2019-2021美团技术年货

CODE A BETTER LIFE

【算法篇】

美团 美团

目录

算法 | 2021 1

美团外卖特征平台的建设与实践 1

美团基于知识图谱的剧本杀标准化建设与应用 24

美团商品知识图谱的构建及应用 40

常识性概念图谱建设以及在美团场景中的应用 60

美团外卖美食知识图谱的迭代及应用 83

美团知识图谱问答技术实践与探索 95

本地生活综合性需求图谱的构建及应用 116

外卖商品的标准化建设与应用 138

多业务建模在美团搜索排序中的实践 151

美团搜索多业务商品排序探索与实践 165

情感分析技术在美团的探索与应用 178

细粒度情感分析在到餐场景中的应用 198

预训练技术在美团到店搜索广告中的应用 217

美团智能客服核心技术与实践 235

广告深度预估技术在美团到店场景下的突破与畅想 254

小样本学习及其在美团场景中的应用 274

美团外卖广告智能算力的探索与实践 300

外卖套餐搭配的探索和应用 317

自动驾驶轨迹预测算法：NeurIPS 挑战赛冠军方案 332

KDD 2021 | 美团联合多高校提出多任务学习模型，已应用于联名卡获客场景 341

ACL 2021 | 一文详解美团技术团队 7 篇精选论文 353

ACL 2021 | 美团提出基于对比学习的文本表示模型，效果相比 BERT-flow 提升 8% 368
CVPR 2021 | 基于 Transformer 的端到端视频实例分割方法 382
ICLR 2021 | 美团 AutoML 论文：鲁棒的神经网络架构搜索 DARTS- 396
SIGIR 2021 | 广告系统位置偏差的 CTR 模型优化方案 409

算法 | 2020 424

智能搜索模型预估框架 Augur 的建设与实践 424
Transformer 在美团搜索排序中的实践 446
BERT 在美团搜索核心排序的探索和实践 459
美团智能配送系统的运筹优化实战 483
一站式机器学习平台建设实践 500
美团搜索中 NER 技术的探索与实践 515
KDD Cup 2020 Debiasing 比赛冠军技术方案及在美团的实践 536
ICRA 2020 轨迹预测竞赛冠军的方法总结 555
KDD Cup 2020 AutoGraph 比赛冠军技术方案及在美团的实践 564
KDD Cup 2020 多模态召回比赛亚军方案与搜索业务应用 584
CIKM 2020 | 一文详解美团 6 篇精选论文 602
MT-BERT 在文本检索任务中的实践 615
美团无人车引擎在仿真中的实践 627
美团无人配送 CVPR2020 论文 CenterMask 解读 638
WSDM Cup 2020 检索排序评测任务第一名经验总结 648
美团内部讲座 | 清华大学莫一林：信息物理系统中的安全控制算法 658
KDD Cup 2020 多模态召回比赛季军方案与搜索业务应用 675
对话任务中的“语言 - 视觉”信息融合研究 690
ICDM 论文：探索跨会话信息感知的推荐模型 701
自然场景人脸检测技术实践 712
技术解析 | 横纵一体的无人车控制方案 727

算法 | 2019 738
美团 BERT 的探索和实践 739
深度学习在搜索业务中的探索与实践 762
大众点评搜索基于知识图谱的深度学习排序实践 794
大众点评信息流基于文本生成的创意优化实践 817
AI Challenger 2018：细粒度用户评论情感分析冠军思路总结 841
WSDM Cup 2019 自然语言推理任务获奖解题思路 850
深度学习在美团配送 ETA 预估中的探索与实践 862
配送交付时间轻量级预估实践 875
ICDAR 2019 论文：自然场景文字定位技术详解 891
CVPR 2019 轨迹预测竞赛冠军方法总结 903
顶会论文：基于神经网络 StarNet 的行人轨迹交互预测算法 910

算法

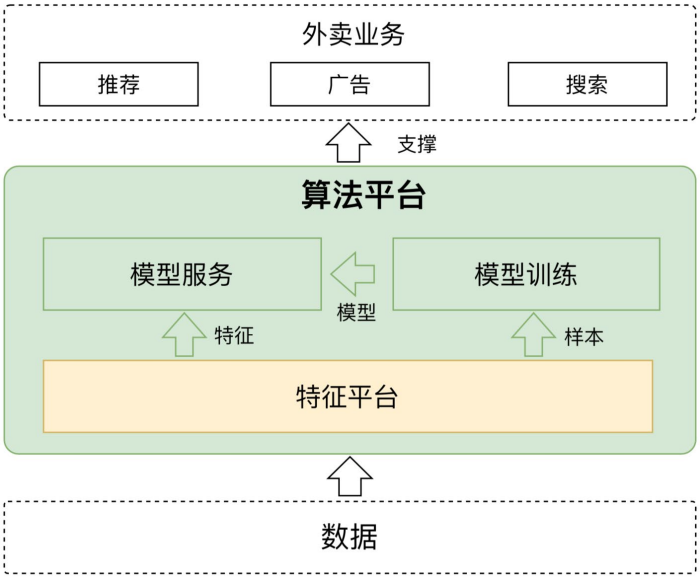
美团外卖特征平台的建设与实践

作者：英亮　陈龙　刘磊　亚劼　乐彬

1. 背景

美团外卖业务种类繁多、场景丰富，根据业务特点可分为推荐、广告、搜索三大业务线以及数个子业务线，比如商家推荐、菜品推荐、列表广告、外卖搜索等等，满足了数亿用户对外卖服务的全方面需求。而在每条业务线的背后，都涉及用户、商家、平台三方面利益的平衡：用户需要精准的展现结果；商家需要尽可能多的曝光和转化；平台需要营收的最大化，而算法策略通过模型机制的优化迭代，合理地维护这三方面的利益平衡，促进生态良性发展。

随着业务的发展，外卖算法模型也在不断演进迭代中。从之前简单的线性模型、树模型，到现在复杂的深度学习模型，预估效果也变得愈发精准。这一切除了受益于模型参数的不断调优，也受益于外卖算法平台对算力增长的工程化支撑。外卖算法平台通过统一算法工程框架，解决了模型 & 特征迭代的系统性问题，极大地提升了外卖算法的迭代效率。根据功能不同，外卖算法平台可划分为三部分：模型服务、模型训练和特征平台。其中，模型服务用于提供在线模型预估，模型训练用于提供模型的训练产出，特征平台则提供特征和样本的数据支撑。本文将重点阐述外卖特征平台在建设过程中遇到的挑战以及优化思路。



诚然，业界对特征系统的研究较为广泛，比如微信 FeatureKV 存储系统聚焦于解决特征数据快速同步问题，腾讯广告特征工程聚焦于解决机器学习平台中 Pre-Trainer 方面的问题，美团酒旅在线特征系统聚焦于解决高并发情形下的特征存取和生产调度问题，而外卖特征平台则聚焦于提供从样本生成 -> 特征生产 -> 特征计算的一站式链路，用于解决特征的快速迭代问题。

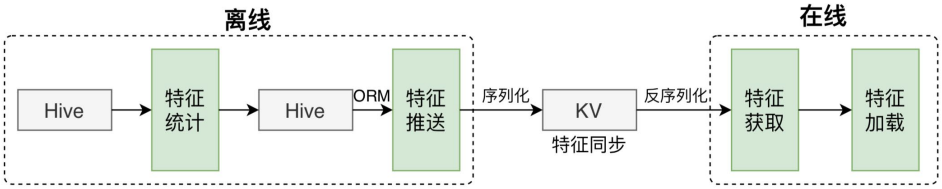
随着外卖业务的发展，特征体量也在快速增长，外卖平台面对的挑战和压力也不断增大。目前，平台已接入特征配置近万个，特征维度近 50 种，日处理特征数据量几十 TB，日处理特征千亿量级，日调度任务数量达数百个。面对海量的数据资源，平台如何做到特征的快速迭代、特征的高效计算以及样本的配置化生成？下文将分享美团外卖在平台建设过程中的一些思考和优化思路，希望能对大家有所帮助或启发。

2. 特征框架演进

2.1 旧框架的不足

外卖业务发展初期，为了提升策略迭代效率，算法同学通过积累和提炼，整理出一套通用的特征生产框架，该框架由三部分组成：特征统计、特征推送和特征获取加载。

如下图所示：



- **特征统计**：基于基础数据表，框架支持统计多个时段内特定维度的总量、分布等统计类特征。
- **特征推送：**框架支持将 Hive 表里的记录映射成 Domain 对象，并将序列化后的结果写入 KV 存储。
- **特征获取加载：**框架支持在线从 KV 存储读取 Domain 对象，并将反序列化后的结果供模型预估使用。

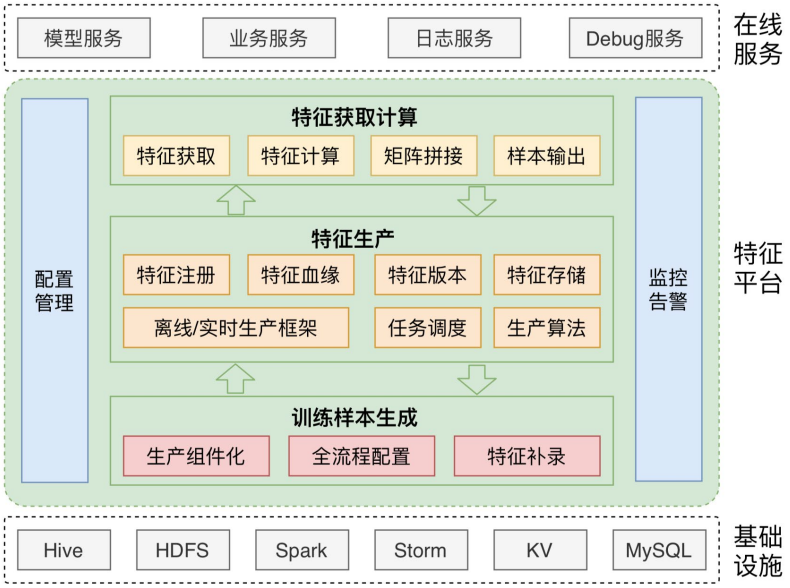
该框架应用在外卖多条业务线中，为算法策略的迭代提供了有力支撑。但随着外卖业务的发展，业务线的增多，数据体量的增大，该框架逐渐暴露以下三点不足：

- **特征迭代成本高：**框架缺乏配置化管理，新特征上线需要同时改动离线侧和在线侧代码，迭代周期较长。
- **特征复用困难：**外卖不同业务线间存在相似场景，使特征的复用成为可能，但框架缺乏对复用能力的很好支撑，导致资源浪费、特征价值无法充分发挥。
- **平台化能力缺失**：框架提供了特征读写的底层开发能力，但缺乏对特征迭代完整周期的平台化追踪和管理能力。

2.2 新平台的优势

针对旧框架的不足，我们在 2018 年中旬开始着手搭建新版的特征平台，经过不断的摸索、实践和优化，平台功能逐渐完备，使特征迭代能力更上一层台阶。

特征平台框架由三部分组成：训练样本生成（离线）、特征生产（近线）以及特征获取计算（在线），如下图所示：



- **训练样本生成**：离线侧，平台提供统一配置化的训练样本生成能力，为模型的效果验证提供数据支撑。
- **特征生产**：近线侧，平台提供面对海量特征数据的加工、调度、存储、同步能力，保证特征数据在线快速生效。
- **特征获取计算**：在线侧，平台提供高可用的特征获取能力和高性能的特征计算能力，灵活支撑多种复杂模型的特征需求。

目前，外卖特征平台已接入外卖多条业务线，涵盖数十个场景，为业务的策略迭代提供平台化支持。其中，平台的优势在于两点：

- **业务提效**：通过特征配置化管理能力、特征 & 算子 & 解决方案复用能力以及离线在线打通能力，提升了特征迭代效率，降低了业务的接入成本，助力业务快速拿到结果。
- **业务赋能**：平台以统一的标准建立特征效果评估体系，有助于特征在业务间的借鉴和流通，最大程度发挥出特征的价值。

3. 特征平台建设

3.1 特征生产：海量特征的生产能力

特征同步的方式有多种，业界常见做法是通过开发 MR 任务 /Spark 任务 / 使用同步组件，从多个数据源读取多个字段，并将聚合的结果同步至 KV 存储。这种做法实现简单，但存在以下问题：

- **特征重复拉取**：同一特征被不同任务使用时，会导致特征被重复拉取，造成资源浪费。
- **缺乏全局调度**：同步任务间彼此隔离，相互独立，缺乏多任务的全局调度管理机制，无法进行特征复用、增量更新、全局限流等操作，影响特征的同步速度。
- **存储方式不够灵活健壮**：新特征存储时，涉及到上下游代码 / 文件的改动，迭代成本高，特征数据异常时，需长时间重导旧数据，回滚效率较低。

围绕上述几点问题，本文将从三个方面进行特征生产核心机制的介绍：

- **特征语义机制**：用于解决平台从数百个数据源进行特征拉取和转化的效率问题。
- **特征多任务调度机制**：用于解决海量特征数据的快速同步问题。
- **特征存储机制**：用于解决特征存储在配置化和可靠性方面的问题。

3.1.1 特征语义

特征平台目前已接入上游 Hive 表数百个、特征配置近万个，其中大部分特征都需天级别的更新。那平台如何从上游高效地拉取特征呢？直观想法是从特征配置和上游 Hive 表两个角度进行考虑：

特征配置角度：平台根据每个特征配置，单独启动任务进行特征拉取。

- 优点：控制灵活。
- 缺点：每个特征都会启动各自的拉取任务，执行效率低且耗费资源。

上游 Hive 表角度：Hive 表中多个特征字段，统一放至同一任务中拉取。

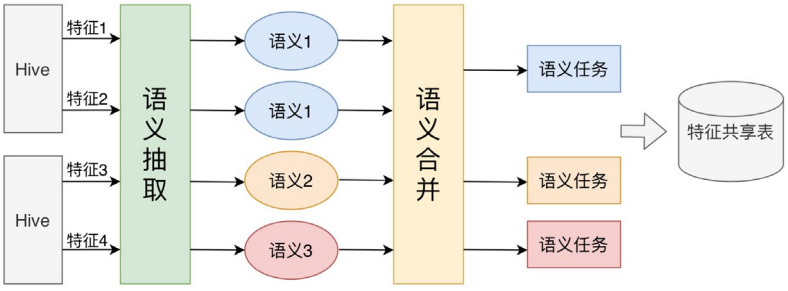
  - 优点：任务数量可控，资源占用低。

- 缺点：任务逻辑耦合较重，新增特征时需感知 Hive 表其它字段拉取逻辑，导致接入成本高。

上述两种方案都存在各自问题，不能很好满足业务需求。因此，特征平台结合两个方案的优点，并经过探索分析，提出了特征语义的概念：

- **特征语义**：由特征配置中的上游 Hive 表、特征维度、特征过滤条件、特征聚合条件四个字段提取合并而成，本质就是相同的查询条件，比如：Select **KeyInHive**,f1,f2 From **HiveSrc** Where **Condition** Group by **Group**，此时该四个字段配置相同，可将 F1、F2 两个特征的获取过程可合并为一个 SQL 语句进行查询，从而减少整体查询次数。另外，平台将语义合并过程做成自动化透明化，接入方只需关心新增特征的拉取逻辑，无需感知同表其它字段，从而降低接入成本。

特征平台对特征语义的处理分为两个阶段：语义抽取和语义合并，如下图所示：



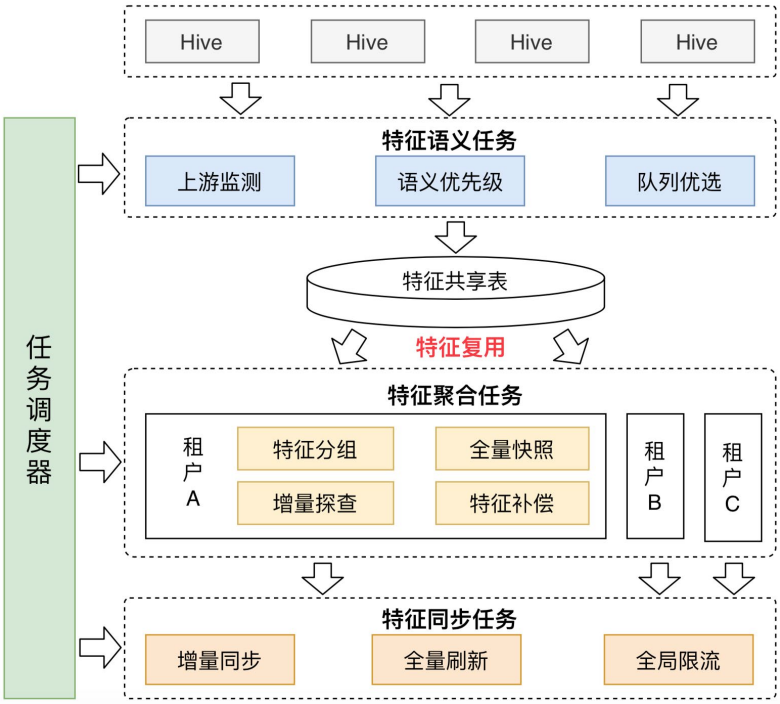
- **语义抽取**：平台解析特征配置，构建 SQL 语法树，通过支持多种形式判同逻辑（比如交换律、等效替换等规则），生成可唯一化表达的 SQL 语句。
- **语义合并**：如果不同特征对应的语义相同，平台会将其抽取过程进行合并，比如：Select KeyInHive, **Extract1 as f1, Extract2 as f2** From HiveSrc Where Condition Group by Group，其中 Extract 即特征的抽取逻辑，f1 和 f2 的抽取逻辑可进行合并，并将最终抽取到的特征数据落地至特征共享表中存储，供多业务方使用。

3.1.2 特征多任务调度

为了保证每天数十 TB 数据量的快速同步，特征平台首先按照特征的处理流程：获取、聚合和同步，分别制定了特征语义任务、特征聚合任务和特征同步任务：

- **特征语义任务**：用于将特征数据从数据源拉取解析，并落地至特征共享表中。
- **特征聚合任务**：用于不同业务线（租户）按照自身需求，从特征共享表中获取特定特征并聚合，生成全量快照以及增量数据。
- **特征同步任务**：用于将增量数据（天级）和全量数据（定期）同步至 KV 存储中。

同时，特征平台搭建了多任务调度机制，将不同类型的任务进行调度串联，以提升特征同步的时效性，如下图所示：



- **任务调度器**：按照任务执行顺序，循环检测上游任务状态，保证任务的有序执行。
- **特征语义任务调度**：当上游 Hive 表就绪后，执行语义任务。
  - 上游监测：通过上游任务调度接口实时获取上游 Hive 表就绪状态，就绪即

拉取，保证特征拉取的时效性。

- 语义优先级：每个语义都会设置优先级，高优先级语义对应的特征会被优先聚合和同步，保证重要特征的及时更新。
- 队列优选：平台会获取多个队列的实时状态，并优先选择可用资源最多的队列执行语义任务，提升任务执行效率。

- **特征复用**：特征的价值在于复用，特征只需接入平台一次，就可在不同业务间流通，是一种业务赋能的体现。
  - 特征统一存储在特征共享表中，供下游不同业务方按需读取，灵活使用。
  - 特征的统一接入复用，避免相同数据的重复计算和存储，节省资源开销。
- **特征聚合任务调度**：当上游语义任务就绪后，执行聚合任务。
  - 多租户机制：多租户是平台面向多业务接入的基础，业务以租户为单位进行特征管理，并为平台分摊计算资源和存储资源。
  - 特征分组：特征分组将相同维度下的多个特征进行聚合，以减少特征 Key 的数量，避免大量 Key 读写对 KV 存储性能造成的影响。
  - 全量快照：平台通过天级别聚合的方式生成特征全量快照，一方面便于增量数据探查，另一方面也避免历史数据的丢失。
  - 增量探查：通过将最新特征数据与全量快照的数值对比，探查出发生变化的特征，便于后续增量同步。
  - 特征补偿：因就绪延迟而未被当天同步的特征，可跨天进行补偿同步，避免出现特征跨天丢失的问题。
- **特征同步任务调度**：当上游聚合任务就绪后，执行同步任务。
  - 增量同步：将经全量快照探查到的增量数据，同步写入 KV 存储，大大降低数据写入量，提升同步效率。
  - 全量刷新：KV 存储中的数据由于过期时间限制，需定期进行全量刷新，避免出现特征过期导致的数据丢失问题。
  - 全局限流：通过监测同步任务的并行度以及 KV 存储状态指标，实时调整全局同步速度，在保证 KV 存储稳定性前提下，充分利用可用资源来提升特征同步效率。

3.1.3 特征存储

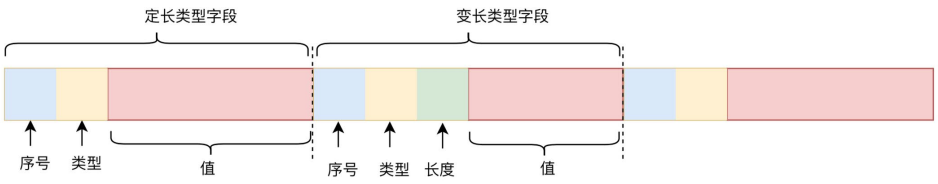
3.1.3.1 特征动态序列化

特征数据通过聚合处理后，需存储到 HDFS/KV 系统中，用于后续任务 / 服务的使用。数据的存储会涉及到存储格式的选型，业界常见的存储格式有 JSON、Object、Protobuf 等，其中 JSON 配置灵活，Object 支持自定义结构，Protobuf 编码性能好且压缩比高。由于特征平台支持的数据类型较为固定，但对序列化反序列化性能以及数据压缩效果有较高要求，因此选择 Protobuf 作为特征存储格式。

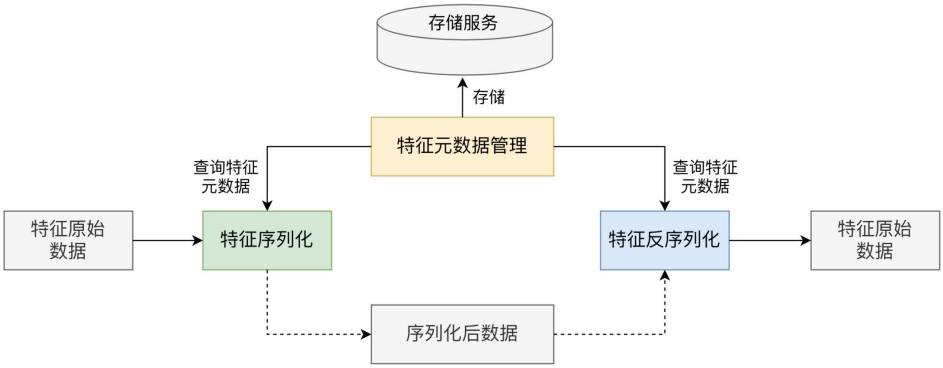
Protobuf 的常规使用方式是通过 Proto 文件维护特征配置。新增特征需编辑 Proto 文件，并编译生成新版本 JAR 包，在离线 & 在线同时发布更新后，才能生产解析新增特征，导致迭代成本较高。Protobuf 也提供了动态自描述和反射机制，帮助生产侧和消费侧动态适配消息格式的变更，避免静态编译带来的 JAR 包升级成本，但代价是空间成本和性能成本均高于静态编译方式，不适用于高性能、低时延的线上场景。

针对该问题，特征平台从特征元数据管理的角度，设计了一种基于 Protobuf 的特征动态序列化机制，在不影响读写性能前提下，做到对新增特征读写的完全配置化。

为方便阐述，先概述下 Protobuf 编码格式。如下图所示，Protobuf 按“键 – 值”形式序列化每个属性，其中键标识了该属性的序号和类型。可以看出，从原理上，序列化主要要依赖键中定义的字段序号和类型。



因此，特征平台通过从元数据管理接口查询元数据，来替换常规的 Proto 文件配置方式，去动态填充和解析键中定义的字段序号和类型，以完成序列化和反序列化，如下图所示：



- **特征序列化**：通过查询特征元数据，获取特征的序号和类型，将特征序号填充至键的序号属性中，并根据特征类型决定键的类型属性以及特征值的填充方式。
- **特征反序列化**：解析键的属性，获取特征序号，通过查询特征元数据，获取对应的特征类型，并根据特征类型决定特征值的解析方式（定长 / 变长）。

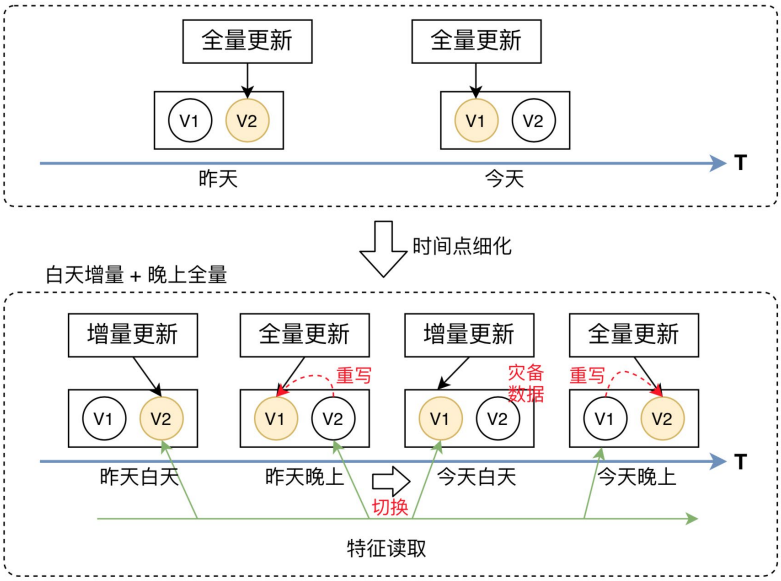
3.1.3.2 特征多版本

特征数据存储于 KV 系统中，为在线服务提供特征的实时查询。业界常见的特征在线存储方式有两种：单一版本存储和多版本存储。

- 单一版本存储即覆盖更新，用新数据直接覆盖旧数据，实现简单，对物理存储占用较少，但在数据异常的时候无法快速回滚。
- 多版本存储相比前者，增加了版本概念，每一份数据都对应特定版本，虽然物理存储占用较多，但在数据异常的时候可通过版本切换的方式快速回滚，保证线上稳定性。

因此，特征平台选择特征多版本作为线上数据存储方式。

传统的多版本方式是通过全量数据的切换实现，即每天在全量数据写入后再进行版本切换。然而，特征平台存在增量和全量两种更新方式，不能简单复用全量的切换方式，需考虑增量和全量的依赖关系。因此，特征平台设计了一种适用于增量 & 全量两种更新方式下的版本切换方式（如下图所示）。该方式以全量数据为基础，白天进行增量更新，版本保持不变，在增量更新结束后，定期进行全量更新（重写），并进行版本切换。



3.2 特征获取计算：高性能的特征获取计算能力

特征获取计算为模型服务、业务系统、离线训练提供特征的实时获取能力和高性能的计算能力，是特征平台能力输出的重要途径。

旧框架中，特征处理分散在业务系统中，与业务逻辑耦合严重，随着模型规模增长和业务系统的架构升级，特征处理性能逐渐成为瓶颈，主要存在以下问题：

- **需要代码开发**：特征处理的代码冗长，一方面会造成易增难改的现象，另一方面相同逻辑代码重复拷贝较多，也会造成复用性逐渐变差，代码质量就会持续恶化。
- **潜在性能风险**：大量实验同时进行，每次处理特征并集，性能会互相拖累。
- **一致性难以保证**：离线训练样本和在线预估对特征的处理逻辑难以统一。

因此，我们在新平台建设中，将特征处理逻辑抽象成独立模块，并对模块的职责边界做了清晰设定：通过提供统一 API 的方式，只负责特征的获取和计算，而不关心业务流程上下文。在新的特征获取和计算模块设计中，我们主要关注以下两个方面：

- **易用性**：特征处理配置的易用性会影响到使用方的迭代效率，如果新增特征或

更改特征计算逻辑需要代码改动，势必会拖慢迭代效率。

- **性能**：特征处理过程需要实时处理大量特征的拉取和计算逻辑，其效率会直接影响到上游服务的整体性能。

围绕以上两点，本文将从下述两个方面分别介绍特征获取计算部分：

- **模型特征自描述 MFDL**：将特征计算流程配置化，提升特征使用的易用性。
- **特征获取流程**：统一特征获取流程，解决特征获取的性能问题。

3.2.1 模型特征自描述 MFDL

模型特征处理是模型预处理的一部分，业界常用的做法有：

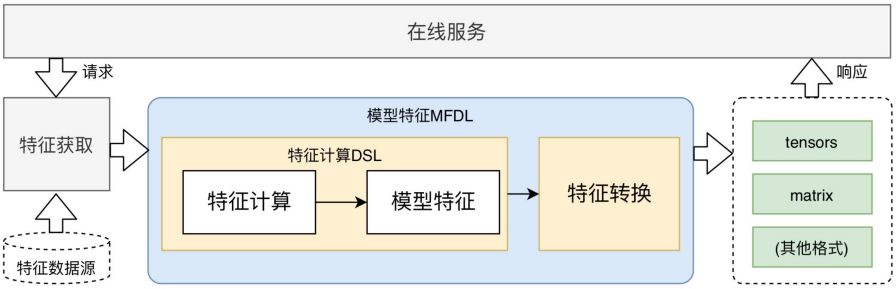
- 将特征处理逻辑和模型打包在一起，使用 PMML 或类似格式描述。优点是配置简洁；缺点是无法单独更新模型文件或特征配置。
- 将特征处理逻辑和模型隔离，特征处理部分使用单独的配置描述，比如 JSON 或 CSV 等格式。优点是特征处理配置和模型文件分离，便于分开迭代；缺点是可能会引起特征配置和模型加载不一致性的问题，增加系统复杂度。

考虑到对存量模型的兼容，我们定义了一套自有的配置格式，能独立于模型文件之外快速配置。基于对原有特征处理逻辑的梳理，我们将特征处理过程抽象成以下两个部分：

- **模型特征计算**：主要用来描述特征的计算过程。这里区分了原始特征和模型特征：将从数据源直接获取到的特征称之为原始特征，将经过计算后输入给模型的特征称之为模型特征，这样就可以实现同一个原始特征经过不同的处理逻辑计算出不同的模型特征。
- **模型特征转换**：将生成的模型特征根据配置转换成可以直接输入给模型的数据格式。由于模型特征计算的结果不能被模型直接使用，还需要经过一些转换逻辑的处理，比如转换成 Tensor、Matrix 等格式。

基于该两点，特征平台设计了 MFDL（Model Feature Description Language）来完整的描述模型特征的生成流程，用配置化的方式描述模型特征计算和转换过程。其

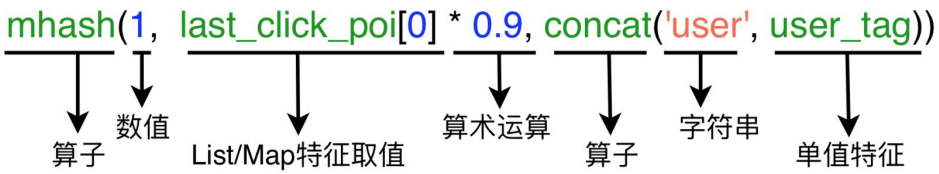
中，特征计算部分通过自定义的 DSL 来描述，而特征转换部分则针对不同类型的模型设计不同的配置项。通过将特征计算和转换分离，就可以很方便的扩展支持不同的机器学习框架或模型结构。



在 MFDL 流程中，特征计算 DSL 是模型处理的重点和难点。一套易用的特征计算规范需既要满足特征处理逻辑的差异性，又要便于使用和理解。经过对算法需求的了解和对业界做法的调研，我们开发了一套易用易读且符合编程习惯的特征表达式，并基于 JavaCC 实现了高性能的执行引擎，支持了以下特性：

- **特征类型**：支持以下常用的特征数据结构：
  - **单值类型（String/Long/Double）**：数值和文本类型特征，如商家好评率。
  - **Map 类型**：交叉或字典类型的特征。
  - **List 类型**：Embedding 或向量特征。
- **逻辑运算**：支持常规的算术和逻辑运算，比如 a>b?(a−b):0。
- **函数算子**：逻辑运算只适合少量简单的处理逻辑，而更多复杂的逻辑通常需要通过函数算子来完成。业务方既可以根据自己的需求编写算子，也可快速复用平台定期收集整理的常用算子，以降低开发成本，提升模型迭代效率。

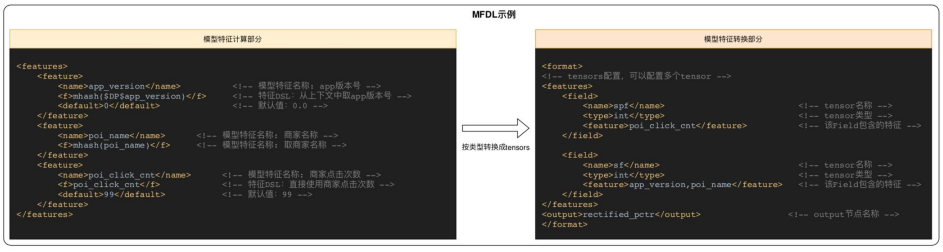
特征计算 DSL 举例如下所示：



基于规范化的 DSL，一方面可以让执行引擎在执行阶段做一些主动优化，包括向量化计算、并行计算等，另一方面也有助于使用方将精力聚焦于特征计算的业务逻辑，而不用关心实现细节，既降低了使用门槛，也避免了误操作对线上稳定性造成的影响。

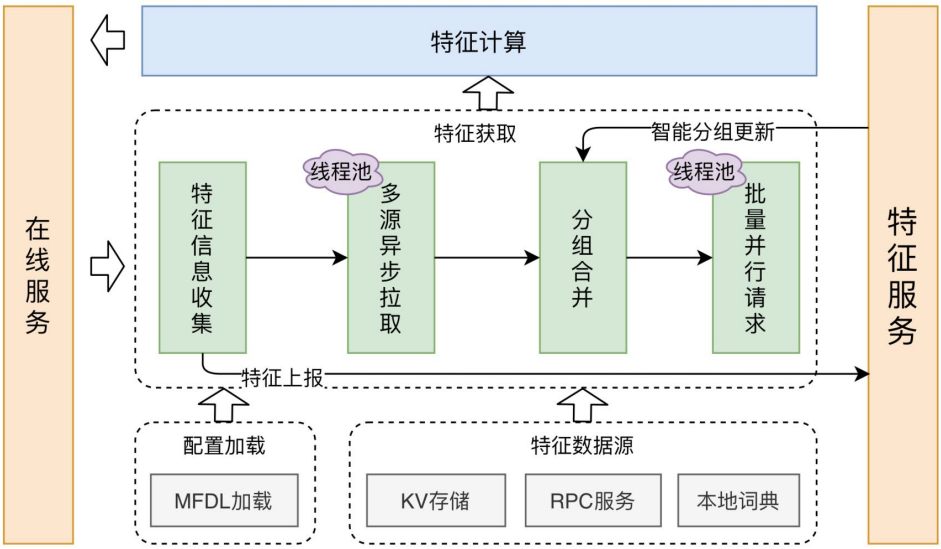
由于 MFDL 是独立于模型文件之外的配置，因此特征更新迭代时只需要将新的配置推送到服务器上，经过加载和预测后即可生效，实现了特征处理的热更新，提升了迭代效率。同时，MFDL 也是离线训练时使用的特征配置文件，结合统一的算子逻辑，保证了离线训练样本 / 在线预估特征处理的一致性。在系统中，只需要在离线训练时配置一次，训练完成后即可一键推送到线上服务，安全高效。

下面是一个 TF 模型的 MFDL 配置示例：



3.2.2 特征获取流程

MFDL 中使用到的特征数据，需在特征计算之前从 KV 存储进行统一获取。为了提升特征获取效率，平台会对多个特征数据源异步并行获取，并针对不同的数据源，使用不同的手段进行优化，比如 RPC 聚合等。特征获取的基本流程如下图所示：



在特征生产章节已经提到，特征数据是按分组进行聚合存储。特征获取在每次访问 KV 存储时，都会读取整个分组下所有的特征数据，一个分组下特征数量的多少将会直接影响到在线特征获取的性能。因此，我们在特征分组分配方面进行了相关优化，既保证了特征获取的高效性，又保证了线上服务的稳定性。

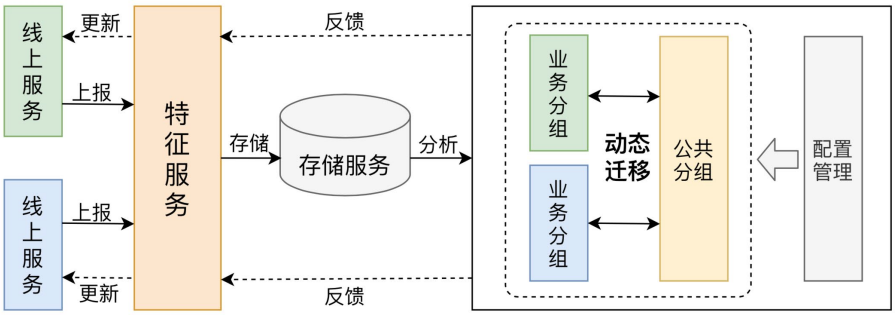
3.2.2.1 智能分组

特征以分组的形式进行聚合，用于特征的写入和读取。起初，特征是以固定分组的形式进行组织管理，即不同业务线的特征会被人工聚合到同一分组中，这种方式实现简单，但却暴露出以下两点问题：

- **特征读取性能差**：线上需要读取解析多个业务线聚合后的特征大 Value，而每个业务线只会用到其中部分特征，导致计算资源浪费、读取性能变差。
- **影响 KV 集群稳定性**：特征大 Value 被高频读取，一方面会将集群的网卡带宽打满，另一方面大 Value 不会被读取至内存，只能磁盘查找，影响集群查询性能（特定 KV 存储场景）。

因此，特征平台设计了智能分组，打破之前固定分组的形式，通过合理机制进行特征分组的动态调整，保证特征聚合的合理性和有效性。如下图所示，平台打通了线上线

下链路，线上用于上报业务线所用的特征状态，线下则通过收集分析线上特征，从全局视角对特征所属分组进行智能化的整合、迁移、反馈和管理。同时，基于存储和性能的折中考虑，平台建立了两种分组类型：业务分组和公共分组：



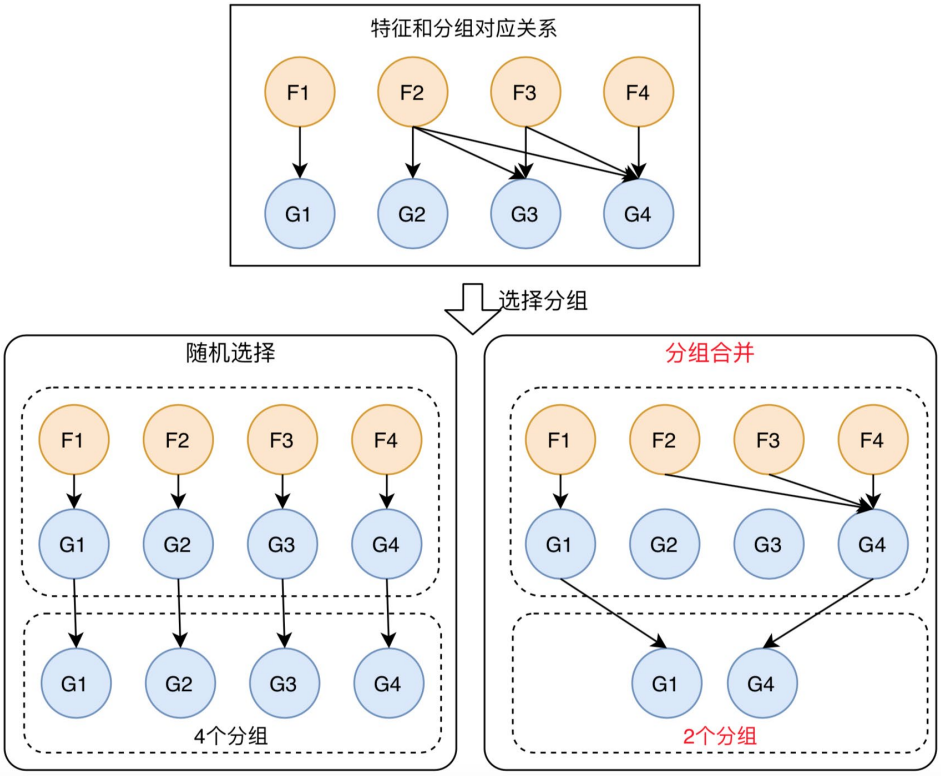
- **业务分组**：用于聚合每个业务线各自用到的专属特征，保证特征获取的有效性。如果特征被多业务共用，若仍存储在各自业务分组，会导致存储资源浪费，需迁移至公共分组（存储角度）。
- **公共分组**：用于聚合多业务线同时用到的特征，节省存储资源开销，但分组增多会带来 KV 存储读写量增大，因此公共分组数量需控制在合理范围内（性能角度）。

通过特征在两种分组间的动态迁移以及对线上的实时反馈，保证各业务对特征所拉即所用，提升特征读取性能，保证 KV 集群稳定性。

3.2.2.2 分组合并

智能分组可以有效的提升特征获取效率，但同时也引入了一个问题：在智能分组过程中，特征在分组迁移阶段，会出现一个特征同时存在于多个分组的情况，造成特征在多个分组重复获取的问题，增加对 KV 存储的访问压力。为了优化特征获取效率，在特征获取之前需要对特征分组进行合并，将特征尽量放在同一个分组中进行获取，从而减少访问 KV 存储的次数，提升特征获取性能。

如下图所示，经过分组合并，将特征获取的分组个数由 4 个（最坏情况）减少到 2 个，从而对 KV 存储访问量降低一半。



3.3 训练样本构建：统一配置化的一致性训练样本生成能力

3.3.1 现状分析

训练样本是特征工程连接算法模型的一个关键环节，训练样本构建的本质是一个数据加工过程，而这份数据如何做到“能用”（数据质量要准确可信）、“易用”（生产过程要灵活高效）、“好用”（通过平台能力为业务赋能）对于算法模型迭代的效率和效果至关重要。

在特征平台统一建设之前，外卖策略团队在训练样本构建流程上主要遇到几个问题：

- **重复性开发**：缺少体系化的平台系统，依赖一些简单工具或定制化开发 Hive/Spark 任务，与业务耦合性较高，在流程复用、运维成本、性能调优等方面都表现较差。
- **灵活性不足**：样本构建流程复杂，包括但不限数据预处理、特征抽取、特征样

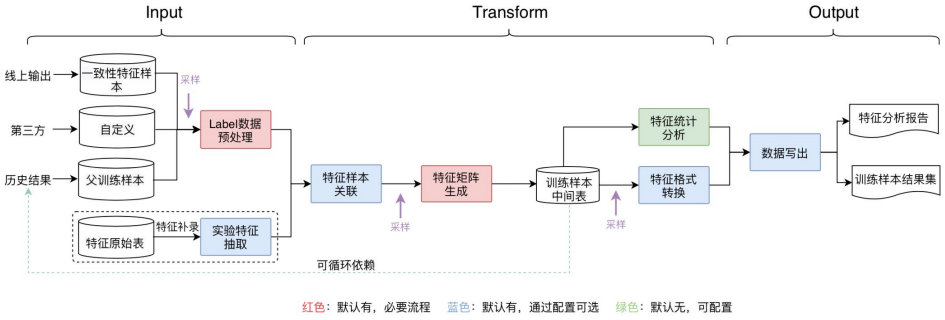
本拼接、特征验证，以及数据格式转换（如 TFRecord）等，已有工具在配置化、扩展性上很难满足需求，使用成本较高。

- **一致性较差**：线上、线下在配置文件、算子上使用不统一，导致在线预测样本与离线训练样本的特征值不一致，模型训练正向效果难保障。

3.3.2 配置化流程

平台化建设最重要的流程之一是“如何进行流程抽象”，业界有一些机器学习平台的做法是平台提供较细粒度的组件，让用户自行选择组件、配置依赖关系，最终生成一张样本构建的 DAG 图。

对于用户而言，这样看似是提高了流程编排的自由度，但深入了解算法同学实际工作场景后发现，算法模型迭代过程中，大部分的样本生产流程都比较固定，反而让用户每次都去找组件、配组件属性、指定关系依赖这样的操作，会给算法同学带来额外的负担，所以我们尝试了一种新的思路来优化这个问题：**模板化 + 配置化**，即平台提供一个基准的模板流程，该流程中的每一个节点都抽象为一个或一类组件，用户基于该模板，通过简单配置即可生成自己样本构建流程，如下图所示：



整个流程模板包括三个部分：**输入（Input）、转化（Transform）、输出（Output）**，其中包含的组件有：Label 数据预处理、实验特征抽取、特征样本关联、特征矩阵生成、特征格式转换、特征统计分析、数据写出，组件主要功能：

- **Label 数据预处理**：支持通过自定义 Hive/Spark SQL 方式抽取 Label 数据，平台也内置了一些 UDF（如 URL Decode、MD5/Murmur Hash 等），通过

自定义 SQL+UDF 方式灵活满足各种数据预处理的需求。在数据源方面，支持如下类型：

- 一致性特征样本：指线上模型预测时，会将一次预测请求中使用到的特征及 Label 相关字段收集、加工、拼接，为离线训练提供基础的样本数据，推荐使用，可更好保障一致性。
- 自定义：不使用算法平台提供的一致性特征样本数据源，通过自定义方式抽取 Label 数据。
- 父训练样本：可依赖之前或其他同学生产的训练样本结果，只需要简单修改特征或采样等配置，即可实现对原数据微调，快速生成新的训练数据，提高执行效率。

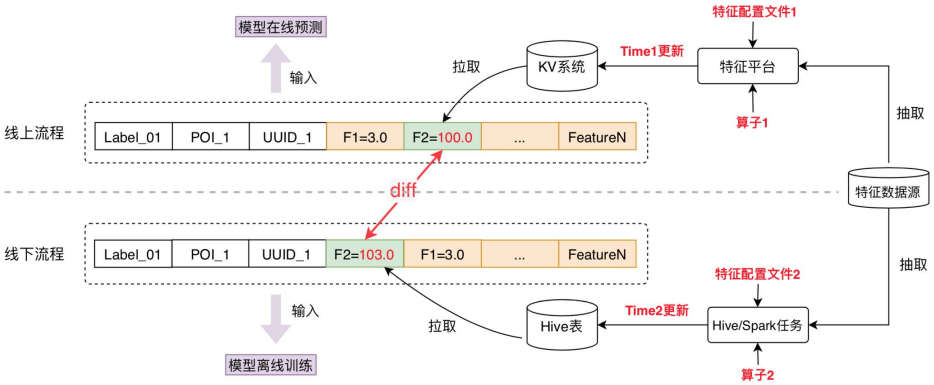
- **实验特征抽取**：线下训练如果需要调研一些新特征（即在一致性特征样本中不存在）效果，可以通过特征补录方式加入新的特征集。
- **特征样本关联**：将 Label 数据与补录的实验特征根据唯一标识（如：poi_id）进行关联。
- **特征矩阵生成**：根据用户定义的特征 MFDL 配置文件，将每一个样本需要的特征集计算合并，生成特征矩阵，得到训练样本中间表。
- **特征格式转换**：基于训练样本中间表，根据不同模型类型，将数据转换为不同格式的文件（如：CSV/TFRecord）。
- **特征统计分析**：辅助功能，基于训练样本中间表，对特征统计分析，包括均值、方差、最大 / 最小值、分位数、空值率等多种统计维度，输出统计分析报告。
- **数据写出**：将不同中间结果，写出到 Hive 表 /HDFS 等存储介质。

上面提到，整个流程是模板化，模板中的多数环节都可以通过配置选择开启或关闭，所以整个流程也支持从中间的某个环节开始执行，灵活满足各类数据生成需求。

3.3.3 一致性保障

（1）为什么会不一致?

上文还提到了一个关键的问题：一致性较差。先来看下为什么会不一致?

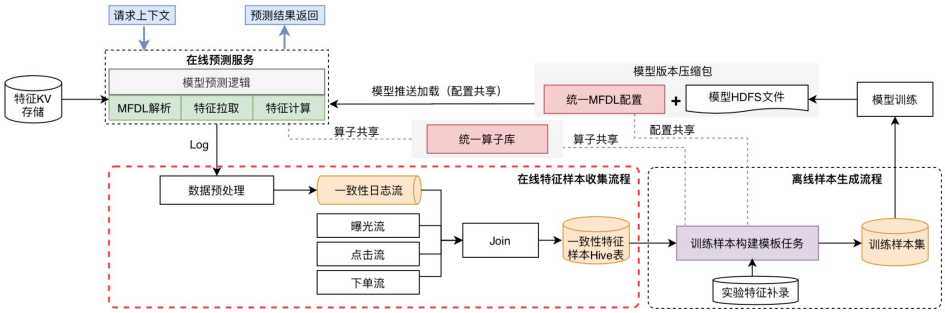


上图展示了在离线训练和在线预测两条链路中构建样本的方式，最终导致离线、在线特征值 Diff 的原因主要有三点：

- **特征配置文件不一致**：在线侧、离线侧对特征计算、编排等配置描述未统一，靠人工较难保障一致性。
- **特征更新时机不一致**：特征一般是覆盖更新，特征抽取、计算、同步等流程较长，由于数据源更新、重刷、特征计算任务失败等诸多不确定因素，在线、离线在不同的更新时机下，数据口径不一致。
- **特征算子定义不一致**：从数据源抽取出来的原始特征一般都需要经过二次运算，线上、线下算子不统一。

（2）如何保证一致性?

明确了问题所在，我们通过如下方案来解决一致性问题：



- **打通线上线下配置**

线下生成训练样本时，用户先定义特征 MFDL 配置文件，在模型训练后，通过平台一键打包功能，将 MFDL 配置文件以及训练输出的模型文件，打包、上传到模型管理平台，通过一定的版本管理及加载策略，将模型动态加载到线上服务，从而实现线上、线下配置一体化。

- **提供一致性特征样本**

通过实时收集在线 Serving 输出的特征快照，经过一定的规则处理，将结果数据输出到 Hive 表，作为离线训练样本的基础数据源，提供一致性特征样本，保障在线、离线数据口径一致。

- **统一特征算子库**

上文提到可以通过特征补录方式添加新的实验特征，补录特征如果涉及到算子二次加工，平台既提供基础的算子库，也支持自定义算子，通过算子库共用保持线上、线下计算口径一致。

3.3.4 为业务赋能

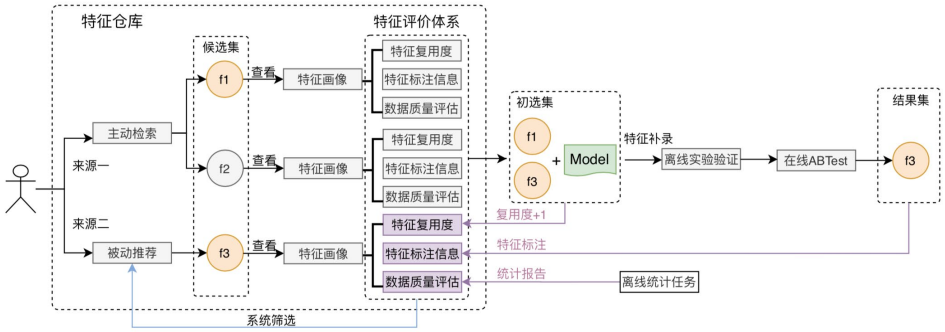
从特征生产，到特征获取计算，再到生成训练样本，特征平台的能力不断得到延展，逐步和离线训练流程、在线预测服务形成一个紧密协作的整体。在特征平台的能力边界上，我们也在不断的思考和探索，希望能除了为业务提供稳定、可靠、易用的特征数据之外，还能从特征的视角出发，更好的建设特征生命周期闭环，通过平台化的能力反哺业务，为业务赋能。在上文特征生产章节，提到了特征平台一个重要能力：特征复用，这也是特征平台为业务赋能最主要的一点。

特征复用需要解决两个问题：

- **特征快速发现**：当前特征平台有上万特征，需要通过平台化的能力，让高质量的特征快速被用户发现，另外，特征的“高质量”如何度量，也需要有统一的评价标准来支撑。

- **特征快速使用**：对于用户发现并筛选出的目标特征，平台需要能够以较低的配置成本、计算资源快速支持使用（参考上文 3.1.2 小节“特征复用”）。

本小节重点介绍如何帮助用户快速发现特征，主要包括两个方面：主动检索和被动推荐，如下图所示：



- 首先，用户可以通过主动检索，从特征仓库筛选出目标特征候选集，然后结合特征画像来进一步筛选，得到特征初选集，最后通过离线实验流程、在线 ABTest，结合模型效果，评估筛选出最终的结果集。其中特征画像主要包括以下评价指标：
  - 特征复用度：通过查看该特征在各业务、各模型的引用次数，帮助用户直观判断该特征的价值。
  - 特征标注信息：通过查看该特征在其他业务离线、在线效果的标注信息，帮助用户判断该特征的正负向效果。
  - 数据质量评估：平台通过离线统计任务，按天粒度对特征进行统计分析，包括特征的就绪时间、空值率、均值、方差、最大 / 小值、分位点统计等，生成特征评估报告，帮助用户判断该特征是否可靠。
- 其次，平台根据特征的评价体系，将表现较好的 Top 特征筛选出来，通过排行榜展现、消息推送方式触达用户，帮助用户挖掘高分特征。

为业务赋能是一个长期探索和实践的过程，未来我们还会继续尝试在深度学习场景中，建立每个特征对模型贡献度的评价体系，并通过自动化的方式打通模型在线上、

线下的评估效果，通过智能化的方式挖掘特征价值。

4. 总结与展望

这篇文章分别从特征框架演进、特征生产、特征获取计算以及训练样本生成四个方面介绍了特征平台在建设与实践中的思考和优化思路。经过两年的摸索建设和实践，外卖特征平台已经建立起完善的架构体系、一站式的服务流程，为外卖业务的算法迭代提供了有力支撑。

未来，外卖特征平台将继续推进从离线 -> 近线 -> 在线的全链路优化工作，在计算性能、资源开销、能力扩展、合作共建等方面持续投入人力探索和建设，并在更多更具挑战的业务场景中发挥平台的价值。同时，平台将继续和模型服务和模型训练紧密结合，共建端到端算法闭环，助力外卖业务蓬勃发展。

5. 作者简介

英亮、陈龙、刘磊、亚劼、乐彬等，美团外卖算法平台工程师。

6. 招聘信息

美团外卖广告工程团队长期招聘后台高级工程师 / 技术专家，负责广告多个方向（推荐 / 搜索 / 召回 / 预估 / 创新）的系统研发工作，坐标北京。欢迎感兴趣的同学加入我们。可投简历至：changyingliang@meituan.com（邮件主题请注明：美团外卖广告工程团队）

美团基于知识图谱的剧本杀标准化建设与应用

作者：李翔　陈焕　志华　晓阳　王奇

剧本杀作为爆发式增长的新兴业务，在商家上单、用户选购、供需匹配等方面存在不足，供给标准化能为用户、商家、平台三方创造价值，助力业务增长。本文介绍了美团到店综合业务数据团队从 0 到 1 快速建设剧本杀供给标准化的过程及算法方案。我们将美团到店综合知识图谱（GENE，GEneral NEeds net）覆盖至剧本杀行业，构建剧本杀知识图谱实现供给标准化建设，包括剧本杀供给挖掘、标准剧本库构建、供给与标准剧本关联等环节，并在多个场景进行应用落地，希望给大家带来一些帮助或启发。

一、背景

剧本杀行业近年来呈爆发式增长态势，然而由于剧本杀是新兴行业，平台已有的类目体系和产品形态，越来越难以满足飞速增长的用户和商户需求，主要表现在下面三个方面：

- **平台类目缺失**：平台缺少专门的“剧本杀”类目，中心化流量入口的缺失，导致用户决策路径混乱，难以建立统一的用户认知。
- **用户决策效率低**：剧本杀的核心是剧本，由于缺乏标准的剧本库，也未建立标准剧本和供给的关联关系，导致剧本信息展示和供给管理的规范化程度低，影响了用户对剧本选择决策的效率。
- **商品上架繁琐**：商品信息需要商户人工一一录入，没有可用的标准模板用以信息预填，导致商户在平台上架的剧本比例偏低，上架效率存在较大的提升空间。

为了解决上述痛点，业务需要进行剧本杀的供给标准化建设：首先建立“剧本杀”新类目，并完成相应的供给（包括商户、商品、内容）的类目迁移。以此为基础，以剧本为核心，搭建标准剧本库，并关联剧本杀供给，继而建立剧本维度的信息分发渠道、评价评分和榜单体系，满足用户“以剧本找店”的决策路径。

值得指出的是，供给标准化是简化用户认知、帮助用户决策、促进供需匹配的重要抓手，标准化程度的高低对平台业务规模的大小有着决定性影响。具体到剧本杀行业，供给标准化建设是助力剧本杀业务持续增长的重要基础，而标准剧本库的搭建是剧本杀供给标准化的关键。由于基于规格如「城限」、背景如「古风」、题材如「情感」等剧本属性无法确定具体的剧本，但剧本名称如「舍离」则能起唯一标识的作用。因此，标准剧本库的搭建，首先是标准剧本名称的建设，其次是规格、背景、题材、难度、流派等标准剧本属性的建设。

综上，美团到店综合业务数据团队与业务同行，助力业务进行剧本杀的供给标准化建设。在建设过程中，涉及了剧本名称、剧本属性、类目、商户、商品、内容等多种类型的实体，以及它们之间的多元化关系构建。而知识图谱作为一种揭示实体及实体间关系的语义网络，用以解决该问题显得尤为合适。特别地，我们已经构建了**美团到店综合知识图谱（GENE，GEneral NEeds net）**，因此，我们基于 GENE 的构建经验快速进行剧本杀这一新业务的知识图谱构建，从 0 到 1 实现剧本杀标准化建设，从而改善供给管理和供需匹配，为用户、商户、平台三方创造出更大的价值。

二、解决方案

我们构建的 GENE，围绕本地生活用户的综合性需求，以行业体系、需求对象、具象需求、场景要素和场景需求五个层次逐层递进，覆盖了玩乐、医美、教育、亲子、结婚等多个业务，体系设计和技术细节可见美团到店综合知识图谱相关的文章。剧本杀作为一项新兴的美团到店综合业务，体现了用户在玩乐上的新需求，天然适配 GENE 的体系结构。因此，我们将 GENE 覆盖至剧本杀新业务，沿用相同的思路来进行相应知识图谱的构建，以实现相应的供给标准化。

基于知识图谱来实现剧本杀标准化建设的关键，是以标准剧本为核心构建剧本杀知识图谱。图谱体系设计如图 1 所示，具体地，首先在行业体系层进行剧本杀新类目的构建，挖掘剧本杀供给，并建立供给（包括商户、商品、内容）与类目的从属关系。在此基础上，在需求对象层，进一步实现标准剧本名称这一核心对象节点和其剧本属性

节点的挖掘以及关系构建，建立标准剧本库，最后将标准剧本库的每个标准剧本与供给和用户建立关联关系。此外，具象需求、场景要素、场景需求三层则实现了对用户在剧本杀上的具象的服务需求和场景化需求的显性表达，这部分由于与剧本杀供给标准化建设的联系不多，在这里不做展开介绍。

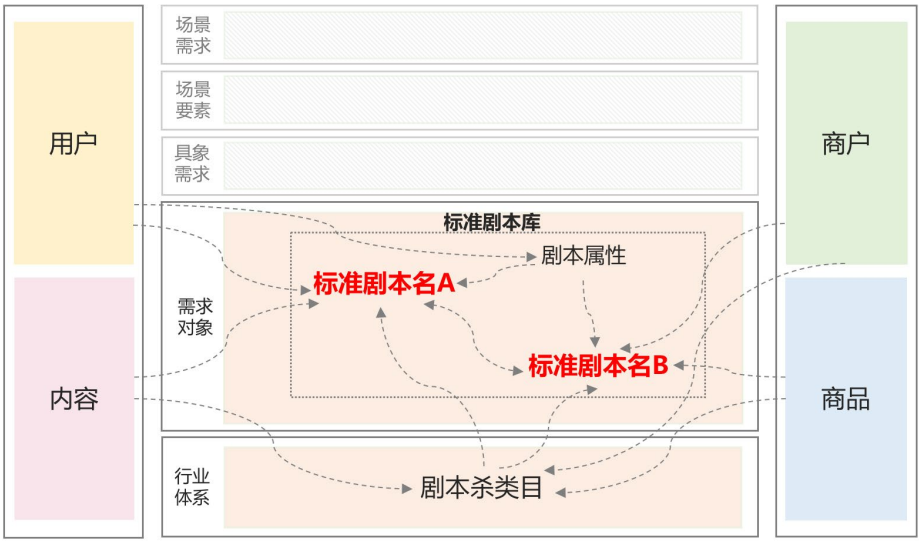


图 1

剧本杀知识图谱中用于供给标准化部分的具体样例如下图 2 所示。其中，标准剧本名称是核心节点，围绕它的各类标准剧本属性节点包括题材、规格、流派、难度、背景、别称等。同时，标准剧本之间可能构建诸如“同系列”等类型的关系，比如「舍离」和「舍离 2」。此外，标准剧本还会与商品、商户、内容、用户之间建立关联关系。

我们基于剧本杀知识图谱的这些节点和关系进行供给标准化，在图谱构建过程中，包括了**剧本杀供给挖掘、标准剧本库构建、供给与标准剧本关联**三个主要步骤，下面对三个步骤的实现细节以及涉及的算法进行介绍。

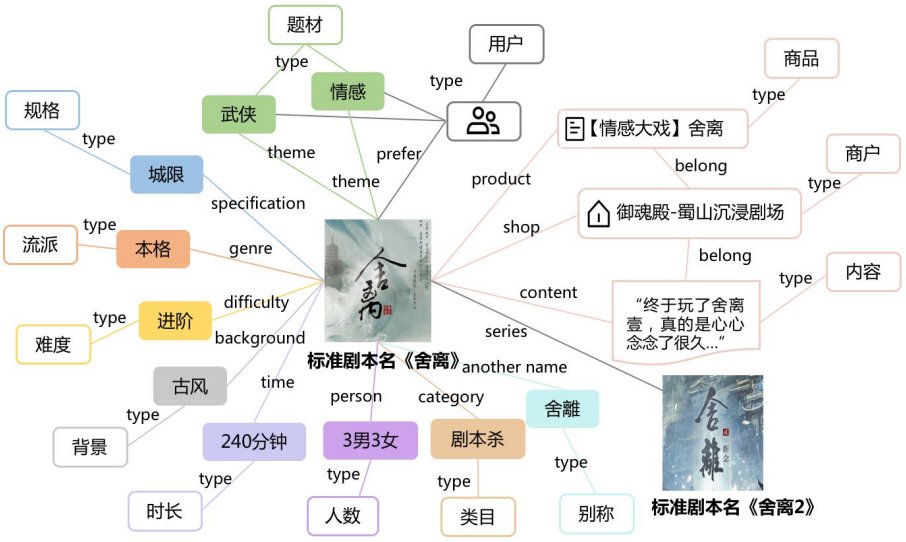


图 2

三、实现方法

3.1 剧本杀供给挖掘

剧本杀作为新兴的业务，已有的行业类目树中并没有相应的类目，无法直接根据类目获取剧本杀的相关供给（包括商户、商品和内容）。因此，我们需要首先进行剧本杀供给的挖掘，即从当前与剧本杀行业相近类目的供给中挖掘出剧本杀的相关供给。

对于剧本杀的商户供给挖掘，需要判断商户是否提供剧本杀服务，判别依据包括了商户名、商品名及商品详情、商户 UGC 三个来源的文本语料。这个本质上是一个多源数据的分类问题，然而由于缺乏标注的训练样本，我们没有直接采用端到端的多源数据分类模型，而是依托业务输入，采用无监督匹配和有监督拟合相结合的方式高效实现，具体的判别流程如下图 3 所示，其中：

- **无监督匹配**：首先构造剧本杀相关的关键词词库，分别在商户名、商品名及商品详情、商户 UGC 三个来源的文本语料中进行精确匹配，并构建基于 BERT[1] 的通用语义漂移判别模型进行匹配结果过滤。最后根据业务规则基于

各来源的匹配结果计算相应的匹配分数。

- **有监督拟合**：为了量化不同来源匹配分数对最终判别结果的影响，由运营先人工标注少量商户分数，用以表征商户提供剧本杀服务的强弱。在此基础上，我们构造了一个线性回归模型，拟合标注的商户分数，获取各来源的权重，从而实现对剧本杀商户的精准挖掘。

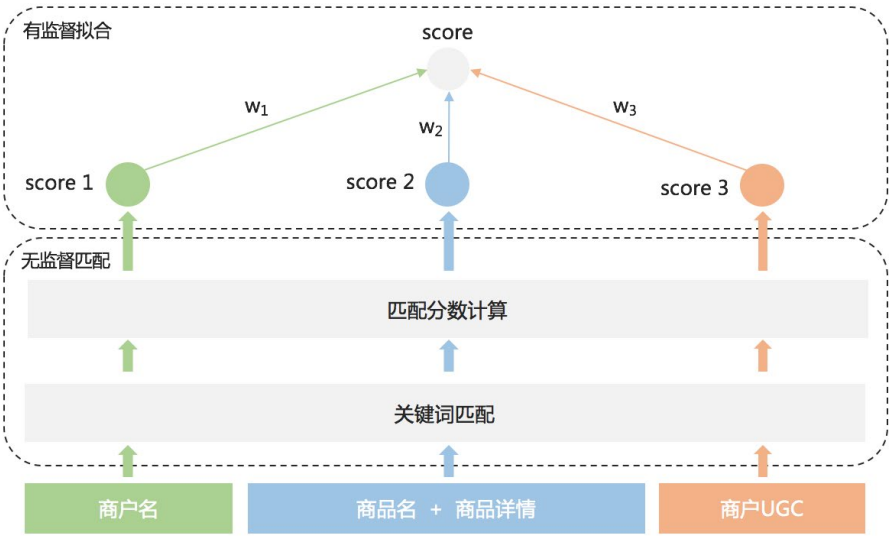


图 3

采用上述方式，实现了桌面和实景两种剧本杀商户的挖掘，准确率和召回率均达到了要求。基于剧本杀商户的挖掘结果，能够进一步对商品进行挖掘，并创建剧本杀类目，从而为后续剧本杀知识图谱构建及标准化建设打好了数据基础。

3.2 标准剧本库构建

标准剧本作为整个剧本杀知识图谱的核心，在剧本杀供给标准化建设中扮演着重要的角色。我们基于剧本杀商品相似聚合的方式，结合人工审核来挖掘标准剧本，并从相关发行方获取剧本授权，从而构建标准剧本库。标准剧本由两部分构成，一个是标准剧本名称，另一个是标准剧本属性。因此，标准剧本库构建也分为标准剧本名称的挖掘和标准剧本属性的挖掘两个部分。

3.2.1 标准剧本名称的挖掘

我们根据剧本杀商品的特点，先后采用了规则聚合、语义聚合和多模态聚合三种方法进行挖掘迭代，从数十万剧本杀商品的名称中聚合得到数千标准剧本名称。下面分别对三种聚合方法进行介绍。

规则聚合

同一个剧本杀商品在不同商户的命名往往不同，存在较多的不规范和个性化。一方面，同一个剧本名称本身就可以有多种叫法，例如「舍离」、「舍离壹」、「舍离 1」就是同一个剧本；另一方面，剧本杀商品名除了包含剧本名称外，商家很多时候也会加入剧本的规格和题材等属性信息以及吸引用户的描述性文字，例如「《舍离》情感本」。所以我们首先考虑剧本杀商品的命名特点，设计相应的清洗策略对剧本杀商品名称进行清洗后再聚合。

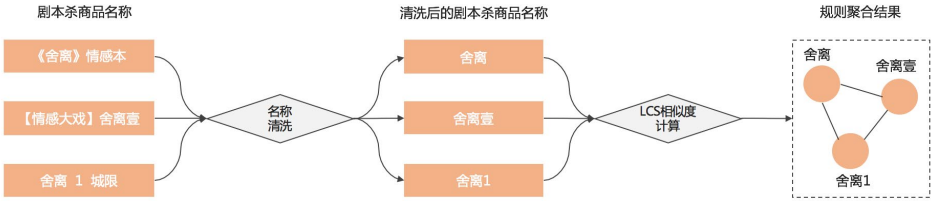


图 4

我们除了梳理常见的非剧本词，构建词库进行规则过滤外，也尝试将其转换为命名实体识别问题 [2]，采用序列标注对字符进行“是剧本名”与“不是剧本名”两个类别的区分。对于清洗后的剧本杀商品名称，则通过基于最长公共子序列（LCS）的相似度计算规则，结合阈值筛选对其进行聚合，例如「舍离」、「舍离壹」、「舍离 1」最后均聚在一起。整个流程如上图 4 所示，采用规则聚合的方式，能够在建设初期帮助业务快速对剧本杀商品名称进行聚合。

语义聚合

规则聚合的方式虽然简单好用，但由于剧本名称的多样性和复杂性，我们发现聚合结果中仍然存在一些问题：1）不属于同一个剧本的商品被聚合，例如「舍离」和「舍离

2」是同一个系列的两个不同剧本，却被聚合在一起。2）属于同一个剧本的商品没有聚合，例如，商品名使用剧本的简称缩写（「唐人街名侦探和猫」和「唐探猫」）或出现错别字（「弗洛伊德之锚」和「佛洛依德之锚」）等情况时则难以规则聚合。

针对这上述这两种问题，我们进一步考虑使用商品名称语义匹配的方式，从文本语义相同的角度来进行聚合。常用的文本语义匹配模型分为交互式和双塔式两种类型。交互式是把两段文本一起输入进编码器，在编码的过程中让其相互交换信息后再进行判别；双塔式模型是用一个编码器分别给两个文本编码出向量，然后基于两个向量进行判别。

由于商品数量众多，采用交互式的方法需要将商品名称两两组合后再进行模型预测，效率较为低下，为此，我们采用双塔式的方法来实现，以 Sentence-BERT[3] 的模型结构为基础，将两个商品名称文本分别通过 BERT 提取向量后，再使用余弦距离来衡量两者的相似度，完整结构如下图 5 所示：

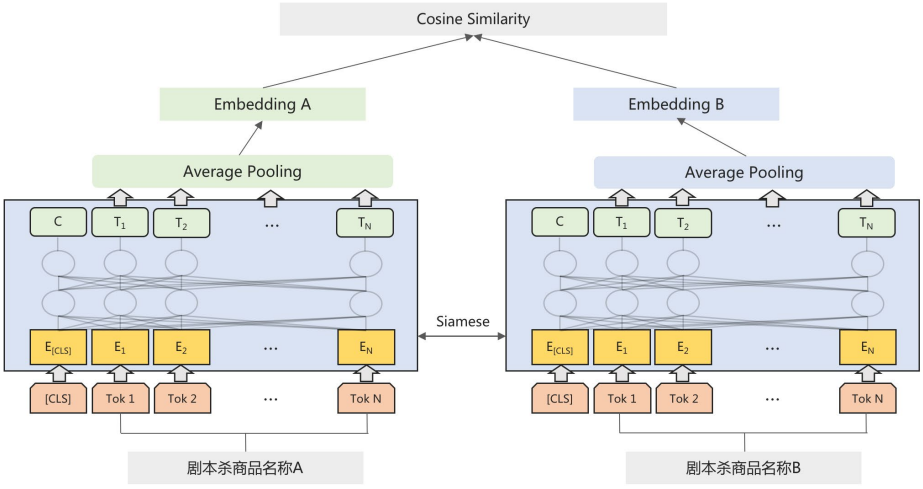


图 5

在训练模型的过程中，我们首先基于规则聚合的结果，通过同聚簇内生成正例和跨聚簇交叉生成负例的方式，构造粗粒度的训练样本，完成初版模型的训练。在此基础上，进一步结合主动学习，对样本数据进行完善。此外，我们还根据上文提到的规则聚合出现的两种问题，针对性的批量生成样本。具体地，通过在商品名称后添加同系

列编号，以及使用错字、别字和繁体字替换等方式来实现样本的自动构造。

多模态聚合

通过语义聚合的方式实现了从商品名称文本语义层面的同义聚合，然而我们通过对聚合结果再分析后发现还存在一些问题：两个商品属于同一个剧本，但仅从商品名称的角度是无法判别。例如，「舍离 2」和「断念」从语义的角度无法聚合，但是它们本质上是一个剧本「舍离 2 · 断念」。虽然这两个商品的名称各异，但是它们的图像往往是相同或相似的，为此，我们考虑引入商品的图像信息来进行辅助聚合。

一个简单的方法是，使用 CV 领域成熟的预训练模型作为图像编码器进行特征提取，直接计算两个商品的图像相似度。为了统一商品图像相似度计算和商品名称语义匹配的结果，我们尝试构建一个剧本杀商品的多模态匹配模型，充分利用商品名称和图像信息来进行匹配。模型沿用语义聚合中使用的双塔式结构，整体结构如下图 6 所示：

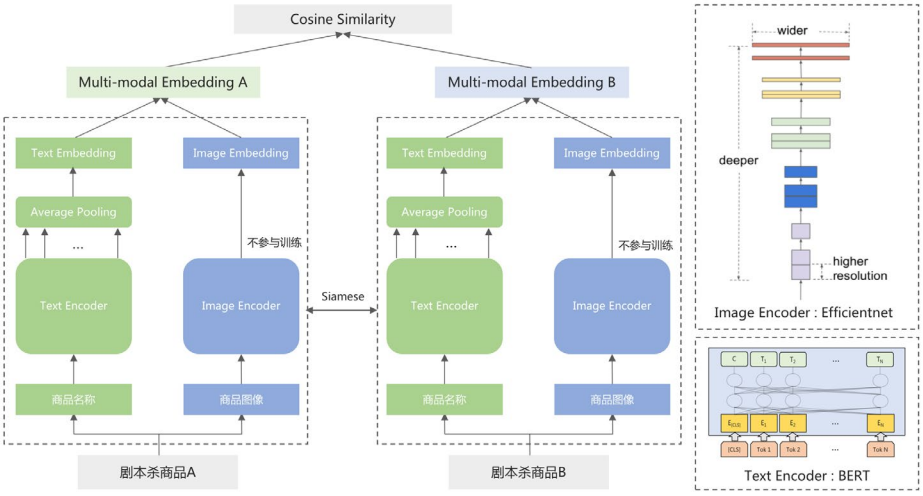


图 6

在多模态匹配模型中，剧本杀商品的名称和图像分别通过文本编码器和图像编码器得到对应的向量表示后，再进行拼接作为最终的商品向量，最后使用余弦相似度来衡量商品之间的相似度。其中：

- **文本编码器**：使用文本预训练模型 BERT[1] 作为文本编码器，将输出平均池化后作为文本的向量表示。
- **图像编码器**：使用图像预训练模型 EfficientNet[4] 作为图像编码器，提取网络最后一层输出作为图像的向量表示。

在训练模型的过程中，文本编码器会进行 Finetune，而图像编码器则固定参数，不参与训练。对于训练样本构建，我们以语义聚合的结果为基础，以商品图像相似度来圈定人工标注样本的范围。具体地，对于同聚簇内商品图像相似度高的直接生成正例，跨聚簇交叉的商品图像相似度低的直接生成负例，而对于剩余的样本对则交由人工进行标注确定。通过多模态聚合，弥补了仅使用文本匹配的不足，与其相比准确率提升了 5%，进一步提升了标准剧本的挖掘效果。

3.2.2 标准剧本属性的挖掘

标准剧本的属性包括了剧本的背景、规格、流派、题材、难度等十余个维度。由于商户在剧本杀商品上单的时候会录入商品的这些属性值，所以对于标准剧本属性的挖掘，本质上是对该标准剧本对应的所有聚合商品的属性的挖掘。

在实际过程中，我们通过投票统计的方式来进行挖掘，即对于标准剧本的某个属性，通过对应的聚合商品在该属性上的属性值进行投票，选择投票最高的属性值，作为该标准剧本的候选属性值，最后由人工审核确认。此外，在标准剧本名称挖掘的过程中，我们发现同一个剧本的叫法多种多样，为了对标准剧本能有更好的描述，还进一步为标准剧本增加了一个别称的属性，通过对标准剧本对应的所有聚合商品的名称进行清洗和去重来获取。

3.3 供给与标准剧本关联

在完成标准剧本库构建后，还需要建立剧本杀的商品、商户和内容三种供给，与标准剧本的关联关系，从而使剧本杀的供给实现标准化。由于通过商品和标准剧本的关联关系，可以直接获取该商品对应商户和标准剧本的关系，所以我们只需要对商品和内容进行标准剧本关联。

3.3.1 商品关联

在 3.2 节中，我们通过聚合存量剧本杀商品的方式来进行标准剧本的挖掘，在这个过程中其实已经构建了存量商品和标准剧本的关联关系。对于后续新增加的商品，我们还需要将其和标准剧本进行匹配，以建立两者之间的关联关系。而对于与标准剧本无法关联的商品，我们则自动进行标准剧本名称和属性的挖掘，经由人工审核后再加入标准剧本库。

整个商品关联流程如下图 7 所示，首先对商品名称进行清洗再进行匹配关联。在匹配环节，我们基于商品和标准剧本的名称及图像的多模态信息，对两者进行匹配判别。

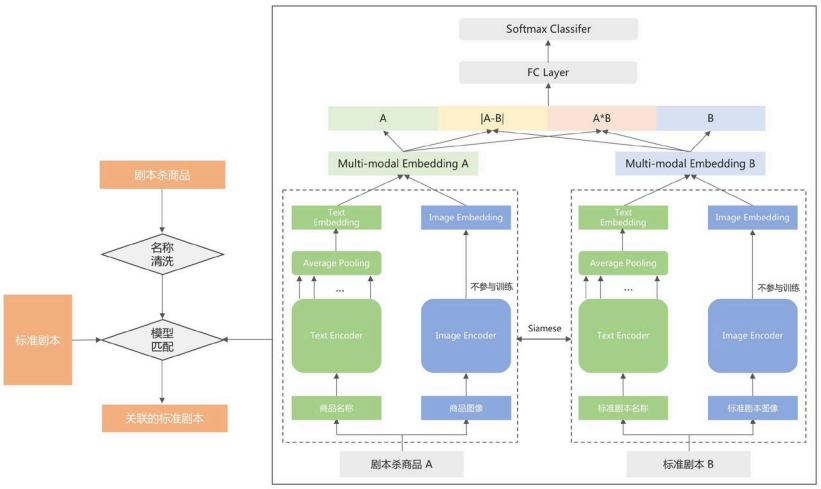


图 7

与商品之间的匹配不同，商品与标准剧本的关联不需要保持匹配的对称性。为了保证关联的效果，我们在 3.2.1 节的多模态匹配模型的结构基础上进行修改，将商品和标准剧本的向量拼接后通过全连接层和 softmax 层计算两者关联的概率。训练样本则直接根据存量商品和标准剧本的关联关系构造。通过商品关联，我们实现了绝大部分剧本杀商品的标准化。

3.3.2 内容关联

对于剧本杀内容关联标准剧本，主要针对用户产生的内容（UGC，例如用户评价）这

一类型的内容和标准剧本的关联。由于一段 UGC 文本通常包含多个句子，且其中只有部分句子会提及标准剧本相关信息，所以我们将 UGC 与标准剧本的匹配，细化为其子句粒度的匹配，同时出于效率和效果的平衡的考虑，进一步将匹配过程分为了召回和排序两个阶段，如下图 8 所示：

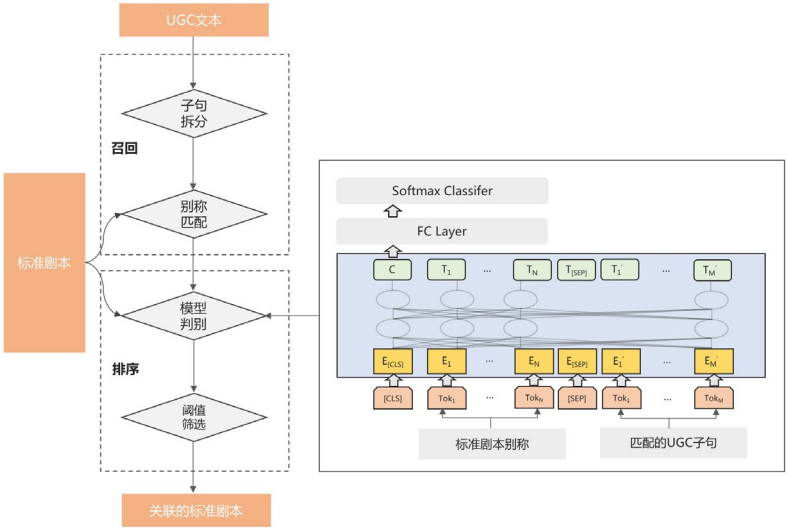


图 8

在召回阶段，将 UGC 文本进行子句拆分，并根据标准剧本名称及其别称，在子句集合中进行精确匹配，对于匹配中的子句则将进入到排序阶段进行精细化的关联关系判别。

在排序阶段，将关联关系判别转换为一个 Aspect-based 的分类问题，参考属性级情感分类的做法 [5]，构建基于 BERT 句间关系分类的匹配模型，将实际命中 UGC 子句的标准剧本别称和对应的 UGC 子句用 [SEP] 相连后输入，通过在 BERT 后增加全连接层和 softmax 层来实现是否关联的二分类，最后对模型输出的分类概率进行阈值筛选，获取 UGC 关联的标准剧本。

与上文中涉及的模型训练不同，UGC 和标准剧本的匹配模型无法快速获取大量训练样本。考虑到训练样本的缺乏，所以首先通过人工少量标注数百个样本，在此基础上，除了采用主动学习外，我们还尝试对比学习，基于 Regularized Dropout[6] 方

法，对模型两次 Dropout 的输出进行正则约束。最终在训练样本不到 1K 的情况下，UGC 关联标准剧本的准确率达到上线要求，每个标准剧本关联的 UGC 数量也得到了大幅提升。

四、应用实践

当前剧本杀知识图谱，以数千标准剧本为核心，关联百万供给。剧本杀供给标准化建设的结果已在美团多个业务场景上进行了初步的应用实践。下面介绍具体的应用方式和应用效果。

4.1 类目构建

通过剧本杀供给挖掘，帮助业务识别出剧本杀商户，从而助力剧本杀新类目和相应剧本杀列表页的构建。剧本杀类目迁移、休闲娱乐频道页的剧本杀入口、剧本杀列表页均已上线，其中，频道页剧本杀 ICON 固定第三行首位，提供了中心化流量入口，有助于建立统一的用户认知。上线示例如图 9 所示（(a) 休闲娱乐频道页剧本杀入口，(b) 剧本杀列表页）。

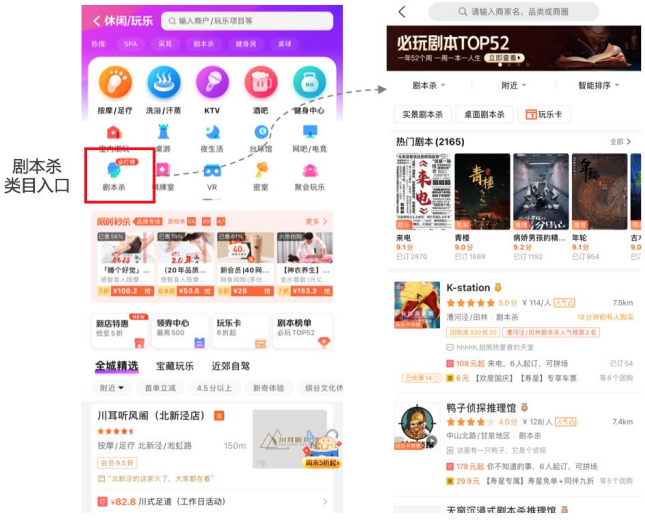


图 9

4.2 个性化推荐

剧本杀知识图谱包含的标准剧本及属性节点，以及其与供给和用户的关联关系，可应用于剧本杀各页面的推荐位。一方面应用于剧本列表页热门剧本推荐（图 10(a)），另一方面还应用于剧本详情页的商品在拼场次推荐（图 10(b) 左）、可玩门店推荐（图 10(b) 左）和相关剧本推荐模块（图 10(b) 右）。这些推荐位的应用，帮助培养了用户在平台找剧本的心智，优化了用户认知和选购体验，提高了用户和供给的匹配效率。

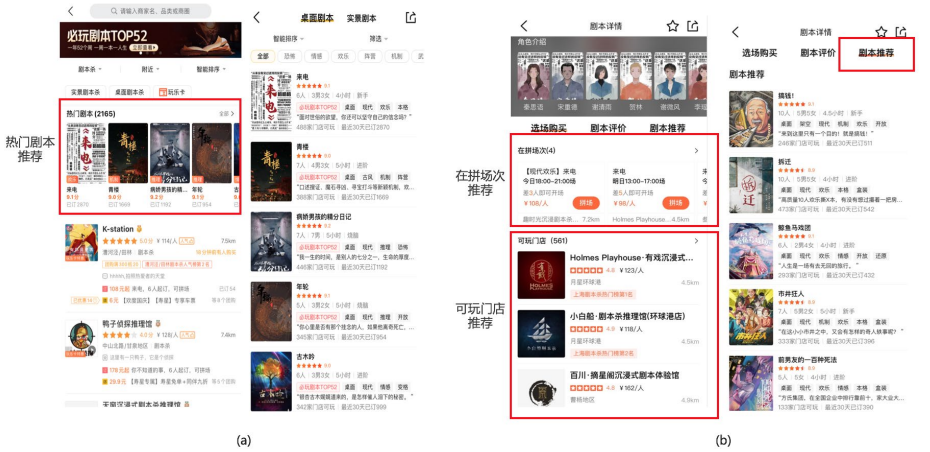


图 10

以剧本列表页的热门剧本推荐模块为例，剧本杀知识图谱包含的节点和关系除了可以直接用于剧本的召回，还可以进一步在精排阶段进行应用。在精排中，我们基于剧本杀知识图谱，结合用户行为，参考 Deep Interest Network（DIN）[7] 模型结构，尝试对用户访问剧本的序列和访问商品的序列进行建模，构建双通道 DIN 模型，深度刻画用户兴趣，实现剧本的个性化分发。其中商品访问序列部分，通过商品与标准剧本的关联关系将其转为为剧本序列，与候选剧本采用 Attention 方式进行建模，具体模型结构如下图 11 所示：

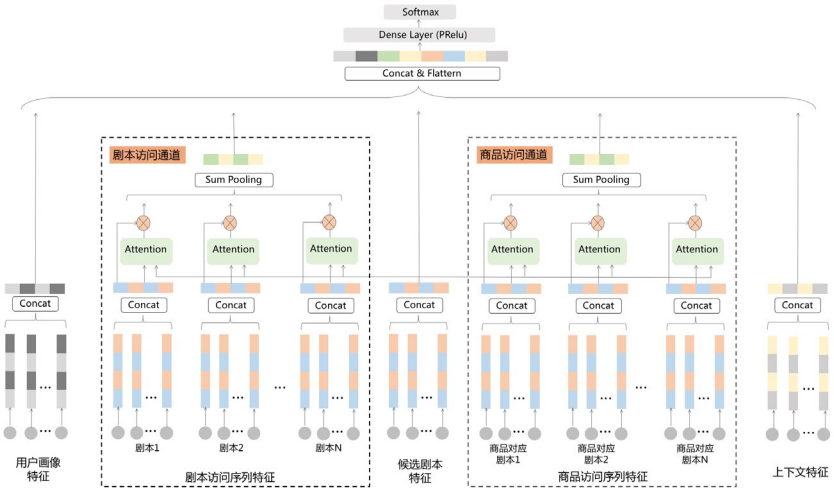


图 11

4.3 信息外露和筛选

基于剧本杀知识图谱中的节点和关系，在剧本杀列表页和在剧本列表页增加相关标签筛选项，并外露剧本的属性和关联的供给信息，相关应用如下图 12 所示。这些标签筛选项和信息的外露，为用户提供了规范的信息展示，降低了用户决策成本，更加方便了用户选店和选剧本。

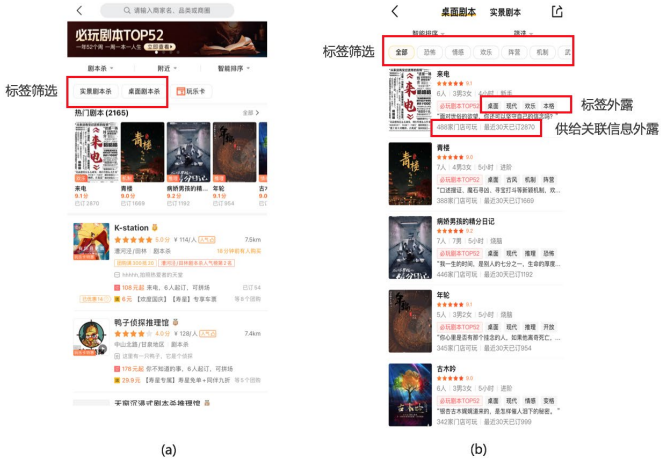


图 12

4.4 评分和榜单

在剧本详情页，内容和标准剧本的关联关系参与到剧本的评分计算中（图 13(a)）。在此基础上，基于剧本维度，形成经典必玩和近期热门的剧本榜单，如图 13(b) 所示，从而为用户的剧本选择决策提供了更多的帮助。

(a) (b)

图 13

五、总结展望

面对剧本杀这一新兴行业，我们快速响应业务，以标准剧本为核心节点，结合行业特点，通过剧本杀供给挖掘、标准剧本库构建、供给与标准剧本关联，构建相应的知识图谱，从 0 到 1 逐步推进剧本杀的供给标准化建设，力求以简单而有效的方法来解决剧本杀业务的问题。

目前剧本杀知识图谱已在剧本杀多个业务场景中取得应用成果，赋能剧本杀业务持续增长，显著提升了用户体验。在未来的工作中，我们将不断进行优化和探索：

- **标准剧本库的持续完善**：优化标准剧本名称和属性以及相应的供给关联关系，保证标准剧本库的质与量俱佳，并尝试引入外部的知识补充当前的标准化结果。
- **剧本杀场景化**：当前剧本杀知识图谱主要以“剧本”这类用户的具象需求对象

为主，后续将深入挖掘用户的场景化需求，探索剧本杀和其他行业的联动，更好的助力剧本杀行业的发展。

- **更多的应用探索**：将图谱数据应用于搜索等模块，在更多的应用场景中提升供给匹配效率，从而创造出更大的价值。

参考文献

[1] Devlin J, Chang M W, Lee K, et al. Bert: Pre-training of deep bidirectional transformers for language understanding[J]. arXiv preprint arXiv:1810.04805, 2018.
[2] Lample G, Ballesteros M, Subramanian S, et al. Neural architectures for named entity recognition[J]. arXiv preprint arXiv:1603.01360, 2016.
[3] Reimers N, Gurevych I. Sentence-bert: Sentence embeddings using siamese bert-networks[J]. arXiv preprint arXiv:1908.10084, 2019.
[4] Tan M, Le Q. EfficientNet: Rethinking model scaling for convolutional neural networks[C]//International Conference on Machine Learning. PMLR, 2019: 6105-6114.
[5] Sun C, Huang L, Qiu X. Utilizing BERT for aspect-based sentiment analysis via constructing auxiliary sentence[J]. arXiv preprint arXiv:1903.09588, 2019.
[6] Liang X, Wu L, Li J, et al. R-Drop: Regularized Dropout for Neural Networks[J]. arXiv preprint arXiv:2106.14448, 2021.
[7] Zhou G, Zhu X, Song C, et al. Deep interest network for click-through rate prediction[C]//Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. 2018: 1059-1068.

作者简介

李翔、陈焕、志华、晓阳、王奇等，均来自美团到店平台技术部到综业务数据团队。

招聘信息

美团到店平台技术部 - 到综业务数据团队，长期招聘算法（自然语言处理 / 推荐算法）、数据仓库、数据科学、系统开发等岗位同学，坐标上海。欢迎感兴趣的同学发送简历至：licong.yu@meituan.com。

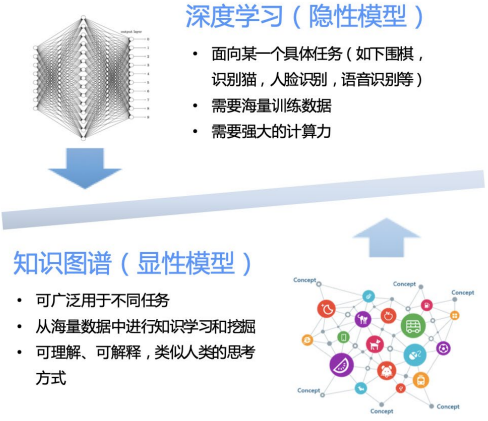
美团商品知识图谱的构建及应用

作者：雪智　凤娇　姿雯　匡俊　林森　武威

背景

美团大脑

近年来，人工智能正在快速地改变人们的生活，背后其实有两大技术驱动力：**深度学习**和**知识图谱**。我们将深度学习归纳为隐性的模型，它通常是面向某一个具体任务，比如说下围棋、识别猫、人脸识别、语音识别等等。通常而言，在很多任务上它能够取得很优秀的结果，同时它也有一些局限性，比如说它需要海量的训练数据，以及强大的计算能力，难以进行跨任务的迁移，并且不具有较好的可解释性。在另一方面，知识图谱作为显式模型，同样也是人工智能的一大技术驱动力，它能够广泛地适用于不同的任务。相比深度学习，知识图谱中的知识可以沉淀，具有较强的可解释性，与人类的思考更加贴近，为隐式的深度模型补充了人类的知识积累，和深度学习互为补充。因此，全球很多大型的互联网公司都在知识图谱领域积极进行布局。



	深度学习	知识图谱
场景示例	人脸识别	语音助手
目前进展	在一些任务上已经接近或超过人类	在知识量上超过人类，在知识推理上不如人类
任务范围	面向具体任务，难以迁移	广泛适用于不同任务
可解释性	较难解释	可解释性强
数据量	海量训练数据	海量知识数据
未来趋势	未来深度融合	

图 1　人工智能两大驱动力

美团连接了数亿用户和数千万商户，背后也蕴含着丰富的日常生活相关知识。2018年，美团知识图谱团队开始构建美团大脑，着力于利用知识图谱技术赋能业务，进一步改善用户体验。具体来说，美团大脑会对美团业务中涉及到的千万级别商家、亿级别的菜品 / 商品、数十亿的用户评论，以及背后百万级别的场景进行深入的理解和结构化的知识建模，构建人、店、商品、场景之间的知识关联，从而形成生活服务领域大规模的知识图谱。现阶段，美团大脑已覆盖了数十亿实体，数百亿三元组，在餐饮、外卖、酒店、金融等场景中验证了知识图谱的有效性。

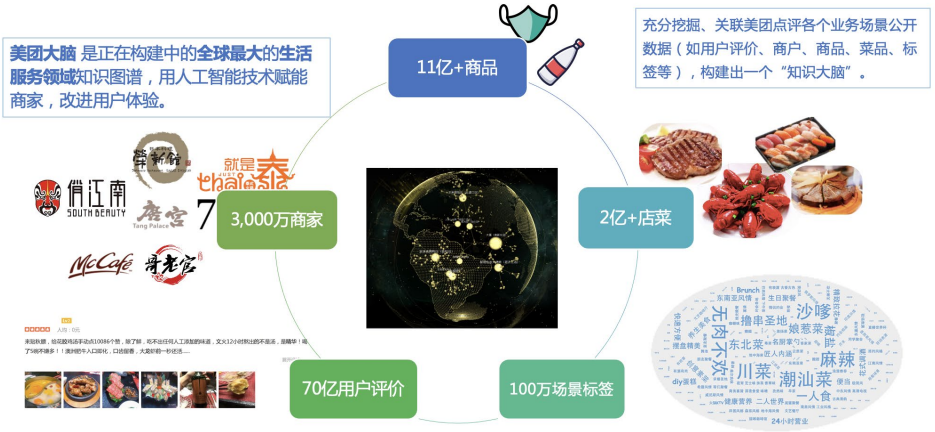


图 2　美团大脑

在新零售领域的探索

美团逐步突破原有边界，在生活服务领域探索新的业务，不仅局限于通过外卖、餐饮帮大家“吃得更好”，近年来也逐步拓展到零售、出行等其他领域，帮大家“生活更好”。在零售领域中，美团先后落地了美团闪购、美团买菜、美团优选、团好货等一系列相应的业务，逐步实现“万物到家”的愿景。为了更好地支持美团的新零售业务，我们需要对背后的零售商品建立知识图谱，积累结构化数据，深入对零售领域内商品、用户、属性、场景等的理解，以便能更好地为用户提供零售商品领域内的服务。

相比于围绕商户的餐饮、外卖、酒店的等领域，零售商品领域对于知识图谱的建设和

应用提出了更大的挑战。一方面，商品数量更加庞大，覆盖的领域范围也更加宽广。另一方面，商品本身所具有的显示信息往往比较稀疏，很大程度上需要结合生活中的常识知识来进行推理，方可将隐藏在背后的数十维的属性进行补齐，完成对商品完整的理解。在下图的例子中，“乐事黄瓜味”这样简单的商品描述其实就对应着丰富的隐含信息，只有对这些知识进行了结构化提取和相应的知识推理后，才能够更好的支持下游搜索、推荐等模块的优化。

图 3　商品结构化信息的应用

商品图谱建设的目标

我们针对美团零售业务的特点，制定了多层级、多维度、跨业务的零售商品知识图谱体系。

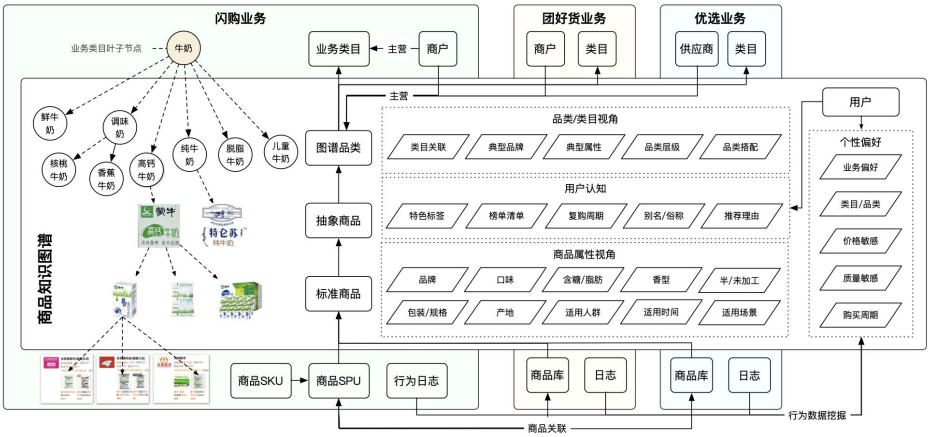


图 4　商品知识图谱体系

多层级

在不同业务的不同应用场景下，对于“商品”的定义会有所差别，需要对各个不同颗粒度的商品进行理解。因此，在我们的零售商品知识图谱中，建立了五层的层级体系，具体包括：

– L1– 商品 SKU/SPU：对应业务中所售卖的商品颗粒度，是用户交易的对象，往往为商户下挂的商品，例如“望京家乐福所售卖的蒙牛低脂高钙牛奶 250ml 盒装”。这一层级也是作为商品图谱的最底层的基石，将业务商品库和图谱知识进行打通关联。

– L2– 标准商品：描述商品本身客观事实的颗粒度，例如“蒙牛低脂高钙牛奶 250ml 盒装”，无论通过什么渠道在什么商户购买，商品本身并没有任何区别。商品条形码则是在标准商品这层的客观依据。在这一层级上，我们可以建模围绕标准商品的客观知识，例如同一个标准商品都会具有同样的品牌、口味、包装等属性。

– L3– 抽象商品：进一步我们将标准商品向上抽象的商品系列，例如“蒙牛低脂高钙牛奶”。在这一层级中，我们不再关注商品具体的包装、规格等，将同系列的商品聚合为抽象商品，承载了用户对于商品的主观认知，包括用户对商品系列的别名俗称、品牌认知、主观评价等。

– L4– 主体品类：描述商品主体的本质品类，列如“鸡蛋”、“奶油草莓”、“台式烤肠”等。这一层作为商品图谱的后台类目体系，以客观的方式对商品领域的品类进行建模，承载了用户对于商品的需求，例如各品牌各产地的鸡蛋都能够满足用户对于鸡蛋这个品类的需求。

– L5– 业务类目：相比于主体品类的后台类目体系，业务类目作为前台类目体系会依据业务当前的发展阶段进行人工定义和调整，各个业务会根据当前业务阶段的特点和需求建立对应的前台类目体系。

多维度

- **商品属性视角**：围绕商品本身，我们需要有海量的属性维度来对商品进行描述。商品属性维度主要分为两类：一类是通用的属性维度，包括品牌、规格、包装、产地等；另一类是品类特有的属性维度，例如对于牛奶品类我们会关注脂肪含量（全脂 / 低脂 / 脱脂牛奶）、存储方式（常温奶、冷藏奶）等。商品属性主要是刻画了商品的客观知识，往往会建立在标准商品这一层级上。
- **用户认知视角**：除了客观的商品属性维度以外，用户往往对于商品会有一系列的主观认知，例如商品的别名俗称（“小黑瓶”、“快乐水”）、对于商品的评价（“香甜可口”、“入口即化”、“性价比高”）、商品的清单 / 榜单（“进口食品榜单”、“夏季消暑常备”）等维度。这些主观认知往往会建立在抽象商品这一层级上。
- **品类 / 类目视角**：从品类 / 类目的视角来看，不同品类 / 类目也会有各自不同的关注点。在这一层级上，我们会建模各个品类 / 类目下有哪些典型的品牌、用户关注哪些典型属性、不同品类的复购周期是多长时间等。

跨业务

美团大脑商品知识图谱的目标是希望能够对客观世界中的商品知识进行建模，而非局限于单个业务之中。在商品图谱的五层体系中，标准商品、抽象商品、品类体系都是与业务解耦的，围绕着客观商品所建立的，包括围绕这些层级建立的各维度数据也均是刻画了商品领域的客观知识。

在应用于各个业务当中时，我们将客观的图谱知识向上关联至业务前台类目，向下关联至业务商品 SPU/SKU，则可以完成各个业务数据的接入，实现各个业务数据和客观知识之间的联通，提供更加全面的跨业务的全景数据视角。利用这样的数据，在用户方面我们可以更加全面的建模、分析用户对于业务、品类的偏好，对于价格、品质等的敏感程度，在商品方面我们可以更准确的建模各品类的复购周期、地域 / 季节 / 节日偏好等。

商品图谱建设的挑战

商品知识图谱的构建的挑战主要来源于以下三个方面：

1. **信息来源质量低**：商品本身所具有的信息比较匮乏，往往以标题和图片为主。尤其在美团闪购这样 LBS 的电商场景下，商户需要上传大量的商品数据，对于商品信息的录入存在很多信息不完整的情况。在标题和图片之外，商品详情虽然也蕴含着大量的知识信息，但是其质量往往参差不齐，并且结构各异，从中进行知识挖掘难度极高。
2. **数据维度多**：在商品领域有众多的数据维度需要进行建设。以商品属性部分为例，我们不仅需要建设通用属性，诸如品牌、规格、包装、口味等维度，同时还要覆盖各个品类 / 类目下特定关注的属性维度，诸如脂肪含量、是否含糖、电池容量等，整体会涉及到上百维的属性维度。因此，数据建设的效率问题也是一大挑战。
3. **依赖常识 / 专业知识**：人们在日常生活中因为有很丰富的常识知识积累，可以通过很简短的描述获取其背后隐藏的商品信息，例如在看到“乐事黄瓜”这样一个商品的时候知道其实是乐事黄瓜味的薯片、看到“唐僧肉”的时候知道其实这不是一种肉类而是一种零食。因此，我们也需要探索结合常识知识的语义理解方法。同时，在医药、个护等领域中，图谱的建设需要依赖较强的专业知识，例如疾病和药品之间的关系，并且此类关系对于准确度的要求极高，需要做到所有知识都准确无误，因此也需要较好的专家和算法相结合的方式来进行高效的图谱构建。

商品图谱建设

在了解了图谱建设的目标和挑战后，接下来我们将介绍商品图谱数据建设的具体方案。

层级体系建设

品类体系建设

本质品类描述了商品本质所属的最细类别，它聚合了一类商品，承载了用户最终的消费需求，如“高钙牛奶”、“牛肉干”等。本质品类与类目也是有一定的区别，类目是若干品类的集合，它是抽象后的品类概念，不能够明确到具体的某类商品品类上，如“乳制品”、“水果”等。

品类打标：对商品图谱的构建来说，关键的一步便是建立起商品和品类之间的关联，即对商品打上品类标签。通过商品和品类之间的关联，我们可以建立起商品库中的商品与用户需求之间的关联，进而将具体的商品展示到用户面前。下面简单介绍下品类打标方法：

1. **品类词表构建**：品类打标首先需要构建一个初步的商品品类词表。首先，我们通过对美团的各个电商业务的商品库、搜索日志、商户标签等数据源进行分词、NER、新词发现等操作，获得初步的商品候选词。然后，通过标注少量的样本进行二分类模型的训练（判断一个词是否是品类）。此外，我们通过结合主动学习的方法，从预测的结果中挑选出难以区分的样本，进行再次标注，继续迭代模型，直到模型收敛。
2. **品类打标**：首先，我们通过对商品标题进行命名实体识别，并结合上一步中的品类词表来获取商品中的候选品类，如识别“蒙牛脱脂牛奶 500ml”中的“脱脂牛奶”、“牛奶”等。然后，在获得了商品以及对应的品类之后，我们利用监督数据训练品类打标的二分类模型，输入商品的 SPU_ID 和候选品类 TAG 构成的 Pair，即，对它进行是否匹配的预测。具体的，我们一方面利用结合业务中丰富的半结构化语料构建围绕标签词的统计特征，另一方面利用命名实体识别、基于 BERT 的语义匹配等模型产出高阶相关性特征，在此基础上，我们将上述特征输入到终判模型中进行模型训练。

3. 品类标签后处理：在这一步中，我们对模型打上的品类进行后处理的一些策略，如基于图片相关性、结合商品标题命名实体识别结果等的品类清洗策略。

通过上述的三个步骤，我们便可以建立起商品与品类之间的联系。

品类体系：品类体系由品类和品类间关系构成。常见的品类关系包括同义词和上下位等。在构建品类体系的过程中，常用的以下几种方法来进行关系的补全。我们主要使用下面的一些方法：

1. **基于规则的品类关系挖掘**。在百科等通用语料数据中，有些品类具有固定模式的描述，如“玉米又名苞谷、苞米棒子、玉蜀黍、珍珠米等”、“榴莲是著名热带水果之一”，因此，可以使用规则从中提取同义词和上下位。
2. **基于分类的品类关系挖掘**。类似于上文中提到的品类打标方法，我们将同义词和上下位构建为的样本，通过在商品库、搜索日志、百科数据、UGC 中挖掘的统计特征以及基于 Sentence-BERT 得到的语义特征，使用二分类模型进行品类关系是否成立的判断。对于训练得到的分类模型，我们同样通过主动学习的方式，选出结果中的难分样本，进行二次标注，进而不断迭代数据，提高模型性能。
3. **基于图的品类关系推理**。在获得了初步的同义词、上下位关系之后，我们使用已有的这些关系构建网络，使用 GAE、VGAE 等方法对网络进行链路预测，从而进行图谱边关系的补全。

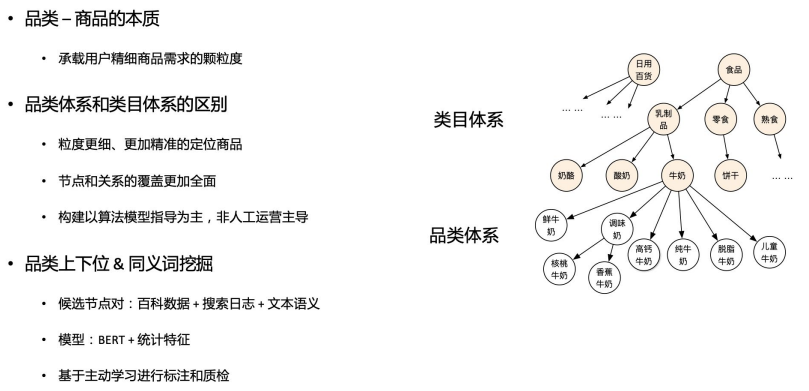


图 5　商品图谱品类体系的构建

标准 / 抽象商品

标准商品是描述商品本身客观事实的颗粒度，和销售渠道和商户无关，而商品条形码是标准商品这层的客观依据。标品关联即将同属于某个商品条形码的业务 SKU/SPU，都正确关联到该商品条形码上，从而在标准商品层级上建模相应的客观知识，例如标准商品对应的品牌、口味和包装等属性。 下面通过一个案例来说明标品关联的具体任务和方案。

案例：下图是一个公牛三米插线板的标准商品。商家录入信息的时候，会把商品直接关联到商品条码上。通过商户录入数据完成了一部分的标品关联，但这部分比例比较少，且存在大量的链接缺失，链接错误的问题。另外，不同的商家对于同样的标品，商品的标题的描述是千奇百怪的。我们的目标是补充缺失的链接，将商品关联到正确的标品上。

图 6　商品图谱标品关联任务

针对标品关联任务，我们构建了商品领域的同义词判别模型：通过远监督的方式利用商户已经提供的少量有关联的数据，作为已有的知识图谱构造远监督的训练样本。在模型中，正例是置信度比较高的标品码；负例是原始数据中商品名或者图像类似但不属于同一标品的 SPU。构造准确率比较高的训练样本之后，通过 BERT 模型进行同义词模型训练。最后，通过模型自主去噪的方式，使得最终的准确率能够达到 99%

以上。总体能做到品牌，规格，包装等维度敏感。

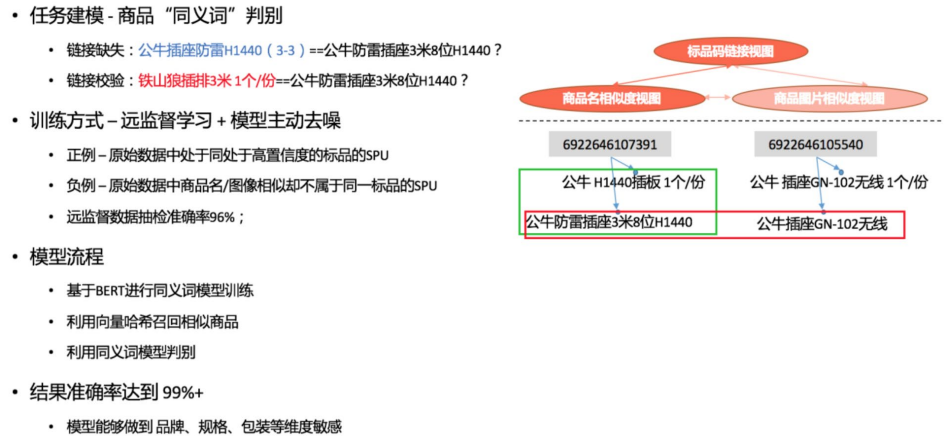


图 7　商品图谱标品关联方法

抽象商品是用户认知的层面，作为用户所评论的对象，这一层对用户偏好建模更加有效。同时，在决策信息的展示上，抽象商品粒度也更符合用户认知。例如下图所示冰淇淋的排行榜中，罗列了用户认知中抽象商品对应的 SKU，然后对应展示不同抽象商品的特点、推荐理由等。抽象商品层整体的构建方式，和标准商品层比较类似，采用标品关联的模型流程，并在数据构造部分进行规则上的调整。

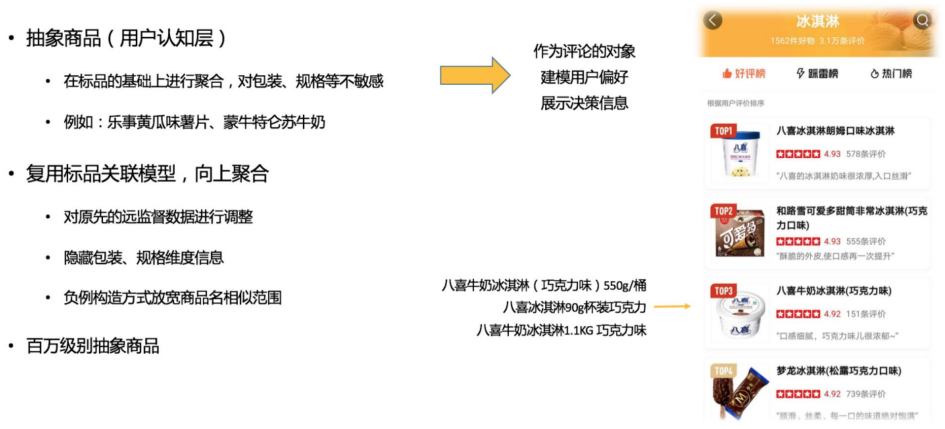


图 8　商品图谱抽象商品聚合

属性维度建设

对一个商品的全面理解，需要涵盖各个属性维度。例如“乐事黄瓜味薯片”，需要挖掘它对应的品牌、品类、口味、包装规格、标签、产地以及用户评论特色等属性，才能在商品搜索、推荐等场景中精准触达用户。商品属性挖掘的源数据主要包含商品标题、商品图片和半结构化数据三个维度。

图 9　商品图谱属性建设

商品标题包含了对于商品最重要的信息维度，同时，商品标题解析模型可以应用在查询理解中，对用户快速深入理解拆分，为下游的召回排序也能提供高阶特征。因此，这里我们着重介绍一下利用商品标题进行属性抽取的方法。

商品标题解析整体可以建模成文本序列标注的任务。例如，对于商品标题“乐事黄瓜薯片”，目标是理解标题文本序列中各个成分，如乐事对应品牌，黄瓜对应口味，薯片是品类，因此我们使用命名实体识别（NER）模型进行商品标题解析。然而商品标题解析存在着三大挑战：（1）上下文信息少；（2）依赖常识知识；（3）标注数据通常有较多的噪音。为了解决前两个挑战，我们首先尝试在模型中引入了图谱信息，主要包含以下三个维度：

- **节点信息**：将图谱实体作为词典，以 Soft-Lexicon 方式接入，以此来缓解 NER 的边界切分错误问题。

- **关联信息**：商品标题解析依赖常识知识，例如在缺乏常识的情况下，仅从标题“乐事黄瓜薯片”中，我们无法确认“黄瓜”是商品品类还是口味属性。因此，我们引入知识图谱的关联数据缓解了常识知识缺失的问题：在知识图谱中，乐事和薯片之间存在着“品牌 – 售卖 – 品类”的关联关系，但是乐事跟黄瓜之间则没有直接的关系，因此可以利用图结构来缓解 NER 模型常识知识缺少的问题。具体来说，我们利用 Graph Embedding 的技术对图谱进行的嵌入表征，利用图谱的图结构信息对图谱中的单字，词进行表示，然后将包含了图谱结构信息的嵌入表示和文本语义的表征进行拼接融合，再接入到 NER 模型之中，使得模型能够既考虑到语义，也考虑到常识知识的信息。
- **节点类型信息**：同一个词可以代表不同的属性，比如“黄瓜”既可以作为品类又可以作为属性。因此，对图谱进行 Graph Embedding 建模的时候，我们根据不同的类型对实体节点进行拆分。在将图谱节点表征接入 NER 模型中时，再利用注意力机制根据上下文来选择更符合语义的实体类型对应的表征 ，缓解不同类型下词语含义不同的问题，实现不同类型实体的融合。

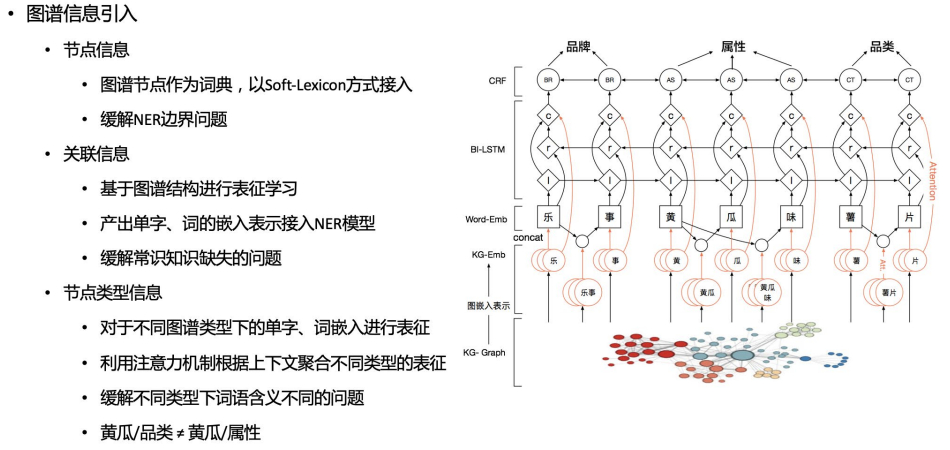


图 10　商品图谱标题解析

接下来我们探讨如何缓解标注噪音的问题。在标注过程中，少标漏标或错标的问题无法避免，尤其像在商品标题 NER 这种标注比较复杂的问题上，尤为显著。对于标注数据中的噪音问题，采用以下方式对噪音标注优化：不再采取原先非 0 即

1 的 Hard 的训练方式，而是采用基于置信度数据的 Soft 训练方式，然后再通过 Bootstrapping 的方式迭代交叉验证，然后根据当前的训练集的置信度进行调整。我们通过实验验证，使用 Soft 训练 +Bootstrapping 多轮迭代的方式，在噪声比例比较大的数据集上，模型效果得到了明显提升。具体的方法可参见我们在 NLPCC 2020 比赛中的论文《Iterative Strategy for Named Entity Recognition with Imperfect Annotations》。

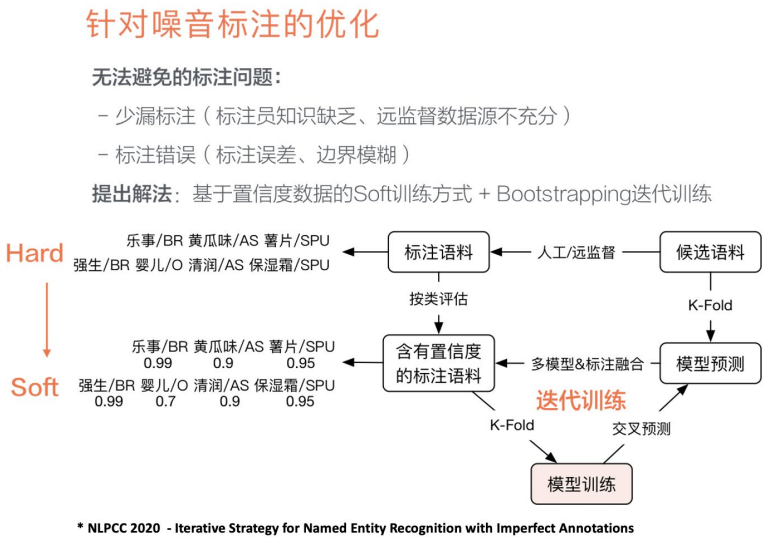


图 11　基于噪音标注的 NER 优化

效率提升

知识图谱的构建往往是针对于各个领域维度的数据单独制定的挖掘方式。这种挖掘方式重人工，比较低效，针对每个不同的领域、每个不同的数据维度，我们都需要定制化的去建设任务相关的特征及标注数据。在商品场景下，挖掘的维度众多，因此效率方面的提高也是至关重要的。我们首先将知识挖掘任务建模为三类分类任务，包括节点建模、关系建模以及节点关联。在整个模型的训练过程中，最需要进行效率优化的其实就是上述提到的两个步骤：(1) 针对任务的特征提取；(2) 针对任务的数据标注。

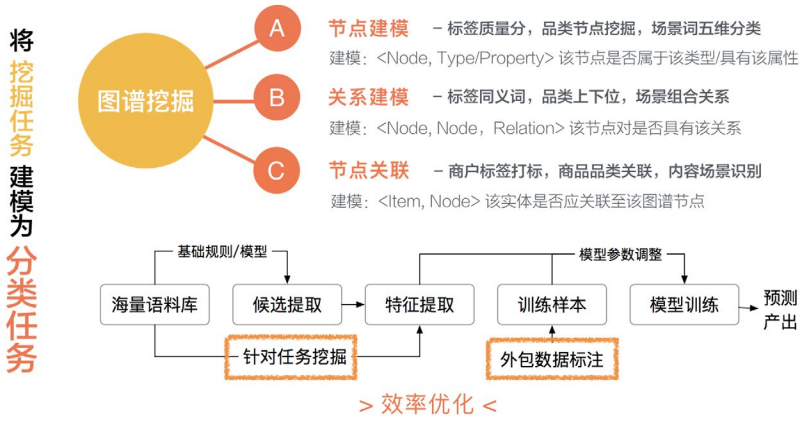


图 12　知识挖掘任务建模

针对特征提取部分，我们摒弃了针对不同挖掘任务做定制化特征挖掘的方式，而是尝试将特征和任务解耦，构建跨任务通用的图谱挖掘特征体系，利用海量的特征库来对目标的节点 / 关系 / 关联进行表征，并利用监督训练数据来进行特征的组合和选择。具体的，我们构建的图谱特征体系主要由四个类型的特征组构成：1. 规则模板型特征主要是利用人工先验知识，融合规则模型能力。2. 统计分布型特征，可以充分利用各类语料，基于不同语料不同层级维度进行统计。3. 句法分析型特征则是利用 NLP 领域的模型能力，引入分词、词性、句法等维度特征。4. 嵌入表示型特征，则是利用高阶模型能力，引入 BERT 等语义理解模型的能力。

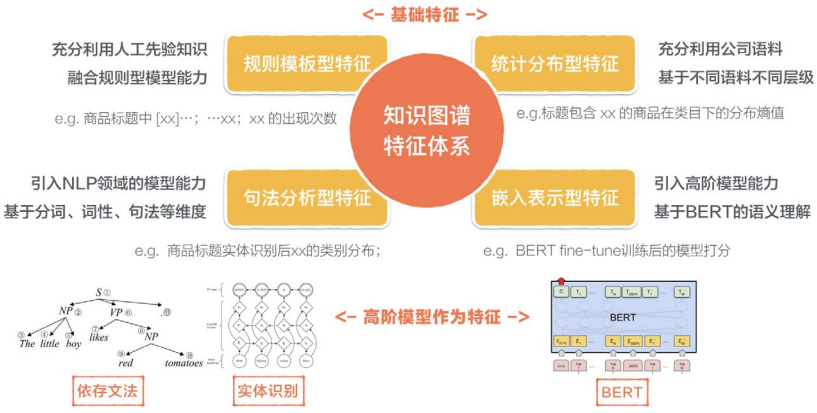


图 13　知识挖掘特征体系

针对数据标注部分，我们主要从三个角度来提升效率。1. 通过半监督学习，充分的利用未标注的数据进行预训练。2. 通过主动学习技术，选择对于模型来说能够提供最多信息增益的样本进行标注。3. 利用远程监督方法，通过已有的知识构造远监督样本进行模型训练，尽可能的发挥出已有知识的价值。

人机结合 – 专业图谱建设

当前医药健康行业结构性正在发生变化，消费者更加倾向于使用在线医疗解决方案和药品配送服务，因此医药业务也逐渐成为了美团的重要业务之一。相比于普通商品知识图谱的建设，药品领域知识具有以下两个特点：(1) 具有极强的专业性，需要有相关背景知识才能判断相应的属性维度，例如药品的适用症状等。(2) 准确度要求极高，对于强专业性知识不允许出错，否则更容易导致严重后果。因此我们采用将智能模型和专家知识结合的方式来构建药品知识图谱。

药品图谱中的知识可以分为弱专业知识和强专业知识两类，弱专业知识即一般人能够较容易获取和理解的知识，例如药品的使用方法、适用人群等；而强专业知识则是需要具有专业背景的人才能够判断的知识，例如药品的主治疾病、适应症状等。由于这两类数据对专家的依赖程度不同，因此我们分别采取不同的挖掘链路：

- **弱专业知识**：对于药品图谱的弱专业知识挖掘，我们从说明书、百科知识等数据源中提取出相应的信息，并结合通过专家知识沉淀出来的规则策略，借助通用语义模型从中提取相应的知识，并通过专家的批量抽检，完成数据的建设。
- **强专业知识**：对于药品图谱的强专业知识挖掘，为了确保相关知识百分百准确，我们通过模型提取出药品相关属性维度的候选后，将这些候选知识给到专家进行全量质检。在这里，我们主要是通过算法的能力，尽可能减少专业药师在基础数据层面上的精力花费，提高专家从半结构化语料中提取专业知识的效率。

在药品这类专业性强的领域，专业知识的表述和用户习惯往往存在差异。因此我们除了挖掘强弱专业知识外，还需要填补专业知识和用户之间的差异，才能将药品图谱更好的与下游应用结合。为此，我们从用户行为日志以及领域日常对话等数据源中，挖

掘了疾病、症状和功效的别名数据，以及药品通用名的俗称数据，来打通用户习惯和专业表述之间的通路。

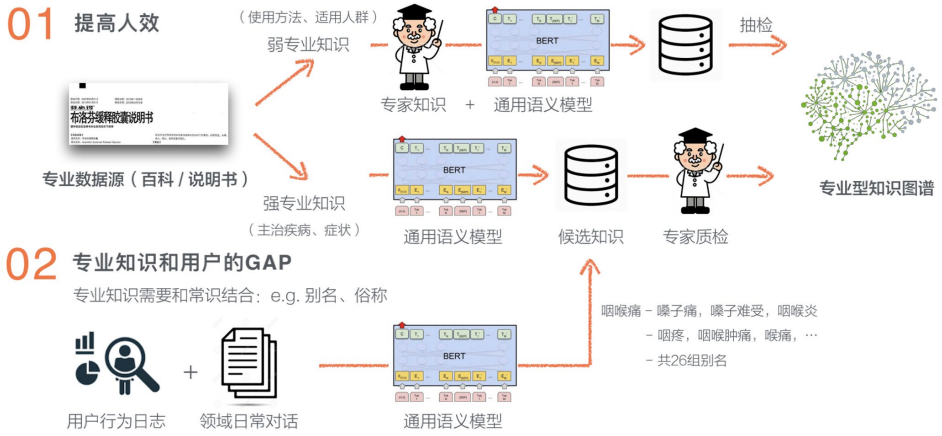


图 14　人机结合的专业知识挖掘

商品图谱的落地应用

自从谷歌将知识图谱应用于搜索引擎，并显著提升了搜索质量与用户体验，知识图谱在各垂直领域场景都扮演起了重要的角色。在美团商品领域中，我们也将商品图谱有效的应用在围绕商品业务的搜索、推荐、商家端、用户端等多个下游场景当中，接下来我们举几个典型的案例进行介绍。

结构化召回

商品图谱的数据，对于商品的理解很有帮助。例如，在商品搜索中，如用户在搜索头疼腰疼时，通过结构化的知识图谱，才能知道什么药品是有止疼功效的；用户在搜索可爱多草莓、黄瓜薯片时，需要依赖图谱的常识知识来理解用户真正需求是冰淇淋和薯片，而不是草莓和黄瓜。

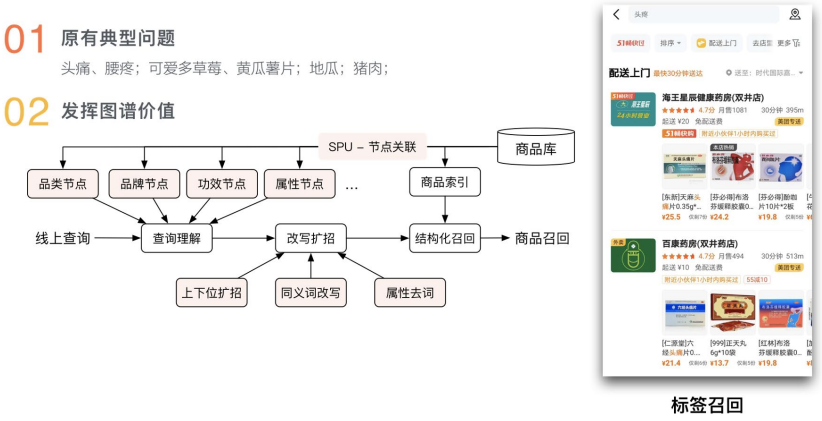


图 15　基于图谱的结构化召回

排序模型泛化性

图谱的类目信息、品类信息、属性信息，一方面可以作为比较强有力的相关性的判断方法和干预手段，另一方面可以提供不同粗细粒度的商品聚合能力，作为泛化性特征提供到排序模型，能有效地提升排序模型的泛化能力，对于用户行为尤为稀疏的商品领域来说则具有着更高的价值。具体的特征使用方式则包括：1. 通过各颗粒度进行商品聚合，以 ID 化特征接入排序模型。2. 在各颗粒度聚合后进行统计特征的建设。3. 通过图嵌入表示的方式，将商品的高维向量表示和排序模型结合。

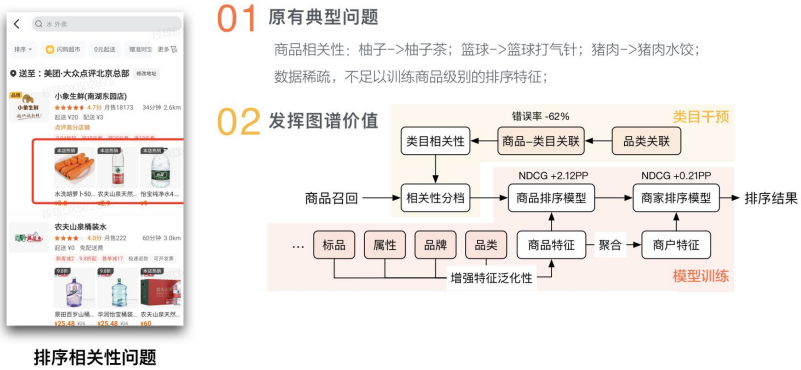
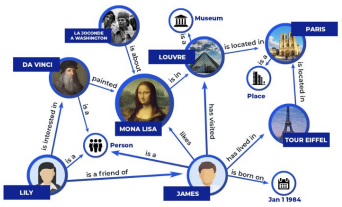


图 16　基于图谱的排序优化

多模态图谱嵌入

现有的研究工作已经在多个领域中证明了，将知识图谱的数据进行嵌入表示，以高维向量表示的方式和排序模型结合，可以有效地通过引入外部知识达到缓解排序 / 推荐场景中数据稀疏以及冷启动问题的效果。然而，传统的图谱嵌入的工作往往忽视了知识图谱中的多模态信息，例如商品领域中我们有商品的图片、商品的标题、商家的介绍等非简单的图谱节点型的知识，这些信息的引入也可以进一步提升图谱嵌入对推荐 / 排序的信息增益。

- 多模态知识图谱
  - 传统图谱仅考虑：Entity、Relation
  - 多模态图谱还考虑：图片、文本、位置等
- 多模态信息价值
  - 商品图片、电影剧照、餐厅风格
  - 简介、标签、用户评论
  - 非图谱方式使用
    - 直接作为Item侧特征接入推荐系统
    - 优势：实现方式基础、简单
    - 劣势：信息无法得到有效传播



* CIKM 2020 - Multi-Modal Knowledge Graphs for Recommender Systems

图 17　基于多模态图谱的推荐 – 背景

现有的图谱嵌入方法在应用到多模态图谱表征的时候会存在一些问题，因为在多模态场景下，图谱中边的含义不再是单纯的语义推理关系，而是存在多模态的信息补充的关系，因此我们也针对多模态图谱的特点，提出了 MKG Entity Encoder 和 MKG Attention Layer 来更好的建模多模态知识图谱，并将其表征有效的接入至推荐 / 排序模型中，具体方法可以参考我们在 CIKM 2020 发表了的论文《Multi-Modal Knowledge Graphs for Recommender Systems》。

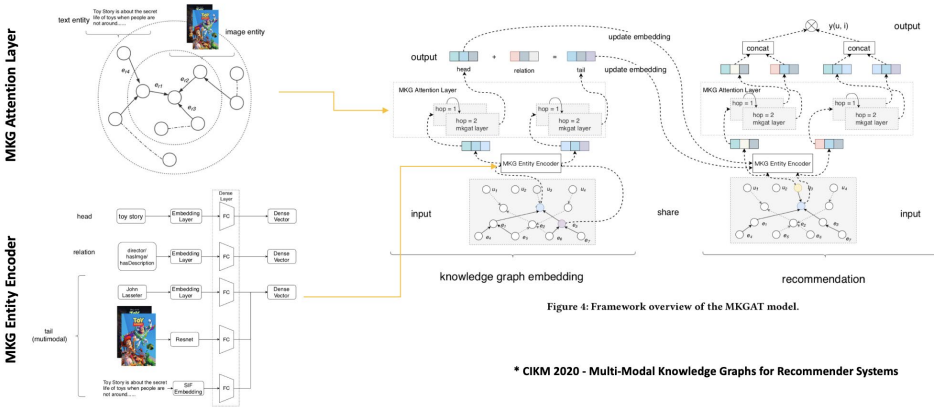


Figure 4: Framework overview of the MKGAT model.

图 18　基于图谱的排序优化 – 模型

用户 / 商家端优化

商品图谱在用户端提供显式化的可解释性信息，辅助用户进行决策。具体的呈现形式包括筛选项、特色标签、榜单、推荐理由等。筛选项的维度受当前查询词对应品类下用户关注的属性类别决定，例如，当用户搜索查询词为薯片时，用户通常关注的是它的口味、包装、净含量等，我们将会根据供给数据在这些维度下的枚举值展示筛选项。商品的特色标签则来源于标题、商品详情页信息与评论数据的提取，以简洁明了的结构化数据展示商品特色。商品的推荐理由通过评论抽取与文本生成两种渠道获得，与查询词联动，以用户视角给出商品值得买的原因，而榜单数据则更为客观，以销量等真实数据，反应商品品质。

在商家端，即商家发布侧，商品图谱则提供了基于商品标题的实时预测能力，帮助商家进行类目的挂载、属性信息的完善。例如，商家填写标题“德国进口德亚脱脂纯牛奶 12 盒”后，商品图谱提供的在线类目预测服务可将其挂载到“食品饮料 – 乳制品 – 纯牛奶”类目，并通过实体识别服务，得到商品的“产地 – 德国”，“是否进口 – 进口”，“品牌 – 德亚”，“脂肪含量 – 脱脂”，“规格 –12 盒”的属性信息，预测完成后，由商家确认发布，降低商家对商品信息的维护成本，并提升发布商品的信息质量。

作者简介

雪智，凤娇，姿雯，匡俊，林森，武威等，均来自美团平台搜索与 NLP 部 NLP 中心。

招聘信息

美团大脑知识图谱团队大量岗位持续招聘中，实习、校招、社招均可，坐标北京 / 上海，欢迎感兴趣的同学加入我们，利用自然语言和知识图谱技术，帮大家吃得更好，生活更好。简历可投递至：caoxuezhi@meituan.com。

常识性概念图谱建设以及在美团场景中的应用

作者：宗宇　俊杰　慧敏等

一、引言

在自然语言处理中，我们经常思考，怎么样才能做好自然语言的理解工作。对我们人类来说，理解某一个自然语言的文本信息，通常都是通过当前的信息，关联自己大脑中存储的关联信息，最终理解信息。例如“他不喜欢吃苹果，但是喜欢吃冰淇淋”，人在理解的时候关联出大脑中的认知信息：苹果，甜的，口感有点脆；冰淇淋，比苹果甜，口感软糯、冰凉，夏天能解暑；小孩更喜欢吃甜食和冰淇淋。所以结合这样的知识，会推理出更喜欢冰淇淋的若干原因。但是现在很多自然语言理解的工作还是聚焦在信息的层面，现在的理解工作类似于一个贝叶斯概率，从已知的训练文本中寻找符合条件的最大化文本信息。

在自然语言处理中做到像人一样去理解文本是自然语言处理的终极目标，所以现在越来越多的研究上，引入了一些额外的知识，帮助机器做好自然语言文本的理解工作。单纯的文本信息只是外部客观事实的表述，知识则是在文本信息基础之上对外部客观事实的归纳和总结，所以在自然语言处理中加入辅助的知识信息，让自然语言理解的更好。

建立知识体系则是一种直接的方式，能够帮助自然语言理解得更准确。知识图谱就是围绕这个思想提出，期望通过给机器显性的知识，让机器能够像人一样进行推理理解。所以在 2012 年 Google 正式提出了知识图谱（Knowledge Graph）的概念，它的初衷是为了优化搜索引擎返回的结果，增强用户的搜索质量及体验。

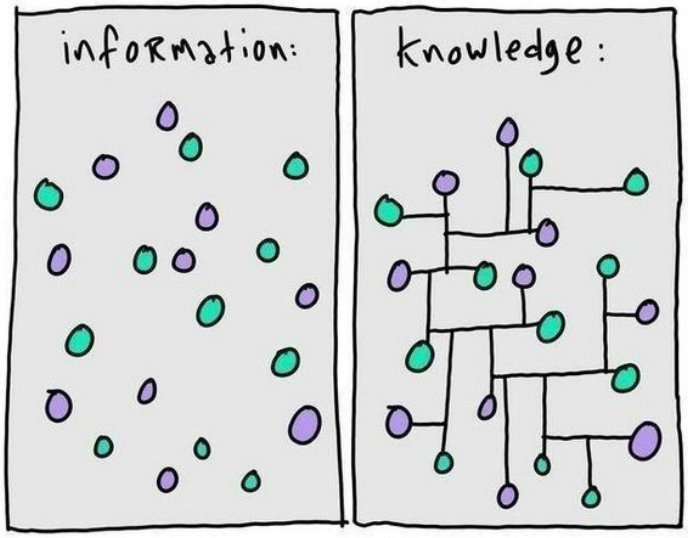


图 1　信息和知识

二、常识性概念图谱介绍

常识性概念图谱就是建立概念与概念之间的关系，帮助自然语言文本的理解。同时我们的常识性概念图谱侧重美团场景，帮助提升美团场景中的搜索、推荐、Feeds 流等的效果。

按照理解的需求，主要是三个维度的理解能力：

1. **是什么**，概念是什么，建立核心概念是什么的关联体系。例如“维修洗衣机”，“维修”是什么，“洗衣机”是什么。
2. **什么样**，核心概念某一方面的属性，对核心概念某一方面的细化。“带露台的餐厅”、“亲子游乐园”、“水果千层蛋糕”中“带露台”、“亲子”、“水果千层”这些都是核心概念某一个方面的属性，所以需要建立核心概念对应属性以及属性值之间的关联。
3. **给什么**，解决搜索概念和承接概念之间的 Gap，例如“阅读”、“逛街”、“遛娃”等没有明确对应的供给概念，所以建立搜索和供给概念之间的关联网络，解决这一类问题。

总结下来，涵盖“是什么”的概念 Taxonomy 体系结构，“什么样”的概念属性关系，“给什么”的概念承接关系。同时 POI（Point of Interesting）、SPU（Standard Product Unit）、团单作为美团场景中的实例，需要和图谱中的概念建立连接。

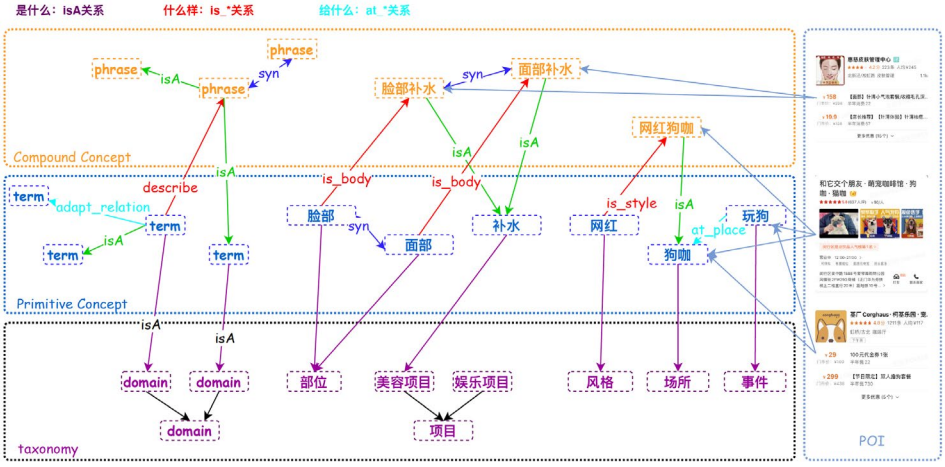


图 2　常识性概念图谱关系

从建设目标出发，拆解整体常识性概念图谱建设工作，拆分为三类节点和四类关系，具体内容如下。

2.1 图谱三类节点

Taxonomy 节点：在概念图谱中，理解一个概念需要合理的知识体系，预定义好的 Taxonomy 知识体系作为理解的基础，在预定义的体系中分为两类节点：第一类在美团场景中可以作为核心品类出现的，例如，食材、项目、场所；另一类是作为对核心品类限定方式出现的，例如，颜色、方式、风格。这两类的节点的定义都能帮助搜索、推荐等的理解。目前预定义的 Taxonomy 节点如下图所示：

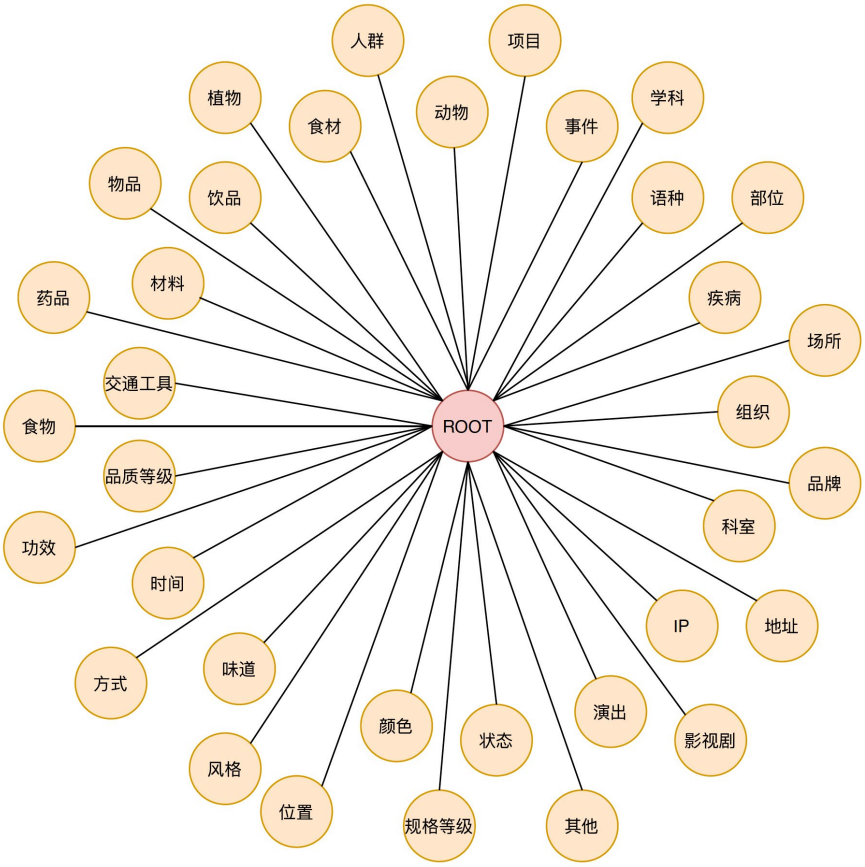


图 3　图谱 Taxonomy 预定义体系

原子概念节点：组成图谱最小语义单元节点，有独立语义的最小粒度词语，例如网红、狗咖、脸部、补水等。定义的原子概念，全部需要挂靠到定义的 Taxonomy 节点之上。

复合概念节点：由原子概念以及对应属性组合而成的概念节点，例如脸部补水、面部补水等。复合概念需要和其对应的核心词概念建立上下位关系。

2.2 图谱四类关系

同义 / 上下位关系：语义上的同义 / 上下位关系，例如脸部补水 –syn– 面部补水等。定义的 Taxonomy 体系也是一种上下位的关系，所以归并到同义 / 上下位关系里。

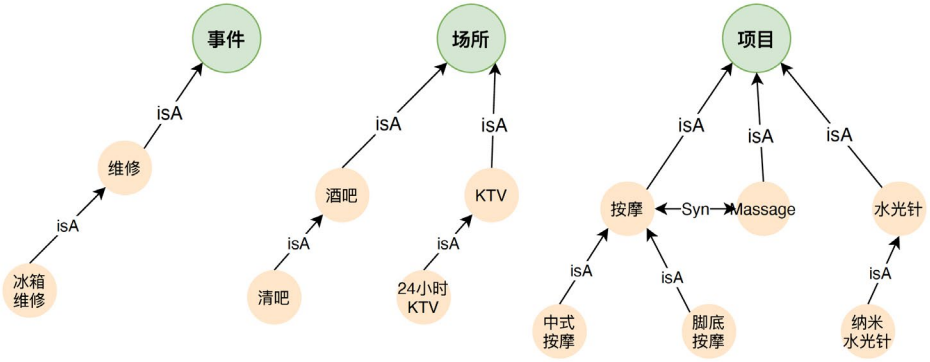


图 4　上下位、同义关系示例

概念属性关系：是典型的 CPV（Concept-Property-Value）关系，从各个属性维度来描述和定义概念，例如火锅 – 口味 – 不辣，火锅 – 规格 – 单人等，示例如下：

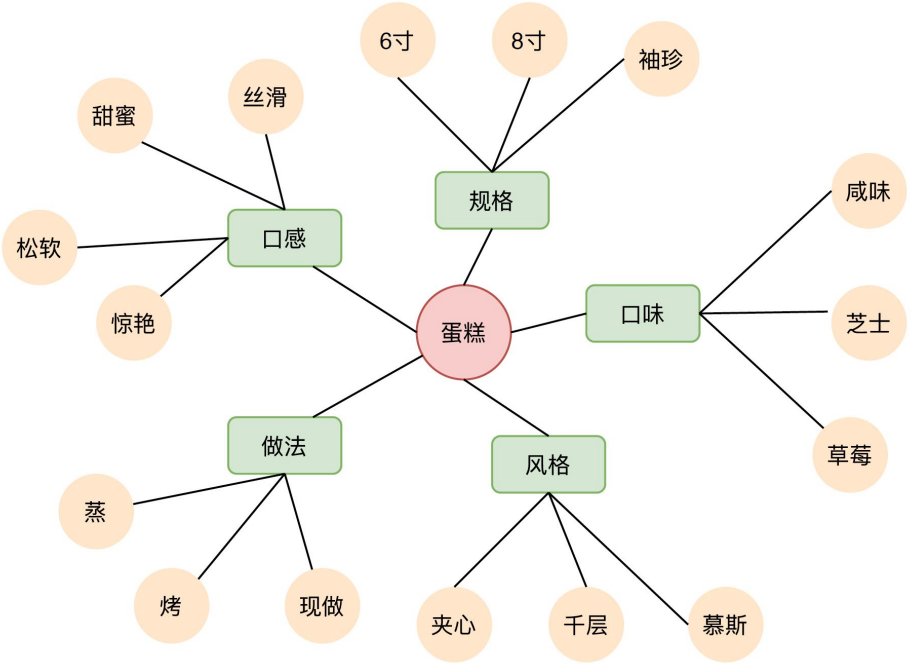


图 5　概念属性关系示例

概念属性关系包含两类。

预定义概念属性：目前我们预定义典型的概念属性如下：

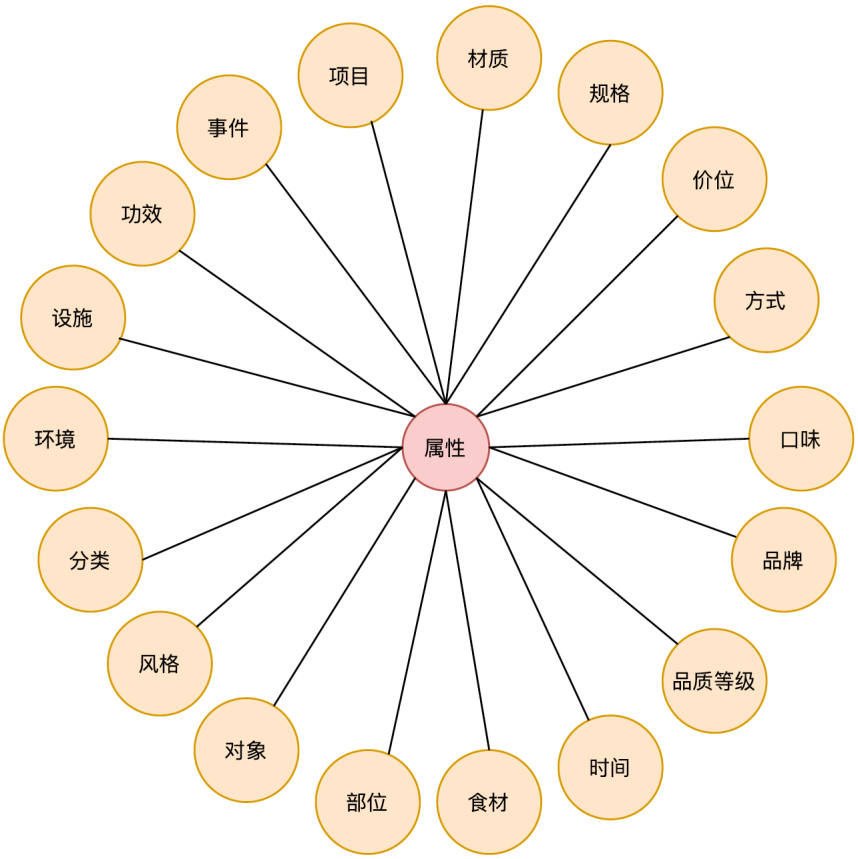


图 6　预定义属性

开放型概念属性：除了我们自己定义的公共的概念属性外，我们还从文本中挖掘一些特定的属性词，补充一些特定的属性词。例如，姿势、主题、舒适度、口碑等。

概念承接关系：这类关系主要建立用户搜索概念和美团承接概念之间的链接，例如踏春 – 场所 – 植物园，减压 – 项目 – 拳击等。

概念承接关系以「事件」为核心，定义了「场所」、「物品」、「人群」、「时间」、「功效」等能够满足用户需求的一类供给概念。以事件“美白”为例，“美白”作为用户的需求，可以有不同的供给概念能够满足，例如美容院、水光针等。目前，定义的几

类承接关系如下图所示：

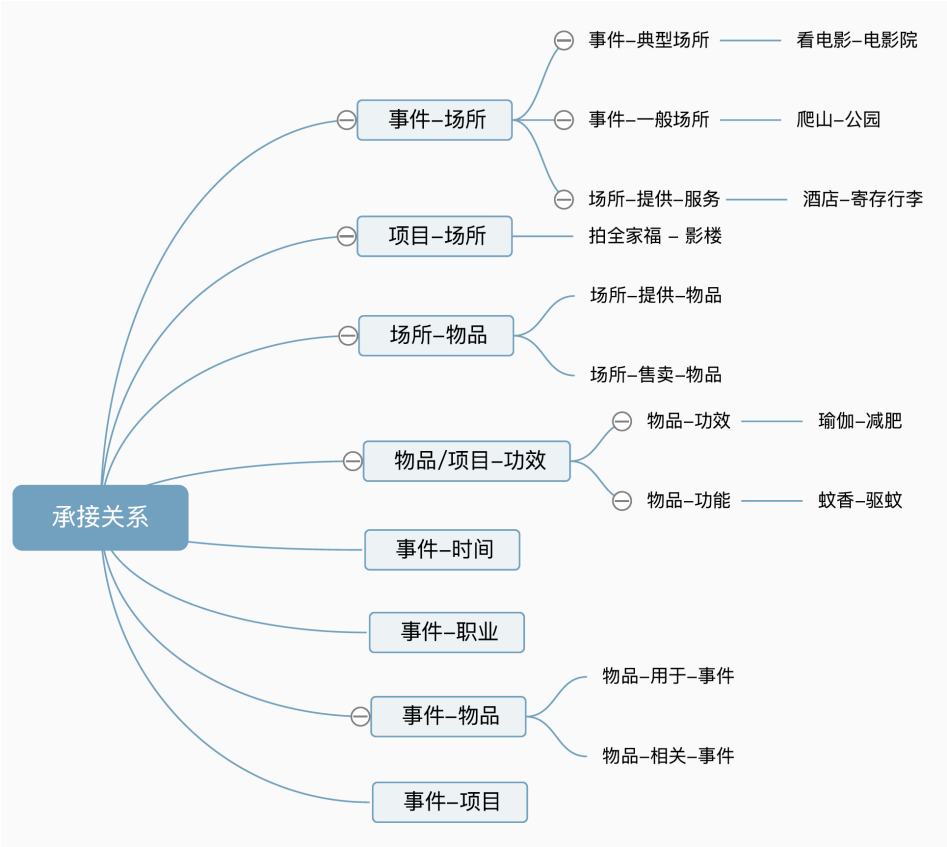


图 7　概念承接关系类型

POI/SPU– 概念关系：POI 作为美团场景中的实例，实例 – 概念的关系作为知识图谱中最后的一站，常常是比较能发挥知识图谱在业务上价值的地方。在搜索、推荐等业务场景，最终的目的是能够展示出符合用户需求的 POI，所以建立 POI/SPU– 概念的关系是整个美团场景常识性概念图谱重要的一环，也是比较有价值的数据。

三、常识性概念图谱构建

图谱构建整体框架如下图所示：

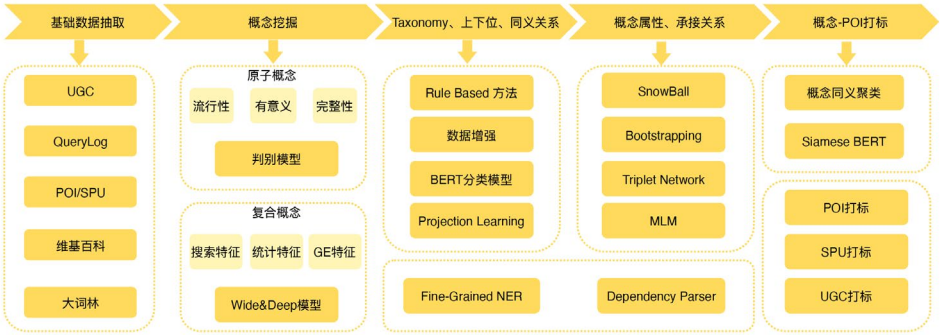


图 8　概念图谱建设整体工作

3.1 概念挖掘

常识性概念图谱的各种关系都是围绕概念构建，这些概念的挖掘是常识性概念图谱建设的第一环。按照原子概念和复合概念两种类型，分别采取相应的方法进行挖掘。

3.1.1 原子概念挖掘

原子概念候选来自于 Query、UGC（User Generated Content）、团单等文本分词后的最小片段，原子概念的判断标准是需要满足流行性、有意义、完整性三个特性的要求。

1. **流行性**，一个概念应是某个或某些语料内流行度较高的词，该特性主要通过频率类特征度量，如“桌本杀”这个词搜索量很低且 UGC 语料中频率也很低，不满足流行性要求。
2. **有意义**，一个概念应是一个有意义的词，该特性主要通过语义特征度量，如“阿猫”、“阿狗”通常只表一个单纯的名称而无其他实际含义。
3. **完整性**，一个概念应是一个完整的词，该特性主要通过独立检索占比（该词作为 Query 的搜索量 / 包含该词的 Query 的总搜索量）衡量，如“儿童设”是一个错误的分词候选，在 UGC 中频率较高，但独立检索占比低。

基于原子概念以上的特性，结合人工标注以及规则自动构造的训练数据训练 XGBoost 分类模型对原子概念是否合理进行判断。

3.1.2 复合概念挖掘

复合概念候选来自于原子概念的组合，由于涉及组合，复合概念的判断比原子概念判断更为复杂。复合概念要求在保证完整语义的同时，在美团站内也要有一定的认知。根据问题的类型，采用 Wide&Deep 的模型结构，Deep 侧负责语义的判断，Wide 侧引入站内的信息。

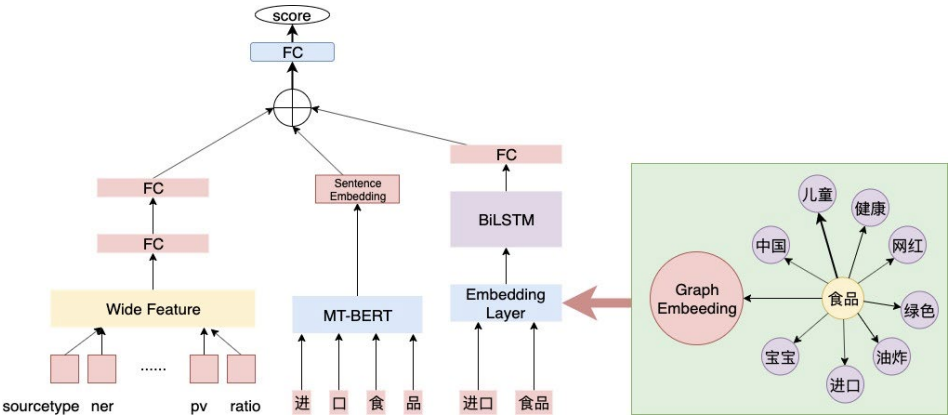


图 9　复合概念挖掘的 Wide&Deep 模型

该模型结构有以下两个特点，对复合概念的合理性进行更准确的判断：

1. **Wide&Deep 模型结构**：将离散特征与深度模型结合起来判断复合概念是否合理。
2. **Graph Embedding 特征**：引入词组搭配间的关联信息，如“食品”可以与“人群”、“烹饪方式”、“品质”等进行搭配。

3.2 概念上下位关系挖掘

在获取概念之后，还需要理解一个概念“是什么”，一方面通过人工定义的 Taxonomy 知识体系中的上下位关系进行理解，另一方面通过概念间的上下位关系进行理解。

3.2.1 概念 -Taxonomy 间上下位关系

概念 -Taxonomy 间上下位关系是通过人工定义的知识体系理解一个概念是什么，由

于 Taxonomy 类型是人工定义好的类型，可以把这个问题转化成一个分类问题。同时，一个概念在 Taxonomy 体系中可能会有多个类型，如“青柠鱼”既是一种“动物”，也属于“食材”的范畴，所以这里最终把这个问题作为一个 Entity Typing 的任务来处理，将概念及其对应上下文作为模型输入，并将不同 Taxonomy 类别放在同一空间中进行判断，具体的模型结构如下图所示：

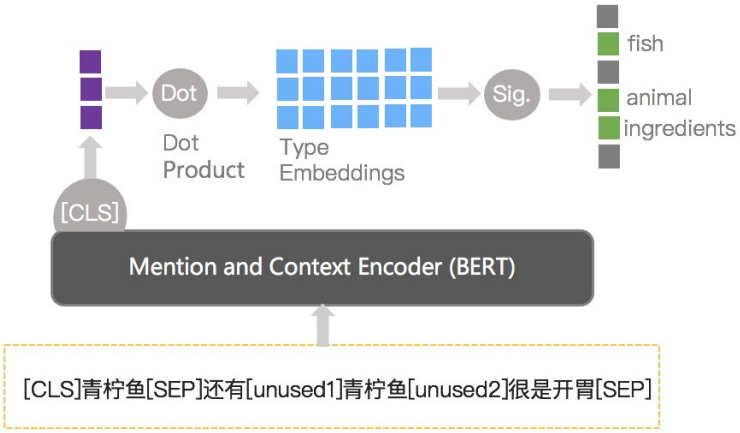


图 10　BERT Taxonomy 关系模型

3.2.2 概念 – 概念间上下位关系

知识体系通过人工定义的类型来理解一个概念是什么，但人工定义的类型始终是有限的，如果上位词不在人工定义的类型中，这样的上下位关系则没办法理解。如可以通过概念 –Taxonomy 关系理解“西洋乐器”、“乐器”、“二胡”都是一种“物品”，但没办法获取到“西洋乐器”和“乐器”、“二胡”和“乐器”之间的上下位关系。基于以上的问题，对于概念 – 概念间存在的上下位关系，目前采取如下两种方法进行挖掘：

基于词法规则的方法：主要解决原子概念和复合概念间的上下位关系，利用候选关系对在词法上的包含关系（如西洋乐器 – 乐器）挖掘上下位关系。

基于上下文判断的方法：词法规则可以解决在词法上存在包含关系的上下位关系对的判断。对于不存在词法上的包含关系的上下位关系对，如“二胡 – 乐器”，首先需要进行上下位关系发现，抽取出“二胡 – 乐器”这样的关系候选，再进行上下位关系判

断，判断“二胡 – 乐器”是一个合理的上下位关系对。考虑到人在解释一个对象时会对这个对象的类型进行相关介绍，如在对“二胡”这个概念进行解释时会提到“二胡是一种传统乐器”，从这样的解释性文本中，既可以将“二胡 – 乐器”这样的关系候选对抽取出来，也能同时实现这个关系候选对是否合理的判断。这里在上下位关系挖掘上分为候选关系描述抽取以及上下位关系分类两部分：

1. **候选关系描述抽取**：两个概念从属于相同的 Taxonomy 类型是一个候选概念对是上下位关系对的必要条件，如“二胡”和“乐器”都属于 Taxonomy 体系中定义的“物品”，根据概念 –Taxonomy 上下位关系的结果，对于待挖掘上下位关系的概念，找到跟它 Taxonomy 类型一致的候选概念组成候选关系对，然后在文本中根据候选关系对的共现筛选出用作上下位关系分类的候选关系描述句。
2. **上下位关系分类**：在获取到候选关系描述句后，需要结合上下文对上下位关系是否合理进行判断，这里将两个概念在文中的起始位置和终止位置用特殊标记标记出来，并以两个概念在文中起始位置标记处的向量拼接起来作为两者关系的表示，根据这个表示对上下位关系进行分类，向量表示使用 BERT 输出的结果，详细的模型结构如下图所示：

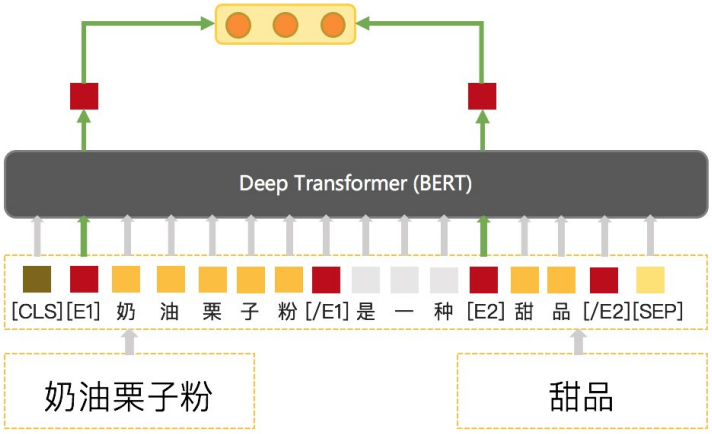


图 11　BERT 上下位关系模型

在训练数据构造上，由于上下位关系表述的句子非常稀疏，大量共现的句子并没有明确的表示出候选关系对是否具有上下位关系，利用已有上下位关系采取远程监督方式

进行训练数据构建不可行，所以直接使用人工标注的训练集对模型进行训练。由于人工标注的数量比较有限，量级在千级别，这里结合 Google 的半监督学习算法 UDA（Unsupervised Data Augmentation）对模型效果进行提升，最终 Precision 可以达到 90%+，详细指标见表 1：

Data Size	BERT			BERT + UDA		
	Precision (%)	Recall (%)	F1 (%)	Precision (%)	Recall (%)	F1 (%)
1300	88.46	80.23	84.15	90.67	79.07	84.47
200	69.07	77.91	73.22	80.82	68.60	74.21

表 1　使用 UDA 在不同训练数据量下的提升效果

3.3 概念属性关系挖掘

概念含有的属性可以按照属性是否通用划分为公共属性和开放属性。公共属性是由人工定义的、大多数概念都含有的属性，例如价位、风格、品质等。开放属性指某些特定的概念才含有的属性，例如，“植发”、“美睫”和“剧本杀”分别含有开放属性“密度”、“翘度”和“逻辑”。开放属性的数量远远多于公共属性。针对这两种属性关系，我们分别采用以下两种方式进行挖掘。

3.3.1 基于复合概念挖掘公共属性关系

由于公共属性的通用性，公共属性关系（CPV）中的 Value 通常和 Concept 以复合概念的形式组合出现，例如，平价商场、日式料理、红色电影高清。我们将关系挖掘任务转化为依存分析和细粒度 NER 任务（可参考《美团搜索中 NER 技术的探索与实践》一文），依存分析识别出复合概念中的核心实体和修饰成分，细粒度 NER 判断出具体属性值。例如，给定复合概念「红色电影高清」，依存分析识别出「电影」这个核心概念，「红色」、「高清」是「电影」的属性，细粒度 NER 预测出属性值分别为「风格（Style）」、「品质评价（高清）」。

依存分析和细粒度 NER 有可以互相利用的信息，例如“毕业公仔”，「时间（Time）」和「产品（Product））」的实体类型，与「公仔」是核心词的依存信息，可以相互促进训练，因此将两个任务联合学习。但是由于两个任务之间的关联程度并不明

确，存在较大噪声，使用 Meta-LSTM，将 Feature-Level 的联合学习优化为 Function-Level 的联合学习，将硬共享变为动态共享，降低两个任务之间噪声影响。

模型的整体架构如下所示：

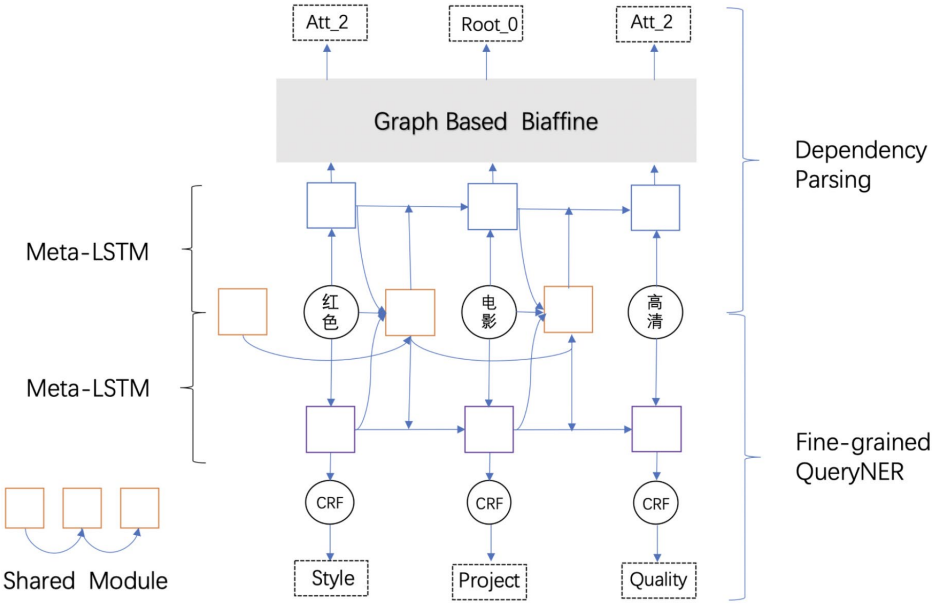


图 12 依存分析 – 细粒度 NER 联合学习模型

目前，概念修饰关系整体准确率在 85% 左右。

3.3.2 基于开放属性词挖掘特定属性关系

开放属性词和属性值的挖掘

开放属性关系需要挖掘不同概念特有的属性和属性值，它的难点在于开放属性和开放属性值的识别。通过观察数据发现，一些通用的属性值（例如：好、坏、高、低、多、少），通常和属性搭配出现（例如：环境好、温度高、人流量大）。所以我们采取一种基于模板的 Bootstrapping 方法自动从用户评论中挖掘属性和属性值，挖掘流程如下：

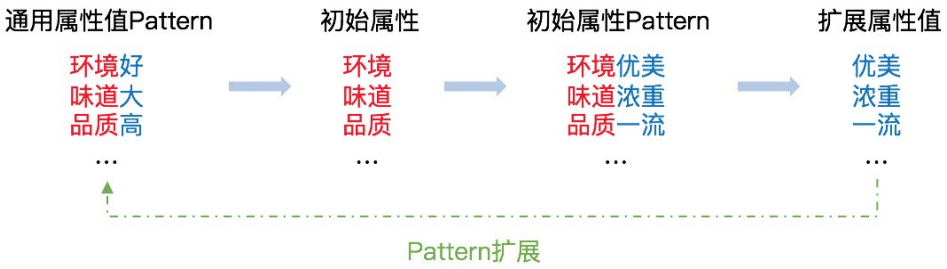


图 13　开放属性挖掘流程

在挖掘了开放属性词和属性值之后，开放属性关系的挖掘拆分为「概念 – 属性」二元组的挖掘和「概念 – 属性 – 属性值」三元组的挖掘。

概念 – 属性的挖掘

「概念 – 属性」二元组的挖掘，即判断概念 Concept 是否含有属性 Property。挖掘步骤如下：

- 根据概念和属性在 UGC 中的共现特征，利用 TFIDF 变种算法挖掘概念对应的典型属性作为候选。
- 将候选概念属性构造为简单的自然表述句，利用通顺度语言模型判断句子的通顺度，保留通顺度高的概念属性。

概念 – 属性 – 属性值的挖掘

在得到「概念 – 属性」二元组后，挖掘对应属性值的步骤如下：

- **种子挖掘**。基于共现特征和语言模型从 UGC 中挖掘种子三元组。
- **模板挖掘**。利用种子三元组从 UGC 中构建合适的模板（例如，“水温是否合适，是选择游泳馆的重要标准。”）。
- **关系生成**。利用种子三元组填充模板，训练掩码语言模型用于关系生成。

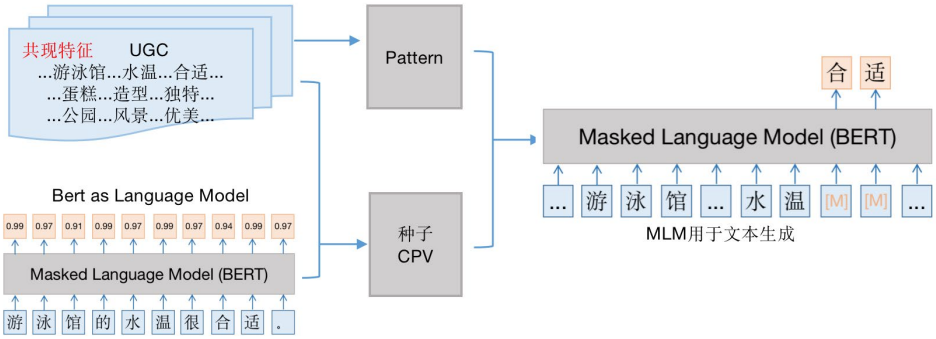


图 14 概念属性关系生成模型

目前，开放领域的概念属性关系准确率在 80% 左右。

3.4 概念承接关系挖掘

概念承接关系是建立用户搜索概念和美团承接概念之间的关联。例如，当用户搜索“踏青”时，真正的意图是希望寻找“适合踏青的地方”，因此平台通过“郊野公园”、“植物园”等概念进行承接。关系的挖掘需要从 0 到 1 进行，所以整个概念承接关系挖掘根据不同阶段的挖掘重点设计了不同的挖掘算法，可以分为三个阶段：①初期的种子挖掘；②中期的深度判别模型挖掘；③后期的关系补全。详细介绍如下。

3.4.1 基于共现特征挖掘种子数据

为了解决关系抽取任务中的冷启动问题，业界通常采用 Bootstrapping 的方法，通过人工设定的少量种子和模板，自动从语料中扩充数据。然而，Bootstrapping 方法不仅受限于模板的质量，而且应用于美团的场景中有着天然缺陷。美团语料的主要来源是用户评论，而用户评论的表述十分口语化及多样化，很难设计通用而且有效的模板。因此，我们抛弃基于模板的方法，而是根据实体间的共现特征以及类目特征，构建了一个三元对比学习网络，自动从非结构化的文本中挖掘实体关系之间潜在的相关性信息。

具体来说，我们观察到不同商户类目下用户评论中实体的分布差异较大。例如，美食类目下的 UGC 经常涉及到“聚餐”、“点菜”、“餐厅”；健身类目下的 UGC 经常涉

及到“减肥”、“私教”、“健身房”；而“装修”、“大厅”等通用实体在各个类目下都会出现。因此，我们构建了三元对比学习网络，使得同类目下的用户评论表示靠近，不同类目的用户评论表示远离。与 Word2Vec 等预训练词向量系统类似，通过该对比学习策略得到的词向量层天然蕴含丰富的关系信息。在预测时，对于任意的用户搜索概念，可以通过计算其与所有承接概念之间的语义相似度，辅以搜索业务上的统计特征，得到一批高质量的种子数据。

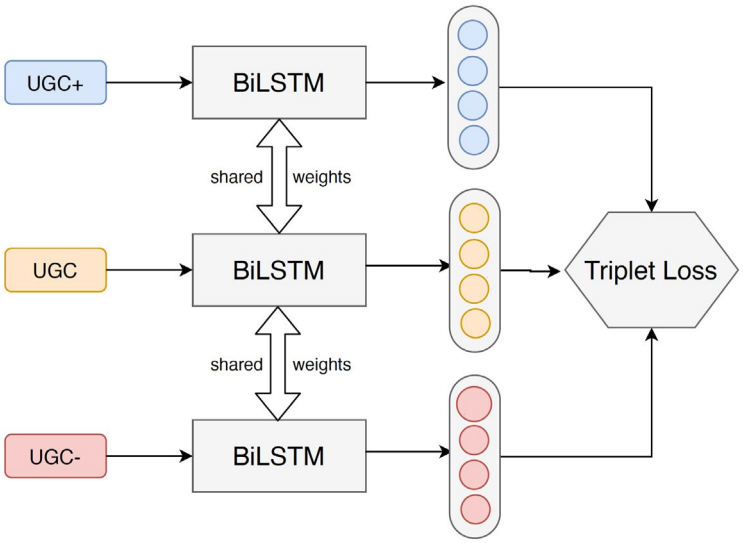


图 15　概念承接关系挖掘的 Triplet Network

3.4.2 基于种子数据训练深度模型

预训练语言模型近两年来在 NLP 领域取得了很大的进展，基于大型的预训练模型微调下游任务，是 NLP 领域非常流行的做法。因此，在关系挖掘中期，我们采用基于 BERT 的关系判别模型（参考《美团 BERT 的探索和实践》一文），利用 BERT 预训练时学到的大量语言本身的知识来帮助关系抽取任务。

模型结构如下图所示。首先，根据实体间的共现特征得到候选实体对，召回包含候选实体对的用户评论；然后，沿用 MTB 论文中的实体标记方法，在两个实体的开始位置和结束位置分别插入特殊的标志符号，经过 BERT 建模之后，将两个实体开始位

置的特殊符号拼接起来作为关系表示；最后，将关系表示输入 Softmax 层判断实体间是否含有关系。

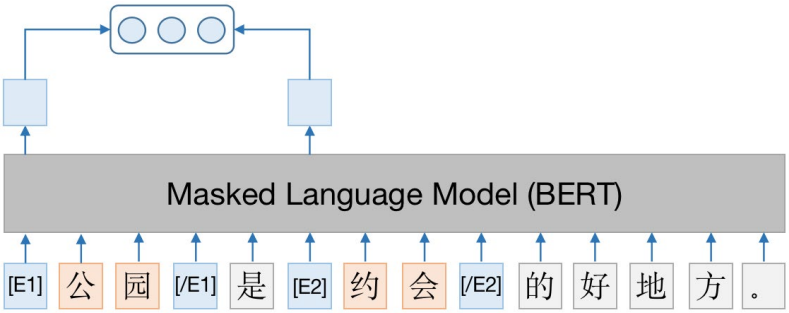


图 16　概念承接关系判别模型

3.4.3 基于已有的图谱结构进行关系补全

通过上述两个阶段，已经从非结构化的文本信息中构建出了一个初具规模的概念承接关系的图谱。但是由于语义模型的局限性，当前图谱中存在大量的三元组缺失。为了进一步丰富概念图谱，补全缺失的关系信息，我们应用知识图谱链接预测中的 TransE 算法以及图神经网络等技术，对已有的概念图谱进行补全。

为了充分利用已知图谱的结构信息，我们采用基于关系的图注意力神经网络（RGAT，Relational Graph Attention Network）来建模图结构信息。RGAT 利用关系注意力机制，克服了传统 GCN、GAT 无法建模边类型的缺陷，更适用于建模概念图谱此类异构网络。在利用 RGAT 得到实体稠密嵌入之后，我们使用 TransE 作为损失函数。TransE 将三元组（h,r,t）中的 r 视为从 h 到 t 的翻译向量，并约定 h+r ≈ t。该方法被广泛适用于知识图谱补全任务当中，显示出极强的鲁棒性和可拓展性。

具体细节如下图所示，RGAT 中每层结点的特征由邻居结点特征的均值以及邻边特征的均值加权拼接而成，通过关系注意力机制，不同的结点和边具有不同的权重系数。在得到最后一层的结点和边特征后，我们利用 TransE 作为训练目标，对训练集中的每对三元组（h,r,t），最小化 ||h+r=t||。在预测时，对于每个头实体和每种关系，图谱所有结点作为候选尾实体与其计算距离，得到最终的尾实体。

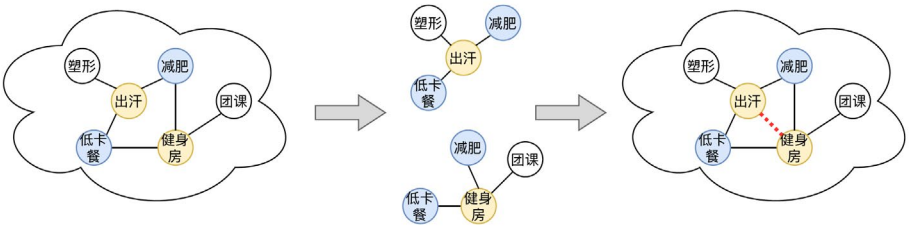


图 17　概念承接关系补足图示

目前概念承接关系整体准确率 90% 左右。

3.5 POI/SPU– 概念关系建设

建立图谱概念和美团实例之间的关联，会利用到 POI/SPU 名称、类目、用户评论等多个维度的信息。建立关联的难点在于如何从多样化的信息中获取与图谱概念相关的信息。因此，我们通过同义词召回实例下所有与概念语义相关的子句，然后利用判别模型判断概念与子句的关联程度。具体流程如下：

- **同义词聚类**。对于待打标的概念，根据图谱同义词数据，获取概念的多种表述。
- **候选子句生成**。根据同义词聚类的结果，从商户名称、团单名称、用户评论等多个来源中召回候选子句。
- **判别模型**。利用概念 – 文本关联判别模型（如下图所示）判断概念和子句是否匹配。

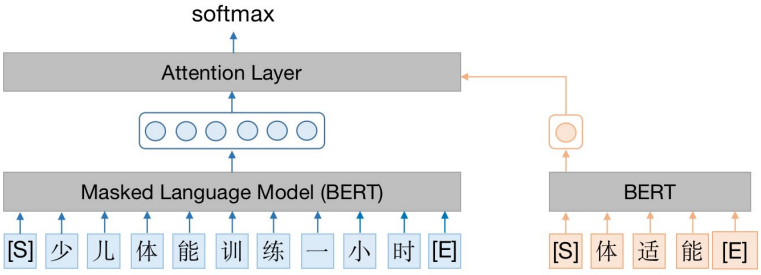


图 18　概念打标判别模型

- **打标结果**。调整阈值，得到最终的判别结果。

四、应用实践

4.1 到综品类词图谱建设

美团到综业务涵盖知识领域较广，包含亲子、教育、医美、休闲娱乐等，同时每个领域都包含更多小的子领域，所以针对不同的领域建设领域内的知识图谱，能够辅助做好搜索召回、筛选、推荐等业务。

在常识性概念图谱中除了常识性概念数据，同时也包含美团场景数据，以及基础算法能力的沉淀，因此可以借助常识性图谱能力，帮助建设到综品类词的图谱数据。

借助常识性图谱，补充欠缺的品类词数据，构建合理的品类词图谱，帮助通过搜索改写，POI 打标等方式提升搜索召回。目前在教育领域，图谱规模从起初的 1000+ 节点扩展到 2000+，同时同义词从千级别扩展到 2 万 +，取得了不错的效果。

品类词图谱建设流程如下图所示：

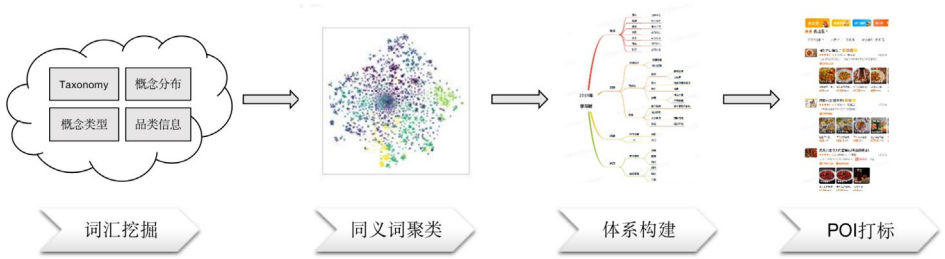


图 19　到综品类词图谱建设流程

4.2 点评搜索引导

点评搜索 SUG 推荐，在引导用户认知的同时帮助减少用户完成搜索的时间，提升搜索效率。所以在 SUG 推荐上需要聚焦两个方面的目标：①帮助丰富用户的认知，从对点评的 POI、类目搜索增加自然文本搜索的认知；②精细化用户搜索需求，当用户在搜索一些比较泛的品类词时，帮助细化用户的搜索需求。

在常识性概念图谱中，建立了很丰富的概念以及对应属性及其属性值的关系，通过一

个相对比较泛的 Query，可以生成对应细化的 Query。例如蛋糕，可以通过口味这个属性，产出草莓蛋糕、芝士蛋糕，通过规格这个属性，产出 6 寸蛋糕、袖珍蛋糕等等。

搜索引导词 Query 产出示例如下图所示：

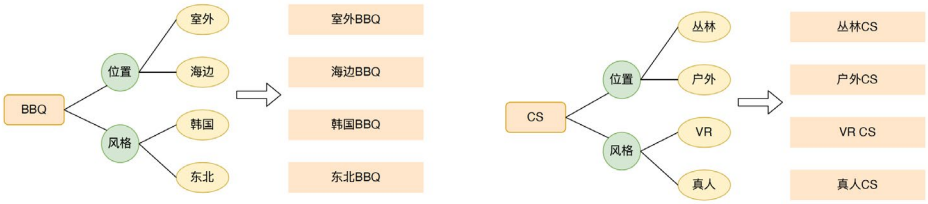


图 20　推荐 Query 挖掘示例

4.3 到综医美内容打标

在医美内容展示上，用户通常会对某一特定的医美服务内容感兴趣，所以在产品形态上会提供一些不同的服务标签，帮助用户筛选精确的医美内容，精准触达用户需求。但是在标签和医美内容进行关联时，关联错误较多，用户筛选后经常看到不符合自己需求的内容。提升打标的准确率能够帮助用户更聚焦自己的需求。

借助图谱的概念 –POI 打标能力和概念 –UGC 的打标关系，提升标签 – 内容的准确率。通过图谱能力打标，在准确率和召回率上均有明显提升。

- **准确率**：通过概念 – 内容打标算法，相比于关键词匹配，准确率从 51% 提升到 91%。
- **召回率**：通过概念同义挖掘，召回率从 77% 提升到 91%。

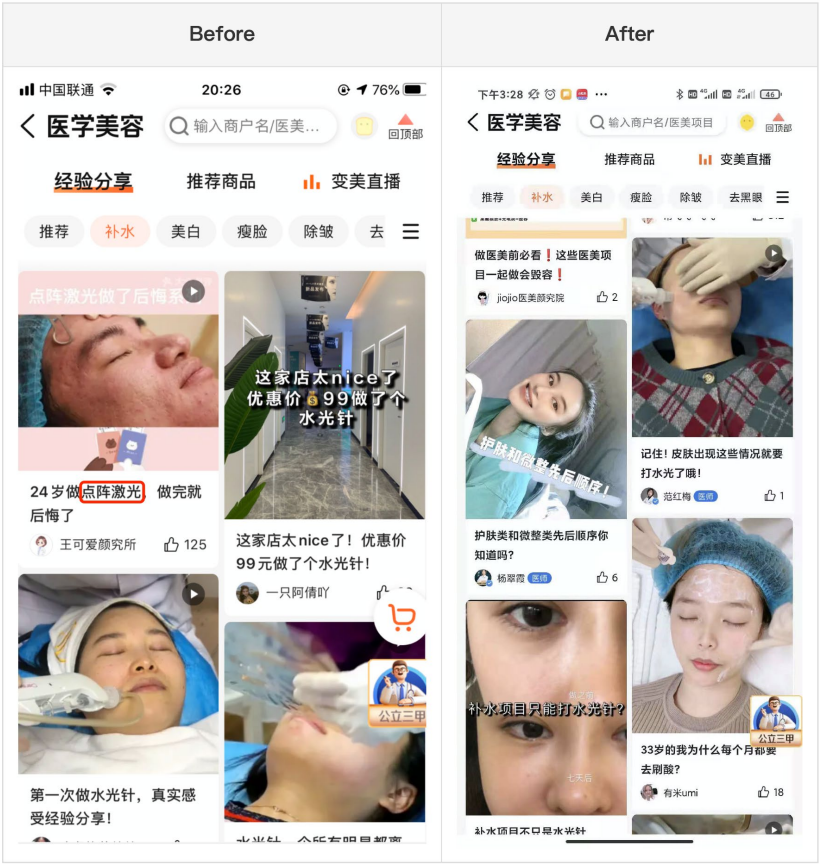


图 21　医美内容打标效果示例

五、总结与展望

我们对常识性概念图谱建设工作以及在美团场景中的使用情况进行了详细的介绍。在整个常识性概念图谱中，按照业务需要包含三类节点和四类的关系，分别介绍了概念挖掘算法、不同种类的关系挖掘算法。

目前，我们常识性概念图谱有 200 万 + 的概念，300 万 + 的概念之间的关系，包含上下位、同义、属性、承接等关系，POI- 概念的关系不包含在内。目前，整体关系准确率在 90% 左右，并且还在不断优化算法，扩充关系的同时提升准确率。后续我们的常识性概念图谱还会继续完善，希望能够做到精而全。

参考资料

[1] Onoe Y, Durrett G. Interpretable entity representations through large-scale typing[J]. arXiv preprint arXiv:2005.00147, 2020.

[2] Bosselut A, Rashkin H, Sap M, et al. Comet: Commonsense transformers for automatic knowledge graph construction[J]. arXiv preprint arXiv:1906.05317, 2019.

[3] Soares L B, FitzGerald N, Ling J, et al. Matching the blanks: Distributional similarity for relation learning[J]. arXiv preprint arXiv:1906.03158, 2019.

[4] Peng H, Gao T, Han X, et al. Learning from context or names? an empirical study on neural relation extraction[J]. arXiv preprint arXiv:2010.01923, 2020.

[5] Jiang, Zhengbao, et al. "How can we know what language models know?." Transactions of the Association for Computational Linguistics 8 (2020): 423-438.

[6] Li X L, Liang P. Prefix-Tuning: Optimizing Continuous Prompts for Generation[J]. arXiv preprint arXiv:2101.00190, 2021.

[7] Malaviya, Chaitanya, et al. "Commonsense knowledge base completion with structural and semantic context." Proceedings of the AAAI Conference on Artificial Intelligence. Vol. 34. No. 03. 2020.

[8] 李涵昱 , 钱力 , 周鹏飞 . "面向商品评论文本的情感分析与挖掘 ." 情报科学 35.1 (2017): 51-55.

[9] 闫波 , 张也 , 宿红毅 等 . 一种基于用户评论的商品属性聚类方法 .

[10] Wang, Chengyu, Xiaofeng He, and Aoying Zhou. "Open relation extraction for chinese noun phrases." IEEE Transactions on Knowledge and Data Engineering (2019).

[11] Li, Feng-Lin, et al. "AliMeKG: Domain Knowledge Graph Construction and Application in E-commerce." Proceedings of the 29th ACM International Conference on Information & Knowledge Management. 2020.

[12] Yang, Yaosheng, et al. "Distantly supervised ner with partial annotation learning and reinforcement learning." Proceedings of the 27th International Conference on Computational Linguistics. 2018.

[13] Luo X, Liu L, Yang Y, et al. AliCoCo: Alibaba e-commerce cognitive concept net[C]//Proceedings of the 2020 ACM SIGMOD International Conference on Management of Data. 2020: 313-327.

[14] Devlin J, Chang M W, Lee K, et al. Bert: Pre-training of deep bidirectional transformers for language understanding[J]. arXiv preprint arXiv:1810.04805, 2018.

[15] Cheng H T, Koc L, Harmsen J, et al. Wide & deep learning for recommender systems[C]//Proceedings of the 1st workshop on deep learning for recommender systems. 2016: 7-10.

[16] Liu J, Shang J, Wang C, et al. Mining quality phrases from massive text corpora[C]//Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data. 2015: 1729-1744.

[17] Shen J, Wu Z, Lei D, et al. Hiexpan: Task-guided taxonomy construction by hierarchical tree expansion[C]//Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. 2018: 2180-2189.

[18] Huang J, Xie Y, Meng Y, et al. Corel: Seed-guided topical taxonomy construction by concept learning and relation transferring[C]//Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. 2020: 1928-1936.
[19] Liu B, Guo W, Niu D, et al. A user-centered concept mining system for query and document understanding at tencent[C]//Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. 2019: 1831-1841.
[20] Choi E, Levy O, Choi Y, et al. Ultra-fine entity typing[J]. arXiv preprint arXiv:1807.04905, 2018.
[21] Xie Q, Dai Z, Hovy E, et al. Unsupervised data augmentation for consistency training[J]. arXiv preprint arXiv:1904.12848, 2019.
[22] Mao X, Wang W, Xu H, et al. Relational Reflection Entity Alignment[C]// Proceedings of the 29th ACM International Conference on Information & Knowledge Management. 2020: 1095-1104.
[23] Chen J, Qiu X, Liu P, et al. Meta multi-task learning for sequence modeling[C]// Proceedings of the AAAI Conference on Artificial Intelligence. 2018, 32(1).

作者简介

宗宇、俊杰、慧敏、福宝、徐俊、谢睿、武威等，均来自美团搜索与 NLP 部 -NLP 中心。

招聘信息

美团搜索与 NLP 部 /NLP 中心是负责美团人工智能技术研发的核心团队，使命是打造世界一流的自然语言处理核心技术和服务能力，依托 NLP（自然语言处理）、Deep Learning（深度学习）、Knowledge Graph（知识图谱）等技术，处理美团海量文本数据，为美团各项业务提供智能的文本语义理解服务。

NLP 中心长期招聘自然语言处理算法专家 / 机器学习算法专家，感兴趣的同学可以将简历发送至 wangzongyu02@meituan.com。

美团外卖美食知识图谱的迭代及应用

作者：杨林　郭同　海超　懋地

0. 写在前面

菜品是外卖交易过程的核心要素，对菜品的理解也是实现外卖供需匹配的重点。本期推送，我们将通过三篇文章系统地介绍了美团外卖美食知识图谱的构建和应用。第一篇文章会介绍外卖知识图谱的体系全貌，包括菜品类目、标准菜品、美食基础属性（食材、口味等）和美食业务主题属性。同时外卖的菜品属于非标品。第二篇，我们将重点介绍外卖菜品标准化建设思路、技术方案和业务应用。由于外卖的业务特点是搭配成单，所以在第三篇我们会针对性地介绍外卖套餐搭配技术的迭代以及应用实践。希望对从事相关工作的同学能够带来一些启发或者帮助。

本文系外卖美食知识图谱系列的第一篇文章，这篇文章系统地介绍了美团外卖美食知识图谱的标签体系结构，包括菜品类目标签、标准菜品名、美食基础属性（食材、口味、菜系等）和美食业务主题属性（商家招牌、类目经典等）。在技术层面，举例对标签体系的具体构建方法进行介绍，例如基于 BERT 预训练的分类模型。在应用方面，介绍了美食知识图谱在美团外卖业务的具体应用，包括支撑套餐搭配的菜品表征、提升搜索和商家推荐等业务的用户体验。

1. 背景

知识图谱，旨在描述真实世界中存在的各种实体和实体之间的关系。在美团外卖业务中，美食商品是美团向用户提供服务的基础，美食知识图谱的建设，可以帮助我们向用户提供更加准确、更加丰富、更加个性化的美食服务。另外，美团外卖业务向用户提供“到家”吃饭的服务，到店餐饮业务则向用户提供“到店”吃饭的服务，而外卖和到店的商家和菜品有相当程度的重合，菜品数据的对齐，为我们进行线上（外卖场景）线下（到店场景）数据的对比分析也提供了一个很好的“抓手”。

本文介绍了外卖美食知识图谱的建设，基于对外卖业务数据（外卖交易数据、商家录入的商品标签信息、专业描述 PGC、用户评论 UGC、商品图片等）和站外数据（百科、菜谱等）的挖掘和分析，形成了针对外卖美食的分类体系（美食类目标签）和标准化体系（标准菜品名标签），并进一步针对不同类型的美食商品，构建包含口味、食材等众多美食基础属性体系。同时，依托美团外卖的业务特性，构建美食商品在外卖业务中涉及的主题属性体系，例如商家招牌、商家主营、类目经典等。目前，外卖美食知识图谱的标签结构如下图 1 所示：

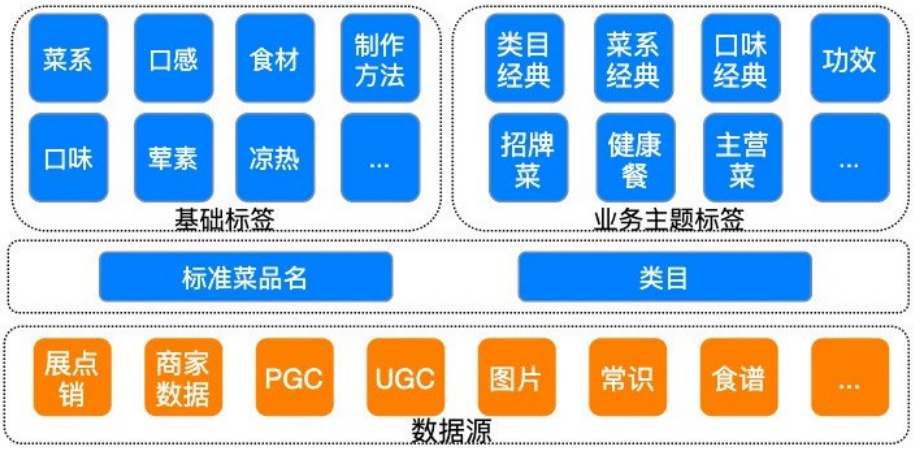


图 1　美食知识图谱标签体系

外卖美食知识图谱包含以下四种维度的标签（以“宫保鸡丁”为例，如下图 2 所示）：

- **类目标签**：包括主食、小吃、菜品等类目，并在每个类目下，形成了层级化的三百多种细分类目。例如“宫保鸡丁”的类目是“菜品”。类目标签是美食商品的基础分类信息，根据类目的不同，美食商品的基础属性也不同。例如“菜品”类目存在“荤素”、“菜系”之分，而“酒水饮品”类目则没有这种属性标签。
- **标准菜品名标签**：标准菜品名标签主要为标准商品信息，例如“宫保鸡丁（招牌必点）”的标准商品是“宫保鸡丁”。因商家输入商品的多样性，标准菜标签的建设，实现了相同美食的聚合。
- **基础属性**：根据美食商品的类目不同，构建包括美食的食材、菜系、口味、制

作方法、荤素等基础属性。例如“宫保鸡丁”的菜系是“川菜”，食材有“鸡胸”和“花生”，荤素标签是“荤”。基础属性的挖掘对我们理解商品起到关键作用，在商品的筛选、展示、商品表征等业务需求方面，提供基本的数据特征。

- **主题属性**：主题属性主要体现美食的业务主题，包括美食在外卖的交易行为、美食在商家的定位、美食在用户反馈中的好评度等。例如某商家的“宫保鸡丁（招牌必点）”是该商家的“招牌菜”。

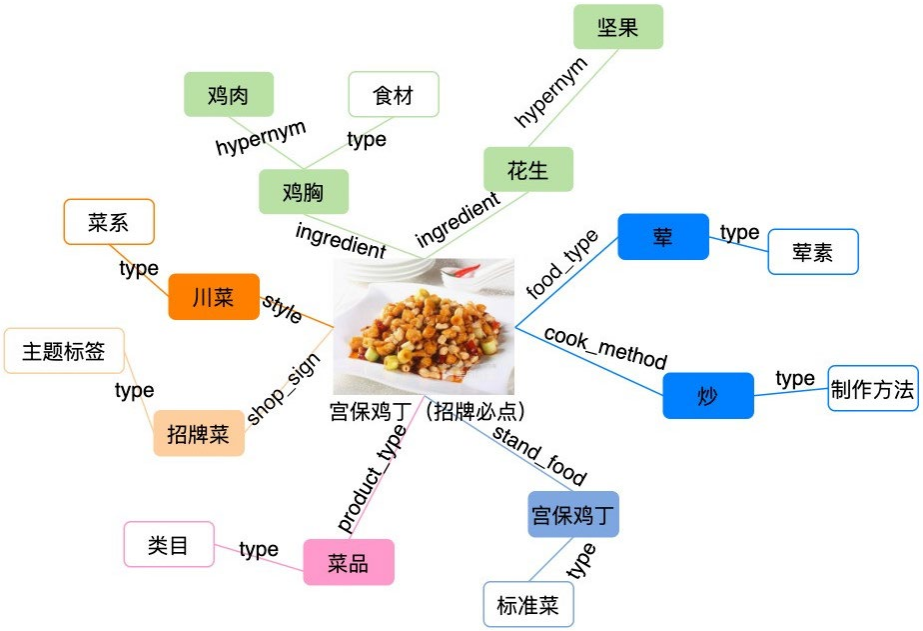


图 2　外卖美食知识图谱样例

菜品对齐，涉及到菜品数据，来自外卖在线菜品、点评推荐菜品、美团商家套餐等。

2. 需求及挑战

目前，外卖美食知识图谱已经应用于美团外卖的多个场景，例如推荐、搜索、套餐搭配、运营分析等。业务的深入发展，对美食知识图谱的建设和迭代也提出了更加复杂的要求，例如：

- 美食商品越来越多样，相应的美食知识图谱则需要越来越精细和准确。例如美

食知识图谱的类目标签从零开始，建设了包含一百多种类目的类目标签体系。但随着业务发展，部分类目存在明显的可细化空间。

- 图谱标签的挖掘，偏向于静态标签的挖掘，对于相同图谱标签下的美食，缺少业务相关的主题属性描述。例如同样包含“花生”的“酒鬼花生”，相比“宫保鸡丁”，更能代表“花生”相关的美食。
- 外卖美食知识图谱主要描述外卖美食商品，而同一商家的美食商品，也可能会出现在该店的线下收银等业务中。通过对齐不同业务的美食商品，可以在美食实体层面，完善美食知识图谱对商家美食的描述，从而指导商品和商家运营。

为满足业务需求，我们对类目标签和基础属性进行了迭代和优化；同时，构建了业务相关的主题属性。另外，我们将外卖菜品和到餐菜品进行了实体对齐。其中，挖掘主题属性，即挖掘业务相关的图谱知识，是一个需要综合考虑外卖业务和商品本身属性的复杂过程。外卖菜品和到餐菜品的对齐，则需要综合考虑菜品多样性表述和菜品主体归一。

外卖美食知识图谱的迭代难点主要体现在以下几点：

- 业务相关的主题属性挖掘，并没有现成的体系可以参考，在构建过程中，涉及大量的分析和体系设计工作。
- 主题属性的挖掘，最重要的是需要从用户的需求出发，分析用户对商品的需求点，并将其反映在商品的图谱层面，形成相应的主题属性标签。同时，商家的商品信息是一个动态变化的过程，例如销量、供给、商品标签等，前后两天的信息可能就会完全不同。因此业务性主题属性的挖掘，一方面需要建设相对完善的体系，另一方面也需要适配业务数据的动态变化过程，也就在图谱挖掘和需求匹配上带来了极大的挑战。
- 商家录入菜品时，对菜品存在多样性表述，例如同一道菜在分量、口味、食材等方面存在的差异。菜品对齐时，则需要对这些多样性表述进行平衡，例如是否忽略分量因素等。但目前并没有现成的对齐标准可以参考。

3. 外卖美食知识图谱的迭代

因篇幅受限，本文主要对其中菜品类目，不同类目、口味、食材、荤素、做法下的经典美食，健康餐等图谱标签的挖掘进行介绍。其中，在图谱标签挖掘中涉及到的数据来源和采用的技术，大致如下表所示：

标签	技术
菜品类目	BERT 分类模型
不同类目、口味、食材、荤素、做法下的经典美食	数据统计、实体识别、关系识别、产品定义（综合考虑销量和供给量）
健康餐	分类模型 + 产品定义（符合一定食材、做法、功效的商品）

3.1 菜品类目

菜品类目标签的挖掘，主要解决美食菜品是什么类别的问题。实现这一目标的挑战有两方面：首先是类目体系如何建立，其次是如何将商品链接到相应的类目节点。在最开始的体系构建时，我们从美食商品的特点以及业务的具体需求出发，从零开始建立起包含一百多种类别的层次化类别体系，部分实例如图 3（左）所示。同时，构建基于 CNN+CRF 的分类模型，对美食商品进行类目分类，如图 4（左）所示。

然而，随着业务的发展，已有分类目已经无法支持现有业务的需求。例如：原先的类目体系，对热菜描述不够详细，譬如没有区分热菜的做法等。为此，我们与外卖的供给规划部合作，将类目体系扩充到细分的三百多种类目标签，划分更加详细，覆盖也更加全面，部分实例如图 3（右）所示。

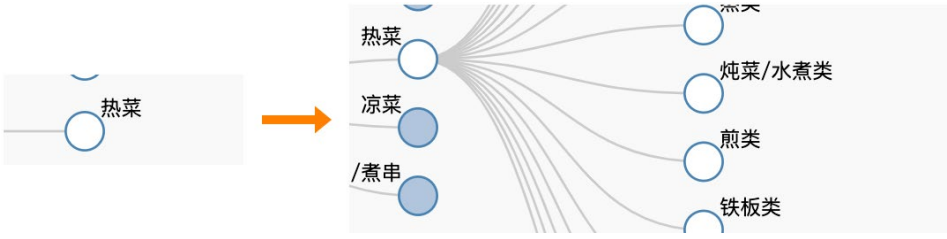


图 3　类目体系的迭代

类目的细分，要求模型更加精确。在进行类目识别时，可用的数据包括菜品名、商

家店内侧边栏分类名称、商家名等。考虑到可使用的信息大多为文本信息，并且，商家录入的文本并没有一定的规范，菜品名也多种多样，为提高模型精度，我们将原先的 CNN+CRF 的分类模型进行了升级，采用模型容量更大的 BERT 预训练+Fine-Tuning 的模型。模型结构如下图 4（右）所示。

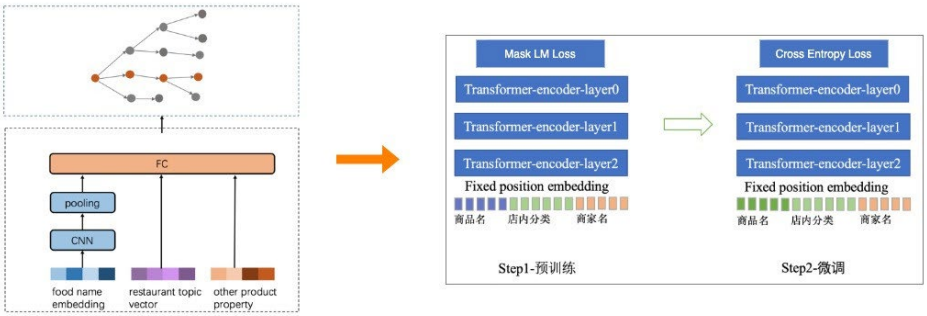


图 4　类目模型迭代

3.2 不同类目、口味、食材、荤素、做法下的经典美食标签

我们在建设主题属性时，首先在基础属性标签维度，综合考虑商品的销量和供给情况，对菜品进行选优。例如类目下的经典美食等。但在建设过程中，我们发现菜系的经典美食识别，假如依据销量和供给进行识别，则识别结果倾向于菜系下的“家常菜”，因此将经典菜系美食进行单独识别。

类目经典美食等指的是销量较高、供给量丰富的类目美食商品，例如主食经典美食、小吃经典美食。口味、食材、做法经典美食标签等也是相似的定义。

在建设过程中，我们发现，假如直接在商品维度进行识别，因为商品的更新频率相对较高，对新录入的暂时没有销量或者暂时销量低的美食商品不友好，销量水平需要考虑在线时间的影响。因此我们使用标准菜品进行类目、口味经典等的识别，并通过标准菜品，泛化到具体的美食商品上。

其中，“标准菜品”借用其它类电商业务中的“标品”概念，虽然绝大部分菜品的生产都不是标准化的过程，但是这里我们只关注主要的共性部分，忽略次要的差异部

分。例如“西红柿鸡蛋”、“番茄炒蛋”都是同一类菜品。从结果上看，目前我们聚合出来的“标准菜品”达到几十万的量级，并且能够覆盖大部分美食商品。

借助标准菜品，我们将类目、口味、食材、荤素、做法等标签聚合到标准菜品维度，并将销量、供给量进行标准菜品维度计算，这样就解决了商品在线时间长短的问题。在具体打标过程中，例如类目经典，我们基于销量和供给，在类目维度对标准菜进行排序，并选择 Top n% 标准菜进行打标，作为类目经典下的商品。例如在“面食”类目下，“西红柿鸡蛋面”的销量和供给量均在 Top n% 的水平，因此就认为“西红柿鸡蛋面”是一个面食类经典美食。

3.3 健康餐

这里的健康餐主要指低脂低卡餐，即低卡路里、低脂肪、高纤维、制作简单、原汁原味、健康营养的食物，一般为蔬菜水果（如罗勒、甘蓝、秋葵、牛油果等），富含优质蛋白的肉类（如三文鱼、虾、贝类、鸡胸等），谷物（主要以粗粮为主，如燕麦、高粱、藜麦等）。烹饪方法也坚持“少油，少盐，少糖”的原则，主要做法为蒸、煮、少煎、凉拌等。

健康餐的识别，主要挑战在于本身的样本较少，但是因为健康餐的特殊性，商家在进行商品录入时，一般会对其进行描述，例如指出这个美食商品是“健康”的、“低卡”的、“健身”类型的，因此我们构建了一个分类模型，对健康餐进行识别。可使用的数据，包括商品名、商家导航栏、商家名称、商家对商品的描述等。而商家类目与商品的类目处于迭代状态，因此并没有对这部分信息进行使用。

识别过程如下：

- **训练数据构建**：因健康餐本身的占比相对较少，因此首先总结和健康餐相关的关键词，使用关键词进行文本匹配，采样概率相对较高的健康餐数据，进行外包数据标注。此处，我们总结出“沙拉、谷物饭、谷物碗、低油、低卡、无糖、减脂、减肥、轻食、轻卡”等关键词。
- **模型构建**：同一个商品因其中使用的配料不同，在健康餐识别方面也会不同，

例如菜名为“招牌沙拉”的商品，假如沙拉中添加了芝士，则有可能商品就不会被识别成健康餐。为了综合考虑商家录入的商品信息，使用商品名、商家名、导航栏名称、商家录入的商品描述等。这四种数据为不同尺度的数据源，商品名等为相对较短的文本，因此在模型构建时，考虑使用类似 Text-CNN[1] 的结构进行字级别的特征提取；商品描述则是相对较长的文本，因此在构建时，考虑使用类似 Transformer[2] 的结构进行特征提取，使用 Multi-head Attention 的机制，提取长文本中，“字”层面的特征。具体结构如下：

- 采用了两种结构：Multihead-attention（Transformer）和 Text-CNN。实验发现，采用两种结构联合的方式，比采用单一结构准确率高。
- 在建模时，均使用字级别特征处理，避免因为分词造成的误差，同时也避免未登录词的影响。

- **数据迭代增强**：因为使用关键词进行样本构建，在模型训练时，模型会朝着包含这些关键词的方向学习，因此存在漏召回的情况。在这里，我们进行了一定的训练数据增强，例如在评估时，选取可识别出健康餐的商家，对该商家中漏召回的数据进行训练数据补充；同时，对部分特征明显的关键词，进行补充并扩充正例。通过对训练样本的多次扩充，最终完成健康餐的高准确率识别。

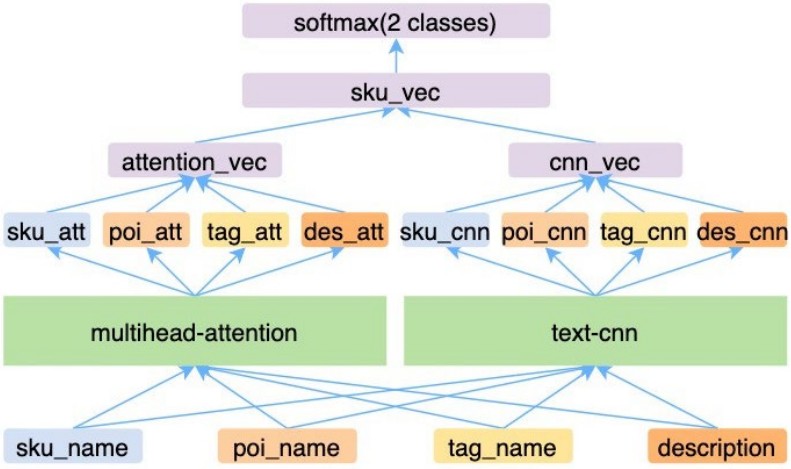


图 5　健康餐识别模型

3.4 菜品实体对齐

考虑到同一商家菜品在不同业务线的菜品名可能略有差异，我们设计了一套菜品名匹配的算法，通过拆解菜品名称的量词、拼音、前后缀、子字符串、顺序等特征，利用美食类目识别、标准菜品名抽取、同义关系匹配等进行菜品实体对齐。例如：碳烧鸽 = 炭烧鸽、重庆辣子鸡 = 重庆歌乐山辣子鸡、茄子肉泥盖饭 = 茄子肉泥盖浇饭、番茄炒蛋 = 西红柿炒蛋等。目前，形成如下图的菜品归一体系：

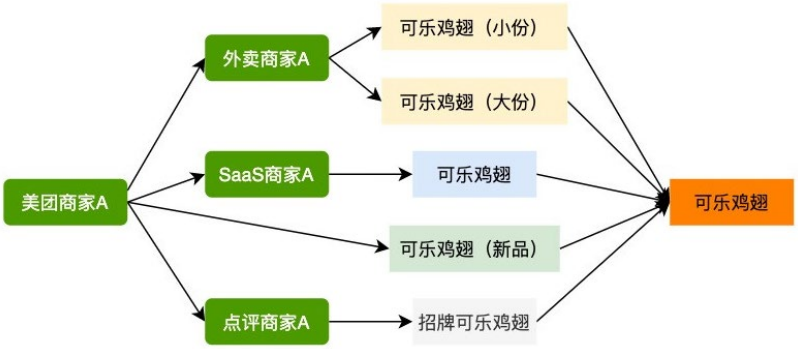


图 6　菜品归一体系

4. 应用

这里对外卖美食知识图谱的应用，进行举例说明。主要涉及套餐搭配、美食商品展示等。

4.1 套餐搭配 – 表征菜品

为满足用户的搭配成单需求，进行套餐搭配技术的探索。套餐搭配技术的关键在于，对美食商品的认知，而外卖美食知识图谱，则提供了最全面的数据基础。我们基于同商家内的商品信息和历史成单信息，对商品的搭配关系进行拟合，参考指针网络 [2] 等结构，构建了基于 Multi-Head Attention[3] 的 Enc-Dec 模型，具体的模型结构如下：

- **Encoder**：对商家菜单进行建模，因菜单为无序数据，因此采用 Attention 的方式进行建模。商品的信息主要包括商品名、商品图谱标签、交易统计数据等

三部分。

- 对菜名、商品标签分别进行 Self-Attention 计算，得到菜名和商品标签对应的向量信息，然后与交易统计数据进行 Concat，作为商品的初步表示。
- 对商品的初步表示进行 Self-Attention 计算，以对同商家的商品有所感知。

- **Deocoder**：对搭配关系进行学习，基于当前已选择的商品，对下一个可能的搭配进行预估。
  - 在搭配输出时，使用 Beam-Search 进行多种搭配结果的输出。
  - 为了保证输出搭配中的商品的多样性，添加 Coverage 机制 [2]。
- 训练之后，将 Encoder 部分分离，进行离线调度，可实现每天的向量产出。

具体的模型结构如下图所示：

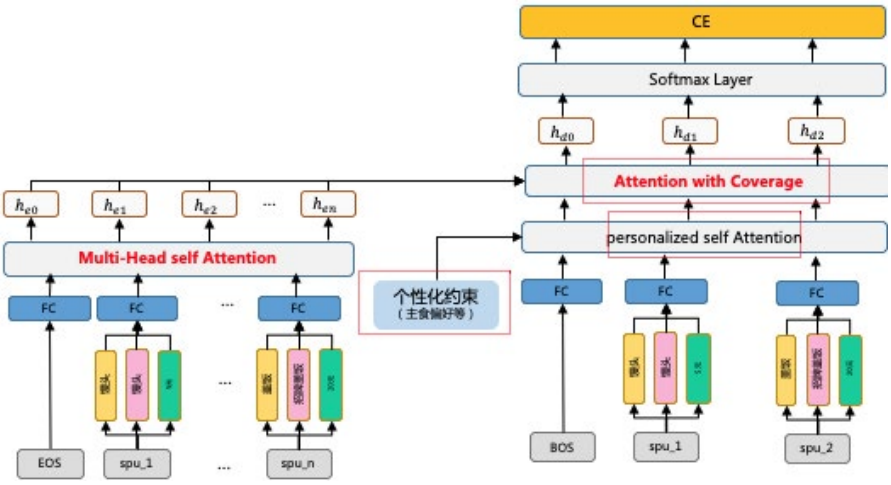


图 7　套餐搭配模型

基于外卖美食知识图谱构建的套餐搭配模型，在多个入口（“满减神器”、“对话点餐”、“菜品详情页”等）取得转化的提升。

4.2 交互式推荐

通过分析外卖用户的需求，发现用户存在跨店相似商品对比的需求，为打破商家界限的选购流程特点，提供便捷的跨店对比决策方式。交互式推荐，通过新的交互模式，

打造推荐产品的突破点。在用户的交互过程中，根据用户的历史偏好、实时的点击行为，向用户推荐可能喜欢的美食商品。如下图 8（左）所示，在向用户进行同类美食的推荐时，美食知识图谱中的标准菜品标签提供了主要的数据支撑。

4.3 搜索

搜索作为外卖核心流量入口，承载了用户明确的外卖需求。用户通过输入关键词，进行菜品检索。在实际使用中，从搜索的关键词类型看，可能是某个具体的菜品，也可能是某种食材、某种菜系。在美食知识图谱中，图谱标签的高准确率和高覆盖，有助于提升搜索入口的用户体验，最新的实验也表明了这一点（新增部分食材、菜系、功效等标签，在搜索的线上实验效果正向）。

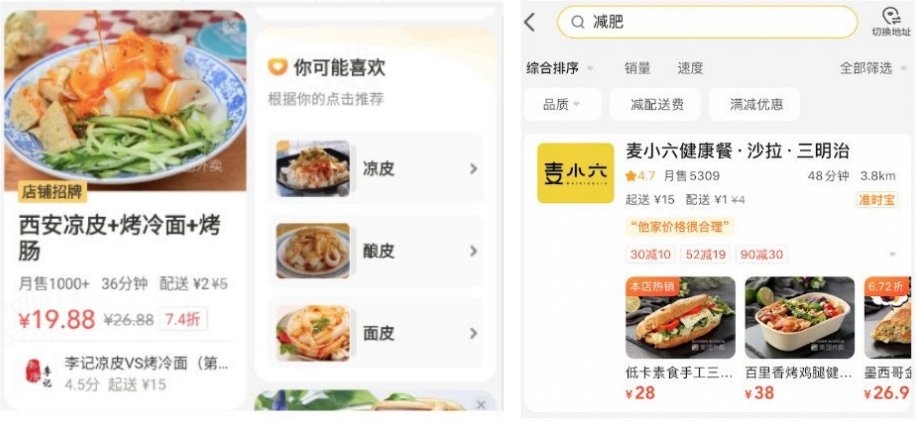


图 8　交互式推荐和搜索

5. 未来规划

5.1 场景化标签的挖掘

美食与我们的生活息息相关，美团外卖每天为千万用户提供美食方面的服务。然而，用户的需求是多种多样的，在不同的环境、不同的场景下，对美食的需求也不尽相同。目前美食知识图谱挖掘，在场景相关的标签较为缺失，例如某些节气、节日等图谱知识；特定天气情况下的图谱知识；特定人群（增肌人群、减肥人群）等的图谱知

识。接下来我们会在场景化标签的挖掘方面进行探索。

在挖掘方法方面，目前的挖掘数据主要为文本信息。在商品图片、描述、结构化标签等信息的融合方面，挖掘不够深入，模型的效果也有待提升。因此在多模态识别模型方面，我们也会进行相应的探索。

5.2 基于图谱的推荐技术研究

美团外卖在理解美食的基础上，向用户进行美食推荐，以更好地满足用户对美食的需求。外卖美食知识图谱和外卖业务数据，作为实现这一点的数据基础，包含上亿的节点信息和十几亿的关系数据。通过对用户的商品搜索、点击、购买等行为进行建模分析，可以更加贴合用户的需求，向用户进行商品推荐，例如，将美食知识图谱和外卖行为数据融合，以用户为起点，进行随机游走，向用户推荐相关的美食。在接下来的图谱应用方面的探索中，我们也会更加深入的探索基于美食知识图谱和用户行为的推荐技术。

6. 参考文献

[1] Kim Y. Convolutional neural networks for sentence classification[J]. arXiv preprint arXiv:1408.5882, 2014.

[2] See A, Liu P J, Manning C D. Get to the point: Summarization with pointer-generator networks[J]. arXiv preprint arXiv:1704.04368, 2017.

[3] Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need[C]//Advances in neural information processing systems. 2017: 5998-6008.

[4] Hamilton W, Ying Z, Leskovec J. Inductive representation learning on large graphs[C]//Advances in Neural Information Processing Systems. 2017: 1024-1034.

7. 作者简介

杨林、郭同、海超、懋地等，均来自美团外卖技术团队。

美团知识图谱问答技术实践与探索

作者：如寐　梁迪　思睿　鸿志　明洋　武威

1. 背景与挑战

问答系统（Question Answering System, QA）是人工智能和自然语言处理领域中一个倍受关注并具有广泛发展前景的方向，它是信息检索系统的一种高级形式，可以用准确、简洁的自然语言回答用户用自然语言提出的问题。这项研究兴起的主要原因是人们对快速、准确地获取信息的需求，因此被广泛应用于工业界的各种业务场景中。美团在平台服务的售前、售中、售后全链路的多个场景中，用户都有大量的问题需要咨询商家。因此我们基于问答系统，以自动智能回复或推荐回复的方式，来帮助商家提升回答用户问题的效率，更快地解决用户的问题。

针对不同问题，美团的智能问答系统包含多路解决方案：

- **PairQA**：采用信息检索技术，从社区已有回答的问题中返回与当前问题最接近的问题答案。
- **DocQA**：基于阅读理解技术，从商家非结构化信息、用户评论中抽取出答案片段。
- **KBQA（Knowledge-based Question Answering）**：基于知识图谱问答技术，从商家、商品的结构化信息中对答案进行推理。

本文主要分享在 KBQA 技术落地中的实践与探索。

在用户的问题中，包括着大量关于商品、商家、景区、酒店等相关基础信息及政策等信息咨询，基于 KBQA 技术能有效地利用商品、商家详情页中的信息，来解决此类信息咨询问题。用户输入问题后，KBQA 系统基于机器学习算法对用户查询的问题进行解析、理解，并对知识库中的结构化信息进行查询、推理，最终将查询到的精确答案返回给用户。相比于 PairQA 和 DocQA，KBQA 的答案来源大多是商家数据，可

信度更高。同时，它可以进行多跳查询、约束过滤，更好地处理线上的复杂问题。

实际落地应用时，KBQA 系统面临着多方面的挑战，例如：

- **繁多的业务场景**：美团平台业务场景众多，包涵了酒店、旅游、美食以及十多类生活服务业务，而不同场景中的用户意图都存在着差别，比如“早餐大概多少钱”，对于美食类商家需要回答人均价格，而对于酒店类商家则需要回答酒店内餐厅的价格明细。
- **带约束问题**：用户的问题中通常带有众多条件，例如“故宫学生有优惠吗”，需要我们对故宫所关联的优惠政策进行筛选，而不是把所有的优惠政策都回答给用户。
- **多跳问题**：用户的问题涉及到知识图谱中多个节点组成的路径，例如“XX 酒店的游泳池几点开”，需要我们在图谱中先后找到酒店、游泳池、营业时间。

下面将详细讲述我们是如何设计高准确、低延时的 KBQA 系统，处理场景、上下文语境等信息，准确理解用户、捕捉用户意图，从而应对上述的挑战。

2. 解决方案

KBQA 系统的输入为用户 Query，输出为答案。总体架构如下图 1 所示。最上层为应用层，包括对话以及搜索等多个入口。在获取到用户 Query 后，KBQA 线上服务通过 Query 理解和召回排序模块进行结果计算，最终返回答案文本。除了在线服务之外，知识图谱的构建、存储也十分重要。用户不仅会关心商户的基本信息，也会询问观点类、设施信息类问题，比如景点好不好玩、酒店停车是否方便等。针对上述无官方供给的问题，我们构建了一套信息与观点抽取的流程，可以从商家非结构化介绍以及 UGC 评论中抽取出有价值的信息，从而提升用户咨询的满意度，我们将在下文进行详细介绍。

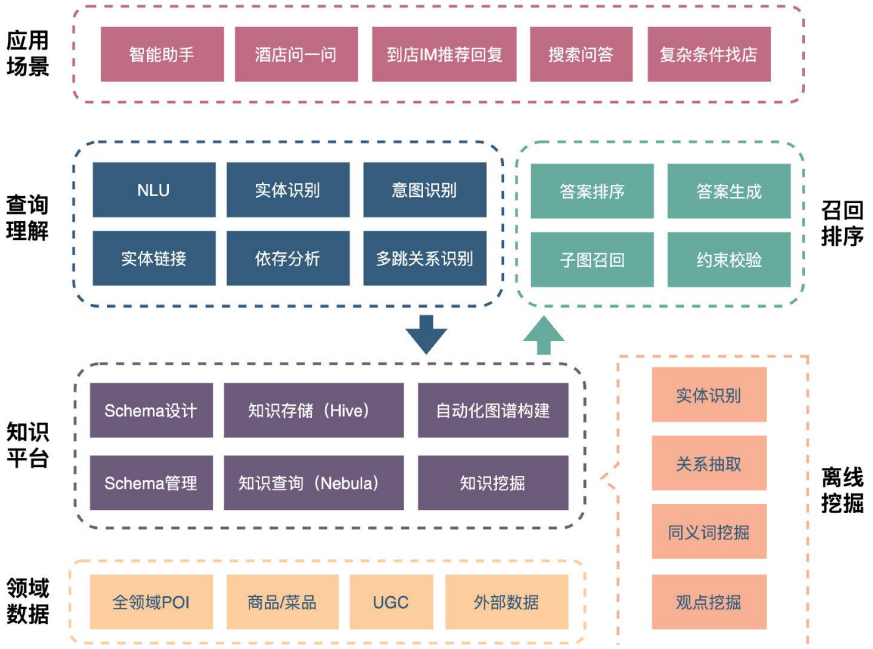


图 1　KBQA 系统架构图

对于 KBQA 模型，目前的主流解决方案有两种，如下图 2 所示：

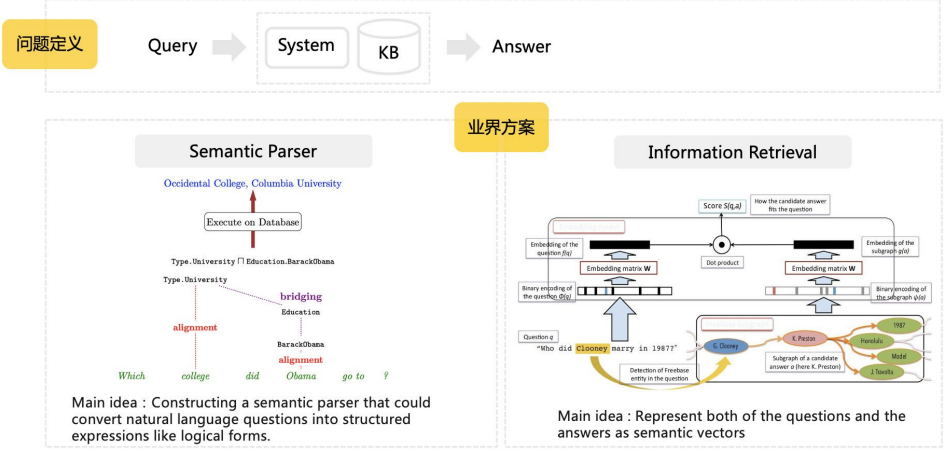


图 2　KBQA 解决方案分类

- **基于语义解析（Semantic Parsing-based）**：对问句进行深度句法解析，并

将解析结果组合成可执行的逻辑表达式（如 SparQL），直接从图数据库中查询答案。

- **基于信息抽取（Information Retrieval）**：先解析出问句的主实体，再从 KG 中查询出主实体关联的多个三元组，组成子图路径（也称多跳子图），之后分别对问句和子图路径编码、排序，返回分数最高的路径作为答案。

基于语义解析的方法可解释性更强，但这种方法需要标注大量的自然语言逻辑表达式，而信息抽取式的方法更偏向端到端的方案，在复杂问题、少样本情况下表现更好，但若子图过大，会显著降低计算的速度。

因此，考虑到两者的优势，我们采用将两者结合的方案。如下图 3 所示，整体的流程分为四大步骤，以“故宫周末有学生票吗”为例：

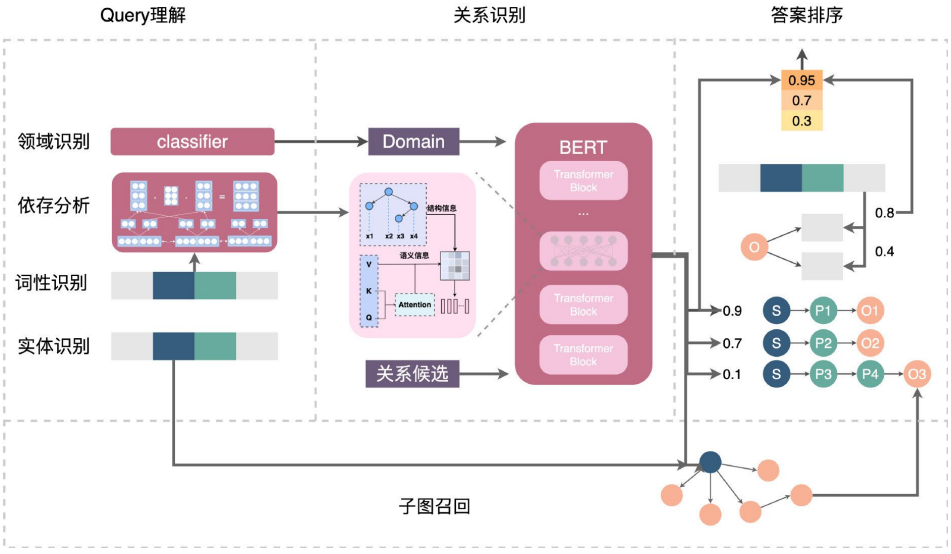


图 3　美团 KBQA 解决方案

- **Query 理解**：输入原始 Query，输出 Query 理解结果。其中会对对 Query 进行句法分析，识别出用户查询的主实体是“故宫”、业务领域为“旅游”、问题类型为一跳（One-hop）。
- **关系识别**：输入 Query、领域、句法解析结果、候选关系，输出每个候选的分

数。在这个模块中，我们借助依存分析强化 Query 的问题主干，召回旅游领域的相关关系，进行匹配排序，识别出 Query 中的关系为“门票”。

- **子图召回**：输入前两个模块中解析的主实体和关系，输出图谱中的子图（多个三元组）。对于上述例子，会召回旅游业务数据下主实体为“故宫”、关系为“门票”的所有子图。
- **答案排序**：输入 Query 和子图候选，输出子图候选的分数，如果 Top1 满足一定阈值，则输出作为答案。基于句法分析结果，识别出约束条件为“学生票”，基于此条件最终对 Query-Answer 对进行排序，输出满足的答案。

下面将介绍我们对于重点模块的建设和探索。

2.1 Query 理解

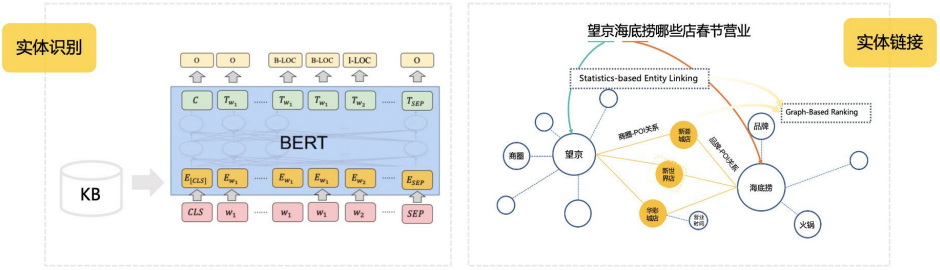
Query 理解是 KBQA 的第一个核心模块，负责对句子的各个成分进行细粒度语义理解，其中两个最重要的模块是：

- 实体识别和实体链接，输出问句中有意义的业务相关实体和类型，如商家名称、项目、设施、人群、时间等。
- 依存分析：以分词和词性识别结果为输入，识别问句的主实体、被提问信息、约束等。

实体识别是句法分析的重要步骤，我们先基于序列标注模型识别实体，再链接到数据库中的节点。对于该模块我们主要做了以下优化：

- 为了提升 OOV（Out-of-Vocabulary）词的识别能力，我们对实体识别的序列标注模型进行了知识注入，利用已知的先验知识辅助新知识的发现。
- 考虑到实体嵌套的问题，我们的实体识别模块会同时输出粗粒度和细粒度的结果，保证后续模块对于 Query 的充分理解。
- 在问答的长 Query 场景下，利用上下文信息进行实体的链接，得到节点 id。

最终，该模块会输出句子中各个重要成分的类型，如下图 4 所示：



Query	NER结果	实体链接结果
颐和园学生门票多少钱	颐和园=POI、学生=PEOPLE、门票=TICKET	颐和园=000101
望京附近环境安静的咖啡厅	望京=AREA、附近=RANGE、环境安静=TAG、安静=TAG、咖啡厅=POI	望京=0341

图 4　Query 理解流程及结果

依存分析是句法分析的一种，它的目的是识别句子中词与词的非对称支配关系，在输出的结果中用有向弧表示，该弧线由从属词（dep）指向支配词（head）。对于 KBQA 任务，我们定义了五种关系，如下图 5 所示：

关系名称	关系含义
ROOT	主实体
ATT_HEAD	主实体约束
SUB_HEAD	主实体主体
HEAD_INFO	主关系
ATT_INFO	关系约束
INFO_NONE	无意义词

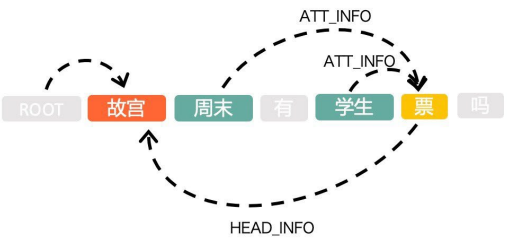


图 5　依存类型定义

依存分析主要有两种方案：基于转移的（Transition-based）和基于图的（Graph-based）。基于转移的依存分析将依存句法树的构建建模为一系列操作，由模型预测每一步的动作（shift、left_arc、right_arc），不断将未处理的节点入栈并赋予关系，最终构成句法树。基于图的方法则致力于在图中找出一棵最大生成树，也就是句子整体依存关系的全局最优解。考虑到基于图的方法是对全局进行搜索，准确率更高，我们采用较为经典的“Deep Biaffine Attention for Neural Dependency Parsing”模

型，它的结构如下图 6 所示：

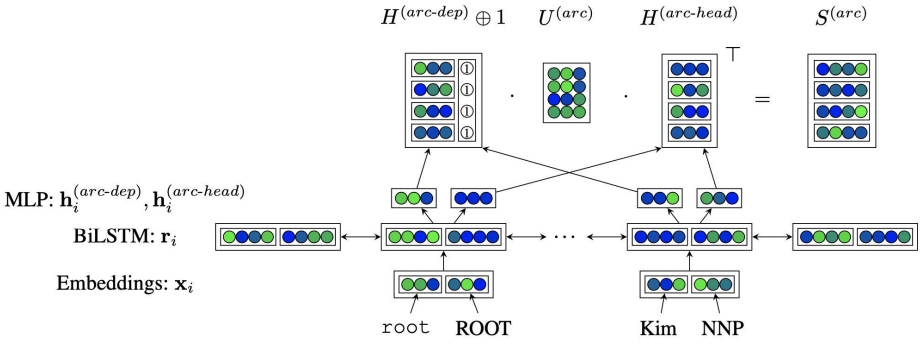


图 6　依存分析模型结构

该模型先通过 BiLSTM 对词与词性的拼接向量进行编码，之后采用对用两个 MLP 头分别编码出 h(arc-head) 和 h(arc-dep) 向量，去除冗余信息。最终将各个时刻的向量拼接起来得到 H(arc-head) 和 H(arc-dep)，且在 H(arc-dep) 上拼接了一个单位向量，加入中间矩阵 U(arc) 进行仿射变换，得到 dep 与 head 的点积分数矩阵 S(arc)，找到每个词依存的 head。

有了依存分析的结果，我们可以更好地识别关系、复杂问题，具体的特征使用方法将在下文进行介绍。

2.2 关系识别

关系识别是 KBQA 中另一个核心模块，目的是识别出用户 Query 所问的关系（Predicate），从而与主实体（Subject）联合确定唯一子图，得到答案（Object）。

在实践中，考虑到图谱中边关系的数量会不断增加，我们将关系识别建模为文本匹配任务，输入用户 Query、Query 特征和候选关系，输出关系匹配的分数。为了解决开头提到的多领域问题，我们在输入的特征中加入了领域信息，从而在领域表示中存储一定的领域相关知识，让模型更好地判断。同时，为了提升复杂 Query 的理解，我们在输入中还融入了句法信息，让模型可以更好地理解带约束、多跳的问题。

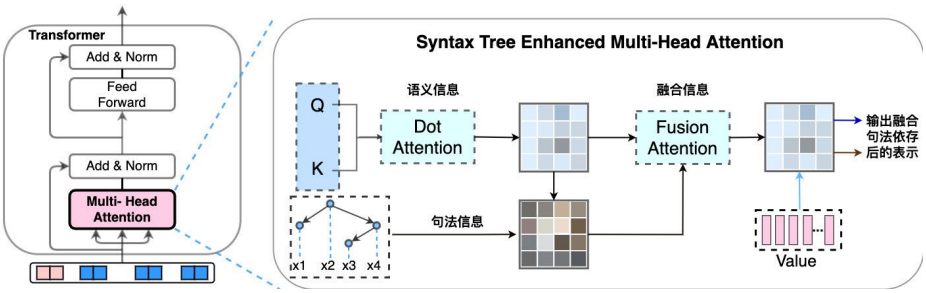


图 7　关系识别模型结构

随着大规模预训练语言模型的出现，BERT 等大模型在匹配任务上取得了 SOTA 的结果，通常业界通用的方法主要归类为以下两种：

- **表示型**：也称“双塔模型”，它的主要思想是将两段文本转换成一个语义向量，然后在向量空间计算两向量的相似度，更侧重对语义向量表示层的构建。
- **交互型**：该方法侧重于学习句子中短语之间的对齐，并学习比较他们之间的对齐关系，最终将对齐整合后的信息聚合到预测层。由于交互型模型可以利用到文本之前的对齐信息，因而精度更高、效果更好，所以在本项目中我们采用交互型模型来解决匹配问题。

为了充分利用 BERT 的语义建模能力，同时考虑实际业务的线上延时要求，我们在推理加速、数据增强、知识增强方面做了以下三点优化：

- **层次剪枝**：BERT 每层都会学到不同的知识，靠近输入侧会学到较为通用的句法知识，而靠近输出则会学习更多任务相关的知识，因此我们参考 DistillBERT，采取 Skip 等间隔式层次剪枝，只保留对任务效果最好的 3 层，比单纯保留前三层的剪枝在 F1-score 上提升了 4%，同时，实验发现不同剪枝方法效果差距可达 7%。
- **领域任务数据预精调**：剪枝后，由于训练数据有限，3 层模型的效果有不小的下降。通过对业务的了解，我们发现美团的“问大家”模块数据与线上数据的一致性很高，并对数据进行清洗，将问题标题和相关问题作为正例，随机选取字面相似度 0.5-0.8 之间的句子作为负例，生成了大量弱监督文本对，预精调

后 3 层模型在准确率上提升超过 4%，甚至超过了 12 层模型的效果。

- **知识增强**：由于用户的表达方式多种多样，准确识别用户的意图，需要深入语意并结合语法信息。为了进一步提升效果，同时解决部分 Case，我们在输入中加入了领域与句法信息，将显式的先验知识融入 BERT，在注意力机制的作用下，同时结合句法依存树结构，准确建模词与词之间的依赖关系，我们在业务数据以及五个大型公开数据集上做验证，对比 BERT Base 模型在准确率上平均提升 1.5%。

经过上述一系列迭代后，模型的速度、准确率都有了大幅的提升。

2.3 复杂问题理解

在真实场景中，大部分问题可以归为以下四类（绿色为答案节点），如下图 8 所示：

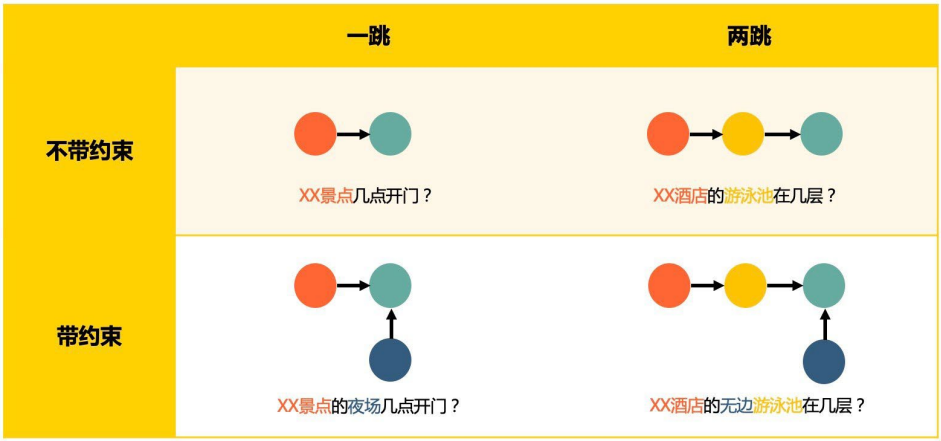


图 8　复杂问题分类

问题的跳数根据实体数量决定，单跳问题通常只涉及商户的基本信息，比如问商户的地址、电话、营业时间、政策等，在知识图谱中都可以通过一组 SPO（三元组）解答；两跳问题主要是针对商户中某些设施、服务的信息提问，比如酒店的健身房在几层、早餐几点开始、以及接送机服务的价格等，需要先找到商户 -> 主实体（设施 / 服务 / 商品等）的路径，再找到主实体的基本信息三元组，也就是 SPX、XPO 两个三元组。约束问题指主实体或答案节点上的约束条件，一般为时间、人群或是定语。

下面介绍针对不同类型的复杂问题，我们所进行的一些改进。

2.3.1 带约束问题

通过对线上日志的挖掘，我们将约束分为以下几类，如下图 9 所示：

图 9　约束问题分类

对于带约束问题的回答涉及两个关键步骤：**约束识别**和**答案排序**。

通过 KBQA 系统中的依存分析模块，我们可以识别出用户在实体或关系信息上所加的约束限制，但约束的说法较多，且不同节点的约束类型也不一样，因此我们在构造数据库查询 SQL 时先保证召回率，尽量召回实体和关系路径下的所有候选节点，并在最终排序模块对答案约束进行打分排序。

为了提升效率，我们首先在知识存储层上进行了优化。在复合属性值的存储方面，Freebase 提出 Compound Value Type (CVT) 类型，如下图 10 所示，来解决这类复合结构化的数据的存储与查询问题。比如欢乐谷的营业时间，对于不同的场次是不一样的。这种复合的属性值可以用 CVT 的形式去承接。

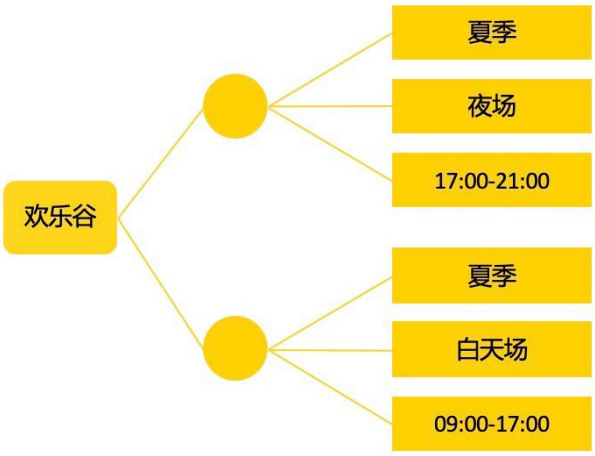


图 10　CVT 类型示例

但是，CVT 存储方式增加查询复杂度、耗费数据库存储。以图“欢乐谷营业时间 CVT”为例：

- 该信息以通常成对 CVT 形式存储，一个 CVT 涉及 3 个三元组存储。
- 对于“欢乐谷夏季夜场几点开始”这样的问题，在查询的时候，涉及四跳，分别为，< 实体 -> 营业时间 CVT>, < 营业时间 CVT -> 季节 = 夏季 >, < 营业时间 CVT -> 时段 = 夜场 >,< 营业时间 CVT -> 时间 >。对业界查询快速的图数据库比如 Nebula 来说，三跳以上的一般查询时间约为几十毫秒，在实际上线使用中耗时较长。
- 一旦属性名称、属性值有不同的但是同意的表达方式，还需要多做一步同义词合并，从而保证查询时能匹配上，没有召回损失。

为了解决上述问题，我们采用 Key-Value 的结构化形式承载属性信息。其中 Key 为答案的约束信息，如人群、时间等可以作为该属性值的约束的信息，都可以放在 Key 中，Value 即为要查的答案。对于上文的例子，我们将所有可能的约束维度的信息组成 Key，如下图 11 所示：

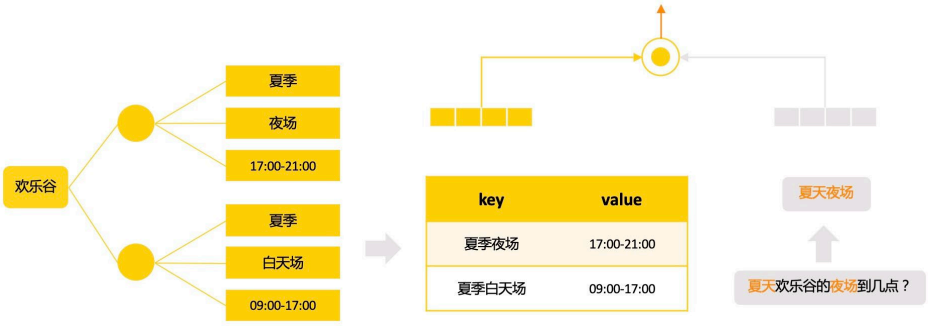


图 11　约束问题解决方案

之后，为了解决约束值说法过多的问题，在实际查询过程中，在找不到完全匹配的情况下，我们用属性值的 Key 和问题中的约束信息进行匹配计算相关度，相关度最高的 Key，对应的 Value 即为答案。因此，Key 的表示方法可以为多种形式：

- **字符串形式**：用文本相似度的方法去计算和约束文本的相关性。
- **文本 Embedding**：如对 Key 的文本形式做 Embedding 形式，与约束信息做相似计算，在训练数据合理的情况下，效果优于字符串形式。
- **其他 Embedding 算法**：如对虚拟节点做 Graph Embedding，约束文本与对应的虚拟节点做联合训练等等。

这种形式的存储方式，相当于只存储一个三元组，即 < 实体 –> 营业时间 KV>，查询过程压缩成了一跳 + 文本匹配排序。基于语义模型的文本匹配可以在一定程度上解决文本表达不同造成的不能完全匹配的问题。对语义模型进行优化后，可以尽量压缩匹配时间，达到十几毫秒。

进行复杂条件优化后，先通过前置模块识别到实体、关系和约束，组成约束文本，再与当前召回子图的 Key 值候选进行匹配，得到最终的答案。

2.3.2 多跳问题

多跳问题是天然适合 KBQA 的一类问题，当用户询问商户中的设施、服务、商品等实体的信息时，我们只需要先在图谱中找到商户，再找到商户下的实体，接着找到下面的基本信息。如果使用 FAQ 问答的解法，就需要为每个复杂问题都设置一个标准

问，比如“健身房的位置”、“游泳馆的位置”等。而在 KBQA 中，我们可以很好地对这类问题进行压缩，不管问什么实体的位置，都问的是“位置”这条边关系，只是起始实体不同。

在 KBQA 系统中，我们先依赖依存分析模块对句子成分间的依赖关系进行识别，之后再通过关系识别模块判断句子所询问的关系跳数以及关系，具体流程如下图 12 所示：

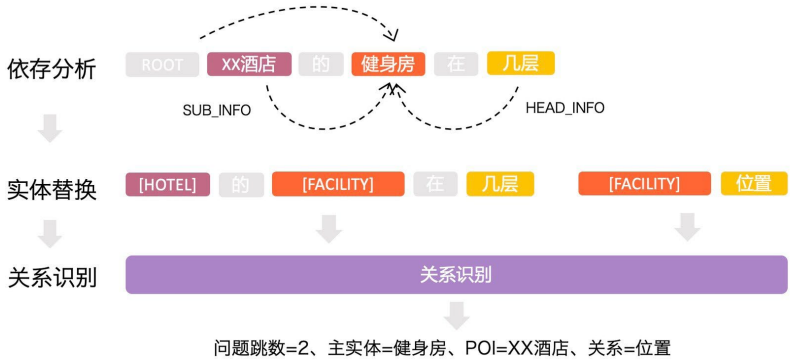


图 12　多跳问题解决方案

借助实体识别的类型，我们可以将句子中的重要成分进行替换，从而压缩候选关系配置的个数、提升关系识别准确率。在对句子进行了充分理解后，系统会基于主实体、关系、跳数对子图进行查询，并输入给答案排序模块进行更细粒度的约束识别和打分。

2.4 观点问答

除了上述基本信息类的查询 Query 外，用户还会询问观点类的问题，比如“迪士尼停车方便吗?”、“XX 酒店隔音好吗?”等。对于主观观点类问题，可以基于 FAQ 或阅读理解技术，从用户评论中找出对应的评论，但这种方法往往只能给出一条或几条评论，可能会太过主观，无法汇总群体的观点。因此我们提出了观点问答方案，给出一个观点的正反支持人数，同时考虑到可解释性，也会给出多数观点的评论证据，在 App 中的实际展示如下图 13 所示：

图 13　观点问答截图

为了自动化地批量挖掘用户观点，我们拆解了两步方案：观点发现和 Evidence 挖掘，如下图 14 所示。

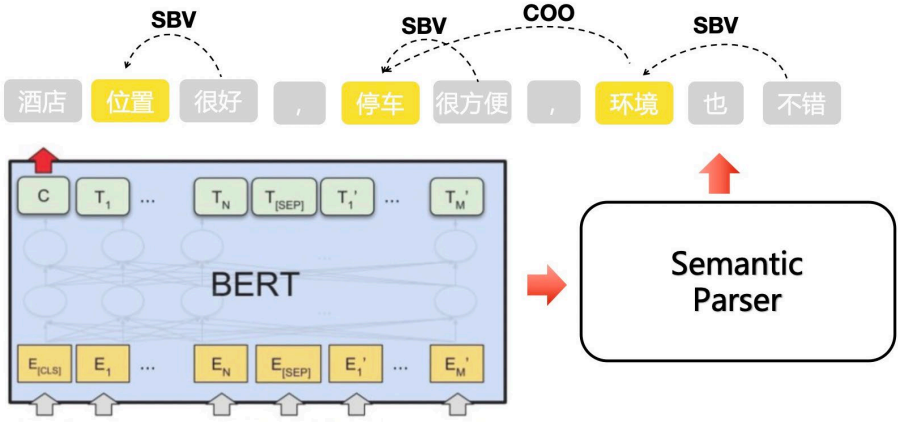


图 14　观点挖掘步骤

第一步，先通过观点发现在用户评论中挖掘出多种观点。我们采用基于序列标注的模型发掘句子中的实体和观点描述，并使用依存分析模型对实体 – 观点的关系进行判断。

第二步，在挖掘到一定数量的观点后，再深入挖掘评论中的证据（Evidence），如下图 15 所示。虽然在第一步观点发现时也能找到部分观点的出处，但还有很多用户评论的观点是隐式的。比如对于“是否可以带宠物”，用户不一定在评论中直接指明，而是说“狗子在这里玩的很开心”。这就需要我们对评论语句进行深度语义理解，从而归纳其中的观点。在方案的落地过程中，最初我们使用了分类模型对观点进行分类，输入用户评论，用编码器对句子进行理解，之后各个观点的分类头判断观点正向程度。但随着自动化挖掘的观点增多，为了减少人工标注分类任务的成本，我们将其转换成了匹配任务，即输入观点标签（Tag）和用户评论，判断评论语句对该观点的支撑程度。最后，为了优化速度，我们对 12 层 Transformer 进行了裁剪，在速度提升 3 倍的情况下效果只降了 0.8%，实现了大批量的观点离线挖掘。

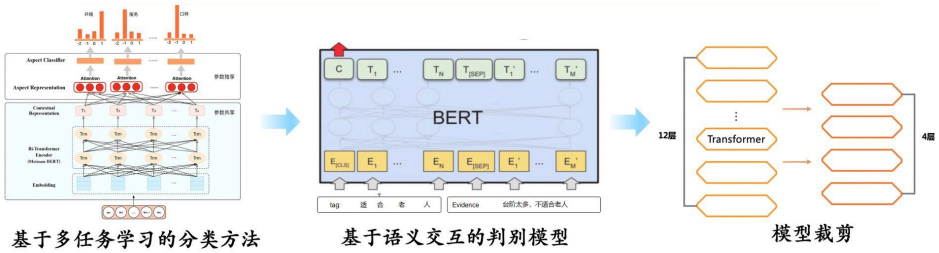


图 15　观点证据挖掘步骤

2.5 端到端方案的探索

在上文中，我们针对多跳、带约束等复杂问题设计了不同的方案，虽然可以在一定程度上解决问题，但系统的复杂度也随之提高。基于关系识别模块的预训练思路，我们对通用的、端到端的解决方案进行了更多的探索，**并在今年的 EMNLP 发表了《Large-Scale Relation Learning for Question Answering over Knowledge Bases with Pre-trained Language Models》论文**。

对于 KBQA，目前学术界有很多研究专注于图学习方法，希望用图学习来更好地表示子图，但却忽略了图谱节点本身的语义。同时，BERT 类的预训练模型是在非结构化文本上训练的，而没接触过图谱的结构化数据。我们期望通过任务相关的数据来消除两者的不一致性，从而提出了三种预训练任务，如下图 16 所示：

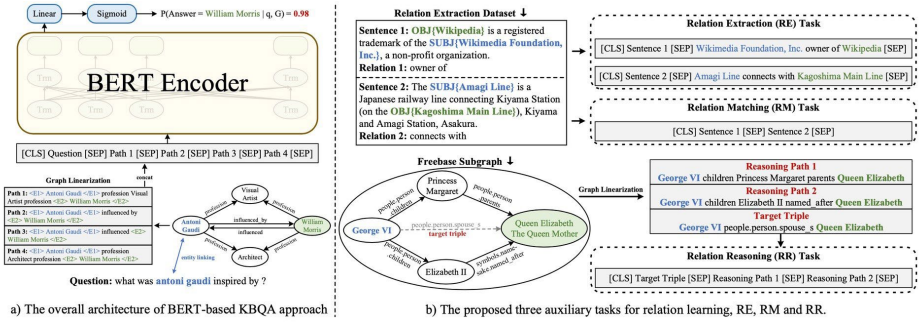


图 16　关系识别预训练任务

- **Relation Extraction**：基于大规模关系抽取开源数据集，生成了大量一跳（[CLS]s[SEP]h, r, t[SEP]）与两跳（[CLS]s1，s2 [SEP]h1，r1，t1 (h2)，r2，t2 [SEP]）的文本对训练数据，让模型学习自然语言与结构化文本间的关系。
- **Relation Matching**：为了让模型更好的捕捉到关系语义，我们基于关系抽取数据生成了大量文本对，拥有相同关系的文本互为正例，否则为负例。
- **Relation Reasoning**：为了让模型具备一定的知识推理能力，我们假设图谱中的 (h, r, t) 缺失，并利用其他间接关系来推理 (h, r, t) 是否成立，输入格式为：[CLS]h, r, t[SEP]p1 [SEP] … pn [SEP]。

经过上述任务预训练后，BERT 模型对于 Query 和结构化文本的推理能力显著提升，并且在非完全 KB 的情况下有更好的表现，如下图 17 所示：

Model	dev set		test set
	Hits@1	F1	Hits@1
Baselines			
KV-Mem†	-	38.6	46.7
GraftNet‡	-	62.4	66.7
PullNet†	-	-	68.1
EmbedKGQA†	-	-	66.6
NSM†	-	67.4	**74.3**
Our implementation			
BERT	71.0	63.4	71.2
$\text{BERT}_{+\text{RE}}$ *pre-train*	68.1	62.1	72.8*
$\text{BERT}_{+\text{RM}}$ *pre-train*	71.8	63.4	72.6*
$\text{BERT}_{+\text{RR}}$ *pre-train*	69.8	61.8	71.7*
$\text{BERT}_{+\text{RE,RM,RR}}$ *pre-train*	69.4	62.5	72.3*
$\text{BERT}_{+\text{RE}}$ *joint*	67.3	57.4	72.4*
$\text{BERT}_{+\text{RM}}$ *joint*	72.2	64.5	**72.9***
$\text{BERT}_{+\text{RR}}$ *joint*	67.3	62.9	71.2
$\text{BERT}_{+\text{RE,RM,RR}}$ *joint*	71.8	60.0	72.0*

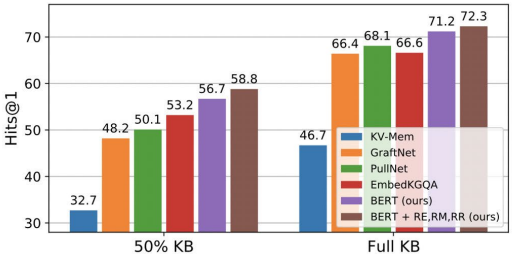


图 17　模型效果

3. 应用实践

经过一年多的建设，当前 KBQA 服务已经接入美团的旅游、酒店、到综等多个业务，辅助商家及时回答用户问题，并提升了用户的满意度和转化率。

3.1 酒店问一问

酒店是用户出行的必备需求之一，但一些中小商家没有开通人工客服入口，无法及时回答用户信息。为满足用户对详情页内信息的快速查找，智能助理辅助未开通客服功能的酒店商家进行自动回复，提升用户下单转化率。用户可询问酒店以及房型页的各类信息，如下图 18 所示：

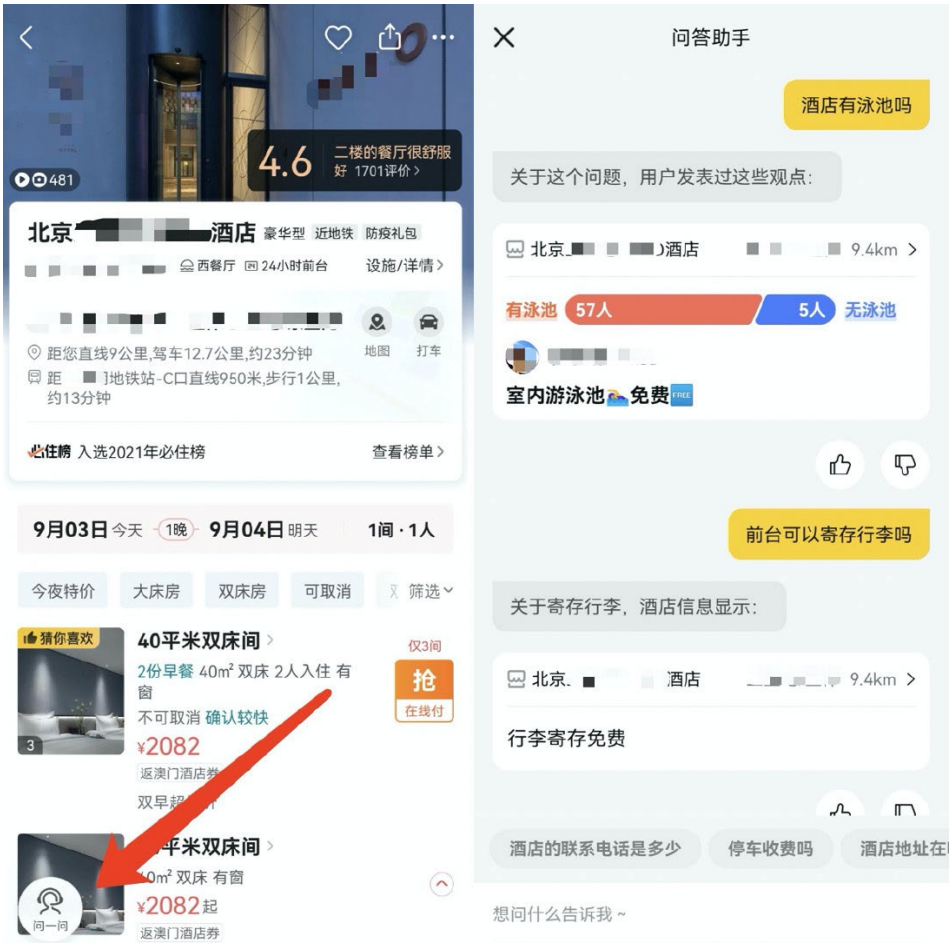


图 18　酒店问一问产品示例

3.2 门票地推

门票地推致力于帮助旅游商家解决主要的卖票业务，在景区高峰时段，线上购票相比于排队更加便捷，然而仍有很多用户保持着线下购票的习惯。美团通过提过二维码以及简单的交互，提升了商户卖票以及用户购票的便捷程度。同时，我们通过在购票页内置「智能购票助手」，解决用户购票过程中的问题，帮用户更快捷地买到合适的门票，如下图 19 所示：

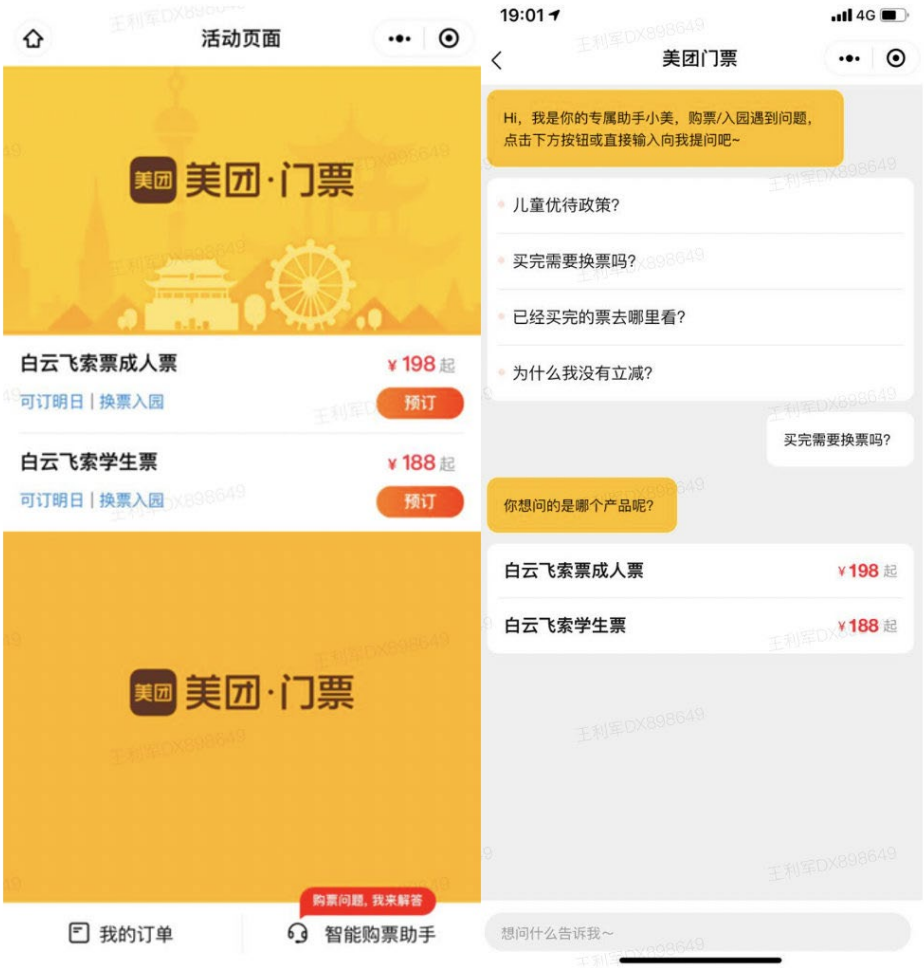


图 19　门票地推产品示例

3.3 商家推荐回复

除了出行场景外，用户在浏览其他本地服务商家时也会有很多问题，比如“理发店是否需要预约?”、“店家最晚几点关门?”，这些都可以通过商家客服进行咨询。但商家本身的人力有限，难免在高峰时期迎接不暇。为了降低用户的等待时间，我们的问答服务会为商家提供话术推荐功能，如下图 20 所示。其中 KBQA 主要负责解答商家、团购相关的信息类问题。

图 20　商家推荐回复产品示例

4 总结与展望

KBQA 不仅是一个热门的研究方向，更是一个复杂的系统，其中涉及到实体识别、句法分析、关系识别等众多算法，不仅要关注整体准确率，更要控制延时，对算法和工程都提出了很大的挑战。经过一年多的技术的探索，我们团队不仅在美团落地多个应用，并在 2020 年获得了 CCKS KBQA 测评的 A 榜第一、B 榜第二和技术创新奖。

同时我们开放出了部分美团数据，与北大合作举办了 2021 年的 CCKS KBQA 测评。

回到技术本身，虽然目前我们的 KBQA 已能解决大部分头部问题，但长尾、复杂问题才是更大的挑战，接下来还有很多前沿技术值得探索，我们希望探索以下方向：

- **无监督领域迁移**：由于 KBQA 覆盖美团酒店、旅游到综等多个业务场景，其中到综包含十多个小领域，我们希望提升模型的 Few-Shot、Zero-Shot 能力，降低标注数据会造成的人力成本。
- **业务知识增强**：关系识别场景下，模型核心词聚焦到不相关的词将对模型带来严重的干扰，我们将研究如何利用先验知识注入预训练语言模型，指导修正 Attention 过程来提升模型表现。
- **更多类型的复杂问题**：除了上述提到的带约束和多跳问题，用户还会问比较类、多关系类问题，未来我们会对图谱构建和 Query 理解模块进行更多优化，解决用户的长尾问题。
- **端到端 KBQA**：不管对工业界还是学术界，KBQA 都是一个复杂的流程，如何利用预训练模型以及其本身的知识，简化整体流程、甚至端到端方案，是我们要持续探索的方向。

也欢迎对 KBQA 感兴趣的同学加入我们团队，一起探索 KBQA 的更多可能性！简历投递地址：wangsirui@meituan.com。

作者简介

如寐、梁迪、思睿、鸿志、明洋、武威，均来自搜索与 NLP 部 NLP 中心知识图谱组。

本地生活综合性需求图谱的构建及应用

作者：李翔

一、背景

1.1 业务现状

以“帮大家吃得更好，生活更好”为使命，美团覆盖了外卖、餐饮、酒店、民宿、旅游、门票、电影 / 演出、休闲 / 玩乐、丽人、医疗、亲子、教育、结婚、生活服务等数百个行业，满足数以亿计用户的多样化生活服务需求。为了持续增加平台价值，除了推动用户和商户在数量和质量上的不断提升之外，更高效地匹配用户需求和商户供给也是重要抓手之一。

为了提升匹配的效率，我们需要更充分、深入地理解用户需求和商户供给，并且尝试从用户视角出发来组织和管理供给。当前，“行业 – 类目 – 商户 – 商品”是一种较为常见的供给组织和管理方式，然而随着业务和行业的快速发展，这种组织方式带来的痛点日渐突显，举例来说：

- 部分指向性不明确的用户需求，较难得到适合的匹配结果。例如“周末陪宝宝去哪儿玩?”由于平台的匹配结果不理想，用户往往只能在线下完成类目决策，决定带宝宝去农家乐烧烤，再到平台上搜索相应的农家乐团购。
- 部分需求跨多个类目，匹配过程不够流畅。例如“周末和朋友去哪里放松一下?”用户在线下完成类目决策后，可选类目包括 KTV、酒吧、密室、桌游等，但各类目的承载页面相互独立，用户需要在承载页面间来回切换。
- 在部分指向明确的类目中，用户仍然难以找到满足自己需求的供给。例如在医美类目下，因为缺乏相关知识，用户往往并不了解商户提供的形形色色的服务项目分别具有什么功效，适合什么部位，应该使用什么材料，无法高效地找到适合自己的服务供给。

上述问题的本质原因在于供给的组织方式主要是从行业视角出发，没有充分考虑到用户视角。在当前以满足用户需求为首要目标的市场环境下，我们需要迭代完善现有的供给组织方式。

1.2 问题解析

为了解决上述问题，我们尝试从一个外部视角，结合第一性原理去进行剖析。在整个人类社会中，按照马斯洛需求层次[1]的理论，人的需求是可以被归纳及分层的。如果将人类社会看作一个系统，人类在其中一个称为“市场”的子系统中，通过交易来满足自己的需求。

人类从需求层次出发，在市场中完成交易，最终需求得以满足，那么在市场中交易的过程可以被拆解为“起心动念 -> 考虑 -> 选择评估 -> 交易购买 -> 履约 / 服务”。并且，经由前三个阶段，用户的需求逐渐从粗粒度演变为细粒度，从抽象过渡到具象。下面将结合具体的例子来进行解读：

- **需求层次**：在马斯洛需求层次中有一个情感层次，对应了人类丰富多样的情感需求，包含亲情、友情、爱情等等。作为一个母亲，与自己的宝宝之间往往存在一种强烈的母子情，并且希望不断去强化这种情感。
- **起心动念**：为此，母亲往往都会尽力花较多的时间去陪伴宝宝。通过增加陪伴方式的维度，变为“陪宝宝玩”；通过增加时间、空间等维度，变为“周末陪宝宝去哪儿玩?”
- **考虑**：对于上述问题，母亲会找出很多种方案，比如户外烧烤、农家采摘、主题公园等等。当她决定选用户外烧烤这一方案时，便会转化为具体的商品需求，如购买烤架。
- **评估选择**：母亲于是在可触达的供给范围中进行选择。正如人们常说的货比三家，选择的依据会各不相同，如价格、质量、口碑等等。
- **交易购买**：当完成选择之后，母亲便会进行交易，以换取商品或服务。
- **履约服务**：即商品到货、完成服务等等。

市场是人类社会的一个子系统，电商平台则是市场中的一个子系统。同时，电商平台是一种线上化的子系统，它提供了搜索、推荐等形式的供给检索能力。当前的一个现状是：用户往往在线下完成了“起心动念”到“考虑”的过程，转化为具体的商品 / 服务需求，随后进入电商平台完成“选择评估”到“履约服务”的后续过程（如下图 1 所示）。而电商平台往往聚焦于后三个阶段的能力提升，容易忽略前两个阶段。

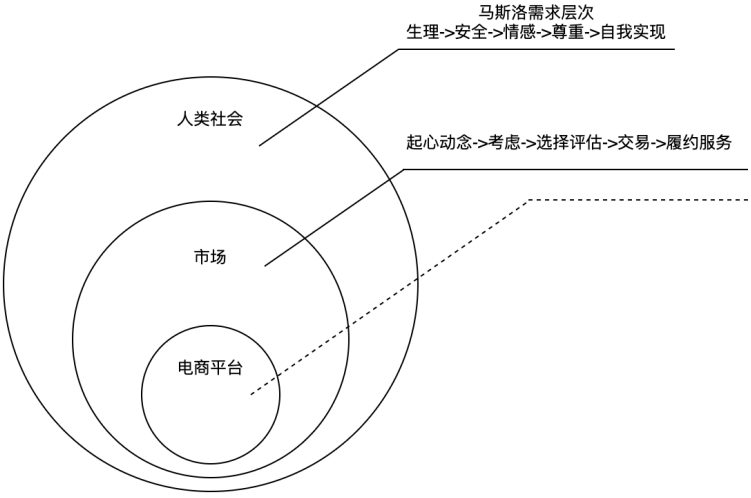


图 1

因此，用户较难形成在平台上完成“起心动念”和“考虑”的心智，而大部分电商平台按“行业 – 类目 – 商户 – 商品”的方式对供给进行组织和管理。最终，用户和电商平台之间形成了一种相互掣肘的关系。

事实上，相对于“烤架”这种明确的商品需求，用户还有很多抽象的、模糊的、不明确的需求，依然停留在前两个阶段。例如，周末陪宝宝去哪儿玩？周末和朋友去哪里放松一下？结婚前怎样才能让自己变得更美？暑假如何培养孩子的动手能力？同时，这样的需求往往跨越多个类目，或者在同类目下存在多种选择。

电商平台只有打破既有的掣肘关系，向用户提供前两个阶段（起心动念、考虑）的能力，才能进一步满足用户的需求。用户的决策成本进一步降低，决策过程更加连贯，

用户体验也能得到相应的提升，同时用户在市场中的交易过程也能够进一步实现线上化。

以“行业 – 类目 – 商户 – 商品”作为参考，**如果电商平台能够识别出用户在前两个阶段的需求，并且在其和供给之间建立新的关联关系，辅以搜索、推荐等检索能力，用户便有可能在线上完成前两个阶段**。而知识图谱作为一种揭示实体之间关系的语义网络，用以解决上述问题显得尤为合适。

二、解决方案

2.1 解决思路

延续 1.2 中的例子，这位母亲将“周末陪宝宝去哪儿玩?”的需求转化为具体的“户外烧烤”需求，延伸至更为具体的“烤架”和“农家乐团购”需求。此时，母亲便会分别前往各类实物电商平台以及以美团为代表的生活服务电商平台进行“选择评估”。两种电商平台分别在实物供给池或服务供给池中采用搜索、推荐之类的召排技术定位到具体的商品 / 服务，并反馈给这位母亲。

对于 1.2 中的所述，技术团队期望达成的目的，目前比较有代表性的参考案例是阿里巴巴电商认知图谱 AliCoCo[2]。它基本构建思路是从用户视角出发，先进行各种类型的原子词挖掘，再进一步对原子词进行组合和挖掘出相关候选短语，再从中识别出真实的用户需求，最后关联至相应的供给。它的层次结构如图 2 所示：

- **分类层**：构建完备的分类体系，包含了的大千世界的各种分类，既有通用的空间、时间等类别，也有电商中涉及的颜色、功能以及最为重要的品类等类别。
- **原子概念层**：在分类层诸多类别的基础上进行扩展，包含了各种类别下的原子概念（例如空间 –> 户外、事件 –> 烧烤、时间 –> 圣诞节、颜色 –> 红色、功能 –> 保暖、品类 –> 连衣裙）及原子概念之间的关系。
- **电商概念层**：在原子概念层之上，包含了由原子概念组成或直接挖掘的短语粒度的用户购物需求，即电商概念（例如户外烧烤），从而将用户购物需求显式地用一个符合自然语言的短语表示。

- **商品层**：包含了商品与各种原子概念、电商概念之间的关联关系（例如户外烧烤 –> 烤架、黄油、锡纸）。

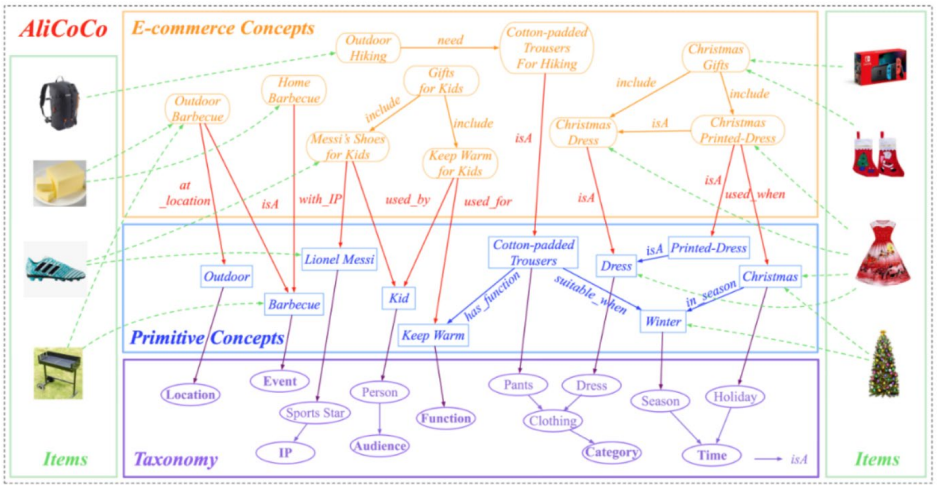


图 2

基于上述图谱，这位母亲可以直接在天猫上表达“户外烧烤”的需求，而非更具体的“烤架”；天猫也会将烤架以外其他和户外烧烤相关的重要商品反馈给这位母亲。从对应关系来看，AliCoCo 的电商概念层对应至“考虑”阶段，商品层对应至“选择评估”阶段。显然，由于 AliCoCo 的存在，天猫能够从“考虑”阶段开始介入用户的交易过程。

由此推论，我们应该可以将图谱构建得更为完整，以覆盖“起心动念”阶段。在此阶段，人类的需求，按马斯洛需求层次，通过增加一个或多个维度约束，而被逐渐具象化。对于这样的维度约束，我们统称为“场景约束”。因此，我们将“考虑”阶段对应的需求称为“具象需求”；将“起心动念”阶段对应的需求称为“场景需求”。为此，我们希望构建一个**本地生活综合性需求图谱**（**GENE**：lifestyle **GE**neral **NE**eds net），如下图 3 所示。对于本地生活场景下所涉及的数百个综合性行业，我们认为新的供给组织方式更能贴近用户需求，也是从用户视角去解决供需匹配的问题。

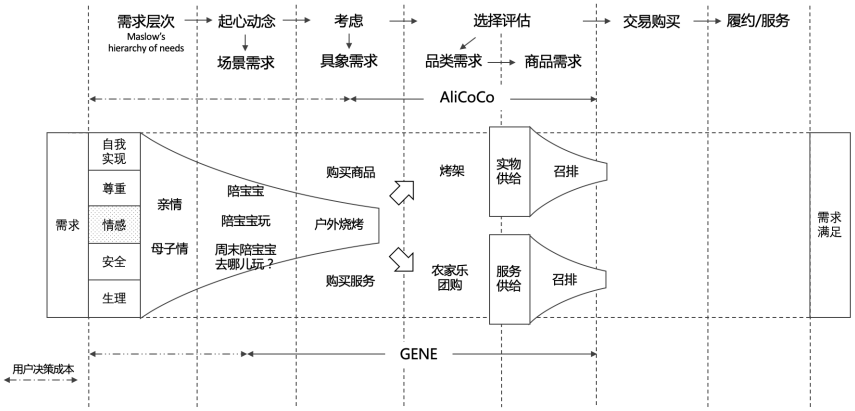


图 3

2.2 具体方案

延续 2.1 中的构建思路，我们尝试构建一套多层次的图谱结构，并将“具象需求”和“场景需求”拆分为独立的层次，既避免了两种类型的需求合在同一层级中而引起混淆，又能对用户场景化的需求进行更详尽的拆解、更丰富的刻画。本地生活综合性需求图谱（GENE），主要由六部分构成，包含场景需求层、场景要素层、具象需求层、需求对象层、行业体系层和供给层，如下图 4 所示：

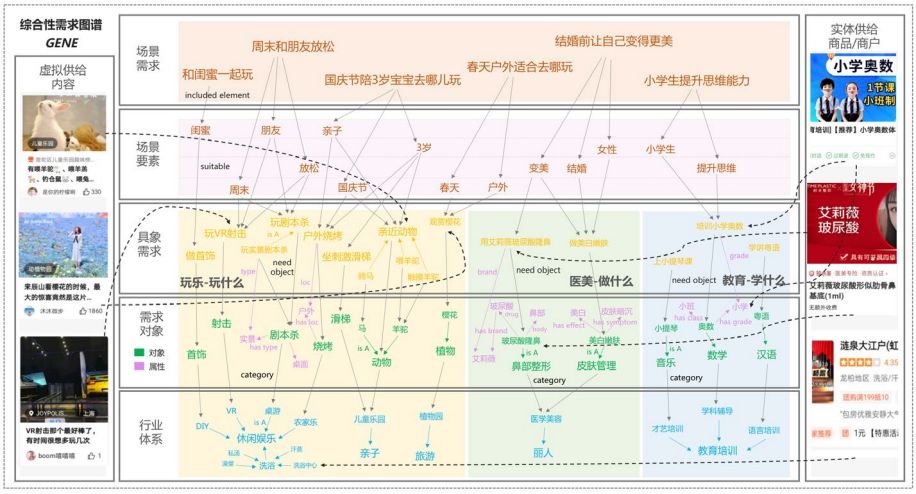


图 4

在场景需求层，我们用 Human-Readable 的短句来表征场景化的用户需求，如“国庆节陪 3 岁宝宝去哪玩”、“结婚前让自己变得更美”、“小学生提升思维能力”等。一个场景需求的表述中，通常会包含人物、目的、时间、空间、方式等要素，以“国庆节陪 3 岁宝宝去哪玩”为例，“3 岁宝宝”是人物，“陪宝宝玩”是目的，“国庆节”是时间。

在场景要素层，为了更好地表达场景需求，我们将这些短句进行拆解，细化成多个细粒度的词汇，用这些词汇对场景需求中的人物、目的、时间、空间、方式等要素进行完整的覆盖和系统的组织，我们将其称为“场景要素”。

在具象需求层，由于场景需求的表述往往并不显式地指向某个具体的服务 / 供给，而是隐含了一批潜在的适合这一场景的服务 / 供给。例如，在“国庆节陪 3 岁宝宝去哪玩”这个例子中，户外烧烤、喂羊驼、玩滑梯、骑小马等都是适合这一场景需求的具体服务。所以我们需要将所有这些具体的服务通过短语的形式显式的展现出来，这些短语直接反应了用户具体的服务需求，被称之为“具象需求”。

在需求对象层，为了进一步的理解具象需求，我们将具象需求分为具体的服务需求所对应的对象，我们称之为“需求对象”，以及该服务中用户和对象之间的交互行为。例如，对于具象需求喂羊驼，可以分为羊驼（需求对象）和喂（服务交互），由于本地生活服务的多样性，围绕羊驼这一需求对象，除了喂羊驼外，还可以产生触摸羊驼、骑羊驼和看羊驼表演等多种具象需求。这一层除了包含需求对象节点外，也会涵盖需求对象的属性信息，以对需求对象进行更详尽地描述。例如对于具象需求户外烧烤，可分为烧烤（需求对象）、户外（需求对象属性）和体验（隐含的服务交互）。

在行业体系层，由于用户的场景需求和具象需求往往会跨越多个传统的服务类目，为了给用户需求确定一个具体的业务范围，我们还需要构建出各行业涉及的类目体系，作为上述各层构建的业务基础。

在供给层，包含内容这类虚拟供给和商户与商品这类实体供给，这些供给将会和具象需求和场景需求等节点进行关联，从而为用户需求提供相对应的供给支撑。例如，一个提供户外烧烤的供给将会关联上具象需求“户外烧烤”，并进一步关联上场景需求

“国庆节陪 3 岁宝宝去哪玩”。

综上所述，在本地生活综合性需求图谱中，用户场景化的需求和具体的服务需求被分别表达成短句级别的场景需求和短语级别的具象需求。这两种需求分别通过场景要素和需求对象进行表达。最后，不同类型的供给都会和场景需求和具象需求相关联，从而以用户需求为纽带，提升供给和用户的匹配效率。

三、实现方法

当前，本地生活综合性需求图谱已经初步涵盖了用户在玩乐、医美和教育三个本地生活相关行业中的多元化需求。我们在图谱的构建过程中，自底向上按照行业体系层、需求对象层、具象需求层、场景要素层、场景需求层的顺序逐层进行构建，并将各层的节点与各种类型的供给建立关联关系。下文将以玩乐行业为例，对图谱每一层的构建细节以及涉及的算法进行介绍。

3.1 行业体系层

3.1.1 行业类目树的构建

在玩乐行业中，行业体系层包含了能够提供玩乐服务的类目，类目信息通过一个树形结构来表征。由于玩乐行业体系的构建，对专家知识的要求非常高，并且这部分的设计对后续各层的知识挖掘至关重要，因此我们没有直接进行人工定义，而是以当前我们成熟的行业类目树为基础，通过对其进行剪枝和分裂来构建。

首先在类目树中筛选出和玩乐相关的一级类目节点，包括“休闲娱乐”、“亲子”、“旅游”、“餐饮”等。对于每个一级类目，我们进一步筛选其和玩乐相关的下一级类目直至叶子类目，并剪去与玩乐无关的类目。此外，我们还对剪枝后的类目树中可以细分的玩乐相关的叶子类目进行分裂，例如将“洗浴”细分为“私汤”、“洗浴中心”等，最终得到完整的玩乐行业类目树。

3.1.2 类目的供给关联

确定了类目树后，我们还需要获取实体供给（商户和商品）和虚拟供给（内容，例如

UGC）与类目的从属关系，从而为后续的一系列挖掘提供数据支持。由于商品和内容均可链接到商户，所以我们只需要获取商户与类目的从属关系即可。玩乐类目树是对我们已有的类目进行剪枝和分裂得到的，除了分裂出的新类目外，其余类目和商户的关系均可直接继承原有的结果。对于新分裂的类目，我们则需要重新构建商户和其之间的从属关系。

要判断一个商户属于哪个类目，最直观的依据就是商户名、商品名及商品详情，然而很多商户的商户名和商品包含的信息往往较少，增加了类目判别的难度。为了确保商户类目判别的准确性，我们引入更多的商户信息，包括商户 UGC 和商户画像，设计了一个多源异构数据融合判别模型，整体模型结构如下图 5 所示：

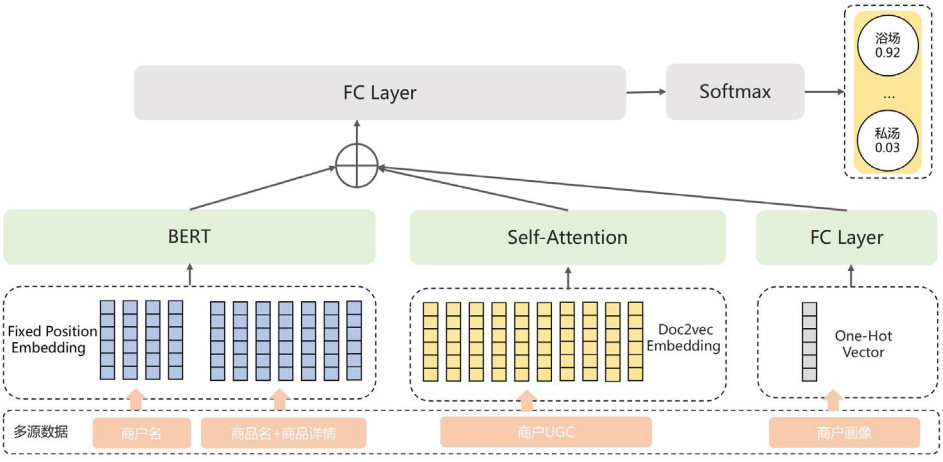


图 5

其中，不同来源数据的特征提取和处理方式如下：

- 商户名、商品名及商品详情：均为文本数据，直接通过 BERT[3] 提取文本特征后输出。
- 商户 UGC：由于商户的 UGC 往往数量非常多，为了对其信息进行有效利用，首先通过 Doc2Vec[4] 的方式进行 Encode 得到 UGC 的特征后，再通过一个 Self-Attention[5] 模块进行特征处理后输出。
- 商户画像：转成 One-Hot 特征后，通过全连接层进行非线性映射后输出。

上述三种特征相连接后进行融合，通过全连接层和 softmax 层实现最终的类目判别。基于多源数据的融合建模，商户信息得到了充分利用。以洗浴细分类目为例，仅利用商户名、商品名及商品详情数据，基于 BERT 判别，准确率为 92%，而基于多源融合模型判别后，准确率提升到 98%。

3.2 需求对象层

在需求对象层，我们希望能够挖掘出玩乐行业体系中各类目涉及的玩乐对象词作为该层的节点，这些词能够描述出用户在实际玩乐过程中的交互对象，这是用于组成具象玩乐需求的基础。为了确保玩乐对象挖掘的全面性，我们采用多源多方法的形式。

在数据上，我们采用来自商户和用户的相关文本作为挖掘语料。在方法上，我们采用两种方式来挖掘玩乐对象词：

- 第一种是无监督的扩充，在开始挖掘前运营会首先根据经验，提供一些玩乐对象词作为种子输入，我们提前利用语料构建无监督的 Skip-Gram 结构的 Word2Vec 模型，对业务输入的种子词提取词向量，并结合余弦相似度，快速扩充相关的对象词。
- 第二种是有监督的标注，我们将其定义序列标注问题，采用基于 BERT+CRF 的模型，在语料中自动识别出新的对象词。

在实践过程中，为了更高效的挖掘，我们将无监督环节扩充和质检后的对象词在语料中进行文本匹配，并将匹配结果转化为有监督标注环节的训练样本；同时对于有监督标注的结果，经过运营质检后，也会将其作为无监督扩充的输入，通过两个环节相结合，我们完成玩乐对象的挖掘，完整流程如下图 6 所示。此外，在运营人工审核玩乐对象词的过程中，对于一些业务已知的核心玩乐对象，也会直接输入业务侧已经沉淀的相关特征作为其属性，进一步完善玩乐对象的信息，例如对于“剧本杀”这一玩乐对象，增加相应的“实景”和“桌面”等类型属性。

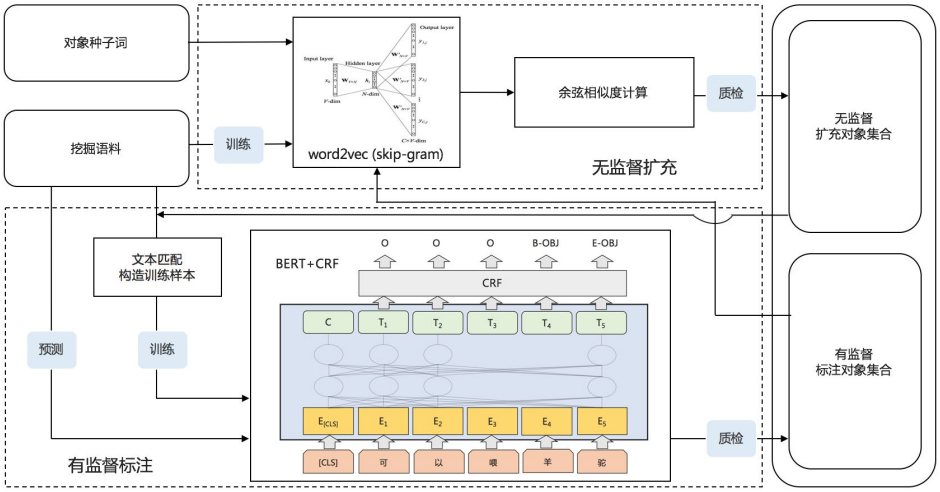


图 6

在获取对象词后，我们还需要知道对象词属于哪个类目，以便下一步的具象需求挖掘及供给关联，为此我们构建对象词和类目之间的关系。通过各类目下的语料文本中提及对象词的次数来衡量两者关系，是最直观且准确率最高的方法。因此，我们直接使用对象词在每个类目下的语料中进行文本匹配，通过词频高低来确定关系。同时，我们进一步构建对象词之间的上下位和同义关系，当前常见的有通过投影和分类（如 BERT 句间关系模型）等方法来进行关系判别等有监督方法。在实际过程中，我们采用规则辅助人工的方式，基于对象词的统计特征及 Pattern 共现的结果指导人工快速完成构建。

3.3 具象需求层

3.3.1 具象需求挖掘

具象需求层可以看成是用户在玩乐行业中的具体服务需求的集合，每个具象玩乐需求是该层的一个节点，由玩乐对象叠加用户与对象之间的多元化的交互行为及对象描述信息得到，它通过短语的形式表达出用户对于玩乐服务供给的本质诉求。具体玩乐需求挖掘的流程流程可以分为两个步骤：

- **候选短语生成**：围绕玩乐对象词，生成大量包含玩乐对象的短语，作为具象玩

乐需求候选集。

- **短语质量判别**：建立一个语义判别模型从候选集中提取真实的具象玩乐需求。

候选短语生成

在步骤 1 中，首先我们以需求对象词为核心，采用与玩乐对象挖掘相同的语料，进行候选短语的生成。常用的短语挖掘算法如 AutoPhrase[6]，是以 Ngram 来进行短语组合，而这种形式对于有需求对象的短语显得过于冗余，所以我们考虑基于句法结构来进行短语挖掘。

为了使生成短语符合句法的要求，我们以预设的句法关系为模板进行挖掘。在大规模语料中为了更高效地挖掘句法关系，我们基于更轻便的 ELECTRA[7] 预训练模型获取句子各成分的 Embedding 后，再利用 BiAffine[8] 预测其句法关系。通过依存句法分析，我们在各个类目的语料中挖掘出包含相应的玩乐对象且符合句法关系的短语。此外，需求对象层中对象的属性也会被用作对象描述进行短语生成。最终所有挖掘的短语，经过词频等统计特征粗筛后将作为具象玩乐需求的候选集，挖掘示例如图 7(a) 所示。

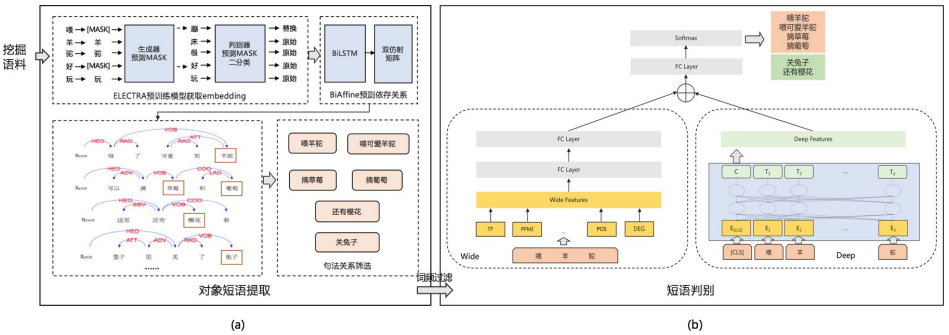


图 7

短语质量判别

在步骤 2 中，经过步骤 1 获取的候选集短语虽然符合预设的句法关系，但是从语义上仍然存在大量与用户实际需求不符的表述，通过抽检分析我们发现符合要求的短语不足 10%。如何从海量的候选短语中挑选出反映真实的用户具象玩乐需求的短语成为

亟需解决的问题。

AutoPhrase 通过基于短语统计特征的判别模型进行短语打分，然而仅通过统计特征难以识别语义质量低的短语，为此我们进一步基于统计和语义特征联合建模，构建一个 Wide&Deep[9] 结构的判别模型，对候选集合中的短语是否为具象玩乐需求进行判别，我们希望判别模型能过滤掉大量的低质短语，从而为运营节省大量的人力成本。判别模型的整体结构如图 7(b) 所示，其中：

- Wide 部分，提取候选短语的全局和上下文的统计特征，通过全连接层进行非线性映射后输出。
- Deep 部分，提取候选短语的深度语义特征，通过 BERT 完成相应特征的提取后输出。

上述 Wide 和 Deep 部分输出的特征相连接后进行融合，优势互补，经过全连接层和 softmax 层来实现最终的短语判别。在实践过程中，除了直接使用已经积累的短语标签作为正样本外，我们还通过预设一些常识性的 Pattern 从候选集中构造正样本，例如观赏 [植物]、触摸 [动物]，并对候选集采样构造负样本，完成初版模型的训练，之后结合主动学习，经过多轮迭代，模型最终达到 92% 的召回率和 85% 的准确率。通过质量判别后保留的短语则会交由运营人工审核提炼后成为最终的具象玩乐需求。

3.3.2 具象需求的供给关联

在具象需求层中，由于具象玩乐需求是由玩乐对象得到的，所以两者之间天然地建立了对应关系。而对于具象玩乐需求之间的上下位和同义关系，则可以基于其对象之间的关系及其句法关系，在人工审核的环节来辅助人工完成构建。除此之外，更重要的是需要将具象玩乐需求与实体供给（商户和商品）及虚拟供给（内容，例如 UGC）进行关联。

我们将这个问题抽象为一个语义匹配的问题，通过具象玩乐需求与其对应的类目的供给的文本信息进行匹配来实现，其中，商户使用商户名文本信息，商品使用商品名和商品详情文本信息，UGC 使用其自身文本信息。由于 UGC 和商品属于商户的一部

分，所以具象玩乐需求与 UGC/ 商品的关系也会加入其与商户关系的构建中。整体匹配流程如下图 8 所示，我们首先进行具象玩乐需求与 UGC/ 商品的匹配，在此基础上再结合商户名文本的匹配结果，一起通过规则聚合后关联到商户。

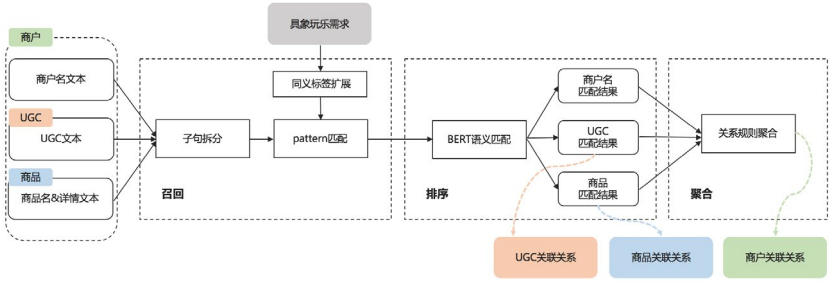


图 8

由于具象玩乐需求数量众多，同时供给的文本信息通常包含多个子句，出于效率和效果的平衡考虑，我们将匹配过程分为了召回和排序两个阶段。

在召回阶段，我们粗筛出和具象玩乐需求可能有潜在关联的子句。对于具象玩乐需求，我们基于构建的同义关系，扩展具象需求的同义标签，并将其与子句文本进行粗粒度 Pattern 匹配，对于匹配中的子句则将进入到排序阶段进行精细化的关联关系计算。

在排序阶段，我们构建基于 BERT 句间关系分类的语义匹配模型，通过在 BERT 后增加全连接层和 softmax 层来实现分类。模型通过对召回阶段得到的粗筛样本进行预测，识别两者在语义上的匹配关系（关联 / 不关联）。最终供给关联关系的平均召回率和准确率分别达到 90% 和 95%。

3.4 场景要素层

3.4.1 场景要素拆解

场景要素层包含了组成用户场景化需求的场景要素。如文初提及，要描述一个场景，需要交代特定的人物、时间、空间、目的等要素。例如，对于“国庆节陪 3 岁宝宝去哪儿玩?”这个场景化需求，我们可做如下拆解：时间 – 国庆节，人物 –3 岁宝宝，目的 – 亲情陪伴（陪宝宝玩）。所以我们按照上述方式，对场景要素进行拆解，以期对场

景要素的挖掘和梳理尽可能的全面与系统。

3.4.2 场景要素挖掘

完成了场景要素拆解后，下一步就是要分别在每个拆解的类别中进行场景要素的挖掘。场景要素作为具象需求的场景化的信息，往往来自于用户的直观感受，所以挖掘的语料我们选择与具象玩乐需求关联的 UGC 的上下文语料。与需求对象挖掘的方法类似，我们将每个类别的已提炼总结的场景要素作为种子词，通过相关要素扩充和序列标注的方式完成场景要素的挖掘。

确定场景要素后，接下来的关键是完成场景要素与具象玩乐需求的关系构建，即对于每个场景要素，找出其适合的具象玩乐需求，如春季适合观赏樱花、孩子适合亲近动物。我们通过对 UGC 文本分析后发现，UGC 中用户说到某一具象玩乐需求的同时，往往也会交代一些相关的场景要素信息，所以我们继续选择与具象玩乐需求关联的 UGC 的上下文语料，作为关系构建的数据来源。

最初我们采用基于 Pattern 的方法，通过归纳可用于判断场景要素和具象玩乐需求关系的 Pattern，从语料中直接抽取包含两者的文本。但由于用户表达的多样性，不仅准确率无法保证，同时有限的 Pattern 也影响了召回，因此我们进一步尝试使用基于模型判别的方法提升泛化性，完善关系的构建。

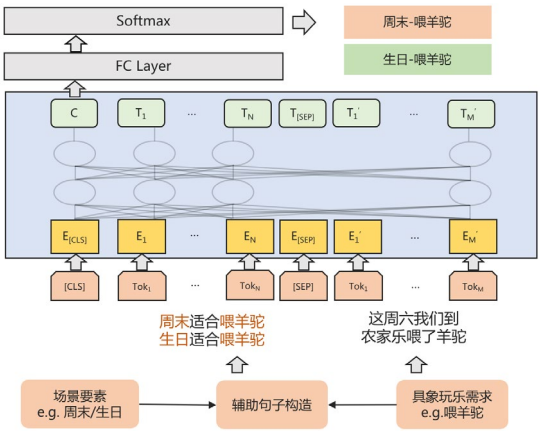


图 9

由于我们使用的语料中的具象玩乐需求是已知的，如果将场景要素看成是具象玩乐需求的属性，那么问题就可以看成是一个属性级（Aspect-Based）的分类问题。参考属性级情感分类的做法 [10]，我们通过预设句子模板的方式，结合场景要素和具象玩乐需求构造出辅助句子，将属性级分类转换为一个类 QA 的句对分类问题。例如，对于已经关联了“喂羊驼”这一具象玩乐需求的某个语料：“这个周六我们到农家乐喂了羊驼”，其中一个辅助句子为“周末适合喂羊驼”。

我们采用 BERT 句间关系分类模型来实现句对分类，如图 9 所示。辅助句子与语料文本通过 [SEP] 相连后输入模型进行判别，模型输出判别结果（适合 / 不适合）。最后我们根据在所有语料上的关系提取结果，对每个场景要素和具象需求之间的关系进行投票计算评分后来确定两者的关系。

3.5 场景需求层

3.5.1 场景需求组装

在场景需求层，我们会将场景要素层和具象需求层的信息进行组装，从而生成大量的场景需求。组装出的场景需求，既可能仅含场景要素，如“国庆节陪 3 岁宝宝去哪儿玩?”不含任何具象需求，又可以同时包含场景要素和具象需求，如“周末去郊外摘草莓”中，周末、郊外是场景要素，摘草莓是具象需求。

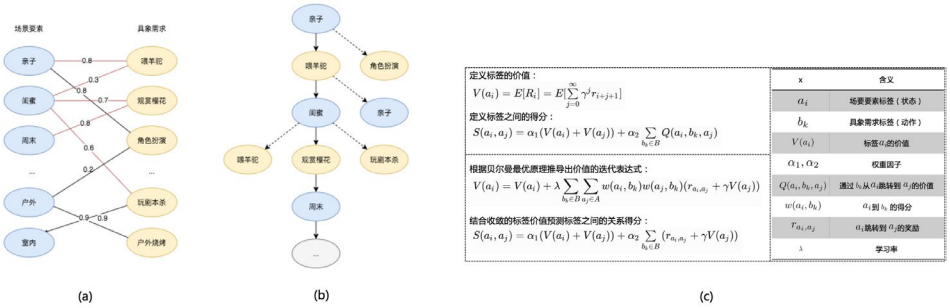
3.5.2 场景需求判别

对于组装得到的场景需求，最重要的是保证其合理性，例如“周末”和“亲子”就是合理的玩乐场景，而“闺蜜”和“亲子”则是矛盾的玩乐场景。为此，我们首先需要计算场景要素之间的关系评分，从而指导场景需求的组装。场景要素只有依托具象需求，搭配上适合的玩法，参与组装的场景需求才有意义。因此，对于场景要素之间的合理关系构建，我们尝试以场景要素和具象需求的关系得分作为依据，通过关系传递评估两个场景要素之间的相关性。

在 3.4.2 节中，我们已经量化了场景要素和具象需求之间的关系评分，一个最直观的想法是通过场景要素 – 具象需求 – 场景要素的关系传递来进行计算场景要素之间关系

的计算。如图 10(a) 所示，以具象需求“喂羊驼”为纽带，可以获取“亲子”和“闺蜜”两个场景要素的关系得分。

我们首先构建场景要素和具象需求的关系评分矩阵，考虑到玩法数量满足长尾分布的情况，对矩阵进行具象需求维度的列归一化处理，同时为了保证场景要素 – 场景要素矩阵的自相关系数为 1，对归一化后的场景要素 – 具象需求矩阵进行 L2 行范数归一化处理，因此该归一化矩阵与自身转置的矩阵乘得到的新矩阵即可作为场景要素 – 场景要素的关系评分矩阵。



(a) (b) (c)

图 10

通过上述方法可以很快得到场景要素之间的关系评分，然而该方法仅以场景要素 – 具象需求 – 场景要素的关系传递模式，计算场景要素在具象需求上的直接共指强弱，导致场景要素关系的覆盖不足。为此，我们扩展为更长的节点关系链传递模式，这种节点之间的传递关系是服从马尔可夫性质的，如图 10(b) 所示。但是随着传递路径的增长，计算成本会呈指数级增长。所以我们采用强化学习 [11] 中的 One-Step 时序差分方法进行求解，以“累计回报的期望最大化”的概念作为节点的价值，以场景要素节点集合作为强化学习概念中的状态空间，具象需求节点集合作为动作空间。

例如当我们处在“亲子”场景要素这个状态下，可以通过选择“喂羊驼”或者“角色扮演”来跳转到下一个状态“闺蜜”或“户外”。该状态跳转过程的决策函数，根据当前场景要素状态关联的所有具象需求中随机抽取一个具象需求节点作为决策行为，抽取概率与得分正相关；状态转移概率则为在该具象需求节点决策下，随机跳转到与

之相关联的场景要素，跳转概率与得分正相关。

同时，对于特定的互斥关系，我们根据实际的业务应用需求制定奖励矩阵实现多样化的场景要素关系评分模型。这样我们将节点关系传递模型转化为了马尔可夫决策模型，结合贝尔曼最优原理推导出的价值迭代表达式和节点对关系得分预测公式，如图 10© 所示。根据图示的公式，在保持策略不变的前提下采用自举迭代的思想计算节点的价值，并进一步计算场景要素之间的关系评分，既能够保证更充分的利用已有的关系网络信息，提高关系覆盖，又能通过奖励矩阵的约束关系降低互斥关系的影响，灵活适应不同业务的需求。

最后基于场景要素之间的关系评分，我们从组装的场景需求集合中选择评分高的场景需求，并根据预设的模板生成最终的场景需求表达，例如“周末和朋友放松”、“和闺蜜一起玩”、“国庆节带孩子户外烧烤好去处”。这些场景需求通过其包含的场景要素 / 具象玩乐需求，可以链接到相应的具象玩乐需求，进而关联相关的供给，从而给用户提供场景化玩乐的解决方案。

四、应用实践

本地生活综合性需求图谱，涵盖了用户的场景需求和具象需求，一方面更前置地参与用户决策，在“起心动念”、“考虑”、“选择评估”等多个阶段影响用户，降低其决策成本，另一方面，提供更多样化的供给选择，高效地进行供需匹配。应用方式上，应用于搜索、推荐等各类业务形态。

经过近一年的建设，当前综合性需求图谱包含数十万核心具象需求和场景需求节点，以及数千万的关系，并在美团的亲子、休闲娱乐、医美、教育培训等多个业务进行了初步的应用实践，下面举例介绍具体的应用方式和应用效果。

4.1 亲子

亲子原频道页用户需求和供给之间匹配效率低下，其中 ICON 按照亲子传统类目划分，无法满足用户不同类型的需求（图 11(a) 左），而底部猜喜的供给形态单一，且体

现用户需求的优质供给不足，决策信息不足（图 11(b) 左），因此对亲子频道页进行改版。为了贴合亲子的业务特色，我们将亲子玩乐相关的需求节点和关系，应用于频道改版后的多个流量位，为其提供标签和供给数据的支撑。

图 11

其中，对于 ICON，基于高频的场景和具象需求，跨类目生成需求 ICON，如“亲近动物”、“带娃泡汤”等（图 11(a) 中）及相应二级页（图 11(a) 右），这些 ICON 包含了原本多个类目中的相似需求，在用户“考虑”阶段为其提供决策信息。

对于底部推荐，我们围绕亲子玩乐的具象需求进行供给优化，将其关联的内容作为优质供给进行推荐，并为每个供给提取包含相应具象需求的文本，作为推荐理由外露，这些句子从用户实际需求的角度展示信息，极大地吸引了用户（图 11(b) 右）。此外，还进一步根据浏览和交易行为，以供给为媒介建立具象需求与用户的关联关系，应用于个性化推荐的召回和排序的优化。改版后的亲子频道页，满足了用户多样化的推荐需求，极大地改善了用户体验。

4.2 休闲娱乐

在休闲娱乐频道页，我们围绕场景需求和具象需求进行了一系列应用。一方面，基于玩乐的场景需求组织新的场景 ICON，如满足用户户外游玩的“踏青赏花”、满足用户在室内潮流玩乐的“室内潮玩”、满足喜欢晚上玩乐用户的“夜生活”、满足和朋友

同事聚会玩乐的“团建聚会”，这些 ICON 从用户场景化的玩乐出发，打破了传统类目的限制，让用户和供给的匹配更加流畅，每个 ICON 的二级页面则会展示每个场景的具象玩法需求及关联的商户和内容。

另一方面，在频道页场景导航模块中，尝试利用场景需求来进一步展示场景化的玩乐信息，包括“一人乐”、“家庭暖”、“过生日”等十多个玩乐场景主题，并针对这些场景的具象需求关联的商户进行推荐。这些场景化的应用（图 12(a)），在“起心动念”阶段即作用于用户，提升了用户的决策效率。

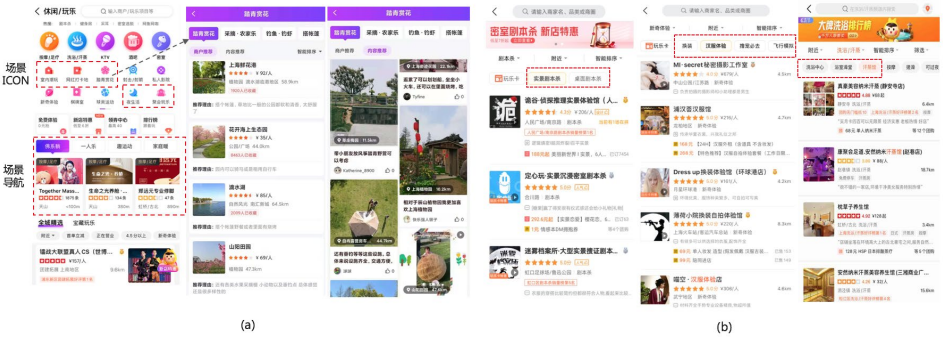


图 12

此外，部分具象需求经过改写可以直接用于相应类目的列表页商户的快筛，例如实景剧本杀 / 桌面剧本杀，换装 / 汉服体验 / 撸宠必去 / 飞行模拟等（图 12(b) 左和中），而我们在行业体系层中类目的细分结果，亦可以成为商户的快筛，例如洗浴类目的细分（图 12(b) 右），这些快筛的应用，更加方便了用户选店。

五、总结展望

在本地生活服务中，如何不断提高供给和用户之间的匹配效率，是摆在我们面前的一道难题。我们尝试以用户关注对象为切入点，通过深挖用户需求并以其为纽带来关联供给和用户。为了全方位挖掘和理解用户需求，我们努力探索并尝试构建本地生活综合性需求图谱，以行业体系层、需求对象层、具象需求层、场景要素层、场景需求层的顺序逐层构建，并为各种类型的供给建立关联关系。

目前，综合性需求图谱的结果可应用于搜索、推荐等各类业务形态，并已在美团多个业务场景中取得实践成果。不过，我们还处在探索的初级阶段，接下来还有很长的迭代之路要走，在此我们提出一些后续的思考和展望：

- **更广的行业覆盖**：一方面加深对已有的玩乐、医美和教育行业的建设，挖掘更多的节点和关系，更好的理解用户需求；一方面向丽人、结婚等更多的行业进行横向覆盖；此外还将进一步扩展到用户决策的全链路，构建服务体验图谱，覆盖履约服务环节，分析其中的用户需求和反馈，更好地赋能商家提升用户体验。
- **更多的数据引入**：当前图谱的构建主要是以平台的用户和商户的文本语料为主，下一步将利用图像等更多模态的数据，并尝试引入外部的知识，对当前的节点和关系进行完善和补充。
- **更深的图谱应用**：现阶段图谱在搜索和推荐上的实践主要集中在标签及其关联供给的直接应用，后续考虑进一步深化图谱的应用，充分利用场景需求和场景要素的信息，为推荐侧更准的用户意图识别提供支撑，从而提升供给和用户的匹配效率，发挥出知识图谱更大的价值。

参考文献

[1] Maslow A H. A theory of human motivation[J]. Psychological review, 1943, 50(4): 370.

[2] Luo X, Liu L, Yang Y, et al. AliCoCo: Alibaba e-commerce cognitive concept net[C]. Proceedings of the 2020 ACM SIGMOD International Conference on Management of Data. 2020: 313-327.

[3] Devlin J, Chang M W, Lee K, et al. Bert: Pre-training of deep bidirectional transformers for language understanding[J]. arXiv preprint arXiv:1810.04805, 2018.

[4] Le Q, Mikolov T. Distributed representations of sentences and documents[C]. International conference on machine learning. PMLR, 2014: 1188-1196.

[5] Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need[J]. arXiv preprint arXiv:1706.03762, 2017.

[6] Shang J, Liu J, Jiang M, et al. Automated phrase mining from massive text corpora[J]. IEEE Transactions on Knowledge and Data Engineering, 2018, 30(10): 1825-1837.

[7] Clark K, Luong M T, Le Q V, et al. Electra: Pre-training text encoders as discriminators rather than generators[J]. arXiv preprint arXiv:2003.10555, 2020.

[8] Dozat T, Manning C D. Deep biaffine attention for neural dependency parsing[J].

arXiv preprint arXiv:1611.01734, 2016.
[9] Cheng H T, Koc L, Harmsen J, et al. Wide & deep learning for recommender systems[C]. Proceedings of the 1st workshop on deep learning for recommender systems. 2016: 7-10.
[10] Sun C, Huang L, Qiu X. Utilizing BERT for aspect-based sentiment analysis via constructing auxiliary sentence[J]. arXiv preprint arXiv:1903.09588, 2019.
[11] Sutton R S, Barto A G. Reinforcement learning: An introduction[J]. 2011.

作者简介

李翔、陈焕、志伟、晓阳、艳婷、旭乐、曹臻等，均来自美团到店平台技术部到综业务数据团队。

招聘信息

美团到店平台技术部 – 到综业务数据团队，长期招聘算法（自然语言处理 / 推荐算法）、数据仓库、数据科学、系统开发等岗位同学，坐标上海。欢迎感兴趣的同学发送简历至：licong.yu@meituan.com。

外卖商品的标准化建设与应用

作者：刘柳　懋地　崇锦　晓星

1. 背景及目标

商品作为外卖交易过程中的核心要素，决定了供需匹配的精准度，直接影响交易行为是否可以达成。外卖平台美食、甜点、饮品类在线商品有亿级之多，其中很多是属性信息一致的相同商品。建立对商品的标准化描述、聚合相同商品，是很多业务场景的诉求。

- **供销分析场景**：想分析一下望京的商家都售卖哪些菜品，有多少商家卖“西红柿炒鸡蛋”？
- **遇到的问题**：由于菜品是非标品，并且商家对菜品命名的个性化程度也较高，因此在外卖平台，同一个菜品名出现不同的命名方式；例如“西红柿炒鸡蛋”有西红柿炒蛋、小番茄炒蛋、西红柿鸡蛋、京城三绝 ~ 番茄炒蛋【正价小份菜】等，没有办法简单通过关键字进行聚合。
- **主题推荐场景**：想出一个菜品粒度的主题，快速筛选“小龙虾”、“烤鱼”、“鸡公煲”、“黄焖鸡”等热门菜品?
- **遇到的问题**：商品分类的颗粒度不够精细，无法快速找到适合颗粒度的菜品。
- **商家上单场景**：像“鱼香肉丝”这样普遍的菜品，每个商家上单都需要录入食材、口味、做法、菜系、荤素等标签，录入成本较高，能不能像淘宝一样，选择“iPhone 12”，它的属性就能够自动关联。
- **遇到的问题**：没有将菜品的属性标准化，菜品和属性之间没有关联关系。

基于上述业务应用的痛点，启动外卖商品的标准化建设。目标是建立商品的标准化名称，实现对相同商品的聚合，从而为业务提供合理粒度的概念划分，赋能运营端供销分析、用户端个性化召回排序、商家端标签生产。

2. 业界调研

对于业界的参考，主要参考淘宝标准化 SPU 建设。SPU 在淘宝体系中决定了商品是什么，是商品信息聚合的最小单位，由关键属性 + 绑定属性来构成。

- **关键属性**：用来约束和定义一个商品的，比如 iPhone X，决定他的就是“苹果”这个品牌和“X”这个系列。
- **绑定属性**：是关键属性的补充和细化，比如当 iPhone X 已经明确了这个产品后，其他的属性也确定，比如网络模型，屏幕尺寸等，进一步补充这些属性内容，逐步明确了一个产品。

可见淘宝对于 SPU 的建设，实际上是对属性的建设，例如格力空调 S1240，通过“格力”品牌、“空调”类目、“S1240”型号来标准化、唯一化。

但对于餐饮行业，对于核心属性食材“牛肉”、做法“炒”、口味“辣”，都无法确定是什么菜，更谈不上唯一化；但如果通过“小炒黄牛肉”来标准化，行业 / 用户对其有普遍的认知，较固定的食材口味做法，适合用来进行标准化。因此淘宝是标准化属性，而餐饮是标准化菜品名称，所以我们称之为标准菜品名。

3. 问题分析及挑战

淘宝的标准化主要针对标品，而餐饮标准化都是针对非标品，难度较大，面临着个性化问题、录入不规范、粒度无行业标准、认知局限性等挑战。

3.1 个性化问题

餐饮商家可以较低成本的自定义生产，个性化程度较高，同一个菜品在不同商家的命名可能不同，需要大量的同义词聚合，而同义词的召回是最大的难点（如何将潜在的同义词挖掘出来进行标注）。例如，京城三绝 – 番茄炒蛋【正价小份菜】、西红柿炒蛋（小份）、小番茄炒蛋、西红柿炒土鸡蛋（小份），都表示“西红柿炒蛋”这个商品。

3.2 录入不规范

商家在录入商品名称时，存在缺失关键信息的问题，例如“缤纷水果”是水果拼盘、

饮品还是披萨，“韭菜鸡蛋”是包子还是饺子。除商品名称外，需要借助商家分类，商品左侧栏 tag 等相关信息，对名称进行推理和补全。

3.3 粒度无行业标准

在进行标准化处理时，没有统一的标准，粒度难以把控：过粗容易产生非菜品错误（例如：“香辣鸡腿”–>“鸡腿”），过细则标准名内聚性偏弱（例如：“传统黄焖鸡【大碗】”本身粒度过细，需要提炼到“黄焖鸡”）。

3.4 认知局限性

中华美食文化博大精深，对于一些不为大众所熟知的小众或者地方特色菜品，需要具有一定的专业背景知识，例如“炒鸡”也是一个标准的名称，还并非商家没有填写完整。

4. 方案

商品标准化的整体方案如图所示：首先，基于美食、甜点饮品类商家全量在线的亿级商品，通过名称清洗、置信度判别、人工检验，获得近菜名主干；通过同义词挖掘，对主干名进一步聚合压缩，映射到标准名主词上；对于单个商品，进行名称纠错、清洗，通过模型匹配，建立商品 – 标准名的映射；为了满足不同业务场景的聚合粒度要求，通过上下级关系挖掘、深度遍历，进一步构建标准名层级树。分别对名称聚合、匹配映射、层级构建三个模块，涉及的算法模型进行介绍。

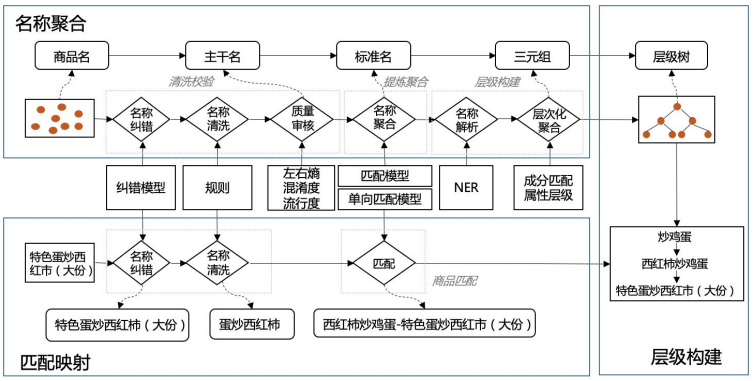


图 1　商品名标准化整体方案

4.1 名称聚合

清洗后的主干名仍然存在很多同义说法，比如土豆烧牛肉、牛肉烧土豆、洋芋烧牛肉、小土豆烧牛肉，表示相同的商品。目标通过挖掘这种潜在的同义关系，进一步提升名称的内聚性。在迭代过程中，先后采用了规则匹配和语义匹配的方法，挖掘潜在同义词；聚合后，根据流行度判别其中的主词，并将原始主干词映射至标准名主词上。分别对两种同义词挖掘方法介绍如下。

4.1.1 规则匹配

一期首先采用了规则匹配的方法，利用 NER 模型对主干名进行成份识别，结合知识图谱构建的属性同义词表，判别两个主干名是否是同义关系。

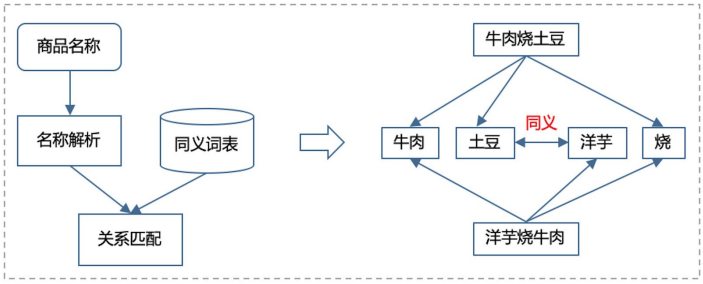


图 2　规则匹配

如图所示，其中“牛肉烧土豆”通过名称解析得到牛肉 – 食材，烧 – 做法，土豆 – 食材；“洋芋烧牛肉”通过名称解析得到洋芋 – 食材，烧 – 做法，牛肉 – 食材。对比两个主干名的成份词，其中土豆和洋芋是一对同义词，其余成份相同，进而获得二者之间是同义关系。

通过这种方式，挖掘了十万级同义词。根据标准名覆盖的商品供给数计算流行度值，将更流行度高的作为主词；人工校验后补充到标准名体系，提升了名称的聚合度。

4.1.2 语意匹配

由于规则匹配挖掘到的同义词有限，比如“担担面”和“担担汤面”，根据 NER 模型，担担面和汤面都会被识别成类目。如此，两个主干名是无法建立同义关系的。

我们二期调研了一些匹配模型，借鉴搜索算法组的经验，采用 BERT+DSSM 的语义匹配模型进行同义关系的扩覆盖。如图所示，首先基于一期积累的同义词，通过组内生成正例、跨组交叉生成负例的方式，构造百万级样本，训练了一版基础模型；为了进一步优化模型性能，通过主动学习和数据增强两种方式，对样本数据进行了迭代。

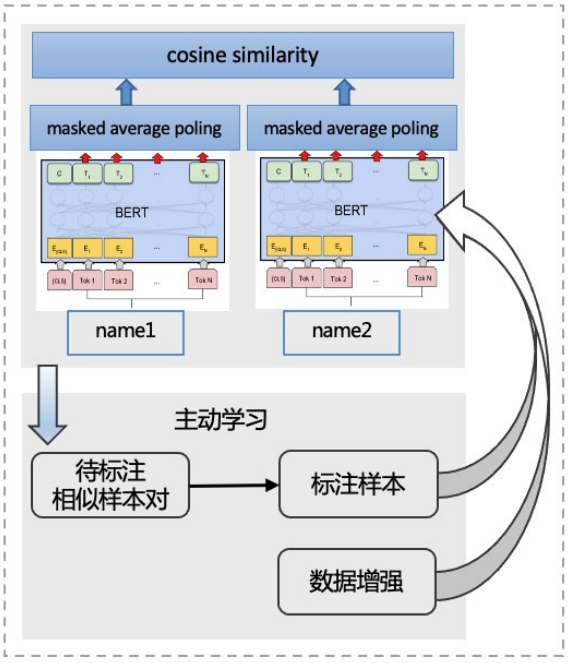


图 3　语义匹配

主动学习的方式是先利用基础模型，圈定一批待标注的相似样本，交与外包标注，将标注正确的样本补充至已有同义词中，标注错误的样本作为负例加入训练集，用于模型的优化迭代。通过主动学习的方式，补充了万级样本，模型准确率取得了明显提升。

进一步分析结果，我们发现了一批很有特点的 Bad Case，比如红烧狮子头和红烧狮子头盖饭、香椿拌豆腐和拌豆腐等，它们都属于字面相似度很高，但核心成份不同的匹配错误。基于这个特点，先根据字面距离圈定了一批字面相似度高的样本，再利用名称解析模型对它们进行成份识别，找出其中的负例。通过这种方式，在不增加标注成本的情况下，自动补充了十万级样本，进一步提升了模型准确率。

利用语义匹配模型，新增了十万级同义词，进一步提升了标准名的内聚性。

4.2 匹配映射

在挖掘到的标准名词表和同义词基础上，为亿级在线商品建立“商品 – 标准名”映射（如“招牌蛋炒小番茄（大份）”映射到“西红柿炒鸡蛋”），以实现对相同商品的标准化描述与聚合。采用“文本 + 图像”相结合的匹配模型，覆盖了绝大部分的美食、甜点饮品类商家的在线商品。

4.2.1 文本匹配

文本匹配流程如图 4 所示，整体上包括召回、排序两个阶段。首先，对商品名称中的规格、分量等描述信息进行清洗，将清洗后的商品名和标准名进行 2–Gram 切片，通过关联相同切片的方式召回待匹配的标准名；基于召回的标准名，通过计算 Jaccard 距离，保留其中 Top 20 的标准名；在此基础上，利用 BERT 向量化模型，生成商品名和标准名的向量表示，通过计算 Jaccard 字面距离以及 Cosine 向量相似度，获取其中综合得分最高的标准名。

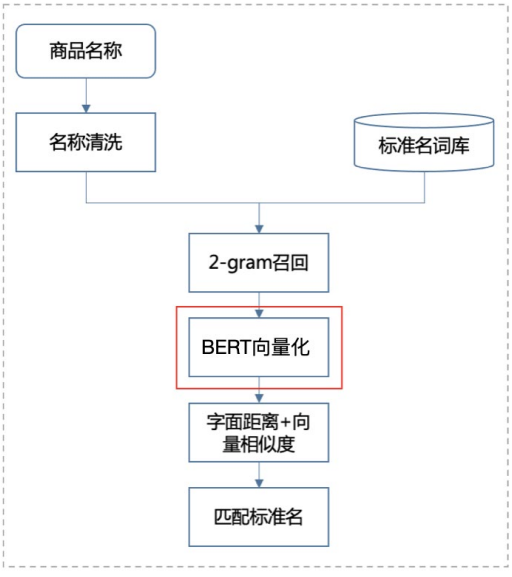


图 4　文本匹配流程

其中，BERT 向量化模型是基于上文提到的同义语义匹配模型，通过级联一维类型编码的方式，对标准名和商品名加以区分，改造成非对称的标准名匹配模型。之所以进行这一改造，是因为与同义匹配不同，标准名匹配是非对称的，如应该将“香锅盔”匹配上相对抽象的标准名“锅盔”，而不是匹配上一个更加具体的标准名“五香锅盔”。改造后，匹配准确率提升显著。

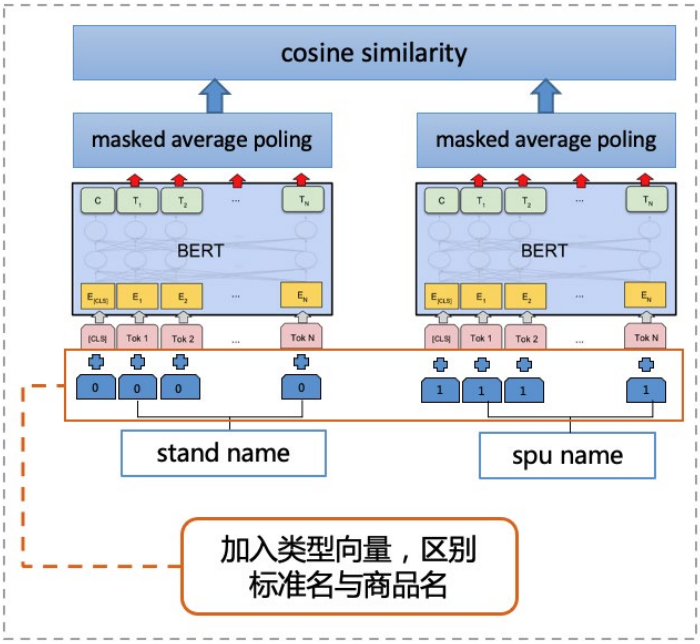


图 5　BERT 向量化模型

4.2.2 图像匹配

由于菜品名称长度有限及商家命名不规范，会导致仅从菜品名称中获取的信息有限，而无法建立到标准名的匹配。通过引入商品图片信息，提升对文本信息不全商品的匹配准确和覆盖。

图像匹配采用的是多分类模型，根据标准名层级聚合（详见 3 层级构建）后的顶级、二级，选择待匹配的标准名标签，并根据文本匹配结果构造样本集。由于采用大规模非人工标注样本，不可避免地要解决样本噪声问题。在本场景下，噪声主要有两个主

要来源：一是文本信息不全，导致样本标签错误；二是由于顶级、二级聚合程度高，导致分类粒度过粗，需要细分多个标签。针对这些问题，采用样本和模型迭代优化的方式，根据基础样本集训练初版模型，利用模型挖掘噪声数据，人工校验后进行模型微调。如此迭代，实现低标注成本的模型优化。

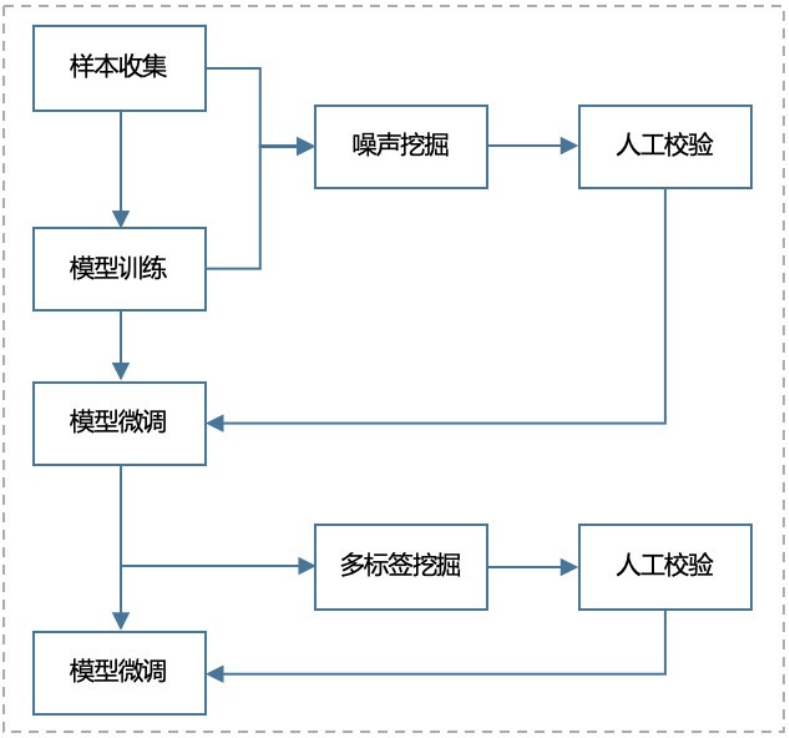


图 6　图像模型迭代流程

图像分类模型选取了对 MBConv 模块的参数进行精细化调整的 Basebone 网络 Efficientnet，通过调整网络的分辨率、深度、宽度，确定最优组合。噪声挖掘方法首先通过 Metric-Learn 的方法，学习获得每个类别的聚类中心，及类内样本与聚类中心距离的均值、方差、中位数，对其进行排序挖掘出类内离散度较大的类别；再借助分类模型在验证集上的预测、O2U-Net 和 Forgetting Event 挖掘样本噪声。通过上述方法优化模型，提升对噪声样本的鲁棒性。

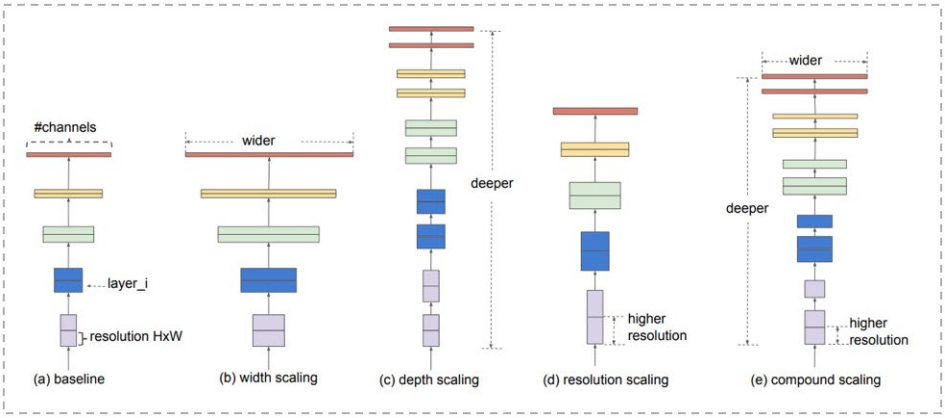


图 7　Efficientnet 网络宽度、深度、输入分辨率的变化组合

4.3 层级构建

推荐场景下，为了保证用户的个性化和多样性体验，需要对商品进行合理粒度的聚合。对于商品列表排序场景，现有的类目过粗会导致多样性不足，标准名过细又会导致结果重复。目标是建立一个层级化商品体系，为业务提供合理的聚合粒度。通过关系挖掘、层级遍历，构造万级顶点的层级树，支持了商品列表、美食排行榜、交互式推荐等业务的上线和优化。构建方法包括规则匹配和模型判别，分别对这两类方法进行介绍。

4.3.1 规则匹配

规则匹配方法是基于已有的 NER 模型和属性词表，通过结构化匹配的方法，挖掘到十万级上下级关系，进一步遍历生成万级顶点的标准名层级树。这种方法比较简单且基于已有工作，开发周期短，在项目初期快速支持上线，并取得了明显的业务收益。

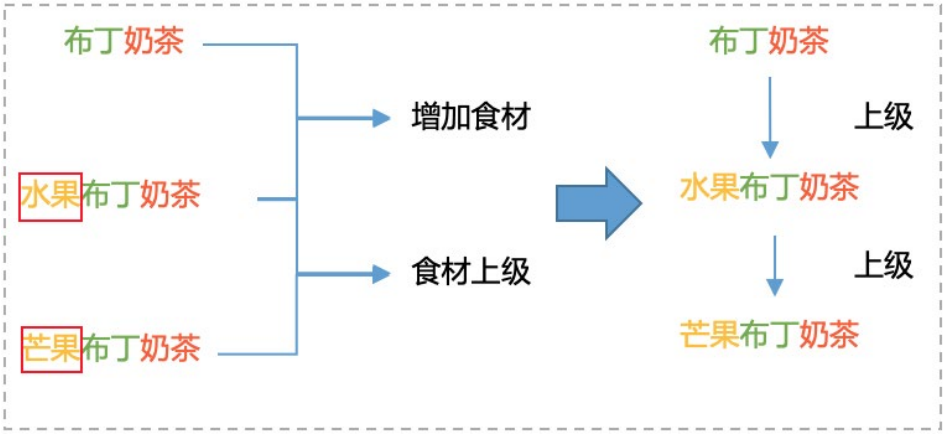


图 8　规则判别

4.3.2 模型判别

规则匹配方法由于 NER 模型错误和属性词关系缺少，导致挖掘到的关系有限，需要通过判别模型进一步提升泛化性。基于 BERT 的关系分类模型如图 8 所示，对待分类一对标准名用 [SEP] 进行拼接，并在开头增加 [CLS] 标识符；将拼接结果编码后，传入 BERT 模型，取出 [CLS] 位的 Embedding；再接一个全联接层和 Softmax 层，输出关系分类结果。标准名关系包括：同义、上级、下级、无关系，一共四个类别。

样本数据包括简单例和难例两部分，其中：简单例基于已有同义词、上下级，以及同义词组间交叉生成无关系，一共构造百万级样本；在此基础上，进一步利用已有的向量化模型，召回相似度较高的标准名对，交与外包标注其类别。第二类样本更加贴近实际分类场景，且属于混淆度较高的分类难例。

利用第一类样本预训练初版模型，并在此基础上，利用第二类样本对模型进行微调，进一步提升了分类模型准确率。经人工校验，进一步补充了万级词关系。

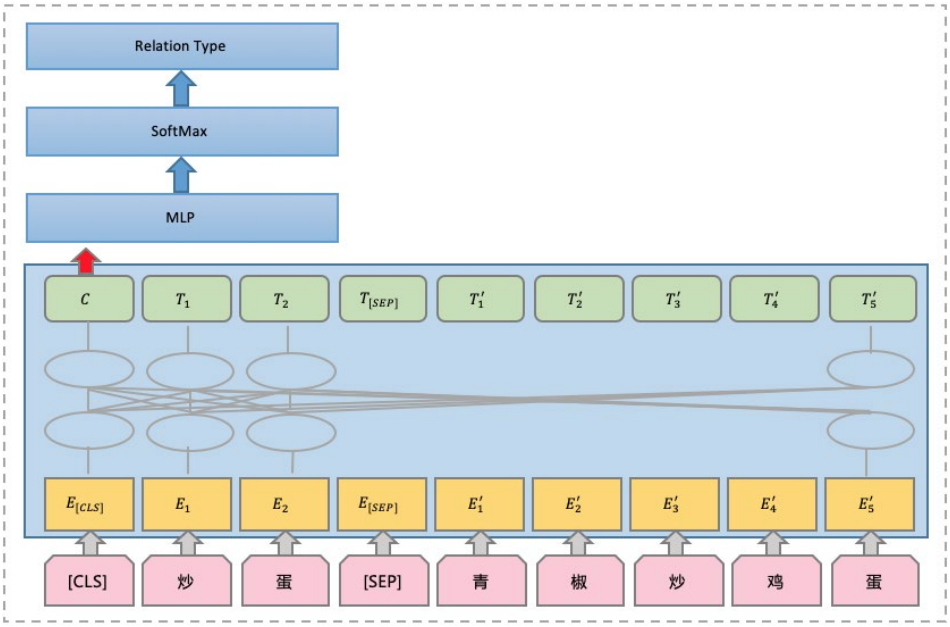


图 9　分类模型

5. 在外卖业务中的应用

标准名作为品类和商品的中间层，为业务提供了更加丰富、合理的聚合粒度，支持了流量转化的策略优化，以及系列产品形态的开发上线。商品列表通过接入标准名层级，实现对商品合理粒度的聚合，解决了线上商品重复的问题；标准名作为基础数据，支持了美食排行榜的开发和上线，帮助提升用户的决策效率；针对用户当前访问的商品，利用标准名召回相关商品，实现交互式推荐。标准名作为重要的基础数据，支持了产品形态多元化、推荐策略优化，对于提升用户粘性、流量转化，搭建商家用户友好的平台生态，具有重要的价值和意义。

图 10　标准名相关应用

6. 总结与展望

目前已完成基本的体系建设，并成功应用于不同的场景、取得业务收益。已经建成的标准名层级体系，覆盖了外卖绝大多数的在线商品。标准名作为基础和特征数据，应用到用户端的商品列表、美食排行榜等业务场景，支持策略优化、带来流量收益；同时，也通过服务接口的形式，跨部门支持 SaaS 点餐推荐的模型优化，以提升用户体验和业务收益。

标准名作为重要的商品特征数据，应用的业务场景广泛。在未来的工作中，需要持续迭代优化，保证标准名自身的准确性和质量；同时加深业务理解，根据业务需要优化层级体系，为业务提供更加合理的聚合粒度，提升转化收益；此外，重点建设一批用户感知强、供给覆盖高的标准名，降低业务方的接入成本和提高收益。

6.1 词表及同义词优化

标准名词表体量大、同义关系复杂，词表建设不是一蹴而就的，需要长期的迭代和优化。针对标准名词表中存在的非菜品、不规范等错误，以及同义词聚合不足、过度的问题，采用算法圈定 + 人工标注的方式，通过模型挖掘潜在同义词、圈定问题词组；人工校验后，进行批量补充和修改，持续优化词表、同义词。

6.2 层级结构合理化

目前标准名层级是通过规则和模型的方式直接生成的，人工参与度较低，与业务场景的结合不够深入。后续会结合业务需要，明确剪枝、聚合粒度准则，合理化层级结

构，优化层级的合理性。从而更加灵活、高效地支持不同的业务应用，提升落地效果。

6.3 核心标准名建设

标准名有 20 万之多，为业务应用造成一定的选择成本、带来不便。结合业务需要，圈定其中用户感知强、供给覆盖高、优质的标准名，作为核心标准名进行重点建设。核心标准名轻量且精炼，更加贴近业务需要，可以帮助业务方降低接入成本、提高收益。

7. 参考文献

[1] Devlin J, Chang M W, Lee K, et al. BERT: Pre-training of deep bidirectional transformers for language understanding[J]. arXiv preprint arXiv:1810.04805, 2018.

[2] Lample G, Ballesteros M, Subramanian S, et al. Neural architectures for named entity recognition[J]. arXiv preprint arXiv:1603.01360, 2016.

[3] Chen Q, Zhu X, Ling Z, et al. Enhanced lstm for natural language inference[J]. arXiv preprint arXiv:1609.06038, 2016.

[4] Huang P S, He X, Gao J, et al. Learning deep structured semantic models for web search using clickthrough data[C]//Proceedings of the 22nd ACM international conference on Information & Knowledge Management. 2013: 2333-2338.

[5] Tan M , Le Q V . EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks[J]. 2019.

8. 作者简介

刘柳、懋地、崇锦、晓星等，均来自美团外卖技术团队。

多业务建模在美团搜索排序中的实践

作者：肖垚　培浩

引言

美团的使命是“帮大家吃得更好，生活更好”，美团 App 承载的业务包括外卖、到店餐饮、买菜、优选、酒店、旅游、休闲娱乐等各类生活服务。美团首页的搜索是美团 App 上各类生活服务最大的流量分发入口，每天为数千万的用户提供各种服务。美团搜索排序是一个典型的多业务混合排序建模问题，一个典型的多业务搜索场景是当用户搜索地点，如“望京”的时候，用户的需求不是很明确，此时搜索的结果页如下图 1 所示，下方的商家列表中会包含望京附近餐饮、电影、休闲娱乐、酒店等多种业务的结果，这就是一个多业务混合排序场景。

图 1　美团搜索结果页面

而多业务场景存在如下几点挑战：

- 因业务之间存在共性和特性，如何让模型兼顾这两种特性，实现更好的数据学习。比如到店餐饮对距离特征非常敏感，而旅游景点业务对距离特征相对不敏感。
- 业务天然存在高频和低频特性（比如外卖和旅游），导致模型的训练数据中多业务样本数量不平衡。
- 各个业务往往有自己不同的主目标，如何满足不同业务的目标，最终能够提升搜索的用户体验。

本文分享了美团搜索中的多业务排序建模优化工作，我们主要聚焦在到店商家多业务场景，后续的内容会分为以下四个部分：第一部分是对美团搜索排序分层架构进行简单介绍；第二部分会介绍多路融合层上的多业务融合建模；第三部分会介绍精排模型的多业务排序建模；最后一部分是总结和展望。希望能对从事相关工作的同学有所启发或者帮助。

排序流程简介

美团搜索系统流程如下图 2 所示，整体流程分为数据层、召回层、排序层和展示层。其中排序层分为以下几个子部分：

- **粗排层**：使用相对简单的模型对召回候选集进行初步过滤，实现排序效果和性能的 Trade-off。
- **多路融合层**：使用查询词特征、上下文场景特征构建配额模型，进行不同业务候选集的数量控制，实现用户需求的精确理解。
- **精排层**：使用亿级别特征的深度学习模型，捕捉各种显式和隐式信号，实现 Item 排序分数的精准预估。
- **重排层**：使用小模型和各种机制对精排后的结果进行调序，实现精细定向的优化。
- **异构排序层**：使用深度学习模型对异构聚块进行排序，实现多业务的高承载。

多层排序架构设计主要是为了平衡排序效果和性能。本文后续提到的多业务建模优化工作主要从多路融合层和精排层进行介绍。

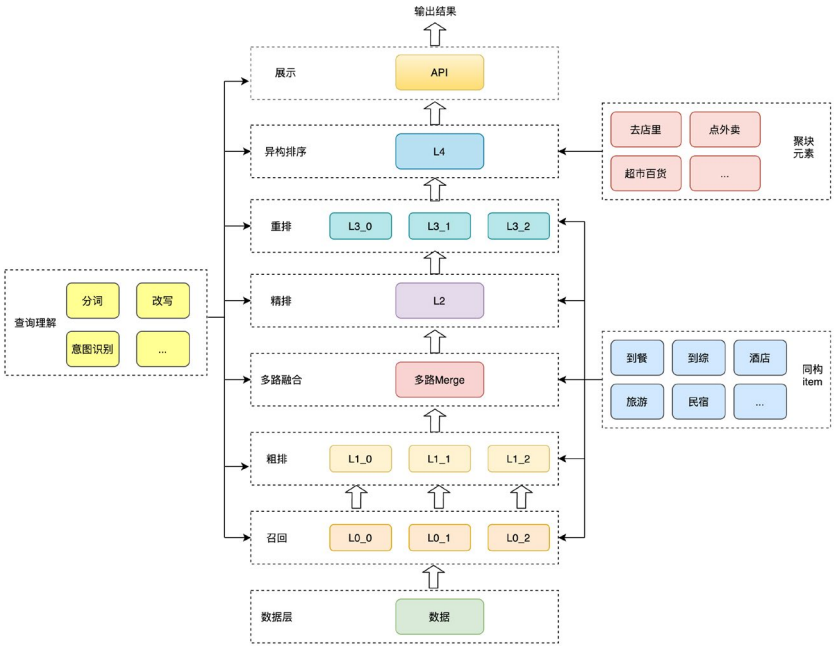


图 2　系统架构图

多业务建模实践

多业务配额模型（多路融合层）

随着美团业务的发展，美团搜索接入了到餐、到综、酒店、旅游等业务。对于业务意图模糊的搜索词，比如用户搜索“五道口”，需要根据用户、查询词、场景等多种因素来综合判断用户的业务意图。为了融合不同业务的召回结果，给 L2 精排一个比例合适的候选集，我们设计了一个多业务配额模型来平衡多业务召回的比例。这种基于配额对多路召回结果进行合并的做法在搜索、推荐场景中十分常用，比如淘宝首页搜索、美团推荐等。

为了多路召回的灵活接入，适配美团搜索业务的发展，我们不断迭代搜索配额模型。

下面将详细介绍美团多业务配额模型的迭代过程，文章后续部分会将多业务配额模型（Multi-Business Quota Model）简称为 MQM。

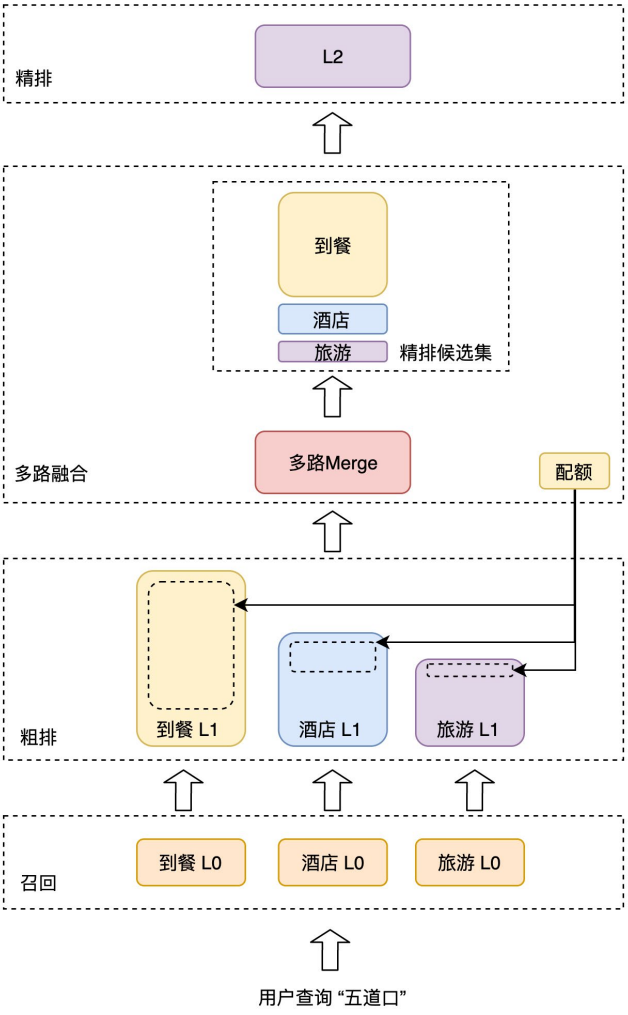


图 3　多路融合过程

一维目标多业务配额

考虑到大搜结果存在多路不同业务的召回，为了刻画用户搜索 Query 对三路业务召回的意图强弱，我们采用多目标的建模方式，以每一路召回是否被点击、下单为

目标进行建模，实现了多业务配额初版模型 MQM-V1。该模型输出各路召回的点击、下单联合概率，作为最终的配额分布。在特征层面，我们使用 Query 维度特征、Context 维度特征、Cross 维度特征、User 维度特征，来刻画用户在不同场景的实时个性化需求。MQM-V1 模型结构如下图 4 所示。

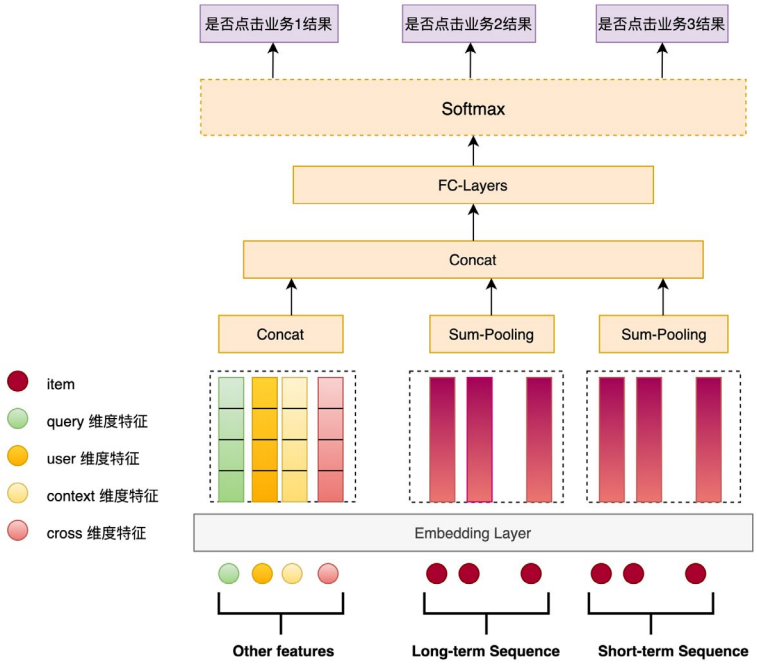


图 4　MQM-V1 模型结构图

MQM-V1 版本上线后，整体线上点击率 +0.53%，各业务访购率基本持平。

二维目标多业务配额

随着大搜召回策略的不断迭代，大搜不仅引入了按业务拆分的召回方式，而且引入了向量检索、地理位置近邻检索等跨多业务的异构召回方式，导致大搜召回策略不断增加，多业务配额模型也面临新召回源带来的冷启动问题。同时，为了加强多业务配额模型的个性化，我们参考借鉴了 [6] 中用户行为序列建模的方法。综上，该版本 MQM-V2 与 MQM-V1 区别如下：

- 建模目标上，从以召回方式点击的一维目标升级到召回方式叉乘业务的二维目标，使得多路融合的粒度也更细、精度更高。
- 行为序列建模模块引入 Transformer Layer。
- 为了解决新召回源接入的冷启动问题，我们引入了人工经验层，包括业务先验和历史统计，综合模型输出决定每一路召回的配额。

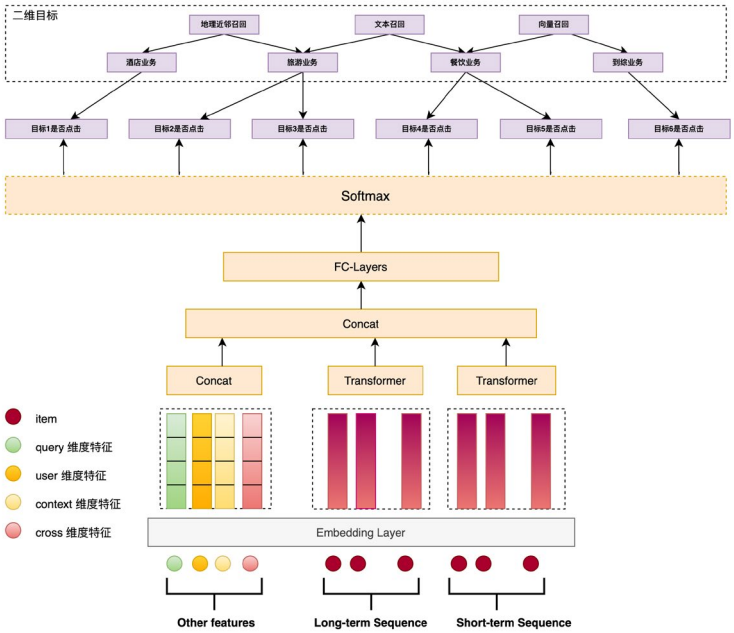


图 5　MQM-V2 模型结构图

MQM-V2 版本上线后，各业务指标率均有提升，其中旅游访购率 +2%，到餐访购率 +0.57%，到综、酒店访购率持平。

多业务排序模型（精排层）

从美团搜索精排模型升级为 DNN 模型，一直到 2019 年底，美团搜索的精排模型结构是业界主流的 Embedding&MLP 的范式结构，期间我们也尝试过业界提出的模型结构比如 PNN[1]、DeepFM[2]、DCN[3]、AutoInt[4]、FiBiNet[5] 等等。

随着迭代的进行，我们发现针对特定业务的优化难以在精排模型发挥作用，为了兼顾

各个业务的特性，支持各个业务更加有效的针对性迭代优化，需要探索出一种模型结构来适配美团搜索这样的多业务场景。下面会具体介绍精排模型在多业务建模上的发展史，文章后续部分将多业务精排模型（Multi-Business Network）简称为 MBN。

独立子网络拆分

考虑到酒店和旅游在美团大搜排序策略的流量里面占比较少，而针对小流量的相关优化在目前统一的 Embedding&MLP 模型结构里面很难体现，我们尝试了如图 6 所示的人工自定义多塔模型 MBN-V1 结构：主网络复用目前的模型结构。具体情况介绍参考 [6] 中的行为序列建模部分，增加酒店和旅游独立子网络；酒店子网络的输入包括酒店独有特征和主网络的打分输出，旅游子网络的输入包括旅游独有特征、主网络的打分输出、主网络最后一层 FC，酒店和旅游子塔输入不同是因为业务逻辑不同导致数据分布差异大，这是实践出的结果，最终的输出是对三个输出的加权求和。

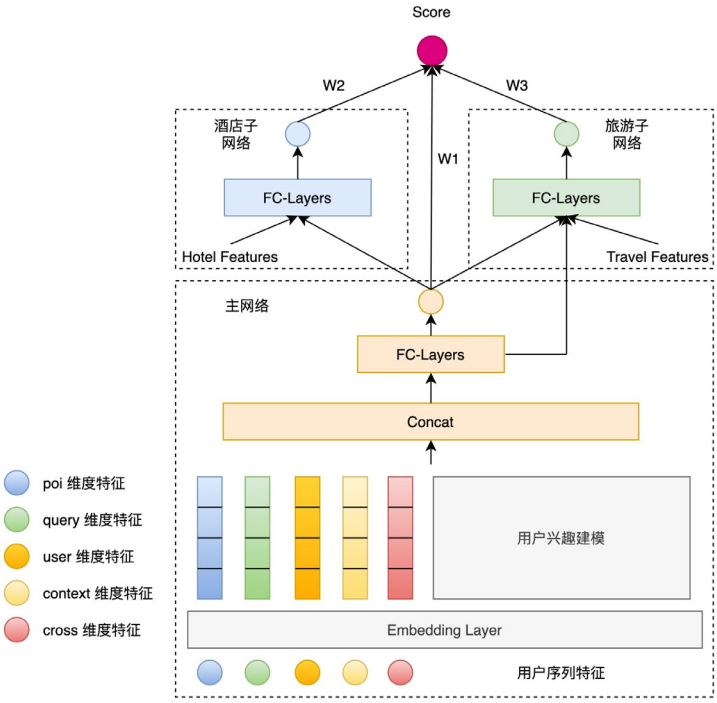


图 6　MBN-V1 模型结构图

针对加权求和的权重部分，我们采取了两种方式对权重进行设定：

- 第一种，采用硬切分的方式，就是说权重向量是一个 One-hot 稀疏向量：对酒店商家进行预测，只选取酒店子网络的输出，其余类推。
- 第二种，采用软切分的方式，把多业务配额模型的输出作为权重值。

线上实验发现第二种方法比第一种好，我们认为采用硬切分会导致子塔分支的参数只能被对应业务的数据更新，而各业务的数据占比不均导致学习不佳，而软切分会达到一种知识迁移的作用。最终线上的效果相比统一的 Embedding& MLP 模型，整体旅游取得了正向效果：其中整体点击率 +0.17%，其余业务访购率效果基本持平。

子网络权重自学习

基于第一版多业务精排模型取得了初步正向效果，我们继续添加美食业务子塔，同时考虑到 MBN-V1 依赖配额模型的输出，这样会导致配额模型的变更可能会对精排模型的效果产生影响，针对这些因素我们上线了第二版多业务模型 MBN-V2，模型结构如图 7 所示。相比 MBN-V1 的区别如下：

- 添加美食业务的独立子网络。
- 解耦精排模型和配额模型，在精排模型中集成权重生成子网络，该子网络的输入主要是一些 Query 维度，Context 维度的特征。

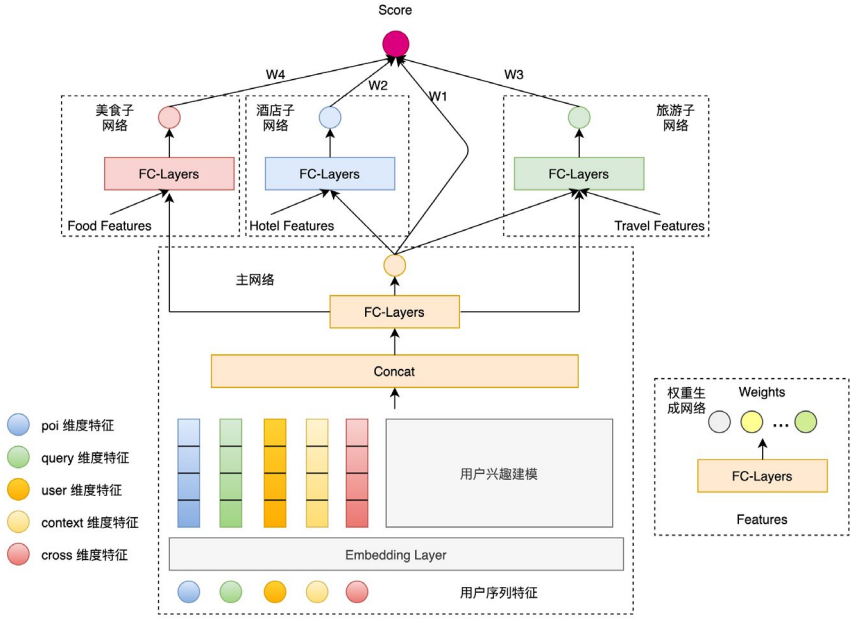


图 7　MBN-V2 模型结构图

线上实验效果：MBN-V2 相比 MBN-V1，整体点击率 +0.1%，业务访购率效果基本持平。

子网络特征自适应

在第二版模型的基础上，我们进一步添加到综业务子塔，随着子网络的变多，目前对于子网络的输入采用人工设计，这种方式需要花费大量的时间进行离线实验。考虑到目前的多业务子塔结构十分类似业界的多任务学习，我们也尝试引入业界的多任务学习结构；同时，我们针对 MBN-V2 中的权重子网络输出进行分析发现其输出的权重对不同业务商户的输出差不多，那么会带来业务的针对性优化不明显。基于上述部分，我们迭代了第三版多业务精排 MBN-V3，结构如下图 8 所示，改进点如下：

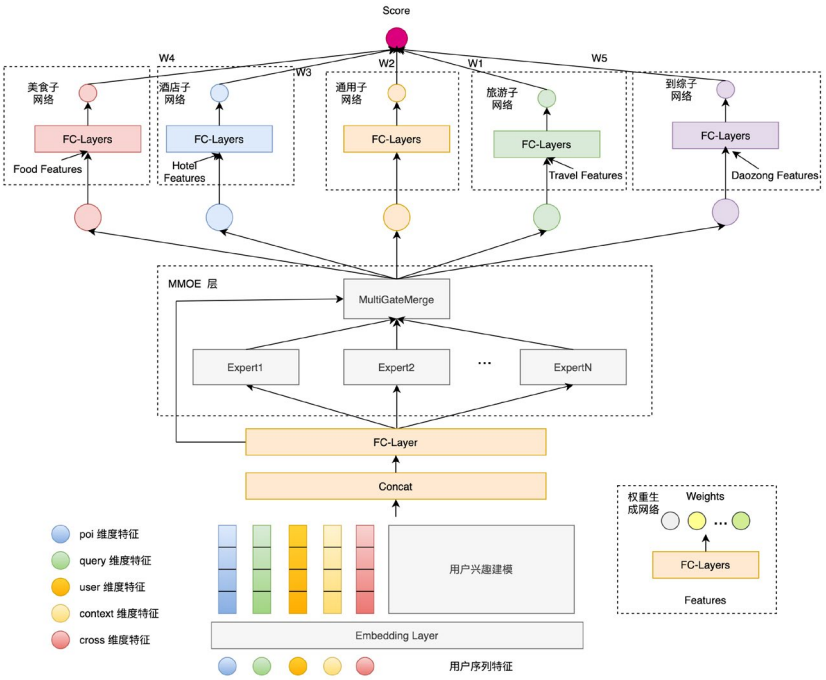


图 8　MBN-V3 模型结构图

- 补充到综子网络，采用 MMoE[7] 多任务学习结构，来自动学习特征表征输出给上层子网络，从而取代人工设计子网络的输入。
- 精排模型的损失函数除了采用用户线上反馈计算的主 LambdaLoss 外，额外添加了业务的分类交叉熵 Loss，达到预测某业务 Item 得分时，对应的业务子塔权重最大的目的。

线上实验效果：MBN-V3 相比 MBN-V2，整体点击率效果持平、美食业务访购率 +0.36%，到综业务访购率 +1.07%，酒店业务访购率 +0.27%，旅游业务访购率 +0.35%。

多业务特征表达优化

虽然 MMoE 多任务学习结构在业界很多场景得到了应用，在我们的多业务建模场景也取得了有效验证，但是我们持续跟进业界前沿，并且结合业务场景进行落地。

我们尝试了腾讯提出的 PLE[8] 结构，迭代出多业务精排 MBN-V4。 PLE 可以看成

是 MMoE 的改进版，它对于每一个任务有自己特定的专家层，不同任务之间有共享的专家层，相比 MMoE 是所有专家输出的加权求和，PLE 子任务的输入是子任务独有的专家和共享专家输出的加权求和，更容易学出业务的特性；同时基于性能考虑，我们选取了单层 PLE 也即 CGC 结构，结构如下图 9 所示：

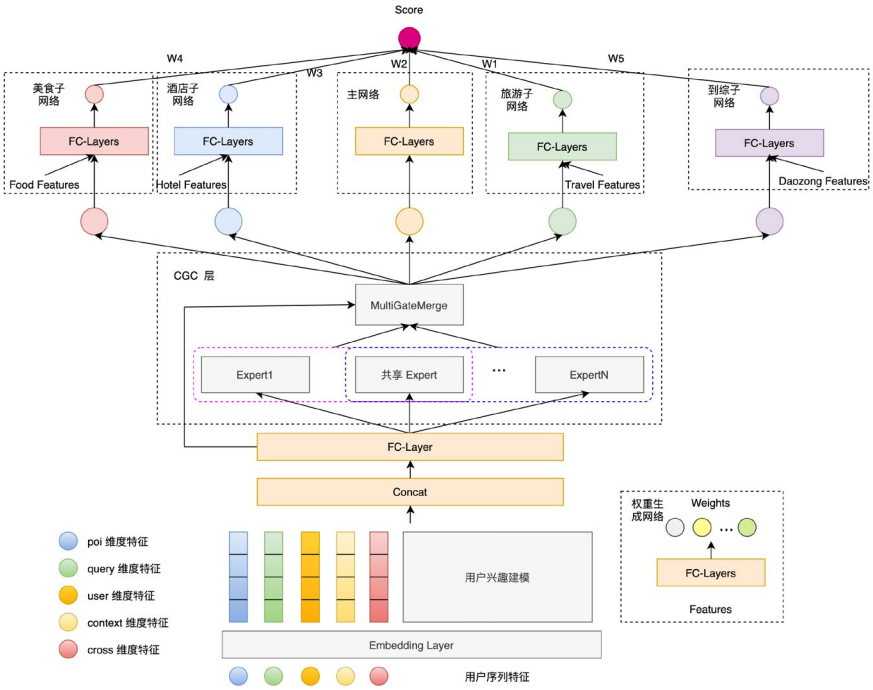


图 9　MBN-V4 模型结构图

线上实验效果：MBN-V4 相比 MBN-V3，整体点击率 +0.1%，美食业务访购率 +0.53%，其余业务访购率波动持平；我们对 MMoE 和 CGC 的专家权重进行可视化如下图 10 所示，分析发现：CGC 结构的专家层权重相比 MMoE 在同一业务多条样本之间的专家权重方差更小，更加稳定，说明 CGC 相比 MMoE 在特征表示上更加有优势。

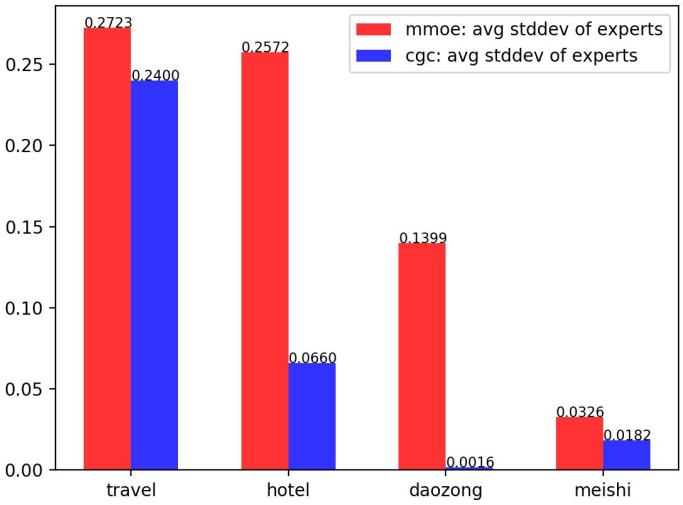


图 10　MMoE、CGC 专家权重分析

总结和展望

2019 年底开始，为了解决实际的多业务召回排序问题，美团搜索进行了大量探索，从工程到算法到产品形态各个层面丰富对多业务的支持。其中排序算法层面主要在多路召回融合层和精排层进行优化。

多路融合层主要完成搜索结果从结果相关到结果优质的筛选过程，需要解决不同召回方式（文本召回、推荐召回、向量召回）和不同业务召回结果的融合截断问题，直接决定了用户能浏览到的结果候选集。这其中最重要的问题就是判断用户对各个业务的需求强弱以及各业务的召回质量，对每个业务结果和召回结果确定合适的精排准入标准。

多业务配额模型通过整合用户实时需求、Query 的历史统计信息、搜索上下文信息和每个召回源质量情况，给出了每路召回、每个业务的应进入精排的比例。该模型保证了不同场景下精排候选集的结果多样性和优质性，实现了新业务、新召回方法接入的少侵入性，降低业务、召回接入成本。同时也为精排层分业务的网络结构提供了融合各个业务子网络结果的先验权重。

精排层在多路融合层的基础上进一步对多业务搜索结果进行精细化排序建模打分。用户的需求和美团业务一样是多样的，为了能充分建模各种场景下的需求，精排多业务排序模型从底层数据（丰富分业务的特征）、模型结构到业务目标融合都进行了多轮迭代。其中模型结构和相应的目标融合直接对各种大小业务、场景和相应的业务目标进行了分片建模，有效地缓解了小业务小场景在统一建模中被大业务样本淹没的问题。同时该模型支持新老业务的快速迭代，各业务可以方便独立地迭代特征、模型结构和相应的目标。

上述优化覆盖了线上全流量，在搜索用户体验和各业务价值均有明显提升，但还有很多工作可以持续优化。

- **业务独有特征利用**：目前我们采用对某些业务添加业务独有特征，其余业务对于这些缺失的独有特征给默认值，但是这样会带来很多冗余的计算量，这部分不管从效果还是性能方面都存在优化空间。
- **样本不平衡学习**：不同业务的数据量在美团搜索上差异大，如何让模型更好的学习出小业务的分布，我们正在探索迁移学习和 Meta-Learning 等方法。
- **多目标优化**：美团搜索既要兼顾用户的搜索体验，也要服务美团各个业务的战略目标，所以各个业务的主优化指标不一定一致，多目标优化也是一个持续探索的方向。

本文叙述的工作集中在美团多业务商家搜索排序上，同时随着优选、买菜、团好货、闪购等商品类业务的发展，我们也正在进行商品类多业务混排以及商家商品异构多业务混排工作。

参考资料

[1] Product-based neural networks for user response prediction
[2] DeepFM: A Factorization-Machine based Neural Network for CTR Prediction
[3] Deep & Cross Network for Ad Click Predictions
[4] AutoInt: Automatic Feature Interaction Learning via Self-Attentive Neural Networks
[5] FiBiNET: Combining Feature Importance and Bilinear feature Interaction for Click-Through Rate Prediction

[6] Transformer 在美团搜索排序中的实践
[7] Modeling Task Relationships in Multi-task Learning with Multi-gate Mixture-of-Experts
[8] Progressive Layered Extraction (PLE): A Novel Multi-Task Learning (MTL) Model for Personalized Recommendations

作者简介

培浩、肖垚、晓江、家琪、陈胜、云森、永超、利前等，均来自美团平台搜索与 NLP 部。

美团搜索多业务商品排序探索与实践

作者：曹越　瑶鹏　诗晓　李想等

引言

美团的使命是“帮大家吃得更好，生活更好”，我们给广大消费者提供买菜、优选、闪购、外卖、到店餐饮、酒店旅游、休闲娱乐等各类商品和服务。首页搜索是美团App 上十分重要的模块，每天服务于数千万用户。随着美团零售商品类业务的不断发展，美团搜索在商品类业务上的相关技术也在不断迭代，排序模块作为整个搜索系统的重要组成部分，极大地影响着用户最终的搜索体验。

近些年，深度学习在排序领域得到广泛的应用，本文分享了美团搜索在零售商品类业务上的排序经验。内容主要分为以下三个部分：第一部分，我们对商品搜索多业务排序面临的挑战进行简单的介绍；第二部分会介绍商品搜索相关排序技术的一些实践经验；最后一部分是总结与展望。

搜索排序的挑战

美团搜索多业务商品排序，面临的挑战主要包括以下几个方面：

1. 各业务供给和履约差异较大，采用统一模型对各类业务商品进行统一的混合排序建模，模型兼顾不同业务的共性和特性难度大。
2. 不同业务的消费频次和业务规模不同，导致各业务在模型训练数据中的分布差异较大。
3. 排序结果进行业务聚合展示情况下，业务间的顺序和首位业务结果展示数目共同影响着用户的搜索体验，两者独立优化只可能得到体验目标的次优解，由于两者的任务差异大，进行联合优化难度较大。

下面主要介绍我们在商品多业务排序上针对上述挑战点所做的一些探索与实践。

排序探索与实践

混排建模

在美团搜索商品多业务排序场景下，排序需要对不同商品业务：闪购、买菜、优选、团好货等进行统一混合排序，不同业务之间既有共性也有自己的特性。下面介绍我们针对商品多业务排序相关的一些实践。

图 1　多业务商品排序示意图

多子塔结构

此前，我们在商家多业务排序场景中已经积累了很多的实践经验，详情可参考《多业务建模在美团搜索排序中的实践》一文，所以直接将多子塔结构应用到商品多业务

排序场景。商品混排的 item 集合涵盖五个业务，模型结构中具有五个子塔：闪购子网络、买菜子网络、外卖子网络、优选子网络和团好货子网络；与此同时，我们引入 ESMM[2] 模型结构来更好的学习下单信息。整体模型结构如下图 2 所示：

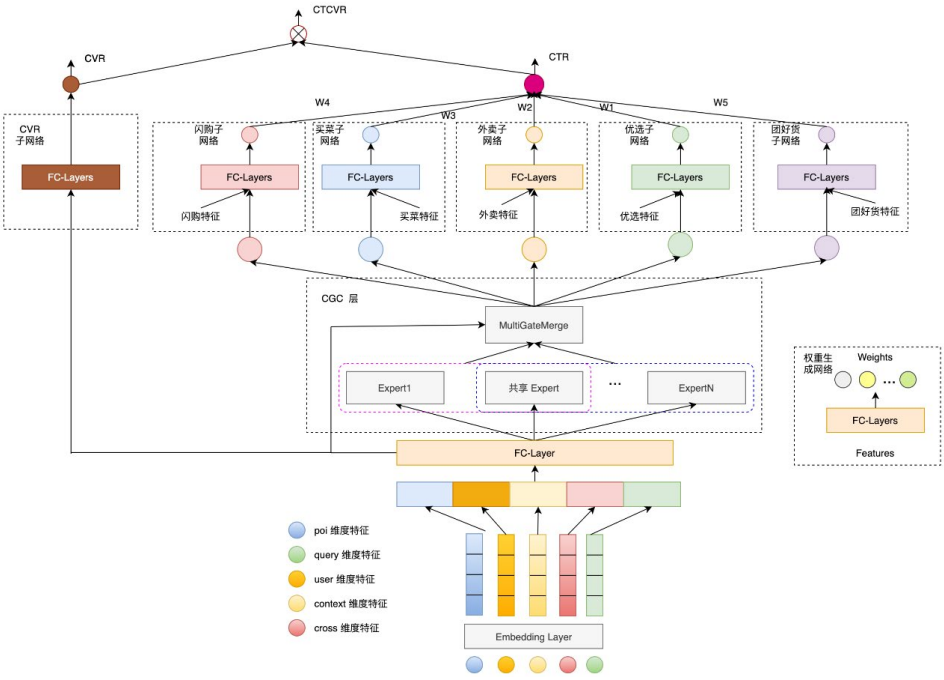


图 2　多子塔模型结构示意图

实践表明，模型离线指标相比基线提升明显；经过线上 AB 实验测试，点击率提升了 20BP（基点），访购率提升了 37BP。

多业务特征选择

目前在多业务子塔结构中，我们采用了一个大的特征候选集来兼顾各业务的需求：底层特征输入是闪购、买菜、外卖、优选、团好货多个业务的特征并集，业务之间互相缺失的特征给予默认值。这种实现，导致特征的数量繁多且杂乱，可能会给模型的训练带来很多的噪声。例如对于优选业务，实际可能只有 10% 的特征起到作用，剩下 90% 的特征对这个业务来说就是噪声。

目前，业界提出的相关特征选择方案，如 AutoFIS [20]、AutoFeature [21] 等等，都是针对单个场景、单个模型来选择的，而我们的目的是针对一个特征集合，让模型学会不同场景下特征重要性的不同。因此，这些方案无法直接应用到我们的场景中。为了缓解上述提到的问题，我们初步尝试了一些方法用于多业务的特征选择。

Group Lasso 正则化

Group Lasso [3] 是 Lasso 的一个扩展。Group Lasso 首先对特征进行分组，然后对组内的特征采用 L2 正则化，组间的特征集合采用 L1 正则化。由于 L1 正则化的稀疏性，因此最后产出的特征集合是稀疏的，也就达到了特征选择的目的。形式化表述，我们在训练目标中加入下述公式（1）正则项：

$$\min \mathcal{L}_{grls} = \lambda_1 \sum_{k=1}^{K} || \theta_k ||_2 + \lambda_2 || \theta ||_1 \quad (1)$$

其中 K 为特征组的数目，θ_k 为第 k 组特征的 Embedding 矩阵，λ_1 和 λ_2 为权重超参数。在上式中，我们对所有特征集合应用 L1 正则化，从而产出一个稀疏的解，也就达到了我们特征选择的目的。

该版模型相比多子塔模型，离线 NDCG 基本持平。

特征选择门

在上述 Group Lasso 方案中，虽然我们达到了特征选择的目的，但选出的特征对应不同业务的权重是相同的。在多业务排序场景下，不同特征对不同业务的重要性显然不一样，例如“配送时间”这个特征对闪购业务比较重要，但对于团好货影响不是很大。为了分业务建模特征重要性，我们借鉴 Group Lasso 的思想，提出了一个自适应多业务特征选择模块。

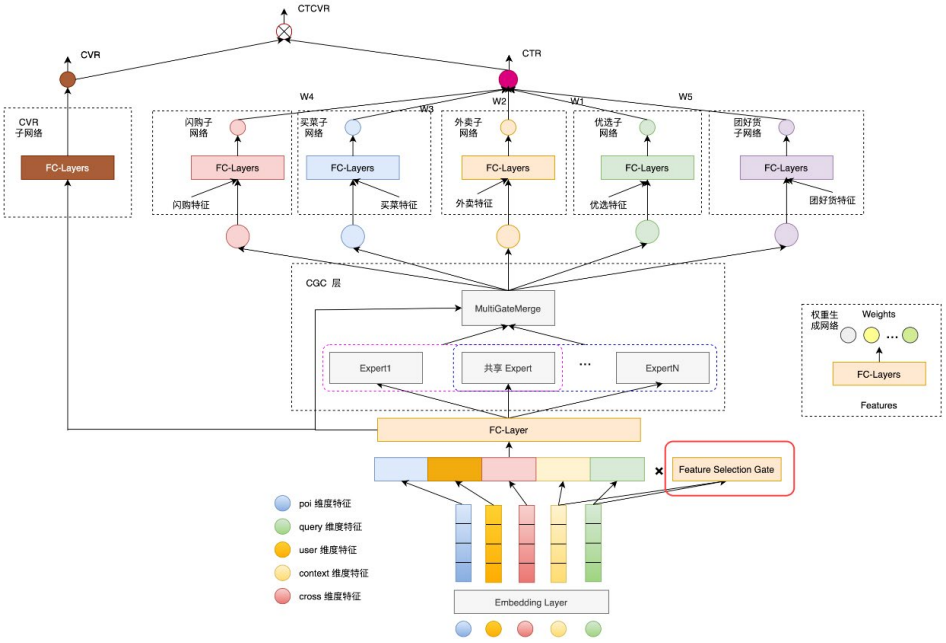


图 3　多子塔特征选择门结构

整体模型结构如上图 3 所示，我们通过一个特征选择门来计算每个特征的权重。特征选择门将 Query 以及上下文相关信息的 Embedding 表征经过一层线性映射以及 Softmax 激活函数，得到归一化的特征的权重向量：

$$g_t = softmax(W_t x_{query} + b_t) \tag{2}$$

其中 $g_t \in R^{h_2}$ 是第 t 个业务的特征权重向量，$W_t \in R^{h_2 \times h_1}$ 是第 t 个业务的投影矩阵，$b_t \in R^{h_2}$ 是对应的 Bias 向量。而后用这个权重向量点乘经过 Embedding 编码后的模型输入：

$$\hat{x}_t = g_t \odot x = \sum_{k=1}^{K} g_{t,k} x_k \tag{3}$$

对于不同的业务，我们用不同的门控 g_t 乘上输入，从而达到了分业务加权的目的。

这版模型相比 Group Lasso 正则化版本，离线 NDCG 提升了 16BP。

多业务概率图模型及训练

在特征选择门方案中，我们针对不同的业务选用了不同的门控来实现自适应特征选择。但是该方案有一个关键的问题：训练和推理阶段，我们怎么知道该选用哪个业务的门控向量 g_t？因此，模型首先需要能够识别当前请求的意图。

此外，多塔子结构的实践本质是想让业务更好地学出自己的特性。在图 2 和图 3 所示的多业务结构中，如果调参不当，很容易出现退化或者模式坍塌的现象。一个极端情况是，所有子塔中只有一个子塔得到了充分的训练，模型所有的输出都依赖于这个子塔，这与我们多业务建模的初衷相违背。

为了实现意图识别以及更好地训练多业务子塔，我们接着提出了一个概率图的框架，用一个隐变量 zz 来学习用户当前需求对应到每个业务的分布，系统性地用一个概率图将输入、业务、以及输出联系起来。模型结构示意图如下图 4 所示，我们在模型中额外引入一个先验网络 p 和一个后验网络 q，先验网络的目标是根据用户、查询词、上下文等输入信息预测采用哪个子塔，即建模 $p(z \mid x)$，后验网络的目标是根据输入以及用户的行为来建模后验概率 $p(z \mid x,y)$。训练目标是优化条件概率分布的一个证据下界（Evidence Lower Bound）：

$$\begin{aligned}\log p(y \mid x) &= \log \int_z p(y \mid x,z)\, p(z \mid x) dz \\ &\geq E_{q(z|x,y)}[\log p(y \mid x,z)] - D_{KL}[q(z \mid x,y) \,\|\, p(z \mid x)] \end{aligned} \tag{4}$$

这个公式第一项表示的含义是在训练时，我们从后验网络中拿到对应业务的分布 z，然后根据 z 来判断模型走哪个子塔并进行训练。第二项表示的含义是让先验网络和后验网络的输出尽可能接近，从而在推理阶段也能够根据先验塔的输出来判断走哪个子塔。这个后验网络只会在训练阶段使用。在具体实现上，我们将后验网络简化成一个规则决策模型：根据用户的行为反馈来预测对应的业务。

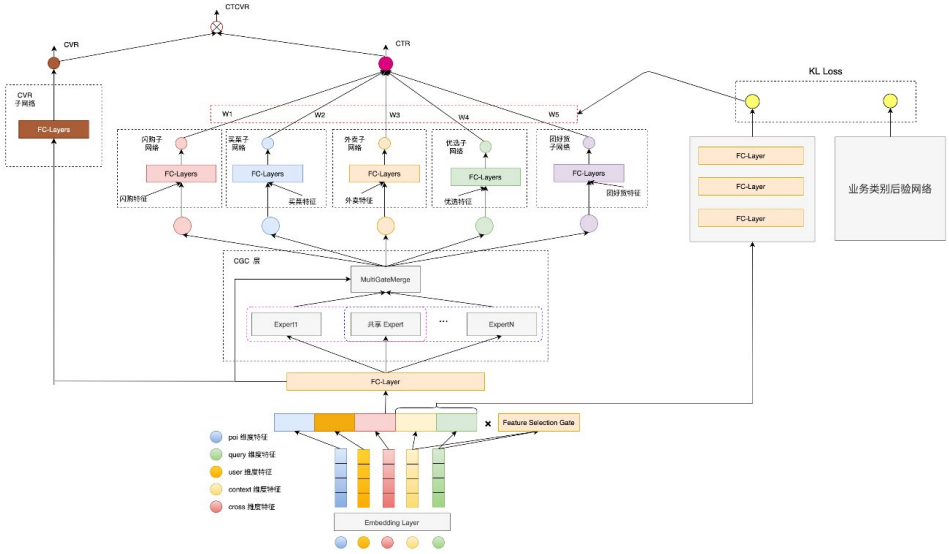


图 4　多业务概率图模型

引入概率图建模后离线 NDCG 提升了 39BP；最终我们将该版本进行线上 AB 测试：线上点击率提升了 25BP。

在多业务建模方面，除了上面的工作我们还陆续尝试了使用 Uncertainty 对分业务 Loss 进行加权 [4]、多业务 Hierarchy 结构 [5] 等，但均没有取得明显效果提升。

此外，我们也调研了阿里妈妈近期提出的 Star[6] 模型，Star 模型主要有以下几点贡献：1. 提出了一个星形（Star）的架构；2. 提出了分片归一化（Partitioned Normalization）的方案；（3）将域的信息显式加入到输出中（辅助网络）。我们尝试过将分片归一化、辅助网络的方案应用到我们的场景中，但均没有取得明显的效果。

聚合建模

当对各业务结果进行如图 5 所示的聚合展示时，聚块的位置顺序和聚块大小（即聚块内 item 的数量）共同决定用户的搜索体验。过去很长一段时间的技术方案是先使用排序模型预测业务的先后顺序，然后根据业务预测分数的相对差值通过人工规则来设置聚块大小。这种方案将聚块顺序和聚块大小分两步进行串行建模，距用户体验最优

解存在较大差距；其次，聚块排序与聚块大小的串行耦合会降低模型的迭代效果：前者的模型迭代替换，会导致后者规则相关的阈值设定不再合适。

图 5　结果聚合展示示意图

针对聚合展示样式带来的技术挑战，我们提出 GSRM (Grouping Search Results Model) 模型，该模型结构如下图 6 所示。这是一个多任务模型，同时进行聚块位置和大小的预测。聚块位置预测任务我们建模成 CTR 预估的分类问题，输出结果代表业务偏好强弱进行聚块的的位置排序；聚块大小预测任务我们建模成一个回归问题，输出聚块展示结果数。模型的底层输入特征包含查询词维度、用户维度、上下文维度、item 维度、以及各个维度的交叉特征。下面重点介绍两个工作：第一个是用户行为序列建模，第二个是聚块大小预测建模。

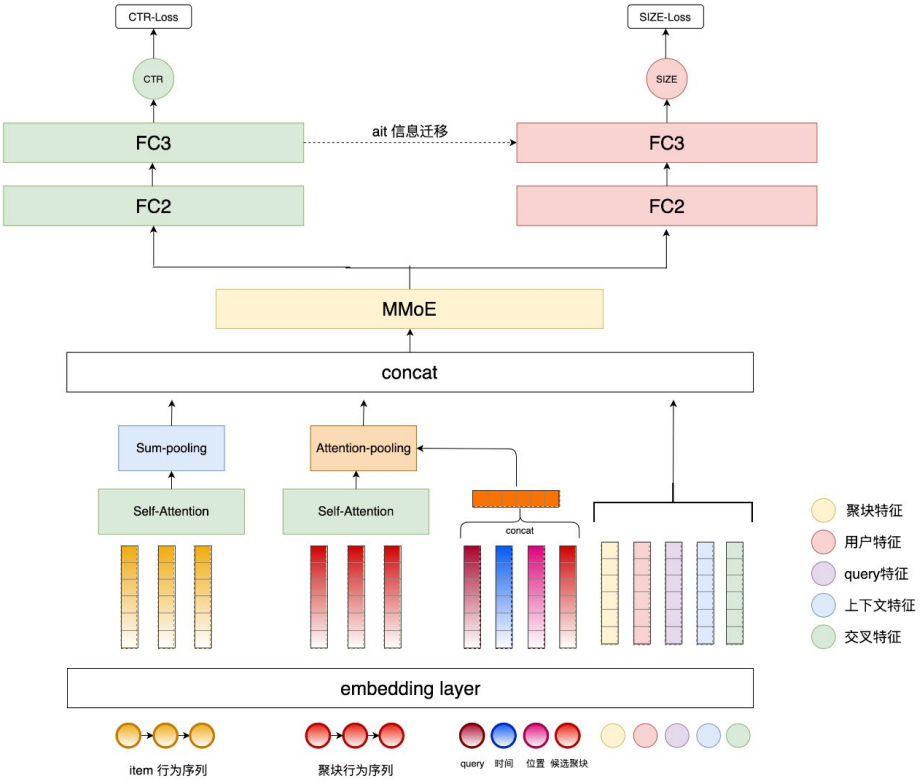


图 6　聚块展示模型示意图

异构双序列建模

众所周知，在深度学习 CTR 预估时代，在模型中引入用户行为序列特征能够极大提升模型的效果。业界关于端到端行为序列建模的工作，主要分为以下三类：第一类是基于注意力机制的模型，代表性工作如 DIN[7]、DIEN[8]、DSIN[9] 以及 BST[10] 等等；第二类是基于记忆网络的模型，代表性工作如 HPMN[11] 和 MIMN[12]；第三类是基于检索的模型，代表性工作如 UBR4CTR[13] 和 SIM[14]。这些工作对美团搜索多业务商品排序有很大的落地实践指导。

考虑到在聚合展示样式下，用户的行为具有异构性：用户既与聚块发生交互行为，也与聚块内的 item 发生交互行为，所以该业务排序场景相比业界大部分排序场景的用户行为序列建模不同，它属于异构行为序列建模。如上图 6 的模型结构所示，模型

使用了用户的 item 行为序列和聚块行为序列，其中我们使用更加丰富的用户全美团 App 业务序列来替换美团搜索聚块行为序列。

得益于我们在美团搜索商家个性化排序上积累的实践经验（参见《Transformer 在美团搜索排序中的实践》一文），两种序列都首先通过 Self-Attention 方式进行编码，得到一个更好的 Embedding 表示；然后对 item 行为序列直接进行 Sum-pooling，对聚块序列进行 Attention-pooling；考虑到在美团 O2O 场景下，用户的业务偏好兴趣与时间和地点有极强的关系，我们将查询词、时间和位置等上下文信息和候选聚块一同作为 Attention 中的 Target 来更好的刻画不同时空场景下用户的兴趣。

模型补充 Context-Aware 的异构双序列建模，离线 NDCG 提升了 10BP；线上 AB 测试也取得稳定正向的效果：点击率提升了 12BP，访购率提升了 9BP。

为了深入理解行为序列建模的效果，我们对 Context-Aware 部分的 Attention 权重进行可视化如下图 7 所示，发现对于用户最终发生行为的聚块，对应到行为序列中同类别聚块的权重也更大，说明 Context-Aware 能较好捕获用户当前请求的兴趣，在抽取兴趣时能够较好抑制其他行为的噪声。

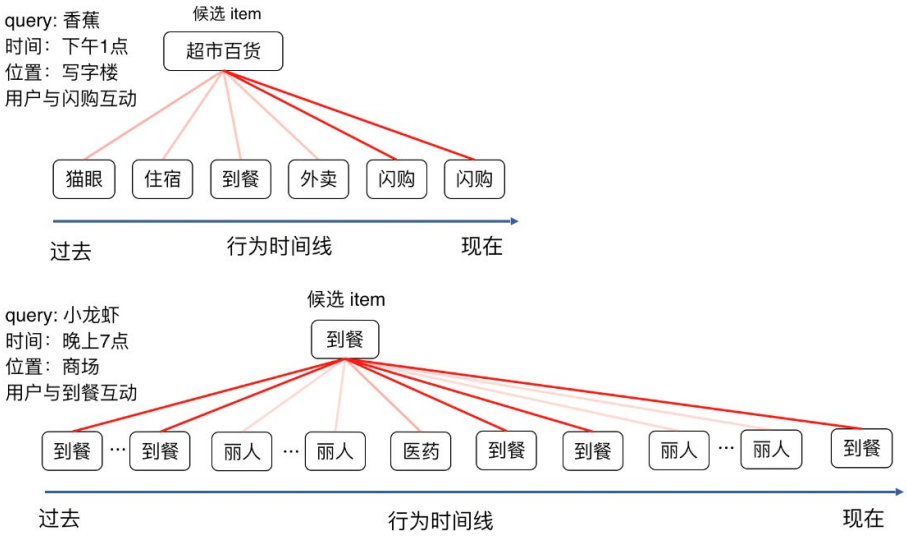


图 7　Target-Attention 权重示意图

聚块大小预测建模

针对聚块大小预测的回归任务，考虑到和之前的预估分类任务差异较大，我们在特征共享层采用目前业界常用的 MMoE[16] 多任务参数共享结构，同时从业务场景来看，当偏好预估认为某个业务很强时，应该给与相对多的业务结果数，所以我们借鉴了 AITM[17] 的工作，迁移偏好预估塔中的隐藏层信息到聚块大小预估塔中，整体模型如上图 6 所示。

模型训练损失函数如公式 (5) 所示，其中 pos 表示聚块的位置信息。

$$Loss = \lambda_1 * Loss_{ctr} + \lambda_2 * \varphi(pos) * Loss_{size} \quad \varphi(pos) = \begin{cases} 1 & pos = 0 \\ 0 & pos \neq 0 \end{cases} \tag{5}$$

其中偏好预估损失函数 $Loss_{ctr}$ 采用 LambdaLoss[18]，如下面公式（6）所示：

$$Loss_{ctr} = \sum_{y_i > y_j} \Delta NDCG(i, j) \log_2(1 + e^{-\sigma(\hat{y}_i - \hat{y}_j)}) \tag{6}$$

聚块大小预估损失函数 $Loss_{size}$ 采用 HuberLoss[19]，如下面公式（7）所示：

$$Loss_{size} = \begin{cases} \frac{1}{2}(label_{size} - pred_{size})^2 & \text{for} \mid label_{size} - pred_{size} \mid \leq \delta \\ \delta(\mid label_{size} - pred_{size} \mid - \frac{1}{2}\delta) & \text{otherwise.} \end{cases} \tag{7}$$

两个损失函数的权重 λ_1 和 λ_2 根据人工经验设定。针对聚块大小 Loss，考虑到实际我们只调整首位的聚块大小，所以聚块大小 Loss 只对首位聚块生效。其中的 $label_{size}$ 会依据用户的行为反馈进行适当设计：当用户点击“查看更多”或者“聚块标题”时，会对线上历史展示个数适当放大；当用户在非首位下单时，会对首位聚块大小进行适当缩小；当用户在首位下单时，会综合考虑用户的浏览个数、最后点击位置以及历史展示大小进行调整。

对聚块大小进行联合预测，离线 NDCG 持平，线上 AB 测试结果：访购率提升了 3BP。

总结和展望

本文介绍了我们在多业务商品排序上的探索与实践经验。关于多业务统一排序，基于过去商家多业务排序的经验，继续从特征选择、引入概率图模型优化模型训练来持续提升多子塔结构的效果。可以看出，我们的工作具有较好的延续性和迭代性；同时当结果采用聚合展示样式时，我们采用 GSRM 聚合模型技术策略来与之适配。这些工作也都取得了不错的效果。

未来，我们会在以下几个方面展开进一步的工作：

1 针对如何使模型更好地学习不同业务的特性和共性，将在特征自适应选择和参数共享两个方面持续迭代模型结构。

2. 针对上层的业务子塔结构，不同的业务可能适用不同的参数量，将结合 AutoML 技术对多业务结构中的子网络进行优化。

3. 针对目前人工设计多任务 Loss 权重带来的调参工作量，将探索对多任务 Loss 权重自适应调整的方法。

4. 针对用户对单个 item 的决策行为受周围展示 item 信息影响，而目前的排序方法没有较好建模这个信息，将探索考虑 Context 信息多业务混排方法。

参考资料

[1] 多业务建模在美团搜索排序中的实践

[2] Ma X, Zhao L, Huang G, et al. Entire space multi-task model: An effective approach for estimating post-click conversion rate[C]//The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval. 2018: 1137-1140.

[3] Friedman et al., A note on the group lasso and a sparse group lasso.

[4] Kendall et al., Multi-Task Learning Using Uncertainty to Weigh Losses for Scene Geometry and Semantics. In CVPR, 2018.

[5] Guo et al., Dynamic Task Prioritization for Multitask Learning. In ECCV, 2018.

[6] Sheng et al., One Model to Serve All: Star Topology Adaptive Recommender for Multi-Domain CTR Prediction. In CIKM, 2021.

[7] Zhou G, Zhu X, Song C, et al. Deep interest network for click-through rate prediction[C]//Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. ACM, 2018: 1059-1068.

[8] Zhou G, Mou N, Fan Y, et al. Deep interest evolution network for click-through rate prediction[C]//Proceedings of the AAAI Conference on Artificial Intelligence. 2019, 33: 5941-5948.
[9] Feng Y, Lv F, Shen W, et al. Deep Session Interest Network for Click-Through Rate Prediction[J]. arXiv preprint arXiv:1905.06482, 2019.
[10] Chen Q, Zhao H, Li W, et al. Behavior sequence transformer for e-commerce recommendation in Alibaba[C]//Proceedings of the 1st International Workshop on Deep Learning Practice for High-Dimensional Sparse Data. 2019: 1-4
[11] Kan Ren, Jiarui Qin, Yuchen Fang, Weinan Zhang, Lei Zheng, Weijie Bian, Guorui Zhou, Jian Xu, Yong Yu, Xiaoqiang Zhu, et al. Lifelong sequential modeling with personalized memorization for user response prediction. In SIGIR, 2019.
[12] Qi Pi, Weijie Bian, Guorui Zhou, Xiaoqiang Zhu, and Kun Gai. Practice on long sequential user behavior modeling for click-through rate prediction. In KDD, 2019.
[13] Jiarui Qin, W. Zhang, Xin Wu, Jiarui Jin, Yuchen Fang, and Y. Yu. User behavior retrieval for click-through rate prediction. In SIGIR, 2020.
[14] Search-based User Interest Modeling with Lifelong Sequential Behavior Data for Click-Through Rate Prediction.
[15] Transformer 在美团搜索排序中的实践
[16] Ma J, Zhao Z, Yi X, et al. Modeling task relationships in multi-task learning with multi-gate mixture-of-experts[C]//Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. 2018: 1930-1939.
[17] Xi D, Chen Z, Yan P, et al. Modeling the Sequential Dependence among Audience Multi-step Conversions with Multi-task Learning in Targeted Display Advertising[J]. arXiv preprint arXiv:2105.08489, 2021.
[18] Burges C J C. From ranknet to lambdarank to lambdamart: An overview[J]. Learning, 2010, 11(23-581): 81.
[19] https://en.wikipedia.org/wiki/Huber_loss
[20] Liu et al., AutoFIS: Automatic Feature Interaction Selection in Factorization Models for Click-Through Rate Prediction, In ADS-KDD, 2020.
[21] Khawar et al., AutoFeature: Searching for Feature Interactions and Their Architectures for Click-through Rate Prediction, In CIKM, 2020.
[22] Tang et al., Progressive Layered Extraction (PLE): A Novel Multi-Task Learning (MTL) Model for Personalized Recommendations, In Recsys, 2020.

作者简介

曹越、瑶鹏、诗晓、李想、家琪、可依、晓江、肖垚、培浩、达遥、陈胜、云森、利前均来自美团平台搜索与 NLP 部。

情感分析技术在美团的探索与应用

作者：任磊　佳昊　金刚等

背景

移动互联网的发展和普及已达到空前规模，亿万网民在互联网上可以获得信息、交流信息，发表自己的观点和分享自己的体验，表达各种情感和情绪，如批评、赞扬以及喜、怒、哀、乐等。作为国内领先的生活服务电子商务平台，美团连接起了百万商户和亿级用户，积累了数十亿真实用户评论。用户评论不仅为消费者选店决策提供参考，也是商家获取消费者反馈的渠道。对用户评论进行情感分析可以帮助美团深入理解商家服务水平和用户偏好，为改进平台服务与获取优质内容提供了最直观的反馈，同时也在美团搜索、个性化推荐、商业智能、内容安全等十多个业务场景中有着广泛应用。

以大众点评 App 为例，用户既可以发表对商家的星级打分（一星到五星），也可以对商家服务或产品发表详细评论。我们对大众点评商户近半年的评论进行了统计分析，用户情感倾向在不同星级商家中的分布如图 1 所示，商家整体星级打分反映出用户对该商家的整体情感倾向，但同时我们发现高星商户的评论中存在大量负向情感内容，低星商户的评论中也存在不少正向情感内容。究其原因在于用户评论中存在大量星级评分与对应评论情感倾向不一致的情况，如图 2 中展示的两条用户评论，高星级评论中存在负向情感内容，低星级评论中也存在正向情感内容。

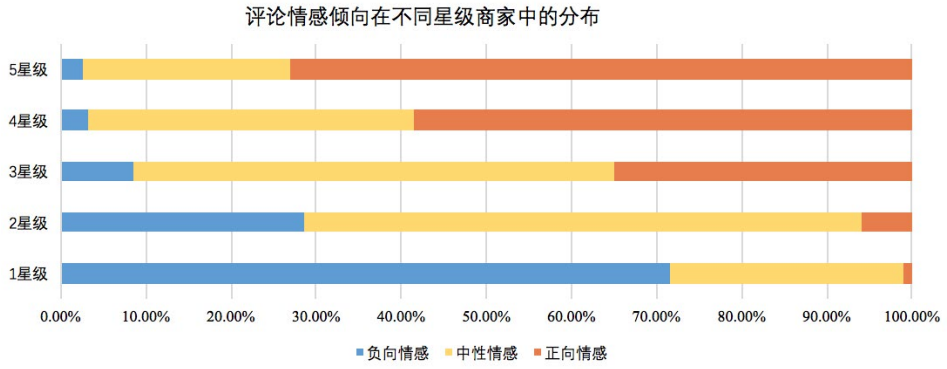


图 1　评论情感倾向在不同星级商家中的分布

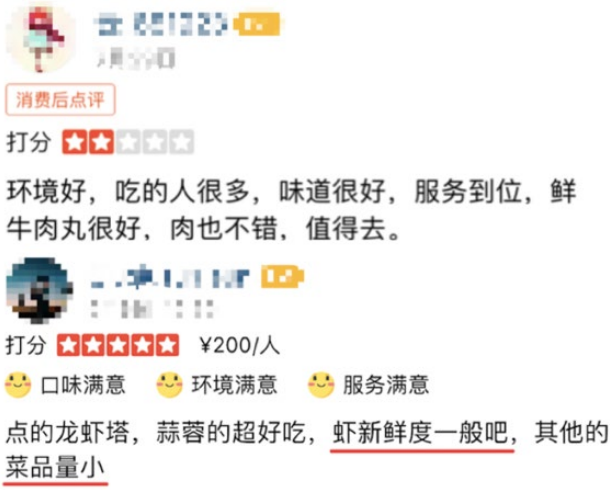


图 2　用户星级评分与评论情感倾向不一致

图 3 展示了大众点评 App 上一家「五星级」咖啡厅，用户对口味、环境、服务各项评分都很高。我们对其所有用户评论分析后发现好评主要集中在口味、服务、性价比等属性，但也有不少用户抱怨说咖啡厅的位置难找，不好停车，座位比较拥挤，空间小等。整体星级评分并不能全面体现用户对商家的真实评价。因此，除了对用户评论内容进行整体情感判断外，我们还需要对用户评论进行更细粒度的情感分析才能精准地挖掘用户对商家的反馈，进而改善商家服务，提升用户体验。

图 3　商家详情页

2021 年 5 月，美团 NLP 中心开源了迄今规模最大的基于真实场景的中文属性级情感分析数据集 ASAP[1]，该数据集相关论文 [2] 被自然语言处理顶会 NAACL2021 录用，同时该数据集加入中文开源数据计划千言 [3]，将与其他开源数据集一起推动中文信息处理技术的进步。借此机会，本文回顾了美团情感分析技术的演进和在典型业务场景中的应用，整体技术框架如图 4 所示。

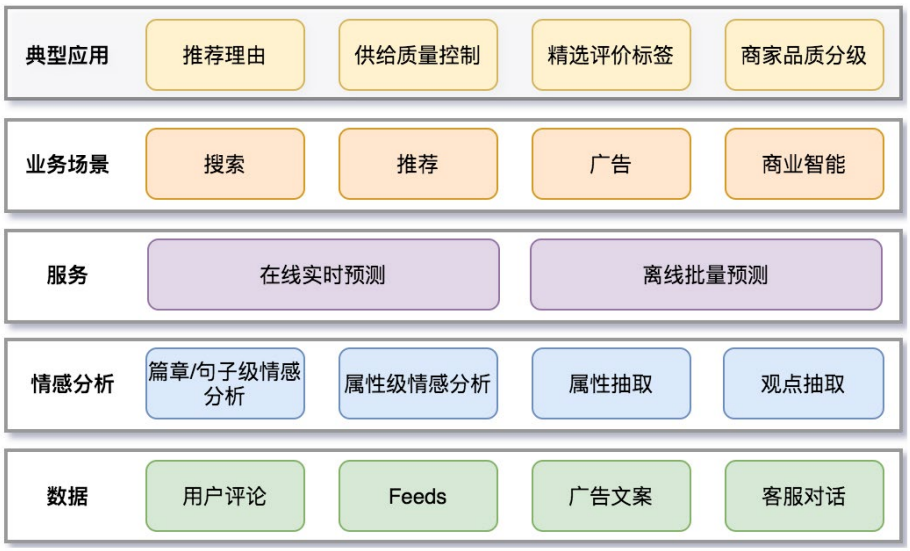


图 4　美团情感分析整体技术框图

情感分析介绍

情感分析的目标是分析文本中表达或蕴含的正向（褒义）、负向（贬义）或中立的情感倾向。根据所处理文本的粒度，情感分析可以分为篇章 / 句子级情感分析和属性级情感分析 [4]，与属性级情感分析密切相关的还有观点三元组分析。

篇章 / 句子级情感分析的目标是判断整篇文档或整句子的情感倾向（即正向、负向或中立）。

属性级情感分析与篇章 / 句子级情感分析不同，需要同时考虑目标信息（属性）及其对应的情感。例如“这家餐厅装修很有格调，饭菜也很好吃，就是位置很不显眼，找

了好久才找到，并且服务员态度也不够满意。”如果从整体看，很难说这句话应该被判断成褒义还是贬义，因为在评论中，用户称赞了餐厅的装修和菜品属性，但是吐槽了餐厅的位置和服务属性。为了得到更全面和精确的情感信息，我们需要识别粒度更细的属性级情感。这种任务学术上被称为 ABSA（Aspect Based Sentiment Analysis，基于属性的情感分析），可以划分为 ACSA（Aspect Category Sentiment Analysis，基于属性类别的情感分析）和 ATSA（Aspect Term Sentiment Analysis，基于属性项的情感分析）[5]，其中 ACSA 是识别在相应预定义属性类别（Aspect Category）上的情感倾向，如上述评论在属性类别“菜品口味”上表达正向情感，在属性类别“服务态度”上表达负向情感。ATSA 中是识别针对文本中出现的相应属性（Aspect Term）的情感倾向，如上述评论中针对属性“饭菜”表达正向情感，针对属性“服务员”表达负向情感。本文所述属性级情感分析主要指 ACSA 任务。

观点三元组分析包括属性抽取（Aspect Extraction）、观点抽取（Opinion Extraction）以及属性 – 观点对（Aspect-Opinion）的情感倾向分析三个任务，在学术上称为 Aspect Sentiment Triplet Extraction[6] 或者 Opinion Triplet Extraction[7]。属性抽取和观点抽取属于信息抽取的范畴，属性 – 观点对的情感倾向分析属于分类任务。属性抽取任务中抽取的属性（Aspect Term），是上述 ATSA 任务的基础，观点抽取任务是抽取人们针对属性所表达的观点，例如评论“宫保鸡丁这道菜真是棒极了，太好吃了！”中的属性“宫保鸡丁”，观点“真是棒极了，太好吃了！”；属性 – 观点对的情感倾向分析是识别基于该属性表达观点的情感倾向，例如“宫保鸡丁——真是棒极了，太好吃了！”情感倾向为正向。

篇章 / 句子级情感分析及应用

篇章 / 句子级情感分析即给定一段文本，通过模型给出该文本整体的情感强度或情感倾向。根据模型输出类型划分，主要分为回归方案和分类方案，其中回归方案的输出为连续的情感强度 [0 ~ 1]（趋向 0 代表负向程度，趋向 1 代表正向程度），分类方案的输出为离散的情感倾向（正向、负向、中立）三分类。调研多个业务场景对情感分析的需求，同时考虑模型在应用中的通用性和灵活性，我们最终采用了回归模型方

案。为了建立精准的情感分析能力，在数据标注时我们将情感强度划分为离散的七档评分——[非常负向 , 负向 , 轻微负向 , 中性 , 轻微正向 , 正向 , 强烈正向]，并采用 Min-Max Normalization 将标注 Label 归一化到 [0 ~ 1]。

在这个任务上，业界主流技术方案主要有 TextCNN[8]、Att-BLSTM[9] 以及 BERT[10] 为代表的预训练模型。其中 Att-BLSTM 提出利用注意力机制获取句子中最重要的信息，且在学习长距离语义上优于 TextCNN，因此第一版模型框架采用了图 5 所示的 Att-BLSTM 方案。2018 年以来以 BERT 为代表的预训练模型在多项 NLP 任务中取得巨大进展，BERT 在特征抽取上具有较强优势，如浅层的句法特征以及深层的语义特征。基于上述考虑，我们用自研 MT-BERT[11] 模型对 Att-BLSTM 架构中的编码器进行升级，并在多个业务评测上取得了显著效果提升，部分实验结果见图 6。

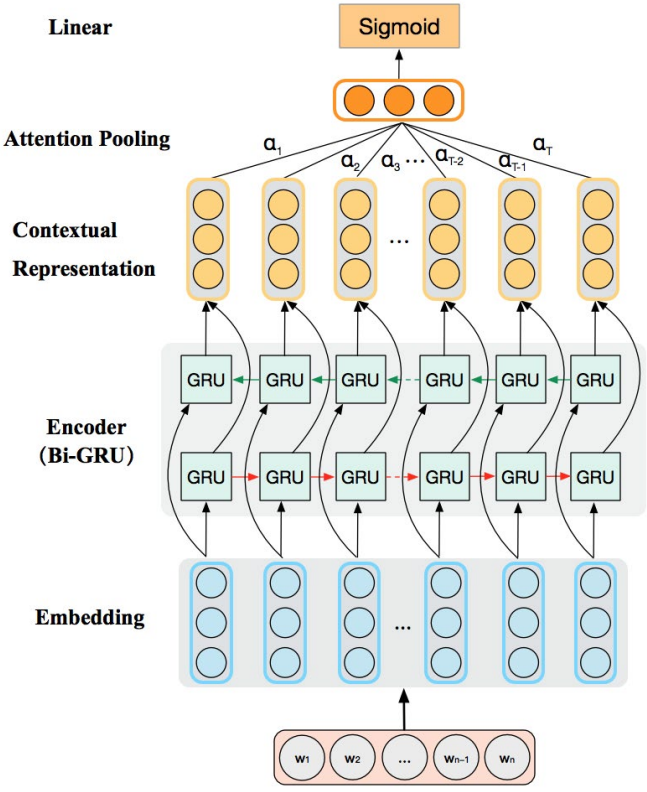


图 5　篇章 / 句子级情感分析模型架构

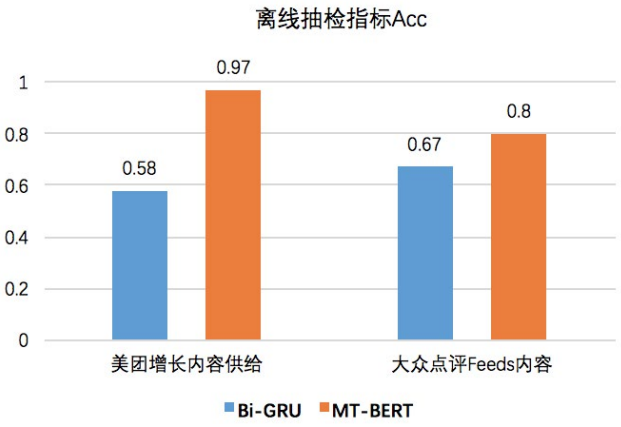
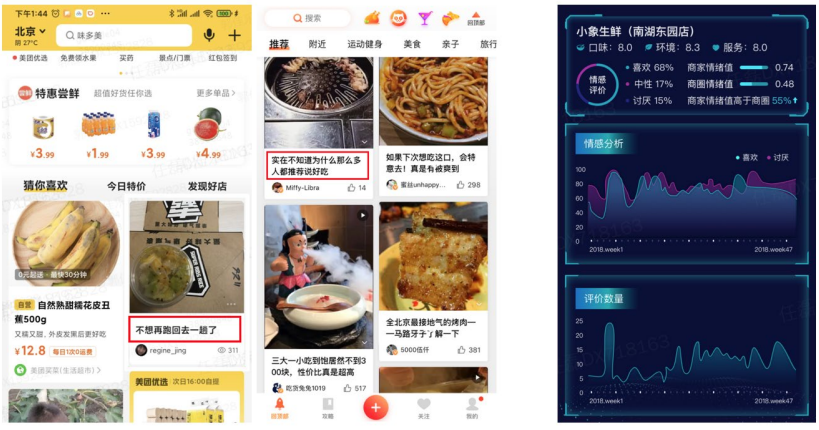


图 6　篇章 / 句子级情感分析模型迭代效果指标

篇章 / 句子级情感分析在美团业务中有着广泛应用，如广告和推荐业务中的文本安全管控和商家运营情感监测等。如图 7（a）所示，美团 App 和大众点评 App 首页信息流文案大部分抽取自用户真实评论，需要从内容供给上严格控制文本质量，通过句子级情感分析对供给内容中的负向内容进行过滤，提升终端用户体验。此外，在商业智能场景中，句子级情感分析还可以帮助商家优化运营策略，如图 7（b）所示，美团商业大脑通过情感分析能力可以监控用户对商家评价的情感曲线以及评论情感占比等指标来改善商家运营。



(a)文本安全管控场景　　(b)商家运营情感监测场景

图 7　篇章 / 句子级情感分析在美团业务场景的应用

属性级情感分析及应用

篇章 / 句子级情感分析只是针对文本整体表达或蕴含的情感倾向分析，无法获取针对评价观点对象（属性）上的情感倾向。为了得到更细粒度的情感分析结果，我们需要针对美团业务场景构建属性级情感分析能力，即给定一段文本，通过模型给出该文本在指定属性上的情感倾向。

数据收集与分析：针对大量用户评论进行统计分析，同时参考 SemEval 数据集 [12]，并与业务产品专家沟通美团业务特点后，我们将餐饮业务下的属性设计为菜品、环境、价格、服务、位置，共 5 类。数据分析中发现美团业务下的用户评论趋于多样化和细粒度化。以该真实用户评论“... 装修看起来很高大上的样子，但是因为主厅在举办婚礼非常混乱，感觉特别吵 ...”为例，用户对装修和噪声两个环境细分属性表现出相反情感倾向。因此我们将属性从上述 5 个类别细分到 18 个子类别，最终设计为两级属性体系，如图 8 所示。

图 8　两级属性体系

为了构建高质量数据集，我们进行了详细数据分析和调研业界标准，最终制定了面向用户评论的属性级情感分析标注规则。对于每条评论，标注人员需要先判断预定义的属性是否在该评论中被提及，再判断被提及属性上的情感倾向。因此对于每个属性共有 4 种标签——[未提及 , 负向 , 中立 , 正向]。

我们对 ASAP 数据集中属性及其对应的情感倾向分布做了统计，如图 9 所示。本次标注主要面向餐饮商家下的用户评论，所以 94.7% 的评论都提及菜品口味这一属性。除此之外，用户们还比较关心服务人员态度，价格水平，装修情况等属性。

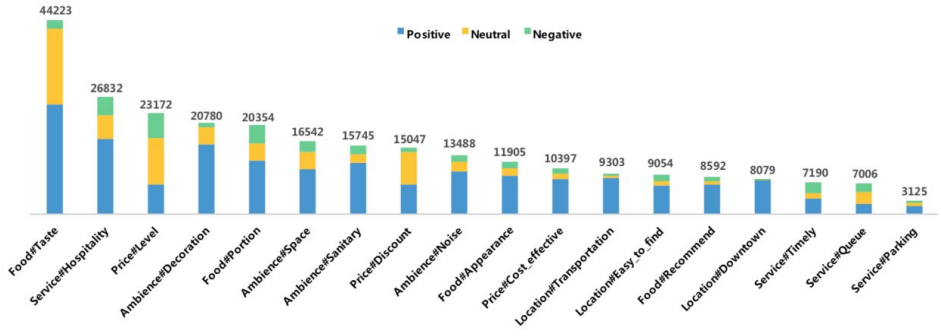


图 9　ASAP 数据集中属性以及对应的情感倾向分布

与属性级情感分析开源数据集 SemEval RESTAURANT[13] 相比，ASAP 在数据量上远远超出，可以支撑深度神经网络模型的探索。相对于单句级别的 RESTAURANT，包含多子句评论级别的 ASAP 具有更长的平均长度，因此往往包含更丰富的属性信息。在 ASAP 数据集中平均每条评论包含 5.8 个属性，是 RESTAURANT 数据集的 4.7 倍，相对于 RESTAURANT 数据集具有更大的挑战性。

表1 ASAP和SemEval RESTAURANT的统计数据和label分布

Dataset	Split	Reviews	Average sentences per review	Average aspects per review	Average length	Positive	Negative	Neutral	1-star	2-star	3-star	4-star	5-star
ASAP	Train	36, 850	8.6	5.8	319.7	133, 721	27, 425	52, 225	1, 219	1, 258	5, 241	13, 362	15, 770
	Dev	4, 940	8.7	5.9	319.9	18, 176	3, 733	7, 192	151	166	784	1, 734	2, 105
	Test	4, 940	8.3	5.7	317.1	17, 523	3, 813	7, 026	165	173	717	1, 867	2, 018
RESTAURANT	Train	2, 855	1	1.2	15.2	2150	822	498	-	-	-	-	-
	Test	749	1	1.3	15.6	645	215	94	-	-	-	-	-

属性级情感分析多任务学习框架：在该类问题上，业界主流技术方案根据模型结构不同可以分为两种，分别为基于非预训练模型的方案和基于预训练模型的方案。基于非预训练模型的方案中，Att-BLSTM[9] 需要为每个属性单独训练一个模型，无论模型训练还是模型应用都比较复杂；ATAE-LSTM[14] 通过属性向量（Aspect Embedding）引入属性信息，同时提出基于注意力（Attention）机制让模型聚焦到对应的属性相关信息上来提升模型准确率；CapsNet[15] 可以用于 ABSA 任务中学习属性（Aspect）和文本（评论）之间关系。基于预训练模型的方案中，QA-BERT[16] 将 ABSA 任务转化为问答形式的句间分类任务，每个属性都和文本构成句对作为输入，实际应用中属性数量较多时，计算量较大；CapsNet-BERT[17] 在 CapsNet 方案的

基础上将语义编码层（Encoder）从 Bi-GRU 升级为 BERT。上述模型都只考虑了属性与文本的关系，把每个属性都单独处理，没有考虑不同属性间的关系。

通过数据分析部分可知，ASAP 数据集在不同属性上的分布非常不均衡，这会导致模型训练时数据量小的属性上更容易出现过拟合，不同属性上的情感倾向表达有共性同时也有各自特定的表达方式。模型设计时需要结合业务数据特点同时兼顾计算效率，最终我们采用了多任务学习框架来兼顾属性间关系，同时处理一条评论的多个属性，模型整体设计为两层，分别为共享层和任务相关层，如图 10 所示。

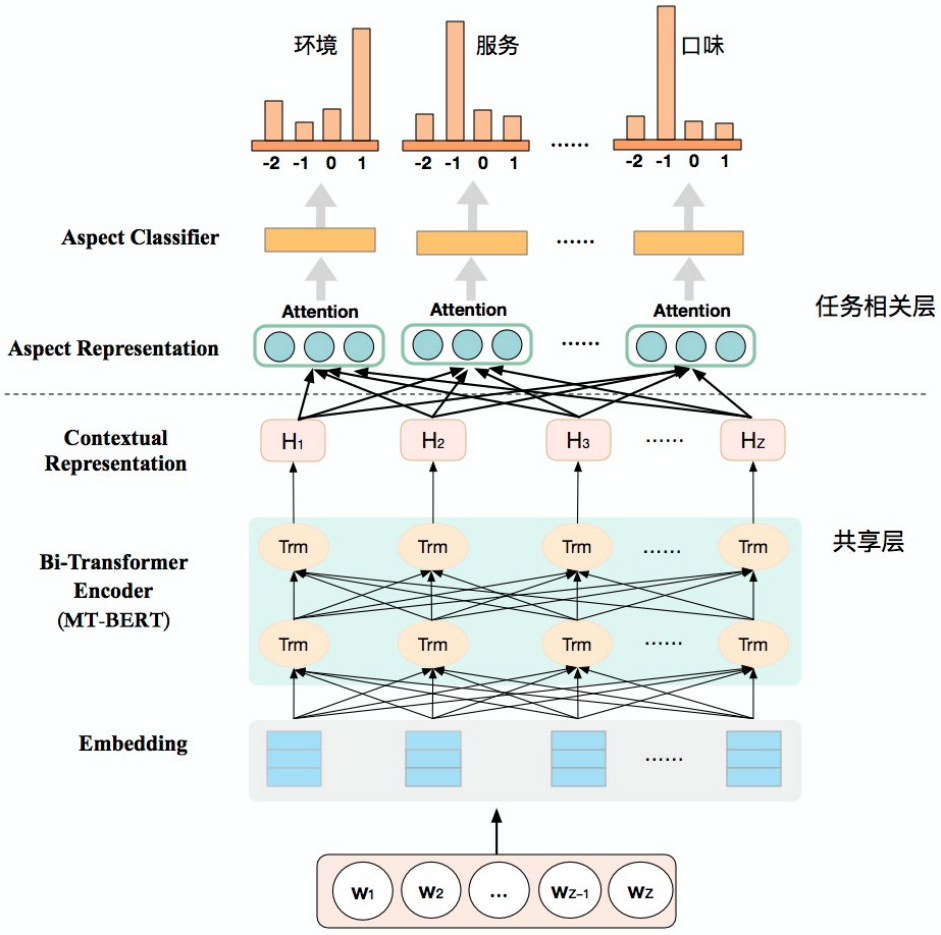


图 10　美团属性级情感分析模型架构

共享层为语义编码层（Encoder），目标是学习文本在不同属性上的上下文深度语义表征。通过参数共享实现知识的迁移可缓解数据不均衡带来的问题，同时对新增属性的扩展具有较好的兼容性。

$$H = Encoder(R) \tag{1}$$

其中，$H = \{h_1, h_2, ..., h_Z\}$ 为评论 $R = \{w_1, w_2, ..., w_Z\}$ 经过语义编码层（EncoderEncoder）编码后得到的向量，Z 代表评论的长度。

任务相关层包括注意力层和分类层，注意力层设计上我们借鉴了 ATAE-LSTM 中注意力池化的方法，让模型可以聚焦到整条评论中和指定属性最相关的句子或者片段，减少其他属性等无关信息的干扰，提升该属性上的情感倾向判断准确率。

$$M_i^a = \tanh(W_i^a * H) \tag{2}$$

$$\alpha_i = softmax(\omega_i^T * M_i^a) \tag{3}$$

$$r_i = \tanh(W_i^p * H * \alpha_i^T) \tag{4}$$

其中，α_i 是注意力权重向量，对句子中不同重要程度的词语给予不同的权重。r_i 是评论针对第 i 个属性经过注意力池化后得到的加权表征。

最后模型经过 Softmax 分类器，

$$\hat{y}_i = softmax(W_i^q * r_i + b_i^q) \tag{5}$$

对于给定的评论 RR，属性级情感分析的损失如下：

$$L_{ACSA} = \frac{1}{N}\sum_{i=1}^{N}\sum_{C} y_i * \log \hat{y}_i \tag{6}$$

其中，N 为属性（Aspect）数量，C 是分类类别数。

随着预训练语言模型技术的快速发展，结合团队自研的预训练模型 MT-BERT 不断迭代，我们在上述多任务框架的基础上，对参数共享层的语义编码（Encoder）部分做了持续的迭代，由最初的 Bi-GRU 到 Google-BERT-Base，再到 MT-BERT-

Base，以及目前的 MT-BERT-Large，模型在 Benchmark 上的指标得到了进一步的提升，如图 11 所示。

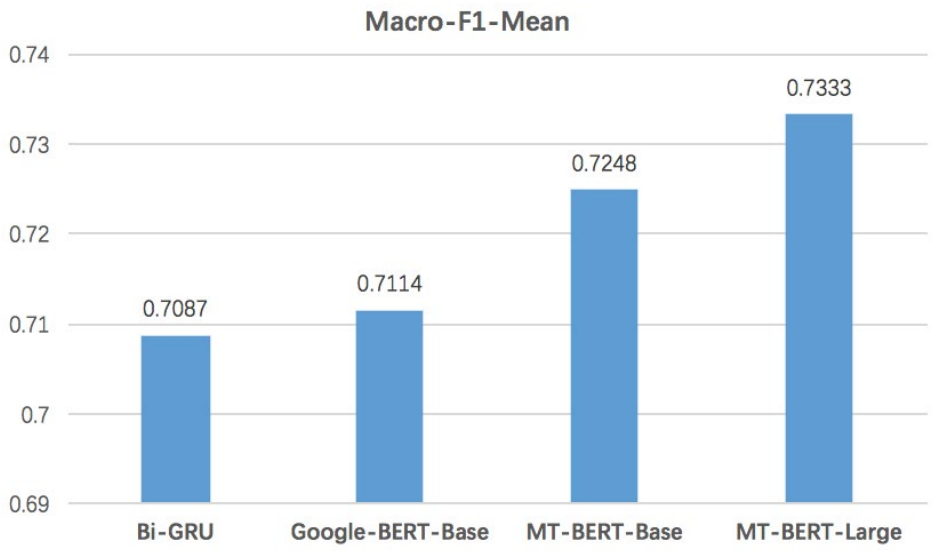


图 11　美团属性级情感分析模型迭代指标

典型应用：属性级情感分析在美团业务中有非常多的应用场景，如精选点评、个性化推荐、智能搜索、商家运营等。大众点评 App 的精选点评模块是属性级情感分析的典型应用场景之一，如图 12 所示。精选点评模块作为点评 App 用户查看高质量评论的入口，其中精选点评标签承载着内容结构化聚合的作用，支撑着用户高效查找目标评论内容的需求。鉴于属性级情感分析能够挖掘用户评论在不同属性上表达的情感倾向，根据其构建的属性级情感标签能够较好的帮助用户筛选查看，同时外露更多的商家信息，帮助用户高效的从评论中获取消费指南。

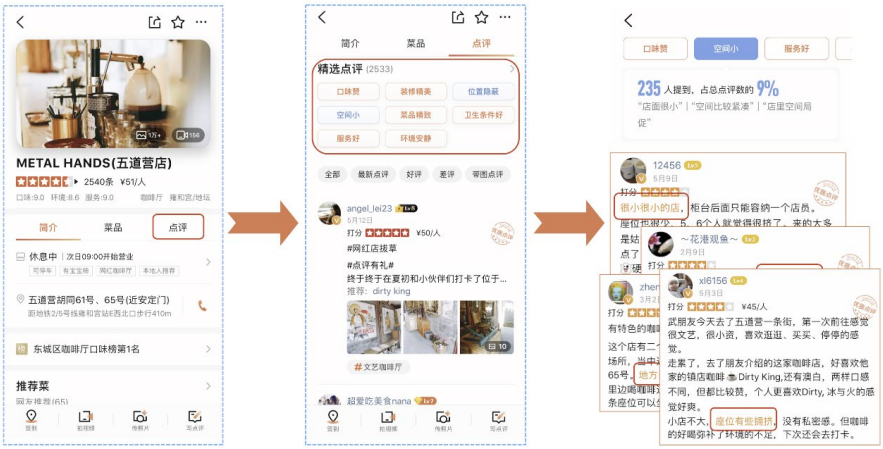


图 12　大众点评精选点评模块产品形态

美团商业大脑运营分析是属性级情感分析在美团商业智能典型应用场景，如图 13 所示。美团商业大脑是美团搭建起的面对生活服务业商家提供大数据分析决策支持的知识图谱平台，通过细致刻画商家服务现状、商家竞争力分析，以及商圈洞察等方式，为商户提供精细化经营建议。美团商业大脑通过对用户评论进行全方属性级情感分析与归纳总结，从而可以发现商家在市场上的竞争优势 / 劣势、用户对于商家的总体印象趋势、商家菜品的受欢迎程度变化，并对商家提供前瞻性经营方向，智能化指导商家精准优化经营模式。

图 13　美团商业大脑商家运营分析

观点三元组分析及应用

在观点三元组抽取任务中，工业界目前主流的方案是基于 Pipline 的方案，即先分别抽取属性和观点，然后抽取两者关系，最后通过句子级情感分析获得属性和观点对的情感倾向。Pipline 方案的优点是整个任务可以模块化，复用关系抽取、情感分析等通用能力，且各模块可以单独迭代优化；缺点是误差会在 Pipline 模块中传导累积，后序模块对前序模块的准确度有较强的依赖性，且计算效率低。

在学术界，最新的一些工作 MTL[7] 和 GTS[18] 等提出了多任务学习的解决方案。结合美团业务特点同时兼顾计算效率的考虑，最终我们选择了多任务学习的方案，模型架构如图 14 所示。模型整体由四部分组成：Pretrained Language Model、Aspect Tagger、Opinion Tagger、Sentiment Parser，其中 Pretrained Language Model 为基于预训练语言模型的编码器，兼容 BERT、RoBERTa[19]、ELECTRA[20] 等主流预训练模型，Aspect Tagger 执行属性序列标注任务，Opinion Tagger 执行观点描述序列标注任务，Sentiment Parser 执行属性与观点描述关系依赖及其情感分析解析任务，主要为一个 Biaffine Scorer[21]。

如图 14 中的例子所示，Aspect Tagger 将“虾滑”和“墨鱼滑”这两个属性抽取出来，而 Opinion Tagger 则对应地将“很不错”和“并不新鲜”这两个观点描述抽取出来，Sentiment Parser 给出句子中的所有字之间的情感依赖关系（即正向、负向、中立、无关）四分类。我们利用抽取出来的属性和观点描述在所有字之间的情感依赖关系上进行索引和记录，最终可以得到“虾滑”和“很不错”之间的 2*3=6 个情感依赖关系中最多的关系为正向情感，而“虾滑”和“并不新鲜”之间的 2*4=8 个情感依赖关系中最多的关系为无关。类似地，我们可以推断出“墨鱼滑”和”并不新鲜“之间的关系为负向情感。模型训练时，模型整体对三个子任务损失进行联合优化。

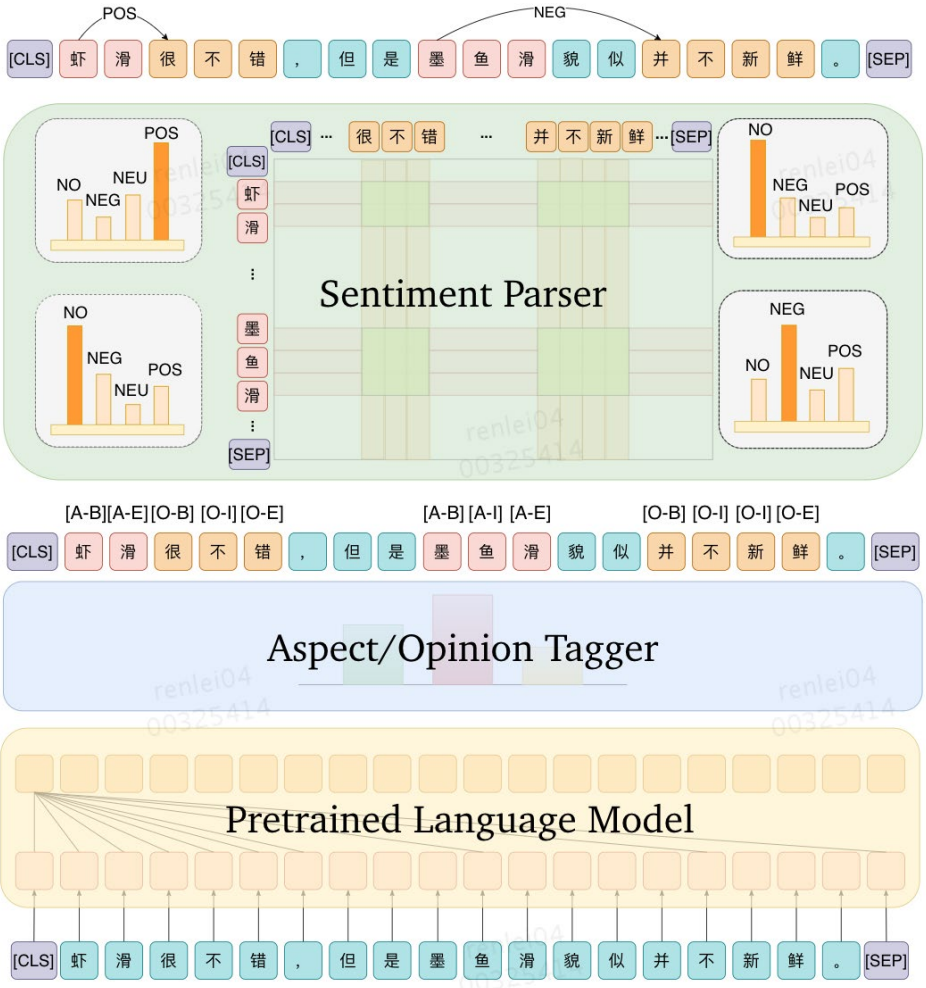


图 14　观点三元组抽取多任务学习模型架构

此外，我们对 Pretrained Language Model 模块进行了优化，由最初的 Google BERT-Base，到自研 MT-BERT-Base，再到目前的 MT-BERT-Large，模型在 Benchmark 上的指标得到了进一步的提升，如图 15 所示。由于观点三元组分析既包括分类任务也包括抽取任务，评价指标上我们采用了 EM-F1 和 Fuzzy-F1，其中 EM-F1 需要任意两个观点三元组（属性，观点描述，情感）中三个元素全部正确整体才算一次命中，Fuzzy-F1 会考虑抽取片段的重合度（包括属性和观点描述），

EM-F1 比 Fuzzy-F1 更加严格。

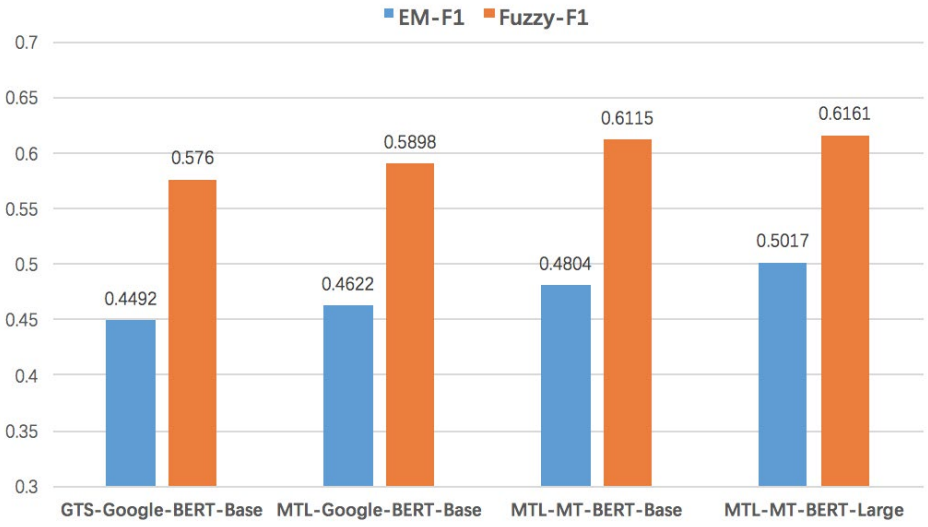


图 15　观点三元组抽取模型指标迭代

通过观点三元组分析技术可以构建基于属性的情感观点知识库，其中每个属性（商品或服务）节点都会被一个或多个带情感依赖的观点描述指向，如图 16 所示。通过【属性 + 正向观点】的模板方法可以构造推荐文案，例如”酸汤鱼片：酸酸甜甜，妙不可言”，可以用于美团 App 和大众点评 App 首页推荐或广告文案场景，同时可以根据用户对属性（商品或服务）的偏好，展示个性化推荐文案，帮助用户更加高效的获取信息。同时这些基于属性（商品或服务）的情感观点信息可以帮助商家更加精细化的运营。

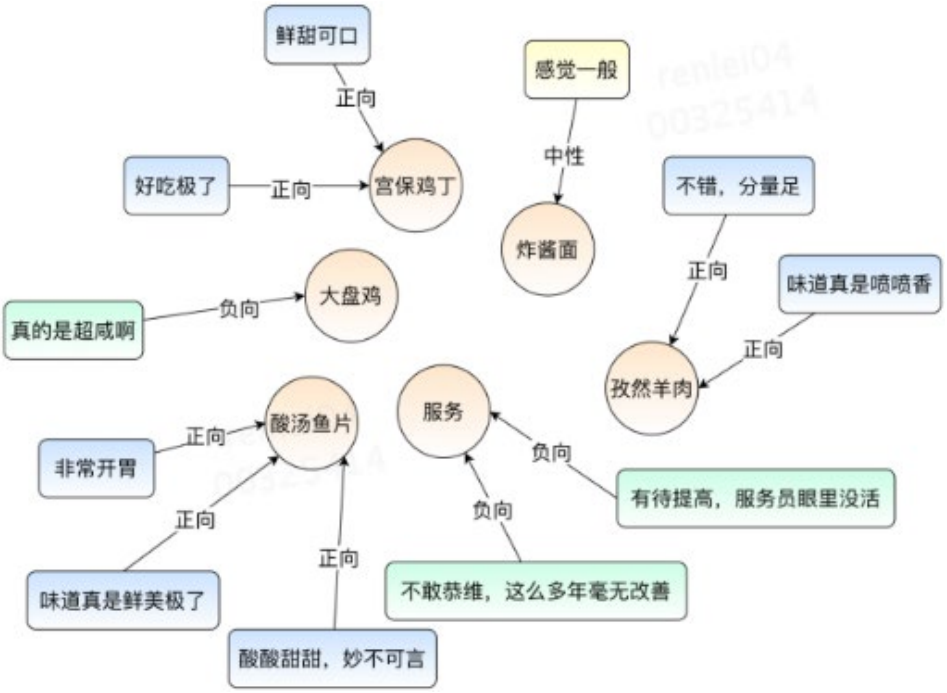


图 16　基于属性（商品或服务）的情感观点知识库样例展示

情感分析服务化

很多美团业务场景对情感分析有迫切需求，例如推荐理由、Feeds 内容、广告文案、精选评价标签、商家运营、对话系统等场景都需要识别文本在整体或属性上的情感倾向。针对美团业务特点我们构建了在线实时预测服务和离线批量预测服务。

在线实时预测服务主要针对需要实时计算的场景，如对话系统需要实时识别用户对话内容的情感倾向，提供 Thrift 接口调用方式，方便美团内部业务服务调用；离线批量预测服务主要针对需要离线批量计算的场景，如每日新生产的 Feeds 内容需要批量进行情感分析后才能进入线上内容供给池。依托于美团机器学习平台 MLP 搭建了情感分析离线计算服务，业务使用方仅需要简单的配置输入 / 输出数据表以及计算队列即可便捷调用，同时该服务支持了定时调度，可以便捷接入到业务方整体的调度流程中。目前情感分析服务已经为美团内部十多个业务场景提供服务。

图 17　在线实时预期

图 18　离线批量预测

总结与展望

本文总结了美团情感分析技术的演进和在典型业务场景中的应用，包括篇章 / 句子级情感分析、属性级情感分析和观点三元组分析，其中属性级情感分析工作向业界开源了迄今规模最大的基于真实场景的中文属性级情感分析数据集 ASAP，该数据集相关论 文《ASAP: A Chinese Review Dataset Towards Aspect Category Sentiment Analysis and Rating Prediction》被自然语言处理顶会 NAACL2021 录用。在技术迭代上，紧跟预训练语言模型技术的快速发展，结合团队自研的 MT-BERT 模型不

断迭代升级。在业务应用上，依托情感分析技术能力构建了在线实时预测服务和离线批量预测服务，截至目前情感分析服务已经为美团内部十多个业务场景提供服务。

属性级情感分析已经在美团多个场景落地应用，但对于某些领域跨度较大的新场景（比如从餐饮领域到休闲娱乐领域），我们总是需要人工预定义新的属性，并进行一定数量的数据标注。预定义属性需要对每个业务都有深入的理解，在实际中，很难把每个业务的属性都预定义得非常全面，尤其是某些占比不高但具有业务特色的属性。这些成本会对属性级情感分析在新业务场景的快速落地有一定程度的影响。我们也在探索迁移学习、少样本学习、属性自动挖掘等技术在情感分析上的应用，以加速情感分析在新领域快速应用，来满足美团业务的快速发展需求。

参考文献

[1] https://github.com/Meituan-Dianping/asap.

[2] Bu J, Ren L, Zheng S, et al. ASAP: A Chinese Review Dataset Towards Aspect Category Sentiment Analysis and Rating Prediction. In Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. 2021.

[3] https://www.luge.ai/

[4] Zhang, L. , S. Wang , and B. Liu . “Deep Learning for Sentiment Analysis : A Survey.” Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery (2018):e1253.

[5] Liu, Bing. “Sentiment analysis and opinion mining.” Synthesis lectures on human language technologies 5.1 (2012): 1-167.

[6] Peng, Haiyun, et al. “Knowing what, how and why: A near complete solution for aspect-based sentiment analysis.” In Proceedings of the AAAI Conference on Artificial Intelligence. Vol. 34. No. 05. 2020.

[7] Zhang, Chen, et al. “A Multi-task Learning Framework for Opinion Triplet Extraction.” In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings. 2020.

[8] Yoon Kim. 2014. Convolutional neural networks for sentence classification. arXiv preprint arXiv:1408.5882.

[9] Peng Zhou, Wei Shi, Jun Tian, Zhenyu Qi, Bingchen Li,Hongwei Hao, and Bo Xu. 2016. Attention-based bidirectional long short-term memory networks for relation classification. In Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), pages 207 - 212.

[10] Devlin, Jacob, et al. “Bert: Pre-training of deep bidirectional transformers for

language understanding." arXiv preprint arXiv:1810.04805 (2018).
[11] 杨扬、佳昊等 . 美团 BERT 的探索和实践 .
[12] Pontiki, Maria, et al. "Semeval-2016 task 5: Aspect based sentiment analysis." International workshop on semantic evaluation. 2016.
[13] Pontiki, M. , et al. "SemEval-2014 Task 4: Aspect Based Sentiment Analysis." In Proceedings of International Workshop on Semantic Evaluation at (2014).
[14] Yequan Wang, Minlie Huang, and Li Zhao. 2016. Attention-based lstm for aspect-level sentiment classification. In Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, pages 606 - 615.
[15] Sara Sabour, Nicholas Frosst, and Geoffrey E Hinton. 2017. Dynamic routing between capsules. In Advances in neural information processing systems, pages 3856 - 3866.
[16] Chi Sun, Luyao Huang, and Xipeng Qiu. 2019. Utilizing bert for aspect-based sentiment analysis via constructing auxiliary sentence. arXiv preprint arXiv:1903.09588.
[17] Qingnan Jiang, Lei Chen, Ruifeng Xu, Xiang Ao, and Min Yang. 2019. A challenge dataset and effective models for aspect-based sentiment analysis. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), pages 6281 - 6286.
[18] Wu, Zhen, et al. "Grid Tagging Scheme for End-to-End Fine-grained Opinion Extraction." In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings. 2020.
[19] Liu, Yinhan, et al. "Roberta: A robustly optimized bert pretraining approach." arXiv preprint arXiv:1907.11692 (2019).
[20] Clark, Kevin, et al. "Electra: Pre-training text encoders as discriminators rather than generators." arXiv preprint arXiv:2003.10555 (2020). 0- [21] Timothy Dozat and Christopher D. Manning. 2017.Deep biaffine attention for neural dependency parsing. In 5th International Conference on Learning Representations, ICLR 2017.

作者介绍

任磊、佳昊、张辰、杨扬、梦雪、马放、金刚、武威等，均来自美团平台搜索与 NLP 部 NLP 中心。

招聘信息

美团搜索与 NLP 部 /NLP 中心是负责美团人工智能技术研发的核心团队，使命是打造世界一流的自然语言处理核心技术和服务能力。

NLP 中心长期招聘自然语言处理算法专家 / 机器学习算法专家，感兴趣的同学可以将简历发送至 renlei04@meituan.com。具体要求如下。

岗位职责

1. 预训练语言模型前瞻探索，包括但不限于知识驱动预训练、任务型预训练、多模态模型预训练以及跨语言预训练等方向；
2. 负责百亿参数以上超大模型的训练与性能优化；
3. 模型精调前瞻技术探索，包括但不限于 Prompt Tuning、Adapter Tuning 以及各种 Parameter-efficient 的迁移学习等方向；
4. 模型 inference/training 压缩技术前瞻探索，包括但不限于量化、剪枝、张量分析、KD 以及 NAS 等；
5. 完成预训练模型在搜索、推荐、广告等业务场景中的应用并实现业务目标；
6. 参与美团内部 NLP 平台建设和推广

岗位要求

1. 2 年以上相关工作经验，参与过搜索、推荐、广告至少其一领域的算法开发工作，关注行业及学界进展；
2. 扎实的算法基础，熟悉自然语言处理、知识图谱和机器学习技术，对技术开发及应用有热情；
3. 熟悉 Python/Java 等编程语言，有一定的工程能力；
4. 熟悉 Tensorflow、PyTorch 等深度学习框架并有实际项目经验；
5. 熟悉 RNN/CNN/Transformer/BERT/GPT 等 NLP 模型并有过实际项目经验；
6. 目标感强，善于分析和发现问题，拆解简化，能够从日常工作中发现新的空间；
7. 条理性强且有推动力，能够梳理繁杂的工作并建立有效机制，推动上下游配合完成目标。

加分项

熟悉模型训练各 Optimizer 基本原理，了解分布式训练基本方法与框架；
对于最新训练加速方法有所了解，例如混合精度训练、低比特训练、分布式梯度压缩等

细粒度情感分析在到餐场景中的应用

作者：储哲 王璐 润宇 马宁 建林 张琨 刘强

一、背景

作为一家生活服务在线电子商务平台，美团致力于通过科技链接消费者和商户，努力为消费者提供品质生活。到店餐饮（简称到餐）作为美团的核心业务之一，是满足用户堂食消费需求、赋能餐饮商户在线运营的重要平台，在服务百万级别的餐饮商户和亿级别 C 端用户的过程中，积累了海量的用户评论信息（User Generated Content, UGC），包含了用户到店消费体验之后的真情实感，如果能够有效提取其中的关键的情感极性、观点表达，不仅可以辅助更多用户做出消费决策，同时也可以帮助商户收集经营状况的用户反馈信息。

近年来，大规模预训练模型（BERT）、提示学习（Prompt）等 NLP 技术飞速发展。文本分类、序列标注、文本生成各类自然语言处理任务的应用效果得到显著提升，情感分析便是其中最常见的应用形式之一。它的任务目标在于通过 NLP 技术手段对输入文本进行分析、处理、归纳、推理，给出文本情感极性判定的结果。

按照情感极性判定粒度，可以细分为篇章 / 整句粒度情感分析、细粒度情感分析（ABSA, Aspect-based Sentiment Analysis）[1]。一般而言，细粒度情感分析的任务目标主要围绕属性（Aspect Term）、观点（Opinion Term）、情感（Sentiment Polarity）三要素展开，可以拆分为属性抽取、观点抽取以及属性 – 观点对的情感倾向判定三个级联任务 [2-5]。例如，对于给定的用户评论“这家店环境不错，但服务很糟糕”，预期的输出结果为（环境，不错，正向）、（服务，糟糕，负向）。

图 1　ABSA 子任务 [5]

到餐算法团队结合到餐业务供给侧、平台侧、需求侧的业务场景，为核心业务链路的智能化提供高效、优质的算法解决方案，通过算法能力辅助业务降本提效。本文结合到餐 B/C 端业务场景，探索细粒度情感分析技术在用户评价挖掘方向的应用实践。

二、目标回顾

2.1 业务问题

秉承“帮大家吃得更好，生活更好”的使命，到餐面向消费者提供包括套餐、代金券、买单、预订等在内的丰富产品和服务，并通过黑珍珠餐厅指南、大众点评必吃榜等榜单，以及搜索、查询、评价等，帮助消费者更好地作出消费决策。同时，为商家提供一站式的营销服务，帮助餐饮商户沉淀口碑、获取用户、增加复购等，进而轻松管理餐厅。

随着餐饮连锁化加速、行业竞争格局激烈，商户管理宽幅和难度逐步加大，商户的经营要求更加精细，数据管理意识更加迫切。用户历史评论中蕴含着大量用户消费后的反馈，是情感分析的重要组成部分，不仅能够描述消费感受，同时也能反映出就餐环境的好坏。因此，做好情感分析有利于帮助餐饮门店提升服务质量，也能够更好地促进消费体验。

(a) (b) (c)

图 2 评论示例：(a) 菜品评价，(b) 服务评价，(c) 食安评价

UGC 评价分析，主要是从评论文本中挖掘出菜品、服务、食品安全（简称食安）等方面相关信息，获取用户在各个维度的细粒度情感，细致刻画商家的服务现状，如上图 2 所示。对于餐饮商户，菜品、服务、食安评价分析问题可以拆解如下：

- 菜品评价，主要包括用户评论中的菜品识别、评价属性提取、菜品观点提取、观点情感分类；
- 服务评价，主要包括用户评论中评价属性提取、服务方面观点提取、观点情感分类；
- 食安评价，主要包括用户评论中评价属性提取、食安方面观点提取、观点情感分类。

其中问题 2 和 3 是典型的三元组抽取任务，即识别服务或食安方面的（属性，观点，情感）。对于问题 1，在服务、食安评价问题的基础上，菜品评价需要识别评论中提及的菜品，相比业界四元组（属性，观点，属性类别，情感）[6] 抽取任务，到餐场景下主要为（菜品，属性，观点，情感）四元组的识别。

2.2 技术调研

在美团内部，我们针对 UGC 评价分析问题，调研了相关工作成果，主要为基于 MT-BERT 预训练模型开发了多任务模型，试图解决情感分析中的 ACSA (Aspect-Category Setiment Analysis) 问题以及（属性，观点，情感）三元组抽取问题，并实现了句子粒度的情感分类工具开发，同时开源了基于真实场景的中文属性级情感分析数据集 ASAP[7-9]。但对于美团到餐业务来说，我们需要基于具体场景提出针对性的解决方案，如四元组抽取任务，不能直接复用其他团队的相关技术和工具，因此有必要建设服务于到餐业务场景的细粒度情感分析技术。

在业界，我们也调研了行业其他团队如腾讯、阿里在细粒度情感分析方面的相关研究。2019 年腾讯 AI Lab 和阿里达摩院合作 [3]，提出了基于两个堆叠的 LSTM 和三个组件（边界引导、情感一致性和意见增强）的模型，将“BIOES”标注体系与情感正向（Positive）、中性（Neutral）、负向（Negative）结合形成统一标签，可以同时识别属性和情感。同年，阿里达摩院提出了 BERT+E2E-ABSA 模型结构，进一步解决属性和情感的联合抽取问题 [10]，同时提出（属性，观点，情感）[2] 三元组抽取任务，并给出了两阶段解决框架，首先分别识别出属性（情感融合为统一标签）和观点，然后判断属性 - 观点是否配对。

自此，业界后续研究开始向三元组联合抽取展开 [11-14]。2021 年 2 月，华为云 [6] 提出（属性，观点，属性类别，情感）四元组抽取多任务模型，其中一个任务识别属性和观点，另一个任务识别属性类别和情感。2021 年 4 月，腾讯 [15] 引入 Aspect-Sentiment-Opinion Triplet Extraction（ASOTE）任务，提出了一个位置感知的 BERT 三阶段模型，解决了（属性，观点，情感）三元组抽取问题。

调研机构	行业	预训练模型	细粒度情感分析问题	阅读理解问题	三元组问题	四元组问题	联合抽取问题
阿里达摩院 [2,10]	电子商务	√	√	X	√	X	√
华为云 [6]	云服务	√	√	X	X	√	√
腾讯 [15]	社交	√	√	X	√	X	X
美团到餐	本地生活	√	√	√	√	√	√

从学术界来看，更关注于如何更好地进行实体抽取、情感分类以及多任务的联合抽取，可能会忽略工业界落地更关注的计算时效性（如多维度标注与情感维度整合，增加计算、存储资源消耗，在有限资源下时长延迟）、效果准确性（如任务模块端到端开发，忽略业务的个性化，直接复用导致准确性降低）等方面要求，导致相关技术方法并不能直接应用于业务场景，需要进一步开发完善才能实现业务的落地。

如上表所示，针对以上调研，我们借鉴了美团搜索与 NLP 部在三元组细粒度情感分析方面的经验，拆解到餐四元组抽取问题，并结合学界最先进的阅读理解、注意力机制等方面的实体抽取、情感分类经验，设计开发了应用于到餐业务的细粒度情感分析解决方案。

2.3 技术目标

如上文所述，菜品评价主要关注菜品、评价属性、菜品观点和观点情感，而服务、食安评价问题，主要关注服务或食安方面的评价属性、观点和情感。就细粒度情感分析任务而言，可以看出，前一个问题涉及四元组信息，而后两个问题仅涉及三元组信息。

假设给定一个长度为n的句子x_i，令$T_n=\{(d,a,o,s)\}$和$T_m=\{(a,o,s)\}$，$T_n=\{(d,a,o,s)\}$表示句子x_i中标注的菜品实体$(d,\underline{dish})$、评价属性$(a,\underline{aspect})$、观点$(o,\underline{opinion})$和情感极性$(s,\underline{sentiment})$，其中情感包括{正向，中性，负向，未提及}。$T_m=\{(a,o,s)\}$表示句子$x_i$中标注的服务或食安方面的评价属性、观点和情感。

对于菜品评价，训练集为$D_n=\{(x_i,T_n)\}$，似然函数为：

$$L(D_n)=\prod_{i=1}^{|D_n|}\prod_{(d,a,o,s)\in T_n}P((d,a,o,s)|x_i), \qquad (1)$$

进一步，对数似然函数为：

$$\begin{aligned} l(D_n)&=\sum_{i=1}^{|D_n|}\sum_{(d,a,o,s)\in T_n}logP((d,a,o,s)|x_i)\\ &=\alpha\sum_{i=1}^{|D_n|}\sum_{d\in T_n}logP(d|x_i)+\beta\sum_{i=1}^{|D_n|}\sum_{d\in T_n}\sum_{a\in T_n|d}logP(a|d,x_i)+\gamma\sum_{i=1}^{|D_n|}\sum_{d\in T_n}\sum_{a\in T_n|d}\left(\lambda_1\sum_{o\in T_n|d,a}logP(o|d,a,x_i)+\lambda_2\sum_{s\in T_n|d,a}logP(s|d,a,x_i)\right) \end{aligned} \qquad (2)$$

其中，超参数$(\alpha,\beta,\gamma,\lambda_1,\lambda_2)\in[0,1]$。由公式(2)可推测，四元组细粒度情感分析问题可以拆解为：给定文本的菜品实体抽取、描述菜品的评价属性抽取、评价属性对应的观点抽取以及情感分类。

对于服务、食安评价，训练集为$D_m=\{(x_i,T_m)\}$，似然函数为：

$$L(D_m)=\prod_{i=1}^{|D_m|}\prod_{(a,o,s)\in T_m}P((a,o,s)|x_i), \qquad (3)$$

进一步，对数似然函数为：

$$\begin{aligned} l(D_m)&=\sum_{i=1}^{|D_m|}\sum_{(a,o,s)\in T_m}logP((a,o,s)|x_i)\\ &=\alpha\sum_{i=1}^{|D_m|}\sum_{a\in T_m}logP(a|x_i)+\beta\sum_{i=1}^{|D_m|}\sum_{a\in T_m}\left(\lambda_1\sum_{o\in T_n|a}logP(o|a,x_i)+\lambda_2\sum_{s\in T_n|a}logP(s|a,x_i)\right) \end{aligned} \qquad (4)$$

其中，超参数$(\alpha,\beta,\lambda_1,\lambda_2)\in[0,1]$。由公式(4)可推测，三元组问题可以拆解为：给定文本的服务或食安属性抽取、评价属性对应的观点抽取以及情感分类。

特别地，公式(2)和(4)最后一项成立的原因是给定属性a条件下，o与s是条件独立的，即$P(o,s|a,x_i)=P(o|a,x_i)*P(s|a,x_i)$，目标是最大化对数似然函数。总而言之，基于联合概率分布拆解，能够将复杂问题拆分为多个简单问题建模。基于以上分析，建立菜品、服务、食安维度等细粒度情感分析模型。算法的目标就是通过菜品评价、服务评价以及食安评价的建模，从UGC文本中挖掘用户评论信息，如用户消费偏好、用户就餐环境以及场景反馈等。值得一提的是这里提到的四元组(d,a,o,s)和三元组(a,o,s)，不仅仅解决以上场景的问题，而是具有更强的泛化性，对于类似场景的细粒度情感分析也同样适用。

在细粒度情感分析中，训练、测试数据$D_n=\{(x_i,T_n)\}$主要来源于UGC标注数据。其中，假设UGC文本间标注是独立的，则任意取一定数据的UGC文本进行测试，其余标注为训练样本，进行模型训练。技术评估指标是任务的准确率、召回率以及多类别平均F1值。此外，业务层面主要依赖于文本抽取标签的Badcase率反映算法的准确性。

2.4 主要挑战

由于三元组问题可以看作是四元组问题的子问题，不失一般性，下文将重点阐述四元组相关技术挑战。

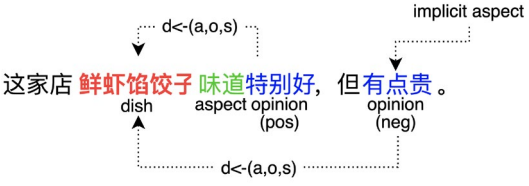


图 3　到餐细粒度情感分析中的 UGC 示例

如图3的示例所示，提取的四元组为"鲜虾馅饺子-口味-特别好-正向"和"鲜虾馅饺子-(?)-有点贵-负向"。在到餐场景中，UGC文本细粒度情感分析较复杂，主要存在挑战：给定一条评论，可能包含多个四元组(d, a, o, s)且存在实体间一对多或者多对一的关系，以及可能存在a缺失情况；如何准确识别用户所有细粒度评论情感倾向？对于以上挑战，可以拆解为如下问题。

问题1：如何准确识别可能存在的多个四元组?

对于一条存在多个四元组的评论，通过序列标注，直接同时识别多个四元组是不现实的。为此，我们在标注数据预处理时，将评论中四元组分别抽取，各自组成一条独立样本，进行训练、预测。此外，如上文所述，四元组的实体间存在一对多或者多对一的关系，而且实体间的间距通常不固定，存在远程抽取的可能。针对这一问题，一个较有效的实践经验是进行分层分块抽取，即pipeline识别。基于pipeline的方法，如公式(2)所示，则需要分别处理：给定文本，菜品实体抽取 $(P(d|x_i))$、描述菜品的评价属性抽取 $(P(a|d, x_i))$、评价属性对应的观点抽取 $(P(o|d, a, x_i))$ 以及情感分类 $(P(s|d, a, x_i))$。

定义交叉熵损失函数为：

$$f_{CE}(\hat{y}_{i,k}, y_{i,k}) = -\frac{1}{n}\sum_{i=1}^{n}\sum_{k}\hat{y}_{i,k}log(y_{i,k}) \qquad (5)$$

其中，$\hat{y}_{i,k}$表示每个模块真实label分布，k表示分类类别数。

那么，当前问题的损失函数集合为

$$\begin{aligned} L(D_n)|_{i=1,2,...,|D_n|} &= \left(L_{d|x_i}, L_{a|d,x_i}, L_{o|d,a,x_i}, L_{s|d,a,x_i}\right) \\ &= \left(f_{CE}(\hat{P}_{d|x_i}, P_{d|x_i}), f_{CE}(\hat{P}_{a|d,x_i}, P_{a|d,x_i}), f_{CE}(\hat{P}_{o|d,a,x_i}, P_{o|d,a,x_i}), f_{CE}(\hat{P}_{s|d,a,x_i}, P_{s|d,a,x_i})\right) \end{aligned} \qquad (6)$$

其中，目标$min\ L(D_n)$，将分别针对每一个任务进行逐个最小化。

问题2：如何解决aspect有缺失的四元组识别?

如图3所示，真实场景可能存在四元组的评价属性缺失。针对这类问题，我们将问题拆解为三个任务：给定评价文本，菜品实体抽取 $(P(d|x_i))$、评价观点抽取 $(P(o|d, x_i))$ 以及观点和情感的分类 $(P(c_a, s|d, o, x_i))$ ，其中c_a是隐藏评价属性的类别。则损失函数为：

$$\begin{aligned} L(D_n)|_{i=1,2,...,|D_n|} &= (L_{d|x_i}, L_{o|d,x_i}, L_{c_a,s|d,o,x_i}) \\ &= \left(f_{CE}(\hat{P}_{d|x_i}, P_{d|x_i}), f_{CE}(\hat{P}_{o|d,x_i}, P_{o|d,x_i}), f_{CE}(\hat{P}_{c_a,s|d,o,x_i}, P_{c_a,s|d,o,x_i})\right) \end{aligned} \qquad (7)$$

问题 3：如何同时对四元组抽取、识别，减少 pipeline 方法的错误累计影响?

减少 pipeline 方法的错误累计影响，典型的解决方案是提出同时处理信息抽取和分类任务，即多任务学习。传统的方法是直接尝试多任务学习的思路，但过程中忽略了实体间依赖的关系，甚至远程关联关系 [2]。当前也在尝试直接将四元组转化成多任务学习过程，将来期望通过建立实体间 pair 或 triplet 关系，进行联合抽取、识别。

综上，对于**问题 1** 和**问题 2**，我们会按照 pipeline 识别的结果，再利用策略进行抽取结果的优化；对于**问题 3**，整合实体、关系及分类任务，进行联合学习，将有助于减少 pipeline 方法的错误累计影响。

三、细粒度情感分析实践

3.1 Pipeline 方法

如上文 2.3 的问题 2 所述，我们采用 pipeline 的方法，将四元组抽取问题拆解为三个任务，分为实体识别、观点抽取、观点类别和情感分类，如下图 4 所示：

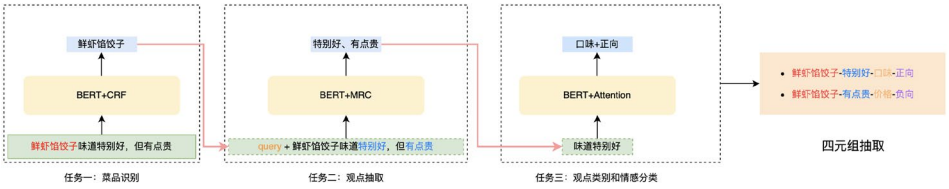


图 4 三阶段模型

3.1.1 实体识别

自 2018 年 BERT[16] 出现以后，NER 模型由传统的 LSTM+CRF 替换为 BERT+CRF（或者 BERT+LSTM+CRF），一度是业界 NER 任务的 SOTA 模型，近两年来 NER 任务主要从以下两个方面进行改进：

- 加入额外的特征 [17-19]：如字特征、词特征、词性特征、句法特征、知识图谱表征；
- 转换任务形式 [20-21]：将 NER 任务转化为问答（QA, Question Answering）任务或者机器翻译任务。

考虑到引入额外特征需要构建人工词典，以及转化问答任务形式依赖于人工模板，成本较高，因此采用 BERT+CRF 模型。

学习率调整，模型策略调优。在实验过程中，我们发现 BERT+CRF 相比简单的 BERT+Softmax 效果提升甚微，究其原因，由于预训练模型经过微调之后可以学习到具有明显区分度的特征，导致增加 CRF 层对实体识别结果几乎没有影响。然而，一个好的 CRF 转移矩阵显然对预测是有帮助的，可以为最后预测的标签添加约束来保证预测结果的合理性。进一步实验后发现，通过调整 BERT 和 CRF 层的学习率，如 BERT 使用较小的学习率而 CRF 层使用 100 倍于 BERT 的学习率

（即 e2/e1>100，如图 5 所示），最终 BERT+CRF 的效果相比 BERT+Softmax 有了较明显的提升。此外，在传统 NER 模型 LSTM+CRF 基础上，我们也实验了 BERT+LSTM+CRF，但效果居然有些许下降，而且预测时间也增加了，因此最终没有引入 LSTM 层。

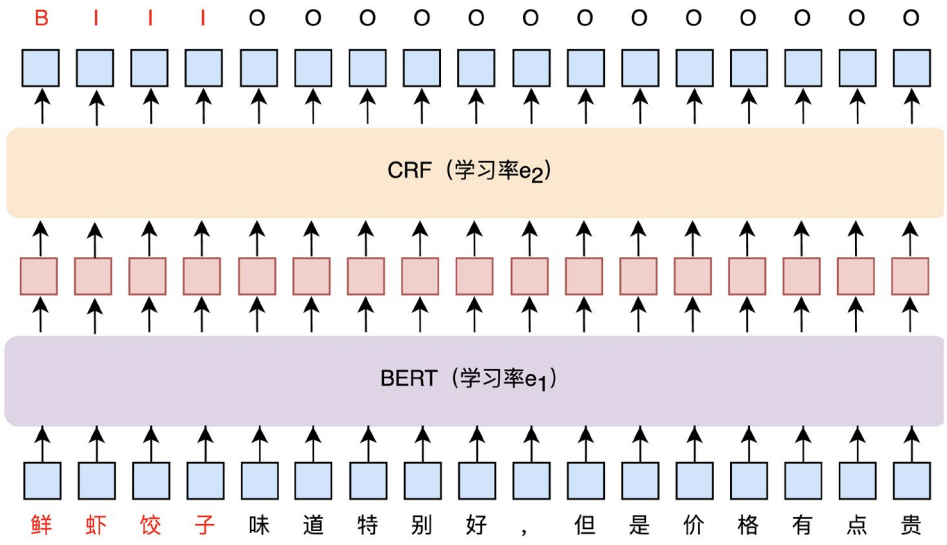


图 5　BERT+CRF（BIO 标记）

3.1.2 观点抽取

观点抽取任务在业界也称为 Target-oriented Opinion Words Extraction（TOWE），旨在从评论句子中抽取出给定目标对应的观点词。观点抽取也可以看作是一种 NER 任务，但若评论涉及多个实体和观点，如何准确抽取所有“实体 - 观点”关系是一个技术挑战。借鉴 MRC（Machine Reading Comprehension）任务的思想，通过构建合理的 Query 引入先验知识，辅助观点抽取。

QA 任务形式，观点抽取建模。如图 6 所示，模型整体由预训练层和输出层两部分组成。输出层我们使用了常规 QA 任务输出，包括开始标签（Start Label）和结束标签（End Label），但需要人工设计 Quey。参考论文 [20] 经验，以图 3 为例，实验发现 Query 设计为“找出鲜虾馅饺子口味、口感、分量、食材、卖相、价格、卫生以及整

体评价”效果最好，可能融入了观点描述信息，更加有助于观点抽取。考虑到 QA 任务天然有类别不平衡的问题，因此损失函数引入针对类别不平衡的 Focal Loss，用于提升观点抽取模型的效果。由于观点抽取也可以看作是 NER 任务，故我们尝试将输出层设计为 CRF 层，但实验效果并不理想，可能由于观点语句长度不一且比较个性化，影响模型识别。另一方面，考虑到 Google 中文预训练模型 BERT 是以字粒度为切分，没有考虑到传统 NLP 中的中文分词，在预训练层我们将 BERT 模型替换为哈工大开源的中文预训练模型，如 BERT-wwm-ext、RoBERTa-wwm 等，最终模型效果取得进一步提升。

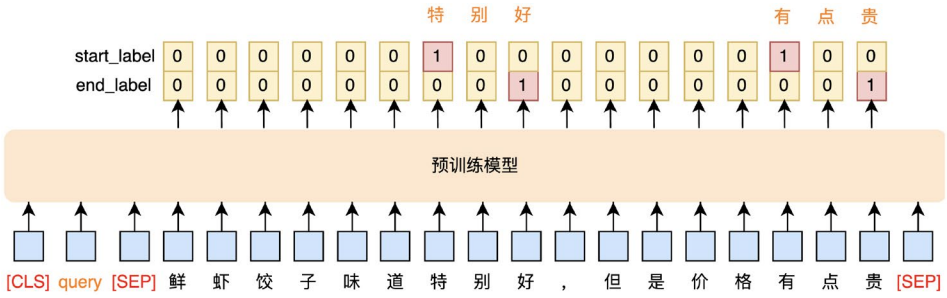


图 6　BERT+MRC（通过开始标签和结束标签，抽取出观点词：特别好、有点贵）

3.1.3 观点类别和情感分类

观点类别和情感分类可以看作两个分类任务，其中菜品评价四元组任务的观点类别包含口感、口味、分量、食材、卖相、价格、卫生、菜品整体等 8 个标签，而情感包含正向、中性、负向、未提及等 4 个标签，都是业务预定义好的。考虑到用户评论提及某个菜品的观点可能涉及多个维度，若每个维度单独建模，需要构建多个模型，较复杂且维护困难。结合 ATAE-LSTM[22] 和 NLP 中心 [7-9] 情感分析的经验和到餐业务特点，模型整体结构设计为多任务多分类学习框架。

多任务多分类模型，联合建模观点类别和情感。如图 7 所示，模型整体分为两个部分，分别为 BERT 共享层和 Attention 独享层，其中 BERT 共享层学习观点 Embedding 表示，Attention 独享层学习观点在各个观点类别的情感倾向。考虑到评论中各部分会聚焦不同的观点维度，通过引入 Attention 结构，使得模型更加关注特定维度相关的

文本信息，进而提升整体效果。

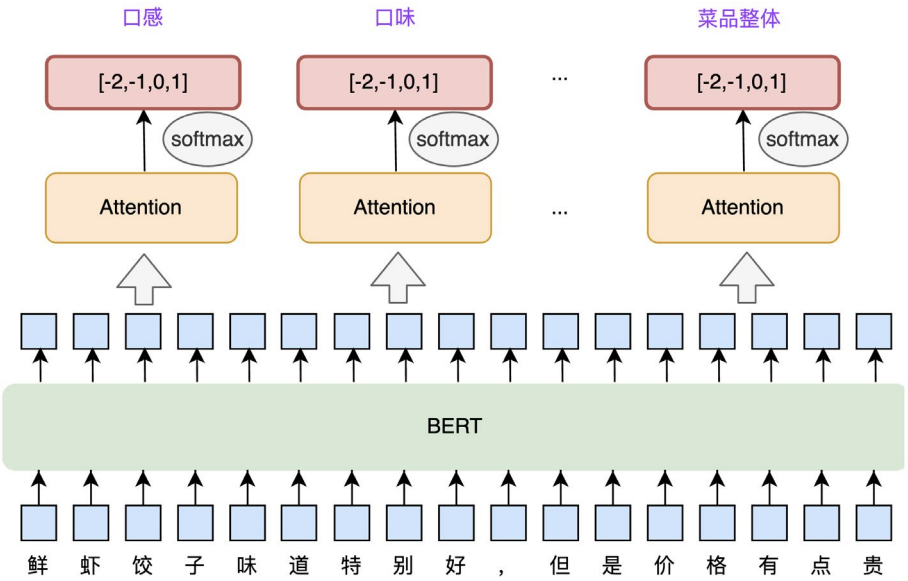


图 7　BERT+Attention

3.2 联合学习

pipeline 方法的优点是将目标问题拆分为多个子模块问题，对子模块分别优化，通过后处理能在一定程度上解决实体间多对多关系的问题。然而，pipeline 方法也会存在一些致命缺陷，主要包括：

- 误差传播，实体识别模块的错误会影响到观点抽取模型的性能；
- 忽略了任务之间的关联性，如实体和观点往往一起出现，如果可以知道观点，那么也能判断出所描述的实体，而 pipeline 方法显然不能利用这些信息；
- 信息冗余，由于需要对识别出来的实体都要进行观点抽取，以及提取出的观点都要进行分类，产生一些无效的匹配对，提升错误率。

参考业界情感分析联合学习现状，主要为（属性，观点，情感）三元组联合抽取。结合到餐业务场景特点（如挑战 2.3 的问题 2 所述），整体设计为两阶段模型，第一阶段为对菜品实体、观点和情感联合训练，第二阶段为对观点进行分类，进而得到四元组

识别的结果。

3.2.1 三元组联合抽取

目前在学术界，三元组（属性，观点，情感）联合抽取的方法主要包括序列标注方法[11]、QA 方法[5,12]、生成式方法[13,14]等。结合菜品分析场景和 pipeline 方法中观点抽取模块的经验，我们采取了 QA 式的联合抽取方法，主要参考模型 Dual-MRC[5]。

Dual-MRC 模型的改进，三元组联合抽取建模。在模型设计过程中，由于 Dual-MRC 模型分类情感倾向是对某个属性的整体评价，即一个属性只对应一个情感。然而，在到餐业务场景中，新增了菜品实体的识别，同时 UGC 评论中存在对同一个菜品实体包含不同观点及情感倾向。如图 3 所示，“味道特别好”表达了对“鲜虾饺子”正向情感，而“有点贵”显然表达了负面情感。因此，我们对 Dual-MRC 模型进行了改造，将观点和情感标签整合成统一标签。如图 8 所示，到餐 Dual-MRC 整体结构基于双塔 BERT 模型，通过引入两个 Query，左边负责抽取菜品实体，右边负责抽取观点和观点情感，从而实现三元组联合抽取。

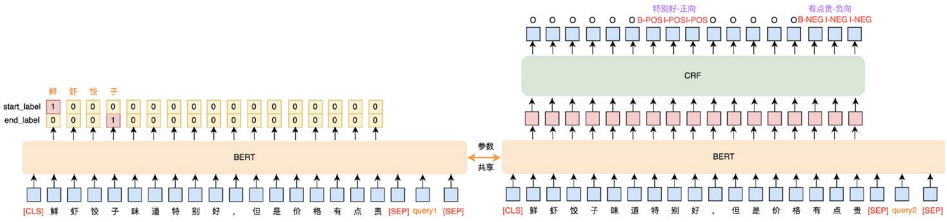


图 8　结合到餐业务特点改进的 Dual-MRC 模型

模型结构说明：

- 整体是由两个部分组成，左边 BERT 抽取菜品实体，右边 BERT 抽取观点和观点情感，将观点和情感构成统一标签 B-{POS，NEU，NEG}，I-{POS，NEU，NEG} 以及 O，其中未提及情感被整合到 O 标签中；
- 参考 pipeline 方法经验，构建两个 Quey，左边 Quey1 构建为“找出评论中的菜品”，右边 Quey2 构建为“找出鲜虾馅饺子口味、口感、分量、食材、卖相、价格、卫生以及整体评价”；

- 训练阶段，对于左边标注的每个菜品实体，都需要重复右边流程，两边模型共享参数进行训练；预测阶段，由于实体不可知，采用 pipeline 方式，首先左边部分抽取出所有的菜品实体，然后对于每个实体输入到右边部分，抽取出观点和观点情感。

在此基础上，我们也探索了四元组联合抽取的可能，具体操作为对右边 Query2 进行改造，如“找出鲜虾馅饺子口味评价”，对于每个观点类别都需要构建 Query 进行预测，从而实现四元组联合抽取。但考虑计算量级较大且耗时较长，最终将观点类别另做预测。

3.2.2 观点类别分类

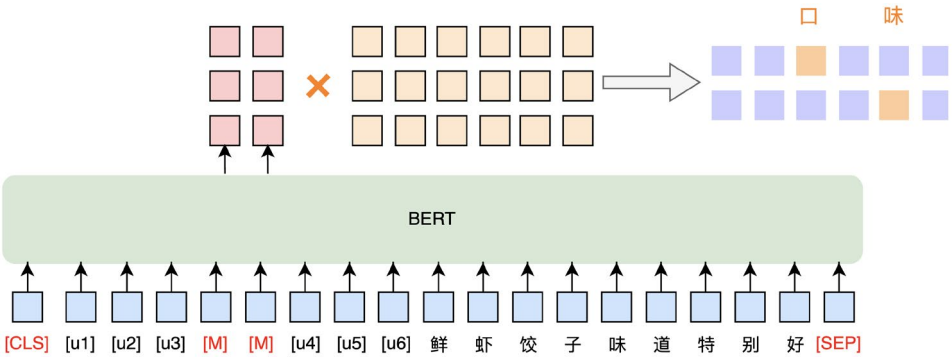


图 9　BERT+P-tuning

观点类别分类，显然是一个文本分类问题，通常做法是基于 BERT 分类，取 [CLS] 位置的 Embedding，接一个全连接层和 Softmax 层即可。在到餐业务场景中，主要面临少样本问题，参考业界 NLP 少样本解决方法，以基于对比学习的 R-drop[23] 方法和基于 Prompt[24] 的第四范式为代表。我们在 BERT 模型结构基础上，分别实验了 Prompt 模板方法（如图 9 所示）和 R-drop 数据增强（如图 10 所示）。其中，Prompt 模板主要借鉴 P-tuning[25] 的思想，采取自动化构建模板的方式，基于 MLM 任务解决问题。

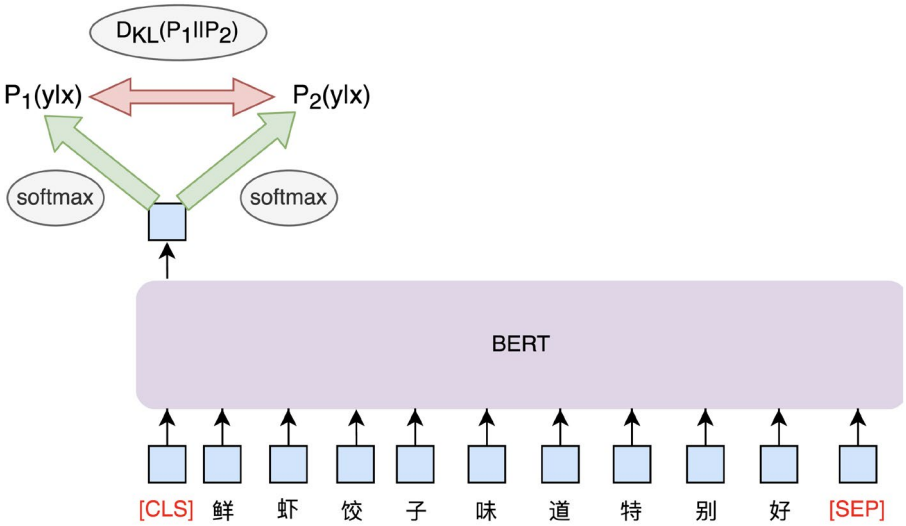


图 10　BERT+R-drop

图 9 中 [u1]~[u6] 代表 BERT 词表里边的 [unused1] ~ [unused6]，即使用未出现的 Token 构建模板，Token 数目为超参数。实验结果发现，基于 BERT 的预训练模型，结合 P-tuning 或 R-drop 结构，分类效果都能得到一定的提升，且 P-tuning 的效果略优于 R-drop，后续还会持续探索少样本解决方法。

四、在到餐业务中的应用

4.1 模型效果对比

利用到餐的 UGC 标注数据，对于四元组识别进行了整体效果测评，最终以整体四元组的精确率和召回率计算 F1 值作为性能评估指标。如图 11 所示，采用经典的 BERT+CRF 模型进行实体抽取，在到餐评论标注数据仅达到 0.61 的 F1，经过学习率等调参（Baseline Tuning）优化之后，F1 值提升 2.61%。如上文所述，在观点抽取模块中，将序列标注问题转化成问答（QA）问题后，采用 BERT+MRC 模型，F1 显著提升至 0.64，提升了 5.9%，表明问题转化获得较大收益。此外，采用哈工大中文预训练 BERT 仍取得一定幅度的提升，F1 提升至 0.65。注意，图 11 中的模型迭代表示四元组问题中重点优化模块的模型，最终评测四元组整体效果来进行对比分析。

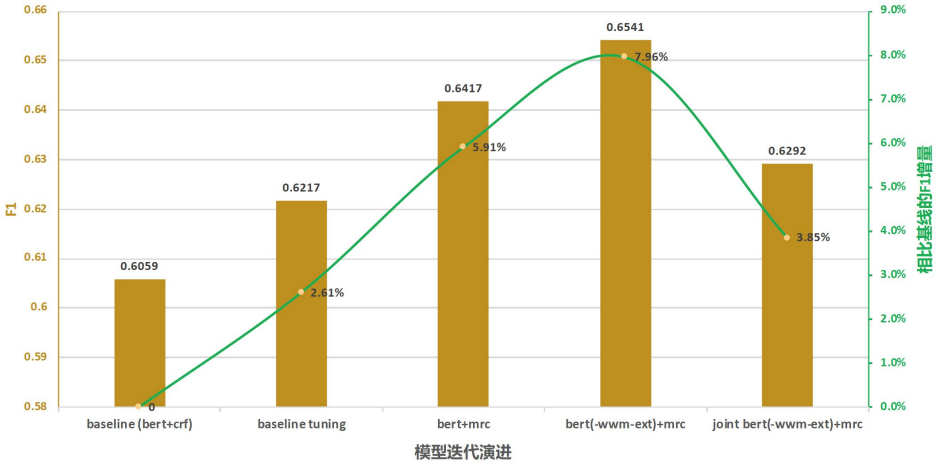


图 11　在到餐场景细粒度情感分析效果演进

4.2 业务应用场景

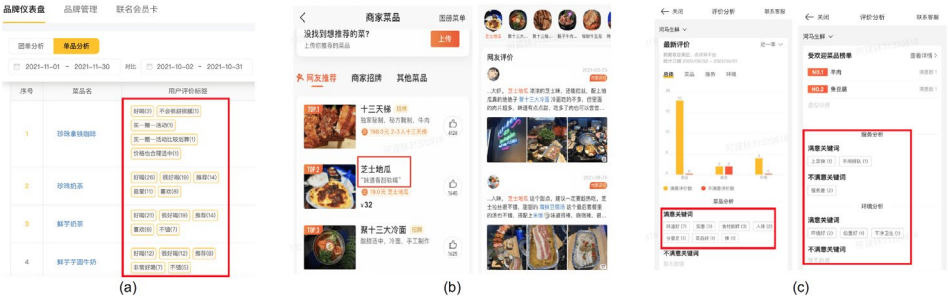


图 12　到餐场景细粒度情感分析的应用：(a) 品牌仪表盘，(b) 到餐商户菜品信息优化，(c) 开店宝评价管理

品牌仪表盘

品牌仪表盘作为旗舰店能力的重要环节，提供品牌层面的数据服务，助力生意增长。产品定位为头部餐饮品牌的数据中心，具备基础的数据披露能力，通过量化业务效果，指导商户经营决策。由于大客在平台沉淀了丰富的线上信息（大量的交易 / 流量 / 评论数据），可挖掘分析空间较大。应用细粒度情感分析技术从评论数据中挖掘菜品维度、服务维度、食品安全维度相关信息，量化商户经营表现，指导经营动作。关于菜品的用户反馈监控，品牌商户更关注菜品、口味、口感等维度的用户反馈。如上文

所述模型迭代后，菜品情感、口味情感、口感情感识别准确率都得到一定的提升。

到餐商户菜品信息优化

随着到餐加强了菜品信息建设，主要包括在生产层面上，整合了商户各来源菜品数据，建设了商户菜品中心，并优化了 C 端菜品 UGC 上传功能，有效补充 UGC 菜品生产；在消费层面上，整合了商户通菜品和网友推荐菜菜品，并基于菜品信息的完善，优化了 C 端菜品信息的内容聚合及展示消费。同时配合到餐业务，持续通过评价信息生产建设赋能，更多的引导用户从评价生产层面进行商户菜品的描述介绍。主要针对到餐商户菜品关联的评价信息，进行信息联动与展示层面的优化，相比迭代前，有评价菜品覆盖率得到较大的提升。

开店宝评价管理

商家通过提供餐饮服务来获取用户，用户消费后通过评价给商家以反馈，促使商家去不断优化，提供更好的服务，从而获取更多的用户，达到正向循环。评价分析的意义在于建立起评价和餐饮服务之间的通道，实现评价对服务的正向促进循环。通过分析评价内容，帮助商家发现餐厅在菜品、服务、环境等方面，做得好和做得不好的地方，进而针对性的改善。相比迭代前，菜品、服务、环境维度关联评论数得到很大的提升。

五、未来展望

经过近一年的建设，情感分析相关能力不但成功应用到到餐商户经营、供应链等业务，而且优化了多源菜品信息，辅助品牌商户进行用户反馈监控，提升商户服务能力。在联合学习探索上，目前主要将四元组问题转化为两阶段模型，如图 11 所示，F1 值有所下降，仅达到 0.63。究其原因，可能是在三元组联合抽取模型中，忽略了实体间的关系，尤其长程关系（如上文 2.4 的问题 3 所述），导致性能不足预期。接下来，将进一步提升情感分析四元组抽取能力，挖掘 UGC 中用户的核心需求以及重要反馈。在技术方面，将持续进行模型迭代演进，主要涉及：

- **持续优化现有模型，保证质量的同时也要提升效率**

  实验结果还有很大的改进空间，需要进一步探索模型优化方法，如优化预训练模型，使用 MT-BERT 等，以及在联合抽取中进一步引入实体间关系，来提升四元组抽取的性能。

- **深度探索情感分析领域，建设四元组联合抽取模型**

  主要通过改造 Query 实现四元组抽取，但是计算量级较大，需要探索模型结构优化，减少冗余的计算量，使其满足四元组联合抽取。

- **建设细粒度情感分析通用框架**

  到餐场景涉及多个情感分析场景，需要建设灵活方便的通用框架，有助于快速支持业务，以及减少资源消耗。

未来，团队将持续优化应用技术，解决到餐业务场景中的情感分析需求。细粒度情感分析是具有挑战和前景的任务，到店餐饮算法团队将和各位读者共同持续探索和研究。

六、参考文献

[1] Liu, B. 2012. Sentiment analysis and opinion mining. Synthesis lectures on human language technologies 5(1):1 – 167.

[2] Peng, H. Xu, L. Bing, L. Huang, F. Lu, W. and Si, L.2020. Knowing What, How and Why: A Near Complete Solution for Aspect-Based Sentiment Analysis. In AAAI, 8600 – 8607.

[3] Li, X. Bing, L. Li, P. and Lam, W. 2019a. A unified model for opinion target extraction and target sentiment prediction. In AAAI, 6714 – 6721.

[4] Zhao, H. Huang, L. Zhang, R. Lu, Q. and Xue, H. 2020. SpanMlt: A Span-based Multi-Task Learning Framework for Pair-wise Aspect and Opinion Terms Extraction. In ACL, 3239 – 3248.

[5] Y. Mao, Y. Shen, C. Yu, and L. Cai. 2021. A joint training dual-mrc framework for aspect based sentiment analysis. arXiv preprint arXiv:2101.00816.

[6] 华为云细粒度文本情感分析及应用

[7] 杨扬、佳昊等 . 美团 BERT 的探索和实践 .

[8] 任磊，步佳昊等 . 情感分析技术在美团的探索与应用 .

[9] Bu J, Ren L, Zheng S, et al. ASAP: A Chinese Review Dataset Towards Aspect Category Sentiment Analysis and Rating Prediction. In Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational

Linguistics: Human Language Technologies. 2021.
[10] Xin Li, Lidong Bing, Wenxuan Zhang, and Wai Lam. Exploiting BERT for end-to-end aspect-based sentiment analysis. In W-NUT@EMNLP, 2019.
[11] Xu, L. Li, H. Lu, W. and Bing, L. 2020. Position-Aware Tagging for Aspect Sentiment Triplet Extraction. In EMNLP, 2339 – 2349.
[12] Chen, S. Wang, Y. Liu, J. and Wang, Y. 2021a. Bidirectional Machine Reading Comprehension for Aspect Sentiment Triplet Extraction. In AAAI.
[13] Yan, H. Dai, J. Qiu, X. Zhang, Z. et al. 2021. A Unified Generative Framework for Aspect-Based Sentiment Analysis. arXiv preprint arXiv:2106.04300.
[14] Wenxuan Zhang, Xin Li, Yang Deng, Lidong Bing, and Wai Lam. 2021. Towards Generative Aspect-Based Sentiment Analysis. In ACL/IJCNLP 2021, 504 – 510.
[15] Li Yuncong, Fang Wang, Zhang Wenjun, Sheng-hua Zhong, Cunxiang Yin, & Yancheng He. 2021. A More Fine-Grained Aspect-Sentiment-Opinion Triplet Extraction Task. arXiv: Computation and Language.
[16] Devlin, J. Chang, M.-W. Lee, K. and Toutanova, K. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In NAACL-HLT, 4171 – 4186.
[17] Yue Zhang and Jie Yang. 2018. Chinese ner using lattice lstm. arXiv preprint arXiv:1805.02023.
[18] Li, X. Yan, H. Qiu, X. and Huang, X. 2020. FLAT: Chinese NER Using Flat-Lattice Transformer. arXiv preprint arXiv:2004.11795 .
[19] Tareq Al-Moslmi, Marc Gallofré Ocaña, Andreas L. Opdahl, and Csaba Veres. 2020. Named entity extraction for knowledge graphs: A literature overview. IEEE Access 8 (2020), 32862 – 32881.
[20] X. Li, J. Feng, Y. Meng, Q. Han, F. Wu, and J. Li. 2020. A unified MRC framework for named entity recognition. In ACL, 5849 – 5859.
[21] Jana Strakova, Milan Straka, and Jan Hajic. 2019. Neural architectures for nested ner through linearization. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, 5326 – 5331.
[22] Yequan Wang, Minlie Huang, Li Zhao, and Xiaoyan Zhu. 2016. Attention-based lstm for aspect-level sentiment classification. In Proceedings of the conference on empirical methods in natural language processing, 606 – 615.
[23] Liang X, Wu L, Li J, et al. R-Drop: Regularized Dropout for Neural Networks[J]. arXiv preprint arXiv:2106.14448, 2021.
[24] P. Liu, W. Yuan, J. Fu, Z. Jiang, H. Hayashi, and G. Neubig. 2021. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. arXiv preprint arXiv:2107.13586.
[25] X. Liu, Y. Zheng, Z. Du, M. Ding, Y. Qian, Z. Yang, and J. Tang. 2021. Gpt understands, too. arXiv preprint arXiv:2103.10385.

七、术语解释

术语	解释
ABSA	细粒度情感分析，Aspect-based Sentiment Analysis
NER	命名实体识别，Named Entity Recognition
TOWE	面向目标的观点词抽取，Target-oriented Opinion Words Extraction
MRC	阅读理解，Machine Reading Comprehension
MLM	语言掩码模型，Masked Language Model
BERT	基于变换器的双向编码器表示，Bidirectional Encoder Representations from Transformers
CRF	条件随机场，Conditional Random Fields
LSTM	长短期记忆，Long Short-Term Memory
R-drop	基于 dropout 的正则策略，regularization strategy upon dropout

八、作者介绍

储哲、王璐、润宇、马宁、建林、张琨、刘强，均来自美团到店事业群 / 平台技术部。

九、招聘信息

美团到店平台技术部的到餐业务数据策略组菜品知识图谱方向主要负责将菜品知识应用到到餐相关业务，使命是为到餐业务提供高效、优质、智能的应用算法解决方案。基于海量的到餐业务数据，应用前沿的实体抽取、关系挖掘、实体表征学习、细粒度情感分析、小样本学习、半监督学习等算法技术，为到餐业务提供算法能力支持。

业务数据策略组菜品知识图谱方向长期招聘自然语言处理算法专家 / 机器学习算法专家，感兴趣的同学可以将简历发送至 hejianlin@meituan.com。

预训练技术在美团到店搜索广告中的应用

作者：邵雯　杨扬　春喜　任磊　晓俊　金刚

引言

美团到店搜索广告负责美团、大众点评双平台站内搜索流量的商业变现，服务于到店餐饮、休娱亲子、丽人医美、酒店旅游等众多本地生活服务商家。在美团搜索场景中广告的展示样式非常原生，用户使用美团服务不会明显区分广告和自然结果，而广告用户体验损失则会显著影响搜索流量。因此，搜索广告除了优化流量变现效率等商业指标外，也需要重点优化用户体验，不断降低不相关广告对用户体验的损害，这样才能保证整个平台生态长期健康地发展。

在优化用户体验的目标下，如何正确的衡量用户体验，定义不相关广告是首要解决的问题。在搜索广告中，受结果列表页广告位置偏差、素材创意等因素影响，我们无法单一使用点击率（CTR）等客观性指标来衡量用户体验，尤其首位、首屏等排序靠前广告的相关性问题被认为是影响用户体验的主要因素。因此，我们首先建立了美团场景下的搜索广告相关性标准和评估体系，主要通过例行采样和人工评估的方式对搜索广告结果进行相关、一般和不相关的分档标注，进而驱动我们的广告相关性模型和策略迭代。然后，使用广告排序前五位的 Badcase 率（即 Badcase@5）作为搜索广告的相关性评估指标。

问题与挑战

在定义出广告相关性问题和评估指标后，使用相关性模型进行搜索关键词（Query）和候选广告（Doc）的相关性打分，类似于 NLP 文本匹配任务，但实际建模中也发现若干困难。美团搜索结果以商户门店（POI）粒度展示，即 Doc 除了 POI 文本外，还包含一系列的团单或商品描述，内容非常丰富但也带来较多冗余或歧义，且不同业务的文本表达差异较大，比如结婚类商品“朱颜【出门出阁红色秀禾】，南国【中式婚

礼嫁衣红色秀禾]”，给广告内容编码带来挑战。

其次，美团广告商户大多没有选择竞价关键词，且 POI 和团单标题并未面向搜索优化，Doc 内容与 Query 可能存在文本表达偏差。比如“满月酒”和“嗨派星球宝宝宴 · 游轮派对，生日宴个性气球拱门”，需要处理这类信息缺失的问题。我们最初采用 ESIM[1] 交互式模型，但实践中发现该模型对我们复杂广告内容的表征能力有限，正负样本区分能力不足，在过滤不相关广告的同时对相关广告的误伤率较高。自 2018 年底以来，以 BERT[2] 为代表的预训练模型在多项 NLP 任务上都取得了突破，我们也开始探索预训练技术在搜索广告相关性上的应用。

业界及美团的解决方案

针对搜索语义匹配任务，Google[3] 和 Bing[4] 的搜索团队已经基于 BERT 来编码 Query 和候选 Doc，进而改善相关性的效果。预训练模型在美团内部的 NLP 场景中也有不少落地实践，美团搜索已经验证了预训练模型在文本相关性任务上的有效性 [5]。

而针对预训练在语义匹配任务中的应用，业界也提出不少的解决方案。中科院计算所郭嘉丰等人提出 PROP[6] 和 B-PROP[7] 等针对搜索任务的预训练方法，主要思想是引入文档中代表词预测 ROP（Representative wOrds Prediction）任务。纽约大学石溪分校曹庆庆等人提出 DeFormer[8] 分解预训练语言模型来做问答等语义匹配任务，在 BERT 的低层分别对问题和文档各自编码，再在高层部分拼接问题和文档的表征进行交互编码，让文档和问题在编码阶段尽可能地独立，从而提升模型的整体效率。百度刘璟等人提出 RocketQA[9] 和 RocketQAv2[10] 等面向端到端问答的检索模型训练方法，通过跨批次负采样、去噪的强负例采样以及数据增强技术大幅提升了双塔模型的效果。陈丹琦等人提出 SimCSE[11]，采用自监督来提升模型的句子表示能力，从而提升语义匹配的效果。

另一方面，2020 年至今，预训练从“大炼模型”迈向了“炼大模型”的阶段，通过设计先进的算法，整合尽可能多的数据，汇聚大量算力，集约化地训练超大模型，持续提升模型效果。不论是公开论文结果还是美团内部实践，均已证明：更大规模的预

训练模型能带来更好的下游任务效果。因此，美团广告平台与美团搜索与 NLP 部进行了合作，尝试利用预训练模型来优化搜索关键词和广告结果的相关性，进一步降低首屏广告 Badcase，提升用户体验。

本文分为算法探索、应用实践和总结规划三个部分，对预训练技术在搜索广告相关性的落地方案进行介绍。在算法探索部分介绍了我们在训练样本上的数据增强、预训练（Pre-training）和微调（Fine-tuning）阶段的模型结构优化；在应用实践部分，本文介绍了以知识蒸馏为主的模型压缩方法、相关性服务链路优化方案，以及所取得的业务效果；最后，我们总结了相关性方面的优化方法，并对未来技术探索进行了展望。希望这些经验和思考能够给从事相关研究的同学带来一些帮助或启发。

算法探索

在美团搜索广告场景中，相关性计算可以看做用户搜索意图（Query）和广告商户（POI/Doc）之间的匹配问题，实践中我们采用了能够从多角度衡量匹配程度的集成方法，具体方案为分别基于 Query 和 POI 的结构化信息匹配、文本匹配和语义匹配等方法进行打分并且进行分数融合。其中，结构化信息匹配主要是对 Query 分析结果与 POI 进行类目、属性等信息的匹配；文本匹配方面借鉴了搜索引擎中的传统相关性方法，包括 Query 和 POI 的 term 共现数、Query term 覆盖率、TF-IDF、BM25 打分等；语义匹配包括传统的隐语义匹配（如基于 LDA 或者 Word2Vec 计算相似度）和深度语义匹配方法。在广告相关性服务中，我们采用学习能力更强的深度语义匹配模型。

深度语义匹配通常分为表示型和交互型两类：表示型模型一般基于双塔结构分别得到两段输入文本的向量表示，最后计算两段文本表示的相似度；该方法的优点是 Doc 向量可提前离线计算缓存，且匹配阶段计算速度很快，适合线上应用；缺点是只在模型最后阶段进行交互，对文本之间匹配关系学习不足。而交互型模型在初期即对两段输入文本进行交互，匹配阶段可以采用更复杂的网络结构，以学习文本间细粒度匹配关系；这种方法往往可以达到更高的精度，主要挑战在于线上应用的性能瓶颈。

美团搜索广告相关性服务的基线模型采用 Transformer+ESIM 的交互式模型结构，在满足性能的前提下有效解决了部分相关性问题，但是实际应用中仍然存在一些不足，主要包括：1. 训练数据中存在标签错误、正负样本分布不一致等问题；2.Doc 除了基础门店信息外还关联了大量商品和团单内容，如果直接将这些信息拼接成长文本作为 Doc 输入，由于模型结构限制往往需要对 Doc 文本进行截断，因而导致信息丢失；3. 基线模型对于长文本的表征能力有限，相关性判别能力不足，很难在控制变现效率影响的同时解决更多的 Badcase。

为了解决这些问题，我们基于 BERT 在训练数据、特征构造和模型方面进行若干探索和实践。下文将逐一展开介绍。

数据增强

由于 BERT 模型微调阶段所需数据量相比 ESIM 模型更少，并且对数据覆盖全面度、标签准确度、数据分布合理性等因素更为敏感，在进行模型结构探索前，我们先按照如下思路产出一份可用性较高的数据。搜索广告涉及的业务众多且差异性大，包含的团单和商品种类多元，我们希望 BERT 的微调数据尽可能覆盖各个场景和主要服务。如果全部人工标注人力和时间成本较高，而用户点击转化行为能一定程度反映出广告是否相关，所以训练数据主要基于曝光点击日志构造，对于部分困难样本加以规则及人工校验。我们根据业务特性对训练数据的主要优化包括以下几点。

正样本置信加权

正样本主要通过点击数据得到，我们对 4 个月内的 Query-POI 点击数据进行统计，并且基于曝光频次和 CTR 进行数据清洗以减少噪声。实际采样流程中，假设对于某个 Query 需要取 N 个 POI 构造 N 条正样本，采样过程中令 POI 被采样的概率与其点击数成正比，这样做主要是基于点击越多相关性越高的认知，既可以进一步保证标签置信，又有利于模型学习到 POI 之间不同的相关程度。

在实验中我们也尝试了另外两个正样本采样方法：1) 对某个 Query 随机取 N 个 POI，2) 对某个 Query 取点击最多的 N 个 POI。实践发现方法 1 会采样到较多的弱

相关样本，而方法 2 得到的大多为强相关样本，这两种方式均不利于模型拟合真实场景的数据分布。

负采样分层

我们按照模型学习的困难程度，从低到高设计了三种负样本采样方式：

- **全局随机负样本**：大多为跨业务的负样本（比如烧烤和密室逃脱），模型学习最容易，可以有效识别跨类目的恶劣 Badcase；
- **一级类目内负样本**：Query 和 POI 属于相同一级类目（比如美食、丽人等），但是属于不同细化类目（比如祛痘和医学美容），这部分样本可以为模型学习增加一定难度，提高模型判别能力；
- **三级类目内负样本**：Query 和 POI 属于相同的细化类目，但是 POI 并不提供 Query 相关的服务（比如光子嫩肤和水光针商户），这部分属于困难负样本，可以提升模型对语义相近但服务不相关的 Badcase 的判别能力，更大程度保障用户体验；但是在三级类目下采样可能取到较多相关样本，所以这部分样本还需要经过基于服务核心词的规则过滤以及人工校验。

采样平滑及分布一致性

- **采样平滑**：在正样本构建过程中对 Query 采样频次做了平滑，避免高频 Query 采样过多，导致模型忽略对中长尾 Query 样本的学习。
- **样本分布一致性**：在负样本构建中，对于每种负样本均需要保证各 Query 出现概率与其在正样本中概率相等，避免样本分布不一致性导致模型学习有偏。

文本关键词提取

美团搜索广告场景下，Query 中可能包含地址词、品牌词、服务核心词等多种成分，Query 文本一般较短，90% 以上的 Query 长度小于 10；POI 的主要文本特征包括门店名称和商品信息，而广告主的商品数量普遍较多，直接拼接商品标题会导致 POI 文本过长，有 26% 的 POI 文本长度超过 240。

由于相关性模型的主要目标是学习 Query 和 POI 之间的服务相关性，大量冗余文本

信息会影响模型性能和学习效果，我们对 Query 和 POI 文本进行如下处理以提取关键文本信息：

- **对于 Query 文本**：基于命名实体识别（NER）和词权重结果过滤掉地址词、分店名等成分，保留服务核心词；
- **对于 POI 文本**：对所有商品标题进行关键词抽取，得到一组能反映商户核心服务的关键词，将其拼接作为 POI 文本。相比直接拼接原始商品文本，长度大幅下降，仅有 5% 的 POI 长度超过 240，并且 POI 文本质量更高，模型学习效果更好。

最终，我们的微调样本包括约 50 万条数据，涵盖餐饮、休娱、亲子和丽人等 20 个主要类目，其中正负样本比例为 1:5，三种负样本比例为 2:2:1。

模型优化

基于多任务学习的多业务模型

由于美团搜索广告涉及餐饮、休娱亲子、丽人医美等大量业务场景，并且不同场景之间差异较大。从过去的实践经验可知，对于某个业务场景下的相关性优化，利用该业务数据训练的子模型相比利用全业务数据训练的通用模型往往效果更佳，但这种方法存在几个问题：1) 多个子模型的维护和迭代成本更高；2) 某些小场景由于训练数据稀疏难以正确学习到文本表示。

受到多业务子模型优缺点的启发，我们尝试了区分业务场景的多任务学习，利用 BERT 作为共享层学习各个业务的通用特征表达，采用对应不同业务的多个分类器处理 BERT 输出的中间结果，实际应用中根据多个小场景的业务相似程度划分成 N 类，亦对应 N 个分类器，每个样本只经过其对应的分类器。多业务模型的主要优势在于，能够利用所有数据进行全场景联合训练，同时一定程度上保留每个场景的特性，从而解决多业务场景下的相关性问题，模型结构如下图 1 所示：

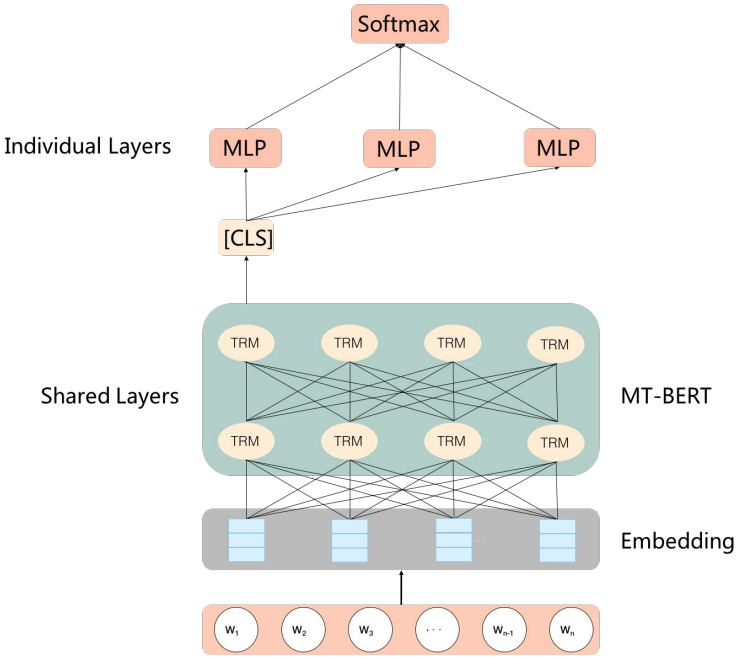


图 1　多业务模型结构

引入品类信息的预训练

由于美团商户 POI 和商品标题可能缺乏有效信息表达，有时仅根据 Query 和 POI 商品文本很难准确判断两者之间的语义相关性。例如【租车公司，< 上水超跑俱乐部；宝马，奥迪 > 】，Query 和 POI 文本的相关性不高，而该商户的三级品类是“养车 – 用车租车 – 租车”，我们认为引入品类信息有助于提高模型效果。

为了更合理的引入品类信息，我们对 BERT 模型的输入编码部分进行改造，除了与原始 BERT 一致的 Query、Doc 两个片段外，还引入了品类文本作为第三个片段，将品类文本作为额外片段的作用是防止品类信息对 Query、Doc 产生交叉干扰，使模型对于 POI 文本和品类文本区别学习。

下图 2 为模型输入示意图，其中红色框内为品类片段的编码情况，Ec 为品类片段的片段编码（Segment Embedding）。由于我们改变了 BERT 输入部分的结构，无法直接基于标准 BERT 进行相关性微调任务。我们对 BERT 重新进行预训练，并对预

训练方式做了改进，将 BERT 预训练中用到的 NSP（Next Sentence Prediction）任务替换为更适合搜索广告场景的点击预测任务，具体为“给定用户的搜索关键词、商户文本和商户品类信息，判断用户是否点击”。预训练数据采用自然及广告搜索曝光点击数据，大约 6 千万样本。

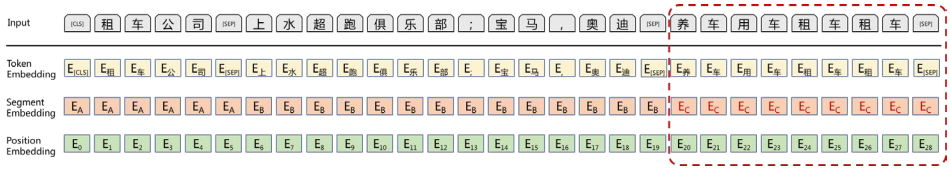


图 2　BERT 输入部分引入 POI 品类信息

模型优化离线效果

为了清晰准确地反映模型迭代的离线效果，我们通过人工标注的方法构建了一份广告相关性任务 Benchmark。基线 ESIM 模型、BERT 模型以及本文提到的优化后 BERT 模型在 Benchmark 上的评估指标如下表 1 所示：

Model	Accuracy	AUC	F1-Score
ESIM（基线，旧训练数据）	67.73%	76.94%	72.62%
MT-BERT-Base	74.88%	82.65%	75.85%
MT-BERT-Base- 多业务	75.41%	83.03%	76.49%
MT-BERT-Base- 引入品类信息	77.33%	83.85%	77.93%
MT-BERT-Large- 引入品类信息	77.87%	85.06%	79.14%

表 1　广告相关性任务模型优化迭代指标

我们首先利用上文介绍的数据增强后的训练样本训练了 MT-BERT-Base 模型（12 层 768 维），与 ESIM 模型相比，各项指标均显著提升，其中 AUC 提升 6.6PP。在 BERT 模型优化方面，多任务学习和引入品类信息这两种方式均能进一步提升模型效果，其中引入品类信息的 MT-BERT-Base 模型效果更佳，相比标准的 MT-BERT-Base 模型 AUC 提升 1.2PP。

在 BERT 模型规模方面，实验发现随着其规模增长，模型效果持续提升，但是预训练和部署成本也相应增长，最终我们选取了大约 3 亿参数量的 MT-BERT-Large 模型（24 层 1024 维），在同样引入品类信息的条件下，相比 MT-BERT-Base 模型 AUC 增长 1.21PP，相比 ESIM 模型 AUC 增长 8.12PP。

应用实践

在模型的实践落地过程中，我们也遇到若干挑战，并且针对性的设计了优化方案。第一个挑战是 BERT 模型的前向耗时无法满足线上性能要求，我们通过知识蒸馏和低精度量化方法对模型进行压缩，并且采用离线缓存与实时预测结合的方式进一步提升了服务性能。

另一个挑战是，在广告业务场景下，需要综合考虑平台变现效率、用户体验、商户供给及转化等因素，如何使相关性分数在广告整体链路中发挥出更好的作用。我们目前采用了低质量广告过滤、重排阶段考虑相关性因子以及 TOP 位次广告门槛控制等策略。下文对应用实践方面的具体方案进行介绍。

模型压缩

由于 BERT 模型的庞大参数量和前向预测耗时，直接部署上线会面临很大的性能挑战，通常需要将训练好的模型压缩为符合一定要求的小模型，业内常用模型压缩方案包括模型裁剪、低精度量化和知识蒸馏等。知识蒸馏 [12] 旨在有效地从大模型（教师模型）中迁移知识到小模型（学生模型）中，在业内得到了广泛的研究和应用，如 HuggingFace 提出的 DistillBERT[13] 和华为提出的 TinyBERT[14] 等蒸馏方法，均在保证效果的前提下大幅提升了模型性能。

经过在搜索等业务上的探索和迭代，美团 NLP 团队沉淀了一套基于两阶段知识蒸馏的模型压缩方案，包括通用型知识蒸馏和任务型知识蒸馏，具体过程如下图 3 所示。在通用型知识蒸馏阶段，使用规模更大的预训练 BERT 模型作为教师模型，对学生模型在无监督预训练语料上进行通用知识蒸馏，得到通用轻量模型，该模型可用于初

始化任务型知识蒸馏里的学生模型或直接对下游任务进行微调。在任务型知识蒸馏阶段，使用在有监督业务语料上微调的 BERT 模型作为教师模型，对学生模型在业务语料上进行领域知识蒸馏，得到最终的任务轻量模型，用于下游任务。实验证明，这两个阶段对于模型最终效果的提升都至关重要。

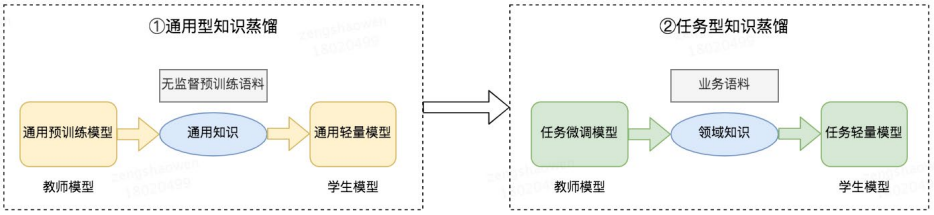


图 3　两阶段知识蒸馏

在美团搜索广告场景下，首先我们基于 MT-BERT-Large（24 层 1024 维）在大规模无监督广告语料上进行第一阶段通用型知识蒸馏，得到 MT-BERT-Medium（6 层 384 维）通用轻量模型，在下游的广告相关性任务上进行微调。MT-BERT-Medium 属于单塔交互结构，如图 4(a) 所示。

目前，每个 Query 请求会召回上百个 POI 候选，交互模型需要分别对上百个 Query-POI 对进行实时推理，复杂度较高，很难满足上线条件。常见解决方案是将交互模型改造成如图 4(b) 所示的双塔结构，即分别对 Query 和 POI 编码后计算相似度。由于大量候选 POI 编码可以离线完成，线上只需对 Query 短文本实时编码，使用双塔结构后模型效率大幅提升。我们使用通用型蒸馏得到的 MT-BERT-Medium 模型对双塔模型中 Query 和 POI 的编码网络进行初始化，并且在双塔在微调阶段始终共享参数，因此本文将双塔模型记为 Siamese-MT-BERT-Medium（每个塔为 6 层 384 维）。双塔结构虽然带来效率的提升，但由于 Query 和 POI 的编码完全独立，缺少上下文交互，模型效果会有很大损失，如表 2 所示，Siamese-MT-BERT-Medium 双塔模型相比 MT-BERT-Medium 交互模型在相关性 Benchmark 上各项指标都明显下降。

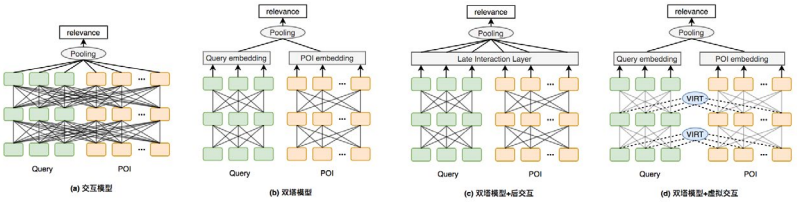


图 4　相关性模型结构对比

为了充分结合交互结构效果好和双塔结构效率高的优势，Facebook Poly-encoder[15]、斯坦福大学 ColBERT[16] 等工作在双塔结构的基础上引入不同复杂程度的后交互层（Late Interaction Layer）以提升模型效果，如图 4© 所示。后交互网络能提升双塔模型效果，但也引入了更多的计算量，在高 QPS 场景仍然很难满足上线要求。针对上述问题，在第二阶段任务型知识蒸馏过程中，我们提出了虚拟交互机制（Virtual InteRacTion mechanism, VIRT），如图 4(d) 所示，通过在双塔结构中引入虚拟交互信息，将交互模型中的知识迁移到双塔模型中，从而在保持双塔模型性能的同时提升模型效果。

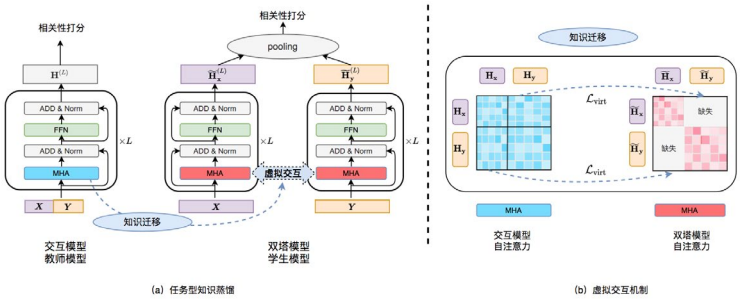


图 5　任务型知识蒸馏 & 虚拟交互

任务型知识蒸馏及虚拟交互的具体过程如上图 5 所示。在任务型知识蒸馏阶段，我们首先基于 MT-BERT-Large 交互模型在业务语料上进行微调得到教师模型。由于学生模型 Siamese-MT-BERT-Medium 缺乏上下文交互，如图 5(b) 所示，注意力矩阵中的灰色部分代表了 2 块缺失的交互信息，我们通过虚拟交互机制对缺失部分进行模拟，计算公式如下为：

$$\widetilde{\mathbf{M}}_{\mathbf{x}\to\mathbf{y}} = \mathrm{softmax}\Big(\mathrm{Attention}(\widetilde{\mathbf{H}}_{\mathbf{x}}\widetilde{\mathbf{W}}_q, \widetilde{\mathbf{H}}_{\mathbf{y}}\widetilde{\mathbf{V}}_k)\Big)$$

$$\widetilde{\mathbf{M}}_{\mathbf{y}\to\mathbf{x}} = \mathrm{softmax}\Big(\mathrm{Attention}(\widetilde{\mathbf{H}}_{\mathbf{y}}\widetilde{\mathbf{V}}_q, \widetilde{\mathbf{H}}_{\mathbf{x}}\widetilde{\mathbf{W}}_k)\Big)$$

其中，$\widetilde{\mathbf{H}}_{\mathbf{x}}$和$\widetilde{\mathbf{H}}_{\mathbf{y}}$分别代表双塔模型中Query和POI表示，$\widetilde{\mathbf{W}}$和$\widetilde{\mathbf{V}}$分别是Query和POI进行编码时的模型参数，$\widetilde{\mathbf{M}}_{\mathbf{x}\to\mathbf{y}}$代表了$\widetilde{\mathbf{H}}_{\mathbf{x}}$到$\widetilde{\mathbf{H}}_{\mathbf{y}}$的注意力（即图5(b)右上角缺失部分），$\widetilde{\mathbf{M}}_{\mathbf{y}\to\mathbf{x}}$代表了$\widetilde{\mathbf{H}}_{\mathbf{y}}$到$\widetilde{\mathbf{H}}_{\mathbf{x}}$的注意力（即图5(b)左下角缺失部分）。而交互模型包含了Query和POI的全交互，计算公式为：

$$\begin{aligned}\mathbf{S} &= \mathrm{Attention}(\mathbf{H}\mathbf{W}_q,\ \mathbf{H}\mathbf{W}_k)\\ &= \mathrm{Attention}((\mathbf{H}_{\mathbf{x}}, \mathbf{H}_{\mathbf{y}})\mathbf{W}_q,\ (\mathbf{H}_{\mathbf{x}}, \mathbf{H}_{\mathbf{y}})\mathbf{W}_k)\\ &= \begin{bmatrix}\mathbf{S}_{\mathbf{x}\to\mathbf{x}} & \mathbf{S}_{\mathbf{x}\to\mathbf{y}}\\ \mathbf{S}_{\mathbf{y}\to\mathbf{x}} & \mathbf{S}_{\mathbf{y}\to\mathbf{y}}\end{bmatrix}\end{aligned}$$

其中，$\mathbf{H}$是交互模型中Query和POI的融合表示，可以分解为$\mathbf{H}_{\mathbf{x}}$和$\mathbf{H}_{\mathbf{y}}$，分别代表Query和POI，$\mathbf{W}$是模型参数。交互模型的自注意力矩阵可以分解为4个部分，其中$\mathbf{S}_{\mathbf{x}\to\mathbf{y}}$和$\mathbf{S}_{\mathbf{y}\to\mathbf{x}}$则是Query和POI之间的交互，也即双塔模型的缺失部分。我们对交互模型的交互矩阵和双塔模型的虚拟交互矩阵之间的L2距离进行最小化，从而将交互模型中的核心交互知识迁移到双塔模型中，计算过程为：

$$\mathbf{M}_{\mathbf{x}\to\mathbf{y}} = \mathrm{softmax}(\mathbf{S}_{\mathbf{x}\to\mathbf{y}})\mathbf{M}_{\mathbf{y}\to\mathbf{x}} = \mathrm{softmax}(\mathbf{S}_{\mathbf{y}\to\mathbf{x}})$$

$$\mathcal{L}_{\mathrm{virt}} = \left\|\widetilde{\mathbf{M}}_{\mathbf{x}\to\mathbf{y}} - \mathbf{M}_{\mathbf{x}\to\mathbf{y}}\right\|_2 + \left\|\widetilde{\mathbf{M}}_{\mathbf{y}\to\mathbf{x}} - \mathbf{M}_{\mathbf{y}\to\mathbf{x}}\right\|_2$$

我们对蒸馏阶段各个模型进行了 Benchmark 上的效果评估以及线上 QPS=50 时的性能测试，结果如表 2 所示。通过虚拟交互进行任务型知识蒸馏得到的任务轻量模型 Siamese-MT-BERT-Medium 相较于直接对通用轻量模型进行微调得到的同结构的 Siamese-MT-BERT-Medium（W/O 任务型知识蒸馏）模型，各项效果指标明显提升，其中 Accuracy 提升 1.18PP，AUC 提升 1.66PP，F1-Score 提升 1.54PP。最终我们对任务轻量模型 Siamese-MT-BERT-Medium 进行上线，相较于最初的 MT-BERT-Large 模型，线上推理速度提升 56 倍，完全满足线上服务的性能要求。

Model	模型规模 / 模型结构	Accuracy	AUC	F1-Score	参数量	耗时
MT-BERT-Large	24 层 1024 维 / 交互	77.87%	85.06%	79.14%	340M	227.5ms
通用轻量模型：MT-BERT-Medium	6 层 384 维 / 交互	77.62%	84.79%	78.63%	21M	16.8ms
Siamese-MT-BERT-Medium（w/o 任务型知识蒸馏）	6 层 384 维 / 双塔	74.23%	81.65%	75.37%	21M	4.0ms
任务轻量模型：Siamese-MT-BERT-Medium	6 层 384 维 / 双塔	75.41%	83.31%	76.91%	21M	4.0ms

表 2　模型效果对比

相关性服务链路优化

相关性计算

为了更好地衡量广告召回结果的相关程度，除了基于模型得出的语义相关性之外，我们还计算了文本相关性、类目相关性等分数，并对所有分数进行融合得到最终的相关性分数。其中，文本相关性的计算借鉴了搜索引擎场景的检索相关性方法，例如 Query 和 POI 的字符串包含关系、短语匹配数和匹配率、以及 BM25 分等。

另外，文本匹配同时考虑了原串匹配、核心词匹配及同义词匹配等多维度指标；类目相关性主要基于 Query 意图分类和商户类目信息进行匹配。分数融合模型可以选择 LR 或者 GBDT 等复杂度比较低的模型，并基于高质量标注数据集训练得到。

相关性应用

通过模型结构和分数融合策略的迭代优化可以得到更加准确合理的相关性分数，但是在实际的相关性应用中，还需要紧密结合广告业务场景，综合考虑平台变现效率、用户体验、广告主供给及转化等多方面因素。基于“减少低质量的不相关广告”和“相关广告排序应该尽量靠前”两个基本要求，我们设计了几种相关性分数的具体应用方式：

- **过滤低质量广告**：完全不相关的广告无疑会严重影响用户体验，需要进行过滤。考虑到不同召回策略和不同业务流量在变现效率及 Badcase 严重程度等方面存在差异，过滤阈值被设计成召回策略 × 类目的二维超参矩阵；
- **重排序参考相关性**：在广告系统的竞价排序模块，在考虑点击率、转化率、交易额和出价等因素的同时，也需要考虑相关性分数；
- **TOP 位次相关性门槛**：首位、首屏等排序靠前的广告结果对于用户体验影响更大，因此针对 TOP 位次设置了相关性门槛，进一步改善用户体验。

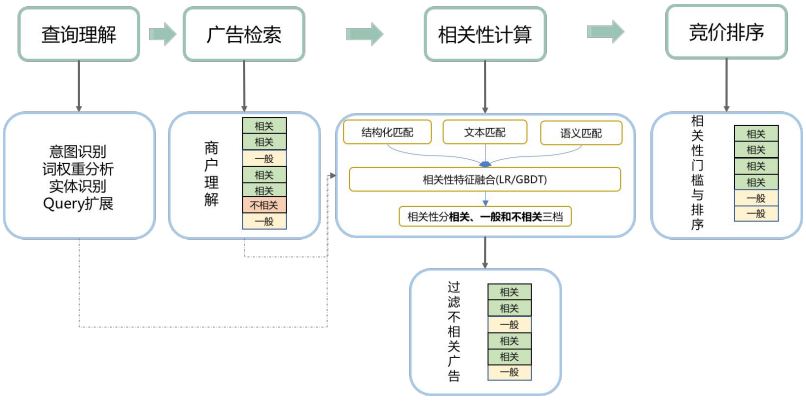


图 6　相关性服务链路示意图

模型部署

为了进一步提升服务性能并且能有效利用计算资源，模型部署阶段我们采用高频流量缓存、长尾流量实时计算的方案。对高频 Query–POI 对进行离线相关性计算并写入缓存，每日对新增或商品信息变化的 Query–POI 对进行增量计算并更新缓存，线上相关性服务优先取缓存数据，如果失效则基于蒸馏后的任务轻量模型进行实时计算。对于输入相关性服务的 Query–POI 对，缓存数据的覆盖率达到 90% 以上，有效缓解了在线计算的性能压力。

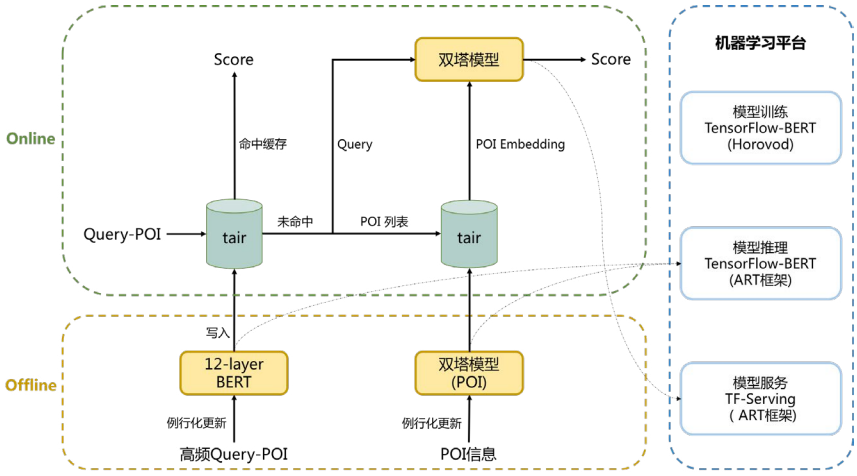


图 7　相关性分数离线 / 在线计算流程图

线上实时计算的任务轻量模型使用 TF-Serving 进行部署，TF-Serving 预测引擎支持使用美团机器学习平台的模型优化工具—ART 框架（基于 Faster-Transformer 改进）进行加速，在将模型转为 FP16 精度后，最终加速比可达到 5.5，数值平均误差仅为 5e-4，在保证精度的同时极大地提高了模型预测的效率。

线上效果

为了更加直接客观地反映线上广告相关性情况，我们建立了美团场景下的搜索广告相关性标准和评估体系，对搜索关键词和广告结果进行相关、一般和不相关的分档标注，采用排序前五位广告的 Badcase 率（即 Badcase@5）作为搜索广告的相关性评估核心指标。

除此之外，由于 CTR 能够通过用户行为间接反映广告的相关程度，并且便于在线上进行 AB 实验评估，而 NDCG 可以反映相关性分数用于广告列表排序的准确性，所以我们选取 CTR 和 NDCG 作为间接指标来辅助验证相关性模型迭代的有效性。我们对本文的优化进行了线上小流量实验，结果显示，实验组 CTR 提升 1.0%，覆盖率降低 1.0%，变现效率基本没有损失。并且经过人工评测，Badcase@5 降低 2.2PP，NDCG 提升 2.0PP，说明优化后的相关性模型能够对召回广告列表进行更加准确的校验，有效提升了广告相关性，从而给用户带来更好的搜索体验。

下面列举了两个 Badcase 解决示例，图 8(a) 和 8(b) 分别包含了搜索“登记照”和“头皮 SPA”时的基线返回结果（左侧截图）和实验组返回结果（右侧截图），截图第一位是广告结果。在这两个示例中，实验组相关性模型将不相关结果“麻朵新生儿摄影”和“莲琪科技美肤抗衰中心”检测出来并过滤掉，让相关广告得以首位展示曝光。

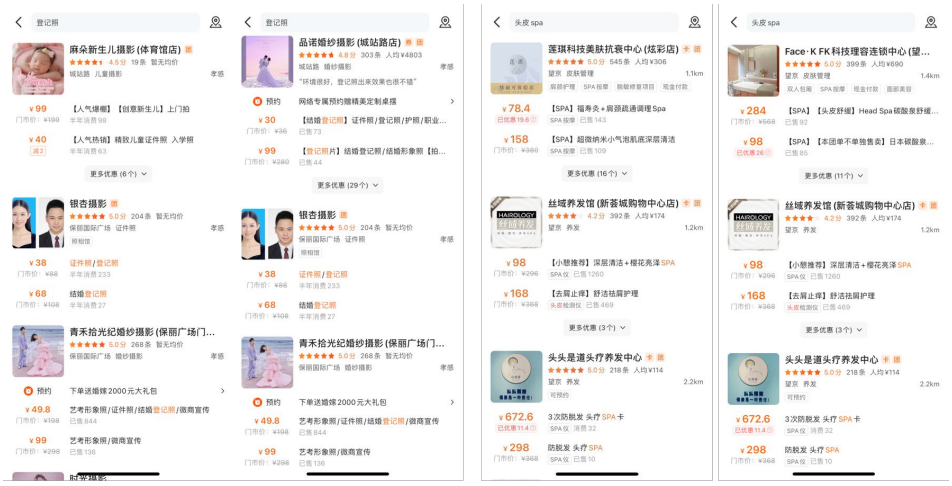


图 8　Badcase 解决示例

总结与展望

本文介绍了预训练技术在美团到店搜索广告相关性上的应用，主要包括样本数据增强、模型结构优化、模型轻量化及线上部署等优化方案。在数据增强方面，为了基于曝光点击数据构造出适合美团广告场景下相关性任务的训练数据，我们构造了多种类型负样本，在采样时考虑正样本置信度、关键词频率平滑、正负样本均衡等因素，另外也对 POI 和团单商品文本进行关键词抽取得到更加简短有效的文本特征。

在模型结构优化方面，我们尝试了对不同业务场景做多任务学习，以及在 BERT 输入中引入品类文本片段这两种方案使模型更好地拟合美团搜索广告业务数据，并利用规模更大的预训练模型进一步提升了模型的表达能力。

在实践应用中，为了同时满足模型效果和线上性能要求，我们对中高频流量进行离线打分和缓存，并且利用 MT-BERT-Large 蒸馏得到的双塔模型进行线上实时预测以覆盖长尾流量。最终，在保证广告平台收入的前提下，有效降低了搜索广告 Badcase 率，提升了用户在平台的搜索体验。

目前，广告相关性打分主要应用于阈值门槛，目的是端到端的过滤掉不相关广告，从

而快速降低广告 Badcase。在此基础上，我们期望相关性模型继续提升区分相关和一般相关广告的能力，从而在重排序中作为排序因子更好的平衡变现效率和用户体验指标，更准确的度量用户体验损失和变现效率提升的兑换关系。此外，在本地搜索类场景下，局部供给经常比较匮乏，实际召回效果对比全局供给的情况更依赖相关性打分的能力，所以我们依然需要在相关性模型上持续深入迭代，并支撑广告召回模型和策略的进一步优化。

在具体技术方向上，相关性门槛阈值设置、广告长文本表达和业务知识融合等方面依然存在优化和提升空间：

- **阈值搜索**：目前的阈值策略需要对每个类目分别调参，缺乏整体性且难以达到全局优化效果。我们正在实验将阈值搜索看作可变现流量上的最优化问题，在限定消耗损失及其他业务约束的条件下，找到一组门槛阈值使得整体 Badcase 解决最大化，并已经取得初步的效果。
- **特征表达**：目前广告 Doc 特征主要采用团单商品标题的关键词抽取结果，但是 Doc 文本仍然较长并且存在一些冗余信息，有必要对 Doc 信息抽取方法继续探索，比如融合外部知识进行信息抽取，或者通过优化 Transformer 注意力机制使模型在相关性打分时更加关注某些重要词项或者行业相关的关键词。
- **联合优化**：Query 和 POI 文本中的蕴含的类目信息、实体成分等对于判断相关性很有帮助，我们计划将相关性任务与搜索广告场景下其他任务联合优化，比如命名实体识别、Query 类目识别等，期望通过引入辅助任务增强模型的学习能力，更全面准确的学习语义相关性。

参考资料

[1] Chen, Qian, et al. “Enhanced lstm for natural language inference.” arXiv preprint arXiv:1609.06038 (2016).
[2] Devlin, Jacob, et al. “BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding.” arXiv preprint arXiv: 1810.04805 (2018).
[3] Pandu Nayak, “Understanding searches better than ever before.” Google blog (2019).
[4] Wenhao Lu, et al. “TwinBERT: Distilling Knowledge to Twin-Structured BERT Models for Efficient Retrieval.” arXiv preprint arXiv: 2002.06275 (2020).

[5] 李勇 , 佳昊 , 杨扬等 . BERT 在美团搜索核心排序的探索和实践 .
[6] Ma, Xinyu, et al. “PROP: Pre-training with Representative Words Prediction for Ad-hoc Retrieval.” Proceedings of the 14th ACM International Conference on Web Search and Data Mining (2021).
[7] Ma, Xinyu, et al. “B-PROP: Bootstrapped Pre-training with Representative Words Prediction for Ad-hoc Retrieval.” arXiv preprint arXiv: 2104.09791 (2021).
[8] Cao, Qingqing, et al. “DeFormer: Decomposing Pre-trained Transformers for Faster Question Answering.” arXiv preprint arXiv:2005.00697 (2020).
[9] Qu, Yingqi, et al. “RocketQA: An Optimized Training Approach to Dense Passage Retrieval for Open-Domain Question Answering.” arXiv preprint arXiv: 2010.08191 (2021).
[10] Ren, Ruiyang, et al. “RocketQAv2: A Joint Training Method for Dense Passage Retrieval and Passage Re-ranking.” arXiv preprint arXiv: 2110.07367 (2021).
[11] Gao, Tianyu, et al. “SimCSE: Simple Contrastive Learning of Sentence Embeddings.” arXiv preprint arXiv: 2104.08821 (2021).
[12] Hinton, Geoffrey, Oriol Vinyals, and Jeff Dean. “Distilling the knowledge in a neural network.” arXiv preprint arXiv:1503.02531 (2015).
[13] Sanh, Victor, et al. “DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter.” arXiv preprint arXiv:1910.01108 (2019).
[14] Jiao, Xiaoqi, et al. “Tinybert: Distilling bert for natural language understanding.” arXiv preprint arXiv:1909.10351 (2019).
[15] Humeau, Samuel, et al. “Poly-encoders: Transformer architectures and pre-training strategies for fast and accurate multi-sentence scoring.” arXiv preprint arXiv:1905.01969 (2019).
[16] Khattab, Omar, and Matei Zaharia. “Colbert: Efficient and effective passage search via contextualized late interaction over bert.” Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval. (2020).

作者简介

邵雯、春喜、晓俊、程佳、雷军等，来自美团广告平台技术部。
杨扬、任磊、金刚、武威等，来自美团平台 / 搜索与 NLP 部。

美团智能客服核心技术与实践

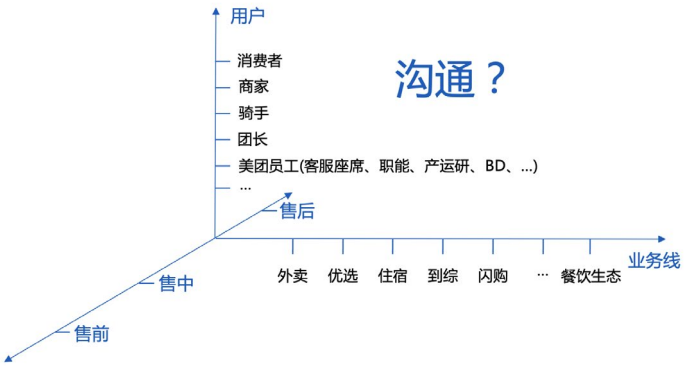
作者：会星

客服是在用户服务体验不完美的情况下，尽可能帮助体验顺畅进行下去的一种解决办法，是问题发生后的一种兜底方案。而智能客服能让大部分简单的问题得以快速自助解决，让复杂问题有机会被人工高效解决。在用户服务的全旅程中，美团平台 / 搜索与 NLP 部提供了问题推荐、问题理解、对话管理、答案供给、话术推荐和会话摘要等六大智能客服核心能力，以期达到低成本、高效率、高质量地与用户进行沟通的目的。本文主要介绍了美团智能客服核心技术以及在美团的实践。

1. 背景

目前，美团的年交易用户量为 6.3 亿，服务了 770 万生活服务类商家。此外，在美团优选业务中还有一个很大的团长群体。美团平台涵盖吃、住、行、游、购、娱等 200 多个生活服务品类，在平台服务的售前、售中、售后各个环节，都有大量信息咨询、订单状态获取以及申诉投诉等沟通诉求。另外，作为一家拥有几万名员工的上市企业，员工之间亦有大量的沟通诉求。面对以上这些需求，如果都是通过人力进行实现，显然不符合公司长远发展的目标，这就需要引入智能客服。

交互立方



1.1 面对不同场景的智能客服落地

首先，我们看看日常生活中几种最为常见的客服场景。

- **售前场景**：比如消费者在平台选择入住酒店，对房型价格、酒店设施、入退房政策等，下单前都有很强的信息咨询诉求。
- **售中场景**：比如外卖催单还没到，添加备注不要辣、加开发票等咨询等等，售前和售中场景主要发生在消费者和商家或平台之间。
- **售后场景**：比如外卖场景投诉菜品少送、骑手送餐超时、要求退款等，酒店场景投诉酒店到店无法入住等，售后往往涉及到客服座席、消费者、骑手和商家，需要多方协同解决。
- **办公场景**：比如 IT、人力资源、财务、法务等咨询，产运研对提供的接口产品的咨询答疑，产品对销售顾问的答疑，以及销售顾问对商家的答疑等等。

1.2 面对不同人群的智能客服落地

沟通是人类的一项基本需求，在绝大多数场景下，我们对沟通的追求都是以低成本、高效率和高质量为目标，而对话机器人也需要同时满足这三点要求。目前我们按照服务的群体进行划分，智能客服落地场景大体可以分为以下四类：

- **面向用户**：提供智能客服机器人，来帮助他们自助解决大部分的问题。
- **面向座席**：用话术推荐或者会话摘要等能力来提升人工座席的工作效率，改善人工座席的工作体验。
- **面向商家**：打造商家助手来降低商家回复的费力度，改善消费者和商家的沟通体验。
- **面向员工**：通过对话机器人，可以自助给员工进行答疑，从而提升办公效率。

1.3 智能客服是什么

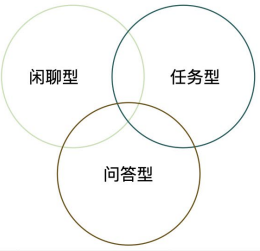
要回答智能客服是什么，可以先看看客服是什么。我们的理解是，客服是在用户服务体验不完美的时候，来帮助体验顺畅进行下去的一种解决办法，是问题发生后的一种兜底方案。而智能客服能让大部分简单的问题得以快速自助解决，让复杂问题有机会被人工高效解决。

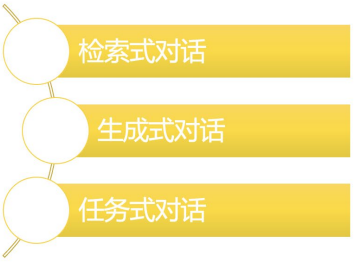
上图展示的是用户服务旅程。首先，用户会通过在线打字或者拨打热线电话的方式进线寻求服务，其中在线咨询流量占比在 85% 以上。当用户进入到服务门户后，先是用户表达需求，然后是智能机器人响应需求，过程中机器人先要理解问题，比如是追加备注或是修改地址，还是申请退款等等，继而机器人尝试自助解决。如果解决不了，再及时地流转到人工进行兜底服务。最后，当用户离开服务时，系统会发送调查问卷，期待用户对本次服务进行评价。

2. 智能客服核心技术

2.1 对话交互技术概述

对话交互技术概述





	推荐		QABot(问答)			TaskBot(任务)		ChatBot(闲聊)	
	问题推荐	商品推荐	FAQ	阅读理解	图谱问答	Taskflow	多轮话题引导	检索式话术	生成式话术
售后客服	O		O			O	O	O	O
商家助手	O	O	O	O	O	O	O	O	O
企业办公	O		O	O		O		O	

智能客服背后的技术主要是以对话交互技术为核心。常见的对话任务可分为闲聊型、任务型和问答型：

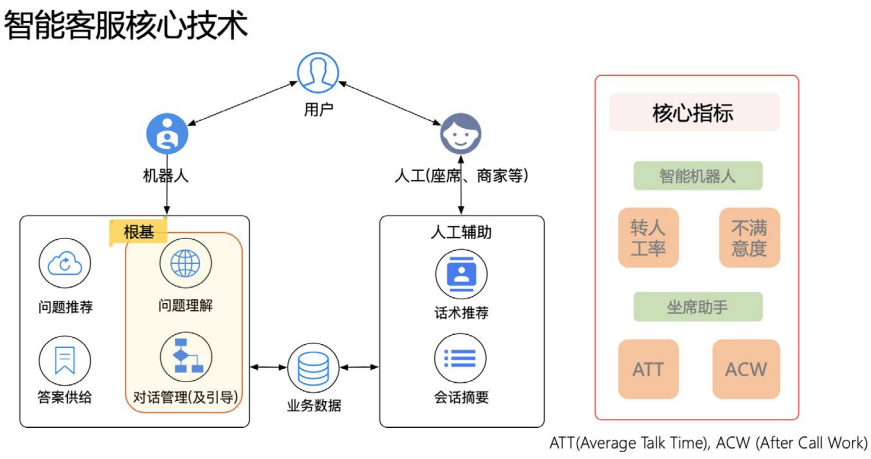
- **闲聊型**：通常是不关注某项特定任务，它的主要的目标是和人进行开放领域的对话，关注点是生成流畅、合理且自然的回复。
- **任务型**：通常是帮助用户完成某项任务指令，如查找酒店、查询订单状态、解决用户的退款申请等等。用户的需求通常比较复杂，需要通过多轮交互来不断收集任务所需的必要信息，进而根据信息进行决策，执行不同的动作，最终完成用户的指令。
- **问答型**：侧重于一问一答，即直接根据用户的问题给出精准答案。问答型和任务型最本质的区别在于，系统是否需要维护一个用户目标状态的表示和是否需要一个决策过程来完成任务。

在技术实现上，通常又可以划分为检索式、生成式和任务式：

- **检索式**：主要思路是从对话语料库中找出与输入语句最匹配的回复，这些回复通常是预先存储的数据。

- **生成式**：主要思路是基于深度学习的 Encoder-Decoder 架构，从大量语料中习得语言能力，根据问题内容及相关实时状态信息直接生成回答话术。
- **任务式**：就是任务型对话，通常要维护一个对话状态，根据不同的对话状态决策下一步动作，是查询数据库还是回复用户等等。

闲聊、问答、任务型对话本质都是在被动地响应用户需求。在具体业务中还会有问题推荐、商品推荐等来主动引导用户交互。在美团的业务场景里主要是任务型和问答型，中间也会穿插一些闲聊，闲聊主要是打招呼或者简单情绪安抚，起到润滑人机对话的作用。

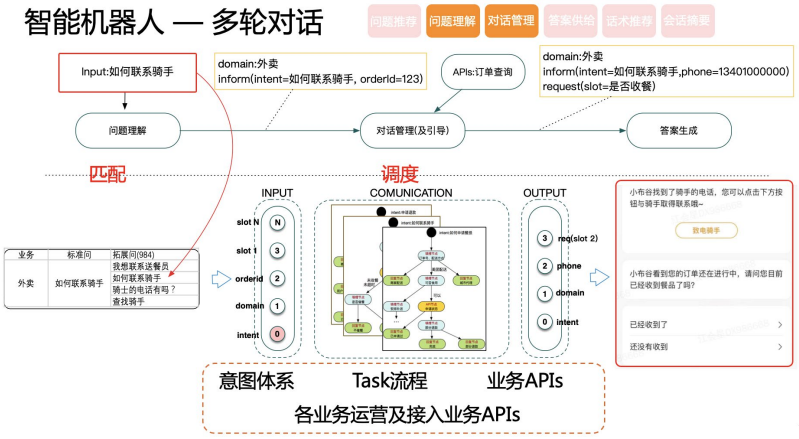


如前面用户服务流程所介绍的那样，用户的沟通对象可能有两个，除了跟机器人沟通外，还可能跟人工沟通。如果是找客服场景人工就是客服座席，如果是找商家场景人工就是商家。机器人的能力主要包括问题推荐、问题理解、对话管理以及答案供给。

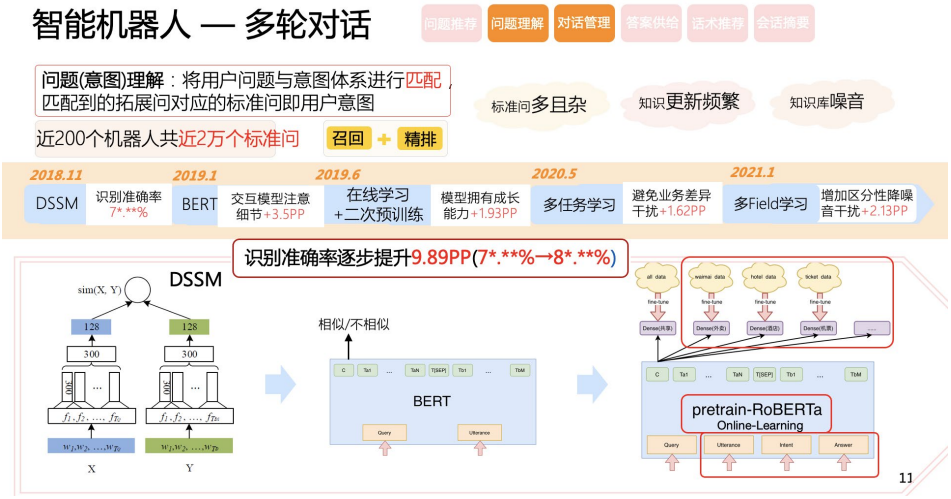
目前，衡量机器人能力好坏的核心输出指标是不满意度和转人工率，分别衡量问题解决的好坏，以及能帮人工处理多少问题。而在人工辅助方面，我们提供了话术推荐和会话摘要等能力，核心指标是 ATT 和 ACW 的降低，ATT 是人工和用户的平均沟通时长，ACW 是人工沟通后的其它处理时长。

2.2 智能机器人——多轮对话

这是一个真实的多轮对话的例子。当用户进入到服务门户后，先选择了一个推荐的问题“如何联系骑手”，机器人给出了联系方式致电骑手。同时为了进一步厘清场景，询问用户是否收到了餐品，当用户选择“还没有收到”的时候，结合预计送达时间和当前时间，发现还未超时，给出的方案是“好的，帮用户催一下”，或者是“我再等等吧”，这时候用户选择了“我再等等吧”。



这个例子背后的机器人是怎么工作的呢？首先当用户输入“如何联系骑手”的时候，问题理解模块将它与知识库中的拓展问进行匹配，进而得到对应的标准问即意图“如何联系骑手”。然后对话管理模块根据意图“如何联系骑手”触发相应的任务流程，先查询订单接口，获取骑手电话号码，进而输出对话状态给到答案生成模块，根据模板生成最终结果，如右边的红框内容所示。在这个过程中涉及到要先有意图体系、定义好 Task 流程，以及订单的查询接口，这些都是业务强相关的，主要由各业务的运营团队来维护。那么，对话系统要做的是什么呢？一是将用户的输入与意图体系中的标准问进行匹配，二是完成多轮交互里面的调度。

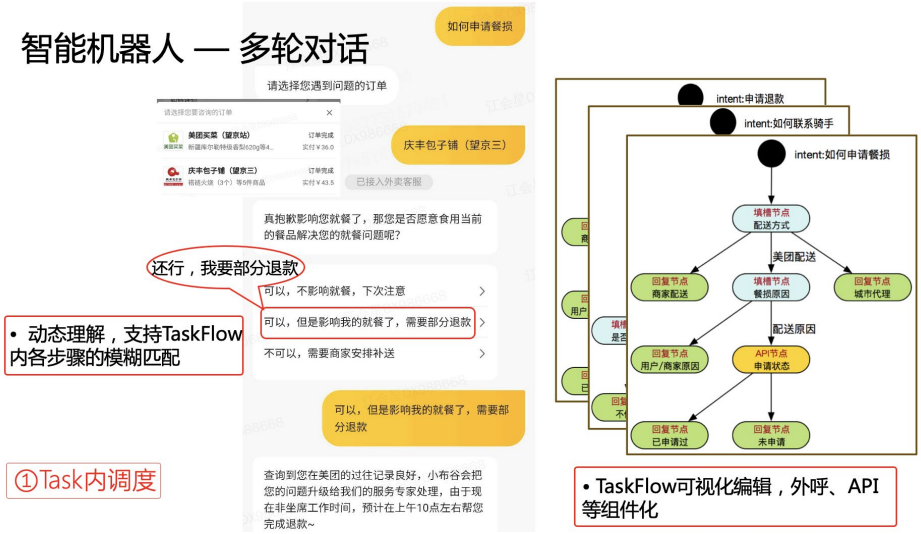


问题理解是将用户问题与意图体系进行匹配，匹配到的拓展问所对应的标准问即用户意图。机器人的工作过程实际是要做召回和精排两件事情。召回更多地是用现有检索引擎实现，技术上更多地关注精排。

美团自研的智能客服系统是从 2018 年开始搭建的，在建设的过程中，我们不断地将业界最先进的技术引入到我们的系统中来，同时根据美团业务的特点，以及问题理解这个任务的特点，对这些技术进行适配。

比如说，当 2018 年底 BERT（参见《美团 BERT 的探索和实践》一文）出现的时候，我们很快全量使用 BERT 替换原来的 DSSM 模型。后面，又根据美团客服对话

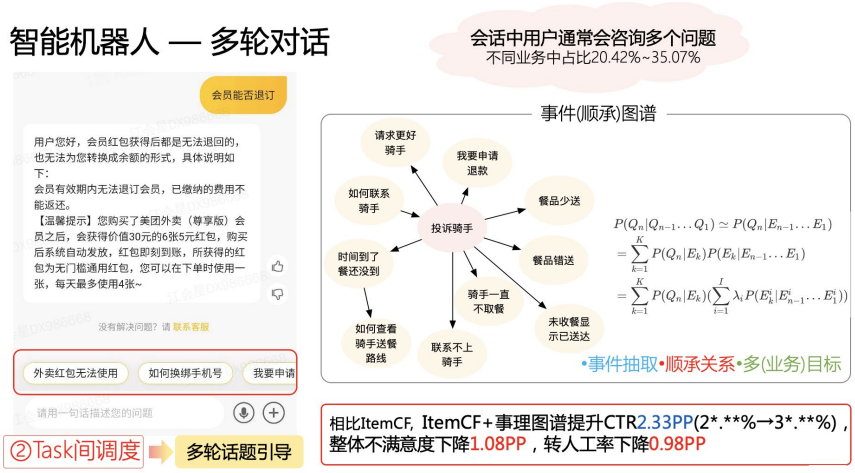
的特点，我们将 BERT 进行了二次训练及在线学习改造，同时为了避免业务之间的干扰，以及通过增加知识区分性降低噪音的干扰，我们还做了多任务学习（各业务在上层为独立任务）以及多域学习（Query 与拓展问匹配，改为与拓展问、标准问和答案的整体匹配），最终我们的模型为 Online Learning based Multi-task Multi-Field RoBERTa。经过这样一系列技术迭代，我们的识别准确率也从最初不到 80% 到现在接近 90% 的水平。



理解了用户意图后，有些问题是可以直接给出答案解决的，而有些问题则需要进一步厘清。比如说“如何申请餐损”这个例子，不是直接告诉申请的方法，而是先厘清是哪一个订单，是否影响食用，进而厘清一些用户的诉求是部分退款还是想安排补送，从而给出不同的解决方案。这样的一个流程是跟业务强相关的，需要由业务的运营团队来进行定义。如右边任务流程树所示，我们首先提供了可视化的 TaskFlow 编辑工具，并且把外呼、地图以及 API 等都组件化，然后业务运营人员可以通过拖拽的方式来完成 Task 流程设计。

对话引擎在与用户的真实交互中，要完成 Task 内各步骤的匹配调度。比如这个例子里用户如果不是点选”可以但影响就餐了…”这条，而是自己输入说“还行，我要部分退款”，怎么办？这个意图也没有提前定义，这就需要对话引擎支持 Task 内各步

骤的模糊匹配。我们基于 Bayes Network 搭建的 TaskFlow Engine 恰好能支持规则和概率的结合，这里的模糊匹配算法复用了问题理解模型的语义匹配能力。



这是另外一个例子，在用户问完“会员能否退订”后，机器人回复的是“无法退回”，虽然回答了这个问题，但这个时候用户很容易不满意，转而去寻找人工服务。如果这个时候我们除了给出答案外，还去厘清问题背后的真实原因，引导询问用户是“外卖红包无法使用”或者是“因换绑手机导致的问题”，基于顺承关系建模，用户大概率是这些情况，用户很有可能会选择，从而会话可以进一步进行，并给出更加精细的解决方案，也减少了用户直接转人工服务的行为。

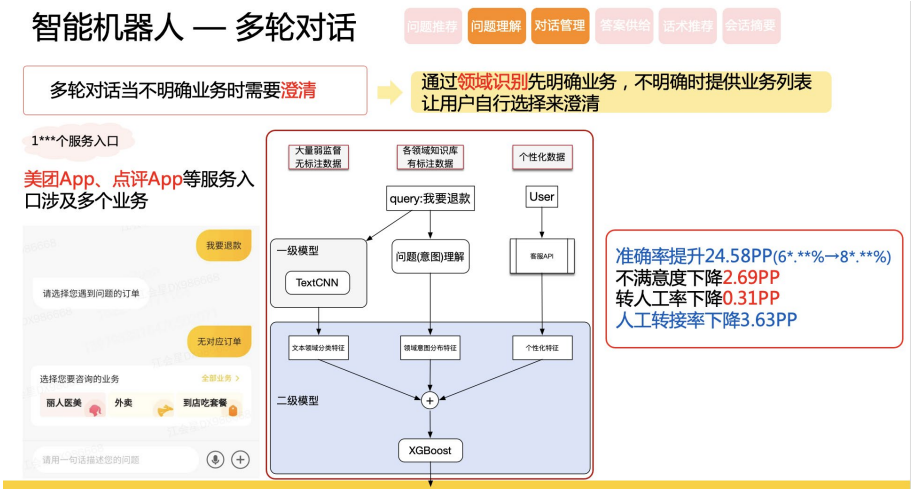
这个引导任务称为多轮话题引导，具体做法是对会话日志中的事件共现关系以及顺承关系进行建模。如右边图所示，这里原本是要建模句子级之间的引导，考虑到句子稀疏性，我们是将其抽象到事件之间的引导，共现关系我们用的是经典的协同过滤方式建模。另外，考虑到事件之间的方向性，我们对事件之间的顺承关系进行建模，公式如下：

$$P(Q_n|E_{n-1}\dots E_1) = \sum_{k=1}^{K} P(Q_n|E_k)P(E_k|E_{n-1}\dots E_1)$$

并通过多目标学习，同时考虑点击指标和任务指标，如在非转人工客服数据和非不满意数据上分别建模顺承关系，公式如下：

$$P(E_k|E_{n-1}\dots E_1) = \sum_{i=1}^{I} \lambda_i P(E_k^i|E_{n-1}^i\dots E_1^i)$$

最终，我们在点击率、不满意度、转人工率层面，都取得了非常正向的收益。



美团平台涵盖吃、住、行、游、购、娱等 200 多个生活服务品类，当用户是从美团 App 或点评 App 等综合服务门户入口进入服务时，需要先行确定用户要咨询的是哪个业务，这里的一个任务是“判断用户 Query 是属于哪个业务”，该任务我们叫做领域识别。若能明确判断领域时，则直接用该领域知识来解答；当不能明确判断时，则还需要多轮对话交互与用户进行澄清。比如用户输入“我要退款”，在多个业务里都存在退款意图，这个时候就需要我们先判断是哪个业务的退款意图，如果判断置信度不高，则给出业务列表让用户自行选择来进行澄清。

领域识别模型主要是对三类数据建模：各领域知识库的有标数据、各领域大量弱监督无标数据和个性化数据。

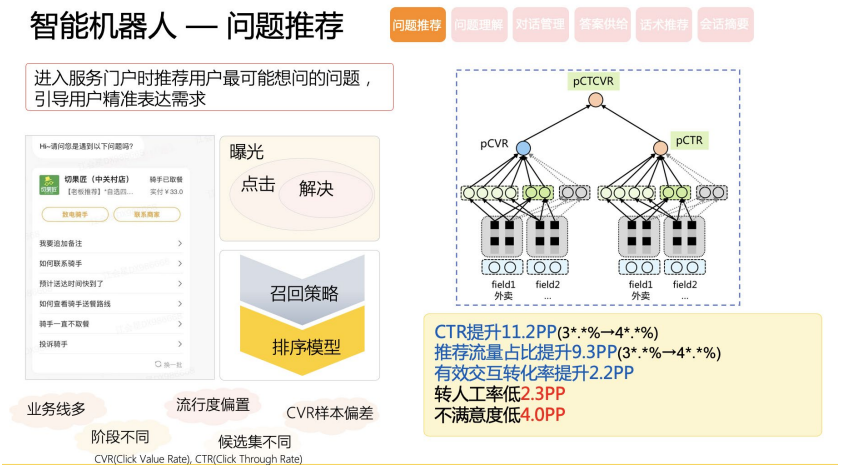
- 依据从各领域知识库的有标数据中学习得到的问题理解模型信号，可以判断用户输入属于各业务各意图的可能性。
- 我们注意到除了美团 App、点评 App 等综合服务入口涉及多个业务外，还有大量能够明确业务的入口，比如说订单入口，从商品详情页进来的入口，这些入口进来的对话数据是有明确业务标签信息的。因此，我们可以得到大量的弱

监督的各业务领域的数据，基于这些数据我们可以训练一个一级分类模型。

- 同时，有些问题是需要结合用户订单状态等个性化数据才能进一步明确的。比如“我要退款”，多个业务里都会有。因此，又要结合用户状态特征一起来训练一个二级模型，最终来判断用户的输入属于哪个业务。

最终，该二级领域识别模型在满意度、转人工率以及成功转接率指标上都取得了非常不错的收益。

2.3 智能机器人——问题推荐

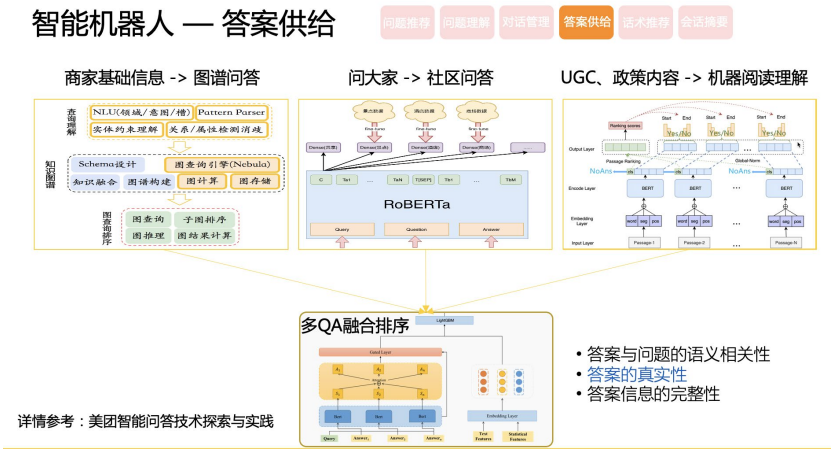


在介绍完多轮对话基础模块问题理解和对话管理后，接下来我们介绍一下智能机器人的另外两个模块：问题推荐和答案供给。如前面多轮对话的例子所示，当用户进入服务门户后，机器人首先是要如何引导用户精准地表达需求，这样即可降低用户迷失或者直接转人工服务，也降低了若机器人不能正确理解时带来的多轮澄清等无效交互。

该问题是一个标准的曝光点击问题，它的本质是推荐问题。我们采用了 CTR 预估任务经典的 FM 模型来作为基础模型，同时结合业务目标，期望用户点击的问题的解决方案能够解决用户问题，该问题最终定义为“曝光、点击、解决”问题，最终的模型是结合多目标学习的 ESSM-FM，对有效交互的转化率、转人工率和不满意度等指标上都带来了提升。

2.4 智能机器人——答案供给

售后客服场景通常问题较集中，且问题的解决多依赖业务内部系统数据及规则，通常是业务部门维护知识库，包括意图体系、Task 流程和答案等。但在售前场景，知识多来自于商户或商品本身、用户体验及评价信息等，具有用户问题开放、知识密度高、人工难以整理答案等特点。比如去哪个城市哪个景点游玩，附近有哪些酒店，酒店是否有浴缸，酒店地址在哪里等，都需要咨询”决策”，针对这些诉求，我们通过智能问答来解决咨询以及答案供给问题。



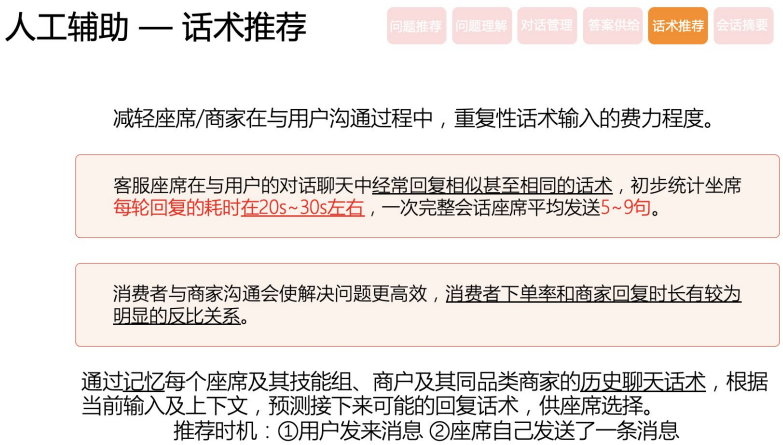
智能问答就是从美团数据中习得答案供给，来快速回答用户的问题，基于不同的数据源，我们建设了不同的问答技术。

- 针对商家基础信息，比如问营业时间、地址、价格等，我们通过图谱问答（KBQA）来解决。利用商家基础信息构建图谱，通过问题理解模型来理解问题，进而查询图谱获取准确的答案。
- 针对社区数据，即商户详情页中“问大家”模块的用户问用户答的社区数据，构建社区问答（Community QA）能力，通过对用户问题与问大家中的”问答对”的相似度建模，选择相似度最高的作为答案，来回答用户的一些开放性问题。
- 针对 UGC 评论数据以及商户政策等无结构化数据，构建文档问答（Document QA）能力，针对用户问题利用机器阅读理解技术从文档中抽取答案，类似我们小时候语文考试中的阅读理解题，进一步回答用户的一些开放性问题。

最后，针对多个问答模块给出的答案，进行多答案来源的答案融合排序，来挑选最终的答案，此外这里还考察了答案真实性，即对“相信多数认为正确的则正确”建模。这部分的详细介绍大家可以参考《美团智能问答技术探索与实践》一文。

3. 人工辅助核心技术

3.1 人工辅助——话术推荐

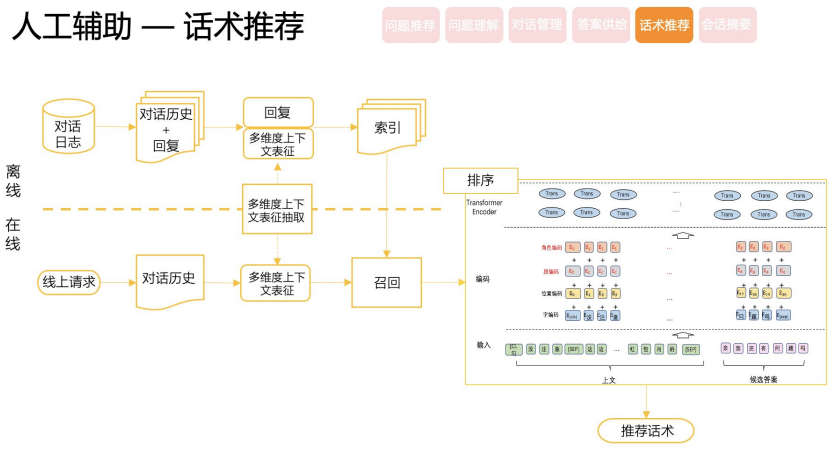


前文介绍的都是智能机器人技术，用户除了跟机器人沟通外，还可能是跟人工沟通。我们在客服座席职场调研过程中发现，座席在与用户的对话聊天中经常回复相似甚至

相同的话术，他们一致期望提供话术推荐的能力来提高效率。此外，除了请求客服座席帮助外，很多情况下用户与商家直接沟通会使得解决问题更高效，而沟通效率不仅影响到消费者的体验，也影响到了商家的经营。比如在外卖业务中，消费者的下单率和商家的回复时长有较为明显的反比关系，无论是客服座席还是商家，都有很强的话术推荐诉求。

那么，话术推荐具体要怎么做呢？常见的做法是先准备好常用通用话术库，部分座席或商家也会准备个人常见话术库，然后系统根据用户的 Query 及上下文来检索最合适的话术来推荐。我们根据调查发现，这部分知识库维护得很不好，既有业务知识变更频繁导致已维护的知识很快不可用因素，也有座席或商家本身意愿不强的因素等。另外，针对新客服座席或者新商家，可用的经验更少。因此我们采用了自动记忆每个座席及其同技能组的历史聊天话术，商家及其同品类商家的历史聊天话术，根据当前输入及上下文，预测接下来可能的回复话术，无需人工进行整理，大大提升了效率。

我们将历史聊天记录构建成“N+1”QA 问答对的形式建模，前 N 句看作问题 Q，后 1 句作为回复话术 A，整个框架就可以转化成检索式的问答模型。在召回阶段，除了文本信息召回外，我们还加入了上文多轮槽位标签，Topic 标签等召回优化，排序为基于 BERT 的模型，加入角色信息建模，角色为用户、商家或者座席。



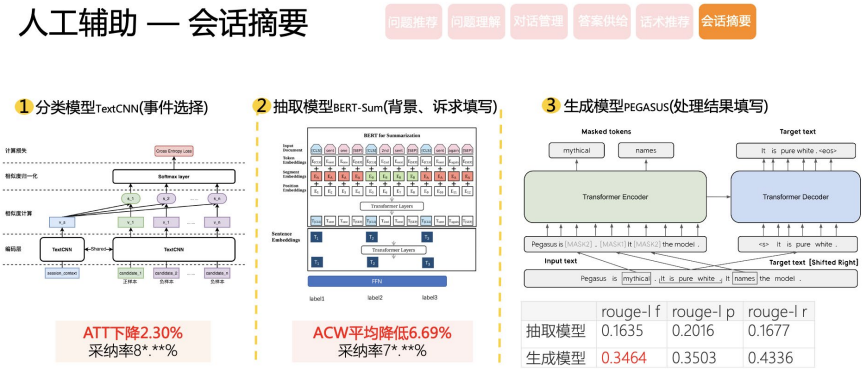
整个架构如上图所示，分为离线和在线两部分。另外上线后我们也加入了一层 CTR

预估模型来提升采纳率。当前多个业务的话术推荐平均采纳率在 24% 左右，覆盖率在 85% 左右。话术推荐特别是对新座席员工价值更大，新员工通常难以组织话术，通过采纳推荐的话术可以来缩减熟练周期，观测发现，3 个月内座席员工的平均采纳率是 3 个月以上座席员工的 3 倍。

3.2 人工辅助——会话摘要

- 提升客服坐席工作效率，改善其办公体验
  - 背景、诉求、处理结果的填写
- 经营分析，产品改善
  - 事件、标签

在客服场景座席跟用户沟通完后，还需要对一些必要信息进行工单纪要，包括是什么事件，事件发生的背景是什么，用户的诉求是什么，最后的处理结果是什么等等。而填写这些内容对座席来说其实是很不友好，通常需进行总结归纳，特别是有些沟通进行的时间还比较长，需要来回翻看对话历史才能正确总结。另外，为了持续对于服务产品进行改善，也需要对会话日志进行相应事件抽取及打上标签，从而方便经营分析。

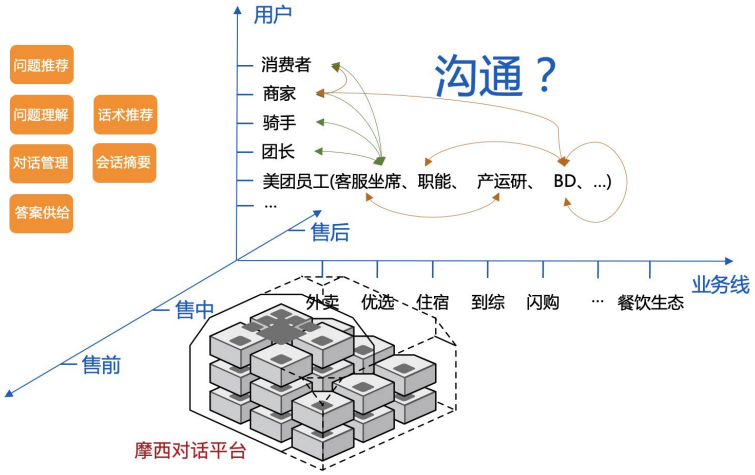


	rouge-l f	rouge-l p	rouge-l r
抽取模型	0.1635	0.2016	0.1677
生成模型	0.3464	0.3503	0.4336

这里有些问题是选择题，有些问题是填空题，比如这通会话具体聊的是哪个事件，我们提前整理有比较完整的事件体系，可以看成是个选择题，可以用分类或者语义相似度计算模型来解决。又比如说事件发生的背景，如外卖退款的背景是因餐撒了、酒店退款的背景是到店没有房间等，是个开放性问题，分析发现可以很好地从对话内容中抽取，可以用摘要抽取模型来解决。而对于处理结果，不仅仅依赖对话内容，还包括是否外呼，外呼了是否商家接通了，后续是否需要回访等等，我们实验发现生成模型更有效。具体使用的模型如上图所示，这里事件选择考虑到经常有新事件的添加，我们转成了双塔的相似度计算任务，背景抽取采用的是 BERT-Sum 模型，处理结果采用的是谷歌的 PEGASUS 模型。

4. 小结与下一步计划

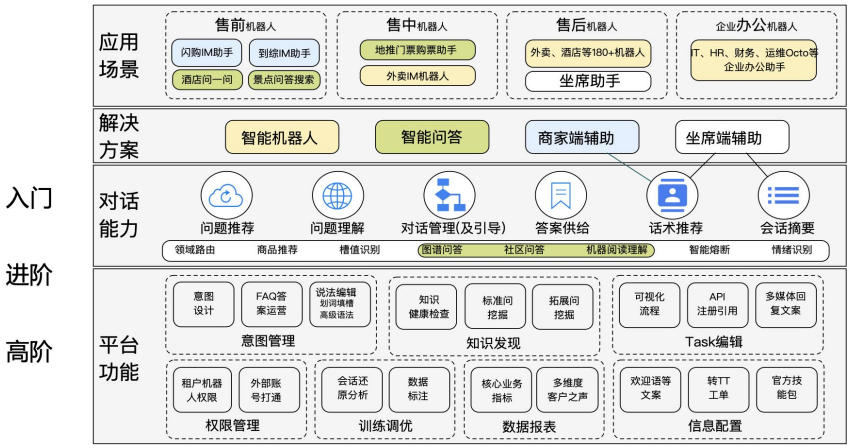
4.1 小结——交互立方



前面介绍了美团智能客服实践中的一些核心技术，过程中也穿插着介绍了客服座席与消费者 / 商家 / 骑手 / 团长等之间的沟通提效，以及消费者与商家之间的沟通提效。除了这两部分之外，在企业办公场景，其实还有员工之间、销售顾问与商家之间的大量沟通。如果一个个去做，成本高且效率低，解决方案是把智能客服中沉淀的能力进行平台化，最好“一揽子”进行解决，以固定成本来支持更多的业务需求。于是我们

搭建了美团的对话平台 – 摩西对话平台，用“一揽子”方案以固定成本来解决各业务的智能客服需求。

4.2 小结——对话平台“摩西”



构建一个怎么样的对话平台，才能提供期望的没有 NLP 能力的团队也能拥有很好的对话机器人呢？首先是把对话能力工具化和流程化。如上图所示，系统可分为四层：应用场景层、解决方案层、对话能力层、平台功能层。

- **应用场景层**：在售前应用场景，一类需求是商家助手，如图中所列的美团闪购 IM 助手和到综 IM 助手，需要辅助商家输入和机器人部分接管高频问题能力；还有一类需求是在没有商家 IM 的场景需要智能问答来填补咨询空缺，比如图中所列的酒店问一问和景点问答搜索；另外售中、售后以及企业办公场景，各自需求也不尽相同。
- **解决方案层**：这就要求我们有几套解决方案，大概可以分为智能机器人、智能问答、商家辅助、座席辅助等。每个解决方案的对话能力要求也有所不同，这些解决方案是需要很方便地对基础对话能力进行组装，对使用方是透明的，可以拿来即用。
- **对话能力层**：前面也进行了相应的介绍，六大核心能力包括问题推荐、问题理解、对话管理、答案供给、话术推荐和会话摘要。

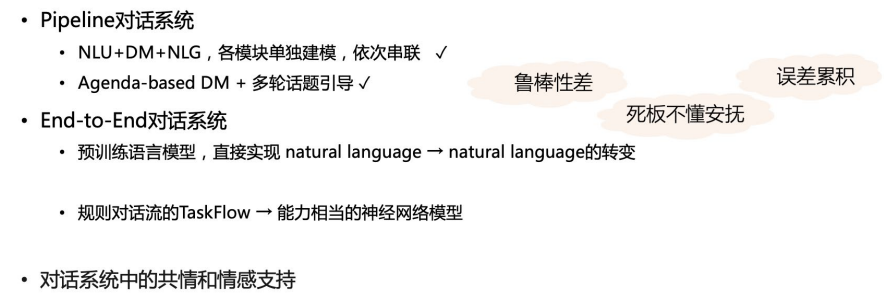
- **平台功能层**：此外，我们需要提供配套的运营能力，提供给业务方的运营人员来日常维护知识库、数据分析等等。

其次，提供“一揽子”的解决方案，还需要针对处在不同阶段的业务提供不同阶段的解决方案。

- 有些业务只希望维护好常用的问答，能回答高频的问题就好，那么他们只需要维护一个入门级的机器人，只需要在意图管理模块来维护它的意图，意图的常见说法以及答案就可以了。
- 而对于有运营资源的团队，他们希望不断地去丰富知识库来提升问答能力，这个时候可以使用知识发现模块，可以自动地从每天的日志里面发现新意图及意图的新说法，运营人员只需要每天花一点时间来确认添加及维护答案即可，这是一个进阶的业务方。
- 还有一些高级的业务方希望调用他们业务中的 API 来完成复杂问题的求解。这个时候他们可以使用 TaskFlow 编辑引擎，在平台上直接注册业务的 API，通过可视化拖拽的方式来完成 Task 编辑。

此外，为了进一步方便更多的业务介入，我们也提供了一些闲聊、通用指令、地区查询等官方技能包，业务方可以直接勾选使用。另外，随着我们不断在业务中沉淀，也会有越来越多的官方行业技能包。整体方向上是逐步让业务方使用的门槛变得越来越低。

4.3 下一步计划



前文所介绍的对话系统是一种 Pipeline 式对话系统，按照功能划分为不同的模块，各个模块单独建模，依次串联。这种方式的好处是可以做到不同团队职责的有效分工，比如研发同学专注于建设好问题推荐模型、问题理解模型和 Task 引擎等；业务运营同学专注于意图体系维护、Task 流程设计以及答案设计等等。它的劣势也很明显，模块耦合，误差累积，很难联合优化，进而各模块负责的同学可能会去修修补补，容易导致动作变形。

另一类建模方式是 End-to-End，将 Pipeline 式对话系统的各个模块联合建模成一个模型，直接实现语言到语言的转变，此类方法最初应用在闲聊式对话系统里面，近期随着大规模预训练模型的快速发展，学术上也逐渐开始研究基于预训练模型的端到端任务型对话系统。它的优点是模型可以充分利用无监督人人会话，用数据驱动可以快速迭代；缺点是模型的可控性差，不易解释且缺乏干预能力。目前主要以学术研究为主，未见成熟的应用案例。

除了使用这种大量无监督的人人会话日志外，还有一种思路是基于 Rule-Based TaskFlow 构建规则的用户模拟器，进行交互以生成大量的对话数据，进而训练对话模型。为了保证对话系统的鲁棒性，也可使用类似对抗攻击的方法优化，可以模拟 Hard User 的行为，不按顺序执行 TaskFlow，随机打断、跳转某个对话节点等等。

此外，通过对比分析人机对话日志和人人对话日志，人机对话比较僵硬死板，无法有效捕捉用户的情绪，而人就很擅长这方面。这在客服场景非常重要，用户往往进来就是带着负面情绪的，机器人需要有共情能力。而端到端数据驱动的对话和对话共情能力建设，也将是接下来一段时间我们尝试的重点方向。

广告深度预估技术在美团到店场景下的突破与畅想

作者：胡可

背景与简介

1. 背景

点击率 / 转化率 / 交易额预估问题，是广告 / 推荐领域的核心，预估值准确性与广告 / 推荐平台的营收和发展直接相关。因此，该方向吸引了学术界与工业界的广泛研究，成为机器学习 / 深度学习技术应用最成功的领域之一。

得益于大规模深度学习框架的推广以及在图像、NLP 等方向的成功应用，深度学习技术被引入广告和推荐领域，尤其在点击率 / 转化率 / 交易额预估问题上，取得了巨大成功，已经成为业界的主流方法。对比传统的机器学习方法 [1][2]，DNN（深度神经网络）[3][4][5] 兼具记忆与泛化性 [6]，在拟合能力上体现出更大优势。然而，经过几年的发展，深度模型的优化变得越来越困难，既往“拿锤子找钉子”——单方面提升模型复杂度的模式已经不再奏效。在这种大背景下，如何打破新的瓶颈呢?

美团到店广告质量预估团队在过去的两年中不断进行实践和研究：通过与美团业务特点深度结合，发挥 DNN 结构设计灵活多变的优势，大量新模型新技术喷涌而出，团队探索出了一条破局之路。

2. 业务特点与挑战

模型整体化方案设计同业务特点紧密相关，团队的业务形态，主要场景覆盖美团 / 大众点评双侧的搜索广告，筛选列表广告，信息流广告三种业务形式（如图 1 所示），意图依次从强到弱。每一个业务也具体包括餐饮、丽人、休闲娱乐、结婚、酒店等不同品类。丰富的业务特点使得算法需要从多个角度进行结合，为团队不断进行新模型的灵活设计提供了广阔的空间。

图 1　美团 / 大众点评双平台各自三种广告形式

进一步理解业务特点，上述的 O2O(Online to Offline) 业务类型，将线下的商业机会与互联网结合，同传统的线上业务的核心差异在两方面，一方面是 Online to Offline 中的“to”，从线上到线下过程中，用户会重点考虑线上线下间的距离因素，因此空间维度的 LBS(Location Based Service) 距离约束是一个重要特点；另一方面是 Online to Offline 中的“Offline”，线下门店消费习惯，行为周期具有明显时间维度特点，例如在月级别固定消费的美甲行为，因此长周期性为另一重要特点。

基于以上 O2O 业务特点，模型整体建模设计有以下空间与时间维度两方面的挑战。

a. LBS 空间距离约束：

- **特点**：线上门店 / 商品需要在有限的距离约束下展示，不同空间位置差异性大。
- **挑战**：带来了候选量差异大，上下文刻画重要等挑战。

b. 长周期性：

- **特点**：门店消费习惯具有周期性强，长期行为影响大等特点。
- **挑战**：用户的长期兴趣建模挑战大，并且因常规模型训练方式较难学习周期性，训练方式存在遗忘历史信息挑战。

3. 深度预估技术

针对 LBS 距离约束与长周期性两方面挑战，下文将简要介绍 CTR(Click-through Rate) 预估技术方案。从机器学习分类的角度理解，CTR 预估是一个有监督机器学习问题：$min_f \ell(f(u,a,c),y)$，最优预测函数$\hat{f}$被用户 (u)、广告 (a)、上下文 (c) 三个维度的函数输入，以及训练过程的函数求解（min_f）所决定，其中上下文包括了请求 Query、请求列表上下文、请求地理位置等实时信息。

如图 2 所示，我们结合 LBS 距离约束和长周期性这两个核心挑战，对 CTR 预估建模函数（用户、广告、上下文和训练过程）进行优化：

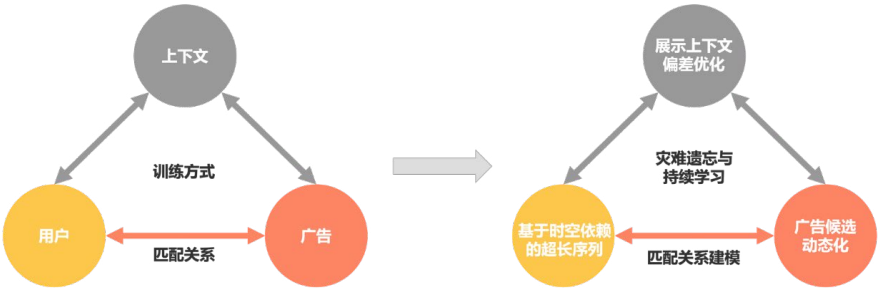


图 2　CTR 预估建模关系图

a. 函数输入：上下文、用户、广告三方面建模

基于位次组合的上下文偏差感知预估（上下文）：由于 LBS 距离约束挑战，空间位置差异引起不同展示上下文影响大，因此利用深度位次交叉网络将广告和位次组合建模，并基于位次信息强化不同请求下的上下文偏差感知。**基于时空依赖的超长序列建模（用户）**：针对长周期性与 LBS 距离约束两方面挑战，一方面基于预训练机制实现了超长序列建模，捕获用户周期性行为；另一方面，实现行为序列的时间与空间联合建模，优化 LBS 距离约束问题。**广告候选动态化（广告）**：由于 LBS 距离约束挑战，空间差异性引起不同位置候选差异大，广告与用户两者表达存在匹配关系（如图 2 所示），只有保障了广告供给的充足性，才能对用户广告关系进行更好的建模。因此进行广告候选量 / 候选类型 / 算力资源三方面动态化，提供用户需求匹配的供给。

b. 函数求解：模型训练方式

灾难遗忘与持续学习（训练方式）：针对长周期性挑战，高维稀疏模型在学习最近数据的同时会遗忘历史数据，形成灾难遗忘。通过具备数据回放能力的记忆结构进行持续学习，实现对周而复始的历史数据有效刻画。**整体网络如图 3 所示，结合上述 User/Ad/Context/Training 几方面优化，实现方式通过表示单元（Representation Unit），记忆单元（Memory Unit），组合单元（Combination Unit）三个模块构成高精度高量级的预估系统**。其中表示单元通过加载流式数据，结合超长序列和实时序列，构建时空依赖激活（图中 Representation Unit 的 User 部分）；记忆单元整合历史与当前数据，在数据流控制的基础上，一方面提供历史超长序列的表达学习（图中 Memory Unit 的 User 部分），实现序列长度为 2000~10000 用户全生命周期兴趣建模。另一方面面对灾难遗忘问题 [10]，通过具备数据回放能力的记忆结构进行持续学习（图中 Training 部分），实现高精度的单点预估。组合单元提供组合能力，在广告方面，可以将记忆单元的综合表示与多个泛商品表示进行组合（图中 Ad 部分），在上下文方面，可以与多个位次表示进行组合，将单门店的预估能力扩展到多位次多商品（图中 Context 部分），进而感知展示上下文偏差，从而实现高精度高量级的组合预估系统（**精排在线预估量级 3000+**）。

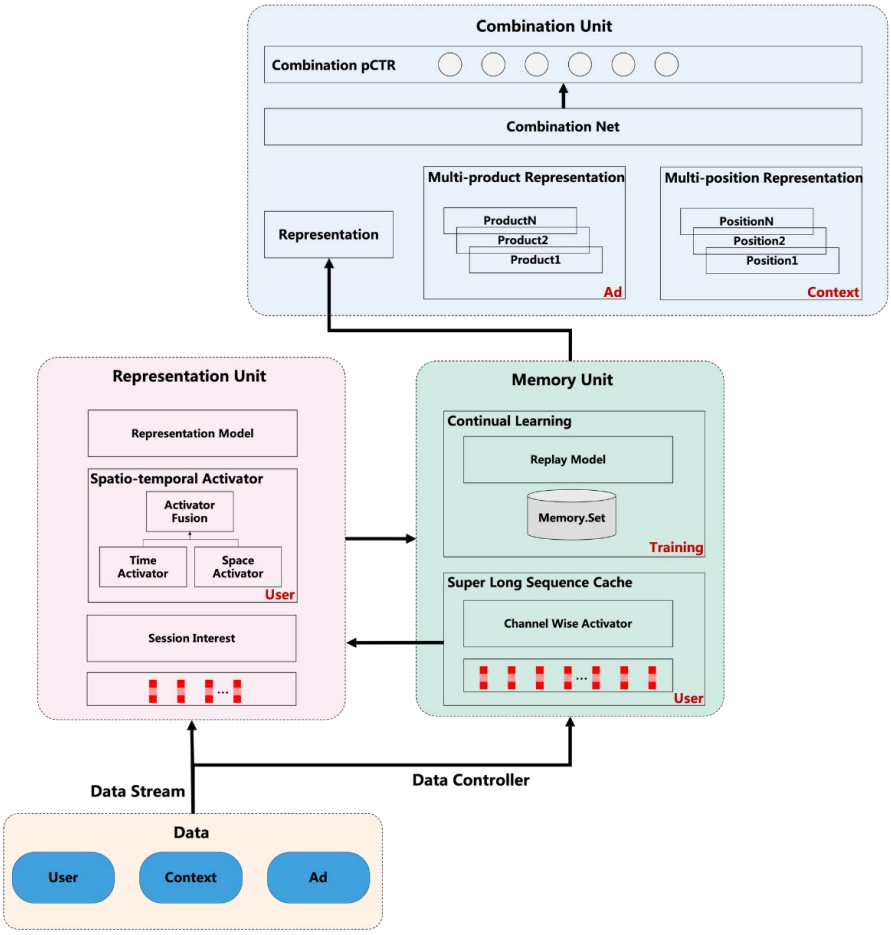


图 3　CTR 预估整体网络图

下文会就 Context、User、Ad、Training 四个方面具体介绍技术突破。

预估技术突破

1. 上下文：基于位次组合的上下文偏差感知预估

由于 CTR 预估模型的训练通常采用曝光点击数据，该数据是一种隐式反馈数据，所以会不可避免地产生各种偏差问题，例如位次偏差、选择性偏差、流行度偏差等。以位次偏差为例（通常也被称为位置偏差，为与本文 LBS 的位置区分开，我们称其为位

次偏差），由于在搜索列表页场景中，用户存在着从上至下的浏览行为习惯，靠前位次的广告通常会受到更多的关注从而容易有更高的 CTR，这种位次倾向会累积到数据中，从而导致数据中存在着偏差问题。此外，受整个列表上下文所产生的点击差异性而导致的偏差也逐渐受到业界关注，在本文该偏差被称之上下文偏差。在 LBS 空间距离约束挑战下，位次偏差和上下文偏差相较于传统电商场景更突出，下文具体介绍该问题。

展示上下文偏差问题：在 O2O 场景的 LBS 距离约束下，不同地理位置相同搜索词所展示的列表差异极大，而传统电商由于不存在 LBS 约束，对于同一搜索词往往更容易根据热度去排序展示，在此比较下，LBS 的展示上下文对于广告有着更大的影响，这就导致了在不同的上下文中数据存在较大不同的偏差。对于位次偏差而言，传统电商的位次偏差更趋向于一个固定偏差，而 LBS 约束下的位次偏差在不同上下文中是有明显差异的，例如一线城市相较于三线城市而言，由于广告的供给更加充足且质量更高，靠前位次广告容易吸引用户点击，更易形成用户点击靠前位次的行为习惯。同时，一线城市商户质量差异小，三线城市商户质量参差不齐，这就导致广告受其上下文自然的影响呈现差异，进一步而言，相同广告在不同自然上下文中点击率也是有差异的，用户容易受到不同上下文的影响从而有不一样的决策，这种差异同样会累积到数据中导致数据存在上下文偏差。为此，本文将位次偏差及广告受不同上下文自然影响所带来的上下文偏差这两类偏差归纳为展示上下文偏差。在美团业务中，搜索广告场景的用户意图较强，其更关注于靠前广告位，位次偏差较大，而在筛选列表广告场景中，意图不明确的用户偏向于去浏览更多的潜在优质商家，因此上下文偏差较大，团队结合这些业务特点在各个场景上推进了基于位次组合的上下文偏差感知预估技术。

位次组合预估技术框架：基于位次偏差问题，我们首先沉淀出一个深度位次交叉网络（Deep Position-wise Interaction Network）（DPIN）模型框架去同时预估多个位次的 CTR。如图 4 所示，DPIN 模型由三个模块组成，分别是处理 J 个（例如 200 个）候选广告的基础模块（Base Module），处理 K 个（例如 10 个）候选广告位次的深度位次交叉模块（Deep Position-wise Interaction Module）以及组合 J X K（例如 2000 个）个交叉候选的位次组合模块（Position-wise Combination Module），不

同模块需预估的样本数量不一样，复杂模块预估的样本数量少，简单模块预估的样本数量多，由此来提高模型性能和保障服务性能。通过这三个模块的组合，DPIN 模型有能力在服务性能的限制下预估每个广告在每个位次上的 CTR，并学习位次信息和其他信息的深度非线性交叉表示 [6]。

上下文偏差感知学习：DPIN 技术框架具备位次信号的深度交叉能力，从而实现多位次的 CTR 预估。在此基础上，由于自然广告分别排序及广告插入自然列表的业务特点，我们在广告预估时可以去考虑已排序好的自然列表，而多位次预估的技术框架赋予了广告可以去感知其局部自然上下文的能力。因此，我们在 DPIN 的基础上考虑每个位次的邻域自然信息，如图 4 所示，深度位次交叉模块在考虑位次、上下文、用户位次历史行为的基础上通过上下文自然聚合（Position-wise Context Aggregation）模块去感知并聚合局部自然上下文信息，从而更好地缓解上下文偏差。进一步地，相同上下文中的自然和广告具有同质性，我们通过基于异质的 Listwise 损失函数加强自然广告间的竞争性建模，通过基于同质的自适应采样加强广告间关系建模。

基于位次组合的上下文偏差感知预估在多个主广告位落地，并都在衡量广告营收的 RPM(Revenue Per Mille) 指标提升 2%~3%。

详情请参考团队的 SIGIR 2021 论文，Deep Position-wise Interaction Network for CTR Prediction[6]。

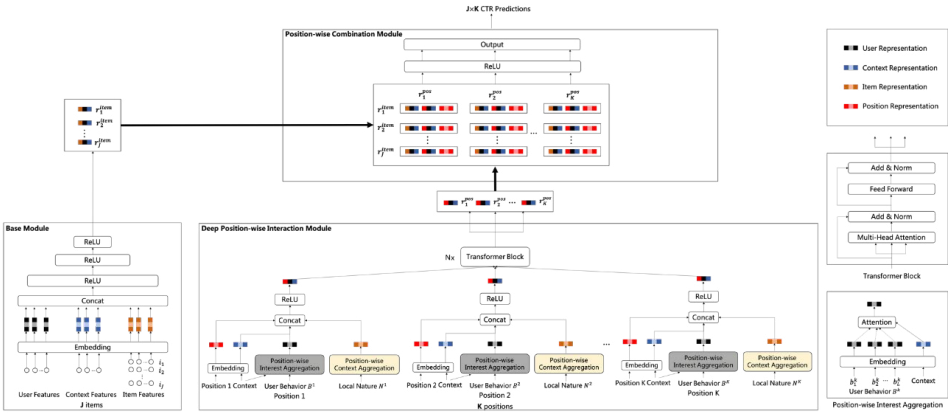


图 4　基于位次组合的上下文偏差感知网络

2. 用户：基于时空依赖的超长序列建模

针对长周期性和 LBS 约束两个挑战，进行用户建模深度优化。一方面针对长周期问题，基于“长短分离”用户行为序列设计原则，通过预训练方式建模长期兴趣；另一方面针对 LBS 约束问题，通过时空建模进行时间与空间的多重依赖关系学习，有效刻画在不同时空条件的用户表达。

用户长期行为复杂性：在美团 O2O 场景下，用户行为习惯存在明显月级周期性（如图 5 所示），并且行为间周期性差异较大，例如脱毛 / 水光针类的周期性大于火锅 / 烧烤类，用户超长序列建模是捕获周期性行为特点的必要条件。在时空依赖方面，用户点击体现出明显的“长分散、短聚合”特点（如图 6 所示），即较短时间内用户的地理位置聚集在单点，但拉长来看，用户将漫游到多个地理位置；超长周期内，大部分用户还会有跨省跨城流动。因此时空联合联合建模存在潜能。

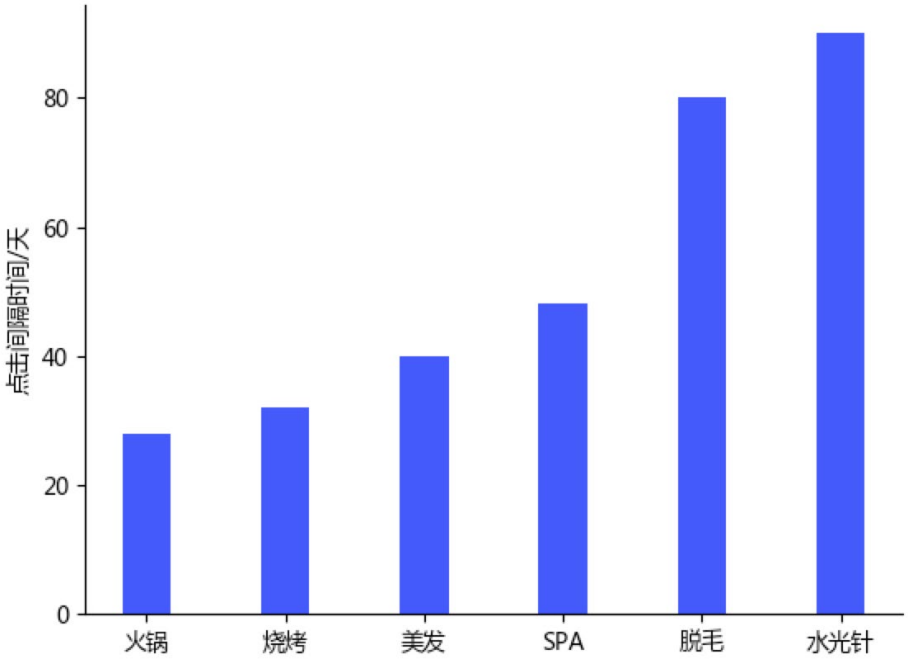


图 5　不同类目行为的平均点击间隔天数

图 6　用户地理位置漫游“长分散，短聚合”特点示例

综上两方面问题，整体建模如下：

$$P(click=1)=f(a,u,c,h(s))\ ,$$

$$\text{where } s=\{(a_i,t_i,g_i)\}_{i\in N}\ ,$$

其中包括 1). 超长序列预训练：$h(s)$ 部分，将其建模为一个仅依赖于历史序列 s 作为输入、用户兴趣向量作为输出的子网络，支撑千级别的超长序列推理性能；2). 时空联合建模：$s=\{(a_i,t_i,g_i)\}_{i\in N}$ 部分的建模。下文具体介绍。

超长序列预训练：针对以上超长序列的性能问题，基于“长短分离”设计，将超长序列单独预训练（如图 7 所示），同短期表达分离，超长序列模块负责将用户全生命周期序列进行激活和降维，提取用户长期兴趣的表达。主模型负责将用户长期表达与短期表达结合。线上推理时，超长序列可以直接读取天级别缓存的表达向量，通过短期表达部分补充因无法实时更新的效果损失，然后再进行长短期兴趣表达结合，最终线上性能增加由原先 30+ms 降至 1ms 以内，实现千级别序列复杂推理。为能捕捉用户周期性行为兴趣，采用 Channel-wise Activator Layer 结构设计（如图 7 所示），利用长期行为兴趣漫游的特点将超长序列切分为能表达不同兴趣的时间子序列（Sub

Seq），并增加行为动态时间间隔表征后，同差异性属性进行多通道表达激活，实现基于类目周期性差异的兴趣表达。对比业界基于相关检索的超长序列方案 [7]，上述超长序列预训练并没有丢弃不相似的表达，充分建模全部行为。例如烧烤与素食、健身房和快餐之间意图负相关，但互相却又强影响，预训练建模以上负相关等更通用的行为影响，在美团数据集中也取得比检索方法 [7] 更显著的效果。

时空联合建模：依赖美团 O2O 的业务特点与挑战，我们充分考量其时间与空间属性，对用户的历史时空信息和当前时空信息进行了充分的建模和交互。我们分辨出用户行为的三重时空信息，即：用户点击发生时的时间、用户请求发出的地理位置、用户所点击的商户的地理位置。基于上述三重时空信息，我们提出 Spatio-temporal Activator Layer（如图 7 所示）：三边时空注意力机制神经网络来对用户历史行为进行建模，通过对请求经纬度信息、商户经纬度信息和请求时间的交互，囊括了同构空间信息深度交叉（用户历史请求位置和用户当前请求位置、用户历史点击商户位置和用户当前目标商户位置等）、异构空间信息深度交叉（用户历史请求位置和用户当前目标 POI 位置、用户历史点击商户位置和用户当前请求位置）、时间信息深度交叉的三边时空信息交互机制。针对空间信息交叉，我们进一步采用地理位置哈希编码和球面距离相结合的方式，以综合表达绝对空间信息和相对位置信息；针对时间信息交叉，我们也采用绝对与相对时间相结合的方式，有效实现用户行为序列在不同时空条件下的三边表达。从业务角度，针对不同的业务特性，我们进行空间信息表达的动态归一。比如在美食业务中，我们刻画用户搜索地和商户的相对距离信息；而在空间变迁属性更强的酒店业务中，当用户搜索词包含地址词时，我们刻画该地址和目标商户的相对距离信息。最后，经上述网络编码后的时空信息经过注意力机制网络融合，得到 LBS 场景下用户超长行为序列对不同请求候选的个性化表达。

基于时空依赖的超长序列建模在各个主广告位落地，RPM 提升 2%~5%。

详情请参考团队的 CIKM 2021 论文，Trilateral Spatiotemporal Attention Network for User Behavior Modeling in Location-based Search[8]。

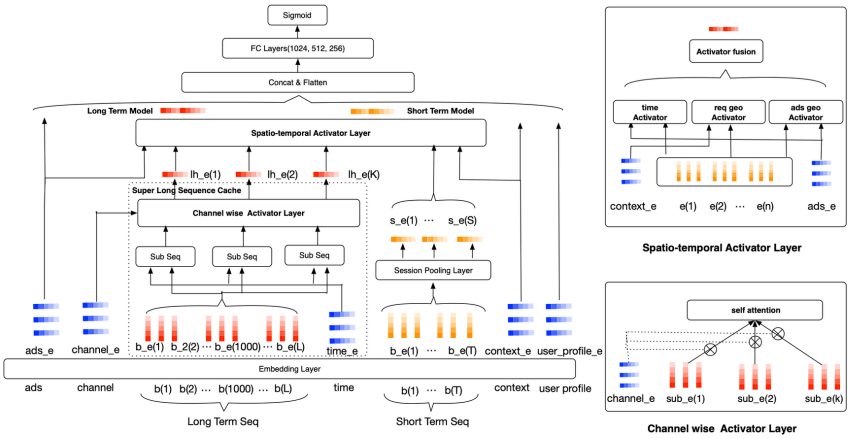


图 7　基于时空依赖的超长序列模型图

3. 广告：广告候选动态化

广告同用户存在着匹配关系，在用户意图精准理解的基础上，充分的广告供给能够进一步释放潜能。而在 LBS 距离约束挑战下，不同请求的广告候选存在很大差异。

因此如图 8 所示，我们从候选量、候选类型、候选算力三个维度构造广告候选动态化系统，实现精细化的供给匹配，具体优化如下：

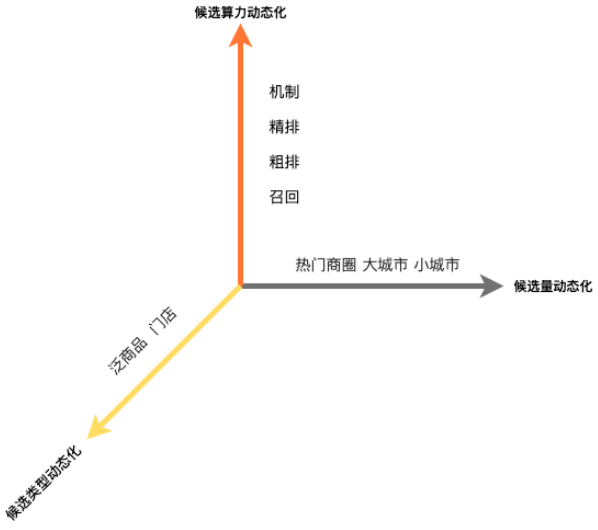


图 8　广告候选动态化三维图

候选量动态化：在广告 / 推荐领域，由于性能和资源限制，召回 –> 粗排 –> 精排 –> 机制，各层之间普遍存在漏斗候选截断，其中的精排候选数一般设置固定值。美团业务的距离约束特点显著，大型城市是小型城市平均候选数的 59 倍。根据城市、商圈等空间粒度进行候选量动态化后，可以在资源固定下最大限度释放热门地区的排序效果，特别是大型城市效果明显，RPM 提升约 20%。

由于候选量大幅度提升，精排模型有更严重的选择性偏差问题——线上面对的扩量候选与此前系统的候选不一致，为扩量前系统候选的超集。将候选扩量对应的流量理解为 Explore 流量，对 CTR 模型训练进行了多阶段优化，进一步取得效果 [13]，整体优化在部分筛选列表广告 RPS 提升 5%~10%。

候选类型动态化：在候选量动态化的基础上，我们进一步进行了候选类型动态化。因不同类型均为广告，我们在此处不采用 Ad 概念，在某些类目上，受 LBS 限制门店数较少，而每个门店下的泛商品（门店下更细粒度候选，同门店差异见上图，例如在结婚品类下的某婚纱照为商品）较多，将候选类型从门店扩展到门店 + 泛商品后，候选供给在部分类目可以扩大约 30 倍。并且在业务中，用户意图逐渐呈现具体化、明确化与多样化特点，例如“写真”等搜索关键词增多，粗粒度的门店候选已经不足以匹配细粒度用户意图。通过深度个性化的异构候选混排，能够匹配细粒度的用户意图，给用户展示最符合其兴趣偏好的候选类型。如图 9 所示，我们通过组合预估实现异构混排系统，主网络的高复杂性门店表征通过共有表达的迁移学习，和商品网络的高数量级商品表征进行维度动态转换对齐，实现数千量级的门店 / 泛商品异构混排系统。在部分筛选列表广告 RPM 提升 10%~15%。

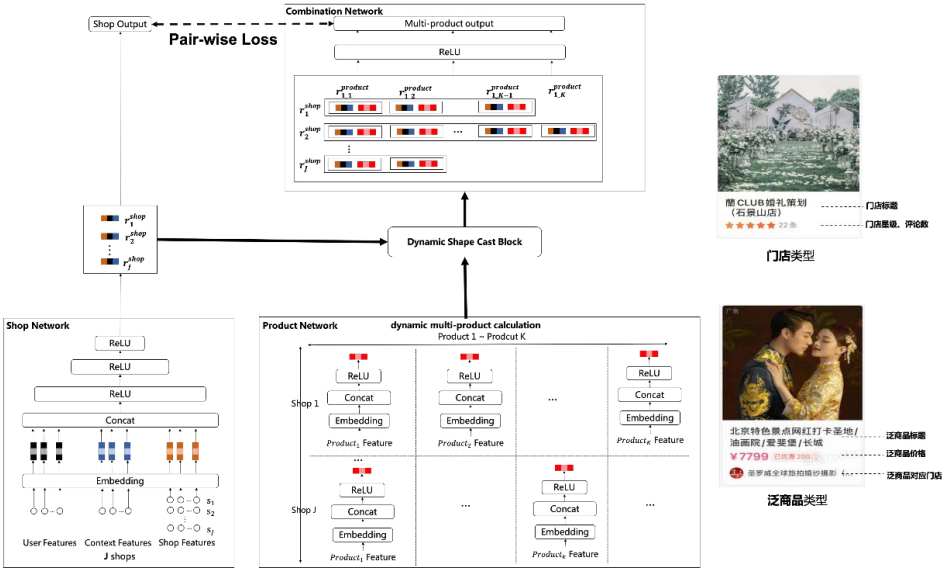


图 9　导构混排网络

候选算力动态化：通过动态的候选供给我们发现更多的候选能够带来效果空间，在此基础上我们进一步进行算力动态化，从性能角度进一步提升候选量级。广告系统收到请求后，会经过召回、粗排、精排、机制模块，最终生成广告展现顺序。其中各模块的超时时间和资源是固定的，导致总体算力缺乏弹性，容易因候选过多出现总体请求超时而不出广告的情况，或是候选过少出现多余算力浪费的现象。因此我们进行候选算力动态化，根据流量价值调整每个模块的耗时，通过全局优化实现在有限算力下收益最大化：首先在工程架构上增加各个服务算力弹性变量，如超时时间、模型复杂度和计算并行度等等，实现了效果和算力的灵活兑换。之后在算法上通过流量价值预估模块预估当前请求价值，确定各阶段打分数。在接下来的各服务阶段，结合当前算力消耗与预估价值情况，即时调整后续阶段算力弹性变量，进一步释放了效果空间。在筛选列表广告 RPM 提升约 2%。

候选动态化效果在各个主广告位落地效果差异较大，RPM 提升范围在 2%~15%。同时广告侧优化我们通过多模态表达学习取得效果，详情后续整理为论文公开。

4. 训练方式：灾难遗忘与持续学习

灾难遗忘 (Catastrophic Forgetting)[10] 是模型学习同人脑学习的一个重要差异，指的是同一个网络模型，在学习新的任务的时候，因分布变化而导致老任务学习的模型权重损坏。在美团广告业务，无论是用户和商户均存在周期性特点，并且高维稀疏网络特性会放大这个特点，现象是在拟合新数据模式过程中，遗忘具备周期性的老数据，形成了灾难遗忘。下面会介绍长周期性挑战，与高维稀疏网络结构问题，以及对应的持续学习解决方案。

周期性数据模式：不同于传统机器学习数据独立同分布假设 (i.i.d.)，CTR 预估模型面临着动态非平稳的外部环境。特别地，在我们的 O2O 业务中，新用户、新商户不停出现，流行元素如小龙虾、烤肉等不时演变，夹杂着周中简餐周末大餐等节日性、季节性、星期级，小时级特征，对 CTR 预估模型的持续学习提出了挑战，要求我们的模型能够快速拟合新出现的数据模式。针对这类时变数据的学习任务，一种常见选择是流式训练。流式训练将数据按照时间顺序依次喂入模型，以 Regret[11] 为目标让模型实时动态地调整模型参数，达到拟合最新数据分布的目的。但是，单纯地流式训练有过分强调新数据模式而遗忘过往学习到的老数据模式的危险。特别地，在我们的 O2O 业务场景下，存在大量周期性数据模式，下面将以星期级周期为例进行介绍。

高维稀疏网络的结构匹配性：不同于 NLP/CV 领域，广告 / 推荐领域的 DNN 模型具有显著的参数不平衡性，即高维度稀疏的 Embedding 层约有亿级 ~ 千亿级别参数量的参数量，而连接 Embedding 到 Output 的 FC 层 (Fully Connected Layers) 却低维稠密，网络中极度稀疏的 Embedding 层占据 99.99%+ 的参数量，该不平衡性易于引发老数据模式的遗忘问题。以天级更新模式的星期几特征为例，线上推理的数据是周六 (t, w(t) = 6) 数据，训练时在未充分纳 t 时刻数据时，会更多的拟合在周五 (t−1) 特征上，而实际上在上一周的周六 (t−7) 与线上推理数据 (t) 具有更强的分布一致性。特别地，如果我们使用星期几这一离散特征，在高维稀疏的 Embedding 层在线上推理的 t 时刻会直接查询到 t−7 天的 Embedding，而低维稠密的 FC 层会查询到 t−1 的 FC 层，会引起严重的结构不匹配问题，形成星期属性模式的灾难性遗忘。

更具一般性地，训练中存在着周中简餐，周末大餐等复杂模式的灾难遗忘，引起效果大幅退化。

面向未来分布的持续学习：为系统性地解决上述问题，我们采用持续学习中的数据回放技术，在流式训练的基础上进一步拟合回放的数据，加强模型在无遗忘约束下的持续学习能力。具体地，为结合流式训练和持续学习的数据回放技术，我们采用 FC 层分多塔的结构来达成预训练模型可流式更新、持续学习模型可依赖最新预训练模型更好地拟合当前数据的效果。如下图 10 所示，通过流式训练我们维护一个预训练模型；而为避免模型过分拟合最新数据而导致历史稀疏 Embedding 层参数和上层 FC 层参数不匹配的问题，通过数据控制器 (Data Controller) 维护一份记忆集 (Memory Set)，其中的数据为根据未来分布预测，在历史数据中采样得到，并将该份记忆集提供给基础模型用于持续学习，从而在加强对未来分布数据的模型预估能力。

持续学习技术在各个主广告位落地，RPM 提升 2%~3%。更多详情已整理为论文投稿中，后续会进行公开。

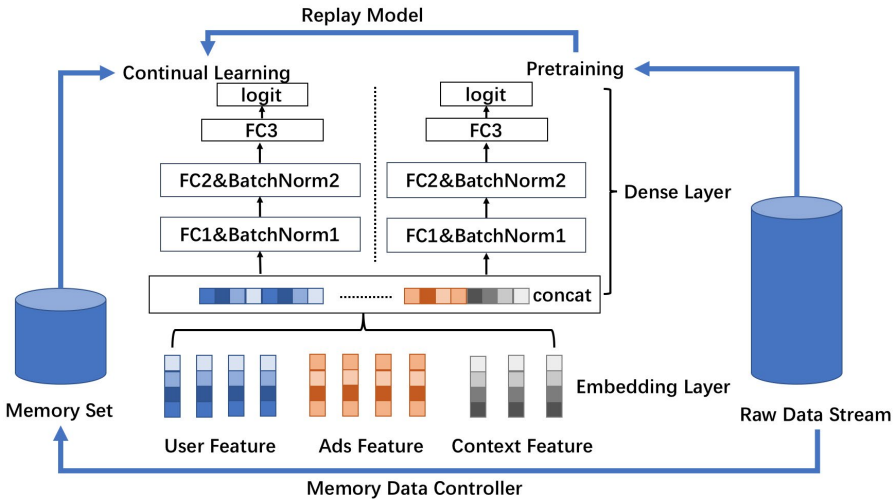


图 10　灾难遗忘与持续学习训练模式

预估技术新趋势

上文介绍了基于 O2O 特点的 CTR 技术突破，我们将优化方法推广在转化率、交易额预估问题，其中 LBS 距离约束在转化率中起到更重要的影响，因方法接近，不再进行赘述。

同一技术的点击率预估优化效果在不同业务类型中也存在一定差异，意图强的搜索广告场景，上下文差异性更大，Context 相关优化有更明显效果。而在意图弱的信息流广告场景，User 建模有更大的空间。

那么预估技术如何持续突破的系统上线？迭代路径从前期通过模型复杂度提升，到现阶段结合问题的模型灵活设计均取得效果，而未来我们认为将越来越剥离模块与目标设置的现状，消除建模中人为的设置假设。

如何抽丝剥茧，回归问题本质？我们认为有如下新趋势。

a. 模型内推理量级动态化

广告系统通常分为召回、粗排、精排（预估）等多个模块，模块漏斗遵循的一个核心设计逻辑是简单模块打分量级大（例如粗排），复杂模块打分量级低（例如精排）。以递归视角进一步理解子模块设计，将设计逻辑泛化到模型内，同样存在简单的网络部分打分量大，复杂的网络部分打分量低，同一个网络打分量差异化的优化方式。例如粗排常用的双塔网络结构 (Two-tower Network) 在 User 端打分一次，Ad 端打分多次。

更具一般性的形式并不局限在表达能力受限的双塔结构，而在于更充分的应用打分量与复杂度的关系进行灵活设计，例如本文阐述的 DPIN(Deep Position-wise Interaction Network)[7] 组合网络部分打分量大，基础网络部分打分量小，同在本文阐述的异构混排网络，超长序列预训练优化同样适用于这一趋势。

未来将在更细化的层面进行推理量级动态化，通过每一个局部精度与性能的平衡达到整体的最优平衡。

b. 问题间评估指标差异化

评测指标是一个比优化方式甚至更重要的问题，点击率预估最经典的评测指标为AUC，而随着模型精度的提升，离线 AUC 提升同线上不一致的情况逐渐增多。回归到预估问题本身，实际排序系统是在固定请求固定位次下的最优排序，等价寻优User × Context × Time 限定条件的 Ad 最优序。

time 条件在持续变化难建模，为简化问题，以 User/Context 维度分组，近似寻优Ad 最优序。基于以上假设，User/Context 分组的“组间序”同“组内序”并不具备等价的业务意义。例如个性化优化问题，每一个 User 作为一组进行分组为例，User Group AUC[12] 同线上具有更强的一致性；而在 Context 中的 Position Bias 问题中，我们发现通过 Position Group AUC(PAUC)[7] 评估同线上系统具有更强的一致性。在全局 AUC 增长挑战性变大的背景下，不同问题进行差异性评估指标设计，进而指导模型优化，是取得效果的一个关键趋势。

c. 链路模块间目标多元化

系统整体目标面对排序问题，先是解耦为召回、粗排、精排等多模块独立优化，而后针对整体性排序问题，链路目标一致性优化成为一个明显的演进趋势。在召回 –> 粗排 –> 精排目标一致，取得明显效果的同时，存在着因多模块过于同质化、协同共振引起的马太效应。以召回为例，是否应该由其他路模块，例如探索路召回来解决这个问题?

更具泛化性地，多路召回之间的强互补性，能够提供给下游更优的候选超集，召回各个路之间面向 Diversity Loss 进行优化是未来的探索性方向之一。随着模块间关系动态变化，广告系统目标设计存在着一致性与差异性的辩证关系。

d. 搜索推荐边界模糊化

从广告视角，团队内的搜索广告、推荐广告间做法呈现逐步边界模糊化特点。传统搜索广告围绕意图明确进行 Query 理解，进而内容适配。逐渐 Query 相关的刻画不能完全满足用户意图的细化，搜索问题逐渐转变为在 Query 约束下的个性化匹配问题。

边界模糊化带来的一个技术现象就是全域数据的运用，搜索、推荐间的全方位深度迁移学习是取得效果的一个关键。

e. 模型学习非监督化

如图 11 所示，整个广告系统中，是在用户交互、数据、模型之间不断产生反馈循环，在这个过程中会有偏差循环放大的问题，例如产生偏差的数据会喂入模型影响展示，数据回溯到模型形成进一步归纳偏差。

传统的建模方式是将整个动态的系统简化为监督问题进行预估建模，优化集合仅为线上集合的子集，效果限制随迭代愈发明显。未来针对监督模型无法充分建模的候选超集依然有较大空间。在实际动态演化的广告系统中，无论是新目标定义、模型优化还是 AB Test 评测方式都有较大的探索潜能。

Debiasing 问题优化详情参照团队的 KDD Cup 2020 冠军技术分享，**KDD Cup 2020 Debiasing 比赛冠军技术方案及在美团的实践** [13]。

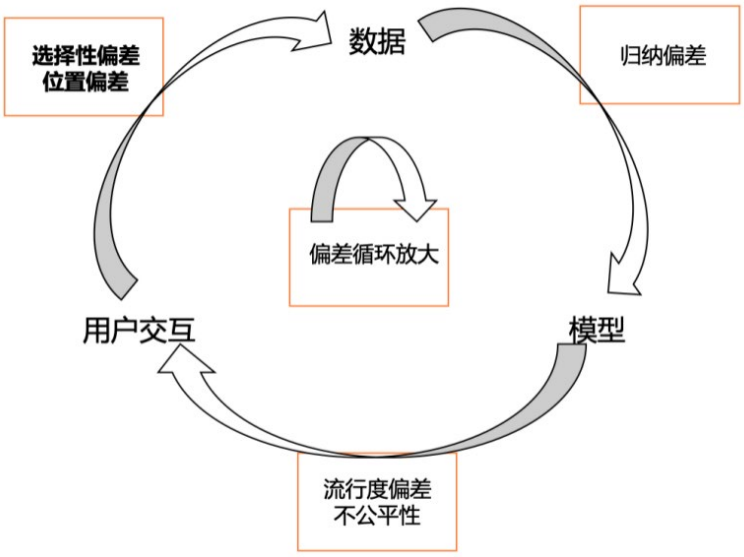


图 11　广告系统中的反馈环路，偏差积累循环

作者简介

胡可、坚强、张博、漆毅、庆涛、曲檀、程佳、雷军等，均来自美团广告平台技术部。

参考文献

[1] Friedman J H . Greedy Function Approximation: A Gradient Boosting Machine[J]. Annals of Statistics, 2001, 29(5):1189-1232.
[2] Rendle S. Factorization machines[C]//2010 IEEE International conference on data mining. IEEE, 2010: 995-1000.
[3] HT Cheng, et al. Wide & Deep Learning for Recommender Systems, 2016
[4] Zhou, Guorui, et al. "Deep interest network for click-through rate prediction." Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. 2018
[5] Modeling Task Relationships in Multi-task Learning with Multi-gate Mixture-of-Experts. ACM, 2018.
[6] Wen, Ling, Chua. A closer look at strategies for memorization.[J]. Clavier Companion, 2014, 6(6):50-52.
[7] Huang J, Hu K, Tang Q, et al. Deep Position-wise Interaction Network for CTR Prediction[J]. arXiv preprint arXiv:2106.05482, 2021.
[8] Search-based User Interest Modeling with Lifelong Sequential Behavior Data for Click-Through Rate Prediction
[9] Qi, Yi, et al. "Trilateral Spatiotemporal Attention Network for User Behavior Modeling in Location-based Search", CIKM 2021.
[10] Overcoming catastrophic forgetting in neural networks[J]. Proceedings of the National Academy of Sciences of the United States of America, 2017.
[11] M. Zinkevich. Online convex programming and generalized infinitesimal gradient ascent. In ICML, 2003.
[12] Optimized Cost per Click in Taobao Display Advertising[C]// the 23rd ACM SIGKDD International Conference. ACM, 2017.
[13] https://tech.meituan.com/2020/08/20/kdd-cup-debiasing-practice.html

招聘信息

美团到店广告平台广告算法团队立足广告场景，探索深度学习、强化学习、人工智能、大数据、知识图谱、NLP 和计算机视觉前沿的技术发展，探索本地生活服务电商的价值。主要工作方向包括：

- 触发策略：用户意图识别、广告商家数据理解，Query 改写，深度匹配，相关性建模。
- 质量预估：广告质量度建模。点击率、转化率、客单价、交易额预估。
- 机制设计：广告排序机制、竞价机制、出价建议、流量预估、预算分配。
- 创意优化：智能创意设计。广告图片、文字、团单、优惠信息等展示创意的优化。

岗位要求

- 有三年以上相关工作经验，对 CTR/CVR 预估、NLP、图像理解，机制设计至少一方面有应用经验。
- 熟悉常用的机器学习、深度学习、强化学习模型。
- 具有优秀的逻辑思维能力，对解决挑战性问题充满热情，对数据敏感，善于分析 / 解决问题。
- 计算机、数学相关专业硕士及以上学历。

具备以下条件优先：

- 有广告 / 搜索 / 推荐等相关业务经验。
- 有大规模机器学习相关经验。

感兴趣的同学可投递简历至：chengxiuying@meituan.com（邮件标题请注明：美团广平算法团队）。

小样本学习及其在美团场景中的应用

作者：骆颖　徐俊　谢睿　武威

1. 背景

美团的各个业务有着丰富的 NLP 场景，而这些场景中模型的构建需要很多的标注资源，成本很高。小样本学习致力于在数据资源稀少的情况下训练出比较好的模型，主要有以下两个评价标准：

- **提升算法效果**：在标注资源一定的情况下，使用小样本学习能够尽可能多地提升相应的指标。
- **节省标注数据**：在算法效果希望达到一定水平的情况下，希望尽可能减少标注数据。

在 NLP 领域少样本的场景下，主要存在以下三种场景，小样本学习针对这些问题采取了不同的措施：

1. 样本空间稀缺（图 1 左）：样本数量较少时，分布稀缺，数据增强旨在更好利用样本 /Embedding 之间的关系，提高模型泛化性能。
2. 样本分布在局部空间（图 1 中）：某个领域往往只有少量标注数据，而有大量的未标注数据。根据对未标注数据的使用方式不同，我们将其划分为两种，第一种是半监督学习，是在模型 Finetune 过程中同时学习标注样本和未标注样本，利用了模型对未标注数据的预测一致性；第二种是集成学习 + 自训练，强调的是融合多个模型对未标注数据的预测结果作为伪标注数据加入训练。
3. 不同领域之间的样本分布差异（图 1 右）：在某个领域充分学习到标注信息之后，因样本空间有差异，无法直接用到其他领域，迁移学习旨在学习到一个领域的充分知识后，能够快速学习到其他领域的知识。

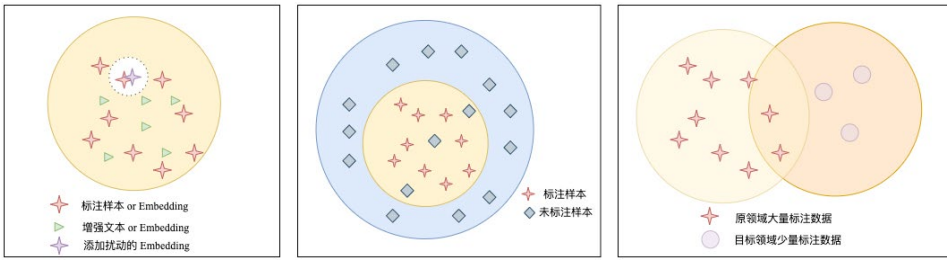


图 1　小样本学习三种应用场景

除了上面提到的三种场景，还有一种是，如何在有限的标注成本中选择更有针对性的样本进行人工标注（主动学习）。因此我们将小样本学习划分为下面几种：

- **数据增强**：数据增强可以分为样本增强和 Embedding 增强。样本增强早先在计算机视觉中对图像进行数据增强，图像的一些简单操作，如将图像旋转或将其转换为灰度，并不会改变其语义，语义不变变换的存在使增强成为计算机视觉研究中的一个重要工具。在 NLP 领域中的样本增强也试图不改变句子主旨来扩充文本数据，主要方法有简单文本替换、预训练语言模型生成相似句子等，对得到的增强数据可以使用课程学习的方式由简到难地进行学习。Embedding 增强在模型的 Embedding 层进行操作，可以通过对 Embedding 加扰动 / 插值等，提升模型的鲁棒性。
- **半监督学习**：监督学习往往需要大量的标注数据，而标注数据的成本比较高，因此如何利用大量的无标注数据来提高监督学习的效果，具有十分重要的意义。近年来，半监督深度学习取得了长足的进展，特别是在计算机视觉领域；相关的研究主要着力于如何针对未标注数据构建无监督信号，与监督学习联合建模，目前的主要方法都是基于无标注数据的一致性正则构建损失函数。
- **集成学习 + 自训练**：机器学习的有监督学习目标是学习出一个稳定的且在各个方面表现都较好的模型，但实际情况中有时只能得到多个有偏好的模型（弱监督模型，在某些方面表现的比较好）。集成学习就是组合这里的多个弱监督模型以期得到一个更好更全面的强监督模型，集成学习潜在的思想是即便某一个弱分类器得到了错误的预测，其他的弱分类器也可以将错误纠正回来。单纯使

用多个模型在预测时做集成会增加线上负担，因此我们利用多个模型对大量无标注数据进行预测，选取组合置信度较高的数据合并到训练集进行训练，最后将多个模型的优势集成到一个统一的模型上。这部分也属于半监督学习的一种，但是与上面提到的半监督学习的主要区别是后者提到的方法都是强调在 Finetune 阶段利用模型对未标注数据的预测一致性，而这部分强调融合多个模型的预测结果，故单独将这部分单独列出。

- **Few-shot Learning/ 领域迁移**：人类在具备一定的知识储备后可以快速学习新的知识，研究人员希望机器学习模型也能具备这种能力，一个模型已经从一定类别的大量数据学习许多信息后，对于新的类别可以依据新的少量的标记样本得到新的分类器，该分类器可以在新的类别中识别出该类样本。
- **主动学习**：主动学习是一个机器学习和人工参与迭代的过程，通过机器学习的方法筛选出合适的候选集给人工标注。大致思路是：通过机器学习的方法获取到那些比较“难”分类的样本数据，让人工再次确认和审核，然后再将人工标注得到的数据再次使用有监督学习模型进行训练，逐步提升模型效果。

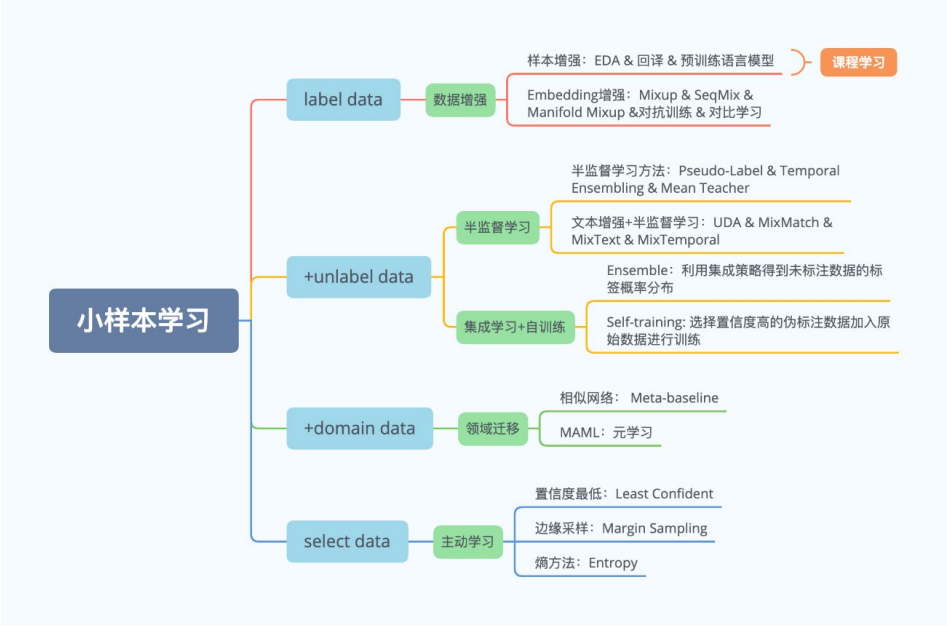


图 2　小样本学习相关工作

2. 方法综述

预训练语言模型 BERT，在 NLP 许多任务中取得非常好的结果。BERT 是基于 Transformer 的深度双向语言表征模型，利用 Transformer 结构构造了一个多层双向的 Encoder 网络。我们采取 BERT 作为 Baseline 模型，对预训练好的模型在特定任务样本上进行 Finetune。

2.1 数据增强

数据增强可以分为数据扩充和 Embedding 增强，数据扩充在保持语义不变的情况下，变换文本的表达形式，如回译、同义词替换、随机删除等等；模型增强主要有 Mixup 和对抗训练，Mixup 在 NLP 领域有一系列的变体，包括 SeqMix、Manifold Mixup 等。数据增强提高了模型的鲁棒性，使得模型更关注文本的语义信息，并对文本的局部噪声不再敏感。在少样本场景，文本增强技术可以有效提高模型的鲁棒性，提高其泛化能力。

2.1.1 样本增强

2.1.1.1 简单数据增强 EDA

EDA[1] 通过知识库来替换句子中的某些词 / 短语，主要包含以下四种操作：

- **同义词替换（Synonym Replacement，SR）**：不考虑停用词，在句子中随机抽取 n 个词，然后从同义词词典中随机抽取同义词，并进行替换。
- **随机插入（Random Insertion，RI）**：随机的找出句中某个不属于停用词集的词，并求出其随机的同义词，将该同义词插入句子的一个随机位置。重复 n 次。
- **随机交换（Random Swap，RS）**：随机的选择句中两个单词并交换它们的位置。重复 n 次。
- **随机删除（Random Deletion，RD）**：以概率 p 随机删除句子中每个单词。

2.1.1.2 回译（Back Translation）

回译是 NLP 在机器翻译中经常使用的一个数据增强的方法，它的本质就是快速产生

一些翻译结果达到增加数据的目的。回译将原有数据翻译为其他语言再翻译回原语言，由于语言逻辑顺序等的不同，回译的方法也往往能够得到和原数据差别较大的新数据。

2.1.1.3 预训练语言模型

- 基于上下文信息的文本增强[2]，利用训练好的语言模型，随机 Mask 文中的一个词或字，再将文本输入语言模型，选择语言模型所预测的 Top k 个词去替换原文中被去掉的词，以形成 k 条新的文本。
- 基于语言生成模型的文本增强 Lambda[3]，Lambda 基于预训练语言模型 GPT，使模型能够捕获语言的结构，从而能产生连贯的句子。在不同任务的少量数据集上对模型进行微调，并使用微调后的模型生成新的句子。

2.1.2 增强样本使用

上面几种方式生成了一批数据增强文本，增强后的文本数量多、带噪音；原始标注数据数据量少、不含噪音。NAACL2021[4] 的一篇工作提出了一种课程学习的方式更好地学习这两种样本。

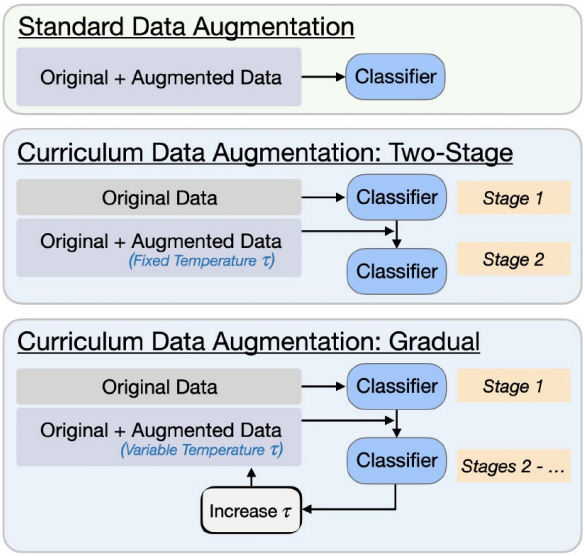


图 3　增强数据三种学习方式

Standard Data Augmentation

直接将原始数据和增强后的数据融合起来进行训练。

Curriculum Data Augmentation

课程学习先学习简单的标注数据，具备一定知识之后再学习带噪音的增强数据，有以下两种使用方式：

1. Two Stage：先在原始数据上进行训练，等到开发集收敛之后再在原始数据和增强数据上一起训练。
2. Gradual：先在原始数据上进行训练，然后逐步以线性方式添加增强数据，通过控制扰动变量 \tau τ，即一开始 \tau=0 τ=0，然后每次线性增加 0.1，直到增加到 0.5。

2.1.3 Embedding 增强

2.1.3.1 Mixup

自然语言具备系统组合性（Systematic Compositionality），能够理解通过适当重组的句子。考虑以下样例，左边两个为原始文本，通过对其中的某些词进行替换 / 重组可以生成右边可理解的样本 [5]。

图 4　自然语言重组变换样例

Mixup[6,7] 提出了一种更通用的，基于向量增强的模型，从训练数据中任抽样两个样本，构造混合样本和混合标签，作为新的增广数据。其中 (x_i, y_i) 和 (x_j, y_j) 为原始样

本，$(\hat{x}, \hat{y})$ 为重组生成的新样本。当 λ 的取值限制在 {0, 1}，就会产生图 4 的组合。

$$\hat{x} = \lambda x_i + (1-\lambda)x_j \ ,$$

$$\hat{y} = \lambda y_i + (1-\lambda)y_j \ ,$$

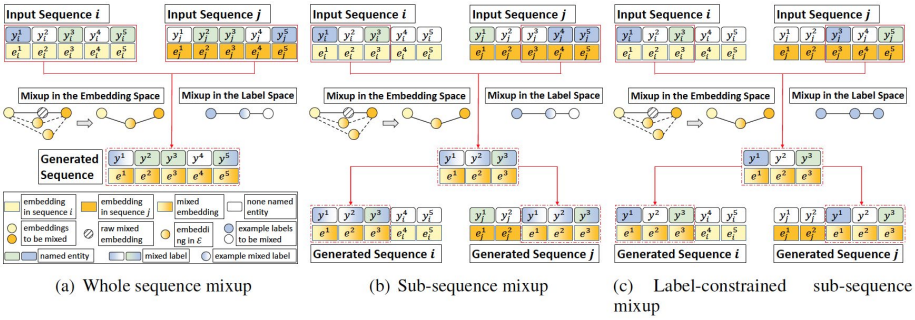
实验中 λ 的取值满足 Beta 分布。

$$\lambda \sim \text{Beta}(\alpha, \alpha)$$

直观来看，它要求当模型的输入为另外两个输入的线性组合时，输出也是这两个数据单独输入模型后所得输出的线性组合，这其实就是要求模型近似为一个线性系统，防止模型过拟合，Mixup 变换本身可以视为一种正则化技术。 下面介绍几种 Mixup 的变体，SeqMix 和 Manifold Mixup。

a. SeqMix

SeqMix[8] 对 Mixup 生成的句子的困惑度打分，用 Discriminator 判断生成的句子困惑度，只选择困惑度较低的句子，减少噪音句子的影响。同时对于更关注于词的任务（如命名实体识别）替换 Sub-Sequence 而不是整个句子。



(a) Whole sequence mixup (b) Sub-sequence mixup (c) Label-constrained sub-sequence mixup

图 5 SeqMix 的几种变体

- 图 5（a）最原始的 Mixup，对整个序列做 Mixup，生成新序列。
- 图 5（b）子序列 Mixup，对合法的子序列做 Mixup，并分别代替原来的子序列，生成两个新序列。

- 图 5（c）标签限制的子序列 Mixup，只对同类别标签的子序列做 Mixup，分别代替原来的子序列，生成两个新序列。

b. Manifold Mixup

流形混合算法 Manifold Mixup[9] 将上述的 Mixup 操作泛化到特征上；特征具有更高阶的语义信息，在其维度上插值可能会产生更有意义的样本。在类似于 BERT 的模型中，随机选择层数 k，对该层的特征表示进行 Mixup，具体操作如下：

- 随机选取网络第 k 层（包括输入层）。
- 传两批数据给网络，前向传播直到第 k 层，得到隐藏特征表示 $(g_k(x), y)$ 和 $(g_k(x'), y')$。
- 对其进行 Mixup 生成新样本 $(\tilde{g}_k, \tilde{y}) = (\text{Mixup}_\lambda(g_k(x), g_k(x')), (\text{Mixup}_\lambda(y, y'))$
- 继续前向传播得到输出。
- 计算损失值和梯度。

2.1.3.2 对抗训练

对抗训练 (Adversarial training, AT)[10] 通过在输入样本上增加微小的扰动来显著提高模型 Loss，对抗训练就是训练一个能有效识别原始样本和对抗样本的模型。如果模型在对抗噪声下，依然能够保持光滑，那么整个网络就能够表现出很好的一致性。当将对抗训练应用到分类器时，对抗训练对应的损失函数为（在原有损失函数的基础上又增加了一项）：

$$-\log(p(y \mid x + r_{adv}); \theta); \text{where } r_{adv} = \arg\min_{r, \|r\| \le \epsilon} \log p(y \mid x + r; \tilde{\theta})$$

其中 x 为输入序列，θ 为模型参数，r 为输入上的扰动，$\tilde{\theta}$ 表示将模型的当前参数设置为常数，即表明在构建对抗样本时，反向传播算法不会对当前模型参数进行更新。其中 r_{adv} 的计算为：

$$r_{adv} = \epsilon \frac{g}{\| g \|_2} \text{ where } g = \nabla_x \log p(y \mid x; \tilde{\theta})$$

2.1.3.3 对比学习 R-Drop

Dropout 是深度学习模型常用的正则化手段，Regularized Dropout (R-Drop)[11] 对同一个句子做两次 Dropout，并且强制由 Dropout 生成的不同子模型的输出概率保持一致。具体来说，对于每个训练样本，R-Drop 最小化了由不同 Dropout 生成的子模型的输出概率之间的 KL 散度。

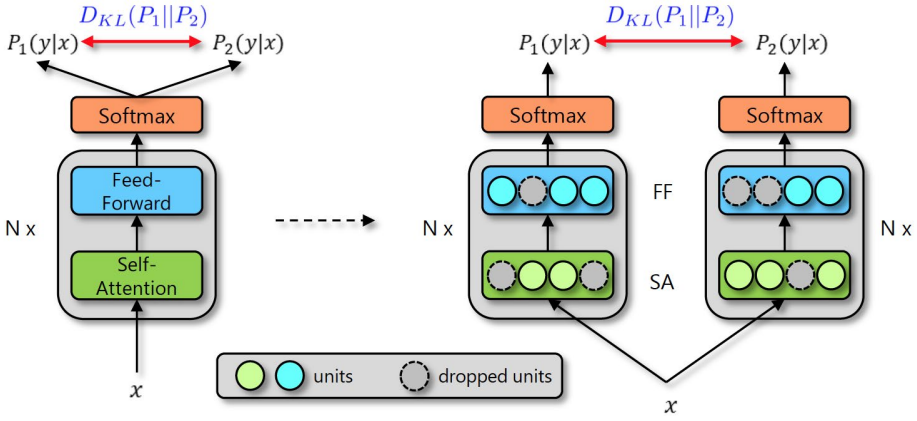


图 6　R-Drop 模型图

图 6 中左边图表示了每个输入样本 (x, y) 都会经过模型两次，得到两个概率分布，右图展示了由于 Dropout 本身的随机性，对同一个样本重复两次就可以得到两个子模型。

给定训练数据 $D = \{(x_i, y_i)\}_{i=1}^{n}$，模型为 $P^w(y \mid x)$，训练数据的交叉熵损失为：

$$L_{CE} = \frac{1}{n}\sum_{i=1}^{n} -\log P^w(y_i \mid x_i)$$

通过不同的 Dropout 可以得到两个子模型 $P_1^w(y \mid x)$ 和 $P_2^w(y \mid x)$，R-Drop 最小化两个子模型输出概率的双向 KL 散度：

$$L_{KL}^i = \frac{1}{2}(D_{KL}(P_1^w(y_i \mid x_i) \,||\, P_2^w(y_i \mid x_i)) + D_{KL}(P_2^w(y_i \mid x_i) \,||\, P_1^w(y_i \mid x_i)))$$

对于训练数据 (x_i, y_i)，最终的损失函数为：

$$L^i = L^i_{CE} + \alpha L^i_{KL} = -\log P_1^w(y_i \mid x_i) - \log P_2^w(y_i \mid x_i) + \frac{\alpha}{2}(D_{KL}(P_1^w(y_i \mid x_i) \| P_2^w(y_i \mid x_i)) + D_{KL}(P_2^w(y_i \mid x_i) \| P_1^w(y_i \mid x_i)))$$

2.2 半监督学习

在实际业务中，有大量的未标注数据，对数据进行标注需要大量的人力。半监督学习的核心目标是，希望通过标注的少量有标签数据，结合大量的无标签数据，训练出具备强泛化能力的模型，从而解决实际中的问题。

- 输入数据：大量的同领域未标注数据和少量的有标注数据。
- 主要基于以下理论基础：

1. 自洽正则化（Consistency Regularization）：对未标记的数据进行增广，产生的增广数据输入模型，它的预测值应该与原数据的预测值一致，即保持自洽。
2. 最小化熵（Entropy Minimization）：基于一个规则，分类器的分类边界不应该穿过边际分布的高密度区域。具体做法为强迫分类器对未标记数据作出低熵预测。
3. 传统正则化（Traditional Regularization）：为了提高模型泛化能力，防止过拟合的方法，如 L2 正则化等。

- 如何构造半监督学习的自洽正则化项是很多半监督模型研究的重点。
  - Temporal Ensemble、Mean Teacher、MixTemporal 这几种方法的思想都是利用了对历史模型的集成，利用历史模型预测结果 / 模型参数构造一致性正则。
  - VAT 则是利用了给 Embedding 加上扰动构造正则项。
  - MixMatch、MixText、UDA 则是利用了样本增强的结果，MixMatch 利用了图像的旋转、缩放等，MixText 和 UDA 依赖于通过回译生成的样本之间的一致性，模型结果很大一部分取决于样本增强的质量。

2.2.1 Temporal Ensembling

Temporal Ensembling[12] 采用时序融合模型，将历史预测结果的指数滑动平均（EMA）作为未标注数据的伪标签，从而和当前的预测结果构造一致性正则项。

- 有标注数据计算交叉熵损失。
- 未标注数据采用历史多次预测结果的指数滑动平均（EMA）作为重构目标计算 MSE 损失，避免采用模型一次的预测作为重构项会产生较大误差，有利于平滑单次预测中的噪声。
- 有标注数据的交叉熵损失和未标注数据的均方误差加权和作为最终损失函数。

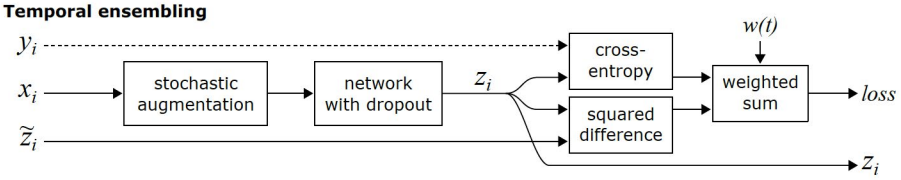


图 7　Temporal Ensembling 模型结构

其中 x_i 中包含有标注数据和未标注数据文本，y_i 表示标注数据标签，z_i 表示当前模型对 x_i 的预测结果，$\tilde{z}_i$ 表示对未标注数据多个 epoch 预测结果的滑动平均值作为其概率分布，$w(t)$ 表示未标注数据 MSE 所占权重，模型刚开始训练时 $w(t) \to 0$，此时模型倾向于学习标注数据标签，随着迭代次数增加，$w(t)$ 慢慢增加，此时模型学习大量未标注数据。

2.2.2 Mean Teacher

Mean Teacher[13] 思想上与 Temporal Ensembling 模型基本一致，Temporal Ensembling 需要矩阵保存历史预测概率，Mean Teacher 模型则是对模型参数做指数滑动平均 EDA 作为 Teacher 模型，原模型作为 Student 模型。用 Teacher 模型和 Student 模型的预测结构构造一致性正则。

- 在 Temporal Ensembling 中，无标签数据的目标标签来自模型前几个 Epoch 预测结果的加权平均。而在 Mean Teacher 中，无标签数据的目标标签来自

Teacher 模型的预测结果。

- 由于是通过模型参数的平均来实现标签预测，因此在每个 Step 都可以把无标签中的信息更新到模型中，而不必像 Temporal Ensembling 模型需要等到一个 Epoch 结束再更新。

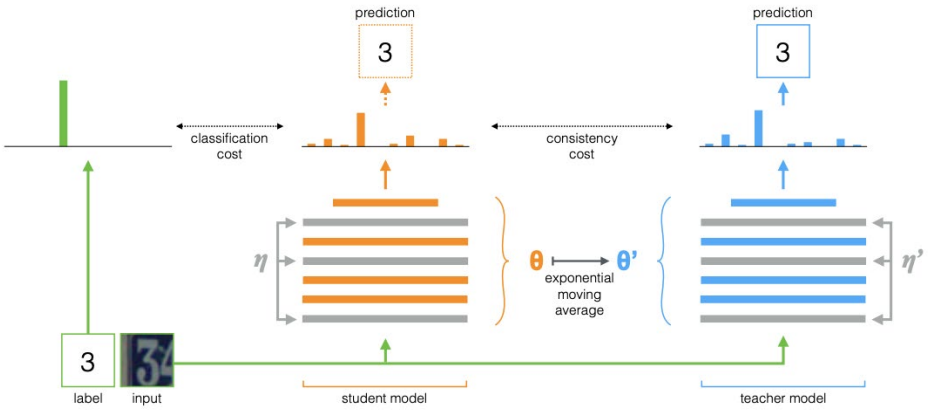


图 8　Mean Teacher 模型结构

2.2.3 VAT

虚拟对抗网络（Virtual adversarial training, VAT）[14] 与 Temporal Ensembling 模型的不同之处在于，后者采用数据增强、Dropout 来对无标签数据施加噪声，而前者施加的则是模型变化最陡峭方向上的噪声，即所谓的对抗噪声。如果模型在对抗噪声下，依然能够保持光滑，那么整个网络就能够表现出很好的一致性。虚拟对抗训练通过对模型添加正则使得一个样本的输出分布与加扰动之后的输出分布相同来将对抗训练扩展到半监督领域。

- 首先对未标注数据抽取随机标准正态扰动 $d \sim \mathcal{N}(0,1)$ 加到 Embedding 上，并用 KL 散度计算梯度。
- 然后用得到的梯度计算对抗扰动，并进行对抗训练。
- 标注数据的交叉熵损失和未标注数据对抗损失作为最终损失函数。

2.2.4 MixMatch

MixMatch[15] 是集大成者，将数据增强、Mixup、Sharpening 等方法融合起来，起重要作用的两个模块就是 Mixup 和 Sharpening。

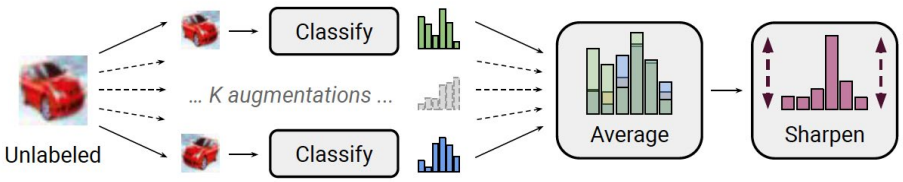


图 9　MixMatch 无标注数据标签构造

- 在图像领域未标注数据的 k 条增强数据来自图像的旋转、缩放等。
- 输入标注数据 $X=(x_b,p_b)$，未标注数据 $U=(u_b)$，利用图 9 生成未标注数据的标签 q_b，q_b 即是模型对未标注数据和它的条增强数据预测结果的平均值。
- 将 $X=(x_b,p_b)$ 和 $U=(u_b,q_b)$ 混合做 Mixup 操作。
- 计算标注数据的交叉熵损失和未标注数据的一致性正则损失项。

2.2.5 MixText

MixText[16] 整个思路和 MixMatch 是一致的。首先利用 Back Translation 对未标注数据做数据增强，利用模型对增强后的数据和原数据的预测结果的加权和作为未标注数据的标签，再对有标注数据和未标注数据同时做 Mixup。

- 无标注数据通过 Back Translation 做数据增强。
- 利用模型对增强后的数据和原数据的预测结果的加权和作为无标注数据的标签。
- 同时对有标注数据和无标注数据做 Mixup。
- 学习有标注数据的交叉熵损失和无标注数据的一致性正则。

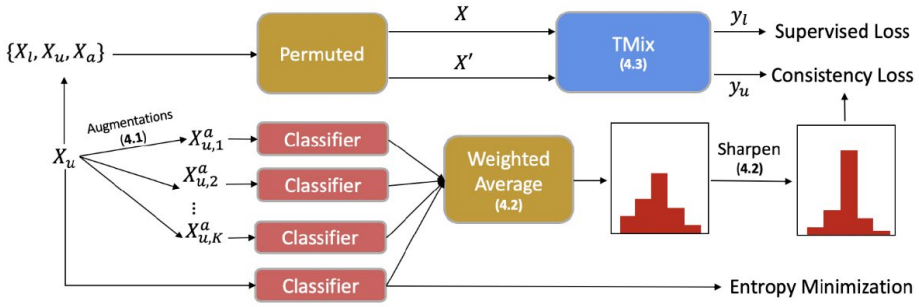


图 10　MixText 模型结构图

其中 x_l 是标注样本，x_u 是未标注样本，x_a 是未标注样本的回译增强样本。与 Manifold Mixup 相似，同样是对模型第 mm 层的隐藏表示做 Mixup，可以挖掘到句子之间的隐式关系。对有标签和无标签数据同时进行插值，可以在学习有标签句子的同时利用无标签句子的信息。

2.2.6 MixTemporal

借鉴 MixMatch 和 MixText 思想，但对无标注数据的预测不是通过 EDA 或者回译这种数据增强，而是利用 Temporal Ensemling 的思想，对多个 Epoch 历史预测结果的指数滑动平均（EMA）作为未标注数据的伪标签，不需要额外的同义词典和翻译工具等，实现较为方便。

最终损失函数计算两部分组成：Mixup 之后的交叉熵 Loss+ 无标注数据一致性正则 Loss。

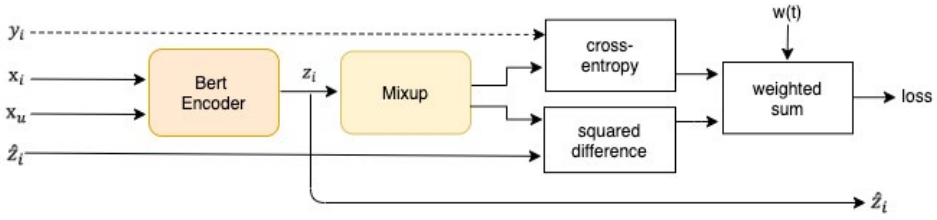


图 11　MixTemporal 模型结构图

其中 x_i 是标注样本，y_i 表示标注样本标签，z_i 表示当前模型对 x_i 的预测结果，$\tilde{z}_i$ 表示对未标注数据多个 Epoch 预测结果的滑动平均值作为其概率分布，$w(t)$ 表示未标注数据 MSE 所占权重。

2.2.7 UDA

UDA[17] 来自 Google，同样也是采用一致性正则。对于图像 UDA 针对图像采取一种高质量的数据增强方法 RandAugment，对于文本 UDA 使用的是回译和非核心词替换，用 TF-IDF 值来衡量一个词对于一段文本的重要性，来判断该词是否要被替换，再联合回译做替换。

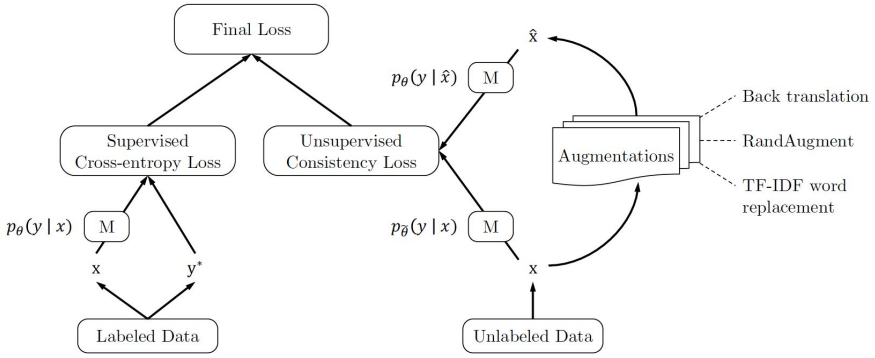


图 12　UDA 模型结构图

训练技巧：

- **Confidence-based masking**：通过阈值筛选一致性正则项。
- **Sharpening Predictions**：让预测的概率分布更加极端化。
- **Domain-relevance Data Filtering**：去掉领域不相关的数据，同样根据置信度来选择。

2.3 集成学习 + 自训练

在有监督学习算法中，目标是学习出一个稳定的且在各个方面表现都较好的模型，但实际情况往往不这么理想，有时只能得到多个有偏好的模型（弱监督模型，在某些方面表现的比较好）。集成学习就是组合这里的多个弱监督模型以期得到一个更好更全

面的强监督模型，集成学习潜在的思想是即便某一个弱分类器得到了错误的预测，其他的弱分类器也可以将错误纠正回来。如果待组合的各个模型之间差异性（Diversity）比较显著，那么 Ensemble 之后通常会有一个较好的结果。

自训练使用少量的标记数据和大量的未标记数据对模型进行联合训练，首先使用经过训练的分类器来预测所有未标记数据的标签，然后选择置信度较高的标签作为伪标注数据，将伪标记数据与人工标记的训练数据联合起来重新训练分类器。

- 集成学习：训练多个不同的模型，如 BERT 模型、Mixup 模型、半监督学习模型。
  - 分别用每个模型预测数据池 U（Unlabeled Data）的标签概率分布。
  - 计算标签概率分布的加权和，得到数据池 U 的概率分布 Soft Prediction。
- 自训练：训练一个模型用于组合其他各个模型。
  - Student 模型学习数据池 U 中高置信度样本的 Soft Prediction。
  - Student 模型作为最后的强学习器。

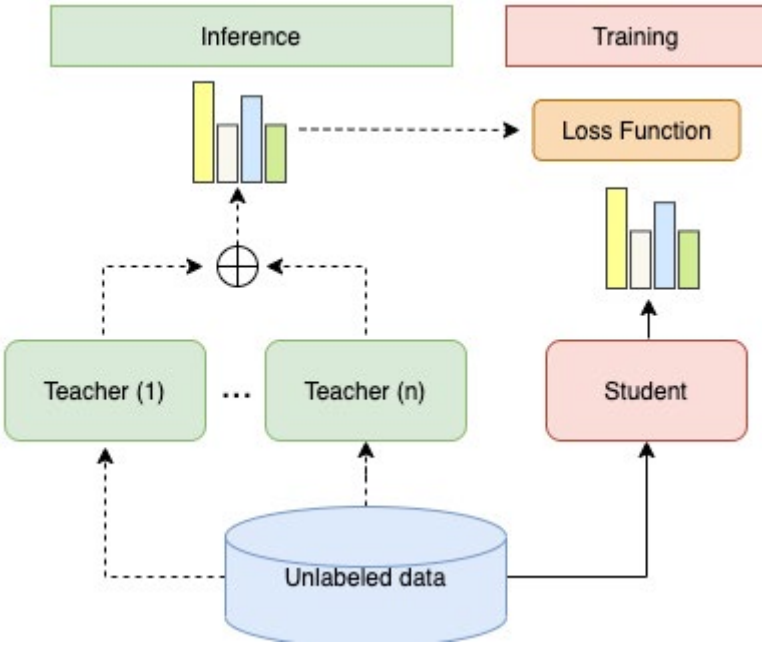


图 13　集成学习 + 自训练结构图

2.4 领域迁移

领域迁移主要解决少样本，多类别的问题。

MAML[18]：2017 年发表被广泛引用的方法，但是训练过程较为复杂，后面被 Meta-Baseline 方法在多个领域超越了。

Meta-Baseline[19]：2020 年发表的新基线方法，大幅超越前者，且方法简单，因此称为 New Baseline。Meta-Baseline 方法，通过在所有 Base Classes 上预先训练分类器，并在基于最近质心的少样本（Few-Shot）分类算法上进行元学习。

- Classifier-Baseline: 在所有的 Base Classes 上使用交叉熵损失预训练分类器，然后移除最后一层 FC，得到 Feature Encoder f_θ。
- Meta-Baseline
  - Pre-Training Stage：训练 Classifier-Baseline 模型。
  - Meta-Learning Stage：给定一个 Few-Shot Task（N-way k-shot），计算每个类别的特征平均值 $w_c = \frac{1}{|S_c|}\sum_{x \in s_c} f_\theta(x)$。
  - 针对于每个 Query-Set 中的样本用余弦相似度计算与每个类别的相似度，减少类内方差。

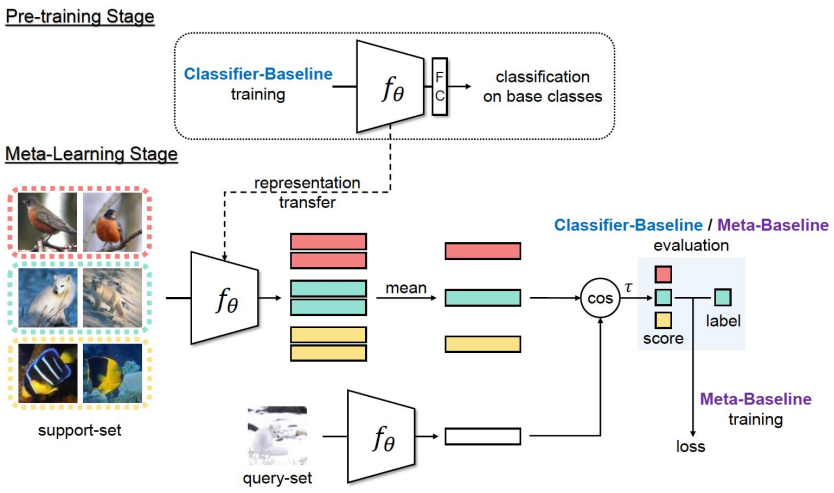


图 14　Meta-Learning Stage 结构图

2.5 主动学习

Active Learning（主动学习）通过一定的算法查询最有用的未标记样本，并交由专家进行标记，然后用查询到的样本训练分类模型来提高模型的精确度。模型抛出的未标注数据为“Hard Sample”，对于“Hard Sample”的不同定义可以衍生出一大堆的方法，如可以是 Ambiguous Sample，即模型最难区分的样本；可以是对模型提升（改变）最大的样本，如梯度提升最大；可以是方差减小等等，相比与有监督学习，主动学习通过让模型更多的关注或学习“Hard Sample”，以期在较少的训练样本下获得较好的模型。

- 查询策略是主动学习的核心之处，不确定性采样的查询是最常用的，不确定性采样的关键是描述样本的不确定性，通常有如下思路：
  - 置信度最低（Least Confident），如对于三分类的情况下，对两个样本预测的概率分别是 (0.8, 0.1, 0.1) 和 (0.51, 0.31, 0.18)，在这种情况下第二个数据更“难”被区分，因此更有标注价值。
  - 边缘采样（Margin Sampling），选择那些更容易被判定成两类的样本数据，即这些数据被判定成两类的概率相差不大，边缘采样选择模型预测最大和第二大的概率差值最小的样本。
  - 熵方法（Entropy），用熵来衡量一个系统的不确定性，熵越大表示系统的不确定性越大，熵越小表示系统的不确定性越小，可以选择那些熵比较大的样本数据作为待标注数据。
- 迭代思路：输入，初始少量标注数据 L_0、未标注数据池 U、深度学习模型 M。

  1. 标注数据集 $L \leftarrow L_0$。
  2. 用 L 训练模型 M，并对未标注数据池 U 进行预测。
  3. 用对应的查询策略选择 U 中需要标注的样本进行标注，并将其加入到标注数据集 L 中。
  4. 重复 2-3 过程，直到达到精度或者或者标注预算。

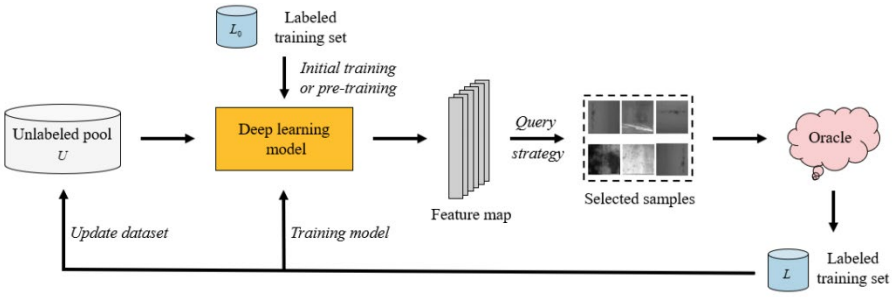


图 15　主动学习迭代过程

3. 应用实践

图 16 展示了基于 BERT[20] 的模型微调，包括句间关系任务和单句分类任务。

1. **单句分类任务**：BERT 模型预训练中的 NSP 任务使得 BERT 中的“[CLS]”位置的输出包含了整个句子对（句子）的信息，我们利用其在有标注的数据上微调模型，给出预测结果。常见单句分类任务如文本分类、情感分类等。
2. **句间关系任务**：对两个句子用“[SEP]”进行拼接，用“[CLS]”位置的输出判断两个句子之间的关系。常见句间关系任务如自然语言推理、语义相似度判断等。

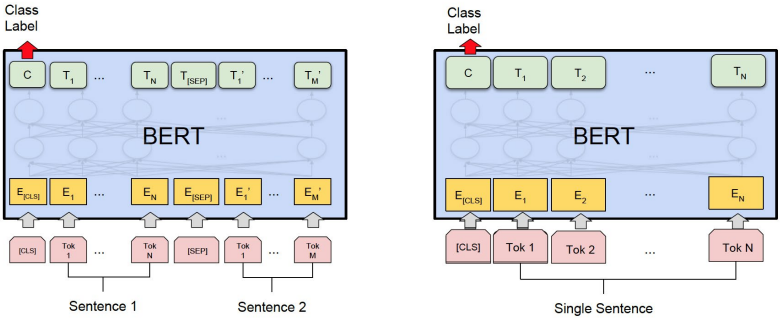


图 16　基于 BERT 的微调模型

3.1 实验结果

在上面我们提到了主要有以下两个评价标准，提升算法效果和节省标注数据，我们分

别在美团业务和通用 Benchmark 数据集上进行了微调实验。

3.1.1 小样本学习，提升算法效果

数据集	数据量	base	数据增强				半监督学习		集成学习+自训练
		base(%)	mixup(%)	AT(%)	R-Drop(%)	manifold-mixup(%)	temporal(%)	mixTemporal(%)	ensemble(%)
医美题材（单句分类）	500	83.73	84.27	83.73	83.87	82.66	**84.67**	**84.40**	84.67
	1000	86.42	**87.23**	86.96	**87.09**	86.29	86.15	85.21	87.09
	3000	88.03	89.11	88.97	**89.78**	88.17	88.57	89.24	**89.24**
POI打标情感分析（句间关系）	500	85.76	85.6	85.8	85.59	86.54	**86.74**	85.46	**86.91**
	1000	85.32	86.44	86.74	86.87	**87.11**	85.66	86.84	**87.28**
	3000	87.88	87.98	88.79	87.95	88.72	88.52	**89.09**	**88.32**

图 17　小样本学习在美团业务 Benchmark 上效果对比（加粗表示了得分最高的两个结果，下划线表示超出 Baseline 结果）

数据集	数据量	base	数据增强				半监督学习		集成学习+自训练
		base(%)	mixup(%)	AT(%)	R-Drop(%)	manifold-mixup(%)	temporal(%)	mixTemporal(%)	ensemble(%)
TNEWS 今日头条中文新闻（单句分类）	500	59.42	59.46	60.36	**60.31**	59.29	60.25	59.14	**60.81**
	1000	60.06	60.08	60.6	60.9	60.67	**61.03**	60.85	**61.63**
	3000	61.48	61.33	62.19	61.72	61.76	**62.69**	61.93	**62.84**
BQ 百度文本相似度（句间关系）	500	74.01	74.51	74.61	73.91	74.62	**75.01**	74.15	**74.93**
	1000	76.49	76.92	77.16	76.96	76.28	**77.71**	77.21	**77.45**
	3000	79.21	79.26	80.04	**80.66**	79.11	79.96	80.44	**81.36**
AFQMC 蚂蚁金融语义相似度（句间关系）	500	62.11	64.29	64.66	62.55	62.23	**67.3**	**67.07**	65.33
	1000	65.01	65.26	65.73	62.53	63.92	**67.4**	**67.44**	65.89
	3000	68.67	68.37	67.65	68.67	68.48	68.35	**70.04**	**69.94**
OCNLI 中文原版自然语言推理（句间关系）	500	53.66	54.41	**56.00**	54.98	48.30	50.71	50.71	**55.35**
	1000	59.22	59.89	60.00	60.4	58.84	58.33	**60.74**	**61.38**
	3000	65.49	66.2	66.4	66.16	**66.61**	66.37	65.89	**67.83**

图 18　小样本学习在中文通用 Benchmark 上效果对比（Dev 集结果，加粗表示了得分最高的两个结果，下划线表示超出 Baseline 结果）

图 17 和图 18 列出了四种 Embedding 增强，两种半监督学习，以及集成学习 + 自训练模型的结果。我们也尝试过上文提到的其他方法，如 UDA/MixText，这些方法依赖于外部翻译软件翻译的结果，效果不稳定。因此我们主要使用了以上模型，几种模型的比较如下：

- **数据增强**：四种 Embedding 增强结果中，对抗训练（AT）表现最为稳定，平均可提升模型 1pp。
- **半监督学习**：数据增强相对于半监督学习结果较为稳定，但半监督学习可以带来的提升较为明显，可提升模型 1.5pp−2pp，其中在 AFQMC 数据集上提升模型 2pp−4pp。
- **集成学习 + 自训练**：集成学习 + 自训练融合了单模型在未标注数据上的预测结果，基本可以达到或这接近最好的结果，平均可提升模型 1.5pp−2pp。

3.1.2 主动学习，减少训练样本

主动学习（Active Learning）选择置信度较低的样本用作下一批标注，可以减少重复 / 相似样本标注，降低标注成本。在新项目标注数据从无到有的时候，可以使用 active learning 选择启动样本。

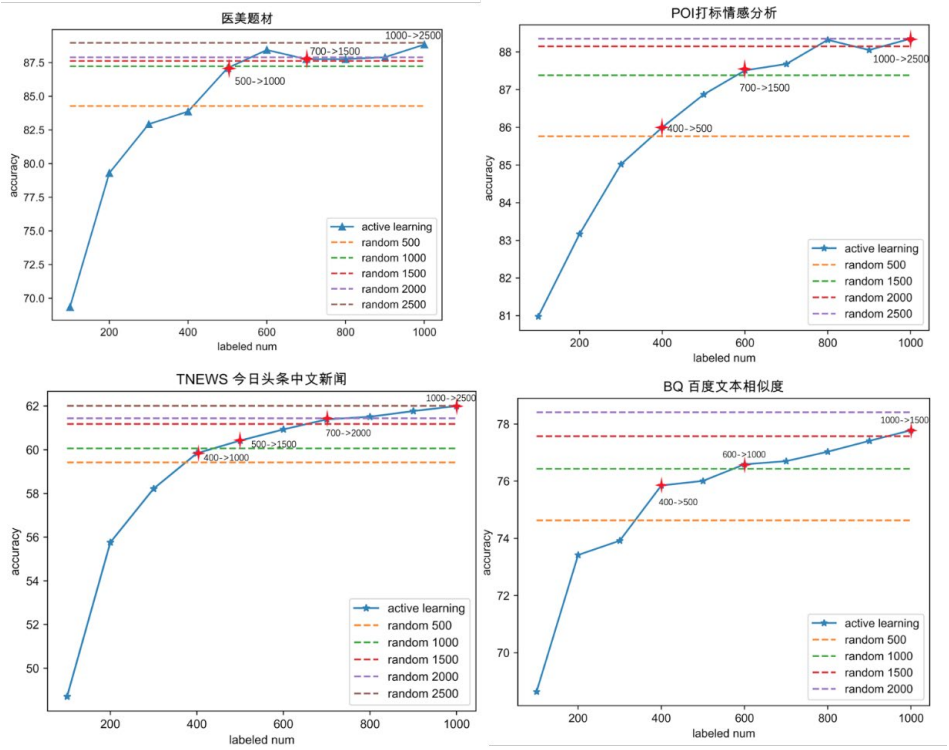


图 19　Active Learning 与随机选择样本在小样本情况下比较

在目前数据集下，平均 500 条 Active Learning 选择数据可以达到 1000 条随机样本结果，900 条数据可以接近 1500 条随机样本结果。在新项目标注数据从无到有的时候，可以使用 Active Learning 选择启动样本。

3.2 在美团业务中的应用

3.2.1 医美题材分类

对美团和点评的笔记内容按题材分为 8 类：猎奇、探店、测评、真人案例、治疗过程、避坑、效果对比、科普。用户点击某一种题材时，返回对应的笔记内容，上线至美团和点评 App 医疗美容频道的百科页、方案页经验分享。该任务是典型的文本分类任务，小样本学习利用 2,989 条训练数据准确率达到了 89.24%。

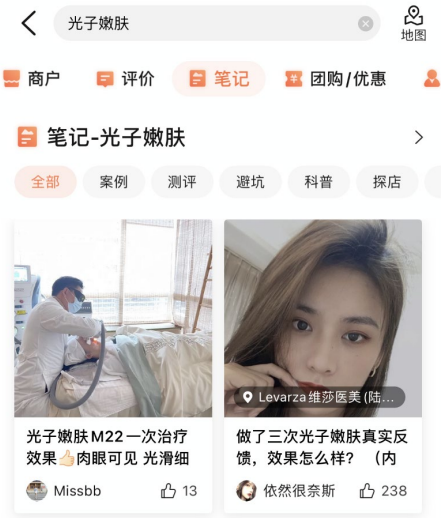


图 20　医美题材分类示例

3.2.2 攻略识别

从 UGC 和笔记中挖掘旅游攻略，提供旅游攻略的内容供给，应用于景点精搜下的攻略模块，召回内容为描述旅游攻略的笔记。该任务为二分类任务，小样本学习利用 384 条训练数据准确率达到了 87%。

图 21　攻略识别示例

3.2.3 医美功效打标

对美团和点评的笔记内容按功效进行召回，功效的类型有：补水、美白、瘦脸、除皱等，上线至医美频道页。该任务是句间关系任务，全量笔记内容为 104 万，有 110 种功效类型需要打标，小样本学习仅用 2909 条训练数据准确率达到了 91.88%。

图 22　医美功效打标示例

3.2.4 医美品牌打标

品牌上游企业有针对旗下产品进行品牌宣传和营销的诉求，而内容营销是当前主流、有效的营销方式之一。品牌打标就是为每种品牌如“伊肤泉”、“术唯可”召回详细介绍该品牌的笔记内容。先用品牌词进行匹配，再判断品牌词和匹配到的笔记内容的相关性，是详细描述该品牌，还是简单提及到品牌名字，上线至医美品牌馆。该任务是句间关系任务，有 103 种品牌，大类品牌 15 种每种标注 64 条数据，其余品牌每种标注 5–8 条数据，小样本学习仅用 1676 条训练数据准确率达到了 88.59%。

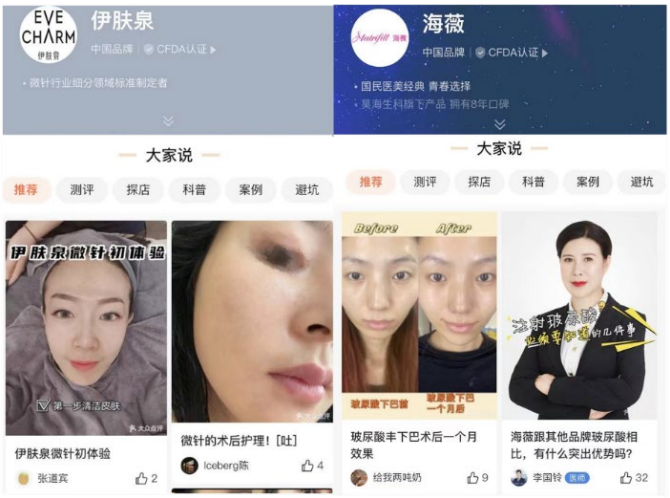


图 23　医美品牌打标示例

3.2.5 其他业务应用

- **医美评价真实性**：点评和美团医美业务如果存在虚假评论，会对体验有比较大的伤害，需要通过模型检测出来。通过 Active Learning 选取标注数据，数据增强、半监督学习和集成学习 + 自训练优化模型，最后业务方仅标注 1757 条数据，模型在检测感知虚假评价的准确率上就能达到 95.88%，超出业务预期。
- **POI 打标情感分析**：该任务为句间关系任务，判断 Query 和内容的情感导向（正面、负面、不确定和不相关）。已有模型在 1 万条数据上训练，通过小样本学习，在已有模型上提升模型精度 0.63pp。

- **学城（美团内部学习工具）文本分类**：该任务为文本分类任务，将文本分为 17 种类别。已有模型在 700 条数据上训练，通过小样本学习，在已有模型上提升模型精度 2.5pp。

4. 未来展望

1. **持续改进现有模型、探索更多模型**。目前的实验结果还有很大的改进空间，需要不断探索，改进模型；同时探索更多的领域迁移模型，并应用到业务中，达成业务方可以用最少的数据训练最好的结果。
2. **在更多任务类型上进行实验**。目前主要是单句分类、句间分类等任务类型上进行实验，需要进一步 MRC 模型、命名实体识别模型。
3. **深入探索领域迁移，训练通用模型**。目前，我们对接的业务较多，因此也积累了不少领域的文本分类和句间关系数据集，希望能够训练一个通用模型用于该领域的各项任务，从而在一个新的业务中使用更少量的数据就能达到很好的业务效果。如可以通过 Facebook 的 EFL[21] 模型，将该领域下的文本分类任务和句间关系任务都重新表述为文本蕴含任务训练一个通用模型，可以直接在新的业务上进行迁移。
4. **建设小样本学习平台**。目前正在将小样本学习能力集成到公司统一的 BERT 平台，开放给公司各业务方灵活使用。后续，在对小样本学习进行了更加深入的探索之后，我们会尝试建立单独的小样本学习平台，提供更多的低资源学习能力。

参考文献

[1] Wei J, Zou K. Eda: Easy data augmentation techniques for boosting performance on text classification tasks[J]. arXiv preprint arXiv:1901.11196, 2019.
[2] Kobayashi S. Contextual augmentation: Data augmentation by words with paradigmatic relations[J]. arXiv preprint arXiv:1805.06201, 2018.
[3] Anaby-Tavor A, Carmeli B, Goldbraich E, et al. Not Enough Data? Deep Learning to the Rescue![J]. arXiv preprint arXiv:1911.03118, 2019.
[4] Wei J, Huang C, Vosoughi S, et al. Few-Shot Text Classification with Triplet Networks, Data Augmentation, and Curriculum Learning[J]. arXiv preprint

arXiv:2103.07552, 2021.
[5] Andreas J. Good-enough compositional data augmentation[J]. arXiv preprint arXiv:1904.09545, 2019.
[6] Zhang H, Cisse M, Dauphin Y N, et al. Mixup: Beyond empirical risk minimization[J]. arXiv preprint arXiv:1710.09412, 2017.
[7] Guo D, Kim Y, Rush A M. Sequence-level mixed sample data augmentation[J]. arXiv preprint arXiv:2011.09039, 2020.
[8] Zhang R, Yu Y, Zhang C. Seqmix: Augmenting active sequence labeling via sequence mixup[J]. arXiv preprint arXiv:2010.02322, 2020.
[9] Verma V, Lamb A, Beckham C, et al. Manifold mixup: Better representations by interpolating hidden states[C]//International Conference on Machine Learning. PMLR, 2019: 6438-6447.
[10] Miyato T, Dai A M, Goodfellow I. Adversarial training methods for semi-supervised text classification[J]. arXiv preprint arXiv:1605.07725, 2016.
[11] Liang X, Wu L, Li J, et al. R-Drop: Regularized Dropout for Neural Networks[J]. arXiv preprint arXiv:2106.14448, 2021.
[12] Laine S, Aila T. Temporal ensembling for semi-supervised learning[J]. arXiv preprint arXiv:1610.02242, 2016.
[13] Tarvainen A, Valpola H. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results[J]. arXiv preprint arXiv:1703.01780, 2017.
[14] Miyato T, Maeda S, Koyama M, et al. Virtual adversarial training: a regularization method for supervised and semi-supervised learning[J]. IEEE transactions on pattern analysis and machine intelligence, 2018, 41(8): 1979-1993.
[15] Berthelot D, Carlini N, Goodfellow I, et al. Mixmatch: A holistic approach to semi-supervised learning[J]. arXiv preprint arXiv:1905.02249, 2019.
[16] Chen J, Yang Z, Yang D. Mixtext: Linguistically-informed interpolation of hidden space for semi-supervised text classification[J]. arXiv preprint arXiv:2004.12239, 2020.
[17] Xie Q, Dai Z, Hovy E, et al. Unsupervised data augmentation for consistency training[J]. arXiv preprint arXiv:1904.12848, 2019.
[18] Finn C, Abbeel P, Levine S. Model-agnostic meta-learning for fast adaptation of deep networks[C]//International Conference on Machine Learning. PMLR, 2017: 1126-1135.
[19] Chen Y, Wang X, Liu Z, et al. A new meta-baseline for few-shot learning[J]. arXiv preprint arXiv:2003.04390, 2020.
[20] Devlin J, Chang M W, Lee K, et al. Bert: Pre-training of deep bidirectional transformers for language understanding[J]. arXiv preprint arXiv:1810.04805, 2018.
[21] Wang S, Fang H, Khabsa M, et al. Entailment as Few-Shot Learner[J]. arXiv preprint arXiv:2104.14690, 2021.

作者简介

骆颖、徐俊、谢睿、武威等，均来自美团搜索与 NLP 部 /NLP 中心。

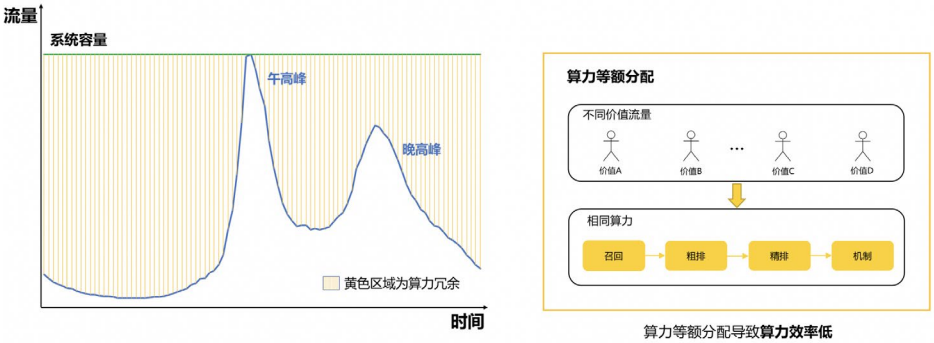
美团外卖广告智能算力的探索与实践

作者：顺辉　家宏　宋伟　国梁　乾龙　乐彬

1. 业务背景

目前，美团外卖日订单量已突破 4000 万，成为美团最为重要的业务之一。外卖广告服务也从最初的支撑单一业务线发展到现在的十余条业务线，服务整体承受的流量日益增长，所消耗的机器资源也达到了一定的规模。

在外卖场景下，流量呈现明显的双峰结构，即午餐和晚餐两个时段为流量波峰，其余时段流量较小。在这种流量特点下，外卖广告服务在高峰时段面临较大的性能压力，在非高峰时段存在大量算力冗余，从全局来看，机器算力分配效率低，流量价值仍然具有较大的挖掘空间。一方面，流量消耗的算力没有根据流量价值进行动态的分配，造成系统在高价值流量上算力分配不足，价值没有得到充分的挖掘，而在低价值流量上，却存在浪费大量算力的现象；另一方面，在非高峰时段，系统流量较低，导致系统的整体资源利用率偏低，限制了系统获取更高的业务收益。



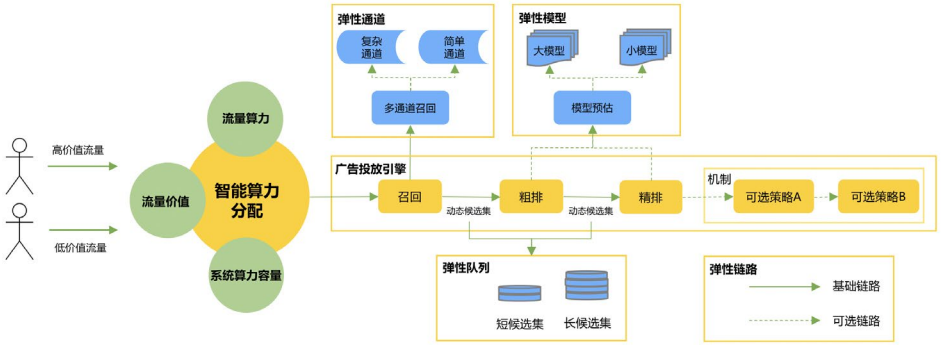
算力等额分配导致**算力效率低**

因此，算力需要进行更合理的分配来得到更加高效的利用。当前业界对于动态算力分配的研究较少，以阿里巴巴定向广告平台的 DCAF[1] 为代表，该方案根据流量价值进行差异化的算力分配，对不同价值的流量分配不同的候选队列长度，实现有限资源约束下的收益最大化。DCAF 提供了一种优秀的解决方案，但该方案在外卖广告场景下

存在一定的局限性。

针对外卖广告场景，外卖广告技术团队在 DCAF 方案的基础上进行了一系列探索和改进，首次进行了队列弹性分配和模型弹性分配相结合的尝试，并取得了不错的收益。一方面，在机器资源持平的情况下，CPM 可以提升 2.3%；另一方面，在业务收益持平的情况下，机器资源可以减少 40%，最后我们在外卖列表广告精排阶段推全了机器资源持平的方案。

2. 整体思路



在外卖广告投放引擎中，为应对极大的在线流量压力和庞大的候选集，我们将整个检索过程设计成候选集依次递减的漏斗型级联架构，主要包含召回、粗排、精排、机制等模块。

实现智能算力的整体思路，是在系统算力容量约束下，对不同价值流量进行差异化算力分配，从而提高在广告检索过程中的算力分配效率，实现收益的最大化。智能算力主要包括以下四个要素：

1. 流量价值量化：流量价值指流量为平台、广告商家、用户三方带来的收益，系统需要具备量化流量价值的能力。

2. 流量算力量化：流量算力指流量在系统内消耗的机器资源，在外卖广告场景下，流量消耗的算力和候选集大小、召回通道数量、模型大小、链路复杂度等系统变量紧密相关，同样系统需要具备量化流量算力的能力。

3. 系统算力容量量化：系统算力容量指系统的机器资源总和，和流量算力量化的量纲一致，通常可以通过压测等手段获取系统的算力容量；在系统算力分配过程中，需要保证总体流量算力消耗不超过系统的算力容量。

4. 智能算力分配：基于以上三个要素，在广告投放引擎全链路进行智能算力分配，我们把算力分配的手段定义为“**弹性动作**”，在外卖广告场景下，我们主要归纳了以下四种动作：

- **弹性队列**：线上检索是一个漏斗的过程，不同价值流量可以在级联漏斗的各模块中分配不同候选队列长度。
- **弹性模型**：在模型预估服务中，不同价值流量可以分配不同模型，大模型相对小模型预估效果更好的同时消耗的算力也更多。
- **弹性通道**：在多通道召回中，不同价值流量可以分配不同的召回通道。
- **弹性链路**：在检索链路上，不同价值流量可以分配不同复杂度的检索链路。 这些弹性动作的可选范围，定义为“**弹性档位**”，如队列长度 100 和 200 对应弹性队列的两个不同档位。**在智能算力下，算力的分配过程即弹性动作和弹性档位的智能决策过程**。

挑战分析

为了让智能算力在外卖广告场景下落地，我们主要面临着如下的挑战：

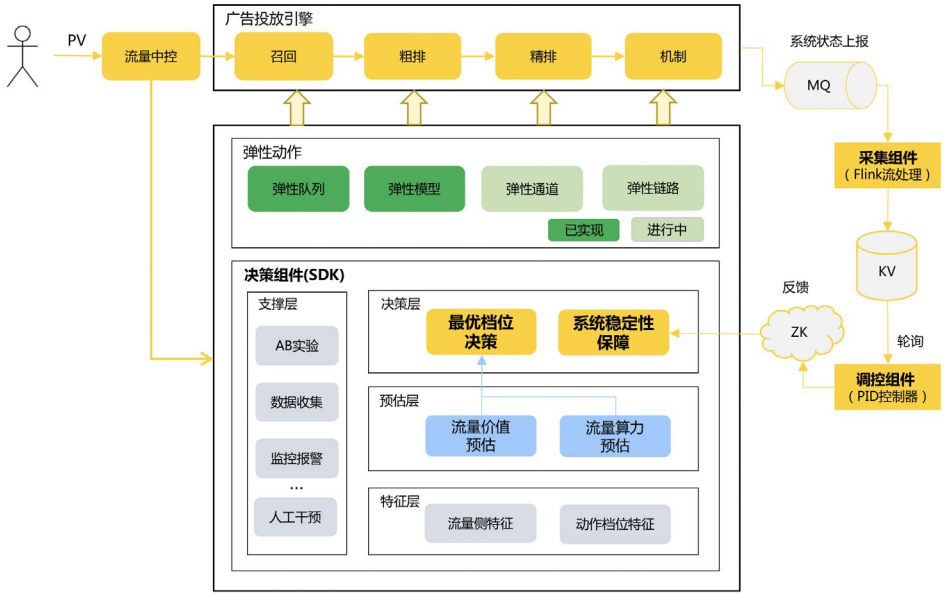
- **问题求解**
  - **挑战点**：智能算力的目标是最优化算力资源分配，这就要求我们对“在系统算力约束下，最大化流量收益”这个问题进行求解。
  - **应对思路**：参考已有方案，将问题拆解为流量价值预估、流量算力预估以及算力分配三个子问题进行求解，并针对外卖广告场景进行探索和改进。
- **系统稳定性保障**
  - **挑战点**：将系统算力分配托管给智能算力框架，从等额算力分配到智能算力分配，不仅需要保证智能算力框架本身的稳定性，更需要保证系统全链路的

平稳运行。

- **应对思路**：除了监控报警、熔断降级等常规保障手段，我们实现了基于系统状态的实时调控功能，保障系统稳定性。

- **通用性 & 扩展性**
  - **挑战点**：兼顾基础能力的复用和个性化能力的扩展，支撑外卖广告推荐和搜索两大方向，多个业务场景的接入。
  - **应对思路**：核心组件以 SDK 形式提供可复用、可扩展能力，同时基于通用的价值评估指标、算力评估指标和智能算力框架，支撑多弹性动作的组合决策和多业务场景的高效接入。

3. 方案设计



经过工程团队和算法团队深度 Co-Design 后，我们设计了一套多动作组合决策的智能算力框架。整个框架由决策组件、采集组件和调控组件组成，其中决策组件作为智能算力框架的核心，以 SDK 形式嵌入应用服务，提供可复用可扩展的多动作组合**最优档位决策能力**和**系统稳定性保障能力**，为广告投放引擎的各阶段赋能；采集组件和

调控组件为系统稳定性保障提供支撑。下面主要对最优档位决策和系统稳定性保障两个模块进行详细介绍。

3.1 最优档位决策

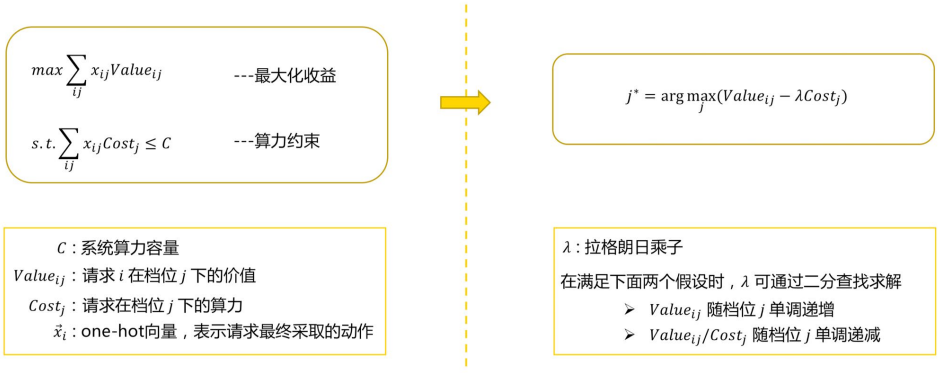
在业内已有弹性队列求解方案的基础上，我们进行一系列探索和改进：

- 通过选用更加通用的流量算力评估指标，同时增加流量算力预估模块，保证量化指标通用的同时提高精确度，解决**算力量化不通用和不精确**的问题。
- 通过构建特殊档位，首次尝试了**弹性队列和弹性模型的结合**，解决了**部分流量无法建模**的问题。

基于以上策略，我们实现了多弹性动作组合最优化档位决策。

3.1.1 问题建模

现有方案

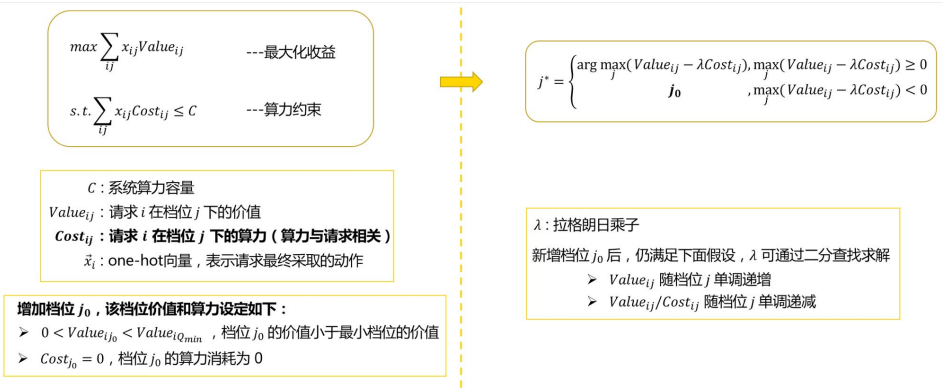


DCAF[1] 将该问题转化为对应的对偶问题进行求解，得到决策公式，实现弹性队列分配。

方案改进

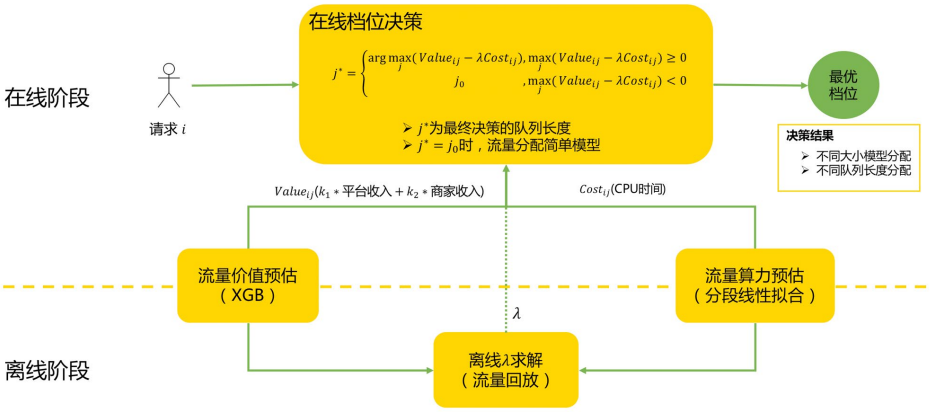
上述建模方案在外卖广告场景中存在以下问题：

- **算力量化问题**
  - **问题描述**：在原论文中，算力仅与队列长度相关。实际上，外卖广告系统中，相同长度的不同流量算力消耗可能不同（比如不同来源的流量会走不同的模型和链路），仅用队列长度无法准确量化算力。
  - **解决方案**：为了建模流量的算力消耗，同时拟合流量侧特征，我们将算力由$Cost_j$拓展为$Cost_{ij}$，并增加流量算力预估模块。
- **低价值流量建模问题**
  - **问题描述**：推导中要求对任意请求满足$max_j(Value_{ij} - \lambda Cost_{ij}) \geq 0$，此时，一些价值低、算力消耗大的流量（$max_j(Value_{ij} - \lambda Cost_{ij}) < 0$）将无法被建模。另外，在外卖广告场景中，由于返回队列必须满足填充率等业务要求，队列长度不能随意选取，档位空间必须满足$Q_{min} \leq j \leq Q_{max}$（$Q_{min}$和$Q_{max}$分别表示当前请求的档位上下界，且$Q_{min}$的值一般为最小填充广告数），这进一步导致了更多的流量不满足约束。
  - **解决方案（弹性模型）**：为了覆盖外卖广告场景中的所有流量，我们增加档位$j = j_0$，在实际落地中该档位对应为将请求分配给简单模型；相对复杂模型，流量分配简单模型后，算力消耗趋近于0（简单模型可以是词表或LR模型等）。



如图所示，由于档位 $j=j0$ 的算力和价值已知，无需对不同模型的价值和算力进行预估，后续的流量价值预估和流量算力预估工作都面向弹性队列即可。

3.1.2 决策框架



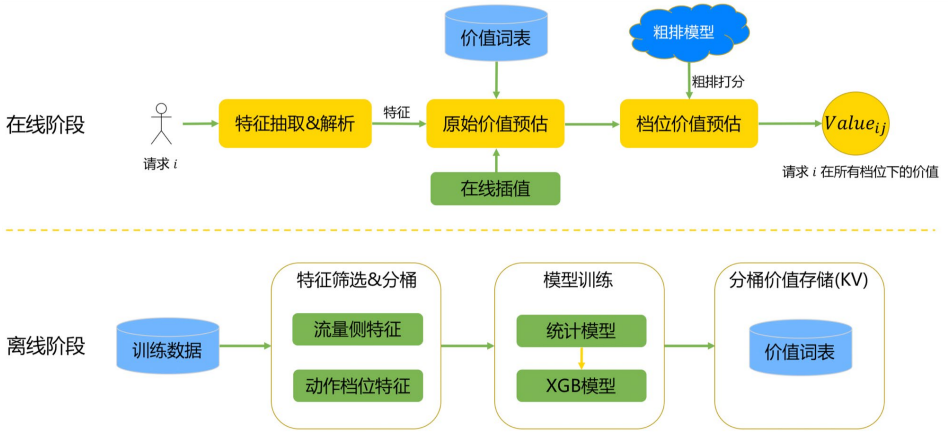
如图所示，最优档位决策模块分为离线和在线两个阶段，包含如下四个子模块：

- **流量价值预估模块**（离线 + 在线）：预估流量在不同档位下的价值。
- **流量算力预估模块**（离线 + 在线）：预估流量在不同档位下的算力。
- **离线 λ 求解模块**（离线）：通过回放历史流量，利用二分查找算法求解最优 λ。
- **在线决策模块**（在线）：对线上流量，基于档位决策公式计算最优档位，并根据计算结果分配不同模型和队列长度。

3.1.3 流量价值预估

流量价值预估是智能算力决策的核心，要求具备一定准确性。而线上进行模型预估会增加检索链路耗时，我们采用离线 XGB 模型预估 + 在线查找词表的方案，既保证了预估的准确性，又足够轻量。

价值评估指标选取：一般来说，流量价值指当前流量为广告平台带来的收入；在外卖广告场景中，我们关注平台收入的同时，也关注商家的收入，因此我们的流量预估的指标选定为 k_1* 平台收入 +k_2* 商家收入。



价值评估指标：**平台收入+商家收入**

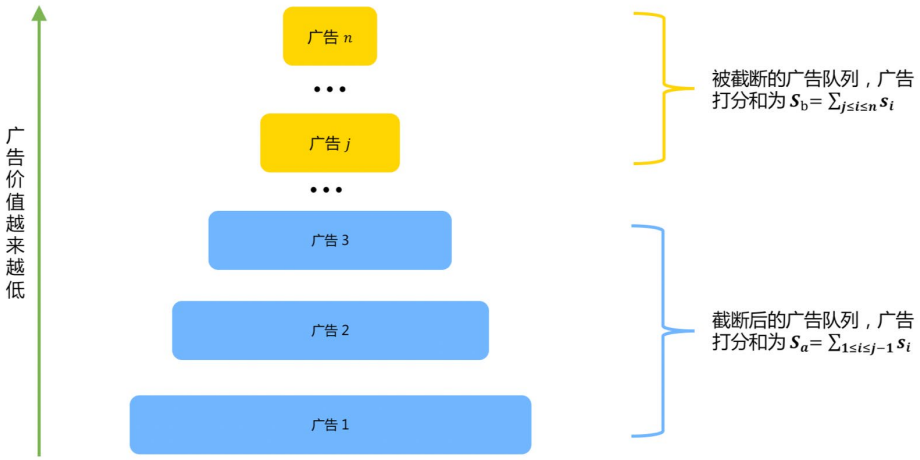
如图所示，流量价值预估模块包含离线和在线两个阶段。

离线阶段：

- **特征筛选 & 分桶**：基于离线特征重要性分析和分布情况，进行特征筛选和分桶。
- **模型训练**
  - 主要问题：初期我们采用统计方案，在特征分桶较多的情况下，数据稀疏问题严重，且队列长度越大，数据越稀疏。
  - 解决方案：采用 XGB 模型代替统计方案，增强泛化能力。
- 分桶价值存储：不同特征分桶的价值预估结果以 KV 结构写入词表。

在线阶段：

- **特征抽取 & 解析**：进行特征抽取、解析，按照离线分桶规则生成 Key 值。
- **原始价值预估**：根据 Key 值查找对应分桶的价值，然后通过线性插值计算原始队列长度下请求的价值。
- **档位价值预估**：如下图所示，借助粗排打分，通过计算不同档位下价值的衰减情况，实现不同档位的价值预估。



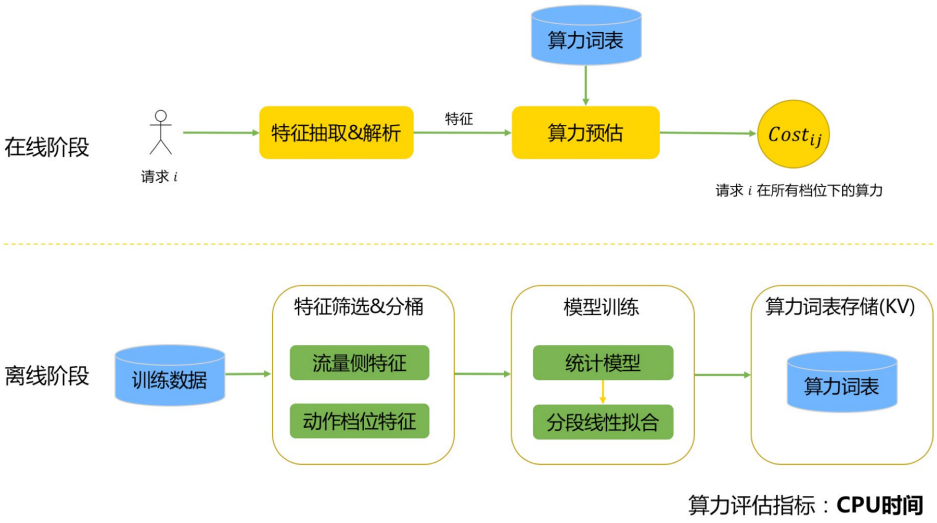
假设广告队列价值为原始流量价值为 V_0 ,则截断后价值为 $V = \frac{S_a}{S_a + S_b} V_0$

3.1.4 流量算力预估

业内智能算力的落地以弹性队列为主，一般采用队列长度作为流量算力评估指标，而队列长度作为流量算力评估指标面临以下两个问题：

- **通用性问题**：在弹性模型、弹性通道和弹性链路中流量算力消耗不由队列长度唯一确定，比如不同来源的流量可能走不同模型（不同模型算力消耗可能不同）。
- **精确度问题**：外卖广告场景下，即使对于弹性队列动作，流量算力消耗和队列长度也不是简单的线性关系。

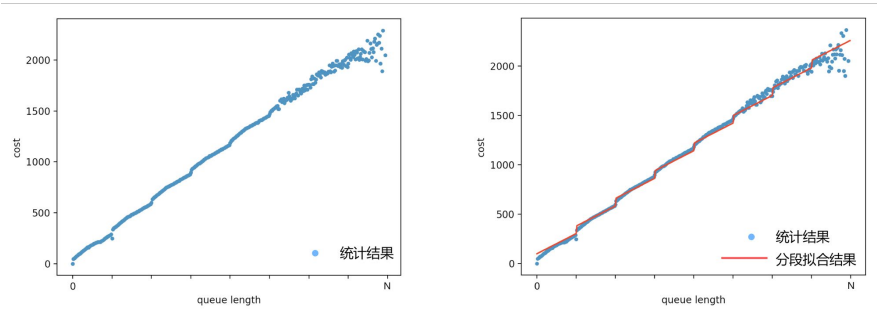
算力评估指标选取：为了解决以上问题，我们采用流量消耗的 CPU 时间作为流量算力评估指标。



如图所示，流量算力预估包括离线和在线两个阶段。

离线阶段：

- **特征筛选 & 分桶**：基于离线特征重要性分析和分布情况，进行特征筛选和分桶。
- **模型训练**
  - 训练过程：首先将样本划分到不同特征分桶中（同一分桶中队列长度不同，其他特征相同），然后分别对不同分桶拟合算力和队列长度的关系。
  - 主要问题：由于数据分布不平均，队列长度大于一定阈值后，由于数据稀疏，分桶内算力统计值开始出现波动，不利于线上决策。
  - 为了解决数据稀疏的问题，并拟合真实业务中的**拆包现象**，我们使用了**分段线性拟合**方案，将算力和队列长度之间的关系拟合为分段线性函数（下图是某个特征分桶内算力和队列长度关系拟合结果）。



拟合前

➢ 队列越长，数据越稀疏

拟合后

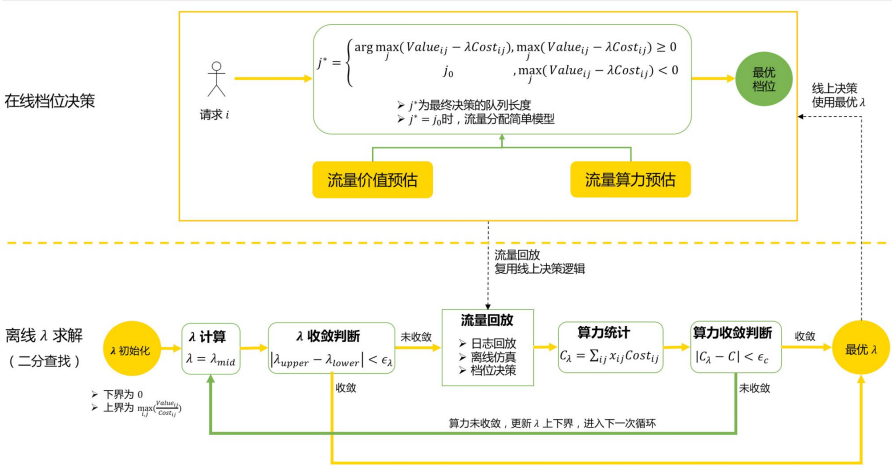
➢ 从短队列泛化到长队列

- **算力词表存储**：不同特征分桶的算力预估结果以 KV 结构写入词表。

在线阶段：

- **特征抽取 & 解析**：进行特征抽取、解析，按照离线分桶规则生成 Key 值。
- **算力预估**：根据 Key 值查找对应分桶的算力。

3.1.5 档位决策



1. 离线 λ 求解

基于价值预估和算力预估模块，通过回放历史流量，利用二分查找算法求解最优 λ。

离线 λ 求解的核心步骤是流量回放：通过重放历史同时段流量，复用线上逻辑，为每个请求模拟决策当前 λ 下的最优档位。

主要问题及解决方案

- 问题描述：离线流量大，将流量划分为多个时间片后，λ 的求解过程需要进行多次流量回放，仿真复杂度高。
- 解决方案：通过流量采样、多个时间片并行化求解等方案，快速求解最优 λ 。

2. 在线档位决策

首先根据业务场景定义候选档位集合（不包括j_0），然后遍历当前请求的候选档位，计算每个档位的收益$Value_{ij} - \lambda Cost_{ij}$，最后按照如下规则进行决策。

最优档位计算：

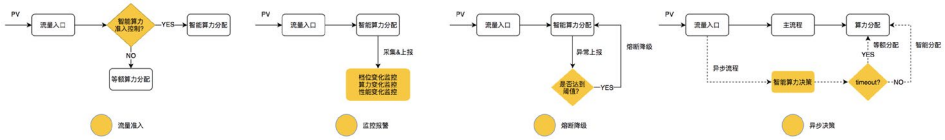
$$j^* = \begin{cases} arg\,max_j(Value_{ij} - \lambda Cost_{ij}), max_j(Value_{ij} - \lambda Cost_{ij}) \geq 0 \\ j_0, \qquad\qquad\qquad\qquad\qquad\quad max_j(Value_{ij} - \lambda Cost_{ij}) < 0 \end{cases}$$

根据档位执行算力分配动作：

- 当$max_j(Value_{ij} - \lambda Cost_{ij}) \geq 0$，即最大档位对应的收益 ≥ 0时，选择收益最大化档位。
- 当$max_j(Value_{ij} - \lambda Cost_{ij}) < 0$，即最大档位对应的收益 < 0时，流量分配简单模型。

3.2 系统稳定性保障

在智能算力下，系统从等额算力分配转换成动态算力分配，为保障系统服务稳定性，我们提供了熔断降级等常规措施，还实现了基于系统状态的 PID 实时调控功能。



- **流量准入**：提供分广告位、城市、时段等不同流量准入规则，控制智能算力流量。
- **监控报警**：提供 PV 和系统整体两个维度的档位、性能、算力等变化监控和阈值报警。
- **熔断降级**：实时监控智能算力的异常，达到配置的异常阈值后，智能算力会自

动熔断降级，进行等额算力分配。

- **异步决策**：为保障主流程整体链路耗时不增长，智能算力决策是一个异步的过程，超时后进行等额算力分配。

3.2.2 基于系统状态的 PID 实时调控

PID（Proportion Integration Differentiation）是一个通过比例、积分、微分进行控制的主流控制算法，我们通过对系统状态的实时监控和变化感知，基于 PID 算法对系统状态进行实时调控，保证系统状态的稳定。

系统状态通常可以用系统的 CPU/GPU 利用率、QPS、RT（Avg、TP99、TP999 等）、调用失败率（FailRate）等指标衡量。

调控目标

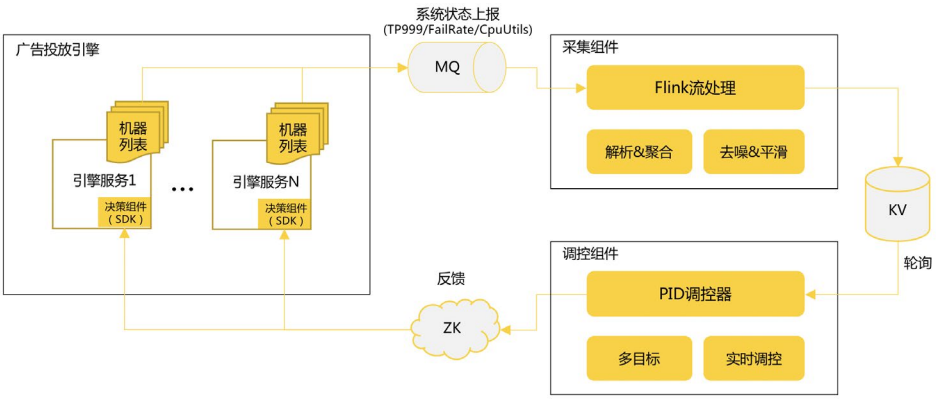
调控目标应该选取最能够快速反映出系统状态变化的指标，基于这个原则，我们选取了 TP999、FailRate、CpuUtils 作为调控目标。

调控策略

基于 PID 调控器，支持多种调控策略：

- **调控动作档位**：对于弹性队列，可以调控队列长度的系数或者上限，从而调节候选队列长度。
- **调控决策公式**：可以通过调控决策公式的 λ 系数，从而调节系统整体的算力消耗。

调控过程



- **系统状态上报**：投放引擎服务通过监控系统，实时反馈系统当前状态，同步到 MQ，其中核心指标具有 10s 的时效性。
- **采集组件**：基于 Flink 流处理框架，实时解析聚合系统状态数据，并进行去噪和平滑处理，处理后的数据写到 KV 存储。
- **调控组件**：基于 PID 算法，通过轮询感知系统状态变化，根据选择的调控目标，实时调控系统算力，将调控结果反馈到投放引擎的决策组件，形成调控闭环。

智能算力接入 PID 实时调控后，在系统负载偏高时，可以快速稳定有效的反馈调节，将系统性能保持在目标水位。

4. 实验

4.1 实验设置

- **系统算力容量的选取**：为了保证线上系统能根据实时流量快速调整，结合外卖的流量特点，我们选择 15min 作为最小的调控单元；实际场景中，我们选取过去多天内流量最大且性能稳定的高峰期时间片，并统计该时间片内消耗的总 CPU 时间作为系统算力容量 C。
- **Baseline 选取**：选取线上无智能算力的流量作为对照组。
- **流量价值预估**：选取最近多天流量作为训练数据，使用 XGB 模型进行预估并将结果写入词表。

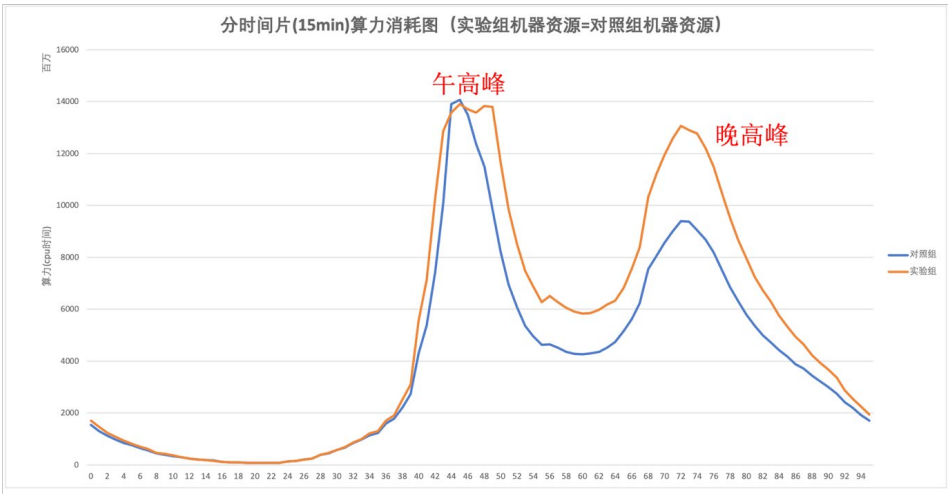
- **流量算力预估**：选取最近多天流量作为训练数据，将算力随队列长度的变化拟合为分段线性函数，并将最终预估结果写入词表。
- **离线 λ 求解**：外卖场景中，同环比流量变化趋势基本一致，我们通过重放昨日流量，离线计算当日每个时间片内（15 分钟为一个时间片）最优 λ 并存储为词表。
- **实验思路**：离线仿真时使用的系统算力容量 C 可控制线上算力的消耗，故而可通过调节离线求解使用的系统算力容量 C 实现机器资源持平或业务收益持平的实验。

4.2 实验一：机器资源持平，提高业务收益

	CPM	ROI	CTR	CVR	机器资源
Baseline（系统算力容量 =C）	+0.00%	+0.00%	+0.00%	+0.00%	+0.00%
智能算力（系统算力容量 =C）	**+2.36%**	−1.40%	+0.94%	+0.09%	+0.46%

收益来源分析

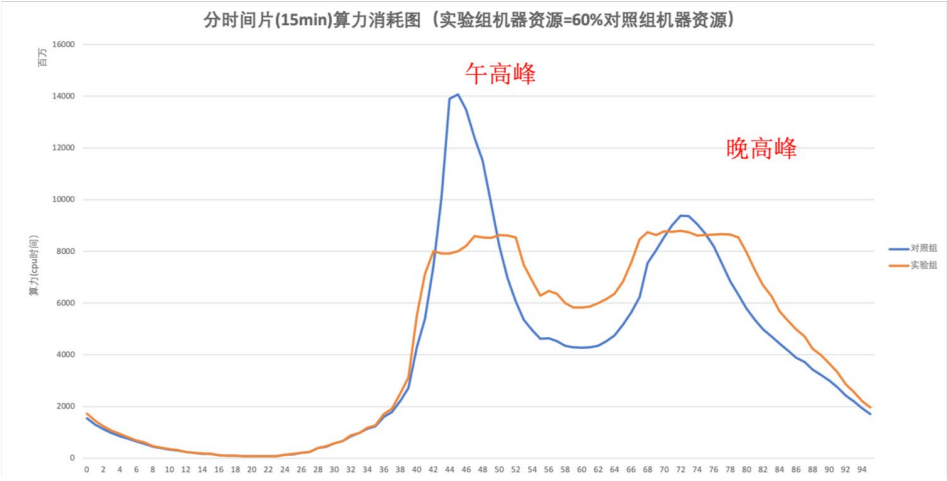
- 流量高峰时段，保证系统稳定的同时，通过差异化算力分配提升业务收益。
- 非流量高峰时段，提高系统资源利用率，将空闲机器资源转化为业务收益。



4.3 实验二：业务收益持平，减少机器资源

	CPM	ROI	CTR	CVR	机器资源
Baseline（系统算力容量 =C）	+0.00%	+0.00%	+0.00%	+0.00%	+0.00%
智能算力（系统算力容量 =60% * C）	+0.70%	−1.22%	+0.15%	+0.85%	**−40.8%**

收益来源分析：– 通过压制午高峰和晚高峰算力消耗，达到减少机器资源的目标。从下图可以看出，在高峰时段，实验组的机器资源消耗约占对照组的 60%。 – 同时，在高峰时段进行差异化算力分配，在非高峰时段提高资源利用率，填平整体业务收益。



5. 总结与展望

这篇文章主要从最优档位决策和系统稳定性保障两个方面，介绍了智能算力在外卖广告从 0 到 1 建设过程中的思考和优化思路。

未来在算法策略上，我们将尝试基于进化算法和强化学习算法，对系统全链路组合下的算力最优分配问题进行建模和求解；在引擎架构上，对系统仿真能力、在线决策能力和稳定性保障能力进行持续优化，同时尝试和公司弹性伸缩系统相结合，发挥智能算力的更大价值。

6. 参考文献

[1] Jiang, B., Zhang, P., Chen, R., Luo, X., Yang, Y., Wang, G., … & Gai, K. (2020). DCAF: A Dynamic Computation Allocation Framework for Online Serving System. arXiv preprint arXiv:2006.09684.

7. 作者简介

顺辉、家宏、宋伟、国梁、乾龙、乐彬等，均来自美团外卖广告技术团队。

8. 招聘信息

美团外卖广告技术团队大量岗位持续招聘中，诚招广告后台 / 算法开发工程师及专家，坐标北京。欢迎感兴趣的同学加入我们。可投简历至：maoshunhui@meituan.com（邮件主题请注明：美团外卖广告技术团队）

外卖套餐搭配的探索和应用

作者：瑞玉　文斌　杨林　懋地

1. 背景

让用户更方便快捷地选购到满意的外卖商品，是美团外卖一直在努力的方向。本文主要介绍了针对美食类商家的套餐搭配技术以及应用实践。在外卖点餐的选择过程中，用户一般会考虑单品偏好、组合搭配等因素，挑选商家和商品的过程耗时较长。我们通过套餐搭配技术，基于商家的候选商品，自动搭配优质套餐，轻松解决用户“选择困难症”，能够提升用户的决策效率。

2. 业务目标和挑战

2.1 业务目标

目前，美团外卖 App 的套餐搭配应用有很多，有“今日套餐推荐”、“满减神器”、“套餐搭配推荐”等。由于目前外卖商家自行搭配套餐的能力较弱而且意愿不强，导致外卖套餐底层供给对业务场景和商家的覆盖率偏低，不能满足套餐相关推荐排序应用的需求。因此，外卖套餐搭配的业务目标是对于美食类商家搭配出候选的套餐组合，给套餐相关应用方提供更丰富的套餐供给。

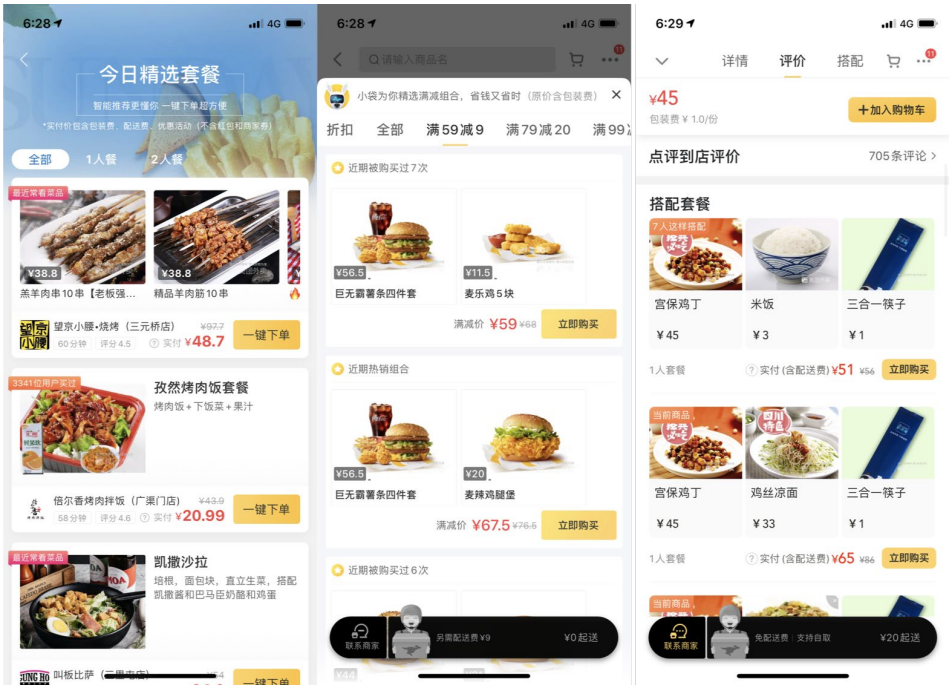


图 1 “套餐推荐”、“满减神器套餐推荐”、“菜品详情页套餐搭配”应用示例

对于套餐相关应用方，我们进行了业务分析：“今日推荐”、“满减神器”等业务的搭配条件相对较弱且可以离线获取搭配条件，归类为推荐型相关业务，此类业务需要保证商家的套餐覆盖率提升，从而保证商家的推荐露出。详情页、满减加购等业务的搭配条件较强，而且偏实时。例如，详情页是用户指定一个菜品后进行搭配，满减加购场景是用户选择一个菜品和特定的价格区间作为条件。这些属于搭配型业务，此类业务需要保证套餐对实时场景的覆盖，从而保证套餐搭配 Tab 的露出。套餐搭配算法的目标有：①提升套餐组合的覆盖度，从而给下游的套餐相关应用提供场景覆盖率高、多样性充足的套餐组合。②保证搭配套餐的质量。

2.2 业务挑战

商品搭配在电商场景下也存在较多的应用，例如淘宝的购物车搭配、服装搭配，化妆品搭配。购物车搭配是基于用户购物车、已购买商品的打包推荐，例如，用户在加购牙刷后，可以给出牙膏的推荐。这类方法主要基于商品的购买行为进行相关性的推

荐，目标不是形成一个完成的搭配组合。但是，外卖美食商品搭配则需要考虑整个组合的合理性，而不单纯基于商品是否相关。比如：大量订单中有“小炒肉 + 番茄鸡蛋汤 + 米饭”、“鱼香肉丝 + 番茄鸡蛋汤 + 米饭”等组合方式，但是“番茄鸡蛋汤 + 米饭”并不构成一个好的套餐搭配。

服装搭配和化妆品搭配为面向组合的搭配推荐，对此类搭配问题的解决方案大体分成两类，一类是：搭配模式用于模型挑选商品过程的剪枝，搭配模式可以是人工或者模型的方式先验给出，参考文献中的论文 4、5 采用此思路，该方法的特点是搭配效果靠剪枝策略 + 质量评估模型共同保证。另一类是通过端到端的网络参数学习搭配模式思路，论文 6 和我们的离线套餐搭配采用此思路，该方案的特点是搭配效果更依赖于端到端模型保证，但同时搭配模型更复杂。

相比电商场景的商品搭配，美食搭配面临独特的业务挑战：

- 套餐搭配的业务场景和搭配条件比较多样，因此套餐搭配方案需要满足各类业务、各种搭配条件下的需求。
- 美食商品属于非标品，且不同商家销售的商品各不相同，导致套餐搭配模式因商家而异。例如，不同商家销售的宫保鸡丁，在份量、口味、食材、价格上都不相同，因此对于宫保鸡丁这道菜也会有不同的套餐搭配方式。
- 算法搭配不可避免会产生低质量的搭配结果，而商品的非标品属性，导致我们更难衡量美食搭配的质量。低质量的搭配可能包括：a. 包含不适宜单独销售、非美食品类的搭配，例如包含赠品、锅具、餐具的搭配。b. 搭配结果不符合常规搭配模式，例如两份饮料，饮料 + 馒头等。

为此，我们的解决方案是：

- 为了解决业务场景和搭配条件多样的问题，我们形成了离线、实时结合的算法搭配框架。对于推荐型相关业务，我们采用离线搭配方法预先搭配出套餐候选，再在业务场景中做个性化排序。离线搭配本着规则到模型的迭代思路，规则搭配依赖知识图谱的商品表示，通过高频聚合 + 规则搭配泛化，产出相对高质量套餐以确保头部商家覆盖。模型搭配可以在保证搭配质量的同时，通过模

型泛化提高套餐的场景覆盖率。对于实时搭配型的业务，算法会依据业务的搭配条件实时搭配套餐，进一步提升各个实时场景下的套餐的覆盖率。

- 为了解决美食商品非标品的问题，我们引入外卖美食图谱对菜品进行了多方位的刻画。我们基于外卖知识图谱提取了菜品丰富的信息表征，例如菜品的标准菜品、菜品品类、口味、食材、做法等，减弱非标品带来的影响。
- 为了保证搭配套餐的质量，我们开发了套餐质量评估的模型。

总体，我们在非标品的商品表示、商家表示、套餐搭配模型、套餐搭配质量评估上都进行了相关的探索和迭代，形成了如下图 2 所示的套餐搭配框架。

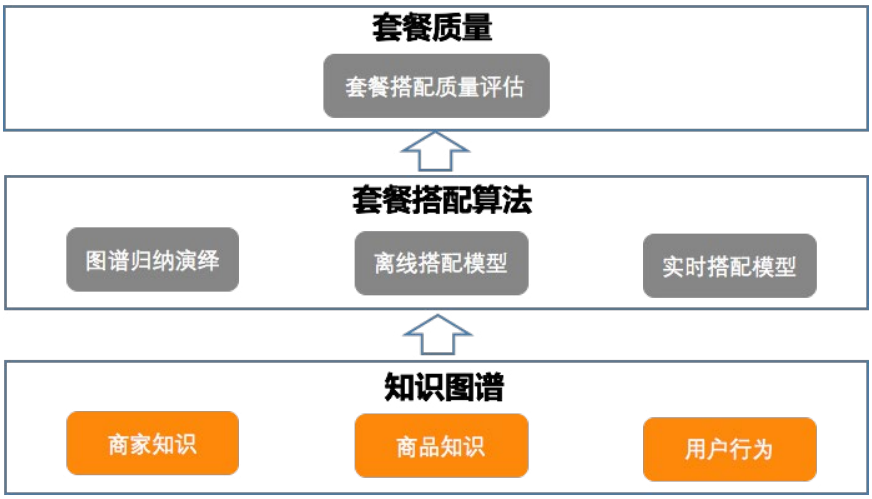


图 2　套餐搭配框架

3. 套餐搭配模型

3.1 基于图谱标签归纳的套餐搭配模型

我们面临的一个问题是外卖商品为非标品，菜品数据质量较差、属性缺失。为此，我们基于商家菜单、菜谱、商品描述等多种信息源，通过信息抽取、关系识别、知识融合等多种方法，构建了以美食为核心的知识图谱，对菜品建立了品类、口味、做法、功效等多个维度的表示。

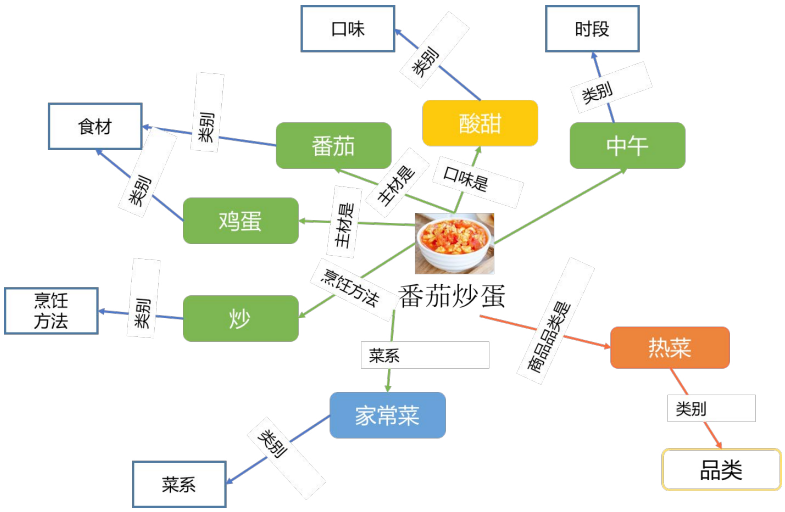


图 3　外卖美食知识图谱

商家的历史高销量套餐一般可认为是优质的套餐，但是，中低销量商家的高销量套餐数量较少，难以支持套餐的个性化推荐等应用。依赖美食图谱对菜品的语义表达，我们首先尝试了基于知识图谱的直接归纳演绎进行套餐搭配的方案。例如，通过高频订单可以归纳得出，{ 热菜 }+{ 米饭 }+{ 汤 } 是一个常见的套餐搭配方式，进而对于商家演绎出“番茄炒鸡蛋 + 番茄鸡蛋汤 + 米饭”的套餐搭配。

图谱归纳演绎的过程是高频聚合和基于搭配模板的泛化过程，我们通过订单聚合、同品牌、同标签、同菜品模板泛化，来产生高质量的套餐搭配，同时套餐的商家覆盖率有了显著的提升。但是搭配模板的问题在于较难对搭配质量和泛化程度取得折中。约束性较强的搭配模板可以确保搭配质量，但泛化能力不足，套餐覆盖度较低。如果用单一或少量标签描述搭配项，会导致模式过度泛化，准确率不能保证。为此，我们引入了基于模型的套餐搭配方法。

3.2 基于 Encoder-Decoder 的套餐搭配模型

用户搭配套餐也是一个信息编码到信息输出的过程：用户浏览商家菜单即是编码过程，得到该商家和商品信息的一个整体概况，再基于这个概况去进行套餐的搭配。贴合该过程的一个思路是采用 Encoder-Decoder 框架进行套餐搭配模型的建设，

Encoder 类比用户浏览菜单的过程，学习菜单的语义信息，Decoder 负责搭配出套餐。Encoder-Decoder 是一种深度学习网络框架，已经被广泛应用于文本摘要、机器翻译、对话生成等应用中，其建模的方式是通过编码（特征提取）和解码（目标拟合），学习从 Encoder 输入数据到 Decoder 输出数据的映射。常见的编码方式有 CNN、RNN、Transformer 等结构，解码方式也类似。

3.2.1 基于 LSTM 的套餐搭配模型

套餐生成问题是要从一个商家的所有候选商品集合中提取出多个商品子集，形成方便用户筛选、可直接下单的套餐。套餐生成的数据源主要是该商家的候选商品信息（如商品的名称、标签、价格、销量等），再结合用餐价格区间、用餐人数等约束条件，以及用户偏好等信息。最初我们采用了 LSTM 作为 Encoder 和 Decoder 的神经网络进行套餐搭配。我们基于图谱语义抽取商品语义表示，并输入 Encoder 的 RNN 模型。Encoder 编码过程类似于用户翻阅商家候选商品的过程，Encoder 端输入菜品名称、菜品标签、菜品的业务属性（价格、销量等），通过 LSTM 对非标品菜品进行特征抽取。如下图 4 所示，每个商品的名称经过 Embedding 层、CNN+Pooling 层提取特征，并同菜品标签、类别的 Embedding 以及价格和销量等连续特征进行拼接，最终作为 Encoder RNN 中每步的输入。

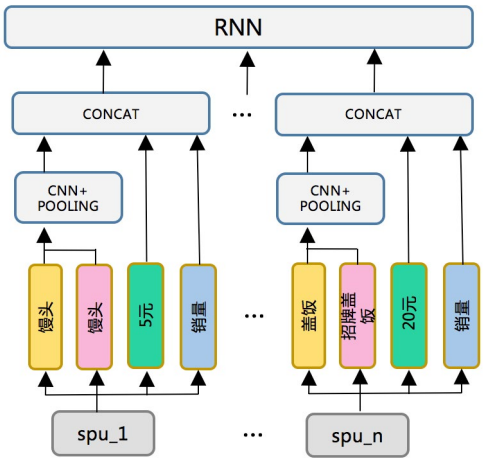


图 4　Encoder 网络结构

Decoder 在解码过程中一般依赖一个固定的字典或词典作为候选集，每一步输出候选集中的字、词被选中的概率分布。对于套餐搭配网络来说，Decoder 解码的候选集来源于 Encoder 输入端商家内的商品列表，而非固定维度的外部菜品词表，Pointer Network 是建模这一问题的有效架构。Pointer Network 是基于 Seq2seq 的扩展，主要解决的是候选集不固定的问题，这一模型架构已被成功应用于抽取式文本摘要，以及旅行商问题、凸包问题等组合优化问题的解决方案中。

套餐搭配解码的具体过程为，Decoder 每一步预估目标菜品来自菜品列表的概率分布。在第 n（n>=1）步时，这个概率分布向量表达了在已经选择了 n−1 个商品的情况下，某个商品或者终止位被选中的概率。若终止位对应的概率较大，则模型倾向以前 n−1 个已选商品形成一个完整的套餐搭配。解码过程中，我们结合 BeamSearch 算法产生 TopN 个结果，保证搭配的多样性。

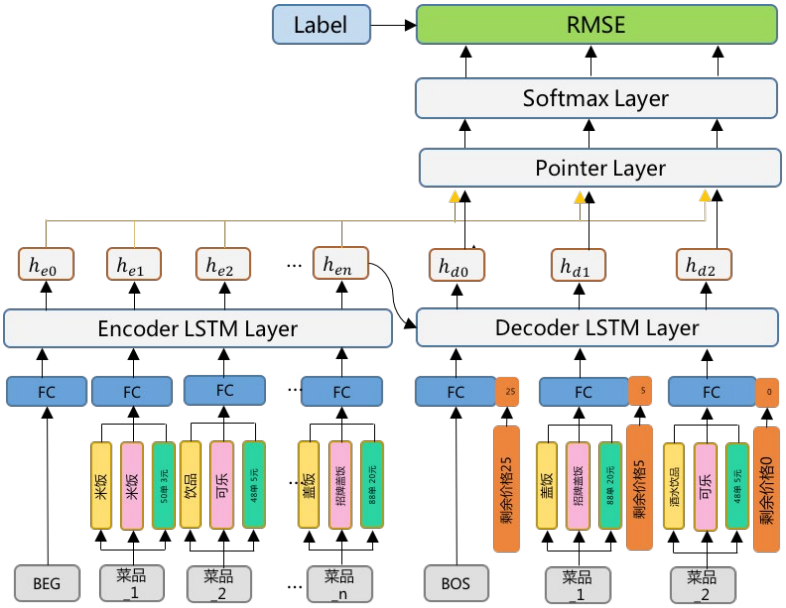


图 5　Encoder−Decoder 网络结构

3.2.2 套餐搭配模型的优化

套餐搭配模型的学习目标

为了解决菜品搭配模式因商家而异的问题，模型通过拟合该商家的历史订单来学习商家的搭配特性。一种较为主流的训练形式是基于商家真实订单，采用 Teacher Forcing 的形式进行训练，使得模型预测的菜品逐位匹配上真实订单内的菜品。Teacher Forcing 式的训练方法使得预测菜品的概率倾向于 0-1 分布，但是现实的菜品搭配通常是个性化、多样性的，例如在 Decoder 已经输出“宫保鸡丁”菜品的基础上，下一步选择的主食为“米饭”或者“炒饭”皆可。

为此，我们对商家历史成单的套餐搭配模式进行统计，计算出商品选择的概率分布，Decoder 以此商品选择的概率分布作为训练目标，同预估的分布计算 MSE Loss，并最小化该值指导模型的训练。Teacher Forcing 的另一个问题是较难引入搭配质量、套餐的点击购买行为等外部知识来指导模型训练，为此我们尝试了采用强化学习的思路进行改进。在解码过程的时刻 T，我们通过蒙特卡洛算法采样（Monte Carlo Sampling）出完整的套餐候选，并计算套餐候选的搭配质量分作为 Reward，结合 MSE Loss 和搭配质量分进行模型训练。

套餐搭配的约束

套餐搭配过程会面临多种业务约束，例如，对于“满减神器”来说，搭配的套餐需要满足给定的满减价格档位。“智能助手”的套餐搭配过程需要考虑用户选定的筛选条件，例如条件可能为“主食为米饭”且“价格为 30 元以下”。我们通过剪枝策略保证搭配过程满足约束，以“满减神器”的价格区间约束为例，Decoder 端在单步产生候选菜品时，会基于剩余的价格过滤掉超出剩余价格范围的菜品。如下图 6 所示，对于商家的 A、B、C、D、E 菜品，Decoder 会利用剩余价格的区间“15 元以内”对后一轮菜品 A、B、C、D、E 进行剪枝，并删除超过价格区间的 C、D 两个菜品。

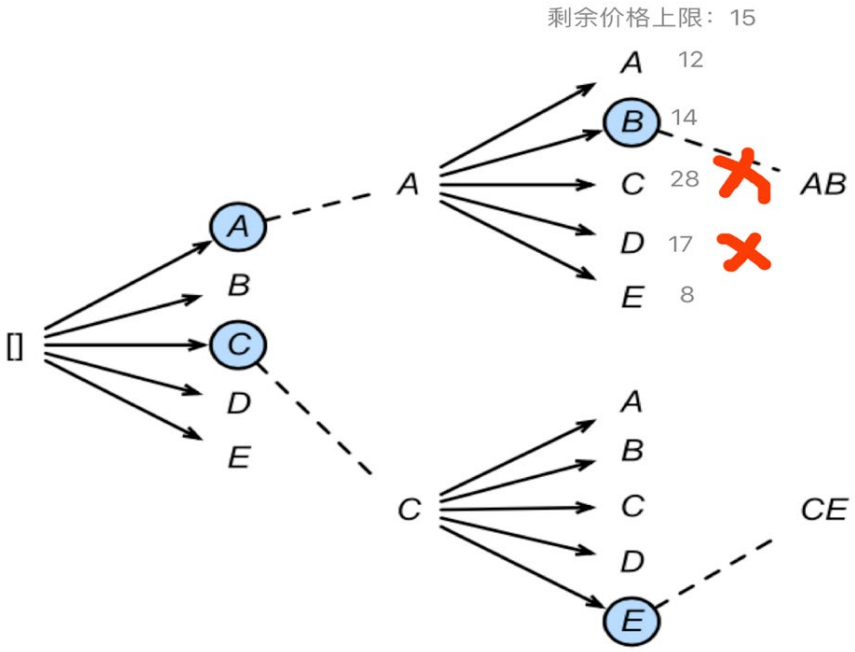


图 6　套餐搭配价格约束

基于 Attention 网络的套餐搭配模型

基于 LSTM 网络的商家内菜品特征提取面临的问题有：第一，商家菜单的菜品本身无序，而 RNN 网络依赖序列进行建模。第二，菜品之间可能存在长距离的语义依赖，例如，菜单内是否有“米饭”、“馒头”等菜品会影响对于“宫保鸡丁”菜品的搭配。

为了更好的对无序菜单和菜品之间依赖信息进行表征，我们尝试了基于 Attention 结构的 Encoder-Decoder 模型。Encoder 部分采用层次化 Attention 结构提取菜品的语义信息，包含底层单菜品级的 Attention 和菜品之间的 Attention 两部分。对于单菜品级的 Attention，我们在字维度采用 Multi-Head Attention 结构得到菜品名称的语义向量，菜品标签同样采用 Multi-Head Attention 得到菜品标签语义向量，对于菜品的交易属性，我们采用多层全连接网络提取交易特征的语义向量。

最后，菜品名称语义向量、菜品标签语义向量、交易特征语义向量拼接后经过全连接层 + 层归一化得到菜品语义向量。对于菜品之间的 Attention 层，我们对该店的菜品

语义向量列表采用多层 Multi-Head Attention 得到该店的菜单级别语义向量。模型 Decoder 部分同样采用 Multi-Head Attention 进行解码，输入信息包括用户偏好信息、历史时刻的解码输入、价格约束等上下文信息，模型在每步输出商家菜单中的菜品被选择的概率分布。Decoder 过程中我们对用户偏好信息同商家菜单级别的语义向量进行 Multi-Head Attention，在套餐搭配过程中考虑用户的用餐偏好。

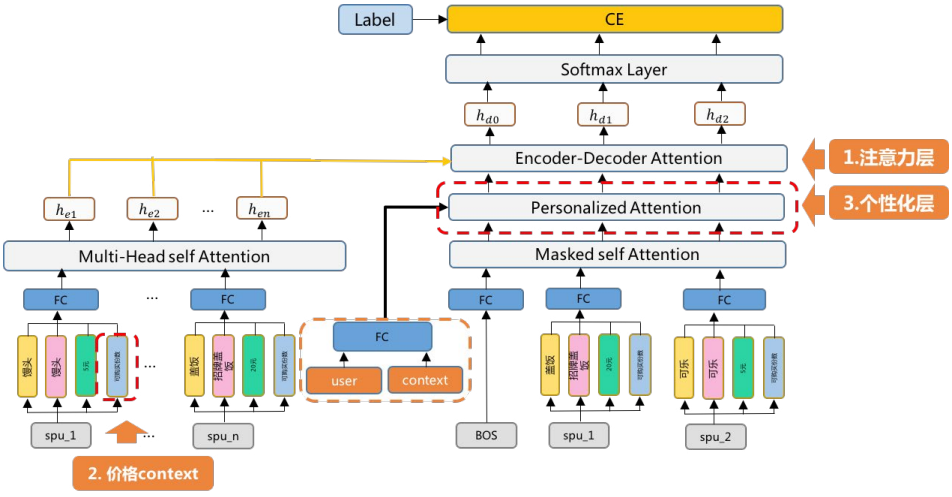


图 7　基于 Attention 的套餐搭配网络

3.2.3 套餐搭配模型分析

我们认为商家高质量的搭配可以从订单的销量体现，一种评估方法即是评估模型输出的套餐对商家真实高销量套餐的覆盖。通过离线和在线评估，我们发现该模型可以拟合出商家高销量套餐。在人工评估部分，我们把算法搭配的套餐和真实成单订单进行混合并让人工进行区分，发现人工无法分辨模型搭配的订单和真实成单订单的差异。同时，该模型具备良好的泛化能力，显著提升了套餐对商家和特定业务场景的覆盖度。

我们对模型输出的菜品表征向量进行了分析，以了解模型的套餐搭配模式。利用 TSNE 对向量进行降维和聚类，通过聚类图观察发现“主食”、“主菜”、“小食”类的菜品各自聚集在一起，可以看出模型识别了菜品的“主食”、“菜品”、”小食”等品类语义属性，并参考此语义进行套餐搭配。

主食类：”馄饨”的TOP N相似菜品	菜品类：”红烧肉”的TOP N相似菜品
鸡汤大馄饨 0.981	黄瓜拌牛肉 0.975
豌杂酸辣粉 0.979	鲜菇牛肉 0.977
猪肉馄饨 0.975	毛家红烧肉 0.980
清汤牛肉面 0.975	白菜肥肠 0.973
皮肚肥肠面 0.974	拌小肠 0.976
海鲜炒乌冬面 0.974	猪头肉 0.981
大葱肉锅贴 0.973	红烧小土豆 0.975
豌杂米粉 0.971	拌牛肉 0.980

3.3 实时套餐搭配模型

利用离线搭配产生套餐候选的方案可以满足推荐型业务的需求，但对于一些搭配型的业务场景仍然覆盖不足，例如，目前离线套餐对菜品的覆盖度较低，即对于菜品详情页等应用只保证部分 PV 的搭配模块露出。

一种解决方案是通过离线搭配提升套餐对美食商品的覆盖，但该方案的存储成本较高，为此我们采用实时套餐搭配方案。实时生成方案的难点在于既要保证套餐质量，又要满足各种搭配条件，最重要的还要保证实时性。最初我们把离线搭配模型应用于线上实时搭配，发现在性能方面存在瓶颈。因此，我们对离线模型进行了精简，精简的思路是将选菜的过程精简为选择菜品类目的过程，将菜品维度的搭配关系精简为菜品类目的搭配关系，实现整个解空间的缩小。如下图 8 所示，具体流程如下：

- **搭配模板挖掘**：通过商家历史订单挖掘出商家高销的类目层面的搭配关系，即搭配模板，例如“热菜 + 主食”。
- **搜索剪枝**：在搭配选菜时，依据搭配模板中的菜品类目，进行菜品的选择。例如上述的例子，首先进行“热菜”的选择，然后进行“主食”的选择。在选择的过程中，根据用户的实时需求，例如指定必选菜、指定价格、指定主食类型等约束条件，对整体的选择过程进行剪枝。
- **筛选评估**：在搭配完成后，对得到的候选搭配结果进行质量评估，这里基于性能考虑，使用树模型进行质量评估，筛选出 Top N 的搭配结果。

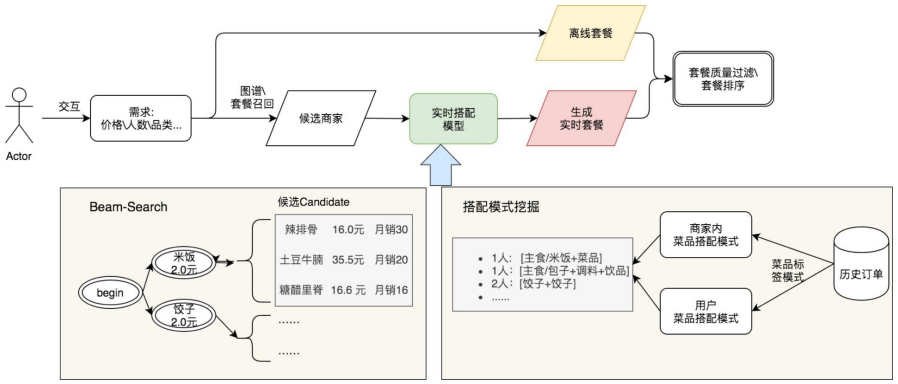


图 8　实时套餐搭配和离线套餐搭配

4. 套餐质量评估

高销量的订单中也存在搭配质量稍低的套餐，加上模型泛化的精度问题，搭配模型很容易生成质量较差的搭配组合。如下图 9 右侧，模型生成的后两个套餐搭配相对不是特别合理。为进一步保证用户体验，我们建立了套餐搭配质量模型，对套餐的质量进行统一评估。套餐质量分类模型将套餐搭配质量转化为一个分类问题。因为套餐组合是有多个菜品组成的，所以我们基于菜品名称、标签等信息构造菜品的表示，然后通过 Global-Attention 来实现菜品间重要程度的考量，同时添加总商品个数、总份数等全局特征来表示整体搭配的信息，模型结构具体见下图 9 所示：

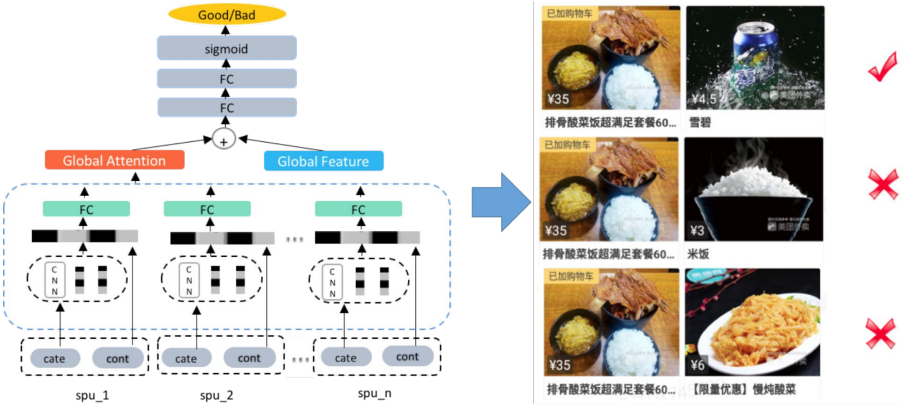


图 9　套餐质量分类

我们对套餐的搭配质量进行了细粒度的分档：极差、差、中、好，且四种类别存在有序关系（极差 < 差 < 中 < 好），对应模型有四个输出值，每一个表示该位为 1 的概率，例如“极差”表示为“1,0,0,0”，“差”表示为“1,1,0,0”，“中”表示为“1,1,1,0”，“好”表示为“1,1,1,1”。模型的损失采用 Pair Hinge Loss 函数，避免前面节点为 0 后面节点为 1 的情况，保证模型的准确度。套餐的搭配质量分值为四个输出节点的求和平均值，使得预测值更加可信。模型结构同一般的分类模型大体一致，目标函数如下：

$$-\frac{1}{m}\sum_{i=1}^{m}\sum_{k=1}^{4}(y_{ik}log\hat{y_{ik}}+(1-y_{ik})log(1-\hat{y_{ik}})+\lambda*\sum_{k=1}^{k'=4}\sum_{k<k'}max(0,-(\hat{y_{ik}}-\hat{y_{ik'}}))$$

图 10　套餐质量分类

套餐质量模型的建设过程中，负例样本主要来自用户反馈的 Bad Case，和通过人工构造不合理搭配模式版筛出的套餐。这种方式存在的问题是：Bad Case 和人工构造的搭配负样本有偏、多样性较差，且负样本同正样本的配比不容易调节。

为此，我们引入一个预训练任务学习历史订单的搭配模式，为套餐搭配质量模型引入更多的搭配先验知识。预训练过程如下图 11 所示，我们随机 Mask 已成单搭配组合中的一个菜品，然后训练 Transformer 模型还原被 Mask 掉的菜品。在这个过程中，考虑一些次优套餐的合理性（例如：”宫保鸡丁 + 米饭 + 可乐”，Mask 掉”宫爆鸡丁”，生成器生成出来的是“鱼香肉丝”，“鱼香肉丝 + 米饭 + 可乐”就可理解为一个次优套餐），我们在最后的损失函数添加预测菜品与目标菜品类别相似性的判别器来解决这类情况。预训练得到的参数最终用来初始化套餐搭配质量分档模型，并基于少量的人工标注语料进行模型的调优。

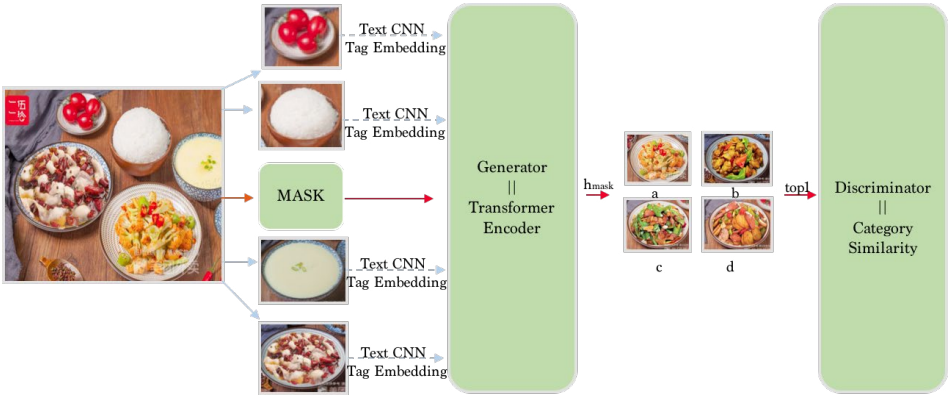


图 11　套餐质量分类

5. 套餐搭配的应用和未来展望

目前，外卖已打造了多款以套餐为核心供给的产品，“今日套餐推荐”帮助用户解决不知道吃什么、选购慢的问题，店铺页内的“满减神器”、“单品搭配推荐”解决用户凑单难、搭配难的问题。为了解决了各业务场景下套餐搭配难题，套餐搭配算法针对覆盖度、搭配质量、搭配多样性进行了持续优化，为业务提供了重要的技术和数据支撑。离线套餐搭配用于”满减神器“、”今日套餐推荐“等业务，显著提高了套餐商家的覆盖率，实时套餐搭配用于“菜品详情页套餐搭配”等业务，并取得较好的业务收益。

后续工作，一方面我们会继续优化菜品知识图谱的构建，完善对非标品菜品的刻画，通过引入图像等多模态数据进一步提升数据准确度和覆盖度，通过构建场景知识图谱更好地刻画用户需求和供给。另一方面我们会探索场景化的套餐搭配：目前我们在场景化套餐搭配方面的工作较少，而且用户在不同场景下会有不同的套餐需求，例如天气寒冷偏爱吃火锅类套餐、腊八节吃粥类套餐，在异地希望吃当地特色类套餐。接下来，我们会在场景化套餐搭配方面进行探索，针对节气、节日、人群等场景搭配套餐，更好的满足用户个性化、场景化的用餐需求。

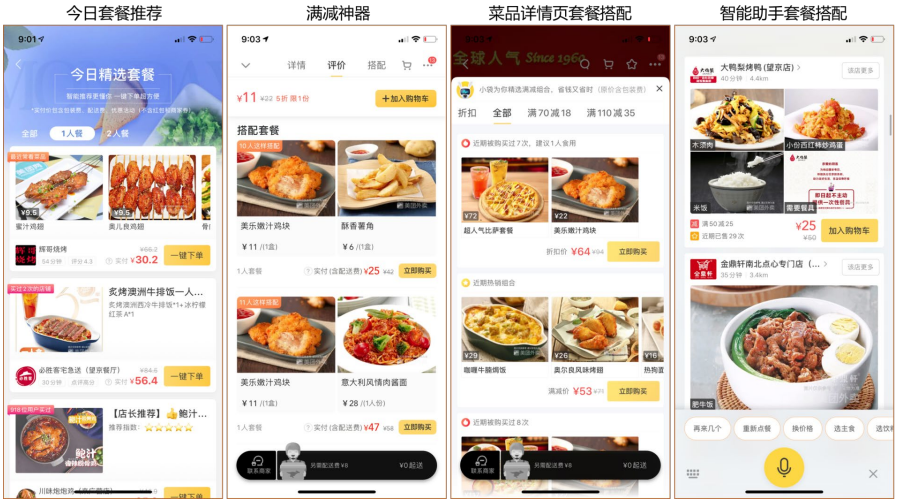


图 12　套餐搭配相关应用

6. 参考文献

[1] Vinyals, Oriol, Meire Fortunato, and Navdeep Jaitly. “Pointer networks.” Advances in neural information processing systems. 2015.

[2] See, Abigail, Peter J. Liu, and Christopher D. Manning. “Get to the point: Summarization with pointer-generator networks.” arXiv preprint arXiv:1704.04368 (2017).

[3] Gong, Jingjing, et al. “End-to-end neural sentence ordering using pointer network.” arXiv preprint arXiv:1611.04953 (2016).

[4] Han, Xintong, et al. “Learning fashion compatibility with bidirectional lstms.” Proceedings of the 25th ACM international conference on Multimedia. 2017.

[5] Alashkar, Taleb, et al. “Examples-Rules Guided Deep Neural Network for Makeup Recommendation.” AAAI. 2017.

[6] Chen, Wen, et al. “Pog: Personalized outfit generation for fashion recommendation at alibaba ifashion.” Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. 2019.

[7] Rush, Alexander M., Sumit Chopra, and Jason Weston. “A neural attention model for abstractive sentence summarization.” arXiv preprint arXiv:1509.00685 (2015).

[8] Paulus, Romain, Caiming Xiong, and Richard Socher. “A deep reinforced model for abstractive summarization.” arXiv preprint arXiv:1705.04304 (2017).

[9] See, Abigail, Peter J. Liu, and Christopher D. Manning. “Get to the point: Summarization with pointer-generator networks.” arXiv preprint arXiv:1704.04368 (2017).

7. 作者简介

瑞玉、文斌、杨林、懋地，均来自美团外卖技术团队。

自动驾驶轨迹预测算法：NeurIPS 挑战赛冠军方案

作者：炎亮　傅壮　德恒　冬淳

01 背景

NeurIPS（Conference on Neural Information Processing Systems）是机器学习和计算神经科学相关的学术会议，也是人工智能方向的顶级会议。INTERPRET 轨迹预测挑战赛（INTERACTION-Dataset-based PREdicTion Challenge）隶属于 NeurIPS 2020 Workshop: Competition Track Saturday。该竞赛由 UC Berkeley MSC Lab 主办，旨在建立一个公共数据集来评估自动驾驶领域各类轨迹预测算法的性能。美团无人车配送中心团队获得 NeurIPS 2020 INTERPRET 轨迹预测挑战赛 Generalizability 赛道冠军，Regular 赛道亚军。

02 赛题简介

INTERPRET 竞赛共包含两条赛道：Generalizability Track 和 Regular Track。Generalizability 赛道中，测试集轨迹与训练集差异较大（采自不同的场景），且不带有高精地图；而 Regular 赛道中测试集轨迹分布与训练集相同（采自相同的场景），同时带有高精地图。数据集采自于美国、中国、德国等多个国家，包含并线 / 变道的高速与城市道路、带停车 / 让车标志的环形路、无保护左转路口等场景。此外，场景中的障碍物包含行人、自行车和机动车 3 种类型。

本次竞赛中，参赛队伍需要根据每个障碍物过去 1 秒（10 帧）的运动轨迹，预测出它在未来 3 秒（30 帧）的轨迹。障碍物的轨迹使用离散采样点集合来表示，采样的频率是 10 赫兹，即每 0.1 秒采样一个轨迹点。竞赛允许参赛队伍对于每个障碍物输出 50 条预测轨迹，但只根据最优轨迹（排序第 1 条）的平均位移误差（Average Displacement Error，ADE）来排名。平均位移误差的计算方式为：

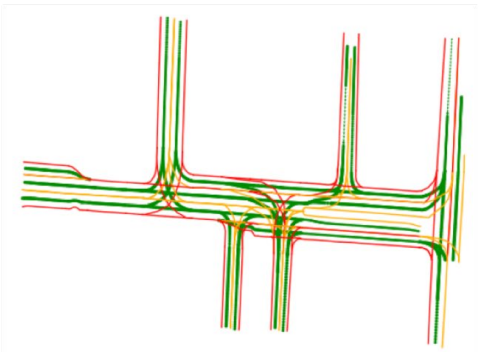
$$ADE = \frac{1}{NT_p}\sum_{i=1}^{N}\sum_{t=1}^{T_p}\| Y_i^t - \hat{Y}_i^t \|_2$$

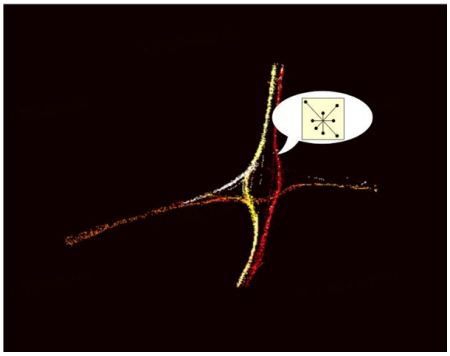
其中 N 表示障碍物的数量，T_p 表示预测时刻数，Y 表示真实轨迹，$\hat{Y}$ 表示预测轨迹。

03 算法介绍

Part 1 地图数据处理

由于 Generalizability Track 和 Regular Track 数据形式不一致（前者带有高精地图，后者不带高精地图），为保证算法的有效性，我们使用了两种形式来更好地表达场景。如下图 1：





在 Regular Track 中，所有测试集都附带高精地图，我们可以通过查询地图的方式得到任意位置附近的车道线（如图 1- 左所示，场景中道路拓扑非常完整）；而对于 Generalizability Track，测试集没有给出对应的高精地图，无法获取完整的道路结构化信息。对此，我们设计了一种基于地理位置的语义地图来描述非结构化场景下的可行驶区域。基于地理位置的语义地图依赖于场景中障碍物的历史观测轨迹，具体绘制流程主要分为 3 个步骤：

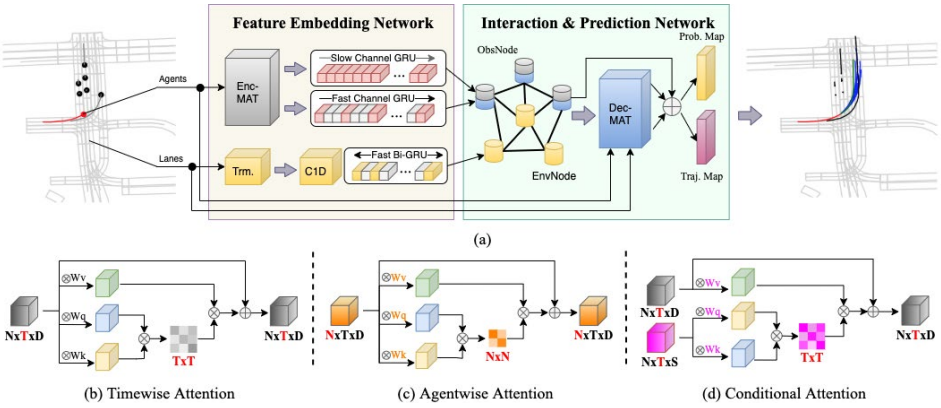
1. 将一定大小（e.g., 50 米 x50 米）的实际场景区域以 d（e.g., 0.2 米）的分辨率划分成 $W \times H$ 个栅格（W=250, H=250）；
2. 根据障碍物的历史轨迹，计算每个轨迹点的方向，放入对应位置的栅格中；
3. 统计所有栅格的运动方向信息，形成方向描述子。具体方法是：将 360 度划分成 8 个区间，45 度一个区间，统计每个区间内的轨迹点个数，然后归一化。

最终，基于地理位置的语义地图的大小为 $H \times W \times 8$ 。

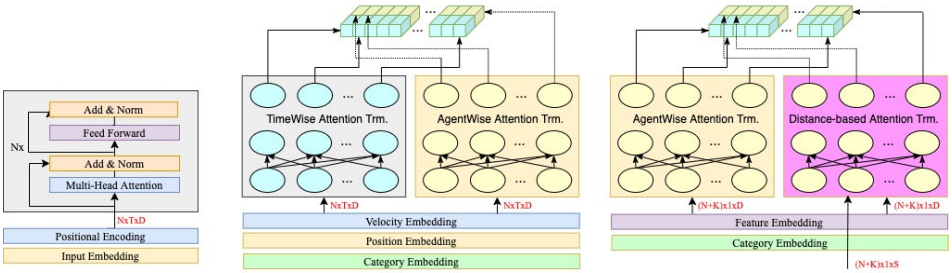
Part 2 预测模型设计

轨迹预测算法设计过程中通常需要考虑一个重要问题：即在预测时如何建模障碍物与周围环境的复杂交互，这里周围环境通常包含多类交通元素，例如其他交通参与者，路网拓扑，交通信号灯等。现有的预测算法中，对障碍物交互的建模方式也不尽相同，例如较早期基于简单位置关系的交互 [1-3]，基于语义地图 +CNN 编码的交互 [4-6]，基于（图）注意力机制的交互 [7-11] 等。随着对障碍物交互认知的加深以及新技术的迭代，

轨迹预测算法的精度也在逐步提高。本次竞赛中，我们提出一种基于混合注意力机制的预测算法，以通用的形式解决两个赛道的预测问题。算法是基于目前主流的图注意力机制，整体设计思路是通过引入混合注意力机制，促使算法更准确地提取障碍物运动特性与车道拓扑特性，同时编码障碍物之间、障碍物与车道间的复杂交互。



上图 2 是算法的整体结构，整个模型基于主流的 Encoder-Decoder 结构，包含特征编码网络（Feature Embedding Network）和交互 & 预测网络（Interaction & Prediction Network）。特征编码网络使用 Timewise + Agentwise Attention 双注意力机制与双通道 GRU 对障碍物轨迹和地图信息进行高质量特征强化与时序编码；交互预测网络则使用 Agentwise + Conditional Attention 双注意力机制建模智能体间交互行为，并输出多模态预测轨迹及其概率。上述两个网络都是基于混合注意力的图网络，其核心是 Enc-MAT 和 Dec-MAT（Mixture Attention Transformer encoder）模块。Enc-MAT 和 Dec-MAT 是现有 BERT-like 模型（Transformer encoder）的改进结构，下图 3 对比了传统 Transformer encoder、Enc-MAT 和 Dec-MAT 的区别。



(a) Transformer Encoder (b) Enc-MAT (c) Dec-MAT

从图 3 可以看到，相比于传统 Transformer encoder，Enc-MAT 和 Dec-MAT 改进并额外新增加了一个注意力通道，混合注意力机制也由此而来。Enc-MAT 编码器使用了 Timewise 和 Agentwise 混合注意力机制；Dec-MAT 编码器则是使用了 Agentwise 和 Conditional（同图 3-c 中的 Distance-base Attention）混合注意力机制。算法使用混合注意力代替原有单注意力机制，目的在于结合实际需求来强化障碍物与环境拓扑的特征表达。图 2 下半部分展示了三种 Attention 结构，从计算形式上看，三种注意力方式的计算公式是一致的：

$$Attention(Q,K,V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$

区别在于三种注意力方式中，Attention 模块的 QKV 特征的产生方式有区别：

1. 在 Timewise Attention 中，QKV 计算方式为：

$$Q = \phi_{q,ta}(e; w_{q,ta}), K = \phi_{k,ta}(e; w_{k,ta}), V = \phi_{v,ta}(e; w_{v,ta}), e \in \mathbb{R}^{N\times T\times D}$$

2. 在 Agentwise Attention 中，QKV 计算方式为：

$$Q = \phi_{q,aa}(e; w_{q,aa}), K = \phi_{k,aa}(e; w_{k,aa}), V = \phi_{v,aa}(e; w_{v,aa}), e \in \mathbb{R}^{T\times N\times D}$$

3. 在 Conditional Attention 中，QKV 计算方式为：

$$Q = \phi_{q,ca}(c; w_{q,ca}), K = \phi_{k,ca}(e; w_{k,ca}), V = \phi_{v,ca}(e; w_{v,ca}), e \in \mathbb{R}^{N\times T\times D}, c \in \mathbb{R}^{N\times T\times S}$$

上述公式中 e 是输入数据（例如轨迹、车道等）的 Embedding 编码特征，大小为 $N\times T\times D$；c 是额外输入包含人工先验的条件信息，下文会做解释，大小为 $N\times T\times S$；$\phi_*(\cdot)$

是编码函数，例如 Linear 函数，w_* 是对应的编码函数参数。此外，三类 Attention 模块内部也额外增加了 ADD 输入的操作。

Part 3 轨迹预测流程

首先对符号作一些说明：

假设场景中障碍物的数量为 N，历史观测时长为 T，所有障碍物的历史观测轨迹为 P。则 P 大小为 $N \times T \times 6$，6 维特征包含坐标、速度和类别。预测的时长为 T_p，预测轨迹为 $\hat{Y}$，未来真实轨迹为 Y，在输出一条轨迹的情况下，$\hat{Y}$ 和 Y 大小均为 $N \times T_p \times 2$。

此外，当存在高精地图时，假设场景中车道的数量为 K，车道的采样点个数为 L，场景（离散车道）可以被表示为 M。M 大小为 $K \times L \times 2$；当不存在高精地图时，直接使用基于地理位置的语义地图，场景（语义图像）同样可以被表示为 M。M 大小为 $H \times W \times 8$。

编码过程主要包含障碍物历史轨迹编码与场景拓扑编码。以障碍物特征编码为例，流程主要分为两步：

1. **Dec-MAT 编码**。给定障碍物观测轨迹 P，算法首先使用 Enc-MAT 强化每个障碍物的特征。Enc-MAT 中第一个注意力通道在时间维度对数据进行 Timewise Attention 操作，目的是对于每个障碍物，根据其他历史时刻的信息，来强化某个时刻的信息；第二个注意力通道在障碍物维度进行 Agentwise Attention 操作，目的是对于每个时刻，根据其他障碍物的位置信息，来强化某个障碍物的信息。最终两个通道的特征拼接得到障碍物轨迹特征 R_{agent}Ragent；
2. **Slow+Fast Channel GRU 时序编码**。特征 R_{agent} 是经过其它时刻（或者障碍物）强化过的障碍物信息，然后我们通过 Slow Channel GRU 和 Fast Channel GRU 来分别提取障碍物的精细化运动特征和主体运动特征。Slow Channel 是用特征 R_{agent} 的全部时序信息（$N \times T \times 2D$），经过一个 Slow GRU 进行时序编码得到特征 H_1；Fast Channel 则先对特征 R_{agent} 在时序上降采样

（$N \times T_{down} \times 2D$），再经过 Fast GRU 得到特征 H_2。最终障碍物的时序编码结果 H_{agent} 由 H_1 和 H_2 拼接得到，它记录了障碍物的时序运动特性。

对于道路拓扑的编码也采用相似的方式，但与轨迹编码有两个区别：

1. **特征强化区别**。为了简化计算，我们只使用常规的 Transformer Encoder 对场景信息进行 Timewise Attention 单通道注意力特征强化，得到特征 R_{env}。高精地图模式下，R_{env} 为三维特征，大小为 $K \times L \times D$；语义地图模式下，R_{env} 为三维特征，大小为 $1 \times HW \times D$；
2. **时序特征编码区别**。基于上述特征 R_{env}，高精地图模式下，我们使用 Fast Channel Bi-GRU 直接进行双向道路拓扑特征编码；而语义地图模式下，我们使用 Fast X Bi-GRU 和 Fast Y Bi-GRU 对图像横纵两个方向进行时序编码。最终得到特征 H_{env}。

解码过程主要包含高层交互和轨迹预测两个阶段。前者采用混合注意力网络 Dec-MAT，后者使用基础的 MLP 实现轨迹与概率的多任务预测。在介绍流程前，我们先阐述两个相对合理的事实：

- 事实 1：障碍物运动方向和场景中车道走向存在关联（运动趋势关联）。
- 事实 2：障碍物运动更依赖与距离它更近的邻近车道（相对位置关联）。

基于上述两个事实，解码器两个阶段的流程分别可以描述为：

1. 高层交互阶段。障碍物时序特征 H_{agent} 和场景拓扑特征 H_{env} 特征组成一张全局的图，这张图可以表示为一个混合矩阵 H_{mix}（高精地图模式下大小为 $(N+L) \times 1 \times 2D$，语义图模式下为 $(N+1) \times 1 \times 2D$）。基于这张混合图，我们同样使用双通道混合注意力模块 Dec-MAT 进行特征强化与编码。其中，第一个通道对于特征 H_{mix} 的第一维度进行 Agentwise Attention 操作，目的是根据每个障碍物各自的时序运动特征来做全局交互，这基于上述事实 1；第二个通道对于特征 H_{mix}，引入位置信息 c 作为条件进行 Conditional Attention 操作，目的是根据障碍物之间的相对位置关系来做全局交互，这基于上述事实

2。条件特征 c 是障碍物和车道的绝对位置坐标，因此 Conditional Attention 也称为 Distance-based Attention。最终两个通道的编码特征以及输入 H_{mix} 特征相加，过滤掉道路信息后得到障碍物特征 G_{agent}（大小为 $N \times 2D$）；

2. 轨迹预测阶段。G_{agent} 通过 Backbone 以后，经过两个 MLP head 分别输出预测轨迹 $\hat{Y}$ 和轨迹概率 Pr。

最终在 Generalizability 赛道上，我们以 ADE 0.5339 米获得冠军；Regular 赛道上，我们以 ADE 0.1912 米获得亚军。

04 总结

障碍物轨迹预测对无人车安全行驶具有重要的意义，它也是学界与工业界公认有很大挑战性的课题。我们希望通过努力做出更好的解决方案，持续提升自动驾驶系统对障碍物的预测能力，为美团实际业务及出行领域提供更多的技术支持。

05 参考文献

[1] Alahi A, Goel K, Ramanathan V, et al. Social lstm: Human trajectory prediction in crowded spaces[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2016: 961-971.

[2] Gupta A, Johnson J, Fei-Fei L, et al. Social gan: Socially acceptable trajectories with generative adversarial networks[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2018: 2255-2264.

[3] Zhu Y, Qian D, Ren D, et al. StarNet: Pedestrian trajectory prediction using deep neural network in star topology[C]//Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems. 2019: 8075-8080.

[4] Chai Y, Sapp B, Bansal M, et al. Multipath: Multiple probabilistic anchor trajectory hypotheses for behavior prediction[J]. arXiv preprint arXiv:1910.05449, 2019.

[5] Chang M F, Lambert J, Sangkloy P, et al. Argoverse: 3d tracking and forecasting with rich maps[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2019: 8748-8757.

[6] Liang J, Jiang L, Niebles J C, et al. Peeking into the future: Predicting future person activities and locations in videos[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2019: 5725-5734.

[7] Mohamed A, Qian K, Elhoseiny M, et al. Social-STGCNN: A Social Spatio-

Temporal Graph Convolutional Neural Network for Human Trajectory Prediction[C]// Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2020: 14424-14432.

[8] Liang M, Yang B, Hu R, et al. Learning lane graph representations for motion forecasting[C]//European Conference on Computer Vision. Springer, Cham, 2020: 541-556.

[9] Huang Y, Bi H K, Li Z, et al. STGAT: Modeling spatial-temporal interactions for human trajectory prediction[C]//Proceedings of the IEEE International Conference on Computer Vision. 2019: 6272-6281.

[10] Gao J, Sun C, Zhao H, et al. VectorNet: Encoding HD maps and agent dynamics from vectorized representation[J]. arXiv preprint arXiv:2005.04259, 2020.

[11] Zhao H, Gao J, Lan T, et al. Tnt: Target-driven trajectory prediction[J]. arXiv preprint arXiv:2008.08294, 2020.

06 作者简介

炎亮、傅 壮、德恒、冬淳，均为美团无人车配送中心算法工程师。

KDD 2021 | 美团联合多高校提出多任务学习模型，已应用于联名卡获客场景

作者：冬博　陈振

> 论文下载：《Modeling the Sequential Dependence among Audience Multi-step Conversions with Multi-task Learning in Targeted Display Advertising》
> 源代码：https://github.com/xidongbo/AITM

Modeling the Sequential Dependence among Audience Multi-step Conversions with Multi-task Learning in Targeted Display Advertising

Dongbo Xi[1,*], Zhen Chen[1,*], Peng Yan[1], Yinger Zhang[1,2],
Yongchun Zhu[3,4], Fuzhen Zhuang[5,6], and Yu Chen[1]
[1]Meituan
[2]Zhejiang University, Hangzhou 310027, China
[3]Key Lab of Intelligent Information Processing of Chinese Academy of Sciences (CAS),
Institute of Computing Technology, CAS, Beijing 100190, China
[4]University of Chinese Academy of Sciences, Beijing 100049, China
[5]Institute of Artificial Intelligence, Beihang University, Beijing 100191, China
[6]SKLSDE, School of Computer Science, Beihang University, Beijing 100191, China
{xidongbo,chenzhen06,yanpeng04,chenyu17}@meituan.com,
zhangyinger@zju.edu.cn, zhuyongchun18s@ict.ac.cn, zhuangfuzhen@buaa.edu.cn

一、摘要

很多应用通常都需要用定向展示广告来获客，比如电子商务平台或者金融应用。随着这类应用的爆炸式增长，如何进行持续有效的获客，已经成为现实世界中这些大规模应用最大的挑战之一。在这些应用中，获客通常是一个用户多步转化的过程。例如在电子商务平台中，用户转化通常表现为一个**曝光 -> 点击 -> 购买**的过程。而在金融广告，比如信用卡业务中，用户转化通常是一个**曝光 -> 点击 -> 申请 -> 核卡 -> 激活**的过程。所以，信用卡广告中获客比传统广告更加具有挑战性：

- 用户多步转化过程中的路径序列依赖更长。
- 在多步转化的长序列依赖中，正反馈（正样本）是逐步稀疏的，类别不平衡越来越严重。

在美团的信用卡业务中，我们通常希望用户能完成最后两个转化阶段（即**曝光 -> 核卡**、**曝光 -> 激活**），这才被认为是有效转化。因此，用前边任务的丰富的正样本信息，来缓解后续任务的类别不平衡问题是很有必要的。而在这个方向上，多任务学习是一个提高端到端获客转化率的典型解决方案。

二、背景介绍

在我们的信用卡业务中：

- **曝光（Impression）**：意味着广告被展示给了特定的用户，这些用户是根据一些排序指标选中的，比如 CTR、CVR 等等。
- **点击（Click）**：用户如果对这个广告内容感兴趣的话，他会点击这个广告，点击后，就会进入申请表格页。
- **申请（Application）**：进入申请表格页后用户可以填写申请表格，并且点击申请按钮来申请一张信用卡。
- **核卡（Approval）**：也称授信，意味着用户信用良好，通过申请并被授予了一定的信用卡额度。在我们的系统中，这在多数情况下也是一个系统实时判断的过程。
- **激活（Activation）**：用户在授信并且收到邮寄的信用卡之后，可以激活信用卡并使用。是否激活的标签通常比较难获得，因为信用卡邮寄需要一定的时间，而且用户主动去激活也需要一定的表现期，所以这里类别不平衡更为严重。在这里，我们通常看用户是否会在核卡后 14 天内激活信用卡。

以上的转化步骤有序列依赖，这意味着只有前一个步骤发生了，后一个步骤才可能发生。基于这个约束，用户只可能有 5 个不同的转化阶段如下图所示：

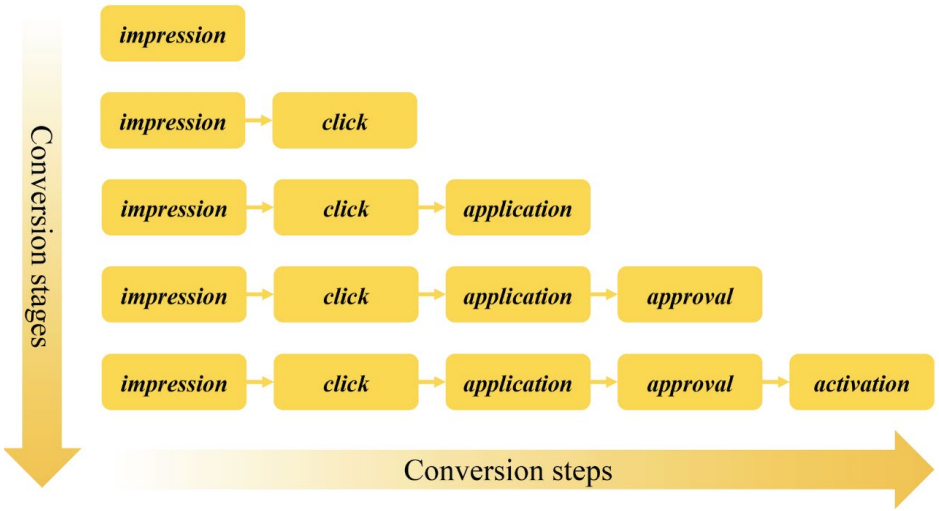


图 1　5 个不同的转化阶段

在工业界和学术界，多任务学习是一个提高端到端获客转化率的典型解决方案。近年来，在多任务学习中建模任务间关系的研究取得了很大的进展。我们将这些主要研究分为两大类：

1）控制多任务模型底部的专家模块如何在任务间参数共享 [1,2,3]，顶部的多塔模块分别处理每个任务，正如图 3（a）所示，我们称这一类为 ** 专家底（Expert-Bottom）** 模式。然而，专家底模式只能在任务间传递浅层表示，但在靠近输出层的网络中往往包含更丰富、更有用的表示 [4,5]，这已被证明能带来更多的增益 [6]。另外，由于专家底模式不是专门为具有序列依赖的任务设计的，因此这些具有专家底模式的模型不能显式地对序列依赖进行建模。

2）在不同任务的输出层中迁移概率 [7,8,9,10]，如图 3（b）所示，我们称之为 ** 概率迁移（Probability-Transfer）** 模式。概率迁移模式只能通过标量乘积传递简单的概率信息，而忽略了向量空间中更丰富、更有用的表示，导致了信息的损失。如果其中任何一个概率没有被准确预测，多个任务将会受到影响。

本文针对序列依赖任务，提出了一种自适应信息迁移多任务（**Adaptive Information Transfer Multi-task，AITM**）框架，该框架通过自适应信息迁移（AIT）模块对用户

多步转化之间的序列依赖进行建模。AIT 模块可以自适应地学习在不同的转化阶段需要迁移什么和迁移多少信息。此外，通过在损失函数中加入行为期望校准器，AITM 框架可以更准确地识别端到端转化。该框架被部署在美团 APP 中，利用它来为对美团联名卡有高转化率的用户实时展示联名卡广告。

三、系统概览

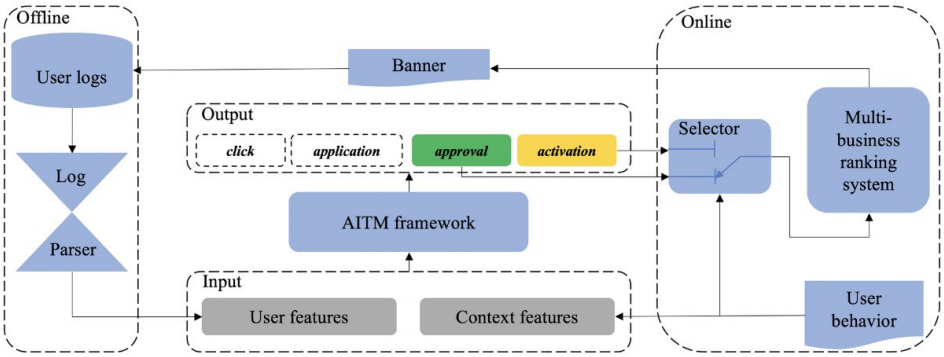


图 2　美团 App 中的多任务排序系统

上图给出了一个多任务排序系统示意图。在我们的信用卡业务中，除了被动的曝光步骤外，我们建模了四个任务。其中核卡、激活是主要任务，点击、申请是辅助任务。这是因为如果用户只完成了点击和申请步骤，而核卡步骤还没有完成，那么就不是一次有效转化，就会造成资源的浪费（例如计算和流量资源）。因此，我们主要关注最后两个端到端的转化任务，即**曝光 –> 核卡**和**曝光 –> 激活**。由于后两个任务的正样本数较少，且激活是延迟反馈的，而前两个辅助任务的正样本数较多，因此可以通过前两个任务来缓解后两个任务的类别不平衡问题。

另外，美团联名信用卡是与不同的银行合作发行的，不同的银行处于不同的业务发展阶段，因此对核卡和激活有不同的要求。信用卡业务刚刚起步的银行往往希望发行更多的信用卡以迅速占领市场，而发展成熟的银行则希望提高激活率以实现快速盈利。因此，我们的系统中有一个选择器，可以为不同的银行输出不同的转化目标，多任务框架可以很好地处理不同的业务需求。

此外，由于美团不同的业务都需要流量来为各自业务获客，而且不同用户对不同业务的兴趣也不同，所以不能简单地将流量划分给不同的业务。我们需要一个多业务排序机制来分配流量从而最大化整体的利益。

四、模型介绍

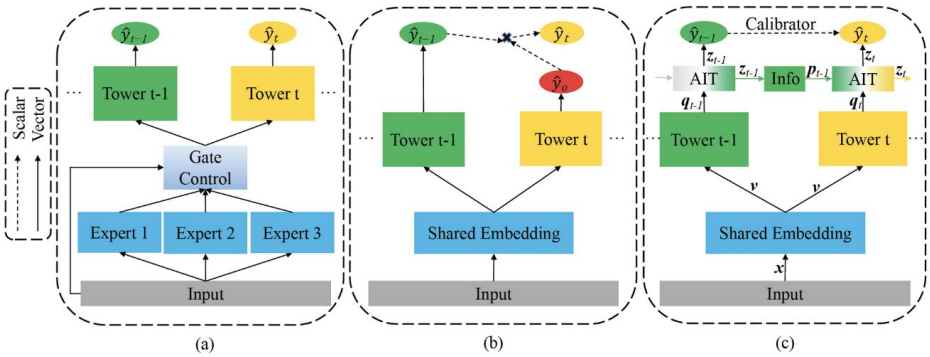


图 3 （a）专家底模式（b）概率迁移模式（c）AITM 框架

上图（c）中展示了我们提出的 AITM 框架。该框架利用 AIT 模块来建模用户多步转化过程中的序列依赖。这个模型图显示了两个相邻的任务：$t-1$ 和 t。

首先，输入特征 $\boldsymbol{x}$ 经过任务共享的 Embedding 分别输出到多个 Tower 网络中。通过共享 Embedding 模块，一方面可以用前边任务的丰富的正样本信息来学习 Embedding 表示，从而缓解后续任务的类别不平衡问题，另一方面可以减少模型参数。Tower 网络可以根据自己需要定制，这里我们只使用了简单的 MLP。然后，AIT 模块利用当前任务 Tower 输出的向量 $\boldsymbol{q}_t$ 以及前一个任务传来的信息 $\boldsymbol{p}_{t-1}$ 来学习任务间如何融合信息。AIT 模块如下定义：

$$\boldsymbol{z}_t = AIT(\boldsymbol{p}_{t-1}, \boldsymbol{q}_t) \ ,$$

$$\boldsymbol{p}_{t-1} = g_{t-1}(\boldsymbol{z}_{t-1}) \ .$$

这里其实利用了特殊设计的注意力机制来自动为迁移信息 $\boldsymbol{p}_{t-1}$ 和原始信息 $\boldsymbol{q}_t$ 来分配权重。而迁移的信息 $\boldsymbol{p}_{t-1}$ 是通过函数 $g_{t-1}(\cdot)$ 来学习的，这里 $g_{t-1}(\cdot)$ 可以是一个简单的全

连接层，用来学习两个相邻的任务间应该迁移什么信息。而该函数的输入 $\boldsymbol{z}_{t-1}$ 是前一个任务的 AIT 模块的输出。

具体地，AIT 模块中的注意力机制如下设计：

$$\boldsymbol{z}_t = \sum_{\boldsymbol{u} \in \{\boldsymbol{p}_{t-1}, \boldsymbol{q}_t\}} w_u h_1(\boldsymbol{u}) \ ,$$

$$w_u = \frac{exp(\hat{w}_u)}{\sum_u exp(\hat{w}_u)}, \hat{w}_u = \frac{< h_2(\boldsymbol{u}), h_3(\boldsymbol{u}) >}{\sqrt{k}} \ .$$

其中，w_u 是自动学习的迁移信息的权重。$h_1(\cdot)$，$h_2(\cdot)$，$h_3(\cdot)$ 利用前馈神经网络将输入投影到新的向量空间。

最后，我们通过在损失函数中施加校准器来约束概率的输出尽量满足序列依赖。损失函数是交叉熵 + 校准器约束，其中 $\alpha \in [0,1]$ 定义了校准器约束强度：

$$L(\theta) = L_{ce}(\theta) + \alpha L_{lc}(\theta) \ .$$

具体地，交叉熵定义如下：

$$L_{ce}(\theta) = -\frac{1}{N} \sum_{t=1}^{T} \sum_{(\boldsymbol{x}, y_t) \in D}^{N} \left(y_t log \hat{y}_t + (1 - y_t) log(1 - \hat{y}_t) \right) \ .$$

校准器约束定义如下：

$$L_{lc}(\theta) = \frac{1}{N} \sum_{t=2}^{T} \sum_{\boldsymbol{x} \in D}^{N} max(\hat{y}_t - \hat{y}_{t-1}, 0) \ .$$

如果 $\hat{y}_t$ 大于 $\hat{y}_{t-1}$，校准器将输出一个正的惩罚项，否则输出 0。

五、实验分析

在本节中，我们将在工业和公开现实数据集上进行实验，对比各种基线模型来评估我们提出的 AITM 框架。接下来，首先会介绍使用的数据集，然后是离线和在线实验结果，最后是进一步的实验分析（包括消融实验、超参数实验和 Case Study）。

5.1 数据集

我们使用两个数据集进行了离线实验。工业数据集是美团联名卡广告曝光的所有样本，我们取了一段时间的样本。该数据集有 4 个任务，分别是点击、申请、核卡、激活。在我们的业务中，我们只关注核卡和激活的转化指标，这两步转化才是有效转化。公开数据集是使用的阿里的点击转化预估数据集 [9,11]，这个数据集有点击、购买两个任务。

Dataset	#Task	#Train	#Validation	#Test	%Positive(%)
Industrial	4	20M	3M	26M	23.29/1.84/1.30/1.00
Public	2	38M	4.2M	43M	3.89/0.02

表 1　数据集统计信息。其中，“%Positive”代表采样后训练集中每个任务的正样本占比

5.2 离线、在线实验

Model	Industrial dataset				Public dataset			
	approval AUC	*activation* AUC	Gain		*click* AUC	*purchase* AUC	Gain	
LightGBM	0.8392±0.0011	0.8536±0.0035	-	-	0.5837±0.0005	0.5870±0.0038	-	-
MLP	0.8410±0.0010	0.8602±0.0014	+0.0018	+0.0066	0.6048±0.0013	0.5806±0.0035	+0.0211	-0.0064
ESMM	0.8443±0.0028	0.8691±0.0025	+0.0051	+0.0155	0.6022±0.0020	0.6291±0.0061	+0.0185	+0.0421
OMoE	0.8438±0.0022	0.8714±0.0009	+0.0046	+0.0178	**0.6049±0.0020**	0.6405±0.0041	**+0.0212**	+0.0535
MMoE	0.8444±0.0026	0.8705±0.0009	+0.0052	+0.0169	0.6047±0.0017	0.6420±0.0031	+0.0210	+0.0550
PLE	0.8518±0.0006	0.8731±0.0016	+0.0126	+0.0195	0.6039±0.0014	0.6417±0.0013	+0.0202	+0.0547
AITM	**0.8534±0.0011****	**0.8770±0.0005***	**+0.0142**	**+0.0234**	0.6043±0.0016	**0.6525±0.0024****	+0.0206	**+0.0655**

表 2　工业和公开数据集上的 AUC（均值 ± 标准差）表现。Gain 表示与 LightGBM 模型相比的 AUC 提升，下划线代表最好的基线模型。“*”表示与最好的基线模型相比，AITM 在配对样本 t 检验上 p-value<0.05。“*”表示与最好的基线模型相比，AITM 在配对样本 t 检验上 p-value<0.01

Model	Gain	
	approval	*activation*
MLP vs LightGBM	+16.95%	+17.55%
AITM vs MLP	+25.00%	+42.11%

表 3　在线 A/B 实验

我们进行了离线和在线实验。随着业务发展，我们先后部署了 LightGBM、MLP 和 AITM 3 个模型到线上。离线和在线实验都证明了 AITM 的显著表现。

5.3 消融实验

我们设计了 AIT 模块来建模用户多步转化过程中的序列依赖，为了证明 AIT 模块的有效性，我们对 AIT 模块进行了消融研究。我们先分别随机取了 500 个激活正、负样本，然后对它们的激活任务预测分进行排序。越 Top 的预测分，表明模型预测这些用户越容易激活联名卡。然后我们利用 t-SNE 在激活任务上画出了原始信息 $\boldsymbol{q}_t$，迁移 + 原始信息 $\boldsymbol{p}_{t-1}+\boldsymbol{q}_t$，以及 AIT 模块学习到的信息 $\boldsymbol{z}_t$ 的二维图。

从下图中，我们可以看到当用户激活比较容易预测时（即激活分 Top0%-Top50%），三个模块都能够较好地区分正负样本。但是，当用户激活比较难预测时（即激活分 Top50%-Top100%），原始信息 $\boldsymbol{q}_t$、迁移 + 原始信息 $\boldsymbol{p}_{t-1}+\boldsymbol{q}_t$ 都不能很好地区分正负样本，而 AIT 模块在 Top50%-Top100% 预测分上显著优于其他两个部分，说明了 AIT 模块的有效性。

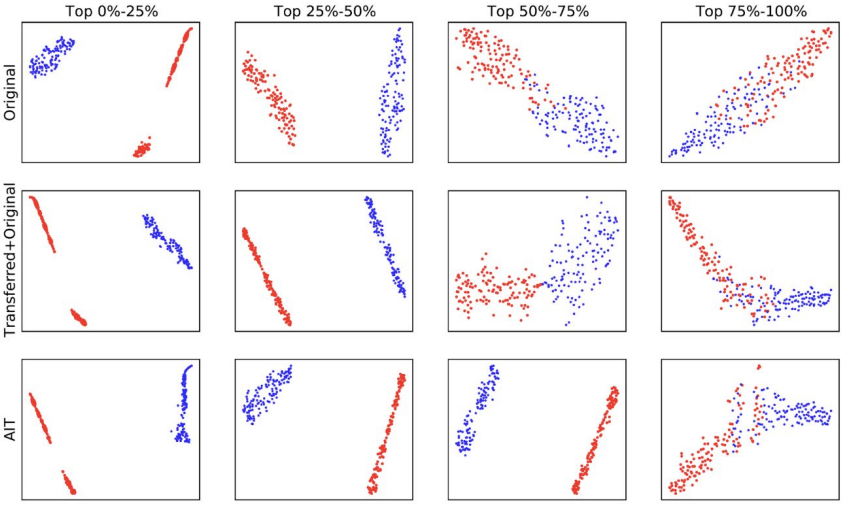


图 4　AIT 中不同模块的 t-SNE 可视化

5.4 超参数实验

另外，为了研究 AITM 框架的稳定性以及对超参数的依赖，我们进行了超参数研究。我们通过对 Embedding 向量维度、校准器强度、正样本比例、任务数量取不同的值进行多次实验，我们有以下发现：

- 前两个超参数实验表明，AITM 模块在 Embedding 向量维度、校准器强度这些超参数下都表现比较稳定。
- 第三个超参数实验表明合适的下采样比例是缓解类别不平衡的一个重要举措。
- 最后一个图显示了更多的任务可以带来更多有用的信息，从而进一步提升端到端转化率。

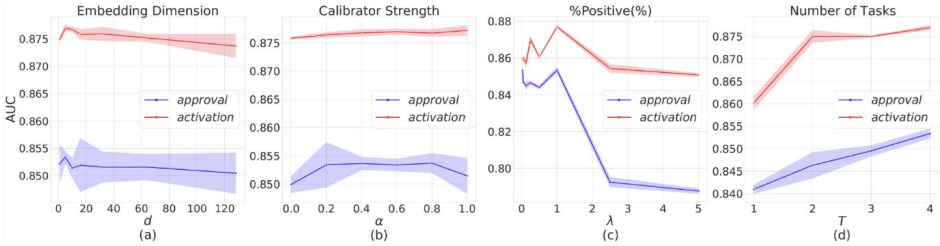


图 5　不同超参数设置下的平均 AUC 表现，阴影代表多次实验的标准差

5.5 Case Study

最后，为了理解 AIT 模块在不同的阶段迁移了什么信息以及多少信息，我们进行了 Case Study。下图中 W_uWu 是迁移信息的权重。我们先随机采样 4 万个测试样本，然后将它们按照相邻两个任务的 Label 分成 3 组：00/10/11（分别对应图中红色、蓝色、绿色的线），并且在每个组中根据 Logloss 对 Top 500 个样本进行排序，越 Top 的样本表示预测越准。

- 首先，看红色的线，由于存在序列依赖，当前一个任务 Label 是 0 时，后一个任务 Label 也只能是 0。所以我们看到前一个任务迁移了很强的信息给后一个任务（权重几乎为 1）。
- 然后，看蓝色和绿色的线，当前一个任务 Label 是 1 时，后一个任务 Label

是不确定的。随着 Logloss 的增加，迁移信息的权重会逐渐增大，这表明后一个任务的预测结果可能会被前一个任务所误导。

- 另外，看绿色的线，当后一个任务 Label 是 1 时，前一个任务迁移了很少的信息（权重很小），这表明后一个任务主要根据自己任务本身来识别正样本。
- 从以上的结果中，我们可以看到 AIT 模块能学习到两个相邻的任务间应该迁移多少信息。

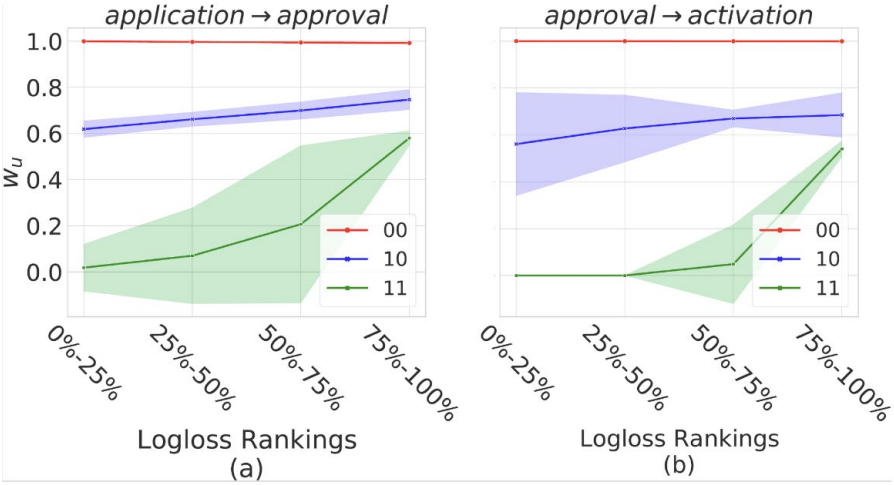


图 6　不同转化阶段的迁移信息平均权重，阴影代表标准差

六、总结

本文提出了一种自适应信息迁移多任务（AITM）框架来建模用户多步转化之间的序列依赖关系。本文提出的自适应信息迁移（AIT）模块配合行为期望校准器，可以学习在不同的转化阶段需要迁移什么和迁移多少信息，以提高序列依赖多任务学习的表现。离线和在线实验结果表明，与最新的基线模型相比该框架有显著的提升。该框架被部署在美团 App 中，利用它来为对美团联名卡有高转化率的用户实时展示联名卡广告。更多的技术细节，欢迎大家参考我们的论文。

七、作者简介

冬博、陈振、燕鹏、陈彧等，来自美团金融智能应用团队。
张颖而：美团实习生、浙江大学本科生
朱勇椿：中科院计算所直博生
庄福振：北京航空航天大学人工智能研究院研究员

招聘信息

美团金融智能应用团队算法岗位持续热招中，诚招优秀算法工程师及专家，坐标北京 / 上海。招聘岗位包括：

营销算法工程师 / 专家

- 服务美团金融各业务场景，负责营销获客、留存促活等场景的算法设计与开发，综合机器学习与优化技术，解决金融营销问题；
- 沉淀算法平台能力，提升算法应用的效率，提供客群挖掘、权益分配、素材匹配、动态创意、运筹规划、精准触达等智能解决方案；
- 结合美团金融业务场景，对深度学习、强化学习、知识图谱等人工智能前沿技术探索创新，实施创新技术沉淀和落地。

风控算法工程师 / 专家

- 通过机器学习模型与策略的开发优化，持续提升对于金融风险行为的识别能力；
- 深入理解业务，应用机器学习技术提高风控工作的自动化程度，全面提升业务效率；
- 跟进人工智能的前沿技术，并在金融风控场景中探索落地。

NLP 算法工程师 / 专家

- 基于美团金融业务场景，结合自然语言处理和机器学习相关技术，落地智能对话机器人到金融营销、风险管理、客服等多个场景；
- 参与研发对话机器人的相关项目，包括但不限于语义理解、多轮对话管理等相关算法的开发和优化；
- 持续跟进学术界和工业界相关技术的发展，并快速应用于项目中。

欢迎感兴趣的同学发送简历至：chenzhen06@meituan.com（邮件标题注明：美团金融智能应用团队）。

参考文献

[1] Jiaqi Ma, Zhe Zhao, Xinyang Yi, Jilin Chen, Lichan Hong, and Ed H Chi. 2018. Modeling task relationships in multi-task learning with multi-gate mixture-of-experts. In KDD. 1930 - 1939.

[2] Zhen Qin, Yicheng Cheng, Zhe Zhao, Zhe Chen, Donald Metzler, and Jingzheng Qin. 2020. Multitask Mixture of Sequential Experts for User Activity Streams. In KDD. 3083 – 3091.

[3] Hongyan Tang, Junning Liu, Ming Zhao, and Xudong Gong. 2020. Progressive Layered Extraction (PLE): A Novel Multi-Task Learning (MTL) Model for Personalized Recommendations. In RecSys. 269 – 278.

[4] Yixuan Li, Jason Yosinski, Jeff Clune, Hod Lipson, and John E Hopcroft. 2016. Convergent Learning: Do different neural networks learn the same representations? In ICLR.

[5] Matthew D Zeiler and Rob Fergus. 2014. Visualizing and understanding convolutional networks. In ECCV. 818 – 833.

[6] Eric Tzeng, Judy Hoffman, Ning Zhang, Kate Saenko, and Trevor Darrell. 2014. Deep domain confusion: Maximizing for domain invariance. arXiv preprint arXiv:1412.3474 (2014).

[7] Chen Gao, Xiangnan He, Dahua Gan, Xiangning Chen, Fuli Feng, Yong Li, Tat-Seng Chua, and Depeng Jin. 2019. Neural multi-task recommendation from multi-behavior data. In ICDE. 1554 – 1557.

[8] Chen Gao, Xiangnan He, Danhua Gan, Xiangning Chen, Fuli Feng, Yong Li, Tat-Seng Chua, Lina Yao, Yang Song, and Depeng Jin. 2019. Learning to Recommend with Multiple Cascading Behaviors. TKDE (2019).

[9] Xiao Ma, Liqin Zhao, Guan Huang, ZhiWang, Zelin Hu, Xiaoqiang Zhu, and Kun Gai. 2018. Entire space multi-task model: An effective approach for estimating post-click conversion rate. In SIGIR. 1137 – 1140.

[10] Hong Wen, Jing Zhang, Yuan Wang, Fuyu Lv, Wentian Bao, Quan Lin, and Keping Yang. 2020. Entire Space Multi-Task Modeling via Post-Click Behavior Decomposition for Conversion Rate Prediction. In SIGIR. 2377 – 2386.

[11] https://tianchi.aliyun.com/datalab/dataSet.html?dataId=408

ACL 2021 | 一文详解美团技术团队 7 篇精选论文

作者：可清　翔宇　恒通　会星等

计算语言学协会年会（ACL 2021）于 2021 年 8 月 1 日至 6 日在泰国曼谷举办（虚拟线上会议）。ACL 是计算语言学和自然语言处理领域最重要的顶级国际会议，该会议由国际计算语言学协会组织，每年举办一次。据谷歌学术计算语言学刊物指标显示，ACL 影响力位列第一，是 CCF-A 类推荐会议。今年 ACL 的主题是“NLP for Social Good”。据官方统计信息，本次会议共收到 3350 篇有效投稿，共计接收 710 篇主会论文（接受率为 21.3%），493 篇 Findings 论文（接受率为 14.9%）。

美团技术团队共有 7 篇论文（其中 6 篇长文，1 篇短文）被 ACL 2021 接收，这些论文是美团在事件抽取、实体识别、意图识别、新槽位发现、无监督句子表示、语义解析、文档检索等自然语言处理任务上的一些技术沉淀和应用。

针对于事件抽取，我们显示地利用周边实体的语义级别的论元角色信息，提出了一个双向实体级解码器（BERD）来逐步对每个实体生成论元角色序列；针对于实体识别，我们首次提出了槽间可迁移度的概念，并为此提出了一种槽间可迁移度的计算方式，通过比较目标槽与源任务槽的可迁移度，为不同的目标槽寻找相应的源任务槽作为其源槽，只基于这些源槽的训练数据来为目标槽构建槽填充模型；针对于意图识别，我们提出了一种基于监督对比学习的意图特征学习方法，通过最大化类间距离和最小化类内方差来提升意图之间的区分度；针对于新槽位发现，我们首次定义了新槽位识别（Novel Slot Detection, NSD）任务，与传统槽位识别任务不同的是，新槽位识别任务试图基于已有的域内槽位标注数据去挖掘发现真实对话数据里存在的新槽位，进而不断地完善和增强对话系统的能力。

此外，为解决 BERT 原生句子表示的“坍缩”现象，我们提出了基于对比学习的句子表示迁移方法—ConSERT，通过在目标领域的无监督语料上 Fine-Tune，使模型生成的句子表示与下游任务的数据分布更加适配。我们还提出了一种新的无监督的

语义解析方法——同步语义解码（SSD），它可以联合运用复述和语法约束解码同时解决语义鸿沟与结构鸿沟的问题。我们还从改进文档的编码入手来提高文档编码的语义表示能力，既提高了效果也提高了检索效率。

接下来，我们将对这 7 篇学术论文做一个更加详细的介绍，希望能对那些从事相关研究的同学有所帮助或启发，也欢迎大家在文末评论区留言，一起交流。

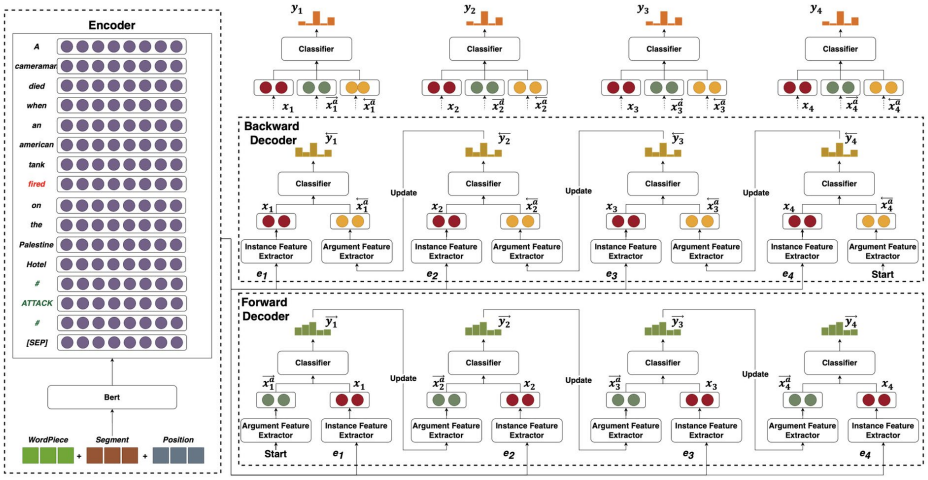
01 Capturing Event Argument Interaction via A Bi-Directional Entity-Level Recurrent Decoder

| 论文下载 | 论文作者：习翔宇，叶蔚（北京大学），张通（北京大学），张世琨（北京大学），王全修（RICHAI），江会星，武威 | 论文类型：Main Conference Long Paper（Oral）

事件抽取是信息抽取领域一个重要且富有挑战性的任务，在自动文摘、自动问答、信息检索、知识图谱构建等领域有着广泛的应用，旨在从非结构化的文本中抽取出结构化的事件信息。事件论元抽取对具体事件的描述信息（称之为论元信息）进行抽取，包括事件参与者、事件属性等信息，是事件抽取中重要且难度极大的任务。绝大部分论元抽取方法通常将论元抽取建模为针对实体和相关事件的论元角色分类任务，并且针对一个句子中实体集合的每个实体进行分离地训练与测试，忽略了候选论元之间潜在的交互关系；而部分利用了论元交互信息的方法，都未充分利用周边实体的语义级别的论元角色信息，同时忽略了在特定事件中的多论元分布模式。

针对目前事件论元检测中存在的问题，本文提出显示地利用周边实体的语义级别的论元角色信息。为此，本文首先将论元检测建模为实体级别的解码问题，给定句子和已知事件，论元检测模型需要生成论元角色序列；同时与传统的词级别的 Seq2Seq 模型不同，本文提出了一个双向实体级解码器（BERD）来逐步对每个实体生成论元角色序列。具体来说，本文设计了实体级别的解码循环单元，能够同时利用当前实例信息和周边论元信息；并同时采用了前向和后向解码器，能够分别从左往右和从右往左地对当前实体进行预测，并在单向解码过程中利用到左侧 / 右侧的论元信息；最终，本文在两个方向解码完成之后，采用了一个分类器结合双向编码器的特征来进行最终

预测，从而能够同时利用左右两侧的论元信息。



本文在公开数据集 ACE 2005 上进行了实验，并与多种已有模型以及最新的论元交互方法进行对比。实验结果表明该方法性能优于现有的论元交互方法，同时在实体数量较多的事件中提升效果更加显著。

Model	P	R	F1
DMBERT	56.9	57.4	57.2
HMEAE	**62.2**	56.6	59.3
BERT(Inter)	58.4	57.1	57.8
BERT(Intra)	56.4	61.2	58.7
BERD	59.1	**61.5**	**60.3**

Table 2: Overall performance on ACE 2005 (%).

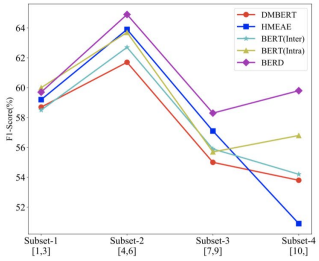


Figure 2: Comparison on four subsets with different range of entity numbers. F1-Score(%) is listed.

02 Slot Transferability for Cross-domain Slot Filling

| 论文下载 | 论文作者：陆恒通（北京邮电大学），韩卓芯（北京邮电大学），袁彩霞（北京邮电大学），王小捷（北京邮电大学），雷书彧，江会星，武威 | 论文类型：Findings of ACL 2021, Long Paper

槽填充旨在识别用户话语中任务相关的槽信息，是任务型对话系统的关键部分。当某

个任务（或称为领域）具有较多训练数据时，已有的槽填充模型可以获得较好的识别性能。但是，对于一个新任务，往往只有很少甚至没有槽标注语料，如何利用一个或多个已有任务（源任务）的标注语料来训练新任务（目标任务）中的槽填充模型，这对于任务型对话系统应用的快速扩展有着重要的意义。

针对该问题的现有研究主要分为两种，第一种通过建立源任务槽信息表示与目标任务槽信息表示之间的隐式语义对齐，来将用源任务数据训练的模型直接用于目标任务，这些方法将槽描述、槽值样本等包含槽信息的内容与词表示以一定方式进行交互得到槽相关的词表示，之后进行基于“BIO”的槽标注。第二种思路采用两阶段策略进行，将所有槽值看作实体，首先用源任务数据训练一个通用实体识别模型识别目标任务所有候选槽值，之后将候选槽值通过与目标任务槽信息的表示进行相似度对比来分类到目标任务的槽上。

现有的工作，大多关注于构建利用源 – 目标任务之间关联信息的跨任务迁移模型，模型构建时一般使用所有源任务的数据。但是，实际上，并不是所有的源任务数据都会对目标任务的槽识别具有可迁移的价值，或者不同源任务数据对于特定目标任务的价值可能是很不相同的。例如：机票预定任务和火车票预定任务相似度高，前者的槽填充训练数据会对后者具有帮助，而机票预定任务和天气查询任务则差异较大，前者的训练数据对后者没有或只具有很小的借鉴价值，甚至起到干扰作用。

再进一步，即使源任务和目标任务很相似，但是并不是每个源任务的槽的训练数据都会对目标任务的所有槽都有帮助，例如，机票预定任务的出发时间槽训练数据可能对火车票预定任务的出发时间槽填充有帮助，但是对火车类型槽就没有帮助，反而起到干扰作用。因此，我们希望可以为目标任务中的每一个槽找到能提供有效迁移信息的一个或多个源任务槽，基于这些槽的训练数据构建跨任务迁移模型，可以更为有效地利用源任务数据。

为此，我们首先提出了槽间可迁移度的概念，并为此提出了一种槽间可迁移度的计算方式，基于可迁移度的计算，我们提出了一种为目标任务选择出源任务中能够提供有效迁移信息的槽的方法。通过比较目标槽与源任务槽的可迁移度，为不同的目标槽寻

找相应的源任务槽作为其源槽，只基于这些源槽的训练数据来为目标槽构建槽填充模型。具体来说，可迁移度融合了目标槽和源槽之间的槽值表示分布相似度，以及槽值上下文表示分布相似度作为两个槽之间的可迁移度，然后对源任务槽依据其与目标槽之间的可迁移度高低进行排序，用可迁移度最高的槽所对应训练语料训练一个槽填充模型，得到其在目标槽验证集上的性能，依据按照可迁移度排序加入新的源任务槽对应训练语料训练模型并得到对应的验证集性能，选取性能最高的点对应的源任务槽及可迁移度高于该槽的源任务槽作为其源槽。利用选择出来的源槽构建目标槽槽填充模型。

槽填充模型依据槽值信息及槽值的上下文信息对槽值进行识别，所以我们在计算槽间可迁移度时，首先对槽值表示分布与上下文表示分布上的相似性进行了度量，然后我们借鉴了 F 值对于准确率及召回率的融合方式，对槽值表示分布相似性及槽值上下文表示分布相似性进行了融合，最后利用 Tanh 将所得到的值归一化到 0–1 之间，再用 1 减去所得到的值，为了符合计算得到的值越大，可迁移度越高的直观认知。下式是我们所提出的槽间可迁移度的计算方式：

$$\mathrm{STM}_{\beta}(s_a, s_b) = 1 - \tanh\left(\frac{(1+\beta^2)sim(p_v(s_a), p_v(s_b))sim(p_c(s_a), p_c(s_b))}{\beta^2 sim(p_v(s_a), p_v(s_b)) + sim(p_c(s_a), p_c(s_b))}\right)$$

$sim(p_v(s_a), p_v(s_b))$ 和 $sim(p_c(s_a), p_c(s_b))$ 分别表示槽 a 与槽 b 在槽值表示分布与上下文表示分布上的相似性，我们采用最大均值差异（MMD）来衡量分布之间的相似度。

我们并没有提出新的模型，但是我们提出的源槽选择方法可以与所有的已知模型进行结合，在多个已有模型及数据集上的实验表明，我们提出的方法能为目标任务槽填充模型带来一致性的性能提升（ALL 所在列表示已有模型原始的性能，STM1 所在列表示用我们的方法选出的数据训练的模型性能。）

Model		CT		RZT		Coach		Coach+TR	
Data/Domain ↓ Training Setting →		ALL	STM_1	ALL	STM_1	ALL	STM_1	ALL	STM_1
MWS	attraction	74.52	**84.99**	74.96	**84.22**	67.26	**73.35**	65.06	**76.16**
	hotel	58.81	47.73	50.63	43.82	59.03	**61.12**	59.00	**60.74**
	restaurant	69.93	63.47	66.29	61.01	78.65	65.36	79.00	71.53
	taxi	51.61	**69.32**	52.82	**66.25**	63.88	**81.17**	70.04	**79.34**
	train	80.78	79.66	80.02	**81.37**	77.68	**82.85**	77.91	**85.23**
	Average F1	67.13	**69.03**	64.94	**67.33**	69.30	**72.77**	70.20	**74.60**
SNIPS	AddToPlaylist	38.82	**41.95**	42.77	**42.92**	45.23	**53.36**	50.90	50.54
	BookRestaurant	27.54	**31.17**	30.68	30.21	33.45	32.60	34.01	32.89
	GetWeather	46.45	**53.03**	50.28	**62.32**	47.93	**60.91**	50.47	**62.38**
	PlayMusic	32.86	23.09	33.12	22.33	28.89	**35.60**	32.01	**34.45**
	RateBook	14.54	**15.39**	16.43	**25.37**	25.67	16.37	22.06	**25.39**
	SearchCreativeWork	39.79	38.72	44.45	42.63	43.91	**49.88**	46.65	**52.21**
	SearchScreeningEvent	13.83	**14.13**	12.25	**15.15**	25.64	23.75	25.63	**26.05**
	Average F1	30.55	**31.07**	32.85	**34.42**	35.82	**38.92**	37.39	**40.56**

03 Modeling Discriminative Representations for Out-of-Domain Detection with Supervised Contrastive Learning

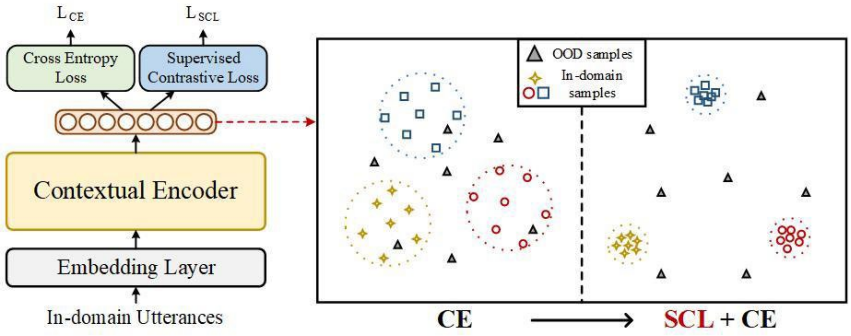
| 论文下载 | 论文作者：曾致远（北京邮电大学），何可清，严渊蒙（北京邮电大学），刘子君（北京邮电大学），吴亚楠（北京邮电大学），徐红（北京邮电大学），江会星，徐蔚然（北京邮电大学）| 论文类型：Main Conference Short Paper (Poster)

在实际的任务型对话系统中，异常意图检测（Out-of-Domain Detection）是一个关键的环节，其负责识别用户输入的异常查询，并给出拒识的回复。与传统的意图识别任务相比，异常意图检测面临着语义空间稀疏、标注数据匮乏的难题。现有的异常意图检测方法可以分为两类：一类是有监督的异常意图检测，是指训练过程中存在有监督的 OOD 意图数据，此类方法的优势是检测效果较好，但缺点是依赖于大量有标注的 OOD 数据，这在实际中并不可行。另一类是无监督的异常意图检测，是指仅仅利用域内的意图数据去识别域外意图样本，由于无法利用有标注 OOD 样本的先验知识，无监督的异常意图检测方法面临着更大的挑战。因此，本文主要是研究无监督的异常意图检测。

无监督异常意图检测的一个核心问题是，如何通过域内意图数据学习有区分度的语义表征，我们希望同一个意图类别下的样本表征互相接近，同时不同意图类别下的样本互相远离。基于此，本文提出了一种基于监督对比学习的意图特征学习方法，通过最

大化类间距离和最小化类内方差来提升特征的区分度。

具体来说，我们使用一个 BiLSTM/BERT 的上下文编码器获取域内意图表示，然后针对意图表示使用了两种不同的目标函数：一种是传统的分类交叉熵损失，另一种是监督对比学习（Supervised Contrastive Learning）损失。监督对比学习是在对比学习的基础上，改进了原始的对比学习仅有一个 Positive Anchor 的缺点，使用同类样本互相作为正样本，不同类样本作为负样本，最大化正样本之间的相关性。同时，为了提高样本表示的多样性，我们使用对抗攻击的方法来进行虚拟数据增强（Adversarial Augmentation），通过给隐空间增加噪声的方式来达到类似字符替换、插入删除、回译等传统数据增强的效果。模型结构如下：



a) The in-domain intent classifier　　b) Using intent representations for OOD detection

我们在两个公开的数据集上验证模型的效果，实验结果表明我们提出的方法可以有效的提升无监督异常意图检测的性能，如下表所示。

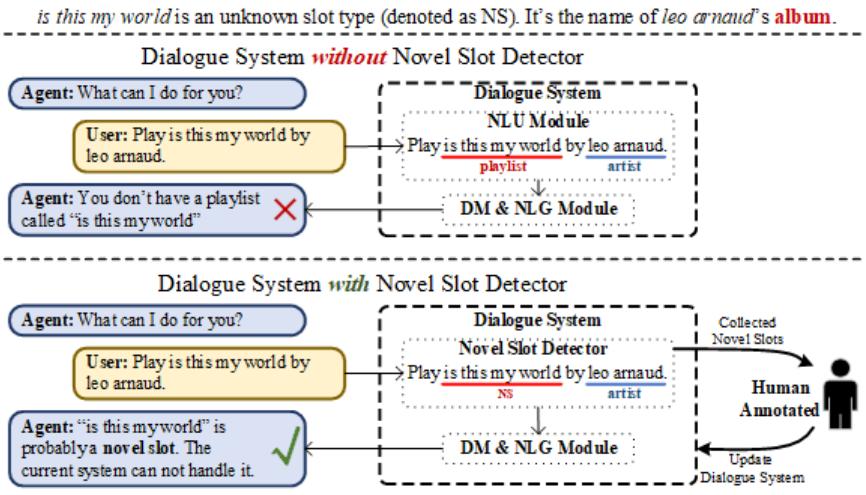
Models		CLINC-Full				CLINC-Small			
		IND		OOD		IND		OOD	
		ACC	F1	Recall	F1	ACC	F1	Recall	F1
LSTM	CE	86.34	87.73	63.72	65.23	84.24	84.30	60.40	61.07
	LMCL	86.83	87.90	64.14	65.79	84.46	84.87	60.72	61.89
	SCL+CE(ours)	87.01	88.28	66.80	67.68	85.73	86.61	63.96	64.44
	SCL+LMCL(ours)	**87.37**	**88.60**	**66.92**	**68.04**	**85.93**	**87.02**	**64.16**	**64.70**
BERT	CE	88.13	88.98	64.24	66.17	86.68	86.20	61.64	62.58
	LMCL	88.57	89.12	64.76	66.80	86.76	86.64	62.20	63.11
	SCL+CE(ours)	88.97	89.57	66.84	68.03	87.65	88.07	64.44	64.52
	SCL+LMCL(ours)	**89.20**	**90.03**	**67.28**	**68.21**	**87.87**	**88.30**	**64.64**	**65.01**

04 Novel Slot Detection: A Benchmark for Discovering Unknown Slot Types in the Task-Oriented Dialogue System

| 论文下载 | 论文作者：吴亚楠（北京邮电大学），曾致远（北京邮电大学），何可清，徐红（北京邮电大学），严渊蒙（北京邮电大学），江会星，徐蔚然（北京邮电大学）| 论文类型：Main Conference Long Paper（Oral）

槽填充（Slot Filling）是对话系统中一个重要的模块，负责识别用户输入中的关键信息。现有的槽填充模型只能识别预先定义好的槽类型，但是实际应用里存在大量域外实体类型，这些未识别的实体类型对于对话系统的优化至关重要。

在本文中，我们首次定义了新槽位识别（Novel Slot Detection, NSD）任务，与传统槽位识别任务不同的是，新槽位识别任务试图基于已有的域内槽位标注数据去挖掘发现真实对话数据里存在的新槽位，进而不断地完善和增强对话系统的能力，如下图所示：

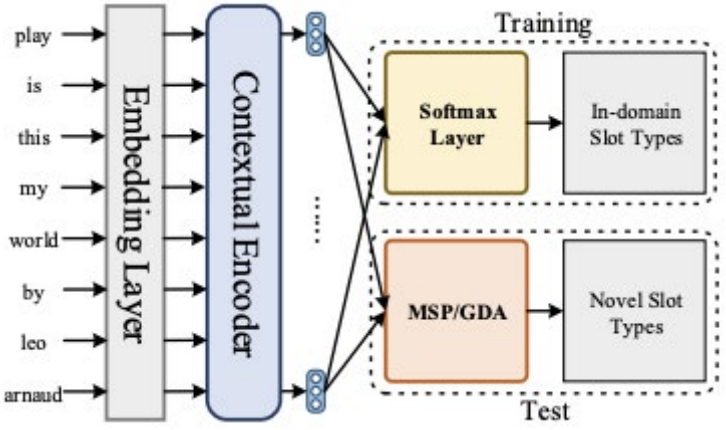


对比现有的 OOV 识别任务和域外意图检测任务，本文提出的 NSD 任务具有显著的差异性：一方面，与 OOV 识别任务相比，OOV 识别的对象是训练集中未出现过的新槽值，但这些槽值所属的实体类型是固定的，而 NSD 任务不仅要处理 OOV 的问

题，更严峻的挑战是缺乏未知实体类型的先验知识，仅仅依赖域内槽位信息来推理域外实体信息；另一方面，和域外意图检测任务相比，域外意图检测仅需识别句子级别的意图信息，而 NSD 任务则面临着域内实体和域外实体之间上下文的影响，以及非实体词对于新槽位的干扰。整体上来看，本文提出的新槽位识别（Novel Slot Detection, NSD）任务与传统的槽填充任务、OOV 识别任务以及域外意图检测任务有很大的差异，并且面临着更多的挑战，同时也给对话系统未来的发展提供了一个值得思考和研究的方向。

基于现有的槽填充公开数据集 ATIS 和 Snips，我们构建了两个新槽位识别数据集 ATIS-NSD 和 Snips-NSD。具体来说，我们随机抽取训练集中部分的槽位类型作为域外类别，保留其余类型作为域内类别，针对于一个句子中同时出现域外类别和域内类别的样例，我们采用了直接删除整个样本的策略，以避免 O 标签引入的 bias，保证域外实体的信息仅仅出现在测试集中，更加的贴近实际场景。同时，我们针对于 NSD 任务提出了一系列的基线模型，整体的框架如下图所示。模型包含两个阶段：

- **训练阶段**：基于域内的槽标注数据，我们训练一个 BERT-based 的序列标注模型（多分类或者是二分类），以获取实体表征。
- **测试阶段**：首先使用训练的序列标注模型进行域内实体类型的预测，同时基于得到的实体表征，使用 MSP 或者 GDA 算法预测一个词是否属于 Novel Slot，也即域外类型，最后将两种输出结果进行合并得到最终的输出。



我们使用实体识别的 F1 作为评价指标，包括 Span-F1 和 Token-F1，二者的区别在于是否考虑实体边界，实验结果如下：

Models			5%			15%			30%		
			IND	NSD		IND	NSD		IND	NSD	
detection method	objective	distance strategy	Span F1	Span F1	Token F1	Span F1	Span F1	Token F1	Span F1	Span F1	Token F1
MSP	binary	-	87.21	12.34	25.16	71.44	12.31	39.50	58.88	8.73	40.38
	multiple	-	88.05	14.04	30.50	79.71	20.97	40.02	78.52	25.26	46.91
	binary+multiple	-	89.59	23.58	37.55	83.72	24.70	45.32	79.08	30.66	52.10
GDA	binary	difference	87.95	23.83	35.83	83.65	22.06	43.99	78.72	32.50	44.13
	binary	minimum	61.29	10.36	17.08	49.11	16.91	31.10	48.07	15.56	33.78
	multiple	difference	**93.14**	29.73	45.99	90.07	31.96	53.02	85.56	36.16	54.55
	multiple	minimum	93.10	**31.67***	**46.97***	**90.18**	**32.19**	**53.75***	**86.26***	**38.64***	**55.24***

我们通过大量的实验和分析来探讨新槽位识别面临的挑战：1. 非实体词与新实体之间混淆；2. 不充分的上下文信息；3. 槽位之间的依赖关系；4. 开放槽（Open Vocabulary Slots）。

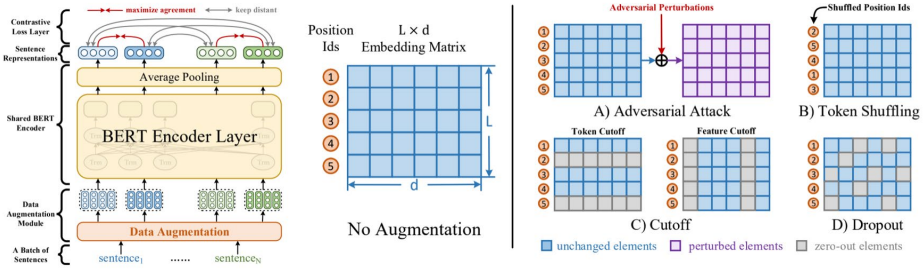
05 ConSERT: A Contrastive Framework for Self-Supervised Sentence Representation Transfer

| 论文下载 | 论文作者：严渊蒙，李如寐，王思睿，张富峥，武威，徐蔚然（北京邮电大学）| 论文类型：Main Conference Long Paper（Poster）

句向量表示学习在自然语言处理（NLP）领域占据重要地位，许多 NLP 任务的成功离不开训练优质的句子表示向量。特别是在文本语义匹配（Semantic Textual Similarity）、文本向量检索（Dense Text Retrieval）等任务上，模型通过计算两个句子编码后的 embedding 在表示空间的相似度来衡量这两个句子语义上的相关程度，从而决定其匹配分数。尽管基于 BERT 的模型在诸多 NLP 任务上取得了不错的性能（通过有监督的 Fine-Tune），但其自身导出的句向量（不经过 Fine-Tune，对所有词向量求平均）质量较低，甚至比不上 Glove 的结果，因而难以反映出两个句子的语义相似度。

为解决 BERT 原生句子表示这种“坍缩”现象，本文提出了基于对比学习的句子表示迁移方法—ConSERT，通过在目标领域的无监督语料上 fine-tune，使模型生成的句子表示与下游任务的数据分布更加适配。同时，本文针对 NLP 任务提出了

对抗攻击、打乱词序、裁剪、Dropout 四种不同的数据增强方法。在句子语义匹配（STS）任务的实验结果显示，同等设置下 ConSERT 相比此前的 SOTA（BERT-Flow）大幅提升了 8%，并且在少样本场景下仍表现出较强的性能提升。



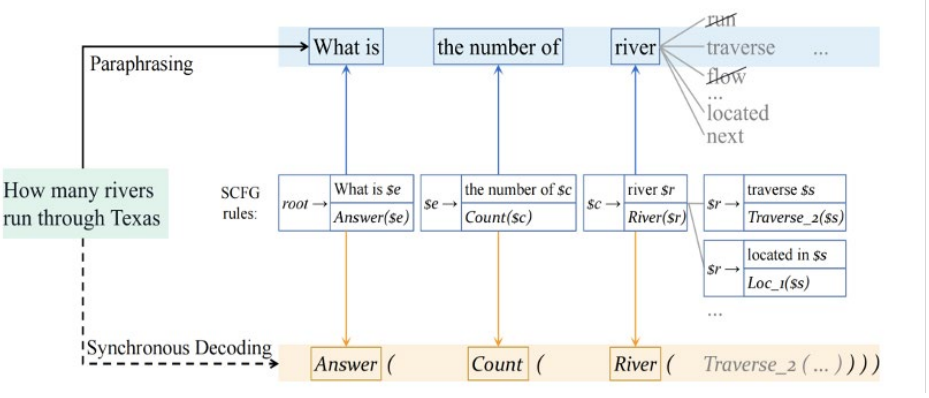
在无监督实验中，我们直接基于预训练的 BERT 在无标注的 STS 数据上进行 Fine-Tune。结果显示，我们的方法在完全一致的设置下大幅度超过之前的 SOTA—BERT-Flow，达到了 8% 的相对性能提升。

Method	STS12	STS13	STS14	STS15	STS16	STSb	SICK-R	Avg.
Unsupervised baselines								
Avg. GloVe embeddings†	55.14	70.66	59.73	68.25	63.66	58.02	53.76	61.32
$BERT_{base}$‡	35.20	59.53	49.37	63.39	62.73	48.18	58.60	53.86
$BERT_{large}$‡	33.06	57.64	47.95	55.83	62.42	49.66	53.87	51.49
$CLEAR_{base}$†	49.0	48.9	57.4	63.6	65.6	75.6	72.5	61.8
IS-$BERT_{base}$-NLI†	56.77	69.24	61.21	75.23	70.16	69.21	64.25	66.58
$BERT_{base}$-CT†	66.86	70.91	72.37	78.55	77.78	-	-	-
$BERT_{large}$-CT†	69.50	75.97	74.22	78.83	78.92	-	-	-
Using STS unlabeled texts								
$BERT_{base}$-flow†	63.48	72.14	68.42	73.77	75.37	70.72	63.11	69.57
$BERT_{large}$-flow†	65.20	73.39	69.42	74.92	**77.63**	72.26	62.50	70.76
$ConSERT_{base}$‡	64.64	78.49	69.07	79.72	75.95	73.97	67.31	72.74
$ConSERT_{large}$‡	**70.69**	**82.96**	**74.13**	**82.78**	76.66	**77.53**	**70.37**	**76.45**

06 From Paraphrasing to Semantic Parsing: Unsupervised Semantic Parsing via Synchronous Semantic Decoding

| 论文下载 | 论文作者：吴杉（中科院软件所），陈波（中科院软件所），辛春蕾（中科院软件所），韩先培（中科院软件所），孙乐（中科院软件所），张伟鹏，陈见耸，杨帆，蔡勋梁 | 论文类型：Main Conference Long Paper

语义解析（Semantic Parsing）是自然语言处理中的核心任务之一，它的目标是把自然语言转换为计算机语言，从而使得计算机真正理解自然语言。目前语义解析面临的一大挑战是标注数据的缺乏。神经网络方法大都十分依赖监督数据，而语义解析的数据标注非常费时费力。因此，如何在无监督的情况下学习语义解析模型成为非常重要的问题，同时也是有挑战性的问题，它的挑战在于，语义解析需要在无标注数据的情况下，同时跨越自然语言和语义表示间的语义鸿沟和结构鸿沟。之前的方法一般使用复述作为重排序或者重写方法以减少语义上的鸿沟。与之前的方法不同，我们提出了一种新的无监督的语义解析方法——同步语义解码（SSD），它可以联合运用复述和语法约束解码同时解决语义鸿沟与结构鸿沟。



语义同步解码的核心思想是将语义解析转换为复述问题。我们将句子复述成标准句式，同时解析出语义表示。其中，标准句式和逻辑表达式存在一一对应关系。为了保证生成有效的标准句式和语义表示，标准句式和语义表示在同步文法的限制中解码生成。

我们通过复述模型在受限的同步文法上解码，利用文本生成模型对标准句式的打分，找到得分最高的标准句式（如上所述，空间同时受文法限制）。本文给出了两种不同的算法：Rule-Level Inference 以语法规则为搜索单元和 Word-Level Inference 使用词作为搜索单元。

我们使用 GPT2.0 和 T5 在复述数据集上训练序列到序列的复述模型，之后只需要使用同步语义解码算法就可以完成语义解析任务。为了减少风格偏差影响标准句式的生成，我们提出了适应性预训练和句子重排序方法。

Unsupervised									
Cross-domain Zero Shot	-	28.3	53.6	52.4	55.3	60.2	61.7	-	-
GENOVERNIGHT	15.6	27.7	17.3	45.9	46.7	26.3	61.3	9.7	31.3
SYNTH-SEQ2SEQ	16.1	23.6	16.1	30.2	36.6	26.9	43.1	9.2	25.2
SYNTHPARA-SEQ2SEQ	28.4	37.3	33.9	38.1	39.1	41.7	62.7	23.3	38.1
SSD (Word-Level)	68.3	54.9	51.2	55.0	54.7	60.2	65.4	33.6	55.4
SSD (Grammar-Level)	68.8	58.1	56.5	56.1	57.8	59.3	66.9	37.1	**57.6**

我们在三个数据集上进行了实验：Overnight（λ-DCS）、GEO（FunQL）和 GEOGranno。数据覆盖不同的领域和语义表示。实验结果表明，在不使用有监督语义解析数据的情况下，我们的模型在各数据集上均能取得最好的效果。

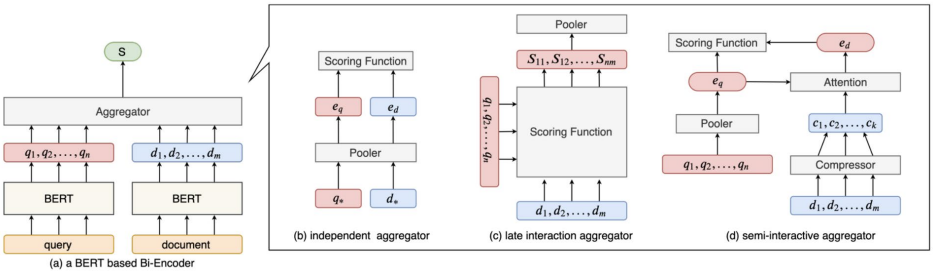
07 Improving Document Representations by Generating Pseudo Query Embeddings for Dense Retrieval

| 论文下载 | 论文作者：唐弘胤，孙兴武，金蓓弘（中科院软件所），王金刚，张富峥，武威 | 论文类型：Main Conference Long Paper（Oral）

文档检索任务的目标是在海量的文本库中检索出和给定查询语义近似的文本。在实际场景应用中，文档文档库的数量会非常庞大，为了提高检索效率，检索任务一般会分成两个阶段，即初筛和精排阶段。在初筛阶段中，模型通过一些检索效率高的方法筛选出一部分候选文档，作为后续精排阶段的输入。在精排阶段，模型使用高精度排序方法来对候选文档进行排序，得到最终的检索结果。

随着预训练模型的发展和应用，很多工作开始将查询和文档同时送入预训练进行编码，并输出匹配分数。然而，由于预训练模型的计算复杂度较高，对每个查询和文档都进行一次计算耗时较长，这种应用方式通常只能在精排阶段使用。为了加快检索速率，一些工作开始使用预训练模型单独编码文档和查询，在查询前提前将文档库中的文档编码成向量形式，在查询阶段，仅需利用查询编码和文档编码进行相似度计算，减少了时间消耗。由于这种方式会将文档和查询编码为稠密向量形式，因此这种检索也称作“稠密检索”（Dense Retrival）。

一个基本的稠密检索方法会将文档和查询编码成为一个向量。然而由于文档包含的信息较多，容易造成信息丢失。为了改进这一点，有些工作开始对查询和文档的向量表示进行改进，目前已有的改进方法大致可分为三种，如下图所示：



我们的工作从改进文档的编码入手来提高文档编码的语义表示能力。首先，我们认为稠密检索的主要瓶颈在于编码时，文档编码器并不知道文档中的哪部分信息可能会被查询，在编码过程中，很可能造成不同的信息互相影响，造成信息被改变或者丢失。因此，我们在编码文档的过程中，对每个文档构建了多个“伪查询向量”（Pseudo Query Embeddings），每个伪查询向量对应每个文档可能被提问的信息。

具体而言，我们通过聚类算法，将 BERT 编码的 Token 向量进行聚类，对每个文档保留 Top-k 个聚类向量，这些向量包含了多个文档 Token 向量中的显著语义。另外，由于我们对每个文档保留多个伪查询向量，在相似度计算时可能造成效率降低。我们使用 Argmax 操作代替 Softmax，来提高相似度计算的效率。在多个大规模文档检索数据集的实验表明，我们的方法既可以提高效果也提高了检索效率。

Models	MRR@10	Recall@1k
DeepCT	24.3	91.0
docT5Query	27.7	94.7
RepBERT	30.4	94.3
ANCE(First-P)	33.0	95.9
ME-BERT	33.4	-
ME-BERT+BM25	33.8	-
ColBERT	36.0	96.8
Ours	34.5	96.4

Table 1: Results on MS MARCO passage ranking dev set.

Models	MRR@100	NDCG@10
ANCE(First-P)	37.2*	61.5
ME-BERT	33.2	-
ME-BERT+BM25	34.6	-
Ours	39.2	62.8

Table 2: Results on MS MARCO document ranking dev set(MRR@100) and TREC test set(NDCG@10) The value with * is obtained by the public available code and checkpoint in https://github.com/microsoft/ANCE

Operation	offline	online
Per Document BERT Forward	0.9ms	-
Per Document K-means	2.1ms	-
Per Document Encoding	2.3ms	-
Per Query BERT Forward	-	0.5ms
Retrieval	-	180ms
Retrieval(w/o optimization)	-	880ms
Retrieval(independent)	-	100ms
Retrieval(late interaction)	-	940ms

写在后面

以上这些论文是美团技术团队与各高校、科研机构通力合作，在事件抽取、实体识别、意图识别、新槽位发现、无监督句子表示、语义解析、文档检索等领域所做的一些科研工作。论文是我们在实际工作场景中遇到并解决具体问题的一种体现，希望对大家能够有所帮助或启发。

美团科研合作致力于搭建美团各部门与高校、科研机构、智库的合作桥梁和平台，依托美团丰富的业务场景、数据资源和真实的产业问题，开放创新，汇聚向上的力量，围绕人工智能、大数据、物联网、无人驾驶、运筹优化、数字经济、公共事务等领域，共同探索前沿科技和产业焦点宏观问题，促进产学研合作交流和成果转化，推动优秀人才培养。面向未来，我们期待能与更多高校和科研院所的老师和同学们进行合作，欢迎大家跟我们联系（meituan.oi@meituan.com）。

ACL 2021 | 美团提出基于对比学习的文本表示模型，效果相比 BERT-flow 提升 8%

作者：渊蒙　如寐　思睿

尽管基于 BERT 的模型在 NLP 诸多下游任务中取得了成功，直接从 BERT 导出的句向量表示往往被约束在一个很小的区域内，表现出很高的相似度，因而难以直接用于文本语义匹配。为解决 BERT 原生句子表示这种“坍缩”现象，美团 NLP 中心知识图谱团队提出了基于对比学习的句子表示迁移方法——ConSERT，通过在目标领域的无监督语料上 Fine-tune，使模型生成的句子表示与下游任务的数据分布更加适配。在句子语义匹配（STS）任务的实验结果显示，同等设置下 ConSERT 相比此前的 SOTA（BERT-flow）大幅提升了 8%，并且在少样本场景下仍表现出较强的性能提升。

- 论文：《ConSERT: A Contrastive Framework for Self-Supervised Sentence Representation Transfer》
- 会议：ACL 2021
- 下载链接：https://arxiv.org/abs/2105.11741
- 开源代码：https://github.com/yym6472/ConSERT

1. 背景

句向量表示学习在自然语言处理（NLP）领域占据重要地位，许多 NLP 任务的成功离不开训练优质的句子表示向量。特别是在文本语义匹配（Semantic Textual Similarity）、文本向量检索（Dense Text Retrieval）等任务上，模型通过计算两个句子编码后的 Embedding 在表示空间的相似度来衡量这两个句子语义上的相关程度，从而决定其匹配分数。

尽管基于 BERT 的模型在诸多 NLP 任务上取得了不错的性能（通过有监督的 Fine-tune），但其自身导出的句向量（不经过 Fine-tune，对所有词向量求平均）质量较低，甚至比不上 Glove 的结果，因而难以反映出两个句子的语义相似度[1][2][3][4]。我们在研

究的过程中进一步分析了 BERT 导出的句向量所具有的特性，证实了以下两点：

1. BERT 对所有的句子都倾向于编码到一个较小的空间区域内，这使得大多数的句子对都具有较高的相似度分数，即使是那些语义上完全无关的句子对（如图 1a 所示）。我们将此称为 BERT 句子表示的“坍缩（Collapse）”现象。

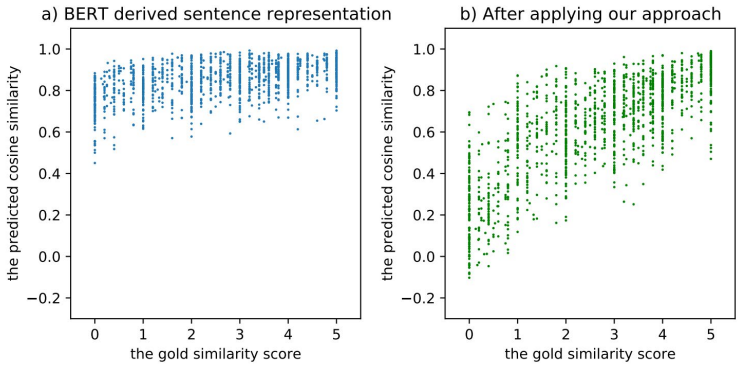


图 1　左：BERT 表示空间的坍缩问题（横坐标是人工标注的相似度分数，纵坐标是模型预测的余弦相似度）；右：经过我们的方法 Fine-tune 之后

2. BERT 句向量表示的坍缩和句子中的高频词有关。具体来说，当通过平均词向量的方式计算句向量时，那些高频词的词向量将会主导句向量，使之难以体现其原本的语义。当计算句向量时去除若干高频词时，坍缩现象可以在一定程度上得到缓解（如图 2 蓝色曲线所示）。

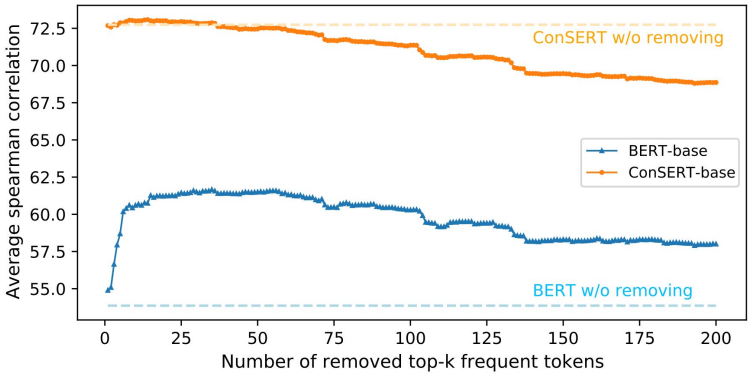


图 2　计算句向量时移除 Top-K 高频词后的性能变化

BERT 导出的句向量难以直接用于下游的语义匹配任务，而用于 Fine-tune 的监督语料又是昂贵的。因此我们希望寻找一种自监督的方法，只需要收集少量来自于下游任务无标注的文本用于 Fine-tune，就能解决 BERT 句向量的“坍缩”问题，同时让其表征更适用于下游任务。

在本文中，我们使用了对比学习（Contrastive Learning）来达到上述目的。对比学习是目前被广泛应用的自监督任务之一，其核心思想为：人类是通过“对比”来辨别对象的，因此相似的事物在编码后的表示空间中应当相近，不同的事物则应当相距尽可能远。通过对同一样本施加不同的数据增强方法，我们能够得到一系列“自相似”的文本对作为正例，同时将同一个 Batch 内的其他文本作为负例，以此为监督信号去规范 BERT 的表示空间。在实验中，我们发现对比学习能够出色地消解高频词对句子语义表示的干扰（如图 2 橙色曲线所示）。在经过对比学习训练之后，模型生成的句子表示将不再由高频词主导（体现在移除前几个高频词后，性能没有出现非常明显的变化）。这是因为对比学习“辨别自身”的学习目标能够天然地识别并抑制这类高频特征，从而避免语义相差较大的句子表示过于相近（即坍缩现象）。

在对比学习中，我们进一步分析了不同的数据增强方法在其中的影响，同时验证了我们的方法在少样本情况下的性能表现。实验结果显示，即使是在非常有限的数据量情况下（如 1000 条无标注样本），我们的方法仍然表现出很强的鲁棒性，能够十分有效地解决 BERT 表示空间的坍缩问题，提升在下游语义匹配任务上的指标。

2. 研究现状和相关工作

2.1 句子表征学习

句子表征学习是一个很经典的任务，分为以下三个阶段：

- 有监督的句子表征学习方法：早期的工作 [5] 发现自然语言推理（Natural Language Inference，NLI）任务对语义匹配任务有较大的帮助，他们使用 BiLSTM 编码器，融合了两个 NLI 的数据集 SNLI 和 MNLI 进行训练。Universal Sentence Encoder[6]（USE）使用了基于 Transformer 的架构，并使

用 SNLI 对无监督训练进行增强。SBERT[1] 进一步使用了一个共享的预训练的 BERT 编码器对两个句子进行编码，在 NLI 数据集上进行训练（Fine-tune）。

- 自监督的 Sentence-level 预训练：有监督数据标注成本高，研究者们开始寻找无监督的训练方式。BERT 提出了 NSP 的任务，可以算作是一种自监督的句子级预训练目标。尽管之后的工作指出 NSP 相比于 MLM 其实没有太大帮助。Cross-Thought[7]、CMLM[8] 是两种思想类似的预训练目标，他们把一段文章切成多个短句，然后通过相邻句子的编码去恢复当前句子中被 Mask 的 Token。相比于 MLM，额外添加了上下文其他句子的编码对 Token 恢复的帮助，因此更适合句子级别的训练。SLM[9] 通过将原本连贯的若干个短句打乱顺序（通过改变 Position Id 实现），然后通过预测正确的句子顺序进行自监督预训练。
- 无监督的句子表示迁移：预训练模型现已被普遍使用，然而 BERT 的 NSP 任务得到的表示表现更不好，大多数同学也没有资源去进行自监督预训练，因此将预训练模型的表示迁移到任务才是更有效的方式。BERT-flow[2]：CMU&字节 AI Lab 的工作，通过在 BERT 之上学习一个可逆的 Flow 变换，可以将 BERT 表示空间映射到规范化的标准高斯空间，然后在高斯空间进行相似度匹配。BERT-whitening[10]：苏剑林和我们同期的工作。他们提出对 BERT 表征进行白化操作（均值变为 0，协方差变为单位矩阵）就能在 STS 上达到媲美 BERT-flow 的效果。SimCSE[11]：陈丹琦组在 2021 年 4 月份公开的工作。他们同样使用基于对比学习的训练框架，使用 Dropout 的数据增强方法，在维基百科语料上 Fine-tune BERT。

2.2 对比学习

对比学习是 CV 领域从 2019 年末开始兴起的预训练方法，同时最近也被广泛应用到了 NLP 任务中，我们简要介绍两个领域下的进展：

- 计算机视觉（CV）领域的对比学习：2019 年年末 ~ 2020 年年初，Facebook 提出 MoCo[14]，谷歌提出 SimCLR[15]，自此对比学习开始在无监督图像表示

预训练领域大放光彩。SimCLR 提出了一种简单的对比学习框架，通过对同一个图像进行增强，得到两个不同版本，随后通过 ResNet 对图像编码，再使用一个映射层将其映射到对比学习空间，使用 NT-Xent 损失进行预训练。本文的框架也主要受到 SimCLR 的启发。

- NLP 领域的对比学习（用于文本表示学习）：随着对比学习在 CV 无监督图像表示预训练任务上大获成功，许多工作也试图将对比学习引入到 NLP 的语言模型预训练中。下面是一些代表性的工作及其总结：

名称	结构	数据增强方法	作用阶段
BERT-CT[18]	用两个相同结构、不同参数的模型产生两个 View	用两个相同结构、不同参数的模型产生两个 View	微调 BERT
IS-BERT[16]	在 BERT 之上添加额外的 CNN 层	全局 Embedding 作为一个 View；CNN 层局部的 Embedding 作为另一个 View	微调 BERT，最大化全局 Embedding 和局部 Embedding 的互信息，类似 DeepInfoMax（DIM）
CERT[17]	类似 MoCo，用一个动量编码器	回译	先在任务上用对比无监督精调，再迁到有监督任务
CLEAR[20]	类似 SimCLR	Token、Span 的删除（Deletion）、交换位置（Reordering）、替换（Substitution）	预训练阶段，用 MLM 和对比损失联合训练模型
DeCLUTR[19]	类似 SimCLR	从文本中抽取 Span	预训练阶段，用 MLM 和对比损失联合训练模型

3. 模型介绍

3.1 问题定义

给定一个类似 BERT 的预训练语言模型 $\mathbf{M}$，以及从目标领域数据分布中收集的无标签文本语料库 D，我们希望通过构建自监督任务在 D 上对 $\mathbf{M}$ 进行 Fine-tune，使得 Fine-tune 后的模型能够在目标任务（文本语义匹配）上表现最好。

3.2 基于对比学习的句子表示迁移框架

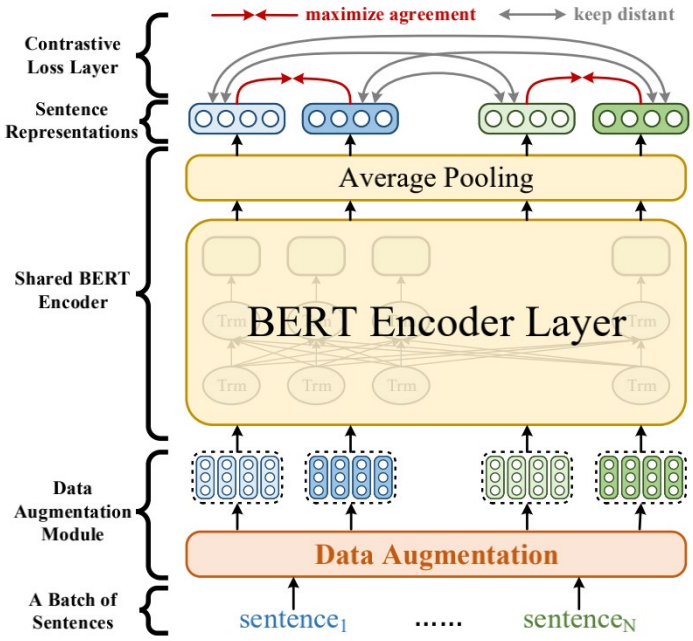


图 3　ConSERT 的基本框架

如图 3 所示，我们受到 SimCLR 的启发对 BERT 编码器进行了改进，提出 ConSERT，主要包含三个部分：

- 一个数据增强模块（详见后文），作用于 Embedding 层，为同一个句子生成两个不同的增强版本（View）。
- 一个共享的 BERT 编码器，为输入的句子生成句向量。
- 一个对比损失层，用于在一个 Batch 的样本中计算对比损失，其思想是最大化同一个样本不同增强版本句向量的相似度，同时使得不同样本的句向量相互远离。

训练时，先从数据集 D 中采样一个 Batch 的文本，设 Batch size 为 N。通过数据增强模块，每一个样本都通过两种预设的数据增强方法生成两个版本，得到总共 $2N$ 条样本。这 $2N$ 条样本均会通过共享的 BERT 编码器进行编码，然后通过一个平均池化层，得到 $2N$ 个句向量。我们采用和 SimCLR 一致的 NT-Xent 损失对模型进行 Fine-tune：

$$\mathcal{L}_{i,j} = -\log \frac{\exp(\text{sim}(r_i, r_j)/\tau)}{\sum_{k=1}^{2N} \mathbf{1}_{[k \neq i]} \exp(\text{sim}(r_i, r_k)/\tau)}$$

这里的 sim() 函数为余弦相似度函数；r 表示对应的句向量；τ 表示 temperature，是一个超参数，实验中取 0.1。该损失从直观上理解，是让 Batch 内的每个样本都找到其对应的另一个增强版本，而 Batch 内的其他 $2N-2$ 个样本将充当负样本。优化的结果就是让同一个样本的两个增强版本在表示空间中具有尽可能大的一致性，同时和其他的 Batch 内负样本相距尽可能远。

3.3 用于文本领域的数据增强方法探索

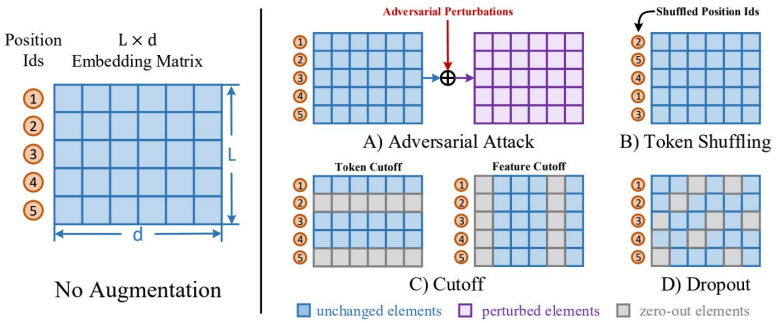


图 4　四种高效的数据增强方法：Adversarial Attack、Token Shuffling、Cutoff、Dropout，均作用于 Embedding 层

图像领域可以方便地对样本进行变换，如旋转、翻转、裁剪、去色、模糊等等，从而得到对应的增强版本。然而，由于语言天然的复杂性，很难找到高效的、同时又保留语义不变的数据增强方法。一些显式生成增强样本的方法包括：

- 回译：利用机器翻译模型，将文本翻译到另一个语言，再翻译回来。
- CBERT[12][13]：将文本的部分词替换成 [MASK]，然后利用 BERT 去恢复对应的词，生成增强句子。
- 意译（Paraphrase）：利用训练好的 Paraphrase 生成模型生成同义句。

然而这些方法一方面不一定能保证语义一致，另一方面每一次数据增强都需要做一次

模型 Inference，开销会很大。鉴于此，我们考虑了在 Embedding 层隐式生成增强样本的方法，如图 4 所示：

- **对抗攻击（Adversarial Attack）**：这一方法通过梯度反传生成对抗扰动，将该扰动加到原本的 Embedding 矩阵上，就能得到增强后的样本。由于生成对抗扰动需要梯度反传，因此这一数据增强方法仅适用于有监督训练的场景。
- **打乱词序（Token Shuffling）**：这一方法扰乱输入样本的词序。由于 Transformer 结构没有“位置”的概念，模型对 Token 位置的感知全靠 Embedding 中的 Position Ids 得到。因此在实现上，我们只需要将 Position Ids 进行 Shuffle 即可。
- **裁剪（Cutoff）**：又可以进一步分为两种：
  - Token Cutoff：随机选取 Token，将对应 Token 的 Embedding 整行置为零。
  - Feature Cutoff：随机选取 Embedding 的 Feature，将选取的 Feature 维度整列置为零。
- **Dropout**：Embedding 中的每一个元素都以一定概率置为零，与 Cutoff 不同的是，该方法并没有按行或者按列的约束。

这四种方法均可以方便地通过对 Embedding 矩阵（或是 BERT 的 Position Encoding）进行修改得到，因此相比显式生成增强文本的方法更为高效。

3.4 进一步融合监督信号

除了无监督训练以外，我们还提出了几种进一步融合监督信号的策略：

- 联合训练（joint）：有监督的损失和无监督的损失通过加权联合训练模型。
- 先有监督再无监督（sup-unsup）：先使用有监督损失训练模型，再使用无监督的方法进行表示迁移。
- 联合训练再无监督（joint-unsup）：先使用联合损失训练模型，再使用无监督的方法进行表示迁移。

4. 实验分析

我们主要在文本语义匹配（Semantic Textual Similarity，STS）任务上进行了实验，包括七个数据集：STS12、STS13、STS14、STS15、STS16、STSb、SICK-R。其中 STS12-16 为 SemEval2012 ~ 2016 评测比赛放出的数据集；STSb 为 STS benchmark，来自于 SemEval2017 评测赛；SICK-R 表示 SICK-Relatedness，是 SICK（Sentences Involving ComPositional Knowledge）数据集中的一个子任务，目标是推断两个句子时间的语义相关性（即 Relatedness）。这些数据集中的样本均包含两个短文本 text1 和 text2，以及人工标注的位于 0 ~ 5 之间的分数，代表 text1 和 text2 语义上的匹配程度（5 表示最匹配，即“两句话表达的是同一个语义”；0 表示最不匹配，即“两句话表达的语义完全不相关”）。下面给出了两条样本作为示例：

text1	text2	人工标注的语义相似度
a man is cutting paper with a sword .	a woman is cutting a tomato .	0.6
a woman is dancing in the rain .	a woman dances in the rain out side .	5.0

在测试时，我们根据此前的工作 [1][2] 选择了斯皮尔曼相关系数（Spearman correlation）作为评测指标，它将用于衡量两组值（模型预测的余弦相似度和人工标注的语义相似度）之间的相关性，结果将位于 [-1, 1] 之间，仅当两组值完全正相关时取到 1。对于每个数据集，我们将其测试样本全部融合计算该指标，并且报告了七个数据集的平均结果。考虑到简洁性，会在表格中报告乘以 100 倍的结果。

4.1 无监督实验

Method	STS12	STS13	STS14	STS15	STS16	STSb	SICK-R	Avg.
Unsupervised baselines								
Avg. GloVe embeddings†	55.14	70.66	59.73	68.25	63.66	58.02	53.76	61.32
$BERT_{base}$‡	35.20	59.53	49.37	63.39	62.73	48.18	58.60	53.86
$BERT_{large}$‡	33.06	57.64	47.95	55.83	62.42	49.66	53.87	51.49
$CLEAR_{base}$†	49.0	48.9	57.4	63.6	65.6	75.6	72.5	61.8
IS-$BERT_{base}$-NLI†	56.77	69.24	61.21	75.23	70.16	69.21	64.25	66.58
$BERT_{base}$-CT†	66.86	70.91	72.37	78.55	77.78	-	-	-
$BERT_{large}$-CT†	69.50	75.97	74.22	78.83	78.92	-	-	-
Using STS unlabeled texts								
$BERT_{base}$-flow†	63.48	72.14	68.42	73.77	75.37	70.72	63.11	69.57
$BERT_{large}$-flow†	65.20	73.39	69.42	74.92	**77.63**	72.26	62.50	70.76
$ConSERT_{base}$‡	64.64	78.49	69.07	79.72	75.95	73.97	67.31	72.74
$ConSERT_{large}$‡	**70.69**	**82.96**	**74.13**	**82.78**	76.66	**77.53**	**70.37**	**76.45**

图 5　无监督设置下的实验结果

在无监督实验中，我们直接基于预训练的 BERT 在无标注的 STS 数据上进行 Fine-tune。结果显示，我们的方法在完全一致的设置下大幅度超过之前的 SOTA—BERT-flow，达到了 8% 的相对性能提升。

4.2 有监督实验

Method	STS12	STS13	STS14	STS15	STS16	STSb	SICK-R	Avg.
Using NLI supervision								
InferSent - GloVe†	52.86	66.75	62.15	72.77	66.87	68.03	65.65	65.01
Universal Sentence Encoder†	64.49	67.80	64.61	76.83	73.18	74.92	76.69	71.22
$SBERT_{base}$-NLI†	70.97	76.53	73.19	79.09	74.30	77.03	72.91	74.89
$SBERT_{large}$-NLI†	72.27	78.46	74.90	80.99	76.25	79.23	73.75	76.55
$SBERT_{base}$-NLI (re-impl.)‡	69.89	75.77	72.36	78.51	73.67	76.75	72.76	74.24
$SBERT_{large}$-NLI (re-impl.)‡	72.69	78.77	75.13	80.95	76.89	79.53	73.25	76.74
$BERT_{base}$-CT†	68.80	74.58	76.62	79.72	77.14	-	-	-
$BERT_{large}$-CT†	69.80	75.45	76.47	81.34	78.11	-	-	-
$ConSERT_{base}$ *joint*‡	70.53	79.96	74.85	81.45	76.72	78.82	77.53	77.12
$ConSERT_{large}$ *joint*‡	**73.26**	**82.36**	**77.73**	**83.84**	**78.75**	**81.54**	**78.64**	**79.44**
Using NLI supervision and STS unlabeled texts								
$BERT_{base}$-flow†	68.95	78.48	77.62	81.95	78.94	81.03	74.97	77.42
$BERT_{large}$-flow†	70.19	80.27	78.85	82.97	**80.57**	81.18	74.52	78.36
$ConSERT_{base}$ *sup-unsup*‡	73.51	84.86	77.44	83.11	77.98	81.80	74.29	79.00
$ConSERT_{large}$ *sup-unsup*‡	75.26	**86.01**	79.00	83.88	79.45	82.95	76.54	80.44
$ConSERT_{base}$ *joint-unsup*‡	74.07	83.93	77.05	83.66	78.76	81.36	76.77	79.37
$ConSERT_{large}$ *joint-unsup*‡	**77.47**	85.45	**79.41**	**85.59**	80.39	**83.42**	**77.26**	**81.28**

图 6　有监督设置下的实验结果

在有监督实验中，我们额外使用了来自 SNLI 和 MNLI 的训练数据，使用上面提到的

融合额外监督信号的三种方法进行了实验。实验结果显示，我们的方法在“仅使用 NLI 有标注数据”和“使用 NLI 有标注数据 + STS 无标注数据”的两种实验设置下均超过了基线。在三种融合监督信号的实验设置中，我们发现 joint-unsup 方法取得了最好的效果。

4.3 不同的数据增强方法分析

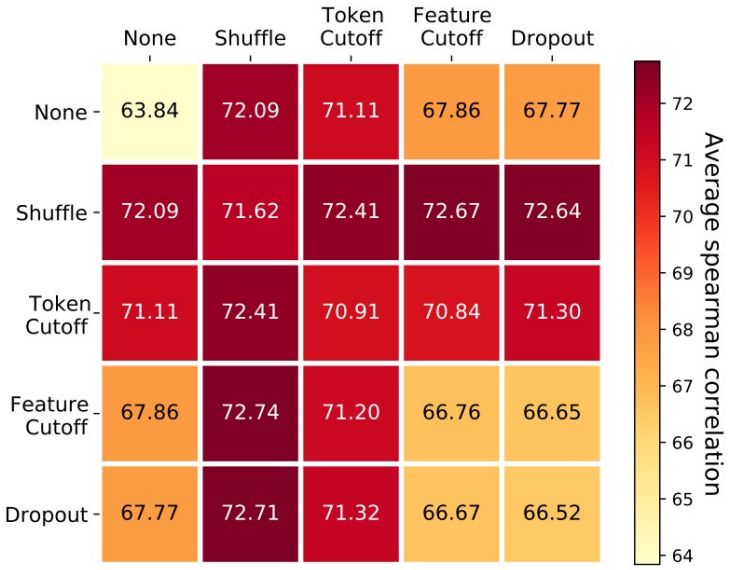


图 7　不同数据增强组合方法的性能

我们对不同的数据增强组合方法进行了消融分析，结果如图 7 所示。我们发现 Token Shuffle 和 Feature Cutoff 的组合取得了最优性能（72.74）。此外，就单种数据增强方法而言，Token Shuffle > Token Cutoff >> Feature Cutoff ≈ Dropout >> None。

4.4 少样本设置下的实验分析

我们进一步分析了数据量（无标注文本的数目）对效果的影响，结果如图 8 所示。结果显示，我们的方法仅需较少的样本就能近似达到全数据量的效果；同时，在样本量很少的情况下（如 100 条文本的情况下）仍相比于 Baseline 表现出不错的性能提升。

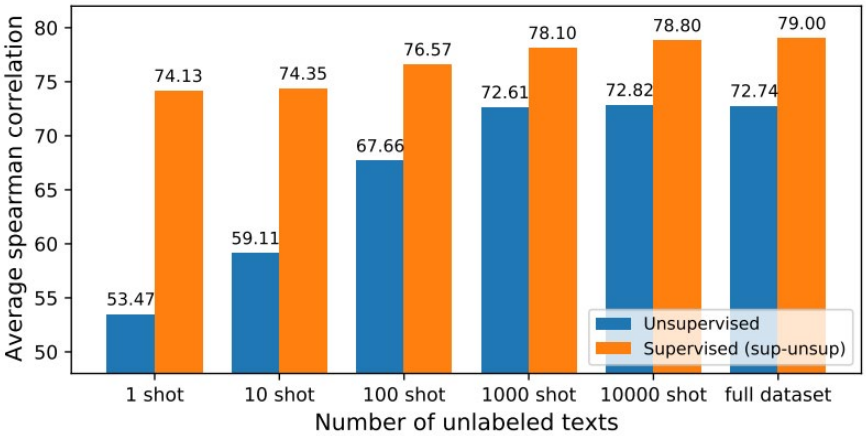


图 8　ConSERT 在小样本情况下的性能

4.5 Temperature 超参的实验分析

在实验中，我们发现对比学习损失函数中的温度超参数（τ）对于结果有很大影响。从图 9 的分析实验中可以看到，当 τ 值在 0.08 到 0.12 之间时会得到最优结果。这个现象再次证明了 BERT 表示的塌缩问题，因为在句子表示都很接近的情况下，τ 过大会使句子间相似度更平滑，编码器很难学到知识。而 τ 如果过小，任务就太过简单，所以需要调整到一个合适的范围内。

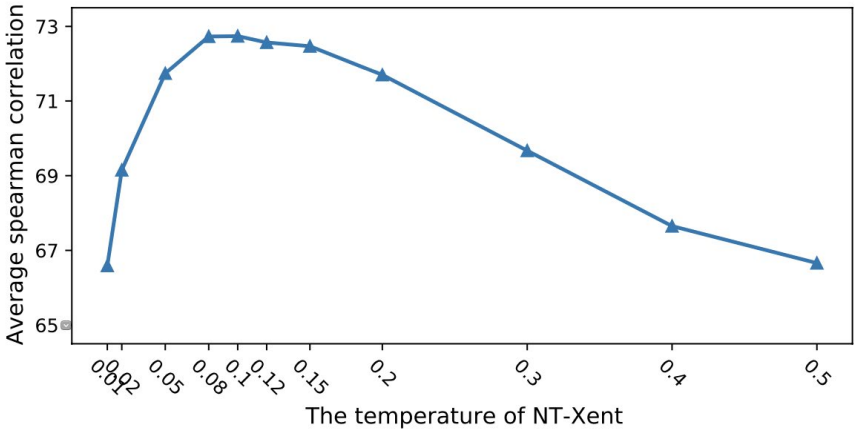


图 9　不同超参数 τ 下的性能

4.6 Batch size 超参的实验分析

在图像领域的对比学习中，Batch size 会对结果有很大影响，因此我们也对比了不同 Batch size 下模型的表现。从图 10 可以看到两者基本是成正比的，但提升很有限。

Batch Size	16	48	96	192	288
Avg. Spearman	72.63	72.60	72.74	72.86	72.98
Number of Steps	6175	2459	1530	930	620

图 10　不同 Batch size 下的性能

5. 总结

在此工作中，我们分析了 BERT 句向量表示空间坍缩的原因，并提出了一种基于对比学习的句子表示迁移框架 ConSERT。ConSERT 在无监督 Fine-tune 和进一步融合监督信号的实验中均表现出了不错的性能；同时当收集到的样本数较少时，仍能有不错的性能提升，表现出较强的鲁棒性。

同时，在美团的业务场景下，有大量不同领域的短文本相关性计算需求，目前 ConSERT 已经在知识图谱构建、KBQA、搜索召回等业务场景使用。未来将会在美团更多业务上进行探索落地。目前，相关代码已经在 GitHub 上开源，欢迎大家使用。

参考文献

[1] Reimers, Nils, and Iryna Gurevych. “Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks.” Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP). 2019.

[2] Li, Bohan, et al. “On the Sentence Embeddings from Pre-trained Language Models.” Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP). 2020.

[3] Gao, Jun, et al. “Representation Degeneration Problem in Training Natural Language Generation Models.” International Conference on Learning Representations. 2018.

[4] Wang, Lingxiao, et al. “Improving Neural Language Generation with Spectrum Control.” International Conference on Learning Representations. 2019.

[5] Conneau, Alexis, et al. “Supervised Learning of Universal Sentence Representations from Natural Language Inference Data.” Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing. 2017.
[6] Cer, Daniel, et al. “Universal Sentence Encoder for English.” Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations. 2018.
[7] Wang, Shuohang, et al. “Cross-Thought for Sentence Encoder Pre-training.” Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP). 2020.
[8] Yang, Ziyi, et al. “Universal Sentence Representation Learning with Conditional Masked Language Model.” arXiv preprint arXiv:2012.14388 (2020).
[9] Lee, Haejun, et al. “SLM: Learning a Discourse Language Representation with Sentence Unshuffling.” Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP). 2020.
[10] Su, Jianlin, et al. “Whitening sentence representations for better semantics and faster retrieval.” arXiv preprint arXiv:2103.15316 (2021).
[11] Gao, Tianyu, Xingcheng Yao, and Danqi Chen. “SimCSE: Simple Contrastive Learning of Sentence Embeddings.” arXiv preprint arXiv:2104.08821 (2021).
[12] Wu, Xing, et al. “Conditional bert contextual augmentation.” International Conference on Computational Science. Springer, Cham, 2019.
[13] Zhou, Wangchunshu, et al. “BERT-based lexical substitution.” Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. 2019.
[14] He, Kaiming, et al. “Momentum contrast for unsupervised visual representation learning.” Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2020.
[15] Chen, Ting, et al. “A simple framework for contrastive learning of visual representations.” International conference on machine learning. PMLR, 2020.
[16] Zhang, Yan, et al. “An Unsupervised Sentence Embedding Method by Mutual Information Maximization.” Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP). 2020.
[17] Fang, Hongchao, et al. “Cert: Contrastive self-supervised learning for language understanding.” arXiv preprint arXiv:2005.12766 (2020).
[18] Carlsson, Fredrik, et al. “Semantic re-tuning with contrastive tension.” International Conference on Learning Representations. 2021.
[19] Giorgi, John M., et al. “Declutr: Deep contrastive learning for unsupervised textual representations.” arXiv preprint arXiv:2006.03659 (2020).
[20] Wu, Zhuofeng, et al. “CLEAR: Contrastive Learning for Sentence Representation.” arXiv preprint arXiv:2012.15466(2020).

作者简介

渊蒙、如寐、思睿、富峥、武威等，美团平台 / 搜索与 NLP 部。
徐蔚然，北京邮电大学人工智能学院，模式识别实验室，副教授，博士生导师。

CVPR 2021 | 基于 Transformer 的端到端视频实例分割方法

作者：钰晴　昭良

前言

实例分割是计算机视觉中的基础问题之一。目前，静态图像中的实例分割业界已经进行了很多的研究，但是对视频的实例分割（Video Instance Segmentation，简称 VIS）的研究却相对较少。而真实世界中的摄像头所接收的，无论是自动驾驶背景下车辆实时感知的周围场景，还是网络媒体中的长短视频，大多数都是视频流信息而非纯图像信息。因而研究对视频建模的模型有着十分重要的意义，本文系美团无人配送团队在 CVPR2021 发表的一篇 Oral 论文：《End-to-End Video Instance Segmentation with Transformers》的解读。本届 CVPR 大会共收到 7015 篇有效投稿，最终共 1663 篇论文被接收，论文录用率为 23.7%，Oral 的录用率仅为 4%。

背景

图像的实例分割指的是对静态图像中感兴趣的物体进行检测和分割的任务。视频是包含多帧图像的信息载体，相对于静态图像来说，视频的信息更为丰富，因而建模也更为复杂。不同于静态图像仅含有空间的信息，视频同时含有时间维度的信息，因而更接近对真实世界的刻画。其中，视频的实例分割指的是对视频中感兴趣的物体进行检测、分割和跟踪的任务。如图 1 所示，第一行为给定视频的多帧图像序列，第二行为视频实例分割的结果，其中相同颜色对应同一个实例。视频实例分割不光要对单帧图像中的物体进行检测和分割，而且要在多帧的维度下找到每个物体的对应关系，即对其进行关联和跟踪。

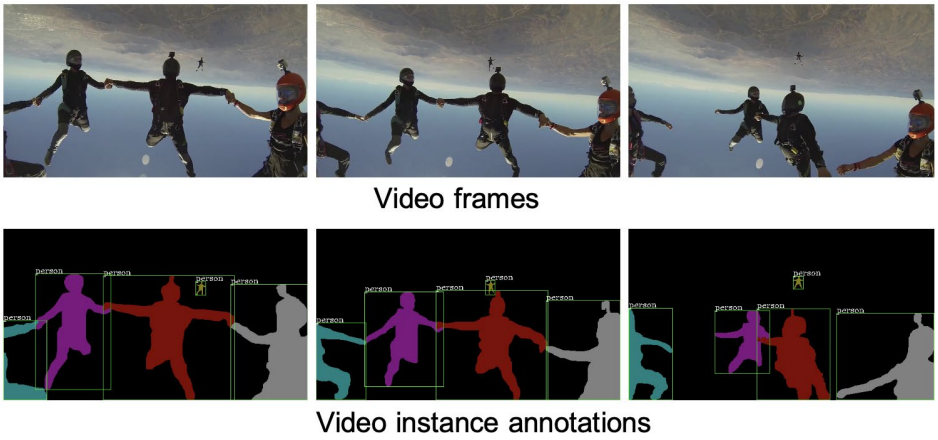


图 1　视频实例分割任务示意

相关工作

现有的视频实例分割算法通常为包含多模块、多阶段的复杂流程。最早的 Mask Track R-CNN[1] 算法同时包含实例分割和跟踪两个模块，通过在图像实例分割算法 Mask R-CNN[2] 的网络之上增加一个跟踪的分支实现，该分支主要用于实例特征的提取。在预测阶段，该方法利用外部 Memory 模块进行多帧实例特征的存储，并将该特征作为实例关联的一个要素进行跟踪。该方法的本质仍然是单帧的分割加传统方法进行跟踪关联。Maskprop[3] 在 Mask Track R-CNN 的基础上增加了 Mask Propagation 的模块以提升分割 Mask 生成和关联的质量，该模块可以实现当前帧提取的 mask 到周围帧的传播，但由于帧的传播依赖于预先计算的单帧的分割 Mask，因此要得到最终的分割 Mask 需要多步的 Refinement。该方法的本质仍然是单帧的提取加帧间的传播，且由于其依赖多个模型的组合，方法较为复杂，速度也更慢。

Stem-seg[4] 将视频实例分割划分为实例的区分和类别的预测两个模块。为了实现实例的区分，模型将视频的多帧 Clip 构建为 3D Volume，通过对像素点的 Embedding 特征进行聚类实现不同物体的分割。由于上述聚类过程不包含实例类别的预测，因此需要额外的语义分割模块提供像素的类别信息。根据以上描述，现有的算法大多沿袭单帧图像实例分割的思想，将视频实例分割任务划分为单帧的提取和多帧的关联

多个模块，针对单个任务进行监督和学习，处理速度较慢且不利于发挥视频时序连续性的优势。本文旨在提出一个端到端的模型，将实例的检测、分割和跟踪统一到一个框架下实现，有助于更好地挖掘视频整体的空间和时序信息，且能够以较快的速度解决视频实例分割的问题。

VisTR 算法介绍

重新定义问题

首先，我们对视频实例分割这一任务进行了重新的思考。相较于单帧图像，视频含有关于每个实例更为完备和丰富的信息，比如不同实例的轨迹和运动模态，这些信息能够帮助克服单帧实例分割任务中一些比较困难的问题，比如外观相似、物体邻近或者存在遮挡的情形等。另一方面，多帧所提供的关于单个实例更好的特征表示也有助于模型对物体进行更好的跟踪。因此，我们的方法旨在实现一个端到端对视频实例目标进行建模的框架。为了实现这一目标，我们第一个思考是：**视频本身是序列级别的数据，能否直接将其建模为序列预测的任务？**比如，借鉴自然语言处理（NLP）任务的思想，将视频实例分割建模为序列到序列（Seq2Seq）的任务，即给定多帧图像作为输入，直接输出多帧的分割 Mask 序列，这时需要一个能够同时对多帧进行建模的模型。

第二个思考是：**视频的实例分割实际同时包含实例分割和目标跟踪两个任务，能否将其统一到一个框架下实现？**针对这个我们的想法是：分割本身是像素特征之间相似度的学习，而跟踪本质是实例特征之间相似度的学习，因此理论上他们可以统一到同一个相似度学习的框架之下。

基于以上的思考，我们选取了一个同时能够进行序列的建模和相似度学习的模型，即自然语言处理中的 Transformer[5] 模型。Transformer 本身可以用于 Seq2Seq 的任务，即给定一个序列，可以输入一个序列。并且该模型十分擅长对长序列进行建模，因此非常适合应用于视频领域对多帧序列的时序信息进行建模。其次，Transformer 的核心机制，自注意力模块（Self-Attention），可以基于两两之间的相似度来进行特征的学习和更新，使得将像素特征之间相似度以及实例特征之间相似度

统一在一个框架内实现成为可能。以上的特性使得 Transformer 成为 VIS 任务的恰当选择。除此之外，Transformer 已经有被应用于计算机视觉中进行目标检测的实践 DETR[6]。因此我们基于 transformer 设计了视频实例分割（VIS）的模型 VisTR。

VisTR 算法流程

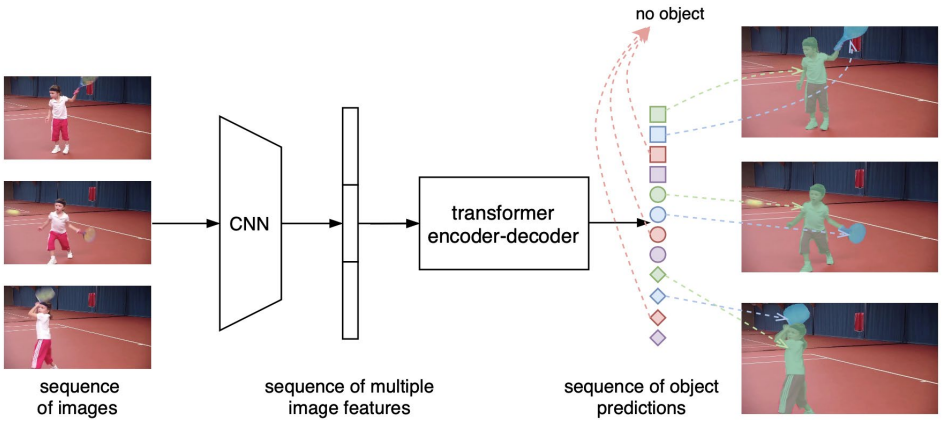


图 2　VisTR 算法框架

遵照上述思想，VisTR 的整体框架如图 2 所示。图中最左边表示输入的多帧原始图像序列（以三帧为例），右边表示输出的实例预测序列，其中相同形状对应同一帧图像的输出，相同颜色对应同一个物体实例的输出。给定多帧图像序列，首先利用卷积神经网络（CNN）进行初始图像特征的提取，然后将多帧的特征结合作为特征序列输入 Transformer 进行建模，实现序列的输入和输出。

不难看出，首先，VisTR 是一个端到端的模型，即同时对多帧数据进行建模。建模的方式即：将其变为一个 Seq2Seq 的任务，输入多帧图像序列，模型可以直接输出预测的实例序列。虽然在时序维度多帧的输入和输出是有序的，但是单帧输入的实例的序列在初始状态下是无序的，这样仍然无法实现实例的跟踪关联，因此我们强制使得每帧图像输出的实例的顺序是一致的（用图中同一形状的符号有着相同的颜色变化顺序表示），这样只要找到对应位置的输出，便可自然而然实现同一实例的关联，无需任何后处理操作。为了实现此目标，需要对属于同一个实例位置处的特征进行序列维

度的建模。针对性地，为了实现序列级别的监督，我们提出了 Instance Sequence Matching 的模块。同时为了实现序列级别的分割，我们提出了 Instance Sequence Segmentation 的模块。端到端的建模将视频的空间和时间特征当做一个整体，可以从全局的角度学习整个视频的信息，同时 Transformer 所建模的密集特征序列又能够较好的保留细节的信息。

VisTR 网络结构

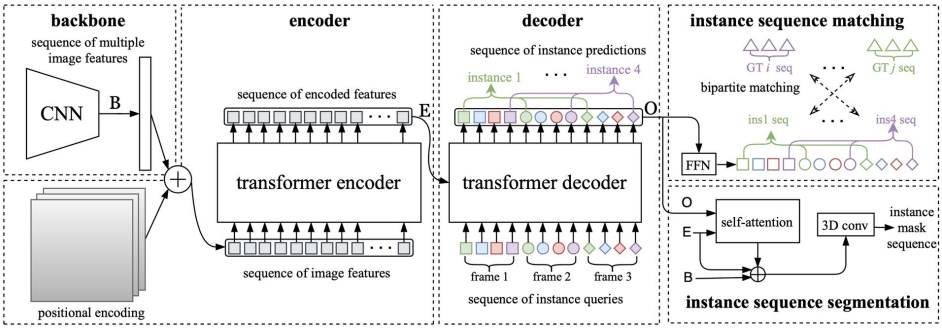


图 3　VisTR 网络结构

VisTR 的详细网络结构如图 3 所示，以下是对网络的各个组成部分的介绍：

- **Backbone**：主要用于初始图像特征的提取。针对序列的每一帧输入图像，首先利用 CNN 的 Backbone 进行初始图像特征的提取，提取的多帧图像特征沿时序和空间维度序列化为多帧特征序列。由于序列化的过程损失了像素原始的空间和时序信息，而检测和分割的任务对于位置信息十分敏感，因此我们将其原始的空间和水平位置进行编码，作为 Positional Encoding 叠加到提取的序列特征上，以保持原有的位置信息。Positional Encoding 的方式遵照 Image Transformer[7] 的方式，只是将二维的原始位置信息变为了三维的位置信息，这部分在论文中有详细的说明。
- **Encoder**：主要用于对多帧特征序列进行整体的建模和更新。输入前面的多帧特征序列，Transformer 的 Encoder 模块利用 Self-Attention 模块，通过点和

点之间相似度的学习，进行序列中所有特征的融合和更新。该模块通过对时序和空间特征的整体建模，能够对属于同一个实例的特征进行更好的学习和增强。

- **Decoder**：主要用于解码输出预测的实例特征序列。由于 Encoder 输入 Decoder 的是密集的像素特征序列，为了解码出稀疏的实例特征，我们参考 DETR 的方式，引入 Instance Query 进行代表性的实例特征的解码。Instance Query 是网络自身学习的 Embedding 参数，用于和密集的输入特征序列进行 Attention 运算选取能够代表每个实例的特征。以处理 3 帧图像，每帧图像预测 4 个物体为例，模型一共需要 12 个 Instance Query，用于解码 12 个实例预测。和前面的表示一致，用同样的形状表示对应同一帧图像的预测，同样的颜色表示同一个物体实例在不同帧的预测。通过这种方式，我们可以构造出每个实例的预测序列，对应为图 3 中的 Instance 1…Instance 4，后续过程中模型都将单个物体实例的序列看作整体进行处理。
- **Instance Sequence Matching**：主要用于对输入的预测结果进行序列级别的匹配和监督。前面已经介绍了从序列的图像输入到序列的实例预测的过程。但是预测序列的顺序其实是基于一个假设的，即在帧的维度保持帧的输入顺序，而在每帧的预测中，不同实例的输出顺序保持一致。帧的顺序比较容易保持，只要控制输入和输出的顺序一致即可，但是不同帧内部实例的顺序其实是没有保证的，因此我们需要设计专门的监督模块来维持这个顺序。在通用目标检测之中，在每个位置点会有它对应的 Anchor，因此对应每个位置点的 Ground Truth 监督是分配好的，而在我们的模型中，实际上是没有 Anchor 和位置的显式信息，因此对于每个输入点我们没有现成的属于哪个实例的监督信息。为了找到这个监督，并且直接在序列维度进行监督，我们提出了 Instance Sequence Matching 的模块，这个模块将每个实例的预测序列和标注数据中每个实例的 Ground Truth 序列进行二分匹配，利用匈牙利匹配的方式找到每个预测最近的标注数据，作为它的 Groud Truth 进行监督，进行后面的 Loss 计算和学习。
- **Instance Sequence Segmentation**：主要用于获取最终的分割结果序列。

前面已经介绍了 Seq2Seq 的序列预测过程，我们的模型已经能够完成序列的预测和跟踪关联。但是到目前为止，我们为每个实例找到的只是一个代表性的特征向量，而最终要解决的是分割的任务，如何将这个特征向量变为最终的 Mask 序列，就是 Instance Sequence Segmentation 模块要解决的问题。前面已经提到，实例分割本质是像素相似度的学习，因此我们初始计算 Mask 的方式就是利用实例的预测和 Encode 之后的特征图计算 Self-Attention 相似度，将得到的相似度图作为这个实例对应帧的初始 Attention Mask 特征。为了更好的利用时序的信息，我们将属于同一个实例的多帧的 Attention Mask 作为 Mask 序列输入 3D 卷积模块进行分割，直接得到最终的分割序列。这种方式通过利用多帧同一实例的特征对单帧的分割结果进行增强，可以最大化的发挥时序的优势。

VisTR 损失函数

根据前面的描述，网络学习中需要计算损失的主要有两个地方，一个是 Instance Sequence Matching 阶段的匹配过程，一个是找到监督之后最终整个网络的损失函数计算过程。

Instance Sequence Matching 过程的计算公式如式 1 所示：由于 Matching 阶段只是用于寻找监督，而计算 Mask 之间的距离运算比较密集，因此在此阶段我们只考虑 Box 和预测的类别 c 两个因素。第一行中的 yi 表示对应第 i 个实例的 Ground Truth 序列，其中 c 表示类别，b 表示 Boundingbox，T 表示帧数，即 T 帧该实例对应的类别和 Bounding Box 序列。第二行和第三行分别表示预测序列的结果，其中 p 表示在 ci 这个类别的预测的概率，b 表示预测的 Bounding Box。序列之间距离的运算是通过两个序列对应位置的值两两之间计算损失函数得到的，图中用 Lmatch 表示，对于每个预测的序列，找到 Lmatch 最低那个 Ground Truth 序列作为它的监督。根据对应的监督信息，就可以计算整个网络的损失函数。

$$y_i = \{(c_i, c_i..., c_i), (b_{i,0}, b_{i,1}..., b_{i,T})\}$$

$$\hat{p}_{(\sigma(i))}(c_i) = \{\hat{p}_{(\sigma(i),0)}(c_i)..., \hat{p}_{(\sigma(i),T)}(c_i)\}$$

$$\hat{b}_{\sigma(i)} = \left\{\hat{b}_{(\sigma(i),0)}, \hat{b}_{(\sigma(i),1)}..., \hat{b}_{(\sigma(i),T)}\right\}$$

$$\mathcal{L}_{\text{match}}\left(y_i, \hat{y}_{\sigma(i)}\right) = -\hat{p}_{\sigma(i)}\left(c_i\right) + \mathcal{L}_{\text{box}}\left(b_i, \hat{b}_{\sigma(i)}\right)$$

$$\hat{\sigma} = \arg\min_{\sigma \in S_n} \sum_{i}^{n} \mathcal{L}_{\text{match}}\left(y_i, \hat{y}_{\sigma(i)}\right)$$

公式 1　Instance Sequence Matching 计算过程

由于我们的方法是将分类、检测、分割和跟踪做到一个端到端网络里，因此最终的损失函数也同时包含类别、Bounding Box 和 Mask 三个方面，跟踪通过直接对序列算损失函数体现。公式 2 表示分割的损失函数，得到了对应的监督结果之后，我们计算对应序列之间的 Dice Loss 和 Focal Loss 作为 Mask 的损失函数。

$$\mathcal{L}_{\text{mask}}\left(m_i, \hat{m}_{\sigma(i)}\right) = \lambda_{\text{mask}} \frac{1}{T} \sum_{t=1}^{T} \Big[\mathcal{L}_{\text{Dice}}(m_{i,t}, \hat{m}_{\sigma(i),t}) + \mathcal{L}_{\text{Focal}}(m_{i,t}, \hat{m}_{\sigma(i),t})\Big].$$

公式 2　分割损失函数

最终的损失函数如公式 3 所示，为同时包含分类（类别概率）、检测（Bounding Box）以及分割（Mask）的序列损失函数之和。

$$\mathcal{L}_{\text{Hung}}(y, \hat{y}) = \sum_{i=1}^{N} \Big[(-\log \hat{p}_{\hat{\sigma}(i)}(c_i)) + \mathcal{L}_{\text{box}}\left(b_i, \hat{b}_{\hat{\sigma}}(i)\right) + \mathcal{L}_{\text{mask}}\left(m_i, \hat{m}_{\hat{\sigma}}(i)\right)\Big].$$

公式 3　VisTR 整体损失函数

实验结果

为了验证方法的效果，我们在广泛使用的视频实例分割数据集 YouTube-VIS 上进行了实验，该数据集包含 2238 个训练视频，302 个验证视频以及 343 个测试视频，以及 40 个物体类别。模型的评估标准包含 AP 和 AR，以视频维度多帧 Mask 之间的 IOU 作为阈值。

时序信息的重要性

相对于现有的方法，VisTR 的最大区别是直接对视频进行建模，而视频和图像的主要区别在于视频包含着丰富的时序信息，如果有效的挖掘和学习时序信息是视频理解的关键，因此我们首先探究了时序信息的重要性。时序包含两个维度：时序的多少（帧数）以及有序和无序的对比。

Length	AP	AP_{50}	AP_{75}	AR_1	AR_{10}
18	29.7	50.4	31.1	29.5	34.4
24	30.5	47.8	33.0	29.5	34.4
30	31.7	53.2	32.8	31.3	36.0
36	33.3	53.4	35.1	33.1	38.5

表 1　不同帧数模型的训练效果对比

表 1 中展示了我们利用不同帧数的 Clip 训练模型最终的测试效果，不难看出，随着帧数从 18 提升至 36，模型的精度 AP 也在不断提升，证明多帧提供的更丰富的时序信息对模型的学习有所帮助。

time order	AP	AP_{50}	AP_{75}	AR_1	AR_{10}
random	32.3	52.1	34.3	33.8	37.3
in order	33.3	53.4	35.1	33.1	38.5

表 2　打乱时序 vs. 按照时序训练的效果对比

表 2 中展示了利用打乱物理时序的 Clip 以及按照物理时序的 Clip 进行模型训练的效果对比，可以看出，按照时间顺序训练的模型有一个点的提升，证明 VisTR 有学到物理时间下物体变化和规律，而按照物理时间顺序对视频建模也有助于视频的理解。

Query 探究

第二个实验是对于 Query 的探究。由于我们的模型直接建模的 36 帧图像，对每帧图像预测 10 个物体，因此需要 360 个 Query，对应表 3 最后一行的结果（Prediction Level）。我们想探究属于同一帧或者同一个实例的 Query 之间是否存在一定的关联，即是否可以共享。针对这个目标，我们分别设计了 Frame Level 的实验：即一帧只使用一个 Query 的特征进行预测，以及 Instance level 的实验：一个实例只使用一个 Query 的特征进行预测。

可以看到，Instance Level 的结果只比 Prediction Level 的结果少一个点，而 Frame Level 的结果要低 20 个点，证明不同帧属于同一个 Instance 的 Query 可以共享，而同一帧不同 Instance 的 Query 信息不可共享。Prediction Level 的 Query 是和输入的图像帧数成正比的，而 Instance Level 的 Query 可以实现不依赖输入帧数的模型。只限制模型要预测的 Instance 个数，不限制输入帧数，这也是未来可以继续研究的方向。除此之外，我们还设计了 Video Level 的实验，即整个视频只用一个 Query 的 Embedding 特征进行预测，这个模型可以实现 8.4 的 AP。

	number	AP	AP_{50}	AP_{75}	AR_1	AR_{10}
video level	1	8.4	13.2	9.5	20.0	20.8
frame level	36	13.7	23.3	14.5	30.4	35.1
ins. level	10	32.0	52.8	34.0	31.6	37.2
pred. level	360	33.3	53.4	35.1	33.1	38.5

表 3　不同类型 Query 的效果对比

其他设计

以下是实验过程中我们发现有效的其他设计。

	AP	AP_{50}	AP_{75}	AR_1	AR_{10}
w/o	28.4	50.1	29.5	29.6	33.3
w	33.3	53.4	35.1	33.1	38.5

表 4　有无 positional encoding 的实验效果对比

由于在特征序列化的过程中会损失原有的空间和时间信息，我们提供了原始的 Positional Encoding 特征以保留原有的位置信息。在表 5 中进行了有无该模块的对比，Positional Encoding 提供的位置信息可以带来约 5 个点的提升。

	AP	AP_{50}	AP_{75}	AR_1	AR_{10}
CNN	32.0	54.5	31.5	31.6	37.7
Transformer	33.3	53.4	35.1	33.1	38.5

表 5　CNN-encoded 的特征 vs.Transformer-Encoded 的特征对分割的效果对比

在分割的过程中，我们通过计算实例的 Prediction 与 Encoded 之后特征的 Self-Attention 来获取初始的 Attention Mask。在表 6 中，我们进行了利用 CNN-Encoded 的特征和利用 Transformer-Encoded 的特征进行分割的效果对比，利用 Transformer 的特征可以提升一个点。证明了 Transformer 进行全局特征更新的有效性。

	AP	AP_{50}	AP_{75}	AR_1	AR_{10}
w/o	33.3	53.4	35.1	33.1	38.5
w	34.4	55.7	36.5	33.5	38.9

表 6　有无 3D 卷积的实验效果对比

表 6 中展示了在分割模块有无 3D 卷积的效果对比，使用 3D 卷积可以带来一个点的提升，证明了利用时序信息直接对多帧 mask 分割的有效性。

可视化结果

图 4　VisTR 可视化效果

VisTR 在 YouTube-VIS 的可视化效果如图 4 所示，其中每行表示同一个视频的序列，相同颜色对应同一个实例的分割结果。可以看出无论是在 (a). 实例存在遮挡 (b). 实例位置存在相对变化 ©. 同类紧邻易混淆的实例 以及 (d). 实例处于不同姿态情形下，模型都能够实现较好的分割个跟踪，证明在有挑战性的情况下，VisTR 仍具有很好的效果。

方法对比

Method	backbone	FPS	AP	AP_{50}	AP_{75}	AR_1	AR_{10}
DeepSORT [28]	ResNet-50	-	26.1	42.9	26.1	27.8	31.3
FEELVOS [24]	ResNet-50	-	26.9	42.0	29.7	29.9	33.4
OSMN [31]	ResNet-50	-	27.5	45.1	29.1	28.6	33.1
MaskTrack R-CNN [30]	ResNet-50	20.0	30.3	51.1	32.6	31.0	35.5
STEm-Seg [1]	ResNet-50	-	30.6	50.7	33.5	31.6	37.1
STEm-Seg [1]	ResNet-101	2.1	34.6	55.8	37.9	34.4	41.6
MaskProp [2]	ResNet-50	-	40.0	-	42.9	-	-
MaskProp [2]	ResNet-101	-	42.5	-	45.6	-	-
VisTR	ResNet-50	30.0/69.9	36.2	59.8	36.9	37.2	42.4
VisTR	ResNet-101	27.7/57.7	40.1	64.0	45.0	38.3	44.9

表 7　VisTR 可视化效果

表 7 是我们的方法和其他方法在 YoutubeVIS 数据集上的对比。我们的方法实现了单一模型的最好效果（其中 MaskProp 包含多个模型的组合），在 57.7FPS 的速度下实现了 40.1 的 AP。其中前面的 27.7 指的是加上顺序的 Data Loading 部分的速度（这部分可以利用并行进行优化），57.7 指的是纯模型 Inference 的速度。由于我们的方法直接对 36 帧图像同时进行建模，因此相对于同样模型的单帧处理，理想情况下能带来大约 36 倍的速度提升，更有助于视频模型的广泛应用。

总结与展望

视频实例分割指的是同时对视频中感兴趣的物体进行分类，分割和跟踪的任务。现有的方法通常设计复杂的流程来解决此问题。本文提出了一种基于 Transformer 的视频实例分割新框架 VisTR，该框架将视频实例分割任务视为直接端到端的并行序列解码和预测的问题。给定一个含有多帧图像的视频作为输入，VisTR 直接按顺序输出视频中每个实例的掩码序列。该方法的核心是一种新的实例序列匹配和分割的策略，能够在整个序列级别上对实例进行监督和分割。 VisTR 将实例分割和跟踪统一到了相似度学习的框架下，从而大大简化了流程。在没有任何 trick 的情况下，VisTR 在所有使用单一模型的方法中获得了最佳效果，并且在 YouTube-VIS 数据集上实现了最快的速度。

据我们所知，这是第一个将 Transformers 应用于视频分割领域的方法。希望我们的方法能够启发更多视频实例分割的研究，同时也希望此框架未来能够应用于更多视频理解的任务。关于论文的更多细节，请参考原文：《End-to-End Video Instance Segmentation with Transformers》，同时代码也已经在 GitHub 上开源：https://github.com/Epiphqny/VisTR，欢迎大家来了解或使用。

作者

美团无人车配送中心：钰晴、昭良、保山、申浩等。
阿德莱德大学：王鑫龙、沈春华。

参考文献

[1] Video Instance Segmentation, https://arxiv.org/abs/1905.04804.
[2] Mask R-CNN, https://arxiv.org/abs/1703.06870.
[3] Classifying, Segmenting, and Tracking Object Instances in Video with Mask Propagation, https://arxiv.org/abs/1912.04573.
[4] STEm-Seg: Spatio-temporal Embeddings for Instance Segmentation in Videos, https://arxiv.org/abs/2003.08429.
[5] Attention Is All You Need, https://arxiv.org/abs/1706.03762.
[6] End-to-End Object Detection with Transformers, https://arxiv.org/abs/2005.12872.
[7] Image Transformer, https://arxiv.org/abs/1802.05751.

招聘信息

美团无人车配送中心大量岗位持续招聘中，诚招算法 / 系统 / 硬件开发工程师及专家。欢迎感兴趣的同学发送简历至：ai.hr@meituan.com（邮件标题注明：美团无人车团队）。

ICLR 2021 | 美团 AutoML 论文：鲁棒的神经网络架构搜索 DARTS-

作者：祥祥　晓星　张勃　晓林

背景

美团日益增长的用户侧和商家侧业务对人工智能（AI）技术有着非常广泛和强烈的诉求。从用户角度出发，美团 AI 在外卖之外，有到店消费、酒店旅游等 200 多个生活服务场景，均需要 AI 来提升用户体验。从商家角度出发，美团 AI 将帮助商家提高效率、分析运营状况，比如能对用户评论进行细粒度分析，来刻画出商家服务现状、商家竞争力分析，以及商圈洞察等等，为商户提供精细化经营建议。

目前，美团 AI 涉及的研发领域，包括自然语言理解、知识图谱、搜索、语音识别、语音生成、人脸识别、文字识别、视频理解、图像编辑、AR、环境预测、行为规划、运动控制等。AI 技术在这些场景中落地的两个关键部分是规模化的数据和先进的深度学习模型，其中高质量模型的设计和更新迭代是当前 AI 生产开发的痛点和难点，亟需自动化技术来辅助并提升生产效率。在此情景下应运而生的技术叫做自动化机器学习（AutoML）。AutoML 被认为是模型设计的未来解决方案，能够将 AI 算法工程师从手动设计的繁复试错中解放出来。

谷歌 2017 年正式提出神经网络架构搜索（Neural Architecture Search，NAS）[1] 用于自动化生成模型架构，这项技术被业界寄予厚望，成为 AutoML 的核心组成部分。凭借日益增强的算力和持续迭代的 NAS 算法，视觉模型在架构层面诞生了像 EfficientNet、MobileNetV3 等影响力深远的系列模型，NAS 也应用到了视觉、NLP、语音等领域的很多方向 [2,3]。NAS 作为生成 AI 模型的 AI，其重要意义不言而喻。美团在 NAS 方向也开展了深入性的研究，对该领域保持着积极的探索。

本文介绍美团和上海交通大学合作的文章 DARTS-[4]，该文即将发表在 ICLR 2021 顶会上。ICLR（International Conference on Learning Representations）全称是

国际学习表征会议，2013 年由两位深度学习大牛、图灵奖得主 Yoshua Bengio 和 Yann LeCun 牵头创办。ICLR 成立至今仅七年，但它已得到学术界的广泛认可，被认为是“深度学习领域的顶级会议”。ICLR 的 h5 指数为 203，在所有科学出版物中排名第 17，超过了 NeurIPS、ICCV 和 ICML。本届 ICLR 共有 2997 篇论文提交，最终接收 860 篇，包括 53 篇 Oral（接收率 6%），114 篇 Spotlight，693 篇 Poster，接收率为 28.7%。

神经网络架构搜索简介

神经网络架构搜索（NAS）的主要任务是如何在有限时间和资源下搜索得到最优的模型。NAS 主要由搜索空间、搜索算法、模型评估三部分组成。NAS 最早在视觉分类任务中验证，在分类任务中常见的搜索空间分为基于子结构单元（Cell）和基于子结构块（Block）两种，前者的特点是具有丰富的图结构，将相同的单元串联再组成最终的网络。后者是直筒型的，搜索的焦点就在于每层子结构块的选取。

按搜索算法分类，NAS 主要包括基于强化学习（Reinforcement Learning，RL）、基于遗传算法（Evolutionary Algorithm，EA）、基于梯度优化（Gradient-Based）的方法。RL 方法通过生成并评估模型来获取反馈，根据反馈来调整生成的策略，从而生成新的模型，循环这一过程直到最优。 EA 方法将模型结构编码为可以交叉和变异的“基因”，通过不同的遗传算法来获取新一代的基因，直到达到最好。EA 方法的优点在于可以处理多种目标，比如一个模型的优劣有参数量、计算延迟、性能指标等多个考察维度，EA 方法便很适合在多个维度进行探索和演进。但 RL 和 EA 均比较耗时，主要受限于模型评估部分，一般采取全量、少量训练的方法。最新的 One-Shot 路线采用训练一个包含所有子结构的超网来评估所有子网的方式，可以从很大程度上提高 NAS 的效率。但同期，基于梯度优化的 DARTS 方法更为高效，成为当下 NAS 方法主流的选择。

DARTS 由卡耐基梅隆大学（CMU）的研究者刘寒骁等人提出，全称为可微分的神经网络架构搜索（Differentiable Architecture Search，DARTS）[5]，大幅提高了

搜索效率，受到了业界的广泛认可。可微分方法（DARTS）基于梯度优化，它首先其定义了一个基于有向无环图（DAG）的子结构（Cell），DAG 有四个中间节点（下图 Figure 1 中灰色方框），每条边有多个可选算子（由不同颜色的边表示），通过 softmax 加和不同边的结果作为到下个节点的输入。堆叠这样的子结构可以形成网络的主干。DARTS 将搜索过程看作对堆叠而成的主干网络（也称为超网，或过参数化网络）的优化过程。这里每条边被赋予了不同的结构权重，并和网络权重一起交叉进行梯度更新。优化完成后结构权重大（由粗线条表示）的作为最终子网的算子，并将该子网作为搜索结果（Figure 1d 展示了最终的 Cell 结构）。这个过程（ Figure 1 从 c 到 d）将连续的结构权重硬性截断为离散值，比如 0.2 变为 1，0.02 变为 0，而这样会产生所谓的离散化偏差（Discretization Gap）。

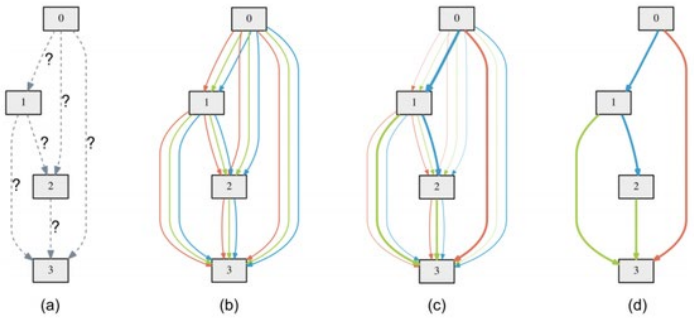


Figure 1: An overview of DARTS: (a) Operations on the edges are initially unknown. (b) Continuous relaxation of the search space by placing a mixture of candidate operations on each edge. (c) Joint optimization of the mixing probabilities and the network weights by solving a bilevel optimization problem. (d) Inducing the final architecture from the learned mixing probabilities.

神经网络架构搜索的难点

简单总结目前神经网络架构搜索主要需要解决的难点在于：

- 搜索过程的**高效性**：搜索算法耗费的计算资源和时间要在可接受的范围，从而可以在实践中得到广泛的应用，直接支撑面向业务数据集的模型结构搜索；
- 搜索结果的**有效性**：搜索得到的模型要在多个数据集上有很好的性能，且又很好的泛化性能和领域迁移能力，比如搜索得到的分类主干网可以很好地迁移到

检测和分割任务，并且有很好的表现；

- 搜索结果的**鲁棒性**：在具有有效性的同时，多次搜索的结果要相对稳定，即提高搜索的可靠性，降低试错成本。

可微分方法的缺点和改进方法

可微分神经网络架构搜索方法的不足之处就是鲁棒性较差，容易产生性能崩塌，即搜索过程中的超网性能表现很好但推断出的子网存在大量的跳跃连接（Skip Connection），严重削弱了最终模型的性能。基于 DARTS 涌现出了非常多的改进工作，比如 Progessive DARTS[6]，Fair DARTS[7]，RobustDARTS[8]，Smooth DARTS[9] 等。其中，ICLR 2020 满分论文 RobustDARTS 提出用 Hessian 特征根作为衡量 DARTS 出现性能崩塌的征兆，但计算特征根又非常耗时。而且在标准的 DARTS 搜索空间下，RobustDARTS 在 CIFAR-10 数据集上的搜索得到的模型性能并不突出。这促使我们思考怎么提高鲁棒性，同时又提高有效性。针对这两个问题，业内有不同的分析和解决方法，代表性的分别是 Fair DARTS（ECCV 2020）、RobustDARTS（ICLR 2020）和 Smooth DARTS（ICML 2020）。

Fair DARTS 观察到大量跳跃连接的存在，并着重分析了其可能的产生原因。文章认为，跳跃连接在可微分的优化过程中，存在竞争环境下的不公平优势（Unfair Advantage），导致跳跃连接容易在竞争中胜出。因此，FairDARTS 提出放宽竞争环境（Softmax 加和）为合作环境（Sigmoid 加和），使得不公平优势带来的影响失效。最终选取算子方式也与 DARTS 不同，通过采取阈值截断，比如选取结构权重高于 0.8 的算子，此时跳跃连接可以和其他算子同时出现，但这样等同于增大了搜索空间：原先的子网中，两个节点之间最终只选取一个。

RobustDARTS（简称 R-DARTS）通过计算 Hessian 特征根来判断优化过程是否出现崩塌，文章认为，损失函数地貌（Loss Landscape）存在尖锐的局部最优点（Sharp Local Minima，Figure 5a 右侧点），离散化过程（α^* 到 α^{disc}）会导致从优化较好的尖锐点偏移到优化较差的地方，从而导致最终模型性能下降。R-DARTS 发现这个过程和 Hessian 特征根关系密切（Figure 5b）。因此可认为，Hessian 特

征根变化幅度过大时优化应该停止，或者通过正则化手段来避免 Hessian 特征根产生大幅变化。

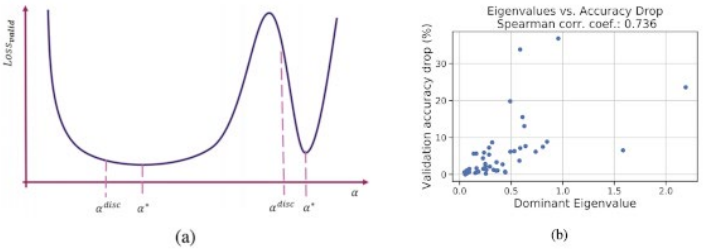


Figure 5: (a) Hypothetical illustration of the loss function change in the case of flat vs. sharp minima. (b) Drop in accuracy after discretizing the search model vs. the sharpness of minima (by means of λ^{α}_{max}).

Smooth DARTS（简作 SDARTS）遵循了 R-DARTS 的判断依据，采取基于扰动的正则化方法，对 Hessian 特征根进行了隐式的约束。具体来讲，SDARTS 对结构权重给予了一定程度的随机扰动，使得超网具有更好的抗干扰性，同时对损失函数地貌有平滑作用。

DARTS- 跳跃连接的工作机制分析

我们首先从跳跃连接的工作机制进行分析性能崩塌现象。ResNet [11] 中引入了跳跃连接，从而使得反向传播时，网络的浅层总包含对深层的梯度，因此可以缓解梯度消失的现象。如下式（i，j，k 表示层数，X 为输入，W 为权重，f 为计算单元）。

$$\frac{\partial \mathcal{L}}{\partial W_i} = \frac{\partial \mathcal{L}}{\partial X_{i+j}} \cdot \prod_{k=1}^{j} \left(\frac{\partial f_{i+k}}{\partial X_{i+k-1}} + \mathbb{1} \right) \cdot \frac{\partial f_i}{\partial W_i} \tag{2}$$

为了理清跳跃连接对残差网络性能的影响，我们在 ResNet 上做了一组验证性试验，即对跳跃连接加上可学习的结构权重参数 β，此时我们的梯度计算则变为下式：

$$\frac{\partial \mathcal{L}}{\partial X_i} = \frac{\partial \mathcal{L}}{\partial X_{i+1}} \cdot \left(\frac{\partial f_{i+1}}{\partial X_i} + \beta \right) \tag{3}$$

三次实验分别初始化 β 为 {0, 0.5, 1.0}，我们发现 β 总能快速增长到 1 附近（Figure 2）来增大深层梯度向浅层的传递，进而缓解梯度消失的现象。

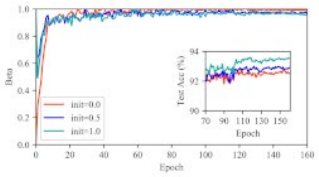


Figure 2: Tendency of trainable coefficient β (initialized with $\{0, 0.5, 1\}$) of the skip connection in ResNet50 and test accuracy (inset figure) vs. epochs. The residual structure is proved to learn a large β to ease training in all three cases. All models are trained and tested on CIFAR-10.

在 DARTS 中，跳跃连接跟 ResNet 类似，当具有可学习参数时，其结构参数也具有这种趋势，从而促进超网的训练。但正如 Fair DARTS [7] 提到的，同时带来的问题就是对其他算子来讲跳跃连接存在不公平优势。

解决崩塌的办法：增加辅助跳跃连接

根据上述的分析，DARTS− 指出跳跃连接（下图 Figure 1 中 Skip）存在双重作用：

- 作为可选算子本身，参与构建子网。
- 与其他算子形成残差结构，从而产生对超网优化产生促进作用。

第一个作用是预期其要发挥的作用，从而与其他算子公平竞争。第二个作用是跳跃连接具有不公平优势的原因，促进了优化，但干扰了我们对最终搜索结果的推断。

为了将第二个作用剥离出来，我们提出额外增加一条跳跃连接（Auxiliary Skip），并使其结构权重 β 从 1 衰减到 0（简便起见，使用线性衰减），这样可以使得超网和子网保持结构上的一致性。Figure 1（b）图示出了一个子结构中两个节点间的连接情况。

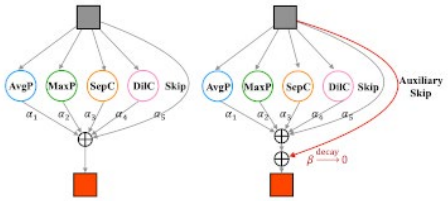


Figure 1: Schematic illustration about (a) DARTS and (b) the proposed DARTS-, featuring an auxiliary skip connection (thick red line) with a decay rate β between every two nodes to remove the potential unfair advantage that leads to performance collapse.

除了增加的辅助跳跃连接，DARTS- 优化过程和 DARTS 大同小异。首先根据 Figure 1（b）构建超网，选取一种 β 衰减策略，然后采取交替迭代优化超网权重 w 和结构权重 α，具体见下面的算法描述（Algorithm 1）。

Algorithm 1 DARTS-

Require:
Network weights w; Architecture parameters α;
Number of search epochs E;
Decay strategy for $\beta_e, e \in \{1, 2, ..., E\}$.

Ensure:
Searched architecture parameters α.

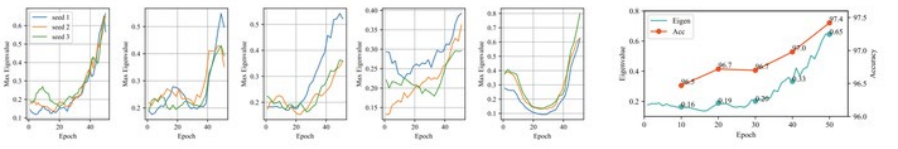
1: Construct a super-network by stacking cells in which there is an auxiliary skip connection between every two nodes of choice
2: **for** each $e \in [1, E]$ **do**
3: Update weights w by $\nabla_w \mathcal{L}_{train}(w, \alpha, \beta_e)$
4: Update parameters α by $\nabla_\alpha \mathcal{L}_{val}(w, \alpha, \beta_e)$
5: **end for**
6: Derive the final architecture based on learned α from the best validation supernet.

在本方法中，我们去掉了用指示信号（Indicator）发现性能崩塌的做法，比如 R-DARTS 中的特征根，从而消除了 DARTS 的性能崩塌，因此命名为 DARTS-。另外根据 PR-DARTS [12] 的收敛理论来分析，辅助跳跃连接具有平衡算子间竞争的作用，且当 β 衰减后，算子间的公平竞争依然保持。

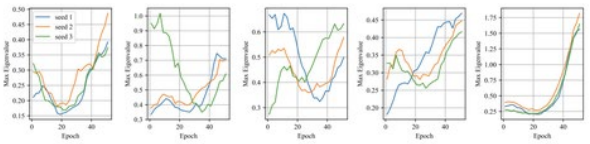
分析和验证

Hessian 特征根变化趋势

在 R-DARTS 以及 DARTS 采用的多个搜索空间下，DARTS- 发现了子网性能增长（Figure 4b）但 Hessian 特征根变化幅度过大（Figure 4a）的情形，这个结果成为了 R-DARTS 所提出原则的反例，即采用 R-DARTS 判定准则，我们会漏掉一些好的模型。这也说明了 DARTS- 可以带来不同于 R-DARTS 的模型结构。



(a) Trajectory of eigenvalues in S0-S4 (left to right) on CIFAR-10 (b) Sampled models' performance

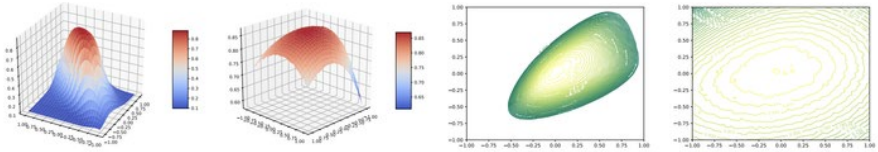


(c) Trajectory of eigenvalues in S0-S4 (left to right) on CIFAR-100

Figure 4: The evolution of maximal eigenvalues of DARTS- when searching in different search spaces S0-S4 on CIFAR-10 (a) and CIFAR-100 (c). We run each experiment 3 times on different seeds. (b) DARTS-'s growing Hessian eigenvalues don't induce poor performance. Among the sampled five models, the one corresponding to the highest eigenvalue has the best performance. This example is done in S0 on CIFAR-10.

验证集准确率地貌

验证集准确率的地貌可以一定程度上说明模型的优化过程难易。DARTS（Figure 3a）在最优解附近范围的地貌相对陡峭，等高线比较疏密不均，而 DARTS- 则表现得舒缓平滑，等高线更为均匀。另外更为光滑的地貌也不容易出现尖锐的局部最优点，一定程度也减少了离散化偏差。



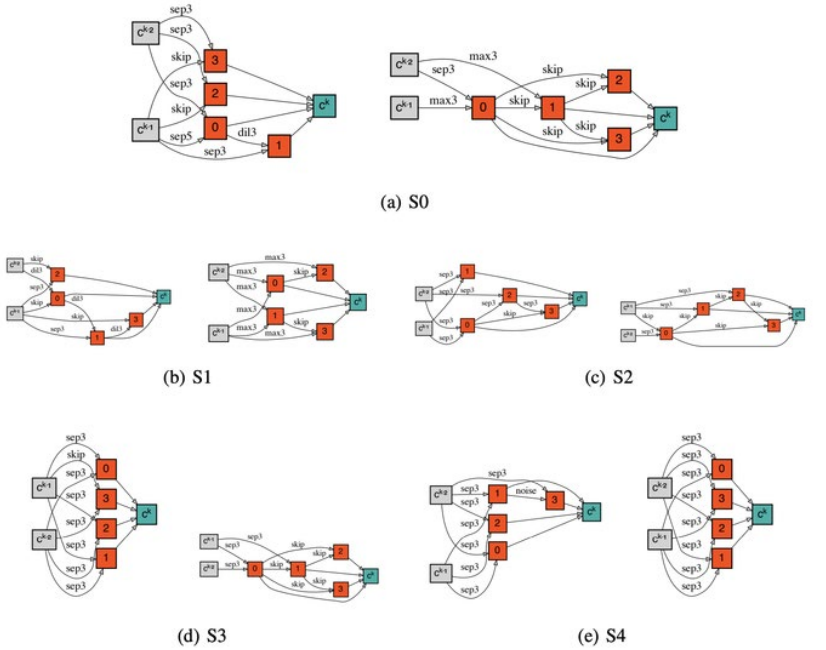
(a) DARTS (b) DARTS- (c) DARTS (d) DARTS-

Figure 3: Comparison of the validation accuracy landscape of (a) DARTS and (b) DARTS- w.r.t. α on CIFAR-10 in S3. Their contour maps are shown respectively in (c) and (d), where we set the step of contour map as 0.1. The accuracy of the derived models are 94.84% (a,c) and 97.58% (b,d), while the maximum Hessian eigenvalues are similarly high (0.52 and 0.42)

实验结果

模型结构

Figure 9 给出了我们在 DARTS 搜索空间 S0 和 Robust DARTS 搜索空间 S1–S4 得到的网络结构。Figure 10 是在 MobileNetV2 的搜索空间下 ImageNet 数据集上进行直接搜索的结果。



(a) S0 (b) S1 (c) S2 (d) S3 (e) S4

Figure 9: The best found normal cell and reduction cell in search spaces S0-S4 on CIFAR-10 dataset.

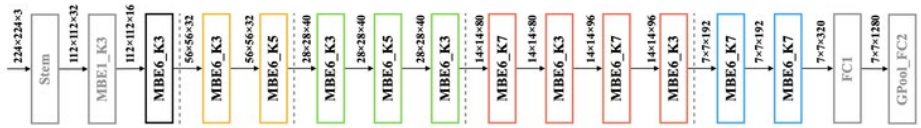


Figure 10: Architecture of DARTS-A searched on ImageNet dataset.

分类任务结果

在标准分类数据集 CIFAR-10 和 ImageNet 上 DARTS- 均取得了业界领先的结果，如下表所示：

Table 2: Comparison of the state-of-the-art models on CIFAR-10 (left) and ImageNet (right). On CIFAR-10 dataset, our average result is obtained on 5 independently searched models to assure the robustness. For ImageNet, networks in the top block are directly searched on ImageNet; the middle indicates that architectures are searched on CIFAR-10 and then transferred to ImageNet; the bottom indicates models have SE and Swish. We search in S0 for CIFAR-10 and S5 for ImageNet.

Models	Params (M)	FLOPs (M)	Acc (%)	Cost GPU Days
NASNet-A (2018)	3.3	608†	97.35	2000
ENAS (2018)	4.6	626†	97.11	0.5
DARTS (2019b)	3.3	528†	97.00±0.14*	0.4
SNAS (2019)	2.8	422†	97.15±0.02*	1.5
GDAS (2019b)	3.4	519†	97.07	0.2
P-DARTS (2019b)	3.4	532†	97.5	0.3
PC-DARTS (2020)	3.6	558†	97.43	0.1
DARTS- (best)	3.5	568	97.5	0.4
P-DARTS (2019b)‡	3.3±0.21	540±34	97.19±0.14	0.3
R-DARTS (2020)	-	-	97.05±0.21	1.6
SDARTS-ADV (2020)	3.3	-	97.39±0.02	1.3
DARTS- (avg.)	3.5±0.13	583±22	97.41±0.08	0.4

† Based on the provided genotypes.
‡ 5 independent searches using their released code.
* Training the best searched model for several times (*whose average doesn't indicate the stability of the method*)

Models	FLOPs (M)	Params (M)	Top-1 (%)	Top-5 (%)	Cost (GPU days)
AmoebaNet-A (2019)	555	5.1	74.5	92.0	3150
MnasNet-92 (2019)	388	3.9	74.79	92.1	3791‡
FBNet-C (2019)	375	5.5	74.9	92.3	9
FairNAS-A (2019b)	388	4.6	75.3	92.4	12
SCARLET-C (2019a)	365	6.7	76.9	93.4	10
FairDARTS-D (2020b)	440	4.3	75.6	92.6	3
PC-DARTS (2020)	597	5.3	75.8	92.7	3.8
DARTS- (ours)	467	4.9	76.2	93.0	4.5
NASNet-A (2018)	564	5.3	74.0	91.6	2000
DARTS (2019b)	574	4.7	73.3	91.3	0.4
SNAS (2019)	522	4.3	72.7	90.8	1.5
PC-DARTS (2020)	586	5.3	74.9	92.2	0.1
FairDARTS-B (2020b)	541	4.8	75.1	92.5	0.4
MobileNetV3 (2019)	219	5.4	75.2	92.2	≈3000
MoGA-A (2020a)	304	5.1	75.9	92.8	12
MixNet-M (2019)	360	5.0	77.0	93.3	≈3000
EfficientNet B0 (2019)	390	5.3	76.3	93.2	≈3000
NoisyDARTS-A◇	449	5.5	77.9	94.0	12
DARTS- (ours)◇	470	5.5	77.8	93.9	4.5

‡ Estimated by Wu et al. (2019).
◇ SE modules and Swish enabled.

在 RobustDARTS 提出的检验鲁棒性的多个搜索空间 S1–S4 中，DARTS- 搜索得到的模型性能优于 R-DARTS 和 SDARTS。

Table 7: Comparison in various search spaces. We report the **lowest error rate** of 3 found architectures. †: under Chen & Hsieh (2020)'s training settings where all models have 20 layers and 36 initial channels (the best is shown in boldface). ‡: under Zela et al. (2020)'s settings where CIFAR-100 models have 8 layers and 16 initial channels (The best is in boldface and underlined).

Benchmark		DARTS‡	R-DARTS‡		DARTS‡		Ours‡	PC-DARTS†	SDARTS†		Ours†
			DP	L2	ES	ADA			RS	ADV	
C10	S1	3.84	3.11	2.78	3.01	3.10	**2.68**	3.11	2.78	2.73	**2.68**
	S2	4.85	3.48	3.31	3.26	3.35	**2.63**	3.02	2.75	2.65	**2.63**
	S3	3.34	2.93	2.51	2.74	2.59	**2.42**	2.51	2.53	2.49	**2.42**
	S4	7.20	3.58	3.56	3.71	4.84	**2.86**	3.02	2.93	2.87	**2.86**
C100	S1	29.46	25.93	24.25	28.37	24.03	**22.41**	18.87	17.02	**16.88**	16.92
	S2	26.05	22.30	22.24	23.25	23.52	**21.61**	18.23	17.56	17.24	**16.14**
	S3	28.90	22.36	23.99	23.73	23.37	**21.13**	18.05	17.73	17.12	**15.86**
	S4	22.85	22.18	21.94	**21.26**	23.20	21.55	17.16	17.17	**15.46**	17.48

NAS 算法评测

NAS-Bench-201[10] 是用于衡量 NAS 算法的评测基准工具之一，DARTS- 也取得了优于其他 NAS 算法的结果，而且最好结果基本逼近了基准中最好的模型。

Table 3: Searching performance on NAS-Bench-201 (Dong & Yang, 2020). Our method robustly obtains new SOTA. Averaged on 4 runs of searching. ^{1st}: first-order, ^{2nd}: second-order

Method	Cost (hours)	CIFAR-10 valid	CIFAR-10 test	CIFAR-100 valid	CIFAR-100 test	ImageNet16-120 valid	ImageNet16-120 test
DARTS^{1st} (2019b)	3.2	39.77±0.00	54.30±0.00	15.03±0.00	15.61±0.00	16.43±0.00	16.32±0.00
DARTS^{2nd} (2019b)	10.2	39.77±0.00	54.30±0.00	15.03±0.00	15.61±0.00	16.43±0.00	16.32±0.00
GDAS (2019b)	8.7	89.89±0.08	93.61±0.09	71.34±0.04	70.70±0.30	41.59±1.33	41.71±0.98
SETN (2019a)	9.5	84.04±0.28	87.64±0.00	58.86±0.06	59.05±0.24	33.06±0.02	32.52±0.21
DARTS-	3.2	**91.03±0.44**	**93.80±0.40**	**71.36±1.51**	**71.53±1.51**	**44.87±1.46**	**45.12±0.82**
DARTS- (best)	3.2	91.55	94.36	73.49	73.51	46.37	46.34
optimal	n/a	91.61	94.37	73.49	73.51	46.77	47.31

迁移能力

DARTS-A 作为主干网在 COCO 数据集目标检测任务上也优于之前 NAS 模型的性能，mAP 达到了 32.5%。

Table 11: Transfer results on COCO datasets of various drop-in backbones.

Backbones	Params (M)	Acc	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
MobileNetV2 (2018)	3.4	72.0	28.3	46.7	29.3	14.8	30.7	38.1
SingPath NAS (2019)	4.3	75.0	30.7	49.8	32.2	15.4	33.9	41.6
MnasNet-A2 (2019)	4.8	75.6	30.5	50.2	32.0	16.6	34.1	41.1
MobileNetV3 (2019)	5.4	75.2	29.9	49.3	30.8	14.9	33.3	41.1
MixNet-M (2019)	5.0	77.0	31.3	51.7	32.4	17.0	35.0	41.9
FairDARTS-C (2020b)	5.3	77.2	31.9	51.9	33.0	17.4	35.3	43.0
DARTS-A (Ours)	5.5	77.8	32.5	52.8	34.1	18.0	36.1	43.4

综合来看，DARTS− 方法继承了 DARTS 的高效率，且在标准数据集、NAS 基准评测、R−DARTS 搜索空间中证明了其鲁棒性和有效性，在检测任务中也证明了其领域迁移的能力，从而印证了搜索方法本身的优越性，解决了当前神经网络架构搜索中的一些难题，将会对 NAS 研究和应用产生积极的推动作用。

总结及展望

本次美团在 ICLR 2021 被收录的文章 DARTS−，重新了梳理 DARTS 搜索结果不够鲁棒的原因，分析了跳跃连接的双重作用，并提出了增加带衰减系数的辅助跳跃连接来对其进行分离的方法，使得内层原生的跳跃连接只表现其作为可选操作的功能。我们同时对 R−DARTS 所依赖的特征根进行深入分析，发现了其作为性能崩塌标志会出现反例的情形。未来 DARTS− 作为高效、鲁棒且通用的搜索方法，期望在其他领域任务和落地中得到更多的拓展和应用。关于文章的更多细节，请参考原文。实验代码已经在 GitHub 上开源。

AutoML 技术可以适用于计算机视觉、语音、NLP、搜索推荐等领域，视觉智能中心 AutoML 算法团队旨在通过 AutoML 技术赋能公司业务、加速算法落地。目前该论文已经申请了专利，本文算法也集成到美团自动化视觉平台系统中，加速自动化模型生产和迭代。除了视觉场景外，我们后续将探索在搜索推荐、无人车、优选、语音等业务场景中的应用。

作者简介

祥祥、晓星、张勃、晓林等，均来自美团视觉智能中心。

参考文献

[1] Learning Transferable Architectures for Scalable Image Recognition, https://arxiv.org/abs/1707.07012.
[2] NAS-FPN: Learning scalable feature pyramid architecture for object detection, https://arxiv.org/abs/1904.07392.
[3] Auto-deeplab: Hierarchical neural architecture search for semantic image segmentation, https://arxiv.org/abs/1901.02985.
[4] DARTS-: Robustly Stepping out of Performance Collapse Without Indicators, https://openreview.net/forum?id=KLH36ELmwIB.
[5] DARTS: Differentiable Architecture Search, https://arxiv.org/pdf/1806.09055.pdf.
[6] Progressive Differentiable Architecture Search: Bridging the Depth Gap between Search and Evaluation, https://arxiv.org/pdf/1904.12760.
[7] Fair DARTS: Eliminating Unfair Advantages in Differentiable Architecture Search, https://arxiv.org/pdf/1911.12126.pdf.
[8] Understanding and Robustifying Differentiable Architecture Search, https://openreview.net/pdf?id=H1gDNyrKDS.
[9] Stabilizing Differentiable Architecture Search via Perturbation-based Regularization, https://arxiv.org/abs/2002.05283.
[10] NAS-Bench-201: Extending the Scope of Reproducible Neural Architecture Search , https://openreview.net/forum?id=HJxyZkBKDr.
[11] Deep Residual Learning for Image Recognition, https://arxiv.org/abs/1512.03385.
[12] Theory-inspired path-regularized differential network architecture search, https://arxiv.org/abs/2006.16537.

SIGIR 2021 | 广告系统位置偏差的 CTR 模型优化方案

作者：坚强、胡可、庆涛、明健、漆毅、程佳、雷军

近些年来，由于人工智能技术的高速发展，所带来的公平性问题也愈发受到关注。同样的，广告技术也存在着许多公平性问题，由于公平性问题造成的偏差对广告系统的生态会产生较大的负面影响。图 1 所示的是广告系统中的反馈环路 [1]，广告系统通过累积的用户交互反馈数据基于一定的假设去训练模型，模型对广告进行预估排序展示给用户，用户基于可看到的广告进行交互进而累积到数据中。在该环路中，位置偏差、流行度偏差等各种不同类型的偏差会在各环节中不断累积，最终导致广告系统的生态不断恶化，形成“强者愈强、弱者愈弱”的马太效应。

由于偏差对广告系统和推荐系统的生态有着极大的影响，针对消除偏差的研究工作也在不断增加。比如国际信息检索会议 SIGIR 在 2018 年和 2020 年组织了一些关注于消除偏差主题的专门会议，同时也给一些基于偏差和公平性的论文颁发了最佳论文奖（Best Paper）[2,3]。KDD Cup 2020 的其中一个赛道也基于电子商务推荐中的流行度偏差进行开展 [1]。

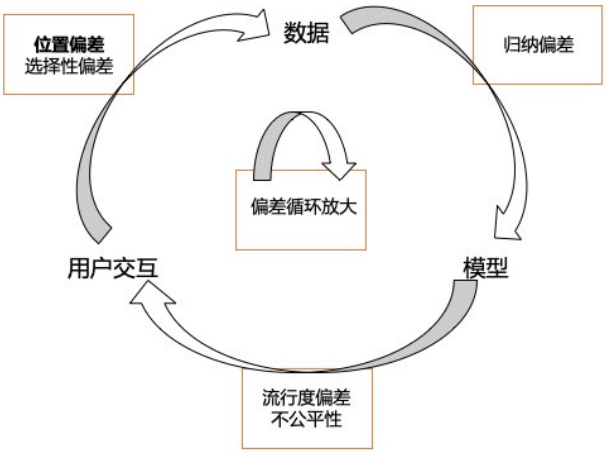


图 1　广告系统中的反馈环路，各种偏差被不断循环累积

美团到店广告平台算法团队基于多年来在广告领域上积累的经验，一直在数据偏差等业界挑战性问题不断进行深入优化与算法创新。在之前分享的《KDD Cup 2020 Debiasing 比赛冠军技术方案与广告业务应用》一文[4]中，团队分享了在 KDD Cup 比赛中取得冠军的选择性偏差以及流行度偏差的解决方案，同时也分享了在广告业务上偏差优化的技术框架。

本文基于这一技术框架进行继续介绍，聚焦于位置偏差问题的最新进展，并详细地介绍团队在美团广告取得显著业务效果的位置偏差 CTR 模型优化方案，以该方案为基础形成的论文《Deep Position-wise Interaction Network for CTR Prediction》也被国际顶级会议 SIGIR 2021 录用。

1. 背景

美团到店广告平台算法团队基于美团和点评双侧的广告业务场景，不断进行广告前沿技术的深入优化与算法创新。在大多数广告业务场景下，广告系统被分为四个模块，分别是触发策略、创意优选、质量预估以及机制设计，这些模块构成一个广告投放漏斗从海量广告中过滤以及精选出优质广告投放给目标用户。其中，触发策略从海量广告中挑选出满足用户意图的候选广告集合，创意优选负责候选广告的图片和文本生成，质量预估结合创意优选的结果对每一个候选广告进行质量预估，包括点击率（CTR）预估、转化率（CVR）预估等，机制排序结合广告质量以及广告出价进行优化排序。在本文中，我们也将广告称之为 item。

CTR 预估，作为质量预估的一个环节，是计算广告中最核心的算法之一。在每次点击付费（CPC）计费模式下，机制设计可以简单地按每千次展示收入（eCPM）来对广告进行排序以取得广告收入最大化。由于 eCPM 正比于 CTR 和广告出价（bid）的乘积。因此，CTR 预估会直接影响到广告的最终收入和用户体验。为了有更高的 CTR 预估精度，CTR 预估从早期的 LR[5]、FM[6]、FFM[7] 等支持大规模稀疏特征的模型，到 XGBoost[8]、LightGBM[9] 等树模型的结合，再到 Wide&Deep[10]、Deep&Cross[11]、DeepFM[12]、xDeepFM[13] 等支持高阶特征交叉的深度学习模型，

进一步演化到 DIN[14]、DIEN[15]、DSIN[16] 等结合用户行为序列的深度学习模型，一直作为工业界以及学术界研究的热点领域之一，被不断探索和不断创新。

由于 CTR 预估模型的训练通常采用曝光点击数据，该数据是一种隐式反馈数据，所以会不可避免地产生各种偏差问题。其中，位置偏差因对 CTR 影响极大而备受关注。如图 2 所示，随机流量上不同位置的 CTR 分布反应了用户通常倾向于点击靠前位置的广告，并且 CTR 会随着曝光位置的增大而迅速下降。因此，直接在曝光点击数据上进行训练，模型不可避免地会偏向于靠前位置的广告集合，造成位置偏差问题。图 2 显示正常流量相比随机流量 CTR 分布更加集中在高位置广告上，通过反馈环路，这一问题将不断地放大，并且进一步损害模型的性能。因此，解决好位置偏差问题不仅能够提升广告系统的效果，而且还能平衡广告系统的生态，促进系统的公平性。

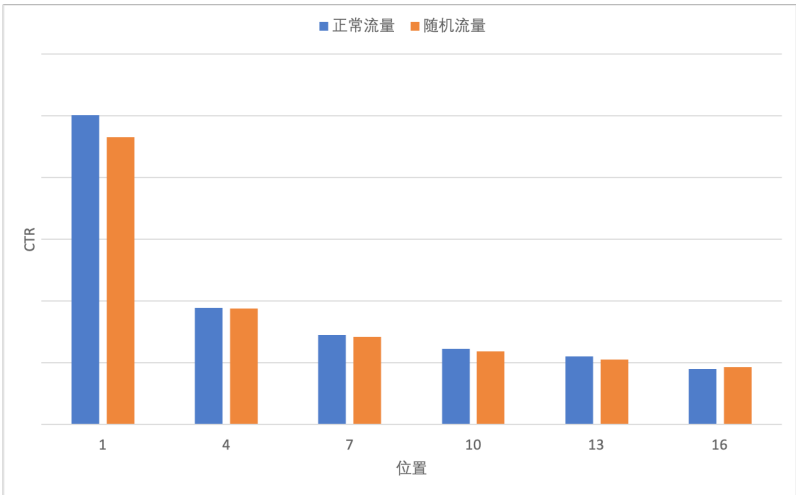


图 2　美团广告正常流量和随机流量在不同位置上的 CTR 分布

广告最终的真实曝光位置信息在线上预估时是未知的，这无疑进一步增大了位置偏差问题的解决难度。现有的解决位置偏差的方法可以大致分为以下两种：

- **神经网络位置特征建模**：该方法将位置建模为神经网络中的特征，由于在预估过程中并不知道真实位置信息，故而有些方法 [17-19] 把位置信息放于网络的 Wide 部分，在线下训练时使用真实位置，在线上预估时使用固定位置，这种

方法由于其简单性和有效性，在工业界被广泛应用。为了在线上预估时无需使用位置信息，如图 3 所示，PAL[20] 将样本的 CTR 建模为 ProbSeen 乘以 pCTR，其中 ProbSeen 仅使用位置特征建模，而 pCTR 使用其他信息建模，在线上只使用 pCTR 作为 CTR 预估值。

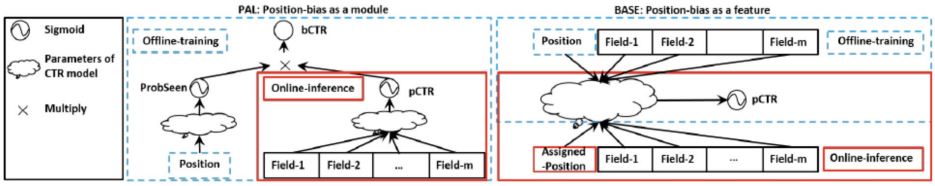


图 3　PAL 框架

- **Inverse Propensity Weighting（IPW）**：该方法被学术界广泛研究 [21–29]，其在模型训练时给不同曝光位置的样本赋予不同的样本权重，直观地看，应该将具有较低接收反馈倾向的广告样本（曝光位置靠后的广告）分配较高的权重。因此，这种方法的难点就在于不同位置的样本权重如何确定，一个简单的方法是使用广告随机展示的流量来准确地计算位置 CTR 偏差，但不可避免地损害用户体验。故而，许多方法致力于在有偏的流量上来准确地预估位置偏差。

上述的方法通常基于一个较强的假设，即点击伯努利变量 C 依赖于两个潜在的伯努利变量 E 和 R，如下式所示：

$$p(C = 1|u, c, i, k) = p(E = 1|k, [s])p(R = 1|u, c, i), \qquad (1)$$

其中，等式左边指的是用户 u 在上下文 c 中点击第 k 个广告 i 的概率，我们定义上下文 c 为实时的请求信息。等式右边第一项指的是位置 k 被查看的概率，其中 $[s]$ 通常为上下文 c 的一个子集，大部分方法假设 $[s]$ 为空集，即位置 k 被查看的概率仅与 k 有关。等式右边第二项指的是相关性概率（例如用户 u 在上下文 c 中对广告 i 的的真实兴趣）。上述方法通常显式或隐式地估计查看概率，然后利用反事实推理（Counterfactual Inference）得出相关性概率，最终在线上将相关性概率作为 CTR 的预估值。训练和预估之间位置信息的不同处理将不可避免地导致线下线上间的不一致问题，进一步导

致次优的模型性能。

此外，已有方法通常假设查看概率仅依赖于位置及部分上下文信息，其假设过于简单。不同的用户通常具有不同的浏览习惯，有些用户可能倾向于浏览更多 item，而有些用户通常能快速做出决定，并且同一个用户在不同的上下文中搜索意图中也会有不同的位置偏好，例如商场等地点词的搜索往往意图不明确导致高低位置的 CTR 差异并不大。故而，位置偏差与用户，上下文有关，甚至可能与广告本身也有关，建模它们间的关系能更好地解决位置偏差问题。

不同于上述的方法，本文提出了一个基于深度位置交叉网络（Deep Position-wise Interaction Network）（DPIN）模型的多位置预估方法去有效地直接建模 $CTR_k^j = p(C=1 \mid u,c,i,k)$ 来提高模型性能，其中 CTR_k^j 是第 j 个广告在第 k 个位置的 CTR 预估值。该模型有效地组合了所有候选广告和位置，以预估每个广告在每个位置的 CTR，实现线下线上的一致性，并在在线服务性能限制的情况下支持位置、用户、上下文和广告之间的深度非线性交叉。广告的最终序可以通过最大化 $\sum CTR_k^j bid^j$ 来确定，其中 bid^j 为广告的出价，本文在线上机制采用一个位置自顶向下的贪婪算法去得到广告的最终序。本文的贡献如下：

- 本文在 DPIN 中使用具有非线性交叉的浅层位置组合模块，该模块可以并行地预估候选广告和位置组合的 CTR，达到线下线上的一致性，并大大改善了模型性能。
- 不同于以往只对候选广告进行用户兴趣建模，本次首次提出对候选位置也进行用户兴趣建模。DPIN 应用一个深度位置交叉模块有效地学习位置，用户兴趣和上下文之间的深度非线性交叉表示。
- 根据对于位置的新处理方式，本文提出了一种新的评估指标 PAUC（Position-wise AUC），用于测量模型在解决位置偏差问题上的模型性能。本文在美团广告的真实数据集上进行了充分的实验，验证了 DPIN 在模型性能和服务性能上都能取得很好的效果。同时本文还在线上部署了 A/B Test，验证了 DPIN 与高度优化的已有基线相比有显著提升。

2. 深度位置交叉网络（Deep Position-wise Interaction Network

本节主要介绍深度位置交叉网络（Deep Position-wise Interaction Network）（DPIN）模型。如图 4 所示，DPIN 模型由三个模块组成，分别是处理 J 个候选广告的基础模块（Base Module），处理 K 个候选位置的深度位置交叉模块（Deep Position-wise Interaction Module）以及组合 J 个广告和 K 个位置的位置组合模块（Position-wise Combination Module），不同模块需预估的样本数量不一样，复杂模块预估的样本数量少，简单模块预估的样本数量多，由此来提高模型性能和保障服务性能。通过这三个模块的组合，DPIN 模型有能力在服务性能的限制下预估每个广告在每个位置上的 CTR，并学习位置信息和其他信息的深度非线性交叉表示。下文将会详细地介绍这三个模块。

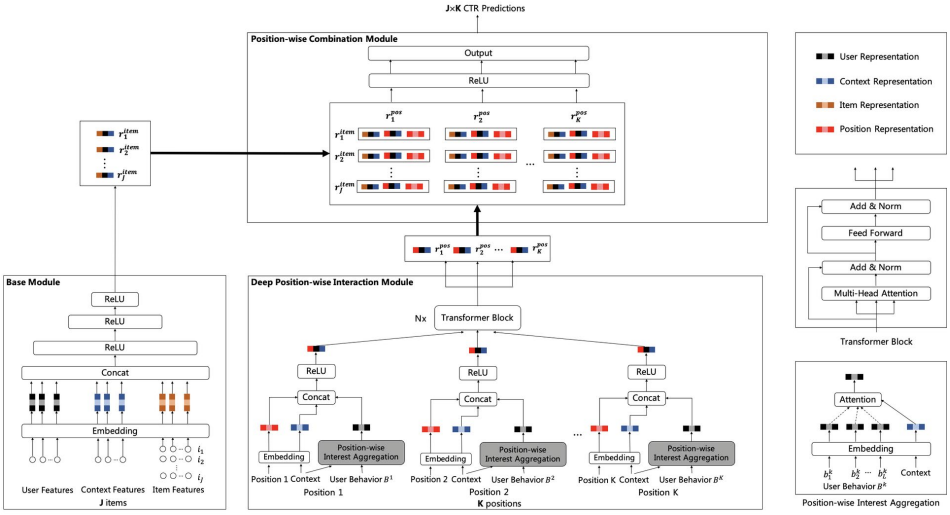


图 4　Deep Position-wise Interaction Network 模型结构

2.1 基础模块（Base Module）

与大多数深度学习 CTR 模型 [10-16] 类似，本文采用 Embedding 和 MLP（多层感知机）的结构作为基础模块。 对于一个特定请求请求，基础模块将用户、上下文和 J 个候选广告作为输入，将每个特征通过 Embedding 进行表示，拼接 Embedding 表示

输入多层 MLP，采用 ReLU 作为激活函数，最终可以得到每个广告在该请求下的表示。第 j 个广告的表示 r_j^{item} 可以通过如下公式得到：

$$\mathbf{u} = Concat(E(u_1), ..., E(u_m)), \quad (2)$$

$$\mathbf{c} = Concat(E(c_1), ..., E(c_n)), \quad (3)$$

$$\mathbf{i_j} = Concat(E(i_1^j), ..., E(i_o^j)), \quad (4)$$

$$\mathbf{r_j^{item}} = MLP(Concat(\mathbf{u}, \mathbf{c}, \mathbf{i_j})), \quad (5)$$

其中 $u_1,\cdots,u_m$， $c_1,\cdots,c_m$， $i_1^j,\cdots,i_o^j$ 分别是当前用户特征集合、当前上下文特征集合以及第 j 个广告的特征集合， $E(\cdot)\in\mathbb{R}$ 是 Embedding 映射。

2.2 深度位置交叉模块（Deep Position-wise Interaction Module）

在大多数业务场景中，基础模块通常已经被高度优化，包含了大量特征甚至用户序列等信息，其目的是捕捉用户在该上下文中对不同广告的兴趣。因此，基础模块的推理时间复杂度通常较大，直接在基础模块中加入位置特征对所有广告在所有位置上进行 CTR 预估是不可接受的。因此，本文提出了一个与基础模块并行的深度位置交叉模块，不同于针对广告进行兴趣建模的基础模块，该模块针对于位置进行兴趣建模，学习每个位置与上下文及用户兴趣的深度非线性交叉表示。

在深度位置交叉模块中，我们提取用户在每个位置的行为序列，将其用于各位置上的用户兴趣聚合，这样可以消除整个用户行为序列上的位置偏差。接着，我们采用一层非线性全连接层来学习位置、上下文与用户兴趣非线性交叉表示。最后，为了聚合用户在不同位置上的序列信息来保证信息不被丢失，我们采用了 Transformer[30] 来使得不同位置上的行为序列表示可以进行交互。

位置兴趣聚合（Position-wise Interest Aggregation）。我们令 $B_k = b_1^k, b_2^k, \cdots, b_L^k$ 为用户在第 k 个位置的历史行为序列，其中 $b_l^k = [v_l^k, c_l^k]$ 为用户在第 k 个位置上的历史第 l 个行为记录， v_l 为点击的 item 特征集合， c_l^k 为发生该行为时的上下文（包括搜索关键词、请求地理位置、一周中的第几天、一天中的第几个小时等），行为记录的 Embedding

表示 $\mathbf{b_l^k}$ 可以通过下式得到：

$$\mathbf{b_l^k} = Concat(E(v_1^{k_l}), ..., E(v_o^{k_l}), E(c_1^{k_l}), ..., E(c_n^{k_l}), E(dif^{k_l})), \quad (6)$$

其中 $v_1^{k_l}, v_o^{k_l}$， $c_1^{k_l}, c_n^{k_l}$ 分别为 v_l^k 和 c_l^k 的特征集合， dif^{kl} 为该行为与当前上下文的时间差。

第 k 个位置行为序列的聚合表示 $\mathbf{b_k}$ 可以通过注意力机制获取，如以下公式所示：

$$a_l^k = Concat(\mathbf{b_l^k}, \mathbf{c})\mathbf{W_a} + b_a, \quad (7)$$

$$\mathbf{b_k} = \sum_{l=1}^{L} \frac{\exp(a_l^k)}{\sum_{i=1}^{L} \exp(a_i^k)} \mathbf{b_l^k}, \quad (8)$$

其引入当前上下文 $\mathbf{c}$ 去计算注意力权重，对于与上下文越相关的行为可以给予越多的权重。

位置非线性交叉（Position-wise Non-linear Interaction）：我们采用一层非线性全连接层来学习位置、上下文与用户兴趣非线性交叉表示，如下式所示：

$$\mathbf{v_k} = ReLU(Concat(E(k), \mathbf{c}, \mathbf{b_k})\mathbf{W_v} + \mathbf{b_v}), \quad (9)$$

其中， $\mathbf{W_v}, \mathbf{bv}$, 将拼接的向量映射到 $dmodel$ 维度。

Transformer Block：如果将 V_k 直接作为第 k 个位置的非线性交叉表示，那么会丢失用户在其他位置上的行为序列信息。因此，我们采用 Transformer 去学习不同位置兴趣的交互。令 $\mathbf{Q} = \mathbf{K} = \mathbf{V} = Concat(\mathbf{v_1}, \mathbf{v_2}, \cdots, \mathbf{v_K})$ 为 Transformer 的输入，Tranformer 的多头自注意力结构可以由以下公式表示：

$$MultiHead(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = Concat(\mathbf{head_1}, ..., \mathbf{head_h})\mathbf{W^O}, \quad (10)$$

$$\mathbf{head_i} = Attention(\mathbf{QW_i^Q}, \mathbf{KW_i^K}, \mathbf{VW_i^V}), \quad (11)$$

$$Attention(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = softmax(\frac{\mathbf{QK^T}}{\sqrt{d_k}})\mathbf{V}, \quad (12)$$

其中， $dk = dmodel / h$ 是每个头的维度。因为 $\mathbf{v_k}$ 中已经包含位置信息，故而我们不需要 Transformer 中的位置编码。同样的，我们也沿用 Transformer 中的前馈网络

（Position-wise Feed-forward Network）、残差连接（Residual Connections）以及层标准化（Layer Normalization）。N 个 Transformer Block 会被使用去加深网络。

最终，深度位置交叉模块会产出每个位置的深度非线性交叉表示，其中第 k 个位置被表示为 r_k^{pos} 。

2.3 位置组合模块（Position-wise Combination Module）

位置组合模块的目的是去组合 J 个广告和 K 个位置来预估每个广告在每个位置上的 CTR，我们采用一层非线性全连接层来学习广告、位置、上下文和用户的非线性表示，第 j 个广告在第 k 个位置上的 CTR 可以由如下公式得出：

$$CTR_k^j = \sigma(ReLU(Concat(\mathbf{r_j^{item}}, \mathbf{r_k^{pos}}, E(k))\mathbf{W_1} + \mathbf{b_1})\mathbf{W_2} + b_2), \quad (13)$$

其中包括了一层非线性连接层和一层输出层，是 $E(k)$ 位置 k 的 embedding 表示，$\sigma(\cdot)$ 是 sigmoid 函数。

整个模型可以使用真实位置通过批量梯度下降法进行训练学习，我们采用交叉熵作为我们的损失函数。

3. 实验

在本节中，我们评估 DPIN 的模型性能和服务性能，我们将详细描述实验设置和实验结果。

3.1 实验设置

数据集：我们使用美团搜索关键词广告数据集训练和评估我们的 CTR 模型。训练数据量达到数亿，测试数据量大约一千万。测试集被划分为两个部分，一部分是线上收集的常规流量日志，另一部分是线上 Top-k 随机的探索流量日志。Top-k 随机的探索流量日志是更适合用来评估位置偏差问题，因为它大大削弱了相关性推荐对位置偏差的影响。

评估指标：我们使用 AUC（Area Under ROC）作为我们的评估指标之一。为了更好的针对位置偏差问题进行评估，我们提出 PAUC（Position-wise AUC）作为我们的另一个评估指标，其由以下公式计算：

$$PAUC = \frac{\sum_{k=1}^{K} \#impression_k PAUC@k}{\sum_{k=1}^{K} \#impression_k}, \tag{14}$$

其中，$\#impression_k$ 是第 k 个位置的曝光数量，$PAUC@k$ 是第 k 个位置曝光数据的 AUC。PAUC 指标衡量每个位置上相关性排序的质量，忽略了位置偏差对排序质量的影响。

对比的方法。为了公平且充分地对比不同模型的效果，我们所有实验中所使用的模型输入使用等量且深度结合美团业务的特征，不同模型中的相同模块都使用一致的参数，并且对比的基线 DIN[14] 模型经过高度优化，以下为我们具体进行对比的实验：

- **DIN**：该模型训练和预估时都没有使用位置信息。
- **DIN+PosInWide**：这个方法在网络的 Wide 部分建模位置特征，在评估时采用第一个位置作为位置特征的默认值去评估。
- **DIN+PAL**：这个方法采用 PAL 框架去建模位置信息。
- **DIN+ActualPosInWide**：这个方法在网络的 Wide 部分建模位置特征，在评估时采用真实位置特征去评估。
- **DIN+Combination**：这个方法在 DIN 的基础上添加了位置组合模块，评估时采用真实位置特征去评估。
- **DPIN-Transformer**：这个方法在我们提出的 DPIN 模型上去除了 Transformer 结构，来验证 Transformer 的作用。
- **DPIN**：这是我们提出的 DPIN 模型。
- **DPIN+ItemAction**：我们在 DPIN 的基础模块 MLP 层前添加深度位置交叉模块，并在位置兴趣聚合和位置非线性交叉中引入候选广告的信息，这个实验是我们方法模型性能的理论上界，然而服务性能是不可接受的。

3.2 离线评估

模型	常规流量		随机流量	
	AUC(%)	PAUC(%)	AUC(%)	PAUC(%)
DIN	0	0	0	0
DIN+PosInWide	-1.22	+0.19	-1.11	+0.16
DIN+PAL	-0.83	+0.38	-0.73	+0.31
DIN+AcutalPosInWide	+1.10	+0.19	+1.02	+0.16
DIN+Combination	+1.52	+0.82	+1.49	+0.71
DPIN-Transformer	+1.43	+0.58	+1.48	+0.60
DPIN	**+1.76**	**+1.26**	**+1.79**	**+1.27**
DPIN+ItemAction	+1.81	+1.33	+1.83	+1.33

表 1　在常规流量和随机流量上的离线实验评估对比结果

表 1 展示了我们所进行的对比方法在常规流量和随机流量上的离线实验评估结果，其中的数值为各个模型相对于 DIN 模型的效果差异，我们首先分析在常规流量上不同方法的差异。与 DIN 相比，DIN+PosInWide 和 DIN+PAL 的模型在 AUC 指标上有所下降，但在 PAUC 上有所提升，这表明了这两种方法都可以有效地缓解位置偏差，但会导致离线和在线之间的不一致。

DIN+AcutalPosInWide 通过在评估过程中引入实际位置来解决不一致问题，这可以通过位置组合模块来实现，但是在 wide 部分建模位置会导致位置特征只是一个偏差，不能提升 PAUC 指标，虽然能更准确地预估各位置上的 CTR，但没有对数据中固有的位置偏差进行更好的学习。

DIN+Combination 通过在 DIN 中引入位置组合模块，我们取得了 1.52% 的 AUC 增益和 0.82% 的 PAUC 增益，达到线下线上一致性的同时也进一步地缓解了位置偏差，这个结果说明了位置偏差与上下文、用户等信息不独立，在不同的用户及上下文中会有不同的位置偏差。更进一步的，DPIN 建模位置、上下文、用户的深度非线性交叉关系，也消除了用户行为序列中存在的位置偏差，对比 DIN+Combination 取得了 0.24% 的 AUC 增益以及 0.44% 的 PAUC 增益。

DPIN-Transformer 的效果说明了丢失其他位置的用户兴趣会影响模型的性能，因为这将损失大部分用户兴趣信息。对比 DPIN 和 DPIN+ItemAction，我们发现 DPIN 的模型性能接近于这个暴力方法，说明 DPIN 模型逼近了我们方法的理论上界。最终，相较于我们的线上基线模型 DIN+PosInWide，DPIN 取得了 2.98% 的 AUC 增益和 1.07% 的 PAUC 增益，这在我们的业务场景中是一次极大的 AUC 和 PAUC 提升。

为了确保我们的方法能够学习位置偏差而不是单纯地过度拟合系统的选择性偏差，我们进一步在随机流量上评估我们的方法。表 1 的结果表明了在常规流量和随机流量上不同方法之间的差异是一致的，这说明了就算系统的推荐结果有了巨大的差异，该模型仍能有效地学习到在不同用户及上下文中的位置偏差，模型学到的位置偏差受系统推荐列表的影响很小，这也说明我们的模型可以不受系统选择性偏差的影响从而泛化到其他推荐方法的流量上。

3.3 服务性能

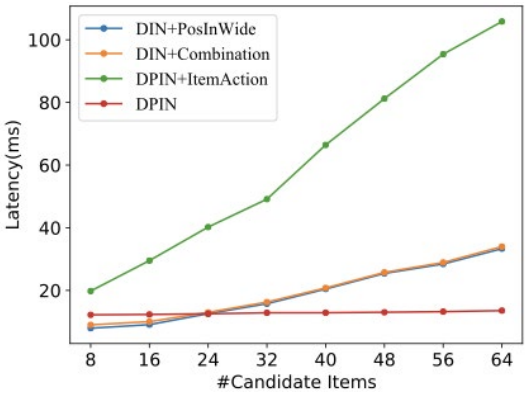


图 5　不同的方法下服务延迟随着不同候选广告数量的变化图

我们从数据集中检索出一些具有不同候选广告数量的请求，以评估不同候选广告数量下的服务性能。如图 5 所示，由于用户序列操作的延迟在服务延迟中占了很大比例，因此与 DIN 模型相比，位置组合模块服务延迟可以忽略不计。DPIN 的服务延迟随着广告数量的增加而缓慢增加，这是因为相比较于 DIN，DPIN 将用户序列从基础模块移动到深度位置交叉模块，而深度位置交叉模块的服务性能与广告数量无关。与

DIPIN+ItemAction 方法相比，DPIN 在服务性能方面有了很大的改进，对模型性能的损害很小，这表明我们提出的方法既高效又有效。

3.4 在线评估

我们在线上部署了 A/B 测试，有稳定的结果表明，与基线相比，DPIN 在 CTR 上提高了 2.25%，在 RPM（每千次展示收入）上提高了 2.15%。如今，DPIN 已在线部署并服务于主要流量，为业务收入的显着增长做出了贡献。

4. 总结与展望

在本文中，我们提出了一种新颖的深度位置交叉网络模型（Deep Position-wise Interaction Network）以缓解位置偏差问题，该模型有效地组合了所有候选广告和位置以估算每个广告在每个位置的点击率，实现了离线和在线之间的一致性。该模型设计了位置、上下文和用户之间的深层非线性交叉，可以学习到不同用户、不同上下文中的位置偏差。为了评估位置偏向问题，我们提出了一种新的评估指标 PAUC，离线实验表明，所提出的 DPIN 的效果和效率均优于已有方法。目前，DPIN 已部署到美团搜索关键词广告系统并服务于主要流量。

值得一提的是，我们的并行组合思想不仅可以用在广告和位置的组合上，也可以用在广告和创意的组合等广告领域常见的组合排序问题。在未来，我们将在这些问题上继续实践我们的方法，并进一步地设计更完善的网络结构来解决类似的组合排序问题。我们也将在偏差领域上进行更多的探索，解决更多的问题，进一步维护广告系统的生态平衡。

作者简介

坚强、胡可、庆涛、明健、漆毅、程佳、雷军等，均来自美团广告平台技术部。

参考文献

[1] Chen, Jiawei, et al. “Bias and Debias in Recommender System: A Survey and

Future Directions." arXiv preprint arXiv:2010.03240 (2020).
[2] Cañamares, Roc í o, and Pablo Castells. "Should I follow the crowd? A probabilistic analysis of the effectiveness of popularity in recommender systems." The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval. 2018.
[3] Morik, Marco, et al. "Controlling fairness and bias in dynamic learning-to-rank." Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval. 2020.
[4]《KDD Cup 2020 Debiasing 比赛冠军技术方案及在美团的实践》。
[5] Richardson, Matthew, Ewa Dominowska, and Robert Ragno. "Predicting clicks: estimating the click-through rate for new ads." Proceedings of the 16th international conference on World Wide Web. 2007.
[6] Rendle, Steffen. "Factorization machines." 2010 IEEE International Conference on Data Mining. IEEE, 2010.
[7] Juan, Yuchin, et al. "Field-aware factorization machines for CTR prediction." Proceedings of the 10th ACM conference on recommender systems. 2016.
[8] Chen, Tianqi, and Carlos Guestrin. "Xgboost: A scalable tree boosting system." Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining. 2016.
[9] Ke, Guolin, et al. "Lightgbm: A highly efficient gradient boosting decision tree." Advances in neural information processing systems 30 (2017): 3146-3154.
[10] Cheng, Heng-Tze, et al. "Wide & deep learning for recommender systems." Proceedings of the 1st workshop on deep learning for recommender systems. 2016.
[11] Wang, Ruoxi, et al. "Deep & cross network for ad click predictions." Proceedings of the ADKDD'17. 2017. 1-7.
[12] Guo, Huifeng, et al. "DeepFM: a factorization-machine based neural network for CTR prediction." arXiv preprint arXiv:1703.04247 (2017).
[13] Lian, Jianxun, et al. "xdeepfm: Combining explicit and implicit feature interactions for recommender systems." Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. 2018.
[14] Zhou, Guorui, et al. "Deep interest network for click-through rate prediction." Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. 2018.
[15] Zhou, Guorui, et al. "Deep interest evolution network for click-through rate prediction." Proceedings of the AAAI conference on artificial intelligence. Vol. 33. No. 01. 2019.
[16] Feng, Yufei, et al. "Deep session interest network for click-through rate prediction." arXiv preprint arXiv:1905.06482 (2019).
[17] Ling, Xiaoliang, et al. "Model ensemble for click prediction in bing search ads." Proceedings of the 26th International Conference on World Wide Web Companion. 2017.

[18] Zhao, Zhe, et al. “Recommending what video to watch next: a multitask ranking system.” Proceedings of the 13th ACM Conference on Recommender Systems. 2019.
[19] Haldar, Malay, et al. “Improving Deep Learning For Airbnb Search.” Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. 2020.
[20] Guo, Huifeng, et al. “PAL: a position-bias aware learning framework for CTR prediction in live recommender systems.” Proceedings of the 13th ACM Conference on Recommender Systems. 2019.
[21] Wang, Xuanhui, et al. “Learning to rank with selection bias in personal search.” Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval. 2016.
[22] Joachims, Thorsten, Adith Swaminathan, and Tobias Schnabel. “Unbiased learning-to-rank with biased feedback.” Proceedings of the Tenth ACM International Conference on Web Search and Data Mining. 2017.
[23] Ai, Qingyao, et al. “Unbiased learning to rank with unbiased propensity estimation.” The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval. 2018.
[24] Wang, Xuanhui, et al. “Position bias estimation for unbiased learning to rank in personal search.” Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining. 2018.
[25] Agarwal, Aman, et al. “Estimating position bias without intrusive interventions.” Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining. 2019.
[26] Hu, Ziniu, et al. “Unbiased lambdamart: an unbiased pairwise learning-to-rank algorithm.” The World Wide Web Conference. 2019.
[27] Ovaisi, Zohreh, et al. “Correcting for selection bias in learning-to-rank systems.” Proceedings of The Web Conference 2020. 2020.
[28] Yuan, Bowen, et al. “Unbiased Ad click prediction for position-aware advertising systems.” Fourteenth ACM Conference on Recommender Systems. 2020.
[29] Qin, Zhen, et al. “Attribute-based propensity for unbiased learning in recommender systems: Algorithm and case studies.” Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. 2020.
[30] Vaswani, Ashish, et al. “Attention is all you need.” arXiv preprint arXiv:1706.03762 (2017).

算法

智能搜索模型预估框架 Augur 的建设与实践

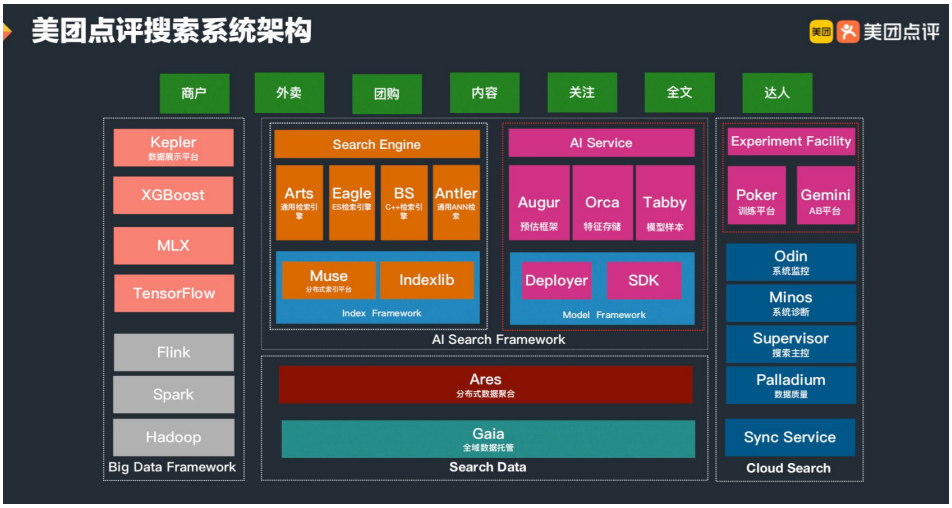
作者：朱敏　紫顺　乐钦　洪晨　乔宇　武进　孝峰　俊浩等

1. 背景

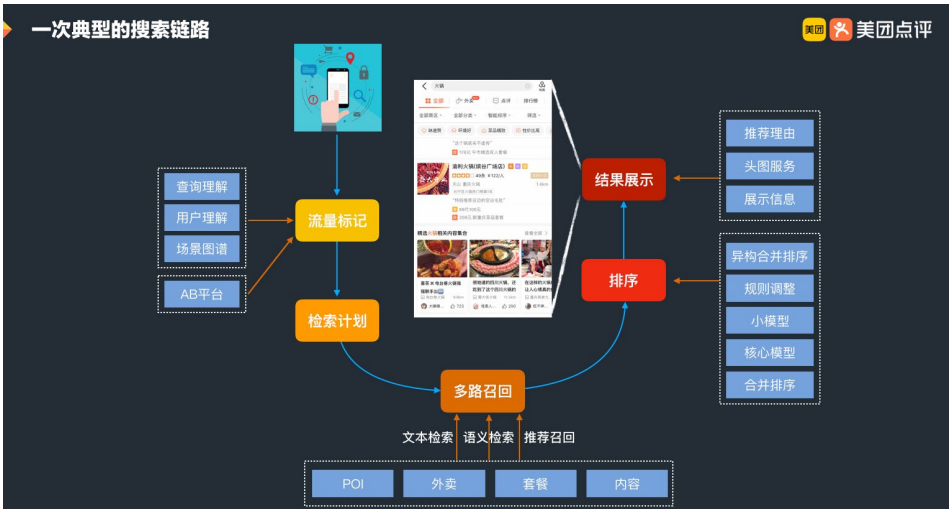
在过去十年，机器学习在学术界取得了众多的突破，在工业界也有很多应用落地。美团很早就开始探索不同的机器学习模型在搜索场景下的应用，从最开始的线性模型、树模型，再到近两年的深度神经网络、BERT、DQN 等，并在实践中也取得了良好的效果与产出。

本文将与大家探讨美团搜索与 NLP 部使用的统一在线预估框架 Augur 的设计思路、效果、优势与不足，希望对大家有所帮助或者启发。

搜索优化问题，是个典型的 AI 应用问题，而 AI 应用问题首先是个系统问题。经历近 10 年的技术积累和沉淀，美团搜索系统架构从传统检索引擎升级转变为 AI 搜索引擎。当前，美团搜索整体架构主要由搜索数据平台、在线检索框架及云搜平台、在线 AI 服务及实验平台三大体系构成。在 AI 服务及实验平台中，模型训练平台 Poker 和在线预估框架 Augur 是搜索 AI 化的核心组件，解决了模型从离线训练到在线服务的一系列系统问题，极大地提升了整个搜索策略迭代效率、在线模型预估的性能以及排序稳定性，并助力商户、外卖、内容等核心搜索场景业务指标的飞速提升。



首先，让我们看看在美团 App 内的一次完整的搜索行为主要涉及哪些技术模块。如下图所示，从点击输入框到最终的结果展示，从热门推荐，到动态补全、最终的商户列表展示、推荐理由的展示等，每一个模块都要经过若干层的模型处理或者规则干预，才会将最适合用户（指标）的结果展示在大家的眼前。



为了保证良好的用户体验，技术团队对模型预估能力的要求变得越来越高，同时模型与特征的类型、数量及复杂度也在与日俱增。算法团队如何尽量少地开发和部署上

线，如何快速进行模型特征的迭代？如何确保良好的预估性能？在线预估框架 Augur 应运而生。经过一段时间的实践，Augur 也有效地满足了算法侧的需求，并成为美团搜索与 NLP 部通用的解决方案。下面，我们将从解读概念开始，然后再分享一下在实施过程中我们团队的一些经验和思考。

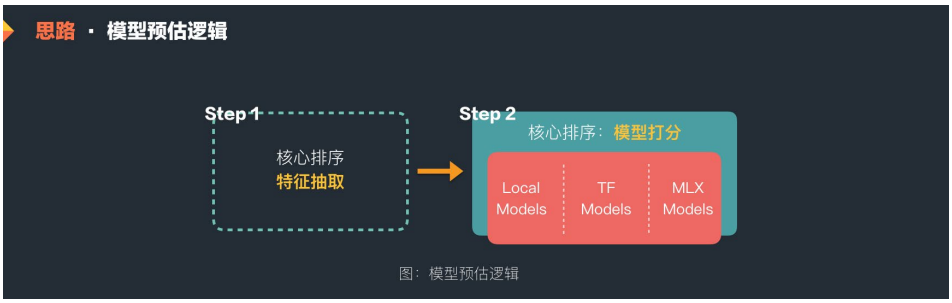
2. 抽象过程：什么是模型预估

其实，模型预估的逻辑相对简单、清晰。但是如果要整个平台做得好用且高效，这就需要框架系统和工具建设（一般是管理平台）两个层面的配合，需要兼顾需求、效率与性能。

那么，什么是模型预估呢？如果忽略掉各种算法的细节，我们可以认为模型是一个函数，有一批输入和输出，我们提供将要预估文档的相关信息输入模型，并根据输出的值（即模型预估的值）对原有的文档进行排序或者其他处理。

纯粹从一个工程人员视角来看：模型可以简化为一个公式（ 举例：f(x1,x2)= ax1 + bx2 +c ），训练模型是找出最合适的参数 abc。所谓特征，是其中的自变量 x1 与 x2，而模型预估，就是将给定的自变量 x1 与 x2 代入公式，求得一个解而已。（当然实际模型输出的结果可能会更加复杂，包括输出矩阵、向量等等，这里只是简单的举例说明。）

所以在实际业务场景中，一个模型预估的过程可以分为两个简单的步骤：第一步，特征抽取（找出 x1 与 x2）；第二步，模型预估（执行公式 f，获得最终的结果）。



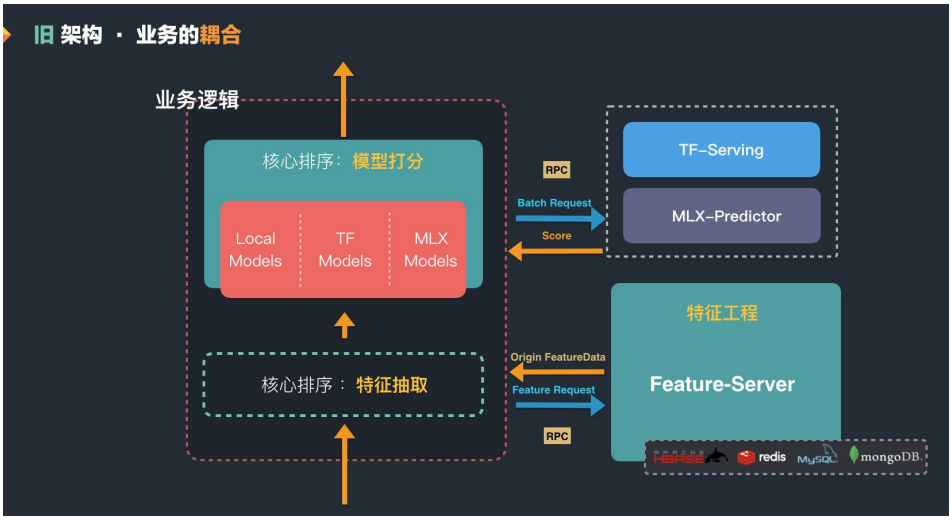
图：模型预估逻辑

模型预估很简单，从业务工程的视角来看，无论多复杂，它只是一个计算分数的过程。对于整个运算的优化，无论是矩阵运算，还是底层的 GPU 卡的加速，业界和美团内部都有比较好的实践。美团也提供了高性能的 TF-Serving 服务（参见《基于 TensorFlow Serving 的深度学习在线预估》一文）以及自研的 MLX 模型打分服务，都可以进行高性能的 Batch 打分。基于此，我们针对不同的模型，采取不同的策略：

- **深度学习模型**：特征多，计算复杂，性能要求高；我们将计算过程放到公司统一提供的 TF-Serving/MLX 预估服务上；
- **线性模型、树模型**：搜索场景下使用的特征相对较少，计算逻辑也相对简单，我们将在构建的预估框架内部再构建起高性能的本机求解逻辑，从而减少 RPC。

图：模型预估逻辑

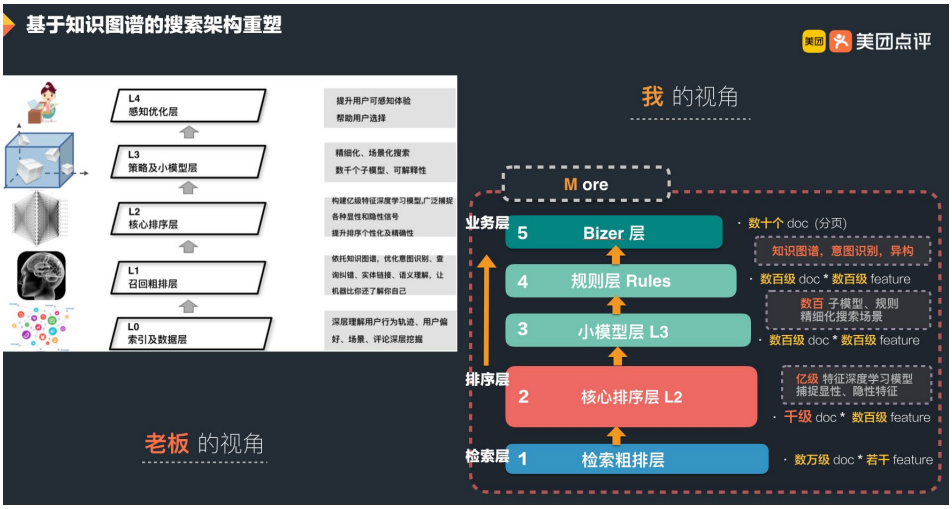
这一套逻辑很简单，构建起来也不复杂，所以在建设初期，我们快速在主搜的核心业务逻辑中快速实现了这一架构，如下图所示。这样的一个架构使得我们可以在主搜的核心排序逻辑中，能够使用各类线性模型的预估，同时也可以借助公司的技术能力，进行深度模型的预估。关于特征抽取的部分，我们也简单实现了一套规则，方便算法同学可以自行实现一些简单的逻辑。



3. 预估框架思路的改变

3.1 老框架的局限

旧架构中模型预估与业务逻辑耦合的方式，在预估文档数和特征数量不大的时候可以提供较好的支持。但是，从 2018 年开始，搜索业务瓶颈开始到来，点评事业部开始对整个搜索系统进行升级改造，并打造基于知识图谱的分层排序架构（详情可以参见点评搜索智能中心在 2019 年初推出的实践文章《大众点评搜索基于知识图谱的深度学习排序实践》）。这意味着：更多需要模型预估的文档，更多的特征，更深层次的模型，更多的模型处理层级，以及更多的业务。在这样的需求背景下，老框架开始出现了一些局限性，主要包括以下三个层面：



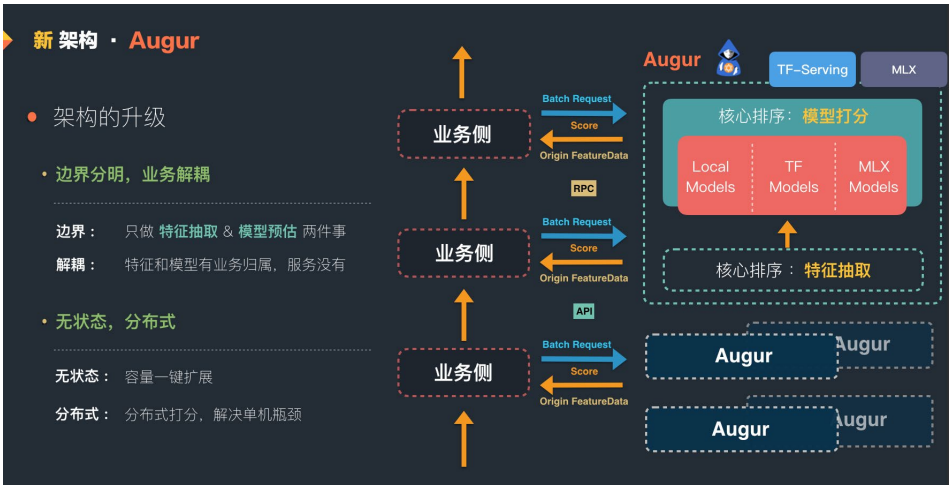
- 性能瓶颈：核心层的模型预估的 Size 扩展到数千级别文档的时候，单机已经难以承载；近百万个特征值的传输开销已经难以承受。
- 复用困难：模型预估能力已经成为一个通用的需求，单搜索就有几十个场景都需要该能力；而老逻辑的业务耦合性让复用变得更加困难。
- 平台缺失：快速的业务迭代下，需要有一个平台可以帮助业务快速地进行模型和特征的管理，包括但不限于配置、上线、灰度、验证等等。

3.2 新框架的边界

跟所有新系统的诞生故事一样，老系统一定会出现问题。原有架构在少特征以及小模型下虽有优势，但业务耦合，无法横向扩展，也难以复用。针对需求和老框架的种种问题，我们开始构建了新的高性能分布式模型预估框架 Augur，该框架指导思路是：

- **业务解耦，设定框架边界**：只做特征抽取和模型预估，对预估结果的处理等业务逻辑交给上层处理。

- **无状态，且可以做到分布式模型预估**，无压力支持数千级别文档数的深度模型预估。



架构上的改变，让 Augur 具备了复用的基础能力，同时也拥有了分布式预估的能力。可惜，系统架构设计中没有“银弹”：虽然系统具有了良好的弹性，但为此我们也付出了一些代价，我们会在文末进行解释。

4. 预估平台的构建过程

框架思路只能解决“能用”的问题，平台则是为了“通用”与“好用”。一个优秀的预估平台需要保证高性能，具备较为通用且接口丰富的核心预估框架，以及产品级别的业务管理系统。为了能够真正地提升预估能力和业务迭代的效率，平台需要回答以下几个问题：

- 如何解决特征和模型的高效迭代?
- 如何解决批量预估的性能和资源问题?
- 如何实现能力的快速复用并能够保障业务的安全?

下面，我们将逐一给出答案。

4.1 构建预估内核：高效的特征和模型迭代

4.1.1 Operator 和 Transformer

在搜索场景下，特征抽取较为难做的原因主要包括以下几点：

- 来源多：商户、商品、交易、用户等数十个维度的数据，还有交叉维度。由于美团业务众多，难以通过统一的特征存储去构建，交易相关数据只能通过服务来获取。
- 业务逻辑多：大多数据在不同的业务层会有复用，但是它们对特征的处理逻辑又有所不同。
- 模型差异：同一个特征，在不同的模型下，会有不同的处理逻辑。比如，一个连续型特征的分桶计算逻辑一样，但“桶”却因模型而各不相同；对于离散特征的低频过滤也是如此。
- 迭代快：特征的快速迭代，要求特征有快速在线上生效的能力，如果想要改动一个判断还需要写代码上线部署，无疑会拖慢了迭代的速度。模型如此，特征也是如此。

针对特征的处理逻辑，我们抽象出两个概念：

Operator：通用特征处理逻辑，根据功能的不同又可以分为两类：

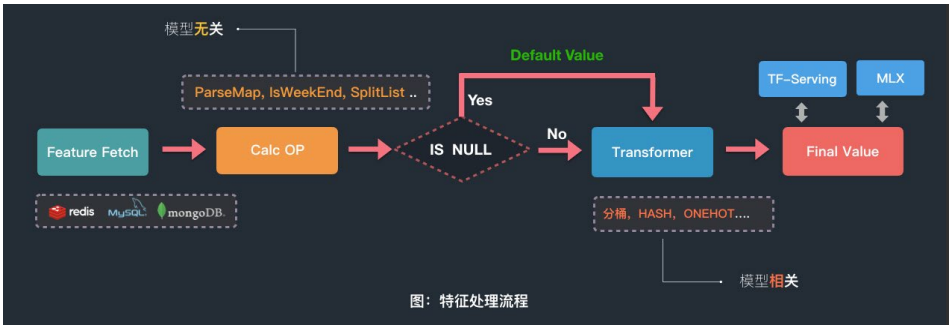
- IO OP：用处理原始特征的获取，如从 KV 里获取数据，或者从对应的第三方服务中获取数据。内置批量接口，可以实现批量召回，减少 RPC。
- Calc OP：用于处理对获取到的原始特征做与模型无关的逻辑处理，如拆分、判空、组合。业务可以结合需求实现特征处理逻辑。

通过 IO、计算分离，特征抽取执行阶段就可以进行 IO 异步、自动聚合 RPC、并行计算的编排优化，从而达到提升性能的目的。

Transformer：用于处理与模型相关的特征逻辑，如分桶、低频过滤等等。一个特征可以配置一个或者多个 Transformer。Transformer 也提供接口，业务方可以根据自己的需求定制逻辑。

离在线统一逻辑：Transformer 是特征处理的模型相关逻辑，因此我们将 Transformer 逻辑单独抽包，在我们样本生产的过程中使用，保证离线样本生产与线上特征处理逻辑的一致性。

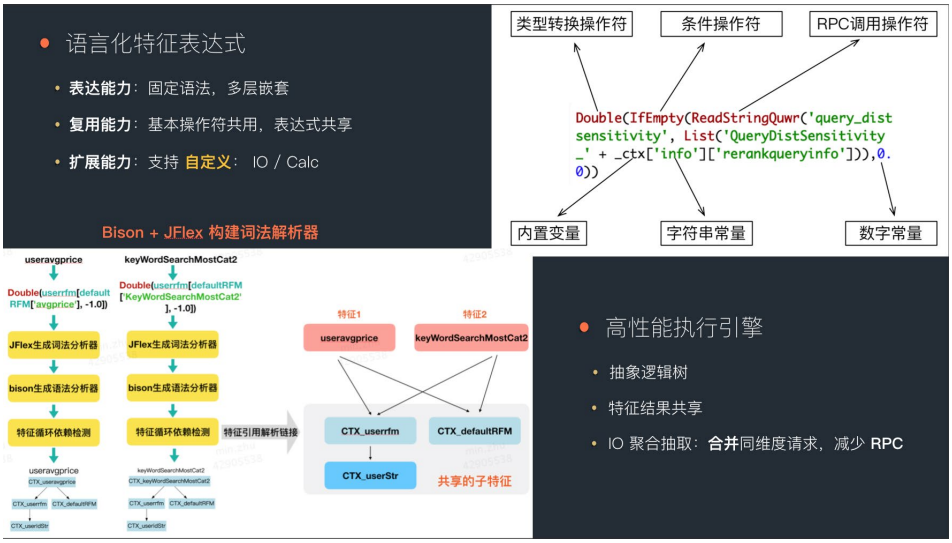
基于这两个概念，Augur 中特征的处理流程如下所示：首先，我们会进行特征抽取，抽取完后，会对特征做一些通用的处理逻辑；而后，我们会根据模型的需求进行二次变换，并将最终值输入到模型预估服务中。如下图所示：



图：特征处理流程

4.1.2 特征计算 DSL

有了 Operator 的概念，为了方便业务方进行高效的特征迭代，Augur 设计了一套弱类型、易读的特征表达式语言，将特征看成一系列 OP 与其他特征的组合，并基于 Bison&JFlex 构建了高性能语法和词法解析引擎。我们在解释执行阶段还做了一系列优化，包括并行计算、中间特征共享、异步 IO，以及自动 RPC 聚合等等。



举个例子：

```
// IO Feature: binaryBusinessTime;  ReadKV 是一个 IO 类型的 OP
ReadKV('mtptpoionlinefeatureexp','_id',_id,'ba_search.platform_poi_
business_hour_new.binarybusinesstime','STRING')
// FeatureA : CtxDateInfo;   ParseJSON 是一个 Calc 类型的 OP
ParseJSON(_ctx['dateInfo']);
// FeatureB : isTodayWeekend 需要看 Json 这种的日期是否是周末，便可以复用
CtxDateInfo  这个特征；IsWeekend 也是是一个 Calc 类型的 OP
IsWeekend(CtxDateInfo['date'])
```

在上面的例子中，ParseJSON 与 IsWeekend 都是 OP，CtxDateInfo 与 isTodayWeekend 都是由其他特征以及 OP 组合而成的特征。通过这种方式，业务方根据自己的需求编写 OP，可以快速复用已有的 OP 和特征，创造自己需要的新特征。而在真实的场景中，IO OP 的数量相对固定。所以经过一段时间的累计，OP 的数量会趋于稳定，新特征只需基于已有的 OP 和特征组合即可实现，非常的高效。

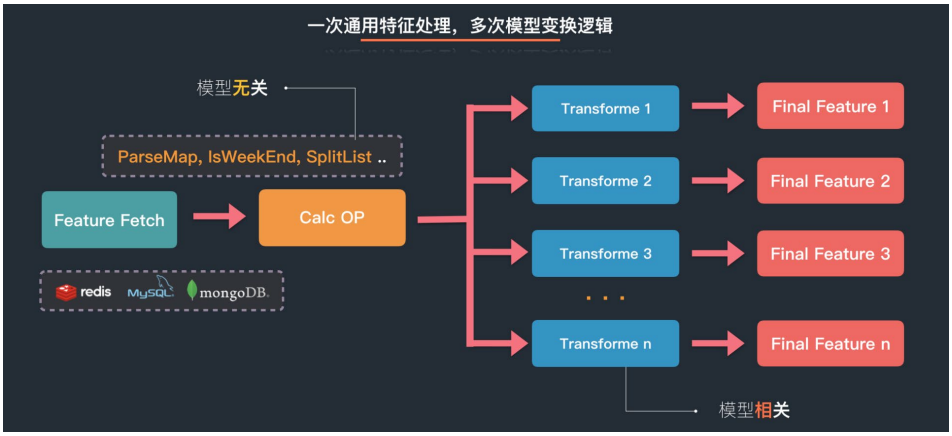
4.1.3 配置化的模型表达

特征可以用利用 OP、使用表达式的方式去表现，但特征还可能需要经过 Transformer 的变换。为此，我们同样为模型构建一套可解释的 JSON 表达模板，模型中每一个特征可以通过一个 JSON 对象进行配置，以一个输入到 TF 模型里的特征结构为例：

```
// 一个的特征的 JSON 配置
{
    "tf_input_config": {"otherconfig"},
    "tf_input_name": "modulestyle",
    "name": "moduleStyle",
    "transforms": [                          // Transfomers：模型相关的处理逻辑，
可以有多个，Augur 会按照顺序执行
      {
        "name": "BUCKETIZE",                 // Transfomer 的名称：这里是分桶
        "params": {
          "bins": [0,1,2,3,4]               // Transfomer 的参数
        }
      }
    ],
    "default_value": -1
}
```

通过以上配置，一个模型可以通过特征名和 Transformer 的组合清晰地表达。因此，模型与特征都只是一段纯文本配置，可以保存在外部，Augur 在需要的时候可以动态的加载，进而实现模型和特征的上线配置化，无需编写代码进行上线，安全且高效。

其中，我们将输入模型的特征名（tf_input_name）和原始特征名（name）做了区分。这样的话，就可以只在外部编写一次表达式，注册一个公用特征，却能通过在模型的结构体中配置不同 Transfomer 创造出多个不同的模型预估特征。这种做法相对节约资源，因为公用特征只需抽取计算一次即可。



此外，这一套配置文件也是离线样本生产时使用的特征配置文件，结合统一的 OP&Transformer 代码逻辑，进一步保证了离线 / 在线处理的一致性，也简化了上线的过程。因为只需要在离线状态下配置一次样本生成文件，即可在离线样本生产、在线模型预估两个场景通用。

4.2 完善预估系统：性能、接口与周边设施

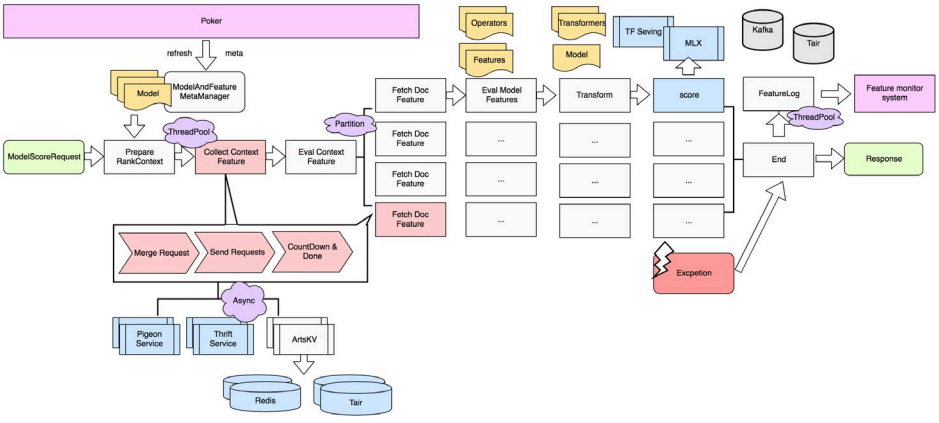
4.2.1 高效的模型预估过程

OP 和 Transformer 构建了框架处理特征的基本能力。实际开发中，为了实现高性能的预估能力，我们采用了分片纯异步的线程结构，层层 Call Back，最大程度将线程资源留给实际计算。因此，预估服务对机器的要求并不高。

为了描述清楚整个过程，这里需要明确特征的两种类型：

- ContextLevel Feature：全局维度特征，一次模型预估请求中，此类特征是通用的。比如时间、地理位置、距离、用户信息等等。这些信息只需计算一次。
- DocLevel Feature：文档维度特征，一次模型预估请求中每个文档的特征不同，需要分别计算。

一个典型的模型预估请求，如下图所示：



Augur 启动时会加载所有特征的表达式和模型，一个模型预估请求 ModelScore-

Request 会带来对应的模型名、要打分的文档 id（docid）以及一些必要的全局信息 Context。 Augur 在请求命中模型之后，将模型所用特征构建成一颗树，并区分 ContextLevel 特征和 DocLevel 特征。由于 DocLevel 特征会依赖 ContextLevel 特征，故先将 ContextLevel 特征计算完毕。对于 Doc 维度，由于对每一个 Doc 都要加载和计算对应的特征，所以在 Doc 加载阶段会对 Doc 列表进行分片，并发完成特征的加载，并且各分片在完成特征加载之后就进行打分阶段。也就是说，打分阶段本身也是分片并发进行的，各分片在最后打分完成后汇总数据，返回给调用方。 期间还会通过异步接口将特征日志上报，方便算法同学进一步迭代。

在这个过程中，为了使整个流程异步非阻塞，我们要求引用的服务提供异步接口。若部分服务未提供异步接口，可以将其包装成伪异步。这一套异步流程使得单机（16c16g）的服务容量提升超过 100%，提高了资源的利用率。

4.2.2 预估的性能及表达式的开销

框架的优势：得益于分布式，纯异步流程，以及在特征 OP 内部做的各类优化（公用特征 、RPC 聚合等），从老框架迁移到 Augur 后，上千份文档的深度模型预估性能提升了一倍。

至于大家关心的表达式解析对对于性能的影响其实可以忽略。因为这个模型预估的耗时瓶颈主要在于原始特征的抽取性能（也就是特征存储的性能）以及预估服务的性能（也就是 Serving 的性能）。而 Augur 提供了表达式解析的 Benchmark 测试用例，可以进行解析性能的验证。

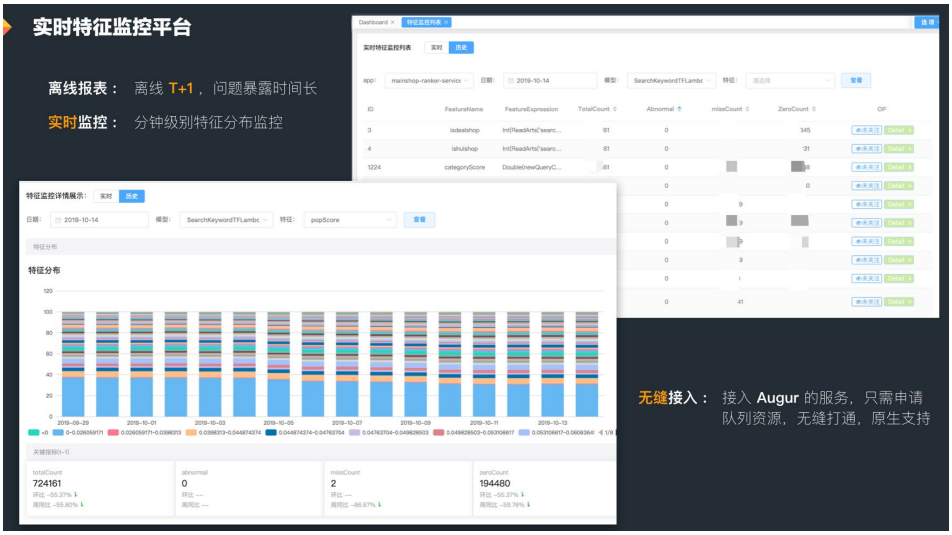
```
_I(_I('xxx'))
Benchmark              Mode  Cnt  Score   Error  Units
AbsBenchmarkTest.test  avgt   25  1.644 ± 0.009  ms/op
```

一个两层嵌套的表达式解析 10W 次的性能是 1.6ms 左右。相比于整个预估的时间，以及语言化表达式对于特征迭代效率的提升，这一耗时在当前业务场景下，基本可以忽略不计。

4.2.3 系统的其他组成部分

一个完善可靠的预估系统，除了“看得见”的高性能预估能力，还需要做好以下几个常被忽略的点：

- **几个常被忽略的点**：预估时产出的特征日志，需要通过框架上传到公司日志中心或者以用户希望的方式进行存储，方便模型的迭代。当然，必要的时候可以落入本地，方便问题的定位。
- **方便问题的定位**：系统监控不用多说，美团内部的 Cat& 天网，可以构建出完善的监控体系。另一方面，特征的监控也很重要，因为特征获取的稳定性决定了模型预估的质量，所以我们构建了实时的特征分布监控系统，可以分钟级发现特征分布的异常，最大限度上保证模型预估的可靠性。



- **丰富的接口**：除了预估接口，还需要有特征抽取接口、模型打分 Debug 接口、特征表达式测试接口、模型单独测试接口、特征模型刷新接口、特征依赖检等等一系列接口，这样才可以保证整个系统的可用性，并为后面管理平台的建设打下基础。

Augur 在完成了以上多种能力的建设之后，就可以当做一个功能相对完善且易扩展的在

线预估系统。由于我们在构建 Augur 的时候，设立了明确的边界，故以上能力是独立于业务的，可以方便地进行复用。当然，Augur 的功能管理，更多的业务接入，都需要管理平台的承载。于是，我们就构建了 Poker 平台，其中的在线预估管理模块是服务于 Augur，可以进行模型特征以及业务配置的高效管理。我们将在下一小节进行介绍。

4.3 建设预估平台：快速复用与高效管理

4.3.1 能力的快速复用

Augur 在设计之初，就将所有业务逻辑通过 OP 和 Transformer 承载，所以跟业务无关。考虑到美团搜索与 NLP 部模型预估场景需求的多样性，我们还为 Augur 赋予多种业务调用的方式。

- **种业务调用的方式。**：即基于 Augur 构建一个完整的 Service，可以实现无状态分布式的弹性预估能力。
- **布式的弹性预估能**：Java 服务化版本中内置了对 Thrift 的支持，使不同语言的业务都可以方便地拥有模型预估能力。
- **地拥有**：Augur 支持同一个服务同时提供 Pigeon（美团内部的 RPC 框架）以及 Thrift 服务，从而满足不同业务的不同需求。
- **不同业务的不同需**：Augur 同样支持以 SDK 的方式将能力嵌入到已有的集群当中。但如此一来，分布式能力就无法发挥了。所以，我们一般应用在性能要求高、模型比较小、特征基本可以存在本地的场景下。

其中服务化是被应用最多的方式，为了方便业务方的使用，除了完善的文档外，我们还构建了标准的服务模板，任何一个业务方基本上都可以在 30 分钟内构建出自己的 Augur 服务。服务模板内置了 60 多个常用逻辑和计算 OP，并提供了最佳实践文档与配置逻辑，使得业务方在没有指导的情况下可以自行解决 95% 以上的问题。整个流程如下图所示：

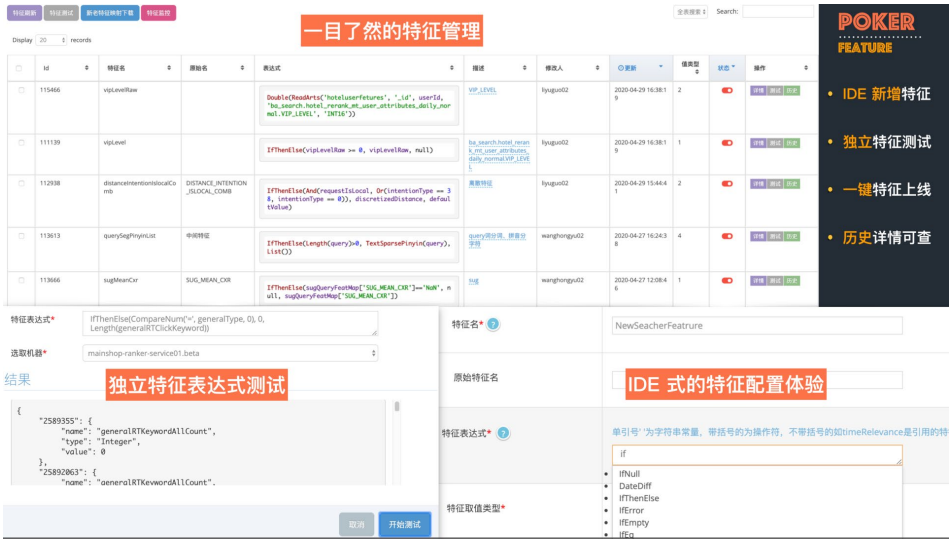
当然，无论使用哪一种方式去构建预估服务，都可以在美团内部的 Poker 平台上进行服务、模型与特征的管理。

4.3.2 Augur 管理平台 Poker 的构建

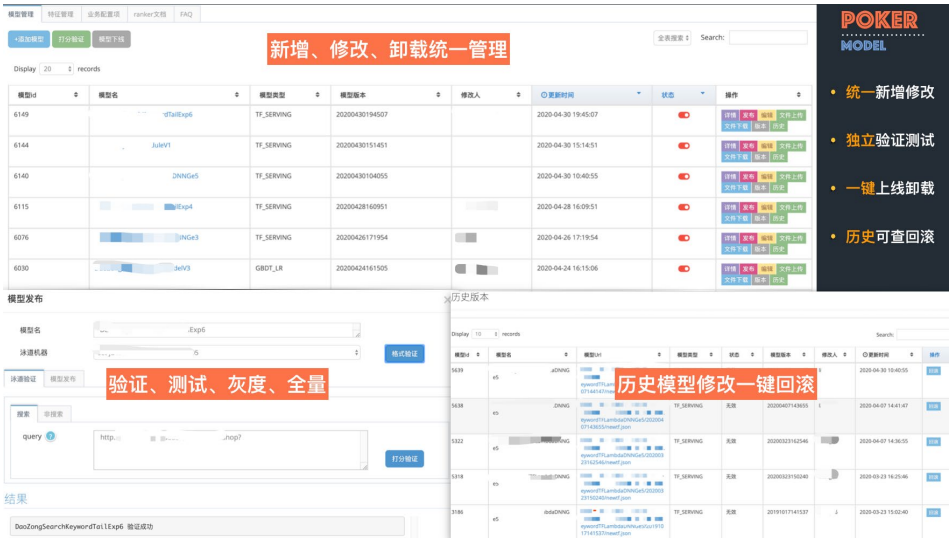
实现一个框架价值的最大化，需要一个完整的体系去支撑。而一个合格的在线预估平台，需要一个产品级别的管理平台辅助。于是我们构建了 Poker（搜索实验平台），其中的在线预估服务管理模块，也是 Augur 的最佳拍档。Augur 是一个可用性较高的在线预估框架，而 Poker+Augur 则构成了一个好用的在线预估平台。下图是在线预估服务管理平台的功能架构：

首先是预估核心特征的管理，上面说到我们构建了语言化的特征表达式，这其实是个较为常见的思路。Poker 利用 Augur 提供的丰富接口，结合算法的使用习惯，构建

了一套较为流畅的特征管理工具。可以在平台上完成新增、测试、上线、卸载、历史回滚等一系列操作。同时，还可以查询特征被服务中的哪些模型直接或者间接引用，在修改和操作时还有风险提示，兼顾了便捷性与安全性。



模型管理也是一样，我们在平台上实现了模型的配置化上线、卸载、上线前的验证、灰度、独立的打分测试、Debug 信息的返回等等。同时支持在平台上直接修改模型配置文件，平台可以实现模型多版本控制，一键回滚等。配置皆为实时生效，避免了手动上线遇到问题后因处理时间过长而导致损失的情况。

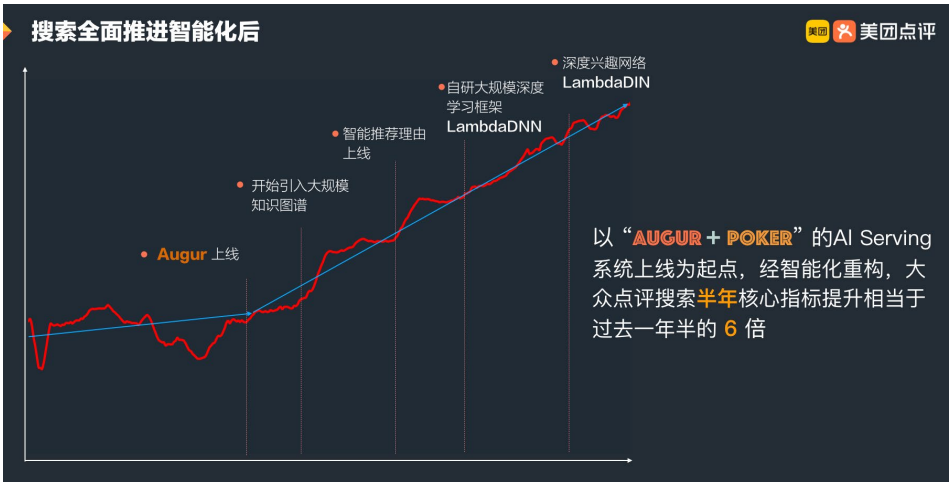


4.3.3 Poker + Augur 的应用与效果

随着 Augur 和 Poker 的成熟，美团搜索与 NLP 部门内部已经有超过 30 个业务方已经全面接入了预估平台，整体的概况如下图所示：

预估框架使用迁移 Augur 后，性能和模型预估稳定性上均获得了较大幅度的提升。更加重要的是，Poker 平台的在线预估服务管理和 Augur 预估框架，还将算法同学从繁复且危险的上线操作中解放出来，更加专注于算法迭代，从而取得更好的效果。以点评搜索为例，在 Poker+Augur 稳定上线之后，经过短短半年的时间，点评搜索核心 KPI 在高位基础上仍然实现了大幅提升，是过去一年半涨幅的六倍之多，提前半

年完成全年的目标。



4.4 进阶预估操作：模型也是特征

4.4.1 Model as a Feature，同构 or 异构?

在算法的迭代中，有时会将一个模型的预估的结果当做另外一个模型输入特征，进而取得更好的效果。如美团搜索与 NLP 中心的算法同学使用 BERT 来解决长尾请求商户的展示顺序问题，此时需要 BERT as a Feature。一般的做法是离线进行 BERT 批量计算，灌入特征存储供线上使用。但这种方式存在时效性较低（T+1）、覆盖度差等缺点。最好的方式自然是可以在线实时去做 BERT 模型预估，并将预估输出值作为特征，用于最终的模型打分。这就需要 Augur 提供 Model as a Feature 的能力。

得益于 Augur 抽象的流程框架，我们很快超额完成了任务。Model as a feature，虽然要对一个 Model 做预估操作，但从更上层的模型角度看，它就是一个特征。既然是特征，模型预估也就是一个计算 OP 而已。 所以我们只需要在内部实现一个特殊的 OP，ModelFeatureOpreator 就可以干净地解决这些问题了。

我们在充分调研后，发现 Model as a Feature 有两个维度的需求：同构的特征和异构的特征。同构指的是这个模型特征与模型的其他特征一样，是与要预估的文档统

一维度的特征，那这个模型就可以配置在同一个服务下，也就是本机可以加载这个 Stacking 模型；而异构指的是 Model Feature 与当前预估的文档不是统一维度的，比如商户下挂的商品，商户打分需要用到商品打分的结果，这两个模型非统一维度，属于两个业务。正常逻辑下需要串行处理，但是 Augur 可以做得更高效。为此我们设计了两个 OP 来解决问题：

- **LocalModelFeature**：解决同构 Model Feature 的需求，用户只需像配置普通特征表达式一样即可实现在线的 Model Stacking；当然，内部自然有优化逻辑，比如外部模型和特征模型所需的特征做统一整合，尽可能的减少资源消耗，提升性能。 该特征所配置的模型特征，将在本机执行，以减少 RPC。
- **RemoteModelFeature**：解决异构 Model Feature 的需求，用户还是只需配置一个表达式，但是此表达式会去调用相应维度的 Augur 服务，获取相应的模型和特征数据供主维度的 Augur 服务处理。虽然多了一层 RPC，但是相对于纯线性的处理流程，分片异步后，还是有不少的性能提升。

美团搜索内部，已经通过 LocalModelFeature 的方式，实现了 BERT as a Feature。在几乎没有新的使用学习成本的前提下，同时在线上取得了明显的指标提升。

4.4.2 Online Model Ensemble

Augur 支持有单独抽取特征的接口，结合 Model as a Feature，若需要同时为一个文档进行两个或者多个模型的打分，再将分数做加权后使用，非常方便地实现离线 Ensemble 出来模型的实时在线预估。我们可以配置一个简单的 LR、Empty 类型模型（仅用于特征抽取），或者其他任何 Augur 支持的模型，再通过 LocalModelFeature 配置若干的 Model Feature，就可以通过特征抽取接口得到一个文档多个模型的线性加权分数了。而这一切都被包含在一个统一的抽象逻辑中，使用户的体验是连续统一的，几乎没有增加学习成本。

除了上面的操作外，Augur 还提供了打分的同时带回部分特征的接口，供后续的业务规则处理使用。

5. 更多思考

当然，肯定没有完美的框架和平台。Augur 和 Poker 还有很大的进步空间，也有一些不可回避的问题。主要包括以下几个方面。

被迫“消失”的 Listwise 特征

前面说到，系统架构设计中没有“银弹”。在采用了无状态分布式的设计后，请求会分片。所以 ListWise 类型的特征就必须在打分前算好，再通过接口传递给 Augur 使用。在权衡性能和效果之后，算法同学放弃了这一类型的特征。

当然，不是说 Augur 不能实现，只是成本有些高，所以暂时 Hold 。我们也有设计过方案，在可量化的收益高于成本的时候，我们会在 Augur 中开放协作的接口。

单机多层打分的缺失

Augur 一次可以进行多个模型的打分，模型相互依赖（下一层模型用到上一层模型的结果）也可以通过 Stacking 技术来解决。但如果模型相互依赖又逐层减少预估文档（比如，第一轮预估 1000 个，第二轮预估 500），则只能通过多次 RPC 的方式去解决问题，这是一个现实问题的权衡。分片打分的性能提升，能否 Cover 多次 RPC 的开销？在实际开发中，为了保持框架的清晰简单，我们选择了放弃多层打分的特性。

离线能力缺失?

Poker 是搜索实验平台的名字。我们设计它的初衷，是解决搜索模型实验中，从离线到在线所有繁复的手工操作，使搜索拥有一键训练、一键 Fork、一键上线的能力。与公司其他的训练平台不同，我们通过完善的在线预估框架倒推离线训练的需求，进而构建了与在线无缝结合的搜索实验平台，极大地提升了算法同学的工作效。

未来，我们也会向大家介绍产品级别的一站式搜索实验平台，敬请期待。

6. 未来展望

在统一了搜索的在线预估框架后，我们会进一步对 Augur 的性能 & 能力进行扩展。

未来，我们将会在检索粗排以及性能要求更高的预估场景中去发挥它的能力与价值。同时 ，我们正在将在线预估框架进一步融合到我们的搜索实验平台 Poker 中，与离线训练和 AB 实验平台做了深度的打通，为业务构建高效完整的模型实验基础设施。

如果你想近距离感受一下 Augur 的魅力，欢迎加入美团技术团队！

7. 作者简介

朱敏，紫顺，乐钦，洪晨，乔宇，武进，孝峰，俊浩等，均来自美团搜索与 NLP 部。

Transformer 在美团搜索排序中的实践

作者：肖垚

引言

美团搜索是美团 App 连接用户与商家的一种重要方式，而排序策略则是搜索链路的关键环节，对搜索展示效果起着至关重要的效果。目前，美团的搜索排序流程为多层排序，分别是粗排、精排、异构排序等，多层排序的流程主要是为了平衡效果和性能。其中搜索核心精排策略是 DNN 模型，我们始终贴近业务，并且结合先进技术，从特征、模型结构、优化目标角度对排序效果进行了全面的优化。

近些年，基于 Transformer[1] 的一些 NLP 模型大放光彩，比如 BERT[2] 等等，将 Transformer 结构应用于搜索推荐系统也成为业界的一个潮流。比如应用于对 CTR 预估模型进行特征组合的 AutoInt[3]、行为序列建模的 BST[4] 以及重排序模型 PRM[5]，这些工作都证明了 Transformer 引入搜索推荐领域能取得不错的效果，所以美团搜索核心排序也在 Transformer 上进行了相关的探索。

本文旨在分享 Transformer 在美团搜索排序上的实践经验。内容会分为以下三个部分：第一部分对 Transformer 进行简单介绍，第二部分会介绍 Transfomer 在美团搜索排序上的应用以及实践经验，最后一部分是总结与展望。希望能对大家有所帮助和启发。

Transformer 简介

Transformer 是谷歌在论文《Attention is all you need》[1] 中提出来解决 Sequence to Sequence 问题的模型，其本质上是一个编解码（Encoder-Decoder ）结构，编码器 Encoder 由 6 个编码 block 组成，Encoder 中的每个 block 包含 Multi-Head Attention 和 FFN（Feed-Forward Network）；同样解码器 Decoder 也是 6 个解码 block 组成，每个 block 包含 Multi-Head Attention、Encoder-Decoder Attention

和 FFN。具体结构如图 1 所示，其详细的介绍可参考文献 [1,6]。

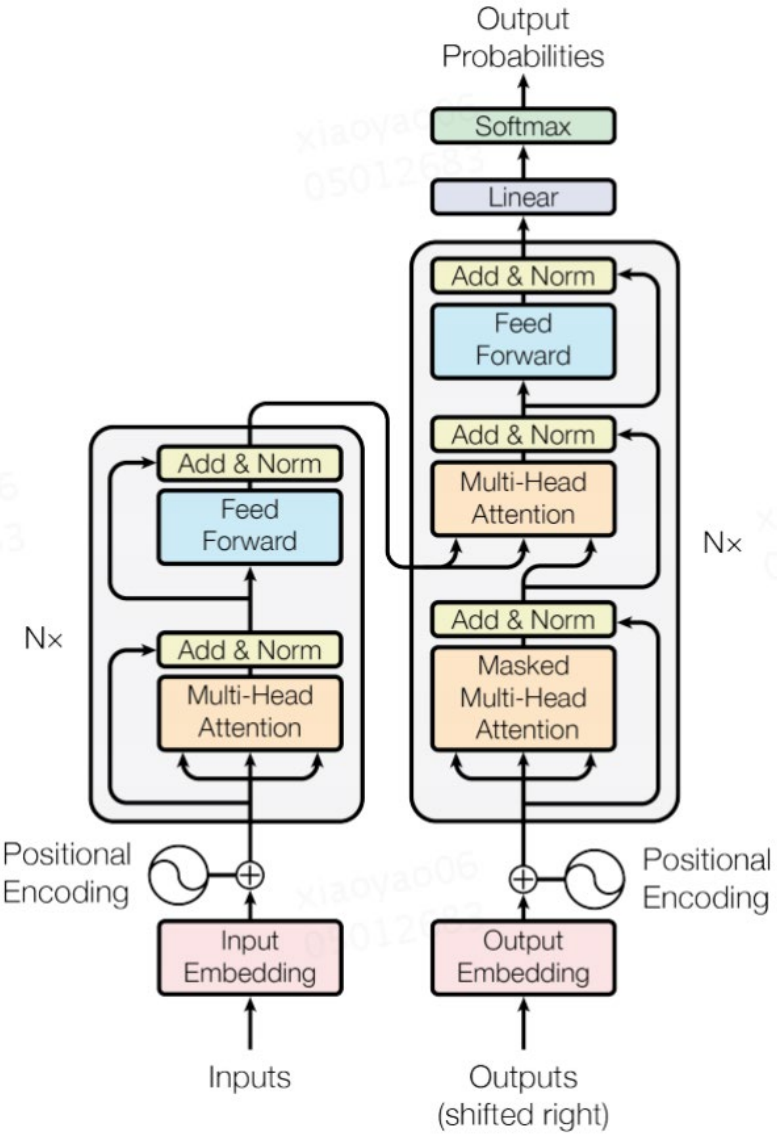


图 1　Transformer 结构示意图

考虑到后续内容出现的 Transformer Layer 就是 Transformer 的编码层，这里先对它做简单的介绍。它主要由以下两部分组成：

Multi-Head Attention

Multi-Head Attention 实际上是 h 个 Self-Attention 的集成，h 代表头的个数。其中 Self-Attention 的计算公式如下：

$$Attention\left(\mathbf{K},\mathbf{Q},\mathbf{V}\right)=softmax\left(\frac{\mathbf{Q}\mathbf{K}^{T}}{\sqrt{d}}\right)\mathbf{V}$$

其中，Q 代表查询，K 代表键，V 代表数值。

在我们的应用实践中，原始输入是一系列 Embedding 向量构成的矩阵 E ，矩阵 E 首先通过线性投影：

$$\mathbf{W}^{Q},\mathbf{W}^{K},\mathbf{W}^{V}\in R^{d\times d}$$

得到三个矩阵：

$$\mathbf{E}\mathbf{W}^{Q}\ \ \mathbf{E}\mathbf{W}^{K}\ \ \mathbf{E}\mathbf{W}^{V}$$

然后将投影后的矩阵输入到 Multi-Head Attention。计算公式如下：

$$head_{i}=Attention\left(\mathbf{E}\mathbf{W}_{i}^{Q},\mathbf{E}\mathbf{W}_{i}^{K},\mathbf{E}\mathbf{W}_{i}^{V}\right)$$

$$\mathbf{S}=MH\left(\mathbf{E}\right)=Concat\left(head_{1},head_{2},...,head_{h}\right)\mathbf{W}^{H}$$

Point-wise Feed-Forward Networks

该模块是为了提高模型的非线性能力提出来的，它就是全连接神经网络结构，计算公式如下：

$$\mathbf{S}'=LayerNorm(\mathbf{E}+Dropout(MH(\mathbf{E})))$$

$$\mathbf{F}=LayerNorm(\mathbf{S}'+Dropout(Relu(\mathbf{S}'\mathbf{W}^{(1)}+b^{(1)})\mathbf{W}^{(2)}+b^{(2)}))$$

Transformer Layer 就是通过这种自注意力机制层和普通非线性层来实现对输入信号的编码，得到信号的表示。

美团搜索排序 Transformer 实践经验

Transformer 在美团搜索排序上的实践主要分以下三个部分：第一部分是特征工程，第二部分是行为序列建模，第三部分是重排序。下面会逐一进行详细介绍。

特征工程

在搜索排序系统中，特征工程的输入特征维度高但稀疏性很强，而准确的交叉特征对模型的效果又至关重要。所以寻找一种高效的特征提取方式就变得十分重要，我们借鉴 AutoInt[3] 的方法，采用 Transformer Layer 进行特征的高阶组合。

模型结构

我们的模型结构参考 AutoInt[3] 结构，但在实践中，根据美团搜索的数据特点，我们对模型结构做了一些调整，如下图 2 所示：

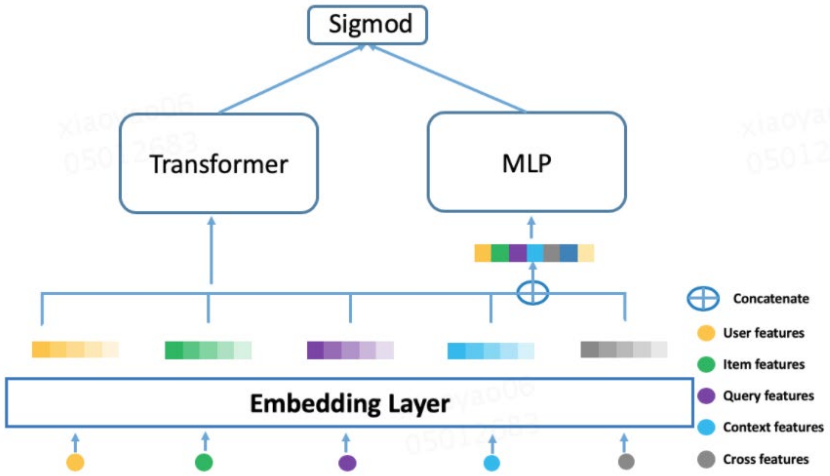


图 2　Transformer&Deep 结构示意图

相比 AutoInt[3]，该结构有以下不同：

- 保留将稠密特征和离散特征的 Embedding 送入到 MLP 网络，以隐式的方式学习其非线性表达。
- Transformer Layer 部分，不是送入所有特征的 Embedding，而是基于人工经验选择了部分特征的 Embedding，第一点是因为美团搜索场景特征的维度高，全输入进去会提高模型的复杂度，导致训练和预测都很慢；第二点是，所有特征的 Embedding 维度不完全相同，也不适合一起输入到 Transformer Layer 。

Embedding Layer 部分：众所周知在 CTR 预估中，除了大规模稀疏 ID 特征，稠密类型的统计特征也是非常有用的特征，所以这部分将所有的稠密特征和稀疏 ID 特征都转换成 Embedding 表示。

Transformer 部分：针对用户行为序列、商户 、品类 、地理位置等 Embedding 表示，使用 Transformer Layer 来显示学习这些特征的交叉关系。

MLP 部分：考虑到 MLP 具有很强的隐式交叉能力，将所有特征的 Embedding 表示 concat 一起输入到 MLP。

实践效果及经验

效果：离线效果提升，线上 QV_CTR 效果波动。

经验：

- 三层 Transformer 编码层效果比较好。
- 调节多头注意力的“头”数对效果影响不大 。
- Transformer 编码层输出的 Embedding 大小对结果影响不大。
- Transformer 和 MLP 融合的时候，最后结果融合和先 concat 再接一个全连接层效果差不多。

行为序列建模

理解用户是搜索排序中一个非常重要的问题。过去，我们对训练数据研究发现，在训练数据量很大的情况下，item 的大部分信息都可以被 ID 的 Embedding 向量进行表示，但是用户 ID 在训练数据中是十分稀疏的，用户 ID 很容易导致模型过拟合，所以需要大量的泛化特征来较好的表达用户。这些泛化特征可以分为两类：一类是偏静态的特征，例如用户的基本属性（年龄、性别、职业等等）特征、长期偏好（品类、价格等等）特征；另一类是动态变化的特征，例如刻画用户兴趣的实时行为序列特征。而用户实时行为特征能够明显加强不同样本之间的区分度，所以在模型中优化用户行为序列建模是让模型更好理解用户的关键环节。

目前，主流方法是采用对用户行为序列中的 item 进行 Sum-pooling 或者 Mean-pooling 后的结果来表达用户的兴趣，这种假设所有行为内的 item 对用户的兴趣都是等价的，因而会引入一些噪声。尤其是在美团搜索这种交互场景，这种假设往往是不能很好地进行建模来表达用户兴趣。

近年来，在搜索推荐算法领域，针对用户行为序列建模取得了重要的进展：DIN 引入注意力机制，考虑行为序列中不同 item 对当前预测 item 有不同的影响 [7]；而 DIEN 的提出，解决 DIN 无法捕捉用户兴趣动态变化的缺点 [8]。DSIN 针对 DIN 和 DIEN 没有考虑用户历史行为中的 Session 信息，因为每个 Session 中的行为是相近的，而在不同 Session 之间的差别很大，它在 Session 层面上对用户的行为序列进行建模 [9]；BST 模型通过 Transformer 模型来捕捉用户历史行为序列中的各个 item 的关联特征，与此同时，加入待预测的 item 来达到抽取行为序列中的商品与待推荐商品之间的相关性 [4]。这些已经发表过的工作都具有很大的价值。接下来，我们主要从美团搜索的实践业务角度出发，来介绍 Transformer 在用户行为序列建模上的实践。

模型结构

在 Transformer 行为序列建模中，我们迭代了三个版本的模型结构，下面会依次进行介绍。

模型主要构成：所有特征（user 维度、item 维度、query 维度、上下文维度、交叉维度）经过底层 Embedding Layer 得到对应的 Embedding 表示；建模用户行为序列得到用户的 Embedding 表示；所有 Embedding concat 一起送入到三层的 MLP 网络。

第一个版本：因为到原来的 Sum-pooling 建模方式没有考虑行为序列内部各行为的关系，而 Transformer 又被证明能够很好地建模序列内部之间的关系，所以我们尝试直接将行为序列输入到 Transformer Layer，其模型结构如图 3 所示：

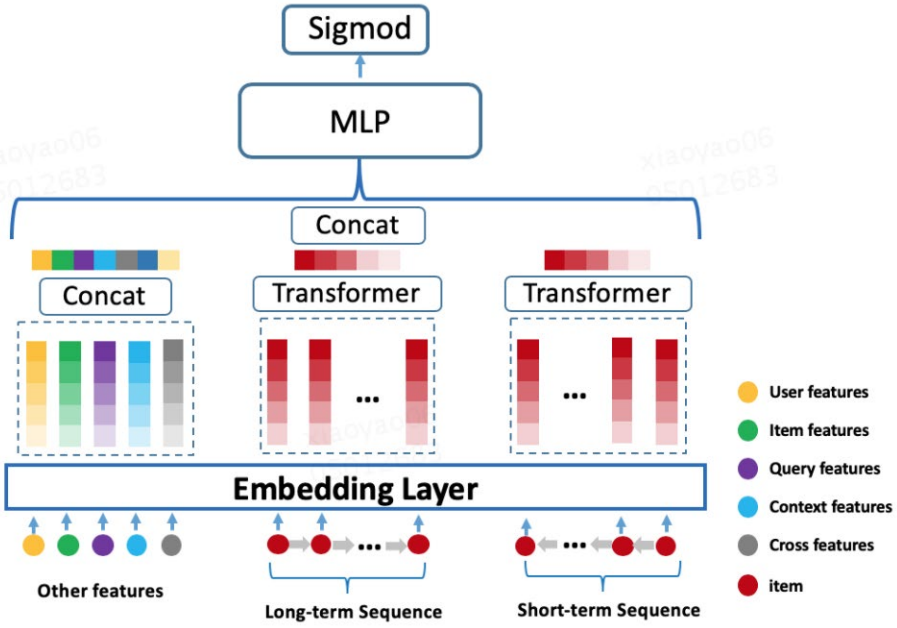


图 3　Transformer 行为序列建模

行为序列建模

输入部分：

- 分为短期行为序列和长期行为序列。
- 行为序列内部的每个行为原始表示是由商户 ID，以及一些商户泛化信息的 Embedding 进行 concat 组成。
- 每段行为序列的长度固定，不足部分使用零向量进行补齐。

输出部分：对 Transformer Layer 输出的向量做 Sum-pooling（这里尝试过 Mean-pooling、concat，效果差不多）得到行为序列的最终 Embedding 表示。

该版本的离线指标相比线上 Base（行为序列 Sum-pooling）模型持平，尽管该版本没有取得离线提升，但是我们继续尝试优化。

第二个版本：第一个版本存在一个问题，对所有的 item 打分的时候，用户的 Embedding 表示都是一样的，所以参考 BST[4]，在第一个版本的基础上引入 Target-item，这样可以学习行为序列内部的 item 与 Target-item 的相关性，这样在对不同的 item 打分时，用户的 Embedding 表示是不一样的，其模型结构如下图 4 所示：

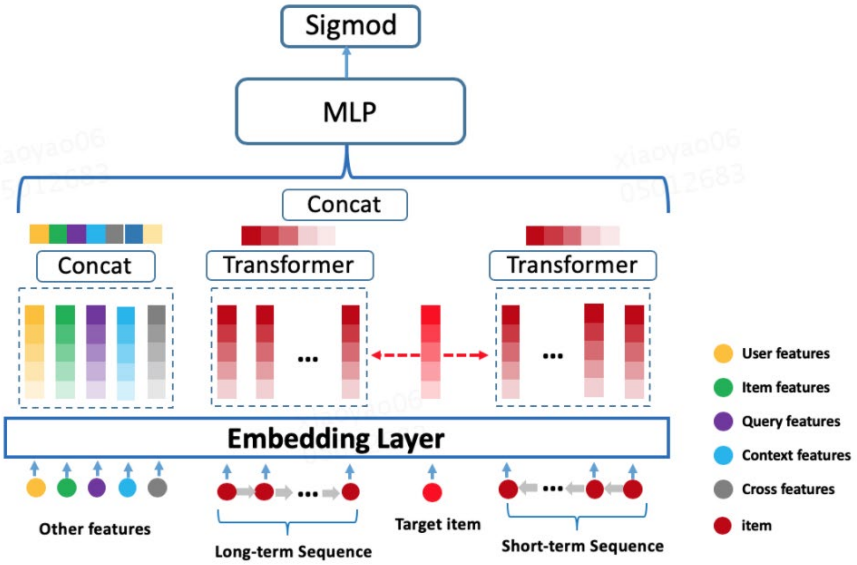


图 4　Transformer 行为序列建模

该版本的离线指标相比线上 Base（行为序列 Sum-pooling）模型提升，上线发现效果波动，我们仍然没有灰心，继续迭代优化。

第三个版本：和第二个版本一样，同样针对第一个版本存在的对不同 item 打分，用户 Embedding 表示一样的问题，尝试在第一个版本引入 Transformer 的基础上，叠加 DIN[7] 模型里面的 Attention-pooling 机制来解决该问题，其模型结构如图 5 所示：

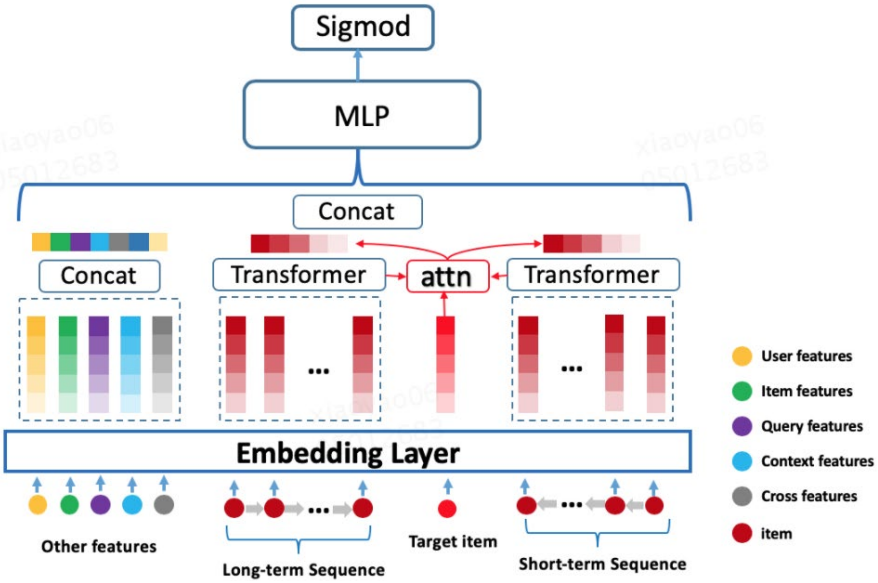


图 5　Transformer 行为序列建模

该版本的离线指标相比第二个版本模型有提升，上线效果相比线上 Base（行为序列 Sum-pooling）有稳定提升。

实践效果及经验

效果：第三个版本（Transformer + Attention-pooling）模型的线上 QV_CTR 和 NDCG 提升最为显著。

经验：

- Transformer 编码为什么有效？ Transformer 编码层内部的自注意力机制，能够对序列内 item 的相互关系进行有效的建模来实现更好的表达，并且我们离线实验不加 Transformer 编码层的 Attention-pooling，发现离线 NDCG 下降，从实验上证明了 Transformer 编码有效。
- Transformer 编码为什么优于 GRU？忽略 GRU 的性能差于 Transformer；我们做过实验将行为序列长度的上限往下调，Transformer 的效果相比 GRU 的效果提升在缩小，但是整体还是行为序列的长度越大越好，所以

Transformer 相比 GRU 在长距离时，特征捕获能力更强。

- 位置编码（Pos-Encoding）的影响我们试过加 Transformer 里面原生的正余弦以及距当前预测时间的时间间隔的位置编码都无效果，分析应该是我们在处理行为序列的时候，已经将序列切割成不同时间段，一定程度上包含了时序位置信息。为了验证这个想法，我们做了仅使用一个长序列的实验（对照组不加位置编码，实验组加位置编码，离线 NDCG 有提升），这验证了我们的猜测。
- Transformer 编码层不需要太多，层数过多导致模型过于复杂，模型收敛慢效果不好。
- 调节多头注意力的“头”数对效果影响不大。

重排序

在引言中，我们提到美团搜索排序过去做了很多优化工作，但是大部分都是集中在 PointWise 的排序策略上，未能充分利用商户展示列表的上下文信息来优化排序。一种直接利用上下文信息优化排序的方法是对精排的结果进行重排，这可以抽象建模成一个序列（排序序列）生成另一个序列（重排序列）的过程，自然联想到可以使用 NLP 领域常用的 Sequence to Sequence 建模方法进行重排序建模。

目前业界已有一些重排序的工作，比如使用 RNN 重排序 [10-11]、Transformer 重排序 [5]。考虑到 Transformer 相比 RNN 有以下两个优势：（1）两个 item 的相关性计算不受距离的影响（2）Transformer 可以并行计算，处理效率比 RNN 更高；所以我们选择 Transformer 对重排序进行建模。

模型结构

模型结构参考了 PRM[5]，结合美团搜索实践的情况，重排序模型相比 PRM 做了一些调整。具体结构如图 6 所示，其中 D1，D2，…，Dn 是重排商户集合，最后根据模型的输出 Score(D1)，Score(D2)，…，Score(Dn) 按照从大到小进行排序。

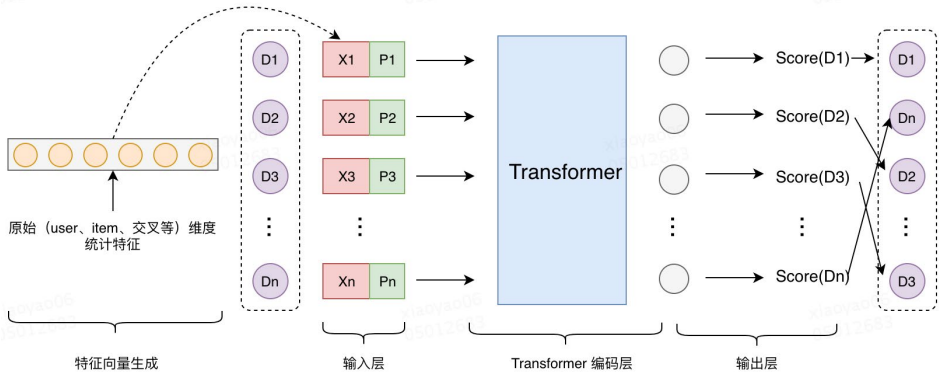


图 6　Transformer 重排序

主要由以下几个部分构成：

- **特征向量生成**：由原始特征（user、item、交叉等维度的稠密统计特征）经过一层全连接的输出进行表示。
- **输入层**：其中 X 表示商户的特征向量，P 表示商户的位置编码，将特征向量 X 与位置向量 P 进行 concat 作为最终输入。
- **Transformer 编码层**：一层 Multi-Head Attention 和 FFN 的。
- **输出层**：一层全连接网络得到打分输出 Score。

模型细节：

- 特征向量生成部分和重排序模型是一个整体，联合端到端训练。
- 训练和预测阶段固定选择 TopK 进行重排，遇到某些请求曝光 item 集不够 TopK 的情况下，在末尾补零向量进行对齐。

实践效果及经验

效果：Transformer 重排序对线上 NDCG 和 QV_CTR 均稳定正向提升。

经验：

- 重排序大小如何选择？考虑到线上性能问题，重排序的候选集不能过大，我们

分析数据发现 95% 的用户浏览深度不超过 10，所以我们选择对 Top10 的商户进行重排。

- 位置编码向量的重要性：这个在重排序中很重要，需要位置编码向量来刻画位置，更好的让模型学习出上下文信息，离线实验发现去掉位置向量 NDCG@10 下降明显。
- 性能优化：最初选择商户全部的精排特征作为输入，发现线上预测时间太慢；后面进行特征重要性评估，筛选出部分重要特征作为输入，使得线上预测性能满足上线要求。
- 调节多头注意力的“头”数对效果影响不大。

总结和展望

2019 年底，美团搜索对 Transformer 在排序中的应用进行了一些探索，既取得了一些技术沉淀也在线上指标上取得比较明显的收益，不过未来还有很多的技术可以探索。

- 在特征工程上，引入 Transformer 层进行高阶特征组合虽然没有带来收益，但是在这个过程中也再次验证了没有万能的模型对所有场景数据有效。目前搜索团队也在探索在特征层面应用 BERT 对精排模型进行优化。
- 在行为序列建模上，目前的工作集中在对已有的用户行为数据进行建模来理解用户，未来要想更加深入全面的认识用户，更加丰富的用户数据必不可少。当有了这些数据后如何进行利用，又是一个可以探索的技术点，比如图神经网络建模等等。
- 在重排序建模上，目前引入 Transformer 取得了一些效果，同时随着强化学习的普及，在美团这种用户与系统强交互的场景下，用户的行为反馈蕴含着很大的研究价值，未来利用用户的实时反馈信息进行调序是个值得探索的方向。例如，根据用户上一刻的浏览反馈，对用户下一刻的展示结果进行调序。

除了上面提到的三点，考虑到美团搜索上承载着多个业务，比如美食、到综、酒店、旅游等等，各个业务之间既有共性也有自己独有的特性，并且除了优化用户体验，也需要满足业务需求。为了更好的对这一块建模优化，我们也正在探索 Partition Model

和多目标相关的工作，欢迎业界同行一起交流。

参考资料

[1] Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need[C]//Advances in neural information processing systems. 2017: 5998-6008.

[2] Devlin J, Chang M W, Lee K, et al. Bert: Pre-training of deep bidirectional transformers for language understanding[J]. arXiv preprint arXiv:1810.04805, 2018.

[3] Song W, Shi C, Xiao Z, et al. Autoint: Automatic feature interaction learning via self-attentive neural networks[C]//Proceedings of the 28th ACM International Conference on Information and Knowledge Management. 2019: 1161-1170.

[4] Chen Q, Zhao H, Li W, et al. Behavior sequence transformer for e-commerce recommendation in Alibaba[C]//Proceedings of the 1st International Workshop on Deep Learning Practice for High-Dimensional Sparse Data. 2019: 1-4.

[5] Pei C, Zhang Y, Zhang Y, et al. Personalized re-ranking for recommendation[C]// Proceedings of the 13th ACM Conference on Recommender Systems. 2019: 3-11.

[6] http://jalammar.github.io/illustrated-transformer/

[7] Zhou G, Zhu X, Song C, et al. Deep interest network for click-through rate prediction[C]//Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. ACM, 2018: 1059-1068.

[8] Zhou G, Mou N, Fan Y, et al. Deep interest evolution network for click-through rate prediction[C]//Proceedings of the AAAI Conference on Artificial Intelligence. 2019, 33: 5941-5948.

[9] Feng Y, Lv F, Shen W, et al. Deep Session Interest Network for Click-Through Rate Prediction[J]. arXiv preprint arXiv:1905.06482, 2019.

[10] Zhuang T, Ou W, Wang Z. Globally optimized mutual influence aware ranking in e-commerce search[J]. arXiv preprint arXiv:1805.08524, 2018.

[11] Ai Q, Bi K, Guo J, et al. Learning a deep listwise context model for ranking refinement[C]//The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval. 2018: 135-144.

作者简介

肖垚，家琪，周翔，陈胜，云森，永超，仲远，均来自美团 AI 平台搜索与 NLP 部。

招聘信息

美团搜索核心排序组，长期招聘搜索推荐算法工程师，坐标北京。欢迎感兴趣的同学发送简历到：tech@meituan.com（邮件标题请注明：美团搜索核心排序组）

BERT 在美团搜索核心排序的探索和实践

作者：李勇　佳昊　杨扬　金刚　周翔　朱敏等

为进一步优化美团搜索排序结果的深度语义相关性，提升用户体验，搜索与 NLP 部算法团队从 2019 年底开始基于 BERT 优化美团搜索排序相关性，经过三个月的算法迭代优化，离线和线上效果均取得一定进展。本文主要介绍探索过程以及实践经验。

引言

美团搜索是美团 App 上最大的连接人和服务的入口，覆盖了团购、外卖、电影、酒店、买菜等各种生活服务。随着用户量快速增长，越来越多的用户在不同场景下都会通过搜索来获取自己想要的服务。理解用户 Query，将用户最想要的结果排在靠前的位置，是搜索引擎最核心的两大步骤。但是，用户输入的 Query 多种多样，既有商户名称和服务品类的 Query，也有商户别名和地址等长尾的 Query，准确刻画 Query 与 Doc 之间的深度语义相关性至关重要。基于 Term 匹配的传统相关性特征可以较好地判断 Query 和候选 Doc 的字面相关性，但在字面相差较大时，则难以刻画出两者的相关性，比如 Query 和 Doc 分别为“英语辅导”和“新东方”时两者的语义是相关的，使用传统方法得到的 Query-Doc 相关性却不一致。

2018 年底，以 Google BERT[1] 为代表的预训练语言模型刷新了多项 NLP 任务的最好水平，开创了 NLP 研究的新范式：即先基于大量无监督语料进行语言模型预训练（Pre-training），再使用少量标注语料进行微调（Fine-tuning）来完成下游的 NLP 任务（文本分类、序列标注、句间关系判断和机器阅读理解等）。美团 AI 平台搜索与 NLP 部算法团队基于美团海量业务语料训练了 MT-BERT 模型，已经将 MT-BERT 应用到搜索意图识别、细粒度情感分析、点评推荐理由、场景化分类等业务场景中 [2]。

作为 BERT 的核心组成结构，Transformer 具有强大的文本特征提取能力，早在多项 NLP 任务中得到了验证，美团搜索也基于 Transformer 升级了核心排序模型，取

得了不错的研究成果[3]。为进一步优化美团搜索排序结果的深度语义相关性，提升用户体验，搜索与 NLP 部算法团队从 2019 年底开始基于 BERT 优化美团搜索排序相关性，经过三个月的算法迭代优化，离线和线上效果均取得一定进展，本文主要介绍 BERT 在优化美团搜索核心排序上的探索过程以及实践经验。

BERT 简介

近年来，以 BERT 为代表的预训练语言模型在多项 NLP 任务上都获得了不错的效果。下图 1 简要回顾了预训练语言模型的发展历程。2013 年，Google 提出的 Word2vec[4] 通过神经网络预训练方式来生成词向量（Word Embedding），极大地推动了深度自然语言处理的发展。针对 Word2vec 生成的固定词向量无法解决多义词的问题，2018 年，Allen AI 团队提出基于双向 LSTM 网络的 ELMo[5]。ELMo 根据上下文语义来生成动态词向量，很好地解决了多义词的问题。2017 年底，Google 提出了基于自注意力机制的 Transformer[6] 模型。

相比 RNN 模型，Transformer 语义特征提取能力更强，具备长距离特征捕获能力，且可以并行训练，在机器翻译等 NLP 任务上效果显著。Open AI 团队的 GPT[7] 使用 Transformer 替换 RNN 进行深层单向语言模型预训练，并通过在下游任务上 Fine-tuning 验证了 Pretrain-Finetune 范式的有效性。在此基础上，Google BERT 引入了 MLM（Masked Language Model）及 NSP（Next Sentence Prediction，NSP）两个预训练任务，并在更大规模语料上进行预训练，在 11 项自然语言理解任务上刷新了最好指标。BERT 的成功启发了大量后续工作，总结如下：

- 融合更多外部知识的百度 ERNIE[8]，清华 ERNIE[9] 和 K-BERT[10] 等；
- 优化预训练目标的 ERNIE 2.0[11]，RoBERTa[12]，SpanBERT[13]，Struct-BERT[14] 等；
- 优化模型结构或者训练方式的 ALBERT[15] 和 ELECTRA[16]。关于预训练模型的各种后续工作，可以参考复旦大学邱锡鹏老师最近的综述[17]，本文不再赘述。

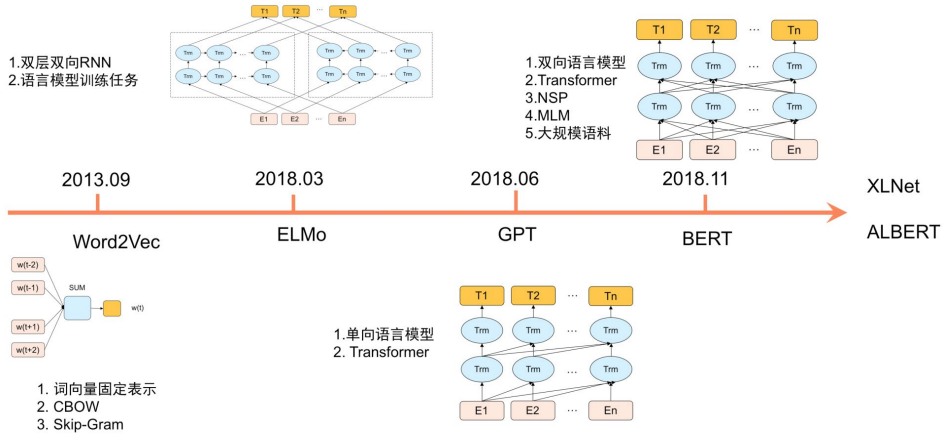


图 1　NLP 预训练发展历程

基于预训练好的 BERT 模型可以支持多种下游 NLP 任务。BERT 在下游任务中的应用主要有两种方式：即 Feature-based 和 Finetune-based。其中 Feature-based 方法将 BERT 作为文本编码器获取文本表示向量，从而完成文本相似度计算、向量召回等任务。而 Finetune-based 方法是在预训练模型的基础上，使用具体任务的部分训练数据进行训练，从而针对性地修正预训练阶段获得的网络参数。该方法更为主流，在大多数任务上效果也更好。

由于 BERT 在 NLP 任务上的显著优势，一些研究工作开始将 BERT 应用于文档排序等信息检索任务中。清华大学 Qiao 等人 [18] 详细对比了 Feature-based 和 Finetune-based 两种应用方式在段落排序（Passage Ranking）中的效果。滑铁卢大学 Jimmy Lin 团队 [19] 针对文档排序任务提出了基于 Pointwise 和 Pairwise 训练目标的 MonoBERT 和 DuoBERT 模型。此外，该团队 [20] 提出融合基于 BERT 的 Query-Doc 相关性和 Query-Sentence 相关性来优化文档排序任务的方案。为了优化检索性能和效果，Bing 广告团队 [21] 提出一种双塔结构的 TwinBERT 分别编码 Query 和 Doc 文本。2019 年 10 月，Google 在其官方博客介绍了 BERT 在 Google 搜索排序和精选摘要（Featured Snippet）场景的应用，BERT 强大的语义理解能力改善了约 10% 的 Google 搜索结果 [22]，除了英文网页，Google 也正在基于 BERT 优化其他语言的搜索结果。值得一提的是美团 AI 平台搜索与 NLP 部在

WSDM Cup 2020 检索排序评测任务中提出了基于 Pairwise 模式的 BERT 排序模型和基于 LightGBM 的排序模型，取得了榜单第一名的成绩 [23]。

搜索相关性

美团搜索场景下相关性任务定义如下：给定用户 Query 和候选 Doc（通常为商户或商品），判断两者之间相关性。搜索 Query 和 Doc 的相关性直接反映结果页排序的优劣，将相关性高的 Doc 排在前面，能提高用户搜索决策效率和搜索体验。为了提升结果的相关性，我们在召回、排序等多个方面做了优化，本文主要讨论在排序方面的优化。通过先对 Query 和 Doc 的相关性进行建模，把更加准确的相关性信息输送给排序模型，从而提升排序模型的排序能力。Query 和 Doc 的相关性计算是搜索业务核心技术之一，根据计算方法相关性主要分为字面相关性和语义相关性。

字面相关性

早期的相关性匹配主要是根据 Term 的字面匹配度来计算相关性，如字面命中、覆盖程度、TFIDF、BM25 等。字面匹配的相关性特征在美团搜索排序模型中起着重要作用，但字面匹配有它的局限，主要表现在：

- **词义局限**：字面匹配无法处理同义词和多义词问题，如在美团业务场景下“宾馆”和“旅店”虽然字面上不匹配，但都是搜索“住宿服务”的同义词；而“COCO”是多义词，在不同业务场景下表示的语义不同，可能是奶茶店，也可能是理发店。
- **结构局限**：“蛋糕奶油”和“奶油蛋糕”虽词汇完全重合，但表达的语义完全不同。 当用户搜“蛋糕奶油”时，其意图往往是找“奶油”，而搜“奶油蛋糕”的需求基本上都是“蛋糕”。

语义相关性

为了解决上述问题，业界工作包括传统语义匹配模型和深度语义匹配模型。传统语义匹配模型包括：

- **隐式模型**：将 Query、Doc 都映射到同一个隐式向量空间，通过向量相似度来计算 Query-Doc 相关性，例如使用主题模型 LDA[24] 将 Query 和 Doc 映射到同一向量空间；
- **翻译模型**：通过统计机器翻译方法将 Doc 进行改写后与 Query 进行匹配 [25]。

这些方法弥补了字面匹配方法的不足，不过从实际效果上来看，还是无法很好地解决语义匹配问题。随着深度自然语言处理技术的兴起，基于深度学习的语义匹配方法成为研究热点，主要包括基于表示的匹配方法（Representation-based）和基于交互的匹配方法（Interaction-based）。

基于表示的匹配方法：使用深度学习模型分别表征 Query 和 Doc，通过计算向量相似度来作为语义匹配分数。微软的 DSSM[26] 及其扩展模型属于基于表示的语义匹配方法，美团搜索借鉴 DSSM 的双塔结构思想，左边塔输入 Query 信息，右边塔输入 POI、品类信息，生成 Query 和 Doc 的高阶文本相关性、高阶品类相关性特征，应用于排序模型中取得了很好的效果。此外，比较有代表性的表示匹配模型还有百度提出 SimNet[27]，中科院提出的多视角循环神经网络匹配模型（MV-LSTM）[28] 等。

基于交互的匹配方法：这种方法不直接学习 Query 和 Doc 的语义表示向量，而是在神经网络底层就让 Query 和 Doc 提前交互，从而获得更好的文本向量表示，最后通过一个 MLP 网络获得语义匹配分数。代表性模型有华为提出的基于卷积神经网络的匹配模型 ARC-II[29]，中科院提出的基于矩阵匹配的的层次化匹配模型 MatchPyramid[30]。

基于表示的匹配方法优势在于 Doc 的语义向量可以离线预先计算，在线预测时只需要重新计算 Query 的语义向量，缺点是模型学习时 Query 和 Doc 两者没有任何交互，不能充分利用 Query 和 Doc 的细粒度匹配信号。基于交互的匹配方法优势在于 Query 和 Doc 在模型训练时能够进行充分的交互匹配，语义匹配效果好，缺点是部署上线成本较高。

BERT 语义相关性

BERT 预训练使用了大量语料，通用语义表征能力更好，BERT 的 Transformer 结构特征提取能力更强。中文 BERT 基于字粒度预训练，可以减少未登录词（OOV）的影响，美团业务场景下存在大量长尾 Query（如大量数字和英文复合 Query）字粒度模型效果优于词粒度模型。此外，BERT 中使用位置向量建模文本位置信息，可以解决语义匹配的结构局限。综上所述，我们认为 BERT 应用在语义匹配任务上会有更好的效果，基于 BERT 的语义匹配有两种应用方式：

- **Feature-based**：属于基于表示的语义匹配方法。类似于 DSSM 双塔结构，通过 BERT 将 Query 和 Doc 编码为向量，Doc 向量离线计算完成进入索引，Query 向量线上实时计算，通过近似最近邻（ANN）等方法实现相关 Doc 召回。
- **Finetune-based**：属于基于交互的语义匹配方法，将 Query 和 Doc 对输入 BERT 进行句间关系 Fine-tuning，最后通过 MLP 网络得到相关性分数。

Feature-based 方式是经过 BERT 得到 Query 和 Doc 的表示向量，然后计算余弦相似度，所有业务场景下 Query-Doc 相似度都是固定的，不利于适配不同业务场景。此外，在实际场景下为海量 Doc 向量建立索引存储成本过高。因此，我们选择了 Finetune-based 方案，利用搜索场景中用户点击数据构造训练数据，然后通过 Fine-tuning 方式优化 Query-Doc 语义匹配任务。图 2 展示了基于 BERT 优化美团搜索核心排序相关性的技术架构图，主要包括三部分：

- **数据样本增强**：由于相关性模型的训练基于搜索用户行为标注的弱监督数据，我们结合业务经验对数据做了去噪和数据映射。为了更好地评价相关性模型的离线效果，我们构建了一套人工标注的 Benchmark 数据集，指导模型迭代方向。
- **BERT 领域适配**：美团业务场景中，Query 和 Doc 以商户、商品、团购等短文本为主，除标题文本以外，还存在商户 / 商品描述、品类、地址、图谱标签等结构化信息。我们首先改进了 MT-BERT 预训练方法，将品类、标签等文

本信息也加入 MT-BERT 预训练过程中。在相关性 Fine-tuning 阶段，我们对训练目标进行了优化，使得相关性任务和排序任务目标更加匹配，并进一步将两个任务结合进行联合训练。此外，由于 BERT 模型前向推理比较耗时，难以满足上线要求，我们通过知识蒸馏将 12 层 BERT 模型压缩为符合上线要求的 2 层小模型，且无显著的效果损失。

- **排序模型优化**：核心排序模型（本文记为 L2 模型）包括 LambdaDNN[31]、TransformerDNN[3]、MultiTaskDNN 等深度学习模型。给定，我们将基于 BERT 预测的 Query-Doc 相关性分数作为特征用于 L2 模型的训练中。

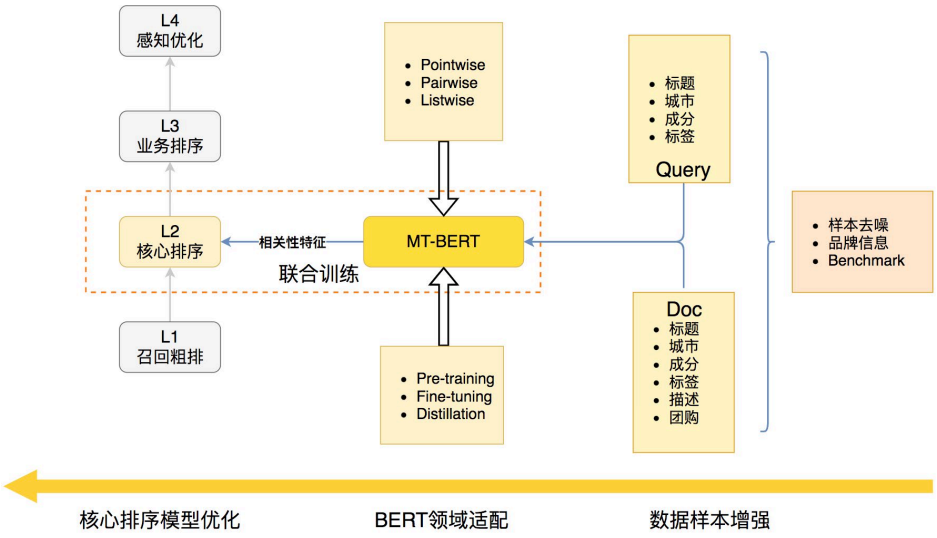


图 2　美团搜索核心排序相关性优化技术架构图

算法探索

数据增强

BERT Fine-tuning 任务需要一定量标注数据进行迁移学习训练，美团搜索场景下 Query 和 Doc 覆盖多个业务领域，如果采用人工标注的方法为每个业务领域标注一批训练样本，时间和人力成本过高。我们的解决办法是使用美团搜索积累的大量用户行为数据（如浏览、点击、下单等），这些行为数据可以作为弱监督训练数据。在

DSSM 模型进行样本构造时，每个 Query 下抽取 1 个正样本和 4 个负样本，这是比较常用的方法，但是其假设 Query 下的 Doc 被点击就算是相关的，这个假设在实际的业务场景下会给模型引入一些噪声。

此处以商家（POI）搜索为例，理想情况下如果一个 POI 出现在搜索结果里，但是没有任何用户点击，可认为该 POI 和 Query 不相关；如果该 POI 有点击或下单行为，可认为该 POI 和 Query 相关。下单行为数据是用户“用脚投票”得来的，具有更高的置信度，因此我们使用下单数据作为正样本，使用未点击过的数据构造负样本，然后结合业务场景对样本进一步优化。数据优化主要包括对样本去噪和引入品牌数据两个方面。此外，为了评测算法离线效果，我们从构造样本中随机采样 9K 条样本进行了人工标注作为 Benchmark 数据集。

样本去噪

无意义单字 Query 过滤。由于单字 Query 表达的语义通常不完整，用户点击行为也比较随机，如 < 优，花漾星球专柜（中央大道倍客优）>，这部分数据如果用于训练会影响最终效果。我们去除了包含无意义单字 Query 的全部样本。

正样本从用户下单的 POI 中进行随机采样，且过滤掉 Query 只出现在 POI 的分店名中的样本，如 < 大润发，小龙坎老火锅（大润发店）>，虽然 Query 和 POI 字面匹配，但其实是不相关的结果。

负样本尝试了两种构造方法：全局随机负采样和 Skip-Above 采样。

- **全局随机负采样**：用户没有点击的 POI 进行随机采样得到负例。我们观察发现随机采样同样存在大量噪声数据，补充了两项过滤规则来过滤数据。① 大量的 POI 未被用户点击是因为不是离用户最近的分店，但 POI 和 Query 是相关的，这种类型的样例需要过滤掉，如 < 蛙小侠 ，蛙小侠（新北万达店）>。② 用户 Query 里包含品牌词，并且 POI 完全等同于品牌词的，需要从负样本中过滤，如 < 德克士吃饭 ，德克士 >。
- **Skip-Above 采样**：受限于 App 搜索场景的展示屏效，无法保证召回的 POI

一次性得到曝光。若直接将未被点击的 POI 作为负例，可能会将未曝光但相关的 POI 错误地采样为负例。为了保证训练数据的准确性，我们采用 Skip-Above 方法，剔除这些噪音负例，即从用户点击过的 POI 之上没有被点击过的 POI 中采样负例（假设用户是从上往下浏览的 POI）。

品牌样本优化

美团商家中有很多品牌商家，通常品牌商家拥有数百上千的 POI，如“海底捞”、“肯德基”、“香格里拉酒店”等，品牌 POI 名称多是“品牌 + 地标”文本形式，如“北京香格里拉饭店”。对 Query 和 POI 的相关性进行建模时，如果仅取 Query 和 POI 名进行相关性训练，POI 名中的“地标”会给模型带来很大干扰。例如，用户搜“香格里拉酒店”时会召回品牌“香格里拉酒店”的分店，如“香格里拉酒店”和“北京香格里拉饭店”等，相关性模型受地标词影响，会给不同分店会打出不同的相关性分数，进而影响到后续排序模型的训练。因此，我们对于样本中的品牌搜索样本做了针对性优化。搜索品牌词有时会召回多个品牌的结果，假设用户搜索的品牌排序靠后，而其他品牌排序靠前会严重影响到用户体验，因此对 Query 和 POI 相关性建模时召回结果中其他品牌的 POI 可认为是不相关样本。针对上述问题，我们利用 POI 的品牌信息对样本进行了重点优化。

- **POI 名映射到品牌**：在品牌搜 Query 不包含地标词的时候，将 POI 名映射到品牌（非品牌 POI 不进行映射），从而消除品牌 POI 分店名中地标词引入的噪声。如 Query 是“香格里拉酒店”，召回的“香格里拉大酒店”和“北京香格里拉饭店”统一映射为品牌名“香格里拉酒店”。Query 是“品牌 + 地标”形式（如“香格里拉饭店 北京”）时，用户意图明确就是找某个地点的 POI，不需要进行映射，示例如下图 3 所示。
- **负样本过滤**：如果搜索词是品牌词，在选取负样本的时候只在其他品牌的样本中选取。如 POI 为“香格里拉实力希尔顿花园酒店”、“桔子香格里拉古城酒店”时，同 Query“香格里拉酒店”虽然字面很相似，但其明显不是用户想要的品牌。

经过样本去噪和品牌样本优化后，BERT 相关性模型在 Benchmark 上的 Accuracy 提升 23BP，相应地 L2 排序排序模型离线 AUC 提升 17.2BP。

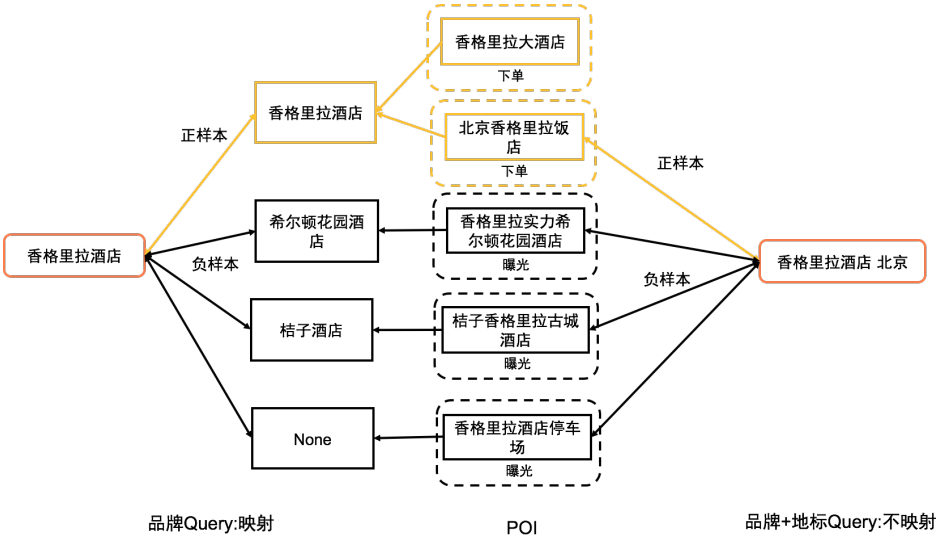


图 3　POI 品牌信息优化样本示意图

模型优化

知识融合

我们团队基于美团业务数据构建了餐饮娱乐领域知识图谱—“美团大脑”[32]，对于候选 Doc（POI/SPU），通过图谱可以获取到该 Doc 的大量结构化信息，如地址、品类、团单，场景标签等。美团搜索场景中的 Query 和 Doc 都以短文本为主，我们尝试在预训练和 Fine-tuning 阶段融入图谱品类和实体信息，弥补 Query 和 Doc 文本信息的不足，强化语义匹配效果。

引入品类信息的预训练

由于美团搜索多模态的特点，在某些情况下，仅根据 Query 和 Doc 标题文本信息很难准确判断两者之间的语义相关性。如 < 考研班，虹蝶教育 >，Query 和 Doc 标题文本相关性不高，但是“虹蝶教育”三级品类信息分别是“教育 – 升学辅导 – 考研”，

引入相关图谱信息有助于提高模型效果，我们首先基于品类信息做了尝试。

在相关性判别任务中，BERT 模型的输入是对。对于每一个输入的 Token，它的表征由其对应的词向量（Token Embedding）、片段向量（Segment Embedding）和位置向量（Position Embedding）相加产生。为了引入 Doc 品类信息，我们将 Doc 三级品类信息拼接到 Doc 标题之后，然后跟 Query 进行相关性判断，如图 4 所示。

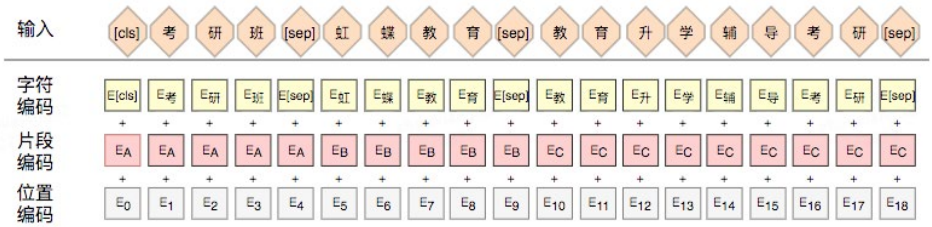


图 4　BERT 输入部分加入 Doc（POI）品类信息

对于模型输入部分，我们将 Query、Doc 标题、三级类目信息拼接，并用 [SEP] 分割，区分 3 种不同来源信息。对于段向量，原始的 BERT 只有两种片段编码 EA 和 EB，在引入类目信息的文本信息后，引入额外的片段编码 EC。引入额外片段编码的作用是防止额外信息对 Query 和 Doc 标题产生交叉干扰。由于我们改变了 BERT 的输入和输出结构，无法直接基于 MT-BERT 进行相关性 Fine-tuning 任务。我们对 MT-BERT 的预训练方式做了相应改进，BERT 预训练的目标之一是 NSP（Next Sentence Prediction），在搜索场景中没有上下句的概念，在给定用户的搜索关键词和商户文本信息后，判断用户是否点击来取代 NSP 任务。

添加品类信息后，BERT 相关性模型在 Benchmark 上的 Accuracy 提升 56BP，相应地 L2 排序模型离线 AUC 提升 6.5BP。

引入实体成分识别的多任务 Fine-tuning

在美团搜索场景中，Query 和 Doc 通常由不同实体成分组成，如美食、酒店、商圈、品牌、地标和团购等。除了文本语义信息，这些实体成分信息对于 Query-Doc 相关性判断至关重要。如果 Query 和 Doc 语义相关，那两者除了文本语义相似

外，对应的实体成分也应该相似。例如，Query 为“Helens 海伦司小酒馆”，Doc 为“Helens 小酒馆（东鼎购物中心店）”，虽然文本语义不完全匹配，但二者的主要的实体成分相似（主体成分为品牌 +POI 形式），正确的识别出 Query/Doc 中的实体成分有助于相关性的判断。微软的 MT-DNN[33] 已经证明基于预训练模型的多任务 Fine-tuning 可以提升各项子任务效果。由于 BERT Fine-tuning 任务也支持命名实体识别（NER）任务，因而我们在 Query-Doc 相关性判断任务的基础上引入 Query 和 Doc 中实体成分识别的辅助任务，通过对两个任务的联合训练来优化最终相关性判别结果，模型结构如下图 5 所示：

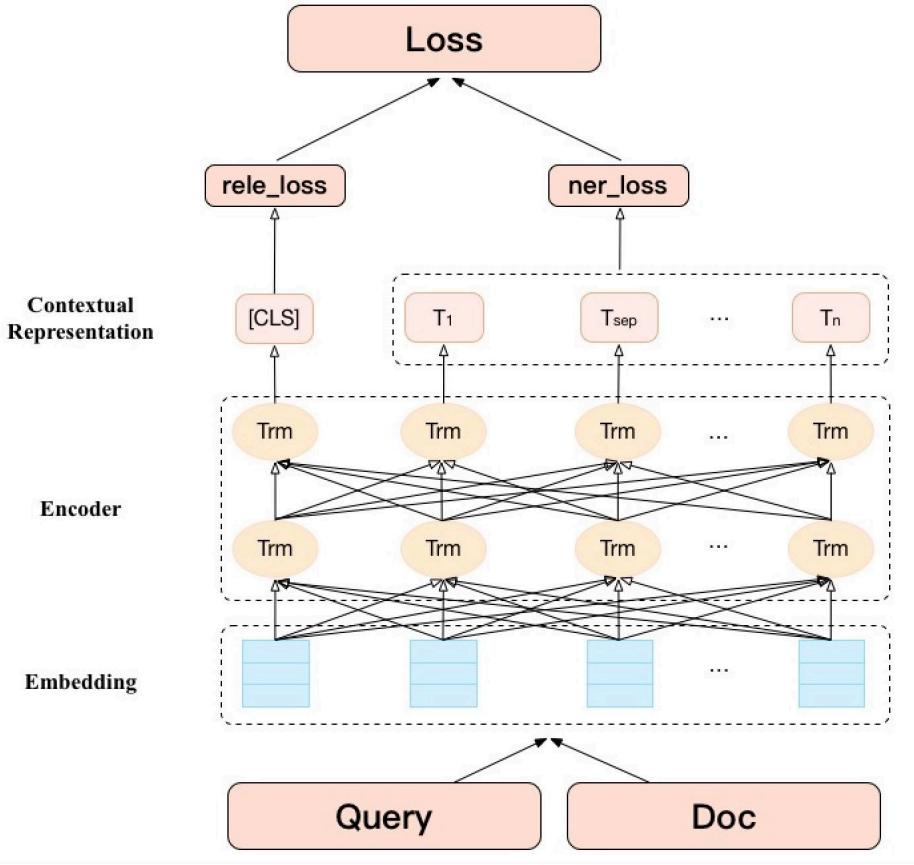


图 5　实体成分一致性学习模型结构

多任务学习模型的损失函数由两部分组成，分别是相关性判断损失函数和命名实体识别损失函数。其中相关性损失函数由 [CLS] 位的 Embedding 计算得到，而实体成分识别损失函数由每个 Token 的 Embedding 计算得到。2 种损失函数相加即为最终优化的损失函数。在训练命名实体识别任务时，每个 Token 的 Embedding 获得了和自身实体相关的信息，从而提升了相关性任务的效果。

引入实体成分识别的多任务 Fine-tuning 方式后，BERT 相关性模型在 Benchmark 上的 Accuracy 提升 219BP，相应地 L2 排序模型 AUC 提升 17.8BP。

Pairwise Fine-tuning

Query-Doc 相关性最终作为特征加入排序模型训练中，因此我们也对 Fine-tuning 任务的训练目标做了针对性改进。基于 BERT 的句间关系判断属于二分类任务，本质上是 Pointwise 训练方式。Pointwise Fine-tuning 方法可以学习到很好的全局相关性，但忽略了不同样本之前的偏序关系。如对于同一个 Query 的两个相关结果 DocA 和 DocB，Pointwise 模型只能判断出两者都与 Query 相关，无法区分 DocA 和 DocB 相关性程度。为了使得相关性特征对于排序结果更有区分度，我们借鉴排序学习中 Pairwise 训练方式来优化 BERT Fine-tuning 任务。

Pairwise Fine-tuning 任务输入的单条样本为三元组，对于同一 Query 的多个候选 Doc，选择任意一个正例和一个负例组合成三元组作为输入样本。在下游任务中只需要使用少量的 Query 和 Doc 相关性的标注数据（有监督训练样本），对 BERT 模型进行相关性 Fine-tuning，产出 Query 和 Doc 的相关性特征。Pairwise Fine-tuning 的模型结构如下图 6 所示：

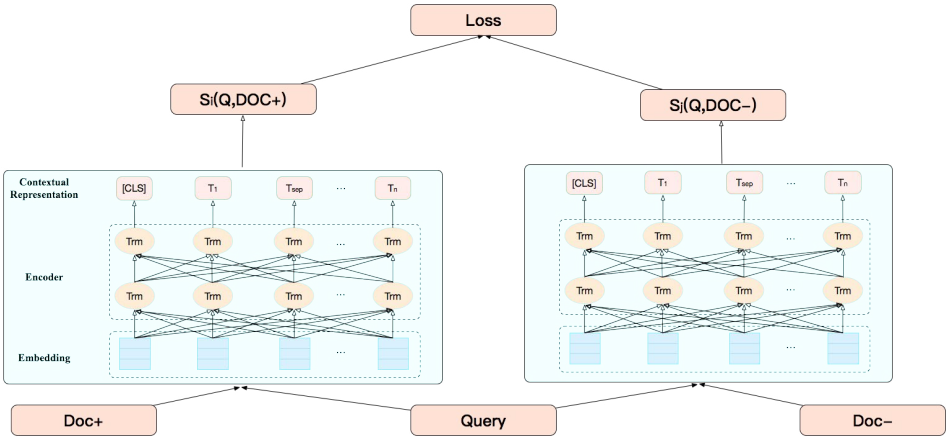


图 6　Pairwise Fine-tuning 模型结构

对于同一 Query 的候选 Doc，选择两个不同标注的 Doc，其中相关文档记为 Doc+，不相关文档记 Doc-。输入层通过 Lookup Table 将 Query, Doc+ 以及 Doc- 的单词转换为 Token 向量，同时会拼接位置向量和片段向量，形成最终输入向量。接着通过 BERT 模型可以分别得到（Query, Doc+）以及（Query, Doc-）的语义相关性表征，即 BERT 的 CLS 位输出。 经过 Softmax 归一化后，可以分别得到（Query, Doc+）和（Query, Doc-）的语义相似度打分。

对于同一 Query 的候选 Doc，选择两个不同标注的 Doc，其中相关文档记为 Doc+，不相关文档记 Doc-。输入层通过 Lookup Table 将 Query, Doc+ 以及 Doc- 的单词转换为 Token 向量，同时会拼接位置向量和片段向量，形成最终输入向量。接着通过 BERT 模型可以分别得到（Query, Doc+）以及（Query, Doc-）的语义相关性表征，即 BERT 的 CLS 位输出。 经过 Softmax 归一化后，可以分别得到（Query, Doc+）和（Query, Doc-）的语义相似度打分。

Pairwise Fine-tuning 除了输入样本上的变化，为了考虑搜索场景下不同样本之间的偏序关系，我们参考 RankNet[34] 的方式对训练损失函数做了优化。 令 PijPij 为同一个 Query 下 DociDoci 相比 DocjDocj 更相关的概率，其中 sisi 和 sjsj 分别为 DociDoci 和 DocjDocj 的模型打分，则：

$$P_{ij} = \frac{1}{1 + e^{-\sigma(s_i - s_j)}}$$

使用交叉熵损失函数，令SijSij表示样本对的真实标记，当DociDoci比DocjDocj更相关时（即DociDoci为正例而$Docj为负例），为负例），*S{ij}$为1，否则为-1，损失函数可以表示为：

$$C = \sum_{(i,j) \in N} \frac{1}{2}(1 - S_{ij})\sigma(s_i - s_j) + log(1 + e^{-\sigma(s_i - s_j)})$$

其中NN表示所有在同Query下的Doc对。

使用Pairwise Fine-tuning方式后，BERT相关性模型在Benchmark上的Accuracy提升925BP，相应地L2排序模型的AUC提升19.5BP。

联合训练

前文所述各种优化属于两阶段训练方式，即先训练BERT相关性模型，然后训练L2排序模型。为了将两者深入融合，在排序模型训练中引入更多相关性信息，我们尝试将BERT相关性Fine-tuning任务和排序任务进行端到端的联合训练。

由于美团搜索涉及多业务场景且不同场景差异较大，对于多场景的搜索排序，每个子场景进行单独优化效果好，但是多个子模型维护成本更高。此外，某些小场景由于训练数据稀疏无法学习到全局的Query和Doc表征。我们设计了基于Partition-model的BERT相关性任务和排序任务的联合训练模型，Partition-model的思想是利用所有数据进行全场景联合训练，同时一定程度上保留每个场景特性，从而解决多业务场景的排序问题，模型结构如下图7所示：

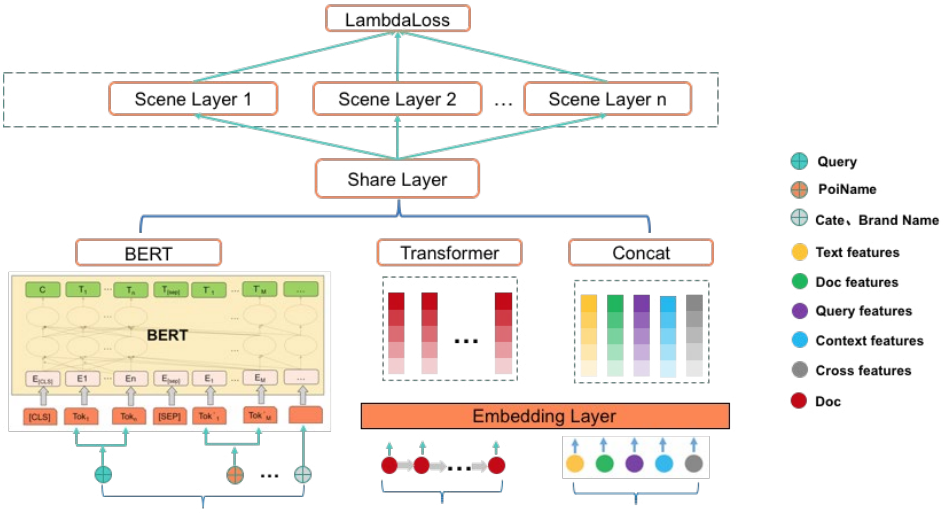


图 7　联合训练模型结构

输入层：模型输入是由文本特征向量、用户行为序列特征向量和其他特征向量 3 部分组成。

- 文本特征向量使用 BERT 进行抽取，文本特征主要包括 Query 和 POI 相关的一些文本（POI 名称、品类名称、品牌名称等）。将文本特征送入预训练好的 MT-BERT 模型，取 CLS 向量作为文本特征的语义表示。
- 用户行为序列特征向量使用 Transformer 进行抽取 [3]。
- 其他特征主要包括：① 统计类特征，包含 Query、Doc 等维度的特征以及它们之间的交叉特征，使用这些特征主要是为了丰富 Query 和 Doc 的表示，更好地辅助相关性任务训练。② 文本特征，这部分的特征同 1 中的文本特征，但是使用方式不同，直接将文本分词后做 Embedding，端到端的学习文本语义表征。③ 传统的文本相关性特征，包括 Query 和 Doc 的字面命中、覆盖程度、BM25 等特征，虽然语义相关性具有较好的作用，但字面相关性仍然是一个不可或缺的模块，它起到信息补充的作用。

共享层：底层网络参数是所有场景网络共享。

场景层：根据业务场景进行划分，每个业务场景单独设计网络结构，打分时只经过所在场景的那一路。

损失函数：搜索业务更关心排在页面头部结果的好坏，将更相关的结果排到头部，用户会获得更好的体验，因此选用优化 NDCG 的 Lambda Loss[34]。

联合训练模型目前还在实验当中，离线实验已经取得了不错的效果，在验证集上 AUC 提升了 234BP。目前，场景切分依赖 Query 意图模块进行硬切分，后续自动场景切分也值得进行探索。

应用实践

由于 BERT 的深层网络结构和庞大参数量，如果要部署上线，实时性上面临很大挑战。在美团搜索场景下，我们对基于 MT-BERT Fine-tuning 好的相关性模型（12 层）进行了 50QPS 压测实验，在线服务的 TP99 增加超过 100ms，不符合工程上线要求。我们从两方面进行了优化，通过知识蒸馏压缩 BERT 模型，优化排序服务架构支持蒸馏模型上线。

模型轻量化

为了解决 BERT 模型参数量过大、前向计算耗时的问题，常用轻量化方法有三种：

- **知识蒸馏**：模型蒸馏是在一定精度要求下，将大模型学到的知识迁移到另一个轻量级小模型上，目的是降低预测计算量的同时保证预测效果。Hinton 在 2015 年的论文中阐述了核心思想 [35]，大模型一般称作 Teacher Model，蒸馏后的小模型一般称作 Student Model。具体做法是先在训练数据上学习 Teacher Model，然后 Teacher Model 对无标注数据进行预测得到伪标注数据，最后使用伪标注数据训练 Student Model。HuggingFace 提出的 DistilBERT[36] 和华为提出的 TinyBERT[37] 等 BERT 的蒸馏模型都取得了不错的效果，在保证效果的情况下极大地提升了模型的性能。
- **模型裁剪**：通过模型剪枝减少参数的规模。

- **低精度量化**：在模型训练和推理中使用低精度（FP16 甚至 INT8、二值网络）表示取代原有精度（FP32）表示。

在 Query 意图分类任务[2]中，我们基于 MT-BERT 裁剪为 4 层小模型达到了上线要求。意图分类场景下 Query 长度偏短，语义信息有限，直接裁剪掉几层 Transformer 结构对模型的语义表征能力不会有太大的影响。在美团搜索的场景下，Query 和 Doc 拼接后整个文本序列变长，包含更复杂的语义关系，直接裁剪模型会带来更多的性能损失。因此，我们在上线 Query-Doc 相关性模型之前，采用知识蒸馏方式，在尽可能在保持模型性能的前提下对模型层数和参数做压缩。两种方案的实验效果对比见下表 1：

Model	Accuracy	AUC
MT-BERT裁剪（2 Layers）	73.11%	75.47%
MT-BERT蒸馏（2 Layers）	73.89%	76.07%
MT-BERT（12 Layers）	74.42%	77.32%

表 1　裁剪和知识蒸馏方式效果对比

在美团搜索核心排序的业务场景下，我们采用知识蒸馏使得 BERT 模型在对响应时间要求苛刻的搜索场景下符合了上线的要求，并且效果无显著的性能损失。知识蒸馏（Knowledge Distillation）核心思想是通过迁移知识，从而通过训练好的大模型得到更加适合推理的小模型。首先我们基于 MT-BERT（12 Layers），在大规模的美团点评业务语料上进行知识蒸馏得到通用的 MT-BERT 蒸馏模型（6 Layers），蒸馏后的模型可以作为具体下游任务 Fine-tuning 时的初始化模型。在美团搜索的场景下，我们进一步基于通用的 MT-BERT 蒸馏模型（6 Layers）进行相关性任务

Fine-tuning，得到 MT-BERT 蒸馏（2 Layers）进行上线。

排序服务架构优化

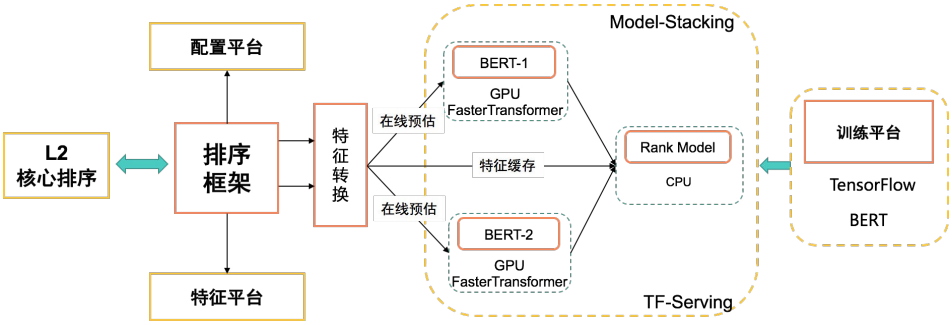


图 8　核心排序框架图

美团搜索线上排序服务框架如上图 8 所示，主要包括以下模块：

- **模型在线预估框架（Augur）**：支持语言化定义特征，配置化加载和卸载模型与特征，支持主流线性模型与 TF 模型的在线预估；基于 Augur 可以方便地构建功能完善的无状态、分布式的模型预估服务。为了能方便将 BERT 特征用于排序模型，Augur 团队开发了 Model Stacking 功能，完美支持了 BERT as Feature；这种方式将模型的分数当做一个特征，只需要在 Augur 服务模型配置平台上进行特征配置即可，很好地提升了模型特征的迭代效率。
- **搜索模型实验平台（Poker）**：支持超大规模数据和模型的离线特征抽取、模型训练，支持 BERT 模型自助训练 /Fine-tuning 和预测；同时打通了 Augur 服务，训练好的模型可以实现一键上线，大大提升了模型的实验效率。

TF-Serving 在线模型服务：L2 排序模型、BERT 模型上线使用 TF-Serving 进行部署。TF-Serving 预测引擎支持 Faster Transformer[38] 加速 BERT 推理，提升了线上的预估速度。

为了进一步提升性能，我们将头部 Query 进行缓存只对长尾 Query 进行在线打分，线上预估结合缓存的方式，即节约了 GPU 资源又提升了线上预估速度。经过上述优

化，我们实现了 50 QPS 下，L2 模型 TP99 只升高了 2ms，满足了上线的要求。

线上效果

针对前文所述的各种优化策略，除了离线 Benchmark 上的效果评测之外，我们也将模型上线进行了线上 AB 评测，Baseline 是当前未做任何优化的排序模型，我们独立统计了各项优化在 Baseline 基础上带来的变化，由于线上真实环境影响因素较多，为了确保结论可信，我们同时统计了 QVCTR 和 NDCG 两个指标，结果如表 2 所示：

Model	Accuracy	AUC
样本去噪	+7.95BP*	-1.9BP
品牌信息	+23BP*	+3.3BP*
品类知识	+12.8BP*	+4.3BP*
Pairwise	+0.4BP	+4.9BP*
All	+26.3BP*	+3.8BP*

表 2　线上 AB 效果对比（* 表示 AB 一周稳定正向）

从表 2 可以看出，各项优化对线上排序核心指标都带来稳定的提升。用户行为数据存在大量噪声不能直接拿来建模，我们基于美团搜索排序业务特点设计了一些规则对训练样本进行优化，还借助 POI 的品牌信息对样本进行映射和过滤。通过人工对样本进行评测发现，优化后的样本更加符合排序业务特点以及“人”对相关性的认知，同时线上指标的提升也验证了我们优化的有效性。知识融合的 BERT 模型引入大量结

构化文本信息，弥补了 POI 名本身文本信息少的问题，排序模型 CTR 和 NDCG 都有明显的提升。对数据样本的优化有了一定的效果。为了更加匹配业务场景，我们从模型的角度进行优化，模型损失函数改用排序任务常用的 Pairwise Loss，其考虑了文档之间的关系更加贴合排序任务场景，线上排序模型 NDCG 取得了一定的提升。

总结与展望

本文总结了搜索与 NLP 算法团队基于 BERT 在美团搜索核心排序落地的探索过程和实践经验，包括数据增强、模型优化和工程实践。在样本数据上，我们结合了美团搜索业务领域知识，基于弱监督点击日志构建了高质量的训练样本；针对美团搜索多模态特点，在预训练和 Fine-tuning 阶段融合图谱品类和标签等信息，弥补 Query 和 Doc 文本较短的不足，强化文本匹配效果。

在算法模型上，结合搜索排序优化目标，引入了 Pairwise/Listwise 的 Fine-tuning 训练目标，相比 Pointwise 方式在相关性判断上更有区分度。这些优化在离线 Benchmark 评测和线上 AB 评测中带来了不同幅度的指标提升，改善了美团搜索的用户体验。

在工程架构上，针对 BERT 在线预估性能耗时长的问题，参考业界经验，我们采用了 BERT 模型轻量化的方案进行模型蒸馏裁剪，既保证模型效果又提升了性能，同时我们对整体排序架构进行了升级，为后续快速将 BERT 应用到线上预估奠定了良好基础。

搜索与 NLP 算法团队会持续进行探索 BERT 在美团搜索中的应用落地，我们接下来要进行以下几个优化：

- 融合知识图谱信息对长尾流量相关性进行优化：美团搜索承接着多达几十种生活服务的搜索需求，当前头部流量相关性问题已经较好地解决，长尾流量的相关性优化需要依赖更多的高质量数据。我们将利用知识图谱信息，将一些结构化先验知识融入到 BERT 预训练中，对长尾 Query 的信息进行增强，使其可以更好地进行语义建模。

- 相关性与其他任务联合优化：美团搜索场景下 Query 和候选 Doc 都更结构化，除文本语义匹配外，Query/Doc 文本中蕴含的实体成分、意图、类目也可以用于辅助相关性判断。目前，我们将相关性任务和成分识别任务结合进行联合优化已经取得一定效果。后续我们考虑将意图识别、类目预测等任务加入相关性判断中，多视角、更全面地评估 Query-Doc 的相关性。
- BERT 相关性模型和排序模型的深入融合：当前两个模型属于两阶段训练方式，将 BERT 语义相关性作为特征加入排序模型来提升点击率。语义相关性是影响搜索体验的重要因素之一，我们将 BERT 相关性和排序模型进行端到端联合训练，将相关性和点击率目标进行多目标联合优化，提升美团搜索排序的综合体验。

参考资料

[1] Devlin, Jacob, et al. “BERT: Pre-training of deep bidirectional transformers for language understanding.” arXiv preprint arXiv:1810.04805 (2018).
[2] 杨扬、佳昊等 . 美团 BERT 的探索和实践
[3] 肖垚、家琪等 . Transformer 在美团搜索排序中的实践
[4] Mikolov, Tomas, et al. “Efficient estimation of word representations in vector space.” arXiv preprint arXiv:1301.3781 (2013).
[5] Peters, Matthew E., et al. “Deep contextualized word representations.” arXiv preprint arXiv:1802.05365 (2018).
[6] Vaswani, Ashish, et al. “Attention is all you need.” Advances in neural information processing systems. 2017.
[7] Radford, Alec, et al. “Improving language understanding by generative pre-training.”
[8] Sun, Yu, et al. “Ernie: Enhanced representation through knowledge integration.” arXiv preprint arXiv:1904.09223 (2019).
[9] Zhang, Zhengyan, et al. “ERNIE: Enhanced language representation with informative entities.” arXiv preprint arXiv:1905.07129 (2019).
[10] Liu, Weijie, et al. “K-bert: Enabling language representation with knowledge graph.” arXiv preprint arXiv:1909.07606 (2019).
[11] Sun, Yu, et al. “Ernie 2.0: A continual pre-training framework for language understanding.” arXiv preprint arXiv:1907.12412 (2019).
[12] Liu, Yinhan, et al. “Roberta: A robustly optimized bert pretraining approach.” arXiv preprint arXiv:1907.11692 (2019).
[13] Joshi, Mandar, et al. “Spanbert: Improving pre-training by representing and

predicting spans." Transactions of the Association for Computational Linguistics 8 (2020): 64-77.
[14] Wang, Wei, et al. "StructBERT: Incorporating Language Structures into Pre-training for Deep Language Understanding." arXiv preprint arXiv:1908.04577 (2019).
[15] Lan, Zhenzhong, et al. "Albert: A lite bert for self-supervised learning of language representations." arXiv preprint arXiv:1909.11942 (2019)
[16] Clark, Kevin, et al. "Electra: Pre-training text encoders as discriminators rather than generators." arXiv preprint arXiv:2003.10555 (2020).
[17] Qiu, Xipeng, et al. "Pre-trained Models for Natural Language Processing: A Survey." arXiv preprint arXiv:2003.08271 (2020).
[18] Qiao, Yifan, et al. "Understanding the Behaviors of BERT in Ranking." arXiv preprint arXiv:1904.07531 (2019).
[19] Nogueira, Rodrigo, et al. "Multi-stage document ranking with BERT." arXiv preprint arXiv:1910.14424 (2019).
[20] Yilmaz, Zeynep Akkalyoncu, et al. "Cross-domain modeling of sentence-level evidence for document retrieval." Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP). 2019.
[21] Wenhao Lu, et al. "TwinBERT: Distilling Knowledge to Twin-Structured BERT Models for Efficient Retrieval." arXiv preprint arXiv: 2002.06275
[22] Pandu Nayak.
[23] 帅朋、会星等 .WSDM Cup 2020 检索排序评测任务第一名经验总结
[24] Blei, David M., Andrew Y. Ng, and Michael I. Jordan. "Latent dirichlet allocation." Journal of machine Learning research 3.Jan (2003): 993-1022.
[25] Jianfeng Gao, Xiaodong He, and JianYun Nie. Click-through-based Translation Models for Web Search: from Word Models to Phrase Models. In CIKM 2010.
[26] Huang, Po-Sen, et al. "Learning deep structured semantic models for web search using clickthrough data." Proceedings of the 22nd ACM international conference on Information & Knowledge Management. 2013.
[27] SimNet.
[28] Guo T, Lin T. Multi-variable LSTM neural network for autoregressive exogenous model[J]. arXiv preprint arXiv:1806.06384, 2018.
[29] Hu, Baotian, et al. "Convolutional neural network architectures for matching natural language sentences." Advances in neural information processing systems. 2014.
[30] Pang, Liang, et al. "Text matching as image recognition." Thirtieth AAAI Conference on Artificial Intelligence. 2016.
[31] 非易、祝升等 . 大众点评搜索基于知识图谱的深度学习排序实践 .
[32] 仲远、富峥等 . 美团餐饮娱乐知识图谱——美团大脑揭秘 .
[33] Liu, Xiaodong, et al. "Multi-task deep neural networks for natural language understanding." arXiv preprint arXiv:1901.11504 (2019).

[34] Burges, Christopher JC. “From ranknet to lambdarank to lambdamart: An overview.” Learning 11.23-581 (2010): 81.
[35] Hinton, Geoffrey, Oriol Vinyals, and Jeff Dean. “Distilling the knowledge in a neural network.” arXiv preprint arXiv:1503.02531 (2015).
[36] Sanh, Victor, et al. “DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter.” arXiv preprint arXiv:1910.01108 (2019).
[37] Jiao, Xiaoqi, et al. “Tinybert: Distilling bert for natural language understanding.” arXiv preprint arXiv:1909.10351 (2019).
[38] Faster Transformer.

作者简介

李勇，美团 AI 平台搜索与 NLP 部算法工程师。
佳昊，美团 AI 平台搜索与 NLP 部算法工程师。
杨扬，美团 AI 平台搜索与 NLP 部算法工程师。
金刚，美团 AI 平台搜索与 NLP 部算法专家。
周翔，美团 AI 平台搜索与 NLP 部算法专家。
朱敏，美团 AI 平台搜索与 NLP 部技术专家。
富峥，美团 AI 平台搜索与 NLP 部资深算法专家。
陈胜，美团 AI 平台搜索与 NLP 部资深算法专家。
云森，美团 AI 平台搜索与 NLP 部研究员。
永超，美团 AI 平台搜索与 NLP 部高级研究员。

美团智能配送系统的运筹优化实战

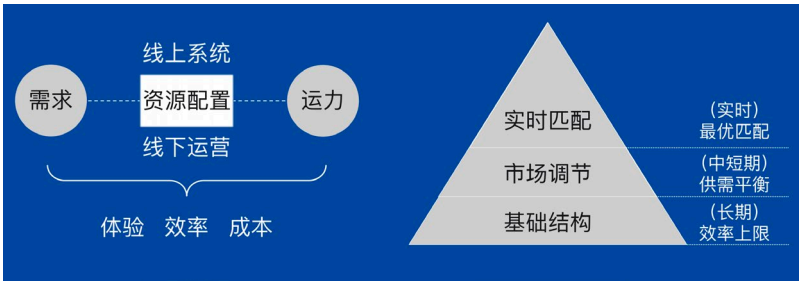
作者：王圣尧

深入各个产业已经成为互联网目前的主攻方向，线上和线下存在大量复杂的业务约束和多种多样的决策变量，为运筹优化技术提供了用武之地。作为美团智能配送系统最核心的技术之一，运筹优化是如何在美团各种业务场景中进行落地的呢？本文根据美团配送技术团队资深算法专家王圣尧在 2019 年 ArchSummit 全球架构师峰会北京站上的演讲内容整理而成。

美团智能配送系统架构

美团配送业务场景复杂，单量规模大。下图这组数字是 2019 年 5 月美团配送品牌发布时的数据。

更直观的规模数字，可能是美团每年给骑手支付的工资，目前已经达到几百亿这个量级。所以，在如此大规模的业务场景下，配送智能化就变得非常重要，而智能配送的核心就是做资源的优化配置。



资源优化配置

外卖配送是一个典型的 O2O 场景。既有线上的业务，也有线下的复杂运营。配送连接订单需求和运力供给。为了达到需求和供给的平衡，不仅要在线下运营商家、运营骑手，还要在线上将这些需求和运力供给做合理的配置，其目的是提高整体的效率。只有将配送效率最大化，才能带来良好的顾客体验，实现较低的配送成本。而做资源优化配置的过程，实际上是有分层的。根据我们的理解，可以分为三层：

1. 基础层是结构优化，它直接决定了配送系统效率的上限。这种基础结构的优化，周期比较长，频率比较低，包括配送网络规划、运力结构规划等等。
2. 中间层是市场调节，相对来说是中短期的，主要通过定价或者营销手段，使供需达到一个相对理想的平衡状态。
3. 再上层是实时匹配，通过调度做实时的资源最优匹配。 实时匹配的频率是最高的，决策的周期也最短。

智能配送系统架构

根据智能配送的这三层体系，配送算法团队也针对性地进行了运作。如上图所示，右边三个子系统分别对应这三层体系，最底层是规划系统，中间层是定价系统，最上层是调度系统。同样非常重要的还包括图中另外四个子系统，在配送过程中做精准的数据采集、感知、预估，为优化决策提供准确的参数输入，包括机器学习系统、IoT 和感知系统、LBS 系统，这都是配送系统中非常重要的环节，涉及大量复杂的机器学习问题。

而运筹优化则是调度系统、定价系统、规划系统的核心技术。接下来，我们分享几个典型的运筹优化案例。

实战业务项目

智能区域规划

为了帮助大家快速理解配送业务的基本背景，这里首先分享智能区域规划项目中经常遇到的问题及其解决方案。

配送网络基本概念

配送连接的是商家、顾客、骑手三方，配送网络决定了这三方的连接关系。当用户打开 App，查看哪些商家可以点餐，这由商家配送范围决定。每个商家的配送范围不一

样，看似是商家粒度的决策，但实际上直接影响每个 C 端用户得到的商流供给，这本身也是一个资源分配或者资源抢夺问题。商家配送范围智能化也是一个组合优化问题，但是我们这里讲的是商家和骑手的连接关系。

用户在美团点外卖，为他服务的骑手是谁呢？又是怎么确定的呢？这些是由配送区域边界来决定的。配送区域边界指的是一些商家集合所对应的范围。为什么要划分区域边界呢？从优化的角度来讲，对于一个确定问题来说，约束条件越少，目标函数值更优的可能性就越大。做优化的同学肯定都不喜欢约束条件，但是配送区域边界实际上就是给配送系统强加的约束。

在传统物流中，影响末端配送效率最关键的点，是配送员对他所负责区域的熟悉程度。这也是为什么在传统物流领域，配送站或配送员，都会固定负责某几个小区的原因之一。因为越熟悉，配送效率就会越高。

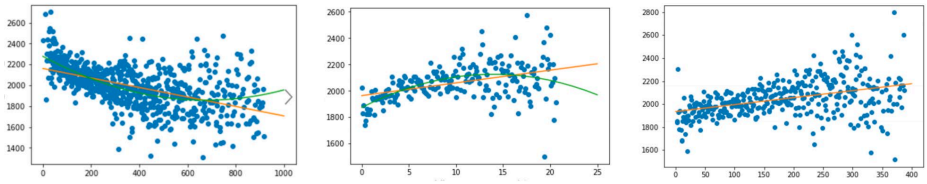
即时配送场景也类似，每个骑手需要尽量固定地去熟悉一片商家或者配送区域。同时，对于管理而言，站点的管理范围也比较明确。另外，如果有新商家上线，也很容易确定由哪个配送站来提供服务。所以，这个问题有很多运营管理的诉求在其中。

区域规划影响配送效率

当然，区域规划项目的发起，存在很多问题需要解决。主要包括以下三种情况：1. 配送区域里的商家不聚合。这是一个典型站点，商家主要集中在左下角和右上角，造成骑手在区域里取餐、送餐时执行任务的地理位置非常分散，需要不停往返两个商圈，

无效跑动非常多。 2. 区域奇形怪状，空驶严重。之前在门店上线外卖平台的发展过程中，很多地方原本没有商家，后来上线的商家多了，就单独作为一个配送区域。这样的区域形状可能就会不规则，导致骑手很多时候在区域外跑。 而商家和骑手都有绑定关系，骑手只能服务自己区域内的商家，因此骑手无法接到配送区域外的取餐任务，空驶率非常高。很多时候骑手送完餐之后，只能空跑回来才可能接到新任务。 3. 站点的大小不合理。图三这个站点，每天的单量只有一二百单。如果从骑手平均单量的角度去配置骑手的话，只能配置 3~4 个骑手。如果某一两个人突然有事要请假，可想而知，站点的配送体验一定会变得非常差，运营管理难度会很高。 反之，如果某一个站点变得非常大，站长也不可能管得了那么多的骑手，这也是一个问题。所以，需要给每个站点规划一个合理的单量规模。

既然存在这么多的问题，那么做区域规划项目就变得非常有必要。那么，什么是好的区域规划方案？基于统计分析的优化目标设定。



多目标优化问题

优化的三要素是：目标、约束、决策变量。

第一点，首先要确定优化目标。在很多比较稳定或者传统的业务场景中，目标非常确定。而在区域规划这个场景中，怎么定义优化目标呢？首先，我们要思考的是区域规划主要影响的是什么。从刚才几类问题的分析可以发现，影响的主要是骑手的顺路性、空驶率，也就是骑手平均为每一单付出的路程成本。所以，我们将问题的业务目标定为优化骑手的单均行驶距离。基于现有的大量区域和站点积累的数据，做大量的统计分析后，可以定义出这样几个指标：商家聚合度、订单的聚合度、订单重心和商家重心的偏离程度。数据分析结果说明，这几个指标和单均行驶距离的相关性很强。

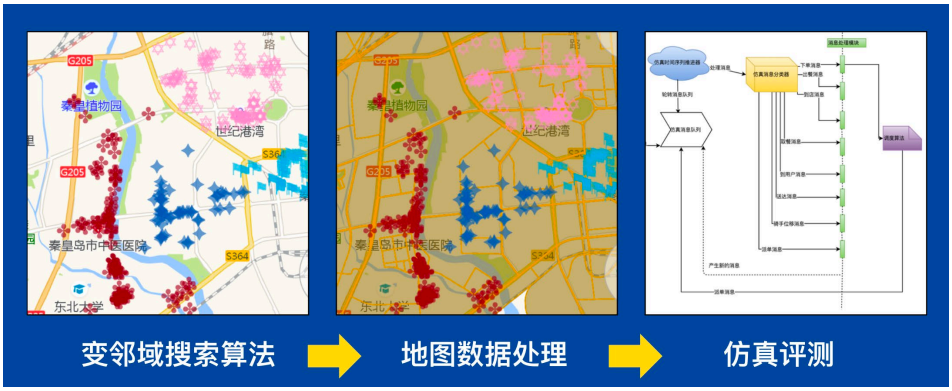
经过这一层的建模转化，问题明确为优化这三个指标。

第二点，需要梳理业务约束。在这方面，我们花费了大量的时间和精力。比如：区域单量有上限和下限；区域之间不能有重合，不能有商家归多个区域负责；所有的 AOI 不能有遗漏，都要被某个区域覆盖到，不能出现商家没有站点的服务。

基于业务场景的约束条件梳理

- 区域单量的上下限
- 区域彼此无交集
- 所有AOI无遗漏
- 区域边界沿路网

最难的一个问题，其实是要求区域边界必须沿路网。起初我们很难理解，因为本质上区域规划只是对商家进行分类，它只是一个商家集合的概念，为什么要画出边界，还要求边界沿路网呢？其实刚才介绍过，区域边界是为了回答如果有新商家上线到底属于哪个站点的问题。而且，从一线管理成本来讲，更习惯于哪条路以东、哪条路以南这样的表述方式，便于记忆和理解，提高管理效率。所以，就有了这样的诉求，我们希望区域边界更“便于理解”。



整体方案设计

在目标和约束条件确定了之后，整体技术方案分成三部分：

1. 首先，根据三个目标函数，确定商家最优集合。这一步比较简单，做运筹优化的同学都可以快速地解决这样一个多目标组合优化问题。
2. 后面的步骤比较难，怎么把区域边界画出来呢？为了解决这个问题，配送团队和美团地图团队进行合作。先利用路网信息，把城市切成若干互不重叠的多边形，然后根据计算几何，将一批商家对应的多边形拼成完整的区域边界。
3. 最后，用美团自主研发的配送仿真系统，评测这样的区域规划对应的单均行驶距离和体验指标是否符合预期。因为一线直接变动的成本非常高，仿真系统就起到了非常好的作用。

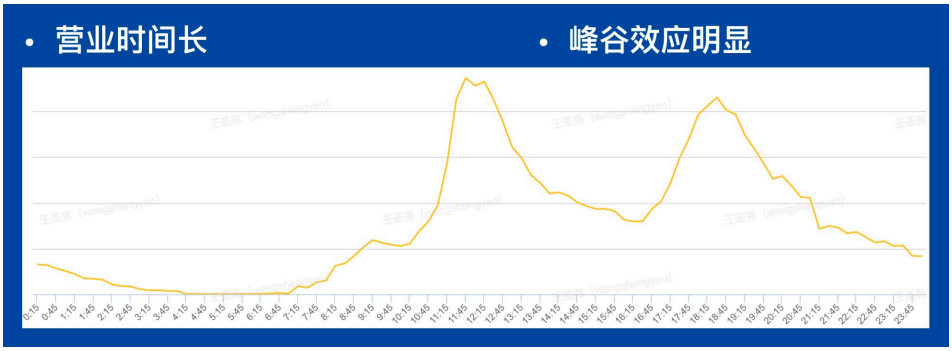
下面是一个实际案例，我们用算法把一个城市做了重新的区域规划。当然，这里必须要强调的是，在这个过程中，人工介入还是非常必要的。对于一些算法很难处理好的边角场景，需要人工进行微调，使整个规划方案更加合理。中间的图是算法规划的结果。经过试点后，测试城市整体的单均行驶距离下降了 5%，平均每一单骑手的行驶距离节省超过 100 米。可以想象一下，在这么庞大的单量规模下，每单平均减少 100 米，总节省的路程、节省的电瓶车电量，都是一个非常可观的数字。更重要的是，可以让骑手自己明显感觉到自己的效率得到了提升。

落地应用效果显著

智能骑手排班

业务背景

这是随着外卖配送的营业时间越来越长而衍生出的一个项目。早期，外卖只服务午高峰到晚高峰，后来大家慢慢可以点夜宵、点早餐。到如今，很多配送站点已经提供了 24 小时服务。但是，骑手不可能全天 24 小时开工，劳动法对每天的工作时长也有规定，所以这一项目势在必行。



另外，外卖配送场景的订单“峰谷效应”非常明显。上图是一个实际的进单曲线。可以看到全天 24 小时内，午晚高峰两个时段单量非常高，而闲时和夜宵相对来说单量又少一些。因此，系统也没办法把一天 24 小时根据每个人的工作时长做平均切分，也需要进行排班。

对于排班，存在两类方案的选型问题。很多业务的排班是基于人的维度，好处是配置的粒度非常精细，每个人的工作时段都是个性化的，可以考虑到每个人的诉求。但

是，在配送场景的缺点也显而易见。如果站长需要为每个人去规划工作时段，其难度可想而知，也很难保证分配的公平性。

	按人排	按组排
优点	配置粒度精细	便于站长管理 轮班制保证公平
缺点	管理难度大 难保证公平性	配置粒度粗

排班方案选型

配送团队最终选用的是按组排班的方式，把所有骑手分成几组，规定每个组的开工时段。然后大家可以按组轮岗，每个人的每个班次都会轮到。

这个问题最大的挑战是，我们并不是在做一项业务工具，而是在设计算法。而算法要有自己的优化目标，那么排班的目标是什么呢？如果你要问站长，怎么样的排班是好的，可能他只会说，要让需要用人的时候有人。但这不是算法语言，更不能变成模型语言。

- 时间离散化
- 人数归一化
- 单量归一化
- 定义运力满足

最大化满足运力需求的时间单元数

$$\sum_{i} \min\left\{ sgn\left(\sum_{j} X_{j,i} \cdot R_j - O_i \right) + 1,1 \right\}$$

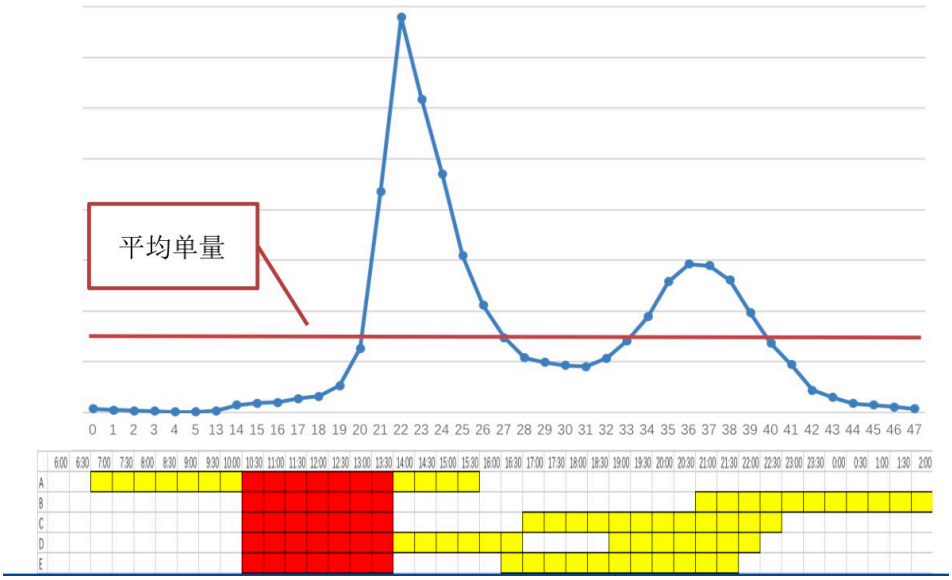
决策变量及目标设计

为了解决这个问题，首先要做设计决策变量，决策变量并没有选用班次的起止时刻和结束时刻，那样做的话，决策空间太大。我们把时间做了离散化，以半小时为粒度。对于一天来讲，只有 48 个时间单元，决策空间大幅缩减。然后，目标定为运力需求满足订单量的时间单元最多。这是因为，并不能保证站点的人数在对应的进单曲线情况下可以满足每个单元的运力需求。所以，我们把业务约束转化为目标函数的一部分。这样做，还有一个好处，那就是没必要知道站点的总人数是多少。

在建模层面，标准化和通用的模型才是最优选。所以，我们把人数做了归一化，算法分配每个班次的骑手比例，但不分人数。最终只需要输入站点的总人数，就得到每个班次的人数。在算法决策的时候，不决策人数、只决策比例，这样也可以把单量进行归一化。每个时间单元的进单量除以每天峰值时间单元的单量，也变成了 0~1 之间的数字。这样就可以认为，如果某个时间单元内人数比例大于单量比例，那么叫作运力得到满足。这样，通过各种归一化，变成了一个通用的问题，而不需要对每种场景单独处理。

另外，这个问题涉及大量复杂的强约束，涉及各种管理的诉求、骑手的体验。约束有很多，比如每个工作时段尽量连续、每个工作时段持续的时间不过短、不同工作时段之间休息的时间不过短等等，有很多这样的业务约束，梳理之后可以发现，这个问题的约束太多了，求最优解甚至可行解的难度太大了。另外，站长在使用排班工具的时候，希望能马上给出系统排班方案，再快速做后续微调，因此对算法运行时间要求也比较高。

算法核心思想



基于约束条件的构造算法与局部搜索

综合考虑以上因素，我们最终基于约束条件，根据启发式算法构造初始方案，再用局部搜索迭代优化。使用这样的方式，求解速度能够达到毫秒级，而且可以给出任意站点的排班方案。整体的优化指标还不错，当然，不保证是最优解，只是可以接受的满意解。

落地应用效果

站点体验指标良好，一线接受度高。

排班时间节省：2h/ 每站点每次

这种算法也在自营场景做了落地应用，跟那些排班经验丰富的站长相比，效果基本持平，一线的接受程度也比较高。最重要的是带来排班时间的节省，每次排班几分钟就搞定了，这样可以让站长有更多的时间去做其它的管理工作。

骑手路径规划

具体到骑手的路径规划问题，不是简单的路线规划，不是从 a 到 b 该走哪条路的问题。这个场景是，一个骑手身上有很多配送任务，这些配送任务存在各种约束，怎样选择最优配送顺序去完成所有任务。这是一个 NP 难问题，当有 5 个订单、10 个任务点的时候，就存在 11 万多条可能的顺序。而在高峰期的时候，骑手往往背负的不止 5 单，甚至有时候一个骑手会同时接到十几单，这时候可行的取送顺序就变成了一个天文数字。

算法应用场景

再看算法的应用场景，这是智能调度系统中最为重要的一个环节。系统派单、系统改派，都依赖路径规划算法。在骑手端，给每个骑手推荐任务执行顺序。另外，用户点了外卖之后，美团会实时展示骑手当前任务还需要执行几分钟，要给用户提供更多预估信息。这么多应用场景，共同的诉求是对时效要求非常高，算法运行时间要越短越好。

但是，算法仅仅是快就可以吗？并不是。因为这是派单、改派这些环节的核心模块，所以算法的优化求解能力也非常重要。如果路径规划算法不能给出较优路径，可想而知，上层的指派和改派很难做出更好的决策。

所以，对这个问题做明确的梳理，核心的诉求是优化效果必须是稳定的好。不能这次的优化结果好，下次就不好。另外，运行时间一定要短。

需求	优化效果稳定	运行时间短
原有问题	随机搜索策略的不确定性	问题结构特征利用不足，迭代搜索效率低
改造方案	启发式定向搜索	基于知识的定制化搜索

核心设计思想

在求解路径规划这类问题上，很多公司的技术团队，都经历过这样的阶段：起初，采用类似遗传算法的迭代搜索算法，但是随着业务的单量变大，发现算法耗时太慢，根本不可接受。然后，改为大规模邻域搜索算法，但算法依然有很强的随机性，因为没有随机性在就没办法得到比较好的解。而这种基于随机迭代的搜索策略，带来很强的不确定性，在问题规模大的场景会出现非常多的 Bad Case。另外，迭代搜索耗时太长了。主要的原因是，随机迭代算法是把组合优化问题当成一个单纯的 Permutation 问题去求解，很少用到问题结构特征。这些算法，求解 TSP 时这样操作，求解 VRP 时也这样操作，求解 Scheduling 还是这样操作，这种类似“无脑”的方式很难有出色的优化效果。

所以，在这个项目中，基本可以确定这样的技术路线。首先，只能做启发式定向搜索，不能在算法中加随机扰动。不能允许同样的输入在不同运行时刻给出不一样的优化结果。然后，不能用普通迭代搜索，必须把这个问题结构特性挖掘出来，做基于知识的定制化搜索。

路径规划问题	Flow-shop Scheduling
订单	Job
取送餐任务	Two Operations
通行时间	Sequence-dependent Setup Time
承诺送达时间	Earliness and Tardiness Criteria

说起来容易，具体要怎么做呢？我们认为，最重要的是看待这个问题的视角。这里的路径规划问题，对应的经典问题模型，是开环 TSP 问题，或是开环 VRP 的变种么？可以是，也可以不是。我们做了一个有意思的建模转换，把它看作流水线调度问题：每个订单可以认为是 job；一个订单的两个任务取餐和送餐，可以认为是一个 job 的 operation。任意两个任务点之间的通行时间，可以认为是序列相关的准备时间。每一单承诺的送达时间，包括预订单和即时单，可以映射到流水线调度问题中的提前和拖期惩罚上。

算法应用效果

做了这样的建模转换之后，流水线调度问题就有大量的启发式算法可以借鉴。我们把一个经典的基于问题特征的启发式算法做了适当适配和改进，可以得到非常好的效果。相比于之前的算法，耗时下降 70%，优化效果不错。因为这是一个确定性算法，所以运行多少次的结果都一样。我们的算法运行一次，跟其它算法运行 10 次的最优结果相比，优化效果是持平的。

订单智能调度

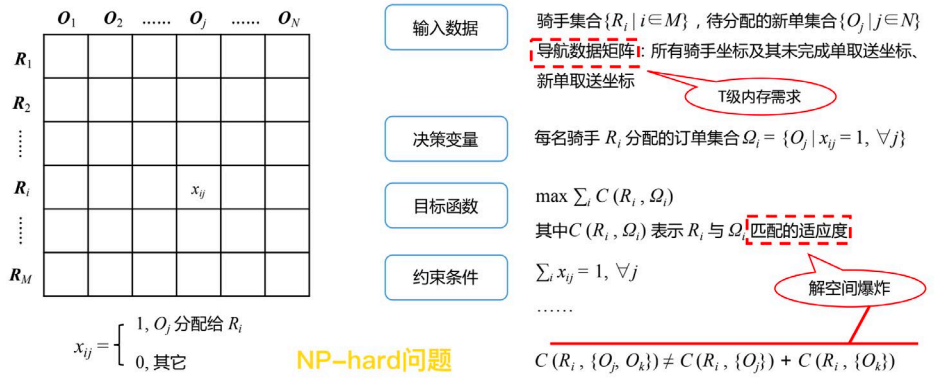
配送调度场景，可以用数学语言描述。它不仅是一个业务问题，更是一个标准的组合优化问题，并且是一个马尔可夫决策过程。

在时间段ΔT内，新订单连续动态到达；对于时刻t，待分配订单集合为$\mathrm{R}(t)=\{r_i|i=1,\ldots,n_t\}$。定义$G(t)=\{V(t),A(t)\}$为$t$时刻问题的图，其中$V(t)$为$t$时刻的位置点集，包含$t$时刻：骑手集合$\{q_j,j=1\ldots m\}$的位置$\mathrm{D}(\mathrm{t})=\{d_j(t),j=1,\ldots,m\}$，$n_t$个订单的取餐位置$P^+(t)=\{1^+,2^+,\ldots,n_t{}^+\}$，送餐位置$P^-(t)=\{1^-,2^-,\ldots,n_t{}^-\}$。$A(t)$为$t$时刻的弧集$A(t)=\{\mathrm{arc(a,b)}|\forall \mathrm{a,b}\in \mathrm{V(t)},\ \mathrm{a}\neq \mathrm{b}\}$，每条弧arc(a,b)对应的距离为$d_{ab}$，行驶时间为$t_{ab}$。点集$\{i^+,i^-\}$表示订单$r_i$需在$i^+$取餐，到$i^-$送餐。$r_i$进单时间为$J_i$，预计出餐时间为$to_i$，预计(承诺)交付时间为$ETA_i$。每个骑手以一定速度在各个取餐点与送餐点以完成订单的运送，其单程运输量与承单量均有限。

对于ΔT内的每个时刻t，如何将$R(t)$分配给骑手并确定每个骑手q_j的节点访问顺序$S_j(\mathrm{t})$，使得ΔT内整体的顾客体验、骑手效率等指标最优。

调度问题的数学描述

并非对于某个时刻的一批订单做最优分配就足够，还需要考虑整个时间窗维度，每一次指派对后面的影响。每一次订单分配，都影响了每个骑手后续时段的位置分布和行进方向。如果骑手的分布和方向不适合未来的订单结构，相当于降低了后续调度时刻最优性的天花板。所以，要考虑长周期的优化，而不是一个静态优化问题。



问题简化分析

为了便于理解，我们还是先看某个调度时刻的静态优化问题。它不仅仅是一个算法问题，还需要我们对工程架构有非常深刻的理解。因为，在对问题输入数据进行拆解的

时候，会发现算法的输入数据太庞大了。比如说，我们需要任意两个任务点的导航距离数据。

而我们面临的问题规模，前几年只是区域维度的调度粒度，一个商圈一分钟峰值 100 多单，匹配几百个骑手，但是这种乘积关系对应的数据已经非常大了。现在，由于美团有更多业务场景，比如跑腿和全城送，是会跨非常多的商圈，甚至跨越半个城市，所以只能做城市级的全局优化匹配。目前，调度系统处理的问题的峰值规模，是 1 万多单和几万名骑手的匹配。而算法允许的运行时间只有几秒钟，同时对内存的消耗也非常大。

另外，配送和网约车派单场景不太一样。打车的调度是做司机和乘客的匹配，本质是个二分图匹配问题，有多项式时间的最优算法：KM 算法。打车场景的难点在于，如何刻画每对匹配的权重。而配送场景还需要解决，对于没有多项式时间最优算法的情况下，如何在指数级的解空间，短时间得到优化解。如果认为每一单和每个骑手的匹配有不同的适应度，那么这个适应度并不是可线性叠加的。也就意味着多单对多人的匹配方案中，任意一种匹配都只能重新运算适应度，其计算量可想而知。

总结一下，这个问题有三类挑战：

1. 性能要求极高，要做到万单对万人的秒级求解。我们之前做了一些比较有意思的工作，比如基于历史最优指派的结果，用机器学习模型做剪枝。基于大量的历史数据，可以帮助我们节省很多无用的匹配方案评价。
2. 动态性。 作为一个 MDP 问题，需要考虑动态优化场景，这涉及大量的预估

环节。在只有当前未完成订单的情况下，骑手如何执行、每一单的完成时刻如何预估、未来时段会进哪些结构的订单、对业务指标和效率指标产生怎样的影响……可能会觉得这是一个典型的强化学习场景，但它的难点在于决策空间太大，甚至可以认为是无限大的。目前我们的思路，是通过其它的建模转换手段进行解决。

3. 配送业务的随机因素多。比如商家的出餐时间，也许是很长时间内都无法解决的随机性。就连历史每一个已完成订单，商家出餐时间的真值都很难获得，因为人为点击的数据并不能保证准确和完整。商家出餐时刻不确定，这个随机因素永远存在，并且非常制约配送效率的提升。另外，在顾客位置交付的时间也不确定。写字楼工作日的午高峰，上电梯、下电梯的时间，很难准确进行预估。当然，我们也在不断努力让预估变得更精准，但随机性永远存在。对于骑手来说，平台没法规定每个骑手的任务执行顺序。骑手在配送过程中可以自由发挥，所以骑手执行顺序的不确定性也一直存在。为了解决这些问题，我们尝试用鲁棒优化或是随机规划的思想。但是，如果基于随机场景采样的方式，运算量又会大幅增加。所以，我们需要进行基于学习的优化，优化不是单纯的机器学习模型，也不是单纯的启发式规则，优化算法是结合真实数据和算法设计者的经验，学习和演进而得。只有这样，才能在性能要求极高的业务场景下，快速的得到鲁棒的优化方案。

目前，美团配送团队的研究方向，不仅包括运筹优化，还包括机器学习、强化学习、数据挖掘等领域。这里具有很多非常有挑战的业务场景，欢迎大家加入我们。

作者简介

圣尧，2017 年加入美团，美团配送团队资深算法专家。

一站式机器学习平台建设实践

作者：艳伟

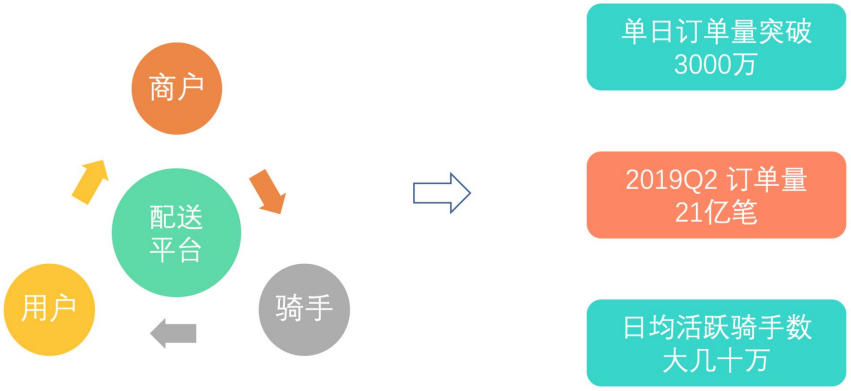
本文根据美团配送资深技术专家郑艳伟在 2019 SACC（中国系统架构师大会）上的演讲内容整理而成，主要介绍了美团配送技术团队在建设一站式机器学习平台过程中的经验总结和探索，希望对从事此领域的同学有所帮助。

0. 写在前面

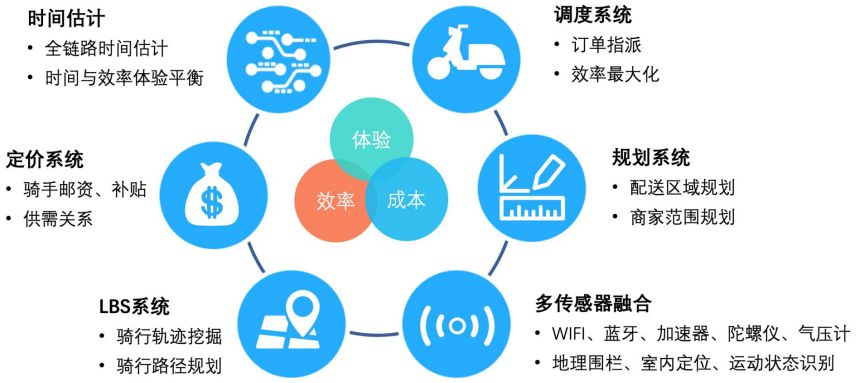
AI 是目前互联网行业炙手可热的“明星”，无论是老牌巨头，还是流量新贵，都在大力研发 AI 技术，为自家的业务赋能。配送作为外卖平台闭环链条上重要的一环，配送效率和用户体验是配送业务的核心竞争力。随着单量上涨、骑手增多、配送场景复杂化，配送场景的各种算法在更快（算法需要快速迭代、快速上线）、更好（业务越来越依赖机器学习算法产生正向的效果）、更准（算法的各种预测如预计送达时间等，需要准确逼近真实值）的目标下也面临日益增大的挑战。算法从调研到最终上线发挥作用，需要有一系列的工程开发和对接，由此引发了新的问题：如何界定算法和工程的边界，各司其职，各善其长？如何提升算法迭代上线的速度和效率？如何快速准确评估算法的效果？本文将为大家分享美团配送技术团队在建设一站式机器学习平台过程中的一些经验和探索，希望对大家能有所帮助或者启发。

1. 业务背景

2019 年 7 月份，美团外卖的日订单量已经突破 3000 万单，占有了相对领先的市场份额。围绕着用户、商户、骑手，美团配送构建了全球领先的即时配送网络，建设了行业领先的美团智能配送系统，形成了全球规模最大的外卖配送平台。



如何让配送网络运行效率更高，用户体验更好，是一项非常有难度的挑战。我们需要解决大量复杂的机器学习和运筹优化等问题，包括 ETA 预测、智能调度、地图优化、动态定价、情景感知、智能运营等多个领域。同时，我们还需要在体验、效率和成本之间达到平衡。



2. 美团配送机器学习平台演进过程

2.1 为什么建设一站式机器学习平台

如果要解决上述的机器学习问题，就需要有一个功能强大且易用的机器学习平台来辅助算法研发人员，帮助大家脱离繁琐的工程化开发，把有限的精力聚焦于算法策略的

迭代上面。

目前业界比较优秀的机器学习平台有很多，既有大公司研发的商用产品，如微软的 Azure、亚马逊的 SageMaker、阿里的 PAI 平台、百度的 PaddlePaddle 以及腾讯的 TI 平台，也有很多开源的产品，如加州大学伯克利分校的 Caffe、Google 的 TensorFlow、Facebook 的 PyTorch 以及 Apache 的 Spark MLlib 等。而开源平台大都是机器学习或者深度学习基础计算框架，聚焦于训练机器学习或深度学习模型；公司的商用产品则是基于基础的机器学习和深度学习计算框架进行二次开发，提供一站式的生态化的服务，为用户提供从数据预处理、模型训练、模型评估、模型在线预测的全流程开发和部署支持，以期降低算法同学的使用门槛。

公司级的一站式机器学习平台的目标和定位，与我们对机器学习平台的需求不谋而合：为用户提供端到端的一站式的服务，帮助他们脱离繁琐的工程化开发，把有限的精力聚焦于算法策略的迭代上面。鉴于此，美团配送的一站式机器学习平台应运而生。

美团配送机器学习平台的演进过程可以分为两个阶段：

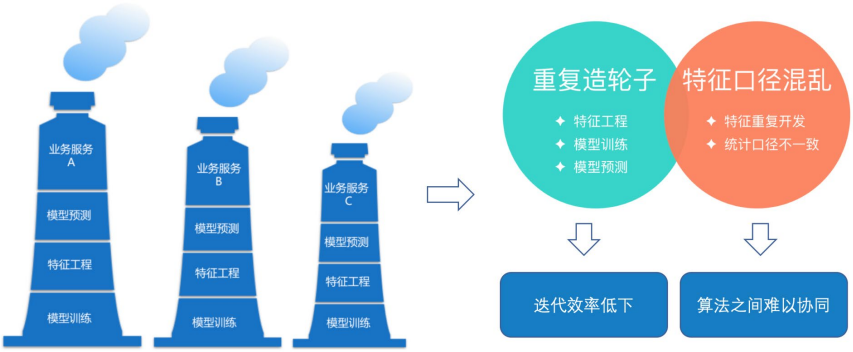
MVP 阶段：灵活，快速试错，具备快速迭代能力。

平台化阶段：业务成指数级增长，需要机器学习算法的场景越来越多，如何既保证业务发展，又能解决系统可用性、扩展性、研发效率等问题。

2.2 MVP 阶段

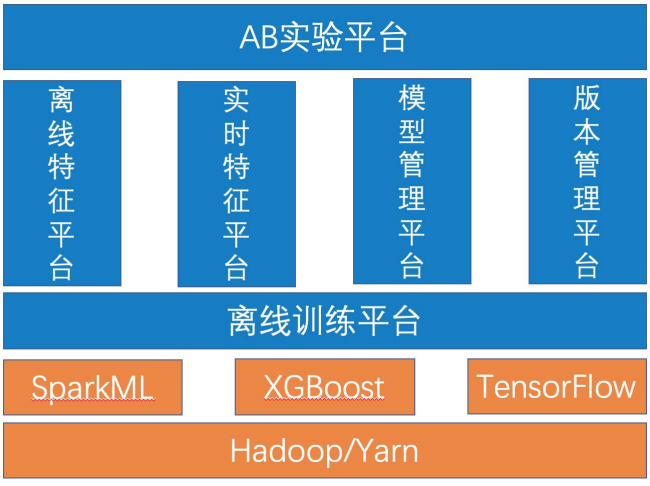
初始阶段，大家对机器学习平台要发展成什么样子并不明确，很多事情也想不清楚。但是为了支撑业务的发展，必须快速上线、快速试错。因此，在此阶段，各个业务线独自建设自己的机器学习工具集，按照各自业务的特殊需求进行各自迭代，快速支持机器学习算法上线落地应用到具体的业务场景，也就是我们所熟知的“烟囱模式”。此种模式各自为战，非常灵活，能够快速支持业务的个性化需求，为业务抢占市场赢得了先机。但随着业务规模的逐渐扩大，这种“烟囱模式”的缺点就凸显了出来，主要表现在以下两个方面：

- **重复造轮子**：特征工程、模型训练、模型在线预测都是各自研发，从零做起，算法的迭代效率低下。
- **特征口径混乱**：各个业务方重复开发特征，相同特征的统计口径也不一致，导致算法之间难以协同工作。



2.3 平台化阶段

为了避免各部门重复造轮子，提升研发的效率，同时统一业务指标和特征的计算口径，标准化配送侧的数据体系，美团配送的研发团队组建了一个算法工程小组，专门规整各业务线的机器学习工具集，希望建设一个统一的机器学习平台，其需求主要包括以下几个方面：



- 该平台底层依托于 Hadoop/Yarn 进行资源调度管理，集成了 Spark ML、XGBoost、TensorFlow 三种机器学习框架，并保留了扩展性，方便接入其它机器学习框架，如美团自研的 MLX（超大规模机器学习平台，专为搜索、推荐、广告等排序问题定制，支持百亿级特征和流式更新）。
- 通过对 Spark ML、XGBoost、TensorFlow 机器学习框架的封装，我们实现了可视化离线训练平台，通过拖拉拽的方式生成 DAG 图，屏蔽多个训练框架的差异，统一模型训练和资源分配，降低了算法 RD 的接入门槛。
- 模型管理平台，提供统一的模型注册、发现、部署、切换、降级等解决方案，并为机器学习和深度学习模型实时计算提供高可用在线预测服务。
- 离线特征平台，收集分拣线下日志，计算提炼成算法所需要的特征，并将线下的特征应用到线上。
- 实时特征平台，实时收集线上数据，计算提炼成算法所需要的特征，并实时推送应用到线上。
- 版本管理平台，管理算法的版本以及算法版本所用的模型、特征和参数。
- AB 实验平台，通过科学的分流和评估方法，更快更好地验证算法的效果。

3. 图灵平台

平台化阶段，我们对美团配送机器学习平台的目标定位是：一站式机器学习平台，给算法同学提供一站式服务，覆盖算法同学调研、开发、上线、评估算法效果的全流程，包括：数据处理、特征生产、样本生成、模型训练、模型评估、模型发布、在线预测和效果评估。为了响应这个目标，大家还给平台取了个大胆的名字——Turing，中文名称为图灵平台，虽然有点“胆大包天”，但是也算是对我们团队的一种鞭策。

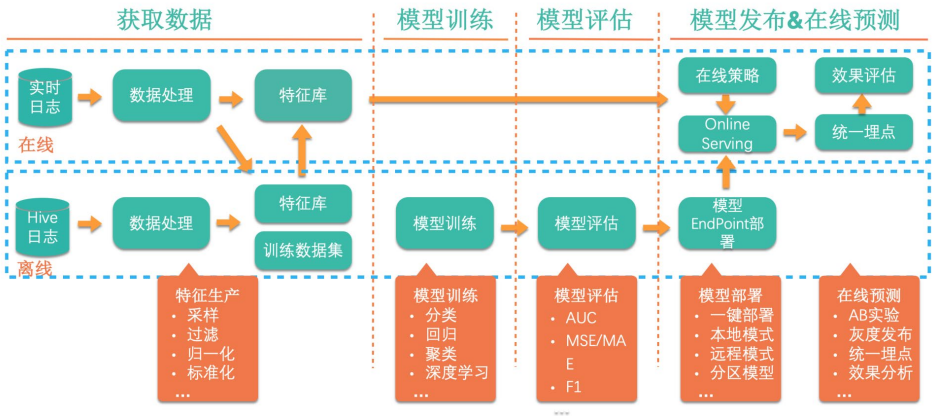
1）首先在获取数据阶段，支持在线和离线两个层面的处理，分别通过采样、过滤、归一化、标准化等手段生产实时和离线特征，并推送到在线的特征库，供线上服务使用。

2）模型训练阶段，支持分类、回归、聚类、深度学习等多种模型，并支持自定义 Loss 损失函数。

3）模型评估阶段，支持多种评估指标，如 AUC、MSE、MAE、F1 等。

4）模型发布阶段，提供一键部署功能，支持本地和远程两种模式，分别对应将模型部署在业务服务本地和部署在专用的在线预测集群。

5）在线预测阶段，支持 AB 实验，灵活的灰度发布放量，并通过统一埋点日志实现 AB 实验效果评估。



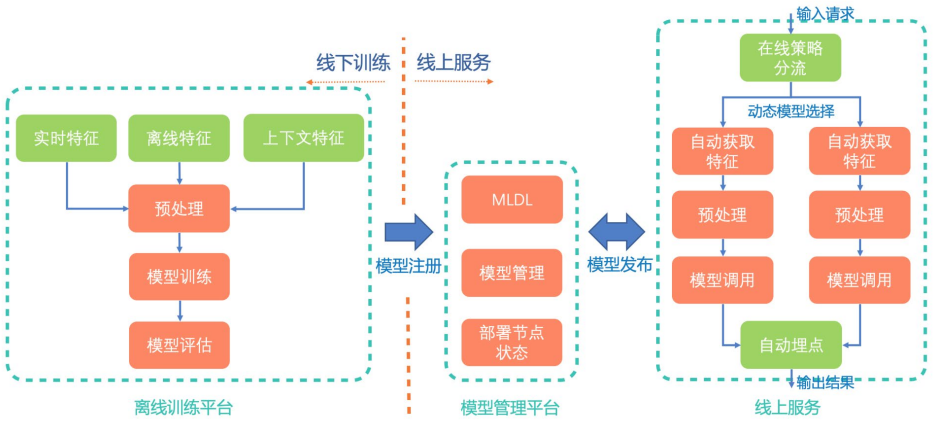
3.1 离线训练平台

离线训练平台的目标是：搭建可视化训练平台，屏蔽多个训练框架的差异，降低算法 RD 的接入门槛。

为了降低算法 RD 进入机器学习领域的门槛，我们开发了带有可视化界面的离线训练平台，通过各种组件的拖拉拽组合成 DAG 图，从而生成一个完整的机器学习训练任务。

目前支持的组件大致分为：输入、输出、特征预处理、数据集加工、机器学习模型、深度学习模型等几大类，每种类别都开发了多个不同的组件，分别支持不同的应用场景。同时为了不失去灵活性，我们也花费了一番心思，提供了多种诸如自定义参数、自动调参、自定义 Loss 函数等功能，尽量满足各个不同业务方向算法同学各种灵活性的需求。

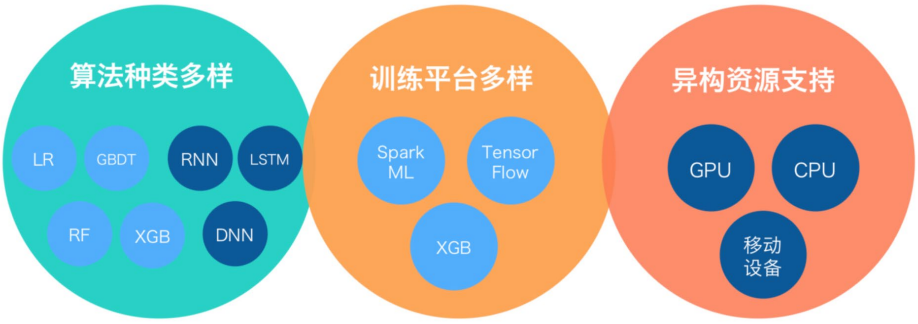
我们的离线训练平台在产出模型时，除了产出模型文件之外，还产出了一个 MLDL（Machine Learning Definition Language）文件，将各模型的所有预处理模块信息写入 MLDL 文件中，与模型保存在同一目录中。当模型发布时，模型文件连带 MLDL 文件作为一个整体共同发布到线上。在线计算时，先自动执行 MLDL 中的预处理逻辑，然后再执行模型计算逻辑。通过 MLDL 打通了离线训练和在线预测，贯穿整个机器学习平台，使得线下和线上使用同一套特征预处理框架代码，保证了线下和线上处理的一致性。



在发布模型时，我们还提供了模型绑定特征功能，支持用户把特征和模型的入参关联起来，方便在线预测时模型自动获取特征，极大地简化了算法 RD 构造模型输入时获取特征的工作量。

3.2 模型管理平台

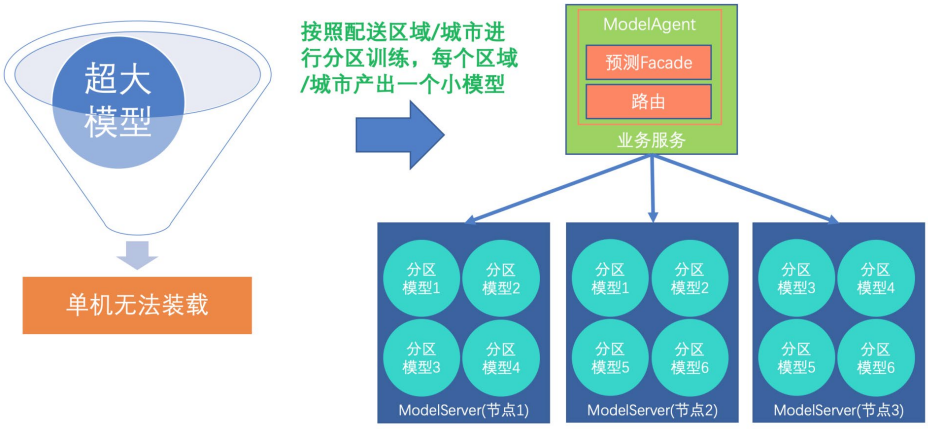
前面介绍了，我们的图灵平台集成了 Spark ML、XGBoost、TensorFlow 三种底层训练框架，基于此，我们的训练平台产出的机器学习模型种类也非常多，简单的有 LR、SVM，树模型有 GBDT、RF、XGB 等，深度学习模型有 RNN、DNN、LSTM、DeepFM 等等。而我们的模型管理平台的目标就是提供统一的模型注册、发现、部署、切换、降级等解决方案，并为机器学习和深度学习模型提供高可用的线上预测服务。



模型管理平台支持本地和远程两种部署模式：

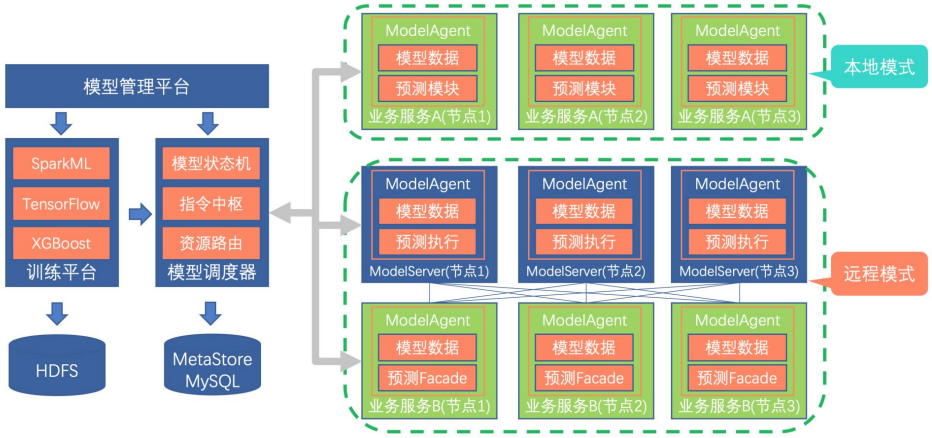
- 本地：模型和 MLDL 统一推送到业务方服务节点上，同时图灵平台提供一个 Java 的 Lib 包，嵌入到业务方应用中，业务方通过本地接口的方式调用模型计算。
- 远程：图灵平台维护了一个专用的在线计算集群，模型和 MLDL 统一部署到在线计算集群中，业务方应用通过 RPC 接口调用在线计算服务进行模型计算。

部署模式	优势	劣势	适用的业务场景
远程模式	· 高度并行化 · 异构计算资源支持	· 额外的网络开销	适合分片存储的超大规模模型
本地模式	· 本地计算性能高 · 无网络开销	· 占用业务方服务器资源 · 单节点执行，并发性能存在瓶颈	适合单节点可以集中存放的小模型



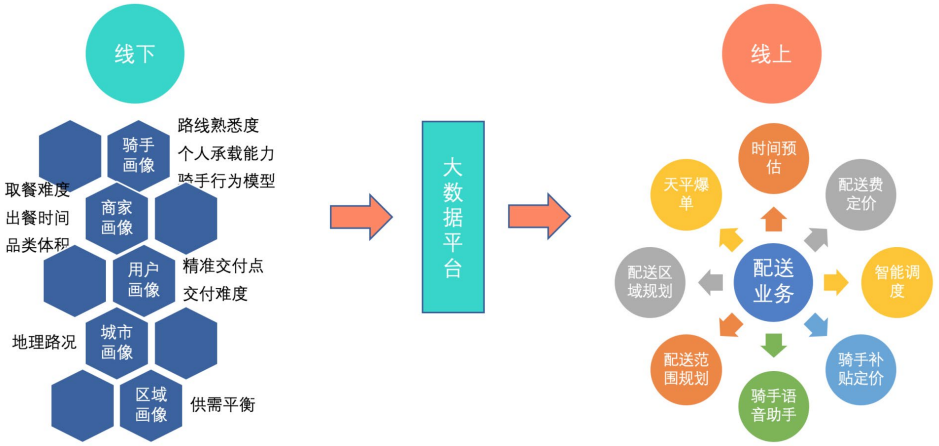
对于超大规模模型，单机无法装载，需要对模型进行 Sharding。鉴于美团配送的业务特性，可以按照配送城市 / 区域进行分区训练，每个城市或区域产出一个小模型，多个分区模型分散部署到多个节点上，解决单节点无法装载大模型的问题。分区模型要求我们必须提供模型的路由功能，以便业务方精准地找到部署相应分区模型的节点。

同时，模型管理平台还收集各个服务节点的心跳上报信息，维护模型的状态和版本切换，确保所有节点上模型版本一致。



3.3 离线特征平台

配送线上业务每天会记录许多骑手、商家、用户等维度的数据，这些数据经过 ETL 处理得到所谓的离线特征，算法同学利用这些离线特征训练模型，并在线上利用这些特征进行模型在线预测。离线特征平台就是将存放在 Hive 表中的离线特征数据生产到线上，对外提供在线获取离线特征的服务能力，支撑配送各个业务高并发及算法快速迭代。

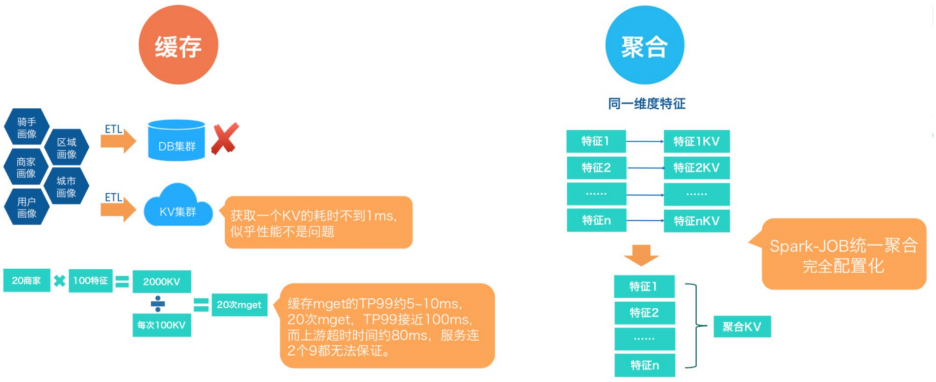


最简单的方案，直接把离线特征存储到 DB 中，线上服务直接读取 DB 获取特征 Value。读取 DB 是个很重的操作，这种方案明显不能满足互联网大并发的场景，直接被 Pass 掉。

第二种方案，把各个离线特征作为 K-V 结构存储到 Redis 中，线上服务直接根据特征 Key 读取 Redis 获取特征 Value。此方案利用了 Redis 内存 K-V 数据库的高性能，乍一看去，好像可以满足业务的需求，但实际使用时，也存在着严重的性能问题。

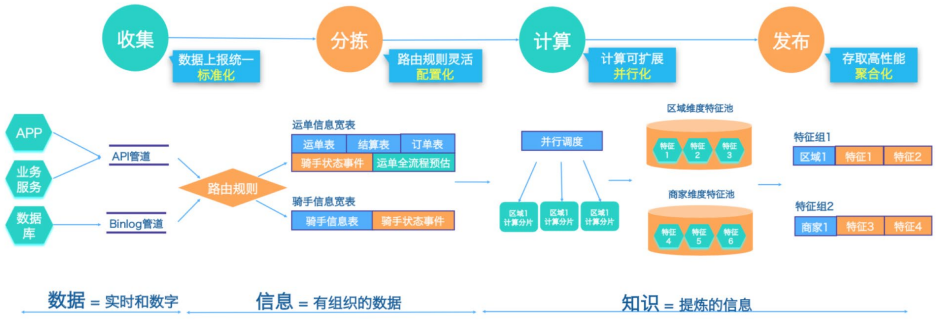
典型的业务场景：比如我们要预测 20 个商家的配送时长，假设每个商家需要 100 个特征，则我们就需要 20*100=2000 个特征进行模型计算，2000 个 KV。如果直接单个获取，满足不了业务方的性能需求；如果使用 Redis 提供的批量接口 Mget，如果每次获取 100 个 KV，则需要 20 次 Mget。缓存 mget 的耗时 TP99 约 5ms，20 次 Mget，TP99 接近 100ms，也无法满足业务方的性能需求（上游服务超时时间约 50ms）。

因此，我们需要对离线特征从存储和获取进行优化。我们提出了特征组的概念，同一维度的特征，按照特征组的结构进行聚合成一个 KV，大大减少了 Key 的数目；并且提供了相对完善的管理功能，支持对特征组的动态调整（组装、拆分等）。



3.4 实时特征平台

相比于传统配送，即时配送无论是在位置信息、骑手负载，还是在当前路网情况，以及商家出餐情况等方面都是瞬息变化的，实时性要求非常高。为了让机器学习算法能够即时的在线上生效，我们需要实时地收集线上各种业务数据，进行计算，提炼成算法所需要的特征，并实时更新。



3.5 AB 实验平台

AB 实验并不是个新兴的概念，自 2000 年谷歌工程师将这一方法应用在互联网产品以来，AB 实验在国内外越来越普及，已成为互联网产品运营精细度的重要体现。简单来说，AB 实验在产品优化中的应用方法是：在产品正式迭代发版之前，为同一个目标制定两个（或以上）方案，将用户流量对应分成几组，在保证每组用户特征相同的前提下，让用户分别看到不同的方案设计，根据几组用户的真实数据反馈，科学的

帮助产品进行决策。

互联网领域常见的 AB 实验，大多是面向 C 端用户进行流量选择，比如基于注册用户的 UID 或者用户的设备标识（移动用户 IMEI 号 /PC 用户 Cookie）进行随机或者哈希计算后分流。此类方案广泛应用于搜索、推荐、广告等领域，体现出千人千面个性化的特点。此类方案的特点是实现简单，假设请求独立同分布，流量之间独立决策，互不干扰。此类 AB 实验之所以能够这样做是因为：C 端流量比较大，样本足够多，而且不同用户之间没有相互干扰，只要分流时足够随机，即基本可以保证请求独立同分布。

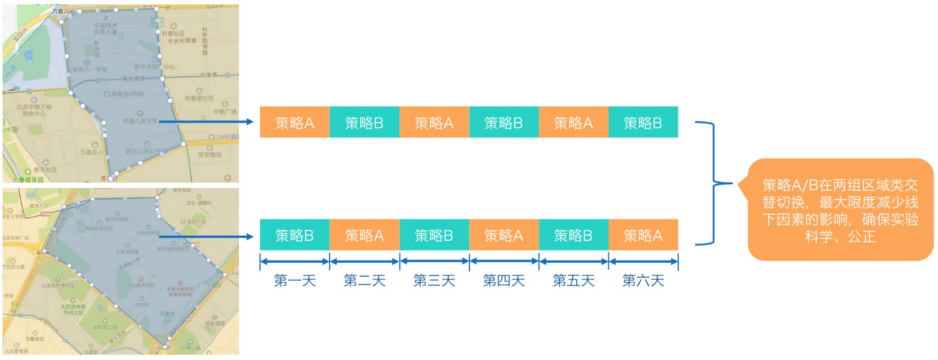
即时配送领域的 AB 实验是围绕用户、商户、骑手三者进行，用户 / 商户 / 骑手之间不再是相互独立的，而是相互影响相互制约的。针对此类场景，现有的分流方案会造成不同策略的互相干扰，无法有效地评估各个流量各个策略的优劣。

鉴于上述的问题，我们将配送侧的 AB 实验分为三个阶段：事前的 AA 分组，事中的 AB 分流，事后的效果评估。

- AA 分组：将候选流量按照既定的规则预先分为对照组和实验组，基于数理统计的理论确保对照组和实验组在所关注的业务指标上没有显著差异。
- AB 分流：将线上请求实时分到对照或者实验版本。

- 效果评估：根据对照组和实验组的数据对比评估 AB 实验的效果。

由于即时配送的场景较为特殊，比如按照配送区域或城市进行 AB 实验时，由于样本空间有限，很难找到没有差异的对照组和实验组，因此我们设计了一种分时间片 AB 对照的分流方法：支持按天、小时、分钟进行分片，多个时间片进行轮转切换，在不同区域、不同时间片之间，对不同的策略进行交替切换进行 AB 分流，最大限度减少线下因素的影响，确保实验科学公正。



4 总结与展望

目前图灵平台支撑了美团配送、小象、LBS 平台等 BU 的算法离线训练、在线预测、AB 实验等，使算法 RD 更加关注算法策略本身的迭代优化，显著提高了算法 RD 的效率。未来我们会在以下方面继续深入探索：

1）加强深度学习的建设。

- 加强深度学习的建设，全面支持深度学习，实现深度学习相关组件与机器学习组件一样，在可视化界面可以和任意组件组合使用。
- 离线训练支持更多常用深度学习模型。
- 支持直接写 Python 代码自定义深度学习模型。

2）在线预测平台化，进一步解耦算法和工程。

- 简化图灵平台 SDK，剥离主体计算逻辑，建设在线预测平台。
- 在线预测平台动态加载算法包，实现算法、业务工程方、图灵平台的解耦。

作者简介

艳伟，美团配送技术团队资深技术专家。

招聘信息

如果你想近距离感受一下图灵平台的魅力，欢迎加入我们。美团配送技术团队诚招调度履约方向、LBS 方向、机器学习平台、算法工程方向的技术专家和架构师，共建全行业最大的单一即时配送网络和平台，共同面对复杂业务和高并发流量的挑战，迎接配送业务全面智能化的时代。感兴趣同学可投递简历至：tech@meituan.com（邮件标题注明：美团配送技术团队）。

美团搜索中 NER 技术的探索与实践

作者：丽红　星池　燕华　马璐　廖群　志安　刘亮　李超　云森　永超等

1. 背景

命名实体识别（Named Entity Recognition，简称 NER），又称作“专名识别”，是指识别文本中具有特定意义的实体，主要包括人名、地名、机构名、专有名词等。NER 是信息提取、问答系统、句法分析、机器翻译、面向 Semantic Web 的元数据标注等应用领域的重要基础工具，在自然语言处理技术走向实用化的过程中占有重要的地位。在美团搜索场景下，NER 是深度查询理解（Deep Query Understanding，简称 DQU）的底层基础信号，主要应用于搜索召回、用户意图识别、实体链接等环节，NER 信号的质量，直接影响到用户的搜索体验。

下面将简述一下实体识别在搜索召回中的应用。在 O2O 搜索中，对商家 POI 的描述是商家名称、地址、品类等多个互相之间相关性并不高的文本域。如果对 O2O 搜索引擎也采用全部文本域命中求交的方式，就可能会产生大量的误召回。我们的解决方法如下图 1 所示，让特定的查询只在特定的文本域做倒排检索，我们称之为“结构化召回”，可保证召回商家的强相关性。举例来说，对于“海底捞”这样的请求，有些商家地址会描述为“海底捞附近几百米”，若采用全文本域检索这些商家就会被召回，显然这并不是用户想要的。而结构化召回基于 NER 将“海底捞”识别为商家，然后只在商家名相关文本域检索，从而只召回海底捞品牌商家，精准地满足了用户需求。

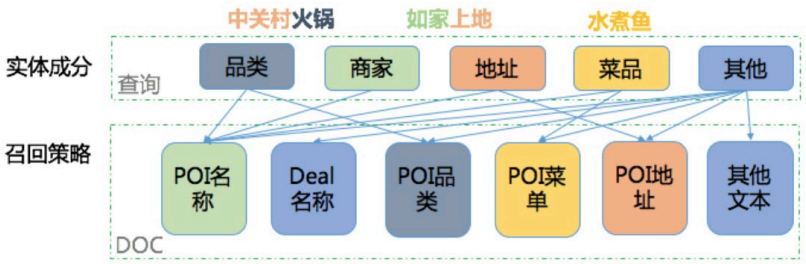


图 1　实体识别与召回策略

有别于其他应用场景，美团搜索的 NER 任务具有以下特点：

- **新增实体数量庞大且增速较快**：本地生活服务领域发展迅速，新店、新商品、新服务品类层出不穷；用户 Query 往往夹杂很多非标准化表达、简称和热词（如“牵肠挂肚”、“吸猫”等），这对实现高准确率、高覆盖率的 NER 造成了很大挑战。
- **领域相关性强**：搜索中的实体识别与业务供给高度相关，除通用语义外需加入业务相关知识辅助判断，比如“剪了个头发”，通用理解是泛化描述实体，在搜索中却是个商家实体。
- **性能要求高**：从用户发起搜索到最终结果呈现给用户时间很短，NER 作为 DQU 的基础模块，需要在毫秒级的时间内完成。近期，很多基于深度网络的研究与实践显著提高了 NER 的效果，但这些模型往往计算量较大、预测耗时长，如何优化模型性能，使之能满足 NER 对计算时间的要求，也是 NER 实践中的一大挑战。

2. 技术选型

针对 O2O 领域 NER 任务的特点，我们整体的技术选型是“实体词典匹配 + 模型预测”的框架，如图下 2 所示。实体词典匹配和模型预测两者解决的问题各有侧重，在当前阶段缺一不可。下面通过对三个问题的解答来说明我们为什么这么选。

为什么需要实体词典匹配?

答：主要有以下四个原因：

一是搜索中用户查询的头部流量通常较短、表达形式简单，且集中在商户、品类、地址等三类实体搜索，实体词典匹配虽简单但处理这类查询准确率也可达到 90% 以上。

二是 NER 领域相关，通过挖掘业务数据资源获取业务实体词典，经过在线词典匹配后可保证识别结果是领域适配的。

三是新业务接入更加灵活，只需提供业务相关的实体词表就可完成新业务场景下的实

体识别。

四是 NER 下游使用方中有些对响应时间要求极高，词典匹配速度快，基本不存在性能问题。

有了实体词典匹配为什么还要模型预测?

答：有以下两方面的原因：

一是随着搜索体量的不断增大，中长尾搜索流量表述复杂，越来越多 OOV（Out Of Vocabulary）问题开始出现，实体词典已经无法满足日益多样化的用户需求，模型预测具备泛化能力，可作为词典匹配的有效补充。

二是实体词典匹配无法解决歧义问题，比如“黄鹤楼美食”，“黄鹤楼”在实体词典中同时是武汉的景点、北京的商家、香烟产品，词典匹配不具备消歧能力，这三种类型都会输出，而模型预测则可结合上下文，不会输出“黄鹤楼”是香烟产品。

实体词典匹配、模型预测两路结果是怎么合并输出的?

答：目前我们采用训练好的 CRF 权重网络作为打分器，来对实体词典匹配、模型预测两路输出的 NER 路径进行打分。在词典匹配无结果或是其路径打分值明显低于模型预测时，采用模型识别的结果，其他情况仍然采用词典匹配结果。

在介绍完我们的技术选型后，接下来会展开介绍下我们在实体词典匹配、模型在线预测等两方面的工作，希望能为大家在 O2O NER 领域的探索提供一些帮助。

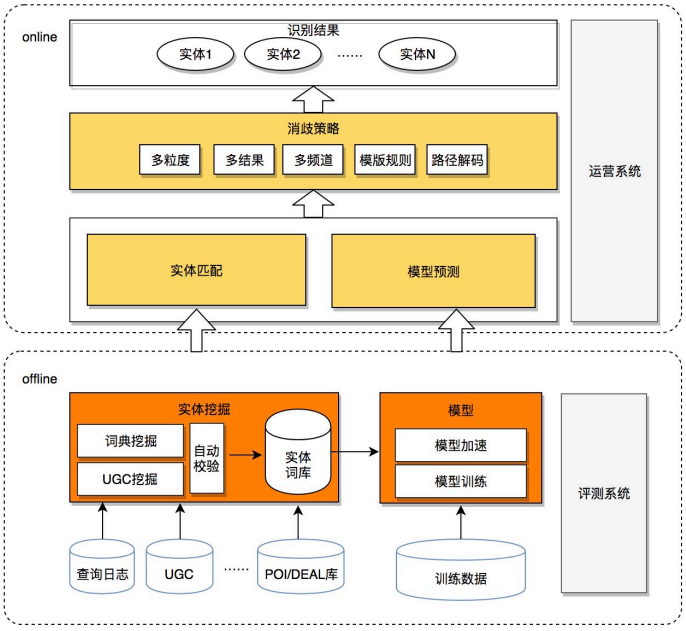


图 2　实体识别整体架构

3. 实体词典匹配

传统的 NER 技术仅能处理通用领域既定、既有的实体，但无法应对垂直领域所特有的实体类型。在美团搜索场景下，通过对 POI 结构化信息、商户评论数据、搜索日志等独有数据进行离线挖掘，可以很好地解决领域实体识别问题。经过离线实体库不断的丰富完善累积后，在线使用轻量级的词库匹配实体识别方式简单、高效、可控，且可以很好地覆盖头部和腰部流量。目前，基于实体库的在线 NER 识别率可以达到 92%。

3.1 离线挖掘

美团具有丰富多样的结构化数据，通过对领域内结构化数据的加工处理可以获得高精度的初始实体库。例如：从商户基础信息中，可以获取商户名、类目、地址、售卖商品或服务等类型实体。从猫眼文娱数据中，可以获取电影、电视剧、艺人等类型实体。然而，用户搜索的实体名往往夹杂很多非标准化表达，与业务定义的标准实体名之间存在差异，如何从非标准表达中挖掘领域实体变得尤为重要。

现有的新词挖掘技术主要分为无监督学习、有监督学习和远程监督学习。无监督学习通过频繁序列产生候选集，并通过计算紧密度和自由度指标进行筛选，这种方法虽然可以产生充分的候选集合，但仅通过特征阈值过滤无法有效地平衡精确率与召回率，现实应用中通常挑选较高的阈值保证精度而牺牲召回。先进的新词挖掘算法大多为有监督学习，这类算法通常涉及复杂的语法分析模型或深度网络模型，且依赖领域专家设计繁多规则或大量的人工标记数据。远程监督学习通过开源知识库生成少量的标记数据，虽然一定程度上缓解了人力标注成本高的问题。然而小样本量的标记数据仅能学习简单的统计模型，无法训练具有高泛化能力的复杂模型。

我们的离线实体挖掘是多源多方法的，涉及到的数据源包括结构化的商家信息库、百科词条，半结构化的搜索日志，以及非结构化的用户评论（UGC）等。使用的挖掘方法也包含多种，包括规则、传统机器学习模型、深度学习模型等。UGC 作为一种非结构化文本，蕴含了大量非标准表达实体名。下面我们将详细介绍一种针对 UGC 的垂直领域新词自动挖掘方法，该方法主要包含三个步骤，如下图 3 所示：

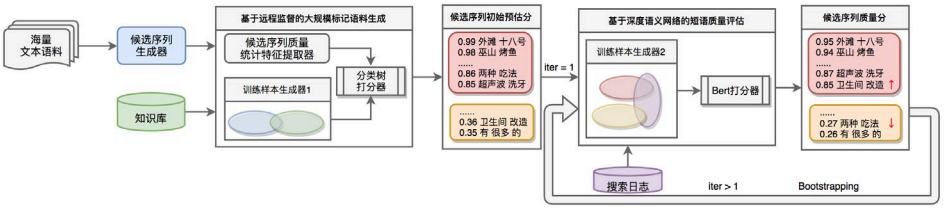


图 3　一种适用于垂直领域的新词自动挖掘方法

Step1：候选序列挖掘。频繁连续出现的词序列，是潜在新型词汇的有效候选，我们采用频繁序列产生充足候选集合。

Step2：基于远程监督的大规模有标记语料生成。频繁序列随着给定语料的变化而改变，因此人工标记成本极高。我们利用领域已有累积的实体词典作为远程监督词库，将 Step1 中候选序列与实体词典的交集作为训练正例样本。同时，通过对候选序列分析发现，在上百万的频繁 Ngram 中仅约 10% 左右的候选是真正的高质新型词汇。因此，对于负例样本，采用负采样方式生产训练负例集 [1]。针对海量 UGC 语料，我

们设计并定义了四个维度的统计特征来衡量候选短语可用性：

- **频率**：有意义的新词在语料中应当满足一定的频率，该指标由 Step1 计算得到。
- **紧密度**：主要用于评估新短语中连续元素的共现强度，包括 T 分布检验、皮尔森卡方检验、逐点互信息、似然比等指标。
- **信息度**：新发现词汇应具有真实意义，指代某个新的实体或概念，该特征主要考虑了词组在语料中的逆文档频率、词性分布以及停用词分布。
- **完整性**：新发现词汇应当在给定的上下文环境中作为整体解释存在，因此应同时考虑词组的子集短语以及超集短语的紧密度，从而衡量词组的完整性。

在经过小样本标记数据构建和多维度统计特征提取后，训练二元分类器来计算候选短语预估质量。由于训练数据负例样本采用了负采样的方式，这部分数据中混合了少量高质量的短语，为了减少负例噪声对短语预估质量分的影响，可以通过集成多个弱分类器的方式减少误差。对候选序列集合进行模型预测后，将得分超过一定阈值的集合作为正例池，较低分数的集合作为负例池。

Step3: 基于深度语义网络的短语质量评估。在有大量标记数据的情况下，深度网络模型可以自动有效地学习语料特征，并产出具有泛化能力的高效模型。BERT 通过海量自然语言文本和深度模型学习文本语义表征，并经过简单微调在多个自然语言理解任务上刷新了记录，因此我们基于 BERT 训练短语质量打分器。为了更好地提升训练数据的质量，我们利用搜索日志数据对 Step2 中生成的大规模正负例池数据进行远程指导，将有大量搜索记录的词条作为有意义的关键词。我们将正例池与搜索日志重合的部分作为模型正样本，而将负例池减去搜索日志集合的部分作为模型负样本，进而提升训练数据的可靠性和多样性。此外，我们采用 Bootstrapping 方式，在初次得到短语质量分后，重新根据已有短语质量分以及远程语料搜索日志更新训练样本，迭代训练提升短语质量打分器效果，有效减少了伪正例和伪负例。

在 UGC 语料中抽取出大量新词或短语后，参考 AutoNER[2] 对新挖掘词语进行类型预测，从而扩充离线的实体库。

3.2 在线匹配

原始的在线 NER 词典匹配方法直接针对 Query 做双向最大匹配，从而获得成分识别候选集合，再基于词频（这里指实体搜索量）筛选输出最终结果。这种策略比较简陋，对词库准确度和覆盖度要求极高，所以存在以下几个问题：

- 当 Query 包含词库未覆盖实体时，基于字符的最大匹配算法易引起切分错误。例如，搜索词“海坨山谷”，词库仅能匹配到“海坨山”，因此出现“海坨山 / 谷”的错误切分。
- 粒度不可控。例如，搜索词“星巴克咖啡”的切分结果，取决于词库对“星巴克”、“咖啡”以及“星巴克咖啡”的覆盖。
- 节点权重定义不合理。例如，直接基于实体搜索量作为实体节点权重，当用户搜索“信阳菜馆”时，“信阳菜 / 馆”的得分大于“信阳 / 菜馆”。

为了解决以上问题，在进行实体字典匹配前引入了 CRF 分词模型，针对垂直领域美团搜索制定分词准则，人工标注训练语料并训练 CRF 分词模型。同时，针对模型分词错误问题，设计两阶段修复方式：

1. 结合模型分词 Term 和基于领域字典匹配 Term，根据动态规划求解 Term 序列权重和的最优解。
2. 基于 Pattern 正则表达式的强修复规则。最后，输出基于实体库匹配的成分识别结果。

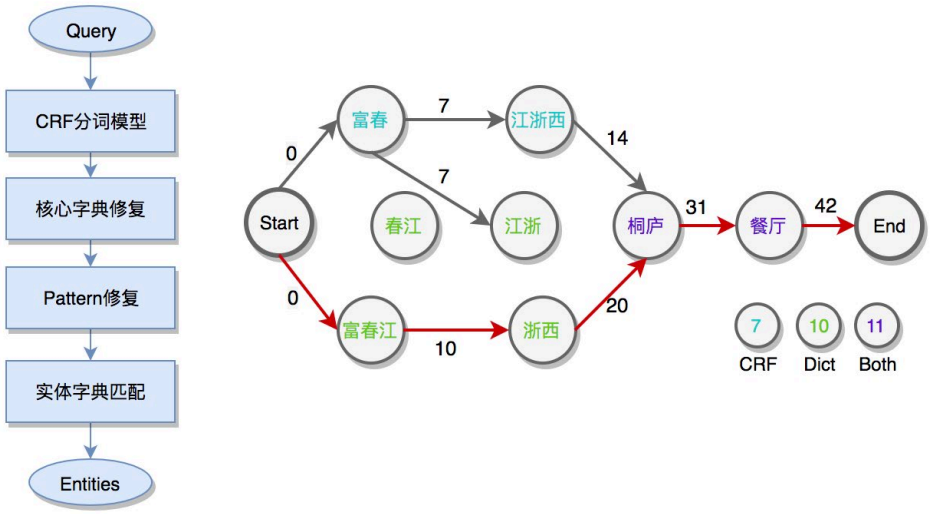


图 4　实体在线匹配

4. 模型在线预测

对于长尾、未登录查询，我们使用模型进行在线识别。 NER 模型的演进经历了如下图 5 所示的几个阶段，目前线上使用的主模型是 BERT[3] 以及 BERT+LR 级联模型，另外还有一些在探索中模型的离线效果也证实有效，后续我们会综合考虑性能和收益逐步进行上线。搜索中 NER 线上模型的构建主要面临三个问题：

1. 性能要求高：NER 作为基础模块，模型预测需要在毫秒级时间内完成，而目前基于深度学习的模型都有计算量大、预测时间较长的问题。
2. 领域强相关：搜索中的实体类型与业务供给高度相关，只考虑通用语义很难保证模型识别的准确性。
3. 标注数据缺乏：NER 标注任务相对较难，需给出实体边界切分、实体类型信息，标注过程费时费力，大规模标注数据难以获取。

针对性能要求高的问题，我们的线上模型在升级为 BERT 时进行了一系列的性能调优；针对 NER 领域相关问题，我们提出了融合搜索日志特征、实体词典信息的知识增强 NER 方法；针对训练数据难以获取的问题，我们提出一种弱监督的 NER 方法。

下面我们详细介绍下这些技术点。

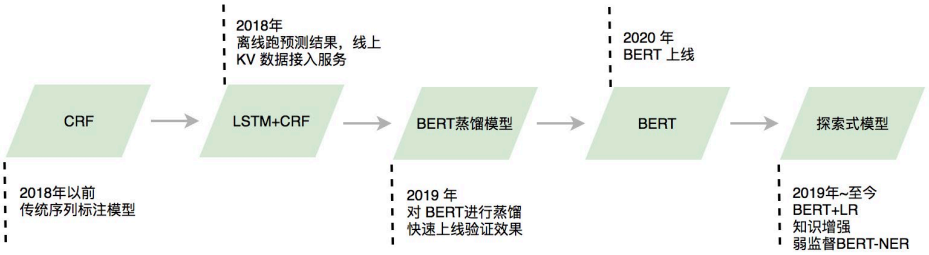


图 5　NER 模型演进

4.1 BERT 模型

BERT 是谷歌于 2018 年 10 月公开的一种自然语言处理方法。该方法一经发布，就引起了学术界以及工业界的广泛关注。在效果方面，BERT 刷新了 11 个 NLP 任务的当前最优效果，该方法也被评为 2018 年 NLP 的重大进展以及 NAACL 2019 的 best paper[4,5]。BERT 和早前 OpenAI 发布的 GPT 方法技术路线基本一致，只是在技术细节上存在略微差异。两个工作的主要贡献在于使用预训练 + 微调的思路来解决自然语言处理问题。以 BERT 为例，模型应用包括 2 个环节：

- 预训练（Pre-training），该环节在大量通用语料上学习网络参数，通用语料包括 Wikipedia、Book Corpus，这些语料包含了大量的文本，能够提供丰富的语言相关现象。
- 微调（Fine-tuning），该环节使用“任务相关”的标注数据对网络参数进行微调，不需要再为目标任务设计 Task-specific 网络从头训练。

将 BERT 应用于实体识别线上预测时面临一个挑战，即预测速度慢。我们从模型蒸馏、预测加速两个方面进行了探索，分阶段上线了 BERT 蒸馏模型、BERT+Softmax、BERT+CRF 模型。

4.1.1 模型蒸馏

我们尝试了对 BERT 模型进行剪裁和蒸馏两种方式，结果证明，剪裁对于 NER 这种复杂 NLP 任务精度损失严重，而模型蒸馏是可行的。模型蒸馏是用简单模型来逼近复杂模型的输出，目的是降低预测所需的计算量，同时保证预测效果。Hinton 在 2015 年的论文中阐述了核心思想 [6]，复杂模型一般称作 Teacher Model，蒸馏后的简单模型一般称作 Student Model。Hinton 的蒸馏方法使用伪标注数据的概率分布来训练 Student Model，而没有使用伪标注数据的标签来训练。作者的观点是概率分布相比标签能够提供更多信息以及更强约束，能够更好地保证 Student Model 与 Teacher Model 的预测效果达到一致。在 2018 年 NeurIPS 的 Workshop 上，[7] 提出一种新的网络结构 BlendCNN 来逼近 GPT 的预测效果，本质上也是模型蒸馏。BlendCNN 预测速度相对原始 GPT 提升了 300 倍，另外在特定任务上，预测准确率还略有提升。关于模型蒸馏，基本可以得到以下结论：

- **模型蒸馏本质是函数逼近**。针对具体任务，笔者认为只要 Student Model 的复杂度能够满足问题的复杂度，那么 Student Model 可以与 Teacher Model 完全不同，选择 Student Model 的示例如下图 6 所示。举个例子，假设问题中的样本（x，y）从多项式函数中抽样得到，最高指数次数 d=2；可用的 Teacher Model 使用了更高指数次数（比如 d=5），此时，要选择一个 Student Model 来进行预测，Student Model 的模型复杂度不能低于问题本身的复杂度，即对应的指数次数至少达到 d=2。
- **根据无标注数据的规模，蒸馏使用的约束可以不同**。如图 7 所示，如果无标注数据规模小，可以采用值（logits）近似进行学习，施加强约束；如果无标注数据规模中等，可以采用分布近似；如果无标注数据规模很大，可以采用标签近似进行学习，即只使用 Teacher Model 的预测标签来指导模型学习。

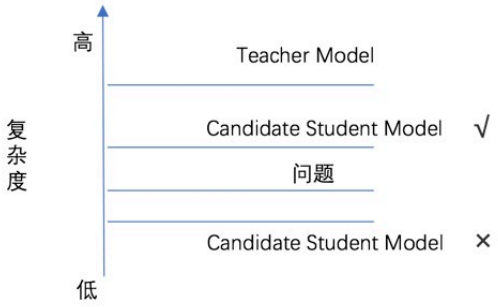


图6 Student Model复杂度选择方法

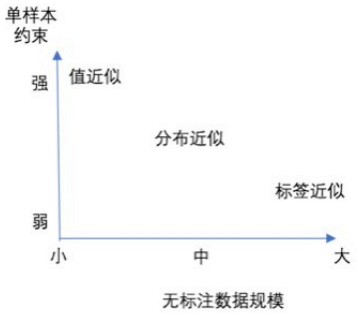


图7 标注数据量级与蒸馏约束关系

有了上面的结论，我们如何在搜索 NER 任务中应用模型蒸馏呢？ 首先先分析一下该任务。与文献中的相关任务相比，搜索 NER 存在有一个显著不同：作为线上应用，搜索有大量无标注数据。用户查询可以达到千万 / 天的量级，数据规模上远超一些离线测评能够提供的数据。据此，我们对蒸馏过程进行简化：不限制 Student Model 的形式，选择主流的推断速度快的神经网络模型对 BERT 进行近似；训练不使用值近似、分布近似作为学习目标，直接使用标签近似作为目标来指导 Student Model 的学习。

我们使用 IDCNN-CRF 来近似 BERT 实体识别模型，IDCNN（Iterated Dilated CNN）是一种多层 CNN 网络，其中低层卷积使用普通卷积操作，通过滑动窗口圈定的位置进行加权求和得到卷积结果，此时滑动窗口圈定的各个位置的距离间隔等于 1。高层卷积使用膨胀卷积（Atrous Convolution）操作，滑动窗口圈定的各个位置的距离间隔等于 d（d>1）。通过在高层使用膨胀卷积可以减少卷积计算量，同时在序列依赖计算上也不会有损失。在文本挖掘中，IDCNN 常用于对 LSTM 进行替换。实验结果表明，相较于原始 BERT 模型，在没有明显精度损失的前提下，蒸馏模型的在线预测速度有数十倍的提升。

4.1.2 预测加速

BERT 中大量小算子以及 Attention 计算量的问题，使得其在实际线上应用时，预测时长较高。我们主要使用以下三种方法加速模型预测，同时对于搜索日志中的高频

Query，我们将预测结果以词典方式上传到缓存，进一步减少模型在线预测的 QPS 压力。下面介绍下模型预测加速的三种方法：

算子融合：通过降低 Kernel Launch 次数和提高小算子访存效率来减少 BERT 中小算子的耗时开销。我们这里调研了 Faster Transformer 的实现。平均时延上，有 1.4x~2x 左右加速比；TP999 上，有 2.1x~3x 左右的加速比。该方法适合标准的 BERT 模型。开源版本的 Faster Transformer 工程质量较低，易用性和稳定性上存在较多问题，无法直接应用，我们基于 NV 开源的 Faster Transformer 进行了二次开发，主要在稳定性和易用性进行了改进：

- 易用性：支持自动转换，支持 Dynamic Batch，支持 Auto Tuning。
- 稳定性：修复内存泄漏和线程安全问题。

Batching：Batching 的原理主要是将多次请求合并到一个 Batch 进行推理，降低 Kernel Launch 次数、充分利用多个 GPU SM，从而提高整体吞吐。在 max_batch_size 设置为 4 的情况下，原生 BERT 模型，可以在将平均 Latency 控制在 6ms 以内，最高吞吐可达 1300 QPS。该方法十分适合美团搜索场景下的 BERT 模型优化，原因是搜索有明显的高低峰期，可提升高峰期模型的吞吐量。

混合精度：混合精度指的是 FP32 和 FP16 混合的方式，使用混合精度可以加速 BERT 训练和预测过程并且减少显存开销，同时兼顾 FP32 的稳定性和 FP16 的速度。在模型计算过程中使用 FP16 加速计算过程，模型训练过程中权重会存储成 FP32 格式，参数更新时采用 FP32 类型。利用 FP32 Master-weights 在 FP32 数据类型下进行参数更新，可有效避免溢出。混合精度在基本不影响效果的基础上，模型训练和预测速度都有一定的提升。

4.2 知识增强的 NER

如何将特定领域的外部知识作为辅助信息嵌入到语言模型中，一直是近些年的研究热点。K-BERT[8]、ERNIE[9] 等模型探索了知识图谱与 BERT 的结合方法，为我们提供了很好的借鉴。美团搜索中的 NER 是领域相关的，实体类型的判定与业务供给高

度相关。因此，我们也探索了如何将供给 POI 信息、用户点击、领域实体词库等外部知识融入到 NER 模型中。

4.2.1 融合搜索日志特征的 Lattice-LSTM

在 O2O 垂直搜索领域，大量的实体由商家自定义（如商家名、团单名等），实体信息隐藏在供给 POI 的属性中，单使用传统的语义方式识别效果差。Lattice-LSTM[10] 针对中文实体识别，通过增加词向量的输入，丰富语义信息。我们借鉴这个思路，结合搜索用户行为，挖掘 Query 中潜在短语，这些短语蕴含了 POI 属性信息，然后将这些隐藏的信息嵌入到模型中，在一定程度上解决领域新词发现问题。与原始 Lattice-LSTM 方法对比，识别准确率千分位提升 5 个点。

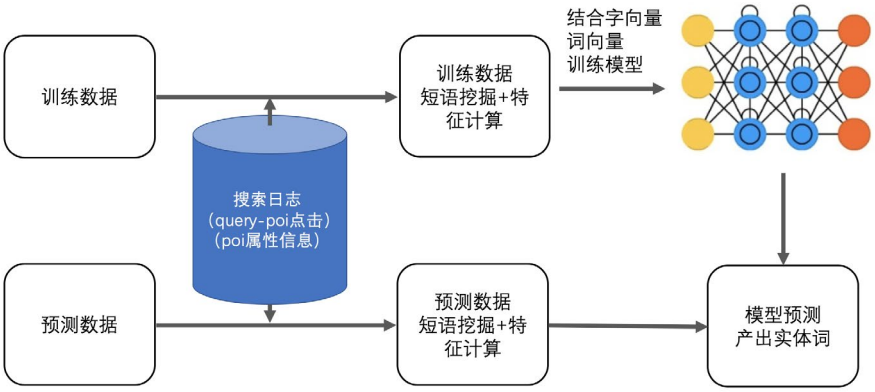


图 8　融合搜索日志特征的 Lattice-LSTM 构建流程

1) 短语挖掘及特征计算

该过程主要包括两步：匹配位置计算、短语生成，下面详细展开介绍。

图 9　短语挖掘及特征计算

Step1：匹配位置计算。对搜索日志进行处理，重点计算查询与文档字段的详细匹配情况以及计算文档权重（比如点击率）。如图 9 所示，用户输入查询是“手工编织”，对于文档 d1（搜索中就是 POI），“手工”出现在字段“团单”，“编织”出现在字段“地址”。对于文档 2，“手工编织”同时出现在“商家名”和“团单”。匹配开始位置、匹配结束位置分别对应有匹配的查询子串的开始位置以及结束位置。

Step2：短语生成。以 Step1 的结果作为输入，使用模型推断候选短语。可以使用多个模型，从而生成满足多个假设的结果。我们将候选短语生成建模为整数线性规划（Integer Linear Programmingm，ILP）问题，并且定义了一个优化框架，模型中的超参数可以根据业务需求进行定制计算，从而获得满足不用假设的结果。对于一个具体查询 Q，每种切分结果都可以使用整数变量 xij 来表示：xij=1 表示查询 i 到 j 的位置构成短语，即 Qij 是一个短语，xij=0 表示查询 i 到 j 的位置不构成短语。优化目标可以形式化为：在给定不同切分 xij 的情况下，使收集到的匹配得分最大化。优化目标及约束函数如图 10 所示，其中 p：文档，f：字段，w：文档 p 的权重，wf：字段 f 的权重。xijpf：查询子串 Qij 是否出现在文档 p 的 f 字段，且最终切分方案会考虑该观测证据，Score(xijpf)：最终切分方案考虑的观测得分，w(xij)：切分 Qij 对应的权重，yijpf：观测到的匹配，查询子串 Qij 出现在文档 p 的 f 字段中。χmax：查询包含的最大短语数。这里，χmax、wp、wf 、w(xij) 是超参数，在求解 ILP 问题前需要完成设置，这些变量可以根据不同假设进行设置：可以根据经验人工设置，另外也可以基于其他信号来设置，设置可参考图 10 给出的方法。最终短语的特征向量表征为在 POI 各属性字段的点击分布。

$$\max \mathbf{Score}(x_{ijpf}) = \sum_{p,f \in S} w_p w_f \sum_{i=1}^{N}\sum_{j=i}^{N} x_{ijpf} w(x_{ij})$$

subject to

$$\sum_{i=1}^{k}\sum_{j=k}^{N} x_{ij} = 1 \qquad \forall k = 1, \ldots, N$$

$$x_{ij} - x_{ijpf} \geq 0 \qquad 1 \leq i \leq j \leq N \ \text{ if } y_{mnpf} = 1$$

$$\sum_{i=m}^{n}\sum_{j=i}^{n} x_{ijpf} \leq 1 \qquad \text{if } y_{mnpf} = 1$$

$$\sum_{i=1}^{N}\sum_{j=i}^{N} x_{ij} \leq \chi_{max}$$

$$x_{ij} \in \{0,1\} \qquad 1 \leq i \leq j \leq N$$

$$\begin{cases} x_{ijpf} \in \{0,1\} & m \leq i \leq j \leq n \text{ if } y_{mnpf} = 1 \\ x_{ijpf} = 0 & \text{elsewise} \end{cases}$$

变量	具体设置方法
w_p	点击率
	文档质量
w_f	根据查询意图确定字段权重
	字段质量
$w(x_{ij})$	子串长度的函数，α^{j-i}
	α<1，答案倾向于选择短的子串
	α>1，答案倾向于选择长的子串

图 10　短语生成问题抽象以及参数设置方法

2) 模型结构

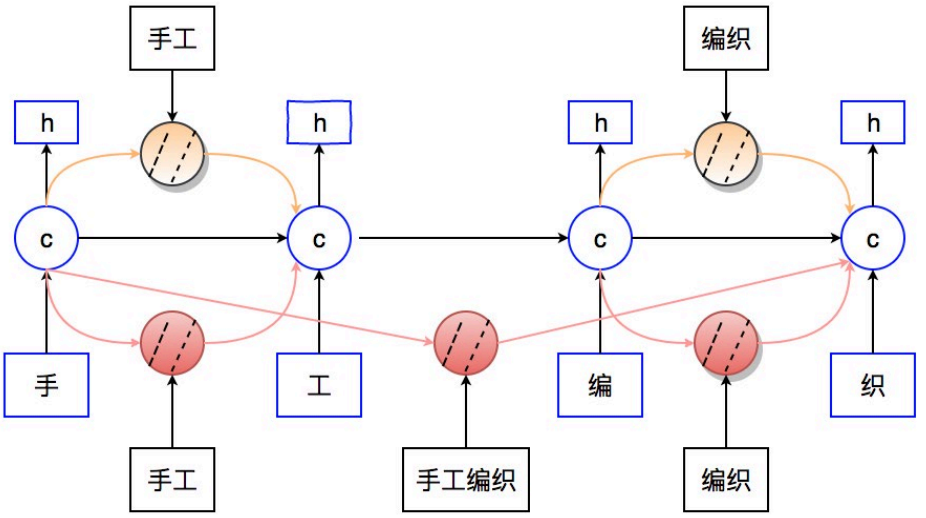


图 11　融合搜索日志特征的 Lattice-LSTM 模型结构

模型结构如图 11 所示，蓝色部分表示一层标准的 LSTM 网络（可以单独训练，也可以与其他模型组合），输入为字向量，橙色部分表示当前查询中所有词向量，红色部分表示当前查询中的通过 Step1 计算得到的所有短语向量。对于 LSTM 的隐状

态输入，主要由两个层面的特征组成：当前文本语义特征，包括当前字向量输入和前一时刻字向量隐层输出；潜在的实体知识特征，包括当前字的短语特征和词特征。下面介绍当前时刻潜在知识特征的计算以及特征组合的方法：(下列公式中，σ 表示 sigmoid 函数，⊙表示矩阵乘法)

潜在实体知识特征：通过LSTM单元的特性，将实体（词+短语）特征向量与前文语义特征向量相结合。结合方式如公式1所示，其中 $\mathbf{x}_{b,e}^{phrase}$ 表示短语特征向量，$\mathbf{i}_{b,e}^{phrase}$，$\mathbf{f}_{b,e}^{phrase}$ 分别表示输入门和遗忘门，$\mathbf{c}_b^c$ 表示词起始字符的隐层输出，例如 匹配短语“编织”，$\mathbf{c}_b^c$ 表示的为“编”对应的隐层输出向量。

$$\begin{bmatrix} \mathbf{i}_{b,e}^{phrase} \\ \mathbf{f}_{b,e}^{phrase} \\ \hat{\mathbf{c}}_{b,e}^{phrase} \end{bmatrix} = \begin{bmatrix} \sigma \\ \sigma \\ \sigma \end{bmatrix} \left(\mathbf{W}^{phrase\intercal} \begin{bmatrix} \mathbf{x}_{b,e}^{phrase} \\ \mathbf{h}_b^c \end{bmatrix} + \mathbf{b}^{phrase} \right)$$

$$\mathbf{c}_{b,e}^{phrase} = \mathbf{f}_{b,e}^{phrase} \odot \mathbf{c}_b^c + \mathbf{i}_{b,e}^{phrase} \odot \hat{\mathbf{c}}_{b,e}^{phrase}$$

公式 1 当前时刻潜在实体知识特征计算

特征组合：在隐状态输出时，对当前的字特征向量、词特征向量与短语特征向量加权求和，权重由模型学习得到。在"手工编织"这个例子中，在“织”输入的时刻，字向量输入为“织”=$\mathbf{x}_3^c$，词向量输入为“编织”=$\mathbf{c}_{2,3}^w$，短语向量输入为“手工编织”=$\mathbf{c}_{0,3}^{phrase}$，“编织”=$\mathbf{c}_{2,3}^{phrase}$，当前时刻的状态为 $\mathbf{c}_3^c = \alpha_{w,c}^{2,3} \odot \mathbf{c}_{2,3}^w + \alpha_{phrase,c}^{0,3} \odot \mathbf{c}_{0,3}^{phrase} + \alpha_{phrase,c}^{2,3} \odot \mathbf{c}_{2,3}^{phrase} + \alpha_3^c \odot \hat{\mathbf{c}}_3^c$ 。

4.2.2 融合实体词典的两阶段 NER

我们考虑将领域词典知识融合到模型中，提出了两阶段的 NER 识别方法。该方法是将 NER 任务拆分成实体边界识别和实体标签识别两个子任务。相较于传统的端到端的 NER 方法，这种方法的优势是实体切分可以跨领域复用。另外，在实体标签识别阶段可以充分使用已积累的实体数据和实体链接等技术提高标签识别准确率，缺点是会存在错误传播的问题。

在第一阶段，让 BERT 模型专注于实体边界的确定，而第二阶段将实体词典带来的信息增益融入到实体分类模型中。第二阶段的实体分类可以单独对每个实体进行预测，但这种做法会丢失实体上下文信息，我们的处理方法是：将实体词典用作训练数据训练一个 IDCNN 分类模型，该模型对第一阶段输出的切分结果进行编码，并将编码信息加入到第二阶段的标签识别模型中，联合上下文词汇完成解码。基于 Benchmark 标注数据进行评估，该模型相比于 BERT-NER 在 Query 粒度的准确率上获得了 1% 的提升。这里我们使用 IDCNN 主要是考虑到模型性能问题，大家可视使用场景替换成 BERT 或其他分类模型。

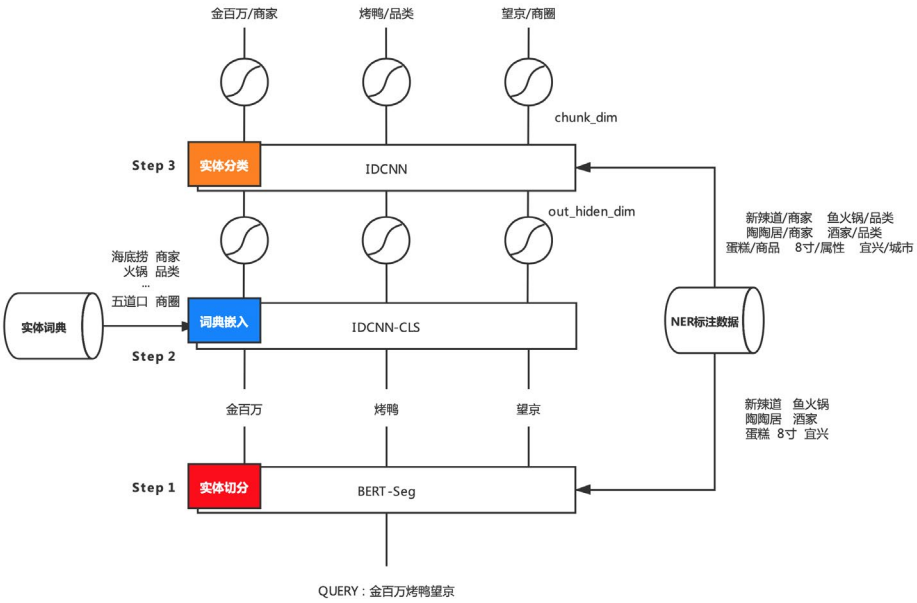


图 12　融合实体词典的两阶段 NER

4.3 弱监督 NER

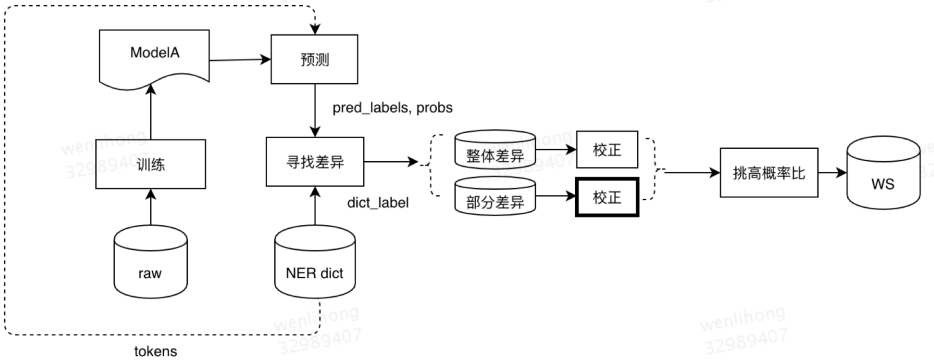


图 13　弱监督标注数据生成流程

针对标注数据难获取问题，我们提出了一种弱监督方案，该方案包含两个流程，分别是弱监督标注数据生成、模型训练。下面详细描述下这两个流程。

Step1：弱监督标注样本生成

1) 初版模型：利用已标注的小批量数据集训练实体识别模型，这里使用的是最新的 BERT 模型，得到初版模型 ModelA。

2) 词典数据预测：实体识别模块目前沉淀下百万量级的高质量实体数据作为词典，数据格式为实体文本、实体类型、属性信息。用上一步得到的 ModelA 预测改词典数据输出实体识别结果。

3) 预测结果校正：实体词典中实体精度较高，理论上来讲模型预测的结果给出的实体类型至少有一个应该是实体词典中给出的该实体类型，否则说明模型对于这类输入的识别效果并不好，需要针对性地补充样本，我们对这类输入的模型结果进行校正后得到标注文本。校正方法我们尝试了两种，分别是整体校正和部分校正，整体校正是指整个输入校正为词典实体类型，部分校正是指对模型切分出的单个 Term 进行类型校正。举个例子来说明，“兄弟烧烤个性 diy”词典中给出的实体类型为商家，模型预测结果为修饰词 + 菜品 + 品类，没有 Term 属于商家类型，模型预测结果和词典有差

异，这时候我们需要对模型输出标签进行校正。校正候选就是三种，分别是“商家 + 菜品 + 品类”、“修饰词 + 商家 + 品类”、“修饰词 + 菜品 + 商家”。我们选择最接近于模型预测的一种，这样选择的理论意义在于模型已经收敛到预测分布最接近于真实分布，我们只需要在预测分布上进行微调，而不是大幅度改变这个分布。那从校正候选中如何选出最接近于模型预测的一种呢？我们使用的方法是计算校正候选在该模型下的概率得分，然后与模型当前预测结果（当前模型认为的最优结果）计算概率比，概率比计算公式如公式 2 所示，概率比最大的那个就是最终得到的校正候选，也就是最终得到的弱监督标注样本。在“兄弟烧烤个性 diy”这个例子中，“商家 + 菜品 + 品类”这个校正候选与模型输出的“修饰词 + 菜品 + 品类”概率比最大，将得到“兄弟 / 商家 烧烤 / 菜品 个性 diy/ 品类”标注数据。

图 14　标签校正

$$dist(A,B)=\frac{e^{\frac{\sum_{i\in A} pi}{N_A}}}{e^{\frac{\sum_{i\in B} pi}{N_B}}}$$ ，pi为第 i 个 term 的模型输出概率；N_A为切分中 term 个数

公式 2　概率比计算

Step2：弱监督模型训练

弱监督模型训练方法包括两种：一是将生成的弱监督样本和标注样本进行混合不区分重新进行模型训练；二是在标注样本训练生成的 ModelA 基础上，用弱监督样本进行 Fine-tuning 训练。这两种方式我们都进行了尝试。从实验结果来看，Fine-tuning 效果更好。

5. 总结和展望

本文介绍了 O2O 搜索场景下 NER 任务的特点及技术选型，详述了在实体词典匹配和模型构建方面的探索与实践。

实体词典匹配针对线上头腰部流量，离线对 POI 结构化信息、商户评论数据、搜索日志等独有数据进行挖掘，可以很好的解决领域实体识别问题，在这一部分我们介绍了一种适用于垂直领域的新词自动挖掘方法。除此之外，我们也积累了其他可处理多源数据的挖掘技术，如有需要可以进行约线下进行技术交流。

模型方面，我们围绕搜索中 NER 模型的构建的三个核心问题（性能要求高、领域强相关、标注数据缺乏）进行了探索。针对性能要求高采用了模型蒸馏，预测加速的方法，使得 NER 线上主模型顺利升级为效果更好的 BERT。在解决领域相关问题上，分别提出了融合搜索日志、实体词典领域知识的方法，实验结果表明这两种方法可一定程度提升预测准确率。针对标注数据难获取问题，我们提出了一种弱监督方案，一定程度缓解了标注数据少模型预测效果差的问题。

未来，我们会在解决 NER 未登录识别、歧义多义、领域相关问题上继续深入研究，欢迎业界同行一起交流。

6. 参考资料

[1] Automated Phrase Mining from Massive Text Corpora. 2018.
[2] Learning Named Entity Tagger using Domain-Specific Dictionary. 2018.
[3] Bidirectional Encoder Representations from Transformers. 2018
[4] https://www.jiqizhixin.com/articles/2018-12-30
[5] https://naacl2019.org/blog/best-papers/
[6] Hinton et al. Distilling the Knowledge in a Neural Network. 2015.
[7] Yew Ken Chia et al.Transformer to CNN: Label-scarce distillation for efficient text classification. 2018.
[8] K-BERT: Enabling Language Representation with Knowledge Graph. 2019.
[9] Enhanced Language Representation with Informative Entities. 2019.
[10] Chinese NER Using Lattice LSTM. 2018.

7. 作者简介

丽红，星池，燕华，马璐，廖群，志安，刘亮，李超，张弓，云森，永超等，均来自美团搜索与 NLP 部。

招聘信息

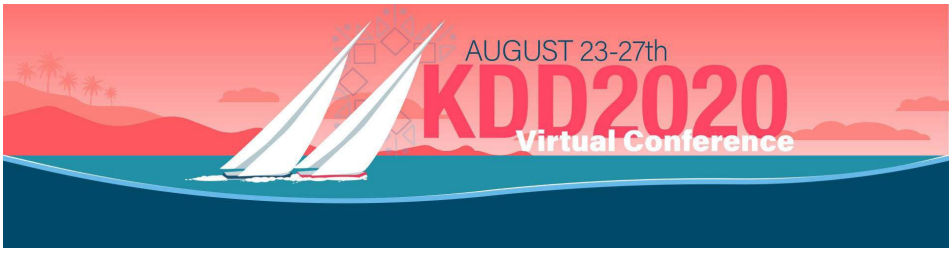
美团搜索部，长期招聘搜索、推荐、NLP 算法工程师，坐标北京。欢迎感兴趣的同学发送简历至：tech@meituan.com（邮件标题注明：搜索与 NLP 部）

KDD Cup 2020 Debiasing 比赛冠军技术方案及在美团的实践

作者：坚强　明健　胡可　曲檀　雷军

背景

ACM SIGKDD（国际数据挖掘与知识发现大会，简称 KDD）是数据挖掘领域的国际顶级会议。KDD Cup 比赛是由 SIGKDD 主办的数据挖掘研究领域的国际顶级赛事，从 1997 年开始，每年举办一次，是目前数据挖掘领域最具影响力的赛事。该比赛同时面向企业界和学术界，云集了世界数据挖掘界的顶尖专家、学者、工程师、学生等参加，为数据挖掘从业者们提供了一个学术交流和研究成果展示的平台。KDD Cup 2020 共设置五道赛题（四个赛道），分别涉及数据偏差问题（Debiasing）、多模态召回问题（Multimodalities Recall）、自动化图学习（AutoGraph）、对抗学习问题和强化学习问题。



美团到店广告平台搜索广告算法团队基于自身的业务场景，一直在不断进行前沿技术的深入优化与算法创新，团队在数据偏差、多模态学习、图学习三个前沿领域均有一定的算法研究与应用，并取得了不错的业务结果。基于这三个领域的技术积累，我们在比赛中选择了三道紧密联系的赛题，希望应用并提升这三个领域技术积累，带来技术与业务的进一步突破。搜索广告算法团队的黄坚强、胡可、漆毅、曲檀、陈明健、郑博航、雷军与中科院大学唐兴元共同组建参赛队伍 Aister，参加了 Debiasing、AutoGraph、Multimodalities Recall 三道赛题，最终在 Debiasing 赛道中获得冠

军（1/1895），在 AutoGraph 赛道中也获得了冠军（1/149），并在 Multimodalities Recall 赛道中获得了季军（3/1433）。

在广告系统中，如何对数据偏差进行消除是最具挑战性的问题之一，也是近年来学术界的研究热点。随着产品形态与算法技术的持续演进，系统会不断积累偏差。搜索广告算法团队在数据偏差问题取得了突破，带来了较显著的业务效果提升。特别是在 Debiasing 赛题中，团队基于偏差消除问题的技术积累，从全球 1895 支队伍的激烈角逐中取得第 1 名，并在最终评测指标（ndcg_half）领先第 2 名 6.0%。下面我们将介绍 Debiasing 赛题的技术方案，以及团队在广告业务中偏差消除的应用与研究，希望对从事相关研究的同学能够有所帮助或者启发。

技术方案开源代码

KDD Cup 2020 Challenges for Modern E-Commerce Platform: Debiasing

Rank	Team	Members
1	aister	Jianqiang Huang, Ke Hu, Mingjian Chen, Bohang Zheng, Xingyuan Tang, Tan Qu, Yi Qi, Jun Lei
2	DeepWisdom	Jin Zhou, Taicheng Guo, Binhao Wu, Chengxuan Ying, Ruirui Guo, Youcheng Xiong, Jinlin Wang, Chenglin Wu
3	TheAvengers	Runxing Zhong, Ziwen Ye, Rui Li, Jin Wei, Yuanfei Luo, Xiufeng Shu, Hengxing Cai
4	hello dog	Zihao Zhao, Yafei Yao, Kui Ma, Zeyu Qiu, Xiangdong Wu, Yong Li, Changping Peng, Yongjun Bao
5	ECNU@KDDCUP20	Wen Wang, Wenwei Liang, Wei Zhang
6	Challengers	Shunyao Wu, Chuanyu Xue, Shouhua Liu, Mingyuan Cheng, Chengbin Huang, Shihuan Liu, Xin Chen, Qing Li, Huan Liu, Chuwen Huang
7	MOH	Shumeng Liu, Pei Shen, Xinchun Ming, Yewen Wang, Yinxiang Zhang, He Wang
8	云儿是真名	Yuner Xuan
9	DB	Zhipeng Luo, Jin Wang, Jiangdong Zhang, Yatao Yang
10	Rush	Taofeng Xue, Jialong Zou, Xinzhou Dong, Jieyang Zhuang, Pengfei Zhao, Hongyu Zhou

图 2　KDD Cup 2020 Debiasing 比赛 TOP 10 榜单

赛题介绍与问题分析

偏差消除问题概述

大多数电子商务和零售公司利用海量数据在其网站上实现搜索和推荐系统，从而来促进销售，随着这样的趋势发展以及流量的大量增加，对推荐系统产生了各式各样的挑战。其中一个值得探索的挑战是推荐系统的人工智能公平性（Fairness）问题 [1,2]，即如果机器学习系统配备了短期目标（例如短期的点击、交易），单纯朝短期目标进行优化将会导致严重的“马太效应”，即热门的商品受到更多的关注，冷门商品则愈发的会被遗忘，产生了系统中的流行度偏差 [3]，并且大多数模型和系统的迭代依赖于页面浏览（Pageview）数据，而曝光数据是实际候选中经过模型选择的一个子集，不断地依赖模型选择的数据与反馈再进行训练，将形成选择性偏差 [3]。上述流行度偏差与选择性偏差不断积累，就会导致系统中的“马太效应”越来越严重。因此，人工智能公平性问题对于推荐系统的不断优化至关重要，并且这将对推荐系统的发展以及生态环境产生深远的影响。

由于不是一个定义充分的优化问题，偏差消除是当前推荐系统非常具有挑战性的问题，也是当前学术界的一个研究热点。本次 KDD 的赛题也是围绕偏差问题展开，基于电子商务中用户下一次点击商品预测（Next-Item Prediction）的问题，进行无偏估计。

赛题官方提供了用户点击数据、商品多模态数据、用户特征数据。其中用户点击数据提供了用户历史点击的商品以及点击的时间戳，商品多模态数据主要为商品的文本向量以及图片向量，用户特征数据有用户的年龄、性别、城市等等。数据涉及超过 100 万次点击，10 万商品和 3 万用户。并根据时间窗口划分数据阶段，一共分为十个阶段，最终评分以最后 3 个阶段为准。

为了关注于消除偏差问题，本次赛题提供的评测指标包括 NDCG@50_full，NDCG@50_half，hitrate@50_full，hitrate@50_half。采用 NDCG@50_full，NDCG@50_half 两项指标进行评估。

- NDCG@50_full：与常规推荐系统评价指标 NDCG 一致，在整个评测数据集上评估了每次用户请求所推荐的前 50 个商品列表的平均排序效果，该评测集我们称之为 full 评测集。
- NDCG@50_half：关注于偏差问题，从整个 full 评测数据集中取出一半历史曝光少的点击商品，对这些商品的推荐列表进行 NDCG 指标评估，该评测集我们称之为 half 评测集。

评分首先通过 NDCG@50_full 筛选出前 10% 的队伍，然后在这些队伍中使用 NDCG@50_half 来进行最终排名。在最终的评估中 NDCG@50_half 将对 Top 名次的差异，在长尾数据预测更重要的评测方式能够更好地评估选手们对于数据偏差的优化。不同于传统的封闭数据集点击率预估问题（CTR 预估），上述数据特点与评测方式侧重于偏差的优化。

数据分析与问题理解

数据分析与问题：用户特征数据中一共有 35444 个用户，但只有 6789 个用户有特征，故而特征覆盖率只有 19.15%，由于覆盖率较低且只有年龄、性别、城市等三个特征，我们发现这些特征对我们的整个任务而言是无用的。商品特征数据中一共有 117720 个商品，有 108916 个商品拥有文本向量及图片向量，覆盖率高达 92.52%，可以根据向量去计算商品间的文本相似度及图片相似度，由于用户信息及商品信息的缺少，如何利用好这些商品多模态向量对于整个任务而言是极其重要的。

选择性偏差分析：如表 1 所示，我们对基于 i2i（item2item）点击共现以及基于 i2i 向量相似度两种 Item-Based 协同过滤的方法所召回的商品候选集做对比，由于系统的性能限制，我们将候选集长度最大值限制到 1000，我们发现两种召回方法在评测集上都有一个较低的 hitrate，则不管使用哪种方法系统都存在着一个较大的选择性偏差，即推荐给用户的样本是根据系统来选择的，而不是所有候选集合，真实的候选集合大大超过了推荐给用户的样本，导致训练数据带有选择性偏差。进一步的，我们发现基于 i2i 点击共现在 full 评测集上相对于 half 评测集有更高的 hitrate，说明其更偏好于流行商品，而基于 i2i 向量相似度在 full 和 half 的评测集上 hitrate 相差不大，说

明其对于流行度无偏好，同时两种方式召回的候选集只有 4% 的重复率，故而我们需要去结合点击共现和向量相似度两种商品关系来生成更大的训练集，从而缓解选择性偏差。

method	hitrate@50_full	hitrate@50_half	hitrate@1000_full	hitrate@1000_half
i2i点击共现	0.1814	0.1384	0.2752	0.2090
i2i向量相似度	0.1218	0.1238	0.1887	0.1879

表 1　i2i 点击共现与 i2i 向量相似度的召回 hitrate

如图 3 所示，我们对商品的流行度进行了分析，其中横坐标商品点击频数，即商品流行度，纵坐标为商品个数。图中我们对流行度做了截断，横坐标最大值本应为 228。可以看出，大部分商品的流行度较低，符合长尾分布。图中的两个箱型图分别是 full 评测数据集商品流行度的分布，以及 half 评测数据集商品流行度的分布。从这两个箱型图可以看出，流行度偏差存在于数据集中，整个 full 评测集中有一半评测数据是基于流行度较低的商品，而另一半评测数据商品的流行度较高，直接通过点击商品去构建样本，会导致数据中拥有较多流行度高的正例商品，从而形成流行度偏差。

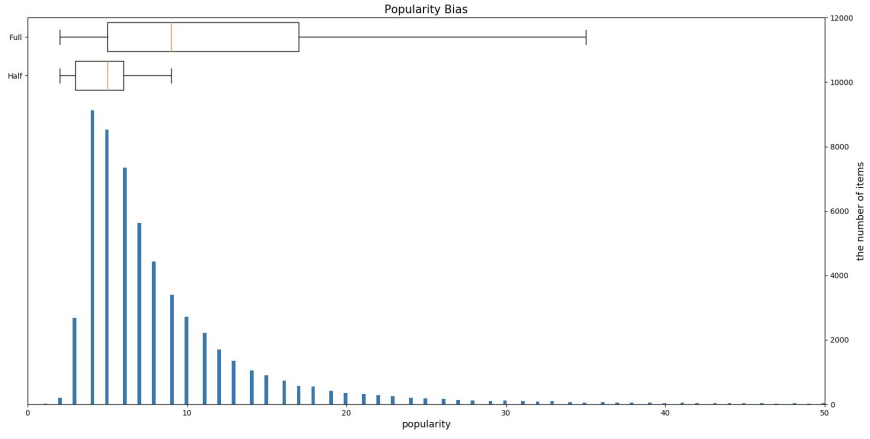


图 3　商品的流行度偏差

问题挑战

该竞赛的主要挑战是消除推荐系统中的偏差，从上述数据分析中可以看出，主要存在两种偏差，选择性偏差（Selection Bias）和流行度偏差（Popularity Bias）。

- 选择性偏差：曝光数据是由模型和系统选择的，与系统中的全部候选集不一致 [4,5]。
- 流行度偏差：商品历史点击次数呈现一个长尾分布，故而流行度偏差存在于头部商品和尾部商品之间，如何解决流行度偏差也是赛题的核心挑战之一 [6,7]。

基于上述偏差，传统的利用 Pageview（曝光）->Click（点击）的点击预估建模思路并不能合理地建模用户的真实兴趣，我们在初步尝试中也发现采用传统建模思路效果较差。不同于传统的用户兴趣建模思路，首先，我们通过 u2i2i（user2item2item）建模转换，采用侧重于 i2i 的建模代替传统 CTR 预估方式中的 u2i（user2item）的兴趣建模。并且，我们采用基于 i2i 图的多跳游走进行候选样本生成，代替基于 Pageview 样本生成思路。同时，在构图过程、i2i 建模过程我们引入了流行度惩罚。最终有效地解决了上面的偏差挑战。

竞赛技术方案

针对选择性偏差和流行度偏差两方面挑战，我们进行了建模设计，有效地优化了上述偏差。已有的 CTR 建模方法可以理解为 u2i 的建模，通常刻画了用户在特定请求上下文中对候选商品的偏好，而我们的建模方式是去学习用户的每个历史点击商品和候选商品的关系，可以理解为 u2i2i 的建模。这种建模方法更有助于学习多种 i2i 关系，并且可以容易地将 i2i 图中的一跳关系拓展到多跳关系，多种 i2i 关系可以探索更多无偏数据来增大商品候选集和训练集，达到了缓解选择性偏差的目的。同时，考虑到流行商品引起的流行度偏差，我们在构图过程中对边权引入流行度惩罚，使得多跳游走时更有机会探索到低流行度的商品，同时在建模过程以及后处理过程中我们也引入了流行度惩罚，缓解了流行度偏差。

最终，我们形成了一个基于 i2i 建模的排序框架，框架图如图 4 所示。在我们的框架中商品推荐过程被分为三个阶段，第一个阶段是基于用户行为数据和商品多模态数据构建 i2i 图，并基于 i2i 图进行多跳游走生成 i2i 候选样本；第二个阶段是拆分用户点击序列，并根据 i2i 候选样本构建 i2i 关系样本集，基于 i2i 样本集进行自动化特征工程，以及使用流行度加权的损失函数进行消除流行度偏差的建模；第三个阶段根据用户点击序列将 i2i 模型生成的 i2i 打分进行聚合，对打分的商品列表进行消除流行度偏差的后处理，从而对商品列表进行排序推荐。我们将详细介绍这三个阶段的方案。

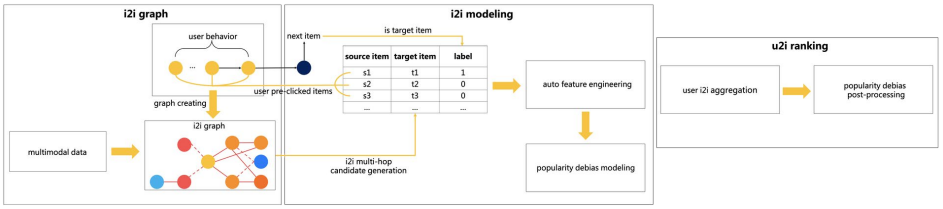


图 4　基于 i2i 建模的排序框架

基于多跳游走的 i2i 候选样本生成

为了探索更多的 i2i 无偏候选样本来进行 i2i 建模，从而缓解选择性偏差，我们构建了一个具有多种边关系的 i2i 图，并在构边过程中引入了流行度惩罚来消除流行度偏差。如下图 5 所示，i2i 图的构建与多跳游走 i2i 候选样本的生成过程被分为三个步骤：i2i 图的构建、i2i 多跳游走以及 i2i 候选样本的生成。

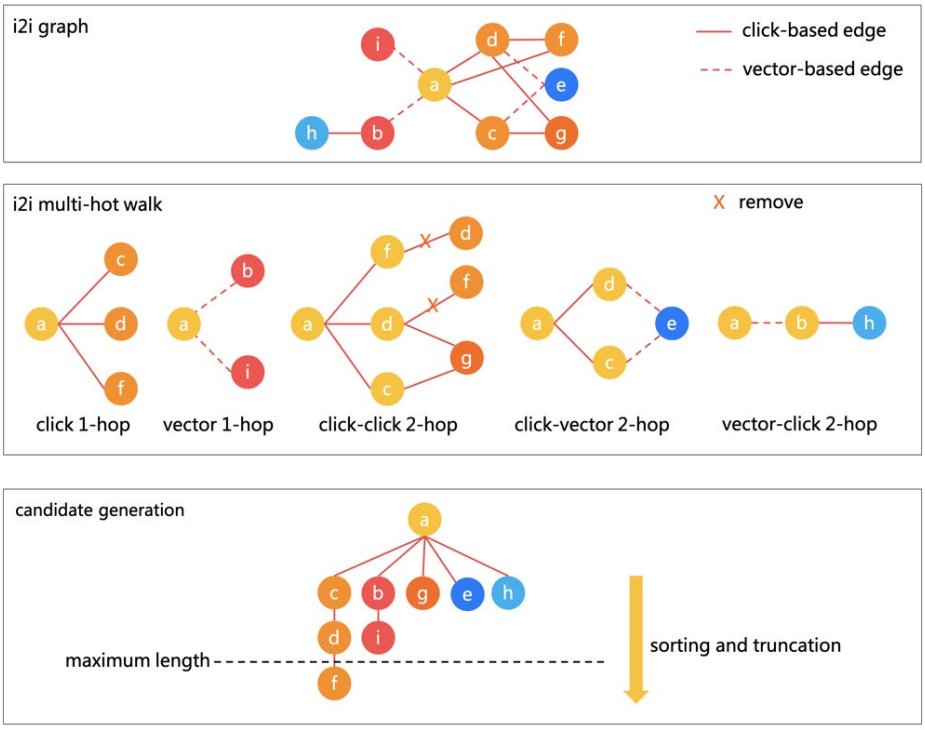


图 5　基于多跳游走的 i2i 候选样本生成

第一个步骤为 i2i 图的构建，图中存在一种结点即商品结点，两种边关系即点击共现边和多模态向量边。点击共现边通过用户的历史商品点击序列所构建，边的权重通过以下的公式得到，其在两个商品间的用户历史点击共现频数的基础上，考虑了每次点击共现的时间间隔因子，并加入了用户活跃度惩罚以及商品流行度惩罚。时间间隔因子考虑到了两个商品间的共现时间越短则这两个商品有更大的相似度；用户活跃度惩罚考虑了活跃用户与不活跃用户的公平性，通过用户历史商品点击次数来惩罚活跃用户；商品流行度惩罚考虑了商品的历史点击频数，对流行商品进行惩罚，缓解了流行度偏差 [8]。多模态向量边则通过两个商品间文本向量及图片向量的余弦相似度进行构建，对一个商品的向量利用 K 最近邻的方法去寻找最邻近的 K 个商品，对这个商品与其最近邻的 K 个商品分别构建 K 条边，向量间的相似度即为边权，多模态向量边与流行度无关，可以缓解流行度偏差。

user activity smoothing

$$W_{c_{ij}} = \sum_{u \in U_i \cap U_j} 1/(\delta(t_{ui} - t_{uj}) log(1 + q_u p_i p_j)$$

item popularity smoothing

第二个步骤是通过多跳游走探索多种 i2i 关系，我们通过枚举不同的一跳 i2i 关系组合构成不同类型的二跳 i2i 关系，并且在构建好二跳 i2i 关系之后删除原本的一跳 i2i 关系以避免冗余。i2i 关系包括基于点击一跳邻居构建 i2i，基于向量一跳邻居构建 i2i，基于点击 – 点击二跳游走构建 i2i，基于点击 – 向量二跳游走构建 i2i，基于向量 – 点击二跳游走构建 i2i，一跳 i2i 关系得分由一跳边权得来，多跳 i2i 关系得分则由以下公式得来，即对每条路径的边权相乘得到路径分，并对所有路径分求平均。通过不同边类型多跳游走的方式，更多的商品有更多的机会和其他商品构建多跳关系，从而扩大了商品候选集，缓解了选择性偏差。

$$W_{2-hop} = 1/K \sum_{k} W_{ik} W_{kj}$$

第三个步骤则基于每种 i2i 关系根据 i2i 得分对所有商品的候选商品集合分别进行排序和截断，每个 i2i 关系间的相似度热图如下图 6 所示，相似度是通过两种 i2i 关系构造的候选集重复度所计算，我们可以根据不同 i2i 关系之间的相似度来确定候选商品集合的数量截断，以得到每种 i2i 关系中每个商品的 i2i 候选集，供后续 i2i 建模使用。

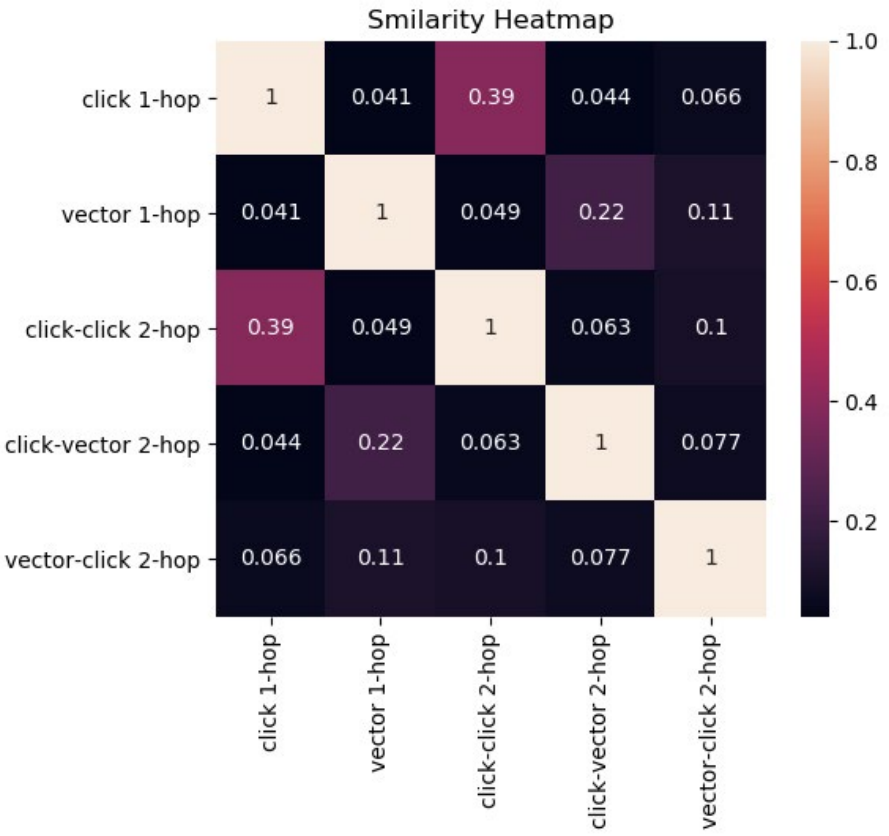


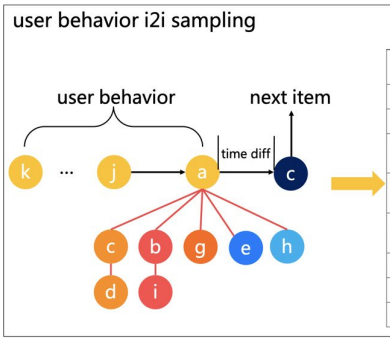
图 6　i2i 关系相似度热图

基于流行度偏差优化的 i2i 建模

我们通过 u2i2i 建模转换，将传统的基于 u2i 的 CTR 预估建模方式转换为 i2i 建模方式，它可以容易地使用多跳 i2i 关系，同时我们引入带流行度惩罚的损失函数，使得 i2i 模型朝着缓解流行度偏差的方向学习。

如下图 7 所示，我们拆分用户前置点击行为序列，将每一个点击的商品作为 source item，从 i2i graph 中的多跳游走候选集中抽取 target item，形成 i2i 样本集。对于 target item 集合，我们将用户下一次点击的商品与 target item 是否一致来引入该样

本的标签。这样，我们将基于用户选择的序列建模 [9] 转变为基于 i2i 的建模，通过两个商品点击的时间差以及点击次数间隔来从侧面引入用户的序列信息，强调了 i2i 的学习，从而达到消除选择性偏差的目的。最终用户的推荐商品排序列表可以基于用户下的 i2i 打分进行 target item 的排序。



user	source item	target item	pre-click K	time diff	type	label
u_1	a	c	1	2s	click	1
u_1	a	b	1	2s	vector	0
u_1	a	g	1	2s	click-click	0
u_1	a	e	1	2s	click-vector	0
u_1	a	h	1	2s	vector-click	0
u_1	...	...	...	...	...	...
u_1	j	...	2	8s	...	...
...	...	...	...	...	...	...

图 7　i2i 训练样本生成

如图 8 所示，我们利用自动化特征工程的思想去探索高阶特征组合，缓解了偏差问题业务含义抽象的问题。我们通过人工构造一些基础特征例如频数特征、图特征、行为特征和时间相关特征等特征后，将这些基本的特征类型划分为 3 种，类别特征、数值特征以及时间特征，基于这些特征做高阶特征组合，每一次组合形成的特征都会加入下一次组合的迭代之中，来降低高阶组合的复杂度，我们并且基于特征重要性和 NDCG@50_half 进行快速的特征选择，从而挖掘到了更深层次的模式并节省了大量的人力成本。

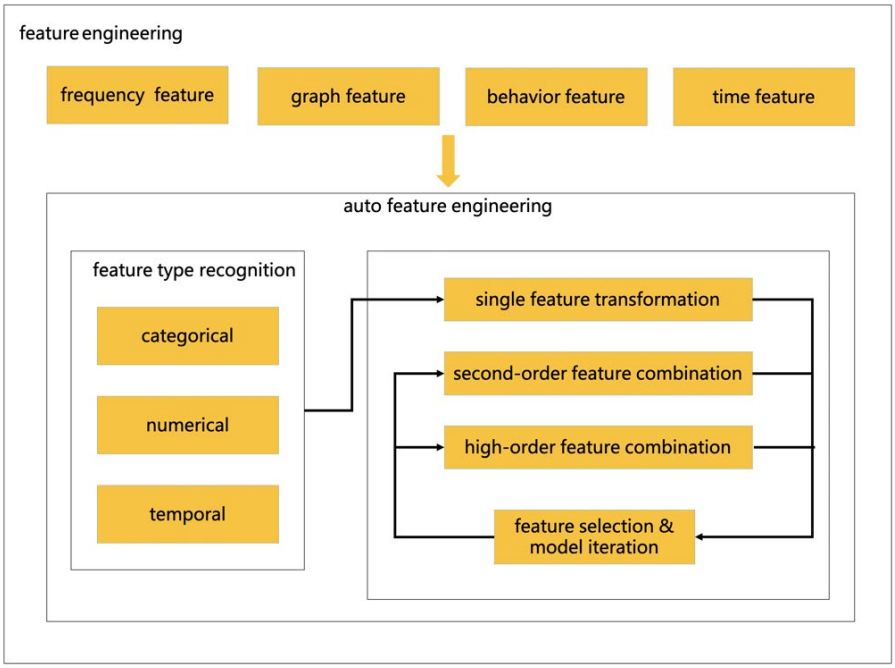


图 8　自动化特征工程

在模型上，我们尝试了 LightGBM、Wide&Deep、时序模型等等，最终由于 LightGBM 在 tabular 上的优异表现力，选择了 LightGBM。

在模型训练中，我们使用商品流行度加权损失去消除流行度偏差 [10]，损失函数 L 如下式所示：

$$L = (\alpha + \beta)ylogp + (1 - y)log(1 - p)$$

其中，参数 α 与流行度成反比，来削弱流行商品的权重，可以消除流行度偏差。参数 β 是正样本权重，用来解决样本不平衡问题。

用户偏好排序

最终，用户的商品偏好排序是通过用户的历史点击商品来引入 i2i，继而对 i2i 引入的所有商品形成最终的排序问题。在排序过程中，根据图 7 所示，target item 集合是

由每一个 source item 分别产出的，所以不同的 source item 以及不同的多跳游走 i2i 关系可能会产出相同的 target item。我们需要考虑如何将相同用户的相同 target item 的模型打分值进行聚合，如果直接进行概率求和会加强流行度偏差，而直接取均值又容易忽略掉一些强信号。最终，我们对一个用户多个相同的 target item 采用最大池化聚合的方式，然后对用户的所有 target item 进行排序，可以在 NDCG@50_half 上取得一个不错的效果。

为了进一步优化 NDCG@50_half 指标，我们对所得到的 target item 打分进行后处理，通过提高低流行度商品的打分权重来进一步打压高流行度的商品，最终在 NDCG@50_half 上取得了一个更好的效果，这其实是一个 NDCG@50_full 与 NDCG@50_half 的权衡。

评估结果

在基于多跳游走的 i2i 候选样本生成过程中，各种 i2i 关系的 hitrate 如表 2 所示，可以发现，在相同长度为 1000 的截断下对多种方法做混合有更高的 hitrate 提升，能引入更多无偏数据来增大训练集和候选集从而缓解系统的选择性偏差。

method	hitrate@50_full	hitrate@50_half	hitrate@1000_full	hitrate@1000_half
click 1-hop	0.1814	0.1384	0.2552	0.2090
vector 1-hop	0.1218	0.1238	0.1787	0.1879
click-click 2-hop	0.1898	0.1237	0.2417	0.1969
click-vector 2-hop	0.1130	0.1158	0.1591	0.1536
vector-click 2-hop	0.1325	0.0827	0.2289	0.1513
mixture	**0.1922**	**0.1523**	**0.2923**	**0.2649**

表 2　不同 i2i 关系的 hitrate

最终，由美团搜索广告团队组建的 Aister 在包括 NDCG 和 hitrate 的各项评价指

标中都取得了第 1 名，如表 3 所示，NDCG@50_half 比第二名高了 6.0%，而 NDCG@50_full 比第二名高了 4.9%，NDCG@50_half 相较于 NDCG@50_full 有更明显的优势，说明我们更好地针对消除偏差问题进行了优化。

Team	Rank	NDCG@50_half	NDCG@50_full
aister	**1**	**0.452**	**0.298**
DeepWisdom	2	0.392	0.249
TheAvengers	3	0.363	0.274
hello dog	4	0.324	0.277
ECNU@KDDCUP	5	0.320	0.258

表 3　不同参赛团队解决方案的 NDCG 评估结果

广告业务应用

搜索广算法团队负责美团与点评双平台的搜索广告与筛选列表广告业务，业务类型涉及餐饮、休闲娱乐、丽人、酒店等，丰富的业务类型为算法优化带来很大空间与挑战。在搜索广告业务问题中，数据偏差问题是个重要且具挑战性的问题。广告系统中有两个重要的数据偏差——位置偏差与选择性偏差，搜索广告算法团队也针对这两个偏差问题进行了较多优化。在位置偏差问题，即位置靠前的点击率天然高于位置靠后的，不同于传统的作为偏差的处理方式，我们引入一致性建模的思想，并通过灵活的深度网络设计达到一致性目标，取得业务效果提升。在选择性偏差问题，整个广告系统投放过程呈现出了一个漏斗图，如图 9 所示，系统分为 Matching、Creative-Select、Ranking、Auction 几个阶段。每一个阶段的候选是由上一阶段选择。以排序阶段为例（Ranking），线上系统排序的候选包含了匹配（Matching）阶

段输出的所有候选，但是排序模型的训练数据是根据模型选择的曝光（Pageview）数据，仅为线上排序系统候选的一个小的子集，模型线上与线下输入数据的差异违反了建模分布一致性假设，上述选择性偏差会导致两方面明显的问题：

1. 模型预估不准确：从曝光样本中学习到的模型存在偏差且不准确，会导致线上预估效果较差，尤其对于同历史曝光样本分布差异大的候选样本。
2. 反馈链路循环影响广告生态：由于模型选择的样本进行曝光，然后进入模型训练进一步选择新的曝光样本，模型基于有偏样本不断学习，使得整体反馈环路不断受到偏差影响，系统选择面越来越窄形成“马太效应”。

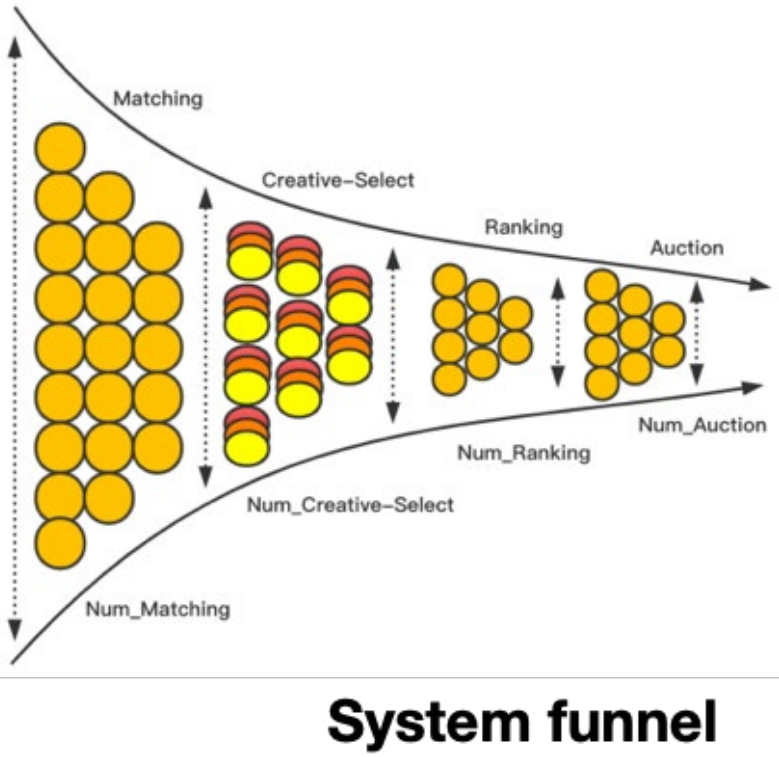


图 9　广告系统的漏斗图

为了解决上面的预估与生态问题，我们通过样本生成和多阶段训练两方面进行算法优化。在样本生成方面，我们进行三方面的数据生成与样本选择。首先，如图 10 所

示，我们采用基于 Beta 分布的 Exploration 算法，通过历史点击率和统计置信度生成 Exploration 候选，算法背后的假设是置信度越大点击率的方差越小。如下图所示，横轴代表预估点击率，纵轴代表概率密度，在黄框中参数的 Beta 分布生成的样本预估点击率分布接近于真实的样本分布，用于补充仅通过模型选择的曝光数据；其次，我们结合随机游走进行负样本优化，并通过采样算法和 Label 优化来控制精度。最后，训练样本大多由系统主流量选择，而在下一次模型优化全量后选择的训练样本会发生较大变化，上述差异性也会导致在 ABTest 时小流量模型精度不符合预期，我们也针对上述不同模型挑选的数据分布差异进行数据选择。

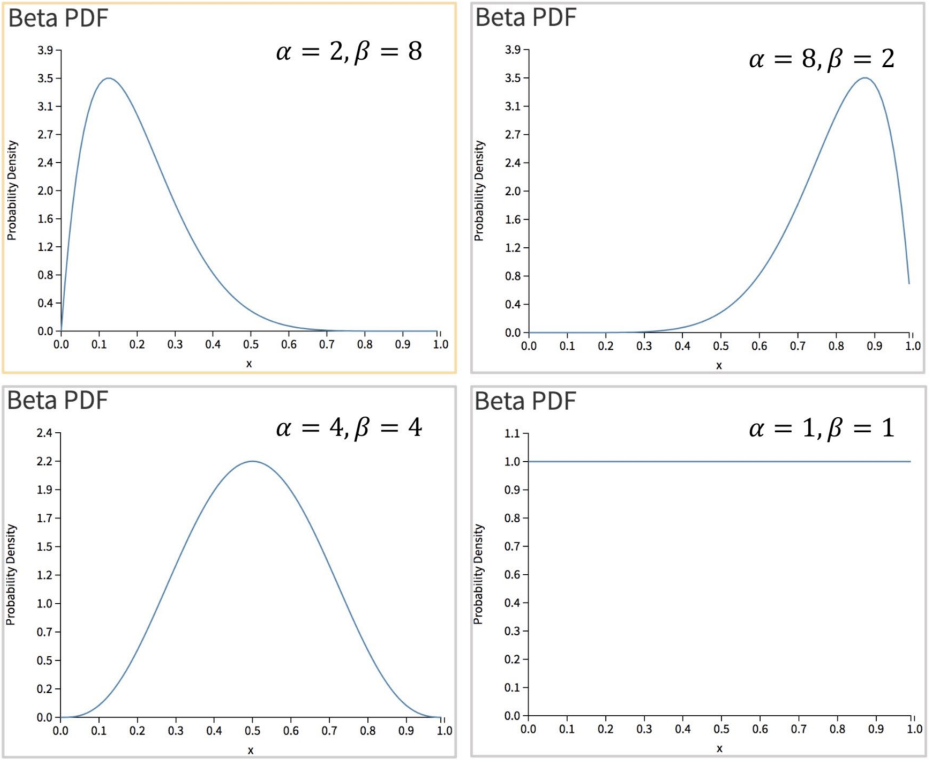


图 10　不同参数的 Beta 分布

并且，结合上述多种样本分布的差异性，通过多阶段训练来优化模型，如图 11 所示，我们基于样本强度控制训练顺序与参数，使得训练数据同线上真实候选分布更一致。

最终不仅在 CTR 预估模型（Ranking 阶段）和创意优选模型（Creative-Select 阶段）两个模块均取得较显著的业务效果提升，并且更一致的建模方式也使得了候选扩量等偏差较重问题的实验由负向变正向，更扎实的验证方式也为未来优化打下了坚实的基础。

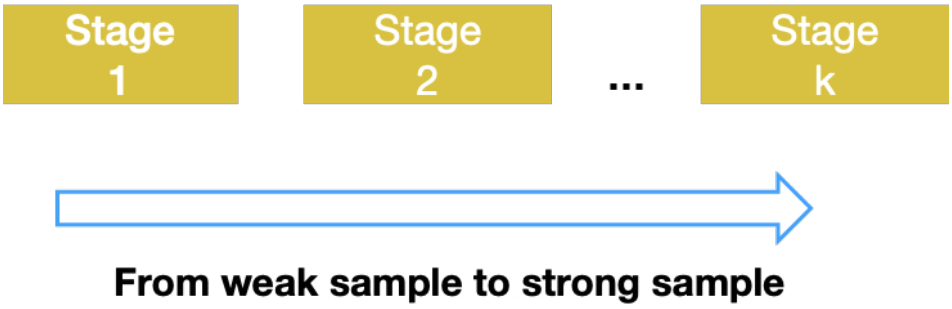


图 11　基于样本强度的多阶段训练

总结与展望

KDD Cup 是同工业界联接非常紧密的比赛，每年赛题紧扣业界热点问题与实际问题，其中历年产出的 Winning Solution 对工业界也有很大的影响。例如，KDD Cup 2012 获胜方案产出了 FFM (Feild-aware Factorization Machine) 与 XGBoost 的原型，在工业界取得广泛应用。

今年 KDD Cup 的 Debiasing 问题也是当前广告与推荐领域中最具挑战性的问题之一，本文介绍了我们在 KDD Cup 2020 Debiasing 赛题上取得第 1 名的解决方案，解决方案不同于以往 CTR 预估方式等 u2i 的兴趣建模方法，我们采用 u2i2i 方式将 u2i 建模转换为 i2i 建模，并构建异构图通过多跳游走探索更多无偏样本，从而缓解了选择性偏差，在建模过程中对图的构建、模型的损失函数以及预估值后处理等过程都引入了流行度惩罚来缓解流行度偏差，最终克服了选择性偏差和流行度偏差两个赛题挑战。

同时本文也介绍我们在美团搜索广告上关于数据选择性偏差问题的业务应用，之前在

广告系统中已经针对偏差问题进行了较多优化，这次比赛也让我们对偏差问题的研究方向有了更进一步的认知。我们希望在未来的工作中会基于本次比赛取得的偏差优化经验进一步地去优化广告系统中的偏差问题，让广告系统变得更加公平。

参考文献

[1] Fairness in Recommender Systems
[2] Singh A, Joachims T. Fairness of exposure in rankings[C]//Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. 2018: 2219-2228.
[3] Stinson C. Algorithms are not Neutral: Bias in Recommendation Systems[J]. 2019.
[4] Ovaisi Z, Ahsan R, Zhang Y, et al. Correcting for Selection Bias in Learning-to-rank Systems[C]//Proceedings of The Web Conference 2020. 2020: 1863-1873.
[5] Wang X, Bendersky M, Metzler D, et al. Learning to rank with selection bias in personal search[C]//Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval. 2016: 115-124.
[6] Abdollahpouri H, Burke R, Mobasher B. Controlling popularity bias in learning-to-rank recommendation[C]//Proceedings of the Eleventh ACM Conference on Recommender Systems. 2017: 42-46.
[7] Abdollahpouri H, Mansoury M, Burke R, et al. The impact of popularity bias on fairness and calibration in recommendation[J]. arXiv preprint arXiv:1910.05755, 2019.
[8] Schafer J B, Frankowski D, Herlocker J, et al. Collaborative filtering recommender systems[M]//The adaptive web. Springer, Berlin, Heidelberg, 2007: 291-324.
[9] Zhang S, Tay Y, Yao L, et al. Next item recommendation with self-attention[J]. arXiv preprint arXiv:1808.06414, 2018.
[10] Yao S, Huang B. Beyond parity: Fairness objectives for collaborative filtering[C]//Advances in Neural Information Processing Systems. 2017: 2921-2930.

作者简介

坚强，明健，胡可，曲檀，雷军等，均来自美团广告平台搜索广告算法团队。

招聘信息

美团广告平台搜索广告算法团队立足搜索广告场景，探索深度学习、强化学习、人工智能、大数据、知识图谱、NLP 和计算机视觉最前沿的技术发展，探索本地生活服务电商的价值。主要工作方向包括：

- **触发策略**：用户意图识别、广告商家数据理解，Query 改写，深度匹配，相关性建模。

- **质量预估**：广告质量度建模。点击率、转化率、客单价、交易额预估。
- **机制设计**：广告排序机制、竞价机制、出价建议、流量预估、预算分配。
- **创意优化**：智能创意设计。广告图片、文字、团单、优惠信息等展示创意的优化。

岗位要求：

- 有三年以上相关工作经验，对 CTR/CVR 预估，NLP，图像理解，机制设计至少一方面有应用经验。
- 熟悉常用的机器学习、深度学习、强化学习模型。
- 具有优秀的逻辑思维能力，对解决挑战性问题充满热情，对数据敏感，善于分析 / 解决问题。
- 计算机、数学相关专业硕士及以上学历。

具备以下条件优先：

- 有广告 / 搜索 / 推荐等相关业务经验。
- 有大规模机器学习相关经验。

感兴趣的同学可投递简历至：tech@meituan.com（邮件标题请注明：广平搜索团队）。

ICRA 2020 轨迹预测竞赛冠军的方法总结

作者：炎亮　佳禾　德恒　冬淳

行人轨迹预测问题是无人驾驶技术的重要一环，已成为近年来的一项研究热点。在机器人领域国际顶级会议 ICRA 2020 上，美团无人配送团队在行人轨迹预测竞赛中夺冠，本文系对该预测方法的一些经验总结，希望能对大家有所帮助或者启发。

一、背景

6 月 2 日，国际顶级会议 ICRA 2020 举办了“第二届长时人类运动预测研讨会”。该研讨会由博世有限公司、厄勒布鲁大学、斯图加特大学、瑞士洛桑联邦理工联合组织，同时在该研讨会上，还举办了一项行人轨迹预测竞赛，吸引了来自世界各地的 104 支队伍参赛。美团无人配送团队通过采用“世界模型”的交互预测方法，夺得了该比赛的第一名。

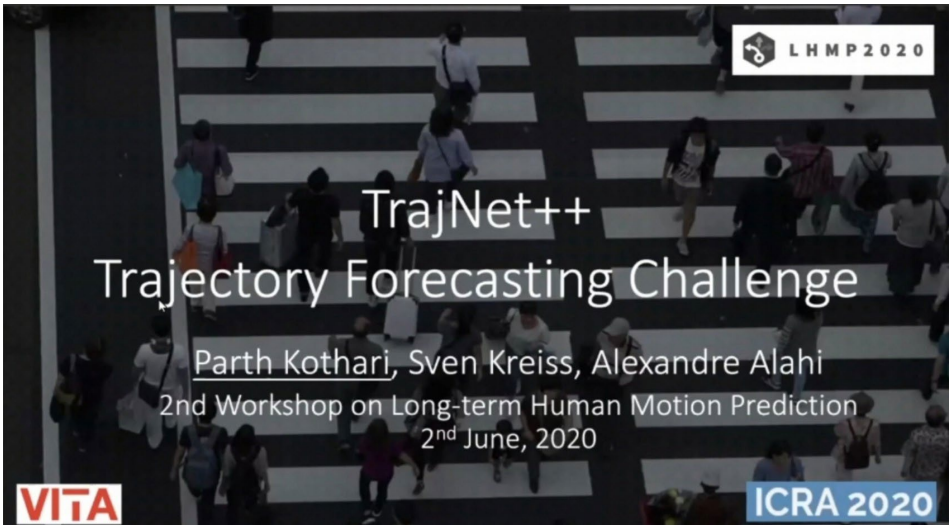


图 1　ICRA 2020 TrajNet++ 轨迹预测竞赛

二、赛题简介

本次竞赛提供了街道、出入口、校园等十个复杂场景下的行人轨迹数据集，要求参赛选手根据这些数据集，利用行人在过去 3.6 秒的轨迹来预测其在未来 4.8 秒的运行轨迹。竞赛使用 FDE（预测轨迹和真实轨迹的终点距离）来对各种算法进行排名。

本次的赛题数据集，主要来源于各类动态场景下的真实标注数据和模拟合成数据，采集频率为 2.5 赫兹，即两个时刻之间的时间差为 0.4 秒。数据集中的行人轨迹都以固定坐标系下的时序坐标序列表示，并且根据行人的周围环境，这些轨迹被分类成不同的类别，例如静态障碍物、线性运动、追随运动、避障行为、团体运动等。在该比赛中，参赛队伍需要根据每个障碍物历史 9 个时刻的轨迹数据（对应 3.6 秒的时间）来预测未来 12 个时刻的轨迹（对应 4.8 秒的时间）。

该竞赛采用多种评价指标，这些评价指标分别对单模态预测模型和多模态预测模型进行评价。单模态模型是指给定确定的历史轨迹，预测算法只输出一条确定的轨迹；而多模态模型则会输出多条可行的轨迹（或者分布）。本次竞赛的排名以单模态指标中的 FDE 指标为基准。

三、方法介绍

其实，美团在很多实际业务中经常要处理行人轨迹预测问题，而行人轨迹预测的难点在于如何在动态复杂环境中，对行人之间的社交行为进行建模。因为在复杂场景中，行人之间的交互非常频繁并且交互的结果将会直接影响他们后续的运动（例如减速让行、绕行避障、加速避障等）。

基于各类带交互数据集，一系列的算法被相继提出，然后对障碍物进行交互预测，这些主流模型的工作重心都是针对复杂场景下行人之间的交互进行建模。常用的方法包括基于 LSTM 的交互算法（SR LSTM[1]、Social GAN[2]、SoPhie[3]、Peeking into[4]、StarNet[5] 等），基于 Graph/Attention 的交互算法（GRIP[6]、Social STGCNN[7]、STGAT[8]、VectorNet[9] 等），以及基于语义地图 / 原始数据的预测算法等。

我们本次的参赛方法就是由自研算法 [10]（如图 2 所示）改进而来，该方法的设计思路是根据场景中所有障碍物的历史轨迹、跟踪信息以及场景信息，建立并维护一个全局的世界模型来挖掘障碍物之间、障碍物与环境之间的交互特性。然后，再通过查询世界模型来获得每个位置邻域内的交互特征，进而来指导对障碍物的预测。

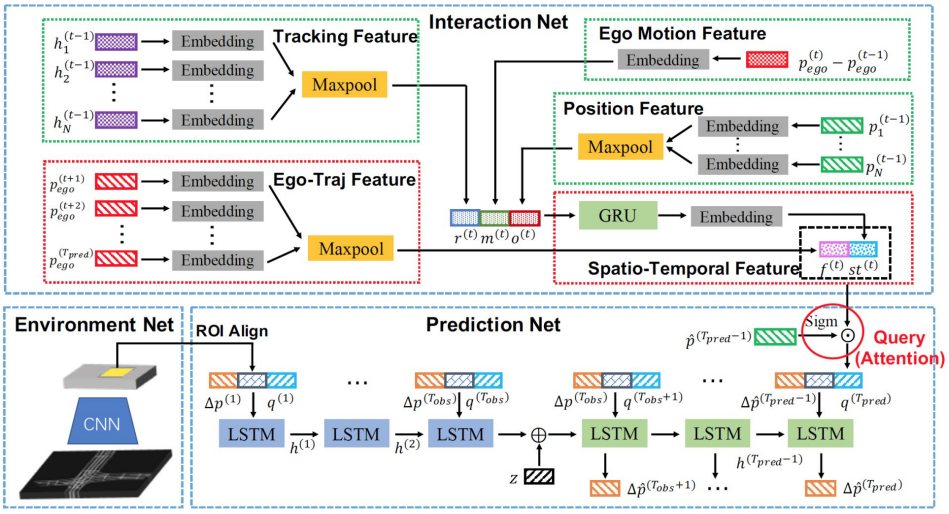


图 2　基于世界模型的预测算法

在实际操作过程中，由于数据集中缺乏场景信息，我们对模型做了适当的调整。在世界模型中（对应上图的 Interaction Net），我们仅使用了现有数据集，以及模型能够提供的位置信息和跟踪信息 LSTM 隐状态信息。最终得到的模型结构设计如下图 3 所示：

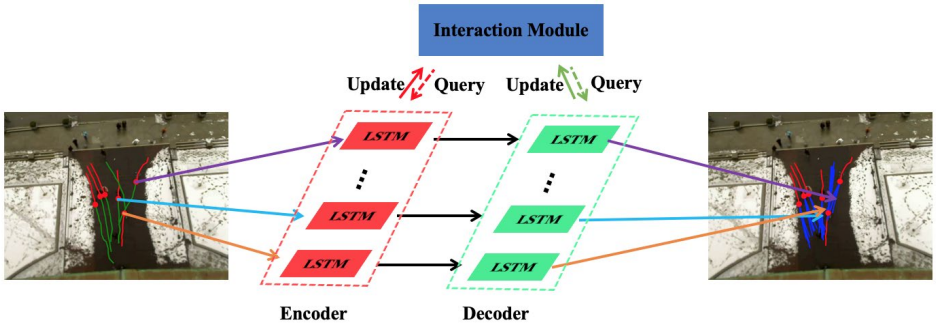


图 3　竞赛使用的基于世界模型的预测算法

整个模型基于 Seq2Seq 结构，主要包含历史轨迹编码模块（Encoder）、世界模型（Interaction Module）和解码预测模块（Decoder）三个部分。其中，编码器的功能在于对行人历史轨迹进行编码，主要提取行人在动态环境中的运动模式；解码器则是利用编码器得到的行人运动模式特征，来预测他们未来的运动轨迹分布。需要强调一下，在整个编码与解码的过程中，都需要对世界模型进行实时更新（Update）与查询（Query）两种操作。更新操作主要根据时序的推进，将行人的运动信息实时编入世界模型中；查询操作则是根据全局的世界地图以及行人的自身位置，来获取行人当前邻域内的环境特征。

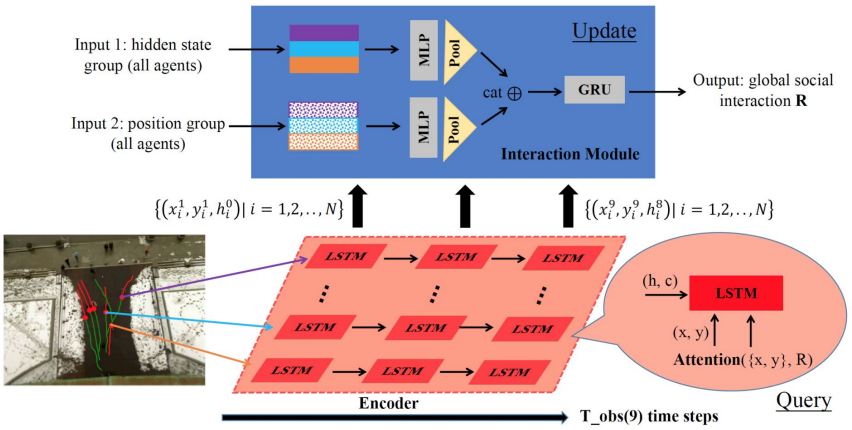


图 4　编码阶段

在图 4 中，展示了我们模型在历史轨迹编码阶段的计算流程。编码阶段共有 9 个时刻，对应 9 个历史观测时间点，每个时刻都执行相同的操作。以 t 时刻为例。

首先，将 t 时刻的所有行人坐标数据，包含：

$$\{(x_i^t, y_i^t) \mid i = 1, 2, ..., N, t = 1, 2, .., T\}$$

位置集合

$$\{(\Delta x_i^t, \Delta y_i^t) \mid i = 1, 2, ..., N, t = 1, 2, .., T\}$$

速度集合

$$\{h_i^t \mid i = 1, 2, ..., N, t = 1, 2, .., T\}$$

所有行人跟踪信息 – 上时刻编码得到的 LSTM 隐状态

将以上信息输入到世界模型中更新地图信息，即 Update 操作。整个 Update 操作经过 MLP、MaxPooling 以及 GRU 等模块获得一个全局的时空地图特征 R；然后，每个 LSTM（对应一个行人），使用其当前观测时刻的坐标信息：

$$(x_i^t, y_i^t, \Delta x_i^t, \Delta y_i^t)$$

解码预测阶段的流程与历史轨迹编码阶段基本一致，但存在两个细微的不同点：

- 区别 1：编码阶段每个行人对应的 LSTM 隐状态的初始化为 0；而解码阶段，LSTM 由编码阶段的 LSTM 隐状态和噪声共同初始化。
- 区别 2：编码阶段行人对应的 LSTM 和世界模型使用的是行人历史观测坐标；而解码阶段使用的是上时刻预测的行人坐标。

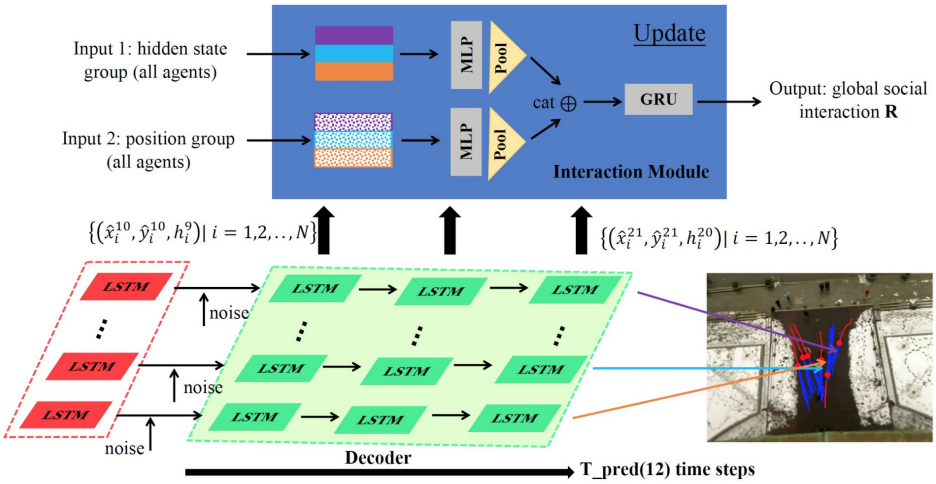


图 5　解码预测阶段

四、数据预处理与后处理

为了对数据有更好的理解，便于使用更适合的模型，我们对训练数据做了一些预处理操作。首先，数据集给出了各个行人的行为标签，这些标签是根据规则得到的，由于我们采用了交互预测的方法，希望模型能自动学习行人与周围主体之间的位置关系、速度关系等，所以我们就不直接使用标注中的“类型”信息；然后这次比赛的数据采集自马路、校园等不同场景中行人的运动轨迹。场景之间的差异性非常大，训练集和测试集数据分布不太一致。

于是，我们做了数据的可视化工作，将所有轨迹数据的起点放置于坐标轴的原点处，根据历史观测轨迹（前 9 个时刻）终点的位置朝向，将所有轨迹分为 4 类：沿左上方运动（top-left moving）、沿右上方运动（top-right moving）、沿左下方运动（bottom-left moving）和沿右下方运动（bottom-right moving）。分布的结果如图 6 所示，可以发现，训练集和测试集的数据分布存在一定的差距。

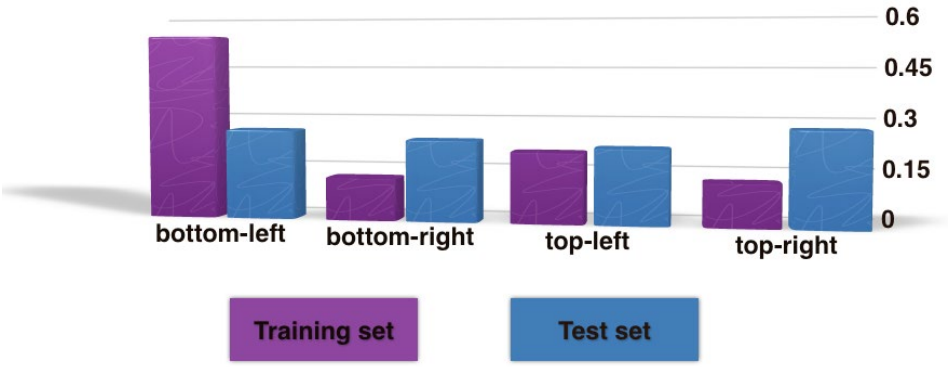


图 6　训练集与测试集历史观测轨迹中行人运动方向分布

针对上述问题，我们对训练集做了 2 项预处理来提高训练集与测试集分布的一致性：

- 平衡性采样；
- 场景数据正则化（缺失轨迹点插值，轨迹中心化以及随机旋转）。

此外，对于预测结果，我们也做了相应的后处理操作进行轨迹修正，主要是轨迹点的裁剪以及基于非极大值抑制的轨迹选择。图 7 展示了两个场景中行人的运动区域，可以看到有明显的边界，对于超出边界的轨迹，我们做了相应的修正，从而保证轨迹的合理性。

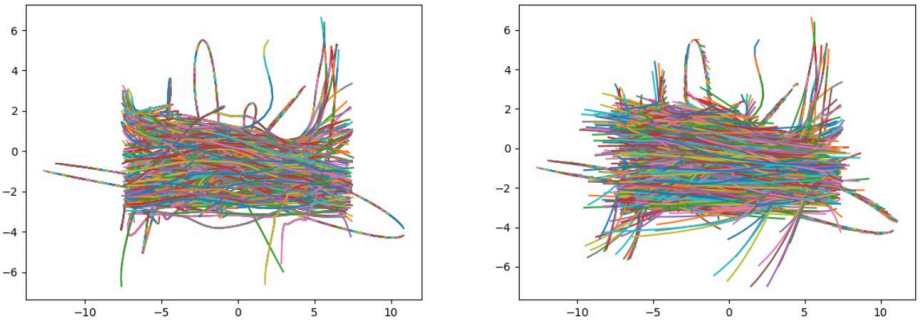


图 7　训练轨迹的可视化

最后在训练技巧上，我们也使用 K-Fold Cross Validation 和 Grid Search 方法来做自适应的参数调优。最终在测试集上取得 FDE 1.24 米的性能，而获得比赛第二名的方法的 FDE 为 1.30 米。

五、总结

行人轨迹预测是当前一个非常热门的研究领域，随着越来越多的学者以及研究机构的参与，预测方法也在日益地进步与完善。美团无人配送团队也期待能与业界一起在该领域做出更多、更好的解决方案。比较幸运的是，这次竞赛的场景与我们美团无人配送的场景具备一定的相似性，所以我们相信未来它能够直接为业务赋能。目前，我们已经将该研究工作在竞赛中进行了测试，也验证了算法的性能，同时为该算法在业务中落地提供了一个很好的支撑。

六、参考文献

[1] Zhang P, Ouyang W, Zhang P, et al. Sr-lstm: State refinement for lstm towards pedestrian trajectory prediction[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2019: 12085-12094.

[2] Gupta A, Johnson J, Fei-Fei L, et al. Social gan: Socially acceptable trajectories with generative adversarial networks[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2018: 2255-2264.

[3] Sadeghian A, Kosaraju V, Sadeghian A, et al. Sophie: An attentive gan for predicting paths compliant to social and physical constraints[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2019: 1349-1358.

[4] Liang J, Jiang L, Niebles J C, et al. Peeking into the future: Predicting future person activities and locations in videos[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2019: 5725-5734.

[5] Zhu Y, Qian D, Ren D, et al. StarNet: Pedestrian trajectory prediction using deep neural network in star topology[C]//Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems. 2019: 8075-8080.

[6] Li X, Ying X, Chuah M C. GRIP: Graph-based interaction-aware trajectory prediction[C]//Proceedings of the IEEE Intelligent Transportation Systems Conference. IEEE, 2019: 3960-3966.

[7] Mohamed A, Qian K, Elhoseiny M, et al. Social-STGCNN: A Social spatio-temporal graph convolutional neural network for human trajectory prediction[J]. arXiv preprint arXiv:2002.11927, 2020.

[8] Huang Y, Bi H K, Li Z, et al. STGAT: Modeling spatial-temporal interactions for human trajectory prediction[C]//Proceedings of the IEEE International Conference on Computer Vision. 2019: 6272-6281.
[9] Gao J, Sun C, Zhao H, et al. VectorNet: Encoding HD maps and agent dynamics from vectorized representation[J]. arXiv preprint arXiv:2005.04259, 2020.
[10] Zhu Y, Ren D, Fan M, et al. Robust trajectory forecasting for multiple intelligent agents in dynamic scene[J]. arXiv preprint arXiv:2005.13133, 2020.

七、作者简介

炎亮，美团无人车配送中心算法工程师。
佳禾，浙江大学在读研究生，美团无人车配送中心实习生。
德恒，美团无人车配送中心算法工程师。
冬淳，美团无人车配送中心算法工程师。

KDD Cup 2020 AutoGraph 比赛冠军技术方案及在美团的实践

作者：坚强　胡可　金鹏　雷军

背景

ACM SIGKDD（国际数据挖掘与知识发现大会，简称 KDD）是数据挖掘领域的国际顶级会议。KDD Cup 比赛是由 SIGKDD 主办的数据挖掘研究领域的国际顶级赛事，从 1997 年开始，每年举办一次，是目前数据挖掘领域最具影响力的赛事。该比赛同时面向企业界和学术界，云集了世界数据挖掘界的顶尖专家、学者、工程师、学生等参加，为数据挖掘从业者们提供了一个学术交流和研究成果展示的平台。KDD Cup 2020 共设置五道赛题（四个赛道），分别涉及数据偏差问题（Debiasing）、多模态召回问题（Multimodalities Recall）、自动化图学习（AutoGraph）、对抗学习问题和强化学习问题。

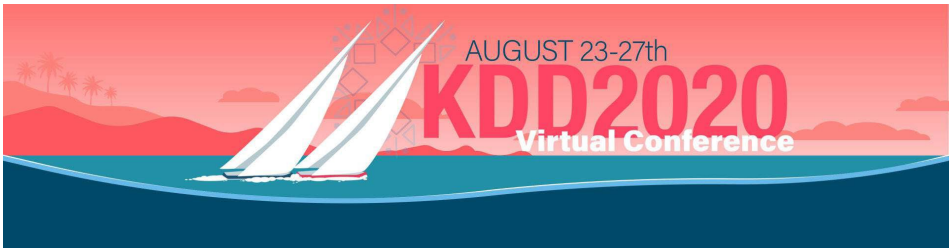
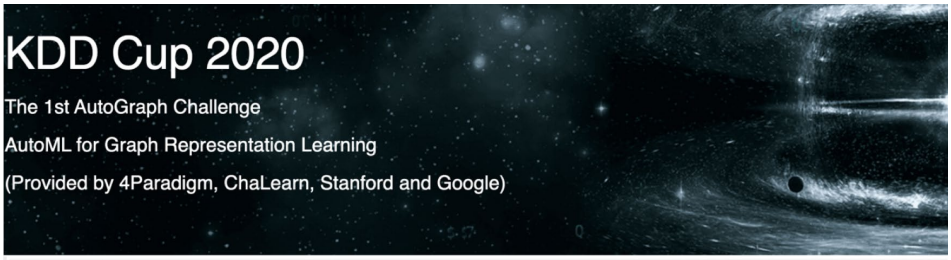


图 1　KDD 2020 会议

美团到店广告平台搜索广告算法团队基于自身的业务场景，一直在不断进行前沿技术的深入优化与算法创新，团队在图学习、数据偏差、多模态学习三个前沿领域均有一定的算法研究与应用，并取得了不错的业务结果。基于这三个领域的技术积累，我们在比赛中选择了三道紧密联系的赛题，希望应用并提升这三个领域技术积累，带来技术与业务的进一步突破。搜索广告算法团队的黄坚强、胡可、漆毅、曲檀、明健、博航、雷军与中科院大学唐兴元共同组建参赛队伍 Aister，参加了 AutoGraph、

Debiasing、Multimodalities Recall 三道赛题，最终在 AutoGraph 赛道中获得了冠军（1/149），在 Debiasing 赛道中获得冠军（1/1895），并在 Multimodalities Recall 赛道中获得了季军（3/1433）。

近些年来，图神经网络（GNN）在广告系统、社交网络、知识图谱甚至生命科学等各个领域都得到了越来越广泛的应用。广告系统中存在着较为丰富的 User-Ad、Query-Ad、Ad-Ad、Query-Query 等结构化关系，搜索广告算法团队成功地将图表示学习应用于广告系统上，业务效果得到了一定的提升。此外，基于广告系统上图学习的技术积累，团队在今年 KDD Cup 的 AutoGraph 赛道中斩获了第一名。本文将介绍 AutoGraph 赛题的技术方案，以及团队在广告系统中图表示学习的应用与研究，希望对从事相关研究的同学能够有所帮助或者启发。



Rank	Team	Code
1	aister	https://github.com/aister2020/KDDCUP_2020_AutoGraph_1st_Place
2	PASA_NJU	https://github.com/Unkrible/AutoGraph2020
3	qqerret	https://github.com/white-bird/kdd2020_GCN
4	common	https://github.com/joneswong/AutoGraph
5	PostDawn	https://github.com/hxttkl/KDDCUP2020_AutoEnsemble

图 2　KDD Cup 2020 AutoGraph 比赛 TOP 5 榜单

赛题介绍与问题分析

AutoGraph 问题概述

自动化图表示学习挑战赛（AutoGraph）是有史以来第一个应用于图结构数据的 AutoML 挑战，是 AutoML 与 Graph Learning 两个前沿领域的结合。KDD Cup

2020 中的 AutoML 赛道挑战，由第四范式、ChaLearn、斯坦福大学和 Google 提供。

图结构数据在现实世界中无处不在，例如社交网络、论文网络、知识图谱等。图表示学习一直是一个非常热门的话题，它的目标是学习图中每个结点的低维表示 ，然后可用于下游任务，例如社交网络中的朋友推荐，或将学术论文分类为引用网络中的不同主题。 传统做法一般利用启发法从图中提取每个结点的特征，例如度统计或基于随机游走的相似性。 近些年来，业界提出了大量用于图表示学习任务的复杂模型，例如图神经网络（GNN）[1]，已经帮助很多任务（例如结点分类或链接预测）取得了新的成果。

然而，无论是传统的启发式方法还是最近基于 GNN 的方法，都需要投入大量的计算和专业知识资源，只有这样才能获得令人满意的任务性能。 例如在 Deepwalk[2] 和 Node2Vec[3] 中，必须对两种众所周知的基于随机游动的方法进行微调，以获得各种不同的超参数，例如每个结点的游走长度和数量、窗口大小等，以获得更好的性能。 而当使用 GNN 模型时，例如 GraphSAGE[4] 或 GAT[5]，我们必须花费大量时间来选择 GraphSAGE 中的最佳聚合函数或 GAT 中多头自注意力头的数量。因此，由于人类专家在调参过程需要付出大量时间和精力，进而限制了现有图表示模型的应用。

AutoML[6] 是降低机器学习应用程序中人力成本的一种有效方法，并且在超参数调整、模型选择、神经体系结构搜索和特征工程方面都取得了令人鼓舞的成绩。 为了使更多的人和组织能够充分利用其图结构数据，KDD Cup 2020 AutoML 赛道举办了针对图结构数据的 AutoGraph 竞赛。在这一竞赛中，参与者应设计一个解决方案来自动化进行图表示学习问题（无需任何人工干预）。该解决方案可以基于图的给定特征、邻域和结构信息，有效而高效地学习每个结点的高质量表示，解决方案应设计为自动提取和利用图中的任何有用信号。

本次 AutoGraph 竞赛针对自动化图学习这一前沿领域，选择了图结点多分类任务来评估表示学习的质量。竞赛官方准备了 15 个图结构数据集，其中 5 个数据集可供下

载，以便参赛者离线开发其解决方案。 除此之外，还将向参与者提供另外 5 个反馈数据集，以评估其 AutoGraph 解决方案的公共排行榜得分。 之后，无需人工干预，竞赛的最后一次提交将在剩余的 5 个数据集里进行评估，这 5 个数据集对于参赛者而言是一直不可见的，评估排名最终会被用来评估所有参赛者的解决方案。 而且，这些数据集是从真实业务中收集的，随机划分为训练集和测试集，每个数据集给予了图结点 id 和结点特征，以及图边和边权信息，并且每个数据集都给了时间预算。参赛者必须在给定的时间预算和算力内存限制下设计一个自动化图学习解决方案，对每个数据集进行结点分类。每个数据集会通过精度（Accuracy）来评估准确性，通过精度可以确定参赛者们在每个数据集的排名，最终排名将根据最后 5 个数据集的平均排名来评估。

数据分析与问题理解

我们对离线五个图数据集进行分析，发现其图的类型多种多样，如下表 1 所示。从图的平均度可以看出离线图 3、4 较为稠密，而图 1、2、5 较为稀疏，从特征数量可以看出图 1、2、3、4 带有结点特征，图 5 无结点特征，同时我们发现图 4 是有向图而图 1、2、3、5 是无向图，我们考虑将图类型划分为有向图 / 无向图、稠密图 / 稀疏图、带特征图 / 无特征图等。

从表 1 中，我们也可以看出大部分图数据集的时间限制都在 100 秒左右，这是一个很短的时间限制，大部分神经网络架构和超参数搜索方案 [7,8,9,10] 都需要一个较长的搜索时间，需要数十个小时甚至长达数天进行架构和超参数搜索。因此，不同于神经网络架构搜索，我们需要一个结构和超参数快速搜索的方案。

图数据集	1	2	3	4	5
结点个数	2708	3327	10000	10000	7521
边数量	5278	4552	733316	5833962	7804
平均度	1.949	1.368	73.3316	583.3962	1.0376
特征数量（含结点ID）	1415	3658	590	294	1
时间限制（秒）	100	100	100	200	100
有向图/无向图	无向图	无向图	无向图	有向图	无向图
类别数	7	6	41	20	3

表 1　离线五个图数据集的概况

如图 3 所示，我们发现在图数据集 5 上存在着模型训练不稳定的问题，模型在某个 epoch 上验证集精度显著下降。我们考虑主要是图数据集 5 易于学习，会发生过拟合现象，因此我们在自动化建模过程中需要保证模型的强鲁棒性。

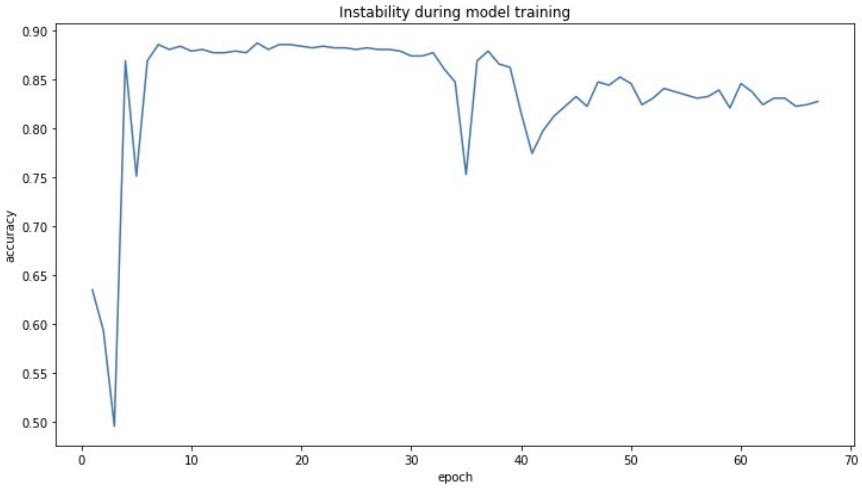


图 3　模型在训练过程中的不稳定性

同时，从下图 4 可以发现，不同于传统的固定数据集评测数据挖掘竞赛，保证多个类型，分布差异大的数据集排名的稳定性相比于优化某个数据集的精度更为重要。例如，数据集 5 模型精度差异仅有 0.15%，却导致了十个名次的差异，数据集 3 模型精度差异有 1.6%，却仅导致 7 个名次的差异，因而我们需要采用排名鲁棒的建模方式来增强数据集排名的稳定性。

<Rank> ▲	Dataset 1 ▲	Dataset 2 ▲	Dataset 3 ▲	Dataset 4 ▲	Dataset 5 ▲
3.2000	0.9291(5)	0.9585(3)	0.9373(1)	0.8846(5)	0.9661(2)
5.0000	0.9261(11)	0.9595(1)	0.9266(3)	0.8849(2)	0.9648(8)
5.0000	0.9338(1)	0.9576(5)	0.9231(4)	0.8845(6)	0.9643(9)
5.8000	0.9243(14)	0.9581(4)	0.9200(6)	0.8847(4)	0.9664(1)
6.2000	0.9295(3)	0.9570(7)	0.9126(10)	0.8846(5)	0.9651(6)
7.0000	0.9268(9)	0.9593(2)	0.9205(5)	0.8832(16)	0.9659(3)
9.6000	0.9238(16)	0.9573(6)	0.9046(12)	0.8850(1)	0.9634(13)
10.2000	0.9315(2)	0.9553(11)	0.9013(24)	0.8847(4)	0.9642(10)
10.6000	0.9276(7)	0.9536(16)	0.9358(2)	0.8846(5)	0.9606(23)
15.2000	0.9226(18)	0.9503(26)	0.9029(19)	0.8845(6)	0.9650(7)

图 4　不同参赛团队在不同数据集上的精度及排名

问题挑战

基于以上数据分析，该赛题中存在以下三个挑战：

- **图数据的多样性**：解决方案要在多个不同的图结构数据上都能达到一个好的效果，图的类型多种多样，包含了有向图 / 无向图、稠密图 / 稀疏图、带特征图 / 无特征图等。
- **超短时间预算**：大部分数据集的时间限制在 100s 左右，在图结构和参数的搜索上需要有一个快速搜索的方案。

- **鲁棒性**：在 AutoML 领域，鲁棒性是非常重要的一个因素，最后一次提交要求选手在之前没见过的数据集上进行自动化建模。

竞赛技术方案

针对以上三个挑战，我们设计了一个自动化图学习框架，如下图 5 所示，我们对输入的图预处理并进行图特征构建。为了克服图的多样性挑战，我们设计了多个图神经网络，每个图神经网络对于不同类型的图有各自的优势。为了克服超短时间预算挑战，我们采用了一个图神经网络结构和超参快速搜索的方法，使用更小的搜索空间以及更少的训练轮数来达到一个更快的搜索速度。为了克服鲁棒性挑战，我们设计了一个多级鲁棒性模型融合策略。最终，我们的自动化图学习解决方案可以在较短的时间内对多个不同图结构数据进行结点分类，并达到鲁棒性效果。接下来，我们将详细地介绍整个解决方案。

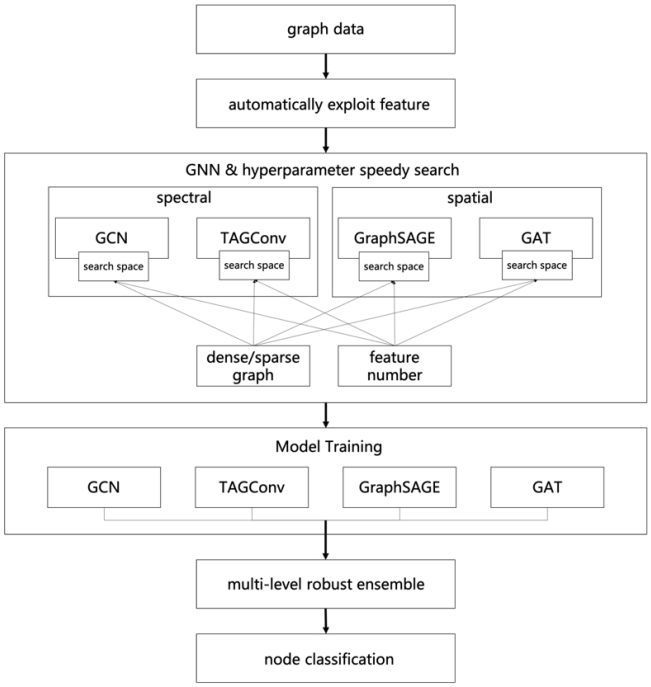


图 5　自动化图学习框架

数据预处理与特征构建

有向图处理：大多数谱域 GNN 方法并不能很好地处理有向图，它们的理论依赖于拉普拉斯矩阵的谱分解，而大多数有向图的邻接矩阵是非对称矩阵，不能直接定义拉普利矩阵及其谱分解。特别的，当一个结点只有入度没有出度时，GCN 等方法并不能有效地获取其邻居信息。由于赛题关注于结点分类而不是链接预测等，考虑大多数图结点分类问题，更为重要的是如何有效地提取图的邻居信息，因而我们将有向图的边进行反转改为无向图，无向图新边的权重与有向图被反转边的权重相等。

特征提取：为了更有效地进行结点的表示学习，提取了一些图的人工特征来让 GNN 进行更好地表示学习，例如结点的度、一阶邻居以及二阶邻居的特征均值等，我们对于数值跨度大的特征进行分桶，对这些特征进行 Embedding，避免过拟合的同时保证了数值的稳定性。

图神经网络模型

为了克服图的多样性挑战，我们结合谱域及空域两类图神经网络方法，采用了 GCN[11]、TAGConv[12]、GraphSAGE[4]、GAT[5] 四个图神经网络模型对多种不同图结构数据进行更好地表示学习，每个模型针对不同类型的图结构数据有各自的优势。

图作为一种非欧式空间结构数据，其邻居结点个数可变且无序，直接设计卷积核是困难的。谱域方法通过图拉普拉斯矩阵的谱分解，在图上进行傅立叶变换得到图卷积函数。GCN 作为谱域的经典方法，公式如下所示，其中 D 是对角矩阵，每个对角元素为对应结点的度，A 是图的邻接矩阵，它通过给每个结点加入自环来使得卷积函数可以获取自身结点信息，图中的 A 帽和 D 帽矩阵即是加自环后的结果，并在傅立叶变换之后使用切比雪夫一阶展开近似谱卷积，使每一个卷积层仅处理一阶邻域信息，可以通过堆叠多个卷积层达到多阶邻域信息传播。GCN 简单且有效，我们将 GCN 应用到所有数据集上，大部分数据集能取得较好的效果。

$$\mathbf{X}' = \hat{\mathbf{D}}^{-1/2}\hat{\mathbf{A}}\hat{\mathbf{D}}^{-1/2}\mathbf{X}\mathbf{\Theta}$$

相较于堆叠多层获取多阶领域信息的 GCN 方法，TAGConv 通过邻接矩阵的多项式拓扑连接来获取多阶领域信息。公式如下所示，可以发现，其通过预先计算邻接矩阵的 k 次幂，相比 GCN 可以在训练过程中实现多阶邻域卷积并行计算，高阶邻域的结果不受低阶邻域结果的影响，从而能加快模型在高阶邻域中的学习。在我们的实验结果上，其在稀疏图上能快速收敛并相比于 GCN 能达到一个更好的效果。

$$\mathbf{X}' = \sum_{k=0}^{K} \mathbf{D}^{-1/2}\mathbf{A}^{k}\mathbf{D}^{-1/2}\mathbf{X}\mathbf{\Theta}_{k}$$

相较于谱域方法利用傅立叶变换来设计卷积核参数，空域方法的核心在于直接聚合邻居结点的信息，难点在于如何设计带参数、可学习的卷积核。GraphSAGE 提出了经典的空域学习框架，其通过图采样与聚合来引入带参数可学习的卷积核，其核心思想是对每个结点采样固定数量的邻居，这样就可以支持各种聚合函数。均值聚合函数的公式如下所示，其中的聚合函数可以替换为最大值聚合，甚至可以替换为带参数的 LSTM 等神经网络。由于 GraphSAGE 带有邻居采样算子，我们引入该图神经网络来极大地加速稠密图的计算。在我们的实验结果上，它在稠密图上的运行时间远小于其他图神经网络，并且能达到一个较好的效果。

$$\mathbf{x}_i' = \mathbf{W}_1\mathbf{x}_i + \mathbf{W}_2 \cdot \mathrm{mean}_{j\in\mathcal{N}(i)}\mathbf{x}_j$$

GAT 方法将 Attention 机制引入图神经网络中，公式如下所示。它通过图结点特征间的 Attention 计算每个结点与其邻居结点的权重，通过权重对结点及其邻居结点进行聚合作为结点的下一层表示。通过 Masked Attention 机制，GAT 能处理可变个数的邻居结点，并且其使用图结点及其邻居结点的特征来学习邻居聚合的权重，能有效利用结点的特征信息来进行图卷积，泛化效果更强，它参考了 Transformer 引入了 Multi-head Attention 来提高模型的拟合能力。GAT 由于利用了结点特征来计算结点与邻居结点间的权重，在带有结点特征的数据集上表现优异，但如果特征维度多就会使得 GAT 计算缓慢，甚至会出现内存溢出的现象，我们需要在特征维度多的情况

下对 GAT 的参数进行搜索限制，要求其在一个参数量更小的空间下搜索。

$$\alpha_{i,j} = \frac{\exp\left(\text{LeakyReLU}\left(\mathbf{a}^{\top}[\mathbf{\Theta}\mathbf{x}_i \parallel \mathbf{\Theta}\mathbf{x}_j]\right)\right)}{\sum_{k\in\mathcal{N}(i)\cup\{i\}}\exp\left(\text{LeakyReLU}\left(\mathbf{a}^{\top}[\mathbf{\Theta}\mathbf{x}_i \parallel \mathbf{\Theta}\mathbf{x}_k]\right)\right)}$$

$$\mathbf{x}'_i = \alpha_{i,i}\mathbf{\Theta}\mathbf{x}_i + \sum_{j\in\mathcal{N}(i)}\alpha_{i,j}\mathbf{\Theta}\mathbf{x}_j$$

超参快速搜索

由于超短时间预算的挑战，我们需要设计一个超参快速搜索方法来保证花较少的时间就可以对每个图模型进行参数搜索，并且在每个数据集上尽可能地使用更多的图模型进行训练和预测。如下图 6 所示，我们将参数搜索分为线下搜索和线上搜索两个部分。

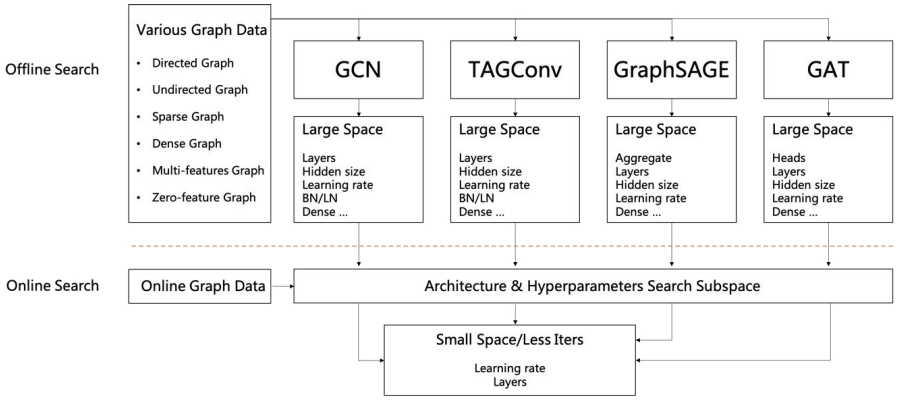


图 6　超参快速搜索

我们在线下搜索时，针对每一个图模型在多个数据集上使用一个大的搜索空间去确定图结构和参数边界，保证每个数据集在这个边界中都有较好的效果。具体地，我们对有向图 / 无向图、稀疏图 / 稠密图、带特征图 / 无特征图等不同图类型都对不同模型的大多数参数进行了搜索，确定了几个重要超参数。例如对于稀疏图，调整

GCN 的层数以及 TAGConv 多项式的阶数，使得其卷积感受野更大，可以迅速对数据集进行拟合，以使得其可以快速收敛；对于特征特别多的图，调整 GAT 的卷积层数、多头自注意力头的数量和隐层神经元个数以使得其训练时间在预算之内并且有较好的效果；对于稠密图，调整 GraphSAGE 的邻居采样，使得其训练可以加速。我们在线下主要确定了不同图模型学习率、卷积层数、隐层神经元个数等这三个重要参数的边界。

由于线上时间预算的限制，我们通过线下的参数边界确定了一个小的参数搜索子空间进行搜索。由于时间预算是相对少的，我们没有充足的时间在参数上做完整的训练验证搜索，因此我们设计了一个快速参数搜索方法。对于每个模型的超参空间，我们通过少量 epochs 的训练来比较验证集精度从而确定超参数。如下图 7 所示，我们通过 16 轮的模型训练来选取验证集精度最优的学习率 0.003，我们的目的是确定哪些超参数可以使得模型快速拟合该数据集，而不追求选择最优的超参数，这样既可以减少超参的搜索时间，也可以减少后续模型训练的时间。通过快速超参搜索，我们保证每个模型在每个数据集上可以在较短内确定超参数，从而利用这些超参数进行每个模型的训练。

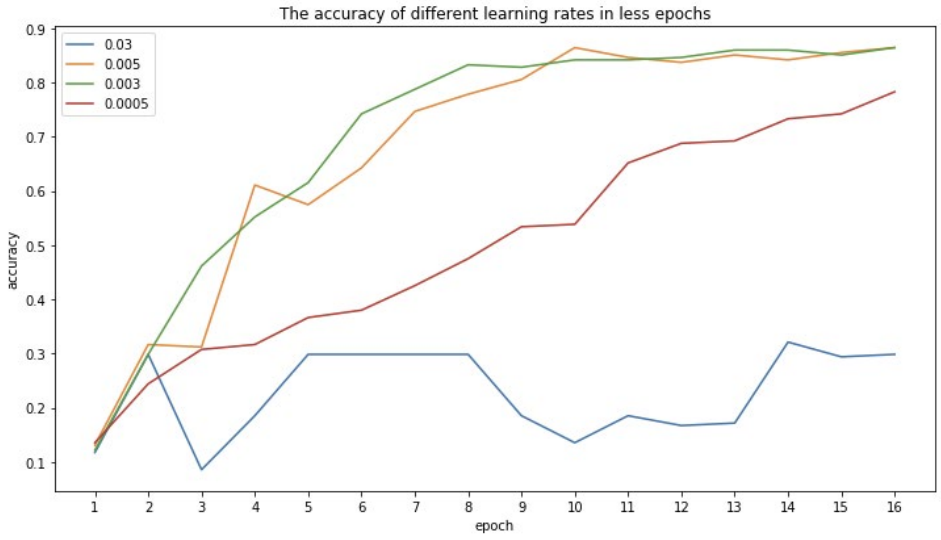


图 7　少量 epochs 模型训练下不同学习率的验证集精度

多级鲁棒模型融合

由于在该次竞赛中是通过数据集排名平均来确定最终排名，故而鲁棒性是特别重要的。为了达到鲁棒效果，我们采用了一个多级鲁棒模型融合策略。如下图 8 所示，我们在数据层面进行切分来进行多组模型训练，每组模型包含训练集及验证集，通过验证集精度使用 Early Stopping 来保证每个模型的鲁棒效果。每组模型包括多种不同的图模型，每种图模型训练进行 n-fold bagging 进行融合来取得稳定效果。不同种类的图模型由于验证精度差异较大，我们需要对不同种类的图模型进行稠密度自适应带权融合来利用不同模型在不同数据集上的差异性。最后，我们再对每组图模型进行均值融合来利用数据间的差异性。

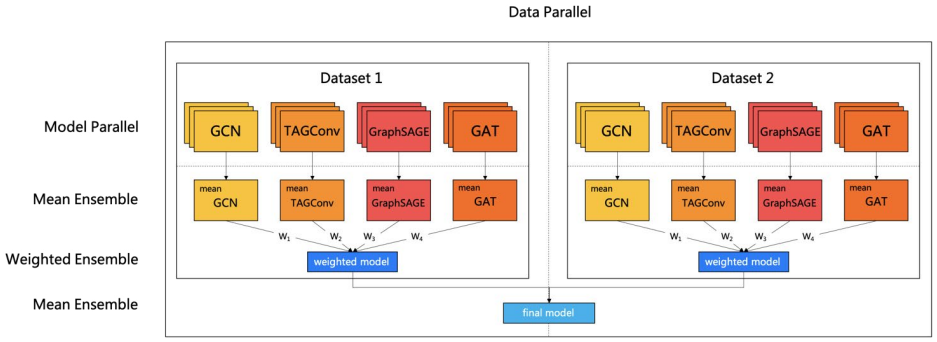


图 8　多级鲁棒模型融合

稠密度自适应带权融合：如图 4 所示，由于某些图数据集较为稀疏且无特征太容易拟合，选手间精度相差小但是排名差异却较大。例如，数据集 5 模型精度差异仅有 0.15%，却导致了十个名次的差异，数据集 3 模型精度差异有 1.6%，却仅导致 7 个名次的差异，因而我们对于多种图模型采用了稠密度自适应的融合方式。

融合权重如以下公式所示，其中 #edges 为边的数量，#nodes 为结点数量，则 #edges/#nodes 表示为图的稠密度，acc（Accuracy）为模型验证集精度，alpha、beta、gamma 为超参数，每个模型的权重由 weight 确定。从以下公式可以看出，如果图足够稠密，则我们只需根据模型精度差异去得到模型权重，无需根据稠密度去

自适应调整，参数 alpha 为是否进行稠密度自适应加权的稠密度临界值；如果图足够稀疏，则模型权重与其验证集精度和数据集的稠密度有关，图越稀疏，则模型权重差异越大。这是由于图越稀疏则模型精度差异性越小，但选手间的排名差异却较大，则我们需要给予更好的模型更大的权重来保证排名的稳定性。

$$density = min(\alpha, log(\frac{\#edges}{\#nodes}) + 1)$$

$$acc_i = acc_i - \frac{1}{n}\sum_{i=1}^{n} acc_i$$

$$acc_i = acc_i - min_i(acc)$$

$$weight_i = softmax(\frac{acc_i}{1 + \frac{(1+density)^{\beta}}{\gamma}})$$

评估结果

表 2 所示的是不同图模型在离线五个图数据集上的测试精度，与图神经网络模型章节所描述的特点一致，GCN 在各个图数据集上有较好的效果。而 TAGConv 在稀疏图数据集 1、2、5 有更优异的效果，GraphSAGE 在稠密图数据集 4 上取得最好的效果，GAT 在有特征的数据集 1、2、4 中表现较为良好，而模型融合在每个数据集上都能取得更稳定且更好的效果。

图数据集	1	2	3	4	5
GCN	0.8864	0.7436	0.9437	0.9392	0.8700
TAGC	0.8808	0.7516	0.8831	0.9377	0.8809
GraphSAGE	0.8740	0.7441	0.9355	0.9559	0.8811
GAT	0.8823	0.7471	0.9318	0.9480	0.8753
Ensemble	0.8895	0.7566	0.9487	0.9607	0.8831

表 2　不同图模型在离线五个图数据集上的测试精度

如下表 3 所示，我们的解决方案在每个图数据集上均达到鲁棒性效果，每个数据集的排行均保持较领先的水平，并避免过度拟合，从而在平均排行上取得了第一，最终我们 Aister 团队在 KDD Cup 2020 AutoGraph 赛题道上赢得了冠军。

team	avg rank	dataset 1	dataset 2	dataset 3	dataset 4	dataset 5
aister	4.8	6	4	4	7	3
PASA_NJU	5.2	2	7	8	4	5
qqerret	5.4	15	6	3	2	1
common	6.6	2	1	2	12	16
PostDawn	7.4	8	10	12	1	6

表 3　Top 5 参赛队伍在最后 5 个数据集上所有图数据集的平均排行及在每个图数据集的单独排行

广告业务应用

搜索广告算法团队负责美团与大众点评双平台的搜索广告与筛选列表广告业务，业务类型涉及餐饮、休闲娱乐、丽人、酒店等，丰富的业务类型为算法优化带来很大空间与挑战。在美团丰富的搜索广告业务场景中，结点类型非常丰富，有用户、Query、Ad、地理位置甚至其他细分的组合结点，结点间的边关系也非常多样化，十分适合通过图学习进行建模。我们在搜索广告的触发模块及点击率预估模块进行图学习的深入优化，带来了业务效果的提升。

不仅结点间具有丰富的边关系，每种结点都有丰富的属性信息，比如 Ad 门店包含结构化的店名、品类、地址位置、星级、销量、客单价以及点击购买次数等统计信息。因此，我们的图是一种典型的异构属性图。目前在搜索广告场景下，我们主要关注包含 Query 和 Ad 两类结点的异构属性图。

如下图 9 所示，我们构建包含了 Query 结点和 Ad 结点的图，应用于触发模块与点击率预估模块。目前，该图使用的边关系主要包括以下几种：

- Query-Query Session：用户在一次会话中的多次 Query 提交；
- Query-Query Similarity Mining：基于用户浏览点击日志挖掘的 Query-Query 相关性数据；
- Query-Ad Click：Query 下 Ad 的点击；
- Ad-Ad CoClick：在同一次请求或用户行为序列中，两个 Ad 的共同点击。

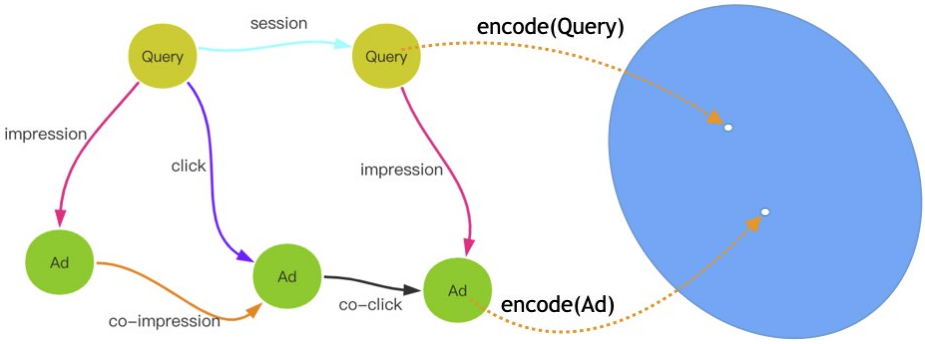


图 9　异构图的构建

图模型在触发模块主要应用于广告 Ad 的向量召回，离线构建 Ad 向量索引，线上实时预估 Query 向量，通过 ANN 检索的方式召回相关性较高的广告 Ad。相比于传统的基于 Bidword 的触发方式，基于图模型的向量化召回在语义相关性及长尾流量上有较明显的优势，通过增加召回率显著提升了广告变现效率。

图 10 所示的是基于图表示多任务学习的触发图网络。我们采用基于 MetaPath 的 Node2Vec 游走生成正例，负例通过全局采样得到。在负例采样时，我们限定负例的品类必须和正例一致，否则由于在特征方面使用了品类特征，模型会轻易地学到使用品类特征区分正负例，弱化了其他特征的学习程度，导致了模型在同品类结点中区分度不好。并且负采样时，使用结点的权重进行 Alias 采样，保证与正例分布一致。为了增强泛化能力解决冷启动问题，我们使用每个结点对应的属性特征而不使用结点 id 特征，这些泛化特征可以有效地缓解冷门结点问题，异构图中未出现的结点，也可以根据它的属性特征，实时预估线上新 Query 或 Ad 的向量。

同时，对于不同结点类型应用不同的深度网络结构，对于 Query 结点，我们采用基于字粒度和词粒度的 LSTM-RNN 网络，Ad 结点采用 SparseEmbedding+MLP 的网络。对于异构边类型，我们希望在模型训练过程中能刻画不同边的影响。对于同一个结点，在不同的边上对应单独的一个深度网络，多个边的深度网络生成的 Embedding 通过 Attention 的方式进行融合，形成结点的最终 Embedding。为了充分利用图的结构信息，我们主要采用 GraphSage 中提出的结点信息汇聚方式。在本结点生成向量的过程中，除了利用本结点的属性特征外，也使用了邻居聚合向量作为特征输入，提升模型的泛化能力。

另外，在美团 O2O 场景下，用户的访问时刻、地理位置等 Context 信息非常重要。因此，我们尝试了图模型和双塔深度模型的多目标联合训练，其中双塔模型使用了用户浏览点击数据，其中包含丰富的 Context 信息。Query 首先经过图模型得到 Context 无关的静态向量，然后与 Context 特征 Embedding 拼接，经过全连接层得到 Context-Aware 的动态 Query 向量。

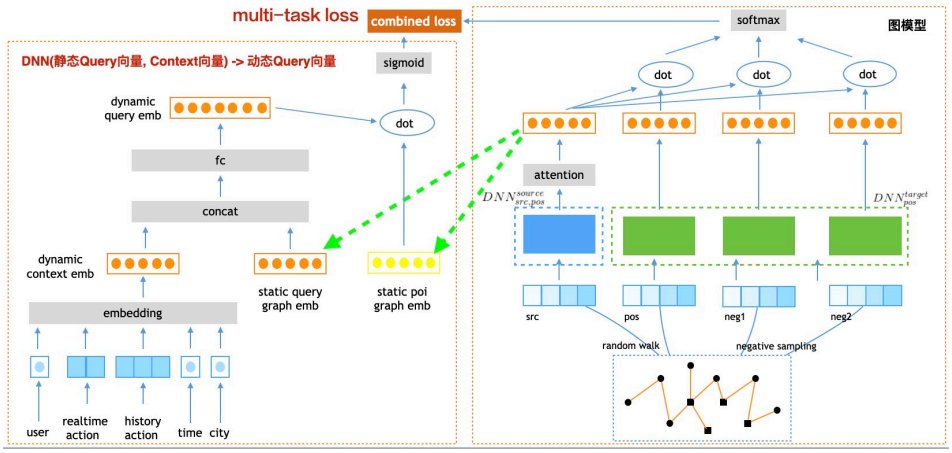


图 10　基于图表示学习的触发网络

在点击率预估模块，相较于侧重于相关性建模的触发模块，更侧重于用户个性化的表达。图结构数据可对用户行为序列进行补充、扩建，起到挖掘用户潜在多峰兴趣的效果，从而提高用户点击率。我们通过在 DSIN（Deep Session Interest Network）网络中引入图神经网络，将更为发散的用户兴趣扩充引入 Session 结构化建模。全局的图结构信息不仅有效扩展了用户潜在兴趣点，并且 GNN Attention 机制可以将目标 Ad 与图中潜在兴趣 Ad 信息结合，进一步挖掘出用户的目标兴趣。

如图 11 所示，对于任意用户行为序列，序列中每一个 Ad，都可以在 Ad 图中进行邻接点遍历，得到其兴趣接近的其余 Ad 表达；用户行为序列是用户的点击序列，可视为用户兴趣的显示表达；经过 Ad 图拓展得到的序列，是行为序列在图数据中最相似的 Ad 组成的序列，可视为用户潜在兴趣的表达。用户原始行为序列的建模，目前基线采用 DSIN 模型；拓展序列的建模，则采用图神经网络的相关方法，利用 GNN attention 处理得到兴趣向量，并和目标 Ad 交叉。我们的实验显示，在 DSIN 基线模型的基础上，拓展序列还能进一步取得精度提升。

未来，我们还会进一步探索图模型在点击率模块中的应用，包括基于用户意图的图模型等。

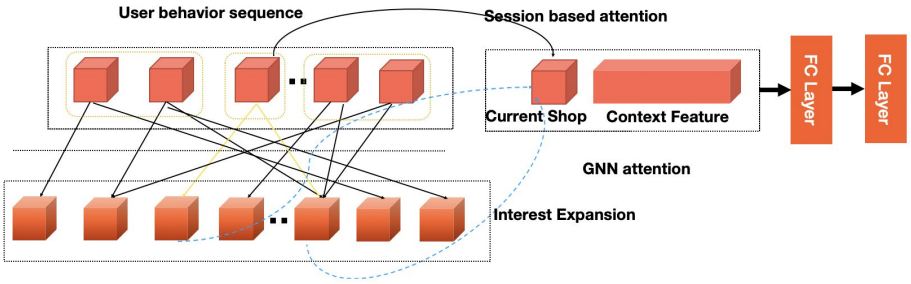


图 11　基于图神经网络的个性化预估网络

总结与展望

KDD Cup 是同工业界联接非常紧密的一项国际比赛，每年赛题紧扣业界热点问题与实际问题，而且历年产出的 Winning Solution 对工业界也都有很大的影响。例如，KDD Cup 2012 产出了 FFM（Feild-Aware Factorization Machine）与 XGBoost 的原型，在工业界已经取得了非常广泛的应用。

今年的 KDD Cup 主要关注在自动化图表示学习以及推荐系统等领域上，图表示学习在近年来既是学术界的热点，也被工业界广泛应用。而 AutoML 领域则致力于探索机器学习端到端全自动化，将 AutoML 与图表示学习两大研究热点相结合，有助于节省在图上进行大量探索的人工成本，解决了复杂度较高的图网络调优问题。

本文介绍了搜索广告算法团队 KDD Cup 2020 AutoGraph 赛题的解决方案，通过对所给的离线数据集进行数据分析，我们定位了赛题的三个主要挑战，采用了一个自动化图学习框架，通过多种图神经网络的结合解决了图数据的多样性挑战，通过超参快速搜索方法来保证自动化建模方案的运行时间在预算之内，以及采用了多级鲁棒模型融合策略来保证在不同类型数据集的鲁棒性。同时，也介绍我们在美团搜索广告触发模块以及点击率预估模块上关于图学习的业务应用，这次比赛也让我们对自动化图表示学习的研究方向有了更进一步的认知。在未来的工作中，我们会基于本次比赛取得的经验进一步优化图模型，并尝试通过 AutoML 技术优化广告系统，解决系统中难以人工遍历的模型优化与特征优化等问题。

参考文献

[1] Wu Z, Pan S, Chen F, et al. A comprehensive survey on graph neural networks[J]. IEEE Transactions on Neural Networks and Learning Systems, 2020.

[2] Perozzi B, Al-Rfou R, Skiena S. Deepwalk: Online learning of social representations[C]//Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining. 2014: 701-710.

[3] Grover A, Leskovec J. node2vec: Scalable feature learning for networks[C]// Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining. 2016: 855-864.

[4] Hamilton W, Ying Z, Leskovec J. Inductive representation learning on large graphs[C]//Advances in neural information processing systems. 2017: 1024-1034.

[5] Veličković P, Cucurull G, Casanova A, et al. Graph attention networks[J]. arXiv preprint arXiv:1710.10903, 2017.

[6] He X, Zhao K, Chu X. AutoML: A Survey of the State-of-the-Art[J]. arXiv preprint arXiv:1908.00709, 2019.

[7] Elsken T, Metzen J H, Hutter F. Neural architecture search: A survey[J]. arXiv preprint arXiv:1808.05377, 2018.

[8] Zhou K, Song Q, Huang X, et al. Auto-gnn: Neural architecture search of graph neural networks[J]. arXiv preprint arXiv:1909.03184, 2019.

[9] Gao Y, Yang H, Zhang P, et al. Graphnas: Graph neural architecture search with reinforcement learning[J]. arXiv preprint arXiv:1904.09981, 2019.

[10] Zhang C, Ren M, Urtasun R. Graph hypernetworks for neural architecture search[J]. arXiv preprint arXiv:1810.05749, 2018.

[11] Kipf T N, Welling M. Semi-supervised classification with graph convolutional networks[J]. arXiv preprint arXiv:1609.02907, 2016.

[12] Du J, Zhang S, Wu G, et al. Topology adaptive graph convolutional networks[J]. arXiv preprint arXiv:1710.10370, 2017.

作者简介

坚强，胡可，金鹏，雷军，均来自美团广告平台搜索广告算法团队。
唐兴元，中国科学院大学。

关于美团 AI

美团 AI 以“帮人们吃得更好，生活更好”为核心目标，致力于在实际业务场景需求上探索前沿的人工智能技术，并将之迅速落地在实际生活服务场景中，完成线下经济的数字化。 美团 AI 诞生于美团丰富的生活服务场景需求之上，具有场景驱动技术的独特性与优势。以业务场景与丰富数据为基础，通过图像识别、语音交互、自然语言处理、配送调度技术，落地于无人配送、无人微仓、智慧门店等真实场景下，覆盖人们生活的方方面面，用科技助力用户生活质量提升，产业智能化升级乃至整个社会的生活服务新基建建设。

更多信息请访问：https://ai.meituan.com/

招聘信息

美团广告平台搜索广告算法团队立足搜索广告场景，探索深度学习、强化学习、人工智能、大数据、知识图谱、NLP 和计算机视觉最前沿的技术发展，探索本地生活服务电商的价值。主要工作方向包括：

- **触发策略**：用户意图识别、广告商家数据理解，Query 改写，深度匹配，相关性建模。
- **质量预估**：广告质量度建模。点击率、转化率、客单价、交易额预估。
- **机制设计**：广告排序机制、竞价机制、出价建议、流量预估、预算分配。
- **创意优化**：智能创意设计。广告图片、文字、团单、优惠信息等展示创意的优化。

岗位要求：

- 有三年以上相关工作经验，对 CTR/CVR 预估，NLP，图像理解，机制设计至少一方面有应用经验。
- 熟悉常用的机器学习、深度学习、强化学习模型。
- 具有优秀的逻辑思维能力，对解决挑战性问题充满热情，对数据敏感，善于分析 / 解决问题。
- 计算机、数学相关专业硕士及以上学历。

具备以下条件优先：

- 有广告 / 搜索 / 推荐等相关业务经验。
- 有大规模机器学习相关经验。

感兴趣的同学可投递简历至：tech@meituan.com（邮件标题请注明：广平搜索团队）。

KDD Cup 2020 多模态召回比赛亚军方案与搜索业务应用

作者：左凯　马潮　东帅　曹佐　金刚　张弓

1. 背景

ACM SIGKDD（ACM SIGKDD Conference on Knowledge Discovery and Data Mining）是世界数据挖掘领域的顶级国际会议。KDD Cup 比赛由 ACM SIGKDD 举办，从 1997 年开始每年举办一次，也是数据挖掘领域最有影响力的赛事之一。该比赛同时面向企业界和学术界，云集了世界数据挖掘界的顶尖专家、学者、工程师、学生等参加，通过竞赛，为数据挖掘从业者们提供了一个学术交流和研究成果展示的理想场所。今年，KDD Cup 共设置四个赛道共五道赛题，涉及数据偏差问题（Debiasing）、多模态召回（Multimodalities Recall）、自动化图学习（AutoGraph）、对抗学习问题和强化学习问题。

美团搜索广告算法团队最终在 Debiasing 赛道中获得冠军（1/1895），在 AutoGraph 赛道中也获得了冠军（1/149）。在 Multimodalities Recall 赛道中，亚军被美团搜索与 NLP 团队摘得（2/1433），美团搜索广告算法团队获得了第三名（3/1433）。

KDD Cup 2020 Challenges for Modern E-Commerce Platform: Multimodalities Recall

Rank	Team	Members
1	WinnieTheBest	Kuei-Chun Huang, Chi-Yu Yang, Ken-Yu Lin
2	MTDP_CVA	Kai Zuo, Chao Ma, Dongshuai Li, Zuo Cao, Xing Xu
3	aister	Jianqiang Huang, Yi Qi, Ke Hu, Bohang Zheng, Mingjian Chen, Xingyuan Tang, Tan Qu, Jun Lei
4	我是余欢水	Tan Yu, Jie Liu, Hongliang Fei, Shujing Wang, Yi Yang, Jinxing Yu, Yi Li, Zhiqiang Xu, Xin Wang, Ping Li
5	烫烫烫烫烫	Donghai Bian, Guangzhi Sheng, Shuai Jiang, Weihua Peng, Yehan Zheng, Qishi Chen, Fayuan Li, Lu Pan, Tianwei Lin
6	WST	Jie Wu, Lei Zhu, Ying Peng
7	NTT DOCOMO LABS	Toshiki Sakai, Yusuke Fukushima, Ryoki Wakamoto, Masato Hashimoto, Katsuaki Kawashima
8	Vaticinator	Qi Zhang, Lixiang Wang, Changyu Li, Peng Zhang, Shengyuan Zheng, Kai Wang, Yudong Xu, Lei Zhong, Chenyu Jin, Jingjie Li
9	ZJU-NESA	Zhe Ma, Xiaoye Qu, Fenghao Liu, Jianfeng Dong, Shouling Ji
10	垫底小分队	Yuhui Ding, Lei Wu, Chengxi Li, Jieyu Yang, Yue Wang

图 1　KDD Cup 2020 Multimodalities Recall 比赛 TOP 10 榜单

跟其它电商公司一样，美团业务场景中除了文本，还存在图片、动图、视频等多种模态信息。同时，美团搜索是典型的多模态搜索引擎，召回和排序列表中存在 POI、图片、文本、视频等多种模态结果，如何保证 Query 和多模态搜索结果的相关性面临着很大的挑战。鉴于多模态召回赛题（Multimodalities Recall）和美团搜索业务的挑战比较类似，本着磨炼算法基本功和沉淀相关技术能力的目的，美团搜索与 NLP 组建团队参与了该项赛事，最终提出的“基于 ImageBERT 和 LXMERT 融合的多模态召回解决方案”最终获得了第二名（2/1433）（KDD Cup2020 Recall 榜单）。本文将介绍多模态召回赛题的技术方案，以及多模态技术在美团搜索场景中的落地应用。

相关代码已经在 GitHub 上开源：https://github.com/zuokai/KDDCUP_2020_MultimodalitiesRecall_2nd_Place。

2. 赛题简介

2019 年，全球零售电子商务销售额达 3.53 万亿美元，预计到 2022 年，电子零售收

入将增长至 6.54 万亿美元。如此快速增长的业务规模表明了电子商务行业的广阔发展前景，但与此同时，这也意味着日益复杂的市场和用户需求。随着电子商务行业规模的不断增长，与之相关的各个模态数据也在不断增多，包括各式各样的带货直播视频、以图片或视频形式展示的生活故事等等。新的业务和数据都为电子商务平台的发展带来了新的挑战。

目前，绝大多数的电子商务和零售公司都采用了各种数据分析和挖掘算法来增强其搜索和推荐系统的性能。在这一过程中，多模态的语义理解是极为重要的。高质量的语义理解模型能够帮助平台更好的理解消费者的需求，返回与用户请求更为相关的商品，能够显著的提高平台的服务质量和用户体验。

在此背景下，今年的 KDD Cup 举办了多媒体召回任务（Modern E-Commerce Platform: Multimodalities Recall），任务要求参赛者根据用户的查询 Query，对候选集合中的所有商品图片进行相关性排序，并找出最相关的 5 个商品图片。举例说明如下：

如图 2 所示，用户输入的 Query 为：

```
leopard-print women's shoes
```

根据其语义信息，左侧图片与查询 Query 是相关的，而右侧的图片与查询 Query 是不相关的。

图 2　多模态匹配示意图

从示例可以看出，该任务是典型的多模态召回任务，可以转化为 Text-Image Matching 问题，通过训练多模态召回模型，对 Query-Image 样本对进行相关性打分，然后对相关性分数进行排序，确定最后的召回列表。

2.1 比赛数据

本次比赛的数据来自淘宝平台真实场景下用户 Query 以及商品数据，包含三部分：训练集（Train）、验证集（Val）和测试集（Test）。根据比赛阶段的不同，测试集又分为 testA 和 testB 两个部分。数据集的规模、包含的字段以及数据样例如表 1 所示。真实样本数据不包含可视化图片，示例图是为了阅读和理解的便利。

比赛数据规模和字段说明				数据示例
数据集		数据量	包含的字段信息	
训练集（Train）		3,000,000	• 图片ID • 图片的长宽 • 目标框的数量 • 目标框的2048维特征	• 图片ID：103059478 • 图片的长宽：800*800 • 目标框的数量：3 • 目标框的2048维特征：Base64编码压缩，此处省略
验证集（Val）		14720		• 目标框的标签：['human face', 'luggage leather goods', 'others'] • 目标框的坐标信息和面积信息(y1,x1,y2,x2,面积占比)：[[0.15375 0.33625 0.35375 0.5075 0.03425] [0.3325 0.33375 0.94 0.84125 0.3083062 5]][0.85375 0.0025 0.94875 0.1225 0.0114]]
测试集（Test）	test A	28830	• 目标框的标签（Label） • 目标框的坐标信息和面积信息	• 用户查询Query：students printed schoolbag valid
	test B	29005	• 用户查询Query • 查询Query的ID	• 查询Query的ID：318

表 1　比赛数据集详情

在数据方面，需要注意的点有：

- 训练集（Train）每条数据代表相关的 Query-Image 样本对，而在验证集（Val）和测试集（Test）中，每条 Query 会有多张候选图片，每条数据表示需要计算相关性的 Query-Image 样本对。
- 赛事主办方已经对所有图片通过目标检测模型（Faster-RCNN）提取了多个目标框，并保存了目标框相应的 2048 维图像特征。因此，在模型中无需再考虑图像特征的提取。

2.2 评价指标

本次比赛采用召回 Top 5 结果的归一化折损累计增益（Normalized Discounted Cumulative Gain，NDCG@5）来作为相关结果排序的评价指标。

3. 经典解法

本次比赛需要解决的问题可以转化为 Text-Image Matching 任务，即对每一个 Query-Image 样本对进行相似性打分，进而对每个 Query 的候选图片进行相关度排序，得到最终结果。多模态匹配问题通常有两种解决思路：

1. 将不同模态数据映射到不同特征空间，然后通过隐层交互这些特征学习到一个不可解释的距离函数，如图 3 (a) 所示。
2. 将不同模态数据映射到同一特征空间，从而计算不同模态数据之间的可解释距离（相似度），如图 3 (b) 所示。

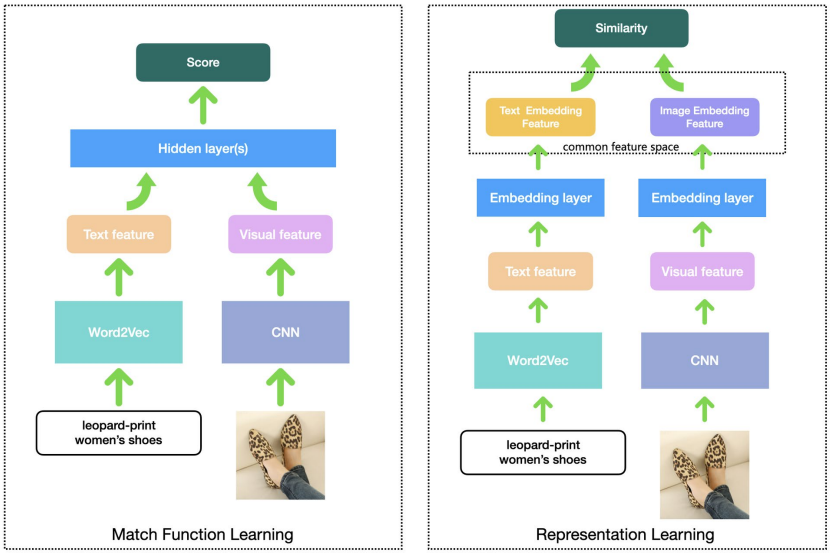


图 3　常用的多模态匹配解决思路

一般而言，同等条件下，由于图文特征组合后可以为模型隐层提供更多的交叉特征信息，因而左侧的模型效果要优于右侧的模型，所以在后续的算法设计中，我们都是围绕图 3 左侧的解决思路展开的。

随着 Goolge BERT 模型在自然语言处理领域的巨大成功，在多模态领域也有越来越多的研究人员开始借鉴 BERT 的预训练方法，发展出融合图片 / 视频（Image/Video）等其他模态的 BERT 模型，并成功应用与多模态检索、VQA、Image Caption 等任务中。因此，考虑使用 BERT 相关的多模态预训练模型（Vision-Language Pre-training, VLP），并将图文相关性计算的下游任务转化为图文是否匹配的二分类问题，进行模型学习。

目前，基于 Transformer 模型的多模态 VLP 算法主要分为两个流派：

- 单流模型，在单流模型中文本信息和视觉信息在一开始便进行了融合，直接一起输入到 Encoder（Transformer）中。典型的单流模型如 ImageBERT [3]，VisualBERT [9]、VL-BERT [10] 等。

- 双流模型，在双流模型中文本信息和视觉信息一开始先经过两个独立的 Encoder（Transformer）模块，然后再通过 Cross Transformer 来实现不同模态信息的融合。典型的双流模型如 LXMERT [4]，ViLBERT [8] 等。

4. 我们的方法：Transformer-Based Ensembled Models TBEM

本次比赛中，在算法方面，我们选用了领域最新的基于 Transformer 的 VLP 算法构建模型主体，并加入了 Text-Image Matching 作为下游任务。除了构建模型以外，我们通过数据分析来确定模型参数，构建训练数据。在完成模型训练后，通过结果后处理策略来进一步提升算法效果。整个算法的流程如下图 4 所示：

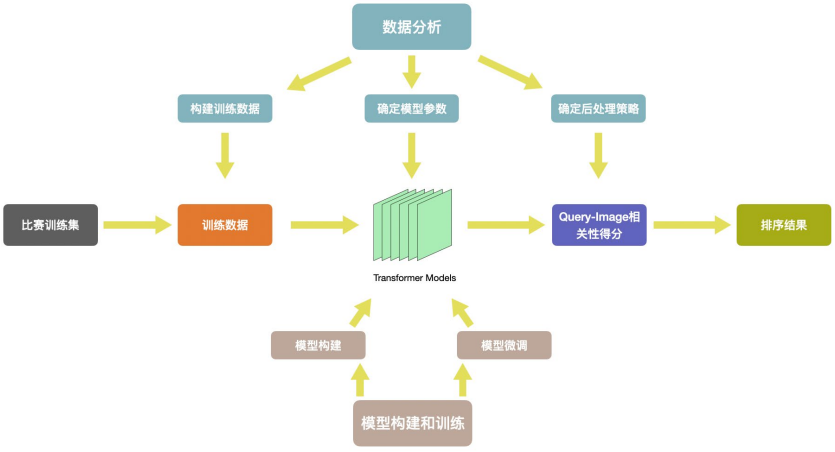


图 4　算法流程图

接下来对算法中的每个环节进行详细说明。

4.1 数据分析 & 处理

数据分析和处理主要基于以下三个方面考虑：

- 正负样本构建：由于主办方提供的训练数据中，只包含了相关的 Query-Image 样本对，相当于只有正样本数据，因此需要通过数据分析，设计策略构建负样本。

- 模型参数设定：在模型中，图片目标框最大数量、Query 文本最大长度等参数需要结合训练数据的分布来设计。
- 排序结果后处理：通过分析 Query 召回的图片数据分布特点，确定结果后处理策略。

常规的训练数据生成策略为：对于每一个 Batch 的数据，按照 1:1 的比例选择正负样本。其中，正样本为训练集（Train）中的原始数据，负样本通过替换正样本中的 Query 字段产生，替换的 Query 是按照一定策略从训练集（Train）中获取。

为了提升模型学习效果，我们在构建负样本的过程中进行了难例挖掘，在构造样本时，通过使正负样本的部分目标框包含同样的类别标签，从而构建一部分较为相似的正负样本，以提高模型对于相似的正负样本的区分度。

难例挖掘的过程如下图 5 所示，左右两侧的相关样本对都包含了“shoes”这一类别标签，使用右侧样本对的 Query 替换左侧图片的 Query，从而构建难例。通过学习这类样本，能够提高模型对于不同类型“shoes”描述的区分度。

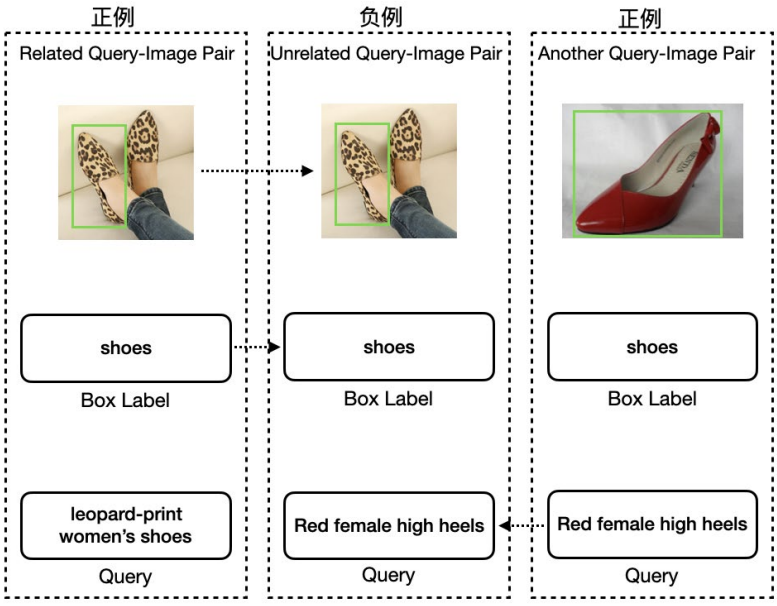


图 5　难例挖掘过程示意图

具体而言，负样本构建策略如表 2 所示：

抽取比例	抽取策略
10%	随机抽取一条Query
40%	抽取一条Query，其对应图片中目标框类别标签与正样本存在重合，类似图5
50%	抽取包含相同实体词的Query，比如sporty men's high-top shoes和 leopard-print women's shoes

表 2　负样本 Query 抽取策略

其次，通过对训练数据中目标框的个数以及 Query 长度的分布情况分析，确定模型的相关参数。图片中目标框的最大个数设置为 10，Query 文本的最大单词个数为 20。后处理策略相关的内容，我们将会在 4.3 部分进行详细的介绍。

4.2 模型构建与训练

4.2.1 模型结构

基于上文中对多模态检索领域现有方法的调研，在本次比赛中，我们分别从单流模型和双流模型中各选择了相应 SOTA 的算法，即 ImageBERT 和 LXMERT。具体而言，针对比赛任务，两种算法分别进行了如下改进：

LXMERT 模型方面主要的改进包括：

- 图片特征部分（Visual Feature）融入了目标框类别标签所对应的文本特征。
- Text-Image Matching Task 中使用两层全连接网络进行图片和文本融合特征的二分类，其中第一个全连接层之后使用 GeLU[2] 进行激活，然后通过 LayerNorm[1] 进行归一化处理。
- 在第二个全连接层之后采用 Cross Entropy Loss 训练网络。

改进后的模型结构如下图 6 所示：

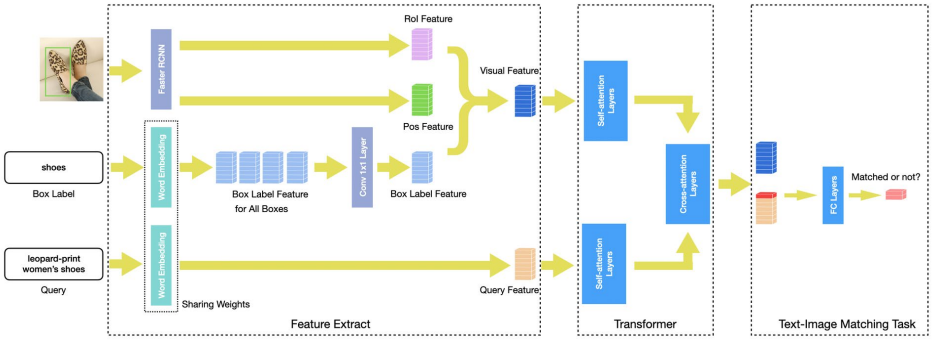


图 6　比赛中使用的 LXMERT 模型结构

特征网络的预训练权重使用了 LXMERT 所提供权重文件，下载地址为：https://github.com/airsplay/lxmert。

ImageBERT：本方案中一共用到了两个版本的 ImageBERT 模型，分别记为 ImageBERT A 和 ImageBERT B，下面会分别介绍改进点。

ImageBERT A：基于原始 ImageBERT 的改进有以下几点。

- 训练任务：不对图片特征和 Query 的部分单词做掩码，仅训练相关性匹配任务，不进行 MLM 等其他任务的训练。
- Segment Embedding：将 Segment Embedding 统一编码为 0，不对图片特征和 Query 文本单独进行编码。
- 损失函数：在 [CLS] 位输出 Query 与 Image 的匹配关系，通过 Cross Entropy Loss 计算损失。

依据上述策略，选用 BERT-Base 模型权重对变量初始化，在此基础上进行 FineTune。其模型结构如下图 7 所示：

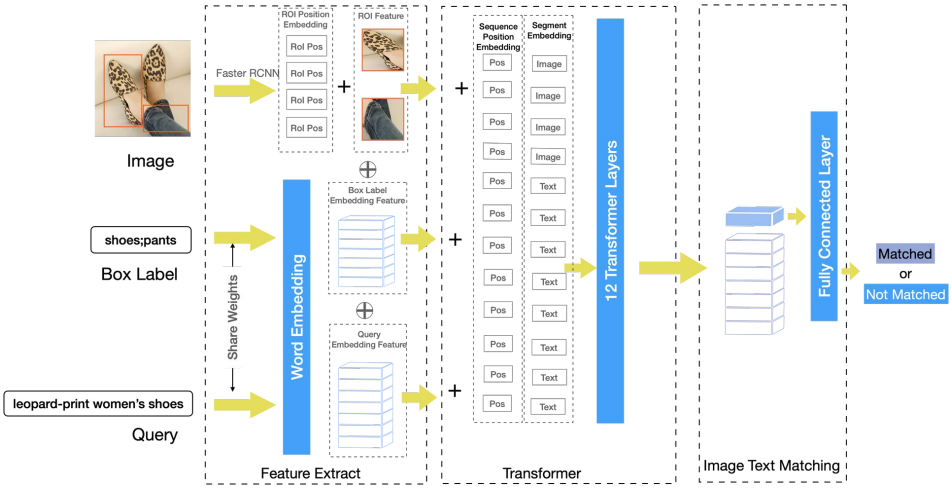


图 7　比赛中使用的 ImageBert 模型结构

ImageBERT B：和 ImageBERT A 的不同点是在 Position Embedding 和 Segment Embedding 的处理上。

- **Position Embedding**：去掉了 ImageBert 中图像目标框位置信息的 Position Embedding 结构。
- **Segment Embedding**：文本的 Segment Embedding 编码为 0，图片特征的 Segment Embedding 编码为 1。

依据上述策略，同样选用 BERT-Base 模型权重对变量初始化，在此基础上进行 FineTune。

三种模型构建中，共性的创新点在于，在模型的输入中引入了图片目标框的标签信息。而这一思路同样被应用在了微软 2020 年 5 月份最新的论文 Oscar [7] 中，但该文的特征使用方式和损失函数设置与我们的方案不同。

4.2.2 模型训练

使用节 4.1 的数据生成策略构建训练数据，分别对上述三个模型进行训练，训练后的模型在验证集（Val）上的效果如表 3 所示。

Model	NDCG@5
LXMERT	0.7049
ImageBERT A	0.6930
ImageBERT B	0.6953

表 3　初步训练后模型在验证集（Val）上的效果

4.2.3 利用损失函数进行模型微调

完成初步的模型训练后，接下来使用不同的损失函数对模型进行进一步的微调，主要有 AMSoftmax Loss [5]、Multi-Similarity Loss [6]。

- AMSoftmax Loss 通过权值归一化和特征归一化，在缩小类内距离的同时增大类间距离，从而提高了模型效果。
- Multi-Similarity Loss 将深度度量学习转化为样本对的加权问题，采用采样和加权交替迭代的策略实现了自相似性，负相对相似性和正相对相似性三种，能够促使模型学习得到更好的特征。

在我们的方案中所采用的具体策略如下：

- 对于 LXMERT，在特征网络后加入 Multi-Similarity Loss，与 Cross Entropy Loss 组成多任务学习网络，进行模型微调。
- 对于 ImageBERT A，使用 AMSoftmax Loss 代替 Cross Entropy Loss。
- 对于 ImageBERT B，损失函数处理方式和 LXMERT 一致。

经过微调，各模型在验证集（Val）上的效果如表 4 所示。

Model	NDCG@5	提升值
LXMERT	0.7094	0.0045
ImageBERT A	0.6980	0.005
ImageBERT B	0.7098	0.0145

表 4　损失函数对模型微调 -- 验证集（Val）上的效果

4.2.4 通过数据过采样进行模型微调

为了进一步提高模型效果，本方案根据训练集（Train）中 Query 字段与测试集（testB）中的 Query 字段的相似程度，对训练集（Test）进行了过采样，采样规则如下：

- 对 Query 在测试集（testB）中出现的样本，或与测试集（testB）中的 Query 存在包含关系的样本，根据其在训练集（Train）出现的次数，按照反比例进行过采样。
- 对 Query 未在测试集（testB）中出现的样本，根据两个数据集 Query 中重复词的个数，对测试集（testB）每条 Query 抽取重复词数目 Top10 的训练集（Train）样本，每条样本过采样 50 次。

数据过采样后，分别对与上述的三个模型按照如下方案进行微调：

- 对于 LXMERT 模型，使用过采样得到的训练样本对 LXMERT 模型进行进一步微调。
- 对于 ImageBERT A 模型，本方案从训练集（Train）选出 Query 中单词和测试集 (Test)Query 存在重合的样本对模型进行进一步微调。
- 对于 ImageBERT B 模型，考虑到训练集（Train）中存在 Query 表达意思相

同，但是单词排列顺序不同的情况，类似”sporty men’s high-top shoes”和”high-top sporty men’s shoes”，为了增强模型的鲁棒性，以一定概率对 Query 的单词（Word）进行随机打乱，对 ImageBERT B 模型进行进一步微调。

训练后各模型在验证集（Val）上的效果如表 5 所示：

Model	NDCG@5	提升值
LXMERT	0.7159	0.0101
ImageBERT A	0.7150	0.0170
ImageBERT B	0.7015	-0.0083

表 5　过采样后验证集（Val）集上的效果

为了充分利用全部有标签的数据，本方案进一步使用了验证集（Val）对模型进行 FineTune。为了避免过拟合，最终提交结果只对 ImageBERT A 模型进行了上述操作。

在 Query-Image 样本对的相关性的预测阶段，本方案对测试集 (testB)Query 所包含的短句进行统计，发现其中“sen department”这一短句在测试集（testB）中大量出现，但在训练集（Train）中从未出现，但出现过“forest style”这个短句。为了避免这组同义短句对模型预测带来的影响，选择将测试集 (testB) 中 Query 的“sen department”替换为“forest style”，并且利用 ImageBERT A 对替换后的测试集进行相关性预测，结果记为 ImageBERT A’。

4.3 模型融合和后处理

经过上述的模型构建、训练以及预测，本方案共得到了 4 个样本对相关性得分的文件。接下来对预测结果进行 Ensemble，并按照一定策略进行后处理，得到 Query 相应的 Image 候选排序集合，具体步骤如下：

（1）在 Ensemble 阶段，本方案选择对不同模型所得相关性分数进行加权求和，作为每一个样本对的最终相关性得分，各模型按照 LXMERT、ImageBERT A、ImageBERT B、ImageBERT A’的顺序的权值为 0.3:0.2:0.3:0.2，各模型的权重利用网格搜索的方式确定，通过遍历 4 个模型的不同权重占比，每个模型权重占比从 0 到 1，选取在 valid 集上效果最优的权重，进行归一化，作为最终权重。

（2）在得到所有 Query-Image 样本对的相关性得分之后，接下来对 Query 所对应的多张候选图片进行排序。验证集（Val）和测试集（testB）的数据中，部分 Image 出现在了多个 Query 的候选样本中，本方案对这部分样本做了进一步处理：

a. 考虑到同一 Image 通常只对应一个 Query，因此认为同一个 Image 只与相关性分数最高的 Query 相关。使用上述策略对 ImageBERT B 模型在验证集（Val）上所得结果进行后处理，模型的 NDCG@5 分数从 0.7098 提升到了 0.7486。

b. 考虑到同一 Image 对应的多条 Query 往往差异较小，其语义也是比较接近的，这导致了训练后的模型对这类样本的区分度较差，较差区分度的相关性分数会一定程度上引起模型 NDCG@5 的下降。针对这种情况我们采用了如下操作：

- 如果同一 Query 的相关性分数中，Top1 Image 和 Top2 Image 相关性分数之差大于一定阈值，计算 NDCG@5 时则只保留 Top 1 所对应的 Query-Image 样本对，删除其他样本对。
- 相反的，如果 Top1 Image 和 Top2 Image 相关性分数之差小于或等于一定阈值，计算 NDCG@5 时，删除所有包含该 Image 的样本对。

使用上述策略对 ImageBERT B 的验证集（Val）结果进行后处理，当选定阈值为 0.92 时，模型的 NDCG@5 分数从 0.7098 提升到了 0.8352。

可以看到，采用策略 b 处理后，模型性能得到了显著提升，因此，本方案在测试集（testB）上，对所有模型 Ensemble 后的相关性得分采用了策略 b 进行处理，得到了最终的相关性排序。

5. 多模态在美团搜索的应用

前面提到过，美团搜索是典型的多模态搜索场景，目前多模态能力在搜索的多个场景进行了落地。介绍具体的落地场景前，先简单介绍下美团搜索的整体架构，美团整体搜索架构主要分为五层，分别为：数据层、召回层、精排层、小模型重排层以及最终的结果展示层，接下来按照搜索的五层架构详细介绍下搜索场景中多模态的落地。

数据层

多模态表示：基于美团海量的文本和图像 / 视频数据，构建平行语料，进行 Image-BERT 模型的预训练，训练模型用于提取文本和图片 / 视频向量化表征，服务下游召回 / 排序任务。

多模态融合：图片 / 视频数据的多分类任务中，引入相关联的文本，用于提升分类标签的准确率，服务下游的图片 / 视频标签召回以及展示层按搜索 Query 出图。

召回层

多模态表示 & 融合：内容搜索、视频搜索、全文检索等多路召回场景中，引入图片 / 视频的分类标签召回以及图片 / 视频向量化召回，丰富召回结果，提升召回结果相关性。

精排层 & 小模型重排

多模态表示 & 融合：排序模型中，引入图片 / 视频的向量化 Embedding 特征，以及搜索 Query 和展示图片 / 视频的相关性特征、搜索结果和展示图片 / 视频的相关性特征，优化排序效果。

展示层

多模态融合：图片 / 视频优选阶段，引入图片 / 视频和 Query 以及和搜索结果的相关

性信息，做到按搜索 Query 出图以及搜索结果出图，优化用户体验。

图 8　多模态在美团搜索的落地场景

6. 总结

在本次比赛中，我们构建了一种基于 ImageBERT 和 LXMERT 的多模态召回模型，并通过数据预处理、结果融合以及后处理策略来提升模型效果。该模型能够细粒度的对用户查询 Query 的相关图片进行打分排序，从而得到高质量的排序列表。通过本次比赛，我们对多模态检索领域的算法和研究方向有了更深的认识，也借此机会对前沿算法的工业落地能力进行了摸底测试，为后续进一步的算法研究和落地打下了基础。此外，由于本次比赛的场景与美团搜索与 NLP 部的业务场景存在一定的相似性，因此该模型未来也能够直接为我们的业务赋能。

目前，美团搜索与 NLP 团队正在结合多模态信息，比如文本、图像、OCR 等，开展 MT-BERT 多模态预训练工作，通过融合多模态特征，学习更好的语义表达，同时

也在尝试落地更多的下游任务，比如图文相关性、向量化召回、多模态特征表示、基于多模态信息的标题生成等。

参考文献

[1] Ba, J. L., Kiros, J. R., and Hinton, G. E. Layer Normalization. arXiv preprint arXiv:1607.06450 (2016).

[2] Hendrycks, D., and Gimpel, K. Gaussian Error Linear Units (GeLUs). arXiv preprint arXiv:1606.08415 (2016).

[3] Qi, D., Su, L., Song, J., Cui, E., Bharti, T., and Sacheti, A. Imagebert: Cross-modal Pre-training with Large-scale Weak-supervised Image-text Data. arXiv preprint arXiv:2001.07966 (2020).

[4] Tan, H., and Bansal, M. LXMERT: Learning Cross-modality Encoder Representations from Transformers. arXiv preprint arXiv:1908.07490 (2019).

[5] Wang, F., Liu, W., Liu, H., and Cheng, J. Additive Margin Softmax for Face Verification. arXiv preprint arXiv:1801.05599 (2018).

[6] Wang, X., Han, X., Huang, W., Dong, D., and Scott, M. R. Multi-similarity Loss with General Pair Weighting for Deep Metric Learning. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2019), pp. 5022 - 5030.

[7] Li X, Yin X, Li C, et al. Oscar: Object-semantics aligned pre-training for vision-language tasks[J]. arXiv preprint arXiv:2004.06165, 2020.

[8] Lu J, Batra D, Parikh D, et al. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks[C]//Advances in Neural Information Processing Systems. 2019: 13-23.

[9] Li L H, Yatskar M, Yin D, et al. Visualbert: A simple and performant baseline for vision and language[J]. arXiv preprint arXiv:1908.03557, 2019.

[10] Su W, Zhu X, Cao Y, et al. Vl-bert: Pre-training of generic visual-linguistic representations[J]. arXiv preprint arXiv:1908.08530, 2019.

[11] 杨扬、佳昊等 . MT-BERT 的探索和实践 . https://tech.meituan.com/2019/11/14/nlp-bert-practice.html

作者简介

左凯，马潮，东帅，曹佐，金刚，张弓等，均来自美团 AI 平台搜索与 NLP 部。

招聘信息

美团搜索与 NLP 部，长期招聘搜索、推荐、NLP 算法工程师，坐标北京 / 上海。欢迎感兴趣的同学发送简历至：tech@meituan.com（邮件注明：搜索与 NLP 部）

CIKM 2020 | 一文详解美团 6 篇精选论文

作者：搜索与 NLP 中心

CIKM 是信息检索、知识管理和数据库领域中顶级的国际学术会议，自 1992 年以来，CIKM 成功汇聚上述三个领域的一流研究人员和开发人员，为交流有关信息与知识管理研究、数据和知识库的最新发展提供了一个国际论坛。大会的目的在于明确未来知识与信息系统发展将面临的挑战和问题，并通过征集和评估应用性和理论性强的顶尖研究成果以确定未来的研究方向。

今年的 CIKM 大会原计划 10 月份在爱尔兰的 Galway 举行，由于疫情原因改为在线举行。美团 AI 平台 / 搜索与 NLP 部 /NLP 中心 / 知识图谱组共有六篇论文（其中 4 篇长文，2 篇短文）被国际会议 CIKM 2020 接收。

这些论文是美团知识图谱组与西安交通大学、中国科学院大学、电子科技大学、中国人民大学、西安电子科技大学、南洋理工大学等高校院所的科研合作成果，是在多模态知识图谱、MT-BERT、Graph Embedding 和图谱可解释性等方向上的技术沉淀和应用。希望这些论文能帮助到更多的同学学习成长。

01《Query-aware Tip Generation for Vertical Search》

| 本论文系美团知识图谱组与西安交通大学郝俊美同学、中国科学院大学李灿佳同学、西安电子科技大学汪自力同学的合作论文。

论文下载

可解释性理由（又称推荐理由）是在搜索结果页和发现页（场景决策、必吃榜单等）展示给用户进行亮点推荐的一句自然语言文本，可以看作是真实用户评论的高度浓缩，为用户解释召回结果，挖掘商户特色，吸引用户点击，并对用户进行场景化引导，辅助用户决策从而优化垂直搜索场景中的用户体验。

(a) Search result page of “Steak”. The upper tip is “*The selected filet steak is enough served.*”, and the bottom one is “*The selected Japanese sirloin steak is fresh, good taste and tender.*”.

(b) An selected channel about “Ramen”. The upper tip is “*Pleasant smell and tasty soup made from pork bones*”, and the bottom one is “*The best Ramen rated by natives.*”.

现有的文本生成工作大部分并未考虑用户的意图信息，这限制了生成式推荐理由在场景化搜索中的落地。本文提出一种 Query 感知的推荐理由生成框架，将用户 Query 信息分别嵌入到生成模型的编码和解码过程中，根据用户 Query 不同会自动生成适配不同场景的个性化推荐理由。本文分别对 Transformer 和递归神经网络（RNN）两种主流模型结构进行了改造。基于 Transformer 结构，本文通过改进 Self-Attention 机制来引入 Query 信息，包括在 Encoder 引入 Query-aware Review Encoder 使得在评论编码最初阶段就开始考虑 Query 相关的信息，在 Decoder 端引入 Query-aware Tip Decoder 使得在评论编码最后阶段考虑 Query 相关的信息。

基于 RNN 结构，在 Encoder 端通过 Selective Gate 方式过滤掉 Query 无关信息，选择原始评论中跟 Query 相关的信息进行编码，并在解码器端将 Query 表示向量加入 Attention 机制的 Context 向量计算，指导解码的过程，一定程度上解决了生成方法解码不可控的问题，从而生成 Query 个性化的推荐理由。

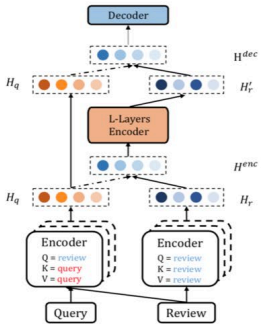


Figure 2: The Transformer-based query-aware tip generation framework. The left encoder encodes review-aware query while the right encoder maintains a self-attention manner to encode review. The two dotted arrows indicate the query information flows to the encoder and the decoder respectively.

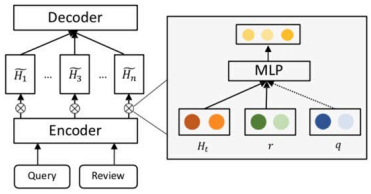


Figure 3: The RNN-based query-aware tip generation framework.

在公开数据集和美团业务数据集上分别进行实验，该论文提出的方法优于现有方法。该论文提出的算法已应用上线，目前在美团的搜索、推荐、类目筛选和榜单等多场景落地。

Table 3: Automatic Evaluation Results on DIANPING and DEBATE datasets.

Group	Methods	DEBATE			DIANPING		
		Semantic	Lexicon	BLEU	Semantic	Lexicon	BLEU
RETRIEVAL	QUERY_LEAD	-	10.23	2.23	-	40.70	23.20
	EXTRACT_BM25	-	14.39	1.12	-	47.18	27.59
	EXTRACT_EMBED	-	**14.43**	1.13	-	37.04	28.29
RNN	RNN	83.87	8.91	11.02	60.08	40.94	40.74
	RNN + QA_ENC	84.37	9.23	15.72	62.65	41.37	48.29
	RNN + QA_DEC	84.17	9.07	15.37	62.77	43.92	46.88
	RNN + BOTH	84.43	9.34	16.58	64.86	44.11	48.38
TRANSFORMER	TRANS	87.17	10.52	30.41	65.64	47.49	48.71
	TRANS + QA_ENC	86.07	13.17	32.03	67.00	49.79	50.39
	TRANS + QA_DEC	84.70	13.46	32.52	62.70	42.61	52.66
	TRANS + BOTH	**88.06**	13.43	**32.93**	**69.79**	**53.75**	**54.20**

02《TABLE: A Task-Adaptive BERT-based ListwisE Ranking Model for Document Retrieval》

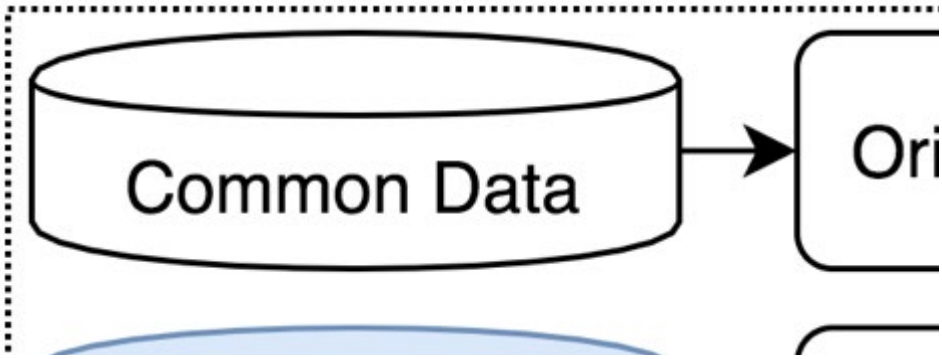
| 本论文系美团知识图谱组与中国科学院软件研究所唐弘胤同学、金蓓弘老师的合作论文。

论文下载

近年来，为了提高模型的自然语言理解能力，越来越多的 MRC 和 QA 数据集开始涌现。但是，这些数据集或多或少存在一些缺陷，比如数据量不够、依赖人工构造 Query 等。针对这些问题，微软提出了一个基于大规模真实场景数据的阅读理解数据集 MS MARCO（Microsoft Machine Reading Comprehension）。该数据集基于 Bing 搜索引擎和 Cortana 智能助手中的真实搜索查询产生，包含 100 万查询、800 万文档和 18 万人工编辑的答案。

基于 MS MARCO 数据集，微软提出了两种不同的任务：一种是给定问题，检索所有数据集中的文档并进行排序，属于文档检索和排序任务；另一种是根据问题和给定的相关文档生成答案，属于 QA 任务。在美团业务中，文档检索和排序算法在搜索、广告、推荐等场景中都有着广泛的应用。此外，直接在所有候选文档上进行 QA 任务的时间消耗是无法接受的，QA 任务必须依靠排序任务筛选出排名靠前的文档，而排序算法的性能直接影响到 QA 任务的表现。基于上述原因，我们主要将精力放在基于 MS MARCO 的文档检索和排序任务上。

自 2018 年 10 月 MACRO 文档排序任务发布后，迄今吸引了包括阿里巴巴达摩院、Facebook、微软、卡内基梅隆大学、清华等多家企业和高校的参与。在美团的预训练 MT-BERT 平台上，我们提出了一种针对该文本检索任务的 BERT 算法方案，称之为 TABLE。值得注意的是，该论文提出的 TABLE 模型在信息检索领域的权威评测微软 MARCO 排行榜上首个超过 0.4% 的模型。



如上图所示，该论文提出了一种基于 BERT 的文档检索模型 TABLE。在 TABLE 的预训练阶段，使用了一种领域自适应策略。在微调阶段，该论文提出了两阶段的任务自适应训练过程，即查询类型自适应的 Pointwise 微调以及 List 微调。实验证明这种任务自适应过程使模型更具鲁棒性。这项工作可以探索查询和文档之间更丰富的匹配特性。因此，该论文显著提升了 BERT 在文档检索任务中的效果。随后在 TABLE 的基础上我们又提出了两个解决 OOV（Out of Vocabulary）错误匹配的方法：精准匹配方法和词还原机制，进一步提升了模型的效果，我们把这个改进后的模型称为 DR-BERT。DR-BERT 的细节详见我们的技术博客：《MT-BERT 在文本检索任务中的实践》。

03《Multi-Modal Knowledge Graphs for Recommender Systems》

| 本论文系美团知识图谱组与中国科学院软件研究所唐弘胤同学、金蓓红老师的合作论文。

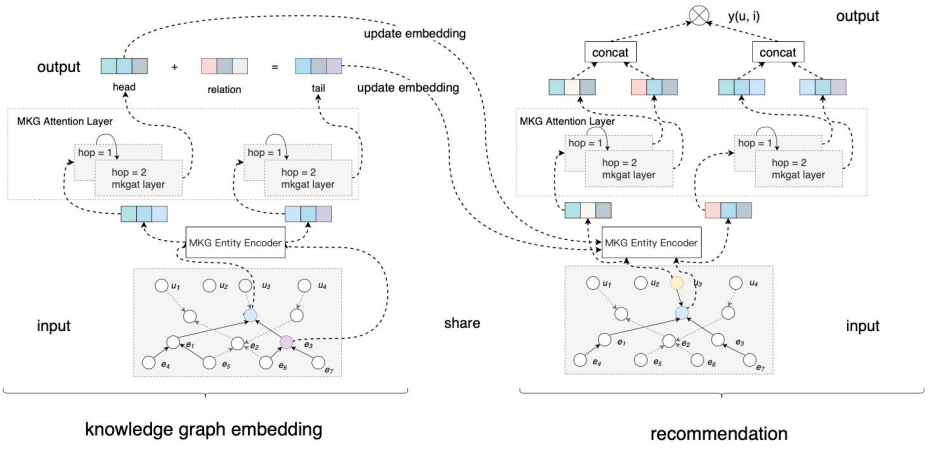
论文下载

随着知识图谱技术发展，其结构化数据被成功的应用在了一系列下游应用当中。在推荐系统方向中，结构化的图谱数据可以利用目标商品更加全面的辅助信息，通过图谱关联进行信息传播，从而有效地对目标商品进行表征建模，缓解推荐系统中用户行为稀疏及冷启动等问题。近年来，已经有不少研究工作利用图谱路径特征、基于图嵌入

的表征学习等方式，成功的将图谱数据和推荐系统进行结合，使得推荐系统准确率得到提升。

在已有的图谱和推荐系统结合的工作当中，人们往往仅关注于图谱节点和节点关系，而没有利用多模态知识图谱中的各个模态的数据进行建模。多模态数据包括图像模态如电影的剧照，文本模态如商户的评论等。这些多模态数据同样可以通过知识图谱图关系进行传播和泛化，并为下游的推荐系统带来高价值的信息。然而，由于多模态知识建模往往是不同模态的辅助信息关系，而非传统图谱中三元组所代表的语义关联关系，故传统的图谱建模方式并不能很好地对多模态知识图谱进行建模。

因此，本文针对多模态知识图谱的特点提出了 MKGAT 模型，首次提出利用多模态知识图谱的结构化信息提升下游推荐系统的预测准确度。MKGAT 的整体模型框架如下图所示：



在 MKGAT 模型中，多模态图谱的嵌入表示学习主要分为三个主要部分：1）我们首先利用多模态实体编码模块（MKG Entity Encoder），将不同类型的输入数据（图像、文本、标签等）编码为高阶隐向量；2）接下来，我们基于多模态图注意力机制模块（MKG Attention Layer），利用实体节点周围的节点（包括多模态及实体节点）来为该节点的刻画提供相应的信息；3）在利用注意力机制综合了多模态信息之后，再利用传统 h+r=t 的训练方法进行图谱嵌入表示学习。

在接入下游推荐系统模型时，我们同样是复用了多模态实体编码和多模态图注意力机制模块对目标实体进行表征，接入推荐系统模型当中。通过上述方法，我们在美团的美食搜索场景和公开数据集 MovieLens 这两个真实数据集上进行了详尽的实验，结果表明在这两个场景中 MKGAT 显著地提高了推荐系统的质量。

04《S^3-Rec: Self-Supervised Learning for Sequential Recommendation with Mutual Information Maximization》

| 本论文系美团知识图谱组与中国人民大学周昆同学、王辉同学、朱余韬同学、赵鑫老师、文继荣老师的合作论文。

论文下载

序列推荐是指利用用户长期的交互历史序列，预测用户未来交互的商品，其通过建模序列信息来增强给用户推荐的准确度。现有的序列推荐模型利用商品预测这一任务来进行模型的参数训练，但是也受限于唯一的训练任务，该类模型很容易受数据稀疏问题影响；它虽然优化的是最终的推荐目标，但是并没有充分地建模上下文数据中的潜在关系，更没有利用该部分信息帮助序列推荐模型。

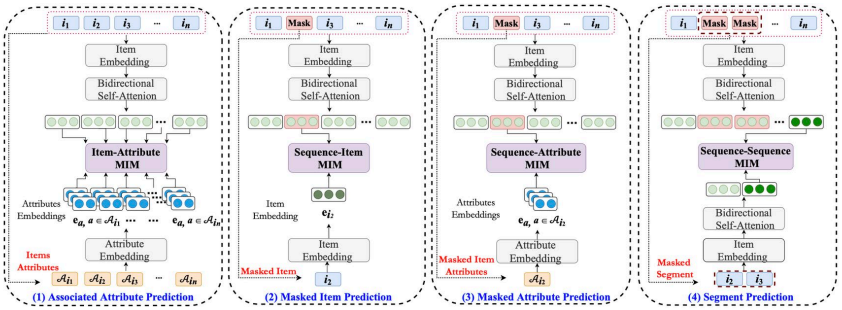


Figure 1: The overview of S^3-Rec in the pre-training stage. We assume that the user sequence is $\{i_1, \cdots, i_n\}$ and each item i is associated with several attributes $\mathcal{A}_i = \{a_1, \cdots, a_m\}$. We incorporate four self-supervised learning objectives: (1) Associated Attribute Prediction (AAP), (2) Masked Item Prediction (MIP), (3) Masked Attribute Prediction (MAP), and (4) Segment Prediction (SP). The embedding layers and bidirectional self-attention blocks are shared by the four pre-training objectives.

为解决以上问题，本文提出了一个新模型 S^3-Rec，它基于自注意力网络结构，采

用自监督学习策略进行表示学习，进而优化序列化推荐任务。该模型基于四种特殊的自监督任务，这些任务分别对属性、商品、自序列和原始序列之间的潜在关系进行学习。由于以上四种信息表示输入数据的四种不同信息粒度视角，本文采用互信息最大化策略来建模这四种信息的潜在关系，进而强化该类数据的表示。本文在包括美团场景的六个真实数据集上进行了大量的实验，以证明该论文提出方法比现有的序列推荐先进方法的优越性，其中在有限的训练数据场景下该模型依旧能保持较好的表现。

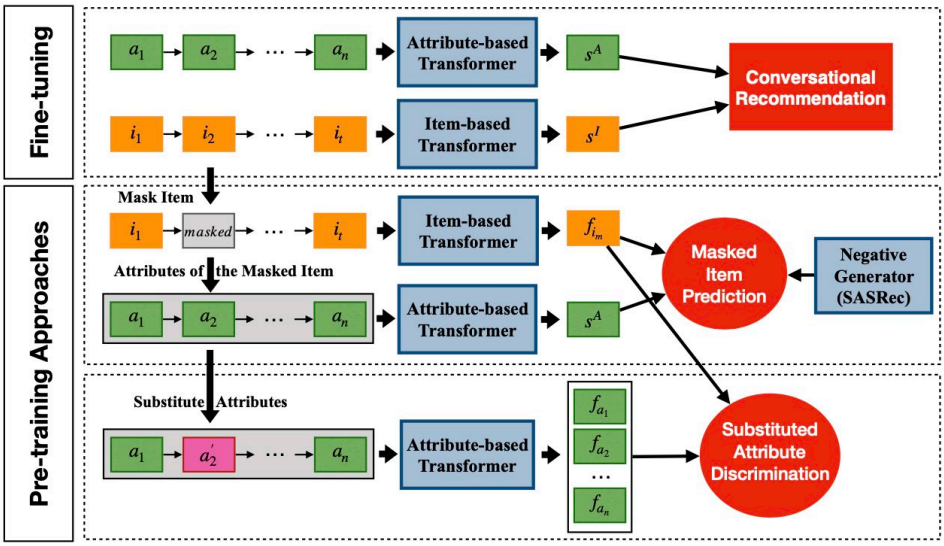
Datasets	Metric	PopRec	FM	AutoInt	GRU4Rec	Caser	SASRec	BERT4Rec	HGN	GRU4Rec$_F$	SASRec$_F$	FDSA	S^3-Rec	Improv.
Meituan	HR@1	0.0946	0.1084	0.0804	0.1194	0.1368	0.1797	0.1381	0.1603	0.1436	0.1746	0.1778	**0.2040***	13.52%
	HR@5	0.2660	0.3218	0.2662	0.3382	0.3812	0.4524	0.3985	0.4110	0.3799	0.4386	0.4595	**0.4925***	7.18%
	NDCG@5	0.1813	0.2170	0.1739	0.2303	0.2619	0.3207	0.2713	0.2887	0.2639	0.3098	0.3236	**0.3527***	8.99%
	HR@10	0.3863	0.4709	0.4077	0.4881	0.5267	0.6053	0.5514	0.5573	0.5378	0.5962	0.6164	**0.6368***	3.31%
	NDCG@10	0.2200	0.2651	0.2194	0.2787	0.3090	0.3700	0.3208	0.3359	0.3149	0.3607	0.3743	**0.3994***	6.71%
	MRR	0.1923	0.2242	0.1854	0.2359	0.2617	0.3146	0.2689	0.2863	0.2666	0.3064	0.3167	**0.3421***	8.02%
Beauty	HR@1	0.0678	0.0405	0.0447	0.1337	0.1337	0.1870	0.1531	0.1683	0.1702	0.1778	0.1840	**0.2192***	17.22%
	HR@5	0.2105	0.1461	0.1705	0.3125	0.3032	0.3741	0.3640	0.3544	0.3727	0.3863	0.4010	**0.4502***	12.27%
	NDCG@5	0.1391	0.0934	0.1063	0.2268	0.2219	0.2848	0.2622	0.2656	0.2759	0.2870	0.2974	**0.3407***	14.56%
	HR@10	0.3386	0.2311	0.2872	0.4106	0.3942	0.4696	0.4739	0.4503	0.4753	0.4843	0.5096	**0.5506***	8.05%
	NDCG@10	0.1803	0.1207	0.1440	0.2584	0.2512	0.3156	0.2975	0.2965	0.3090	0.3185	0.3324	**0.3732***	12.27%
	MRR	0.1558	0.1096	0.1226	0.2308	0.2263	0.2852	0.2614	0.2669	0.2751	0.2844	0.2943	**0.3340***	13.49%
Sports	HR@1	0.0763	0.0489	0.0644	0.1160	0.1135	0.1455	0.1255	0.1428	0.1466	0.1573	0.1585	**0.1841***	16.15%
	HR@5	0.2293	0.1603	0.1982	0.3055	0.2866	0.3466	0.3375	0.3349	0.3547	0.3730	0.3855	**0.4267***	10.69%
	NDCG@5	0.1538	0.1048	0.1316	0.2126	0.2020	0.2497	0.2341	0.2420	0.2535	0.2683	0.2756	**0.3104***	12.63%
	HR@10	0.3423	0.2491	0.2967	0.4299	0.4014	0.4622	0.4722	0.4551	0.4758	0.4912	0.5136	**0.5614***	9.31%
	NDCG@10	0.1902	0.1334	0.1633	0.2527	0.2390	0.2869	0.2775	0.2806	0.2925	0.3064	0.3170	**0.3538***	11.61%
	MRR	0.1660	0.1202	0.1435	0.2191	0.2100	0.2520	0.2378	0.2469	0.2549	0.2680	0.2748	**0.3071***	11.75%
Toys	HR@1	0.0585	0.0257	0.0448	0.0997	0.1114	0.1878	0.1262	0.1504	0.1673	0.1797	0.1717	**0.2003***	6.66%
	HR@5	0.1977	0.0978	0.1471	0.2795	0.2614	0.3682	0.3344	0.3276	0.3695	0.3927	0.3994	**0.4420***	10.67%
	NDCG@5	0.1286	0.0614	0.0960	0.1919	0.1885	0.2820	0.2327	0.2423	0.2719	0.2911	0.2903	**0.3270***	12.33%
	HR@10	0.3008	0.1715	0.2369	0.3896	0.3540	0.4663	0.4493	0.4211	0.4782	0.4981	0.5129	**0.5530***	7.82%
	NDCG@10	0.1618	0.0850	0.1248	0.2274	0.2183	0.3136	0.2698	0.2724	0.3070	0.3252	0.3271	**0.3629***	10.94%
	MRR	0.1430	0.0819	0.1131	0.1973	0.1967	0.2842	0.2338	0.2454	0.2717	0.2886	0.2863	**0.3202***	10.95%
Yelp	HR@1	0.0801	0.0624	0.0731	0.2053	0.2188	0.2375	0.2405	0.2428	0.2293	0.2301	0.2198	**0.2591***	6.71%
	HR@5	0.2415	0.2036	0.2249	0.5437	0.5111	0.5745	0.5976	0.5768	0.5858	0.5937	0.5728	**0.6085***	1.82%
	NDCG@5	0.1622	0.1333	0.1501	0.3784	0.3696	0.4113	0.4252	0.4162	0.4137	0.4178	0.4014	**0.4401***	3.50%
	HR@10	0.3609	0.3153	0.3367	0.7265	0.6661	0.7373	0.7597	0.7411	0.7574	0.7706	0.7555	**0.7725**	0.25%
	NDCG@10	0.2007	0.1692	0.1860	0.4375	0.4198	0.4642	0.4778	0.4695	0.4694	0.4751	0.4607	**0.4934***	3.26%
	MRR	0.1740	0.1470	0.1616	0.3630	0.3595	0.3927	0.4026	0.3988	0.3929	0.3962	0.3834	**0.4190***	4.07%
LastFM	HR@1	0.0725	0.0183	0.0349	0.0642	0.0899	0.1211	0.1220	0.0908	0.1385	0.1147	0.0936	**0.1743***	25.85%
	HR@5	0.1982	0.0954	0.1550	0.1817	0.2982	0.3385	0.3569	0.2872	0.3202	0.3073	0.2624	**0.4523***	26.73%
	NDCG@5	0.1350	0.0552	0.0946	0.1228	0.1960	0.2330	0.2409	0.1896	0.2301	0.2113	0.1766	**0.3156***	31.01%
	HR@10	0.3037	0.1578	0.2596	0.2817	0.4431	0.4706	0.4991	0.4193	0.4670	0.4569	0.4055	**0.5835***	16.91%
	NDCG@10	0.1687	0.0753	0.1285	0.1550	0.2428	0.2755	0.2871	0.2324	0.2775	0.2594	0.2225	**0.3583***	24.80%
	MRR	0.1506	0.0743	0.1122	0.1405	0.2033	0.2364	0.2424	0.1983	0.2410	0.2201	0.1884	**0.3072***	26.73%

05《Leveraging Historical Interaction Data for Improving Conversational Recommender System》

| 本论文系美团知识图谱组与中国人民大学周昆同学、王辉同学、赵鑫老师、文继荣老师的合作论文。

论文下载

近年来，会话推荐系统已经成为了一项重要的研究方向，它在现实生活中也有很多的应用。一个会话推荐系统需要能够通过与用户的对话来了解用户的意图，进而给出合适的推荐，因此它包含一个会话模块和推荐模块。现有的会话推荐系统往往基于学习好的用户表示来完成推荐，这需要对对话内容进行编码。但是实际上仅仅使用对话数据难以准确地预测用户的偏好信息，本论文期望能够通过利用用户的历史交互序列，帮助完成推荐。



基于该设想，会话推荐系统需要同时考虑用户的历史交互序列和会话数据，本论文提出了一种新的预训练方法，通过预训练方法将基于商户的偏好序列（来自历史交互数据）和基于商户属性的偏好序列（来自对话数据）结合起来，提升了会话推荐系统的效果。为了进一步提高性能，该论文还设计了一种负样本生成器，以产生高质量的负样

本来帮助训练。该论文在两个真实数据集上进行了实验，并证明了该方法对改进会话推荐系统是有效的。

Table 2: Performance comparisons of different methods on this task. The number marked with "*" indicates the improvement is statistically significant compared with the best baseline (t-test with p-value < 0.05).

Datasets	Meituan		LastFM	
Models	MRR	NDCG@10	MRR	NDCG@10
CRM	0.0942	0.1009	0.0567	0.0547
EAR	0.0838	0.0869	0.0483	0.0512
GRU_I	0.0964	0.1036	0.0692	0.0734
$SASRec_I$	0.1001	0.1083	0.0783	0.0778
GRU_{I+A}	0.0825	0.0847	0.1034	0.1170
$SASRec_{I+A}$	0.1327	0.1496	0.1231	0.1295
Ours	**0.2026***	**0.2354***	**0.1800***	**0.2032***
-w/o MIP	0.1495	0.1722	0.0828	0.0841
-w/o SAD	0.0627	0.0604	0.1091	0.1216
-w/o NG	0.1760	0.2022	0.1708	0.1926

06《Structural relationship representation learning with graph embedding for personalized product search》

| 本论文系美团知识图谱组与南洋理工大学刘尚同学、丛高老师的合作论文。

论文下载

个性化在商品搜索中非常重要，用户的偏好在很大程度上影响着用户的购买决策。例如，当一个年轻用户在电子商务平台上搜索一件“宽松 T 恤”时，他更有可能购买他感兴趣、有品牌的时尚款式或衬衫。个性化商品搜索（PPS）的目的是针对给定的查询生成用户特有的商品建议，在很多电子商务平台中起着至关重要的作用。

在这项工作中，我们利用从用户 – 查询 – 商品中学习的逻辑结构表示，自然地保留用户 / 查询 / 商品之间的协作信号和交互信息在逻辑路径上，以改进个性化的商品搜索。我们把这些逻辑结构称为“Conjunctive Graph Pattern”。例如，如图 1 所示，有三个关键模式。注意，当分支有三个或更多分支时，我们可以随机抽样其中的两个分支，得到以下模式：

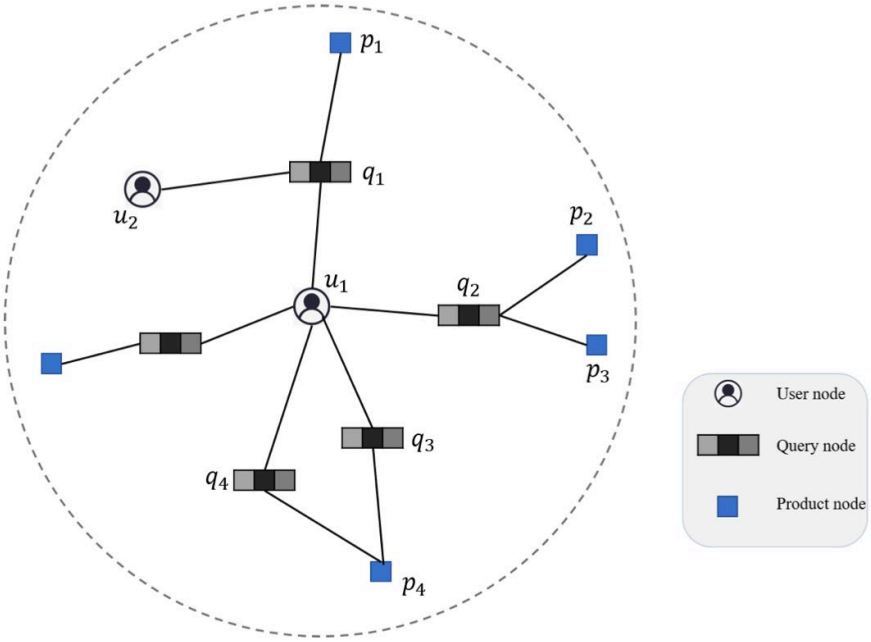


Figure 1: Example of graph construction.

具体来说，我们提出一个新方法：基于逻辑结构表示学习的图嵌入模型（GraphLSR）。GraphLSR 的概念优势在于，它是一个基于嵌入的框架，可以有效地学习逻辑结构的表示，以及用户（查询或商品）在几何操作中的近似关系，并将其整合到个性化的商品搜索中。它背后的关键思想是，我们学习了如何将三种类型的连接图模式嵌入到低维空间中，通过嵌入图来增强个性化商品搜索，框架如图 2 所示，它由两个主要组件组成：图嵌入模块和个性化搜索模块。图 2 下方的图嵌入模块利用设计的三种连接图模式学习嵌入节点进行逻辑表示学习，也便于学习用户（查询或商品）之间的

相似度。然后将表示信息引入个性化搜索模块。

个性化搜索模块以用户、查询、商品以及从图嵌入中学习的表示作为输入，使用多层感知器（MLP）集成相应的信息。将提取出来的用户、查询和商品的短特征和密集特征分别输入到 MLP 网络中，学习用户特有的查询代表和用户特有的商品表示，然后我们将它们一起输入另一个 MLP 来计算预测的概率分数。

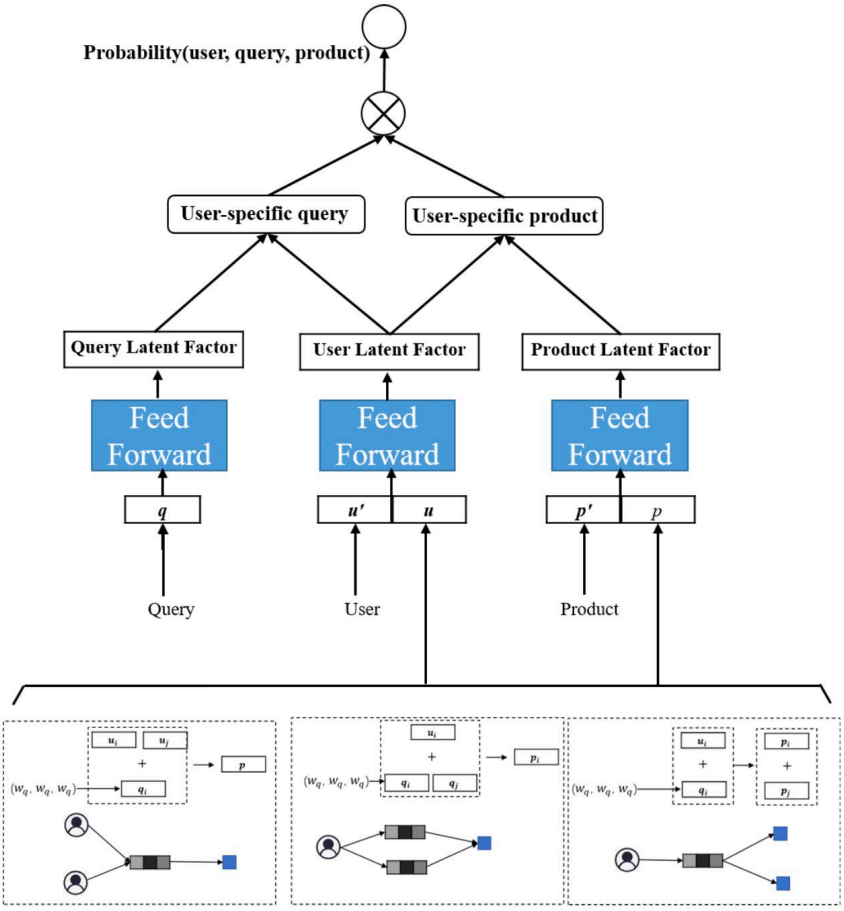


Figure 2: The framework of GraphLSR. The bottom half and top half part illustrate the graph embedding and personalized search module, respectively, which are connected by the transmitting of nodes embedding representation

表 3 比较了 GraphLSR 与四种个性化搜索方法在个性化商品搜索任务中的 MRR、NDCG@10 和 Hit@10 方面的性能：

Table 3: Personalized product search experimental results. The performance of GraphLSR is compared with state-of-the-art product search algorithms, in terms of Hit@10,MRR, and NDCG@10. The best algorithm in each column is highlighted in bold. The last row shows the percentage improvement of Graph@LSR over the best baseline (annotated with * notation). † indicates statistically significant improvement (p < 0.01) by the pairwise t-test over the best baseline. GraphLSR outperforms all the baselines over the four datasets.

	Kindle_Store			Electronics			CIKMCup			Meituan		
	Hit@10	NDCG@10	MRR	Hit@10	NDCG@10	MRR	Hit@10	NDCG@10	MRR	Hit@10	NDCG@10	MRR
LSE	0.008	0.003	0.005	0.109	0.049	0.059	0.001	0.000	0.001	0.004	0.002	0.003
HEM	0.078	0.033	0.039	0.214	0.096	0.110	0.035	0.016	0.022	0.014	0.006	0.009
AEM	0.056	0.023	0.027	0.243	0.109	0.124	0.126	0.052	0.053	0.036	0.016	0.021
DREM	0.258*	0.115*	0.131*	0.309*	0.140*	0.159*	0.233*	0.096*	0.102*	0.128*	0.057*	0.066*
Ours	**0.555**†	**0.267**†	**0.316**†	**0.349**†	**0.161**†	**0.187**†	**0.352**†	**0.143**†	**0.147**†	**0.258**†	**0.115**†	**0.126**†
Gain	115.1%	131.3%	140.3%	13.0%	15.4%	18.0%	50.7%	48.7%	44.3%	102.3%	100.1%	91.8%

总结

以上是搜索与 NLP 部知识图谱组在多模态知识图谱、MT-BERT、Graph-Embedding、图谱可解释性上所做的一些研究工作，论文成果也是我们在实际工作场景中遇到并解决的具体问题，大部分工作已经在实际业务场景如内容搜索、商品搜索、推荐理由等项目上落地，并取得不错的业务收益。美团 AI 平台 / 搜索与 NLP 中心一直致力于通过产研结合，不断将学术成果转化为技术生产力，同时也欢迎更多有志之士加入我们团队。

MT-BERT 在文本检索任务中的实践

作者：兴武

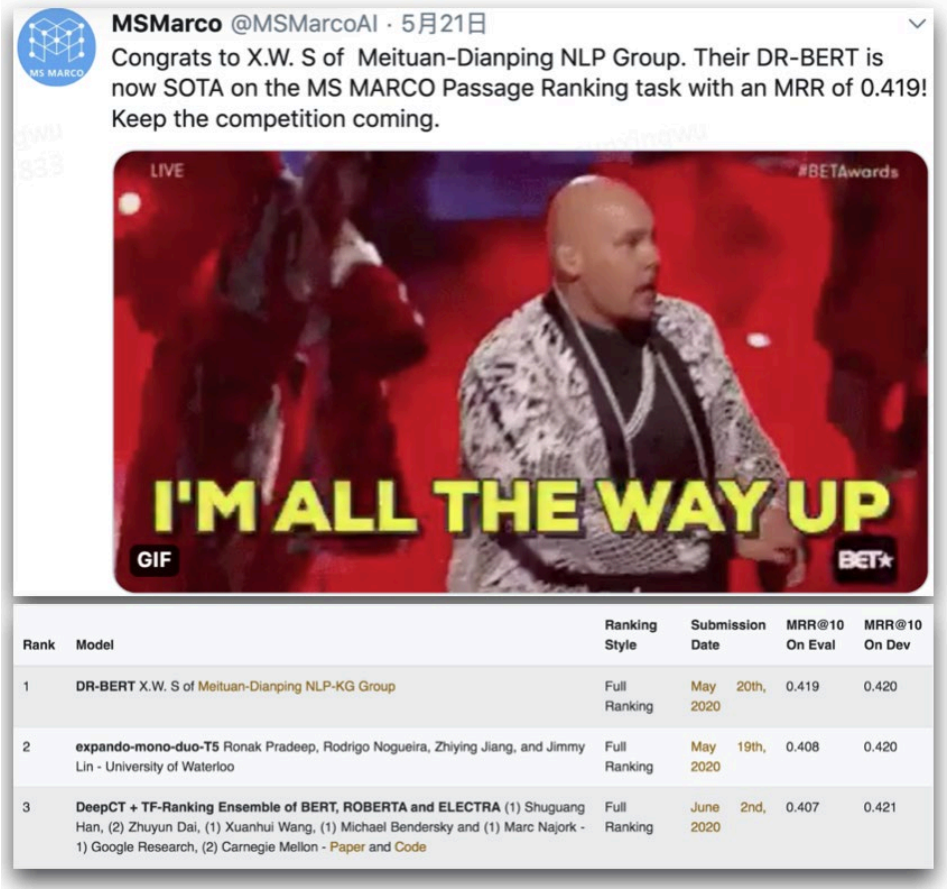
背景

提高机器阅读理解（MRC）能力以及开放领域问答（QA）能力是自然语言处理（NLP）领域的一大重要目标。在人工智能领域，很多突破性的进展都基于一些大型公开的数据集。比如在计算机视觉领域，基于对 ImageNet 数据集研发的物体分类模型已经超越了人类的表现。类似的，在语音识别领域，一些大型的语音数据库，同样使得了深度学习模型大幅提高了语音识别的能力。

近年来，为了提高模型的自然语言理解能力，越来越多的 MRC 和 QA 数据集开始涌现。但是，这些数据集或多或少存在一些缺陷，比如数据量不够、依赖人工构造 Query 等。针对这些问题，微软提出了一个基于大规模真实场景数据的阅读理解数据集 MS MARCO（Microsoft Machine Reading Comprehension）[1]。该数据集基于 Bing 搜索引擎和 Cortana 智能助手中的真实搜索查询产生，包含 100 万查询，800 万文档和 18 万人工编辑的答案。基于 MS MARCO 数据集，微软提出了两种不同的任务：一种是给定问题，检索所有数据集中的文档并进行排序，属于文档检索和排序任务；另一种是根据问题和给定的相关文档生成答案，属于 QA 任务。在美团业务中，文档检索和排序算法在搜索、广告、推荐等场景中都有着广泛的应用。此外，直接在所有候选文档上进行 QA 任务的时间消耗是无法接受的，QA 任务必须依靠排序任务筛选出排名靠前的文档，而排序算法的性能直接影响到 QA 任务的表现。基于上述原因，我们主要将精力放在基于 MS MARCO 的文档检索和排序任务上。

自 2018 年 10 月 MACRO 文档排序任务发布后，迄今吸引了包括阿里巴巴达摩院、Facebook、微软、卡内基梅隆大学、清华等多家企业和高校的参与。在美团的预训练 MT-BERT 平台 [14] 上，我们提出了一种针对该文本检索任务的 BERT 算法方案，称之为 DR-BERT（Enhancing BERT-based Document Ranking Model with

Task-adaptive Training and OOV Matching Method)。DR-BERT 是第一个在官方评测指标 MRR@10 上突破 0.4 的模型，且在 2020 年 5 月 21 日（模型提交日）-8 月 12 日期间位居榜首，主办方也单独发表推文表示了祝贺，如下图 1 所示。DR-BERT 模型的核心创新主要包括领域自适应的预训练、两阶段模型精调及两种 OOV（Out of Vocabulary）匹配方法。



Rank	Model	Ranking Style	Submission Date	MRR@10 On Eval	MRR@10 On Dev
1	**DR-BERT** X.W. S of Meituan-Dianping NLP-KG Group	Full Ranking	May 20th, 2020	0.419	0.420
2	**expando-mono-duo-T5** Ronak Pradeep, Rodrigo Nogueira, Zhiying Jiang, and Jimmy Lin - University of Waterloo	Full Ranking	May 19th, 2020	0.408	0.420
3	**DeepCT + TF-Ranking Ensemble of BERT, ROBERTA and ELECTRA** (1) Shuguang Han, (2) Zhuyun Dai, (1) Xuanhui Wang, (1) Michael Bendersky and (1) Marc Najork - 1) Google Research, (2) Carnegie Mellon - Paper and Code	Full Ranking	June 2nd, 2020	0.407	0.421

图 1　官方祝贺推文及 MARCO 排行榜

相关介绍

Learning to Rank

在信息检索领域，早期就已经存在很多机器学习排序模型（Learning to Rank）用来解决文档排序问题，包括 LambdaRank[2]、AdaRank[3] 等，这些模型依赖很多手工构造的特征。而随着深度学习技术在机器学习领域的流行，研究人员提出了很多神经排序模型，比如 DSSM[4]、KNRM[5] 等。这些模型将问题和文档的表示映射到连续的向量空间中，然后通过神经网络来计算它们的相似度，从而避免了繁琐的手工特征构建。

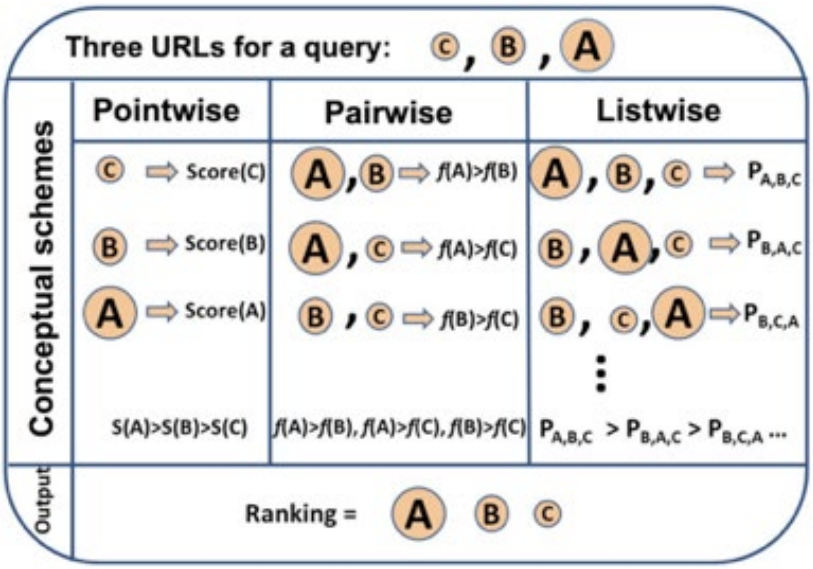


图 2　Pointwise、Pairwise、Listwise 训练的目标

根据学习目标的不同，排序模型大体可以分为 Pointwise、Pairwise 和 Listwise。这三种方法的示意图如上图 2 所示。其中，Pointwise 方法直接预测每个文档和问题的相关分数，尽管这种方法很容易实现，然而对于排序来说，更重要的是学到不同文档之间的排序关系。基于这种思想，Pairwise 方法将排序问题转换为对两两文档的比较。具体来讲，给定一个问题，每个文档都会和其他的文档两两比较，判断该文档是否优于其他文档。这样的话，模型就学习到了不同文档之间的相对关系。

然而，Pairwise 的排序任务存在两个问题：第一，这种方法优化两两文档的比较而非更多文档的排序，跟文档排序的目标不同；第二，随机从文档中抽取 Pair 容易造成训练数据偏置的问题。为了弥补这些问题，Listwise 方法将 Pairwsie 的思路加以延伸，直接学习排序之间的相互关系。根据使用的损失函数形式，研究人员提出了多种不同的 Listwise 模型。比如，ListNet[6] 直接使用每个文档的 top-1 概率分布作为排序列表，并使用交叉熵损失来优化。ListMLE[7] 使用最大似然来优化。SoftRank[8] 直接使用 NDCG 这种排序的度量指标来进行优化。大多数研究表明，相比于 Pointwise 和 Pairwise 方法，Listwise 的学习方式能够产生更好的排序结果。

BERT

自 2018 年谷歌的 BERT[9] 的提出以来，预训练语言模型在自然语言处理领域取得了很大的成功，在多种 NLP 任务上取得了 SOTA 效果。BERT 本质上是一个基于 Transformer 架构的编码器，其取得成功的关键因素是利用多层 Transoformer 中的自注意力机制（Self-Attention）提取不同层次的语义特征，具有很强的语义表征能力。如图 3 所示，BERT 的训练分为两部分，一部分是基于大规模语料上的预训练（Pre-training），一部分是在特定任务上的微调（Fine-tuning）。

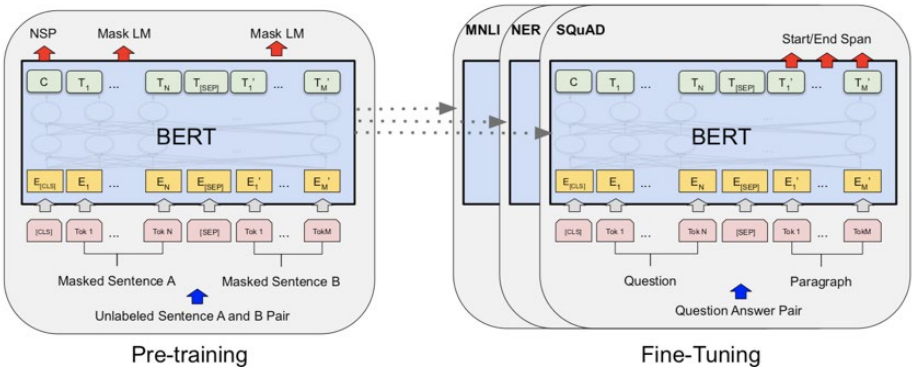


图 3　BERT 的结构和训练模式

在信息检索领域，很多研究人员也开始使用 BERT 来完成排序任务。比如，[10][11] 就使用 BERT 在 MS MARCO 上进行实验，得到的结果大幅超越了当时最好的神经网

络排序模型。[10] 使用了 Pointwise 学习方式，而 [11] 使用了 Pairwise 学习方式。这些工作虽然取得了不错的效果，但是未利用到排序本身的比较信息。基于此，我们结合 BERT 本身的语义表征能力和 Listwise 排序，取得了很大的进步。

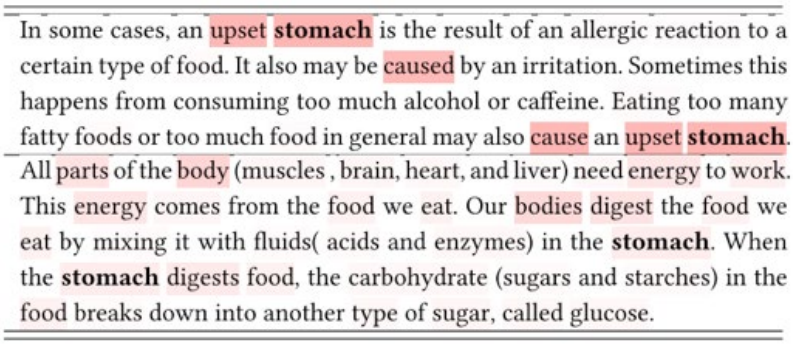
模型介绍

任务描述

给定问题Q以及大规模的文档候选集合D^*，我们的任务是根据文档和问题的相关度产生一个排序，使得排序结果尽量接近目标排序结果$y = y_i{}_{i=1}^{n}$，其中$y_i \in 0, 1$。

基于 DeepCT 候选初筛

由于 MS MARCO 中的数据量很大，直接使用深度神经网络模型做 Query 和所有文档的相关性计算会消耗大量的时间。因此，大部分的排序模型都会使用两阶段的排序方法。第一阶段初步筛选出 top-k 的候选文档，然后第二阶段使用深度神经网络对候选文档进行精排。这里我们使用 BM25 算法来进行第一步的检索，BM25 常用的文档表示方法包括 TF-IDF 等。但是 TF-IDF 不能考虑每个词的上下文语义。DeepCT[12] 为了改进这种问题，首先使用 BERT 对文档单独进行编码，然后输出每个单词的重要性程度分数。通过 BERT 强大的语义表征能力，可以很好衡量单词在文档中的重要性。如下图 4 所示，颜色越深的单词，其重要性越高。其中的“stomach”在第一个文档中的重要性更高。


In some cases, an upset **stomach** is the result of an allergic reaction to a certain type of food. It also may be caused by an irritation. Sometimes this happens from consuming too much alcohol or caffeine. Eating too many fatty foods or too much food in general may also cause an upset **stomach**.

All parts of the body (muscles , brain, heart, and liver) need energy to work. This energy comes from the food we eat. Our bodies digest the food we eat by mixing it with fluids(acids and enzymes) in the **stomach**. When the **stomach** digests food, the carbohydrate (sugars and starches) in the food breaks down into another type of sugar, called glucose.

图 4　DeepCT 估单词的重要性，同一个词在不同文档中的重要性不同

DeepCT 的训练目标如下所示：

$$QTR(t,d) = \frac{|Q_{d,t}|}{|Q_d|}$$

其中$QTR(t,d)$表示文档d中单词t的重要性分数，Q_d表示和文档d相关的问题，$Q_{d,t}$表示文档d对应的问题中包含单词t的子集。输出的分数可以当做词频（TF）使用，相当于对文档的词的重要性进行了重新估计，因此可以直接使用BM25算法进行检索。我们使用DeepCT作为第一阶段的检索模型，得到top-k个文档作为文档候选集合$D = \{D_1, D_2, \ldots, D_k\}$。

领域自适应预训练

由于我们的模型是基于 BERT 的，而 BERT 本身的预训练使用的语料和当前的任务使用的语料并不是同一个领域。我们得出这个结论是基于对两部分语料中 top-10000 高频词的分析，我们发现 MARCO 的 top-10000 高频词和 BERT 基线使用的语料有超过 40% 的差异。因此，我们有必要使用当前领域的语料对 BERT 进行预训练。由于 MS MARCO 属于大规模语料，我们可以直接使用该数据集中的文档内容对 BERT 进行预训练。我们在第一阶段使用 MLM 和 NSP 预训练目标函数在 MS MARCO 上进行预训练。

两阶段精调

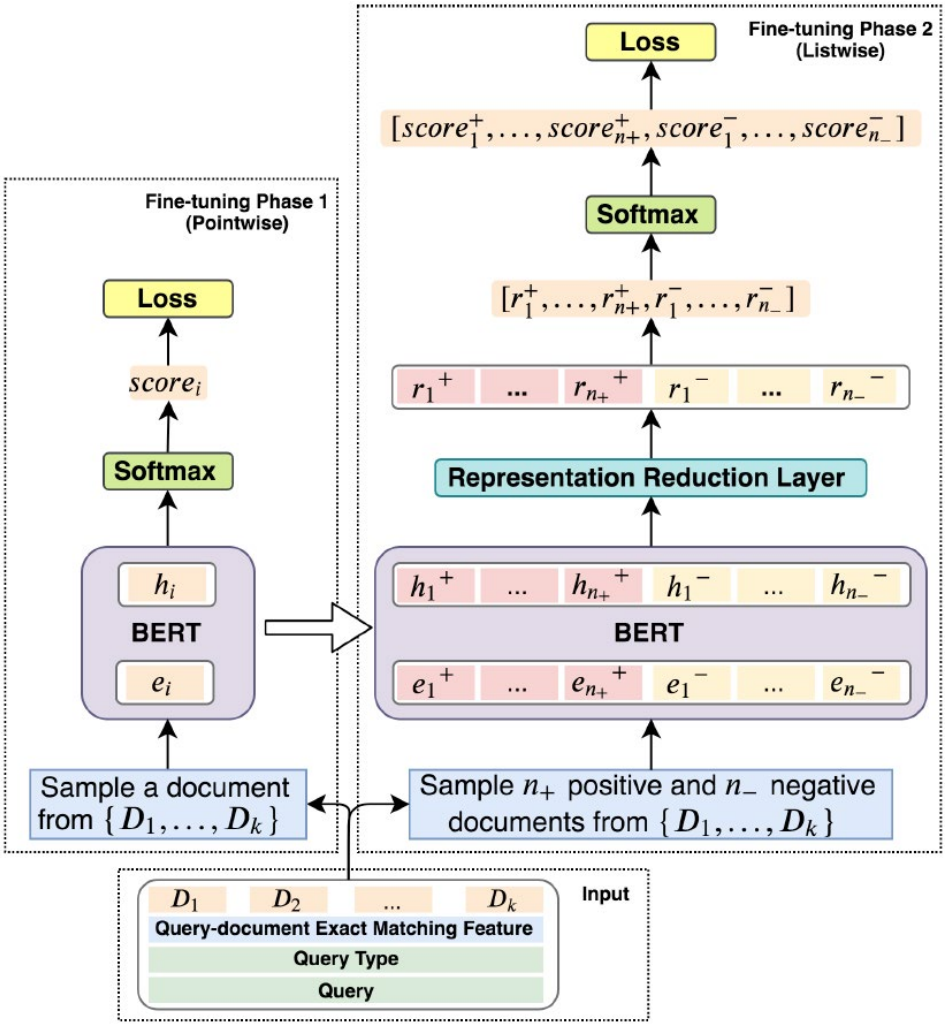


图 5　模型结构

下面介绍我们提出的精调模型，上图 5 展示了我们提出的模型的结构。精调分为两个阶段：Pointwise 精调和 Listwise 精调。

Pointwise 问题类型感知的精调

第一阶段的精调，我们的目标是通过 Pointwise 的训练方式建立问题和文档的关系。

我们将 Query-Document 作为输入，使用 BERT 对其编码，匹配问题和文档。考虑到问题和文档的匹配模式和问题的类型有很大的关系，我们认为在该阶段还需要考虑问题的类型。因此，我们使用问题，问题类型和文档一起通过 BERT 进行编码，得到一个深层交互的语义表示。具体的，我们将问题类型 T, 问题 Q 和第 i 个文档 Di 拼接成一个序列输入，如下式所示：

$$X_i = [< CLS >, T, < SEP >, Q, < SEP >, D_i]$$

其中$< SEP >$表示分隔符，$< CLS >$的位置对应的编码表示Query-Document的关系。其中每个Token x_i^j的Embedding e_i^j由如下三部分组成：

$$e_i^j = e_{tok_i}^j + e_{seg_i}^j + e_{pos_i}^j$$

其中$e_{tok}{}_i^j$，$e_{seg}{}_i^j$，$e_{pos}{}_i^j$ 分别表示第i个文档中第j个Token的Token Embedding、Segment Embedding以及Position Embedding。经过BERT编码后，我们取最后一层中$< CLS >$位置的表示h_i为Query-Document的关系表示。然后通过$Softmax$计算他们的得分，得到：

$$r_i = Softmax(h_i)$$

该分数 ri 通过交叉熵损失函数进行优化。通过以上的预训练，模型对不同的问题学到了不同的匹配模式。该阶段的预训练可以称为类型自适应（Type-Adaptive）模型精调。

Listwise 精调

为了使得模型直接学习不同排序的比较关系，我们通过 Listwise 的方式对模型进行精调。具体的，在训练过程中，对于每个问题，我们采样 n+ 个正例以及 n- 个负例作为输入，这些文档是从候选文档集合 D 中随机产生。注意，由于硬件的限制，我们不能将所有的候选文档都输入到当前模型中。因此我们选择了随机采样的方式来进行训练。

和预训练中使用BERT的方式类似，我们得到正例和负例中每个文档的表示，h_i^+和h_i^-。然后通过一个单层感知机将上面得到的表示降维并转换成一个分数，即

$$r_i = Wh_i + b$$

其中W和b是模型中可学习的参数。接下来对于每个文档的分数，我们通过一个文档级别的比较和归一化得到：

$$\text{score}_i = \frac{e^{r_i}}{\sum_{j=1}^{n_+ + n_-} e^{r_j}}$$

这一步，我们将文档中的正例的分数和负例的分数进行比较，得到Listwise的排名分数。我通过这一步，我们得到了一个文档排序列表，我们可以将文档排序的优化转化为最大化正例的分数。因此，模型可以通过负对数似然损失优化，如下式所示：

$$\mathcal{L} = \frac{\sum_{i=1}^{n_+} -\log(\text{score}_i)}{n_+}$$

至于为什么使用两个阶段的精调模型，主要出于如下两点考虑：

1. 我们发现首先学习问题和文档的相关性特征然后学习排序的特征，相比直接学习排序特征效果好。
2. MARCO 是标注不充分的数据集合。换句话说，许多和问题相关的文档未被标注为 1，这些噪声容易造成模型过拟合。第一阶段的模型用来过滤训练数据中的噪声，从而可以有更好的数据监督第二阶段的精调模型。

解决 OOV 的错误匹配问题

在 BERT 中，为了减少词表的规模以及解决 Out-of-vocabulary（OOV）的问题，使用了 WordPiece 方法来分词。WordPiece 会把不在词表里的词，即 OOV 词拆分成片段，如图 6 所示，原始的问题中包含词“bogue”，而文档中包含词“bogus”。在 WordPiece 方法下，将“bogue”切分成”bog”和“##ue”，并且将“bogus”切分成”bog”和“##us”。我们发现，“bogus”和“bogue”是不相

关的两个词，但是由于 WordPiece 切分出了匹配的片段“bog”，导致两者的相关性计算分数比较高。

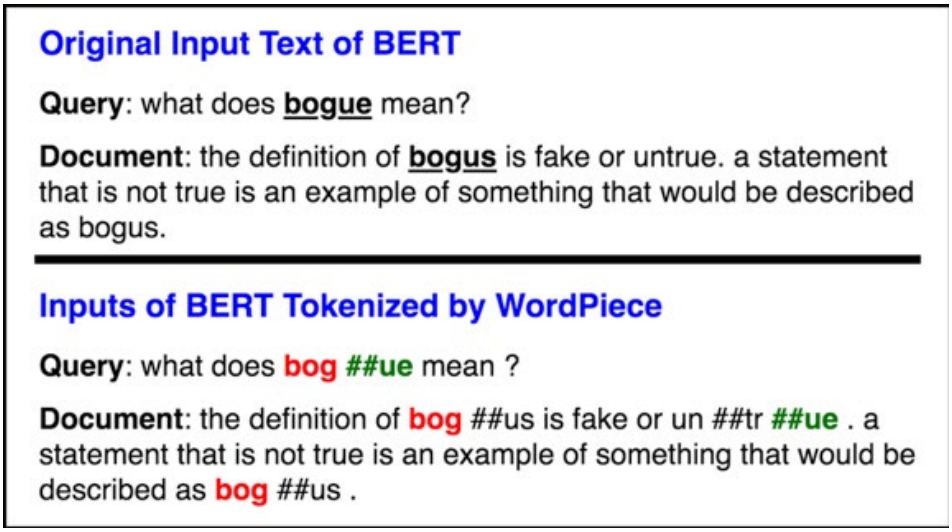


图 6　BERT WordPiece 处理前 / 后的文本

为了解决这个问题，我们提出了一种是对原始词（WordPiece 切词之前）做精准匹配的特征。所谓“精确匹配”，指的是某个词在文档和问题中同时出现。精准匹配是信息检索和机器阅读理解中非常重要的一个技术。根据以往的研究，很多阅读理解模型加入该特征之后都可以有一定的效果提升。具体的，在 Fine-tuning 阶段，我们对于每个词构造了一个精准匹配特征，该特征表示该单词是否出现在问题以及文档中。在编码阶段之前，我们就将这个特征映射到一个向量，和原本的 Embedding 进行组合：

$$e_i^j = e_{tok_i}^j + e_{seg_i}^j + e_{pos_i}^j + \alpha e_{em_i}^j$$

其中，$e_{em}{}_i^j$表示x_i^j的精确匹配特征向量，α表示该特征的重要性，作为一个超参数使用。

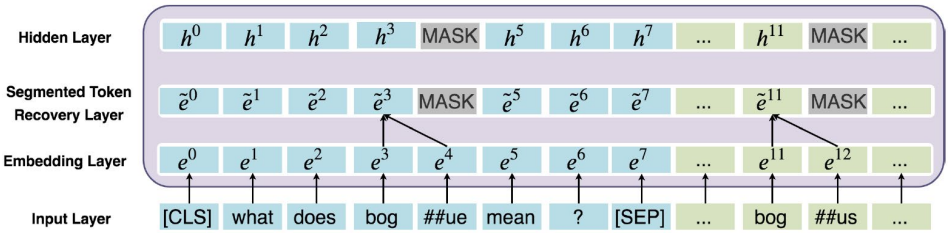


图 7　词还原机制的工作原理

除此之外，我们还提出了一种词还原机制如图 7 所示，词还原机制能够将 WordPiece 切分的 Subtoken 的表示合并，从而能更好地解决 OOV 错误匹配的问题。具体来说，我们使用 Average Pooling 对 Subtoken 的表示合并作为隐层的输入。除此之外，如上图 7 所示，我们使用了 MASK 处理 Subtoken 对应的非首位的隐层位置。值得注意的是，词还原机制也能很好地避免模型的过拟合问题。这是因为 MARCO 的集合标注是比较稀疏的，换句话说，有很多正例未被标注为 1，因此容易导致模型过拟合这些负样本。词还原机制一定程度上起到了 Dropout 的作用。

总结与展望

以上内容就对我们提出的 DR-BERT 模型进行了详细的介绍。我们提出的 DR-BERT 模型主要采用了任务自适应预训练以及两阶段模型精调训练。除此之外，还提出了词还原机制和精确匹配特征提高 OOV 词的匹配效果。通过在大规模数据集 MS MARCO 的实验，充分验证了该模型的优越性，希望这些能对大家有所帮助或者启发。

参考文献

[1] Payal Bajaj, Daniel Campos, et al. 2016. “MS MARCO: A Human Generated MAchine Reading COmprehension Dataset” NIPS.

[2] Christopher J. C. Burges, Robert Ragno, et al. 2006. “Learning to Rank with Nonsmooth Cost Functions” NIPS.

[3] Jun Xu and Hang Li. 2007. “AdaRank: A Boosting Algorithm for Information Retrieval”. SIGIR.

[4] Po-Sen Huang, Xiaodong He, et al. 2013. “Learning deep structured semantic

models for web search using clickthrough data". CIKM.
[5] Chenyan Xiong, Zhuyun Dai, et al. 2017. "End-to-end neural ad-hoc ranking with kernel pooling". SIGIR.
[6] Zhe Cao, Tao Qin, et al. 2007. "Learning to rank: from pairwise approach to listwise approach". ICML.
[7] Fen Xia, Tie-Yan Liu, et al. 2008. "Listwise Approach to Learning to Rank: Theory and Algorithm". ICML.
[8] Mike Taylor, John Guiver, et al. 2008. "SoftRank: Optimising Non-Smooth Rank Metrics". In WSDM.
[9] Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. 2018. "Bert: Pre-training of deep bidirectional transformers for language understanding". arXiv preprint arXiv:1810.04805.
[10] Rodrigo Nogueira and Kyunghyun Cho. 2019. "Passage Re-ranking with BERT". arXiv preprint arXiv:1901.04085 (2019).
[11] Rodrigo Nogueira, Wei Yang, Kyunghyun Cho, and Jimmy Lin. 2019. "Multi-stage document ranking with BERT". arXiv preprint arXiv:1910.14424 (2019).
[12] Zhuyun Dai and Jamie Callan. 2019. "Context-aware sentence/passage term importance estimation for first stage retrieval". arXiv preprint arXiv:1910.10687 (2019)
[13] Hiroshi Mamitsuka. 2017. "Learning to Rank: Applications to Bioinformatics".
[14] 杨扬、佳昊等 . 美团 BERT 的探索和实践 .

作者简介

兴武，弘胤，金刚，富峥，武威等，均来自美团 AI 平台 / 搜索与 NLP 中心。
特别感谢中国科学院软件所研究员金蓓弘老师在 MARCO 比赛和文章撰写过程中给予的指导和帮助。

美团无人车引擎在仿真中的实践

作者：杨磊

1. 引言

过去几年，自动驾驶技术有了飞速发展。国内也出现了许多自动驾驶创业企业，这些公司以百度开源项目 Apollo 为起点，大都可以直接进行公开道路测试，公开道路测试也成为促进技术进步的主要方法。基础问题得以解决之后，行业面临的更多是长尾问题，依靠路测驱动自动驾驶能力建设的方式变得不再高效，离线仿真的地位日益凸显。行业头部企业在仿真的投入十分巨大，Waymo 公司 2019 年公布的仿真里程是 100 亿英里，是路测里程的 1000 倍。

相应地，美团无人车团队在仿真上的投入也在逐渐增大。在仿真平台的建设中，团队发现公开道路测试和仿真测试看似相似，实际上差异巨大：在车载环境下，为了确保系统的稳定运行，通常要保证一定资源处于空闲状态；仿真环境则不同，如何高效利用资源，如何实现压榨资源的同时确保仿真结果与路测结果一致成为了关键目标。在应对这些挑战的过程中，美团提出了无人车引擎的概念，将车载与离线环境的差异隔离起来：功能模块无需任何更改便可以满足两种场景的需要。

本文首先会介绍无人车引擎的概念，并以仿真环境面临的挑战为线索介绍美团无人车引擎的核心设计。

02 无人车引擎

概念

无人车引擎是自动驾驶的基础设施，在机制、工具和计算模型上对功能模块提供支持，隔离自动驾驶所处环境，使各功能模块专注于自身功能。

在机制层，他为各功能模块提供通信、调度、数据、配置、异常监控等支持。

在应用层，引擎为各功能模块提供调试、可视化、性能调优、效果评估等工具支持。

在模块层，引擎为各功能模块提供统一的计算模型和运行环境，确保他们在车上环境、分布式环境、调试环境下的行为一致。

美团无人车引擎的架构图如下：

图 1　无人车引擎布局

如图 1 所示，作为引擎支撑的主要部分，Perception、Localization、Planning 等是自动驾驶系统中重要的功能模块，它们实现了无人车系统的核心功能。引擎则在机制和工具，上下两个方向上支撑他们：各功能模块按照引擎的规范开发，直接或者间接地使用引擎机制层提供的功能并自然而然地获得工具的支持。比如，功能模块只要使用引擎的通信工具，就能直接获得数据落盘、性能报表调试信息可视化的支持，同时基于这些路测数据，在仿真环境下，功能模块会自动获得单步调试、效果评估等功能支持。

自动驾驶引擎面临的挑战

图 1 中所列举的功能是引擎的基础组成部分，引擎所提供的远不止于此，对于多种环境的支持才是美团无人车团队引入引擎概念的真正原因。前面提到，无人车首先运行在车载系统中，随着技术和环境的变化，更多地运行于仿真环境下，二者截然不同。车载环境下，无人车系统的运行环境较好，为了保障各功能模块能够正常运行，

CPU、GPU、内存等资源要提供一定程度的冗余。而仿真环境的要求完全不同：从用户的角度看，仿真的用户是工程师，他们期望仿真任务能够在确定时间内完成尽量多的任务；从集群的角度看，他们希望仿真能够尽量提升资源利用率。接下来的部分将介绍无人车系统在这两类环境下会面临哪些挑战，以及美团无人车团队如何通过引擎应对这些挑战。

行为一致性的挑战

早期，美团无人车团队依赖于 ROS 搭建无人车系统，在车载环境下，ROS 的表现合格。然而在开始仿真建设后，团队遇到很多问题，其中最突出的是“行为一致性问题”，这个问题具体是指：无人车系统在运行过程中，当出现系统资源的变化，行为也随之发生变化。比如，当仿真任务在一台机器上运行时，系统产生的结果和这台机器的状态有关，这台机器被独占地使用或是和其它任务同时运行，结果会有差异。而且，即使不考虑资源利用率，让仿真任务独占机器资源，同一个任务运行两次，结果也会有微弱的扰动。

更严重情况发生在离线环境，此情境追求资源利用率的最大化，意味着计算资源的十分紧张，扰动将变得不再轻微，结果将变得更不可靠，仿真的结果也就失去了价值。

因此，如何在车上和离线两套截然不同的环境下确保结果的一致性，是仿真引擎必须解决的问题。此问题由以下两个原因造成：一是功能模块时序的不一致；二是功能模块内部执行的不一致。

时序一致性

为了介绍什么是时序一致性，首先要介绍一下无人车系统中时序的概念。

无人车系统由多个功能模块组成，功能模块之间有数据依赖关系，比如 Perception 依赖于 Lidar、Camera 的数据，Prediction 依赖于 Prediction 的输出。不同模块的触发条件不同，比如 Planning 是依据时钟触发的而 Prediction 是依赖于 Perception 的数据触发的。由数据关系和触发条件形成的功能模块的执行顺序就是自动驾驶系统的时序。在理想情况下，每个模块都能在满足触发条件时立刻执行并在预期的时间内完

成任务，也就是说，只要保留各模块的输出就可以完全复现线上的问题，离线仿真出现的问题在路测时也必然出现。

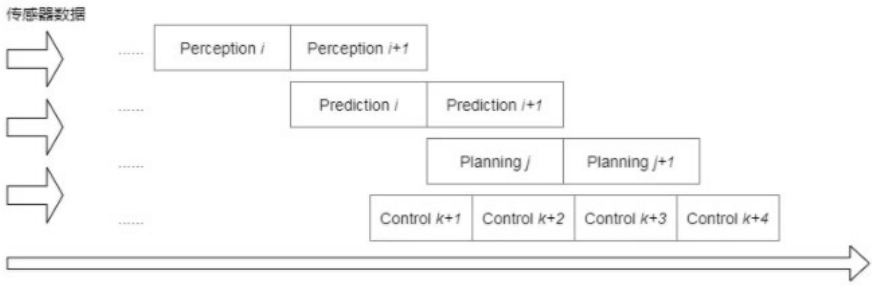


图 2　无人车系统理想时序

然而现实情况远比这复杂，举例来说，当无人车经过拥堵路段时，Perception 需要处理的数据会显著增多，Planning 也可能因为交通参与者过多导致耗时增长，时序必然与理想情况不符合。如下图 3 所示，在车载环境下这种行为方式是没问题的，然而在仿真环境时却会导致严重后果：每一次计算环境的些许变化都有可能导致时序的变化，进而导致系统行为的差异。

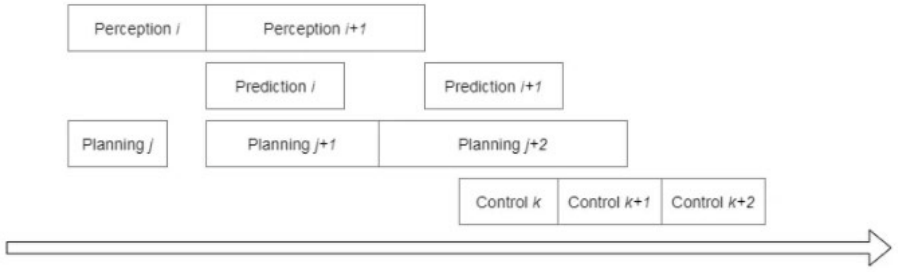


图 3　无人车系统实际时序

这就是时序一致性问题。为了解决这种问题，美团无人车引入了调度器，时序的一致性由调度器保证。此外，引擎按照不同的应用场景，进一步细化了调度器的种类。其中最简单的调度器是“在线调度器”，它的目标只有一个：在功能模块处于 Ready 状态时执行它，车载系统中就是使用的这种调度器，它的行为方式也与

ROS 类似，不过他会记录下调度时序以备使用。除此之外，引擎还提供一组离线调度器，以应对不同的使用场景。这里在线和离线的差异根据数据来源判断，如果数据来自传感器那么就是在线调度器；如果数据来自路测记录那就是离线调度器，具体分类如下图 4 所示：

图 4　调度器分类

以下是美团无人车引擎提供的调度器种类及他们的使用场景：

- **在线调度器**：在满足触发条件时立即触发功能模块，通常在车载环境下会使用；
- **复现调度器**：按照调度器保存的调度信息复现调度时序，在调试时或复现路测场景时使用；
- **理想调度器**：按照理想时序调度资源，通常在仿真时使用；
- **条件驱动调度器**：在条件满足时调度功能模块运行。在这种调度方式下，功能模块的调度密度介于理想调度器和复现调度器之间，他的实现也相对简单，是应用最广泛的调度器。

在他们的帮助下，功能模块执行的时序就能得到保障：只要调度器和输入数据不变，那么无论计算环境如何变化，各功能模块的执行时序总能保持一致。

功能模块的计算模型

时序一致性除了需要调度保证之外，功能模块的内部计算必须是受到调度器调度的。功能模块必须在调度器允许时才能开始执行，在结束时调度器能得到通知。如果存在脱离调度器之外的计算线程，那么系统的一致性必然无法保证。为此，引擎引入了标准计算模型，任何一个功能模块都有应该遵守这个计算模型，从而获得引擎包括一致性保障、单步调试支持、信息可视化等功能的支持。

标准计算模型如下：每一个功能模块都有且仅有一个计算过程并以迭代为单位，每一次调度完成一帧的计算。当然引擎并不控制帧计算内部的细节，帧计算内部的优化由功能模块负责。

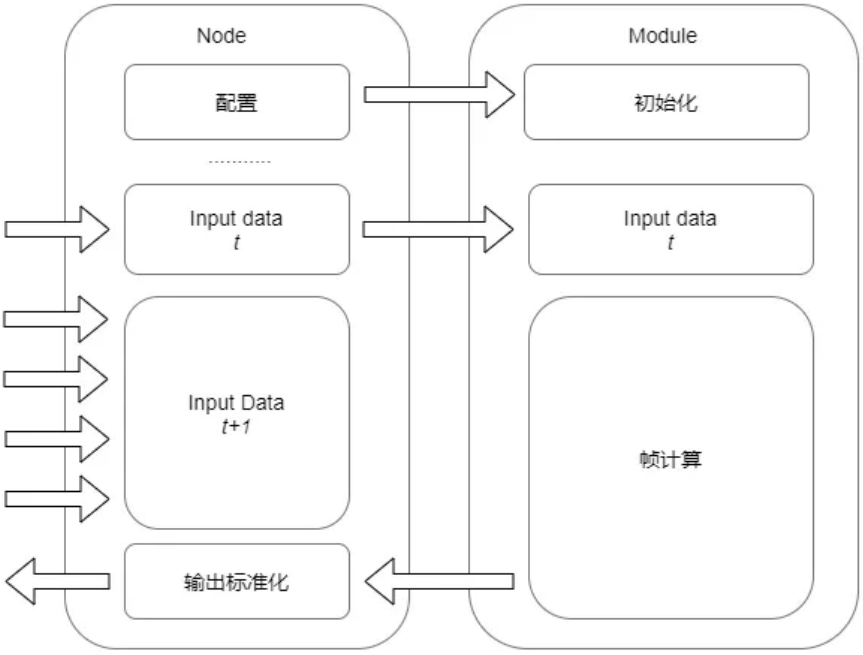


图 5　功能模块的标准模型

标准模型的定义并不一定符合每一个功能模块的实际情况：比如 Localization，它订阅多类频率不同的传感器数据并以不同的频率输出。在实践中，引擎通过对

Localization 功能的重新拆分实现了标准化。此外，对于像 Perception 这类计算量很大、同时兼具异构计算的功能模块来说，多线程，异步 I/O 的机制必须引入，引擎同时提供了相应的支持确保符合标准模型。

在实践过程中，美团无人车团队花费了相当时间来完成这些改造。改造完成后仿真结果的权威性得到了加强，更重要的是：系统的行为不再受外部资源（GPU、CPU、内存等）的影响，这也为离线环境提升资源利用率扫清了障碍。接下来介绍无人车引擎如何在功能模块完全无感的情况下提升资源利用率。

04 资源利用率问题

前面提到过，车载系统和仿真系统环境差异很大：车载系统为了追求系统的平稳运行会保证关键资源有一定程度的富裕；对于仿真系统，保留 idle 就是对资源的浪费。在系统的一致性得到保障之后，资源利用率才能成为引擎的优化目标。优化资源利用率包含了很多方面，比如数据调度等，由于篇幅所限，这里只介绍与引擎相关的优化工作。接下来的部分，将根据无人车系统在仿真环境运行时的特点进行优化，他们分别是资源需求不均、功能模块的重复计算、GPU/CPU 计算不平衡。

数据需求不均匀

从数据的输入规模上讲，各功能模块是极不均衡的：Perception 和 Localization 依赖于 Lidar 和 Camera 数据，数据使用量占到系统的 85% 以上（按照数据存储的规模计算，忽略中间数据，具体比例与开启的 Camera 相关，此处给出概数）。从资源消耗上讲，Perception 和 Prediction 消耗较多的 GPU 计算资源。为了提升云计算资源的效率，无人车引擎必须支持分布式部署：即一套自动驾驶系统分别部署与多台机器甚至是跨机房的机器之上的。

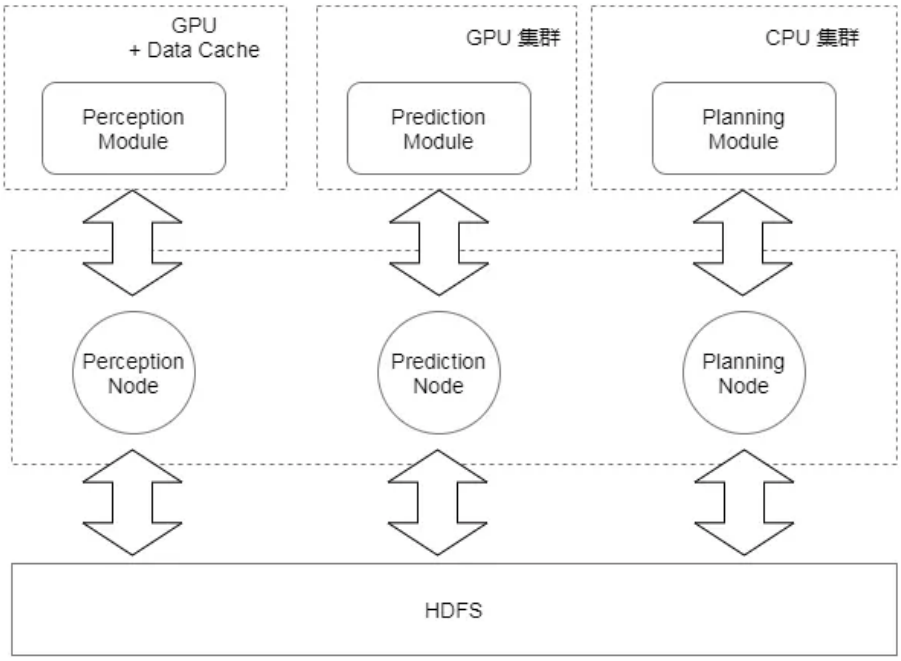
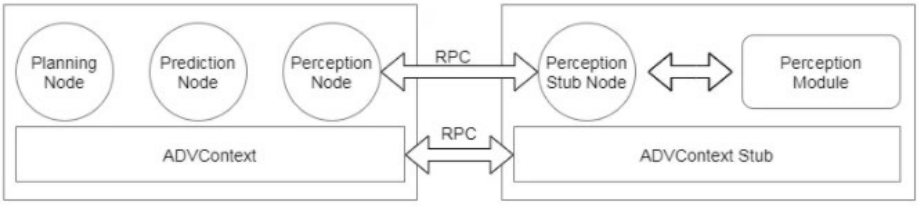


图 6　分布式部署

为了实现分布式部署，引擎参考了计算图模型的概念，采用了类似于 Tensorflow 的设计：将功能模块分成了 Node 和 Module 两个部分。其中 Node 负责定义依赖关系，而 Module 负责完成计算。对于远程部署的 Module 来说，引擎提供了 ADVContext 和 Node Stub 的概念用于协助 Module 完成运算，对于 Module 而言，它对于自身处于环境（远程或者本地）一无所知。



基于图 7 的设计，自动驾驶系统有了分布式部署的能力：一套自动驾驶系统可以运行于一组机器之上。提升离线效率的努力不再局限于单台机器，无人车系统的离线优化

获得了更多的手段和更广的空间。

重复计算

仿真任务分成多种类型，即有运行单个模块的任务，也有同时执行 Perception、Prediction、Planning 的任务。对于同时运行多个 Module 的任务，放在集群的角度看，很多计算都是重复的。试想一个场景：Planning 引入新方法，工程师希望能够在最新 Perception 版本上的获得新方法的效果评估结果。对于仿真而言，这是一个经典场景，常用的方法是离线执行 Perception、Prediction 和 Planning 三个模块并执行 Evaluation 产生报表、评估结果。

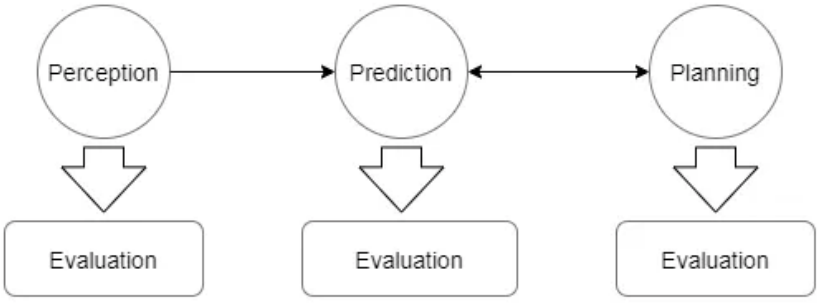


图 8　分模块 Evaluation

一般而言，Perception 的结果受到数据和本身算法迭代的影响，当 Planning 的迭代时，Perception 的结果不会受到影响，它的输出完全可以复用。得益于 Node 和 Module 概念的分离，Perception Node 所绑定的 Module 完全可以是一个非计算单元，而是一个数据服务 Module。

在美团无人车数据平台和无人车引擎共同努力下，通过 Data Service Module, 这个常见的仿真任务的流程在工程师感知不到的情况下变成了图 9 这样。不同版本的 Perception 的输出结果被保存下来，Prediction 和 Planning 只要使用之前的结果，避免了 Perception 的反复计算。

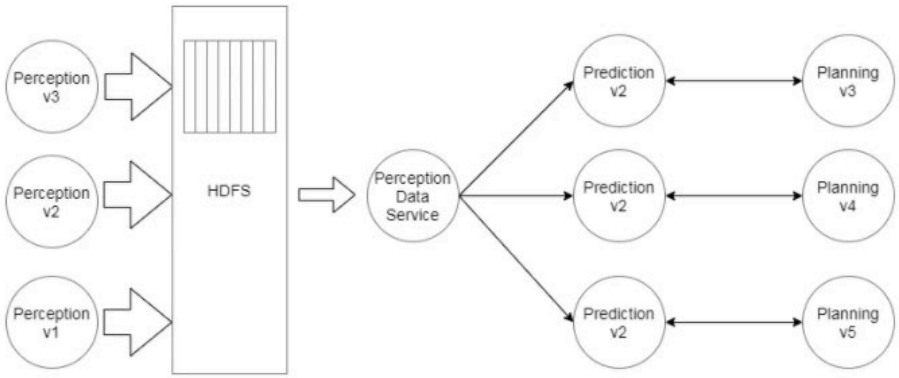


图 9　数据复用

GPU 计算分流

无人车系统是一个同时具备重度 CPU 计算和重度 GPU 计算系统，两部分的计算是不平衡的。引擎为了提升 GPU 资源的利用效率，在内部集成了模型管理的功能同时提供了本地和远程两种 Prediction 的机制。再结合分布式部署方式，系统可以完全部署于 CPU 集群之上，模型相关的计算可以通过 RPC 请求在 Model Serving 上完成。通过 GPU 和 CPU 计算的隔离，引擎帮助提升了 GPU 和 CPU 计算资源的利用率。

05 结论

在持续的实践中，美团无人配送团队抽离出一套自动驾驶引擎，为功能模块提供机制和工具的同时，它还提供了对车载（低时延）和仿真（高吞吐）两套环境的适配。此外，配合美团的大数据基础设施以及在此基础之上专为无人车建立的数据平台，美团无人车逐步建立了完善的自动驾驶基础设施。未来，希望在引擎的帮助下能够隔离功能模块、计算平台、运行环境，使得自动驾驶能力迭代与自动驾驶落地应用两个方向上的工作能够独立开展，齐头并进，加快美团无人车的落地步伐。

关于美团无人配送

美团无人车配送中心成立于 2016 年，由美团首席科学家夏华夏博士领导。美团无人车配送围绕美团外卖、美团跑腿等核心业务，通过与现有复杂配送流程的结合，形成了无人配送整体解决方案，满足在楼宇、园区、公开道路等不同场景下最后三公里的外卖即时配送需求，提升配送效率和用户体验，最终实现“用无人配送让服务触达世界每个角落”的愿景。

招聘信息

美团无人车配送中心大量岗位持续招聘中，诚招算法 / 系统 / 硬件开发工程师及专家。欢迎感兴趣的同学发送简历至：ai.hr@meituan.com（邮件标题注明：美团无人车团队）。

美团无人配送 CVPR2020 论文 CenterMask 解读

作者：钰晴　申浩等

计算机视觉技术是实现自动驾驶的重要部分，美团无人配送团队长期在该领域进行着积极的探索。不久前，高精地图组提出的 CenterMask 图像实例分割算法被 CVPR2020 收录，本文将对该方法进行介绍。

CVPR 的全称是 IEEE Conference on Computer Vision and Pattern Recognition，IEEE 国际计算机视觉与模式识别会议，它和 ICCV、ECCV 并称为计算机视觉领域三大顶会。本届 CVPR 大会共收到 6656 篇投稿，接收 1470 篇，录用率为 22%。

背景

one-stage 实例分割的意义

图像的实例分割是计算机视觉中重要且基础的问题之一，其在众多领域具有十分重要的应用，比如：地图要素提取、自动驾驶车辆感知等。不同于目标检测和语义分割，实例分割需要对图像中的每个实例（物体）同时进行定位、分类和分割。从这个角度看，实例分割兼具目标检测和语义分割的特性，因此更具挑战。当前两阶段（two-stage）目标检测网络（Faster R-CNN[2] 系列）被广泛用于主流的实例分割算法（如 Mask R-CNN[1]）。

2019 年，一阶段（one-stage）无锚点（anchor-free）的目标检测方法迎来了新一轮的爆发，很多优秀的 one-stage 目标检测网络被提出，如 CenterNet[3], FCOS[4] 等。这一类方法相较于 two-stage 的算法，不依赖预设定的 anchor，直接预测 bounding box 所需的全部信息，如位置、框的大小、类别等，因此具有框架简单灵活，速度快等优点。于是很自然的便会想到，实例分割任务是否也能够采用这种 one-stage anchor-free 的思路来实现更优的速度和精度的平衡？我们的论文分析

了该问题中存在的两个难点，并提出 CenterMask 方法予以解决。

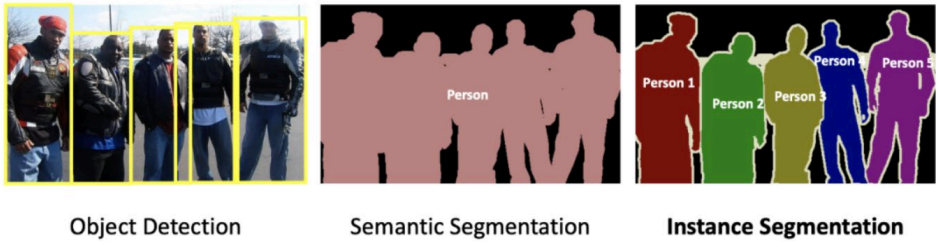


图 1　目标检测，语义分割和实例分割的区别

one-stage 实例分割的难点

相较于 one-stage 目标检测，one-stage 的实例分割更为困难。不同于目标检测用四个角的坐标即可表示物体的 bounding box，实例分割的 mask 的形状和大小都更为灵活，很难用固定大小的向量来表示。从问题本身出发，one-stage 的实例分割主要面临两个难点：

- 如何区分不同的物体实例，尤其是同一类别下的物体实例。two-stage 的方法利用感兴趣区域（Region of Interest，简称 ROI）限制了单个物体的范围，只需要对 ROI 内部的区域进行分割，大大减轻了其他物体的干扰。而 one-stage 的方法需要直接对图像中的所有物体进行分割。
- 如何保留像素级的位置信息，这是 two-stage 和 one-stage 的实例分割面临的普遍问题。分割本质上是像素级的任务，物体边缘像素的分割精细程度对最终的效果有较大影响。而现有的实例分割方法大多将固定大小的特征转换到原始物体的大小，或者利用固定个数的点对轮廓进行描述，这些方式都无法较好的保留原始图像的空间信息。

相关工作介绍

遵照目标检测的设定，现有的实例分割方法可大致分为两类：二阶段（two-stage）实例分割方法和一阶段（one-stage）实例分割方法。

- two-stage 的实例分割遵循先检测后分割的流程，首先对全图进行目标检测得到 bounding box，然后对 bounding box 内部的区域进行分割，得到每个物体的 mask。two-stage 的方法的主要代表是 Mask R-CNN[1]，该方法在 Faster R-CNN[2] 的网络上增加了一个 mask 分割的分支，用于对每个感兴趣区域（Region of Interest，简称 ROI）进行分割。而把不同大小的 ROI 映射为同样尺度的 mask 会带来位置精度的损失，因此该方法引入了 RoIAlign 来恢复一定程度的位置信息。PANet[5] 通过增强信息在网络中的传播来对 Mask R-CNN 网络进行改进。Mask Scoring R-CNN[6] 通过引入对 mask 进行打分的模块来改善分割后 mask 的质量。上述 two-stage 的方法可以取得 SOTA 的效果，但是方法较为复杂且耗时，因此人们也开始积极探索更简单快速的 one-stage 实例分割算法。
- 现有的 one-stage 实例分割算法可以大致分为两类：基于全局图像的方法和基于局部图像的方法。基于全局的方法首先生成全局的特征图，然后利用一些操作对特征进行组合来得到每个实例的最终 mask。比如，InstanceFCN[7] 首先利用全卷积网络 [8]（FCN）得到包含物体实例相对位置信息的特征图（instance-sensitive score maps），然后利用 assembling module 来输出不同物体的分割结果。YOLACT[9] 首先生成全局图像的多张 prototype masks，然后利用针对每个实例生成的 mask coefficients 对 prototype masks 进行组合，作为每个实例的分割结果。基于全局图像的方法能够较好的保留物体的位置信息，实现像素级的特征对齐（pixel-to-pixel alignment），但是当不同物体之间存在相互遮挡（overlap）时表现较差。与此相对应的，基于局部区域的方法直接基于局部的信息输出实例的分割结果。PolarMask[10] 采用轮廓表示不同的实例，通过从物体的中心点发出的射线组成的多边形来描述物体的轮廓，但是含有固定端点个数的多边形不能精准的描述物体的边缘，并且基于轮廓的方法无法很好的表示含有孔洞的物体。TensorMask[11] 利用 4D tensor 来表示空间中不同物体的 mask, 并且引入了 aligned representation 和 tensor bipyramid 来较好的恢复物体的空间位置细节，但是这些特征对齐的操作使得整个网络比 two-stage 的 Mask R-CNN 还要慢一些。

不同于上述方法，我们提出的 CenterMask 网络，同时包含一个全局显著图生成分支和一个局部形状预测分支，能够在实现像素级特征对齐的情况下实现不同物体实例的区分。

CenterMask 介绍

本工作旨在提出一个 one-stage 的图像实例分割算法，不依赖预先设定的 ROI 区域来进行 mask 的预测，这需要模型同时进行图像中物体的定位、分类和分割。为了实现该任务，我们将实例分割拆分为两个平行的子任务，然后将两个子任务得到的结果进行结合，以得到每个实例的最终分割结果。

第一个分支（即 Local Shape 分支）从物体的中心点表示中获取粗糙的形状信息，用于约束不同物体的位置区域以自然地将不同的实例进行区分。第二个分支（即 Global Saliency 分支）对整张图像预测全局的显著图，用于保留准确的位置信息，实现精准的分割。最终，粗糙但 instance-aware 的 local shape 和精细但 instance-un-aware 的 global saliency 进行组合，以得到每个物体的分割结果。

1. 网络整体框架

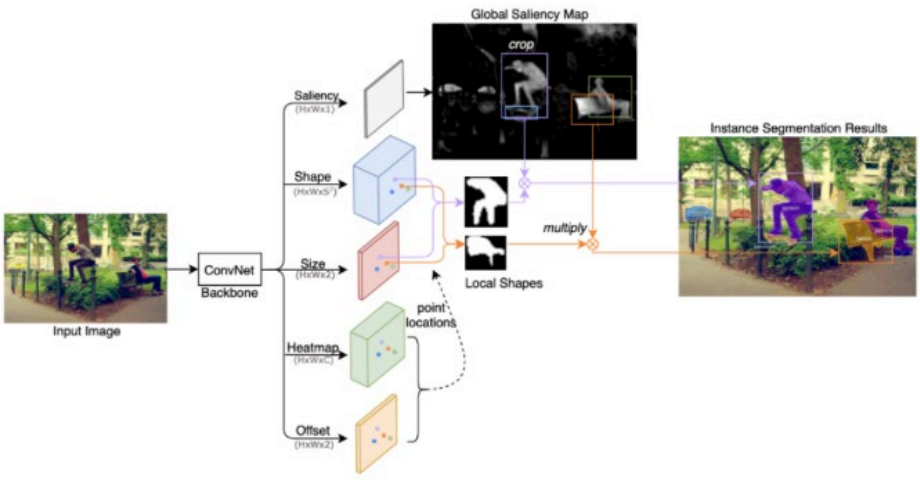


图 2　CenterMask 网络结构图

CenterMask 整体网络结构图如图 2 所示，给定一张输入图像，经过 backbone 网络提取特征之后，网络输出五个平行的分支。其中 Heatmap 和 Offset 分支用于预测所有中心点的位置坐标，坐标的获得遵循关键点预测的一般流程。Shape 和 Size 分支用于预测中心点处的 Local Shape，Saliency 分支用于预测 Global Saliency Map。可以看到，预测的 Local Shape 含有粗糙但是 instance-aware 的形状信息，而 Global Saliency 含有精细但是 instance-aware 的显著性信息。最终，每个位置点处得到的 Local Shape 和对应位置处的 Global Saliency 进行乘积，以得到最终每个实例的分割结果。Local Shape 和 Global Saliency 分支的细节将在下文介绍。

2. Local Shape 预测

为了区分位于不同位置的实例，我们采用每个实例的中心点来对其 mask 进行建模，中心点的定义是该物体的 bounding box 的中心。一种直观的想法是直接采用物体中心点处提取的图像特征来进行表示，但是固定大小的图像特征难以表示不同大小的物体。因此，我们将物体 mask 的表示拆分为两部分：mask 的形状和 mask 的大小，用固定大小的图像特征表示 mask 的形状，用二维向量表示 mask 的大小（高和宽）。以上两个信息都同时可以由物体中心点的表示得到。如图 3 所示，P 表示由 backbone 网络提取的图像特征，shape 和 size 表示预测以上两个信息的分支。用 Fshape(大小为 H x W x S x S) 表示 shape 分支得到的特征图，Fsize(大小为 H x W x 2) 表示 size 分支得到的特征图。假设某个物体的中心点位置为 (x,y)，则该点的 shape 特征为 Fshape(x,y)，大小为 1 x 1 x S x S，将其 reshape 成 S x S 大小的二维平面矩阵；该点的 size 特征为 Fsize(x,y)，用 h 和 w 表示预测的高度和宽度大小，将上述二维平面矩阵 resize 到 h x w 的大小，即得到了该物体的 LocalShape 表示。

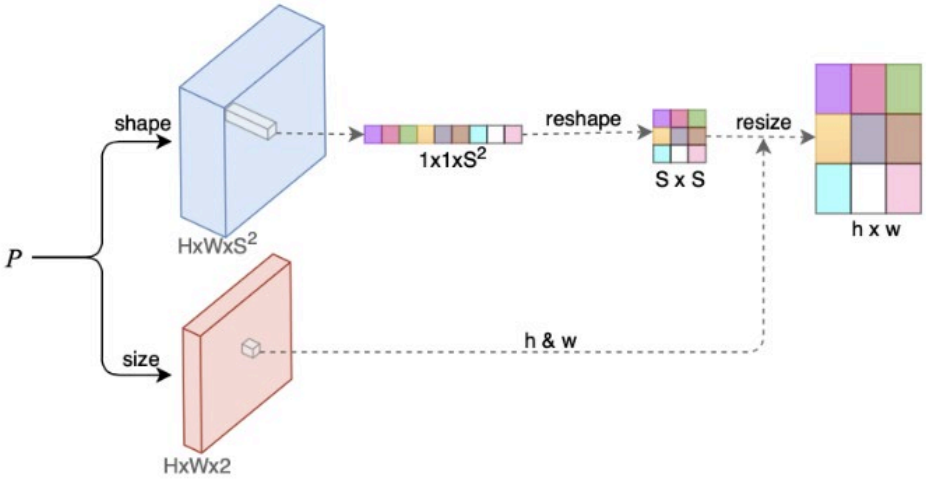


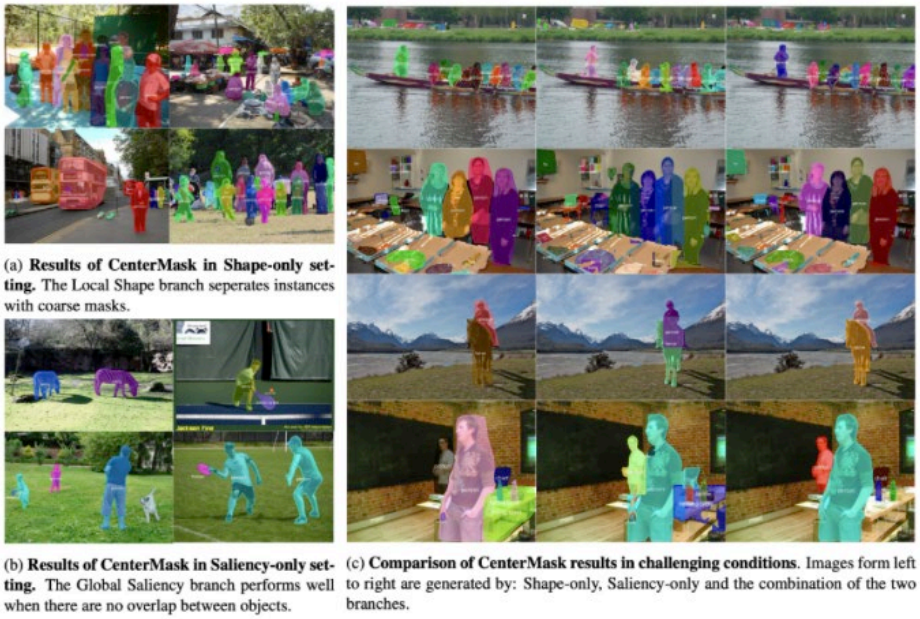
图 3　Local Shape 预测分支

3. Global Saliency 生成

尽管上述 Local Shape 表示可以生成每个实例的 mask，但是由于该 mask 是由固定大小的特征 resize 得到，因此只能描述粗糙的形状信息，不能较好的保留空间位置（尤其是物体边缘处）的细节。如何从固定大小的特征中得到精细的空间位置信息是实例分割面临的普遍问题，不同于其他采用复杂的特征对齐操作来应对此问题的思路，我们采用了更为简单快速的方法。启发于语义分割领域直接对全图进行精细分割的思路，我们提出预测一张全局大小的显著图来实现特征的对齐。平行于 Local Shape 分支，Global Saliency 分支在 backbone 网络之后预测一张全局的特征图，该特征图用于表示图像中的每个像素是属于前景（物体区域）还是背景区域。

实验结果

1. 可视化结果



(a) **Results of CenterMask in Shape-only setting.** The Local Shape branch seperates instances with coarse masks.

(b) **Results of CenterMask in Saliency-only setting.** The Global Saliency branch performs well when there are no overlap between objects.

(c) **Comparison of CenterMask results in challenging conditions**. Images form left to right are generated by: Shape-only, Saliency-only and the combination of the two branches.

图 4　CenterMask 网络不同设定下的分割结果

为了验证本文提出的 Local Shape 和 Global Saliency 两个分支的效果，我们对独立的分支进行了分割结果的可视化，如图 4 所示。其中 (a) 表示只有 Local Shape 分支网络的输出结果，可以看到，虽然预测的 mask 比较粗糙，但是该分支可以较好的区分出不同的物体。(b) 表示只有 Global Saliency 分支网络输出的结果，可以看到，在物体之间不存在遮挡的情形下，仅用 Saliency 分支便可实现物体精细的分割。© 表示在复杂场景下 CenterMask 的表现，从左到右分别为只有 Local Shape 分支，只有 Global Saliency 分支和二者同时存在时 CenterMask 的分割效果。可以看到，在物体之间存在遮挡时，仅靠 Saliency 分支无法较好的分割，而 Shape 和 Saliency 分支的结合可以同时在精细分割的同时实现不同实例之间的区分。

2. 方法对比

Method	Backbone	Resolution	FPS	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
two-stage									
MNC [4]	ResNet-101-C4	-	2.78	24.6	44.3	24.8	4.7	25.9	43.6
FCIS [18]	ResNet-101-C5-dilated	multi-scale	4.17	29.2	49.5	-	7.1	31.3	50.0
Mask R-CNN [12]	ResNeXt-101-FPN	800×1333	8.3	37.1	60.0	39.4	16.9	39.9	53.5
one-stage									
ExtremeNet [31]	Hourglass-104	512×512	3.1	18.9	44.5	13.7	10.4	20.4	28.3
TensorMask [2]	ResNet-101-FPN	800×1333	2.63	37.3	59.5	39.5	17.5	39.3	51.6
YOLACT [1]	ResNet-101-FPN	700×700	23.6	31.2	50.6	32.8	12.1	33.3	47.1
YOLACT-550 [1]	ResNet-101-FPN	550×550	33.5	29.8	48.5	31.2	9.9	31.3	47.7
PolarMask [28]	ResNeXt-101-FPN	768×1280	10.9	32.9	55.4	33.8	15.5	35.1	46.3
CenterMask	DLA-34	512×512	25.2	33.1	53.8	34.9	13.4	35.7	48.8
CenterMask	Hourglass-104	512×512	12.3	34.5	56.1	36.3	16.3	37.4	48.4

图 5　CenterMask 与其他方法在 COCO test-dev 数据集上的对比

CenterMask 与其他方法在 COCO test-dev 数据集上的精度 (AP) 和速度 (FPS) 对比见图 5。其中有两个模型在精度上优于我们的方法：two-stage 的 Mask R-CNN 和 one-stage 的 TensorMask，但是他们的速度分别大约 4fps 和 8fps 慢于我们的方法。除此之外，我们的方法在速度和精度上都优于其他的 one-stage 实例分割算法，实现了在速度和精度上的均衡。CenterMask 和其他方法的可视化效果对比见图 6。

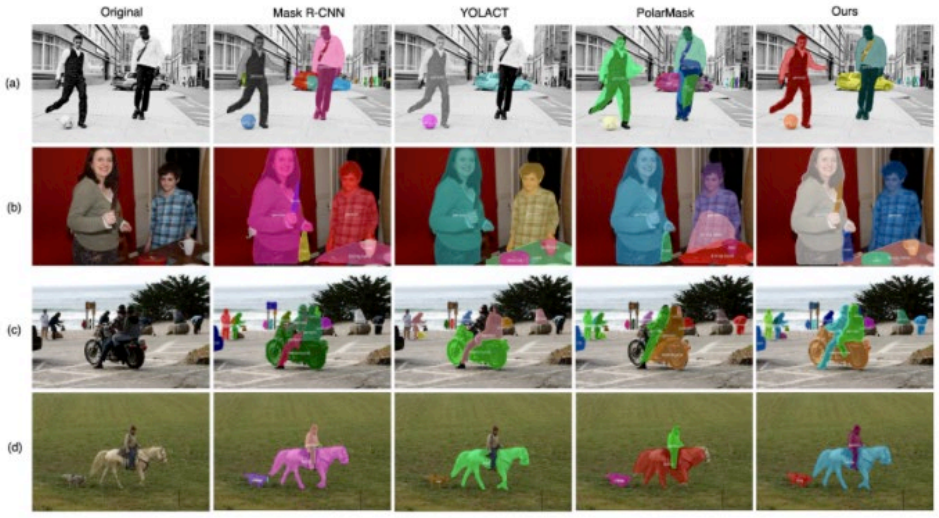


图 6　CenterMask 与其他方法在 COCO 数据集上的可视化对比

除此之外，我们还将提出的 Local Shape 和 Global Saliency 分支迁移至了主流的 one-stage 目标检测网络 FCOS，最终的实验效果见图 7。最好的模型可以实现 38.5 的精度，证明了本方法较好的适用性。

Backbone	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
ResNet-101-FPN	36.1	58.7	38.0	16.5	38.4	51.2
ResNeXt-101-FPN	36.7	59.3	38.8	17.4	38.7	51.4
ResNet-101-FPN-DCN	37.6	60.4	39.8	17.3	39.8	53.4
ResNeXt-101-FPN-DCN	38.5	61.5	41.0	18.7	40.5	54.8

图 7　CenterMask-FCOS 在 COCO test-dev 数据集上的性能

未来展望

首先，CenterMask 方法作为我们在 one-stage 实例分割领域的初步尝试，取得了较好的速度和精度的均衡，但是本质上仍未能完全脱离目标检测的影响，未来希望能够探索出不依赖 box crop 的方法，简化整个流程。其次，由于 CenterMask 预测 Global Saliency 的思想启发自语义分割的思路，而全景分割是同时融合了实例分割和语义分割的任务，未来希望我们的方法在全景分割领域也能有更好的应用，也希望后续有更多同时结合语义分割和实例分割思想的工作被提出。

更多细节见论文：CenterMask: single shot instance segmentation with point representation

招聘信息

美团无人车配送中心大量岗位持续招聘中，诚招感知 / 高精地图 / 决策规划 / 预测算法专家、无人车系统开发工程师 / 专家。无人车配送中心主要是借助无人驾驶技术，依靠视觉、激光等传感器，实时感知周围环境，通过高精定位和智能决策规划，保证无人配送机器人具有全场景的实时配送能力。欢迎感兴趣的同学发送简历至：

hr.mad@meituan.com（邮件标题注明：美团无人车团队）

参考文献

[1] He K, Gkioxari G, Dollár P, et al. Mask r-cnn[C]//Proceedings of the IEEE international conference on computer vision. 2017: 2961-2969.

[2] Ren S, He K, Girshick R, et al. Faster r-cnn: Towards real-time object detection with region proposal networks[C]//Advances in neural information processing systems. 2015: 91-99.

[3] Zhou X, Wang D, Krähenbühl P. Objects as points[J]. arXiv preprint arXiv:1904.07850, 2019.

[4] Tian Z, Shen C, Chen H, et al. Fcos: Fully convolutional one-stage object detection[C]//Proceedings of the IEEE International Conference on Computer Vision. 2019: 9627-9636.

[5] Liu S, Qi L, Qin H, et al. Path aggregation network for instance segmentation[C]// Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2018: 8759-8768.

[6] Huang Z, Huang L, Gong Y, et al. Mask scoring r-cnn[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2019: 6409-6418.

[7] Dai J, He K, Li Y, et al. Instance-sensitive fully convolutional networks[C]//European Conference on Computer Vision. Springer, Cham, 2016: 534-549.

[8] Long J, Shelhamer E, Darrell T. Fully convolutional networks for semantic segmentation[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2015: 3431-3440.

[9] Bolya D, Zhou C, Xiao F, et al. YOLACT: real-time instance segmentation[C]// Proceedings of the IEEE International Conference on Computer Vision. 2019: 9157-9166.

[10] Xie, Enze, et al. "Polarmask: Single shot instance segmentation with polar representation." //Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2020

[11] Chen, Xinlei, et al. "Tensormask: A foundation for dense object segmentation." Proceedings of the IEEE International Conference on Computer Vision. 2019.

WSDM Cup 2020 检索排序评测任务第一名经验总结

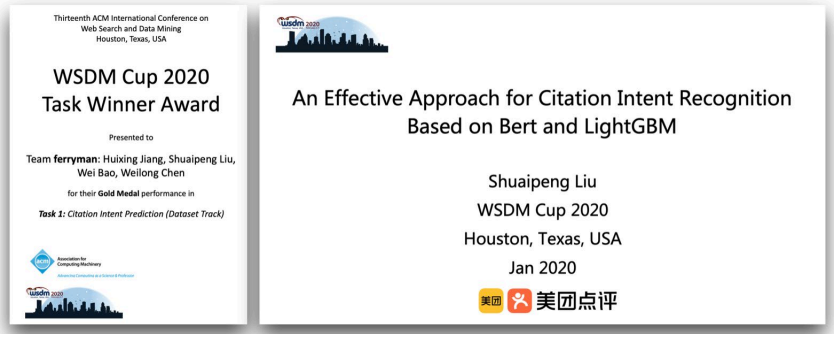
作者：帅朋

1. 背景

第 13 届“国际网络搜索与数据挖掘会议”(WSDM 2020) 于 2 月 3 日在美国休斯敦召开，该会议由 SIGIR、SIGKDD、SIGMOD 和 SIGWEB 四个专委会共同协调筹办，在互联网搜索、数据挖掘领域享有很高学术声誉。本届会议论文录用率仅约 15%，并且 WSDM 历来注重前沿技术的落地应用，每届大会设有的 WSDM Cup 环节提供工业界真实场景中的数据和任务用以研究和评测。

今年的 WSDM Cup 设有 3 个评测任务，吸引了微软、华为、腾讯、京东、中国科学院、清华大学、台湾大学等众多国内外知名机构的参与。美团搜索与 NLP 部继去年获得了 WSDM Cup 2019 第二名后，今年继续发力，拿下了 WSDM Cup 2020 Task 1：Citation Intent Recognition 榜单的第一名。

本次参与的是由微软研究院提出的 Citation Intent Recognition 评测任务，该任务共吸引了全球近 600 名研究者的参与。本次评测中我们引入高校合作，参评团队 Ferryman 由搜索与 NLP 部 -NLP 中心的刘帅朋、江会星及电子科技大学、东南大学的两位科研人员共同组建。团队提出了一种基于 BERT 和 LightGBM 的多模融合检索排序解决方案，该方案同时被 WSDM Cup 2020 录用为专栏论文。



2. 任务简介

本次参与的任务一（WSDM Cup 2020 Task 1: Citation Intent Recognition）由微软研究院发起，任务要求参赛者根据论文中对某项科研工作的描述，从论文库中找出与该描述最匹配的 Top3 论文。举例说明如下：

某论文中对科研工作 [1] 和 [2] 的描述如下：

> An efficient implementation based on BERT [1] and graph neural network (GNN) [2] is introduced.

参赛者需要根据这段科研描述从论文库中检索与 [1][2] 相关工作最匹配论文。

在本例中：

与工作 [1] 最匹配的论文题目应该是：

> [1] BERT: Pre-training of deep bidirectional transformers for language understanding.

与工作 [2] 最匹配的论文题目应该是：

> [2] Relational inductive biases, deep learning, and graph networks.

由上述分析可知，该任务是经典的检索排序任务，即根据文本 Query 从候选 Documents 中找出 Top N 个最相关的 Documents，核心技术包括文本语义理解和搜索排序。

2.1 评测数据

本次评测数据分为论文候选集、训练集、验证集和测试集四个部分，各部分数据的表述如表 1 所示：

表 1　评测数据分析表

数据集	数据量	包含的字段信息
论文候选集	838,939	论文ID、论文标题、摘要、期刊名、发表年份等
训练集	62,976	对相关科研工作的描述，匹配论文的ID
验证集	34,312	对相关科研工作的描述（初赛验证阶段使用）
测试集	34,428	对相关科研工作的描述（决赛评测阶段使用，以此确定算法最终效果）

对本次评测任务及数据分析可以发现本次评测存在以下特点：

- 与工业界的实际场景类似，本次任务数据量规模比较大，要求制定方案时需要同时考虑算法性能和效果，因此相关评测方案可以直接落地应用或有间接参考的价值；
- 为了保证方案具有一定落地实用价值，本任务要求测试集的结果需要在 48 小时内提交，这也对解决方案的整体效率提出了更高的要求，像常见的使用非常多模型的融合提升方案，在本评测中就不太适用；
- 跟自然语言处理领域的一般任跟自然语言处理领域的一般任务不同，本次评测任务中数据多来源于生命科学领域，存在较多的专有词汇和固定表述模式，因此一些常见的方法模型（例如在通用语料上预训练的 BERT、ELMo 等预训练模型）在该任务上的直接应用是不合适的，这也是本次任务的难点之一。

2.2 评测指标

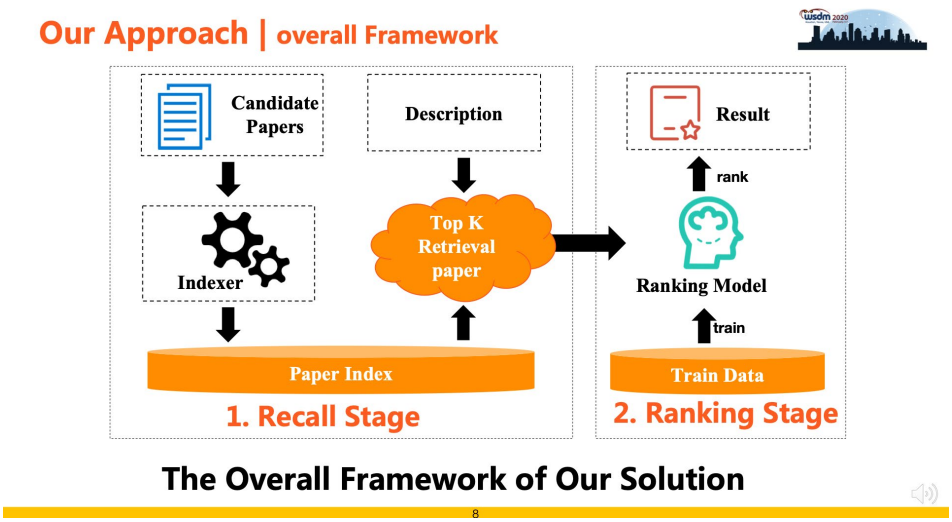
评测使用的评价指标为 Mean Average Precision @3 (MAP@3), 形式如下：

$$MAP@3 = \frac{1}{|U|}\sum_{u=1}^{|U|}\sum_{k=1}^{min(3,n))} P(k))$$

其中，|U| 是需要预测的 description 总个数，P(k) 是在 k 处的精度，n 是 paper 个数。举例来说，如果在第一个位置预测正确，得分为 1；第二个位置预测正确，得分为 1/2；第三个位置预测正确，得分为 1/3。

3. 模型方法

通过对评测数据、任务和评价指标等分析，综合考量方案的效率和精准性后，本次评测中使用的算法架构包括“检索召回”和“精准排序”两个阶段。其中，检索召回阶段负责从候选集中高效快速地召回候选 Documents，从而缩减问题规模，降低排序阶段的复杂度，此阶段注重召回算法的效率和召回率；精准排序阶段负责对召回数据进行重排序，采用 Learning to Rank 相关策略进行排序最优解求解。



3.1 检索召回

目标任务：使用高效的匹配算法对候选集进行粗筛，为后续精排阶段缩减候选排序的数据规模。

性能要求：召回阶段的方案需要权衡召回覆盖率和算法效率两个指标，一方面召回覆盖率决定了后续精排算法的效果上限，另一方面单纯追求覆盖率而忽视算法效率则不能满足评测时效性的要求。

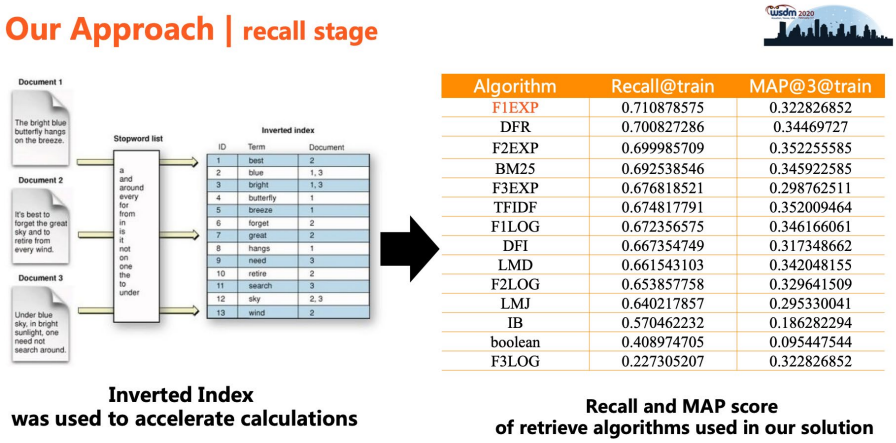
检索召回方案：比赛过程中对比实验了两种召回方案，基于“文本语义向量表征“和“基于空间向量模型 + Bag-of-Ngram”。由于本任务文本普遍较长且专有名词较多等数据特点，实验表明“基于空间向量模型 + Bag-of-Ngram”的召回方案效果更好，下表中列出了使用的相关模型及其实验结果（recall@200）。可以看到相比于传统的 BM25 和 TFIDF 等算法，F1EXP、F2EXP 等公理检索模型（Axiomatic Retrieval Models）可以取得更高的召回覆盖率，该类模型增加了一些公理约束条件，例如基本术语频率约束，术语区分约束和文档长度归一化约束等等。

F2EXP 定义如下：

$$S(Q,D)=\sum_{t\in Q\cap D}C(t,Q)\times\frac{C(t,Q)}{C(t,Q)+s+s\cdot\frac{|D|}{avdl}}\times\left(\frac{N+1}{df(t)}\right)^{k}$$

其中，Q 表示查询 query ,D 表示候选文档，C(t, Q) 是词 t 在 Q 中的频次，|D| 表示文档长度，avdl 为文档的平均长度，N 为文档总数，df(t) 为词 t 的文档频率。

为了提升召回算法的效果，我们使用倒排索引技术对数据进行建模，然后在此基础上实现了 F1EXP、DFR、F2EXP、BM25、TFIDF 等多种检索算法，极大了提升了召回部分的运行效率。为了平衡召回率和计算成本，最后使用 F1EXP、BM25、TFIDF 3 种算法各召回 50 条结果融合作为后续精排候选数据，在验证集上测试，召回覆盖率可以到 70%。



3.2 精准排序

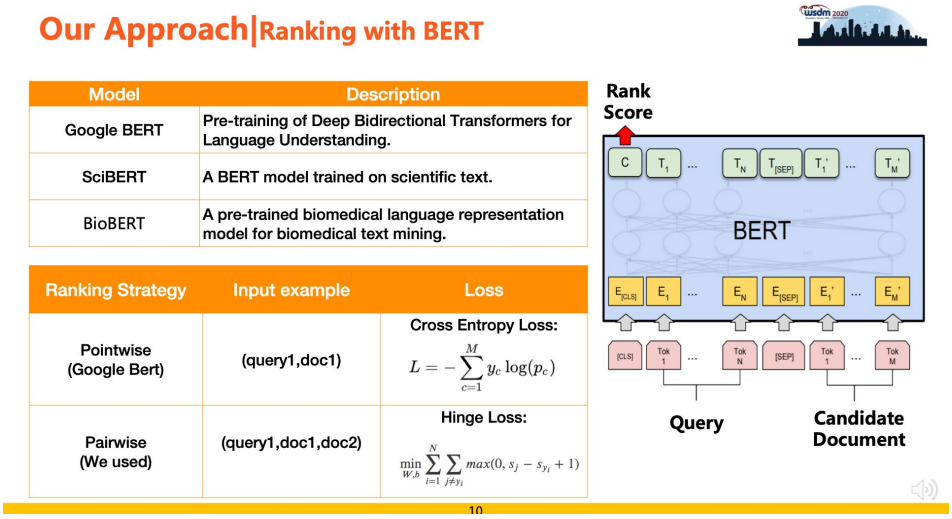
精排阶段基于 Learning to Rank 的思想进行方案设计，提出了两种解决方案，一种是基于 Pairwise-BERT 的方案，另一种是基于 LightGBM 的方案，下面分别进行介绍：

1) 基于 BERT 的排序模型

BERT 是近年来 NLP 领域最重大的研究进展之一，本次评测中，我们也尝试引入 BERT 并对原始模型使用 Pointwise Approach 的模式进行改进，引入 Pairwise Approach 模式，在排序任务上取得了一定的效果提升。原始 BERT 使用 Pointwise 模式把排序问题看做单文档分类问题，Pointwise 优化的目标是单条 Query 与 Document 之间的相关性，即回归的目标是 label。而 Pairwise 方法的优化目标是两个候选文档之间的排序位次（匹配程度），更适合排序任务的场景。具体来说，对原始 BERT 主要有两点改进，如下图中所示：

改进训练样本构造形式：Pointwise 模式下样本是按照形式构造输入，Pairwise 模式下样本按照形式进行构造，其中 Query 与 Doc1 的匹配程度大于与 Doc2 的匹配程度。

改进模型优化目标：Pointwise 模式下模型使用的 Cross Entropy Loss 作为损失函数，优化目标是提升分类效果，而 Pairwise 模式下模型使用 Hing Loss 作为损失函数，优化目标是加大正例和负例在语义空间的区分度。



在基于 BERT 进行排序的过程中，由于评测数据多为生命科学领域的论文，我们还使用了 SciBERT 和 BioBERT 等基于特定领域语料的预训练 BERT 模型，相比 Google 的通用 BERT 较大的效果提升。

2) 基于 LightGBM 的排序模型

不过，上面介绍的基于 BERT 的方案构建的端到端的排序学习框架，仍然存在一些不足。首先，BERT 模型的输入最大为 512 个字符，对于数据中的部分长语料需要进行截断处理，这就损失了文本中的部分语义信息；其次，本任务中语料多来自科学论文，跟已有的预训练模型还是存在偏差，这也在一定程度上限制了模型对数据的表征能力。此外，BERT 模型网络结构较为复杂，在运行效率上不占优势。综合上述三方面的原因，我们提出了基于 LightGBM 的排序解决方案。

LightGBM 是微软 2017 年提出，比 Xgboost 更强大、速度更快的模型。LightGBM 在传统的 GBDT 基础上有如下创新和改进：

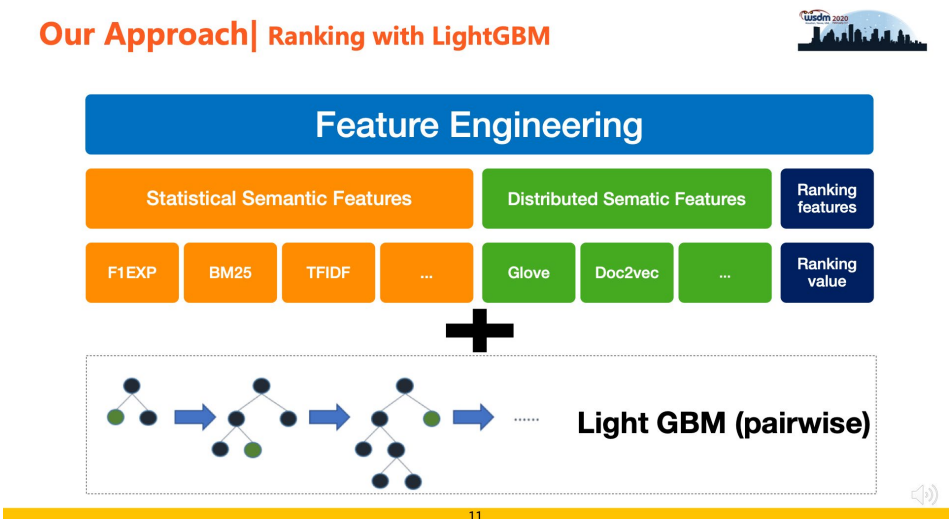
采用 Gradient-based One-Side Sampling(GOSS) 技术去掉很大部分梯度很小的数据，只使用剩下的去估计信息增益，避免低梯度长尾部分的影响；

采用 Exclusive Feature Bundling(EFB) 技术以减少特征的数量；

传统 GBDT 算法最耗时的步骤是使用 Pre-Sorted 方式找到最优划分点，其会在排好序的特征值上枚举所有可能的特征点，而 LightGBM 中会使用 histogram 算法替换了 GBDT 传统的 Pre-Sorted，牺牲一定精度换取了速度。

LightGBM 采用 Leaf-Wise 生长策略，每次从当前所有叶子中找到分裂增益最大的一个叶子，然后分裂，如此循环。因此同 Level-Wise 相比，在分裂次数相同的情况下，Leaf-Wise 可以降低更多的误差，得到更好的精度。

基于 Light GBM 的方案需要特征工程的配合。在我们实践中，特征主要包括 Statistic Semantic Features（包括 F1EXP、F2EXP、TFIDF、BM25 等）、Distributed Semantic Features（包括 Glove、Doc2vec 等）和 Ranking Features（召回阶段的排序序列特征），并且这些特征分别从标题、摘要、关键词等多个维度进行抽取，最终构建成特征集合，配合 LightGBM 的 pairwise 模式进行训练。该方法的优点是运行效率高，可解释性强，缺点是特征工程阶段比较依赖人工对数据的理解和分析。



4. 实验结果

我们分别对比实验了不同方案的效果，可以发现无论是基于 BERT 的排序方案还是基于 LightGBM 的排序方案，Pairwise 的模式都会优于 Pointwise 的模式，具体实验数据如表 2 所示：

表 2　不同方案实验结果

Model	MAP@3 Score
TFIDF	0.212 (valid set)
BM25	0.277 (valid set)
BERT(pointwise)	0.397 (test set)
BERT(pairwise)	0.402 (test set)
LightGBM(pointwise)	0.405 (test set)
LightGBM(pairwise)	0.413 (test set)
Ensemble(BERT(pairwise)+ LightGBM(pairwise))	0.425 (test set)

5. 总结与展望

本文主要介绍了美团搜索与 NLP 部在 WSDM Cup 2020 Task 1 评测中的实践方案，我们构建了召回 + 排序的整体技术框架。在召回阶段引入多种召回策略和倒排索引保证召回的速度和覆盖率；在排序阶段提出了基于 Pairwise 模式的 BERT 排序模型和基于 LightGBM 的排序模型。最终，美团也非常荣幸地取得了榜单第一名的成绩。

当然，在对本次评测进行复盘分析后，我们认为该任务还有较大提升的空间。首先在召回阶段，当前方案召回率为 70% 左右，可以尝试新的召回方案来提高召回率；其次，在排序阶段，还可以尝试基于 Listwise 的模式进行排序模型的训练，相比 Pairwise 的模式，Listwise 模式下模型输入空间变为 Query 跟全部 Candidate Doc，理论上可以使模型学习到更好的排序能力。后续，我们还会再不断进行优化，追求卓越。

6. 落地应用

本次评测任务与搜索与 NLP 部智能客服、搜索排序等业务中多个关键应用场景高度契合。目前，我们正在积极试验将获奖方案在智能问答、FAQ 推荐和搜索核心排序等场景进行落地探索，用最优秀的技术解决方案来提升产品质量和服务水平，努力践行“帮大家吃得更好，生活更好”的使命。

参考文献

[1] Fang H, Zhai C X. An exploration of axiomatic approaches to information retrieval[C]//Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval. 2005: 480-487.

[2] Wang Y, Yang P, Fang H. Evaluating Axiomatic Retrieval Models in the Core Track[C]//TREC. 2017.

[3] Devlin J, Chang M W, Lee K, et al. Bert: Pre-training of deep bidirectional transformers for language understanding[J]. arXiv preprint arXiv:1810.04805, 2018.

[4] Lee J, Yoon W, Kim S, et al. BioBERT: a pre-trained biomedical language representation model for biomedical text mining[J]. Bioinformatics, 2020, 36(4): 1234-1240.

[5] Beltagy I, Lo K, Cohan A. SciBERT: A pretrained language model for scientific text[C]//Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP). 2019: 3606-3611.

[6] Chen W, Liu S, Bao W, et al. An Effective Approach for Citation Intent Recognition Based on Bert and LightGBM. WSDM Cup 2020, Houston, Texas, USA, February 2020.

[7] Ke G, Meng Q, Finley T, et al. Lightgbm: A highly efficient gradient boosting decision tree[C]//Advances in neural information processing systems. 2017: 3146-3154.

作者简介

帅朋，美团 AI 平台搜索与 NLP 部。 会星，美团 AI 平台搜索与 NLP 部 NLP 中心对话平台负责人，研究员。 仲远，美团 AI 平台搜索与 NLP 部负责人，高级研究员、高级总监。

招聘信息

美团 -AI 平台 - 搜索与 NLP 部 -NLP 中心在北京 / 上海长期招聘 NLP 算法专家 / 研究员、对话平台研发工程师 / 技术专家、知识图谱算法专家，欢迎感兴趣的同学发送简历至：tech@meituan.com（邮件标题注明：NLP 中心 - 北京 / 上海）。

美团内部讲座｜清华大学莫一林：信息物理系统中的安全控制算法

作者：莫一林

【Top Talk/ 大咖说】由美团技术学院主办，面向全体技术同学，定期邀请美团各技术团队负责人、业界大咖、高校学者及畅销书作者，为大家分享最佳实践、互联网热门话题、学术界前沿技术进展等内容，从而建立工程师文化、提升技术视野。

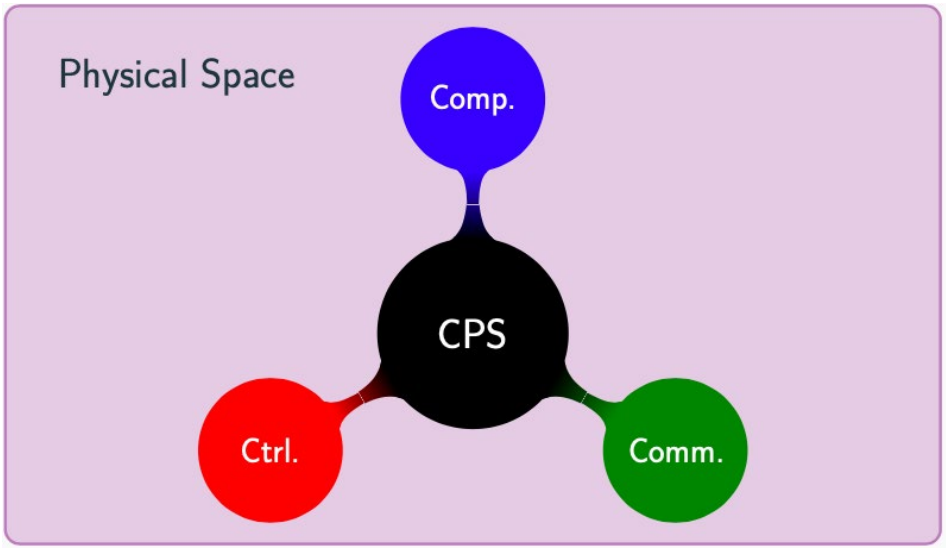
2020 年 9 月 10 日，Top Talk 邀请到了清华大学自动化系莫一林副教授，请他带来题为《信息物理系统中的安全控制算法设计》的分享。本文系本次分享内容的文字实录，希望能对大家有所帮助或者启发。

以下系文字实录：

中文讲的“安全”，其实基本上包含了英文中的两重含义，一个是 Safety，一个是 Security。信息物理系统中的安全问题主要是指 Security，也就是说，如果有人想要攻击你，在这种情况下，怎么能够保证系统的正常运行?

信息物理系统

“信息物理系统”，这个词大概是 2006 年由美国的 National Science Foundation（美国国家科学基金会）最开始提出来的。信息物理系统本质上来说是 Computation（计算）、Communication（通讯）、Control（控制）三 C 的融合，把三个 C 互相之间结合起来，嵌入到一个物理世界当中，从而能够使得整个系统更好地感知或者控制物理世界。比如，无人驾驶就是一个信息物理系统，因为无人驾驶本身是有一个物理的实体，有很多传感器来收集数据，这些数据经过通讯上传到计算机或者云上面，同时车和车之间可能还有通讯，然后去做感知、去做导航，再反馈到无人车的执行单元，最后反馈到物理空间。这些过程既包括了计算，也包括了控制的一些问题。



其实，信息物理系统是一个很大的概念。之前大家说安全，其实更多的是想到计算机安全或者网络安全，现在更多的是手机这类智能设备的安全。但是，信息物理系统安全为什么现在被提上了议程？这其中主要的原因是传统的控制系统，比如汽车内部的通讯是通过 CANBUS 实现的，这样的通讯本质上来说是一个独立的、专用的网络，并不和其他网络产生任何的连接。所以在这种情况下，就很难去大规模地攻击这样一个系统。

信息物理系统的安全威胁

但是，现在智能化逐渐成为了一种趋势，而且智能化的一个核心的事情，就是要使用很多新兴的感知和网络的技术，实现万物互联。在这种背景下，就会产生非常多的问题，因为所有东西都联网了，那么系统受到攻击的可能性就放大了很多。在这种情况下，怎么保证整个系统的稳定性或者说维护系统正常运行，这是一个很大的挑战。

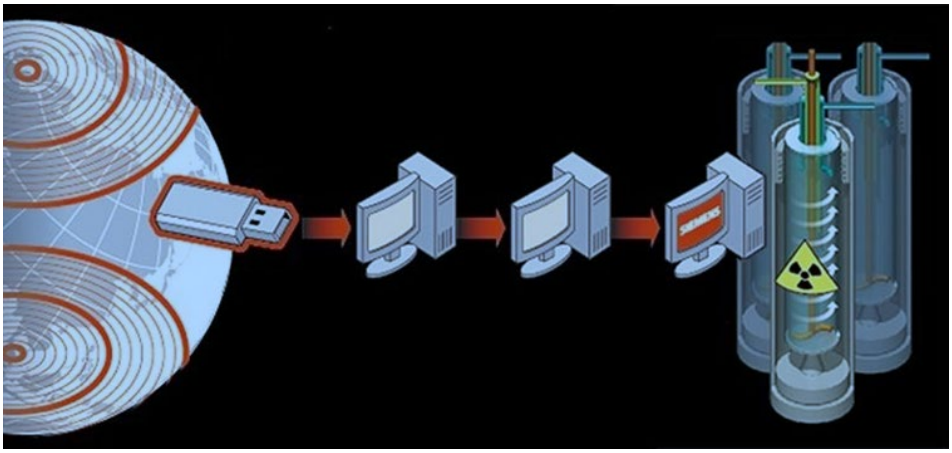
上图是我们之前说的智能家居的一个示意图。智能家居希望人们可以通过比如手机 App 去控制家里的电器，但是一旦家里的电器都上网之后，智能家居控制系统的安全就会变得非常重要，如果当一个攻击者可以操纵成千上万个家庭电器的话，这就可能带来非常严重的问题。

震网病毒

我们实验室从 2010 年前就开始做信息物理系统安全方面的研究。当时，有黑客设计了“震网病毒”，它的目标是伊朗用来提纯铀 235 的离心机。“震网病毒”破坏了伊朗上千台的离心机，最后对伊朗核计划造成了很大的损害。这也算是一个利用“信息战”成功带来破坏的一个例子。也是因为这个事情，让我们把信息物理系统的安全问题迅速提上了日程。

“震网病毒”的例子属于国家行为，可能比较少见，现在针对一般信息物理系统的攻击行为也越来越多。比如 2016 年，美国机构的报告就指出，全年一共有 290 多次针

对美国工业控制系统的攻击，覆盖了制造业、通讯、能源等多个行业。



攻击有很多种类型，先说比较常见的。如果系统联网，可以通过网络去侵入系统；即使系统没有联网，那么还有一些方式方法，比如像“震网病毒”的例子，是通过 U 盘一层一层带进去的。这些方式都能造成很大的破坏。我觉得比较值得关注的是第二个问题，一般做控制或者这种背景的研究人员，很多时候对计算机安全并不是特别有经验，那么开发的系统就可能会存在很多的漏洞。甚至有很多软件的开发者自己都不知道自己的系统有漏洞，但是黑客就可以发现你的漏洞。比如说像“震网病毒”的例子，伊朗不知道系统有问题，但黑客有内部资料，发现这个系统有问题，然后就利用这种漏洞去侵入了伊朗的系统。

大家比较担心的一个事情就是现在的系统变得越来越复杂，比如一辆车可能有几百个 ECU（Electronic Control Unit），一个波音 787 可能有上百万个零件，这上百万零件中的 70% 都是外包出去的，而一级外包商可能再外包给二级、三级，最后整个供应链就变得很复杂，这相当于是全球生产，最后汇集到西雅图去装配。那么在这个过程中，怎么保证装到系统上的每一个零件都是安全的，这也是一个值得思考的问题。

黑客攻击汽车系统

其实，在传统的汽车信息物理系统当中，漏洞还是有很多的。 比如 2015 年有一个比

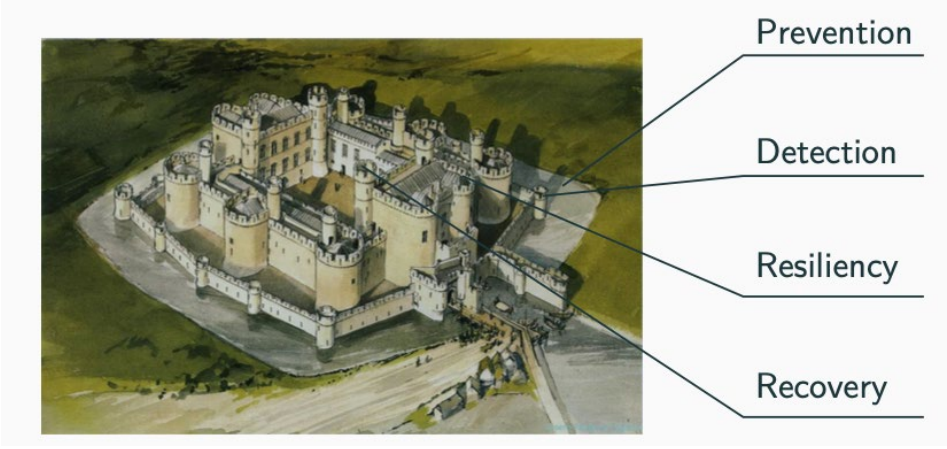
较著名的事件，有两个黑客（这两个人应该只是研究者，并不是真的想要搞破坏）通过攻击克莱斯勒吉普的一款车，入侵了这辆车的显示信息，就是所谓的 implementation，还包括一些娱乐系统，然后通过这个系统，进入车内部的网络来控制这个车，比如说方向、控制刹车，甚至包括安全气囊等等。

有人可能会说，像你们碰到这些问题，在计算机里面也都有，而且已经研究了很长时间了，为什么还要单独提这样一个概念，这个信息物理系统安全中到底有什么新的东西吗？我觉得这个核心价值在于：传统的研究主要都是在所谓的 Cyber Physical System。而信息物理系统，它的核心的是说那些有“物理”的系统，那么这个物理系统就带来了很多的挑战。首先，从传统上来说，如果一个计算机受到攻击，最差的情况，也就是将这个计算机关闭就结束了。但是如果是一个高速行驶的车，无法将它直接关闭，只能让它缓慢地停下来，但是停下来这件事情本身因为有物理系统的参与，所以就不是一个简单的事情。而对于无人机来说，甚至都不能让它停下来，必须让它以一定的速度去飞，因为如果是固定翼无人机，停下来就可能意味着坠毁，所以在这个过程中，物理系统带来了很多的挑战。

另外一个问题是，我们不一定能够让物理系统停下来，比如说如果一个电网受到攻击，那么我们希望能够尽可能把被攻击的地方隔离出来，而不是要让整个地区比如北京市出现停电。这个意思就是说，在系统受到攻击的时候，要让这个系统还能带着“伤”去运行，而不是一旦有一点风吹草动就需要重启，这个也是一个很大的问题。最后，这些物理系统其实都需要非常高的可靠性，比如对飞机来说，我们之前跟波音做过一些项目，他们要求放上飞机的任何东西都需要经过鉴定，必须保证飞机有极高的可靠性，要远远高于杀毒软件要能检测出 99% 病毒的这种可靠性。很多传统的信息安全方法，都是一个所谓的 Best of Effort Approach，就是说，尽可能地提供能提供的最好的服务，但是并不能给出特别多的保证，因为对于一些需要非常高可靠性的系统，即便存在很小的可能也许就隐藏着一个很大的威胁。

为信息物理系统构建“护城河”

在信息物理系统里面，为了保证这个系统的安全，我们需要保证这个系统有非常高的可靠性。其实，任何一个单一的方法都是很难完成这个目标的，我们需要是一个多层的防御机制，它就像一个城堡一样，外面还有一个护城河，中间有一个城墙，里面还有一个城墙，必须要层层地设防，只有这样才能解决这个问题。



我觉得这其中有几个比较关键的点，比如 Prevention、Detection、Resiliency 、Recovery。其中，Prevention 就是怎么防止别人进来，那么这个地方可能是需要更好的防火墙，比如说杀毒软件等等这些东西。当然，我们不可能百分之百地把别人都挡在系统之外，如果有人进来了之后，我们就需要去检测这个系统到底有没有被入侵，包括在一个大的系统怎么去定位哪几个部分被入侵了，这就是所谓的 Detection。另外，我们设计一个系统，需要考虑到一定的 Resiliency，比如说有一个汽车上面有一个零件坏了，这个系统就完蛋了，那么这个系统就不是很有韧性，所以系统能不能带“伤”运行，也是一个需要考虑的因素。最后，当发现这个系统出现问题之后，我们可能就会需要有一个所谓的恢复过程，比如重启等等手段。整体来说，我觉得要通过很多方面，通过多层防御来保证信息物理系统的安全。

信息物理系统中的安全控制算法

今天，我想给大家讲一讲我们在信息物理系统做的一些比较初步的工作，主要讲两个方面：一个是检测，一个是韧性。 首先，从控制这个方式来说，怎么去思考信息物理系统安全性这个问题？当然，并不是说控制就能够彻底解决这个问题，还是需要一个多层的防御。

控制

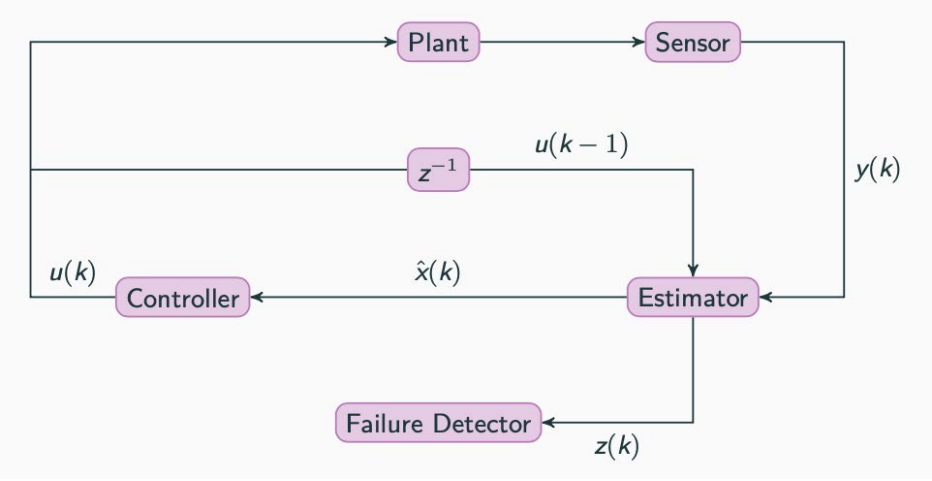
控制，到底能给我们带来什么？以离线设计系统为例，首先我们可以通过控制找到这个系统关键的部件，然后对这些部件做额外的冗余设计。在这个系统没有上线之前，我们可以对系统做一些可控、可观的分析，进而提升系统本身的韧性；上线之后，我们可以利用传统的方法如故障诊断，当然这里就变成了入侵诊断和入侵定位的问题。此外，我们还可以做一些鲁棒控制，保证系统的控制器在有攻击情况下依然能够容错。最后还有一个问题，一个大的系统想要让它更安全，我们到底应该先加固哪个地方？这都是控制可以提供给我们的一些东西。

检测

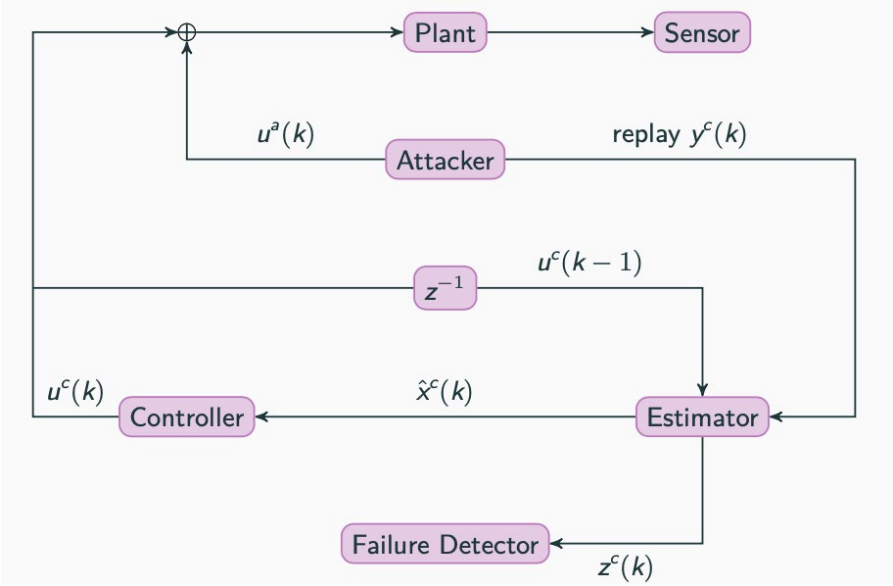
接下来，我主要讲一下关于检测的问题，我们也是受到“震网病毒”例子的启发之后开始做了相关的研究。“震网病毒”是 2010 年被发现的，从计算机的角度来说，它是一个很复杂的、很难防御的病毒。但是从控制角度来说，它的策略其实相当简单。离心机本身是一个类似于快速旋转东西，如果想要毁坏离心机，那就需要让它转的比原本设定的转速快很多，但是如果只是让它转的特别快的话，由于整个系统有传感器，传感器就会发现离心机转速太快，从而触发报警，报警之后就会有技术人员前来检查，整个过程并不能对这个系统造成很大破坏。

但是，“震网病毒”采取的一个策略是先不去攻击这个系统，而是在这个系统正常运行的时候，记录下它的传感器输出是什么情况（比如说离心机的转速）。为了很好地说明问题，我们假设离心机每秒 1000 转，“震网病毒”就记录了很多这样的数据。然后，当真正开始去攻击系统的时候，震网病毒就会把它记录的数据做一个重放，就是

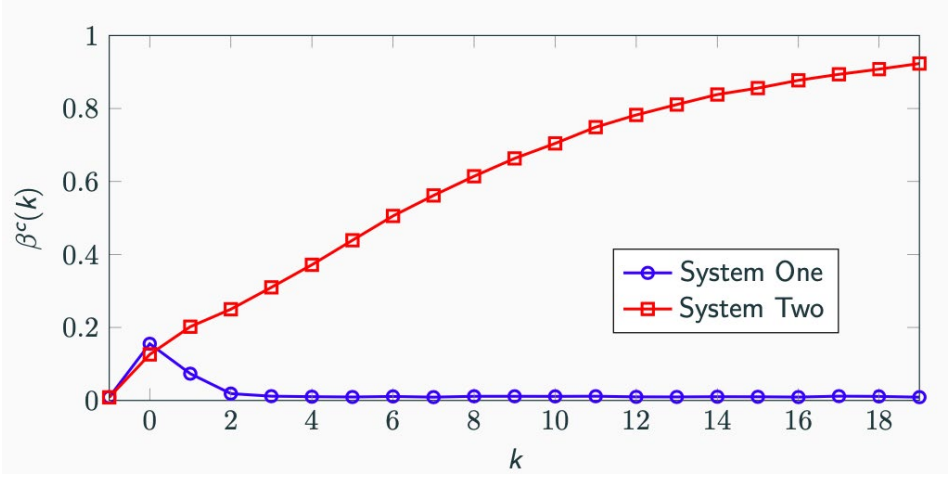
说把整个系统的转速调到比如 2000 转的时候，会使用之前正常的数据去替换异常数据，那么系统操作员看到依然是正常的转速，就不会察觉这个系统已经出现问题了。比如在一些警匪片中，匪徒会去把他们想要抢劫的地方的监控视频的影像做一个重放，比如用前一天没有异常的影像去覆盖掉抢劫的影像，跟“震网病毒”的策略很类似，在信息安全领域也是比较常见的“重放”攻击。



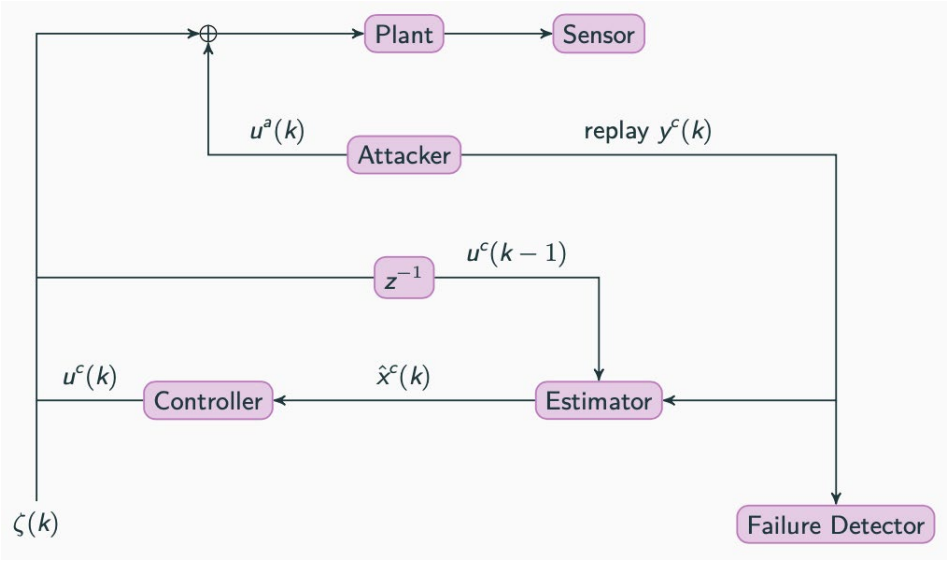
大家可以简单地看一下这个系统的框图，比如说我们做一个离心机或者一个其他的物理系统，需要做控制。那么我们使用传感器去监测这个系统，传感器的输出会给一个估计器，对于无人驾驶来说，可能这个应该叫感知，而不能简单地叫估计，因为它的功能可能会更复杂。但不管怎样的系统，我们都需要通过一个这样的东西，对收集到的信息做一个处理，然后得到系统的状态，再根据状态来设计系统的控制，最后反馈给物理系统。估计器可能还会输出一些信息传递到故障检测器里面，然后故障检测器会检测收到的信息 y(k) 是不是有问题，大概就是这样的一个系统。



对于攻击来说，也分成了两个阶段，第一阶段攻击者先不去修改系统控制这一部分，只是被动的去记录一些传感器的信号。当记录足够多的传感器信号之后，进入第二阶段，开始去修改系统的控制信号，修改这个信号的同时，攻击者还要做的另外一件事，就是要把系统的传感器的这一边给断开，从而把传感器的真实数据替换成之前记录的正常数据。



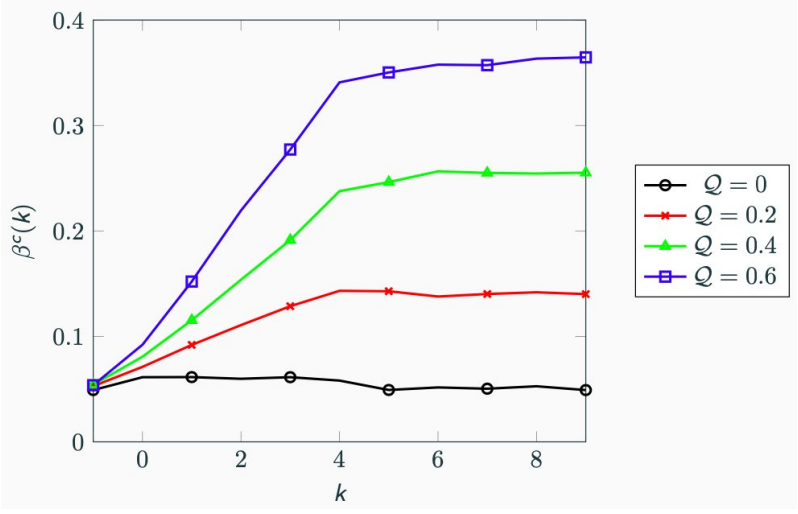
由于系统本身是有故障检测器的，所以最原始的系统在设计的时候根本没有考虑安全性，没有考虑是否能检测出来所谓的“重放”攻击。事实上，“重放”攻击并不是总是有效的，我们发现有一些系统可以检测到“重放”攻击，而有一些系统是不能检测的。如上图所示，Y 轴表示检测器报警的概率，报警概率最大值是 1。整个系统的重放是在时刻零开始，那么作为第一个系统，也就是蓝线标记的系统，我们可以看到在重放开始的时候，它有一个比较短暂的损害过程，就是说这个时候报警的概率有一些，但是也不是特别高，然后很快报警概率就缩减到一个接近于 0 的值；而第二个系统，也就是红线标记的系统，我们发现它的报警概率随着“重放”变得越来越高，最后会趋向于 1。



但是，很多系统跟蓝线标记的系统一样，没有办法检测出“重放”攻击。那就可能会陷入一个问题，攻击者会背着系统的操作人员在系统里面做一些手脚。针对这种问题，我们设计了一种主动检测的方法。刚才所说的检测是被动的检测，通过收集很多的传感器的信息，然后去看传感器的信息是不是和这个系统本身模型是相符合的。 但这个方法存在一个很大的问题，比如控制离心机，它的转速就一直是 1000 转，那么当传感器告诉我们转速是 1000 转时，事实上这个传感器从某种角度来说就没有给我

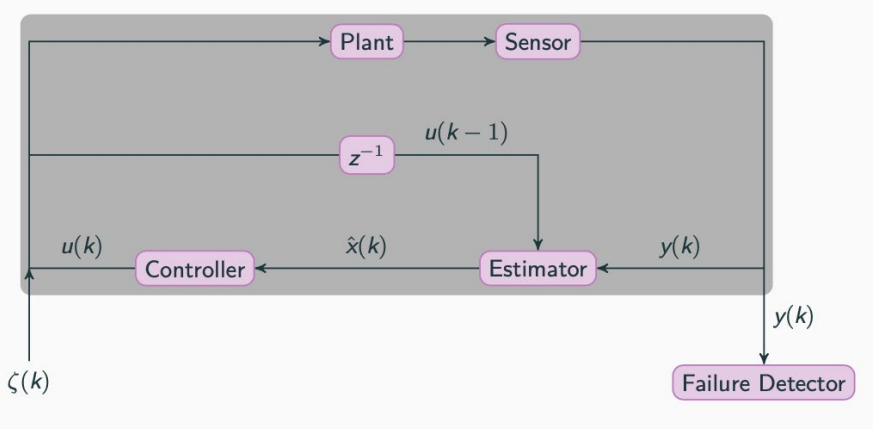
们任何信息，因为系统控制的很好。

我们的想法是能否可以不把系统控制得这么好，我们主动在控制信号中加一个扰动信号，也可以叫做水印信号。这就是说水印信号是藏在真实的控制信号中的一个比较小的噪声。如果我们的系统没有受到攻击的话，那么这个噪声就会被传感器监测到，然后进入估计器，估计器能够在传感器输出的信号中识别这个噪声。当系统遭遇了“重放”攻击，因为系统的这个噪声是完全随机的，那么传感器里面的随机信号跟现在接收到的随机信号就对应不上，因为现在接受到的随机信号是之前的信号“重放”过来的。因此，我们可以通过添加一个这样小的扰动的方式去刺激这个系统，然后让这个系统对这个小的扰动产生一个响应，这样我们就可以去检测出这个系统到底有没有出问题了。我们称这种方式为主动检测方式，主动地去激励系统，而不是被动的去收集信息。其实，这个方法也跟计算机科学领域所说的 Challenge-Response 比较相似，通过给这个系统一个 Challenge，然后这个系统就要返回 Response，这样就能感知到整个系统、整个控制回路是不是都是完好无损的。

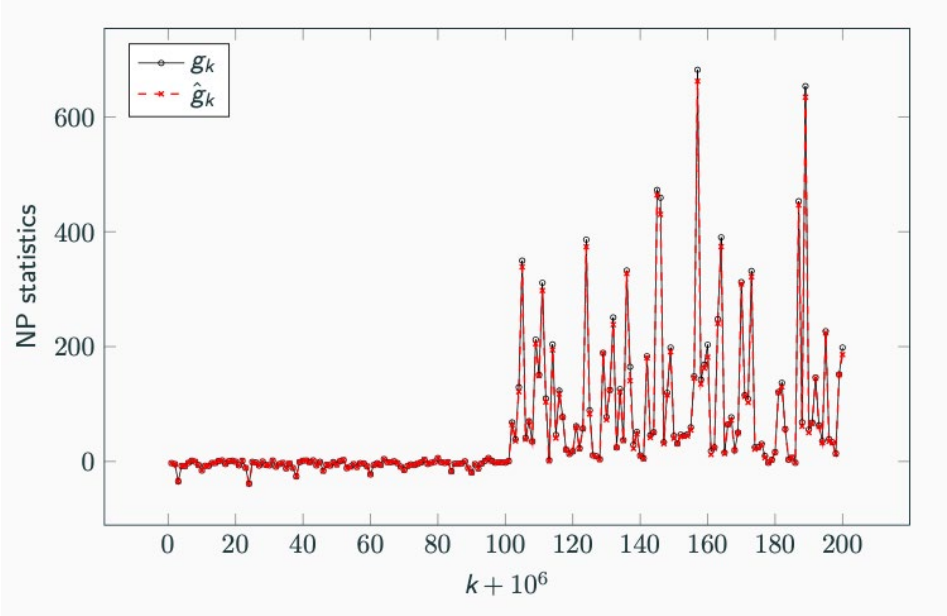


上图是我们做的一些实验的结果。针对的是一个非常简单的系统，大家可以认为这个是噪声的能力，我们发现随着加的能量越高，检测的概率会变得越高。我们大概的处理思路就是这样的，后面会有一些技术的问题，比如说加个扰动信号进去之后，系统

的控制就没那么好，这产生代价会有多大？ 代价和检测性能之间会存在什么样的关系？其中的 Trade Off 到底应该怎么样去做权衡？我们可以把它看成一个类似于优化的问题，然后我们可以去做一个求解。



事实上，我们现在做的这些设计都是基于模型已知的，因为做系统控制通常来说大部分都是假设系统模型已知的。目前，基于数据的方式越来越流行了，我们也尝试去做了一些数据驱动的实验。其中我们需要作一些简单的假设，比如系统本身是稳定的，我们知道 x 到底有几维，其他具体的参数假设是未知的。然后我们可以去做一些数据驱动的方式。想法也很简单，就是我们要在这个地方加一个随机信号，加了这个随机信号，这个系统会产生一些刺激，产生这些刺激之后，我们就可以通过输入和输出的关系来对系统内部的具体参数做一些推断。关于最好的信号具体应该怎么加？检测装置应该怎么去做？这些属于具体的细节，在这里我就不展开仔细讲了。下面是我们针对化工领域常用的 TEP 系统做的一个仿真。虚线是我们在没有模型知识的情况下，通过数据学习了水印信号和检测器的最优设计，得到的检测器的输出。可以看到，跟有模型的实线吻合得非常好。另外，这个系统在 100 时刻收到了“重放”攻击，可以看到我们的水印方法可以有效地检测到攻击。



算法设计

下面我想讲的是一个有韧性的算法到底应该怎么去设计。这里讲的也是一个非常简单的问题，类似于一个简单的状态估计的问题。比如说自动驾驶的汽车里面有很多传感器都可以给这个车做定位，比如雷达、GPS、IMU、视觉传感器，那么我们应该怎么把这个东西给融合起来，这就是一个很传统的状态估计的问题。

这是一个非常简化模型，有很多传感器，每一个传感器都在测量一个叫做状态的东西，这个状态是记做 x，传感器的测量值记做 z。当然这里指的是一个简化的线性高斯模型，就是说测量值是真实状态的一个线性函数加上一定的噪声。比如说最简单的例子：有三个传感器，这三个传感器都在测量位置，这个位置在这里假设它是一个一维的信号，三个传感器都在测量位置，都带有一些噪声。这种情况下融合的规律很简单，就是求这三个测量值的平均值，也可以证明在很多意义下是最大似然，或者是最小均方误差的一个估计，这些都不是很很难。

但是问题是，如果针对这样的一个估计器，假设有一个传感器存在很大的问题，比如

说有一个传感器它可能特别大或者特别小，就会把整体的估计值带偏，那么这就会产生一个非常严重的后果。当然这个问题也可以说非常简单，因为这三个传感器都在测量同样的内容，就知道这三个传感器的测量值应该是接近的，如果其中有一个跟另外两个之间差距很大，那么我们就可以认为这个传感器是存在问题的，就可以把它剔除出去。这其实是一种叫做坏数据检测的想法，就是把跟其他数据不匹配的数据给剔除掉，这样的做法感觉上是比较简单，但是其实也不是那么简单。因为这个模型是一个非常简单的模型，就是三个传感器都在测一个同样的东西、同样的状态，因此可以说如果有一个跟其他两个差得很多，就是这个传感器有问题。但是假设我们在测不同的状态，比如说有的传感器在测这个房间的温度，有的在测走廊的温度，有的在测另一个房间的温度，那么它们之间的数据怎样叫做匹配，怎样叫做不匹配，这个问题就变得很复杂了。再比如说，无人驾驶车的 GPS 和雷达都在测位置，但是两个位置可能是在不同时间测的，假如这个传感器告诉我现在在这里，而另外一个传感器告诉我说下一个时刻在那里，这种情况下判断这两个数据到底匹配不匹配，这个就是一个很复杂的问题。

因此，我们这个地方的想法是能否可以不用这种坏数据检测的方法，因为使用坏数据检测，首先必须要定义什么叫数据匹配。实际上，定义的一般的方法，也是先做一个状态估计，然后通过状态估计去算残差，再通过残差去确定数据是不是匹配，而我们想直接把一个好的状态估计计算出来。所以这里我们考虑这样的一个问题，z 是一个真实的传感器的估计值，我们认为这个估计值等于状态的一个线性函数加上一个噪声，可能会有一些传感器受到攻击，那么要额外在这真实的估计值上加上一个攻击项。z 里面既有噪声也有攻击，噪声一般考虑是一个比较小的数，比如像高斯可能有一个固定的方差，它不会特别大，但是噪声会影响到所有的传感器。而作为攻击来讲，我们认为攻击和噪声不一样，攻击可能是一个任意值，也就是说它可能很大也可能很小，但是这个攻击只能影响有限的传感器，比如整个系统里面有 10 个传感器，可能只有 1 个受到攻击，如果 10 个都受到攻击，当然这个系统基本上就完蛋了，所以一般只有一小部分传感器受到攻击，然后在这种情况下，我们到底应该怎么去解决这个问题。

我们提出利用凸优化的方法去求解这个问题，大概想法是：每一个传感器的测量值，当然这个测量值可能是受到攻击之后的值，假设这个系统既没有攻击也没有噪声的话，那么应该是等于。但是，因为有可能有攻击，也有可能有噪声，所以这个东西肯定是不相等的，就认为它是第个传感器的残差，肯定不是等于 0 的。那么在这种情况下，我们希望找到一个很好的 x 让残差尽可能小，就是希望去最小化这样一个残差的函数。我们在这里假设是凸的，同时它是对称的、非负的。事实上也可以证明有很多种的估计器，都可以写成这种形式，比如说刚才我们说的最小二乘估计器。

我们可以再看一些例子，比如有这样的一个问题，还是跟刚才完全一样，有三个传感器都在监测状态，然后同时有一些噪声，然后假设有一个传感器可能会受到攻击。在这种情况下，如果去优化平方的话，肯定是有问题的，因为平方得出来的是说你的状态估计应该是平均值，平均值本身是不太稳定的。但是你不去优化平方和而是优化绝对值的和，那么这样的话你的估计就会是一个中位数，中位数就是说去掉最大、去掉最小，中间的那个数，那么这样的话，应该是一个比较好的结果。

通过这个东西，我们就可以得到类似于一个安全估计的一个充分条件，就是说你需要保证两件事情，当然这个是从数学上来说比较简单的想法。就是说首先每一个人所能产生的力必须是一个有限的，这个力其实本质上来说就是斜率，就是说它的最大的斜率必须是有限的。然后，另外就是说，你任何 P 个人所产生的力一定要小于剩下的人所产生的力，因为在这里我假设有 P 个人可能是有问题的，这样的话 P 个人会有问题的话，不会把整个系统给带跑，大概是这样。当然，反过来你可以证明这两个条件也是必要的。

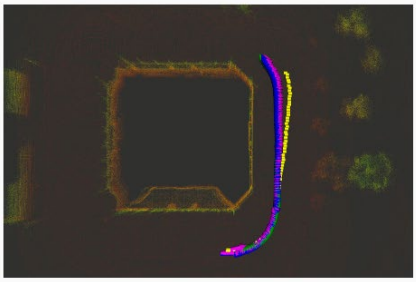


Figure 7: EKF

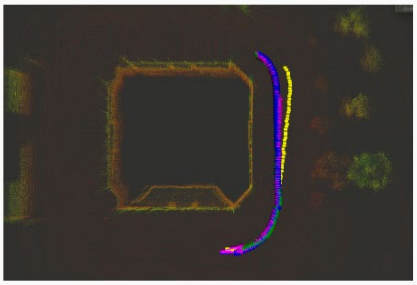


Figure 8: Secure EKF

当然，我们后面还做了很多关于动态系统的，因为时间可能不太多，这里就不展开了。2015 年到 2018 年期间，我在新加坡，当时我们接了新加坡国防部一个关于自动驾驶的项目，他们也是比较关心自动驾驶中的安全问题，这里给大家展示的是我们在一个仿真系统上做的一些东西。

问题是这样的，现在系统里面一共是有三个用于定位的传感器：IMU、雷达、GPS。其中，这条黄色的线是 GPS 告诉我的位置，大家可以看到 GPS 逐渐偏移真实的位置，这个原因是因为我们在这个系统里面加了一个 GPS 的欺骗工具。这种工具也是比较常见，之前也有过伊朗通过 GPS 欺骗攻击，捕获了美国的一个无人机，因为那个无人机认为它已经飞到一个安全的地方，就降落了，但是其实那个地方是伊朗的占领区，然后它就被捕获了。

因为 GPS 信号是一个单方向，通讯员其实是没有办法去确认 GPS 信号是不是真的，所以如果我们有一个发射器，发射一个更强的信号，把真实的 GPS 信号压过去，那么在这种情况下你得到 GPS 就是错的。

这张图是我们采用传统的信息融合方式 EKF 把 IMU、雷达、GPS 做一个融合，粉色的是我们做了一个融合的信息。这个时候，你也可以看到粉色线的已经偏出去了，最后偏的有两米左右这个距离，就大概偏出了一个车道，就因为 GPS 的信号被人劫持了。这张图是我们后面加了一些安全的设计结果，会发现当你偏得比较小的时候，你是没有办法检测出来到底是由噪声引起还是由攻击引起的。

但是，如果当 GPS 偏非常大的时候，我们在这个地方就可以发现 GPS 信号是有问题的，然后我们就会把三个传感器融合变成只用雷达和 IMU 两个传感器去做融合。这样的一个效果，就等于说已经检测出来 GPS 有问题。

总结

今天讲的 Technical 的东西比较多，主要就是想跟大家聊一聊信息物理系统的安全，因为安全本身就是一个挺重要的问题。其次，从控制的角度来说，我们对这一方面的东西有一些自己的思考，可能跟传统的计算机方向相比，大家思考的方式可能也不是特别一样，但是我依然觉得，最终的一个解决方案应该是很多学科共同去协作配合，然后产生一个多层、多种角度、多种手段这样一个防御机制。

我今天主要讲的一个是入侵检测，一个是状态估计这样的两个问题。 事实上这个理念单纯从控制角度来说也面临很多的挑战，现在这个领域大概也就是 10 年左右才能有一些成果，但是我觉得还是要多多学习。我希望做的一些东西，能够给真实的信息物理系统提供一些额外的安全保障。

感谢大家！

KDD Cup 2020 多模态召回比赛季军方案与搜索业务应用

作者：漆毅　坚强　胡可　雷军

背景

美团到店广告平台搜索广告算法团队基于自身的业务场景，一直在不断进行前沿技术的深入优化与算法创新，团队在图学习、数据偏差、多模态学习三个前沿领域均有一定的算法研究与应用，并取得了不错的业务结果。

基于这三个领域的技术积累，团队在 KDD Cup 2020 比赛中选择了三道紧密联系的赛题，希望应用并提升这三个领域技术积累，带来技术与业务的进一步突破。团队的黄坚强、胡可、漆毅、曲檀、陈明健、郑博航、雷军与中科院大学唐兴元共同组建参赛队伍 Aister，参加了 AutoGraph、Debiasing、Multimodalities Recall 三道赛题，最终在 AutoGraph 赛道中获得了冠军（1/149）（解决方案可见：KDD Cup 2020 Debiasing 比赛冠军技术方案与广告业务应用），在 Debiasing 赛道中获得冠军（1/1895）（解决方案可见：KDD Cup 2020 Debiasing 比赛冠军技术方案与广告业务应用），并在 Multimodalities Recall 赛道中获得了季军（3/1433）。

图 1　KDD 2020 会议

要处理自然界、生活中多种模态纠缠、互补着的信息，多模态学习是必由之路。随着互联网交互形态的不断演进，多模态内容如图文、视频等越发丰富；在美团的搜索广

告系统中，也体现出同样的趋势。搜索广告算法团队利用多模态学习相关技术，已在业务上取得了不错的效果，并在今年 KDD Cup 的 Multimodalities Recall 赛道获得了第三名。

本文将介绍 Multimodailites Recall 赛题的技术方案，以及团队在广告业务中多模态学习相关技术的应用与研究，希望对从事相关研究的同学能够有所帮助或者启发。

KDD Cup 2020 Challenges for Modern E-Commerce Platform: Multimodalities Recall

Rank	Team	Members
1	WinnieTheBest	Kuei-Chun Huang, Chi-Yu Yang, Ken-Yu Lin
2	MTDP_CVA	Kai Zuo, Chao Ma, Dongshuai Li, Zuo Cao, Xing Xu
3	aister	Jianqiang Huang, Yi Qi, Ke Hu, Bohang Zheng, Mingjian Chen, Xingyuan Tang, Tan Qu, Jun Lei
4	我是余欢水	Tan Yu, Jie Liu, Hongliang Fei, Shujing Wang, Yi Yang, Jinxing Yu, Yi Li, Zhiqiang Xu, Xin Wang, Ping Li
5	烫烫烫烫烫	Donghai Bian, Guangzhi Sheng, Shuai Jiang, Weihua Peng, Yehan Zheng, Qishi Chen, Fayuan Li, Lu Pan, Tianwei Lin
6	WST	Jie Wu, Lei Zhu, Ying Peng
7	NTT DOCOMO LABS	Toshiki Sakai, Yusuke Fukushima, Ryoki Wakamoto, Masato Hashimoto, Katsuaki Kawashima
8	Vaticinator	Qi Zhang, Lixiang Wang, Changyu Li, Peng Zhang, Shengyuan Zheng, Kai Wang, Yudong Xu, Lei Zhong, Chenyu Jin, Jingjie Li
9	ZJU-NESA	Zhe Ma, Xiaoye Qu, Fenghao Liu, Jianfeng Dong, Shouling Ji
10	垫底小分队	Yuhui Ding, Lei Wu, Chengxi Li, Jieyu Yang, Yue Wang

图 2　KDD Cup 2020 Multimodalities Recall 比赛 TOP 10 榜单

赛题介绍与分析

题目概述

多模态召回赛题由阿里巴巴达摩院智能计算实验室发起并组织，关注电商行业中的多模信息学习问题。2019 年，全世界线上电商营收额已经达到 3530 亿美元。据相关预测，到 2022 年，总营收将增长至 6540 亿美元。大规模的营收和高速增长同时预示着，消费者对于电商服务有着巨大的需求。跟随这一增长，电商行业中各种模态的

信息越来越丰富，如直播、博客等等。怎样在传统的搜索引擎和推荐系统中引入这些多模信息，更好地服务消费者，值得相关从业者深入探讨。

本赛道提供了淘宝商城的真实数据，包括两部分，一是搜索短句（Query）相关，为原始数据；二是商品图片相关，考虑到知识产权等，提供的是使用 Faster RCNN 在图片上提取出的特征向量。两部分数据被组织为基于 Query 的图片召回问题，即有关文本模态和图片模态的召回问题。

为方便理解，本赛道提供了少量真实图片及其对应的原始数据，下面是一个例子。该图例是一个正样例，其 Query 为 Sweet French Dress，图片主体部分是一名身着甜美裙装的女性，主体部分以外，则有大量杂乱信息，包括一个手提包、一些气球以及一些商标和促销文字信息。赛题本身不提供原始图片，而提供的是 Faster RCNN 在图片上提取出的特征向量，即图片中被框出的几个部分。可见，一方面 Faster RCNN 提取了图片中有明显语义的内容，有助于模型学习；另一方面，Faster RCNN 的提取会包含较多的框，这些框体现不出语义的主次之分。怎样利用这些框和文本相匹配，是该赛题的核心内容。

本次赛题设置的评价指标为 NDCG@5。具体来说，在给定的测试集里，每条 Query 会给出约 30 个样本，其中大约 6 条为正样本，其余为负样本。赛题需要选手设计匹配算法，召回出任意 5 条正样本，即可获得该 Query 的全部分数，否则，按照召回的正样本条数来计算 NDCG 指标作为该 Query 的分数。全部 Query 的分数进行平均，即为最终得分。

图 3　Query 和 Product 数据示例

数据分析和理解

本赛道提供了三份数据集，分别称为训练集、验证集和测试集。各个数据集的基本信息如下：

数据集名称	样本条数	有无原始图片	形式	有无label	特征信息
训练集	3,000,000	无	Query-Image对，被视为正样例	无人工标注的正负Label，猜测为点击样本，视为正例	图片ID，图片长宽，目标框数量，目标框图抽取的2048维特征，目标框图的标签，目标框图的位置信息，原始Query，原始Query的ID
验证集	14720	部分有	一条Query下挂多个Image，有正有负	有人工标注的正负Label，公开给选手用于评测	
测试集	29005	无	一条Query下挂多个Image，有正有负	有人工标注的正负Label，不公开，用于比赛评价	

表 1　数据集概况

为进一步探索数据特点，我们将验证集给出的原始图片和特征信息做了聚合展现，下表是一组示例。

Query	正例	负例
breathable and comfortable children's shoes		
men's high collar sweater		
wrought iron nordic-style chair		

表 2　搜索短语与图片的匹配正负例

根据如上探索，我们总结了数据集的三个重要特点：

- 训练集和验证集 / 测试集的数据特点大不相同。训练集量级显著高于验证集 / 测试集，足有三百万条 Query-Image 对，是验证集 / 测试集的一百倍以上。同时，训练集的每条 Query-Image 对均被视为正样本，这和验证集给出的一条 Query 下挂多个有正有负的 Image 截然不同。而通过对验证集原始图片和 Query 进行可视化探索，可见验证集数据质量很高，应该为人工标注。考虑人工标注成本和负样本的缺失，训练集有极大可能描述的是点击关系，而非人工标注的语义匹配关系。我们的解决方案中必须要考虑到训练集分布和测试集分布并不匹配这一基本特点。

- 图片信息复杂，常常包含多个物体。这些物体均被框出，作为给定特征，但各个框之间语义信息并不平等；某些是噪音，如 Query(men's high collar sweater) 下的墨镜、围巾、相机等框图，某些又是因商品展示需要而重复，如 Query(breathable and comfortable children's shoes) 下的重复鞋的框图。平均来说，一张图片有 4 个框，怎么将这多个框包含的语义信息去噪、综合，得到图片的整体语义表达，是建模的一个重点。
- Query 作为给定的原始文本，有着与常用语料截然不同的构造和分布情况。从示例表可见，Query 并非自然语句，而是一些属性和商品实体连缀成的短语。经过统计发现，90% 的 Query 都由 3-4 个单词组成；训练集有约 150 万的不同 Query，其词表大小在 15000 左右；通过最后一个单词，可将全部 Query 归约为大约 2000 类，每一类都是一个具体的商品名词。我们需要考虑文本数据的这些特质，进行针对性处理。

问题挑战

本竞赛是在电商的搜索数据上的一个多模信息匹配任务。从上述数据集的三个特点出发，我们总结了该竞赛的两大主要挑战。

第一，**分布不一致问题**。经典统计机器学习的基础假设是训练集和测试集分布一致，不一致的分布通常会导致模型学偏，训练集和验证集效果难以对齐。我们必须依赖于已有的大规模训练集中的点击信号和小规模的和测试集同分布的验证集，设计可行的数据构建方法和模型训练流程，采取诸如迁移学习等技术，以处理这一问题。

第二，**复杂多模信息匹配问题**。怎么进行多模信息融合是多模态学习中的基础性问题，而怎么对复杂的多模信息进行语义匹配，是本竞赛特有的挑战。从数据看，一方面商品图片多框，信息含量大、噪点多；另一方面，用户搜索 Query 一般具有多个细粒度属性词，且各个词均在语义匹配中发挥作用。这就要求我们在模型设计上针对性处理图和 Query 两方面的复杂性，并做好细粒度的匹配。

针对这两大挑战，下面将详述搜索广告团队的解决方案。

竞赛方案

我们的方案直接回应了上述两个挑战，其主体部分包含两方面的内容，一是通过联合多样化的负采样策略和蒸馏学习以桥接训练数据和测试集的分布，处理分布不一致问题；二是采取细粒度的文本 – 图片匹配网络，进行多模信息融合，处理复杂多模信息匹配问题。最后，通过两阶段训练和多模融合，我们进一步提升了模型表现，整个方案的流程如下图所示。下面详述方案的各个部分。

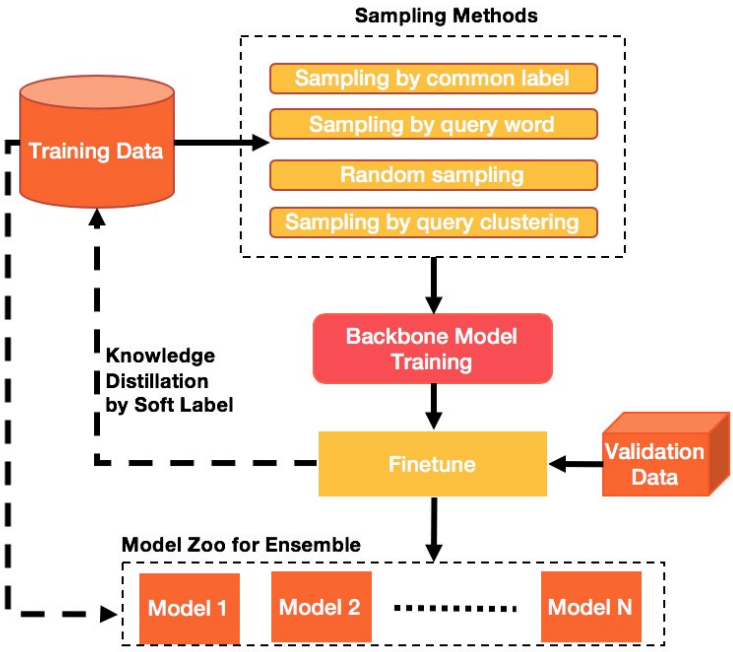


图 4 基于多样化负采样的多阶段蒸馏学习框架

多样负采样策略和预训练

训练集和测试集分布不一致。最直观的不一致是，训练集中只有正样本，没有负样本。我们需要设计负采样策略来构造负样本，并尽可能使得采样出的负样本靠近测试集真实分布。最直观的想法是随机采样。随机采样简单易行，但和验证集区别较大。分析验证集发现，对同一 Query 下的候选图片，通常有着紧密的语义关联。如“甜

美法式长裙”这一 Query 下，待选的图片全是裙装，只是在款式上有不同。这说明，这一多模匹配赛题需要在较细的属性粒度上对文本和图片进行匹配。从图片标签和 Query 词两个角度出发，我们可以通过相应的聚类算法，使得待采样的空间从全局细化为相似语义条目，从而达到负采样更贴近测试集分布的目的。

基于如上分析，我们设计了如下表所示的四种采样策略来构建样本集。这四种策略中，随机采样得到的正负样本最容易被区分，按 Query 最后一词采样得到的正负样本最难被区分；在训练中，我们从基准模型出发，先在最简单的随机采样上训练基准模型，然后在更困难的按图片标签采样、按 Query 的聚类采样的样本集上基于先前的模型继续训练，最后在最难的按 Query 最后一词采样的样本集上训练。这样由易到难、由远到近的训练方式，有助于模型收敛到验证集分布上，在测试集上取得了更好的效果。

采样策略	描述	与验证集贴近程度
随机采样	对训练集的每个Query，在训练集中随机采样图片作为负样本。	弱
按图片标签采样	将全部图片按照其标签（数据集中给定，由Faster RCNN生成）聚类；对训练集中每个Query，在其对应的图片所属的类中，进行随机采样，作为其负样本；原始样本为正样本。	中等
按Query最后一词	将全部Query按照最后一词聚类；对训练集中每张图片，在其对应的query所属的类的，进行随机采样，作为其负样本；原始样本为正样本。	较强
按Query的聚类采样	将全部Query按照Word Embedding进行聚类；对训练集中每张图片，在其对应的Query所属的类的，进行随机采样，作为其负样本；原始样本为正样本。	较强

表 3　多样化负采样

蒸馏学习

尽管使用多种采样策略，可从不同角度去逼近测试集的真实分布，但由于未直接利用测试集信息指导负采样，这些采样策略仍有不足。因而，我们采用蒸馏学习的办法，来进一步优化负采样逻辑，以求拿到更贴近测试集的样本集分布。如下图所示，在通过训练集负采样得到的样本集上预训练以后（第 1 步），我们将该模型在验证集上进一

步 Finetune，得到微调模型（第 2 步）。利用微调模型，我们反过去在训练集上打伪标签，作为 Soft Label，并把 Soft Label 引入 Loss，跟原始的 0-1 Hard Label 联合学习（第 3 步）。这样，训练集的训练上，即直接引入了验证集的分布信息，进一步贴近了验证集分布，提升了预训练模型的表现。

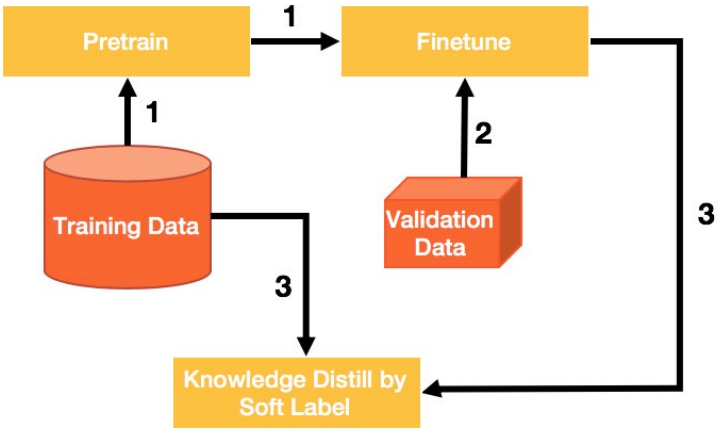


图 5　多阶段蒸馏学习

细粒度匹配网络

多模态学习方兴未艾，各类任务、模型层出不穷。针对我们面临的复杂图片和搜索 Query 匹配的问题，参照 CVPR 2017 的 VQA 竞赛的冠军方案，我们设计了如下的神经网络模型作为主模型。

该模型的设计主要考虑了如下三点：

- 利用带门全连接网络做语义映射。图片和 Query 处于不同语义层级，需利用函数映射到相同的语义空间，我们采取了两个全连接层的方式达到该目的。实验发现，全连接层的隐层大小是比较敏感的参数，适当增大隐层，可在不过分增加计算复杂度的情况下，显著提升模型效果。此外，如文献所述，使用带门的全连接层可进一步提升语义映射网络的效果。

- 采用双向 Attention 机制。图片和 Query 均由更细粒度的子语义单元组成。具体来说，一张图片上可能有多个框，每个框均有独立的语义信息；一个 Query 分为多个词，每个词也蕴含独立的语义信息。这一数据特点是由电商搜索场景决定的。因而，在模型设计时，需考虑到单个子语义单元之间的匹配。我们采用单个词和全部框、单个框和全部词双方向的注意力机制，去捕捉这些子单元的匹配关系和重要程度。
- 使用多样化多模融合策略。多模信息融合有很多手段，大部分最终归结为图片向量和 Query 向量之间的数学操作符。考虑到不同融合方式各有特点，多样融合能够更全面地刻画匹配关系，我们采用了 Kronecker Product、Vector Concatenation 和 Self-Attention 三种融合方式，将经过语义空间转化和 Attention 机制映射后的图片向量和 Query 向量进行信息融合，并最终送入全连接神经网络，得到匹配与否的概率值。

此外，我们采用在训练集样本上预训练词向量的方式得到原始 Query 的表示，而非使用 BERT 模型等流行的预训练模型。这里的主要考虑是，数据分析指出，Query 和常见的自然语句很不同，而更像是一组特定属性 / 品类名词组合在一起的短语，这和 BERT 等预训练模型所使用的语料有明显差异。事实上，我们初步尝试引入 Glove 预训练词向量等，和直接在 Query 文本上预训练相比，并无明显收益。再考虑到 BERT 模型比较笨重，不利于快速迭代，我们最终没有使用相关的语言模型技术。

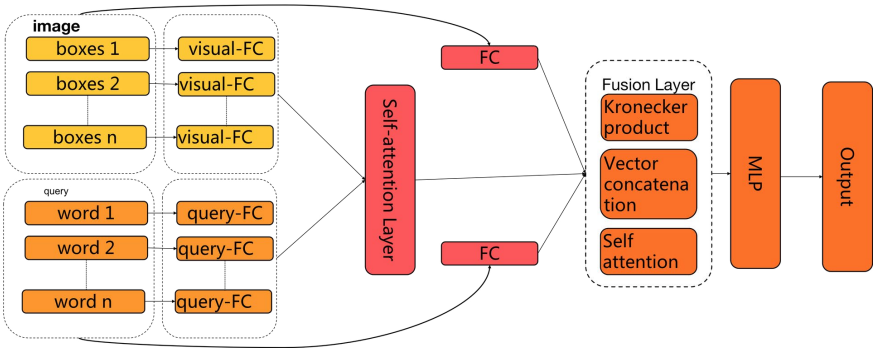


图 6　细粒度匹配网络

多模融合

在上述技术手段的处理下，我们得到了多个基础模型。这些模型均可在验证集上进行 Finetune，从而使其效果更贴近真实分布。一方面，Finetune 阶段可继续使用前述的神经网络匹配模型。另一方面，前述神经网络可作为特征提取器，将其在规模较小的验证集上的输出，放入树模型重新训练。这一好处是树模型和神经网络模型异质性大，融合效果更好。最终，我们提交的结果是多个神经网络模型和树模型融合的结果。

评估结果

我们以随机采样训练的粗粒度（图片表示为所有框的平均，Query 表示为所有词的平均）匹配网络为基准模型。下表列出了我们解决方案的各个部分在基准模型上的提升效果。

方法	NDCG提升比例
多样化采样	+ 4.8%
蒸馏学习	+ 0.5%
细粒度匹配网络	+ 1.9%
多模融合	+ 3.5%

表 4　不同方法的 NDCG 提升

广告业务应用

搜索广告算法团队负责美团与点评双平台的搜索广告与筛选列表广告业务，业务类型涉及餐饮、休闲娱乐、丽人、酒店等，丰富的业务类型为算法优化带来很大空间与挑战。搜索广告中的创意优选阶段，目的在通过当前搜索词或者筛选意图，为用户的每

一个广告展示结果选择高质量的图片。用户的搜索词与图片在维度，表达粒度均有较大差异，我们采用多模态学习来解决这一问题，将跨模表达进行同空间映射。如下图所示，在多模态网络中，将广告特征、请求特征、用户偏好连同图片特征作为输入，其中图片特征通过 CNN 网络提取图片向量表示，其他特征通过多层 MLP 进行交叉得到稠密向量表示，最终通过图片 Loss 和多模 Loss 两个损失函数约束模型训练。通过这样的建模方式，创意优选模型可以根据查询为不同用户的广告结果呈现最合适的图像。

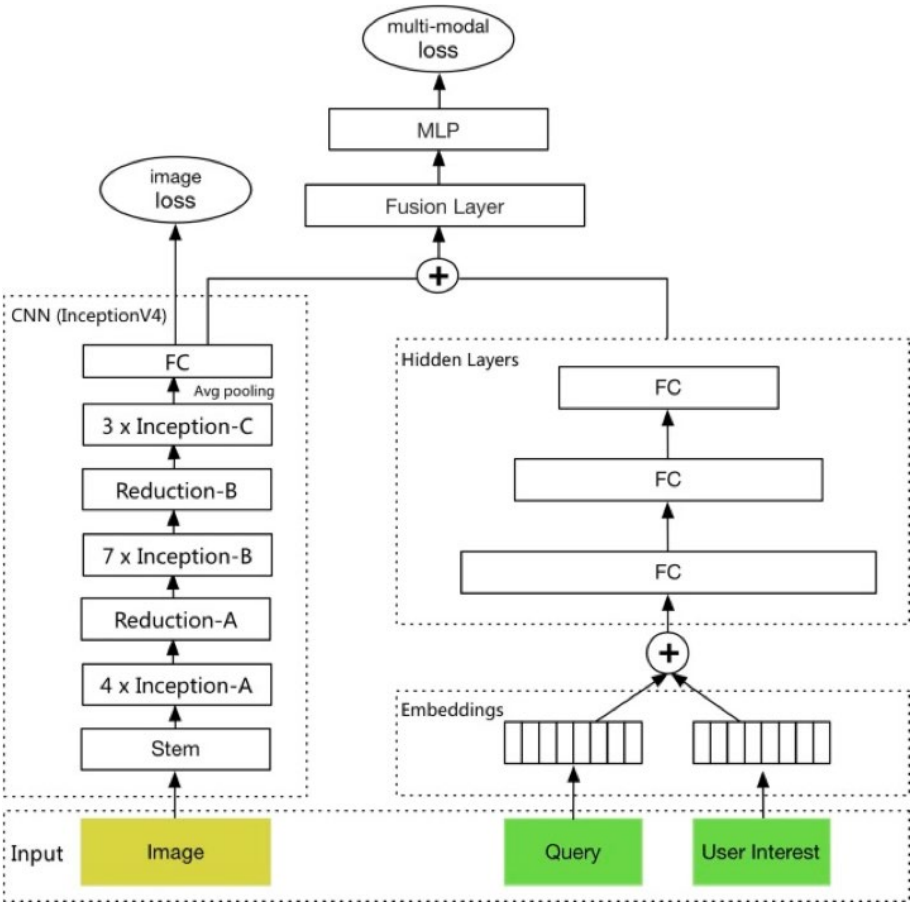


图 7　广告创意业务中的多模态学习

搜索广告系统分为广告触发、创意优选，点击率预估（广告粒度）等模块。其中，创意优选阶段对于每个广告结果有超过十张的图片候选，线上服务的计算量是点击率预

估（广告粒度）的十倍以上，对性能有更高的要求。而为了缩短耗时而减少模型复杂度又必然导致模型精度的的下降。

为了平衡模型的性能和效果，我们借鉴了知识蒸馏的思路来处理这一难题，借用了高表达能力的广告粒度预估模型。如上图 7 所示，左侧模型为复杂的广告粒度点击率预估模型，可以作为教师网络；右侧为简单的创意粒度优选模型，作为学生网络。学生网络的目标损失函数中，除学生网络自身输出 Logit 的 Logloss 以外，还加入了其 Logit 和老师网络输出 Logit 之间的平方误差。这一辅助 Loss 能够迫使学生模型的输出和老师模型的输出更接近。因此，学生模型可以学得与老师模型更接近，从而达到保持相对简单网络规模的同时、提升精度的目的。

除此以外，底层共享 Embedding 的设计，也使得学生模型的底层参数可得到老师模型的训练。并且，在提升精度的同时，多模块之间的一致性（例如 CTR 预估与创意优选）也是系统精度提高的一个关键，在目标与表达学习的 Teacher-Student 联合训练有利于多阶段的目标统一。基于精度提升与多阶段目标的一致性，我们取得线上业务效果较为显著的提升。

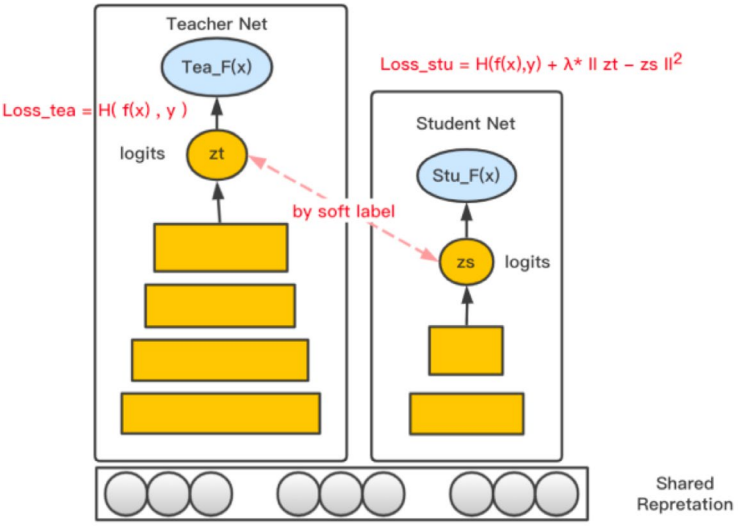


图 8　广告创意业务中的蒸馏学习

总结与展望

KDD Cup 是同工业界联接非常紧密的比赛，每年赛题紧扣业界热点问题与实际问题，其中历年产出的 Winning Solution 对工业界有很大影响。例如，KDD Cup 2012 产出了 FFM (Feild-Aware Factorization Machine) 与 XGBoost 的原型，在工业界取得广泛应用。今年的 KDD Cup 主要关注在自动化图表示学习以及推荐系统等领域上。自然界的信息常常是多种模态混合的，对多模信息的处理和处理是近年来的一大研究热点。同时在工业界的搜索引擎或推荐系统中，涉及到的多模信息处理等，正变得越来越重要。特别是随着直播、短视频等业务形态的兴起，多模态学习已变得不可或缺。

本文主要介绍了 KDD CUP 2020 的多模态竞赛情况以及美团搜索广告算法团队的解决方案。对数据进行充分探索后，我们分析出竞赛数据的三大特点，同时定位了赛题有两大挑战，即训练集和测试集分布不一致和复杂多模信息匹配。我们通过多样化负采样策略、蒸馏学习和预训练与 Finetune 等技术处理了分布不一致问题，并通过细粒度匹配网络处理复杂多模信息匹配问题，两方面思路均取得了效果的显著提升。同时，本文还介绍了多模态学习相关技术在搜索广告业务中的实际应用情况，包括创意优选模型中的图片和用户偏好联合学习、蒸馏学习在创意模型中的应用等。通过比赛高强度、快频率的迭代，团队在多模态学习方面有了更深的理解。在未来的工作中我们会基于本次比赛取得的经验，深入更多的多模态业务场景中进行分析和建模，发挥数据的价值。

参考文献

[1] Teney, Damien, et al. “Tips and tricks for visual question answering: Learnings from the 2017 challenge.” Proceedings of the IEEE conference on computer vision and pattern recognition. 2018.

[2] Hinton, Geoffrey, Oriol Vinyals, and Jeff Dean. “Distilling the knowledge in a neural network.” arXiv preprint arXiv:1503.02531 (2015).

[3] Pennington, Jeffrey, Richard Socher, and Christopher D. Manning. “Glove: Global vectors for word representation.” Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP). 2014.

[4] Devlin, Jacob, et al. “Bert: Pre-training of deep bidirectional transformers for language understanding.” arXiv preprint arXiv:1810.04805 (2018).

[5] Zhou, Bolei, et al. “Simple baseline for visual question answering.” arXiv preprint arXiv:1512.02167 (2015).

[6] Yu, Zhou, et al. “Deep modular co-attention networks for visual question answering.” Proceedings of the IEEE conference on computer vision and pattern recognition. 2019.

作者简介

漆毅，坚强，胡 可，雷军等，均来自美团广告平台搜索广告算法团队。

关于美团 AI

美团 AI 以“帮人们吃得更好，生活更好”为核心目标，致力于在实际业务场景需求上探索前沿的人工智能技术，并将之迅速落地在实际生活服务场景中，完成线下经济的数字化。
美团 AI 诞生于美团丰富的生活服务场景需求之上，具有场景驱动技术的独特性与优势。以业务场景与丰富数据为基础，通过图像识别、语音交互、自然语言处理、配送调度技术，落地于无人配送、无人微仓、智慧门店等真实场景下，覆盖人们生活的方方面面，用科技助力用户生活质量提升，产业智能化升级乃至整个社会的生活服务新基建建设。
更多信息请访问：https://ai.meituan.com/

招聘信息

美团广告平台搜索广告算法团队立足搜索广告场景，探索深度学习、强化学习、人工智能、大数据、知识图谱、NLP 和计算机视觉最前沿的技术发展，探索本地生活服务电商的价值。主要工作方向包括：

- **触发策略**：用户意图识别、广告商家数据理解，Query 改写，深度匹配，相关性建模。
- **质量预估**：广告质量度建模。点击率、转化率、客单价、交易额预估。
- **机制设计**：广告排序机制、竞价机制、出价建议、流量预估、预算分配。
- **创意优化**：智能创意设计。广告图片、文字、团单、优惠信息等展示创意的优化。

岗位要求：

- 有三年以上相关工作经验，对 CTR/CVR 预估，NLP，图像理解，机制设计至少一方面有应用经验。
- 熟悉常用的机器学习、深度学习、强化学习模型。
- 具有优秀的逻辑思维能力，对解决挑战性问题充满热情，对数据敏感，善于分析 / 解决问题。
- 计算机、数学相关专业硕士及以上学历。

具备以下条件优先：

- 有广告 / 搜索 / 推荐等相关业务经验。
- 有大规模机器学习相关经验。

感兴趣的同学可投递简历至：tech@meituan.com（邮件标题请注明：广平搜索团队）。

对话任务中的“语言 – 视觉”信息融合研究

作者：会星　子彭　方向　小捷　玉树　仲远等

目标导向的视觉对话是“视觉 – 语言”交叉领域中一个较新的任务，它要求机器能通过多轮对话完成视觉相关的特定目标。该任务兼具研究意义与应用价值。日前，北京邮电大学王小捷教授团队与美团 AI 平台 NLP 中心团队合作，在目标导向的视觉对话任务上的研究论文《Answer-Driven Visual State Estimator for Goal-Oriented Visual Dialogue-commentCZ》被国际多媒体领域顶级会议 ACMMM 2020 录用。

该论文分享了他们在目标导向视觉对话中的最新进展，即提出了一种响应驱动的视觉状态估计器（Answer-Driven Visual State Estimator，ADVSE）用于融合视觉对话中的对话历史信息和图片信息，其中的聚焦注意力机制（Answer-Driven Focusing Attention，ADFA）能有效强化响应信息，条件视觉信息融合机制（Conditional Visual Information Fusion，CVIF）用于自适应选择全局和差异信息。该估计器不仅可以用于生成问题，还可以用于回答问题。在视觉对话的国际公开数据集 GuessWhat?! 上的实验结果表明，该模型在问题生成和回答上都取得了当前的领先水平。

背景

一个好的视觉对话模型不仅需要理解来自视觉场景、自然语言对话两种模态的信息，还应遵循某种合理的策略，以尽快地实现目标。同时，目标导向的视觉对话任务具有较丰富的应用场景。例如智能助理、交互式拾取机器人，通过自然语言筛查大批量视觉媒体信息等。

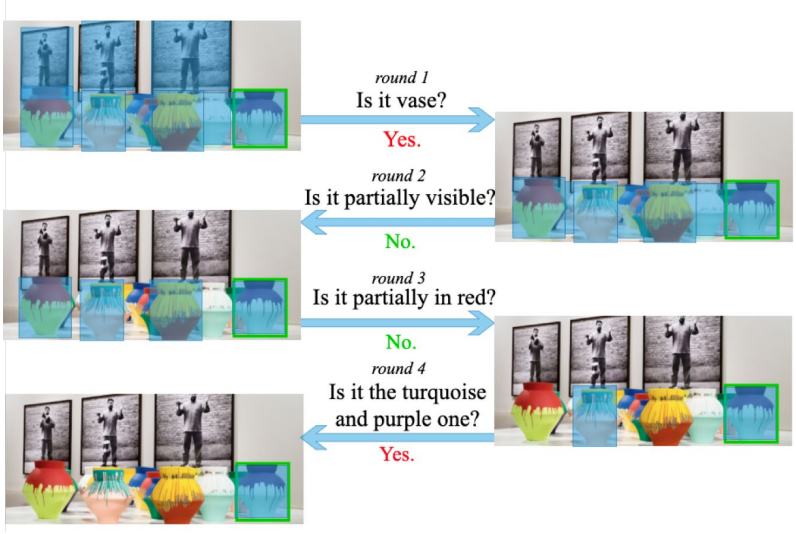


图 1　目标导向的视觉对话

研究现状及分析

为了进行目标导向的和视觉内容一致的对话，AI 智能体应该能够学习到视觉信息敏感的多模态对话表示以及对话策略。对话策略学习的相关工作有很多，如 Strub 等人 [1] 首先提出使用强化学习来探索对话策略，随后的工作则着重于奖励设计 [2，3] 或动作选择 [4，5]。但是，它们中的大多数采用了一种简单的方式来表示多模态对话，分别编码两个模态信息，即由 RNN 编码的语言特征和由预训练 CNN 编码的视觉特征，并将它们拼接起来。

好的多模态对话表示是策略学习的基石。为了改进多模态对话的表示，研究者们提出了各种注意机制 [6，7，8]，从而增强了多模态交互。尽管已有工作取得了许多进展，但是还存在一些重要问题。

1. 在语言编码方面，现有方法的语言编码方式都不能对不同的响应（Answer）进行区分，Answer 通常只是附在 Question 后面编码，由于 Answer 只是 Yes 或 No 一个单词，而 Question 则包含更长的词串，因此，Answer 的作用很微弱。但实际上，Answer 的回答很大程度决定了后续图像关注区域的

变化方向，也决定了对话的发展方向，回答是 Yes 和 No 会导致完全不同的发展方向。例如图 1 中通过对话寻找目标物体的示例，当第一个问题的答案“是花瓶吗?”为“是”，则发问者继续关注花瓶，并询问可以最好地区分多个花瓶的特征；当第三个问题的答案“部分为红色吗?”为“否”，则发问者不再关注红色的花瓶，而是询问有关剩余候选物体的问题。

2. 在视觉以及融合方面的情况也是类似，现有的视觉编码方式或者采用静态编码在对话过程中一直不变，直接和动态变化的语言编码拼接，或者用 QA 对编码引导对视觉内容的注意力机制。因此，也不能对不同的 Answer 进行有效区分。而如前所述，当 Answer 回答不同时，会导致图像关注区域产生非常不同的变化，一般地，当回答为“是”时，图像会聚焦于当前对象，进一步关注其特点，当回答为“否”时，可能需要再次关注图像整体区域去寻找新的可能候选对象。

响应驱动的视觉状态估计器

为此，本文提出一个响应驱动的视觉状态估计器，如下图 2 所示，新框架中包含响应驱动的注意力更新（ADFA-ASU）以及视觉信息的条件融合机制（CVIF）分别解决上述两个问题。

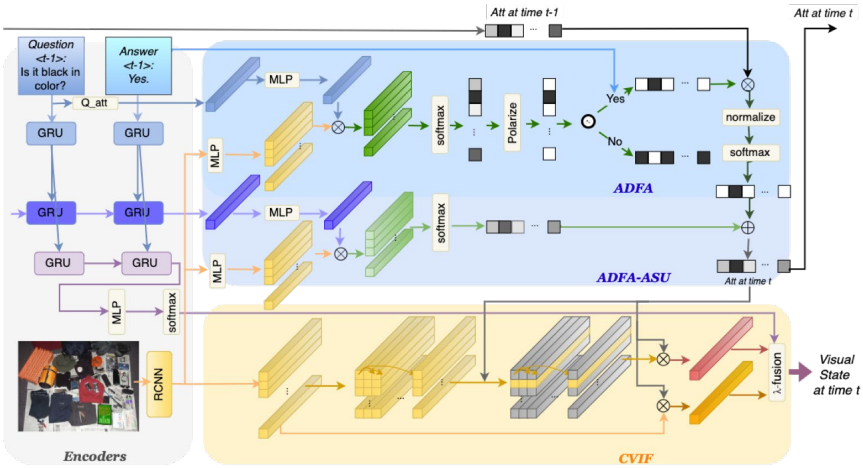


图 2　响应驱动的视觉状态估计器框架图

响应驱动的注意力更新首先采用门限函数极化当前轮次 Question 引导的注意力，随后基于对该 Question 的不同 Answer 进行注意力反转或保持，得到当前 Question-Answer 对对话状态的影响，并累积到对话状态上，这种方式有效地强调了 Answer 对对话状态的影响；CVIF 在当前 QA 的指导下融合图像的整体信息和当前候选对象的差异信息，从而获得估计的视觉状态。

答案驱动的注意力更新（ADFA-ASU）

Questioner在视觉对话过程中，会依据对话历史和当前对话输入改变其对图像的关注，本文称之为对话关注状态att_t，其由当前对话引起的关注更新att_t^q和对话历史引起的累积关注att_t^h两方面共同决定，即$att_t = att_t^q + att_t^h$。下面分别叙述对$att_t^q$和$att_t^h$的具体建模。

首先，介绍由当前对话引起的关注更新，其建模包括如下三步：

- 第一步，以当前轮问题q_t引导对图像区块的关注，得到注意力分布α_t^q。
- 第二步，对α_t^q进行极化，以完全地筛选出当前关注的区块。我们引入一种注意力极化操作，将每个区块i_k对应的注意力权重$\alpha_{t,k}^q \in \alpha_t^q \in \{0,1\}$映射为，从而筛选出当前问题的关注，并根据当前问题对应的回答a_t调整极化方向，得到注意力掩码M_t^q。首先，对α_t^q进行最大最小值归一化得到$norm(\alpha_t^q)$；而后，采用极化操作将区块i_k开始对应的极化注意力转化为一个布尔值$p(a_{t,k}^q)$，代表着i_k是否对应着q_t问到的图像内容。最后，将问题对应的回答来决定基于极化注意力的掩码的方向。如果答案为“是”，说明当前问题锁定到了目标对象，注意力掩码为$p(\alpha_t^q)$；如果答案为“否”，说明当前问题问到的内容不是目标对象，则注意力掩码为$1 - p(\alpha_t^q)$；如果答案是“n/a”，则不作筛选。
- 第三步，更新关注状态att_t^q。将注意力掩码作用于上一轮的关注状态att_{t-1}，归一化和$masked\ softmax$被用于调整累积关注状态att_t^q：

$$att_t^q = maskedSoftmax(Norm(M_t^q \odot att_{t-1})), \quad (1)$$

其次，是对话历史引导的注意力分布：

$$att_t^h = softmax(W_F^H(W_t^H H_t \odot W_I^H I))), \quad (2)$$

将其与att_t^q相加，得到新的关注状态att_t。在att_t中，目标对象的信息被强化。就这样，随着每一轮QA对的生成，关注状态被逐轮更新。

视觉信息的条件融合机制（CVIF）

在差异信息融合机制中，我们依据在注意力聚焦机制中计算得到的当前关注状态att_t，分别计算得到关注图像信息I^t_{att}和关注差异信息D^t_{att}，在当前QA对的引导下融合二者，得到当前的视觉信息编码V_t。

首先，对于区块k，其差异信息D_k为其与其他所有区块图像特征的差：

$$D_k = \{i_k - i_j\}_{j=1}^{N}, k \in \{1, 2, \ldots, N\}, \tag{3}$$

根据注意力聚焦机制计算得到的注意力分布att_t，选择出注意力权重最大的图像区块$i^t_{selected}$，使用att_t对其对应的差异信息$D^t_{selected}$加权求和，得到关注差异信息D^t_{att}：

$$selected_t = argmax(att_t), D^t_{selected} = \{i^t_{selected} - i_j\}_{j=1}^{N}, \tag{4}$$

$$D^t_{att} = D^t_{selected} \otimes att_t, \tag{5}$$

而关注图像信息为：

$$I^t_{att} = I \otimes att_t, \tag{6}$$

当前QA对的信息P_t由GRU_p编码得到，这一信息被映射到二维向量并由$softmax$激活为一二项分布$(\lambda_t, 1-\lambda_t)$，代表着当前QA对锁定到目标候选集的自信度。由λ_t实现对I^t_{att}和D^t_{att}的融合，得到当前轮的视觉信息V_t：

$$h^p_{t,q} = GRU_p(q_t, h_0), p_t = GRU_p(h_t, q^p, \alpha_t), \tag{7}$$

$$(\lambda_t, 1-\lambda_t) = softmax(W_p P_t), \tag{8}$$

$$V_t = \lambda_t \odot I^t_{att} + (1-\lambda_t) \odot D^t_{att}. \tag{9}$$

视觉状态估计是将差异信息与基于当前QA对的整体信息进行的软融合，从而再次增强了当前响应的影响力。

响应驱动的视觉状态估计器用于问题生成和回答

ADVSE 是面向目标的视觉对话的通用框架。因此，我们将其应用于 GuessWhat ?! 中的问题生成（QGen）和回答（Guesser）建模。我们首先将 ADVSE 与经典的层级对话历史编码器结合起来以获得多模态对话表示，而后将多模态对话表示与解码器联合则可得到基于 ADVSE 的问题生成模型；将多模态对话表示与分类器联合则得到基于 ADVSE 的回答模型。

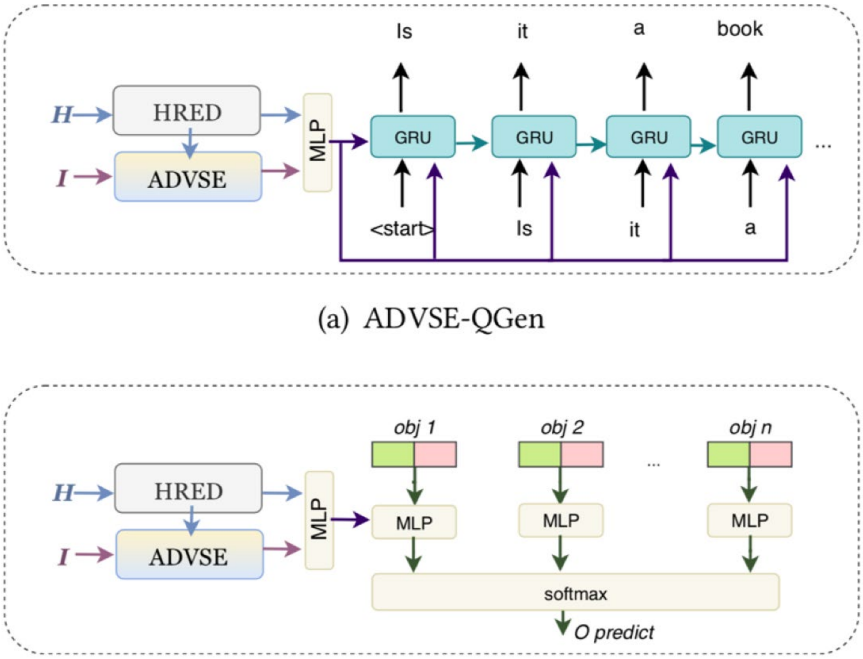


图 3　响应驱动的视觉状态估计器用于问题生成和回答示意图

在视觉对话的国际公开数据集 GuessWhat?! 上的实验结果表明，该模型在问题生成和回答上都取得了当前的领先水平。我们首先给出了 ADVSE-QGen 和 ADVSE-Guesser 与最新模型对比的实验结果。

此外，我们评测了联合使用 ADVSE-QGen 和 ADVSE-Guesser 的性能。最后，我们给出了模型的定性分析内容。我们模型的代码即将可从 ADVSE-GuessWhat 获得。

	Approach	(%)New object				(%)New game			
		Sampling	Greedy	Beam-search	Best	Sampling	Greedy	Beam-Search	Best
Guesser[22]	SL	41.6	43.5	47.1	47.1	39.2	40.8	44.6	44.6
	DM	-	-	-	-	-	-	-	42.19
	VDST-SL	45.02	49.49	-	49.49	44.24	45.94	-	45.94
	ADVSE-QGen	**47.55**	**50.66**	**47.47**	**50.66**	**44.75**	**47.03**	**44.70**	**47.03**
ADVSE-Guesser	**ADVSE-QGen**	**48.01**	**54.06**	**50.66**	**54.06**	**46.32**	**50.94**	**47.89**	**50.94**
Guesser[22]	RL	62.8	58.2	53.9	62.8	60.8	56.3	52.0	60.8
	VQG	63.2	63.6	63.9	63.9	59.8	60.7	60.8	60.8
	Bayesian	61.4	62.1	63.6	63.6	59.0	59.8	60.6	60.6
	GDSE-C	-	-	-	63.3	-	-	-	60.7
	ISM	-	64.2	-	64.2	-	62.1	-	62.1
	TPG	-	-	-	-	-	-	-	62.6
	RIG-1	65.20	63.00	63.08	65.20	64.06	59.00	60.21	64.06
	RIG-2	67.19	63.19	62.57	67.19	65.79	61.18	59.79	65.79
	VDST-RL	69.51	70.55	71.03	71.03	66.76	67.73	67.52	67.73
	ADVSE-QGen	**71.26**	**72.73**	**72.24**	**72.73**	**68.82**	**69.88**	**69.88**	**69.88**
ADVSE-Guesser	**ADVSE-QGen**	**72.38**	**73.59**	**73.73**	**73.73**	**70.61**	**71.10**	**71.27**	**71.27**

表 1　QGen 任务性能对比，评测指标为任务成功率

Model	(%)Test err
HRED	39.0
HRED+VGG	39.6
PLAN	36.6
A-ATT	35.8
Single-hop FiLM	35.7
Multi-hop FiLM	35.0
HACAN	34.1
ADVSE-Guesser	**33.15**

表 2　Guesser 任务性能对比，评测指标为错误率

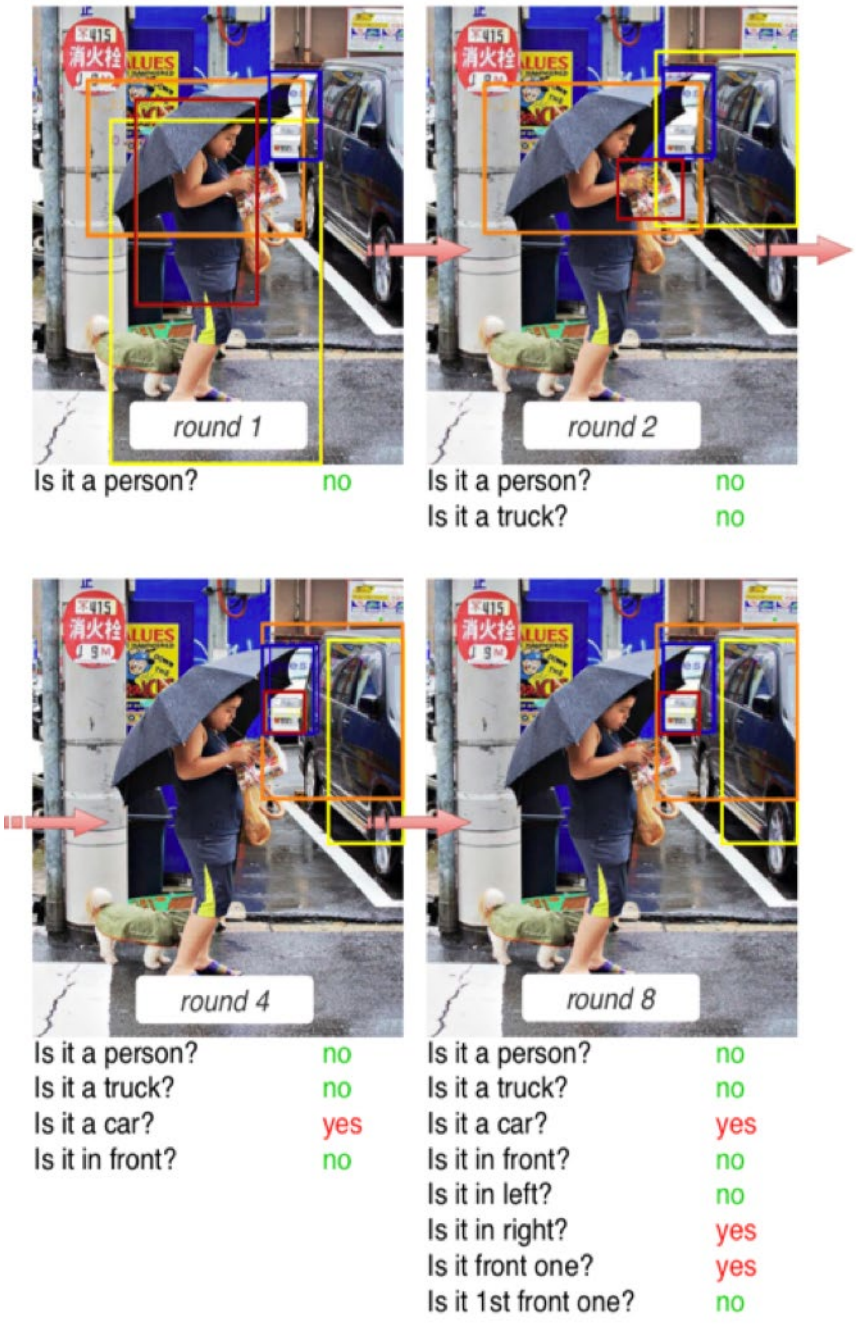


图 4　问题生成过程中响应驱动的注意力转移样例分析

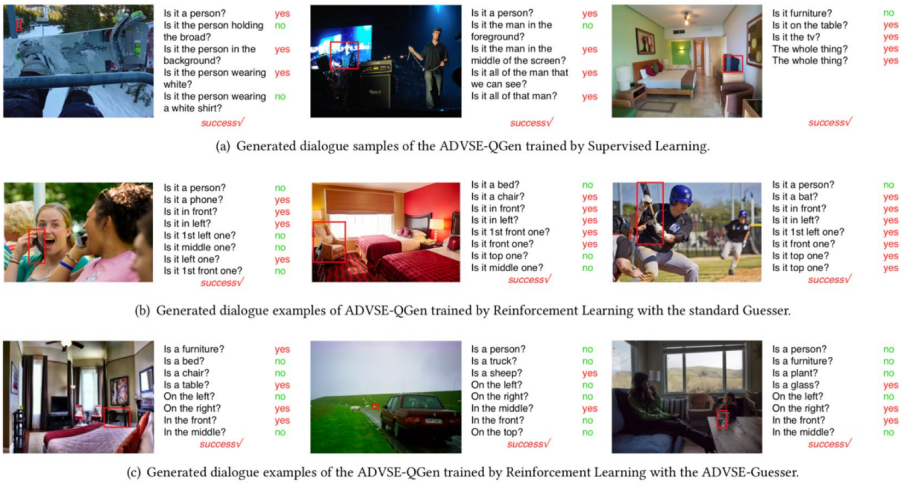


图 5　ADVSE-QGen 对话生成样例

总结

本论文提出了一种响应驱动的视觉状态估计器（ADVSE），以强调在目标导向的视觉对话中不同响应对视觉信息的重要影响。首先，我们通过响应驱动的集中注意力（ADFA）捕获响应对视觉注意力的影响，其中是保持还是移动与问题相关的视觉注意力由每个回合的不同响应决定。

此外，在视觉信息的条件融合机制（CVIF）中，我们为不同的 QA 状态提供了两种类型的视觉信息，然后依情况地将它们融合，作为视觉状态的估计。将提出的 ADVSE 应用于 Guesswhat?! 中的问题生成任务和猜测任务，与这两个任务的现有最新模型相比，我们可以获得更高的准确性和定性结果。后续，我们还将进一步探讨同时使用同源的 ADVSE-QGen 和 ADVSE-Guesser 的潜在改进。

参考文献

[1] FlorianStrub,HarmdeVries,J é r é mieMary,BilalPiot,AaronC.Courville,and Olivier Pietquin. 2017. End-to-end optimization of goal-driven and visually grounded dialogue systems. In Joint Conference on Artificial Intelligence.

[2] Pushkar Shukla, Carlos Elmadjian, Richika Sharan, Vivek Kulkarni, Matthew

Turk, and William Yang Wang. 2019. What Should I Ask? Using Conversationally Informative Rewards for Goal-oriented Visual Dialog.. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. Association for ComputationalLinguistics,Florence,Italy,6442 – 6451. https://doi.org/10.18653/v1/P19-1646

[3] JunjieZhang,QiWu,ChunhuaShen,JianZhang,JianfengLu,andAntonvanden Hengel. 2018. Goal-Oriented Visual Question Generation via Intermediate Re- wards. In Proceedings of the European Conference on Computer Vision.

[4] Ehsan Abbasnejad, Qi Wu, Iman Abbasnejad, Javen Shi, and Anton van den Hengel. 2018. An Active Information Seeking Model for Goal-oriented Vision- and-Language Tasks. CoRR abs/1812.06398 (2018). arXiv:1812.06398 http://arxiv.org/abs/1812.06398

[5] EhsanAbbasnejad,QiWu,JavenShi,andAntonvandenHengel.2018.What's to Know? Uncertainty as a Guide to Asking Goal-Oriented Questions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 4150 – 4159.

[6] Chaorui Deng, Qi Wu, Qingyao Wu, Fuyuan Hu, Fan Lyu, and Mingkui Tan. 2018. Visual Grounding via Accumulated Attention. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 7746 – 7755.

[7] Tianhao Yang, Zheng-Jun Zha, and Hanwang Zhang. 2019. Making History Matter: History-Advantage Sequence Training for Visual Dialog. In Proceedings of the IEEE International Conference on Computer Vision. 2561 – 2569.

[8] BohanZhuang,QiWu,ChunhuaShen,IanD.Reid,andAntonvandenHengel. 2018. Parallel Attention: A Unified Framework for Visual Object Discovery Through Dialogs and Queries. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 4252 – 4261.

作者简介

本文作者包括徐子彭、冯方向、王小捷、杨玉树、江会星、王仲远等等，他们来自北京邮电大学人工智能学院智能科学与技术中心与美团搜索与 NLP 中心团队。

王小捷

在北京航空航天大学获得博士学位，日本奈良先端科学技术大学院大学访问学者。现为北京邮电大学人工智能学院教授，博士生导师，智能科学与技术中心主任，教育部信息网络工程研究中心副主任，北京邮电大学人工智能学科和专业负责人，中国人工智能学会自然语言理解专委会主任、教育工作委员会副主任。主要研究方向为自然语言处理与多模态计算，主持和参与国家级科研项目二十余项，发表学术论文200余篇，曾获中国发明协会科技发明成果一等奖。

招聘信息

美团搜索与 NLP 部，长期招聘搜索、推荐、NLP 算法工程师，坐标北京 / 上海。欢迎感兴趣的同学发送简历至：tech@meituan.com（邮件注明：搜索与 NLP 部）

ICDM 论文：探索跨会话信息感知的推荐模型

作者：叶蕊　张庆　恒亮

会话推荐（Session-based Recommendation）是推荐领域的一个子分支，美团平台增长技术部也在该领域不断地进行探索。不久前，该部门提出的跨会话信息感知的时间卷积神经网络模型（CA-TCN）被国际会议 ICDM NeuRec Workshop 2020 接收。本文会对论文中的 CA-TCN 模型进行介绍，希望能对从事相关工作的同学有所帮助或者启发。

ICDM 的全称 International Conference on Data Mining，是由 IEEE 举办的世界顶级数据挖掘研究会议，该会议涵盖了统计、机器学习、模式识别、数据库、数据仓库、数据可视化、基于知识的系统和高性能计算等数据挖掘相关领域。其中 ICDM NeuRec Workshop 旨在从应用和理论角度系统地讨论推荐系统的浅层和深层神经算法的最新进展，该 Workshop 征集了有关开发和应用神经算法和理论以构建智能推荐系统的最新且重要的贡献。

背景

在大数据时代，推荐系统作为系统中的基础架构，开始扮演着越来越重要的角色，推荐系统可以为用户挑选出自己感兴趣的商品或者内容，从而来减少因信息爆炸带来的一些影响。目前，业界提出的很多推荐模型取得了巨大的成功，但是大部分推荐方法常常是需要根据明确的用户画像信息进行推荐，然而在一些特定的领域，用户画像的信息有可能无法被利用。

为了解决这个问题，会话推荐（Session-based Recommendation）任务被提了出来，会话推荐任务是根据用户在当前会话的行为序列去预测用户的下一个行为，而不需要依赖任何的用户画像信息 [1]。目前，会话推荐任务已广泛应用于多个领域，例如下一个网页推荐、下一个 POI 推荐、下一个商品推荐等等。为了覆盖多个领域，所以“会话”的概念不仅限于交易，而是指一次或者一定时期内的消费或者访问的元素集合。

每一个会话（Session）都是一个 item 的转移序列，所以会话推荐任务可以很自然地被视为序列推荐任务，基于循环神经网络（RNN）的会话推荐模型 [2] 是应用的主流模型。但是基于 RNN 模型只对 item 之间的连续单向转移关系进行建模，而忽略了会话中其他 item 之间的转移关系。随着图神经网络的热点爆发，基于图结构的会话推荐模型如 SR-GNN[3]、GC-SAN[4] 被提出来，希望能够克服该点不足。基于图结构的会话推荐模型将会话的 item 转移序列构建成一个图结构，然后应用图神经网络模型来探索多个 item 之间复杂的转移关系。目前，基于图结构的会话推荐模型已经成为了 State-of-the-art 的解决方法，但它们仍然具有一定的限制，观察如下：

- 观察 1：几乎所有现存的会话推荐方法都仅仅关注于会话的内部信息，而忽略了跨会话的外部信息（跨会话的相互影响），跨会话信息往往包含着非常有价值的补充信息，有利于更准确地推断当前会话的用户偏好。如下图所示，以 Session 3 中的 Item_3 Airpods 为例，现存的方法仅仅关注当前会话 Session3 中的 Item_9 对 Item_3 的影响而忽略了其他会话的影响。对于 Session1 而言，用户可能具有买耳机的意图而进行同品类比较，所以 item_2 和 item_4 会对 item_3 产生一个品类的影响；对于 Session 2 而言，用户可能比较喜欢 Apple 品牌，所以 item_5 和 item_6 会对 item_3 产生一个品牌的影响。根据上面的观察可知，在 Item_Level 层次的跨会话影响对于更好地推断 item 的全局表示至关重要。同时，不同的会话之间也可能具有相似的用户意图和行为模式，所以对于 Session-Level 的跨会话影响对于更准确地预测用户在当前会话中的下一个动作也起着非常重要的作用。

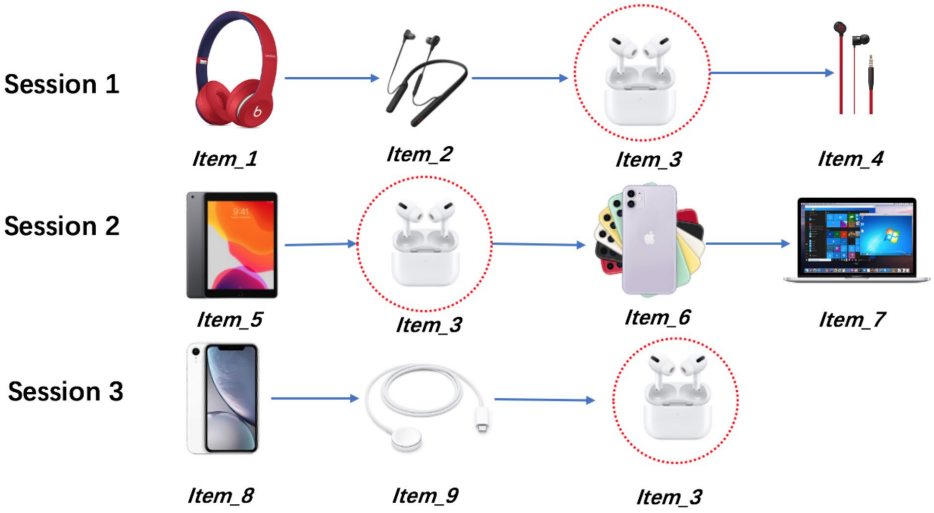


图 1　跨会话 item 的 Toy 样例

- 观察 2：基于图结构的会话推荐方法在构建图的过程中，将出现在不同时间步的相同 item 都视为一个相同节点，这样会丢失序列中的位置信息，以至于不同的序列会话构建出的 Session 图结构是完全相同的。例如两个不同的会话 Session S1:v_i－>v_j－>v_i－>v_k－>v_j－>v_k 与 Session S2: v_i－>v_j－>v_k－>v_j－>v_i－>v_k，在下图 2 中，它们对应的图结构是完全相同的，这不可避免地限制了模型获得准确会话表示的能力。此外，在会话图构造中，仅仅直接连接的两个相邻 item 之间会建立边，意味着只有在当前 item 之前最后点击的 item 才是当前 item 的一阶邻居，如图 2 所示。但出现在一个相同会话中，即使没有被连续点击的 item 之间也具有一定的联系，所以图结构对于保留序列的长期依赖性具有有限的能力。相反，对于时序卷积神经网络（TCN）[5] 模型，Causal Convolution 使当前 item 的接受域中的 items 都可以直接作为一阶邻居进行卷积，并且具有的 Dilated Convolution 使得较远的 items 也可以直接作为一阶邻居对其产生影响。

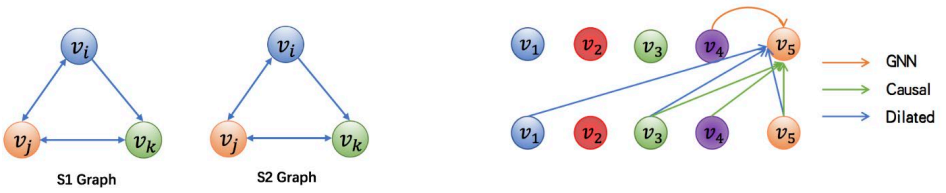


图 2　图结构对于序列数据的建模示意图

相关工作介绍

现有的会话推荐方法大致可以分为两类，分别是基于协同过滤方法和基于深度学习方法：

- **基于协同过滤方法**：协同过滤方法是在推荐系统中被广泛使用的通用方法，协同过滤方法主要可以分为两大类：基于 KNN 查找方法和基于相似度建模方法。 基于 KNN 查找方法是通过查找 Top-K 个相关的 users 或 items 来实现推荐，基于 KNN 查找方法可以通过查找与当前会话中最后一个 item 最相似的 item 来实现基于会话的推荐。 最近，KNN-RNN[6] 探索将 RNN 模型与 KNN 模型相结合，通过 RNN 模型来提取会话序列信息，然后查找在与当前 Session 相似的 Session 中出现的 item 来实现推荐。对于基于相似度建模的方法，CSRM[7] 通过记忆网络将距离当前会话时间最近的 m 个会话中包含的相关信息进行建模，从而来获得更为准确的会话表示，以提高会话推荐的性能。
- **基于深度学习方法**：深度学习方法凭借其强大的特征学习能力在多个领域获得了令人满意的成果，对于会话推荐任务而言，循环神经网络 RNN 是一个直观的选择，可以利用其提取序列特征的优势来捕获会话内复杂的依赖关系。GRU4Rec[2] 利用门控循环单元（GRU）作为 RNN 的一种特殊形式来学习 item 之间的长期依赖性，以预测会话中的下一个动作。之后的一些工作，是通过在基于 RNN 模型的基础上增加注意力机制和记忆机制等对模型进行了改进和扩展，其中 NARM[8] 探索了一种具有注意力机制的层次编码器，可以对当前会话中用户的序列行为和主要意图进行建模。 最近，随着图神经网络模型的飞速发展，出现了依赖图结构的会话推荐模型，SR-GNN 首先提出将每个

会话映射为一个图结构，并利用图神经网络模型 GNN 来建模 item 之间的复杂转移关系。 之后，GC-SAN 通过加入 Self-Attention 机制进一步扩展了 SR-GNN 模型，从而成为了 State-of-the-art 的解决方法。

CA-TCN 模型与现有方法都存在着明显的差异。一方面，CA-TCN 探索 Item-Level 和 Session-Level 的跨会话影响，以提高推荐性能，与其他的协同过滤方法的区别有两个：1. CA-TCN 同时考虑了跨会话信息对 item 和 Session 不同层次的影响，而 CSRM 仅仅考虑了 Session 层次。 2. CA-TCN 构建了跨会话的全局 Cross-Session item 图和 Session-Context 图，通过 GNN 来探索复杂的跨会话影响。另一方面，与基于 RNN 和基于 GNN 的模型相比，CA-TCN 模型克服了 RNN 模型无法并行以及图结构缺失位置和长期依赖信息的不足。

跨会话感知的时间卷积神经网络模型（CA-TCN）

1. 模型整体框架

网络的整体框架如下图 3 所示。给定会话序列数据，首先，我们构造一个 Cross-Session Item-Graph 来链接出现在不同会话中且有关系的 items，然后经过图神经网络输出包含全局信息的 item 向量。 将得到的 item 向量输入到 TCN 模型中输出蕴含会话序列信息的 item 表示，根据 Item-Level Attention 机制来整合 item 的表示进而获得 Session 表示。 此后，根据 Session 表示之间的相似度构建 Session-Context Graph 图以对 Session 层次的跨会话关系进行建模。 最后，根据 Session 的表示以及 item 的表示进行预测。

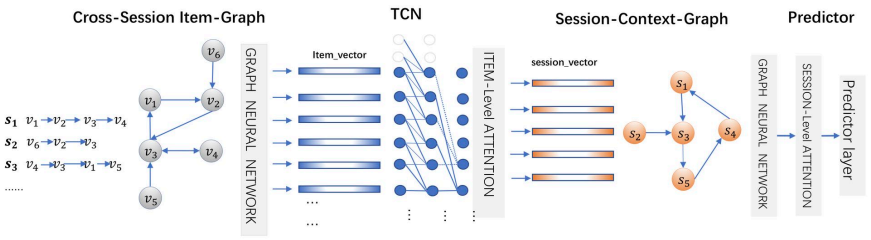


图 3　CA-TCN 模型总体框架图

2. 跨会话 Item 图（Cross-Session Item-Graph）

在第一阶段，我们构建 Cross-Session Item-Graph 有向图 G_ item，其中图中的每个节点代表一个 item，（v_s_i, v_s_i+1）作为一条边，代表在会话 s 中用户在 v_s_i 之后点击了 v_s_i+1。与现有方法相比，跨会话的 G_item 图能够在所有的会话中出现的 item 之间建立链接，因此 G_item 不仅可以获取会话的内部信息，同时可以得到非当前会话的外部信息。G_item 的图的核心在于将所有的 item 放在了一起通盘考虑，然后用各个会话中的点击行为给 item 之间建立链接，不同会话的点击信息汇总在一起使得 item 之间的关系连接更加丰富。

为了充分利用 G_item 图结构中的信息，CA-TCN 将 item 的点击顺序和共现次数考虑在内。对于点击顺序，建立带有方向的邻接矩阵 A_in 和 A_out 来建模输入和输出方向。在邻接矩阵的基础上，根据 item 之间的共现次数为不同的边设置不同的权重，得到权重矩阵 Weight_in 和 Weight_out。通过分配不同的权重，具有更多共现次数的 item 将发挥更大的作用，反之亦然，从而避免了噪音影响。

接下来，我们开发 GNN 模型来捕捉复杂的跨会话信息在 item__level 的影响，GNN 将每一个 item 映射为一个 d 维的 embedding v ∈ R^d，得到包含跨会话信息的全局 item 向量（item_vector）。

$$\mathbf{V}^{l+1} = \sigma(D^{-1}(Weight_{in}\mathbf{V}^{l} + Weight_{out}\mathbf{V}^{l} + \mathbf{V}^{l})W^{l})$$

3. 时间卷积神经网络模型（TCN Model）

在第二阶段，我们采用时间卷积神经网络 TCN 来对会话序列进行建模，获取会话 s 的全局和局部表示。每一个会话 s 由多个 item 组成，输入会话 s 包含的 item 全局向量化表示（item_vector）到时间卷积神经网络（TCN）模型中。对于会话中的每一个 item 进行因果和膨胀卷积的计算，进行会话序列信息的抽取。

$$F(\mathbf{v}_{s,i}) = (\mathbf{v}_{s,i} *_d f) = \sum_{j=0}^{k-1} f(j) \cdot \mathbf{v}_{s,i-d \cdot j}$$

采用会话中最后一个 item 的 TCN 输出作为会话 s 的局部（local）信息，以正确获取用户的当前兴趣：

$$\mathbf{s}^{local} = F(\mathbf{v}_{s,n})$$

此外，采用会话 s 包含的 items 的表示以加权求和的方式得到会话的全局（global）表示（session_vector），捕捉用户的全局信息。其中为了区分不同的 item 对于会话的影响程度不同，采用 item 层次注意力机制，使得会话表示更加专注于重要程度高的 items。

$$\alpha_{s,i} = softmax(\sigma(\mathbf{W}_1 F(\mathbf{v}_{s,n}) + \mathbf{W}_2 F(\mathbf{v}_{s,i})))$$

$$\mathbf{s}^{global} = \sum_{i=1}^{n} \alpha_{s,i} \cdot F(\mathbf{v}_{s,i})$$

4. 会话上下文感知图（Session-Context-Graph）

会话的 local 表示和 global 表示只专注于当前的会话，而忽略了会话间的影响。为了克服该不足，我们构建一个上下文感知的会话图结构（Session-Context-Graph）来考虑不同会话之间复杂的关系。在会话图中，每一个节点代表一个会话 s，边的链接代表两个会话之间具有相似性。我们需要考虑的一个重要问题是如何决定一条边是否存在。对于每一对会话，我们计算其二者表示的相似度，然后采用根据相似度值的 KNN-Graph[9] 模型来决定一个会话节点的邻居。在构建会话图结构之后，我们采用会话层的注意力机制以及图神经网络模型 [10] 来整合会话邻居节点对其自身的影响，同时会话层的注意力将会话之间的相似度也考虑在内，最终得到基于会话上下文敏感的会话表示。

$$\alpha_{i,j} = \frac{exp(LeakyReLU(\mathbf{a}^T[\mathbf{W}\mathbf{s}_i^{global}||\mathbf{W}\mathbf{s}_j^{global}||sim_{ij}]))}{\sum_{k \in N_i} exp(LeakyReLU(\mathbf{a}^T[\mathbf{W}\mathbf{s}_i^{global}||\mathbf{W}\mathbf{s}_k^{global}||sim_{ik}]))}$$

$$\mathbf{s}_i^{cross} = \sum_{j \in N_i} \alpha_{i,j} \mathbf{s}_j^{global}$$

5. 点击预测

为了更好地预测用户的下一个行为，我们采用融合函数将会话的局部表示，全局表示以及基于跨会话信息的表示进行融合，得到最终的会话表示：

$$f = \sigma(\mathbf{W}_l \mathbf{s}^{local} + \mathbf{W}_g \mathbf{s}^{global} + \mathbf{W}_s \mathbf{s}^{cross})$$

$$\mathbf{s}_{final} = f \cdot [\mathbf{s}^{local}||\mathbf{s}^{global}] + (1 - f) \cdot \mathbf{s}^{cross}$$

最后，我们根据 item 和 session 的表示去预测每一个候选 item 成为用户下一个点击的概率，根据概率进行逆序排序，筛选出概率值排在前预设位数对应的商品，作为用户偏好商品并进行推荐。

$$\hat{y}_i = softmax(\mathbf{s}_{final}^T \mathbf{v}_i)$$

实验评估

为了评估所提出的 CA-TCN 的性能，我们使用了两个广泛应用的基准数据集，即 Yoochoose 和 Diginetica，模型性能评估结果如下表所示，CA-TCN 优于目前的基于 RNN 以及图结构的 State-of-the-art 解决方法。

Dataset	Yoochoose 1/64		Yoochoose 1/4		Diginetica	
Metrics(%)	$P@20$	$MRR@20$	$P@20$	$MRR@20$	$P@20$	$MRR@20$
POP*	6.71	1.65	1.33	0.3	0.89	0.20
FPMC*	45.62	15.01	-	-	26.53	6.95
GRU4Rec*	60.04	22.89	59.53	22.60	29.45	8.33
RNN-KNN	68.66	28.64	68.02	28.38	44.61	15.35
NARM*	68.32	28.63	69.73	29.23	49.70	16.17
STAMP*	68.74	29.67	70.44	30.00	45.64	14.32
SR-GNN*	70.57	30.94	71.36	31.89	50.73	17.59
SR-GNN**	69.57	29.67	70.41	29.80	49.97	16.53
CSRM**	69.95	29.79	70.31	29.74	50.89	16.59
GC-SAN	70.47	30.18	70.83	29.95	50.91	17.18
CA-TCN	**71.98**	**31.01**	**72.25**	**32.23**	**53.66**	**18.19**

表 1　会话推荐任务的性能比较

此外，我们进行消融实验以评估 CA-TCN 中每个组成部分的影响，组成部分包括 TCN 模型，Cross-Session Item-Graph 和 Session-Context graph。下图的实验结果证明了 CA-TCN 通过利用 TCN 模型和跨会话信息在会话推荐任务上都实现了性能的逐步提升。

- **CA-TCN(ca.exl)**：是 CA-TCN 的变体，它仅包含时间卷积神经网络、Cross-Session Item-Graph 和 Session-Context graph 的跨会话信息不包含在内。
- **CA-TCN(sc.exl)**：是 CA-TCN 的变体，其中包含了 Cross-Session Item-Graph 的 item-level 的跨会话信息，但不包括 session-level 的 Cross-Session Item-Graph。

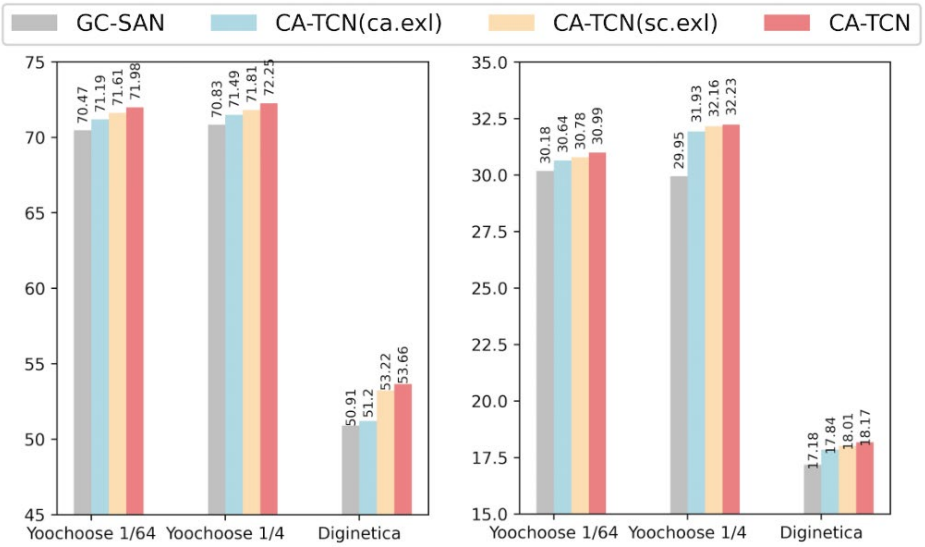


图 4　CA-TCN 模型不同组成部分的实验结果

未来工作

目前该论文已经申请了专利，后续我们将在美团多个业务线的会话推荐和序列推荐任务上进行探索落地。 特别地，CA-TCN 模型在电商数据集 Yoochoose 上进行了性能验证，证明 CA-TCN 模型适用于具有商品属性的电商场景，未来在“团好货”和“美团优选”等具有商品属性的业务线中都可以尝试应用。

参考文献

[1] S. Wang, L. Cao, and Y. Wang, “A survey on session-based recommender systems,” arXiv preprint arXiv:1902.04864, 2019.

[2] B. Hidasi, A. Karatzoglou, L. Baltrunas, and D. Tikk, “Session-based recommendations with recurrent neural networks,” arXiv preprintarXiv:1511.06939, 2015.

[3] S. Wu, Y. Tang, Y. Zhu, L. Wang, X. Xie, and T. Tan, “Session-based recommendation with graph neural networks,” in Proceedings of the AAAI Conference on Artificial Intelligence, vol. 33, 2019, pp. 346 - 353.

[4] C. Xu, P. Zhao, Y. Liu, V. S. Sheng, J. Xu, F. Zhuang, J. Fang, and X. Zhou, “Graph contextualized self-attention network for session-based recommendation.”

inIJCAI, 2019, pp. 3940 – 3946.
[5] S. Bai, J. Z. Kolter, and V. Koltun, “An empirical evaluation of generic convolutional and recurrent networks for sequence modeling,” arXivpreprint arXiv:1803.01271, 2018.
[6] D. Jannach and M. Ludewig, “When recurrent neural networks meet the neighborhood for session-based recommendation,” inRecSys ’ 17, 2017.
[7] M. Wang, P. Ren, L. Mei, Z. Chen, J. Ma, and M. de Rijke, “A collaborative session-based recommendation approach with parallel memory modules,” in Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval, 2019, pp. 345 – 354.
[8] J. Li, P. Ren, Z. Chen, Z. Ren, T. Lian, and J. Ma, “Neural attentive session-based recommendation,” in Proceedings of the 2017 ACM on Conference on Information and Knowledge Management, 2017, pp. 1419 – 1428.
[9] W. Dong, C. Moses, and K. Li, “Efficient k-nearest neighbor graph construction for generic similarity measures,” in Proceedings of the 20thinternational conference on World wide web, 2011, pp. 577 – 586.
[10] P. Veli ˇ ckovi´c, G. Cucurull, A. Casanova, A. Romero, P. Lio, and Y. Bengio, “Graph attention networks,” arXiv preprint arXiv:1710.10903, 2017.

作者信息

本文作者叶蕊、张庆、恒亮，均来自美团平台增长技术部。

招聘信息

美团用户增长技术部，美团用户增长核心团队，长期招聘搜索、推荐、NLP 算法及后台工程师，坐标北京。感兴趣的同学可投递简历至：luohengliang@meituan.com（邮件主题请注明：美团用户增长技术部）。

自然场景人脸检测技术实践

作者：振华　欢欢　晓林

一、背景

人脸检测技术是通过人工智能分析的方法自动返回图片中的人脸坐标位置和尺寸大小，是人脸智能分析应用的核心组成部分，具有广泛的学术研究价值和业务应用价值，比如人脸识别、人脸属性分析（年龄估计、性别识别、颜值打分和表情识别）、人脸 Avatar、智能视频监控、人脸图像过滤、智能图像裁切、人脸 AR 游戏等等。因拍摄的场景不同，自然场景环境复杂多变，光照因素也不可控，人脸本身多姿态以及群体间的相互遮挡给检测任务带来了很大的挑战（如图 1 所示）。在过去 20 年里，该任务一直是学术界和产业界共同关注的热点。

自然场景人脸检测在美团业务中也有着广泛的应用需求，为了应对自然场景应用本身的技术挑战，同时满足业务的性能需求，美团视觉智能中心（Vision Intelligence Center，VIC）从底层算法模型和系统架构两个方面进行了改进，开发了高精度人脸检测模型 VICFace。而且 VICFace 在国际知名的公开测评集 WIDER FACE 上达到了行业主流水平。

图 1　自然场景人脸检测样本示例

二、技术发展现状

跟深度学习不同，传统方法解决自然场景人脸检测会从特征表示和分类器学习两个方面进行设计。最有代表性的工作是 Viola-Jones 算法 [2]，它利用手工设计的 Haar-like 特征和 Adaboost 算法来完成模型训练。传统方法在 CPU 上检测速度快，结果可解释性强，在相对可控的环境下可以达到较好的性能。但是，当训练数据规模成指数增长时，传统方法的性能提升相对有限，在一些复杂场景下，甚至无法满足应用需求。

随着计算机算力的提升和训练数据的增长，基于深度学习的方法在人脸检测任务上取得了突破性进展，在检测性能上相对于传统方法具有压倒性优势。基于深度学习的人脸检测算法从算法结构上可以大致分为三类：

1）基于级联的人脸检测算法。

2）两阶段人脸检测算法。

3）单阶段人脸检测算法。

其中，第一类基于级联的人脸检测方法（如 Cascade CNN[3]、MTCNN[4]）运行速度较快、检测性能适中，适用于算力有限、背景简单且人脸数量较少的场景。第二类两阶段人脸检测方法一般基于 Faster-RCNN[6] 框架，在第一阶段生成候选区域，然后在第二阶段对候选区域进行分类和回归，其检测准确率较高，缺点是检测速度较慢，代表方法有 Face R-CNN[9] 、ScaleFace[10]、FDNet[11]。最后一类单阶段的人脸检测方法主要基于 Anchor 的分类和回归，通常会在经典框架（如 SSD[12]、RetinaNet[13]）的基础上进行优化，其检测速度较两阶段法快，检测性能较级联法优，是一种检测性能和速度平衡的算法，也是当前人脸检测算法优化的主流方向。

三、优化思路和业务应用

在自然场景应用中，为了同时满足精度需求以及达到实用的目标，美团视觉智能中心（Vision Intelligence Center，VIC）采用了主流的 Anchor-Based 单阶段人脸检测

方案，同时在数据增强和采样策略、模型结构设计和损失函数等三方面分别进行了优化，开发了高精度人脸检测模型 VICFace，以下是相关技术细节的介绍。

1. 数据增强和采样策略

单阶段通用目标检测算法对数据增强方式比较敏感，如经典的 SSD 算法在 VOC2007[50] 数据集上通过数据增强性能指标 mAP 提升 6.7。经典单阶段人脸检测算法 S3FD[17] 也设计了样本增强策略，使用了图片随机裁切，图片固定宽高比缩放，图像色彩扰动和水平翻转等。

百度在 ECCV2018 发表的 PyramidBox[18] 提出了 Data-Anchor 采样方法，将图像中一个随机选择的人脸进行尺度变换变成一个更小 Anchor 附近尺寸的人脸，同时训练图像的尺寸也进行同步变换。这样做的好处是通过将较大的人脸生成较小的人脸，提高了小尺度上样本的多样性，在 WIDER FACE[1] 数据集 Easy、Medium、Hard 集合上分别提升 0.4（94.3->94.7），0.4（93.3->93.7），0.6（86.1->86.7）。ISRN[19] 将 SSD 的样本增强方式和 Data-Anchor 采样方法结合，模型检测性能进一步提高。

而 VICFace 在 ISRN 样本增强方式的基础上对语义模糊的超小人脸做了过滤。而 mixup[22] 在图像分类和目标检测中已经被验证有效，现在用于人脸检测，有效地防止了模型过拟合问题。考虑到业务数据中人脸存在多姿态、遮挡和模糊的样本，且这些样本在训练集中占比小，检测难度大，因此在模型训练时动态的给这些难样本赋予更高的权重从而有可能提升这些样本的召回率。

2. 模型结构设计

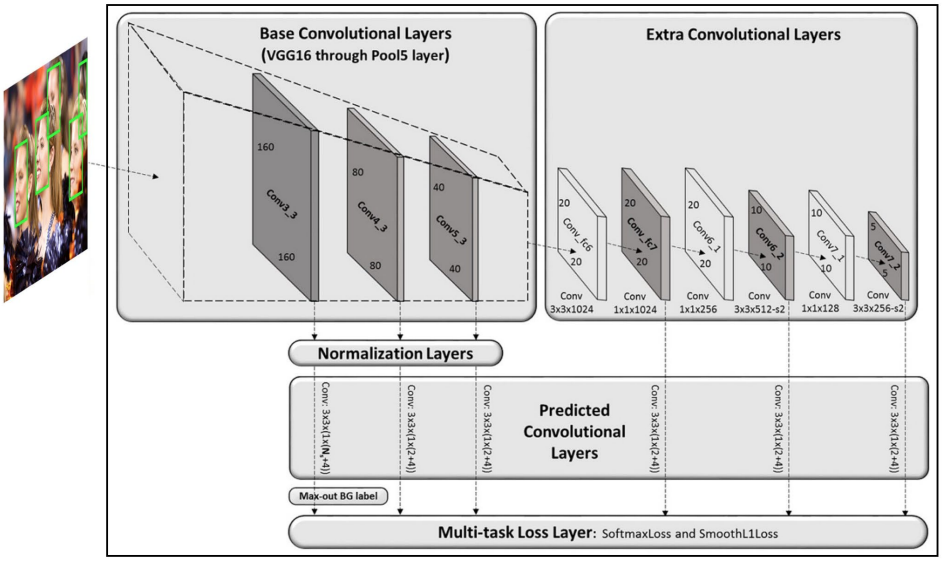
人脸检测模型结构设计主要包括检测框架、主干网络、预测模块、Anchor 设置与正负样本划分等四个部分，是单阶段人脸检测方法优化的核心。

- 检测框架

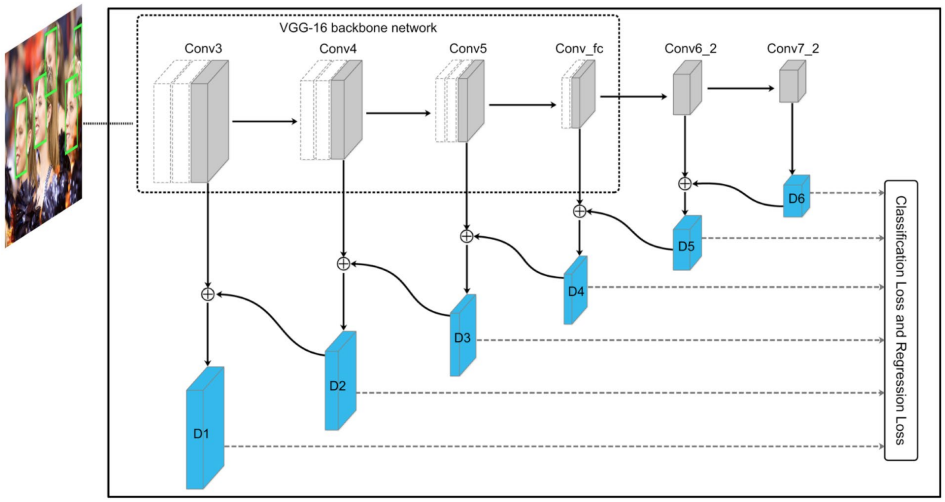
近年来单阶段人脸检测框架取得了重要的发展，代表性的结构有 S3FD[17] 中使用的

SSD，SFDet[25] 中使用的 RetinaNet，SRN[23] 中使用的两步结构（后简称 SRN）以及 DSFD[24] 中使用的双重结构（后简称 DSFD），如下图 2 所示。其中，SRN 是一种单阶段两步人脸检测方法，利用第一步的检测结果，在小尺度人脸上过滤易分类的负样本，改善正负样本数量的均衡性，针对大尺度的人脸采用迭代求精的方式进行人脸定位，改善大尺度人脸的定位精度，提升了人脸检测的准确率。在 WIDER FACE 上测评 SRN 取得了最好的检测效果（按标准协议用 AP 平均精度来衡量），如表 1 所示。

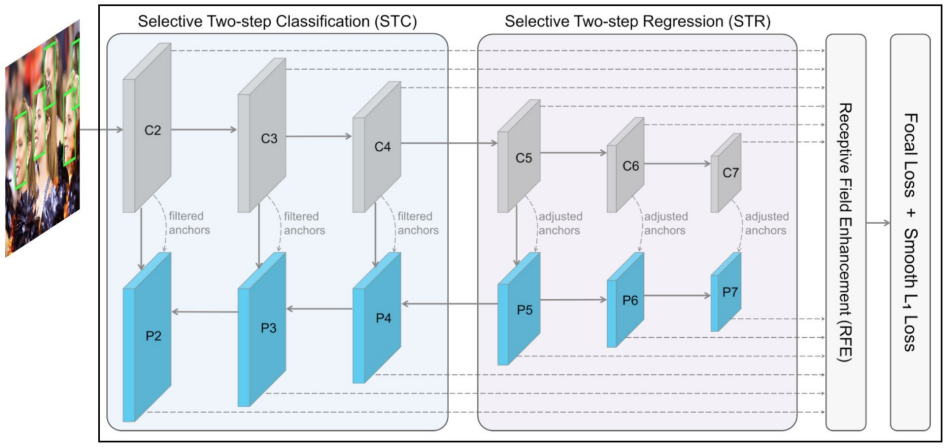
S3FD:



SFDet:


VGG-16 backbone network
Conv3
Conv4
Conv5
Conv_fc
Conv6_2
Conv7_2
D1
D2
D3
D4
D5
D6
Classification Loss and Regression Loss

SRN:


Selective Two-step Classification (STC)
Selective Two-step Regression (STR)
C2
C3
C4
C5
C6
C7
filtered anchors
filtered anchors
filtered anchors
adjusted anchors
adjusted anchors
adjusted anchors
P2
P3
P4
P5
P6
P7
Receptive Field Enhancement (RFE)
Focal Loss + Smooth L1 Loss

DSFD:

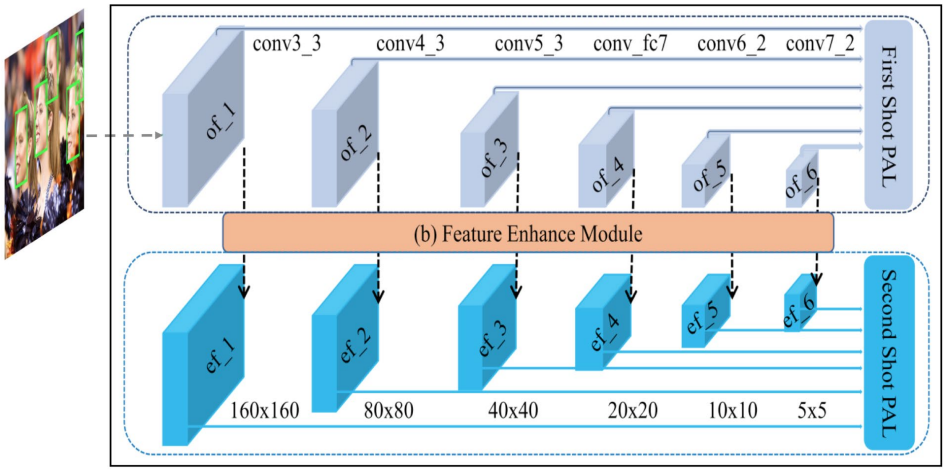


图 2　四种检测结构

检测结构	Easy	Medium	Hard
SSD	0.937	0.922	0.818
RetinaNet	0.950	0.941	0.880
DSFD	0.953	0.944	0.886
SRN	**0.964**	**0.953**	**0.902**

表 1　Backbone 为 ResNet50 时，四种检测结构在 WIDER FACE 上的评估结果

VICFace 继承了当前性能最好的 SRN 检测结构，同时为了更好的融合自底向上和自顶向下的特征，为不同特征不同通道赋予不同的权重，以 P4 为例，其计算式为：

$$P4 = Conv(W_{C4} \cdot Conv(C4) + W_{P4} \cdot Upsample(P5))$$

其中 WC4 向量的元素个数与 Conv(C4) 特征的通道数相等，WP4 与 Upsample(P5) 的通道数相等，WC4 与 WP4 是可学习的，其元素值均大于 0，且 WC4 与

WP4 对应元素之和为 1，结构如图 3 所示。

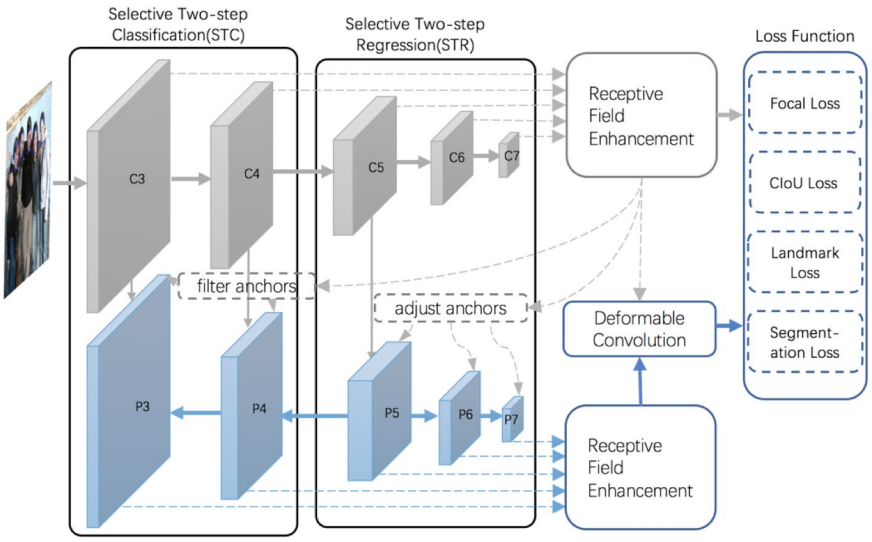


图 3　视觉智能中心 VICFace 网络整体结构图

- 主干网络

单阶段人脸检测模型的主干网络通常使用分类任务中的经典结构（如 VGG[26]、ResNet[27] 等）。其中，主干网络在 ImageNet 数据集上分类任务表现越好，其在 WIDER FACE 上的人脸检测性能也越高，如表 2 所示。为了保证检测网络得到更高的召回，在性能测评时 VICFace 主干网络使用了在 ImageNet 上性能较优的 ResNet152 网络（其在 ImageNet 上 Top1 分类准确率为 80.26），并且在实现时将 Kernel 为 7x7，Stride 为 2 的卷积模块调整为为 3 个 3x3 的卷积模块，其中第一个模块的 Stride 为 2，其它的为 1；将 Kernel 为 1x1，Stride 为 2 的下采样模块替换为 Stride 为 2 的 Avgpool 模块。

主干网络	ImageNet	Easy	Medium	Hard
VGG16	0.7323	0.930	0.914	0.846
ResNet50	0.7736	0.950	0.941	0.880
ResNet101	**0.7834**	**0.958**	**0.951**	**0.897**

表 2　不同主干网络在 ImageNet 的性能对比和其在 RetinaNet 框架下的检测精度

- 预测模块

利用上下文信息可以进一步提高模型的检测性能。SSH[36] 是将上下文信息用于单阶段人脸检测模型的早期方案，PyramidBox、SRN、DSFD 等也设计了不同上下文模块。如图 4 所示，SRN 上下文模块使用 1xk，kx1 的卷积层提供多种矩形感受野，多种不同形状的感受野助于检测极端姿势的人脸；DSFD 使用多个带孔洞的卷积，极大的提升了感受野的范围。

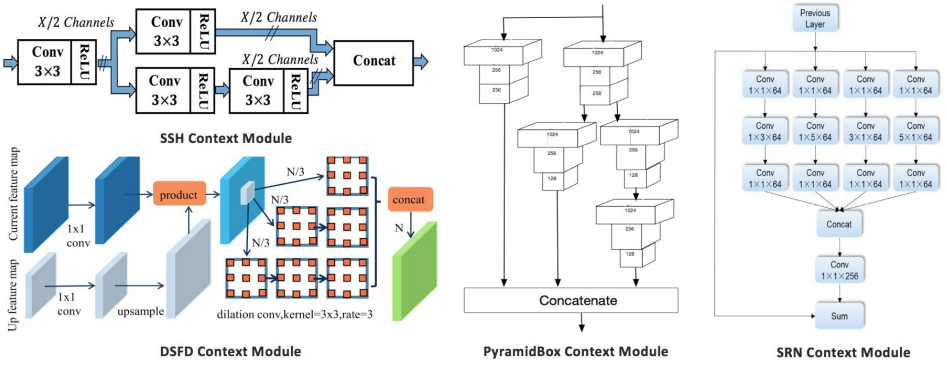


图 4　不同网络结构中的 Context Module

在 VICFace 中，将带孔洞的卷积模块和 1xk，kx1 的卷积模块联合作为 Context Module，既提升了感受野的范围也有助于检测极端姿势的人脸，同时使用 Maxout 模块提升召回率，降低误检率。它还利用 Cn 层特征预测的人脸位置，校准 Pn 层特征对应的区域，如图 5 所示。Cn 层预测的人脸位置相对特征位置的偏移作为可变卷积的 Offset 输入，Pn 层特征作为可变卷积的 Data 输入，经过可变卷积后特征对应

的区域与人脸区域对应更好，相对更具有表示能力，可以提升人脸检测模型的性能。

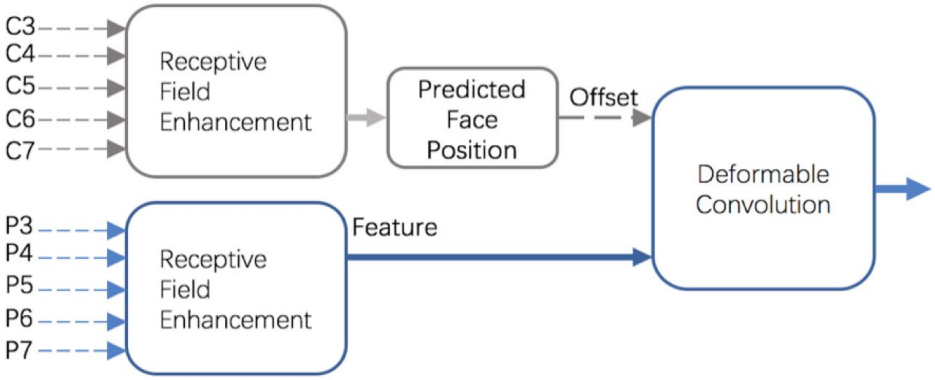


图 5　自研检测模型结构中的预测模块

- Anchor 设置与正负样本划分

基于 Anchor 的单阶段人脸检方法通过 Anchor 的合理设置可以有效的控制正负样本比例和缓解不同尺度人脸定位损失差异大的问题。现有主流人脸检测方法中 Anchor 的大小设置主要有以下三种（S 代表 Stride）：

$$\{4S\},\{2S,\ 2\sqrt{2}S\},\{4S,\ 4\sqrt[3]{2}S,\ 4\sqrt[3]{2^2}S\}$$

根据数据集中人脸的特点，Anchor 的宽高也可以进一步丰富，如 {1}，{0.8}，{1，0.67}。

在自研方案中，在 C3、P3 层，Anchor 的大小为 2S 和 4S，其它层 Anchor 大小为 4S（S 代表对应层的 Stride），这样的 Anchor 设置方式在保证人脸召回率的同时，减少了负样本的数量，在一定程度上缓解了正负样本不均衡现象。根据人脸样本宽高比的统计信息，将 Anchor 的宽高比设置为 0.8，同时将 Cn 层 IoU 大于 0.7 的样本划分为正样本，小于 0.3 的划分为负样本，Pn 层 IoU 大于 0.5 的样本划分为正样本，小于 0.4 的划分为负样本。

3. 损失函数

人脸检测的优化目标不仅需要区分正负样本（是否是人脸），还需要定位出人脸位置和尺寸。S3FD 中区分正负样本使用交叉熵损失函数，定位人脸位置和尺寸使用 Smooth L1 Loss，同时使用困难负样本挖掘解决正负样本数量不均衡的问题。另一种缓解正负样本不均衡带来的性能损失更直接的方式是 Lin 等人提出 Focal Loss[13]。UnitBox[41] 提出 IoU Loss 可以缓解不同尺度人脸的定位损失差异大导致的性能损失。AInnoFace[40] 同时使用 Focal Loss 和 IoU Loss 提升了人脸检测模型的性能。引入其它相关辅助任务也可以提升人脸检测算法的性能，RetinaFace[42] 引入关键点定位任务，提升人脸检测算法的定位精度；DFS[43] 引入人脸分割任务，提升了特征的表示能力。

综合前述方法的优点，VICFace 充分利用人脸检测及相关任务的互补信息，使用多任务方式训练人脸检测模型。在人脸分类中使用 Focal Loss 来缓解样本不均衡问题，同时使用人脸关键点定位和人脸分割来辅助分类目标的训练，从而提升整体的分类准确率。在人脸定位中使用 Complete IoU Loss[47]，以目标与预测框的交并比作为损失函数，缓解不同尺度人脸损失的差异较大的问题，同时兼顾目标和预测框的中心点距离和宽高比差异，从而可以达到更好整体检测性能。

4. 优化结果和业务应用

在集群平台的支持下，美团视觉智能中心的自然场景人脸检测基础模型 VICFace 与现有主流方案进行了性能对比，在国际公开人脸检测测评集 WIDER FACE 的三个验证集 Easy、Medium、Hard 中均达到领先水平（AP 为平均精度，数值越高越好），如图 6 和表 3 所示。

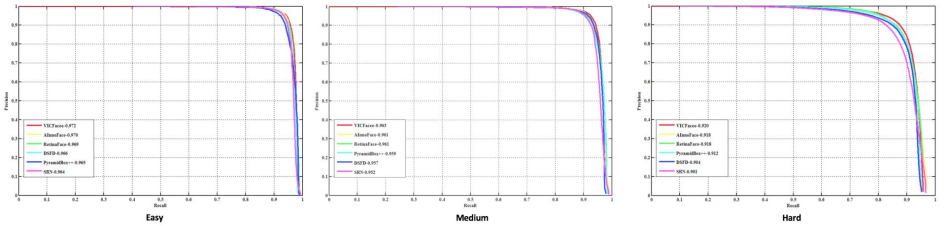


图 6　VICFace 以及当前主流人脸检测方法在 WIDER FACE 上的测评结果

主流方法*	Easy	Medium	Hard
SRN	0.964	0.953	0.902
DSFD	0.966	0.957	0.904
PyramidBox++	0.965	0.959	0.912
AInnoFace	0.969	0.959	0.912
RetinaFace	0.969	0.961	0.918
VICFace	**0.972**	**0.963**	**0.920**

表 3　VICFace 以及当前主流人脸检测方法在 WIDER FACE 上的测评结果

注：SRN 是中科院在 AAAI2019 提出的新方法，DSFD 是腾讯优图在 CVPR2019 提出的新方法，PyramidBox++ 是百度在 2019 年提出的新方法，AInnoFace 是创新奇智在 2019 提出的新方法，RetinaFace 是 ICCV2019 Wider Challenge 亚军。

在业务应用中，自然场景人脸检测服务目前已接入美团多个业务线，满足了业务在 UGC 图像智能过滤和广告 POI 图像展示等应用的性能需求，前者保护用户隐私，预防侵犯用户肖像权，后者可以有效的预防图像中人脸局部被裁切的现象，从而提升了用户体验。此外，VICFace 还为其它人脸智能分析应用提供了核心基础模型，如自动检测后厨工作人员的着装合规性（是否穿戴帽子和口罩），为食品安全增加了一道保障。

在未来的工作中，为了给用户提供更好的体验，同时满足高并发的需求，在模型结构设计和模型推理效率方面将会做进一步探索和优化。此外，在算法设计方面，基于Anchor-Free 的单阶段目标检测方法近年来在通用目标检测领域表现出较高的潜力，也是视觉智能中心未来会关注的重要方向。

参考文献

[1] Yang S, Luo P, Loy C C, et al. Wider face: A face detection benchmark[C]// Proceedings of the IEEE conference on computer vision and pattern recognition. 2016: 5525-5533.

[2] Viola P, Jones M J. Robust real-time face detection[J]. International journal of computer vision, 2004, 57(2): 137-154.

[3] Li H, Lin Z, Shen X, et al. A convolutional neural network cascade for face detection[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2015: 5325-5334.

[4] Zhang K, Zhang Z, Li Z, et al. Joint face detection and alignment using multitask cascaded convolutional networks[J]. IEEE Signal Processing Letters, 2016, 23(10): 1499-1503.

[5] Hao Z, Liu Y, Qin H, et al. Scale-aware face detection[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2017: 6186-6195.

[6] Ren S, He K, Girshick R, et al. Faster r-cnn: Towards real-time object detection with region proposal networks[C]//Advances in neural information processing systems. 2015: 91-99.

[7] Lin T Y, Dollár P, Girshick R, et al. Feature pyramid networks for object detection[C]. Proceedings of the IEEE conference on computer vision and pattern recognition. 2017: 2117-2125.

[8] Jiang H, Learned-Miller E. Face detection with the faster R-CNN[C]//2017 12th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2017). IEEE, 2017: 650-657.

[9] Wang H, Li Zhif, et al. Face R-CNN. arXiv preprint arXiv: 1706.01061, 2017.

[10] Yang S, Xiong Y, Loy C C, et al. Face detection through scale-friendly deep convolutional networks[J]. arXiv preprint arXiv:1706.02863, 2017.

[11] Zhang C, Xu X, Tu D. Face detection using improved faster rcnn[J]. arXiv preprint arXiv:1802.02142, 2018.

[12] Liu W, Anguelov D, Erhan D, et al. Ssd: Single shot multibox detector[C]//European conference on computer vision. Springer, Cham, 2016: 21-37.

[13] Lin T Y, Goyal P, Girshick R, et al. Focal loss for dense object detection[C]// Proceedings of the IEEE international conference on computer vision. 2017: 2980-2988.

[14] Huang L, Yang Y, Deng Y, et al. Densebox: Unifying landmark localization with end to end object detection[J]. arXiv preprint arXiv:1509.04874, 2015.
[15] Liu W, Liao S, Ren W, et al. High-level Semantic Feature Detection: A New Perspective for Pedestrian Detection[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2019: 5187-5196.
[16] Zhang Z, He T, Zhang H, et al. Bag of freebies for training object detection neural networks[J]. arXiv preprint arXiv:1902.04103, 2019.
[17] Zhang S, Zhu X, Lei Z, et al. S3fd: Single shot scale-invariant face detector[C]// Proceedings of the IEEE International Conference on Computer Vision. 2017: 192-201.
[18] Tang X, Du D K, He Z, et al. Pyramidbox: A context-assisted single shot face detector[C]//Proceedings of the European Conference on Computer Vision (ECCV). 2018: 797-813.
[19] Zhang S, Zhu R, Wang X, et al. Improved selective refinement network for face detection[J]. arXiv preprint arXiv:1901.06651, 2019.
[20] Li Z, Tang X, Han J, et al. PyramidBox++: High Performance Detector for Finding Tiny Face[J]. arXiv preprint arXiv:1904.00386, 2019.
[21] Zhang S, Zhu X, Lei Z, et al. Faceboxes: A CPU real-time face detector with high accuracy[C]//2017 IEEE International Joint Conference on Biometrics (IJCB). IEEE, 2017: 1-9.
[22] Zhang H, Cisse M, Dauphin Y N, et al. mixup: Beyond empirical risk minimization[J]. arXiv preprint arXiv:1710.09412, 2017.
[23] Chi C, Zhang S, Xing J, et al. Selective refinement network for high performance face detection[C]//Proceedings of the AAAI Conference on Artificial Intelligence. 2019, 33: 8231-8238.
[24] Li J, Wang Y, Wang C, et al. Dsfd: dual shot face detector[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2019: 5060-5069.
[25] Zhang S, Wen L, Shi H, et al. Single-shot scale-aware network for real-time face detection[J]. International Journal of Computer Vision, 2019, 127(6-7): 537-559.
[26] Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition[J]. arXiv preprint arXiv:1409.1556, 2014.
[27] He K, Zhang X, Ren S, et al. Deep residual learning for image recognition[C]// Proceedings of the IEEE conference on computer vision and pattern recognition. 2016: 770-778.
[28] Xie S, Girshick R, Dollár P, et al. Aggregated residual transformations for deep neural networks[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2017: 1492-1500.
[29] Iandola F, Moskewicz M, Karayev S, et al. Densenet: Implementing efficient convnet descriptor pyramids[J]. arXiv preprint arXiv:1404.1869, 2014.
[30] Howard A G, Zhu M, Chen B, et al. Mobilenets: Efficient convolutional neural networks for mobile vision applications[J]. arXiv preprint arXiv:1704.04861, 2017.

[31] Sandler M, Howard A, Zhu M, et al. Mobilenetv2: Inverted residuals and linear bottlenecks[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2018: 4510-4520.
[32] Bazarevsky V, Kartynnik Y, Vakunov A, et al. BlazeFace: Sub-millisecond Neural Face Detection on Mobile GPUs[J]. arXiv preprint arXiv:1907.05047, 2019.
[33] He Y, Xu D, Wu L, et al. LFFD: A Light and Fast Face Detector for Edge Devices[J]. arXiv preprint arXiv:1904.10633, 2019.
[34] Zhu R, Zhang S, Wang X, et al. Scratchdet: Exploring to train single-shot object detectors from scratch[J]. arXiv preprint arXiv:1810.08425, 2018, 2.
[35] Lin T Y, Maire M, Belongie S, et al. Microsoft coco: Common objects in context[C]// European conference on computer vision. Springer, Cham, 2014: 740-755.
[36] Najibi M, Samangouei P, Chellappa R, et al. Ssh: Single stage headless face detector[C]//Proceedings of the IEEE International Conference on Computer Vision. 2017: 4875-4884.
[37] Sa. Earp, P. Noinongyao, J. Cairns, A. Ganguly Face Detection with Feature Pyramids and Landmarks. arXiv preprint arXiv:1912.00596, 2019.
[38] Goodfellow I J, Warde-Farley D, Mirza M, et al. Maxout networks[J]. arXiv preprint arXiv:1302.4389, 2013.
[39] Zhu C, Tao R, Luu K, et al. Seeing Small Faces from Robust Anchor's Perspective[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2018: 5127-5136.
[40] F. Zhang, X. Fan, G. Ai, J. Song, Y. Qin, J. Wu Accurate Face Detection for High Performance. arXiv preprint arXiv:1905.01585, 2019.
[41] Yu J, Jiang Y, Wang Z, et al. Unitbox: An advanced object detection network[C]// Proceedings of the 24th ACM international conference on Multimedia. ACM, 2016: 516-520.
[42] Deng J, Guo J, Zhou Y, et al. RetinaFace: Single-stage Dense Face Localisation in the Wild[J]. arXiv preprint arXiv:1905.00641, 2019.
[43] Tian W, Wang Z, Shen H, et al. Learning better features for face detection with feature fusion and segmentation supervision[J]. arXiv preprint arXiv:1811.08557, 2018.
[44] Y. Zhang, X. Xu, X. Liu Robust and High Performance Face Detector. arXiv preprint arXiv:1901.02350, 2019.
[45] S. Zhang, C. Chi, Z. Lei, Stan Z. Li RefineFace: Refinement Neural Network for High Performance Face Detection. arXiv preprint arXiv:1909.04376, 2019.
[46] Wang J, Yuan Y, Li B, et al. Sface: An efficient network for face detection in large scale variations[J]. arXiv preprint arXiv:1804.06559, 2018.
[47] Zheng Z, Wang P, Liu W, et al. Distance-IoU Loss: Faster and Better Learning for Bounding Box Regression[J]. arXiv preprint arXiv:1911.08287, 2019.
[48] Bay H, Tuytelaars T, Van Gool L. Surf: Speeded up robust features[C]//European conference on computer vision. Springer, Berlin, Heidelberg, 2006: 404-417.

[49] Yang B, Yan J, Lei Z, et al. Aggregate channel features for multi-view face detection[C]//IEEE international joint conference on biometrics. IEEE, 2014: 1-8.
[50] Everingham M, Van Gool L, Williams C K I, et al. The PASCAL visual object classes challenge 2007 (VOC2007) results[J]. 2007.
[51] Redmon J, Farhadi A. Yolov3: An incremental improvement[J]. arXiv preprint arXiv:1804.02767, 2018.

作者简介

振华、欢欢、晓林，均为美团视觉智能中心工程师。

招聘信息

美团视觉智能中心基础视觉组的主要职责是夯实视觉智能底层核心基础技术，为集团业务提供平台级视觉解决方案。主要方向有基础模型优化、大规模分布式训练、Server 效率优化、移动端适配优化和创新产品孵化。

欢迎计算机视觉相关领域小伙伴加入我们，简历可发邮件至 tech@meituan.com（邮件标题注明：美团视觉智能中心基础视觉组）。

技术解析 | 横纵一体的无人车控制方案

作者：学韬

01 两位“司机”操控下的无人车

一辆车可以同时由两个司机操控行驶吗？好像除了驾校的教练车以外，没有车是这么开的。

但这个在常人看来有些匪夷所思的问题，它的答案对于无人驾驶领域而言却是肯定的——经典的车辆控制方案，是通过两个分别控制纵向、横向的控制器配合实现的。

如图 1 所示，在自动驾驶行业的经典控制方案中，横向控制与纵向控制的求解是模型解耦的独立算法 [1,2]，仅通过单向、单次的依赖参数传递进行配合（虚线框中为控制模块，箭头表示参数依赖关系）。

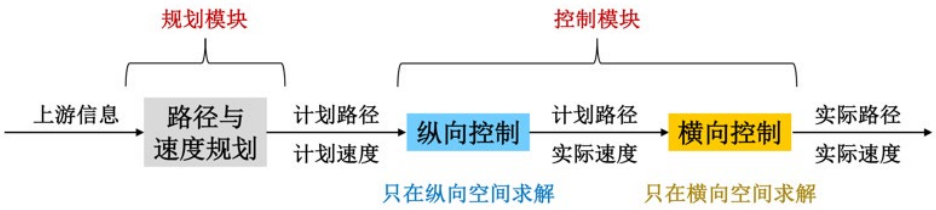


图 1　经典的横纵分离控制方案

不难想象，这相当于“两个司机同开一车”，纵向司机操作踏板（不需看路、只需看速度表）来实现计划速度，横向司机操作方向盘（需要看路）来实现计划路径。控制模块集二人之力，尽可能准确地“实现”上游规划模块给出的规划轨迹（包含计划路径和计划速度）。

这种乍看之下“不合常理”的横纵分离控制方案被业界广泛采用，是有其原因的：

1. 大事化小 ——将轨迹跟踪问题分解为纵、横向两个子问题，能够减小单个问

题的复杂度，并且在模型中避免交叉乘积等非线性（从而采用线性的横向模型），有利于控制的快速求解。

2. 耦合不强 ——常规车辆的转向结构往往限定在 30 度角以内，行车路径的曲率有限，转向轮横向力在车身纵轴的分量很小，纵向控制基本不受横向状态影响、可独立进行。

02 横纵分离方案的潜在问题

经典的横纵分离控制方案虽然是可行的，但显然不符合人类的驾驶方式。

事实上，对于轨迹跟踪这一任务，横向、纵向是紧密联系的 [3–5]，主要包括以下三个方面。

(1) 车辆动态建模包含横纵的相互耦合

真实车辆存在天然的横纵耦合，真正的老司机在操控时会考虑车辆横向行为、纵向行为之间的相互影响。例如，过弯时的纵向车速会影响横向加速度，而转向动作也会有纵向制动效果。事实上，运动学耦合 [1]、轮胎力耦合 [2]、载荷转移耦合 [3] 是车辆模型中的三大横纵耦合 [4]。

横纵分离控制方案中，纵向控制、横向控制各自采用独立的模型，只能通过状态参数进行交互，因此无法在求解前对上述耦合进行合理描述，而模型的准确性会进一步影响到控制解的最优性。

(2) 行车约束构建需要横纵的共同参与

行车约束决定了控制量的可行空间，可以理解成老司机对行驶安全范围和车辆性能极限的把握。有一些行车约束的描述可以基于横 / 纵单个方向的特征，如转向角上限约束、踏板上限约束等；但也有一些行车约束的描述必须要横、纵两个方向特征的共同参与，如位置边界约束（横坐标与纵坐标共同参与）、最大向心加速度约束（横向转角与纵向速度共同参与）等。

横纵分离控制方案中，横、纵控制算法各自只拥有一个方向的求解空间，无法描述横

纵联合参与的行车约束。以最大向心加速度约束为例，分离方案中的横向控制只能通过抑制转向角、不能通过减速来避免向心加速度越限，实际丢失了一部分可行集，如下图所示：

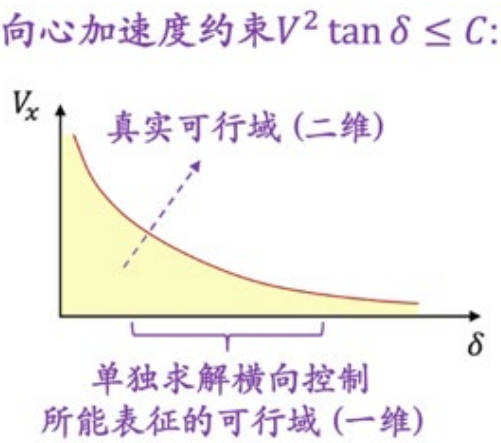


图 2　向心加速度约束所对应可行域的示意图

(3) 跟踪性能评价存在横纵的彼此竞争

轨迹跟踪看重的是横纵两个方向的综合跟踪性能指标，而两个单向指标存在一定的竞争关系 [6]。横纵分离的控制方案中，横、纵控制算法各自为政，分别只盯着一个方向的跟踪性能指标，某些场景下可能顾此失彼、出现横纵跟踪性能失衡的结果。

如高速过弯时，纵向控制能轻易地跟踪目标速度，但会提高横向控制在时间和角度上的精度要求，增大横向误差。但如果从统筹的角度来看，此时的纵向跟踪应适当让步、主动减速，在横向上换取更多的性能改善。

03 横纵一体方案的设计思路

针对横纵分离控制的上述问题，我们尝试开发一种工程可行的横纵一体控制方案。

针对上述提到的、轨迹跟踪任务中横纵之间的三方面联系，横纵一体控制方案的设计要点包括：

采用横纵耦合的车辆建模——进而对车辆的动态特性进行更准确的描述。

采用横纵联合的约束形式——进而对控制量的可行集进行更完备的构建。

采用横纵综合的性能评价——进而对两个单向的跟踪性能进行统筹协调。

需要注意的是，由于横纵耦合建模难以避免交叉乘积等非线性 [1,7]，克服非线性问题、使控制器达到工程实用的计算速度是方案实现中的核心难点。

现有文献中考虑横纵耦合模型的控制研究通常采用基于滑模控制（Sliding Mode Control，SMC）的方案 [3,4,8–10]，构造特定的变结构控制律直接进行非线性控制，使系统状态的演变保持在预先设计的滑动面邻域内（滑动面通常设计为横纵向误差渐趋至零的状态面）。但非线性控制律无法考虑状态约束，且其构造结果与难度与模型强相关，控制律的可迁移性较差。

在这里，我们采用另一种技术方案——线性时变的模型预测控制（Linear Time-Varying Model Predictive Control，LTV-MPC）来处理非线性的横纵耦合模型。

该方案的控制效果和计算性能介于常规（线性时不变）MPC 和非线性 MPC 之间，是一种针对非线性系统或时变系统的实用控制方法 [11–13]，并且有灵活的约束处理能力 [1,13]。

LTV-MPC 方案的具体实现将在下一部分介绍。

概括来说，该方案的本质是在每一个控制帧构建一个控制量序列的优化问题，横纵耦合建模、横纵联合约束、横纵跟踪统筹这三个设计要点分别对应这一优化问题的状态方程约束、不等式约束、评价函数，如图 3 所示。

图 3　横纵一体控制方案的设计思路

需要注意的是，“线性时变模型预测控制”中的“线性”表示进入求解器的状态方程约束、不等式约束在每一帧都是线性的，以此确保计算速度；而“时变”表示在不同帧之间，线性的状态方程约束、不等式约束会发生变化，以此描述真实系统的非线性。

04 LTV-MPC 横纵一体控制的具体实现

4.1 横纵耦合动态与横纵联合约束的构建

建模过程需要对被控车辆全部或主要的横纵耦合进行充分描述，横纵控制量（如横向转向角、纵向加速度）u 和横纵状态量（如车辆位置、横摆角、纵向速度）x 均以待决策变量的形式加入到模型中，并构造更符合真实世界情景的横纵联合约束。

以下图所示的横纵耦合二轮运动学模型为例，取系统状态量为 $x=[Vx \quad Xr \quad Yr \quad \psi]^{\mathrm{T}}$、控制量为 $u=[\delta \quad a]^{\mathrm{T}}$（加速度指令 α 后续会通过查表映射到踏板指令），其动态特性可建模为：

$$\begin{cases} \dot{V}_x = a \\ \dot{X}_{\mathrm{r}} = V_x \cos\psi \\ \dot{Y}_{\mathrm{r}} = V_x \sin\psi \\ \dot{\psi} = V_x \tan\delta \cdot 1/L \end{cases}$$

以最大向心加速度为例，系统的横纵联合约束 $\mathbf{g}(\mathbf{x},\mathbf{u}) \le \mathbf{m}$ 可建模为：

$$\begin{cases} V_x^2 \tan\delta \cdot 1/L \le a_{\mathrm{n,max}} \\ -V_x^2 \tan\delta \cdot 1/L \le a_{\mathrm{n,max}} \end{cases}$$

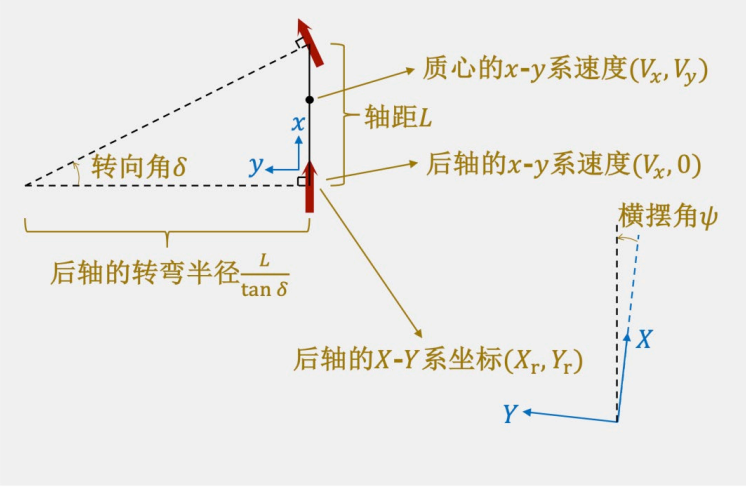


图 4　横纵耦合二轮运动学模型的示意图

二轮运动学模型虽然简单，但已经描述了车辆最重要的横纵耦合——运动学耦合。若采用更复杂的车辆模型，也可用类似的方式完成横纵耦合建模。

4.2 动态特性及约束的线性化

上一步得到的横纵耦合模型通常包含变量的交叉乘积、三角函数等非线性项，需要进行线性化以降低复杂度——这是应对横纵耦合模型非线性困难的核心步骤。其核心思想是，选取当前状态量 $\bar{\mathbf{x}}$ 和上帧控制量 $\bar{\mathbf{u}}$ 作为“基点”，然后将非线性模型在基点 $(\bar{\mathbf{x}},\bar{\mathbf{u}})$ 处进行一阶泰勒展开 [4]。

这一方法的合理性在于，在未来一小段预测域（如 1 秒）之内，系统状态 $\mathbf{x}$ 和控制量 $\mathbf{u}$ 基本会在当前时刻值 $(\bar{\mathbf{x}},\bar{\mathbf{u}})$ 的附近变动，因此在该点所得的线性化模型基本具备足够的描述精度。

具体而言，动态特性 $\dot{\mathbf{x}}=\mathbf{f}(\mathbf{x},\mathbf{u})$ 的线性化结果为：

$$\dot{\mathbf{x}}=\mathbf{A}(\mathbf{x}-\bar{\mathbf{x}})+\mathbf{B}(\mathbf{u}-\bar{\mathbf{u}})+\bar{\mathbf{d}}$$

其中：

$$\mathbf{A}=\frac{\partial \mathbf{f}}{\partial \mathbf{x}_{(\bar{\mathbf{x}},\bar{\mathbf{u}})}},\ \mathbf{B}=\frac{\partial \mathbf{f}}{\partial \mathbf{u}_{(\bar{\mathbf{x}},\bar{\mathbf{u}})}},\ \bar{\mathbf{d}}=\mathbf{f}(\bar{\mathbf{x}},\bar{\mathbf{u}})$$

仍以二轮运动学模型为例，有：

$$\mathbf{A}=\begin{bmatrix} 0 & 0 & 0 & 0 \\ cos\bar{\psi} & 0 & 0 & -\bar{V}_x sin\bar{\psi} \\ sin\bar{\psi} & 0 & 0 & \bar{V}_x cos\bar{\psi} \\ \dfrac{tan\bar{\delta}}{L} & 0 & 0 & 0 \end{bmatrix},\ \mathbf{B}=\begin{bmatrix} 0 & 1 \\ 0 & 0 \\ 0 & 0 \\ \dfrac{\bar{V}_x}{Lcos^2\bar{\delta}} & 0 \end{bmatrix},\ \bar{\mathbf{d}}=\begin{bmatrix} \bar{a} \\ \bar{V}_x cos\bar{\psi} \\ \bar{V}_x sin\bar{\psi} \\ \dfrac{\bar{V}_x tan\bar{\delta}}{L} \end{bmatrix}$$

横纵联合约束 $\mathbf{g}(\mathbf{x},\mathbf{u})\le\mathbf{m}$ 的线性化结果为：

$$\mathbf{C}(\mathbf{x}-\bar{\mathbf{x}})+\mathbf{D}(\mathbf{u}-\bar{\mathbf{u}})+\bar{\mathbf{e}}\le\mathbf{m}$$

其中：

$$\mathbf{C}=\frac{\partial \mathbf{g}}{\partial \mathbf{x}_{(\bar{\mathbf{x}},\bar{\mathbf{u}})}},\ \mathbf{D}=\frac{\partial \mathbf{g}}{\partial \mathbf{u}_{(\bar{\mathbf{x}},\bar{\mathbf{u}})}},\ \bar{\mathbf{e}}=\mathbf{g}(\bar{\mathbf{x}},\bar{\mathbf{u}})$$

仍以最大向心加速度为例，有：

$$\mathbf{C}=\begin{bmatrix} \dfrac{2\bar{V}_x tan\bar{\delta}}{L} & 0 & 0 & 0 \\ -\dfrac{2\bar{V}_x tan\bar{\delta}}{L} & 0 & 0 & 0 \end{bmatrix},\ \mathbf{D}=\begin{bmatrix} \dfrac{\bar{V}_x^2}{Lcos^2\bar{\delta}} & 0 \\ -\dfrac{\bar{V}_x^2}{Lcos^2\bar{\delta}} & 0 \end{bmatrix},\ \bar{\mathbf{e}}=\begin{bmatrix} \dfrac{\bar{V}_x^2 tan\bar{\delta}}{L} \\ -\dfrac{\bar{V}_x^2 tan\bar{\delta}}{L} \end{bmatrix}$$

完成线性化之后，为将连续模型适配数值计算求解器，需按照特定的离散时间间隔 τ（通常就是一个控制帧的长度），采用特定的离散化方法，如零阶离散矩阵变换公式 [14]：

$$\mathbf{A}_{\mathrm{d}}=e^{\mathbf{A}\tau},\ \mathbf{B}_{\mathrm{d}}=\int_0^{\tau}e^{\mathbf{A}t}\mathrm{d}t\mathbf{B}$$

进行尽可能精确的离散化处理，得到离散的线性化模型：

$$\begin{cases}\mathbf{x}_{k+1}-\bar{\mathbf{x}}=\mathbf{A}_{\mathrm{d}}(\mathbf{x}_k-\bar{\mathbf{x}})+\mathbf{B}_{\mathrm{d}}(\mathbf{u}_k-\bar{\mathbf{u}})+\bar{\mathbf{d}}_{\mathrm{d}}\\ \mathbf{C}(\mathbf{x}_k-\bar{\mathbf{x}})+\mathbf{D}(\mathbf{u}_k-\bar{\mathbf{u}})+\bar{\mathbf{e}}\leq\mathbf{m}\end{cases}$$

其中下标 k 表示 $\left(t+k\tau\right)$ 时刻的待解向量值，t 为当前时刻。

4.3 横纵综合评价函数的构建

为求解最优控制量，还需构建系统状态 x 的评价函数 $J_{\mathbf{x}}$。需要注意的是，相较于横纵分离控制针对单个方向跟踪性能的优化，横纵一体控制的目标是横纵两个方向跟踪性能的协调和统筹，因此 $J_{\mathbf{x}}$ 应该描述横纵两个方向的综合跟踪性能。

仍以二轮运动学模型为例，可定义：

$$J_{\mathbf{x}}=q_V(V_x-V_x^{\mathrm{ref}})^2+q_X(X_{\mathrm{r}}-X_{\mathrm{r}}^{\mathrm{ref}})^2+q_Y(Y_{\mathrm{r}}-Y_{\mathrm{r}}^{\mathrm{ref}})^2+q_\psi(\psi-\psi^{\mathrm{ref}})^2$$

其中 q_V 、q_X 、q_Y 、q_ψ 分别为速度误差、纵向位置误差、横向位置误差、横摆角误差在综合评价函数中的惩罚权重——前两者为纵向评价部分，后两者为横向评价部分。上标 ref 表示计划轨迹，由上游的规划模块给出。

在这样的评价函数设计之下，某时刻 $J_{\mathbf{x}}$ 的值越小，则表示该时刻的综合跟踪性能越好。并且不难看出，改善大误差项对 $J_{\mathbf{x}}$ 的边际贡献更大，因此最小化 $J_{\mathbf{x}}$ 可以实现横纵跟踪性能的统筹协调，避免某个方向的误差过大。

例如高速过弯场景下，横纵一体控制器会在过弯处进行适当的主动减速，牺牲少许的纵向跟踪性能来换取更多的横向跟踪性能，使 $J_{\mathbf{x}}$ 尽可能小——这种全局统筹的思想其实更符合真实的运动控制需求。

4.4 控制量求解与下发

选取预测步数为 N（预测域为 $N\tau$），结合 4.2 步所得的线性化模型和 4.3 步所得的评价函数，能够形成如下的 MPC 问题 [5]：

$$
\begin{gathered}
\min_{\{\mathbf{u}_k\}} \sum_{k=1}^{N} \left[(\mathbf{x}-\mathbf{x}^{\mathrm{ref}})^{\mathrm{T}} \mathbf{Q} (\mathbf{x}-\mathbf{x}^{\mathrm{ref}}) + \Delta \mathbf{u}_k^{\mathrm{T}} \mathbf{R} \Delta \mathbf{u}_k \right] \\
\text{s.t.} \quad (\text{for k} = 0, 1, \ldots, \text{N}-1) \\
\mathbf{x}_{k+1} - \bar{\mathbf{x}} = \mathbf{A}_{\mathrm{d}} (\mathbf{x}_k - \bar{\mathbf{x}}) + \mathbf{B}_{\mathrm{d}} (\mathbf{u}_k - \bar{\mathbf{u}}) + \bar{\mathbf{d}}_{\mathrm{d}} \\
\mathbf{C} (\mathbf{x}_k - \bar{\mathbf{x}}) + \mathbf{D} (\mathbf{u}_k - \bar{\mathbf{u}}) + \bar{\mathbf{e}} \leq \mathbf{m} \\
\mathbf{u}_{k+1} = \mathbf{u}_k + \Delta \mathbf{u}_{k+1} \\
\mathbf{u}_{\min} \leq \mathbf{u}_{k+1} \leq \mathbf{u}_{\max}, \ \Delta \mathbf{u}_{\min} \leq \Delta \mathbf{u}_{k+1} \leq \Delta \mathbf{u}_{\max}
\end{gathered}
$$

该问题本质是一个线性约束二次规划问题。接下来，只需同常规 MPC 一样进行求解，即可解得最优的控制序列 $\mathbf{u}_1^* \sim \mathbf{u}_N^*$，且状态预测轨迹 $\mathbf{x}_1^* \sim \mathbf{x}_N^*$ 相伴而生，这一轨迹能够可视化表示为与规划轨迹类似的、包含速度信息的行进轨迹。

将 $\mathbf{u}_1^*$ 下发后，即可等待下一个控制帧的到来，然后重新回到 4.2 步（如需更新模型，则应回到 4.1 步），循环往复执行。

05 小结

本文针对自动驾驶横纵分离控制方案无法描述车辆横纵耦合、不能考虑横纵联合约束、没有横纵统筹能力等不足，给出了一种考虑横纵联合约束的横纵一体车辆控制方案。

该方案支持在建模时考虑车辆的横纵耦合，提高模型准确性；采用时变线性化 MPC 框架来克服横纵耦合及约束所带来的非线性，改善求解耗时和可靠性；在横纵联合可行域中进行综合跟踪性能的寻优，具有一定的横纵统筹能力。

注释

1. 横纵以车身定义，车身旋转时，相邻时刻的横纵物理量会相互贡献

2. 纵向车速高时，横向力会对转向更敏感；横向转向角大时，纵向力会有额外的制动分量
3. 纵向减速时，部分载荷向前轮转移，增强轮胎横向力
4. 另一种常用的线性化方式是在 (0，0) 处展开，但其描述精度相对较差。还有一种线性化方法是为每个预测时刻 $\left(t+k\tau\right)$ 各取一个基点 $\left(\overline{\mathbf{x}}_k,\overline{\mathbf{u}}_k\right)$、得到一簇基点 [12]，也能达到较好的描述精度，但为简化表达，本文不再赘述。
5. 在这里，$J_{\mathbf{x}}$ 写成了目标函数中的矩阵形式 $(\mathbf{x}-\mathbf{x}^{\mathrm{ref}})^{\mathrm{T}}\mathbf{Q}(\mathbf{x}-\mathbf{x}^{\mathrm{ref}})$。

参考文献

[1] 陈慧岩 , 陈舒平 , 龚建伟 . 智能汽车横向控制方法研究综述 [J]. 兵工学报 ,2017, 38(6):1203-1214.

[2] [佐思产研 . 谈谈无人车横向控制](https://mp.weixin.qq.com/s/pqs8UccxqfzaEnLaul7WBQ)[EB/OL].

[3] Pham H, Hedrick K, Tomizuka M. Combined lateral and longitudinalcontrol of vehicles for IVHS[C]//American Control Conference.Baltimore, MD, US: IEEE, 1994: 1205-1206.

[4] Lim E H M, Hedrick J K. Lateral and longitudinal vehicle controlcoupling for automated vehicle operation[C]//Proceedings of theAmerican Control Conference. San Diego, CA, US: IEEE, 1999:3676-3680.

[5] 美团无人配送 .【 技术解析 】无人车横向控制解读 [EB/OL].

[6] Lam D, Manzie C, Good M. Model predictive contouring control[C]//49thIEEE Conference on Decision and Control (CDC). IEEE, 2010: 6137-6142.

[7] Rajamani R. Vehicle Dynamics and Control[M]. Springer Science, 2006.

[8] SwaroopD, Yoon S M. Integrated lateral and longitudinal vehicle control foran emergency lane change manoeuvre design[J]. International Journalof Vehicle Design, 1999, 21(2-3): 161-174.

[9] RajamaniR, Tan H S, Law B K, et al. Demonstration of integrated longitudinaland lateral control for the operation of automated vehicles inplatoons[J]. IEEE Transactions on Control Systems Technology, 2000,8(4): 695-708.

[10] 郭景华 , 罗禹贡 , 李克强 . 智能车辆运动控制系统协同设计 [J]. 清华大学学报 (自然科学版),2015(7): 761-768.

[11] BamiehB, Giarre L. Identification of linear parameter varying models[J]. International Journal of Robust and Nonlinear Control:IFAC - Affiliated Journal, 2002, 12(9): 841-853.

[12] FalconeP, Borrelli F, Tseng H E, et al. Linear time - varying modelpredictive control and its application to active steering systems:Stability analysis and experimental validation[J]. InternationalJournal of Robust and Nonlinear Control: IFAC - Affiliated Journal,2008, 18(8): 862-875.

[13] Xing X, Lin J, Brandon N, et al. Time-varying model predictivecontrol of a reversible-SOC energy-storage plant based on the linearparameter-varying method[J]. IEEE Transactions on Sustainable Energy,2020, 11(3): 1589-1600.
[14] Chen C T. Linear system theory and design[M]. New York, UnitedStates: Oxford University Press, 1999: 90 - 92.

算法

用能力支撑了每天数十亿次的交易量；

用实力攀登算法领域最高学术殿堂；

用技术绘制美团点评最美的 AI 全景图。

美团算法团队正在构建的 AI 相关技术囊括了视觉、语音、自然语言处理、机器学习、知识图谱等。以美团 / 大众点评 App 搜索、推荐为核心，面向外卖配送的策略、调度算法、定价系统，延伸到无人配送的自动驾驶、智能耳机里的语音识别、人脸识别，再到连接用户端的客服系统，连接商家端的金融体系、供应链系统也汇聚了美团点评正在构建的庞大知识图谱……

探索无止境，实践出真知。美团点评算法团队孜孜不倦的学习与探索，并在实际业务场景中引入新技术，总结新领悟。

美团 BERT 的探索和实践

杨扬　佳昊　金刚

2018 年，自然语言处理（Natural Language Processing，NLP）领域最激动人心的进展莫过于**预训练语言模型**，包括基于 RNN 的 ELMo[1] 和 ULMFiT[2]，基于 Transformer[3] 的 OpenAI GPT[4] 及 Google BERT[5] 等。下图 1 回顾了近年来预训练语言模型的发展史以及最新的进展。预训练语言模型的成功，证明了我们可以从海量的无标注文本中学到潜在的语义信息，而无需为每一项下游 NLP 任务单独标注大量训练数据。此外，预训练语言模型的成功也开创了 NLP 研究的新范式 [6]，即首先使用大量无监督语料进行语言模型预训练（Pre-training），再使用少量标注语料进行微调（Fine-tuning）来完成具体 NLP 任务（分类、序列标注、句间关系判断和机器阅读理解等）。

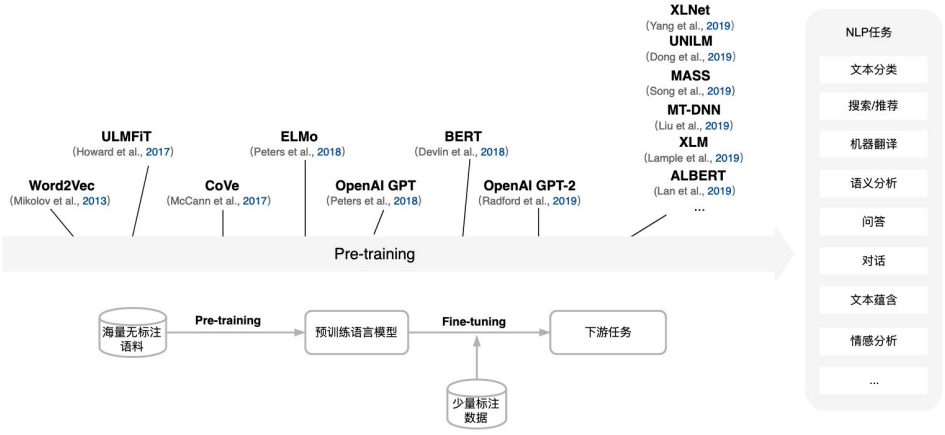


图 1　NLP Pre-training and Fine-tuning 新范式及相关扩展工作

所谓的“预训练”，其实并不是什么新概念，这种“Pre-training and Fine-tuning”的方法在图像领域早有应用。2009 年，邓嘉、李飞飞等人在 CVPR 2009 发布了 ImageNet 数据集 [7]，其中 120 万张图像分为 1000 个类别。基于 ImageNet，以图像分类为目标使用深度卷积神经网络（如常见的 ResNet、VCG、Inception 等）

进行预训练，得到的模型称为预训练模型。针对目标检测或者语义分割等任务，基于这些预训练模型，通过一组新的全连接层与预训练模型进行拼接，利用少量标注数据进行微调，将预训练模型学习到的图像分类能力迁移到新的目标任务。预训练的方式在图像领域取得了广泛的成功，比如有学者将 ImageNet 上学习得到的特征表示用于 PSACAL VOC 上的物体检测，将检测率提高了 20%[8]。

他山之石，可以攻玉。图像领域预训练的成功也启发了 NLP 领域研究，深度学习时代广泛使用的词向量（即词嵌入，Word Embedding）即属于 NLP 预训练工作。使用深度神经网络进行 NLP 模型训练时，首先需要将待处理文本转为词向量作为神经网络输入，词向量的效果会影响到最后模型效果。词向量的效果主要取决于训练语料的大小，很多 NLP 任务中有限的标注语料不足以训练出足够好的词向量，通常使用跟当前任务无关的大规模未标注语料进行词向量预训练，因此预训练的另一个好处是能增强模型的泛化能力。目前，大部分 NLP 深度学习任务中都会使用预训练好的词向量（如 Word2Vec[9] 和 GloVe[10] 等）进行网络初始化（而非随机初始化），从而加快网络的收敛速度。

预训练词向量通常只编码词汇间的关系，对上下文信息考虑不足，且无法处理一词多义问题。如“bank”一词，根据上下文语境不同，可能表示“银行”，也可能表示“岸边”，却对应相同的词向量，这样显然是不合理的。为了更好的考虑单词的上下文信息，Context2Vec[11] 使用两个双向长短时记忆网络（Long Short Term Memory，LSTM）[12] 来分别编码每个单词左到右（Left-to-Right）和右到左（Right-to-Left）的上下文信息。类似地，ELMo 也是基于大量文本训练深层双向 LSTM 网络结构的语言模型。ELMo 在词向量的学习中考虑深层网络不同层的信息，并加入到单词的最终 Embedding 表示中，在多个 NLP 任务中取得了提升。ELMo 这种使用预训练语言模型的词向量作为特征输入到下游目标任务中，被称为 Feature-based 方法。

另一种方法是微调（Fine-tuning）。GPT、BERT 和后续的预训练工作都属于这一范畴，直接在深层 Transformer 网络上进行语言模型训练，收敛后针对下游目标任务进行微调，不需要再为目标任务设计 Task-specific 网络从头训练。关于

NLP 领域的预训练发展史，张俊林博士写过一篇很详实的介绍 [13]，本文不再赘述。

Google AI 团队提出的预训练语言模型 BERT（Bidirectional Encoder Representations from Transformers），在 11 项自然语言理解任务上刷新了最好指标，可以说是近年来 NLP 领域取得的最重大的进展之一。BERT 论文也斩获 NLP 领域顶会 NAACL 2019 的最佳论文奖，BERT 的成功也启发了大量的后续工作，不断刷新了 NLP 领域各个任务的最佳水平。有 NLP 学者宣称，属于 NLP 的 ImageNet 时代已经来临 [14]。

美团点评作为中国领先的生活服务电子商务平台，涵盖搜索、推荐、广告、配送等多种业务场景，几乎涉及到各种类型的自然语言处理任务。以大众点评为例，迄今为止积累了近 40 亿条文本 UGC，如何高效而准确地完成对海量 UGC 的自然语言理解和处理是美团点评技术团队面临的挑战之一。美团点评 NLP 团队一直紧跟业界前沿技术，开展了基于美团点评业务数据的预训练研究工作，训练了更适配美团点评业务场景的 MT-BERT 模型，通过微调将 MT-BERT 落地到多个业务场景中，并取得了不错的业务效果。

BERT 是基于 Transformer 的深度双向语言表征模型，基本结构如图 2 所示，本质上是利用 Transformer 结构构造了一个多层双向的 Encoder 网络。Transformer 是 Google 在 2017 年提出的基于自注意力机制（Self-attention）的深层模型，在包括机器翻译在内的多项 NLP 任务上效果显著，超过 RNN 且训练速度更快。不到一年时间内，Transformer 已经取代 RNN 成为神经网络机器翻译的 State-Of-The-Art（SOTA）模型，包括谷歌、微软、百度、阿里、腾讯等公司的线上机器翻译模型都已替换为 Transformer 模型。关于 Transformer 的详细介绍可以参考 Google 论文《Attention is all you need》[3]。

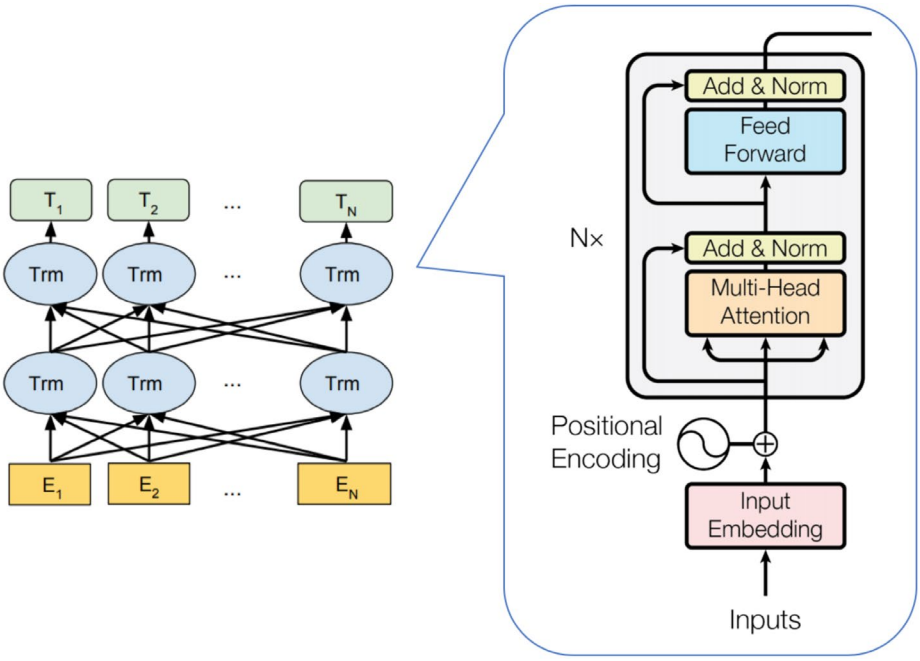


图 2　BERT 及 Transformer 网络结构示意图

模型结构

如表 1 所示，根据参数设置的不同，Google 论文中提出了 Base 和 Large 两种 BERT 模型。

表1　BERT Base和Large模型参数对比

模型	Layers	Hidden Size	Attention Head	参数数量
Base	12	768	12	110M
Large	24	1024	16	340M

输入表示

针对不同的任务，BERT 模型的输入可以是单句或者句对。对于每一个输入的 Token，它的表征由其对应的词表征（Token Embedding）、段表征（Segment Embedding）和位置表征（Position Embedding）相加产生，如图 3 所示：

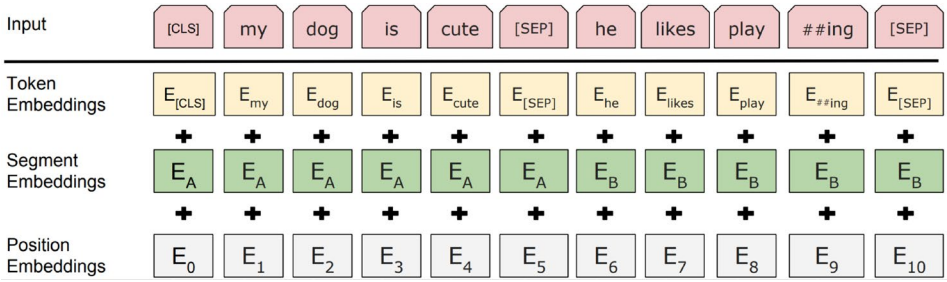


图 3　BERT 模型的输入表示

- 对于英文模型，使用了 Wordpiece 模型来产生 Subword 从而减小词表规模；对于中文模型，直接训练基于字的模型。
- 模型输入需要附加一个起始 Token，记为 [CLS]，对应最终的 Hidden State（即 Transformer 的输出）可以用来表征整个句子，用于下游的分类任务。
- 模型能够处理句间关系。为区别两个句子，用一个特殊标记符 [SEP] 进行分隔，另外针对不同的句子，将学习到的 Segment Embeddings 加到每个 Token 的 Embedding 上。
- 对于单句输入，只有一种 Segment Embedding；对于句对输入，会有两种 Segment Embedding。

预训练目标

BERT 预训练过程包含两个不同的预训练任务，分别是 Masked Language Model 和 Next Sentence Prediction 任务。

Masked Language Model（MLM）

通过随机掩盖一些词（替换为统一标记符 [MASK]），然后预测这些被遮盖的词来训练双向语言模型，并且使每个词的表征参考上下文信息。

这样做会产生两个缺点：（1）会造成预训练和微调时的不一致，因为在微调时 [MASK] 总是不可见的；（2）由于每个 Batch 中只有 15% 的词会被预测，因此模型的收敛速度比起单向的语言模型会慢，训练花费的时间会更长。对于第一个缺点的解

决办法是，把 80% 需要被替换成 [MASK] 的词进行替换，10% 的随机替换为其他词，10% 保留原词。由于 Transformer Encoder 并不知道哪个词需要被预测，哪个词是被随机替换的，这样就强迫每个词的表达需要参照上下文信息。对于第二个缺点目前没有有效的解决办法，但是从提升收益的角度来看，付出的代价是值得的。

Next Sentence Prediction（NSP）

为了训练一个理解句子间关系的模型，引入一个下一句预测任务。这一任务的训练语料可以从语料库中抽取句子对包括两个句子 A 和 B 来进行生成，其中 50% 的概率 B 是 A 的下一个句子，50% 的概率 B 是语料中的一个随机句子。NSP 任务预测 B 是否是 A 的下一句。NSP 的目的是获取句子间的信息，这点是语言模型无法直接捕捉的。

Google 的论文结果表明，这个简单的任务对问答和自然语言推理任务十分有益，但是后续一些新的研究 [15] 发现，去掉 NSP 任务之后模型效果没有下降甚至还有提升。我们在预训练过程中也发现 NSP 任务的准确率经过 1–2 个 Epoch 训练后就能达到 98%–99%，去掉 NSP 任务之后对模型效果并不会有太大的影响。

数据 & 算力

Google 发布的英文 BERT 模型使用了 BooksCorpus（800M 词汇量）和英文 Wikipedia（2500M 词汇量）进行预训练，所需的计算量非常庞大。BERT 论文中指出，Google AI 团队使用了算力强大的 Cloud TPU 进行 BERT 的训练，BERT Base 和 Large 模型分别使用 4 台 Cloud TPU（16 张 TPU）和 16 台 Cloud TPU（64 张 TPU）训练了 4 天（100 万步迭代，40 个 Epoch）。但是，当前国内互联网公司主要使用 Nvidia 的 GPU 进行深度学习模型训练，因此 BERT 的预训练对于 GPU 资源提出了很高的要求。

美团 BERT（MT-BERT）的探索分为四个阶段：（1）开启混合精度实现训练加速；（2）在通用中文语料基础上加入大量美团点评业务语料进行模型预训练，完成领域迁移；（3）预训练过程中尝试融入知识图谱中的实体信息；（4）通过在业务数据上进行微调，支持不同类型的业务需求。MT-BERT 整体技术框架如图 4 所示：

图 4　MT-BERT 整体技术框架

基于美团点评 AFO 平台的分布式训练

正如前文所述，BERT 预训练对于算力有着极大要求，我们使用的是美团内部开发的 AFO[16]（AI Framework On Yarn）框架进行 MT-BERT 预训练。AFO 框架基于 YARN 实现数千张 GPU 卡的灵活调度，同时提供基于 Horovod 的分布式训练方案，以及作业弹性伸缩与容错等能力。Horovod 是 Uber 开源的深度学习工具 [17]，它的发展吸取了 Facebook《一小时训练 ImageNet》论文 [18] 与百度 Ring Allreduce[19] 的优点，可为用户实现分布式训练提供帮助。根据 Uber 官方分别使用标准分布式 TensorFlow 和 Horovod 两种方案，分布式训练 Inception V3 和 ResNet-101 TensorFlow 模型的实验验证显示，随着 GPU 的数量增大，Horovod 性能损失远小于 TensorFlow，且训练速度可达到标准分布式 TensorFlow 的近两倍。相比于 Tensorflow 分布式框架，Horovod 在数百张卡的规模上依然可以保证稳定的加速比，具备非常好的扩展性。

Horovod 框架的并行计算主要用到了两种分布式计算技术：控制层的 Open

MPI 和数据层的 Nvidia NCCL。控制层面的主要作用是同步各个 Rank（节点），因为每个节点的运算速度不一样，运算完每一个 Step 的时间也不一样。如果没有一个同步机制，就不可能对所有的节点进行梯度平均。Horovod 在控制层面上设计了一个主从模式，Rank 0 为 Master 节点，Rank1-n 为 Worker 节点，每个 Worker 节点上都有一个消息队列，而在 Master 节点上除了一个消息队列，还有一个消息 Map。每当计算框架发来通信请求时，比如要执行 Allreduce，Horovod 并不直接执行 MPI，而是封装了这个消息并推入自己的消息队列，交给后台线程去处理。后台线程采用定时轮询的方式访问自己的消息队列，如果非空，Woker 会将自己收到的所有 Tensor 通信请求都发给 Master。因为是同步 MPI，所以每个节点会阻塞等待 MPI 完成。Master 收到 Worker 的消息后，会记录到自己的消息 Map 中。如果一个 Tensor 的通信请求出现了 n 次，也就意味着，所有的节点都已经发出了对该 Tensor 的通信请求，那么这个 Tensor 就需要且能够进行通信。Master 节点会挑选出所有符合要求的 Tensor 进行 MPI 通信。不符合要求的 Tensor 继续留在消息 Map 中，等待条件满足。决定了 Tensor 以后，Master 又会将可以进行通信的 Tensor 名字和顺序发还给各个节点，通知各个节点可以进行 Allreduce 运算。

混合精度加速

当前深度学习模型训练过程基本采用单精度（Float 32）和双精度（Double）数据类型，受限于显存大小，当网络规模很大时 Batch Size 就会很小。Batch Size 过小一方面容易导致网络学习过程不稳定而影响模型最终效果，另一方面也降低了数据吞吐效率，影响训练速度。为了加速训练及减少显存开销，Baidu Research 和 Nvidia 在 ICLR 2018 论文中 [20] 合作提出了一种 Float32（FP32）和 Float16（FP16）混合精度训练的方法，并且在图像分类和检测、语音识别和语言模型任务上进行了有效验证。Nvidia 的 Pascal 和 Volta 系列显卡除了支持标准的单精度计算外，也支持了低精度的计算，比如最新的 Tesla V100 硬件支持了 FP16 的计算加速，P4 和 P40 支持 INT8 的计算加速，而且低精度计算的峰值要远高于单精浮点的计算峰值。

为了进一步加快 MT-BERT 预训练和推理速度，我们实验了混合精度训练方式。混合精度训练指的是 FP32 和 FP16 混合的训练方式，使用混合精度训练可以加速训练过程并且减少显存开销，同时兼顾 FP32 的稳定性和 FP16 的速度。在模型计算过程中使用 FP16 加速计算过程，模型训练过程中权重会存储成 FP32 格式（FP32 Master-weights），参数更新时采用 FP32 类型。利用 FP32 Master-weights 在 FP32 数据类型下进行参数更新可有效避免溢出。 此外，一些网络的梯度大部分在 FP16 的表示范围之外，需要对梯度进行放大使其可以在 FP16 的表示范围内，因此进一步采用 Loss Scaling 策略通过对 Loss 进行放缩，使得在反向传播过程中梯度在 FP16 的表示范围内。

为了提高预训练效率，我们在 MT-BERT 预训练中采用了混合精度训练方式。

加速效果

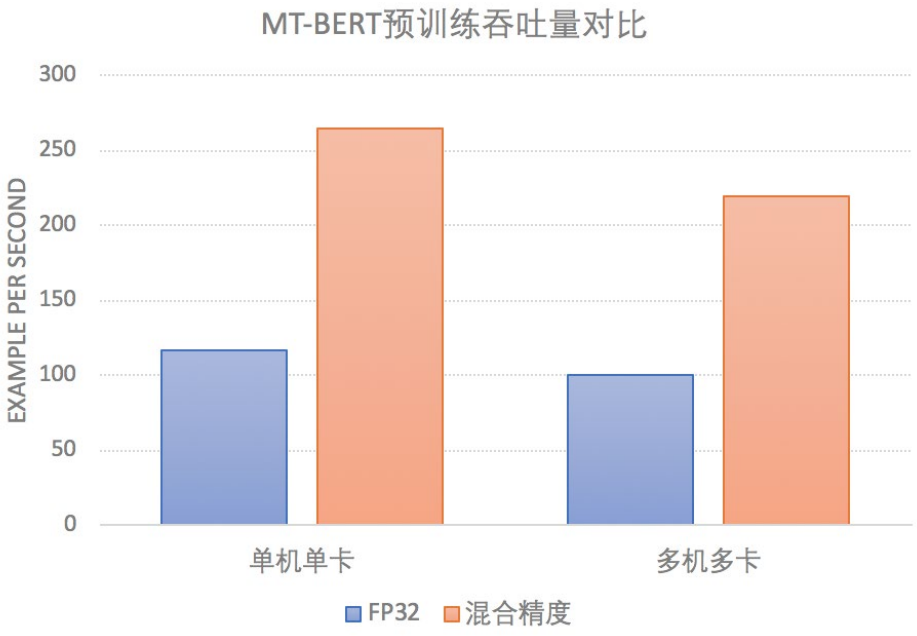


图 5　MT-BERT 开启混合精度在 Nvidia V100 上训练吞吐量的对比
（Tensorflow 1.12；Cuda 10.0；Horovod 0.15.2）

如图 5 所示，开启混合精度的训练方式在单机单卡和多机多卡环境下显著提升了训练速度。为了验证混合精度模型会不会影响最终效果，我们分别在美团点评业务和通用 Benchmark 数据集上进行了微调实验，结果见表 2 和表 3。

表2　开启混合精度训练的MT-BERT模型在美团点评业务Benchmark上效果对比

数据集	Metric	MT-BERT FP32	MT-BERT 混合精度	Google BERT
细粒度情感分析	Macro-F1	72.04%	72.25%	71.63%
Query 意图分类	F1	93.27%	93.13%	92.68%
Query 成分分析（NER）	F1	91.46%	91.05%	90.66%

表3　开启混合精度训练的MT-BERT模型在中文通用Benchmark上效果对比

数据集	Metric	MT-BERT FP32	MT-BERT 混合精度	Google BERT
MSRA-NER	F1	95.89%	95.75%	95.76%
LCQMC	Accuracy	86.74%	85.87%	86.06%
ChnSentiCorp	Accuracy	95.00%	94.92%	92.25%
NLPCC-DBQA	MRR	94.07%	93.24%	93.55%
XNLI	Accuracy	78.10%	76.57%	77.47%

通过表 2 和表 3 结果可以发现，开启混合精度训练的 MT-BERT 模型并没有影响效果，反而训练速度提升了 2 倍多。

领域自适应

Google 发布的中文 BERT 模型是基于中文维基百科数据训练得到，属于通用领域预训练语言模型。由于美团点评积累了大量业务语料，比如用户撰写的 UGC 评论和商家商品的文本描述数据，为了充分发挥领域数据的优势，我们考虑在 Google 中文 BERT 模型上加入领域数据继续训练进行领域自适应（Domain Adaptation），使得模型更加匹配我们的业务场景。实践证明，这种 Domain-aware Continual Training 方式，有效地改进了 BERT 模型在下游任务中的表现。由于 Google 未发布中文 BERT Large 模型，我们也从头预训练了中文 MT-BERT Large 模型。

我们选择了 5 个中文 Benchmark 任务以及 3 个美团点评业务 Benchmark 在内的 8 个数据集对模型效果进行验证。实验结果如表 4 所示，MT-BERT 在通用 Benchmark 和美团点评业务 Benchmark 上都取得了更好的效果。

表4 MT-BERT模型和Google BERT模型在8个Benchmark上的效果对比

Benchmark	Metric	Google BERT	MT-BERT
MSRA-NER	F1	95.76%	95.89%
LCQMC	Accuracy	86.06%	86.74%
ChnSentiCorp	Accuracy	92.25%	95.00%
NLPCC-DBQA	MRR	93.55%	94.07%
XNLI	Accuracy	77.47%	78.10%
细粒度情感分析	Macro-F1	71.63%	72.04%
Query 意图分类	F1	92.68%	93.27%
Query 成分分析（NER）	F1	90.66%	91.46%

知识融入

BERT 在自然语言理解任务上取得了巨大的成功，但也存在着一些不足。其一是常识（Common Sense）的缺失。人类日常活动需要大量的常识背景知识支持，BERT 学习到的是样本空间的特征、表征，可以看作是大型的文本匹配模型，而大量的背景常识是隐式且模糊的，很难在预训练数据中进行体现。其二是缺乏对语义的理解。模型并未理解数据中蕴含的语义知识，缺乏推理能力。在美团点评搜索场景中，需要首先对用户输入的 Query 进行意图识别，以确保召回结果的准确性。比如，对于“宫保鸡丁”和“宫保鸡丁酱料”两个 Query，二者的 BERT 语义表征非常接近，但是蕴含的搜索意图却截然不同。前者是菜品意图，即用户想去饭店消费，而后者则是商品意图，即用户想要从超市购买酱料。在这种场景下，BERT 模型很难像正常人一样做出正确的推理判断。

为了处理上述情况，我们尝试在 MT-BERT 预训练过程中融入知识图谱信息。知识图谱可以组织现实世界中的知识，描述客观概念、实体、关系。这种基于符号语义的计算模型，可以为 BERT 提供先验知识，使其具备一定的常识和推理能力。在我们团队之前的技术文章 [21] 中，介绍了 NLP 中心构建的大规模的餐饮娱乐知识图

谱——美团大脑。我们通过 Knowledge-aware Masking 方法将“美团大脑”的实体知识融入到 MT-BERT 预训练中。

BERT 在进行语义建模时，主要聚焦最原始的单字信息，却很少对实体进行建模。具体地，BERT 为了训练深层双向的语言表征，采用了 Masked LM（MLM）训练策略。该策略类似于传统的完形填空任务，即在输入端，随机地“遮蔽”掉部分单字，在输出端，让模型预测出这些被“遮蔽”的单字。模型在最初并不知道要预测哪些单字，因此它输出的每个单字的嵌入表示，都涵盖了上下文的语义信息，以便把被“掩盖”的单字准确的预测出来。

图 6 左侧展示了 BERT 模型的 MLM 任务。输入句子是“全聚德做的烤鸭久负盛名”。其中，“聚”，“的”，“久”3 个字在输入时被随机遮蔽，模型预训练过程中需要对这 3 个遮蔽位做出预测。

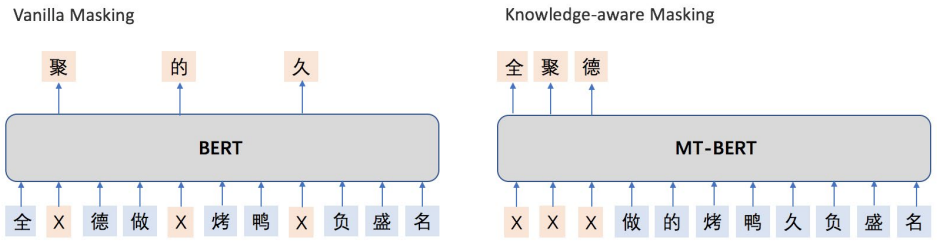


图 6　MT-BERT 默认 Masking 策略和 Whole Word Masking 策略对比

BERT 模型通过字的搭配（比如“全 X 德”），很容易推测出被“掩盖”字信息（“德”），但这种做法只学习到了实体内单字之间共现关系，并没有学习到实体的整体语义表示。因此，我们使用 Knowledge-aware Masking 的方法来预训练 MT-BERT。具体的做法是，输入仍然是字，但在随机”遮蔽”时，不再选择遮蔽单字，而是选择“遮蔽”实体对应的词。这需要我们在预训练之前，对语料做分词，并将分词结果和图谱实体对齐。图 6 右侧展示了 Knowledge-aware Masking 策略，“全聚德”被随机“遮蔽”。MT-BERT 需要根据“烤鸭”，“久负盛名”等信息，准确的预测出“全聚德”。通过这种方式，MT-BERT 可以学到“全聚德”这个实体的语义

表示，以及它跟上下文其他实体之间的关联，增强了模型语义表征能力。基于美团大脑中已有实体信息，我们在 MT-BERT 训练中使用了 Knowledge-aware Masking 策略，实验证明在细粒度情感分析任务上取得了显著提升。

表5 MT-BERT在细粒度情感分析数据集上效果

模型	Macro-F1
BERT (Vanilla masking)	72.04%
MT-BERT (Knowledge-aware Masking)	72.48%

模型轻量化

BERT 模型效果拔群，在多项自然语言理解任务上实现了最佳效果，但是由于其深层的网络结构和庞大的参数量，如果要部署上线，还面临很大挑战。以 Query 意图分类为例，我们基于 MT-BERT 模型微调了意图分类模型，协调工程团队进行了 1000QPS 压测实验，部署 30 张 GPU 线上卡参与运算，在线服务的 TP999 高达 50ms 之多，难以满足上线要求。

为了减少模型响应时间，满足上线要求，业内主要有三种模型轻量化方案。

- 低精度量化。在模型训练和推理中使用低精度（FP16 甚至 INT8、二值网络）表示取代原有精度（FP32）表示。
- 模型裁剪和剪枝。减少模型层数和参数规模。
- 模型蒸馏。通过知识蒸馏方法 [22] 基于原始 BERT 模型蒸馏出符合上线要求的小模型。

在美团点评搜索 Query 意图分类任务中，我们优先尝试了模型裁剪的方案。由于搜索 Query 长度较短（通常不超过 16 个汉字），整个 Sequence 包含的语义信息有限，裁剪掉几层 Transformer 结构对模型的语义表征能力不会有太大影响，同时又能大幅减少模型参数量和推理时间。经过实验验证，在微调过程中，我们将 MT-BERT 模型裁剪为 4 层 Transfomer 结构（MT-BERT-MINI，MBM），实验效果如图 7 所示。可以发现，Query 分类场景下，裁剪后的 MBM 没有产生较大影响。由

于减少了一些不必要的参数运算，在美食和酒店两个场景下，效果还有小幅的提升。

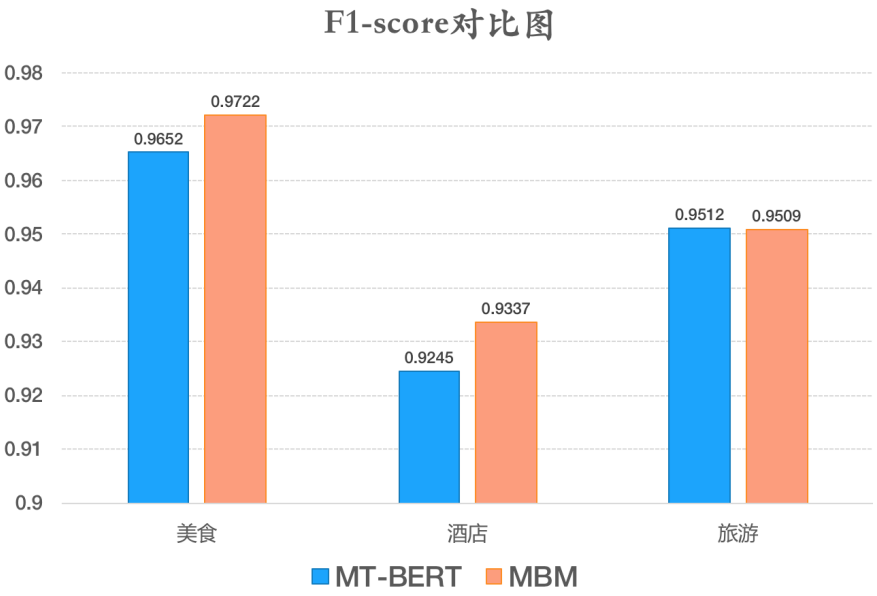


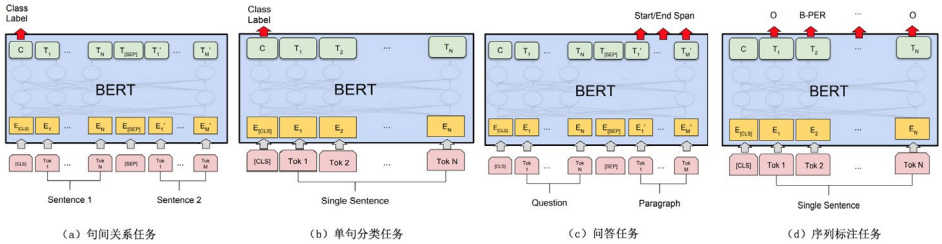
图 7　裁剪前后 MT-BERT 模型在 Query 意图分类数据集上 F1 对比

MBM 在同等压测条件下，压测服务的 TP999 达到了 12-14ms，满足搜索上线要求。除了模型裁剪，为了支持更多线上需求，我们还在进行模型蒸馏实验，蒸馏后的 6 层 MT-BERT 模型在大多数下游任务中都没有显著的效果损失。值得一提的是，BERT 模型轻量化是 BERT 相关研究的重要方向，最近 Google 公布了最新 ALBERT 模型（A Lite BERT）[23]，在减少模型参数量的同时在自然语言理解数据集 GLUE 上刷新了 SOTA。

图 8 展示了基于 BERT 模型微调可以支持的任务类型，包括句对分类、单句分类、问答（机器阅读理解）和序列标注任务。

1. 句对分类任务和单句分类任务是句子级别的任务。预训练中的 NSP 任务使得 BERT 中的“[CLS]”位置的输出包含了整个句子对（句子）的信息，我们利用其在有标注的数据上微调模型，给出预测结果。
2. 问答和序列标注任务都属于词级别的任务。预训练中的 MLM 任务使得每个

Token 位置的输出都包含了丰富的上下文语境以及 Token 本身的信息，我们对 BERT 的每个 Token 的输出都做一次分类，在有标注的数据上微调模型并给出预测。



（a）句间关系任务　（b）单句分类任务　（c）问答任务　（d）序列标注任务

图 8　BERT 微调支持的任务类型

基于 MT-BERT 的微调，我们支持了美团搜索和点评搜索的多个下游任务，包括单句分类任务、句间关系任务和序列标注任务等等。

单句分类

细粒度情感分析

美团点评作为生活服务平台，积累了大量真实用户评论。对用户评论的细粒度情感分析在深刻理解商家和用户、挖掘用户情感等方面有至关重要的价值，并且在互联网行业已有广泛应用，如个性化推荐、智能搜索、产品反馈、业务安全等领域。为了更全面更真实的描述商家各属性情况，细粒度情感分析需要判断评论文本在各个属性上的情感倾向（即正面、负面、中立）。为了优化美团点评业务场景下的细粒度情感分析效果，NLP 中心标注了包含 6 大类 20 个细粒度要素的高质量数据集，标注过程中采用严格的多人标注机制保证标注质量，并在 AI Challenger 2018 细粒度情感分析比赛中作为比赛数据集验证了效果，吸引了学术界和工业届大量队伍参赛。

针对细粒度情感分析任务，我们设计了基于 MT-BERT 的多任务分类模型，模型结构如图 9 所示。模型架构整体分为两部分：一部分是各情感维度的参数共享层（Share Layers），另一部分为各情感维度的参数独享层（Task-specific Layers）。其中参数共享层采用了 MT-BERT 预训练语言模型得到文本的上下文表征。MT-BERT 依赖其深层网络结构以及海量数据预训练，可以更好的表征上下文信息，尤其

擅长提取深层次的语义信息。 参数独享层采用多路并行的 Attention+Softmax 组合结构，对文本在各个属性上的情感倾向进行分类预测。通过 MT-BERT 优化后的细粒度情感分析模型在 Macro-F1 上取得了显著提升。

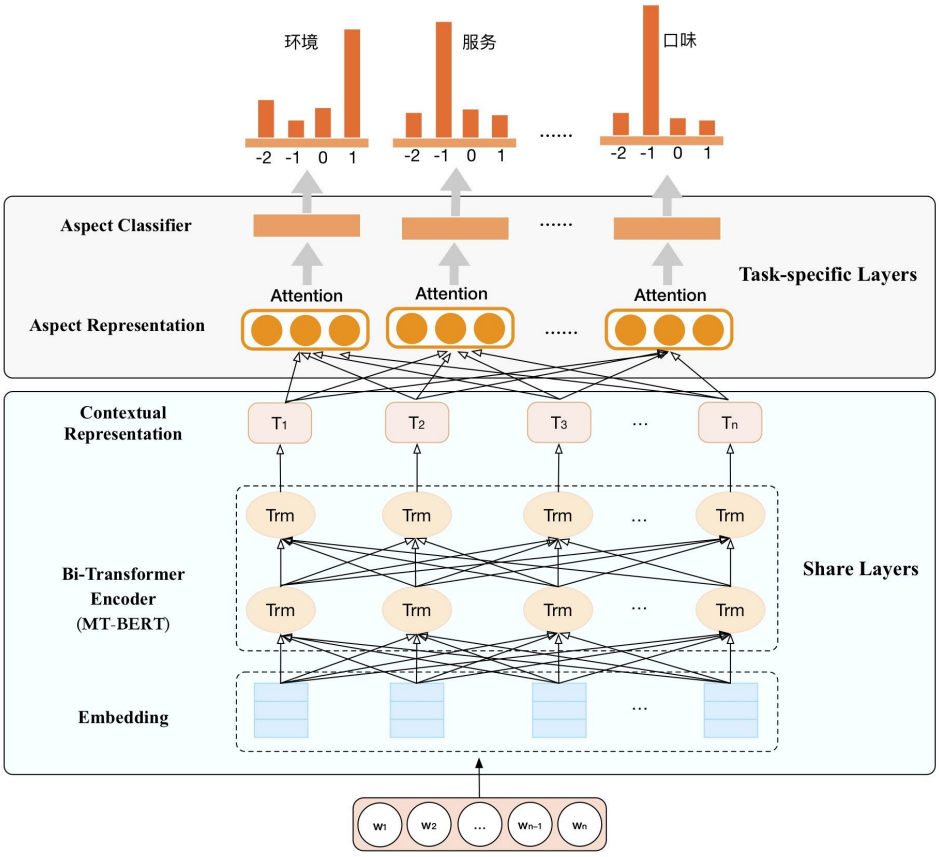


图 9　基于 MT-BERT 的多任务细粒度情感分析模型架构

细粒度情感分析的重要应用场景之一是大众点评的精选点评模块，如图 10 所示。精选点评模块作为点评 App 用户查看高质量评论的入口，其中精选点评标签承载着结构化内容聚合的作用，支撑着用户高效查找目标 UGC 内容的需求。细粒度情感分析能够从不同的维度去挖掘评论的情感倾向。基于细粒度情感分析的情感标签能够较好地帮助用户筛选查看，同时外露更多的 POI 信息，帮助用户高效的从评论中获取消费指南。

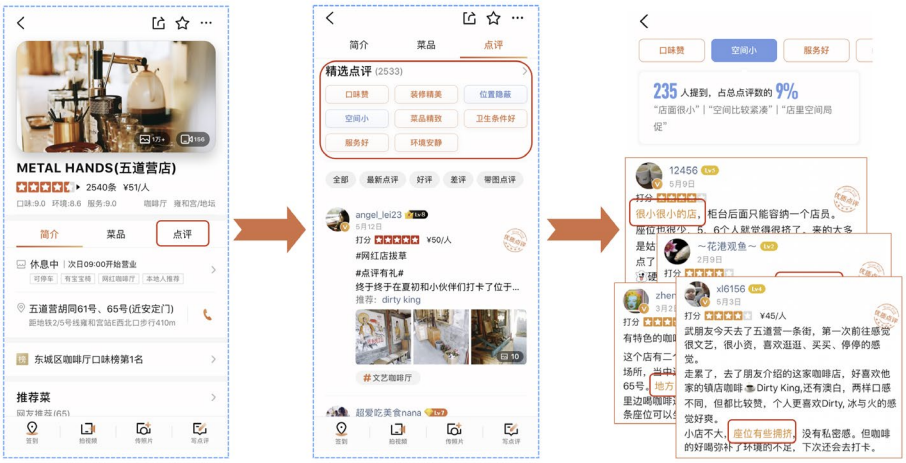
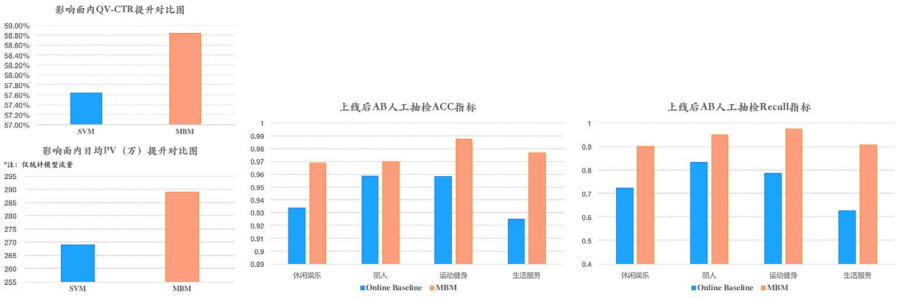


图 10　大众点评精选点评模块产品形态

Query 意图分类

在美团点评的搜索架构中，Deep Query Understanding（DQU）都是重要的前置模块之一。对于用户 Query，需要首先对用户搜索意图进行识别，如美食、酒店、演出等等。我们跟内部的团队合作，尝试了直接使用 MT-BERT 作为 Query 意图分类模型。为了保证模型在线 Inference 时间，我们使用裁剪后的 4 层 MT-BERT 模型（MT-BERT-MINI，MBM 模型）上线进行 Query 意图的在线意图识别，取得的业务效果如图 11 所示：



(a)线上收益指标（以美食频道为例）　　(b) 线下准召指标（以部分频道为例）

图 11　MBM 模型的业务效果

同时对于搜索日志中的高频 Query，我们将预测结果以词典方式上传到缓存，进一步减少模型在线预测的 QPS 压力。MBM 累计支持了美团点评搜索 17 个业务频道的 Query 意图识别模型，相比原有模型，均有显著的提升，每个频道的识别精确度都达到 95% 以上。MBM 模型上线后，提升了搜索针对 Query 文本的意图识别能力，为下游的搜索的召回、排序及展示、频道流量报表、用户认知报表、Bad Case 归因等系统提供了更好的支持。

推荐理由场景化分类

推荐理由是点评搜索智能中心数据挖掘团队基于大众点评 UGC 为每个 POI 生产的自然语言可解释性理由。对于搜索以及推荐列表展示出来的每一个商家，我们会用一句自然语言文本来突出商家的特色和卖点，从而让用户能够对展示结果有所感知，“知其然，更知其所以然”。近年来，可解释的搜索系统越来越受到关注，给用户展示商品或内容的同时透出解释性理由，正在成为业界通行做法，这样不仅能提升系统的透明度，还能提高用户对平台的信任和接受程度，进而提升用户体验效果。在美团点评的搜索推荐场景中，推荐理由有着广泛的应用场景，起到解释展示、亮点推荐、场景化承载和个性化体现的重要作用，目前已经有 46 个业务方接入了推荐理由服务。

对于不同的业务场景，对推荐理由会有不同的要求。在外卖搜索场景下，用户可能更为关注菜品和配送速度，不太关注餐馆的就餐环境和空间，这种情况下只保留符合外卖场景的推荐理由进行展示。同样地，在酒店搜索场景下，用户可能更为关注酒店特色相关的推荐理由（如交通是否方便，酒店是否近海近景区等）。

我们通过内部合作，为业务方提供符合不同场景需求的推荐理由服务。推荐理由场景化分类，即给定不同业务场景定义，为每个场景标注少量数据，我们可以基于 MT-BERT 进行单句分类微调，微调方式如图 8(b) 所示。

图 12　外卖和酒店场景下推荐理由

句间关系

句间关系任务是对两个短语或者句子之间的关系进行分类，常见句间关系任务如自然语言推理（Natural Language Inference, NLI）、语义相似度判断（Semantic Textual Similarity，STS）等。

Query 改写是在搜索引擎中对用户搜索 Query 进行同义改写，改善搜索召回结果的一种方法。在美团和点评搜索场景中，通常一个商户或者菜品会有不同的表达方式，例如“火锅”也称为“涮锅”。有时不同的词语表述相同的用户意图，例如“婚纱摄影”和“婚纱照”，“配眼镜”和“眼镜店”。Query 改写可以在不改变用户意图的情况下，尽可能多的召回满足用户意图的搜索结果，提升用户的搜索体验。为了

减少误改写，增加准确率，需要对改写后 Query 和原 Query 做语义一致性判断，只有语义一致的 Query 改写对才能上线生效。Query 语义一致性检测属于 STS 任务。我们通过 MT-BERT 微调任务来判断改写后 Query 语义是否发生漂移，微调方式如图 8(a) 所示，把原始 Query 和改写 Query 构成句子对，即“[CLS] text_a [SEP] text_b [SEP]”的形式，送入到 MT-BERT 中，通过“[CLS]”判断两个 Query 之间关系。实验证明，基于 MT-BERT 微调的方案在 Benchmark 上准确率和召回率都超过原先的 XGBoost 分类模型。

序列标注

序列标注是 NLP 基础任务之一，给定一个序列，对序列中的每个元素做一个标记，或者说给每一个元素打一个标签，如中文命名实体识别、中文分词和词性标注等任务都属于序列标注的范畴。命名实体识别（Named Entity Recognition，NER），是指识别文本中具有特定意义的实体，主要包括人名、地名、机构名、专有名词等，以及时间、数量、货币、比例数值等文字。

在美团点评业务场景下，NER 主要需求包括搜索 Query 成分分析，UGC 文本中的特定实体（标签）识别 / 抽取，以及客服对话中的槽位识别等。NLP 中心和酒店搜索算法团队合作，基于 MT-BERT 微调来优化酒店搜索 Query 成分分析任务。酒店 Query 成分分析任务中，需要识别出 Query 中城市、地标、商圈、品牌等不同成分，用于确定后续的召回策略。

在酒店搜索 Query 成分分析中，我们对标签采用“BME”编码格式，即对一个实体，第一个字需要预测成实体的开始 B，最后一个字需要预测成实体的结束 E，中间部分则为 M。以图 13 中酒店搜索 Query 成分分析为例，对于 Query“北京昆泰酒店”，成分分析模型需要将“北京”识别成地点，而“昆泰酒店”识别成 POI。MT-BERT 预测高频酒店 Query 成分后通过缓存提供线上服务，结合后续召回策略，显著提升了酒店搜索的订单转化率。

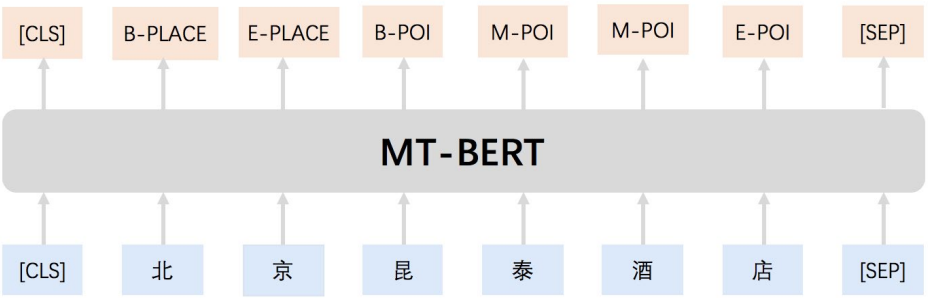


图 13　酒店 Query 的成分分析

一站式 MT-BERT 训练和推理平台建设

为了降低业务方算法同学使用 MT-BERT 门槛，我们开发了 MT-BERT 一站式训练和推理平台，一期支持短文本分类和句间关系分类两种任务，目前已在美团内部开放试用。

基于一站式平台，业务方算法同学上传业务训练数据和选择初始 MT-BERT 模型之后，可以提交微调任务，微调任务会自动分配到 AFO 集群空闲 GPU 卡上自动运行和进行效果验证，训练好的模型可以导出进行部署上线。

融入知识图谱的 MT-BERT 预训练

正如前文所述，尽管在海量无监督语料上进行预训练语言模型取得了很大的成功，但其也存在着一定的不足。BERT 模型通过在大量语料的训练可以判断一句话是否通顺，但是却不理解这句话的语义，通过将美团大脑等知识图谱中的一些结构化先验知识融入到 MT-BERT 中，使其更好地对生活服务场景进行语义建模，是需要进一步探索的方向。

MT-BERT 模型的轻量化和小型化

MT-BERT 模型在各个 NLU 任务上取得了惊人的效果，由于其复杂的网络结构和庞大的参数量，在真实工业场景下上线面临很大的挑战。如何在保持模型效果的前提下，精简模型结构和参数已经成为当前热门研究方向。我们团队在低精度量化、模型裁剪和知识蒸馏上已经做了初步尝试，但是如何针对不同的任务类型选择最合适的

模型轻量化方案，还需要进一步的研究和探索。

参考文献

[1] Peters, Matthew E., et al. “Deep contextualized word representations.” arXiv preprint arXiv:1802.05365 (2018).

[2] Howard, Jeremy, and Sebastian Ruder. “Universal language model fine-tuning for text classification.” arXiv preprint arXiv:1801.06146 (2018).

[3] Vaswani, Ashish, et al. “Attention is all you need.” Advances in neural information processing systems. 2017.

[4] Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving Language Understanding by Generative Pre-Training. Technical report, OpenAI.

[5] Devlin, Jacob, et al. “Bert: Pre-training of deep bidirectional transformers for language understanding.” arXiv preprint arXiv:1810.04805 (2018).

[6] Ming Zhou. “The Bright Future of ACL/NLP.” Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. (2019).

[7] Deng, Jia, et al. “Imagenet: A large-scale hierarchical image database.” 2009 IEEE conference on computer vision and pattern recognition. Ieee, (2009).

[8] Girshick, Ross, et al. “Rich feature hierarchies for accurate object detection and semantic segmentation.” Proceedings of the IEEE conference on computer vision and pattern recognition. 2014.

[9] Mikolov, Tomas, et al. “Distributed representations of words and phrases and their compositionality.” Advances in neural information processing systems. 2013.

[10] Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. Glove: Global vectors for word representation. In EMNLP.

[11] Oren Melamud, Jacob Goldberger, and Ido Dagan.2016. context2vec: Learning generic context embedding with bidirectional lstm. In CoNLL.

[12] Hochreiter, Sepp, and Jürgen Schmidhuber. “Long short-term memory.” Neural computation 9.8 (1997): 1735-1780.

[13] 张俊林 . 从 Word Embedding 到 BERT 模型—自然语言处理中的预训练技术发展史 . https://zhuanlan.zhihu.com/p/49271699

[14] Sebastion Ruder. “NLP’s ImageNet moment has arrived.” http://ruder.io/nlp-imagenet/. (2019)

[15] Liu, Yinhan, et al. “Roberta: A robustly optimized BERT pretraining approach.” arXiv preprint arXiv:1907.11692 (2019).

[16] 郑坤 . 使用 TensorFlow 训练 WDL 模型性能问题定位与调优 . https://tech.meituan.com/2018/04/08/tensorflow-performance-bottleneck-analysis-on-hadoop.html

[17] Uber. “Meet Horovod: Uber’s Open Source Distributed Deep Learning Framework for TensorFlow”. https://eng.uber.com/horovod/

[18] Goyal, Priya, et al. “Accurate, large minibatch sgd: Training imagenet in 1 hour.” arXiv preprint arXiv:1706.02677 (2017).

[19] Baidu. https://github.com/baidu-research/baidu-allreduce
[20] Micikevicius, Paulius, et al. “Mixed precision training.” arXiv preprint arXiv:1710.03740 (2017).
[21] 仲远，富峥等．美团餐饮娱乐知识图谱——美团大脑揭秘．https://tech.meituan.com/2018/11/22/meituan-brain-nlp-01.html
[22] Hinton, Geoffrey, Oriol Vinyals, and Jeff Dean. “Distilling the knowledge in a neural network.” arXiv preprint arXiv:1503.02531 (2015).
[23] Google. ALBERT: A Lite BERT for Self-supervised Learning of Language Representations. https://openreview.net/pdf?id=H1eA7AEtvS. (2019)

作者简介

杨扬，佳昊，礼斌，任磊，峻辰，玉昆，张欢，金刚，王超，王珺，富峥，仲远，都来自美团搜索与 NLP 部。

招聘信息

搜索与 NLP 部是美团人工智能技术研发的核心团队，致力于打造高性能、高扩展的搜索引擎和领先的自然语言处理核心技术和服务能力，依托搜索排序、NLP（自然语言处理）、Deep Learning（深度学习）、Knowledge Graph（知识图谱）等技术，处理美团海量文本数据，打通餐饮、旅行、休闲娱乐等本地生活服务各个场景数据，不断加深对用户、场景、查询和服务的理解，高效地支撑形态各样的生活服务搜索，解决搜索场景下的多意图、个性化、时效性问题，给用户极致的搜索体验，构建美团知识图谱，搭建通用 NLP Service，为美团各项业务提供智能的文本语义理解服务。我们的团队既注重 AI 技术的落地，也开展中长期的搜索、NLP 及知识图谱基础研究。目前项目及业务包括搜索引擎研发、知识图谱、智能客服、语音语义搜索、文章评论语义理解、智能助理等。

美团搜索与 NLP 部诚招智能对话算法专家、推荐算法专家、知识图谱算法专家、NLP 算法专家、数据挖掘专家，以及搜索引擎架构师、高级后台研发工程师、前端技术开发资深工程师、测试工程师、数据工程师等众多岗位，欢迎有兴趣的同学，投递简历至：tech@meituan.com（邮件标题注明：岗位名称 + 美团搜索与 NLP 部）。

深度学习在搜索业务中的探索与实践

艺涛

本文根据美团高级技术专家翟艺涛在 2018 QCon 全球软件开发大会上的演讲内容整理而成，内容有修改。

引言

2018 年 12 月 31 日，美团酒店单日入住间夜突破 200 万，再次创下行业的新纪录，而酒店搜索在其中起到了非常重要的作用。本文会首先介绍一下酒店搜索的业务特点，作为 O2O 搜索的一种，酒店搜索和传统的搜索排序相比存在很大的不同。第二部分介绍深度学习在酒店搜索 NLP 中的应用。第三部分会介绍深度排序模型在酒店搜索的演进路线，因为酒店业务的特点和历史原因，美团酒店搜索的模型演进路线可能跟大部分公司都不太一样。最后一部分是总结。

酒店搜索的业务特点

最大的连接器

美团的使命是帮大家“Eat Better，Live Better”，所做的事情就是连接人与服务。用户在美团平台可以找到他们所需要的服务，商家在美团可以售卖自己提供

的服务，而搜索在其中扮演的角色就是“连接器”。大部分用户通过美团 App 找酒店是从搜索开始的，搜索贡献了大部分的订单，是最大的流量入口。在美团首页点击“酒店住宿”图标，就会进入上图右侧的搜索入口，用户可以选择城市和入住时间并发起搜索。

查询引导

推荐

搜索&排序

酒店搜索技术团队的工作不仅有搜索排序，还有查询引导、推荐等工作，查询引导如搜索智能提示、查询纠错等。之所以还有推荐的工作，是因为很多用户在发起搜索时不带查询词，本质上属于推荐，此外还有特定场景下针对少无结果的推荐等。本文主要介绍搜索排序这方面的工作。

不同搜索对比

现在，大家对搜索都很熟悉，常见的有网页搜索，比如 Google、百度、搜狗等；商品搜索，像天猫、淘宝、京东等；还有就是 O2O（Online To Offline）的搜索，典型的就是酒店的搜索。虽然都是搜索，但是用户使用搜索的目的并不相同，包括找信息、找商品、找服务等等，不同搜索之间也存在很大的差别。

	酒店搜索（O2O）	网页搜索	商品搜索
目标	交易	相关性	交易
个性化	高	低	高
结构化数据	是	否	是
位置约束	高	低	低
供给约束	区域	无	全国

上图对不同搜索进行了简单对比，可以从 5 个维度展开。首先是目标维度。因为用户是来找信息，网页搜索重点是保证查询结果和用户意图的相关性，而在商品搜索和酒店搜索中，用户的主要目的是查找商品或服务，最终达成交易，目标上有较大区别。用户使用不同搜索的目的不同，从而导致不同搜索对个性化程度的要求不同。交易属性的搜索，包括商品搜索和酒店搜索，对个性化程度的要求都比较高，因为不同用户的消费水平不同，偏好也不一样。

在技术层面上，也存在很多不同点。网页搜索会索引全网的数据，这些数据不是它自己生产，数据来源非常多样，包括新闻、下载页、视频页、音乐页等各种不同的形态，所以整个数据是非结构化的，差异也很大。这意味着网页搜索需要拥有两种技术能力，数据抓取能力和数据解析能力，它们需要抓取网页并解析形成结构化数据。在这个层面上，酒店搜索和商品搜索相对就“幸福”一些，因为数据都是商家提交的结构化数据，相对来说更加规范。

此外，酒店作为一种 O2O 的服务，用户在线上（Online）下单，最终需要到线下（Offline）去消费，所以就有一个位置上的约束，而位置的约束也就导致出现供给侧的约束，供给只能在某个特定位置附近。比如北京大学方圆几公里之内的酒店。这两点约束在网页搜索和商品搜索中就不用考虑，网页可以无限次的进行阅读。商品搜索得益于快递业的快速发展，在北京也可以买到来自浙江的商品，供给侧的约束比较小。

介绍完不同搜索产品的特点，接下来看不同搜索产品的优化目标。通用搜索的优化目标是相关性，评价指标是 DCG、NDCG、MAP 等这些指标，要求查询结果和用户意图相关。对商品搜索来说，不同电商平台的优化目标不太一样，有的目标是最大化 GMV，有的目标是最大化点击率，这些在技术上都可以实现。

而对酒店搜索而言，因为它属于 O2O 的业务形态，线上下单，线下消费，这就要求搜索结果必须和用户的查询意图“强相关”。这个“强相关”包括两层含义，显性相关和隐性相关。举个例子，用户搜索“北京大学”，那么他的诉求很明确，就是要找“北京大学”附近的酒店，这种属于用户明确告诉平台自己的位置诉求。但是，如果用户在本地搜索“七天”，即使用户没有明确说明酒店的具体位置，我们也知道，用户可能想找的是距离自己比较近的“七天酒店”，这时候就需要建模用户的隐性位置诉求。

美团是一个交易平台，大部分用户使用美团是为了达成交易，所以要优化用户的购买体验。刻画用户购买体验的核心业务指标是访购率，用来描述用户在美团是否顺畅的完成了购买，需要优化访购率这个指标。总结一下，酒店搜索不仅要解决相关性，尽量优化用户购买体验、优化访购率等指标，同时还要照顾到业务诉求。

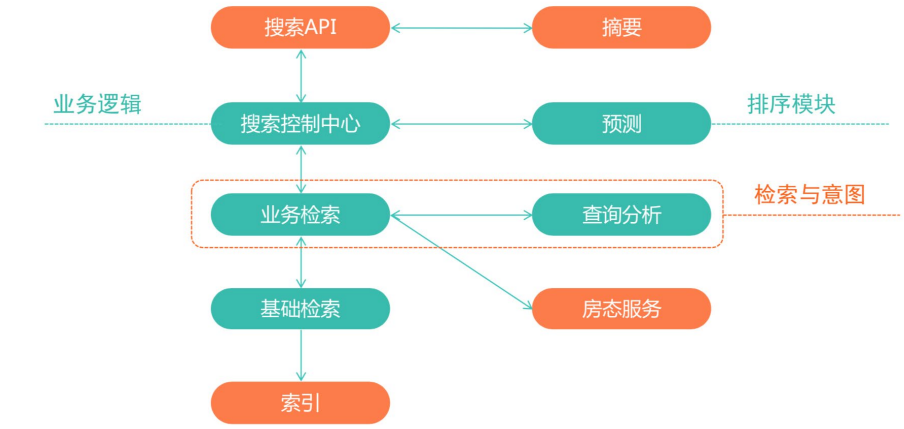
根据上面的分析，酒店搜索的整个搜索框架就可以拆分成三大模块：检索、排序以及业务规则。检索层包括查询理解和召回两部分，主要解决相关性问题。查询理解做的事情就是理解用户意图，召回根据用户意图来召回相关的酒店，两者强耦合，需要放在一起。检索的核心是语义理解，比如用户搜索“北京大学”，平台就知道用户想找的是“北京大学附近的酒店”，所以这个模块的优化方式是问题驱动，不断地发现问题、解决问题来进行迭代。

接下来，从检索模块检索出来的酒店都已经是满足用户需求的酒店了。还是上面“北京大学”的那个例子，检索模块已经检索出来几百家“北京大学”附近的酒店，这些都是和用户的查询词“北京大学”相关的，怎么把用户最有可能购买的酒店排到前面呢？这就是排序模块要做的事情。

排序模块使用机器学习和深度学习的技术提供“千人千面”的排序结果，如果是经常预定经济连锁型酒店的用户，排序模块就把经济连锁型酒店排到前面。针对消费水平比较高，对酒店要求比较高的用户，排序模块就把高档酒店排到前面，对每个用户都可以做到个性化定制。排序属于典型的技术驱动模块，优化目标是访购率，用这个技术指标驱动技术团队不断进行迭代和优化。

最后是业务层面，比如有些商家会在美团上刷单作弊，针对这些商家需要做降权处理。

整体框架

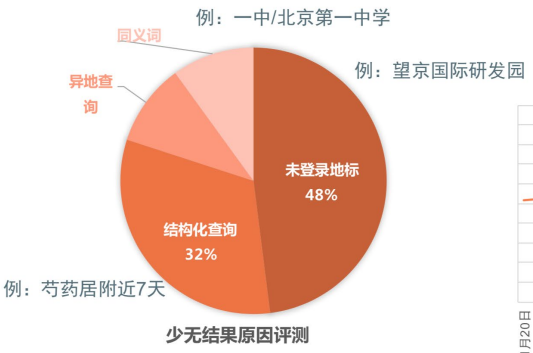


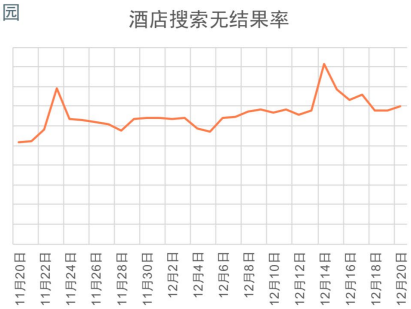
上图是搜索的整体框架，这里详细描述下调用过程：

- 搜索 API 负责接收用户的查询词并发送给搜索控制中心。
- 控制中心把接收到的查询请求发送到检索与意图模块，搜索词会先经过查询分析模块做用户的查询意图分析，分析完之后，会把用户的查询意图分析结果传回去给业务检索模块，业务检索模块根据意图识别结果形成查询条件，然后去基础检索端查询结果。
- 基础检索访问索引得到查询结果后，再把结果返回给上层。
- 业务检索模块获取基础的检索结果后，会调用一些外部服务如房态服务过滤一些满房的酒店，再把结果返回给控制中心。
- 此时，控制中心得到的都是和用户查询意图强相关的结果，这时就需要利用机器学习技术做排序。通过预测模块对每个酒店做访购率预测，控制中心获取预测模块的排序结果后，再根据业务逻辑做一些调整，最终返回结果给搜索 API。

可以看到，模块划分和前文描述的思想一致，检索模块主要解决用户意图识别和召回问题，也就是解决相关性。预测模块做访购率预测，业务逻辑放在搜索控制中心实现。接下来会介绍一下意图理解和排序模块中涉及的一些深度学习技术。

查询理解：问题

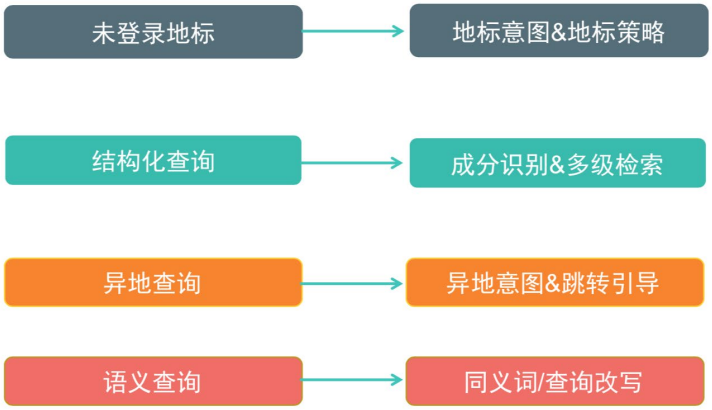




先来看下查询理解的问题，这个模块通过数据分析和 Case 分析，不断的发现问题、解决问题来迭代优化。之前的评测发现少无结果的原因，主要包括以下几种：

- 地标词：比如用户搜索“望京国际研发园”，但是后台没有一家酒店包含“望京国际研发园”这几个字，其实用户想找的是望京国际研发园附近的酒店。
- 结构化查询：比如芍药居附近 7 天，酒店描述信息中没有“附近”这个词，搜索体验就比较差。这种需要对查询词做成分识别，丢掉不重要的词，并且对不用类别的 Term 走不同的检索域。
- 异地查询：用户在北京搜索“大雁塔”没有结果，其实用户的真实意图是西安大雁塔附近的酒店，这种需要做异地需求识别并进行异地跳转。
- 同义词：在北京搜索“一中”和搜索“北京第一中学”，其实都是同一个意思，需要挖掘同义词。

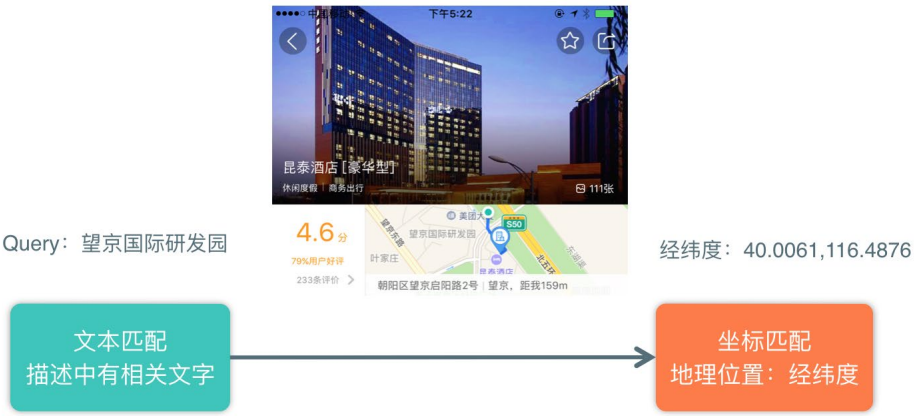
解决方案



针对这几类问题，我们分别作了以下工作：

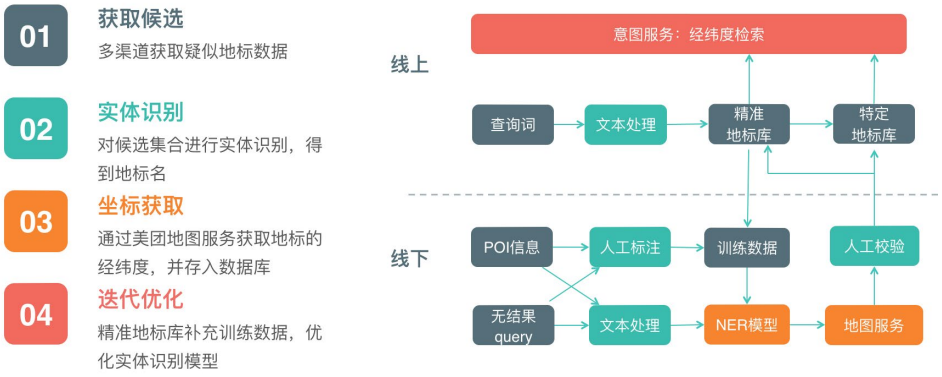
- 针对地标词问题，提供地标意图识别和地标策略，把地标类别的查询词改成按经纬度进行画圈检索。
- 针对结构化查询的问题，我们对查询词做了成分识别，设计了少无结果时的多级检索架构。
- 针对异地查询的问题，做异地意图识别和异地的跳转引导。
- 针对语义查询的问题，做同义词和查询改写。

这里的每一个模块都用到了机器学习和深度学习的技术，本文挑选两个酒店搜索中比较特殊的问题进行介绍。



地标问题是 O2O 搜索的一个典型问题，在网页搜索和商品搜索中都较少出现此类问题。当用户搜索类似“望京国际研发园”这种查询词的时候，因为搜索的相关性是根据文本计算的，需要酒店描述中有相关文字，如果酒店的描述信息中没有这个词，那就检索不出来。比如昆泰酒店，虽然就在望京国际研发园旁边，但是它的描述信息中并没有出现“望京国际研发园”，所以就无法检索出来，这会导致用户体验较差。

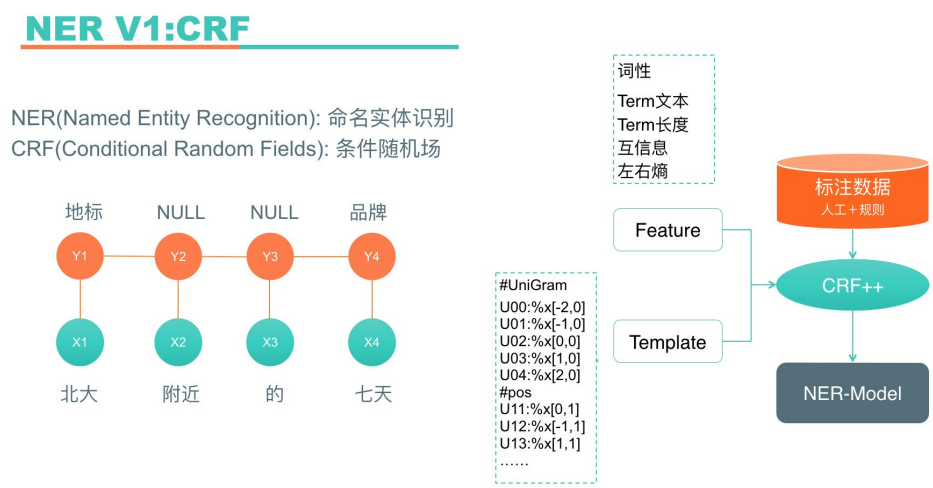
经过分析，我们发现有一类查询词是针对特定地点的搜索，用户的诉求是找特定地点附近的酒店，这种情况下走文本匹配大概率是没有结果的。这个问题的解法是针对这种类型的查询词，从“文本匹配”改成“坐标匹配”，首先分析查询词是不是有地标意图，如果是的话就不走文本匹配了，改走坐标匹配，检索出来这个坐标附近的酒店就可以了。这时就产生了两个问题：第一，怎么确定哪些查询词有地标意图；第二，怎么获取经纬度信息。



针对这个问题，我们做了地标策略，步骤如下：

- 多渠道获取可能包含地标词的候选集，这些候选集包括用户少无结果的查询词，以及一些酒店提供的描述信息。
- 对候选集合进行命名实体识别（NER，Named Entity Recognition），可以得到各个命名实体的类型，标识为“地标”类型的就是疑似地标词。
- 把疑似地标词放到美团地图服务中获取经纬度，经过人工校验无误后，存入线上数据库中；线上来查询请求时，先会去匹配精准地标库，如果匹配成功，说明这个查询词是地标意图，这时就不走文本检索了，直接在意图服务层走经纬度检索。
- 经过人工校验的精准地标库补充到 NER 模型的训练数据中，持续优化 NER 模型。

这里提到了 NER 模型，下面对它做一下详细的介绍。

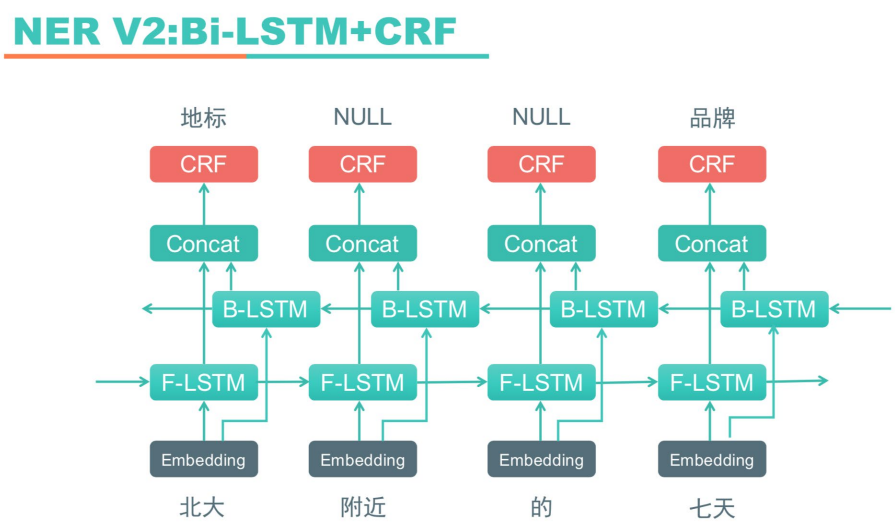


John Lafferty et al. Conditional random fields: Probabilistic models for segmenting and labeling sequence data.ICML2001

NER 是命名实体识别，是机器学习中的序列标注问题，比如输入“北大附近的七天”，就会标注出来每个词的成分，这里“北大”是地标，“七天”是酒店品牌。这里的类别是根据业务特点自己定义的，酒店业务中有地标、品牌、商圈等不同的类别。与分类问题相比，序列标注问题中当前的预测标签不仅与当前的输入特征相关，还与前后的预测标签相关，即预测标签序列之间有强相互依赖关系。

解决序列标注问题的经典模型是 CRF（Conditional Random Field，条件随机场），也是我们刚开始尝试的模型。条件随机场可以看做是逻辑回归的序列化版本，逻辑回归是用于分类的对数线性模型，条件随机场是用于序列化标注的对数线性模型，可以看做是考虑了上下文的分类模型。

机器学习问题的求解就是“数据 + 模型 + 特征”，数据方面先根据业务特点定义了几种实体类别，然后通过“人工 + 规则”的方法标注了一批数据。特征方面提取了包括词性、Term 文本特征等，还定义了一些特征模板，特征模板是 CRF 中人工定义的一些二值函数，通过这些二值函数，可以挖掘命名实体内部以及上下文的构成特点。标注数据、模型、特征都有了，就可以训练 CRF 模型，这是线上 NER 问题的第一版模型。

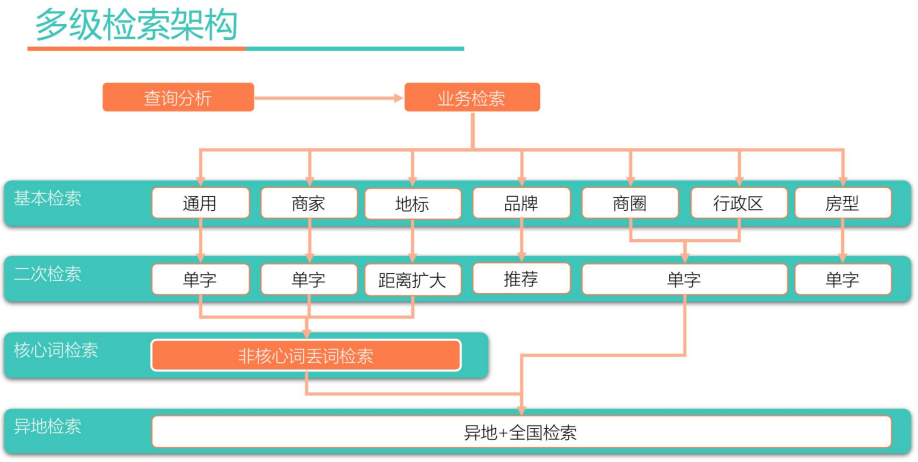


Guillaume Lample et al Neural architectures for named entity recognition. NAACL2016

随着深度学习的发展，用 Word Embedding 词向量作为输入，叠加神经网络单元的方法渐渐成为 NLP 领域新的研究方向。基于双向 LSTM（Long Short-Term Memory）+CRF 的方法成为 NER 的主流方法，这种方法采用双向 LSTM 单元作为特征提取器替代原有的人工特征，不需要专门的领域知识，框架也通用。Embedding 输入也有多种形式，可以是词向量，可以是字向量，也可以是字向量和词向量的拼接。

我们尝试了双向 LSTM+CRF，并在实际应用中做了些改动：由于在 CRF 阶段已经积累了一批人工特征，实验发现把这些特征加上效果更好。加了人工特征的双向 LSTM+CRF 是酒店搜索 NER 问题的主模型。

当然，针对 LSTM+CRF 的方法已经有了很多的改进，比如还有一种 NER 的方法是融合 CNN+LSTM+CRF，主要改进点是多了一个 CNN 模块来提取字级别的特征。CNN 的输入是字级别的 Embedding，通过卷积和池化等操作来提取字级别的特征，然后和词的 Embedding 拼接起来放入 LSTM。这种方法在两个公开数据集上面取得了最好的结果，也是未来尝试的方向之一。



为了解决少无结果的问题，我们设计了多级检索架构，如上图所示，主要分 4 个层次：基本检索、二次检索、核心词检索和异地检索。

- 基本检索会根据查询词的意图选择特定的检索策略，比如地标意图走经纬度检索，品牌意图只检索品牌域和商家名。
- 基本检索少无结果会进行二次检索，二次检索也是分意图的，不同意图类型会有不同的检索策略，地标意图是经纬度检索的，二次检索的时候就需要扩大检索半径；品牌意图的查询词，因为很多品牌在一些城市没有开店，比如香格里拉在很多小城市并没有开店，这时比较好的做法，是推荐给用户该城市最好的酒店。
- 如果还是少无结果，会走核心词检索，只保留核心词检索一遍。丢掉非核心词有多种方式，一种是删除一些运营定义的无意义词，一种是保留 NER 模型识别出来的主要实体类型。此外还有一个 TermWeight 的模型，对每个词都有一个重要性的权重，可以把一些不重要的词丢掉。
- 在还没有结果的情况下，会选择”异地 + 全国“检索，即更换城市或者在全国范围内进行检索。

多级检索架构上线后，线上的无结果率就大幅度降低了。

排序

排序其实是一个典型的技术问题，业界应用比较广泛的有广告排序和推荐排序，广告排序比如 Google 和百度的关键字广告排序，今日头条、腾讯的展示广告排序。推荐排序比如快手、抖音这些短视频平台，以及各大 App、浏览器的信息流。广告排序和推荐排序优化的目标都是点击率，技术栈也比较相似，包括 LR/FTRL、FM/FFM、GBDT、DNN 等模型。

跟以上两种排序应用相比，酒店排序有自己的业务特点，因为美团酒店具有 LBS 属性和交易属性，天生自带很多连续特征，如酒店价格、酒店评分、酒店离用户的距离等，这些连续特征是决定用户购买行为的最重要因素。优化目标也不一样，大部分场景下酒店搜索的优化目标是访购率，部分场景下优化目标是点击率。在技术层面，酒店排序整体的技术栈和广告、推荐比较相似，都可以使用 LR/FTRL、FM/FFM、GBDT、DNN 等模型。

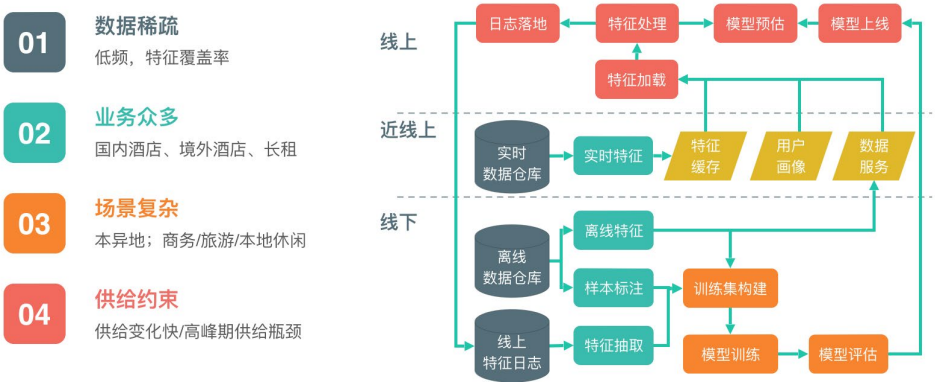
面临的挑战

具体到酒店排序工作，我们面临一些不一样的挑战，主要包括以下 4 点：

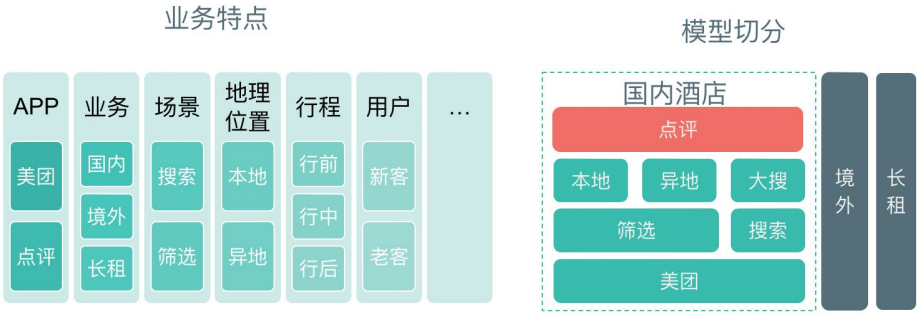
1. 数据稀疏。住酒店本身是一种低频行为，大部分用户一年也就住一两次，导致很多特征的覆盖率比较低。
2. 业务众多。美团酒店包括国内酒店业务、境外酒店业务，以及长租、钟点房

等业务，同时有美团和点评两个不同的 App。

3. 场景复杂。按照用户的位置可以分成本地和异地，按照用户的诉求可以分成商务、旅游、本地休闲等几大类，这些用户之间差异很明显。比如商务用户会有大量复购行为，典型例子是美团员工的出差场景，美团在上海和北京各有一个总部，如果美团的同学去上海出差，大概率会在公司差旅标准内选一家离公司近的酒店，从而会在同一家酒店产生大量的复购行为；但是如果是一个旅游用户，他就很少反复去同一个地方。
4. 供给约束。酒店行业供给的变化很快，一个酒店只有那么多房间，一天能提供的间夜量是固定的，全部订出的话，用户提价也不会提供新的房间，这种情况在劳动节、国庆这种节假日特别明显。



上图右侧是排序的整体架构图，分为线下、线上和近线上三个部分。在线下部分，主要做离线的模型调优和评估，线上部分做预测。这里比较特别的是近线上部分，我们在实时层面做了大量的工作，包括用户的实时行为、酒店实时价格、实时库存等等，以应对供给变化快的特点。



这里介绍一个业务特点导致的比较独特的问题：**模型切分**。美团酒店有很多业务场景，包括国内酒店、境外酒店、长租、钟点房等；还有两个 App，美团 App 和大众点评 App；还有搜索和筛选两种场景，搜索带查询词，筛选没有查询词，两种场景差异较大；从地理位置维度，还可以分成本地和异地两种场景。

面对这么多的业务场景，第一个问题就是模型怎么设计，是用统一的大模型，还是分成很多不同的小模型？我们可以用一个大模型 Cover 所有的场景，用特征来区分不同场景的差异，好处是统一模型维护和优化成本低。也可以划分很多小模型，这里有一个比较好的比喻，多个专科专家会诊，胜过一个全科医生。切分模型后，可以避免差异较大的业务之间互相影响，也方便对特殊场景进行专门的优化。

在模型切分上，主要考虑三个因素：

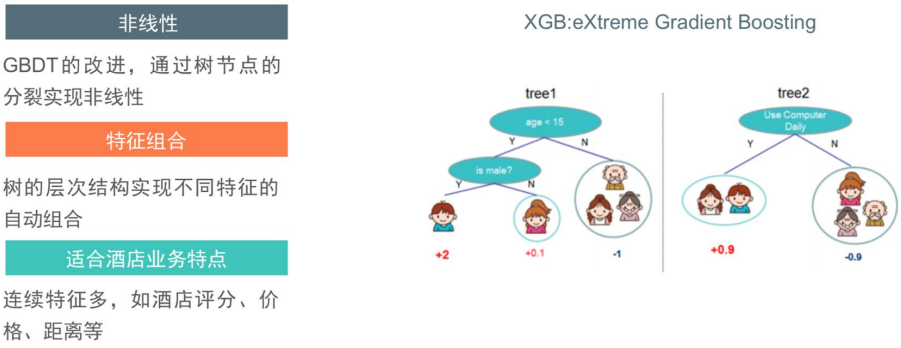
- 第一，业务之间的差异性。比如长租和境外差异很大，国内酒店和境外业务差异也很大，这种需要拆分。
- 第二，细分后的数据量。场景分的越细，数据量就越小，会导致两个问题，一是特征的覆盖率进一步降低；二是数据量变小后，不利于后续的模型迭代，一些复杂模型对数据量有很高的要求。我们做过尝试，国内酒店场景下，美团和大众点评两个 App 数据量都很大，而且用户也很不一样，所以做了模型拆分；但是境外酒店，因为本身是新业务数据量较小，就没有再进行细分。
- 第三，一切以线上指标为准。我们会做大量的实验，看当前数据量下怎么拆分效果更好，比如美团 App 的国内酒店，我们发现把搜索和筛选拆开后，效果

更好；筛选因为数据量特别大，拆分成本、异地效果也更好，但是如果搜索场景拆分成本地、异地模型就没有额外收益了。最终，一切都要以线上的实际表现为准。

模型演进

接下来介绍一下排序模型的演进过程，因为业务特点及历史原因，酒店搜索的排序模型走了一条不一样的演进路线。大家可以看业界其他公司点击率模型的演进，很多都是从 LR/FTRL 开始，然后进化到 FM/FFM，或者用 GBDT+LR 搞定特征组合，然后开始 Wide&Deep。

酒店搜索的演进就不太一样。酒店业务天生自带大量连续特征，如酒店价格、酒店和用户的距离、酒店评分等，因此初始阶段使用了对连续特征比较友好的树模型。在探索深度排序模型的时候，因为已经有了大量优化过的连续特征，导致我们的整个思路也不太一样，主要是借鉴一些模型的思想，结合业务特点做尝试，下面逐一进行介绍。



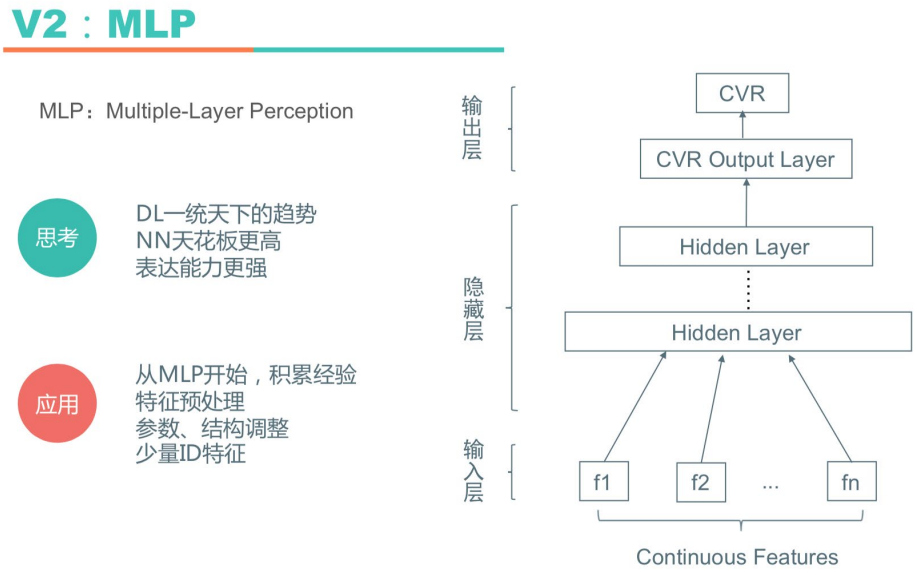
T Chen, C Guestrin. Xgboost: A scalable tree boosting system. KDD2016

初始阶段线上使用的模型是 XGB(XGBoost, eXtreme Gradient Boosting)。作为 GBDT 的改进，XGB 实现了非线性和自动的特征组合。树节点的分裂其实就实现了非线性，树的层次结构实现了不同特征的自动组合，而且树模型对特征的包容性非常好，树的分裂通过判断相对大小来实现，不需要对特征做特殊处理，适合连续特征。

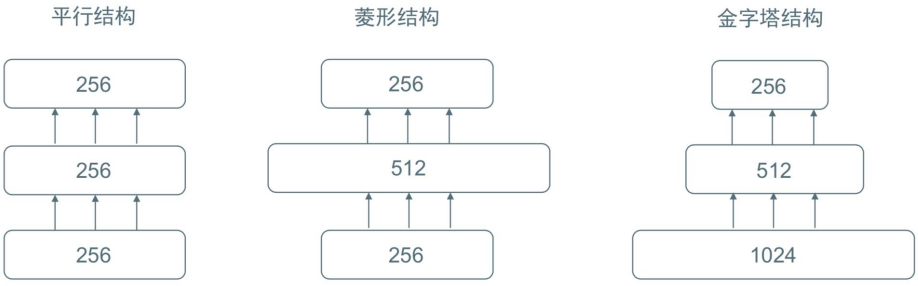
树模型的这些特点确实很适合酒店这种连续特征多的场景，至今为止，XGB 都是数据量较小场景下的主模型。但是树模型优化到后期遇到了瓶颈，比如特征工程收益变小、增大数据量没有额外收益等，此外树模型不适合做在线学习的问题愈发严重。酒店用户在劳动节、国庆节等节假日行为有较大不同，这时需要快速更新模型，我们尝试过只更新最后几棵树的做法，效果不佳。考虑到未来进一步的业务发展，有必要做模型升级。

模型探索的原则是从简单到复杂，逐步积累经验，所以首先尝试了结构比较简单的 MLP(Multiple-Layer Perception) 多层感知机，也就是全连接神经网络。神经网络是一种比树模型“天花板”更高的模型，“天花板”更高两层意思：第一层意思，可以优化提升的空间更大，比如可以进行在线学习，可以做多目标学习；第二层意思，模型的容量更大，“胃口”更大，可以“吃下”更多数据。此外它的表达能力也更强，可以拟合任何函数，网络结构和参数可以调整的空间也更大。但是它的优点同时也是它的缺点，因为它的网络结构、参数等可以调整的空间更大，神经网需要做很

多的参数和网络结构层面的调整。



上图是 MLP 的网络结构图，包含输入层、若干个隐藏层、输出层。在很长一段时间内，在特征相同的情况下，MLP 效果不如 XGB，所以有段时间线上使用的是 XGB 和 MLP 的融合模型。后来经过大量的网络结构调整和参数调整，调参经验越来越丰富，MLP 才逐步超越 XGB。这里额外说明一下，酒店搜索中有少量的 ID 类特征，在第一版 MLP 里 ID 类特征是直接当做连续特征处理的。比如城市 ID，ID 的序关系有一定的物理意义，大城市 ID 普遍较小，小城市开城晚一些，ID 较大。

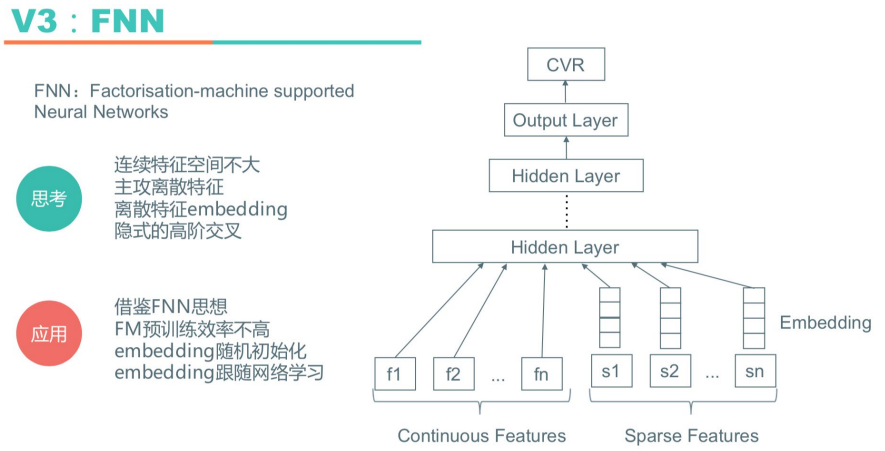


在 MLP 阶段我们对网络结构做了大量实验，尝试过三种网络结构：平行结构、

菱形结构、金字塔结构。在很多论文中提到三者相比平行结构效果最好，但是因为酒店搜索的数据不太一样，实验发现金字塔结构效果最好，即上图最右边的“1024-512-256”的网络结构。同时还实验了不同网络层数对效果的影响，实验发现 3-6 层的网络效果较好，更深的网络没有额外收益而且线上响应时间会变慢，后面各种模型探索都是基于 3 到 6 层的金字塔网络结构进行尝试。

MLP 上线之后，我们开始思考接下来的探索方向。在树模型阶段，酒店搜索组就在连续特征上做了很多探索，连续特征方面很难有比较大的提升空间；同时业界的研究重点也放在离散特征方面，所以离散特征应该是下一步的重点方向。

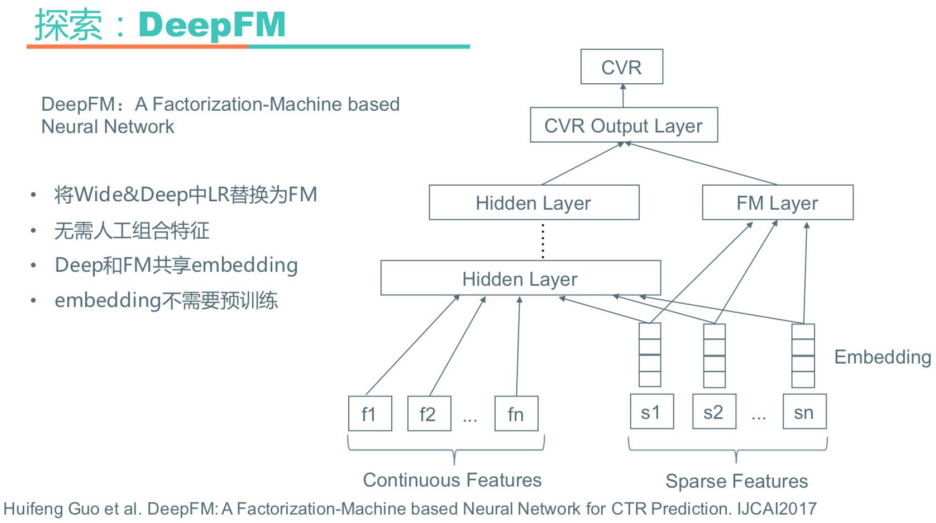
深度排序模型对离散特征的处理有两大类方法，一类是对离散特征做 Embedding，这样离散特征就可以表示成连续的向量放到神经网络中去，另一类是 Wide&Deep，把离散特征直接加到 Wide 侧。我们先尝试了第一种，即对离散特征做 Embedding 的方法，借鉴的是 FNN 的思想。其实离散特征做 Embedding 的想法很早就出现了，FM 就是把离散特征表示成 K 维向量，通过把高维离散特征表示成低维向量增加模型泛化能力。



Weinan Zhang et al. Deep Learning over Multi-Field Categorical Data: A Case Study on User Response Prediction. ECIR 2016

实际使用中，我们稍微做了一些改动，实验中发现使用 FM 预训练的效率不高，所以尝试了不做预训练直接把 Embedding 随机初始化，然后让 Embedding 跟随网络一起学习，实验结果发现比 FM 预训练效果还要好一点。最后的做法是没有用 FM

做预训练，让 Embedding 随机初始化并随网络学习，上图是线上的 V3 模型。



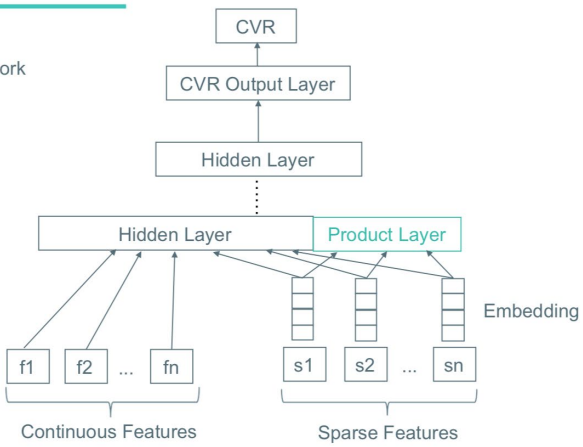
FNN 的成功上线证明离散特征 Embedding 这个方向值得深挖，所以我们接着实验了 DeepFM。DeepFM 相对于 Wide&Deep 的改进，非常类似于 FM 相对 LR 的改进，都认为 LR 部分的人工组合特征是个耗时耗力的事情，而 FM 模块可以通过向量内积的方式直接求出二阶组合特征。DeepFM 使用 FM 替换了 Wide&Deep 中的 LR，离散特征的 Embedding 同时“喂”给神经网和 FM，这部分 Embedding 是共享的，Embedding 在网络的优化过程中自动学习，不需要做预训练，同时 FM Layer 包含了一阶特征和二阶的组合特征，表达能力更强。我们尝试了 DeepFM，线下有提升线上波动提升，并没有达到上线的标准，最终没有全量。

尽管 DeepFM 没有成功上线，但这并没有动摇我们对 Embedding 的信心，接下来尝试了 PNN。PNN 的网络重点在 Product 上面，在点击率预估中，认为特征之间的关系更多是一种 And“且”的关系，而非 Add“加”的关系，例如性别为男且用华为手机的人，他定酒店时属于商务出行场景的概率更高。

探索：PNN

PNN：Product-based Neural Network

- 特征之间‘and’关系
- 显式的2阶特征组合
- 隐式的高阶特征组合
- 内积和外积两种方式



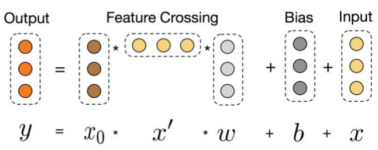
Yanru Qu et al. Product-based neural networks for user response prediction. ICDM2016

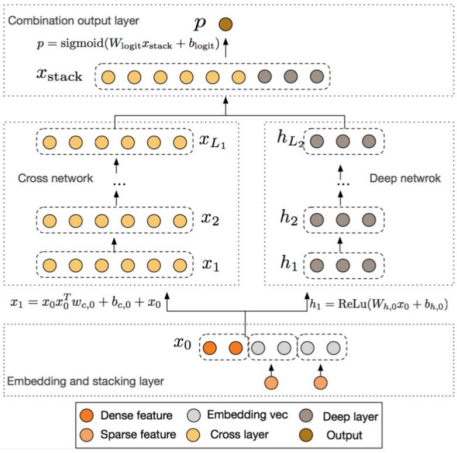
PNN 使用了 Product Layer 进行显式的二阶特征组合。上图右边是 PNN 的网络结构图，依然对离散特征做 Embedding，Embedding 向量同时送往隐层和 Product 层，Product 通过内积或者外积的方式，对特征做显式的二阶交叉，之后再送入神经网的隐层，这样可以做到显式的二阶组合和隐式的高阶特征组合。特征交叉基于乘法的运算实现，有两种方式：内积和外积。我们尝试了内积的方式，线下略有提升线上也是波动提升，没有达到上线标准，所以最终也没有全量上线。

探索：DCN

DCN：Deep & Cross network

- 引入Cross network
- 显式高阶特征交叉
- Cross网络和Deep网络并行
- 显示交叉&隐式交叉
- “元素”级交叉





Ruoxi Wang, et al. Deep & Cross Network for Ad Click Predictions. ADKDD2017

PNN 之后我们认为 Embedding 还可以再尝试一下，于是又尝试了 DCN（Deep&Cross Network）。DCN 引入了一个 Cross Network 进行显式的高阶特征交叉。上图右边是论文中的图，可以看到 Deep&Cross 中用了两种网络，Deep 网络和 Cross 网络，两种网络并行，输入都一样，在最后一层再 Stack 到一起。

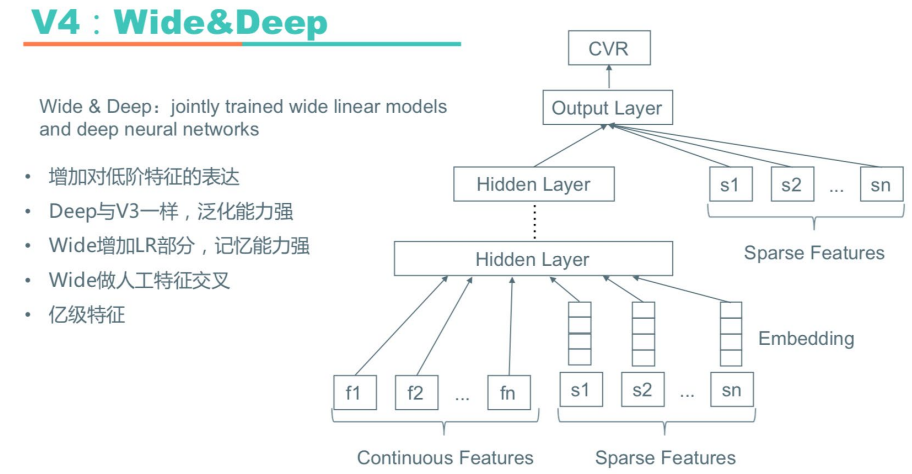
Deep 网络和前面几种网络一样，包括连续特征和离散特征的 Embedding，Cross 网络是 DCN 的特色，在 Cross 网络里面，通过巧妙的设计实现了特征之间的显式高阶交叉。看上图左下角的 Cross 结构示意，这里的 x 是每一层的输入，也就是上一层的输出。Feature Crossing 部分包括了原始输入 x0、本层输入 x 的转置、权重 w 三项，三项相乘其实就做了本层输入和原始输入的特征交叉，x1 就包含了二阶的交叉信息，x2 就包含了三阶的交叉信息，就可以通过控制 Cross 的层数显式控制交叉的阶数。

不得不说，DCN 在理论上很漂亮，我们也尝试了一下。但是很可惜，线下有提升线上波动提升，依然未能达到上线的标准，最终未能全量上线。

经过 DeepFM、PNN、DCN 的洗礼，促使我们开始反思，为什么在学术上特别有效的模型，反而在酒店搜索场景下不能全量上线呢？它们在线下都有提升，在线上也有提升，但是线上提升较小且有波动。

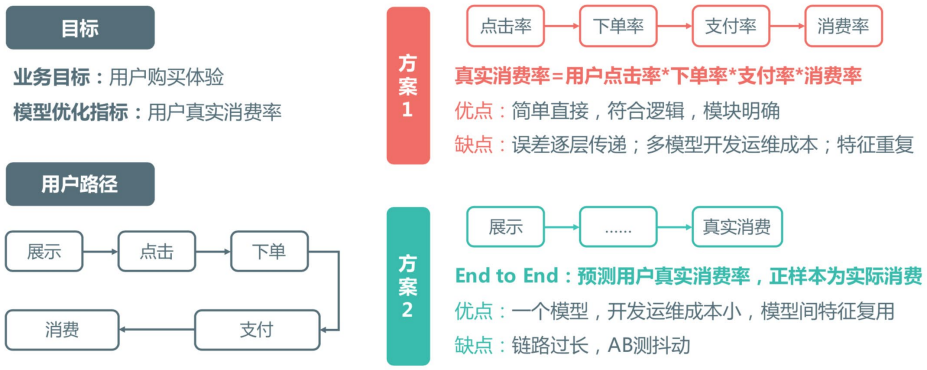
经过认真分析我们发现可能有两个原因：第一，连续特征的影响，XGB 时代尝试了 600 多种连续特征，实际线上使用的连续特征接近 400 种，这部分特征太强了；第二，离散特征太少，离散特征只有百万级别，但是 Embedding 特别适合离散特征多的情况。接下来方向就很明确了：补离散特征的课。

最终，我们还是把目光转回 Wide&Deep。Wide&Deep 同时训练一个 Wide 侧的线性模型和一个 Deep 侧的神经网络，Wide 部分提供了记忆能力，关注用户有过的历史行为，Deep 部分提供了泛化能力，关注一些没有历史行为的 Item。之前的工作主要集中在 Deep 测，对低阶特征的表达存在缺失，所以我们添加了 LR 模块以增加对低阶特征的表达，Deep 部分和之前的 V3 一样。刚开始只用了少量的 ID 类特征，效果一般，后来加了大量人工的交叉特征，特征维度达到了亿级别后效果才得到很好的提升。下图是我们的 V4 模型：



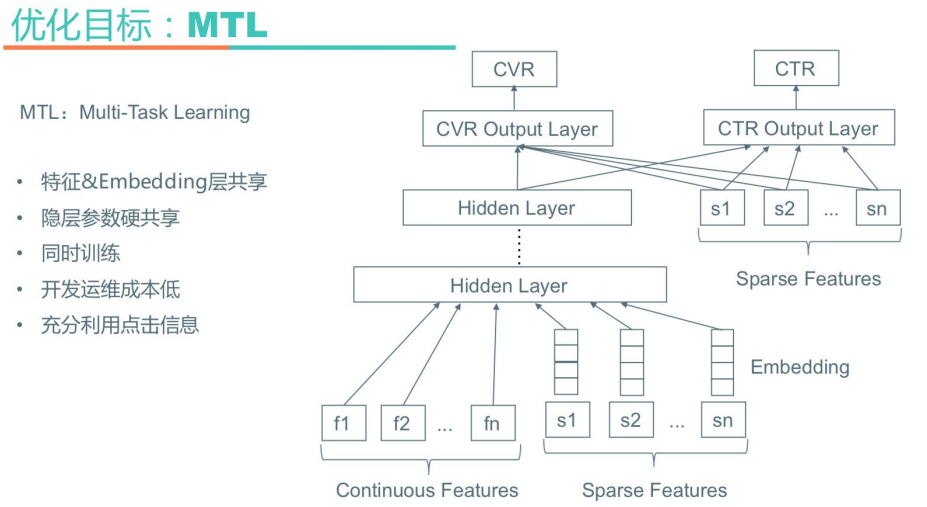
Heng-Tze Cheng et al. 2016. Wide & deep learning for recommender systems. 2016

接下来介绍一下优化目标的迭代过程（后面讲 MTL 会涉及这部分内容）。酒店搜索的业务目标是优化用户的购买体验，模型的优化指标是用户的真实消费率，怎么优化这个目标呢？通过分析用户的行为路径可以把用户的行为拆解成“展示 -> 点击 -> 下单 -> 支付 -> 消费”等 5 个环节，这其中每个环节都可能存在用户流失，比如有些用户支付完成后，因为部分商家确认比较慢，用户等不及就取消了。



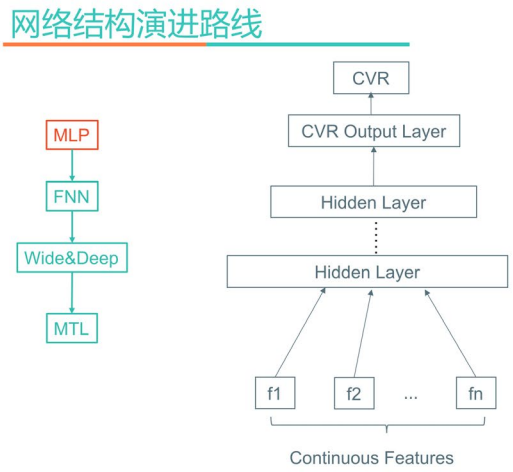
刚开始我们采用了方案 1，对每一个环节建模（真实消费率 = 用户点击率 × 下单率 × 支付率 × 消费率）。优点是非常简单直接且符合逻辑，每个模块分工明确，容易确认问题出在哪里。缺点也很明显，首先是特征重复，4 个模型在用户维度和商

家维度的特征全部一样，其次模型之间是相乘关系且层数过多，容易导致误差逐层传递，此外 4 个模型也增加了运维成本。后来慢慢进化到了方案 2 的“End to End”方式，直接预测用户的真实消费率，这时只需要把正样本设定为实际消费的样本，一个模型就够了，开发和运维成本较小，模型间特征也可以复用，缺点就是链路比较长，上线时经常遇到 AB 测抖动问题。



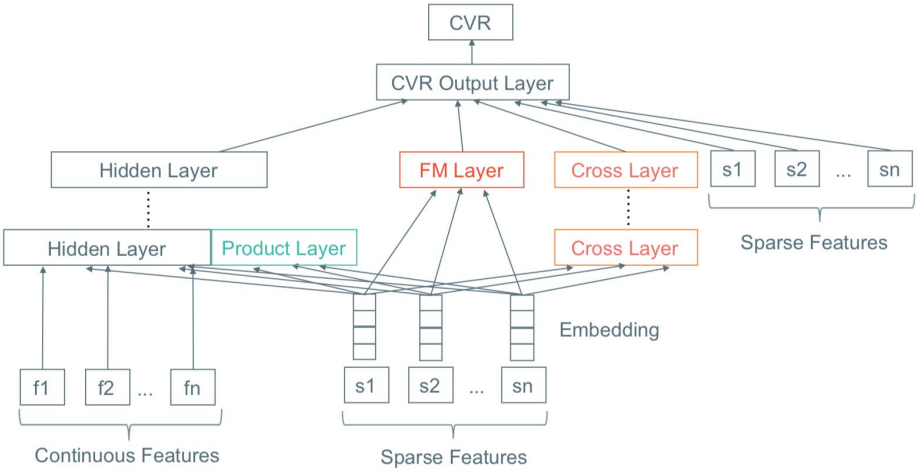
模型切换到神经网络后就可以做多任务学习了，之前树模型时代只预测“End to End”真实访购率，神经网络则可以通过多任务学习同时预测 CTR 展示点击率和 CVR 点击消费率。多任务学习通过硬共享的方式同时训练两个网络，特征、Embedding 层、隐层参数都是共享的，只在输出层区分不同的任务。上图是酒店搜索当前线上的模型，基于 Wide&Deep 做的多任务学习。

网络结构演进路线



上图是酒店搜索排序的深度排序模型演进路线，从 MLP 开始，通过对离散特征做 Embedding 进化到 FNN，中间尝试过 DeepFM、PNN、DCN 等模型，后来加入了 Wide 层进化到 Wide&Deep，现在的版本是一个 MTL 版的 Wide&Deep，每个模块都是累加上去的。

除了上面提到的模型，我们还探索过这个：



这是我们自己设计的混合网络，它融合了 FNN、DeepFM、PNN、DCN、Wide&Deep 等不同网络的优点，同时实现了一阶特征、显式二阶特征组合、显式高阶特征组合、隐式高阶特征组合等，有兴趣的同学可以尝试一下。

不同模型实验结果

不同模型实验结果

模型	离线AUC	线上效果（访购率）
XGBoost	Baseline	Baseline
MLP	+18BP	+0.34%
MLP+XGBoost	+32BP	+0.79%
FNN	+33BP	+0.73%
DeepFM	+39BP	波动提升，未全量
Deep&Cross	+24BP	波动提升，未全量
Wide&Deep	+29BP	无线上实验
Wide&Deep（组合特征）	+54BP	+1.26%

上图是不同模型的实验结果，这里的 BP 是基点（Basis Point），1BP=0.01%。XGB 是 Baseline，MLP 经过很长时间的调试才超过 XGB，MLP 和 XGB 融合模型的效果也很好，不过为了方便维护，最终还是用 FNN 替换了融合模型。Wide&Deep 在开始阶段，提升并没有特别多，后来加了组合特征后效果才好起来。我们 Embedding 上面的尝试，包括 DeepFM、Deep&Cross 等，线下都有提升，线上波动有提升，但是未能达到上线的标准，最终未能全量。

连续特征

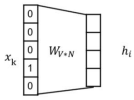
累积分布归一化：$x' = \int_{-\infty}^{x} f(x)dx$

标准化：$x' = \frac{x-\mu}{\sigma}$

根号、对号变换：$x' = \sqrt{x}$　$x' = log_2^x$

离散特征

ID-Embedding：



ID-Combine：$\emptyset_k(x) = \Pi x_i^{c_{ki}}, c_{ki} \in (0,1)$

缺失值

缺失值参数化：$x' = w_{miss} \cdot g(x) + w_{hit} \cdot f(x)$

$g(x) = \begin{cases} 0, x_{正常} \\ 1, x_{缺失} \end{cases}, f(x) = \begin{cases} x, x_{正常} \\ 0, x_{缺失} \end{cases}$

处理方法	AUC
标准化	Baseline
累积分布归一化	+15BP
标准化+累积分布归一化	+26BP
根号+对数等手工变化	+7BP
ID-COMBINE	+25BP
ID-Embedding	+23BP
缺失值处理	+27BP

在特征预处理方面对连续特征尝试了累计分布归一化、标准化，以及手工变换如根号变换、对数变换等；累积分布归一化其实就是做特征分桶，因为连续特征多且分布范围很广，累积分布归一化对酒店搜索的场景比较有效。

离散特征方面尝试了特征 Embedding 及离散特征交叉组合，分别对应 FNN 和 Wide&Deep。这里特别提一下缺失值参数化，因为酒店业务是一种低频业务，特征覆盖率低，大量样本存在特征缺失的情况，如果对缺失特征学一个权重，非缺失值学一个权重效果较好。

激活函数

ReLU　$f(x) = \max(0, x)$

Sigmoid　$f(x) = \frac{1}{1+e^{-x}}$

Leaky_ReLU　$f(x) = \begin{cases} x & , x \geq 0 \\ 0.01x & , x < 0 \end{cases}$

ELU　$f(x) = \begin{cases} x & , x > 0 \\ \alpha(e^x - 1) & , x \leq 0 \end{cases}$

优化器

Adagrad

RMSProp

Adam

$g_t = \nabla_{\theta_{t-1}} f(\theta_{t-1})$

$\Delta\theta_t = -\eta * g_t$

SGD

其他

BN/Dropout/学习率/隐层

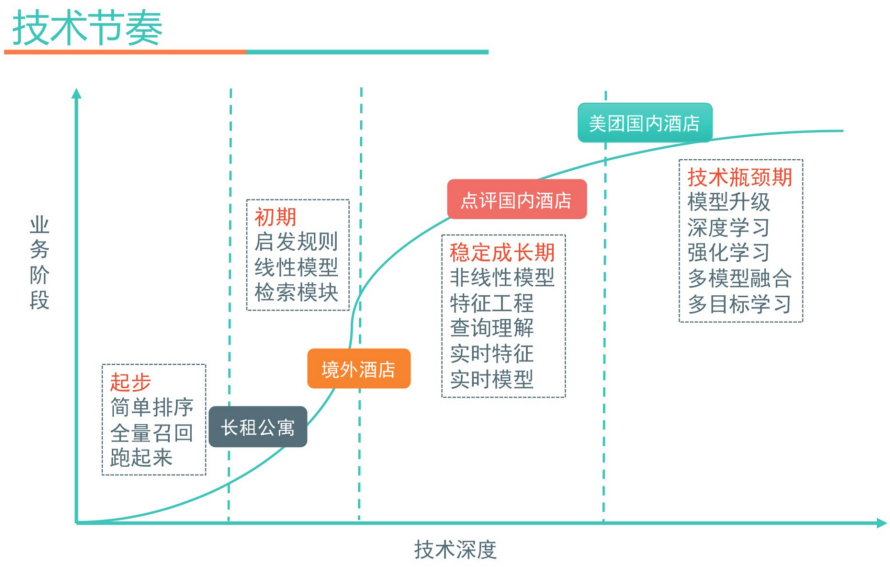
激活函数	优化器	AUC
Sigmoid	Adagrad	-300BP
ReLU	Adagrad	Baseline
ELU	Adagrad	-2BP
LeakyReLU	Adagrad	+5BP
ReLU	RMSProp	+10BP
ReLU	Adam	+15BP

参数调优方面分别尝试了激活函数、优化器等。激活函数尝试过 Sigmoid、

ReLU、Leaky_ReLU、ELU 等；优化器也实验过 Adagrad、Rmsprop、Adam 等；从实验效果看，激活函数 ReLU+Adam 效果最好。刚开始时，加了 Batch Normalization 层和 Dropout 层，后来发现去掉后效果更好，可能和酒店搜索的数据量及数据特点有关。网络结构和隐层数方面用的是 3 到 6 层的金字塔网络。学习率方面的经验是学习率小点比较好，但是会导致训练变慢，需要找到一个平衡点。

下面介绍深度排序模型线上 Serving 架构的演化过程，初始阶段组内同学各自探索，用过各种开源工具如 Keras、TensorFlow 等，线上分别自己实现，预测代码和其他代码都放一起，维护困难且无法复用。

后来组内决定一起探索，大家统一使用 TensorFlow，线上用 TF-Serving，线上线下可以做到无缝衔接，预测代码和特征模块也解耦了。现在则全面转向 MLX 平台，MLX 是美团自研的超大规模机器学习平台，专为搜索、推荐、广告等排序问题定制，支持百亿级特征和流式更新，有完善的线上 Serving 架构，极大地解放了算法同学的生产力。



最后介绍一下我们对搜索排序技术节奏的一些理解，简单来说就是在不同阶段做不同的事情。

在上图中，横轴表示技术深度，越往右技术难度越大，人力投入越大，对人的要求也越高。纵轴是业务阶段。业务阶段对技术的影响包括两方面，数据量和业务价值。数据量的大小，可以决定该做什么事情，因为有些技术在数据量小的时候意义不大；业务价值就更不用说了，业务价值越大越值得“重兵投入”。

- 起步阶段：起步阶段，还没有数据，这时候做简单排序就好，比如纯按价格排序或者距离排序，目的是让整个流程快速地跑起来，能提供最基本的服务。比如 2017 年，美团的长租业务当时就处于起步阶段。
- 业务初期：随着业务的发展，就进入了业务发展初期，订单数慢慢增长，也有了一些数据，这时候可以增加一些启发式规则或者简单的线性模型，检索模型也可以加上。但是由于数据量还比较小，没必要部署很复杂的模型。
- 稳定成长期：业务进一步发展后，就进入了稳定成长期，这时候订单量已经很大了，数据量也非常大了，这段时间是“补课”的时候，可以把意图理解的模块加上，排序模型也会进化到非线性模型比如 XGB，会做大量的特征工程，

实时特征以及实时模型，在这个阶段特征工程收益巨大。

- 技术瓶颈期：这个阶段的特点是基本的东西都已经做完了，在原有的技术框架下效果提升变的困难。这时需要做升级，比如将传统语义模型升级成深度语义模型，开始尝试深度排序模型，并且开始探索强化学习、多模型融合、多目标学习等。

中国有句俗话叫“杀鸡焉用牛刀”，比喻办小事情，何必花费大力气，也就是不要小题大做。其实做技术也一样，不同业务阶段不同数据量适合用不同的技术方案，没有必要过度追求先进的技术和高大上的模型，根据业务特点和业务阶段选择最匹配的技术方案才是最好的。我们认为，**没有最好的模型，只有合适的场景**。

总结

酒店搜索作为 O2O 搜索的一种，和传统的搜索排序相比有很多不同之处，既要解决搜索的相关性问题，又要提供“千人千面”的排序结果，优化用户购买体验，还要满足业务需求。通过合理的模块划分可以把这三大类问题解耦，检索、排序、业务三个技术模块各司其职。在检索和意图理解层面，我们做了地标策略、NER 模型和多级检索架构来保证查询结果的相关性；排序模型上结合酒店搜索的业务特点，借鉴业界先进思想，尝试了多种不同的深度排序模型，走出了一条不一样的模型演进路线。同时通过控制技术节奏，整体把握不同业务的技术选型和迭代节奏，对不同阶段的业务匹配不同的技术方案，只选对的，不选贵的。

参考文献

[1] John Lafferty et al. Conditional random fields: Probabilistic models for segmenting and labeling sequence data.ICML2001.
[2] Guillaume Lample et al Neural architectures for named entity recognition. NAACL2016.
[3] Zhiheng Huang, Wei Xu, and Kai Yu. 2015.
[4] Bidirectional LSTM-CRF models for sequence tagging. arXiv preprint arXiv:1508.01991.
[5] Xuezhe Ma et al.End-to-end Sequence Labeling via Bi-directional LSTM-CNNs-CRF.ACL2016.

[6] T Chen, C Guestrin. XGBoost: A scalable tree boosting system. KDD2016.
[7] Weinan Zhang et al. Deep Learning over Multi-Field Categorical Data: A Case Study on User Response Prediction. ECIR 2016.
[8] Huifeng Guo et al. DeepFM: A Factorization-Machine based Neural Network for CTR Prediction. IJCAI2017.
[9] Yanru Qu et al. Product-based neural networks for user response prediction. ICDM2016.
[10] Heng-Tze Cheng et al. 2016. Wide & deep learning for recommender systems. 2016.In Proceedings of the 1st Workshop on Deep Learning for Recommender Systems.
[11] Ruoxi Wang et al. Deep & Cross Network for Ad Click Predictions. ADKDD2017.

作者简介

艺涛，美团高级技术专家，2016 年加入美团，现负责美团酒店业务搜索排序技术。2010 年毕业于中科院计算所，曾在网易有道等公司工作，先后从事网页搜索、购物搜索、计算广告等方向的研发工作。曾荣获“Kaggle 卫星图像分类大赛”亚军，QCon 明星讲师。

大众点评搜索基于知识图谱的深度学习排序实践

非易　祝升　汤彪　张弓　仲远

1. 引言

挑战与思路

搜索是大众点评 App 上用户进行信息查找的最大入口，是连接用户和信息的重要纽带。而用户搜索的方式和场景非常多样，并且由于对接业务种类多，流量差异大，为大众点评搜索（下文简称点评搜索）带来了巨大的挑战，具体体现在如下几个方面：

1. **意图多样**：用户查找的信息类型和方式多样。信息类型包括 POI、榜单、UGC、攻略、达人等。以找店为例，查找方式包括按距离、按热度、按菜品和按地理位置等多种方式。例如用户按照品牌进行搜索时，大概率是需要寻找距离最近或者常去的某家分店；但用户搜索菜品时，会对菜品推荐人数更加敏感，而距离因素会弱化。
2. **业务多样**：不同业务之间，用户的使用频率、选择难度以及业务诉求均不一样。例如家装场景用户使用频次很低，行为非常稀疏，距离因素弱，并且选择周期可能会很长；而美食多为即时消费场景，用户行为数据多，距离敏感。
3. **用户类型多样**：不同的用户对价格、距离、口味以及偏好的类目之间差异很大；搜索需要能深度挖掘到用户的各种偏好，实现定制化的“千人千面”的搜索。
4. **LBS 的搜索**：相比电商和通用搜索，LBS 的升维效应极大地增加了搜索场景的复杂性。例如对于旅游用户和常驻地用户来说，前者在搜索美食的时候可能会更加关心当地的知名特色商户，而对于距离相对不敏感。

上述的各项特性，叠加上时间、空间、场景等维度，使得点评搜索面临比通

用搜索引擎更加独特的挑战。而解决这些挑战的方法，就需要升级 NLP（Natural Language Processing，自然语言处理）技术，进行深度查询理解以及深度评价分析，并依赖知识图谱技术和深度学习技术对搜索架构进行整体升级。在美团 NLP 中心以及大众点评搜索智能中心两个团队的紧密合作之下，经过短短半年时间，点评搜索核心 KPI 在高位基础上仍然大幅提升，是过去一年半涨幅的六倍之多，提前半年完成全年目标。

基于知识图谱的搜索架构重塑

美团 NLP 中心正在构建全世界最大的餐饮娱乐知识图谱——美团大脑（相关信息请参见《美团大脑：知识图谱的建模方法及其应用》）。它充分挖掘关联各个场景数据，用 NLP 技术让机器“阅读”用户公开评论，理解用户在菜品、价格、服务、环境等方面的喜好，构建人、店、商品、场景之间的知识关联，从而形成一个“知识大脑”[1]。通过将知识图谱信息加入到搜索各个流程中，我们对点评搜索的整体架构进行了升级重塑，图 1 为点评搜索基于知识图谱搭建的 5 层搜索架构。本篇文章是“美团大脑”系列文章第二篇（系列首篇文章请参见《美团餐饮娱乐知识图谱——美团大脑揭秘》），主要介绍点评搜索 5 层架构中核心排序层的演变过程，文章主要分为如下 3 个部分：

1. 核心排序从传统机器学习模型到大规模深度学习模型的演进。
2. 搜索场景深度学习排序模型的特征工程实践。
3. 适用于搜索场景的深度学习 Listwise 排序算法——LambdaDNN。

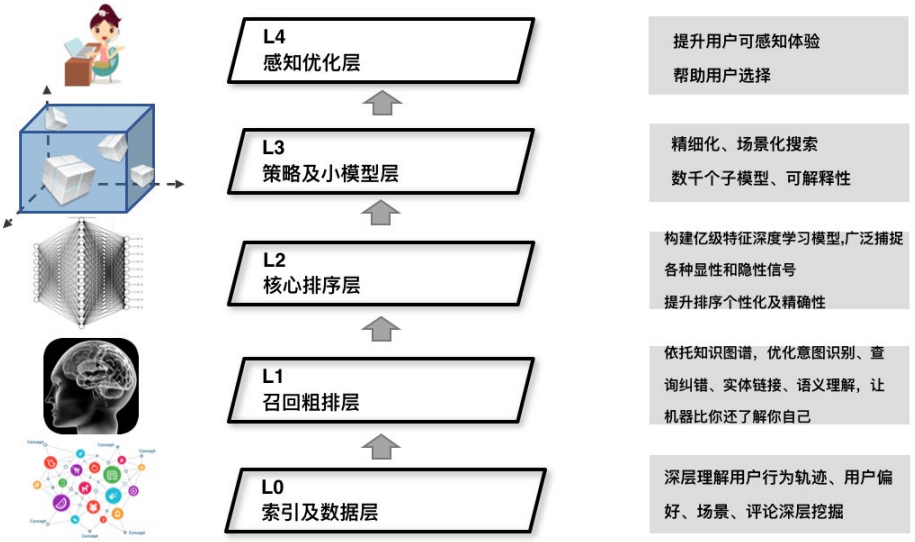


图 1　基于知识图谱的点评搜索 5 层架构

2. 排序模型探索与实践

搜索排序问题在机器学习领域有一个单独的分支，Learning to Rank（L2R）。主要分类如下：

1. 根据样本生成方法和 Loss Function 的不同，L2R 可以分为 Pointwise、Pairwise、Listwise。
2. 按照模型结构划分，可以分为线性排序模型、树模型、深度学习模型，他们之间的组合（GBDT+LR，Deep&Wide 等）。

在排序模型方面，点评搜索也经历了业界比较普遍的迭代过程：从早期的线性模型 LR，到引入自动二阶交叉特征的 FM 和 FFM，到非线性树模型 GBDT 和 GBDT+LR，到最近全面迁移至大规模深度学习排序模型。下面先简单介绍下传统机器学习模型（LR、FM、GBDT）的应用和优缺点，然后详细介绍深度模型的探索实践过程。

传统机器学习模型

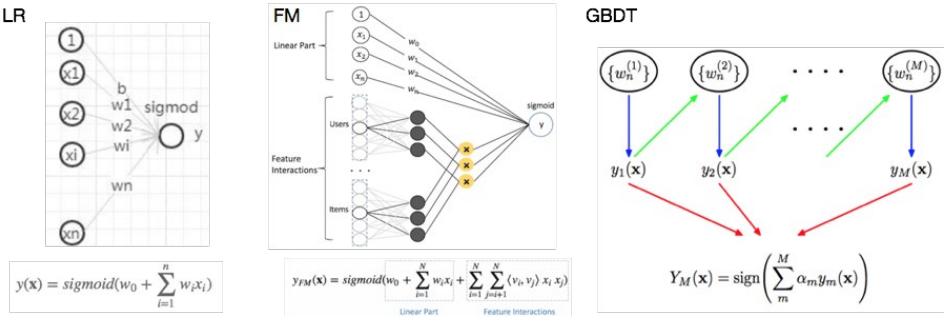


图 2　几种传统机器学习模型结构

1. LR 可以视作单层单节点的线性网络结构。模型优点是可解释性强。通常而言，良好的解释性是工业界应用实践比较注重的一个指标，它意味着更好的可控性，同时也能指导工程师去分析问题优化模型。但是 LR 需要依赖大量的人工特征挖掘投入，有限的特征组合自然无法提供较强的表达能力。
2. FM 可以看做是在 LR 的基础上增加了一部分二阶交叉项。引入自动的交叉特征有助于减少人工挖掘的投入，同时增加模型的非线性，捕捉更多信息。FM 能够自动学习两两特征间的关系，但更高量级的特征交叉仍然无法满足。
3. GBDT 是一个 Boosting 的模型，通过组合多个弱模型逐步拟合残差得到一个强模型。树模型具有天然的优势，能够很好的挖掘组合高阶统计特征，兼具较优的可解释性。GBDT 的主要缺陷是依赖连续型的统计特征，对于高维度稀疏特征、时间序列特征不能很好的处理。

深度神经网络模型

随着业务的发展，在传统模型上取得指标收益变得愈发困难。同时业务的复杂性要求我们引入海量用户历史数据，超大规模知识图谱特征等多维度信息源，以实现精准个性化的排序。因此我们从 2018 年下半年开始，全力推进 L2 核心排序层的主模型迁移至深度学习排序模型。深度模型优势体现在如下几个方面：

1. **强大的模型拟合能力**：深度学习网络包含多个隐藏层和隐藏结点，配合上非线

性的激活函数，理论上可以拟合任何函数，因此十分适用于点评搜索这种复杂的场景。

2. **强大的特征表征和泛化能力**：深度学习模型可以处理很多传统模型无法处理的特征。例如深度网络可以直接中从海量训练样本中学习到高维稀疏 ID 的隐含信息，并通过 Embedding 的方式去表征；另外对于文本、序列特征以及图像特征，深度网络均有对应的结构或者单元去处理。
3. **自动组合和发现特征的能力**：华为提出的 DeepFM，以及 Google 提出的 DeepCrossNetwork 可以自动进行特征组合，代替大量人工组合特征的工作。

下图是我们基于 Google 提出的 Wide&Deep 模型搭建的网络结构 [2]。其中 Wide 部分输入的是 LR、GBDT 阶段常用的一些细粒度统计特征。通过较长周期统计的高频行为特征，能够提供很好的记忆能力。Deep 部分通过深层的神经网络学习 Low-Order、高纬度稀疏的 Categorical 型特征，拟合样本中的长尾部分，发现新的特征组合，提高模型的泛化能力。同时对于文本、头图等传统机器学习模型难以刻画的特征，我们可以通过 End-to-End 的方式，利用相应的子网络模型进行预处理表示，然后进行融合学习。

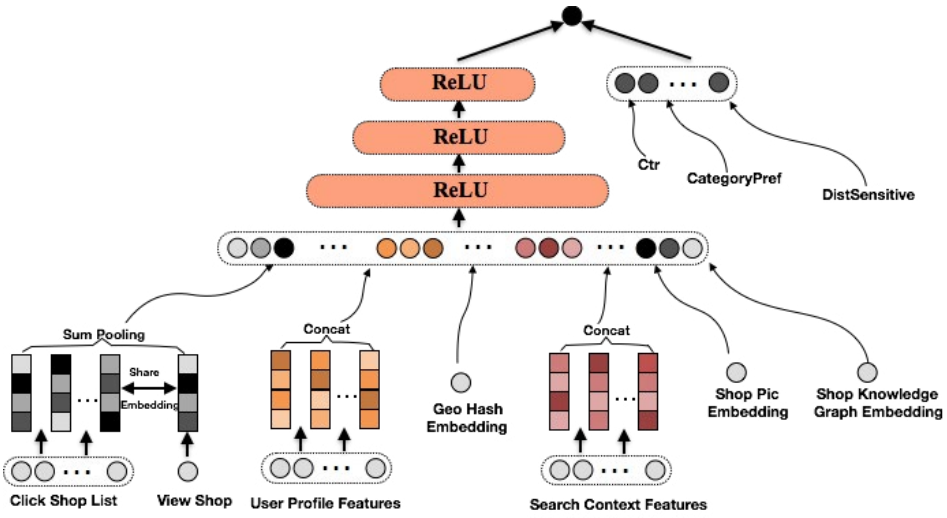


图 3　Deep&Wide 模型结构图

3. 搜索深度排序模型的特征工程实践

深度学习的横空出世，将算法工程师从很多人工挖掘和组合特征的事情中解放出来。甚至有一种论调，专做特征工程的算法工程师可能面临着失业的风险。但是深度学习的自动特征学习目前主要集中体现在 CV 领域，CV 领域的特征数据是图片的像素点——稠密的低阶特征，深度学习通过卷积层这个强力工具，可以自动对低阶特征进行组合和变换，相比之前人工定义的图像特征从效果上来说确实更加显著。在 NLP 领域因为 Transformer 的出现，在自动特征挖掘上也有了长足的进步，BERT 利用 Transformer 在多个 NLP Task 中取得了 State-of-The-Art 的效果。

但是对于 CTR 预估和排序学习的领域，目前深度学习尚未在自动特征挖掘上对人工特征工程形成碾压之势，因此人工特征工程依然很重要。当然，深度学习在特征工程上与传统模型的特征工程也存在着一些区别，我们的工作主要集中在如下几个方面。

3.1 特征预处理

- **特征归一化**：深度网络的学习几乎都是基于反向传播，而此类梯度优化的方法对于特征的尺度非常敏感。因此，需要对特征进行归一化或者标准化以促使模型更好的收敛。
- **特征离散化**：工业界一般很少直接使用连续值作为特征，而是将特征离散化后再输入到模型中。一方面因为离散化特征对于异常值具有更好的鲁棒性，其次可以为特征引入非线性的能力。并且，离散化可以更好的进行 Embedding，我们主要使用如下两种离散化方法：
  - 等频分桶：按样本频率进行等频切分，缺失值可以选择给一个默认桶值或者单独设置分桶。
  - 树模型分桶：等频离散化的方式在特征分布特别不均匀的时候效果往往不好。此时可以利用单特征结合 Label 训练树模型，以树的分叉点做为切分值，相应的叶子节点作为桶号。
- **特征组合**：基于业务场景对基础特征进行组合，形成更丰富的行为表征，为模型提供先验信息，可加速模型的收敛速度。典型示例如下：

- 用户性别与类目之间的交叉特征，能够刻画出不同性别的用户在类目上的偏好差异，比如男性用户可能会较少关注“丽人”相关的商户。
- 时间与类目之间的交叉特征，能够刻画出不同类目商户在时间上的差异，例如，酒吧在夜间会更容易被点击。

3.2 万物皆可 Embedding

深度学习最大的魅力在于其强大的特征表征能力，在点评搜索场景下，我们有海量的用户行为数据，有丰富的商户 UGC 信息以及美团大脑提供的多维度细粒度标签数据。我们利用深度学习将这些信息 Embedding 到多个向量空间中，通过 Embedding 去表征用户的个性化偏好和商户的精准画像。同时向量化的 Embedding 也便于深度模型进一步的泛化、组合以及进行相似度的计算。

3.2.1 用户行为序列的 Embedding

用户行为序列（搜索词序列、点击商户序列、筛选行为序列）包含了用户丰富的偏好信息。例如用户筛选了“距离优先”时，我们能够知道当前用户很有可能是一个即时消费的场景，并且对距离较为敏感。行为序列特征一般有如下图所示的三种接入方式：

- **Pooling**：序列 Embedding 后接入 Sum/Average Pooling 层。此方式接入成本低，但忽略了行为的时序关系。
- **RNN**：LSTM/GRU 接入，利用循环网络进行聚合。此方式能够考虑行为序列的时序关系；代价是增大了模型复杂度，影响线上预测性能。
- **Attention**：序列 Embedding 后引入 Attention 机制，表现为加权的 Sum Pooling；相比 LSTM/GRU 计算开销更低 [4]。

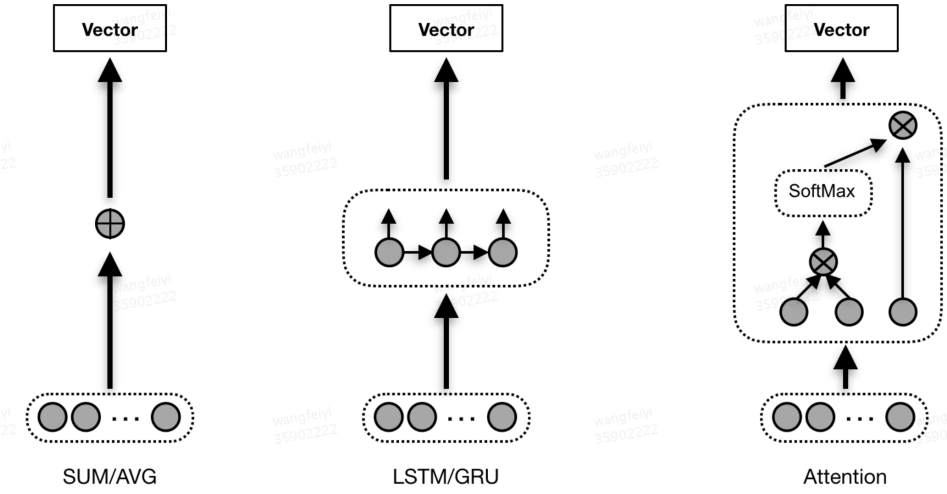


图 4　行为序列特征接入的几种方法

同时，为了突显用户长期偏好和短期偏好对于排序的不同影响，我们按照时间维度对行为序列进行了划分：Session、半小时、一天、一周等粒度，也在线上取得了收益。

3.2.2　用户 ID 的 Embedding

一种更常见的刻画用户偏好的方式，是直接将用户 ID 经过 Embedding 后作为特征接入到模型中，但是最后上线的效果却不尽如人意。通过分析用户的行为数据，我们发现相当一部分用户 ID 的行为数据较为稀疏，导致用户 ID 的 Embedding 没有充分收敛，未能充分刻画用户的偏好信息。

Airbnb 发表在 KDD 2018 上的文章为这种问题提供了一种解决思路 [9]——利用用户基础画像和行为数据对用户 ID 进行聚类。Airbnb 的主要场景是为旅游用户提供民宿短租服务，一般用户一年旅游的次数在 1–2 次之间，因此 Airbnb 的用户行为数据相比点评搜索会更为稀疏一些。

Buckets	1	2	3	4	5	6	7	8
Market	SF	NYC	LA	HK	PHL	AUS	LV	...
Language	en	es	fr	jp	ru	ko	de	...
Device Type	Mac	Msft	Andr	Ipad	Tablet	Iphone	...	
Full Profile	Yes	No						
Profile Photo	Yes	No						
Num Bookings	0	1	2-7	8+				
$ per Night	<40	40-55	56-69	70-83	84-100	101-129	130-189	190+
$ per Guest	<21	21-27	28-34	35-42	43-52	53-75	76+	
Capacity	<2	2-2.6	2.7-3	3.1-4	4.1-6	6.1+		
Num Reviews	<1	1-3.5	3.6-10	> 10				
Listing 5 Star %	0-40	41-60	61-90	90+				
Guest 5 Star %	0-40	41-60	61-90	90+				

图 5　按照用户画像和行为信息聚类

如上图所示，将用户画像特征和行为特征进行离散分桶，拼接特征名和所属桶号，得到的聚类 ID 为：US_lt1_pn3_pg3_r3_5s4_c2_b1_bd2_bt2_nu3。

我们也采取了类似 Airbnb 的方案，稀疏性的问题得到了很好的解决，并且这样做还获得了一些额外的收益。大众点评作为一个本地化的生活信息服务平台，大部分用户的行为都集中自己的常驻地，导致用户到达一个新地方时，排序个性化明显不足。通过这种聚类的方式，将异地有相同行为的用户聚集在一起，也能解决一部分跨站的个性化问题。

3.2.3　商户信息 Embedding

商户 Embedding 除了可以直接将商户 ID 加入模型中之外，美团大脑也利用深度学习技术对 UGC 进行大量挖掘，对商家的口味、特色等细粒度情感进行充分刻画，例如下图所示的“好停车”、“菜品精致”、“愿意再次光顾”等标签。

图 6　美团大脑提供的商家细粒度情感标签

这些信息与单纯的商户星级、点评数相比，刻画的角度更多，粒度也更细。我们将这些标签也进行 Embedding 并输入到模型中：

- **直连**：将标签特征做 Pooling 后直接输入模型。这种接入方式适合端到端的学习方式；但受输入层大小限制，只能取 Top 的标签，容易损失抽象实体信息。
- **分组直连**：类似于直连接入的方式，但是先对标签进行分类，如菜品 / 风格 / 口味等类别；每个分类取 Top N 的实体后进行 Pooling 生成不同维度的语义向量。与不分组的直连相比，能够保留更多抽象信息。

- **子模型接入**：可以利用 DSSM 模型，以标签作为商户输入学习商户的 Embedding 表达。此种方式能够最大化保留标签的抽象信息，但是线上实现和计算成本较高。

3.2.4 加速 Embedding 特征的收敛

在我们的深度学习排序模型中，除了 Embedding 特征，也存在大量 Query、Shop 和用户维度的强记忆特征，能够很快收敛。而 Embedding 特征是更为稀疏的弱特征，收敛速度较慢，为了加速 Embedding 特征的收敛，我们尝试了如下几种方案：

- **低频过滤**：针对出现频率较低的特征进行过滤，可以很大程度上减少参数量，避免过拟合。
- **预训练**：利用多类模型对稀疏 Embedding 特征进行预训练，然后进入模型进行微调：
  - 通过无监督模型如 Word2vec、Fasttext 对用户 – 商户点击关系建模，生成共现关系下的商户 Embedding。
  - 利用 DSSM 等监督模型对 Query– 商户点击行为建模得到 Query 和商户的 Embedding。
- **Multi-Task**：针对稀疏的 Embedding 特征，单独设置一个子损失函数，如下图所示。此时 Embedding 特征的更新依赖两个损失函数的梯度，而子损失函数脱离了对强特征的依赖，可以加快 Embedding 特征的收敛。

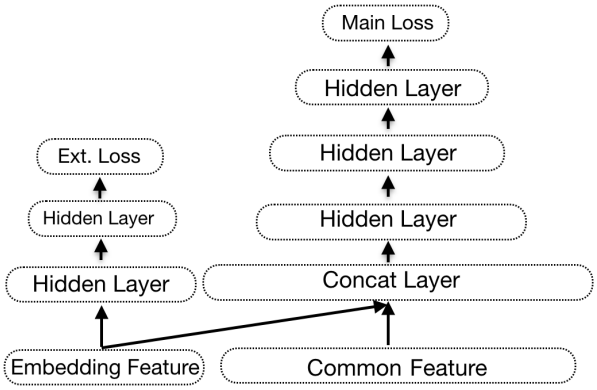


图 7　Multi-Task 加速 Embedding 特征收敛

3.3　图片特征

图片在搜索结果页中占据了很大的展示面积，图片质量的好坏会直接影响用户的体验和点击，而点评商户首图来自于商户和用户上传的图片，质量参差不齐。因此，图片特征也是排序模型中较为重要的一类。目前点评搜索主要用了以下几类图片特征：

- **基础特征**：提取图片的亮度、色度饱和度等基础信息，进行特征离散化后得到图片基础特征。
- **泛化特征**：使用 ResNet50 进行图片特征提取 [3]，通过聚类得到图片的泛化特征。
- **质量特征**：使用自研的图片质量模型，提取中间层输出，作为图片质量的 Embedding 特征。
- **标签特征**：提取图片是否是食物、环境、价目表、Logo 等作为图片分类和标签特征。

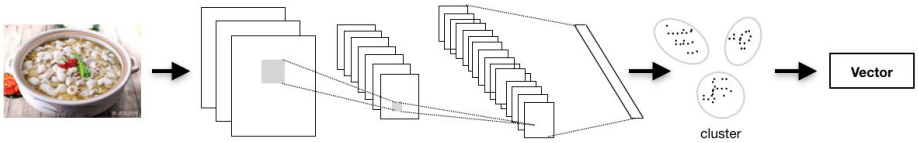


图 8　图片特征接入

4. 适用于搜索场景的深度学习 Listwise 排序算法：LambdaDNN

4.1 搜索业务指标与模型优化目标的 Gap

通常模型的预测目标与业务指标总会存在一些 Gap。如果模型的预测目标越贴近业务目标，越能保证模型优化的同时业务指标也能够有相应的提升；反之则会出现模型离线指标提升，但线上关键业务指标提升不明显，甚至出现负向的问题。工业届大部分深度学习排序采用 Pointwise 的 Log Loss 作为损失函数，与搜索业务指标有较大的 Gap。体现在如下两个方面：

1. 搜索业务常用的指标有 QV_CTR 或者 SSR(Session Success Rate)，更关心的是用户搜索的成功率（有没有发生点击行为）；而 Pointwise 的 Log Loss 更多是关注单个 Item 的点击率。
2. 搜索业务更关心排在页面头部结果的好坏，而 Pointwise 的方法则对于所有位置的样本一视同仁。

图 9　Pointwise 和 Listwise 优化目标的区别

基于上述理由，我们对于深度学习模型的损失函数进行了优化。

4.2 优化目标改进：从 Log Loss 到 NDCG

为了让排序模型的优化目标尽量贴近搜索业务指标，需要按照 Query 计算损失，且不同位置的样本具有不同的权重。搜索系统常用的指标 NDCG(Normalized Discounted Cumulative Gain) 相较于 Log Loss 显然更贴近搜索业务的要求，NDCG 计算公式如下：

$$NDCG@k(l)=\frac{1}{Z_k}\sum_{j=1}^{k}G(l_j)\eta(j)$$

累加部分为 DCG(Discounted Cumulative Gain) 表示按照位置折损的收益，对于 Query 下的结果列表 l，函数 G 表示对应 Doc 的相关度分值，通常取指数函数，即 G(lj)=2lj−1（lj 表示的是相关度水平，如 {0，1，2}）；函数 η 即位置折损，一般采用 η(j)=1/log(j+1)，Doc 与 Query 的相关度越高且位置越靠前则 DCG 值会越大。另外，通常我们仅关注排序列表页前 k 位的效果，Zk 表示 DCG@k 的可能最大值，以此进行归一化处理后得到的就是 NDCG@k。

问题在于 NDCG 是一个处处非平滑的函数，直接以它为目标函数进行优化是不可行的。LambdaRank 提供了一种思路：绕过目标函数本身，直接构造一个特殊的梯度，按照梯度的方向修正模型参数，最终能达到拟合 NDCG 的方法 [6]。因此，如果我们能将该梯度通过深度网络进行反向传播，则能训练一个优化 NDCG 的深度网络，该梯度我们称之为 Lambda 梯度，通过该梯度构造出的深度学习网络称之为 LambdaDNN。

要了解 Lambda 梯度需要引入 LambdaRank。LambdaRank 模型是通过 Pairwise 来构造的，通常将同 Query 下有点击样本和无点击样本构造成一个样本 Pair。模型的基本假设如下式所示，令 Pij 为同一个 Query 下 Doci 相比 Docj 更相关的概率，其中 si 和 sj 分别为 Doci 和 Docj 的模型得分：

$$P_{ij} \equiv P(U_i \rhd U_j) \equiv \frac{1}{1+e^{-\sigma(s_i-s_j)}}$$

使用交叉熵为损失函数，令 Sij 表示样本 Pair 的真实标记，当 Doci 比 Docj 更相关时（即 Doci 有被用户点击，而 Docj 没有被点击），有 Sij=1，否则为 −1；则损失函数可以表示为：

$$C = \frac{1}{2}(1-S_{ij})\sigma(s_i-s_j) + \log(1+e^{-\sigma(s_i-s_j)})$$

在构造样本 Pair 时，我们可以始终令 i 为更相关的文档，此时始终有 Sij ≡ 1，代入上式并进行求导，则损失函数的梯度为：

$$\lambda_{ij} = \frac{\partial C(s_i-s_j)}{\partial s_i} = \frac{-\sigma}{1+e^{\sigma(s_i-s_j)}}$$

到目前为止，损失函数的计算过程中并未考虑样本所在的位置信息。因此进一步对梯度进行改造，考虑 Doci 和 Docj 交换位置时的 NDCG 值变化，下式即为前述的 Lambda 梯度。可以证明，通过此种方式构造出来的梯度经过迭代更新，最终可以达到优化 NDCG 的目的。

$$\lambda_{ij} = \frac{\partial C(s_i-s_j)}{\partial s_i} = \frac{-\sigma}{1+e^{\sigma(s_i-s_j)}}|\Delta_{NDCG}|$$

Lambda 梯度的物理意义如下图所示。其中蓝色表示更相关（用户点击过）的文档，则 Lambda 梯度更倾向于位置靠上的 Doc 得到的提升更大（如红色箭头所示）。有了 Lambda 梯度的计算方法，训练中我们利用深度网络预测同 Query 下的 Doc 得分，根据用户实际点击 Doc 的情况计算 Lambda 梯度并反向传播回深度网络，则可以得到一个直接预测 NDCG 的深度网络。

图 10　Lambda 梯度的物理意义

4.3　LambdaDNN 的工程实施

我们利用 TensorFlow 分布式框架训练 LambdaDNN 模型。如前文所述，Lambda 梯度需要对同 Query 下的样本进行计算，但是正常情况下所有的样本是随机 Shuffle 到各个 Worker 的。因此我们需要对样本进行预处理：

1. 通过 QueryId 进行 Shuffle，将同一个 Query 的样本聚合在一起，同一个 Query 的样本打包进一个 TFRecord。
2. 由于每次请求 Query 召回的 Doc 数不一样，对于可变 Size 的 Query 样本在拉取数据进行训练时需要注意，TF 会自动补齐 Mini-Batch 内每个样本大小一致，导致输入数据中存在大量无意义的默认值样本。这里我们提供两点处理方式：
   - MR 过程中对 Key 进行处理，使得多个 Query 的样本聚合在一起，然后在训练的时候进行动态切分。
   - 读取到补齐的样本，根据设定的补齐标记获取索引位，去除补齐数据。

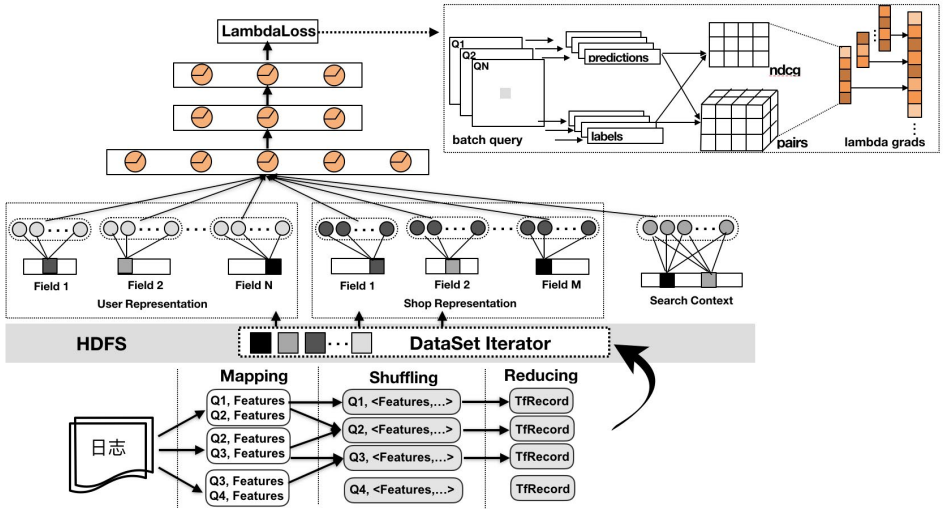


图 11　Lambda 梯度的分布式实现

为了提升训练效率，我们与基础研发平台数据平台中心紧密协同，一起探索并验证了多项优化操作：

1. 将 ID 类特征的映射等操作一并在预处理中完成，减少多轮 Training 过程中的重复计算。
2. 将样本转 TfRecord，利用 RecordDataSet 方式读取数据并计算处理，Worker 的计算性能大概提升了 10 倍。
3. Concat 多个 Categorical 特征，组合成 Multi-Hot 的 Tensor 进行一次 Embedding_Lookup 操作，减少 Map 操作的同时有助于参数做分片存储计算。
4. 稀疏 Tensor 在计算梯度以及正则化处理时保留索引值，仅对有数值的部分进行更新操作。
5. 多个 PS 服务器间进行分片存储大规模 Tensor 变量，减少 Worker 同步更新的通讯压力，减少更新阻塞，达到更平滑的梯度更新效果。

整体下来，对于 30 亿左右的样本量、上亿级别的特征维度，一轮迭代大概在半小时内完成。适当的增加并行计算的资源，可以达到分钟级的训练任务。

4.4 进一步改进优化目标

NDCG 的计算公式中，折损的权重是随着位置呈指数变化的。然而实际曝光点击率随位置变化的曲线与 NDCG 的理论折损值存在着较大的差异。

对于移动端的场景来说，用户在下拉滑动列表进行浏览时，视觉的焦点会随着滑屏、翻页而发生变动。例如用户翻到第二页时，往往会重新聚焦，因此，会发现第二页头部的曝光点击率实际上是高于第一页尾部位置的。我们尝试了两种方案去微调 NDCG 中的指数位置折损：

1. **根据实际曝光点击率拟合折损曲线**：根据实际统计到的曝光点击率数据，拟合公式替代 NDCG 中的指数折损公式，绘制的曲线如图 12 所示。
2. **计算 Position Bias 作为位置折损**：Position Bias 在业界有较多的讨论，其中 [7][8] 将用户点击商户的过程分为观察和点击两个步骤：a. 用户需要首先看到该商户，而看到商户的概率取决于所在的位置；b. 看到商户后点击商户的概率只与商户的相关性有关。步骤 a 计算的概率即为 Position Bias，这块内容可以讨论的东西很多，这里不再详述。

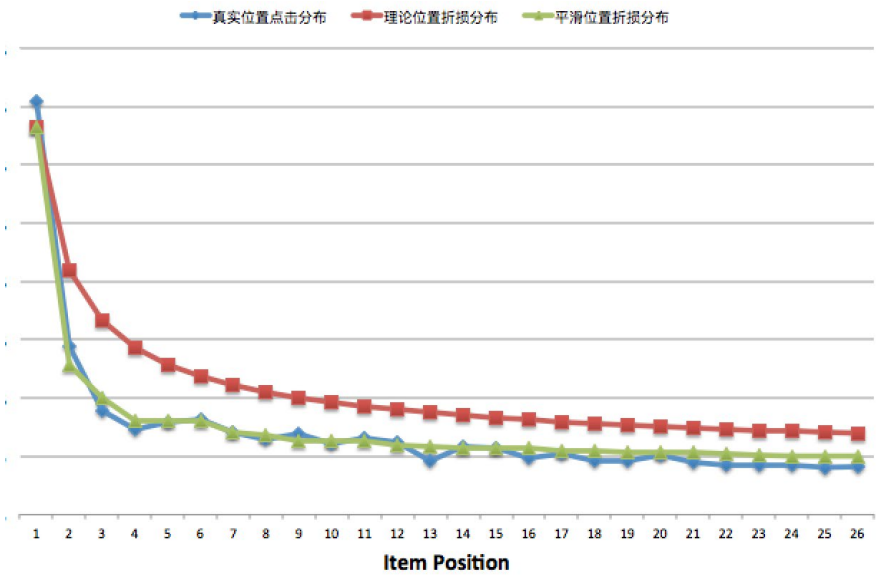


图 12　真实位置折损与理论折损的差别

经过上述对 NDCG 计算改造训练出的 LambdaDNN 模型，相较 Base 树模型和 Pointwise DNN 模型，在业务指标上有了非常显著的提升。

模型	NDCG@10
LambdaMart	0.7141
DNN	0.7378(+3.32%)
LambdaDNN	0.7485(+4.82%)

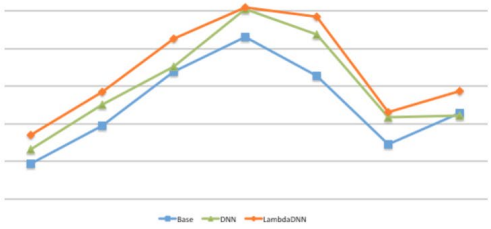


图 13　LambdaDNN 离线 NDCG 指标与线上 PvCtr 效果对比

4.5　Lambda 深度排序框架

Lambda 梯度除了与 DNN 网络相结合外，事实上可以与绝大部分常见的网络结构相结合。为了进一步学习到更多交叉特征，我们在 LambdaDNN 的基础上分别尝试了 LambdaDeepFM 和 LambdaDCN 网络；其中 DCN 网络是一种加入 Cross 的并行网络结构，交叉的网络每一层的输出特征与第一层的原始输入特征进行显性的两两交叉，相当于每一层学习特征交叉的映射去拟合层之间的残差。

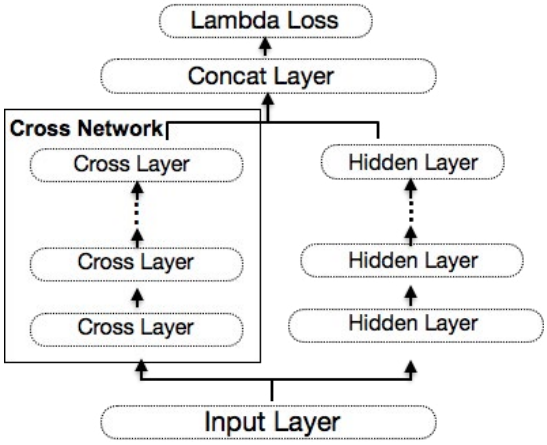


图 14　DCN 模型结构

离线的对比实验表明，Lambda 梯度与 DCN 网络结合之后充分发挥了 DCN 网络的特点，简洁的多项式交叉设计有效地提升模型的训练效果。NDCG 指标对比效

果如下图所示：

模型	NDCG@10
DNN	0.7378
DCN	0.7410(+0.43%)
LambdaDNN	0.7485(+1.45%)
LambdaDCN	0.7496(+1.61%)

图 15　Lambda Loss 与 DCN 网络结果的效果

5. 深度学习排序诊断系统

深度学习排序模型虽然给业务指标带来了大幅度的提升，但由于深度学习模型的“黑盒属性”导致了巨大的解释性成本，也给搜索业务带来了一些问题：

1. **日常搜索 Bad Case 无法快速响应**：搜索业务日常需要应对大量来自于用户、业务和老板们的“灵魂拷问”，“为何这个排序是这样的”，“为什么这家商户质量跟我差不多，但是会排在我的前面”。刚切换到深度学习排序模型的时候，我们对于这样的问题显得手足无措，需要花费大量的时间去定位问题。
2. **无法从 Bad Case 中学习总结规律持续优化**：如果不明白为什么排序模型会得出一个很坏的排序结果，自然也无法定位模型到底出了什么问题，也就无法根据 Bad Case 总结规律，从而确定模型和特征将来的优化方向。
3. **模型和特征是否充分学习无从得知**：新挖掘一些特征之后，通常我们会根据离线评测指标是否有提升决定特征是否上线。但是，即使一个有提升的特征，我们也无法知道这个特征是否性能足够好。例如，模型拟合的距离特征，会不会在特定的距离段出现距离越远反而打分越高的情况。

这些问题都会潜在带来一些用户无法理解的排序结果。我们需要对深度排序模型清晰地诊断并解释。

关于机器学习模型的可解释性研究，业界已经有了一些探索。Lime(Local

Interpretable Model-Agnostic Explanations) 是其中的一种，如下图所示：通过对单个样本的特征生成扰动产生近邻样本，观察模型的预测行为。根据这些扰动的数据点距离原始数据的距离分配权重，基于它们学习得到一个可解释的模型和预测结果[5]。举个例子，如果需要解释一个情感分类模型是如何预测“我讨厌这部电影”为负面情感的，我们通过丢掉部分词或者乱序构造一些样本预测情感，最终会发现，决定“我讨厌这部电影”为负面情感的是因为“讨厌”这个词。

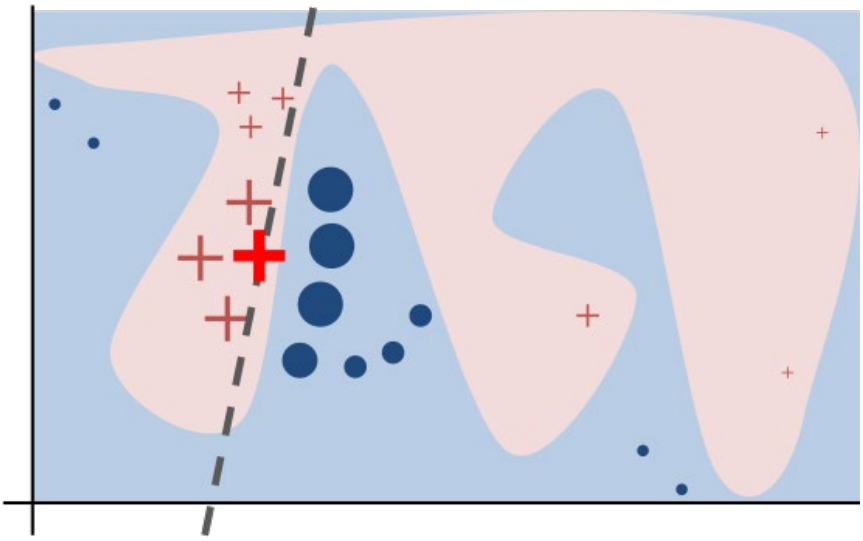


图 16　Lime 解释器的工作原理

基于 Lime 解释器的思想，我们开发了一套深度模型解释器工具——雅典娜系统。目前雅典娜系统支持两种工作模式，Pairwise 和 Listwise 模式：

1. Pairwise 模式用来解释同一个列表中两个结果之间的相对排序。通过对样本的特征进行重新赋值或者替换等操作，观察样本打分和排序位次的变化趋势，诊断出当前样本排序是否符合预期。如下图所示，通过右侧的特征位次面板可以快速诊断出为什么“南京大牌档”的排序比“金时代顺风港湾”要更靠前。第一行的特征位次信息显示，若将“金时代顺风港湾”的 1.3km 的距离特征用“南京大牌档”的 0.2km 的距离特征进行替换，排序位次将上升 10 位；由此得出，“南京大牌档”排在前面的决定性因素是因为距离近。

2. Listwise 模式与 Lime 的工作模式基本类似，通过整个列表的样本生成扰动样本，训练线性分类器模型输出特征重要度，从而达到对模型进行解释的目的。

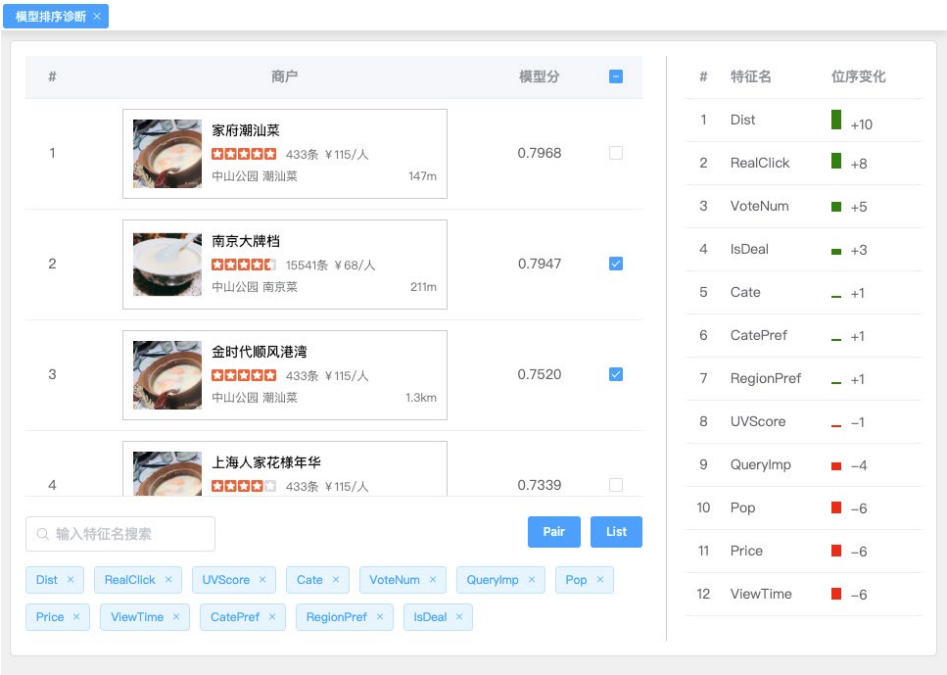


图 17　深度学习排序诊断系统：雅典娜

6. 总结与展望

2018 年下半年，点评搜索完成了从树模型到大规模深度学习排序模型的全面升级。团队在深度学习特征工程、模型结构、优化目标以及工程实践上都进行了一些探索，在核心指标上取得了较为显著的收益。当然，未来依然有不少可以探索的点。

在特征层面，大量知识图谱提供的标签信息尚未充分挖掘。从使用方式上看，简单以文本标签的形式接入，损失了知识图谱的结构信息，因此，Graph Embedding 也是未来需要尝试的方向。同时团队也会利用 BERT 在 Query 和商户文本的深层语义表达上做一些工作。

模型结构层面，目前线上依然以全连接的 DNN 网络结构为主，但 DNN 网络结构在低秩数据的学习上不如 DeepFM 和 DCN。目前 LambdaDeepFM 和

LambdaDCN 在离线上已经取得了收益，未来会在网络结构上做进一步优化。

在模型优化目标上，Lambda Loss 计算损失的时候，只会考虑 Query 内部有点击和无点击的样本对，大量无点击的 Query 被丢弃，同时，同一个用户短时间内在不同 Query 下的行为也包含着一些信息可以利用。因此，目前团队正在探索综合考虑 Log Loss 和 Lambda Loss 的模型，通过 Multi-Task 和按照不同维度 Shuffle 样本让模型充分学习，目前我们已经在线下取得了一些收益。

最后，近期 Google 开源的 TF Ranking 提出的 Groupwise 模型也对我们有一些启发。目前绝大部分的 Listwise 方法只是体现在模型训练阶段，在打分预测阶段依然是 Pointwise 的，即只会考虑当前商户相关的特征，而不会考虑列表上下文的结果，未来我们也会在这个方向上进行一些探索。

参考资料

1. 美团大脑：知识图谱的建模方法及其应用
2. Wide & Deep Learning for Recommender Systems
3. Deep Residual Learning for Image Recognition
4. Attention Is All You Need
5. Local Interpretable Model-Agnostic Explanations: LIME
6. From RankNet to LambdaRank to LambdaMART: An Overview
7. A Novel Algorithm for Unbiased Learning to Rank
8. Unbiased Learning-to-Rank with Biased Feedback
9. Real-time Personalization using Embeddings for Search Ranking at Airbnb

作者简介

非易，2016 年加入美团点评，高级算法工程师，目前主要负责点评搜索核心排序层的研发工作。

祝升，2016 年加入美团点评，高级算法工程师，目前负责点评搜索核心排序层的研发工作。

汤彪，2013 年加入美团点评，高级算法专家，点评平台搜索技术负责人，致力于深层次查询理解和大规模深度学习排序的技术落地。

张弓，2012 年加入美团点评，美团点评研究员。目前主要负责点评搜索业务演进，及集团搜索公共服务平台建设。

仲远，博士，美团 AI 平台部 NLP 中心负责人，点评搜索智能中心负责人。在国际顶级学术会议发表论文 30 余篇，获得 ICDE 2015 最佳论文奖，并是 ACL 2016 Tutorial “Understanding Short Texts” 主讲人，出版学术专著 3 部，获得美国专利 5 项。此前，博士曾担任微软亚洲研究院主管研究员，以及美国 Facebook 公司 Research Scientist。曾负责微软研究院知识图谱、对话机器人项目和 Facebook 产品级 NLP Service。

大众点评信息流基于文本生成的创意优化实践

忆纯　杨肖　明海　众一　扬威　凤阳

1. 引言

信息流是目前大众点评除搜索之外的第二大用户获取信息的入口，以优质内容来辅助用户消费决策并引导发现品质生活。整个大众点评信息流（下文简称点评信息流）围绕个性化推荐去连接用户和信息，把更好的内容推荐给需要的用户。信息流推荐系统涉及内容挖掘、召回、精排、重排、创意等多层机制和排序。本文主要围绕创意部分的工作展开，并选取其中重要的文本创意优化做介绍，分为三个部分：第一部分阐述几个重点问题，包括创意优化是什么，为什么做，以及挑战在哪里；第二部分讲述领域内的应用及技术进展；第三部分介绍我们创意优化的实践，最后做个总结。

什么是创意优化

创意是一个宽泛的概念，它作为一种信息载体对受众展现，可以是文本、图像、视频等任何单一或多类间的组合，如新闻的标题就是经典的创意载体。而创意优化，作为一种方法，指在原有基础上进一步挖掘和激活资源组合方式进而提升资源的价值。在互联网领域产品中，往往表现为通过优化创意载体来提升技术指标、业务目标的过程，在信息流中落地重点包括三个方向：

1. **文本创意**：在文本方面，既包括了面向内容的摘要标题、排版改写等，也包括面向商户的推荐文案及内容化聚合页。它们都广泛地应用了文本表示和文本生成等技术，也是本文的主要方向。
2. **图像创意**：图像方面涉及到首图或首帧的优选、图像的动态裁剪，以及图像的二次生成等。
3. **其他创意**：包括多类展示理由（如社交关系等）、元素创意在内的额外补充信息。

核心目标与推荐问题相似，提升包括点击率、转化率在内的通用指标，同时需要

兼顾考量产品的阅读体验包括内容的导向性等。关于“阅读体验”的部分，这里不作展开。

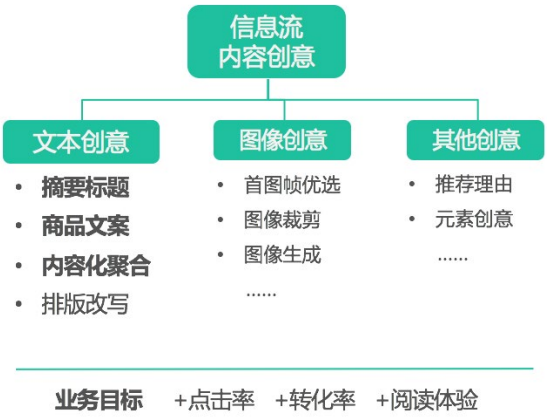


图 1　创意优化的整体应用

为什么要做文本生成

首先文本创意本身为重要的业务发展赋能。在互联网下半场，大众点评平台（下称点评平台）通过内容化去提升用户停留时长，各类分发内容类型在不停地增加，通过优化创意来提升内容的受众价值是必由之路。其次，目前很多内容类型还主要依赖运营维护，运营内容天然存在覆盖少、成本高的问题，无法完全承接需要内容化改造的场景。最后，近几年深度学习在 NLP（Natural Language Processing，自然语言处理）的不同子领域均取得了重大突破。更重要的是，点评平台历经多年，积淀了大量可用的内容数据。从技术层面来说，我们也有能力提供系统化的文本创意生成的解决方案。

对此，我们从文本创意面向对象的角度定义了两类应用形态，分别是面向内容的摘要标题，以及面向商户的推荐文案与内容化聚合页。前者主要应用信息流各主要内容场景，后者则主要应用在信息流广告等内容化场景。这里提前做下产品的简单介绍，帮助大家建立一个立体化的感知。

- **摘要标题**：顾名思义，就是针对某条分发内容生成摘要作标题展示。点评内容

源非常多样，但超过 95% 内容并没有原生标题，同时原生标题质量和多样性等差异也极大。

- **商户文案**：生成有关单个商户核心卖点的描述，一般形式为一句话的短文案。
- **内容聚合**：生成完整的内容页包括标题及多条文案的短篇推荐理由，不同于单商户文案的是，既需要考虑商户的相关性，又要保证理由的多样性。

图 2　文本创意的应用场景

最后需要明确的是，我们做文本创意优化最大的初心，是希望通过创意这个载体显式地连接用户、商户和内容。我们能够知道用户关注什么，知道哪些内容说什么，如何引导用户看，知道哪些商户好、好在哪里，将信息的推荐更进一步。而非为了生成而生成。

面临的挑战

文本创意优化，在业务和技术上分别面临着不同的挑战。首先业务侧，启动创意优化需要两个基础前提：

- 第一，衔接好创意优化与业务目标，因为并不是所有的创意都能优化，也不是所有创意优化都能带来预期的业务价值，方向不对则易蹚坑。
- 第二，创意优化转化为最优化问题，有一定的 Gap。其不同于很多分类排序问

题，本身相对主观，所谓“一千个人眼中有一千个哈姆雷特”，创意优化能不能达到预期的业务目标，这个转化非常关键。

其次，在技术层面，业界不同的应用都面临不一样的挑战，并且尝试和实践对应的解决方案。对文本创意生成来说，我们面临的最大的挑战包括以下三点：

- **带受限的生成** 生成一段流畅的文本并非难事，关键在于根据不同的场景和目标能控制它说什么、怎么说。这是目前挑战相对较大的一类问题，在我们的应用场景中都面临这个挑战。
- **业务导向** 生成能够提升业务指标、贴合业务目标的内容。为此，对内容源、内容表示与建模上提出了更高的要求。
- **高效稳定** 这里有两层含义，第一层是高效，即模型训练预测的效果和效率；第二层是稳定，线上系统应用，需要具备很高的准确率和一套完善的质量提升方案。

2. 文本生成问题综述

我们整体的技术方案演进，可以视作近两年 NLP 领域在深度学习推动下发展的一个缩影。所以在展开之前，先谈一谈整个领域的应用及技术进展。

2.1 相关领域应用

在学界相关领域，文本生成被称为 NLG，其相关任务目标是根据输入数据生成自然语言的文本。而我们在 NLP 领域使用更多的一般是 NLU（Nature Language Understanding 自然语言理解）类任务，如文本分类、命名实体识别等，NLU 的目标则是将自然语言文本转化成结构化数据。NLU 和 NLG 两者表向上是一对相反的过程，但其实是紧密相连的，甚至目前很多 NLU 的任务都受到了生成式模型中表示方法的启发，它们更多只在最终任务上有所区别。

文本生成也是一个较宽泛的概念，如下图所示，广义上只要输出是自然语言文本的各类任务都属于这个范畴。但从不同的输入端可以划分出多种领域应用，从应用相对成熟的连接人和语言的 NMT（神经机器翻译），到 2019 年初，能续写短篇故事的 GPT2 都属于 Text2Text 任务。给定结构化数据比如某些信息事件，来生成

文本比如赛事新闻的属于 Data2Text 类任务，我们的商户文案也属此类。另外还有 Image2Text 等，这块也逐渐在出现一些具有一定可用性又让人眼前一亮的应用，比如各种形式的看图说话。

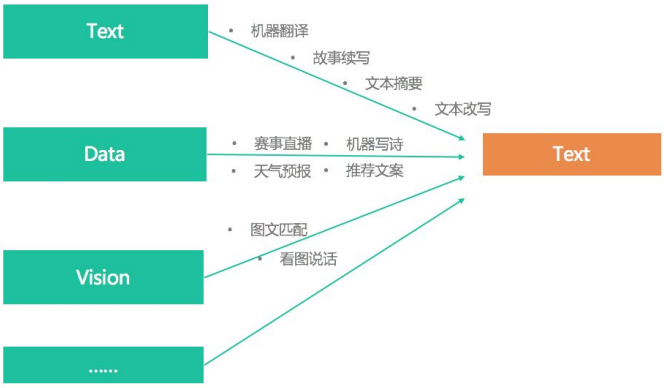


图 3　从输入端划分文本生成的领域应用

2.2　相关技术与进展

文本生成包含文本表示和文本生成两个关键的部分，它们既可以独立建模，也可以通过框架完成端到端的训练。

文本生成

文本生成要解决的一个关键问题，是根据给定的信息如何生成一段文本句子。这是一个简单输入复杂输出的任务，问题的复杂度太大，至今在准确和泛化上都没有兼顾的非常好的方法。2014 年提出的 Seq2Seq Model，是解决这类问题一个非常通用的思路，本质是将输入句子或其中的词 Token 做 Embedding 后，输入循环神经网络中作为源句的表示，这一部分称为 Encoder；另一部分生成端在每一个位置同样通过循环神经网络，循环输出对应的 Token，这一部分称为 Decoder。通过两个循环神经网络连接 Encoder 和 Decoder，可以将两个平行表示连接起来。

另外一个非常重要的，就是 Attention 机制，其本质思想是获取两端的某种权重关系，即在 Decoder 端生成的词和 Encoder 端的某些信息更相关。它也同样可以处理多模态的问题，比如 Image2Text 任务，通过 CNN 等将图片做一个关键特征的向

量表示，将这个表示输出到类似的 Decoder 中去解码输出文本，视频语音等也使用同样的方式（如下图所示）。

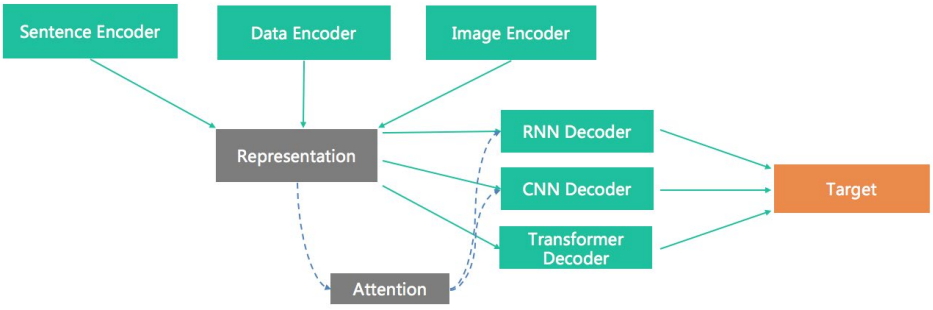


图 4　Seq2Seq 结构图示

可见 Encoder-Decoder 是一个非常通用的框架，它同样深入应用到了文本生成的三种主流方法，分别是规划式、抽取式和生成式，下面看下这几类方法各自的优劣势：

- **规划式**：根据结构化的信息，通过语法规则、树形规则等方式规划生成进文本中，可以抽象为三个阶段。宏观规划解决“说什么内容”，微观规划解决“怎么说”，包括语法句子粒度的规划，以及最后的表层优化对结果进行微调。其优势是控制力极强、准确率较高，特别适合新闻播报等模版化场景。而劣势是很难做到端到端的优化，损失信息上限也不高。
- **抽取式**：顾名思义，在原文信息中抽取一部分作为输出。可以通过编码端的表征在解码端转化为多种不同的分类任务，来实现端到端的优化。其优势在于：能降低复杂度，较好控制与原文的相关性。而劣势在于：容易受原文的束缚，泛化能力不强。
- **生成式**：通过编码端的表征，在解码端完成序列生成的任务，可以实现完全的端到端优化，可以完成多模态的任务。其在泛化能力上具有压倒性优势，但劣势是控制难度极大，建模复杂度也很高。

目前的主流的评估方法主要基于数据和人工评测。基于数据可以从不同角度衡量和训练目标文本的相近程度，如基于 N-Gram 匹配的 BLUE 和 ROUGE 等，基于字符编辑距离（Edit Distance）等，以及基于内容 Coverage 率的 Jarcard 距离等。基于数据的评测，在机器翻译等有明确标注的场景下具有很大的意义，这也是机器翻译领域最先有所突破的重要原因。但对于我们创意优化的场景来说，意义并不大，我们更重要的是优化业务目标，多以线上的实际效果为导向，并辅以人工评测。

另外，值得一提的是，近两年也逐渐涌现了很多利用 GAN（Generative Adversarial Networks，生成对抗网络）的相关方法，来解决文本生成泛化性多样性的问题。有不少思路非常有趣，也值得尝试，只是 GAN 对于 NLP 的文本生成这类离散输出任务在效果评测指标层面，与传统的 Seq2Seq 模型还存在一定的差距，可视为一类具有潜力的技术方向。

文本表示

前文提到，在 Encoder 端包括有些模型在 Decoder 端都需要对句子进行建模，那如何设计一个比较好的模型做表示，既可以让终端任务完成分类、序列生成，也可以做语义推理、相似度匹配等等，就是非常重要的一个部分。那在表示方面，整个 2018 年有两方面非常重要的工作进展：

- **Contextual Embedding**：该方向包括一系列工作，如最佳论文 Elmo(Embeddings from Language Models)，OpenAI 的 GPT(Generative Pre-Training)，以及谷歌大力出奇迹的 BERT(Bidirectional Encoder Representations from Transformers)。解决的核心问题，是如何利用大量的没标注的文本数据学到一个预训练的模型，并通过通过这个模型辅助在不同的有标注任务上更好地完成目标。传统 NLP 任务深度模型，往往并不能通过持续增加深度来获取效果的提升，但是在表示层面增加深度，却往往可以对句子做更好的表征，它的核心思想是利用 Embedding 来表征上下文的的信息。但是这个想法可以通过很多种方式来实现，比如 ELMo，通过双向的 LSTM 拼接后，可以同时得到含上下文信息的 Embedding。而 Transformer 则在

Encoder 和 Decoder 两端，都将 Attention 机制都应用到了极致，通过序列间全位置的直连，可以高效叠加多层（12 层），来完成句子的表征。这类方法可以将不同的终端任务做一个统一的表示，大大简化了建模抽象的复杂度。我们的表示也经历了从 RNN 到拥抱 Attention 的过程。

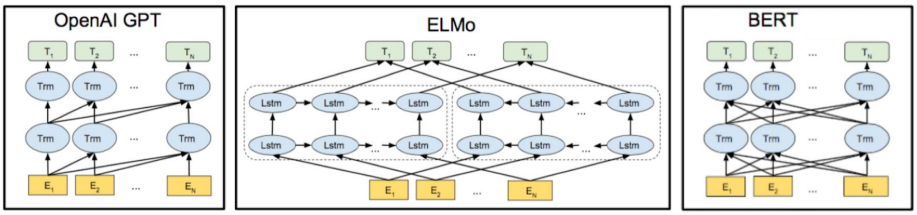


图 5　GPT ELMo BERT 模型结构

- **Tree-Based Embedding**：另外一个流派则是通过树形结构进行建模，包括很多方式如传统的语法树，在语法结构上做 Tree Base 的 RNN，用根结点的 Embedding 即可作为上下文的表征。Tree 本身可以通过构造的方式，也可以通过学习的方式（比如强化学习）来进行构建。最终 Task 效果，既和树的结构（包括深度）有关，也受“表示”学习的能力影响，调优难度比较大。在我们的场景中，人工评测效果并不是很好，仍有很大继续探索的空间。

3. 探索与实践

该部分介绍从 2017 年底至今，我们基于文本生成来进行文本创意优化的一些探索和实践。

3.1　内容源

启动文本生成，首先要了解内容本身，数据的数量和质量对我们的任务重要性无须赘述，这是一切模型的基础。目前我们使用到的数据和大致方法包括：

- **平台渠道**：用户评价、用户笔记、Push、攻略、视频内容、榜单、团单等等。
- **第三方渠道**：合作获取了很多第三方平台的内容来补缺，同时运营侧辅助创意撰写和标注了大量内容，他们同样贡献了可观的数据量。

- **标注数据**：最稀缺的永远是标注数据，尤其是符合业务目标的标注。为此，我们在冷启动阶段设计了 E&E（Explore and Exploit，探索与利用）机制，有意识地积累线上标注，同时尽量引入更多第三方的标注源。

但这些内容的不同特点，也带来了不同的挑战：

- **内容多样**：前面提到的这些内容的结构化程度各不相同，长短差异也极大，对内容表示提出了很高的要求。
- **质量不一**：源内容非常丰富，但事实上质量、质感远远没有达到理想的标准。尤其是占绝对大头的 UGC 的内容，不做好两端的质控将极大影响业务目标的优化，甚至会造成体验问题。
- **聚焦商户**：平台 99% 以上的内容，都以商户作为核心载体，这个对商户的理解和表示同样提出了很高的要求，尤其是在内容化升级的场景下。
- **场景差异**：不同的场景、不同的应用，对模型能力的侧重和优化目标不一样。比如内容和商户，前者要求要有很高的准确率，同时保证优化线上效果；后者更多的是要求有较强的泛化性，并对质感进行优化。

图 7　双平台内容特点与挑战

3.2　基础能力模块

所以，文本创意优化要在业务侧落地产生效果，还需应用到 NLP 领域诸多方向的技术。下图是抽象的整个文本生成应用的基础能力模块，包括用于源和端质量控制的文本质量层，构建 Context 表示的文本表示层，以及面向业务优化的端到端模型

层，其中很多技术应用了公司其他兄弟团队包括内容挖掘组、NLP 中心、离线计算组的出色成果。如针对负面内容过滤的情感分析，多项针对性的文本分类，针对商户表示的标签挖掘等，在这里特别向他们表示感谢。

图 8　文本生成应用的基础能力模块

3.3　信息流标题实践

双平台的内容需要在信息流分发，在创意上最先优化的就是标题，这是用户仅能看到两个要素之一（另一个为首图），而我们超过 95% 的内容并没有原生标题，同时原生标题也存在诸如多样性差非场景导向等问题，还有二次优化的空间。

但是，有两点比较大的挑战，在不同任务上具象可能不一样。它们的本质并没有改变，部分也是业界难点：

- **1. 两个受限条件**：第一，需要以线上点击率转化率为优化目标，线上没效果，写的再好意义都不大；第二，需要与原文强相关，并且容错空间极小，一出现就是 Case。
- **2. 优化评估困难**：第一，模型目标和业务目标间存在天然 Gap；第二，标注数据极度稀缺，离线训练和线上实际预测样本数量之间，往往差距百倍。

对此，我们通过抽取式和生成式的相结合互补的方式，并在流程和模型结构上着手进行解决。

抽取式标题

抽取式方法在用户内容上有比较明显的优势：首先控制力极强，对源内容相关性好，改变用户行文较少，也不容易造成体验问题，可以直接在句子级别做端到端优化。对此，我们把整个标题建模转变为一个中短文本分类的问题，但也无法规避上文提到两个大挑战，具体表现在：

- 在优化评估上，首先标题创意衡量的主观性很强，线上 Feeds 的标注数据也易受到其他因素的影响，比如推荐排序本身；其次，训练预测数据量差异造成 OOV 问题非常突出，分类任务叠加噪音效果提升非常困难。对此，我们重点在语义 + 词级的方向上来对点击 / 转化率做建模，同时辅以线上 E&E 选优的机制来持续获取标注对，并提升在线自动纠错的能力。
- 在受限上，抽取式虽然能直接在 Seq 级别对业务目标做优化，但有时候也须兼顾阅读体验，否则会形成一些“标题党”，亦或造成与原文相关性差的问题。对此，我们抽象了预处理和质量模型，来通用化处理文本创意内容的质控，独立了一个召回模块负责体验保障。并在模型结构上来对原文做独立表示，后又引入了 Topic Feature Context 来做针对性控制。

整个抽取式的流程，可以抽象为四个环节 + 一个在线机制：

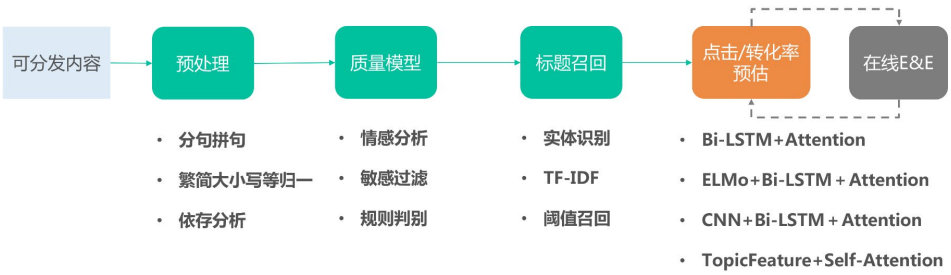


图 9　抽取式生成标题流程

1. 源数据在内容中台完成可分发分析后，针对具体内容，进行系统化插件式的预处理，包括分句拼句、繁简转换、大小写归一等，并进行依存分析。
2. 而后将所有可选内容作质量评估，包括情感过滤、敏感过滤等通用过滤，以

及规则判别等涉及表情、冗余字符处理与语法改写的二次基础优化。

3. 在召回模块中，通过实体识别 +TF-IDF 打分等方式来评估候选内容标题基础信息质量，并通过阈值召回来保证基础阅读体验，从而避免一些极端的 Bad Case。
4. 最后，针对候选标题直接做句子级别的点击 / 转化率预估，负责质感、相关性及最终的业务目标的优化。为此，我们先后尝试了诸多模型结构来解决不同问题，下面重点在这方面做下介绍。

我们第一版 Bi-LSTM + Attention 整个结构并不复杂。我们的输入层是 PreTrain 的 Word Embedding，经过双向 LSTM 给到 Attention 层，Dropout 后全连接，套一个交叉熵的 Sigmod，输出判别，但它的意义非常明显，既可以对整句序列做双向语义的建模，同时可以通过注意力矩阵来对词级进行加权。这个在线上来看，无论是对体感还是点击转化率都较召回打分的原始版本，有了巨大提升。而后，我们还在这个 Base 模型基础上，尝试添加过 ELMo 的 Loss，在模型的第一层双向 LSTM 进行基于 ELMo Loss 的 Pre Train 作为初始化结果，在线上指标也有小幅的提升。

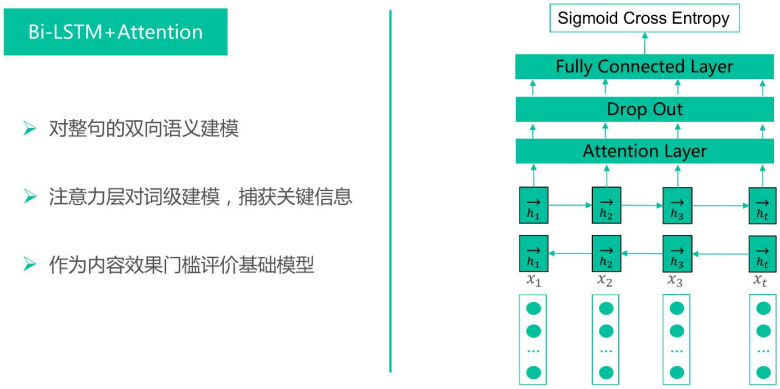


图 10　Bi-LSTM + Attention

但是上述这个结构，将中短文本脱离原文独立建模，显然无法更好地兼顾原文受限这个条件。一个表现，就是容易出现“标题党”、原文不相关等对体验造成影响

的问题。对此，我们在原文与候选标题结合的表示建模方面，做了不少探索，其中以CNN+Bi-LSTM+Attention 的基模型为代表，但其在相关性建模受原文本身长度的影响较大，而且训练效率也不理想。

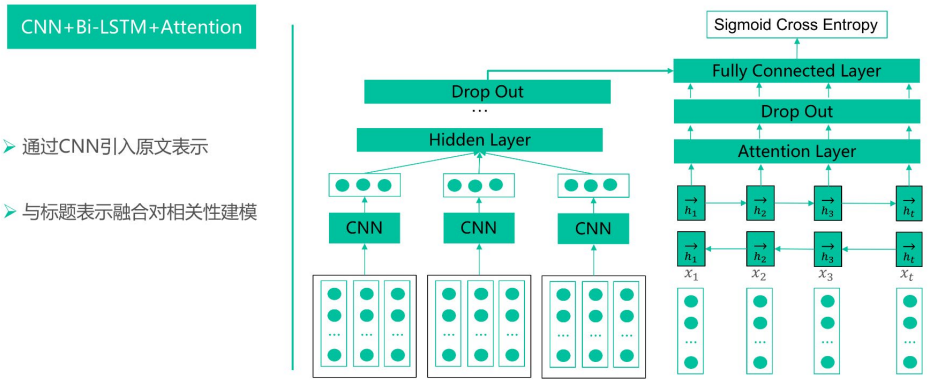


图 11　CNN+Bi-LSTM + Attention

经过一段时间的探索分析，在原文受限问题上，最终既通过深度模型来表征深层的语义，也辅以更多的特征工程，如属性、Topic 等挖掘特征我们统称为 Context，来表征用户能感知到的浅层信息，“两条腿走路”才能被更好的学习，这个在文案生成和标题生成的探索中反过来为抽取式提供了借鉴。

在效率上，我们整体替换了 RNN-LSTM 的循环结构，采用了谷歌那时新提出的自注意力的机制，来解决原文表征训练效率和长依赖问题。采用这个结构在效果和效率上又有了较大的提升。主要问题是，我们的 Context 信息如何更好地建模到 Self-Attention 的结构中。它与生成式模型结构非常类似，在下文生成式部分有所介绍。

另外，需要说明的一点是，除非有两个点以上的巨大提升，一般我们并不会以离线评测指标来评价模型好坏。因为前面提到，我们的标注数据存在不同程度的扰动，而且只是线上预测很小的一个子集，无法避免的与线上存在一定的 Gap，所以我们更关注的是模型影响的基础体验（人工检测通过率即非 Bad Case 率），效率表现（训练预测的时效）最重要的还是线上实际的业务效果。在我们这几个版本的迭代中，这三

个方面都分别获得了不同程度的优化，尤其是包括点击率、总点击量等在内的业务指标，都累计获得了 10% 以上的提升。

	V0.1 TF-IDF	V1.0 Bi-LSTM+Attn	V1.5 CNN+Bi-LSTM+Attn	V2.0 TopicFeature+Self-Attn
业务优化（CTR/CVR/点击曝光）	**+0%**	**+12%**	**+2%**	**+4%**
基础体验（检验通过率）	*+0%*	*+10%*	*+5%*	*+8%*
效率优化（训练预测时效）	*+0%*	*+0%*	*-100%*	*+200%*

图 13　效果数据

受限生成式标题

抽取式标题在包括业务指标和基础体验上都获取了不错的效果，但仍有明显的瓶颈。第一，没有完全脱离原文，尤其在大量质量欠优内容下无法实现创意的二次优化；第二，更好的通过创意这个载体显式的连接用户、商户和内容，这个是生成式标题可以有能力实现的，也是必由之路。

生成式标题，可以抽象描述为：在给定上文并在一定受限条件下，预估下个词的概率的问题。在信息流标题场景，抽取式会面临的问题生成式全部会继承，且在受限优化上面临更大的挑战：

- 原文受限，首先只有表示并学习到原文的语义意图才能更好的控制标题生成，这个本身在 NLU 就是难点，在生成式中就更为突出；其次，标注数据稀缺，原文＋标题对的数据极少，而大部分又存在于长文章。为了保证控制和泛化性，我们初期将标题剥离原文独立建模，通过 Context 衔接，这样能引入更多的非标数据，并在逐步完成积累的情况下，才开始尝试做原文的深度语义表示。
- 优化评估，受限生成式对训练语料的数量和质量要求高很多，首先要保证基础的语义学习也要保证生成端的质量；其次，生成式本质作为语言模型无法在句子层面对业务目标直接做优化，这中间还存在一道 Gap。

在表示上，前面已经提到，我们经历过目标单独建模和结合原文建模的过程，主要原因还是在于仅针对 Target 的理解去构建 Context 衔接，非常容易出现原文相关性问题。所以我们在描述的泛化性方向也做了不少的尝试，比如尽可能地描述广而泛

主题。诸如“魔都是轻易俘获人心的聚餐胜地”，因为只面向上海的商户，内容符合聚餐主题，泛化能力很强，但仍然不能作为一个普适的方案解决问题。

下图为我们一个有初步成效的 RNN-Base 的 Seq2Seq 模型的整体结构。Encoder 端使用的是，包括前面提到的主题（包括商户信息）表示以及原文的双向语义表示，两部分的拼接构成的 Context，输出给注意力层。Decoder 端生成文本时，通过注意力机制学习主题和原文表示的权重关系，这个结构也完整应用到了文案生成，其中控制结构会在文案中展开介绍。

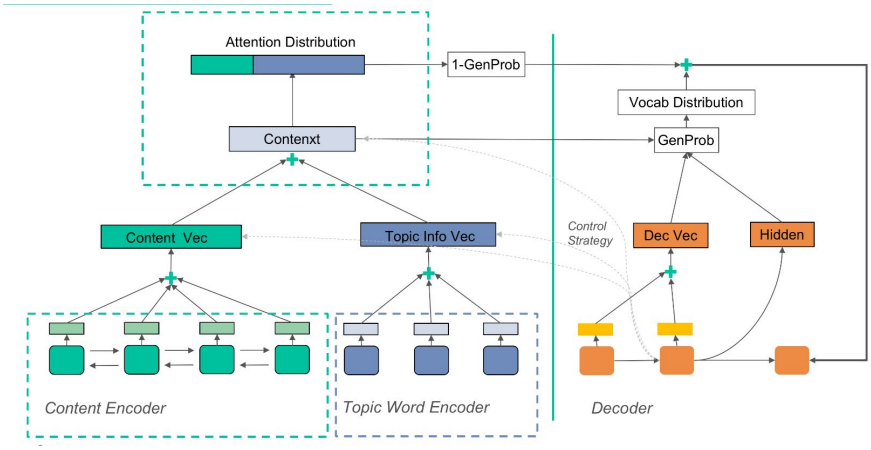


图 14　LSTM Attention Based Seq2Seq 模型结构

在序列建模上，我们经历了一个从 RNN 到自注意力的过程。简单介绍下，序列建模一个核心要点是如何建模序列间的长依赖关系。影响它的重要因素是，信号在网络正向和反向计算中传递的长度（也就是计算次数），较长的依赖关系消失越严重。而在自注意力结构中，每一层都直接与前一层的所有位置直接连接，因此依赖长度均为 O(1)，最大程度保留了序列间的依赖关系。

可以看到，Encoder 包括两部分，一部分是 Source 原文，一部分是基于原文和商户理解的主题 Context，两者共同组成。为此，我们借鉴了 NMT 的一部分研究思想，调整了 Transformer 的结构，在原结构上额外引入了 Context Encoder，并且在 Encoder 和 Decoder 端加入了 Context 的 Attention 层，来强化模型捕捉 Context 信息的能力。

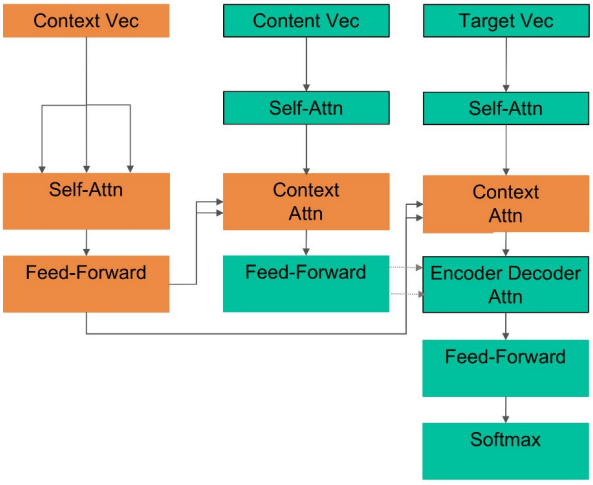


图 15　Transformer Based Seq2Seq Model

我们在生成式方向探索过程中，对低质内容的标题生成，在线上获得了接近 10% 的效果提升，但仍有很多值得进一步的尝试和深挖的空间。

抽取与生成 Combine

在我们的场景中，有两种 Combine 的思路，一个是以业务效果为导向的偏工程化方法，另外一个是我们正在探索的一种 Copy 方法。

工程化的思想非常简洁，在推荐问题上扩充候选，是提升效果的一个可行途径，那生成内容即作为新增的候选集之一，参与整体的预估排序。这个方法能保证最终线上效果不会是负向的，实际上也取得了一定的提升。

另一种方法也是学业界研究的子方向之一，即 Copy 机制，我们也在做重点探索，这里仅作思路的介绍，不再进行展开。

使用 Copy 机制的原始目的，是为了解决生成式的 OOV（超出词表范围）问题。但对于我们的场景来说，大部分的“内容 – 标题”对数据是来自于抽取式，即我们很多标题数据，其实参考了原文。那如何继承这个参考机制，针对业务目标学习何时 Copy 以及 Copy 什么，来更优雅地发挥生成式的优势，就是我们探索 Copy 方法的初衷。我们的方向是对 Copy 和 Generate 概率做独立建模，其中重点解决在受限情况下的“Where To Point”问题。

业务指标与生成式目标的 Gap

我们知道生成式模型其本质是一个 Language Model，它的训练目标是最小化 Word 级别的交叉熵 Loss，而最终我们的需要评价的其实是业务相关的句子级别点击率，这就导致了训练目标和业务指标不一致。

解决这个问题，在我们的场景中有三个可行的方向，第一是在 Context 中显式地标注抽取式模型的 Label，让模型学习到两者的差异；第二是在预测 Decoder 的 Beam Search 计算概率的同时，添加一个打分控制函数；第三则是在训练的 Decoder 中，建立一个全局损失函数参与训练，类似于 NMT 中增加的 Coverage Loss。

考虑到稳定性和实现成本，我们最终尝试了第一和第二种方式，其中第二种方式还是从商户文案迁移过来的，也会在下文进行介绍。在线上，这个尝试并没有在 Combine 的基础上取得更好的效果，但同样值得更加深入的探索。

在线 E&E 机制

最后，介绍一下前面提到过的标题 E&E（Explore and Exploit，探索与利用）机制，用来持续获取标注数据，并提升在线自动纠错的能力。我们采用了一种贪心的 Epsilon Greedy 策略，并做了一点修改，类似经典的 Epsilon 算法，区别是引入创意状态，根据状态将 Epsilon 分成多级。目的是将比较好的创意可以分配给较大概率的流量，而不是均分，差的就淘汰，以此来提升效率。在初期优化阶段，这种方式发挥了很大的作用。

具体我们根据标题和图片的历史表现和默认相比，将状态分成 7 档，从上到下效果表现依次递减，流量分配比例也依次降低，这样可以保证整个系统在样本有噪音的情况下实现线上纠偏。

创意状态	划分原则	Epsilon
Win	曝光充足，且CTR置信大于默认创意	高：70%
Prewin	曝光不充足，且CTR大于默认创意	中：20%
Default	默认创意	
Notsure	曝光不充足	低：10%
Candidate	无曝光	
Prefail	曝光不充足，且CTR低于默认创意	
Fail	曝光充足，且CTR置信小于默认创意	0%

图 17　在线 E&E 选优

3.4　商户文案实践

文案作为一个常见的创意形式，在 O2O 以商户为主要载体的场景下有三点需要：第一，赋予商户以内容调性，丰富创意；第二，通过内容化扩展投放的场景；最后，赋能平台的内容化升级，主要业务目标包括点击率、页面穿透率等等。

文案生成和标题生成能够通用整体的生成模型框架，可以归为 Data2Text 类任务，最大区别是由文案的载体”商户”所决定。不同于内容，准确性的要求低很多，复杂度也大大降低，但同时为泛化能力提出了更高的要求，也带来了与内容生成不同的问题。首先在表示上，对商户的结构化理解变得尤其关键；其次在控制上，有 D2T 任务特有且非常重要的控制要求。前文也提到了生成一段文本从来不是难点，重要的是如何按照不同要求控制 Seq 生成的同时，保证很好的泛化性。下文也会分别介绍卖点控制、风格控制、多样性控制控制等几个控制方法。实现这样的控制，也有很多不同的思路。

商户表示

商户的表示抽象为 Context，如下图中所示，主要分两部分。

第一部分来源于商户的自身理解，一部分则来源于目标文本，两部分有一定交集。其中商户理解的数据为卖点或者 Topic，在初期，为了挖掘商户卖点和 Topic，我们主要使用成本较低、无需标注的 LDA。但是它的准确性相对不可控，同时对产出的卖点主题仍需要进行人工的选择，以便作为新的标注，辅助后续扩展有监督的任

务。我们通过 Key 和 Value 两个 Field，来对卖点和主题进行共同表达（也存在很多只有 Value 的情况），比如下图这个商户“菜品”是个 Key，“雪蟹”是 Value，“约会”则仅是 Value。随着时间的推移，后续我们逐渐利用平台商户标签和图谱信息，来扩展商户卖点的覆盖，以此丰富我们的输入信息。该部分在内容挖掘和 NLP 知识图谱的相关介绍中都有涉及，这里不再进行展开。

第二部分目标文本来源，特意添加这部分进入 Context，主要有三方面原因：

- 第一，仅仅依靠商户理解的 Context，在训练过程中 Loss 下降极慢，并且最终预测生成多样性不理想。本质原因是，目标文本内容与商户卖点、主题间的相关性远远不够。通过不同商户的集合来学习到这个表示关系，非常困难。
- 第二，拓宽可用数据范围，不受商户评论这类有天然标注对的数据限制，从商户衔接扩展到卖点衔接，引入更多的泛化描述数据，比如各类运营文案等等。
- 第三，这也是更为重要的一点，能够间接地实现卖点选择的能力，这个会在下文进行介绍。

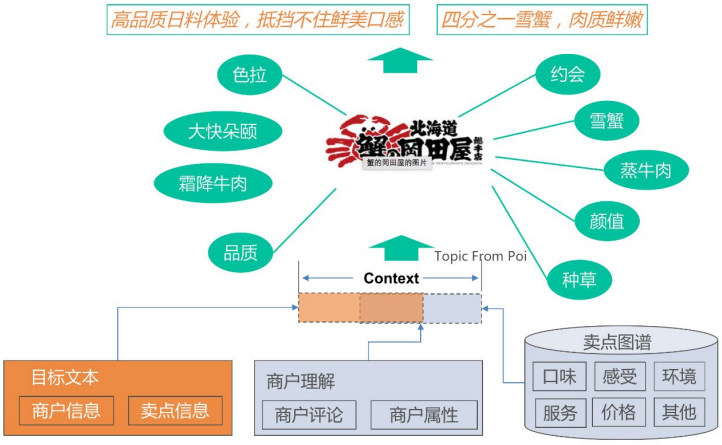


图 18　商户表示

控制端实现

控制，在解码端表现为两类，一类我们称之为 Hard Constrained（强控制），即在数据端给定（或没有给定）的信息，一定要在解码端进行（或不进行）相应描述，这

个适用于地域类目等不能出错的信息。比如这家商户在上海，生成时不能出现除上海以外的地域信息，否则容易造成歧义。另一类称之为 Soft Constrained（弱控制），不同于 NMT 问题，在文案生成上即便是完全相同的输入，不同的输出都是允许的，比如同一商户，最终的文案可以选择不同的卖点去描述不同的内容。

这类同属受限优化的问题，前文提到过有两个思路方向：第一，通过构建机制来让模型自己学习到目标；第二，在 Decoder 的 Beam Search 阶段动态地加入所需的控制目标。我们使用两者相结合的方法，来完成最终的不同控制的实现。

- **两端机制设计**：在具体机制实现上，主要依赖在 Input Context 和 Output Decoder 两端同时生效，让 Context 的 Hard Constrained 来源于 Output，从而使 Model 能够自动学习到强受限关系；而 Soft Constrained 则通过贝叶斯采样的方法，动态添加进 Context，从而帮助 Model 提升泛化能力。
- **Decoder 控制**：简单介绍下 Beam Search，前面提到过，文本生成的预测过程是按 Word 级进行的，每轮预测的候选是整个词汇空间，而往往一般的词表都是十万以上的量级。如果生成序列序列长度为 N，最终候选序列就有十万的 N 次方种可能，这在计算和存储上绝不可行。这时候，就需要使用到 Beam Search 方法，每一步保留最优的前 K（K 一般为 2）个最大概率序列，其他则被剪枝，本质上可以视作一个压缩版的维特比解码。

我们在预测 Beam Search 阶段，除了计算模型概率外，额外增加下图中绿色部分的 Fuction。输入为之前已生成的序列，具体计算逻辑取决于控制目标，可以自由实现。

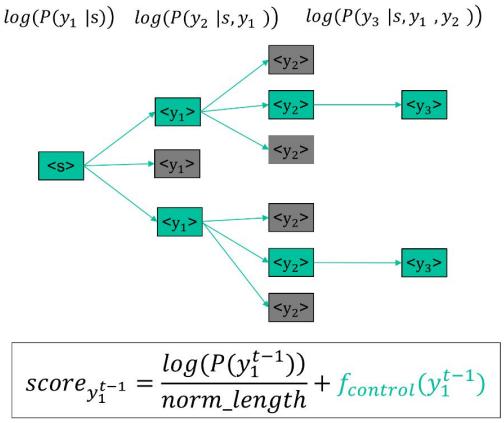


图 19　Decoder Beam_Search 控制

下面简单介绍两个重要的控制实现：

- **卖点控制**：这是最重要的一个控制机制，我们整理了涉及到 Hard Constrained 的卖点和实体，重要的如地域、品类等，在目标理解过程中直接加入 Context。对于 Soft Constrained，我们通过卖点的共现计算一个简单的条件概率，并将卖点依此条件概率随机添加进 Context 中，从而让模型通过注意力学习到受限关系。最后在 Decoder fuction 部分，我们新增了一个 Hard&Soft Constrained 的匹配打分项，参与最终的概率计算。最终的实际结果，也非常符合我们的预期。
- **风格控制**：实现方法和卖点控制非常相似，只是这里的风格，其实是通过不同内容之间的差异来间接进行实现。比如大众点评头条、PGC 类的内容与 UGC 类的的写作风格，就存在极大的差异。那么在文案上，比如聚合页标题上可能更需要 PGC 的风格，而聚合页内容上则需要 UGC 的风格。这样的内容属性，即可作为一个 Context 的控制信号，让模型捕获。

3.5　内容聚合

多样性控制

多样性，在文案生成上是一个比较重要和普遍的问题，尤其对于同一个店铺、同

一个卖点或主题同时生成 N 条内容的聚合页来说，更为突出。本质原因是，在解码预测 Beam Search 时永远选择概率最大的序列，并不考虑多样性。但是如果预测时采用 Decoder 概率 Random Search 的方法，则在通顺度上会存在比较大的问题。

对此，我们直接对全局结果进行优化，在预测时把一个聚合页 Context 放到同一个 batch 中，batch_size 即为文案条数，对已经生成序列上进行实体重复检测和 n-gram 重复检测，将检测判重的加一个惩罚性打分，这个简单的思想已经能非常好的解决多样性问题。

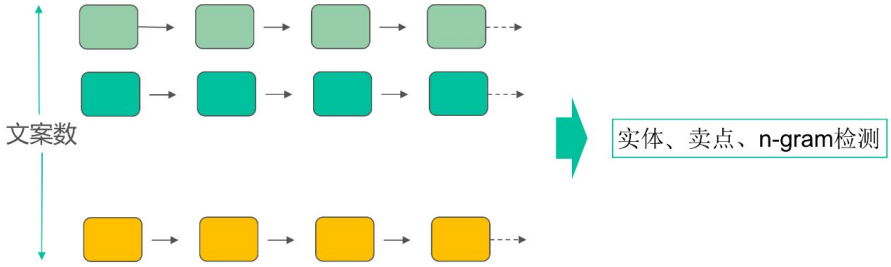


图 20　聚合页多样性控制

4. 动态创意

目前，很多搜索推荐等排序优化场景，都会将创意信息作为特征工程一部分添加进精排或召回模型。那如果把创意优化近似为一个内容级创意排序问题，也可以无缝衔接常用的 Wide&Deep、DNN、FNN 等 CTR 预估模型。但是这之前，需要明确一点非常重要的问题，即它与推荐精排模型的差异，它们之间甚至可能会相互影响，对此，提供下我们的思考。

与精排模型的差异

- 第一，精排模型能否一并完成创意的排序，答案显然是肯定的。但它的复杂度决定了能 Cover 候选集的上限，性能上往往接受不了叉乘创意带来的倍数增长。但此非问题的关键。
- 第二，创意层排序在精排层之前还是之后，直接影响了创意模型的复杂度，也

间接决定了其效果的上限，以及它对精排模型可能的影响程度，从而可能带来全局的影响。此没有最佳实践，视场景权衡。

- 第三，精排模型与创意排序业务目标一致，但实现方式不同。精排模型通过全局排序的最优化来提升业务指标，而创意优化则是通过动态提升内容受众价值来提升业务指标。

最后，我们回到用户视角，当用户在浏览信息流时，其实看到的只有创意本身（标题、图片、作者等信息），但用户却能从中感知到背后的诸多隐含信息，也就是 CTR 预估中的重要内容 / 商户类特征，诸如类目、场景、商户属性等。这个现象背后的本质在于，创意可以表征很多高阶的结构化信息。

基于这一点，在创意优化的特征工程上，方向就很明确了：强化 User/Context，弱化 Item/POI，通过创意表征，来间接学习到弱化的信息从而实现创意层面的最优排序。该部分工作不仅仅涉及到文本，在本文中不再展开。

用户兴趣与文本生成结合的可能性

动态创意为文本生成提供了全新的空间，也提出了更高的要求。动态创意提升受众价值，不仅仅只能通过排序来实现，在正篇介绍的最后部分，我们抛出一个可能性的问题，供各位同行和同学一起思考。也希望能看到更多业界的方案和实践，共同进步。

5. 总结与展望

整个 2018 年，大众点评信息流在核心指标上取得了显著的突破。创意优化作为其中的一部分，在一些方面进行了很多探索，也在效果指标上取得了较为显著的收益。不过，未来的突破，更加任重而道远。

2018 年至 2019 年初，NLP 的各个子领域涌现了非常多令人惊喜的成果，并且这些成果已经落地到业界实践上。这是一个非常好的趋势，也预示着在应用层面会有越来越多的突破。比如 2019 年初，能够续写短篇小说的 GPT2 问世，虽然它真实的泛化能力还未可知，但让我们真切看到了在内容受限下高质量内容生成的可能性。

最后，回到初心，我们希望通过创意的载体显式地连接用户、商户和内容。我们能了解用户关注什么，知道某些内容表达什么，获知哪些商户好，好在哪里，将信息的推荐更进一步。

参考资料

[1] Context-aware Natural Language Generation with Recurrent Neural Networks. arXiv preprint arXiv:1611.09900.
[2] Attention Is All You Need. arXiv preprint arXiv:1706.03762.
[3] Universal Transformers. arXiv preprint arXiv:1807.03819.
[4] A Convolutional Encoder Model for Neural Machine Translation. arXiv preprint arXiv:1611.02344.
[5] Don't Give Me the Details, Just the Summary! Topic-Aware Convolutional Neural Networks for Extreme Summarization. arXiv preprint arXiv:1808.08745.
[6] Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805.
[7] ELMO：Deep contextualized word representations. arXiv preprint arXiv:1802.05365.
[8] openAI GPT：Improving Language Understanding by Generative Pre-Training.
[9] Neural machine translation by jointly learning to align and translate. arXiv preprint arXiv:1409.0473.
[10] Tensor2Tensor for Neural Machine Translation. arXiv preprint arXiv:1803.07416.
[11] A Convolutional Encoder Model for Neural Machine Translation. arXiv preprint arXiv:1611.02344.
[12] Sequence-to-Sequence Learning as Beam-Search Optimization. arXiv preprint arXiv:1606.02960.
[13] A Deep Reinforced Model For Abstractive Summarization. arXiv preprint arXiv:1705.04304.
[14] SeqGAN: Sequence Generative Adversarial Nets with Policy Gradient. arXiv preprint arXiv:1609.05473.
[15] Generating sequences with recurrent neural networks. CoRR,abs/1308.0850.

作者简介

忆纯，2015 年加入美团点评，算法专家，目前负责点评信息流内容创意工作。
杨肖，博士，2016 年加入美团点评，高级算法专家，点评推荐智能中心内容团队负责人。
明海，2016 年加入美团点评，美团点评研究员，点评推荐智能中心团队负责人。
众一，2016 年加入美团点评，算法研发工程师，目前主要负责点评信息流创意相关算法研发工作。
扬威，2018 年初加入美团点评，算法研发工程师，目前主要负责点评信息流动态创意相关算法研发工作。
凤阳，2016 年加入美团点评，算法研发工程师，目前主要负责点评信息流内容运营算法优化的工作。

AI Challenger 2018：细粒度用户评论情感分析冠军思路总结

程惠阁

2018 年 8 月 -12 月，由美团点评、创新工场、搜狗、美图联合主办的“AI Challenger 2018 全球 AI 挑战赛”历经三个多月的激烈角逐，冠军团队从来自全球 81 个国家、1000 多所大学和公司的过万支参赛团队中脱颖而出。其中“后厂村静静”团队 - 由毕业于北京大学的程惠阁（现已入职美团点评）单人组队，勇夺“细粒度用户评论情感分类”赛道的冠军。本文系程惠阁对于本次参赛的思路总结和经验分享，希望对大家能够有所帮助和启发。

背景

在 2018 全球 AI 挑战赛中，美团点评主要负责了其中两个颇具挑战的主赛道赛题：细粒度用户评论情感分析和无人驾驶视觉感知。其中 NLP 中心负责的细粒度用户评论情感分析赛道，最受欢迎，参赛队伍报名数量最多，约占整个报名团队的五分之一。

细粒度用户评论情感分析赛道提供了 6 大类、20 个细分类的中文情感评论数据，标注规模难度之大，在 NLP 语料特别是文本分类相关语料中都属于相当罕见，这份数据有着极其重要的科研学术以及工业应用价值。

赛题简介

在线评论的细粒度情感分析对于深刻理解商家和用户、挖掘用户情感等方面有至关重要的价值，并且在互联网行业有极其广泛的应用，主要用于个性化推荐、智能搜索、产品反馈、业务安全等。本次比赛我们提供了一个高质量的海量数据集，共包含 6 大类 20 个细粒度要素的情感倾向。参赛人员需根据标注的细粒度要素的情感倾向建立算法，对用户评论进行情感挖掘，组委将通过计算参赛者提交预测值和场景真实值之间的误差确定预测正确率，评估所提交的预测算法。

1. 工具介绍

在本次比赛中，采用了自己开发的一个训练框架，来统一处理 TensorFlow 和 PyTorch 的模型。在模型代码应用方面，主要基于香港科技大学开源的 RNet 和 MnemonicReader 做了相应修改。在比赛后期，还加入了一个基于 BERT 的模型，从而提升了一些集成的效果。

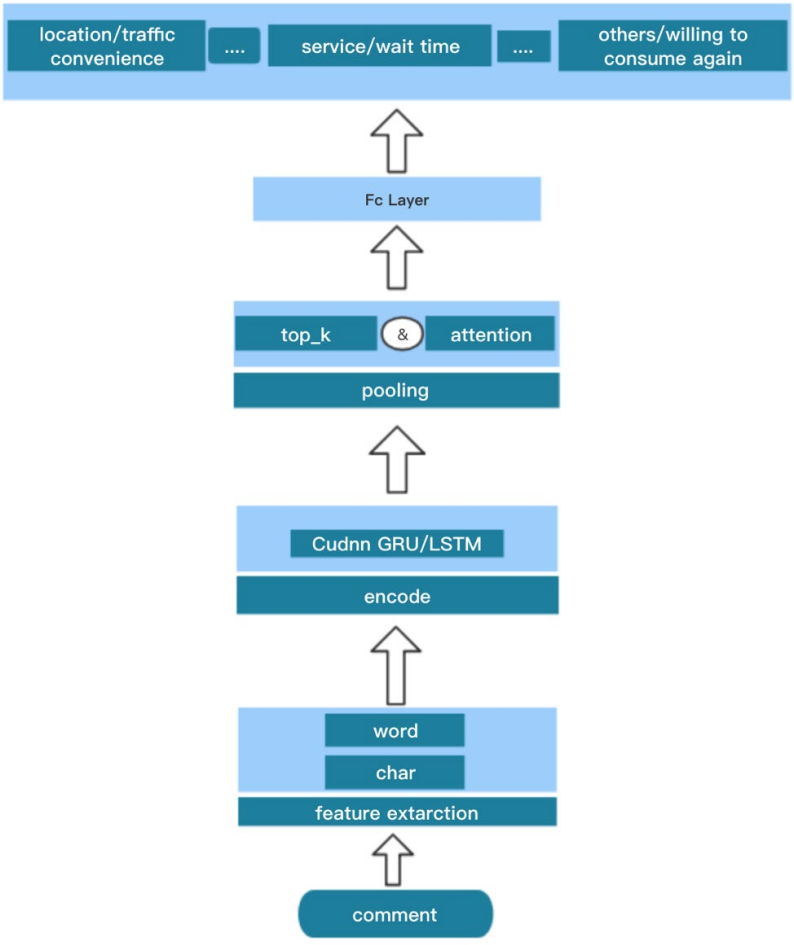
2. 整体思路

整体将该问题看作 20 个 Aspect 的情感多分类问题，采用了传统的文本分类方法，基于 LSTM 建模文本，End2End 多 Aspect 统一训练。

文本分类是业界一个较为成熟的问题，在 2018 年 2 月份，我参加了 Kaggle 的“作弊文本分类”比赛，当时的冠军团队主要依靠基于翻译的数据增强方法获得了成功。2018 年反作弊工作中的一些实践经验，让我意识到，数据是提升文本分类效果的第一关键。因此，我第一时间在网络上寻找到了较大规模的大众点评评论语料，在 Kaggle 比赛的时候，NLP 的语言模型预训练还没有出现，而随着 ELMo 之类模型的成功，也很期待尝试一下预训练语言模型在这个数据集合上的整体效果。

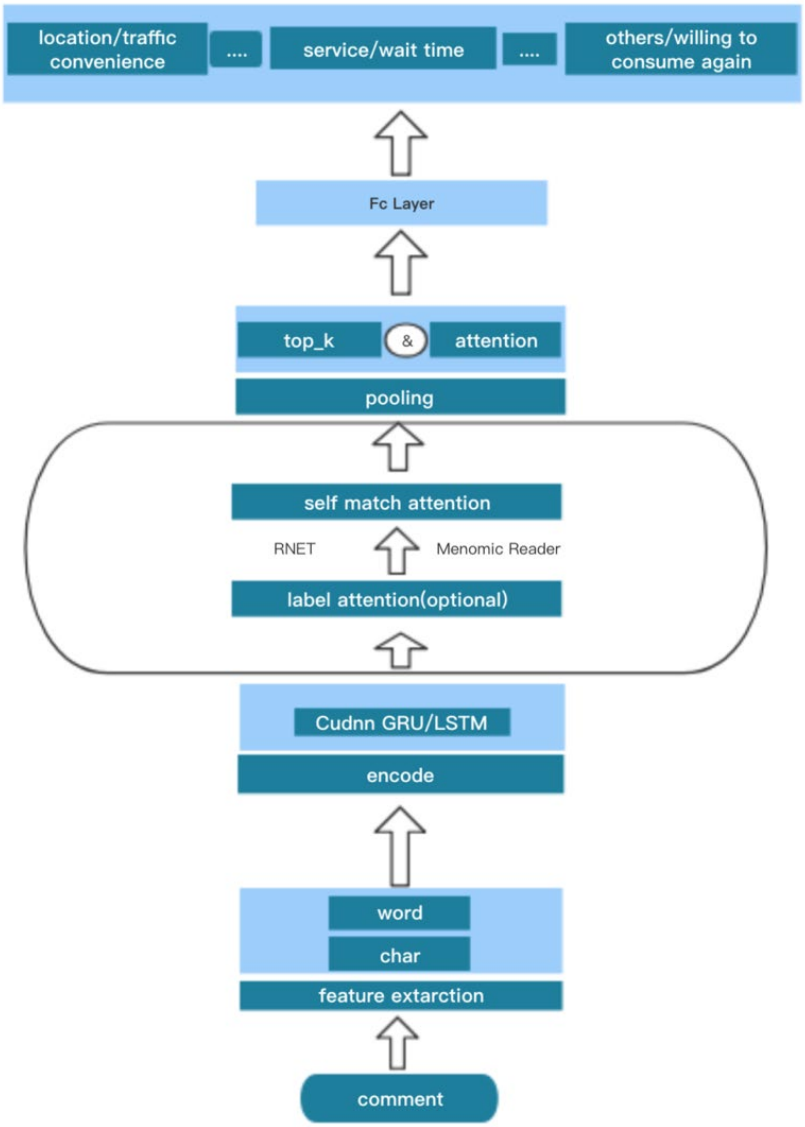
3. 基础模型思路

首先，尝试了不使用预训练语言模型的基础模型，基于 Kaggle Toxic 比赛的经验，直接使用了当时表现最好的 LSTM Encode + Pooling 作为基线模型。在 Kaggle 的比赛中，大家实验的普遍结果是针对中长文本的分类任务的最佳单模型，都是基于 RNN(LSTM/GRU) 或者部分基于 RNN 的模型，比如 RCNN、Capsule + RNN 这样的模型，而其他的模型，比如单纯的 CNN 结构相对表现较差，主要可能是因为 RNN 模型能更好地捕获相对较长距离的顺序信息。



4. 模型层面优化

在基线模型的基础上，效仿阅读理解常见的做法，增加了 Self Attention 层（计算文本到文本自身的 Attention 权重），并将 Attention 之后的输出和原始 LSTM 输出，采用 Gate(RNet) 或者 Semantic Fusion(MnemonicReader) 的方式进行融合。



5. 模型细节处理

更宽的参数更多的模型效果更好

- LSTM 效果好于 GRU。
- Hidden size 400 > 200 > 100。
- Topk Pooling + Attention Pooling 的效果好于单独的 Max 或者 Attention Pooling。
- 共享层前置，Pooling 层 和最后 Fc 层不同 aspect 参数独占效果更好（来自赛后实验，以及其他团队经验）。

这里推测主要原因：是这个数据集有 20 个 Aspect，每个 Aspect 分 4 个不同的类别，所需要的参数相对较多。

三角学习率调节效果最佳

- 参考 BERT 开源代码的学习率设置带来较大效果提升。

采用 Word + Char 的词建模方式

- 这种建模方式能结合分词和字符粒度切分的好处，最大限度避免词汇 UNK 带来的损失。
- 注意对比 Kaggle Toxic 比赛那次比赛是英文语料，对应英文，当时的实验结果是 Word + Ngram 的建模效果更好，收敛更快，所以针对不同 NLP 任务，我们需要具体进行分析。

采用尽可能大的词表

和其他团队相比，我采用了更大的词表 14.4W（Jieba 分词），19.8W（Sentence Piece Unigram 分词），依靠外部大众点评评论数据基于 fastText 预训练词向量，能够支持更大的词表。同时为了避免训练过拟合，采用了只 Finetune 训练中高频的词对低频词固定词向量的处理方式。

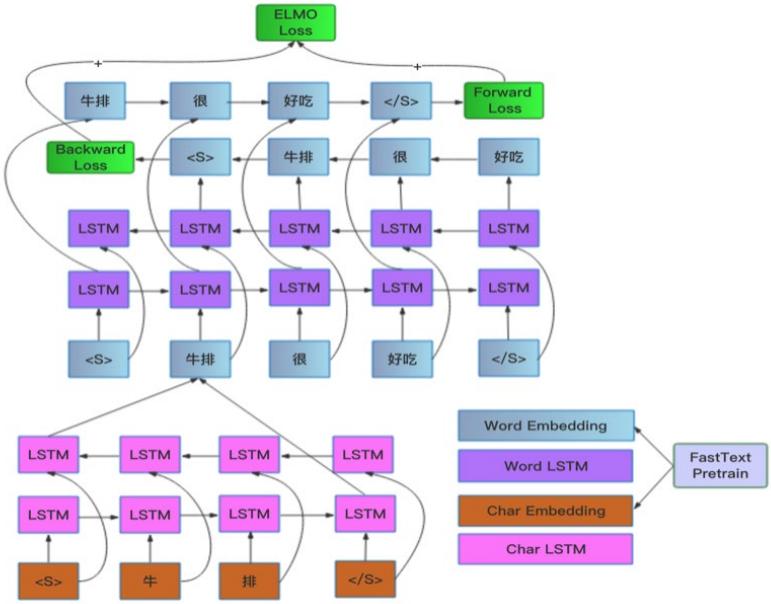
最开始，预计情感相关的词汇相对较少，不需要较大的词表，但是实验过程中发现更大的词表相对地能够提升性能，前提是利用较多的外部数据去比较好的刻画训练数据中低频词的向量。在理论上，我们可以采用一个尽可能大的词表在预测过程中去

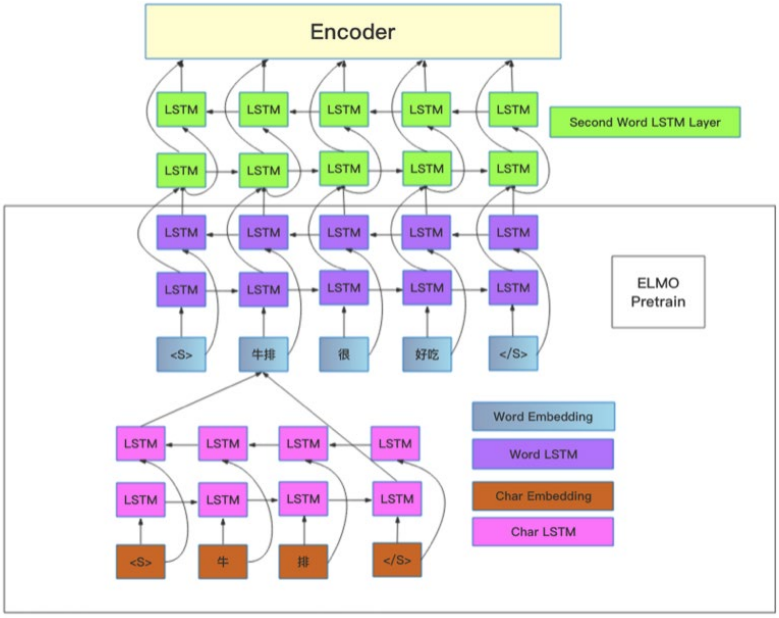
尽可能的减少 UNK 的存在（有论文的结论是对应 UNK 不同的词赋于不同随机向量效果，好于一个固定的 UNK 向量。这里类似，如果我们赋予一个基于无监督外部数据，通过语言模型训练得到的向量则效果更好）。

6. 预训练语言模型

这部分是模型效果提升的关键，这里采用了 ELMo Loss。在简单尝试了官方的 ELMo 版本之后，感觉速度相对比较慢，为此，采用了自己实现的一个简化版的 ELMo，实质上只使用了 ELMo 的 Loss 部分。

在当前双层 LSTM Encoder 的基础上，采用了最小代价的 ELMo 引入，也就是对当前模型的第一层 LSTM 进行基于 ELMo Loss 的预训练，而 Finetune 的时候，模型结构和之前完全不变，只是第一层 LSTM 以及词向量部分采用的 ELMo 预训练的初始化结果，另外在 ELMo 的训练过程中，也采用了基于 fastText 的词向量参数初始化。这个设计使得 ELMo 训练以及 Finetune 训练的收敛，都加快了很多，只需要大概 1 小时的 ELMo 训练，就能在下游任务产生明显受益。值得一提的是，ELMo 和 Self Attention 的搭配在这个数据集合效果非常好。





7. 模型集成

为了取得更好的模型多样性，采用了多种粒度的分词方式，在 Jieba 分词的主要模型基础上，同时引入了基于 SentencePiece 的多种粒度分词。SentencePiece 分词能带来更短的句子长度，但是分词错误相对 Jieba 略多，容易过拟合，因此采用了只 Finetune Char 向量，固定词向量的策略来避免过拟合。多种粒度的分词配合 Word + Char 的建模方式带来了很好的模型多样性。

此外，模型维度的多样性来源自 RNet 结构和 MnemonicReader 结构，以及 BERT 模型的结构的不同。

在模型选择的时候选取了平均 F1 值最优的轮次模型，集成的时候采用了按 Aspect 效果分开加权集成的方式（权重来自 Valid 数据的 F1 分值排序）。基于以上的多样性策略，只需要 7 个单模型集成就能取得较好的效果。

8. 关于 BERT

在实验中基于 Char 的 BERT 单模型，在本次比赛中并没有取得比 ELMo 更好的效果，受限于 512 的长度和只基于 Char 的限制，目前看起来 BERT 模型在这个

数据集合更容易过拟合，Train Loss 下降较快，对应 Valid Loss 效果变差。相信通过适当的优化 BERT 模型能取得更好的效果。

9. 后续优化

F1 的优化是一个有意思的方向。本次比赛中，没有对此做特殊处理，考虑到 F1 是一个全局优化值，如果基于 Batch 强化学习，每个 Batch 可能很难拟合稀有样本分布。

BERT 的进一步优化。因为 BERT 出现之前，基于 Transformer 的模型在长文本分类效果大都是差于基于 LSTM 的模型的，所以如果我们按照 BERT 的 Loss 去预训练基于 LSTM 而不是 Transformer 的模型，在分类问题层面的效果如何？另外，在这个数据集合基于 Transformer 的 BERT，能否取得比 ELMo 更好的分类效果？

对话 AI Challenger 2018 冠军：程惠阁

Q：谈谈对本次参赛的感受？

程惠阁：作为一个多年的算法从业者，我真实的感受到在 AI 时代，技术更新非常之快，比如席卷而来的 ELMo、BERT 等预训练语言模型在工业界影响力之大。包括美团在内的很多公司都快速跟进并上线，而且取得了很好收益，因此技术人员时刻保持学习的心态是非常重要的。

而比赛和工作存在很大的不同，比赛相对更加单纯明确，比赛可以使我在最短时间去学习实验验证一些新的技术，而在标准数据集合验证有效的模型策略，往往在工作中也有实际的价值。对于比赛以及工作中的模型开发，我觉得比较重要的一点首先要做好细致的模型验证部分，在此基础上逐步开发迭代模型才有意义。比如在这次比赛中，我从一开始就监控了包括整体以及各个 Aspect 的包括 F1、AUC、Loss 等等各项指标。

Q：对学习算法的新同学有哪些建议？

程惠阁：如果有时间，可以系统地学习一些名校的深度学习相关的课程，还有很重要的一点，就是实践，我们可以参加去学校项目或者去大公司实习，当然也可以利

用 AI Challenger、Kaggle 这样的竞赛平台进行实践。

Q：为什么会选择参加细粒度用户评论情感分类这个赛道?

程惠阁：因为我之前参加过类似的比赛，并且做过文本分类相关的工作，对这个赛道的赛题也比较感兴趣。

Q：本次比赛最有成就感的事情是什么?

程惠阁：不断迭代提升效果带来的成就感吧，特别是简化版 ELMo 带来的效果提升。

Q：参赛过程中，有哪些收获和成长?

程惠阁：作为一个 TensorFlow 重度用户，我学会了使用 PyTorch 并且体验到 PyTorch 带来的优雅与高效。体验到了预训练语言模型的威力。在比赛中和比赛后，我也收获了很多志同道合的朋友，和他们的交流学习，也帮助我提高了很多。

更重要的是，因为这次比赛，我加入了美团点评这个大家庭，入职这段时间，让我真切地感受到美团点评为了提升用户体验，为了让用户吃的更好，生活更好，在技术方面做了大量的投入。

WSDM Cup 2019 自然语言推理任务获奖解题思路

帅朋

WSDM（Web Search and Data Mining，读音为 Wisdom）是业界公认的高质量学术会议，注重前沿技术在工业界的落地应用，与 SIGIR 一起被称为信息检索领域的 Top2。

刚刚在墨尔本结束的第 12 届 WSDM 大会传来一个好消息，由美团搜索与 NLP 部 NLP 中心的刘帅朋、刘硕和任磊三位同学组成的 Travel 团队，在 WSDM Cup 2019 大赛“真假新闻甄别任务”中获得了第二名的好成绩。队长刘帅朋受邀于 2 月 15 日代表团队在会上作口头技术报告，向全球同行展示了来自美团点评的解决方案。本文将详细介绍他们本次获奖的解决方案。

1. 背景

信息技术的飞速发展，催生了数据量的爆炸式增长。技术的进步也使得了人们获取信息的方式变得更加便捷，然而任何技术都是一把“双刃剑”，信息技术在为人们的学习、工作和生活提供便利的同时，也对人类社会健康持续的发展带来了一些新的威胁。目前亟需解决的一个问题，就是如何有效识别网络中大量存在的“虚假新闻”。

虚假新闻传播了很多不准确甚至虚构的信息，对整个线上资讯的生态造成了很大的破坏，而且虚假新闻会对读者造成误导，干扰正常的社会舆论，严重的危害了整个社会的安定与和谐。因此，本届 WSDM Cup 的一个重要议题就是研究如何实现对虚假新闻的准确甄别，该议题也吸引了全球众多数据科学家的参与。

虽然美团点评的主营业务与在线资讯存在一些差异，但本任务涉及的算法原理是通用的，而且在美团业务场景中也可以有很多可以落地，例如虚假评论识别、智能客服中使用的问答技术、NLP 平台中使用的文本相似度计算技术、广告匹配等。于是，Travel 团队通过对任务进行分析，将该问题转化为 NLP 领域的“自然语言推理”(NLI) 任务，即判断给定的两段文本间的逻辑蕴含关系。因此，基于对任务较为深入理解和平时的技术积累，他们提出了一种解决方案——一种基于多层次深度模型融合框架的虚假新闻甄别技术，该技术以最近 NLP 领域炙手可热的 BERT 为基础模型，并在此基础上提出了一种多层次的模型集成技术。

2. 数据分析

为了客观地衡量算法模型的效果，本届大会组织方提供了一个大型新闻数据集，该数据集包含 32 万多个训练样本和 8 万多个测试样本，这些数据样本均取材于互联网上真实的数据。每个样本包含有两个新闻标题组成的标题对，其中标题对类别标签包括 Agreed、Disagreed、Unrelated 等 3 种。他们的任务就是对测试样本的标签类别进行预测。

“磨刀不误砍柴功”，在一开始，Travel 团队并没有急于搭建模型，而是先对数据进行了全面的统计分析。他们认为，如果能够通过分析发现数据的一些特性，就会有助于后续采取针对性的策略。

首先，他们统计了训练数据中的类别分布情况，如图 1 所示，Unrelated 类别占比最大，接近 70%；而 Disagreed 类占比最小，不到 3%。训练数据存在严重的类别不均衡问题，如果直接用这样的训练数据训练模型，这会导致模型对占比较大类的学习比较充分，而对占比较小的类别学习不充分，从而使模型向类别大的类别进行偏移，存在较严重的过拟合问题。后面也会介绍他们针对该问题提出的对应解决方案。

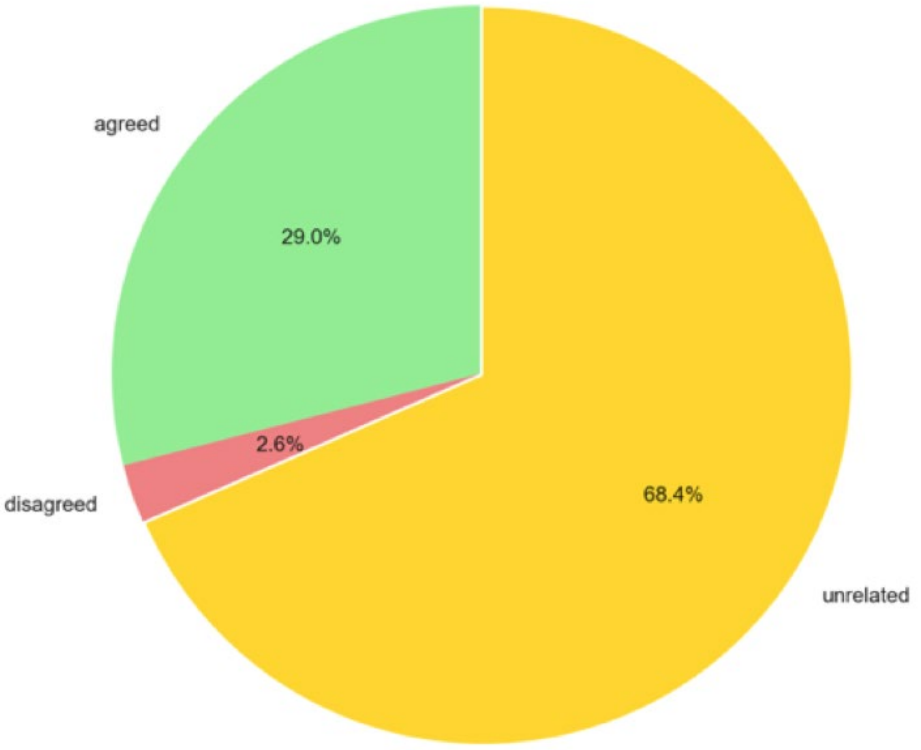


图 1　数据集中类别分布情况

然后，Travel 团队对训练数据的文本长度分布情况进行了统计，如图 2 所示，不同类别的文本长度分布基本保持一致，同时绝大多数文本长度分布在 20 ~ 100 内。这些统计信息对于后面模型调参有着很大的帮助。

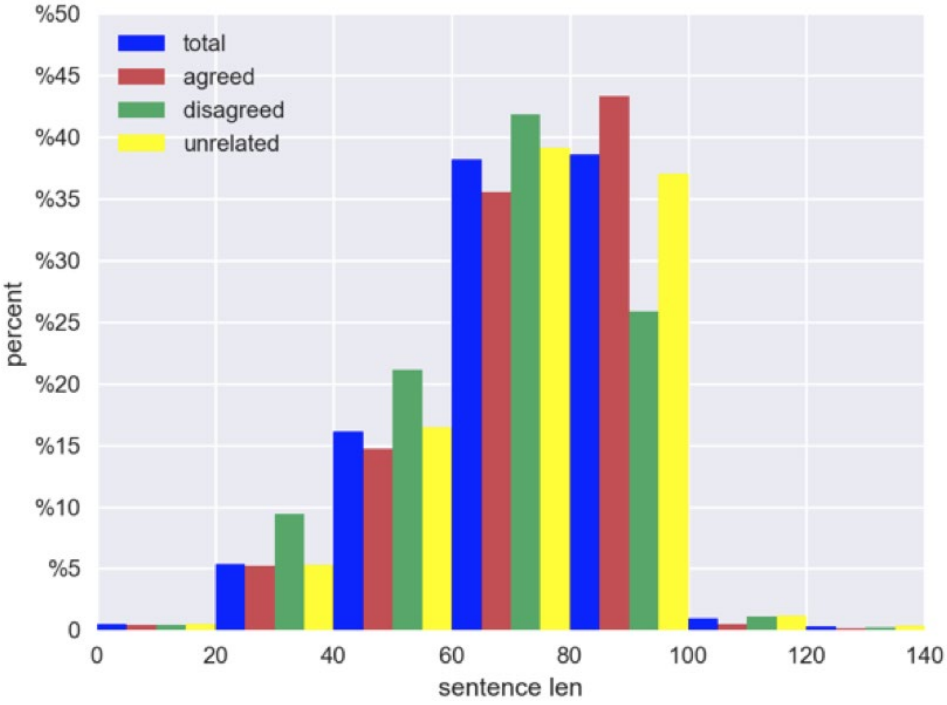


图 2　数据集中文本长度分布情况

3. 数据的预处理与数据增强

本着“数据决定模型的上限，模型优化只是不断地逼近这个上限”的想法，接下来，Travel 团队对数据进行了一系列的处理。

在数据分析时，他们发现训练数据存在一定的噪声，如果不进行人工干预，将会影响模型的学习效果。比如新闻文本语料中简体与繁体共存，这会加大模型的学习难度。因此，他们对数据进行繁体转简体的处理。同时，过滤掉了对分类没有任何作用的停用词，从而降低了噪声。

此外，上文提到训练数据中，存在严重的样本不均衡问题，如果不对该问题做针对性的处理，则会严重制约模型效果指标的提升。通过对数据进行了大量的分析后，他们提出了一个简单有效的缓解样本不均衡问题的方法，**基于标签传播的数据增强方法**。具体方法如图 3 所示：

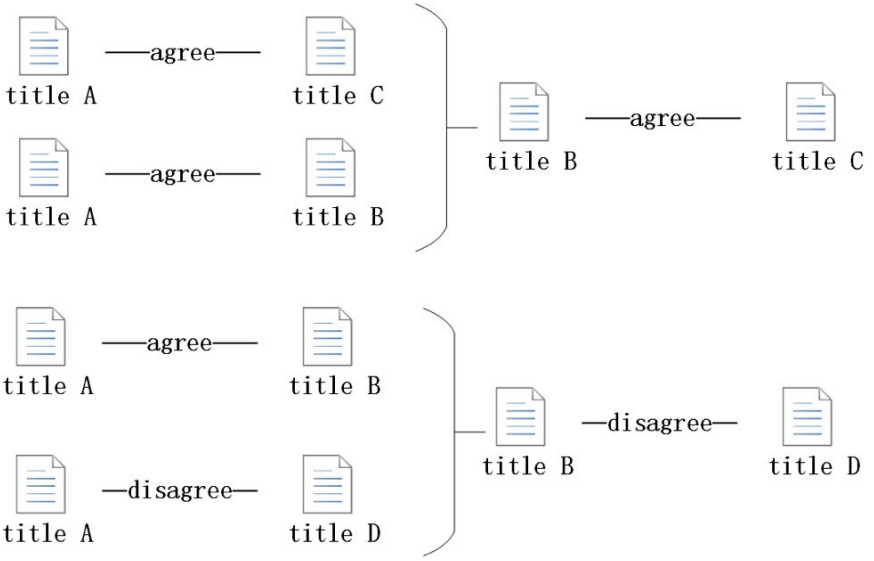


图 3　数据增强策略

如果标题 A 与标题 B 一致，而标题 A 与标题 C 一致，那么可以得出结论，标题 B 与标题 C 一致。同理，如果标题 A 与标题 B 一致，而标题 A 与标题 D 不一致，那么可以得出结论，标题 B 与标题 D 也不一致。此外，Travel 团队还通过将新闻对中的两条文本相互交换位置，来扩充训练数据集。

4. 基础模型

BERT 是 Google 最新推出的基于双向 Transformer 的大规模预训练语言模型，在 11 项 NLP 任务中夺得 SOTA 结果，引爆了整个 NLP 界。BERT 取得成功的一个关键因素是 Transformer 的强大特征提取能力。Transformer 可以利用 Self-Attention 机制实现快速并行训练，改进了 RNN 最被人所诟病的“训练慢”的缺点，可以高效地对海量数据进行快速建模。同时，BERT 拥有多层注意力结构（12 层或 24 层），并且在每个层中都包含有多个“头”（12 头或 16 头）。由于模型的权重不在层与层之间共享，一个 BERT 模型相当于拥有 $12 \times 12=224$ 或 $24 \times 16=384$ 种不同的注意力机制，不同层能够提取不同层次的文本或语义特征，这可以让 BERT 具有超强的文本表征能力。

本赛题作为典型的自然语言推理（NLI）任务，需要提取新闻标题的高级语义特征，BERT 的超强文本表征能力正好本赛题所需要的。基于上述考虑，Travel 团队的基础模型就采用了 BERT 模型，其中 BERT 网络结构如图 4 所示：

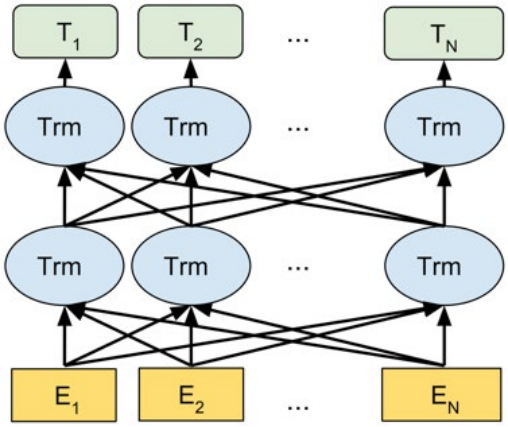


图 4　BERT 网络结构图

在比赛中，Travel 团队在增强后的训练数据上对 Google 预训练 BERT 模型进行了微调（Finetune），使用了如图 5 所示的方式。为了让后面模型融合增加模型的多样性，他们同时 Finetune 了中文版本和英文版本。

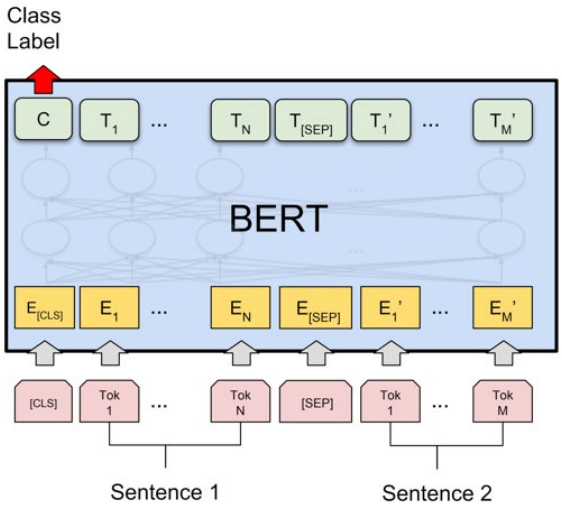


图 5　基于 BERT 的假新闻分类模型结构

5. 多层次深度模型融合框架

模型融合，是指对已有的多个基模型按照一定的策略进行集成以提升模型效果的一种技术，常见的技术包括 Voting、Averaging、Blending、Stacking 等等。这些模型融合技术在前人的许多工作中得到了应用并且取得了不错的效果，然而任何一种技术只有在适用场景下才能发挥出最好的效果，例如 Voting、Averaging 技术的融合策略较为简单，一般来说效果提升不是非常大，但优点是计算逻辑简单、计算复杂度低、算法效率高；而 Stacking 技术融合策略较复杂，一般来说效果提升比较明显，但缺点是算法计算复杂度高，对计算资源的要求较苛刻。

本任务使用的基模型为 BERT，该模型虽然拥有非常强大的表征建模能力，但同时 BERT 的网络结构复杂，包含的参数众多，计算复杂度很高，即使使用了专用的 GPU 计算资源，其训练速度也是比较慢的，因此这就要求在对 BERT 模型融合时不能直接使用 Stacking 这种高计算复杂度的技术，因此我们选择了 Blending 这种计算复杂度相对较低、融合效果相对较好的融合技术对基模型 BERT 做融合。

同时，Travel 团队借鉴了神经网络中网络分层的设计思想来设计模型融合框架，他们想既然神经网络可以通过增加网络深度来提升模型的效果，那么在模型融合中是否也可以通过增加模型融合的层数来提升模型融合的效果呢？基于这一设想，他们提出了一种多层次深度模型融合框架，该框架通过增加模型的层数进而提升了融合的深度，最终取得了更好的融合效果。

具体来说，他们的框架包括三个层次，共进行了两次模型融合。第一层采用 Blending 策略进行模型训练和预测，在具体实践中，他们选定了 25 个不同的 BERT 模型作为基模型；第二层采用 5 折的 Stacking 策略对 25 个基模型进行第一次融合，这里他们选用了支持向量机（SVM）、逻辑回归（LR）、K 近邻（KNN）、朴素贝叶斯（NB），这些传统的机器学习模型，既保留了训练速度快的优点，也保证了模型间的差异性，为后续融合提供了效率和效果的保证；第三层采用了一个线性的 LR 模型，进行第二次模型融合并且生成了最终的结果。模型融合的架构如图 6 所示：

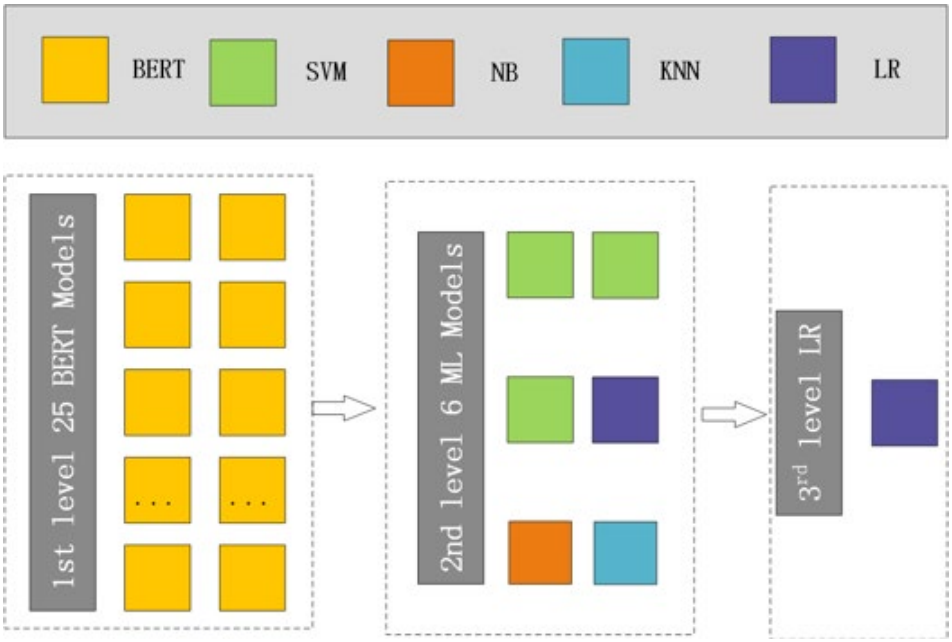


图 6　模型融合架构

整体方案模型训练分为三个阶段，如图 7 所示：

- 第一个阶段，将训练数据划分为两部分，分别为 Train Data 和 Val Data。Train Data 用于训练 BERT 模型，用训练好的 BERT 模型分别预测 Val Data 和 Test Data。将不同 BERT 模型预测的 Val Data 和 Test Data 的结果分别进行合并，可以得到一份新的训练数据 New Train Data 和一份新的测试数据 New Test Data。
- 第二阶段，将上一阶段的 New Train Data 作为训练数据，New Test Data 作为测试数据。本阶段将 New Train Data 均匀的划分为 5 份，使用“留一法”训练 5 个 SVM 模型，用这 5 个模型分别去预测剩下的一份训练数据和测试数据，将 5 份预测的训练数据合并，可以得到一份新的训练数据 NewTrainingData2，将 5 份预测的测试数据采用均值法合并，得到一份新的测试数据 NewTestData2。同样的方法再分别训练 LR、KNN、NB 等模型。
- 第三阶段，将上一阶段的 NewTrainingData2 作为训练数据，NewTestDa-

ta2 作为测试数据，重新训练一个 LR 模型，预测 NewTestData2 的结果作为最终的预测结果。为了防止过拟合，本阶段采用 5 折交叉验证的训练方式。

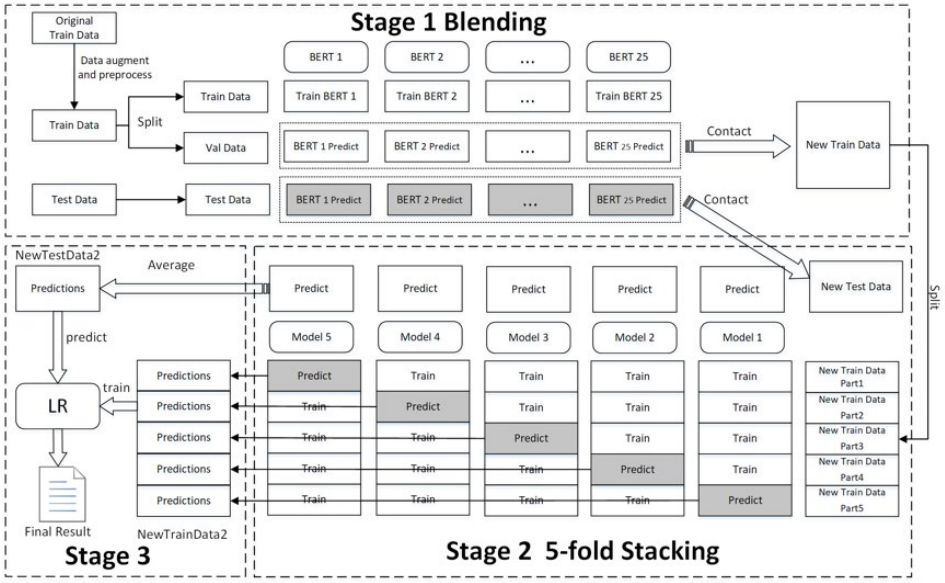


图 7　假新闻分类方案的整体架构和训练流程

6. 实验

6.1　评价指标

为了缓解数据集中存在的类别分布不均衡问题，本任务使用带权重的准确率作为衡量模型效果的评价指标，其定义如下所示：

$$weightedAccuracy(y,\hat{y},\omega)=\frac{1}{n}\sum_{i=1}^{n}\frac{\omega_i(y_i=\hat{y})}{\sum\omega_i}$$

其中，y 为样本的真实类别标签，$\hat{y}$ 为模型的预测结果，ωi 为数据集中第 i 个样本的权重，其权重值与类别相关，其中 Agreed 类别的权重为 1/15，Disagreed 类别的权重为 1/5，Unrelated 类别的权重为 1/16。

6.2 实验结果

在官方测试集上，Travel 团队的最优单模型的准确率达到 0.86750，25 个 BERT 模型简单平均融合后准确率达 0.87700（+0.95PP），25 个 BERT 模型结果以加权平均的形式融合后准确率达 0.87702（+0.952PP），他们提出的多层次模型融合技术准确率达 0.88156（+1.406PP）。实践证明，美团 NLP 中心的经验融合模型在假新闻分类任务上取得了较大的效果提升。

Table 1: Performance of Various Models

Model	Weighted Acc on Private LB
Best Single base model	0.86750
Averaging of 25 BERT	0.87700
Weighted Averaging of 25 BERT	0.87702
Our Empirical Ensemble Model	0.88156

图 8　效果提升

7. 总结与展望

本文主要对解决方案中使用的关键技术进行了介绍，比如数据增强、数据预处理、多层模型融合策略等，这些方法在实践中证明可以有效的提升预测的准确率。由于参赛时间所限，还有很多思路没有来及尝试，例如美团使用的 BERT 预训练模型是基于维基百科数据训练而得到的，而维基百科跟新闻在语言层面也存在较大的差异，所以可以将现有的 BERT 在新闻数据上进行持续地训练，从而使其能够对新闻数据具有更好的表征能。

参考文献

[1] Dagan, Ido, Oren Glickman, and Bernardo Magnini. 2006. The PASCAL recognising textual entailment challenge, Machine learning challenges. evaluating predictive uncertainty, visual object classification, and recognising tectual entailment. Springer, Berlin, Heidelberg, 177–190.

[2] Bowman S R, Angeli G, Potts C, et al. 2015. A large annotated corpus for learning natural language inference. In proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP).

[3] Adina Williams, Nikita Nangia, and Samuel R Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. In NAACL.
[4] Rajpurkar P, Zhang J, Lopyrev K, et al. 2016. Squad: 100,000+ questions for machine comprehension of text. arXiv preprint arXiv:1606.05250.
[5] Luisa Bentivogli, Bernardo Magnini, Ido Dagan, Hoa Trang Dang, and Danilo Giampiccolo. 2009. The fifth PASCAL recognizing textual entailment challenge. In TAC. NIST.
[6] Hector J Levesque, Ernest Davis, and Leora Morgenstern. 2011. The winograd schema challenge. In Aaai spring symposium: Logical formalizations of commonsense reasoning, volume 46, page 47.
[7] Bowman, Samuel R., et al. 2015. "A large annotated corpus for learning natural language inference." arXiv preprint arXiv:1508.05326.
[8] Wang, A., Singh, A., Michael, J., Hill, F., Levy, O., & Bowman, S. R. 2018. GLUE: A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding. arXiv preprint arXiv:1804.07461.
[9] Chen, Q., Zhu, X., Ling, Z., Wei, S., Jiang, H., & Inkpen, D. 2016. Enhanced lstm for natural language inference. arXiv preprint arXiv:1609.06038.
[10] Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding with unsupervised learning. Technical report, OpenAI.
[11] Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805.
[12] David H. Wolpert. 1992. Stacked generalization. Neural Networks (1992). https: // doi.org/10.1016/S0893- 6080(05)80023- 1.

作者简介

刘帅朋，硕士，美团点评搜索与 NLP 部 NLP 中心高级算法工程师，目前主要从事 NLU 相关工作。曾任中科院自动化研究所研究助理，主持研发的智能法律助理课题获 CCTV-1 频道大型人工智能节目《机智过人第二季》报道。

刘硕，硕士，美团点评搜索与 NLP 部 NLP 中心智能客服算法工程师，目前主要从事智能客服对话平台中离线挖掘相关工作。

任磊，硕士，美团点评搜索与 NLP 部 NLP 中心知识图谱算法工程师，目前主要从事美团大脑情感计算以及 BERT 应用相关工作。

会星，博士，担任美团点评搜索与 NLP 部 NLP 中心的研究员，智能客服团队负责人。目前主要负责美团智能客服业务及智能客服平台的建设。在此之前，会星在阿里达摩院语音实验室作为智能语音对话交互专家，主要负责主导的产品有斑马智行语音交互系统，YunOS 语音助理等，推动了阿里智能对话交互体系建设。

富峥，博士，担任美团点评搜索与 NLP 部 NLP 中心的研究员，带领知识图谱算法团队。目前主要负责美团大脑项目，围绕美团吃喝玩乐场景打造的知识图谱及其应用，能够打通餐饮、旅行、休闲娱乐等各个场景数据，为美团各场景业务提供更加智能的服务。张富峥博士在知识图

谱、个性化推荐、用户画像、时空数据挖掘等领域展开了众多的创新性研究，并在相关领域的顶级会议和期刊上发表 30 余篇论文，如 KDD、WWW、AAAI、IJCAI、TKDE、TIST 等，曾获 ICDM2013 最佳论文大奖，出版学术专著 1 部。

仲远，博士，美团点评搜索与 NLP 部负责人。在国际顶级学术会议发表论文 30 余篇，获得 ICDE 2015 最佳论文奖，并是 ACL 2016 Tutorial“Understanding Short Texts”主讲人，出版学术专著 3 部，获得美国专利 5 项。此前，博士曾担任微软亚洲研究院主管研究员，以及美国 Facebook 公司 Research Scientist。曾负责微软研究院知识图谱、对话机器人项目和 Facebook 产品级 NLP Service。

深度学习在美团配送 ETA 预估中的探索与实践

基泽　周越　显杰

1. 背景

ETA（Estimated Time of Arrival，“预计送达时间”），即用户下单后，配送人员在多长时间内将外卖送达到用户手中。送达时间预测的结果，将会以”预计送达时间”的形式，展现在用户的客户端页面上，是配送系统中非常重要的参数，直接影响了用户的下单意愿、运力调度、骑手考核，进而影响配送系统整体成本和用户体验。

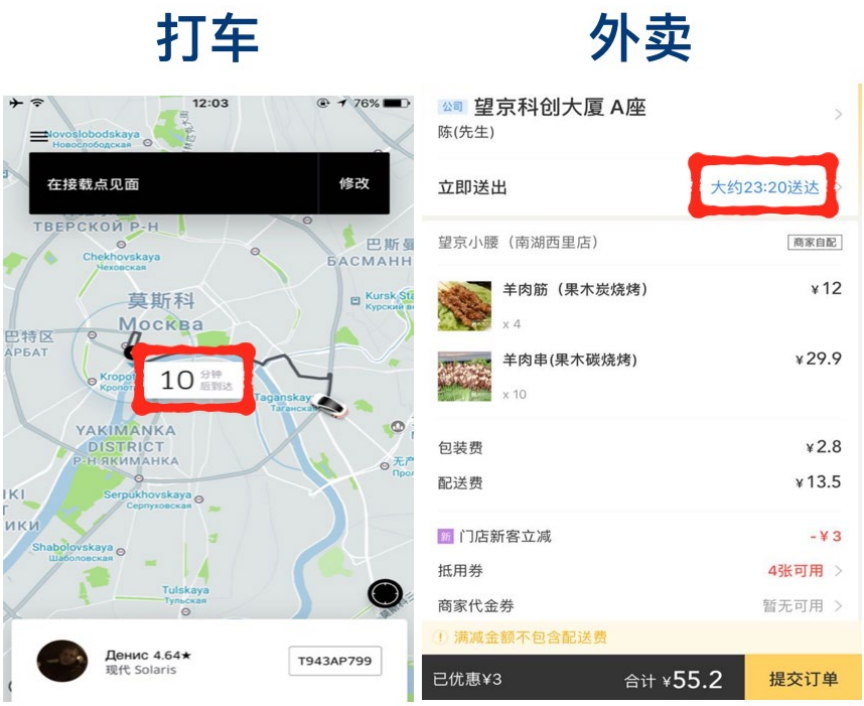
对于整个配送系统而言，ETA 既是配送系统的入口和全局约束，又是系统的调节中枢。具体体现在：

- ETA 在用户下单时刻就需要被展现，这个预估时长继而会贯穿整个订单生命周期，首先在用户侧给予准时性的承诺，接着被调度系统用作订单指派的依据及约束，而骑手则会按照这个 ETA 时间执行订单的配送，配送是否准时还会作为骑手的工作考核结果。
- ETA 作为系统的调节中枢，需要平衡用户 – 骑手 – 商家 – 配送效率。从用户的诉求出发，尽可能快和准时，从骑手的角度出发，太短会给骑手极大压力。从调度角度出发，太长或太短都会影响配送效率。而从商家角度出发，都希望订单被尽可能派发出去，因为这关系到商家的收入。

ETA 在配送系统中作用

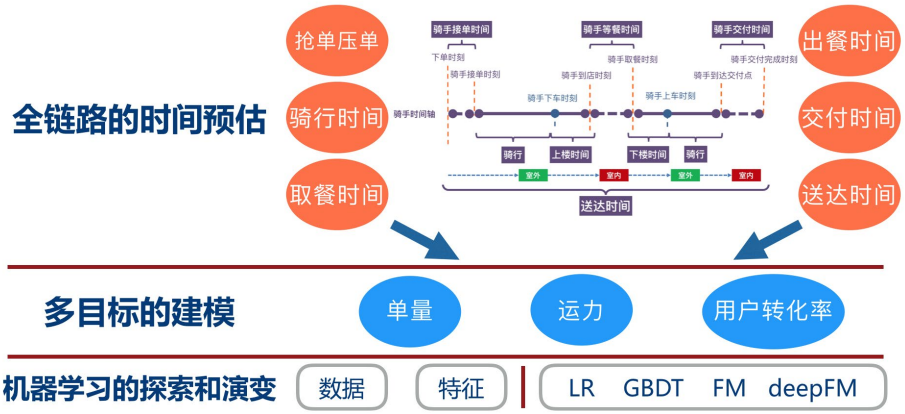
在这样多维度的约束之下，外卖配送的 ETA 的建模和估计会变得更加复杂。与打车场景中的 ETA 做对比，外卖场景的 ETA 面临如下的挑战：

- 外卖场景中 ETA 是对客户履约承诺的重要组成部分，无论是用户还是骑手，对于 ETA 准确性的要求非常高。而在打车场景，用户更加关心是否能打到车，ETA 仅提供一个参考，司机端对其准确性也不是特别在意。
- 由于外卖 ETA 承担着承诺履约的责任，因此是否能够按照 ETA 准时送达，也是外卖骑手考核的指标、配送系统整体的重要指标；承诺一旦给出，系统调度和骑手都要尽力保证准时送达。因此过短的 ETA 会给骑手带来极大的压力，并降低调度合单能力、增加配送成本；过长的 ETA 又会很大程度影响用户体验。
- 外卖场景中 ETA 包含更多环节，骑手全程完成履约过程，其中包括到达商家、商家出餐、等待取餐、路径规划、不同楼宇交付等较多的环节，且较高的合单率使得订单间的流程互相耦合，不确定性很大，做出合理的估计也有更高难度。



外卖及打车中的 ETA

下图是骑手履约全过程的时间轴，过程中涉及各种时长参数，可以看到有十几个节点，其中关键时长达到七个。这些时长涉及多方，比如骑手（接 – 到 – 取 – 送）、商户（出餐）、用户（交付），要经历室内室外的场景转换，因此挑战性非常高。对于 ETA 建模，不光是简单一个时间的预估，更需要的是全链路的时间预估，同时更需要兼顾”单量 – 运力 – 用户转化率”转化率之间的平衡。配送 ETA 的演变包括了数据、特征层面的持续改进，也包括了模型层面一路从 LR–XGB–FM–DeepFM– 自定义结构的演变。



ETA 的探索与演变

具体 ETA 在整个配送业务中的位置及配送业务的整体机器学习实践，请参看《机器学习在美团配送系统的实践：用技术还原真实世界》。

2. 业务流程迭代中的模型改进

2.1 基础模型迭代及选择

与大部分 CTR 模型的迭代路径相似，配送 ETA 模型的业务迭代经历了 LR–> 树模型 –>Embedding–>DeepFM–> 针对性结构修改的路径。特征层面也进行不断迭代和丰富。

- 模型维度从最初考虑特征线性组合，到树模型做稠密特征的融合，到 Embedding 考虑 ID 类特征的融合，以及 FM 机制低秩分解后二阶特征组合，最终通

过业务指标需求，对模型进行针对性调整。

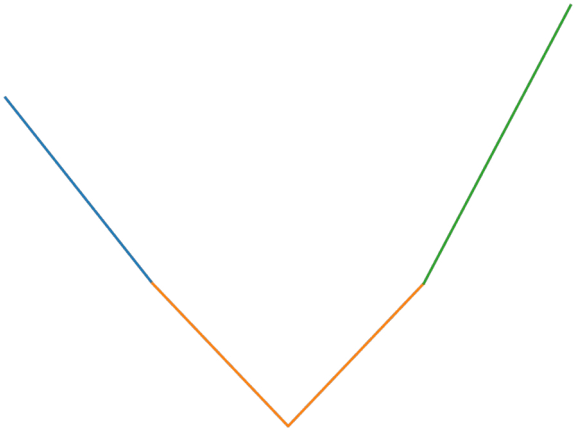
- 特征维度逐步丰富到用户画像 / 骑手画像 / 商家画像 / 地址特征 / 轨迹特征 / 区域特征 / 时间特征 / 时序特征 / 订单特征等维度。

目前版本模型在比较了 Wide&Deep、DeepFM、AFM 等常用模型后，考虑到计算性能及效果，最终选择了 DeepFM 作为初步的 Base 模型。整个 DeepFM 模型特征 Embedding 化后，在 FM（Factorization Machine）基础上，进一步加入 deep 部分，分别针对稀疏及稠密特征做针对性融合。FM 部分通过隐变量内积方式考虑一阶及二阶的特征融合，DNN 部分通过 Feed-Forward 学习高阶特征融合。模型训练过程中采取了 Learning Decay/Clip Gradient/ 求解器选择 /Dropout/ 激活函数选择等，在此不做赘述。

2.2 损失函数

在 ETA 预估场景下，准时率及置信度是比较重要的业务指标。初步尝试将 Square 的损失函数换成 Absolute 的损失函数，从直观上更为切合 MAE 相比 ME 更为严苛的约束。在适当 Learning Decay 下，结果收敛且稳定。

同时，在迭代中考虑到相同的 ETA 承诺时间下，在前后 N 分钟限制下，早到 1min 优于晚到 1min，损失函数的设计希望整体的预估结果能够尽量前倾。对于提前部分，适当降低数值惩罚。对于迟到部分，适当增大数值惩罚。进行多次调试设计后，最终确定以前后 N 分钟以及原点作为 3 个分段点。在原先 absolute 函数优化的基础上，在前段设计 1.2 倍斜率 absolute 函数，后段设计 1.8 倍斜率 absolute 函数，以便让结果整体往中心收敛，且预估结果更倾向于提前送达，对于 ETA 各项指标均有较大幅度提升。



损失函数

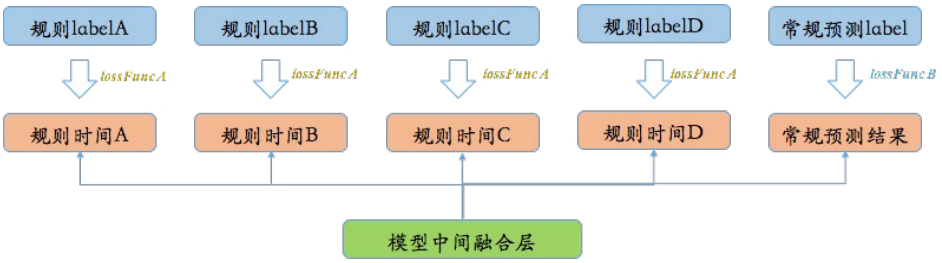
2.3 业务规则融入模型

目前的业务架构是”模型 + 规则”，在模型预估一个 ETA 值之后，针对特定业务场景，会有特定业务规则时间叠加以满足特定场景需求，各项规则由业务指标多次迭代产生。这里产生了模型和规则整体优化的割裂，在模型时间和规则时间分开优化后，即模型训练时并不能考虑到规则时间的影响，而规则时间在一年之中不同时间段，会产生不同的浮动，在经过一段时间重复迭代后，会加大割裂程度。

在尝试了不同方案后，最终将整体规则写入到了 TF 模型中，在 TF 模型内部调整整体规则参数。

- 对于简单的 (a*b+c)*d 等规则，可以将规则逻辑直接用 TF 的 OP 算子来实现，比如当 b、d 为定值时，则 a、c 为可学习的参数。
- 对于过于复杂的规则部分，则可以借助一定的模型结构，通过模型的拟合来代替，过多复杂 OP 算子嵌套并不容易同时优化。

通过调节不同的拟合部分及参数，将多个规则完全在 TF 模型中实现。最终对业务指标具备很大提升效果，且通过对部分定值参数的更改，具备部分人工干涉模型能力。



多目标补时结构

在这里，整体架构就简化为多目标预估的架构，这里采用多任务架构中常用的 Shared Parameters 的结构，训练时按比例采取不同的交替训练策略。结构上从最下面的模型中间融合层出发，分别在 TF 内实现常规预测结构及多个规则时间结构，而其对应的 Label 则仍然从常规的历史值和规则时间值中来，这样考虑了以下几点：

- 模型预估时，已充分考虑到规则对整体结果的影响（例如多个规则的叠加效应），作为整体一起考虑。
- 规则时间作为辅助 Label 传入模型，对于模型收敛及 Regularization，起到进一步作用。
- 针对不同的目标预估，采取不同的 Loss，方便进行针对性优化，进一步提升效果。

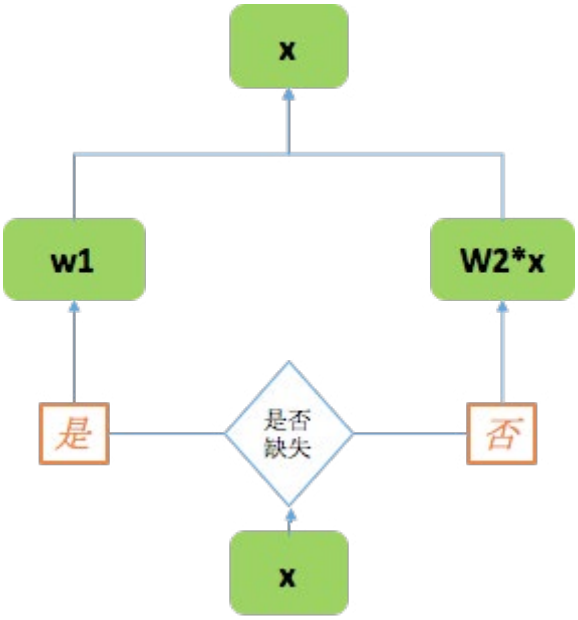
模型结构在进行预估目标调整尝试中：

- 尝试过固定共享网络部分及不固定共享部分参数，不固定共享参数效果明显。
- 通常情况下激活函数差异不大，但在共享层到独立目标层中，不同的激活函数差异很大。

2.4 缺失值处理

在模型处理中，特征层面不可避免存在一定的缺失值，而对于缺失值的处理，完全借鉴了《美团“猜你喜欢”深度学习排序模型实践》文章中的方法。对于特征 x 进入 TF 模型，进行判断，如果是缺失值，则设置 w1 参数，如果不是缺失值则进入模型数值为 w2*x，这里将 w1 和 w2 作为可学习参数，同时放入网络进行训练。以此

方法来代替均值 / 零值等作为缺失值的方法。



缺失值处理

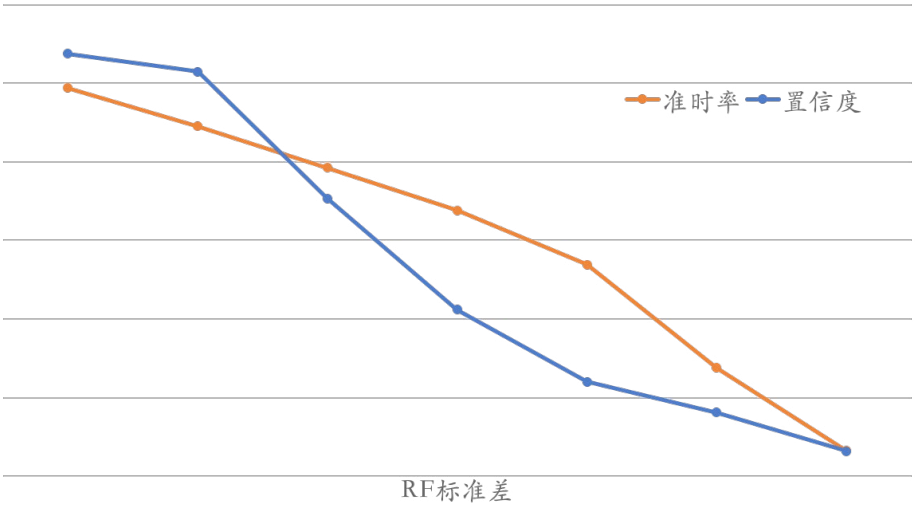
3. 长尾问题优化

3.1 模型预估结果 + 长尾规则补时

基础模型学习的是整体的统计分布，但对于一些长尾情形的学习并不充分，体现在长尾情形下预估时间偏短（由于 ETA 拥有考核骑手的功能，预估偏短对骑手而言意味着很大的伤害）。故将长尾拆解成两部分来分析：

- 业务长尾，即整体样本分布造成的长尾。主要体现在距离、价格等维度。距离越远，价格越高，实际送达时间越长，但样本占比越少，模型在这一部分上的表现整体都偏短。
- 模型长尾，即由于模型自身对预估值的不确定性造成的长尾。模型学习的是整体的统计分布，但不是对每个样本的预估都有“信心”。实践中采用 RF 多棵决策树输出的标准差来衡量不确定性。RF 模型生成的决策树是独立的，每棵

树都可以看成是一个专家，多个专家共同打分，打分的标准差实际上就衡量了专家们的“分歧”程度（以及对预估的“信心”程度）。从下图也可以看出来，随着 RF 标准差的增加，模型的置信度和准时率均在下降。



模型长尾因子

在上述拆解下，采用补时规则来解决长尾预估偏短的问题：长尾规则补时为 < 业务长尾因子，模型长尾因子 > 组合。其中业务长尾因子为距离、价格等业务因素，模型长尾因子为 RF 标准差。最终的 ETA 策略即为模型预估结果 + 长尾规则补时。

4. 工程开发实践

4.1 训练部分实践

整体训练流程

对于线下训练，采取如下训练流程：

Spark 原始数据整合 -> Spark 生成 TFRecord -> 数据并行训练 -> Tensor-Flow Serving 线下 GPU 评估 -> CPU Inference 线上预测

整个例行训练亿级数据多轮 Epoch 下流程持续约 4 小时，其中 TF 训练中，考虑到 TF 实际计算效率并不是很高，有很大比例在数据 IO 部分，通过 Spark 生成

TFRecord 部分，在此可将速度加速约 3.6 倍。而在数据并行训练部分，16 卡内的并行度扩展基本接近线性，具备良好的扩展性。由于 PS 上参数量并未达到单机无法承受，暂时未对参数在 PS 上进行切分。Serving 线下 GPU 评估部分，是整个流程中的非必需项，虽然在训练过程中 Chief Worker 设置 Valid 集合可有一定的指标，但对全量线下，通过 Spark 数据调用 Serving GPU 的评估具备短时间内完成全部流程能力，且可以指定大量复杂自定义指标。

数据并行训练方式

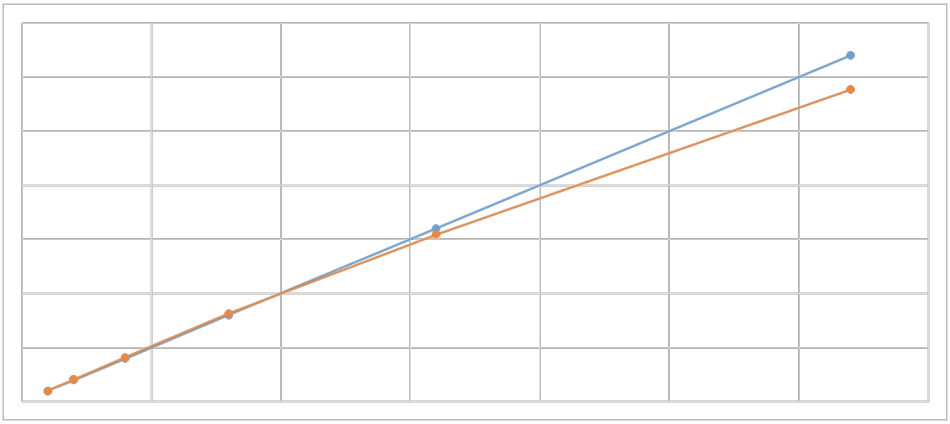
整个模型的训练在美团的 AFO 平台上进行，先后尝试分布式方案及单机多卡方案。考虑到生产及结果稳定性，目前线上模型生产采用单机多卡方案进行例行训练。

- 分布式方案：

采用 TF 自带的 PS-Worker 架构，异步数据并行方式，利用 tf.train.MonitoredTrainingSession 协调整个训练过程。整个模型参数存储于 PS，每个 Step 上每个 Worker 拉取数据进行数据并行计算，同时将梯度返回，完成一次更新。目前的模型单 Worker 吞吐 1~2W/s，亿级数据几轮 Epoch 耗时在几小时内完成。同时测试该模型在平台上的加速比，大约在 16 块内，计算能力随着 Worker 数目线性增加，16 卡后略微出现分离。在目前的业务实践中，基本上 4-6 块卡可以短时间内完成例行的训练任务。

- 单机多卡方案：

采用 PS-Worker 的方案在平台上具备不错的扩展性，但是也存在一定的弊端，使用 RPC 的通讯很容易受到其他任务的影响，整个的训练过程受到最慢 Worker 的影响，同时异步更新方式对结果也存在一定的波动。对此，在线上生产中，最终选取单机多卡的方案，牺牲一定的扩展性，带来整体训练效果和训练速度的稳定性。单机多卡方案采取多 GPU 手动指定 OP 的 Device，同时在各个 Device 内完成变量共享，最后综合 Loss 与梯度，将 Grad 更新到模型参数中。



加速比曲线

TF 模型集成预处理

模型训练过程中，ID 类特征低频过滤需要用到 Vocab 词表，连续型特征都需要进行归一化。这里会产生大量的预处理文件，在线下处理流程中很容易在 Spark 中处理成 Libsvm 格式，然后载入到模型中进行训练。但是在线上预测时，需要在工程开发端载入多个词表及连续型特征的归一化预处理文件（avg/std 值文件等），同时由于模型是按天更新，存在不同日期版本的对齐问题。

为了简化工程开发中的难度，在模型训练时，考虑将所有的预处理文件写入 TF 计算图之中，每次在线预测只要输入最原始的特征，不经过工程预处理，直接可得到结果：

- 对于 ID 类特征，需要进行低频过滤，然后制作成词表，TF 模型读入词表的 list_arr，每次 inference 通过 ph_vals，得到对应词表的 ph_idx。

```
tf_look_up = tf.constant(list_arr, dtype=tf.int64)
table = tf.contrib.lookup.HashTable(tf.contrib.lookup.
KeyValueTensorInitializer(tf_look_up, idx_range), 0)
ph_idx = table.lookup(ph_vals) + idx_bias
```

对于连续型特征，在 Spark 处理完得到 avg/std 值后，直接写入 TF 模型计算图中，作为 constant 节点，每个 ph_in 经过两个节点，得到相应 ph_out。

```
constant_avg = tf.constant(feat_avg, dtype=tf.float32, shape=[feat_dim],
name="avg")
constant_std = tf.constant(feat_std, dtype=tf.float32, shape=[feat_dim],
name="std")
ph_out = (ph_in - constant_avg) / constant_std
```

4.2 TF 模型线上预测

配送机器学习平台内置了模型管理平台，对算法训练产出的模型进行统一管理和调度，管理线上模型所用的版本，并支持模型版本的切换和回退，同时也支持节点模型版本状态的管理。

ETA 使用的 DeepFM 模型用 TensorFlow 训练，生成 SavedModel 格式的模型，需要模型管理平台支持 Tensorflow SavedModel 格式。

实现方案 S 线上服务加载 TensorFlow SavedModel 模型有多种实现方案：

- 自行搭建 TensorFlow Serving CPU 服务，通过 gRPC API 或 RESTful API 提供服务，该方案实现比较简单，但无法与现有的基于 Thrift 的模型管理平台兼容。
- 使用美团 AFO GPU 平台提供的 TensorFlow Serving 服务。
- 在模型管理平台中通过 JNI 调用 TensorFlow 提供的 Java API TensorFlow Java API，完成模型管理平台对 SavedModel 格式的支持。

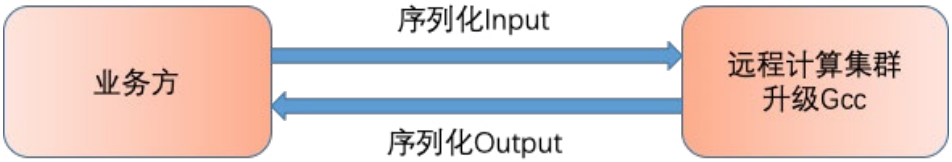
最终采用 TensorFlow Java API 加载 SavedModel 在 CPU 上做预测，测试 batch=1 时预测时间在 1ms 以内，选择方案 3 作为实现方案。

远程计算模式

TensorFlow Java API 的底层 C++ 动态链接库对 libstdc++.so 的版本有要求，需要 GCC 版本不低于 4.8.3，而目前线上服务的 CPU 机器大部分系统为 CentOS 6, 默认自带 GCC 版本为 4.4.7。如果每台线上业务方服务器都支持 TensorFlow SavedModel 本地计算的话，需要把几千台服务器统一升级 GCC 版本，工作量比较大而且可能会产生其他风险。

因此，我们重新申请了几十台远程计算服务器，业务方服务器只需要把 Input 数

据序列化后传给 TensorFlow Remote 集群，Remote 集群计算完后再将 Output 序列化后返回给业务方。这样只需要对几十台计算服务器升级就可以了。



在线序列化

线上性能

模型上线后，支持了多个业务方的算法需求，远程集群计算时间的 TP99 基本上在 5ms 以内，可以满足业务方的计算需求。

接口	调用量	调用量(客户端)	可用率	QPS	TP90	TP95	TP99
all					2	3	3
					2	2	2
					3	3	4

线上效果

总结与展望

模型落地并上线后，对业务指标带来较大的提升。后续将会进一步根据业务优化模型，进一步提升效果：

- 将会进一步丰富多目标学习框架，将取餐、送餐、交付、调度等整个配送生命周期内的过程在模型层面考虑，对订单生命周期内多个目标进行建模，同时提升模型可解释性。
- 模型融合特征层面的进一步升级，在 Embedding 以外，通过更多的 LSTM/CNN/ 自设计结构对特征进行更好的融合。
- 特征层面的进一步丰富。

作者简介

基泽，美团点评技术专家，目前负责配送算法策略部机器学习组策略迭代工作。
周越，2017 年加入美团配送事业部算法策略组，主要负责 ETA 策略开发。
显杰，美团点评技术专家，2018 年加入美团，目前主要负责配送算法数据平台深度学习相关的研发工作。

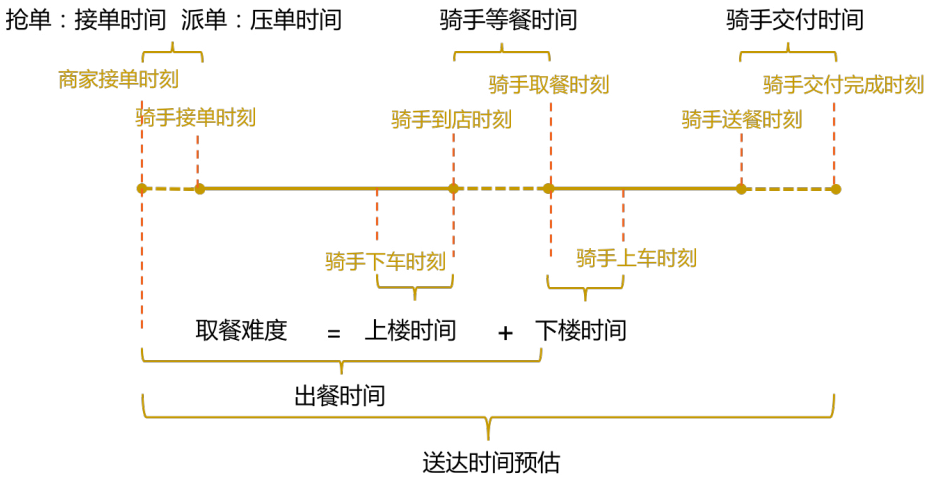
配送交付时间轻量级预估实践

基泽　闫聪

1. 背景

可能很多同学都不知道，从打开美团 App 点一份外卖开始，然后在半小时内就可以从骑手小哥手中拿到温热的饭菜，这中间涉及的环节有多么复杂。而美团配送技术团队的核心任务，就是将每天来自祖国各地的数千万份订单，迅速调度几十万骑手小哥按照最优路线，并以最快的速度送到大家手中。

在这种场景下，骑手的交付时间，即骑手到达用户附近下车后多久能送到用户手中　，就是一个非常重要的环节。下图是一个订单在整个配送链路的时间构成，时间轴最右部分描述了交付环节在整个配送环节中的位置。交付时间衡量的是骑手送餐时的交付难度，包括从骑手到达用户楼宇附近，到将餐品交付到用户手中的整个时间。



交付时间的衡量是非常有挑战的一件事，因为骑手在送餐交付到用户手中时会碰到不同的问题，例如：骑手一次送餐给楼宇内多个用户，骑手对于特定楼宇寻址特别困难，骑手在交付楼宇附近只能步行，老旧小区没有电梯，写字楼无法上楼，或者难

以等到电梯等等。交付时间预估需要具备刻画交付难度的能力，在定价、调度等多个场景中被广泛使用。例如根据交付难度来确定是否调节骑手邮资，根据交付难度来确定是否调节配送运单的顺序，从而避免超时等等。总的来说，交付时间预估是配送业务基础服务的重要一环。

但是，交付时间预估存在如下的困难：

- 输入信息较少，且多为非数值型数据，目前能够被用来预估的仅有如下维度特征：交付地址、交付点的经纬度、区域、城市，适配常规机器学习模型需要重新整理且容易丢失信息。
- 计算性能要求很高。由于是基础服务，会被大量的服务调用，需要性能 TP99 保证在 10ms 以内，整个算法平均响应时间需要控制在 5ms 内，其中包括数据处理及 RPC 的时间。且该标准为 CPU 环境下的性能要求，而非 GPU 下的性能要求。

	Name	Total	Failure	Failure%	Log View	Max	Avg	90Line	95Line	99Line	99.9Line
[:: show ::]	TOTAL	1,225,797	0	0.0000%	L S	85.0	3.8	0.0	0.0	0.0	0.0
[:: show ::]	V3:1	595,184	0	0.0000%	L S	80.0	4.3	8.1	8.8	10.1	23.7
[:: show ::]	V3P2:1	397,471	0	0.0000%	L S	85.0	3.7	5.0	5.1	6.0	18.1
[:: show ::]	V2:1	233,142	0	0.0000%	L S	45.0	2.5	3.4	3.7	4.0	6.6

上图为部分版本所对应的性能，平响时间均在 5ms 内，TP99 基本在 10ms 内

总结起来，交付时间预估的问题，在于需要使用轻量级的解决方案来处理多种数据形式的非数值型数据，并提取有效信息量，得到相对准确的结果。在相同效果的前提下，我们更倾向于性能更优的方案。

在本文中，我们介绍了交付时间预估迭代的三个版本，分别为基于地址结构的树模型、向量召回方案以及轻量级的 End-to-End 的深度学习网络。同时介绍了如何在性能和指标之间取舍，以及模型策略迭代的中间历程，希望能给从事相关工作的同学们有所启发和帮助。

2. 技术迭代路径

首先，在交付时间预估的技术迭代上，我们主要经历了三个大版本的改动，每一

版本在 5ms 计算性能的约束下，追求轻量化的解决方案，在兼顾提升效果的基础上，不显著增加性能的消耗。

本章节分别叙述了 3 个模型的迭代路径，包括技术选型、关键方案及最终效果。

2.1 树模型

技术选型

最早也是最容易被考虑到的是利用规则，核心思路是利用树结构衡量地址相似性，尽可能在相似的交付地址上积聚结构化数据，然后利用局部的回归策略，得到相对充裕的回归逻辑，而未能达到回归策略要求的则走兜底的策略。

为了快速聚积局部数据，树模型是一个较为合适的解决方案，树的规则解析能够有效地聚集数据，同时一个层级并不深的树，在计算速度上，具备足够的优势，能够在较短的时间内，得到相对不错的解决方案。

观察用户填写地址以及联系实际中地址的层级结构，不难发现，一个地址可以由四级结构组成：地址主干词（addr）、楼宇号（building）、单元号（unit）、楼层（floor）。其中的地址主干词在实际中可能对应于小区名或者学校名等地标名称。例如望京花园 1 号楼 2 单元 5 楼，解析为（望京花园，1 号楼，2 单元，5 楼）。通过分析，实际交付时长与楼层高低呈正相关关系，且不同交付地址的交付时长随楼层增加的变化幅度也有所区别，所以可以使用线性回归模型拟合楼层信息和交付时长的关系，而地址主干词、楼宇号、单元号作为其层级索引。但用户填写的地址中并不一定包含完整的四级结构，就会存在一定比例的缺失，所以利用这样的层级结构构建成一棵树，然后充分利用上一层已知的信息进行预估。预测时，只需根据结点的分支找到对应的模型即可，如果缺失，使用上一层结构进行预测。对于没有达到训练模型要求数据量的地址，使用其所在的区域平均交付时长作为交付时长的预估结果，这部分也可以看作区域信息，作为树结构的根节点。

迭代路径

整体的思路是基于离散特征训练树模型，在树的结点上基于楼层训练线性回归模型。树结点训练分裂规则：（1）数据量大于阈值；（2）分裂后 MAE（平均绝对误差）的

和小于分裂前。考虑到数据的时效性，采用加权线性回归增加近期数据的权重。

2.2 树模型 + 向量召回方案

技术选型

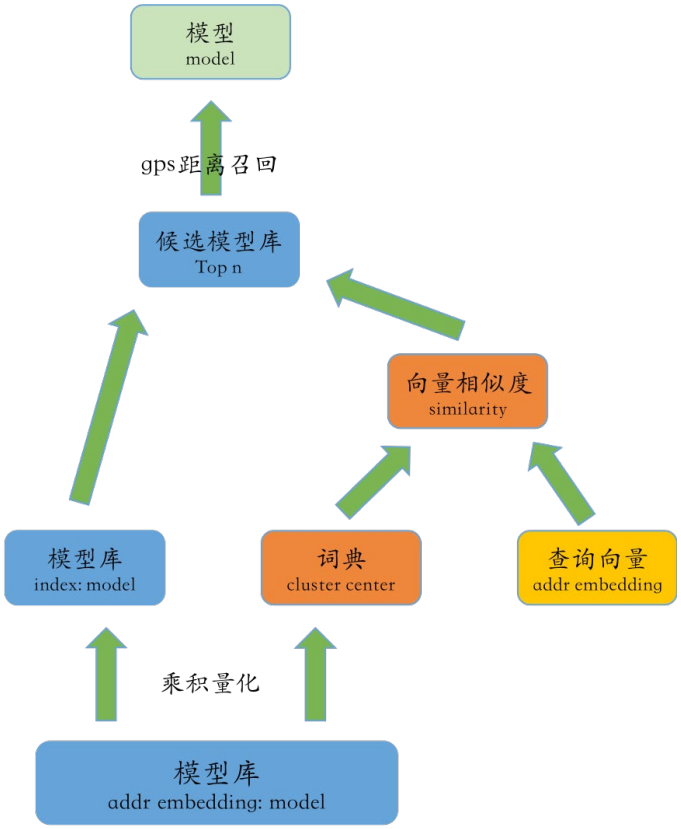
向量召回作为主流的召回方案之一，被业界广泛使用，在使用 LSH、PQ 乘积量化等常用开源工具基础上，高维向量召回性能通常在毫秒量级。

而从算法上考虑，树模型中 NLP 地址解析结果能够达到模型使用要求的仅为 70%+，剩余 20%+ 的地址无法通过训练得到的模型从而只能走降级策略。利用高维向量来表达语义相似性，即利用向量来表达地址相似性，从而用相似数据对应的模型来替代相似但未被召回数据，将地址主干词进行 Embedding 后，摆脱主干词完全匹配的低鲁棒性。

例如，在地址上可能会出现【7 天酒店晋阳街店】数据量比较充足，但【7 天连锁酒店太原高新区晋阳街店】数据量不充足从而无法训练模型的案例，这可能是同一个交付位置。我们希望尽可能扩大地址解析的成功率。

迭代路径

整个技术路径较为清晰简单，即利用 Word2Vec 将 charLevel 字符进行 Embedding，获得该地址的向量表示，并且融入 GPS 位置信息，设计相应兜底策略。



向量召回方案决策路径

最终效果

比较大地提升了整体策略的召回率，提升了 12.20pp，对于未被上一版本树模型召回的地址，指标有了显著的提升，其中 ME 下降 87.14s，MAE 下降 38.13s，1min 绝对偏差率减小 14.01pp，2min 绝对偏差率减小 18.45pp，3min 绝对偏差率减小 15.90pp。

2.3 End-to-End 轻量化深度学习方案

技术选型

在树模型的基础上，迭代到向量召回方案，整个模型的召回率有了较大幅度的增长，但仍然不是 100%。分析发现，召回率提升的障碍在于 NLP 对于地址解析的覆盖率。

整个方案的出发点：

从模型复杂度考虑，同样仅仅使用地址信息的话，在提升模型 VC 维的基础上，使用其他的模型方案至少可以持平树模型的效果，如果在这基础上还能融入其他信息，那么对于原模型的基线，还能有进一步的提升。

考虑到不仅仅需要使用地址数据，同时需要使用 GPS 数据、大量 ID 类的 Embedding，对于各类非数值类型的处理灵活性考虑，采用深度学习的方案，来保证多源且多类型特征能在同一个优化体系下优化学习。

工程上需要考虑的点：

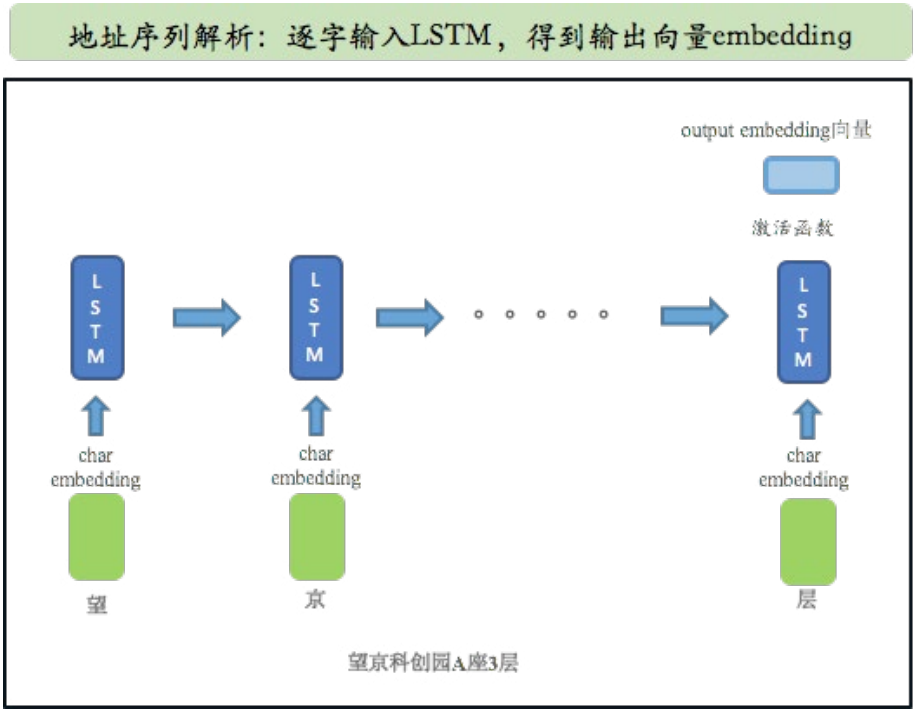
交付模型作为基础模型，被广泛应用在路径构造、定价、ETA 等各个业务中频繁调用，在树模型版本中，对于性能的要求为平均响应时间 5ms，TP99 在 10ms 左右，本方案需要考虑沿袭原业务的性能，不能显著增加计算耗时。

交付模型的难点在于非数值型特征多，信息获取形式的多样化，当前的瓶颈并不在于模型的复杂度低。如果可以轻量地获取信息及融合，没必要对 Fusion 后的信息做较重的处理方案。

所以整体的设计思路为：利用深度学习融合非数值型特征，在简单 Fusion 的基础上，直接得到输出结构，对于组件的选择，尽可能选用 Flops 较低的设计。该设计背后意图是，在充分使用原始输入信息，在尽可能避免信息损失的基础上，将非数值型的信息融入进去。并将信息充分融合，直接对接所需要的目标。而选用的融合组件结构尽可能保证高性能，且具备较高学习效率。这里分别针对地址选用了较为 Robust 的 LSTM，针对 GPS 选用了自定义的双线性 Embedding，兼顾性能和效果。

迭代路径

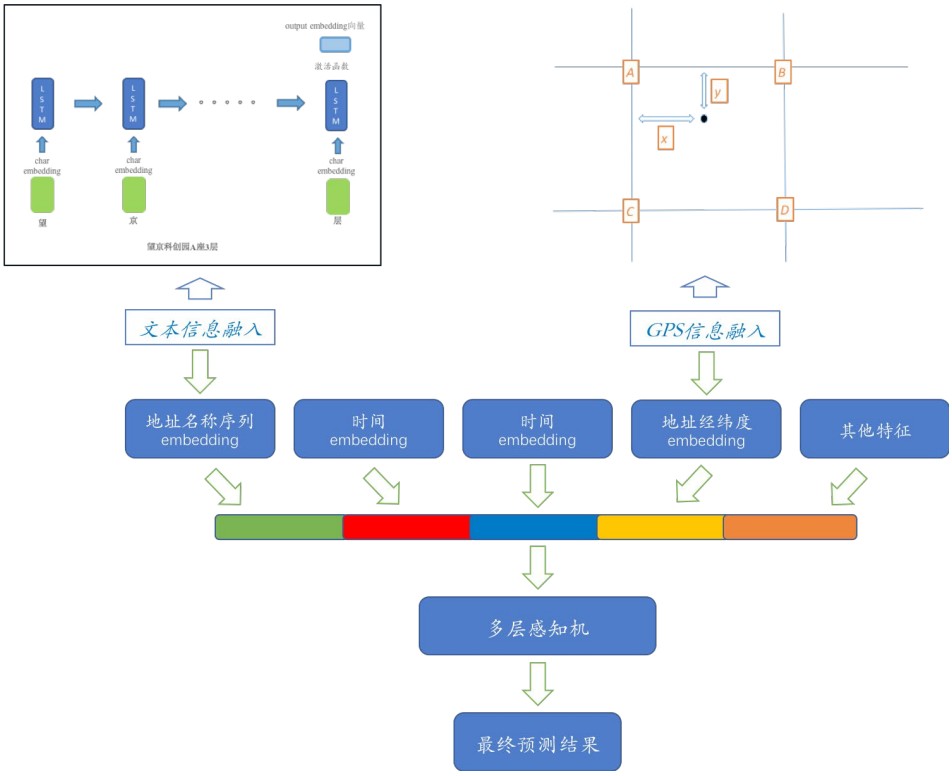
开始采用端到端的深度学习模型，这里首先需要解决的是覆盖率问题，直接采用 LSTM 读取 charLevel 的地址数据，经过全连接层直接输出交付时间。作为第一版本的数据，该版本数据基本持平树模型效果，但对于树模型未召回的 20% 数据，有了较大的提升。



地址信息输入 charLevel 模型

在采用 charLevel 的地址奏效后，我们开始采用加入用户地址 GPS 的信息，由于 GPS 为经纬度信息，非数值型数据，我们使用一种基于地理位置格点的双线性插值方法进行 Embedding。该方案具备一定的扩展性，对不同的 GPS 均能合理得到 Embedding 向量，同时具备平滑特性，对于多对偏移较小的 GPS 点能够很好的进行支持。

最终方案将地址 Embedding 后，以及 GPS 点的 Embedding 化后，加入下单时间、城市 ID、区域 ID 等特征后，再进行特征融合及变换，得到交付模型的时间预估输出。整个模型是一个端到端的训练，所有参数均为 Trainable。



模型结构示意图

扩展组件

在证实 End-to-End 路径可行后，我们开始进行扩展组件建设，包括自定义损失函数、数据采样修正、全国模型统一等操作，得到一系列正向效果，并开发上线。

特征重要性分析

对于深度学习模型，我们有一系列特征重要性评估方案，这里采用依次进行 Feature Permutation 的方式，作为评估模型特征重要性的方式。

考虑 GPS 经纬度和用户地址存在较大程度的信息重叠，评估结果如下。Shuffle 后，用户地址的特征重要性高于 GPS 经纬度的特征重要性。加入 GPS 后 ME 下降不如地址信息明显，主要是地址信息包含一定冗余信息（下文会分析），而其他信息的影响则可以忽略不计。

特征	特征重要排名
用户地址	1
GPS 经纬度	2
其他特征	……

注：在配送的其他案例中，商户 GPS 的经纬度重要性 >> 用户地址重要性 >> 用户 GPS 的经纬度重要性，该特征重要性仅仅为本案例特征重要性排序，不同学习目标下可能会有比较明显差别。

最终效果

End-to-End 深度学习模型的最终效果较为显著：对于树模型及向量召回方案的最痛点，覆盖率得到彻底解决，覆盖率提升到 100%。ME 下降 4.96s，MAE 下降 8.17s，1min 绝对偏差率减小 2.38pp，2min 绝对偏差率减小 5.08pp，3min 绝对偏差率减小 3.46pp。同时，对于之前树模型及向量召回方案未能覆盖到的运单，提升则更为明显。

3. 模型相关分析

在整个技术迭代的过程中，由于整个解决方案对于性能有着较为苛刻的要求，需要单独对方案性能进行分析。本章节对向量召回方案及深度学习方案进行了相应的性能分析，以便在线下确认性能指标，最终保证上线后性能均达到要求。下文分别着重介绍了向量匹配的工具 Faiss 以及 TensorFlow Operation 算子的选取，还有对于整体性能的影响。

同时对比 End-to-End 生成向量与 Word2Vec 生成向量的质量区别，对于相关项目具备一定的借鉴意义。

3.1 向量召回性能

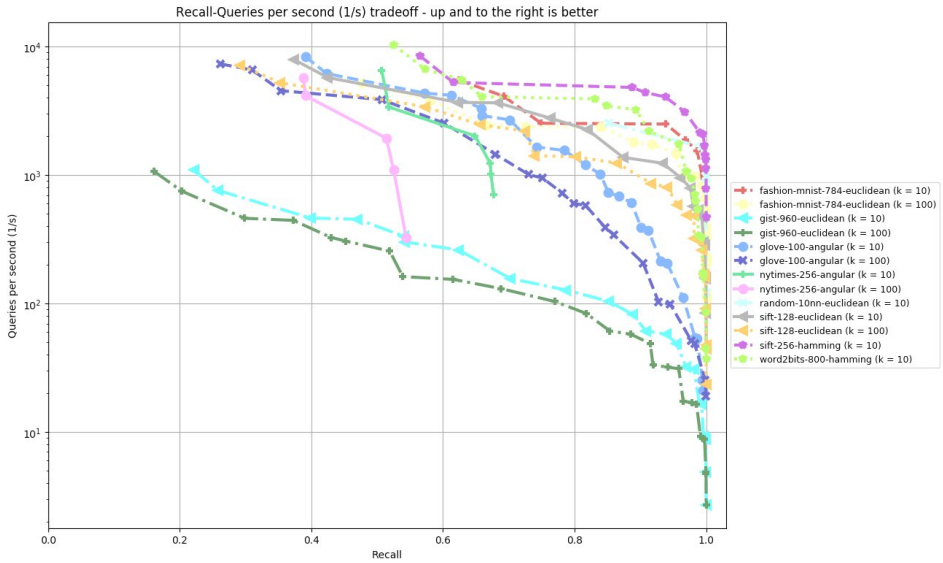
最近邻搜索（Nearest Neighbor Search）指的是在高维度空间内找到与查询点最近点的问题。在数据样本小的时候，通过线性搜索就能满足需求，但随着数据量的增加，如达到上百万、上亿点时候，倾向于将数据结构化表示来更加精确地表达向量信息。

此时近似最近邻搜索 ANN（Approximate Nearest Neighbor）是一个可参考的技术，它能在近似召回一部分之后，再进行线性搜索，平衡效率和精度。目前大体上有以下 3 类主流方法：基于树的方法，如 K-D 树等；基于哈希的方法，例如 LSH；基于矢量量化的方法，例如 PQ 乘积量化。在工业检索系统中，乘积量化是使用较多的一种索引方法。

针对向量召回的工具，存在大量的开源实现，在技术选型的过程中，我们参照 ANN-Benchmarks 以及 Erikbern/ANN-Benchmarks 中的性能评测结果。在众多 ANN 相关的工具包内，考虑到性能、内存、召回精度等因素，同时可以支持 GPU，在向量召回方案的测试中，选择以 Faiss 作为 Benchmark。

Faiss 是 FaceBook 在 2017 年开源的一个用于稠密向量高效相似性搜索和密集向量聚类的库，能够在给定内存使用下，在速度和精度之间权衡。可以在提供多种检索方式的同时，具备 C++/Python 等多个接口，也对大部分算法支持 GPU 实现。

下图为 Faiss 测评曲线：



交付时间模型召回的性能测试如下，可以达到性能需求。

- 召回候选集数量：8W 条向量【由于采用了 GPS 距离作为距离限制，故召回

测试采用 8W 数量级】

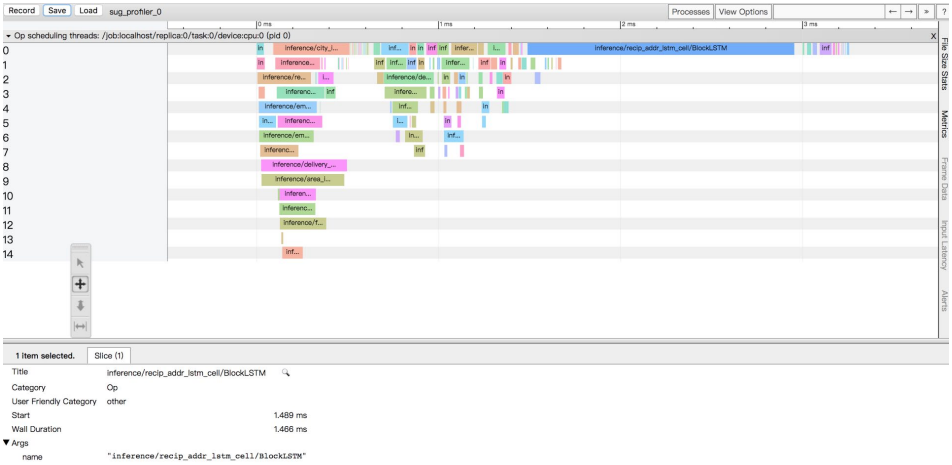
- 测试机器：Mac 本机 CPU【CPU 已满足性能，故不再测试 GPU】

单位 (ms)	最近邻数量	mean	90分位数	max	建索引
IndexFlatL2 dimention=128	1	2.63	2.73	10.83	12.80
	100	2.66	2.69	63.82	
	1000	3.14	3.27	21.54	
IndexIVFFlat L2距离 nlist=100 dimention=128 nprobe=10	1	0.65	0.67	5.59	234.52
	100	0.70	0.72	5.72	
	1000	1.06	1.08	11.95	
IndexIVFFlat Cosine距离 nlist=100 dimention=128 nprobe=10	1	0.36	0.38	5.47	153.99
	100	0.39	0.40	14.41	
	1000	0.61	0.63	5.99	
IndexFlatIP dimention=128	1	2.37	2.20	9.45	17.53
	100	2.92	3.08	17.59	
	1000	3.25	3.29	28.12	

3.2 序列模块性能

在 TensorFlow 系统中，以 C API 为界限，将系统划分为【前端】和【后端】两个子系统，前端扮演 Client 角色，完成计算图的构造，然后由 Protobuf 发送给后端启动计算图计算。计算图的基础单元是 OP，代表的是某种操作的抽象。在 TensorFlow 中，考虑到实现的不同，不同 OP 算子的选择，对于计算性能具有较大影响。

为了评测深度学习交付模型的性能瓶颈，首先对整个模型进行 Profile，下图即为 Profile 后的 Timeline，其中整个计算大部分消耗在序列模块处理部分，即下图中的蓝色部分。故需要对序列模块的计算性能进行 OP 算子的加速。



考虑到序列处理的需求，评估使用了 LSTM/GRU/SRU 等模块，同时在 TensorFlow 中，LSTM 也存在多种实现形式，包括 BasicLSTMCell、LSTMCell、LSTMBlockCell、LSTMBlockFusedCell 和 CuDNNLSTM 等实现，由于整个交付模型运行在 CPU 上，故排除 CuDNNLSTM，同时设置了全连接层 FullyConnect 加入评估。

从评估中可以发现，全连接层速度最快，但是对于序列处理会损失 2.3pp 效果，其余的序列模型效果差异不大，但不同的 OP 实现对结果影响较大。原生的 BasicLSTM 性能较差，contrib 下的 LSTMBlockFusedCell 性能最好，GRU/SRU 在该场景下未取得显著优势。

这是 LSTMBlockFusedCell 的官方说明，其核心实现是将 LSTM 的 Loop 合并为一个 OP，调用时候整个 Timeline 上更为紧凑，同时节约时间和内存：

> This is an extremely efficient LSTM implementation, that uses a single TF op for the entire LSTM. It should be both faster and more memory-efficient than LSTMBlockCell defined above.

以下是序列模块的性能测试：

- 环境：Tensorflow1.10.0，CentOS 7。
- 测试方法：CPU inference 1000 次，取最长的地址序列，求平均时间。

- 结论：LSTMBlockFused 实现性能最佳。【FullyConnect 性能最快，但对性能有损失】

注：在评估中，不仅仅包括了序列模型，也包括了其他功能模块，故参数量及模型大小按照总体模型而言

lstm结构OP	时间(ms)	FLOPs	可训练参数量	模型大小(MB)	效果差异
Fully Connect	1.18	27.83M	7.00M	29.1	−2.3pp
SRU	4.00	27.96M	7.06M	29.4	差异不显著
GRU Block	3.64	28.02M	7.10M	29.6	差异不显著
GRU	4.44	28.02M	7.10M	29.6	差异不显著
LSTMBlockFused	2.48	28.09M	7.13M	29.7	差异不显著
LSTM Block	4.34	28.09M	7.13M	29.7	差异不显著
LSTM	4.85	28.09M	7.13M	29.7	差异不显著
BasicLSTM	4.92	28.09M	7.13M	29.7	差异不显著

3.3 向量效果分析

将向量召回与深度学习模型进行横向比较，二者中间过程均生成了高维向量。不难发现，二者具备一定的相似性，这里就引发了我们的思考：

相较于向量召回，深度学习模型带来的提升主要来自于哪里?

有监督的 lstm 学习到的 Embedding 向量与自监督的 Word2Vec 得到的向量在地址相似性计算中有多大差别，孰优孰劣?

首先，我们分析第一个问题，End-to-End 模型提升主要来自哪里?

ME	MAE	1min绝对偏差率	2min绝对偏差率	3min绝对偏差率
End-to-End 模型 − Word2Vec 模型	4.14	−0.45	−0.31%	0.05%

我们直接将 End-to-End 模型得到的 char embedding 抽取出来，直接放入到 Word2Vec 方案内，取代 Word2Vec 生成的 char embedding，再进行向量召回的评估。结果如下表所示，单独抽取出来的 char embedding 在向量召回方案中，表现与 Word2Vec 生成的向量基本一致，并没有明显的优势。

注：

- 1min 绝对偏差率定义：|pred−label|<=60s
- 2min 绝对偏差率定义：|pred−label|<=120s
- 3min 绝对偏差率定义：|pred−label|<=180s

此时的变量有 2 个方面：

- a）对于 charLevel 地址的学习结构不同，一个为 Word2Vec，一个为 LSTM
- b）输入信息的不同，Word2Vec 的信息输入仅仅为地址主干词，而 End−to−End 的信息输入则包括了地址主干词、地址附属信息、GPS 等其他信息。

注：

- 完整地址：卓玛护肤造型（洞庭湖店）（洞庭湖路与天山路交叉路口卓玛护肤造型）
- 地址主干词：卓玛护肤造型店
- 地址附属信息：（洞庭湖店）（洞庭湖路与天山路交叉路口卓玛护肤造型）

为了排除第二方面的因素，即 b 的因素，使用地址主干词作为输入，而不用地址附属信息和其他模型结构的输入，保持模型输入跟 Word2Vec 一致。在测试集上，模型的效果比完整地址有明显的下降，MAE 增大约 15s。同时将 char embedding 提取出来，取代 Word2Vec 方案的 char embedding，效果反而变差了。结合 2.3 节中的特征重要性，可知，深度学习模型带来的提升主要来自对地址中冗余信息（相较于向量召回）的利用，其次是多个新特征的加入。另外，对比两个 End−to−End 模型的效果，地址附属信息中也包含着对匹配地址有用的信息。

ME	MAE	1min绝对偏差率	2min绝对偏差率	3min绝对偏差率
End−to−End 模型 − Word2Vec 模型	−1.28	0.64	0.90%	0.85%

针对第二个问题，有监督的 End−to−End 学习到的 Embedding 向量，与自监督的 Word2Vec 得到的向量在地址相似性计算中有多大差别，孰优孰劣?

采用地址主干词代替完整地址，作为 End−to−End 模型的输入进行训练，其他

信息均保持不变。使用地址主干词训练得到的 Embedding 向量，套用到向量召回方案中。

从评估结果来看，对于不同的阈值，End-to-End 的表现差异相对 Word2Vec 较小。相同阈值下，End-to-End 召回率更高，但是效果不如 Word2Vec。

从相似计算结果看，End-to-End 模型会把一些语义不相关但是交付时间相近的地址，映射到同一个向量空间，而 Word2Vec 则是学习一个更通用的文本向量表示。

例如，以下两个交付地址会被认为向量距离相近，但事实上只是交付时间相近：南内环西街与西苑南路交叉口金昌盛国会 <=> 辰憬家园迎泽西大街西苑南路路口林香斋酒店

如果想要针对更为复杂的目标和引入更多信息，可以使用 End-to-End 框架；只是计算文本相似性，从实验结果看，Word2Vec 更好一些。同时，通过查看 Case 也可以发现，End-to-End 更关注结果相似性，从而召回一部分语义上完全不相关的向量。两个模型目标上的不同，从而导致了结果的差异。

4. 总结与展望

在本篇中，依次展示了在配送交付场景下的三次模型策略迭代过程，以及在较为苛刻性能要求限制下，如何用轻量化的方案不断提高召回率及效果。同时，对迭代过程中的性能进行简单的分析及衡量，这对相关的项目也具备一定的借鉴意义，最后对 Word2Vec 及 End-to-End 生成的向量进行了比较。

事实上，本文中提及的向量召回及深度学习融合非数值型特征的方案，已经在业界被广泛使用。但对于差异化的场景，本文仍具备一定的借鉴价值，特别是对于订单 – 骑手匹配、订单 – 订单匹配等非搜索推荐领域的场景化应用，以及 TF OP 算子的选用及分析、Embedding 生成方式带来的差异，希望能够给大家提供一些思路和启发。

5. 关联阅读

交付时间预估与 ETA 预估及配送其他业务关系：

- 交付时间预估是 ETA 预估中的重要一环，关于 ETA 预估，请参见《深度学习在美团配送 ETA 预估中的探索与实践》。
- 具体 ETA 在整个配送业务中的位置及配送业务的整体机器学习实践，请参看《机器学习在美团配送系统的实践：用技术还原真实世界》。

6. 作者简介

基泽，美团点评技术专家
闫聪，美团点评算法工程师

7. 招聘信息

美团配送 AI 团队支撑全球领先的即时配送网络——美团配送，并建立了行业领先的美团智能配送系统。美团配送 AI 团队主要来自一线互联网公司以及知名科研机构，研发实力和氛围卓越。目前美团配送业务仍处于快速发展期，新的场景、新的技术难题不断涌现，成长空间巨大。
美团配送 AI 团队现诚聘算法策略工程师、机器学习工程师和运筹优化工程师，欢迎有兴趣的小伙伴投递简历至：tech@meituan.com（邮件标题注明：美团配送 AI 团队）

ICDAR 2019 论文：自然场景文字定位技术详解

刘曦

自然场景文字定位是文字识别中非常重要的一部分。与通用的物体检测相比，文字定位更具挑战性，文字在长宽比、尺度和方向上有更大范围的变化。针对这些问题，本文介绍一种融合文字片段及金字塔网络的场景文字定位方法。该方法将特征金字塔机制应用到单步多框检测器以处理不同尺度文字，同时检测多个文字片段以及学习出文字片段之间 8-neighbor 连接关系，最后通过 8-neighbor 连接关系将文字片段连接起来，实现对不同方向和长宽比的文字定位。此外，针对文字通常较小特点，扩大检测网络中 backbone 模型深层特征图，以获得更好性能。

本文提出的方法已发表在文档分析与识别国际会议 ICDAR2019 (International Conference on Document Analysis and Recognition) 上，审稿人评论该方法为“As it is of more practical uses”，认可了它的实用性。

ICDAR 是由国际模式识别学会 (IAPR) 组织的专业会议之一，专注于文本领域的识别与应用。ICDAR 大会每两年举办一次，目前已发展成文字识别领域的旗舰学术会议。为了提高自然场景的文本检测和识别水平，国际文档分析和识别会议（ICDAR）于 2003 年设立了鲁棒文本阅读竞赛（“Robust Reading Competitions”）。至今已有来自 89 个国家的 3500 多支队伍参与。ICDAR 2019 将于今年 9 月 20-25 日在澳大利亚悉尼举办。 美团今年联合国内外知名科研机构和学者，提出了”中文门脸招牌文字识别”比赛（ICDAR 2019 Robust Reading Challenge onReading Chinese Text on Signboards）。

背景

自然场景图像中的文字识别已被广泛应用在现实生活中，例如拍照翻译，自动驾驶，图像检索和增强现实等，因此也有越来越多的专家学者对其进行研究。自然场景文字定位是指对场景图像中所有文本的精确定位，是自然场景文字识别中第一步也是最重要的一步。由于自然场景下文本颜色、大小、宽高比、字体、方向、光照条件和

背景等具有较大变化（如图 1），因此它是非常具有挑战性的。

图 1　自然场景文字图片

深度学习技术在物体识别和检测等计算机视觉任务方面已经取得了很大进展。许多最先进的基于卷积神经网络（CNN）的目标检测框架，如 Faster RCNN、SSD 和 FPN[1] 等，已被用来解决文本检测问题并且性能远超传统方法。

深度卷积神经网络是一个多层级网络结构，浅层特征图具有高分辨率及小感受野，深层特征图具有低分辨率及大感受野。具有小感受野的浅层特征点对于小目标比较敏感，适合于小目标检测，但是浅层特征具有较少的语义信息，与深层特征相比具有较弱的辨别力，导致小文本定位的性能较差。另一方面，场景文字总是具有夸张的长宽比（例如一个很长的英文单词或者一条中文长句）以及旋转角度（例如基于美学考虑），通用物体检测框架如 Faster RCNN 和 SSD 是无法回归较大长宽比的矩形和旋转矩形。

围绕上面描述的两个问题，本文主要做了以下事情：

1. 为了处理不同尺度的文本，借鉴特征金字塔网络思路，将具有较强判别能力的深层特征与浅层特征相结合，实现在各个层面都具有丰富语义的特征金字塔。另外，当较深层中的小对象丢失时，特征金字塔网络仍可能无法检测到小对象，深层的上下文信息无法增强浅层特征。我们额外扩大了深层的特征图，以更准确地识别小文本。
2. 我们不直接回归文本行，而是将文本行分解为较小的局部可检测的文字片段，并通过深度卷积网络进行学习，最后将所有文字片段连接起来生成最终的文本行。

现有方法

最新的基于深度神经网络的文本定位算法大致可以分为两大类：（1）基于分割的文本定位；（2）基于回归的文本定位。

（1）基于分割的文本定位

当前基于分割的文本定位方法大都受到完全卷积网络（FCN [2]）的启发。全卷积网络（FCN, fully convolutional network），是去除了全连接 (fc) 层的基础网络，最初是用于实现语义分割任务。由于 FCN 网络最后一层特征图的像素分辨率较高，而图文识别任务中需要依赖清晰的文字笔画来区分不同字符（特别是汉字），所以 FCN 网络很适合用来提取文本特征。当 FCN 被用于图文识别任务时，最后一层特征图中每个像素将被分成文字行（前景）和非文字行（背景）两个类别。

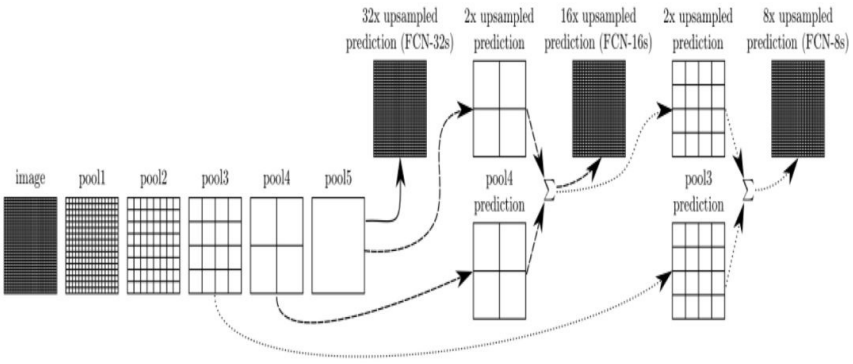


图 2　全卷积网络

（2）基于回归的文本定位

Textboxes [3] 是经典的也是最常用的基于回归的文本定位方法，它基于 SSD 框架，训练方式是端到端，运行速度也较快。为了适应文本行细长型特点，特征层也用长条形卷积核代替了其他模型中常见的正方形卷积核。为了防止漏检文本行，还在垂直方向增加了候选框数量。为了检测大小不同的字符块，在多个尺度的特征图上并行预测文本框，然后对预测结果做 NMS 过滤。

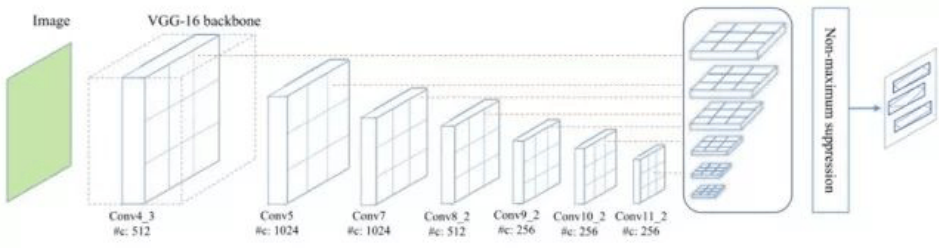


图 3　Textboxes 框架

提出方法

我们的方法也是基于 SSD，整体框架如图 4。为了应对多尺度文字尤其是小文字，对高层特征图进行间隔采样，以保持高层特征图分辨率。同时借鉴特征金字塔网络相关思路，将高层特征图上采样与底层特征叠加，构建一个新的多层级金字塔特征

图（图 4 蓝色框部分）。此外，为了处理各种方向文字，在不同尺度的特征图上预测文字片段以及片段之间的连接关系，然后对预测出的文字片段和连接关系进行组合，得到最终文本框。下面将具体介绍方法。

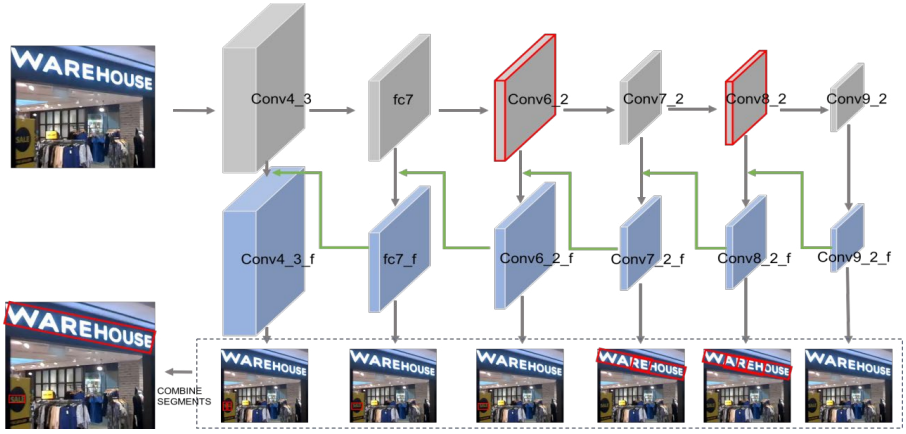


图 4　我们方法框架

（1）扩大高层特征图

深度卷积神经网络通常是逐层下采样，这对于物体分类来说是有效的，但是对于检测任务来说是有损害的。基于时间和性能的权衡考量，我们对卷积网络中最后几层特征进行间隔采样，如图 5，从 Conv6_2 层开始下采样，Conv7_2 层保持原分辨率，Conv8_2 层再下采样。

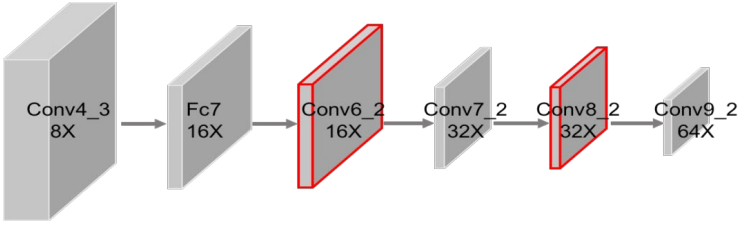


图 5　扩大特征图

（2）构建特征金字塔

虽然通过扩大深度特征图的设计可以更好地检测小文本，但较小的文本仍然难

以检测。为了更好地检测较小的文本，进一步增强较浅层（例如图 5 中 conv4_3，Fc7）的特征。我们通过融合高层和低层的特征构建了一个新的特征金字塔（图 4 中蓝色部分：conv4_3_f,fc7_f,conv6_2_f,conv7_2_f,conv8_2_f 和 conv9_2_f），新的金字塔特征具有更强辨别力和语义丰富性。

高层和低层特征融合策略如图 6 所示，高层特征图先进行上采样使之与低层特征图相同大小，然后与低层特征图进行叠加，叠加后的特征图再连接一个 3*3 卷积，获得固定维度的特征图，我们设定固定维度 d=256。

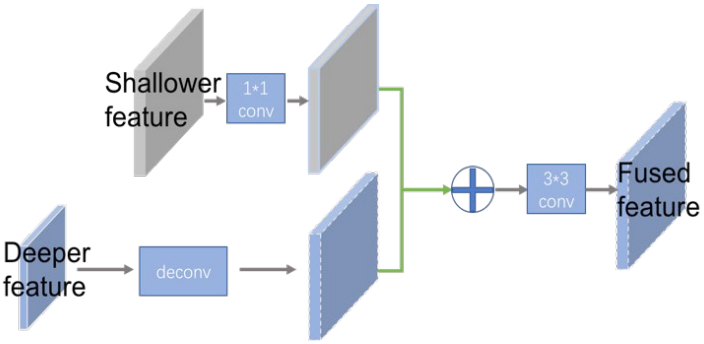


图 6　构建特征金字塔模块

（3）预测文字片段及片段之间连接关系

如图 7，先将每个文字词切割为更易检测的有方向的小文字块（segment），然后用邻近连接（link）将各个小文字块连接成词。这种方案方便于识别长度变化范围很大的、带方向的词和文本行，它不会象 Faster-RCNN 等方案因为候选框长宽比例原因检测不出长文本行，而且处理速度很快。

图 7　小文字块和近邻连接

基于第（2）小节构建的特征金字塔特征图，将每层特征图上特征点用于检测小文字块和文字块连接关系。如图 8，连接关系可以分为八种，上、下、左、右、左上、右上、左下、右下，同一层特征图、或者相邻层特征图上的小文字块都有可能被连接入同一个词中，换句话说，位置邻近、并且尺寸接近的文字块都有可能被预测到同一词中。

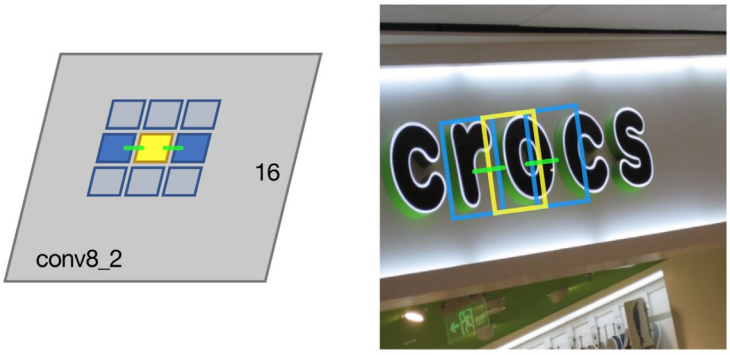


图 8　连接关系示意图

最后基于检测出的小文字块以及文字块连接，组合出文本框（如图 9），具体组合过程如下：

（a）将所有具有连接关系的小文字块组合起来，得到若干小文字块组；（b）对于每组小文字块，找到一条直线能最好的拟合组内所有小文字块中心点；（c）将组内所有小文字块的中心点投影到该直线上，找出距离最远的两个中心点 A 和 B；（d）最终文字框中心点为（A+B）/2，方向为直线斜率，宽度为 A，B 两点直线距离加上 A，B 两点的平均宽度，高度为所有小文字块的平均高度。

图 9　小文字块连接示意图

实验及应用

我们在两个公开数据集上（ICDAR2013，ICDAR2015）对方法进行评测。其中 ICDAR2013 数据集，训练图片 229 张，测试图片 233 张；ICDAR2015 数据集，训练图片 1000 张，测试图片 500 张，它们都来自于自然场景下相机拍摄的图片。

（1）我们首先对比了扩大高层特征图与不扩大高层特征图的性能比较，并在基础上对比加入特征金字塔后的性能比较，在 ICDAR2015 数据集上实验，结果如表 1：

表 1　方法中不同模块有效性验证

Method	Precision	Recall	F-measure
Baseline	79.5	73.4	76.3
扩大高层特征图	81	76.3	78.6
金字塔+扩大高层特征图	88	76.8	82

“baseline”方法是 ssd 框架 + 预测文字片段及片段之间连接关系模块，“扩大高层特征图”是在 baseline 方法基础上对高层特征图进行扩大，“金字塔 + 扩大高层特征图”是在 baseline 方法基础上对高层特征图进行扩大 并且加入特征金字塔。从表 1 中不难发现，扩大高层特征图可以带来精度和召回的提升，尤其是召回有近 3 个点的提升（73.4->76.3），这很好理解，因为更大的特征图产生更多的特征点以及预测结果；在此基础上再加入金字塔机制，精度获得显著提升，说明金字塔结构极大增强低层特征判别能力。

（2）我们也和其他方法也做了比较，具体见表 2 和表 3：

表 2　ICDAR2013 数据集与其他方法比较

Method	Precision	Recall	F-measure
CTPN	93	83	87.7
TextBoxes++	88	74	81
PixelLink	84.4	82.3	83.3
SegLink	87.7	83.0	85.3
OUR	92.4	83.8	87.9

表 3　ICDAR2015 数据集与其他方法比较

Method	Precision	Recall	F	FPS
SegLink	73.1	76.8	75	-
East	80.5	72.8	76.4	6.5
RRPN	82.2	73.2	77.4	-
TextBoxes++	87.2	76.7	81.7	11.6
PixelLink	82.9	81.7	82.3	7.3
OUR	88	76.8	82	10.3

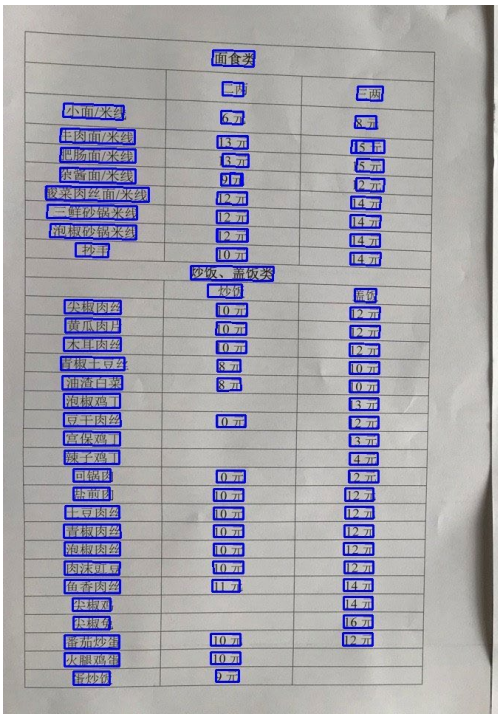
从上表中可以看出，我们的方法在时间和精度上取得很好的权衡。在 ICDAR2015 数据集上，虽然性能不及 PixelLink，但是 FPS 要远高于它；而相比 TextBoxes++，虽然 FPS 略低于它，但是精度更高。图 10 给出一些文字定位结果示例。

图 10　文字定位结果示意图

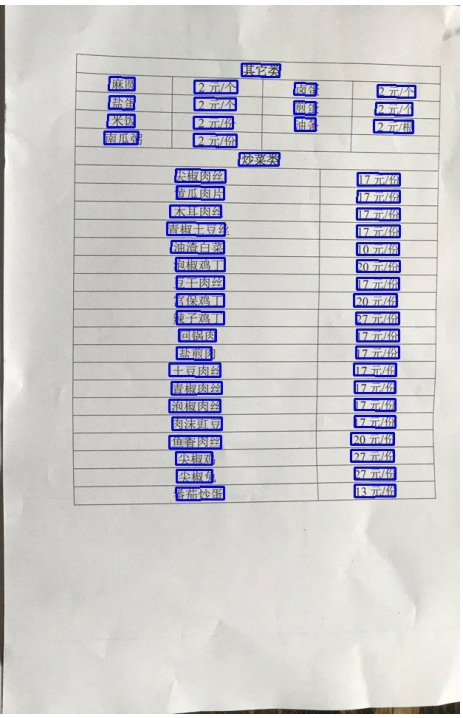
（3）此外，本方法也落地应用于实际业务场景菜单识别中。菜单上文字通常较小、较密，菜名文字可长可短，以及由于拍摄角度导致文字方向倾斜等。如图 11 所示，方法能很好的解决以上问题（小文字、密集文字行、长文本、不同方向）；并且在 500 张真实商家菜单图片上进行评测，相比 SegLink 方法，性能明显提升（近 5 个点提升）。

表 4　菜单测试结果

500菜单测试集	Precision	Recall	F-measure
改进前	85.1	84.7	84.9
改进后	89.5	90.3	89.9



面食类		
	二两	三两
小面/米线	6元	8元
牛肉面/米线	13元	15元
肥肠面/米线	13元	15元
浓酱面/米线	9元	12元
酸菜肉丝面/米线	12元	14元
三鲜砂锅米线	12元	14元
泡椒砂锅米线	12元	14元
抄手	10元	14元
炒饭、盖饭类		
	炒饭	盖饭
尖椒肉丝	10元	12元
黄瓜肉片	10元	12元
木耳肉丝	10元	12元
青椒土豆丝	8元	10元
油渣白菜	8元	10元
泡椒鸡丁		13元
豆干肉丝	10元	12元
宫保鸡丁		13元
辣子鸡丁		14元
回锅肉	10元	12元
盐煎肉	10元	12元
土豆肉丝	10元	12元
青椒肉丝	10元	12元
泡椒肉丝	10元	12元
肉沫豆豆	10元	12元
鱼香肉丝	11元	14元
尖椒鸡		14元
尖椒鸡		16元
蒜茄炒蛋	10元	12元
水腿鸡蛋	10元	
青炒饭	9元	



其它类			
麻圆	2元/个	皮蛋	2元/个
盐蛋	2元/个	咸蛋	2元/个
米饭	2元/份	油条	2元/根
黄瓜蛋	2元/份		
炒菜类			
尖椒肉丝			17元/份
黄瓜肉片			17元/份
木耳肉丝			17元/份
青椒土豆丝			17元/份
油渣白菜			10元/份
泡椒鸡丁			20元/份
豆干肉丝			17元/份
宫保鸡丁			20元/份
辣子鸡丁			27元/份
回锅肉			17元/份
盐煎肉			17元/份
土豆肉丝			17元/份
青椒肉丝			17元/份
泡椒肉丝			17元/份
肉沫豆豆			17元/份
鱼香肉丝			20元/份
尖椒鸡			27元/份
尖椒鸡			27元/份
蒜茄炒蛋			13元/份

图 11　菜单文字定位结果示意图

结论

本文我们提出了一个高效的场景文本检测框架。针对文字特点，我们扩大高层特征图尺寸并构建了一个特征金字塔，以更适用于不同比例文本，同时通过检测文本片段和片段连接关系来处理长文本和定向文本。实验结果表明该框架快速且准确，在 ICDAR2013 和 ICDAR2015 数据集上获得了不错结果，同时应用到公司实际业务场景菜单识别上，获得明显性能提升。下一步，受实例分割的方法 PixelLink [4] 的启发，我们也考虑将文本片段进一步细化到像素级，同时融合检测和分割方法各自优缺点，构建联合检测和分割的文字定位框架。

参考文献

Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, Serge Belongie. “Feature Pyramid Networks for Object Detection.” arXiv preprint. arXiv: 1612.03144, 2017.

J. Long, E. Shelhamer, and T. Darrell. “Fully convolutional networks for semantic segmentation.” In CVPR, 2015.

M. Liao, B. Shi, and X. Bai. “Textboxes++: A single-shot oriented scene text detector.” IEEE Trans. on Image Processing, vol. 27, no. 8, 2018.
D. Deng, H. Liu, X. Li, and D. Cai. “Pixellink: Detecting scene text via instance segmentation.” In AAAI, pages 6773 - 6780, 2018.

作者简介

刘曦，美团视觉图像中心文字识别组算法专家。

招聘信息

美团视觉图像中心文字识别组：针对美团各项业务如商家入驻资质审核、网页信息合规审核等需求，对证照、票据、菜单、网图等图片类型开展文字识别研发工作。利用高性能文字识别功能，帮助业务方和商家实现自动审核 、自动录入，显著提升人效、降低成本，改善体验。

欢迎计算机视觉相关及相关领域小伙伴加入我们，简历可发邮件至 tech@meituan.com（邮件标题注明：美团视觉图像中心文字识别组）。

CVPR 2019 轨迹预测竞赛冠军方法总结

李鑫

背景

CVPR 2019 是机器视觉方向最重要的学术会议，本届大会共吸引了来自全世界各地共计 5160 篇论文，共接收 1294 篇论文，投稿数量和接受数量都创下了历史新高，其中与自动驾驶相关的论文、项目和展商也是扎堆亮相，成为本次会议的“新宠”。

障碍物轨迹预测挑战赛（Trajectory Prediction Challenge）隶属于 CVPR 2019 Workshop on Autonomous Driving — Beyond Single Frame Perception（自动驾驶研讨会），由百度研究院机器人与自动驾驶实验室举办，侧重于自动驾驶中的多帧感知，预测和自动驾驶规划，旨在聚集来自学术界和工业界的研究人员和工程

师，讨论自动驾驶中的计算机视觉应用。美团无人配送与视觉团队此项比赛获得了第一名。

在该比赛中，参赛队伍需要根据每个障碍物过去 3 秒的运动轨迹，预测出它在未来 3 秒的轨迹。障碍物共有四种类型，包括行人、自行车、大型机动车、小型机动车。每种障碍物的轨迹用轨迹上的采样点来表示，采样的频率是 2 赫兹。美团的方法最终以 1.3425 的成绩取得该比赛的第一名，同时我们也在研讨会现场分享了算法和模型的思路。

赛题简介

轨迹预测竞赛数据来源于在北京搜集的包含复杂交通灯和路况的真实道路数据，用于竞赛的标注数据是基于摄像头数据和雷达数据人工标注而来，其中包含各种车辆、行人、自行车等机动车和非机动车。

训练数据：每个道路数据文件包含一分钟的障碍物数据，采样频率为每秒 2 赫兹，每行标注数据包含障碍物的 ID、类别、位置、大小、朝向信息。

测试数据：每个道路数据文件包含 3 秒的障碍物数据，采样频率为每秒 2 赫兹，目标是预测未来 3 秒的障碍物位置。

评价指标

平均位移误差：Average displacement error（ADE），每个预测位置和每个真值位置之间的平均欧式距离差值。

终点位移误差：Final displacement error（FDE），终点预测位置和终点真值位置之间的平均欧式距离差值。

由于该数据集包含不同类型的障碍物轨迹数据，所以采用根据类别加权求和的指标来进行评价。

$$WSADE = D_v \cdot ADE_{\mathrm{v}} + D_p \cdot ADE_{\mathrm{p}} + D_b \cdot ADE_{\mathrm{b}},$$

$$WSFDE = D_v \cdot FDE_{\mathrm{v}} + D_p \cdot FDE_{\mathrm{p}} + D_b \cdot FDE_{\mathrm{b}},$$

现有方法

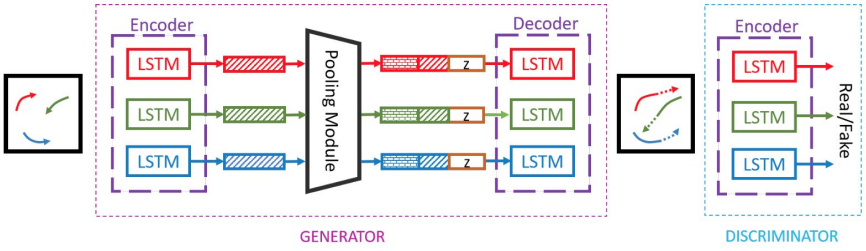
这次竞赛要解决的预测问题不依赖地图和其他交通信号等信息，属于基于非结构化数据预测问题，这类问题现在主流的方法主要根据交互性将其区分为两类：1. **独立预测**，2. **依赖预测**。

独立预测是只基于障碍物历史运动轨迹给出未来的行驶轨迹，依赖预测是会考虑当前帧和历史帧的所有障碍物的交互信息来预测所有障碍物未来的行为。

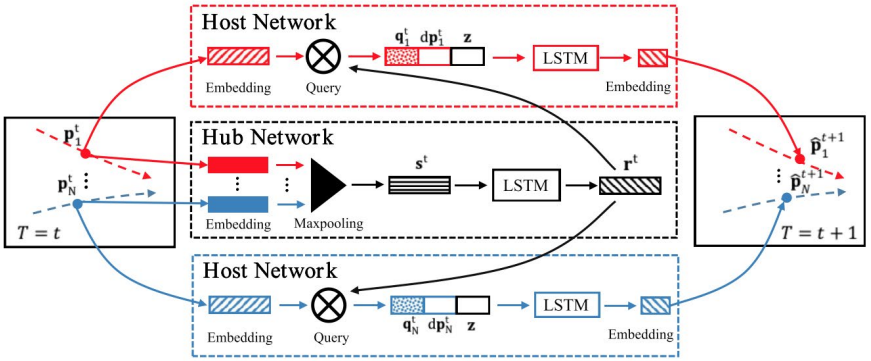
考虑交互信息的依赖预测，是当前学术界研究比较多的一类问题。但是经调研总结，我们发现其更多的是在研究单一类别的交互，比如在高速公路上都是车辆，那预测这些车辆之间的交互；再比如在人行道上预测行人的交互轨迹。预测所有类别障碍物的之间的交互的方法很少。

以下是做行人交互预测的两个方法模型：

方法 1. Social GAN，分别对每个障碍车输入进行 Encoder，然后通过一个统一的 Pooling 模块提取交互信息，再单独进行预测。



方法 2. StarNet，使用一个星型的 LSTM 网络，使用 Hub 网络提取所有障碍物的交互信息，然后再输出给每个 Host 网络独立预测每个障碍物的轨迹。

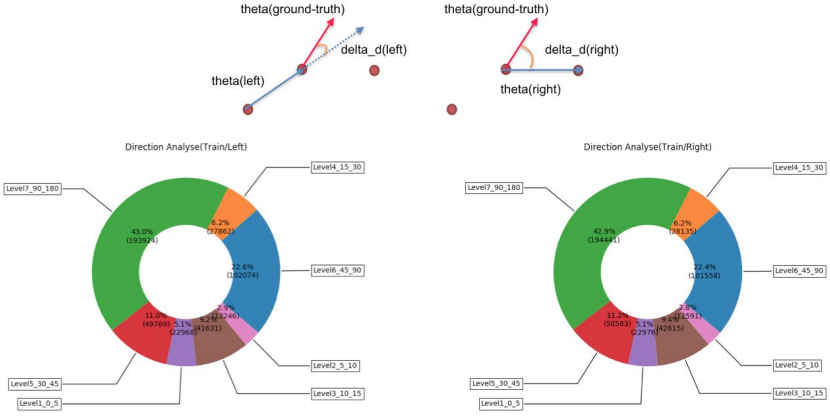


我们的方法

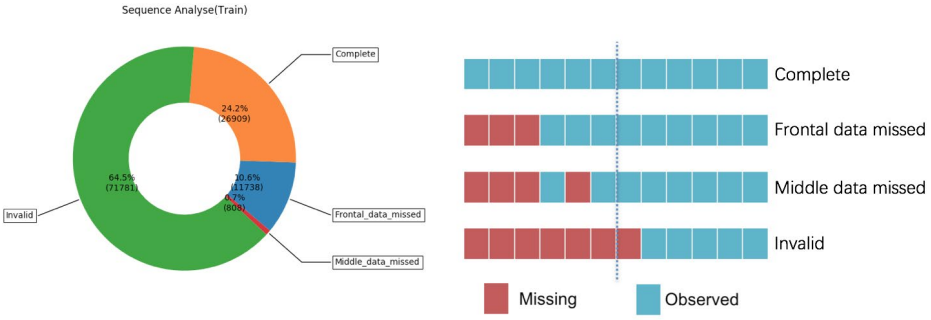
数据分析

拿到赛题之后，我们首先对训练数据做了分析，由于最终的目标是预测障碍物测位置，所以标注数据中的障碍物大小信息不太重要，只要根据类别来进行预测即可。

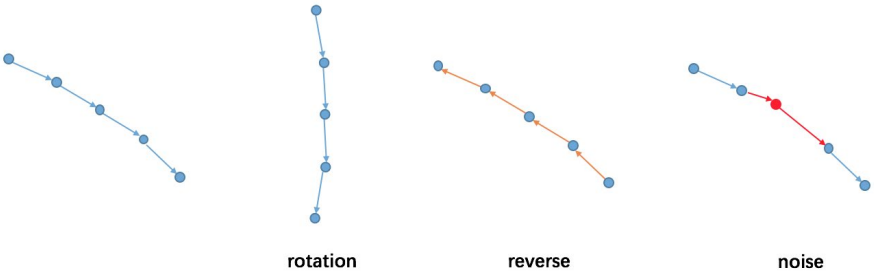
其次，分析朝向信息是否要使用，经统计发现真值标注的朝向信息非常不准确，从下图可以看到，大部分的标注方向信息都和轨迹方向有较大差距，因此决定不使用朝向信息进行预测。



然后，分析数据的完整性，在训练过程中每个障碍物需要 12 帧数据，才可以模拟测试过程中使用 6 帧数据来预测未来 6 帧的轨迹。但是在真实搜集数据的时候，没有办法保证数据的完整性，可能前后或中间都可能缺少数据，因此，我们根据前后帧的位置关系插值生成一些训练数据，以填补数据的缺失。



最后，对数据做了增强，由于我们的方法不考虑障碍物之间的交互，仅依赖每个障碍物自身的信息进行训练，因此障碍物轨迹进行了旋转、反向、噪声的处理。

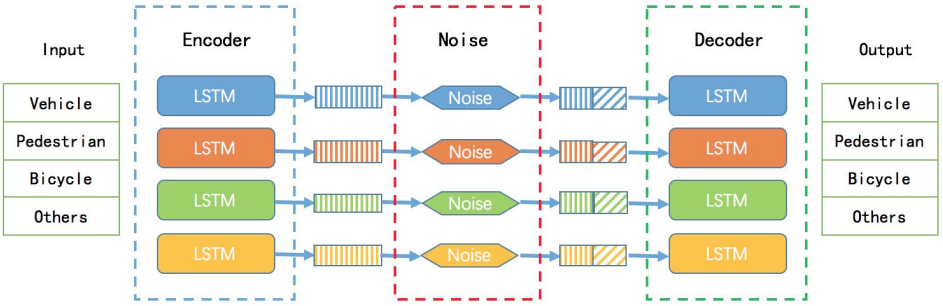


模型结构

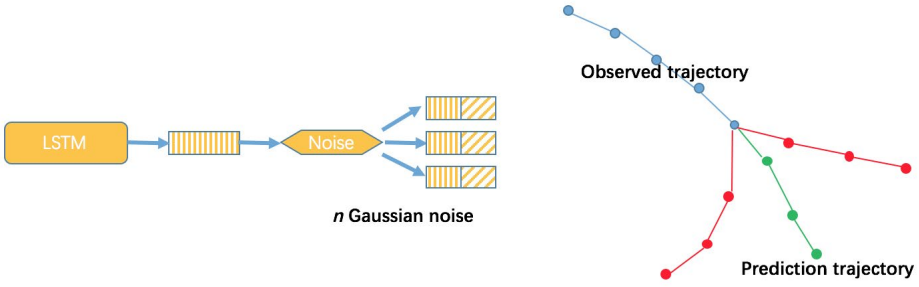
由于这次轨迹预测的问题是预测所有类别的轨迹，所以使用解决单一类别的轨迹预测模型不适用于该问题，而且如果把所有的物体放在单一的交互模型中来，不能正确提取出不同障碍物之间的交互特征。我们尝试了一些方法也证实了这一点。

因此在竞赛中，我们使用了多类别的独立预测方法，网络结构如下图，该方法针对每个类别构造一个 LSTM 的 Encoder-Decoder 模型，并且在 Encoder 和 Decoder 之间加入了 Noise 模块，Noise 模块生成固定维度的高斯噪声，将该噪声

和 Encoder 模块输出的 LSTM 状态量进行连结作为 Decoder 模块的 LSTM 初始状态量，Noise 模块主要作用是负责在多轮训练过程中增加数据的扰动，在推理过程中通过给不同的 Noise 输入，可以生成多个不同的轨迹。



最终，需要在不同的轨迹输出中选择一个最优的轨迹，这里采用了一个简单的规则，选择预测的轨迹方向和历史轨迹方向最接近的轨迹作为最终的轨迹输出。



实验结果

我们仅使用了官方提供的数据进行训练，按照前述数据增强方法先对数据进行增强，然后搭建网络结构进行训练，Loss 采用 Weighted Sum of ADE（WSADE），采用 Adam 优化方法，最终提交测试的 WSADE 结果为 1.3425。

方法	WSADE
我们的方法	1.3425
StarNet（基于交互的方法）	1.8626
TrafficPredict（ApolloScape Baseline 方法）	8.5881

总结

在这次竞赛中，我们尝试了使用多类别的独立预测方法，通过对数据增强和加入高斯噪声，以及最终人工设计规则选择最优轨迹的方法，在这次障碍物轨迹预测挑战赛（Trajectory Prediction Challenge）中获得了较好的成绩。但是，我们认为，基于交互的方法用的好的话应该会比这种独立预测方法还是要好，比如可以设计多类别内部交互和类别间的交互。另外，也关注到现在有一些基于图神经网络的方法也应用在轨迹预测上，今后会在实际的项目中尝试更多类似的方法，解决实际的预测问题。

参考文献

Yanliang Zhu, Deheng Qian, Dongchun Ren and Huaxia Xia. StarNet: Pedetrian Trajectory Prediction using Deep Neural Network in Star Topology[C]//Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). 2019.

Gupta A, Johnson J, Fei-Fei L, et al. Social gan: Socially acceptable trajectories with generative adversarial networks[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2018: 2255-2264.

Apolloscape. Trajectory dataset for urban traffic. 2018. http://apolloscape.auto/trajectory.html.

作者简介

李鑫，美团无人配送与视觉部 PNC 组轨迹预测组算法专家。

炎亮，美团无人配送与视觉部 PNC 组轨迹预测组算法工程师。

德恒，美团无人配送与视觉部 PNC 组轨迹预测组负责人。

冬淳，美团无人配送与视觉部 PNC 组负责人。

顶会论文：基于神经网络 StarNet 的行人轨迹交互预测算法

炎亮　德恒　冬淳　华夏

1. 背景

民以食为天，如何提升超大规模配送网络的整体配送效率，改善数亿消费者在”吃“方面的体验，是一项极具挑战的技术难题。面向未来，美团正在积极研发无人配送机器人，建立无人配送开放平台，与产学研各方共建无人配送创新生态，希望能在一个场景相对简单、操作高度重复的物流配送中，提高物流配送效率。在此过程中，美团无人配送团队也取得了一些技术层面的突破，比如基于神经网络 StarNet 的行人轨迹交互预测算法，论文已发表在 IROS 2019。IROS 的全称是 IEEE/RSJ International Conference on Intelligent Robots and Systems，IEEE 智能机器人与系统国际会议，它和 ICRA、RSS 并称为机器人领域三大国际顶会。

1.1　行人轨迹预测的意义

在无人车行驶过程中，它需要对周围的行人进行轨迹预测，这能帮助无人车更加安全平稳地行驶。我们可以用图 1 来说明预测周围行人的运动轨迹对于无人车行驶的重要性。

图 1 中蓝色方块代表无人车，白色代表行人。上半部分描述的是在不带行人轨迹预测功能情况下无人车的行为。这种情况下，无人车会把行人当做静态物体，但由于每个时刻行人都会运动，导致无人车规划出来的行驶轨迹会随着时间不停地变化，加大了控制的难度，同时还可能产生碰撞的风险，这样违背了安全平稳行驶的目标。下半部分是有了行人轨迹预测功能情况下的无人车行为。这种情况下，无人车会预测周围行人的行驶轨迹，因此在规划自身行驶时会考虑到未来时刻是否会与行人碰撞，最终规划出来的轨迹更具有“预见性”，所以避免了不必要的轨迹变化和碰撞风险。

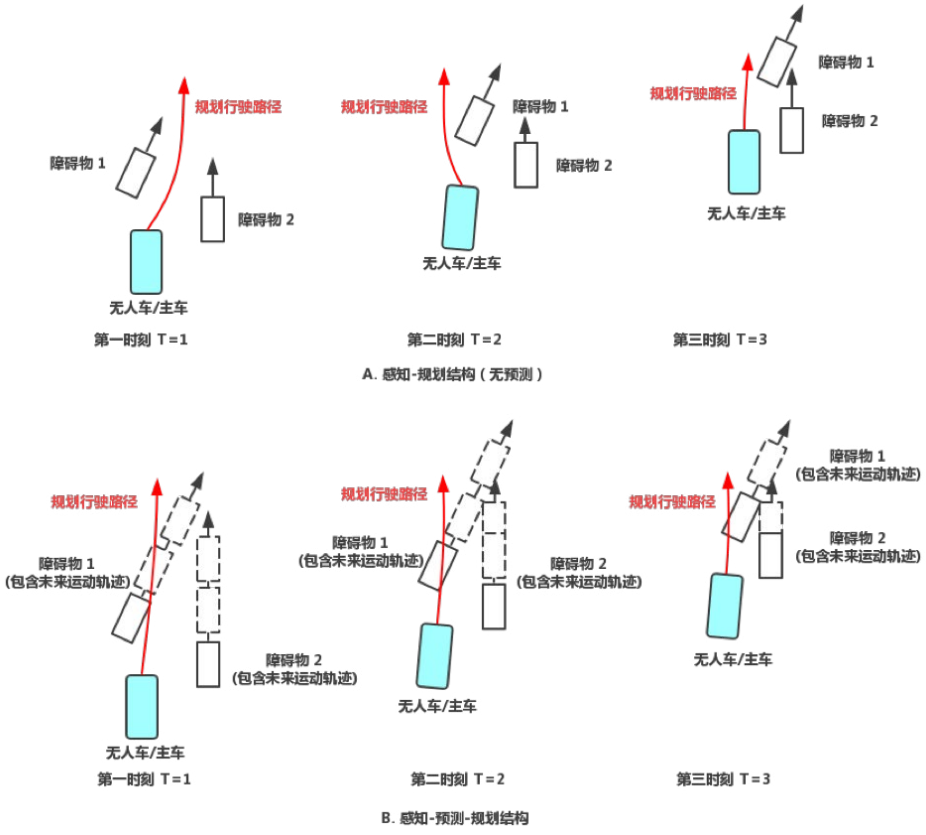


图 1　主车规划轨迹跳变问题

1.2　行人轨迹预测的难点

总体而言，行人轨迹预测的难点主要有两个：

第一，行人运动灵活，预测难度大。本身精确预测未来的运动轨迹是一个几乎不可能完成的任务，但是通过观察某个障碍物历史时刻的运动轨迹，可以根据一些算法来大致估计出未来的运动轨迹（最简单的是匀速直线运动）。在实际中，相比于自行车、汽车等模型，行人运动更加灵活，很难对行人建立合理的动力学模型（因为行人可以随时转弯、停止、运动等），这加剧了行人预测的难度。

第二，行人之间的交互，复杂又抽象。在实际场景中，某一行人未来的运动不仅受自己意图支配，同样也受周围行人的影响（例如避障）。这种交互非常抽象，在算法

中往往很难精确地建模出来。目前，大部分算法都是用相对空间关系来进行建模，例如相对位置、相对朝向、相对速度大小等。

1.3 相关工作介绍

传统算法在做预测工作时会使用一些跟踪的算法，最常见的是各类时序模型，例如卡尔曼滤波（Kalman Filter, KF）、隐马尔可夫（Hidden Markov Model, HMM）、高斯过程（Gaussian Process, GP）等。这类方法都有一个很明显的特点，就是根据历史时序数据，建立时序递推数学公式：$X^t = f\left(X^{t-1}\right)$ 或者 $p\left(X^t \mid X^{t-1}\right)$。因为这类方法具有严格的数学证明和假设，也能处理一些常规的问题，但是对于一些复杂的问题就变得“束手无策”了。这是因为这些算法中都会引入一些先验假设，例如隐变量服从高斯分布，线性的状态转换方程以及观测方程等，而最终这些假设也限制了算法的整体性能。神经网络一般不需要假设固定的数学模型，凭借大规模的数据集促使网络学习更加合理的映射关系。本文我们主要介绍一些基于神经网络的行人预测算法。

基于神经网络的预测算法（主要以长短期记忆神经网络 Long Short Term Memory，LSTM 为主）在最近 5 年都比较流行，预测效果确实比传统算法好很多。在 CVPR（IEEE Conference on Computer Vision and Pattern Recognition）2019 上，仅行人预测算法的论文就有 10 篇左右。这里我们简单介绍 2 篇经典的行人预测算法思路，如果对这方面感兴趣的同学，可以通过文末的参考文献深入了解一下。第一篇是 CVPR 2016 斯坦福大学的工作 Social-LSTM，也是最经典的工作之一。Social-LSTM 为每个行人都配备一个 LSTM 网络预测其运动轨迹，同时提出了一个 Social Pooling Layer 的模块来计算周围其他行人对其的影响。具体的计算思路是将该行人周围的区域划分成 NxN 个网格，每个网络都是相同的大小，落入这些网格中的行人将会参与交互的计算。

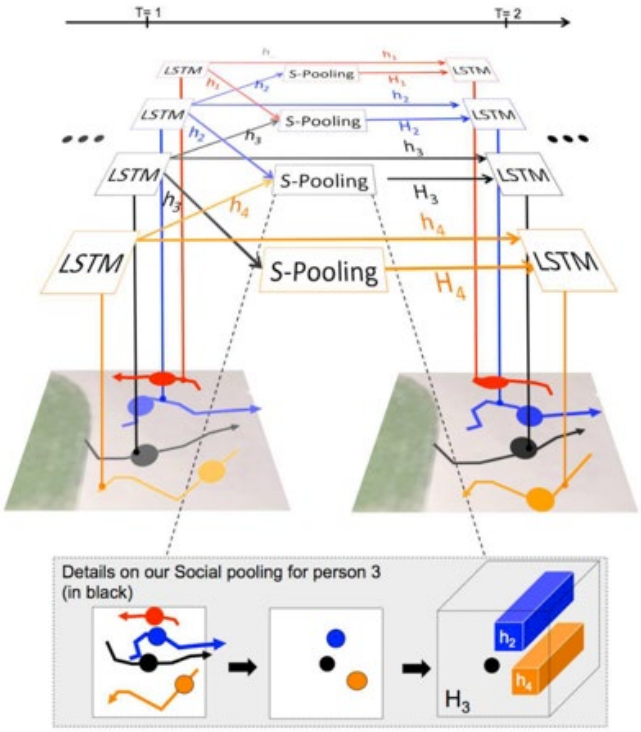


图 2　左：Social LSTM 原理 右：Social Pooling 计算过程

第二篇是 CVPR 2019 卡耐基梅隆大学 & 谷歌 & 斯坦福大学的工作，他们的工作同样使用 LSTM 来接收历史信息并预测行人的未来轨迹。不同于其他算法的地方在于，这个模型不仅接收待预测行人的历史位置信息，同时也提取行人外观、人体骨架、周围场景布局以及周围行人位置关系，通过增加输入信息提升预测性能。除了预测具体的轨迹，算法还会做粗粒度预测（决策预测），输出行人未来时刻可能所在的区域。

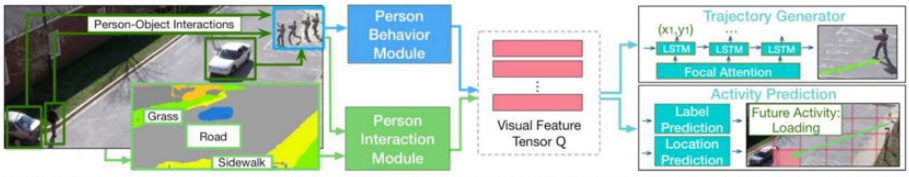


Figure 2. Overview of our model. Given a sequence of frames containing the person for prediction, our model utilizes person behavior module and person interaction module to encode rich visual semantics into a feature tensor.

图 3　算法整体结构

其他的相关工作，还包括基于语义图像 / 占有网格（Occupancy Grid Map, OGM）的预测算法，基于信息传递（Message Passing, MP）的预测算法，基于图网络（Graph Neural Network, GNN）的预测算法（GCN/GAT 等）等等。

2. StarNet 介绍

目前，现有的轨迹预测算法主要还是聚焦在对行人之间交互的建模，轨迹预测通常只使用 LSTM 预测即可。如下图 4 左，现有关于轨迹预测的相关工作基本都是考虑行人之间两两交互，很少有考虑所有行人之间的全局交互（即使是 GCN，也需要设计对应的相似矩阵来构造拉普拉斯矩阵，这也是一个难点）。我们可以举一个例子来说明现有其他算法预测的流程：

- 假设感知模块检测到当前 N 个行人的位置，如何计算第一个行人下一时刻的位置？ Step 1 计算其他人对于第一个行人的交互影响。将第 i 个行人在第 t 时刻的位置记为（一般是坐标 x 和 y）。可以通过以下公式计算第一个行人的交互向量：$Interaction_1^t = f\left(P_2^t - P_1^t, P_3^t - P_1^t, \cdots, P_N^t - P_1^t\right)$

  从上述公式可以大致看到，相对位置关系是最重要的计算指标，计算的函数 f 一般是一个神经网络。 Step 2 计算第一个行人下个时刻的位置。通常需要根据上一时刻的位置与交互向量：$P_1^{t+1} = g\left(P_1^t, Interaction_1^t\right)$ 上述公式中，计算的函数 g 同样是神经网络，即上面提到的长短期记忆神经网络 LSTM。

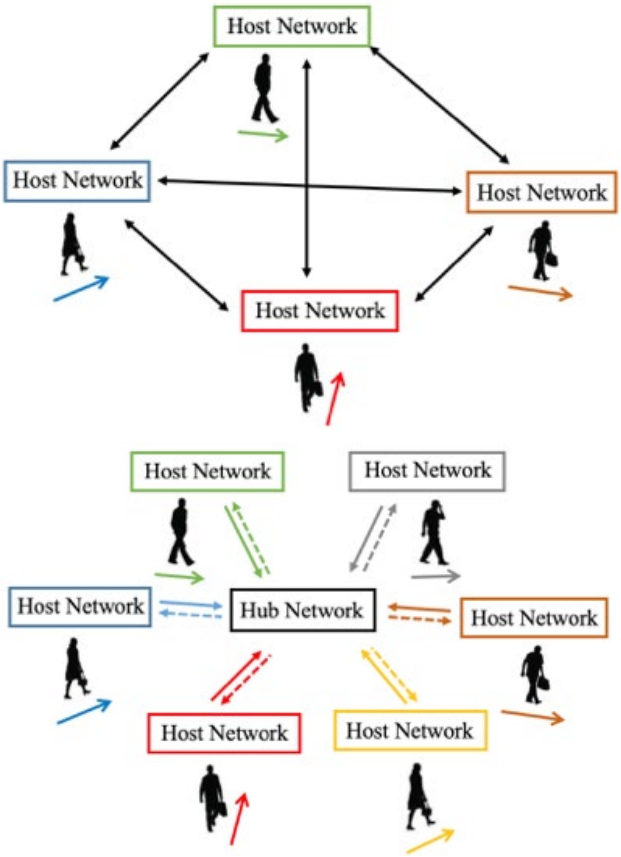


图 4　算法思路对比图 上：传统算法 下：StarNet

两两交互的方式存在两个问题：

1. 障碍物 2 和 3 确实会影响障碍物 1 的运动，但是障碍物 2 和 3 之间同样也存在相互影响，因此不能直接将其他障碍物对待预测障碍物的影响单独剥离出来考虑，这与实际情况不相符。
2. 两两计算消耗的资源大，如果有 N 个障碍物，那么两两交互就需要 N 的平方次计算，随着 N 的变大，计算量呈平方倍增长。我们希望障碍物之间的交互能否只计算 1 次而非 N 次，所有障碍物的轨迹预测都共享这个全局交互那就更好了。

基于上述两个问题，我们提出了一种新的模型，该模型旨在高效解决计算全局交互的问题。因为传统算法普遍存在计算两两交互的问题（即使是基于 Attention 注意力机制的 Message Passing 也很难考虑到全局的交互），本文想尝试通过一些更加简单直观的方式来考虑所有障碍物之间的全局交互，我们的算法大致思路如下：

每个时刻所有障碍物的位置可以构成一张静态的“地图”，随着时间的变化，这些静态地图就变成了一张带有时序信息的动态图。这张动态图中记录了每个区域内的障碍物运动信息，其中运动信息是由所有障碍物一起影响得到的，而非单独地两两交互形成。对于每个障碍物的预测阶段，只要根据该障碍物的位置，就可以在这张时序地图中查询该区域在历史时刻的障碍物运动信息（例如这个区域在历史时刻中，障碍物 1、2、4、5 都有其运动的轨迹）。通过“共享全局交互地图 + 个体查询”的方式，就可以做到计算全局交互以及压缩计算开销。

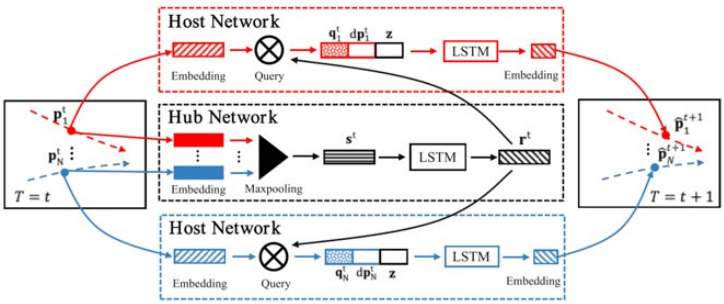


Fig. 2: The process of predicting the coordinates. At time step t, StarNet takes the newly observed (or predicted) coordinates $\{\mathbf{p}_i^t\}_{i=1}^N$ (or $\{\widehat{\mathbf{p}}_i^t\}_{i=1}^N$) and outputs the predicted coordinates $\{\widehat{\mathbf{p}}_i^{t+1}\}_{i=1}^N$.

图 5　StarNet 网络结构图

我们的算法结构如上图 5 所示，Host Network 是基于 LSTM 的轨迹预测网络；Hub Network 是基于 LSTM 的全局时序交互计算网络。在论文具体的实现中，首先 Hub Network 的静态地图模块是通过接受所有障碍物同一时刻的位置信息、全连接网络和最大池化操作得到一个定长的特征向量 s^t；然后动态地图模块使用 LSTM 网络对上述的特征向量 s^t 进行时序编码，最终得到一个全局交互向量 r^t。Host Network 首先根据行人（假设要预测第一个行人下时刻的位置）的位置 P_1^t 去动态地图 r^t 中查询自己当前位置区域内的交互 q_1^t，具体我们采用简单的点乘操作（类似于 Attention 机

制）。最终自己的位置 P_1^t 和交互 q_1^t 一起输入 LSTM 网络预测下时刻的的位置 P_1^{t+1} 。

实验阶段，我们与 4 种经典的算法作比较，使用的数据集为 UCY&ETH 数据集，这两个数据集包含 4 个子场景，分别为 ZARA-1/ZARA-2、UNIV、ETH、HOTEL。在预测过程中，所有算法根据每个行人过去 3.2 秒的运动轨迹，预测出它在未来 3.2 秒的轨迹。每 0.4 秒采样一个离散点，因此 3.2 秒的轨迹可以用 8 个轨迹离散点表示。对比的指标有：

（a）平均距离差 ADE（Average Displacement Error）：用算法预测出的轨迹到真实轨迹所有 8 个点之间的平均距离差。（b）终点距离差 FDE（Final Displacement Error）：用算法预测出的轨迹与真实轨迹最后一个终点之间的距离差。（c）前向预测时间以及参数量。

最终的实验结果如下表：

TABLE II: Prediction Errors

Metric	Dataset	LSTM	Social LSTM	Social GAN	Social Attention	StarNet (Ours)
ADE	ZARA-1	0.25	0.27	**0.21**	1.66	0.25
	ZARA-2	0.31	0.33	0.27	2.30	**0.26**
	UNIV	0.36	0.41	0.36	2.92	**0.21**
	ETH	0.70	0.73	0.61	2.45	**0.31**
	HOTEL	0.55	0.49	0.48	2.19	**0.46**
Average ADE	-	0.43	0.45	0.39	2.30	**0.30**
Variance of ADE	-	0.028	0.026	0.021	0.166	**0.008**
FDE	ZARA-1	0.53	0.56	**0.42**	2.64	0.47
	ZARA-2	0.65	0.70	0.54	4.75	**0.53**
	UNIV	0.77	0.84	0.75	5.95	**0.40**
	ETH	1.45	1.48	1.22	5.78	**0.54**
	HOTEL	1.17	1.01	0.95	4.94	**0.91**
Average FDE	-	0.91	0.91	0.78	4.81	**0.57**
Variance of FDE	-	0.118	0.101	0.802	1.394	**0.031**

TABLE III: Computational Time

Metric	LSTM	Social LSTM	Social GAN	Social Attention	StarNet (Ours)
Inference Time (Seconds)	0.029	0.504	0.202	3.714	**0.073**
Number of Paramters (Kilo)	22.87	156.06	108.03	874.95	**31.90**

从实验结果可以看到，我们的算法在 80% 的场景下都优于其他算法，且实时性高（表中 LSTM 的推理时间为 0.029 秒，最快速是由于该算法不计算交互，因此速度最快参数也最少，但是性能较差）。

总结一下，我们提出算法 StarNet 的优势主要包括以下两点：

- 使用全局动态地图的形式来描述行人之间在时间和空间上的相互影响，更加合理，也更加准确。

- Hub Network 全局共享的特征提升了整个算法的计算效率。

3. 未来工作

首先，我们会进一步探索新的模型结构。虽然我们的算法在数据集上取得了不错的效果，但这是我们的第一次尝试，模型设计也比较简单，如果提升模型结构，相信可以取得更好的结果。

其次，我们会提升预测的可解释性。同现有算法一样，目前的模型对计算到的交互缺乏可解释性，仍然依赖于数据驱动。在今后的工作中，我们将通过对交互的可解释建模来提升预测的准确性。

最后，在构建时序的动态地图过程中，引入对于每个障碍物的跟踪信息。换句话说，我们知道每块区域在各个时间点障碍物的位置，但目前算法没有对障碍物在时序上做跟踪（例如时刻 1 有三个障碍物，时刻 2 三个障碍物运动了得到新的位置，网络输入为三个障碍物的位置信息，但是网络无法理解两个时刻中障碍物的对应关系，这降低了交互的性能），这点在以后的工作中还需要继续改进。

参考文献

[1] A. Alahi, K. Goel, V. Ramanathan, A. Robicquet, F. Li and S. Savarese, “Social lstm: Human trajectory prediction in crowded spaces,” in 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE 2016, pp. 961–971.

[2] H. Wu, Z. Chen, W. Sun, B. Zheng and W. Wang, “Modeling trajectories with recurrent neural networks,” in 28th International Joint Conference on Artificial Intelligence (IJCAI). 2017, pp. 3083–3090.

[3] A. Gupta, J. Johnson, F. Li, S. Savarese and A. Alahi, “Social GAN: Socially acceptable trajectories with generative adversarial networks,” in 2018 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, 2018, pp. 2255–2264.

[4] A. Vemula, K. Muelling and J. Oh, “Social attention: Modeling attention in human crowds,” in 2018 IEEE International Conference on Robotics and Automation (ICRA). IEEE, 2018, pp. 1–7.

[5] Y. Xu, Z. Piao and S. Gao S, “Encoding crowd interaction with deep neural network for pPedestrian trajectory prediction,” in 2018 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, 2018, pp. 5275–5284.

[6] D. Varshneya, G. Srinivasaraghavan, “Human trajectory prediction using spatially aware deep attention models,” arXiv preprint arXiv:1705.09436, 2017.

[7] T. Fernando, S. Denma, S. Sridharan and C. Fookes, “Soft+hardwired attention: An lstm framework for human trajectory prediction and abnormal event detection,” arXiv preprint arXiv:1702.05552, 2017.
[8] J. Liang, L. Jiang, J. C. Niebles, A. Hauptmann and F. Li, “Peeking into the future: Predicting future person activities and locations in videos,” in 2019 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, 2019, pp. 5725–5734.
[9] A. Sadeghian, V. Kosaraju, Ali. Sadeghian, N. Hirose, S. H. Rezatofighi and S. Savarese, “SoPhie: An attentive GAN for predicting paths compliant to social and physical constraints,” in 2019 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, 2019, pp. 5725–5734.
[10] R. Chandra, U. Bhattacharya and A. Bera, “TraPHic: Trajectory prediction in dense and heterogeneous traffic using weighted interactions,” in 2019 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, 2019, pp. 8483–8492.
[11] J. Amirian, J. Hayet and J. Pettre, “Social Ways: Learning multi-modal distributions of pedestrian trajectories with GANs,” arXiv preprint arXiv:1808.06601, 2018.

作者简介

朱炎亮，美团无人配送部
钱德恒，美团无人配送部
任冬淳，美团无人配送部
夏华夏，美团无人配送部

招聘信息

美团轨迹预测组招聘深度学习算法工程师，我们希望你：

- 具有扎实的编程能力，能够熟练使用 C++ 或 Python 作为编程语言。
- 具有深度学习相关知识，能熟练使用 TensorFlow 或 Pytorch 作为深度学习算法研发框架。
- 对预测无人车周围障碍物的未来轨迹感兴趣。

欢迎有兴趣的同学投送简历到 tech@meituan.com（邮件标题注明：美团轨迹预测组）。

虎年大吉