# 2020 美团技术年货

## CODE A BETTER LIFE

【算法篇】


美团 美团

# 目录

# 算法

## 智能搜索模型预估框架 Augur 的建设与实践

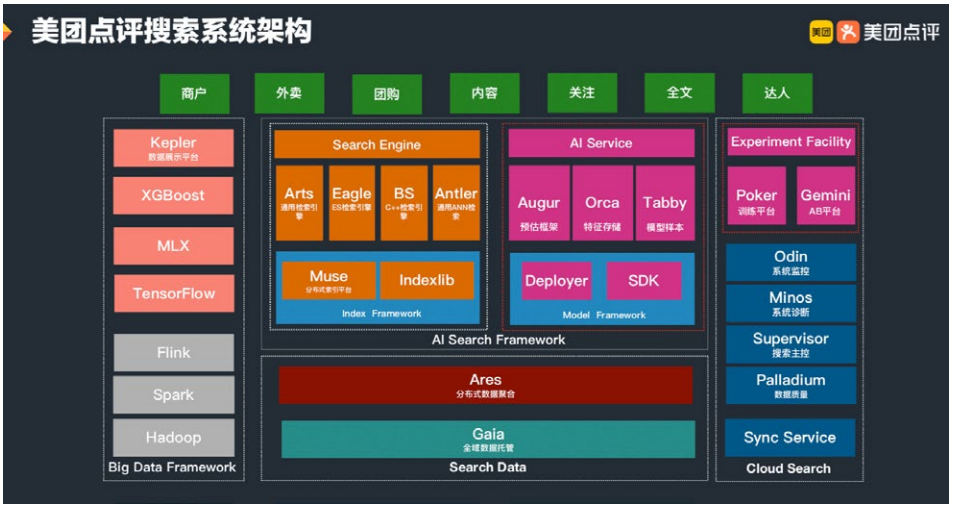
作者：朱敏　紫顺　乐钦　洪晨　乔宇　武进　孝峰　俊浩等

### 1. 背景

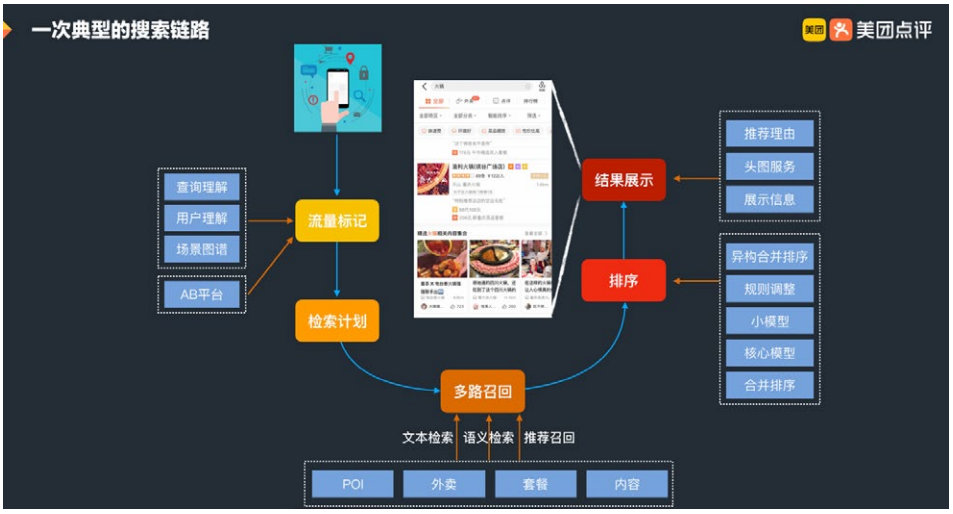
在过去十年，机器学习在学术界取得了众多的突破，在工业界也有很多应用落地。美团很早就开始探索不同的机器学习模型在搜索场景下的应用，从最开始的线性模型、树模型，再到近两年的深度神经网络、BERT、DQN 等，并在实践中也取得了良好的效果与产出。

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。

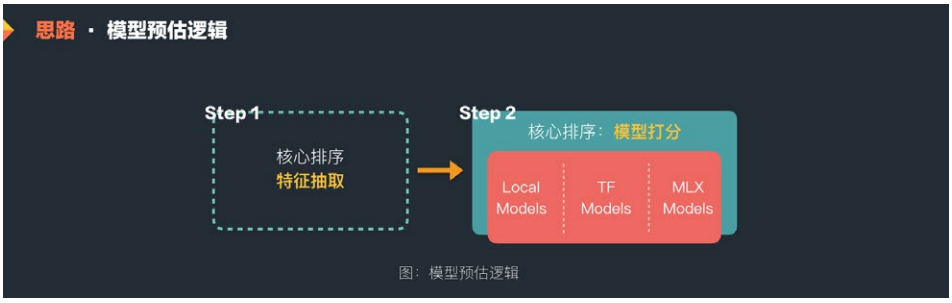
## 2. 抽象过程：什么是模型预估

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）。



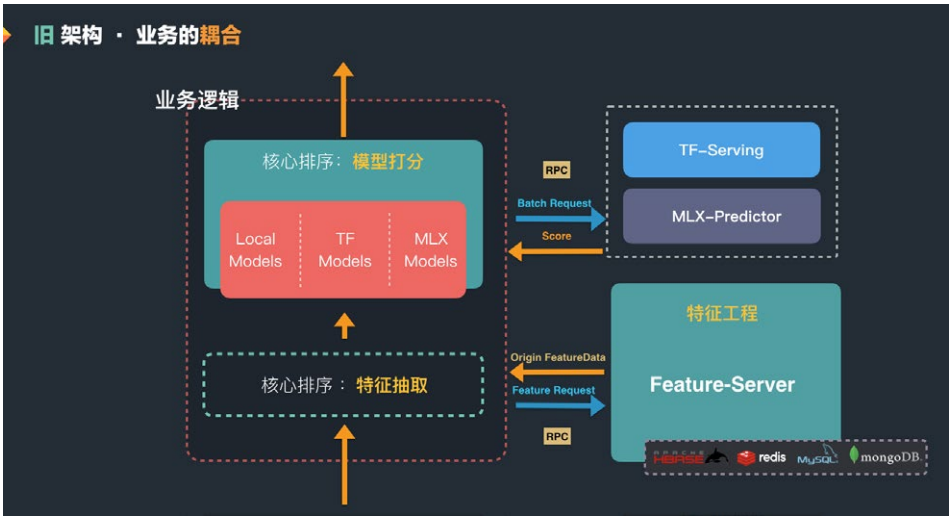

图：模型预估逻辑

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

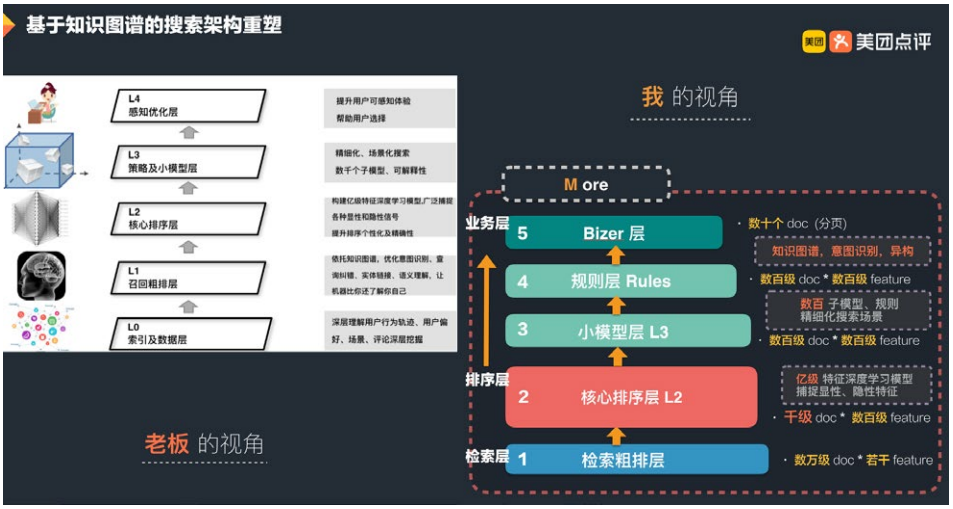
这一套逻辑很简单，构建起来也不复杂，所以在建设初期，我们快速在主搜的核心业务逻辑中快速实现了这一架构，如下图所示。这样的一个架构使得我们可以在主搜的核心排序逻辑中，能够使用各类线性模型的预估，同时也可以借助公司的技术能力，进行深度模型的预估。关于特征抽取的部分，我们也简单实现了一套规则，方便算法同学可以自行实现一些简单的逻辑。




# 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：



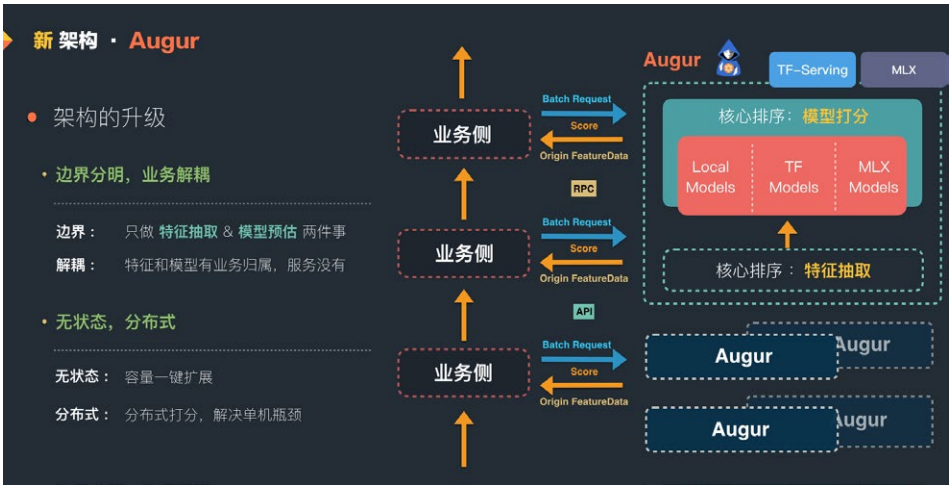

- 性能瓶颈：核心层的模型预估的 Size 扩展到数千级别文档的时候，单机已经难以承载；近百万个特征值的传输开销已经难以承受。
- 复用困难：模型预估能力已经成为一个通用的需求，单搜索就有几十个场景都需要该能力；而老逻辑的业务耦合性让复用变得更加困难。
- 平台缺失：快速的业务迭代下，需要有一个平台可以帮助业务快速地进行模型和特征的管理，包括但不限于配置、上线、灰度、验证等等。


## 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：

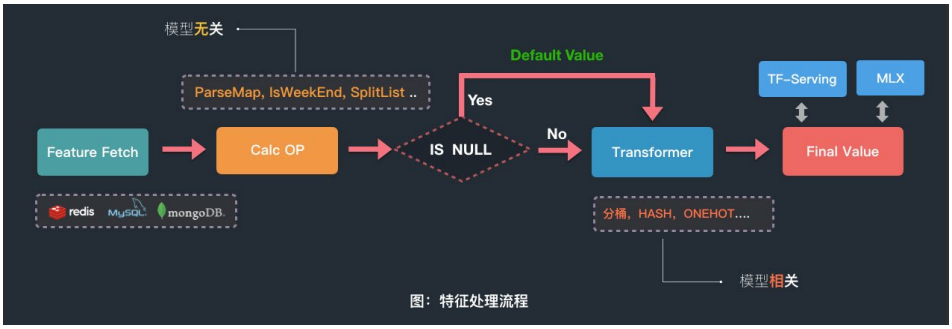
- IO OP：用处理原始特征的获取，如从 KV 里获取数据，或者从对应的第三方服务中获取数据。内置批量接口，可以实现批量召回，减少 RPC。
- Calc OP：用于处理对获取到的原始特征做与模型无关的逻辑处理，如拆分、判空、组合。业务可以结合需求实现特征处理逻辑。

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。

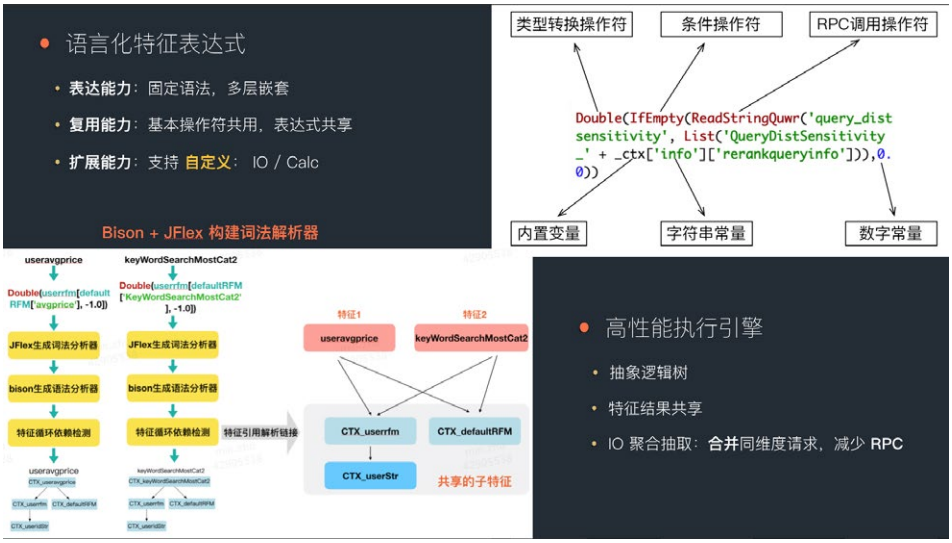
基于这两个概念，Augur 中特征的处理流程如下所示：首先，我们会进行特征抽取，抽取完后，会对特征做一些通用的处理逻辑；而后，我们会根据模型的需求进行二次变换，并将最终值输入到模型预估服务中。如下图所示：



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。

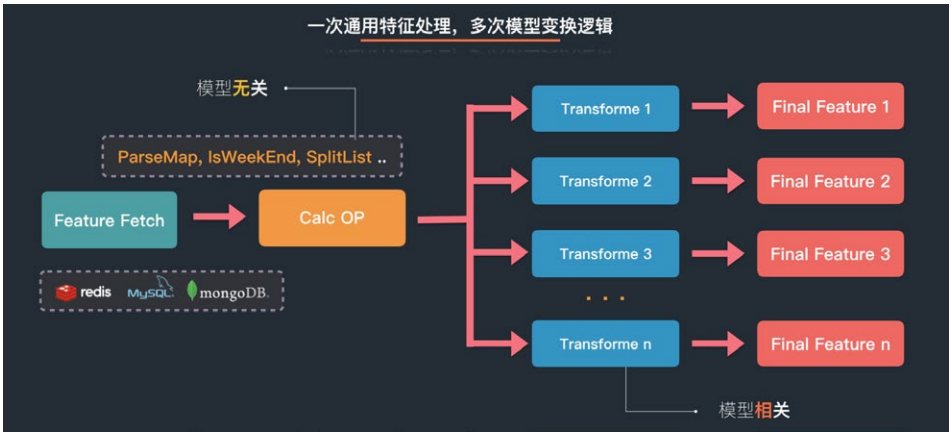
### 4.1.3 配置化的模型表达

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

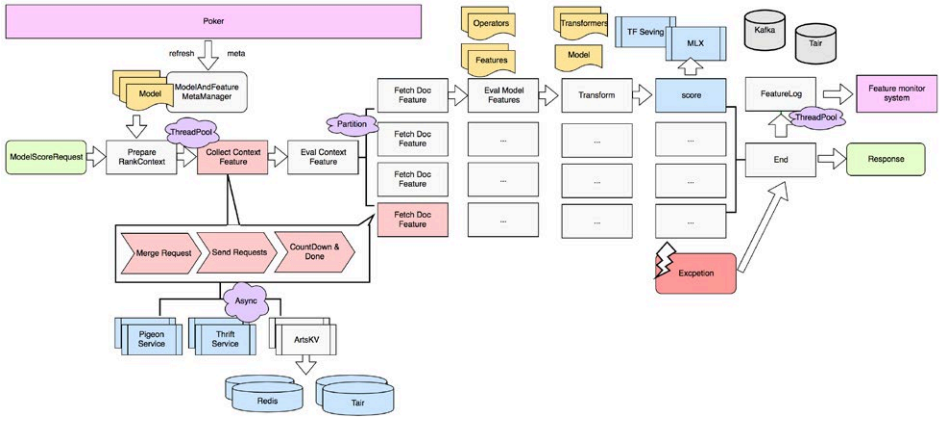
### 4.2.1 高效的模型预估过程

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。

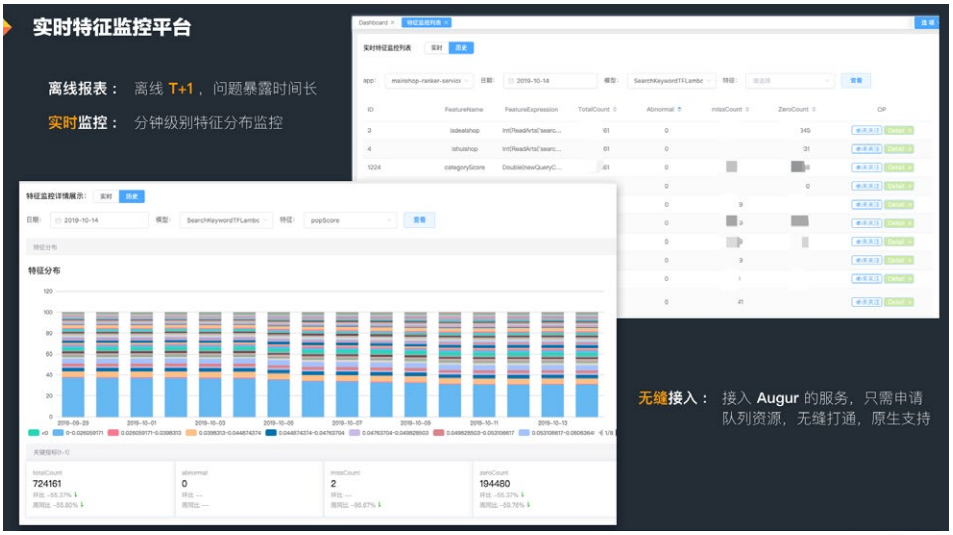
```
_I(_I('xxx'))
Benchmark              Mode  Cnt  Score   Error  Units
AbsBenchmarkTest.test  avgt   25  1.644 ± 0.009  ms/op
```

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：

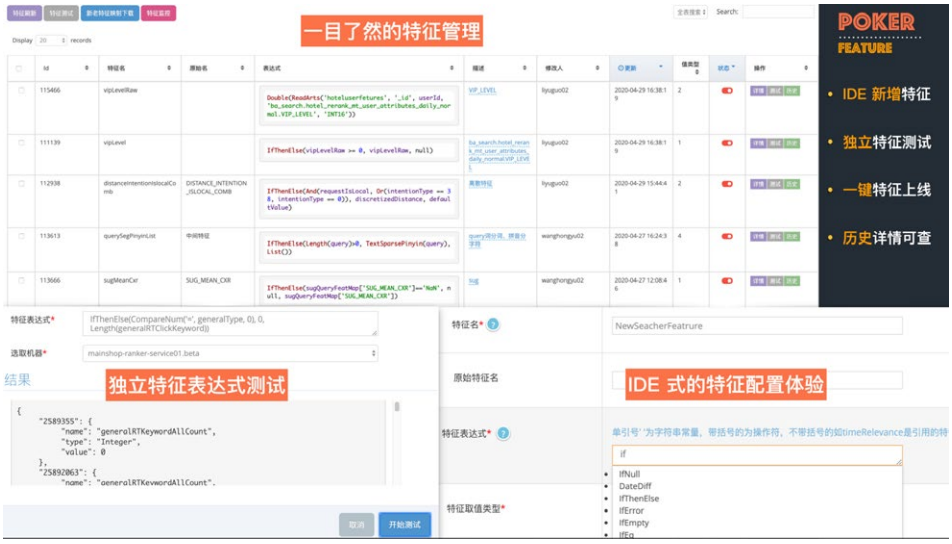

当然，无论使用哪一种方式去构建预估服务，都可以在美团内部的 Poker 平台上进行服务、模型与特征的管理。

### 4.3.2 Augur 管理平台 Poker 的构建

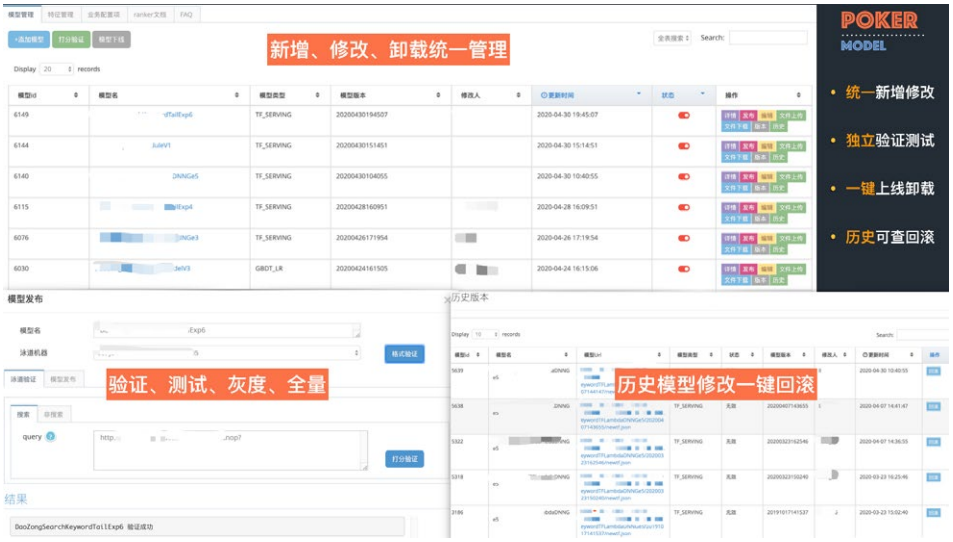
实现一个框架价值的最大化，需要一个完整的体系去支撑。而一个合格的在线预估平台，需要一个产品级别的管理平台辅助。于是我们构建了 Poker（搜索实验平台），其中的在线预估服务管理模块，也是 Augur 的最佳拍档。Augur 是一个可用性较高的在线预估框架，而 Poker+Augur 则构成了一个好用的在线预估平台。下图是在线预估服务管理平台的功能架构：


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。

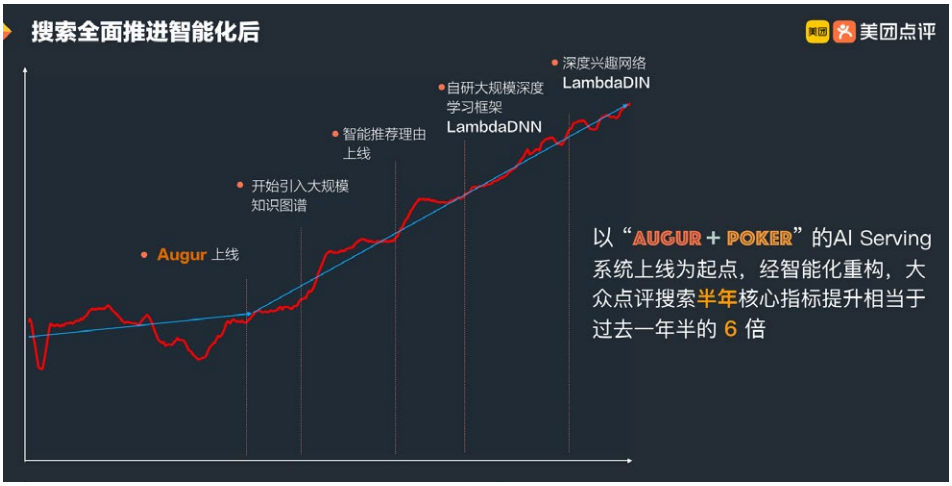



### 4.3.3 Poker + Augur 的应用与效果

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]。

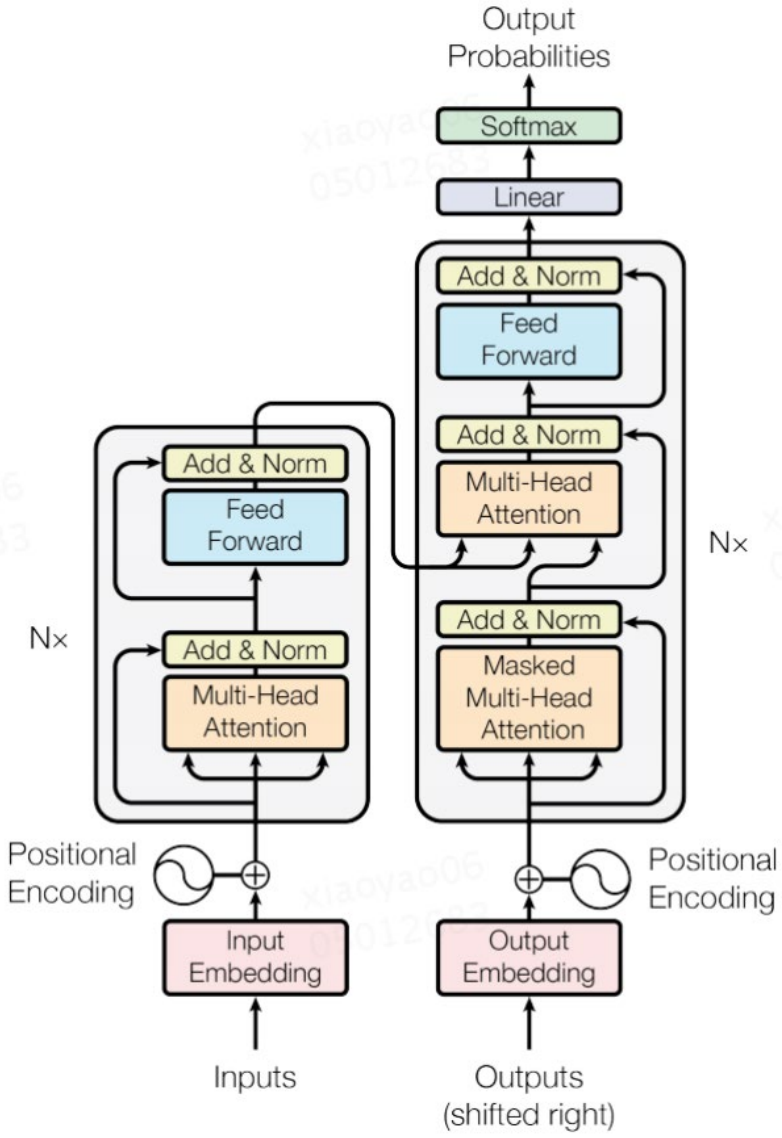



图 1　Transformer 结构示意图

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：

$$\mathbf{EW}^{Q}\ \ \mathbf{EW}^{K}\ \ \mathbf{EW}^{V}$$

然后将投影后的矩阵输入到 Multi-Head Attention。计算公式如下：

$$head_{i}=Attention\left(\mathbf{EW}_{i}^{Q},\mathbf{EW}_{i}^{K},\mathbf{EW}_{i}^{V}\right)$$

$$\mathbf{S}=MH\left(\mathbf{E}\right)=Concat\left(head_{1},head_{2},...,head_{h}\right)\mathbf{W}^{H}$$

### 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：

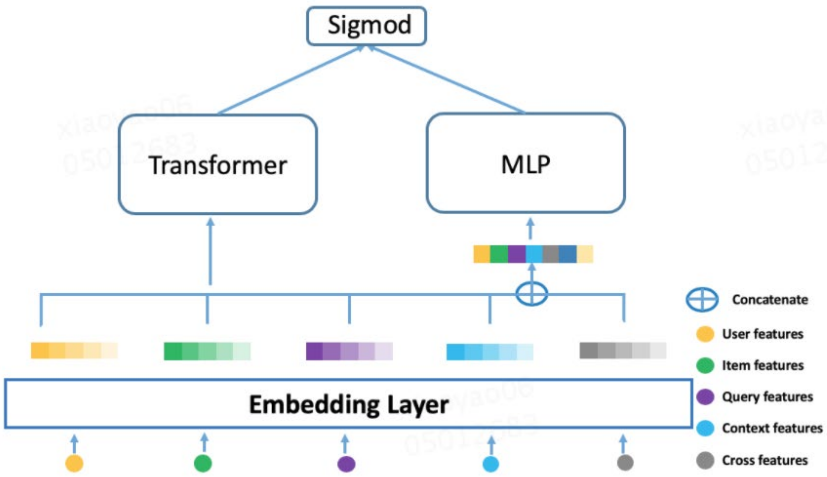



图 2　Transformer&Deep 结构示意图

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：

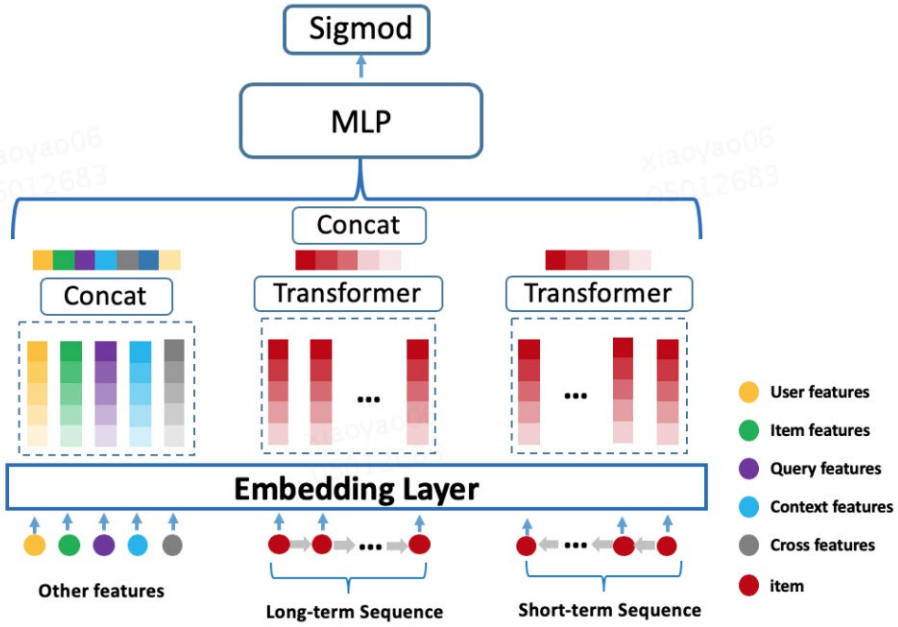



图 3　Transformer 行为序列建模

**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：

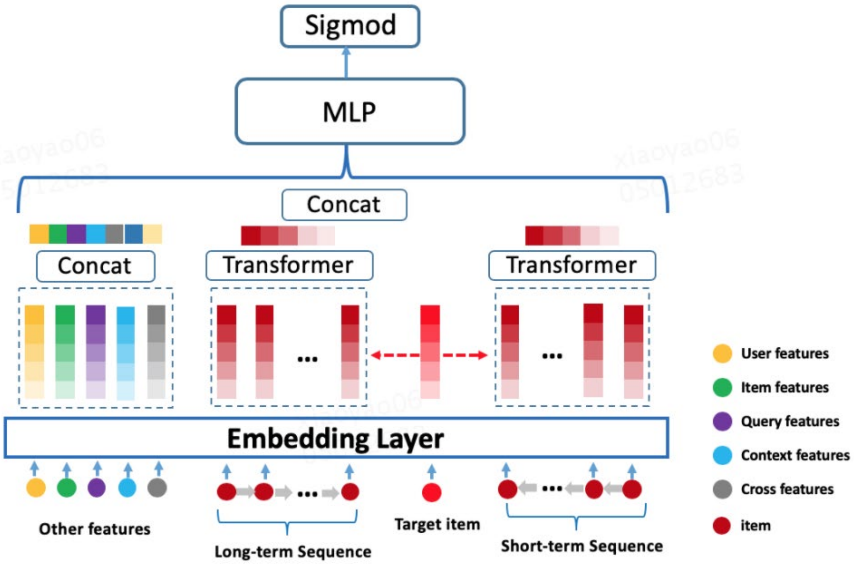



图 4　Transformer 行为序列建模

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：

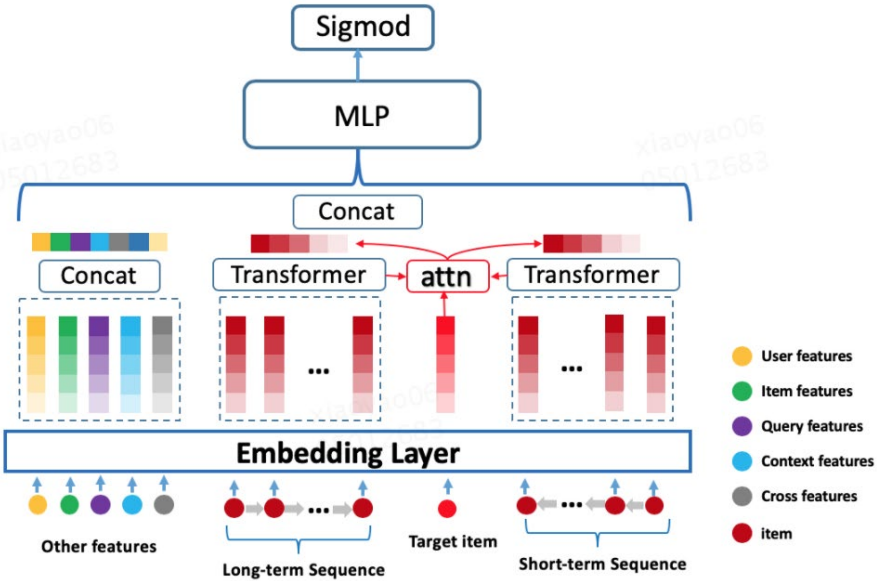



图 5　Transformer 行为序列建模

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。

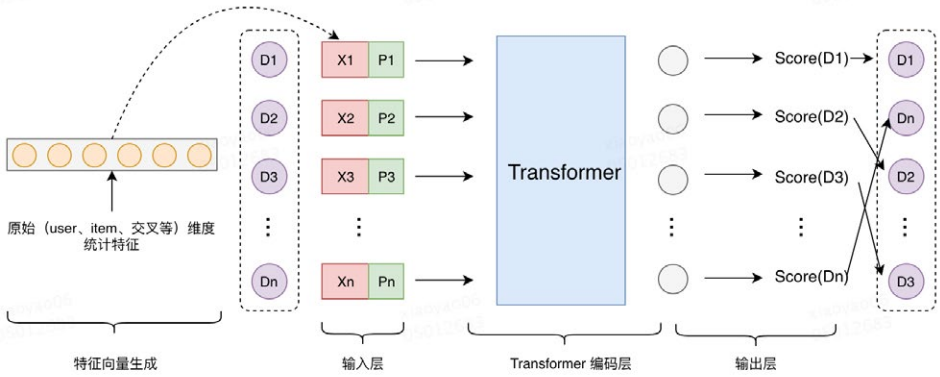



图 6　Transformer 重排序

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。

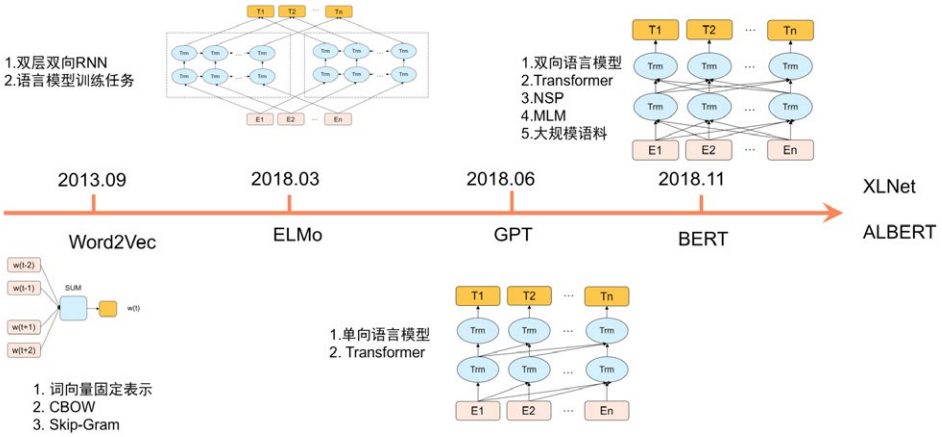



图 1　NLP 预训练发展历程

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。

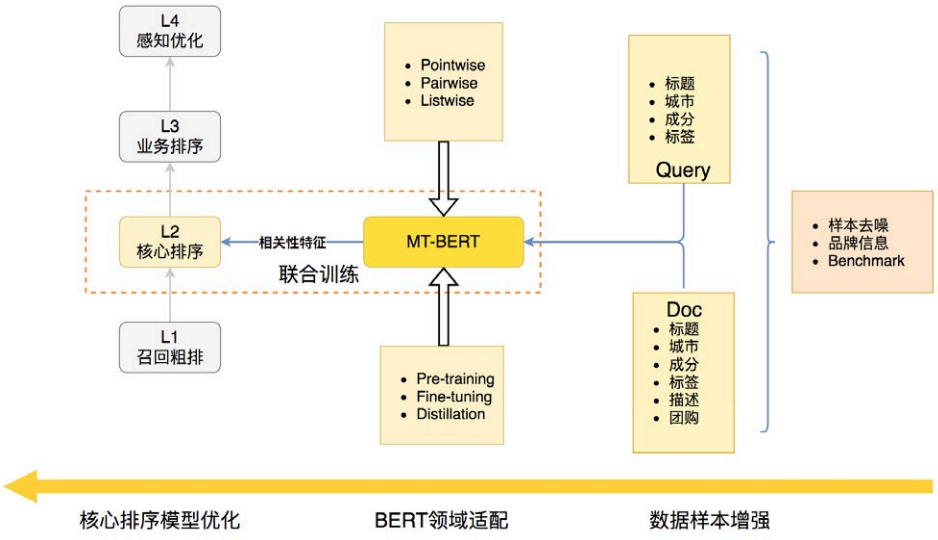



图 2　美团搜索核心排序相关性优化技术架构图

## 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。

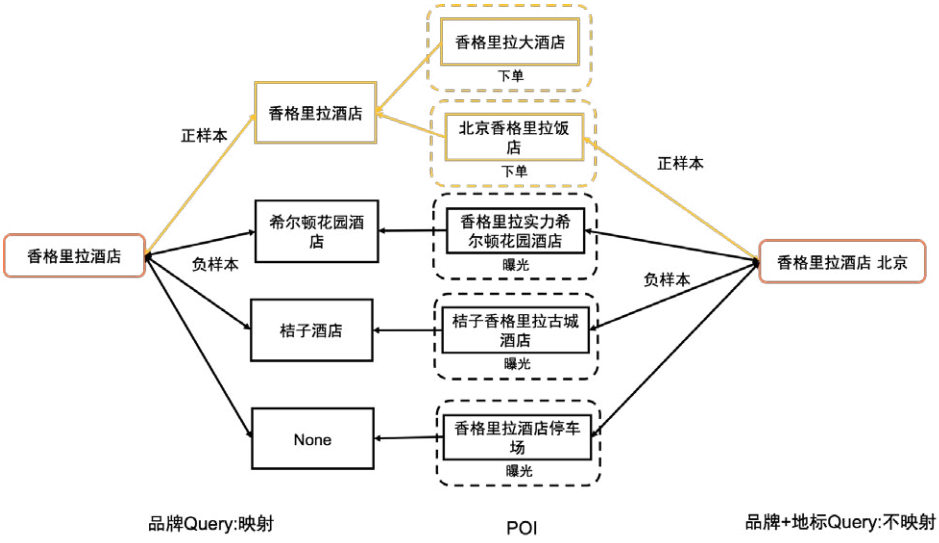



图 3　POI 品牌信息优化样本示意图

## 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。

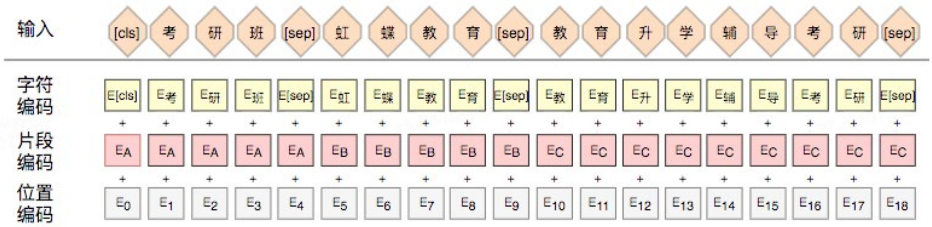



图 4　BERT 输入部分加入 Doc（POI）品类信息

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：

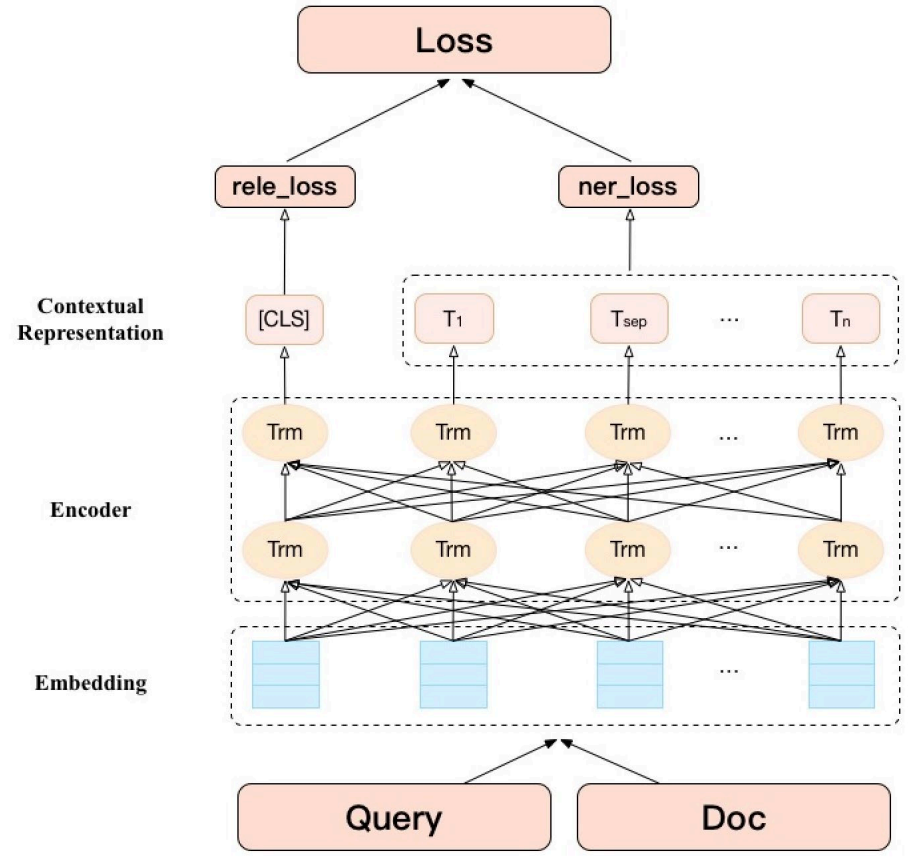



图 5　实体成分一致性学习模型结构

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：

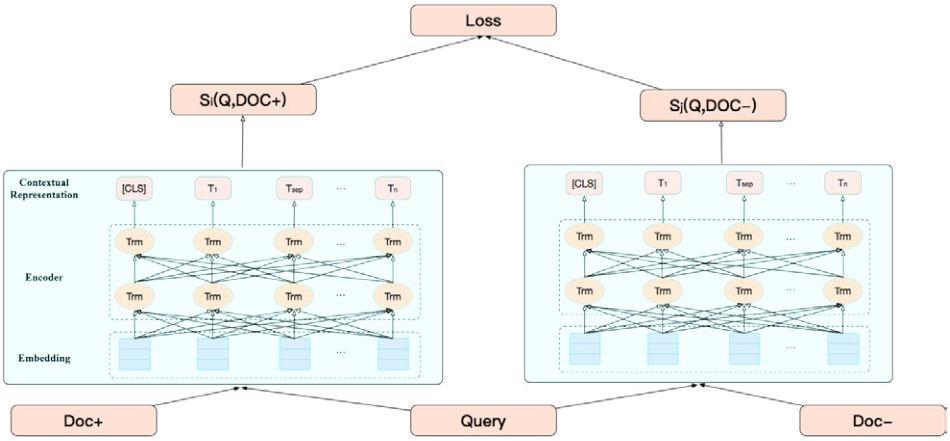



图 6　Pairwise Fine-tuning 模型结构

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 $P_{ij}$ 为同一个 Query 下 $Doc_i$ 相比 $Doc_j$ 更相关的概率，其中 $s_i$ 和 $s_j$ 分别为 $Doc_i$ 和 $Doc_j$ 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：

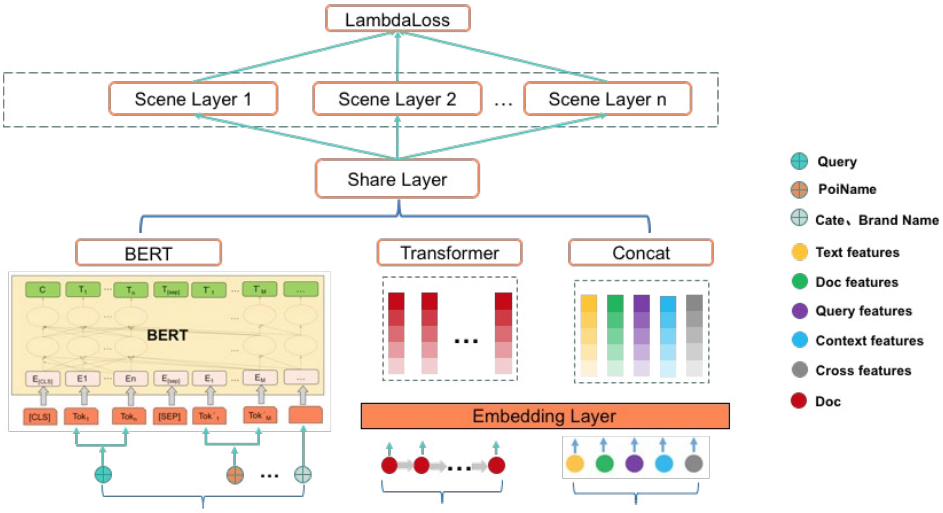



图 7　联合训练模型结构

**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

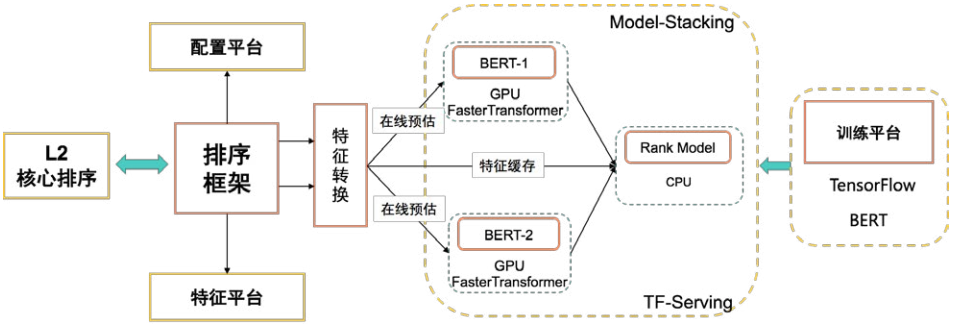



图 8　核心排序框架图

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

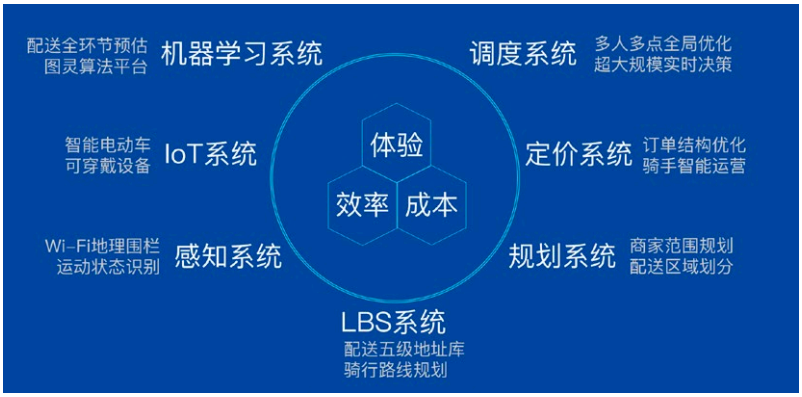
美团配送业务场景复杂，单量规模大。下图这组数字是 2019 年 5 月美团配送品牌发布时的数据。


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。

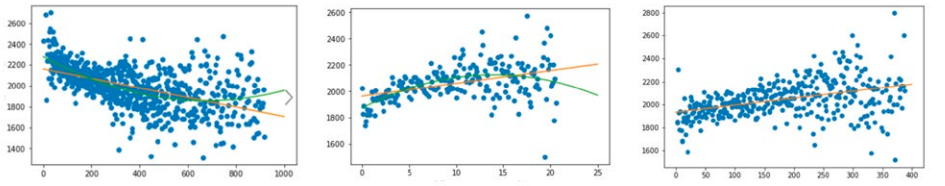
即时配送场景也类似，每个骑手需要尽量固定地去熟悉一片商家或者配送区域。同时，对于管理而言，站点的管理范围也比较明确。另外，如果有新商家上线，也很容易确定由哪个配送站来提供服务。所以，这个问题有很多运营管理的诉求在其中。


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。

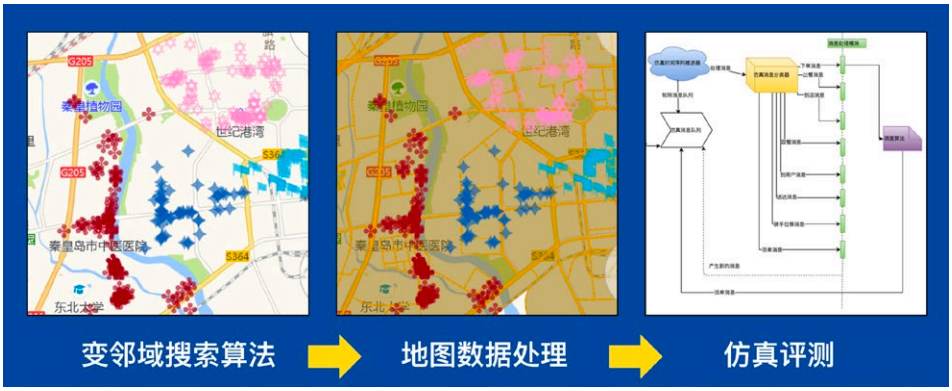
第一点，首先要确定优化目标。在很多比较稳定或者传统的业务场景中，目标非常确定。而在区域规划这个场景中，怎么定义优化目标呢？首先，我们要思考的是区域规划主要影响的是什么。从刚才几类问题的分析可以发现，影响的主要是骑手的顺路性、空驶率，也就是骑手平均为每一单付出的路程成本。所以，我们将问题的业务目标定为优化骑手的单均行驶距离。基于现有的大量区域和站点积累的数据，做大量的统计分析后，可以定义出这样几个指标：商家聚合度、订单的聚合度、订单重心和商家重心的偏离程度。数据分析结果说明，这几个指标和单均行驶距离的相关性很强。

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**


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。

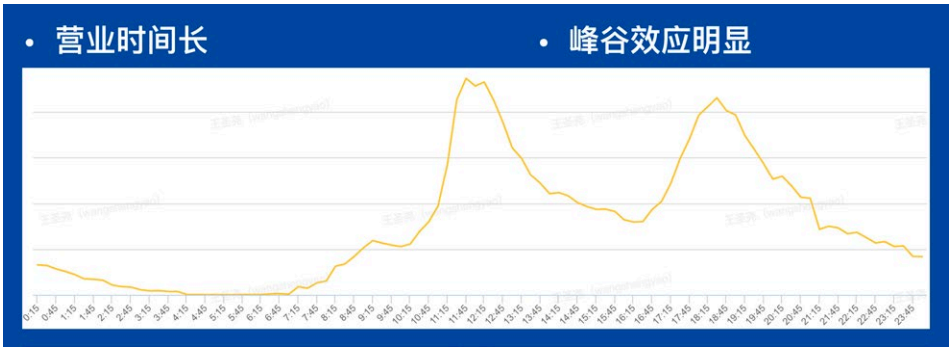
下面是一个实际案例，我们用算法把一个城市做了重新的区域规划。当然，这里必须要强调的是，在这个过程中，人工介入还是非常必要的。对于一些算法很难处理好的边角场景，需要人工进行微调，使整个规划方案更加合理。中间的图是算法规划的结果。经过试点后，测试城市整体的单均行驶距离下降了 5%，平均每一单骑手的行驶距离节省超过 100 米。可以想象一下，在这么庞大的单量规模下，每单平均减少 100 米，总节省的路程、节省的电瓶车电量，都是一个非常可观的数字。更重要的是，可以让骑手自己明显感觉到自己的效率得到了提升。


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。

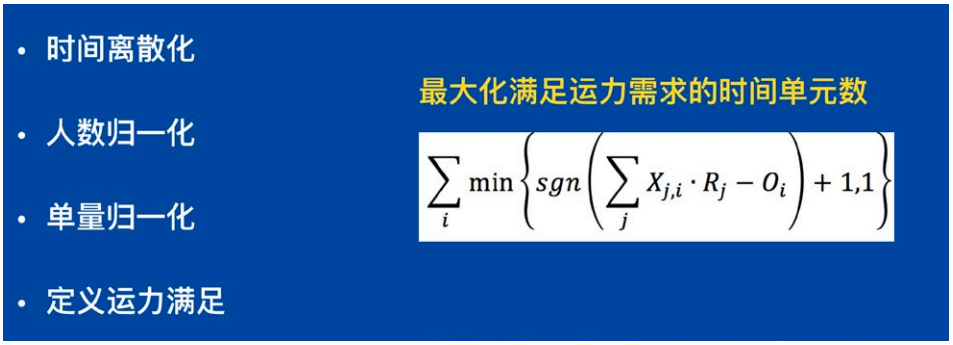
对于排班，存在两类方案的选型问题。很多业务的排班是基于人的维度，好处是配置的粒度非常精细，每个人的工作时段都是个性化的，可以考虑到每个人的诉求。但

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。


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。



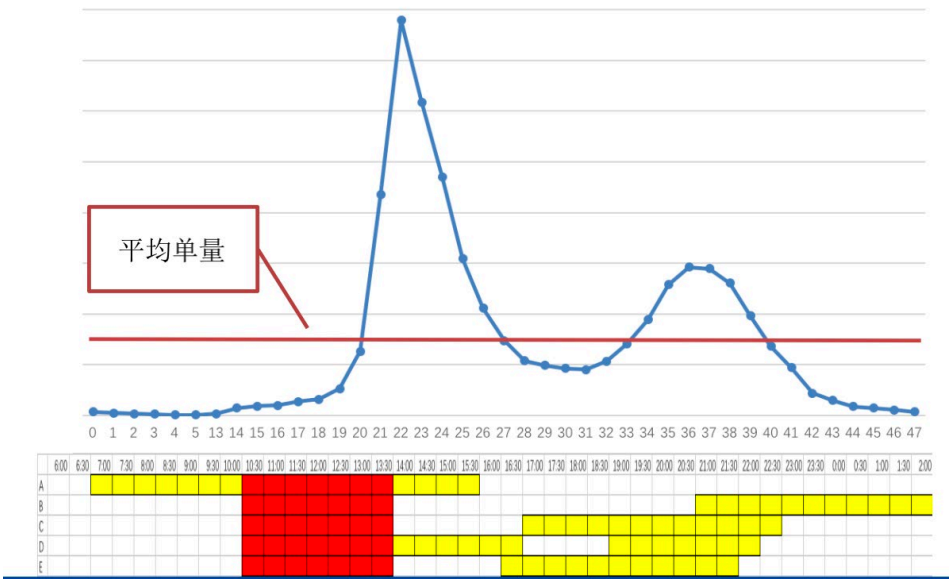

决策变量及目标设计

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。


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

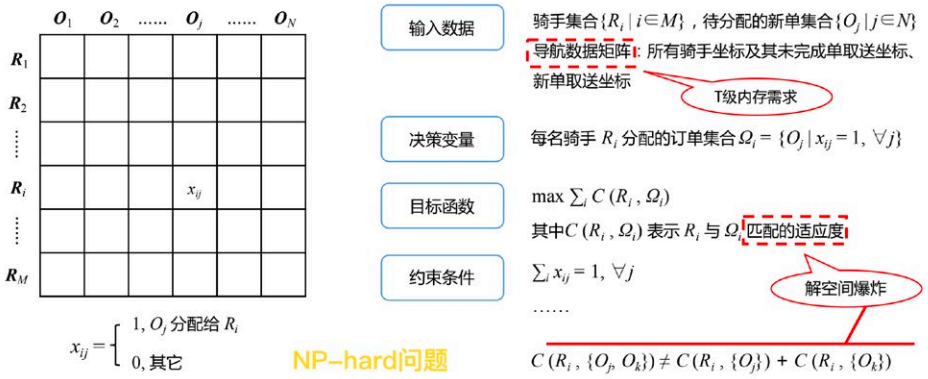
配送调度场景，可以用数学语言描述。它不仅是一个业务问题，更是一个标准的组合优化问题，并且是一个马尔可夫决策过程。

在时间段$\Delta T$内，新订单连续动态到达；对于时刻$t$，待分配订单集合为$R(t)=\{r_i|i=1,\dots,n_t\}$。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$\{q_j,j=1\dots m\}$的位置$D(t)=\{d_j(t),j=1,\dots,m\}$，$n_t$个订单的取餐位置$P^+(t)=\{1^+,2^+,\dots,n_t{}^+\}$，送餐位置$P^-(t)=\{1^-,2^-,\dots,n_t{}^-\}$。$A(t)$为$t$时刻的弧集$A(t)=\{arc(a,b)|\forall a,b\in V(t),\ a\neq b\}$，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$\Delta T$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$S_j(t)$，使得$\Delta T$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

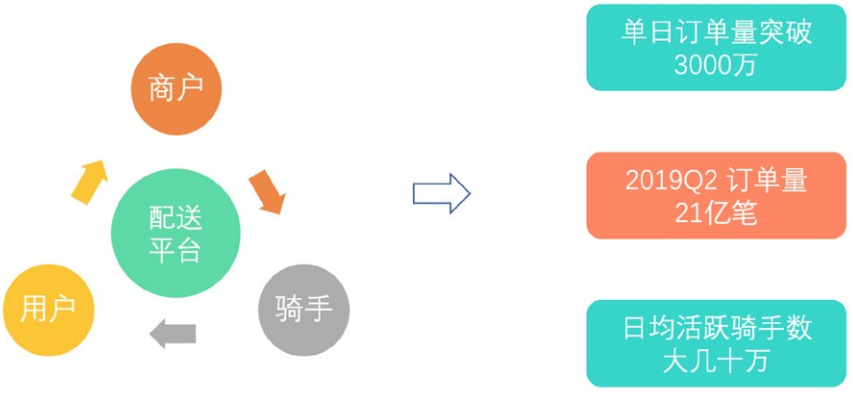
本文根据美团配送资深技术专家郑艳伟在 2019 SACC（中国系统架构师大会）上的演讲内容整理而成，主要介绍了美团配送技术团队在建设一站式机器学习平台过程中的经验总结和探索，希望对从事此领域的同学有所帮助。

## 0. 写在前面

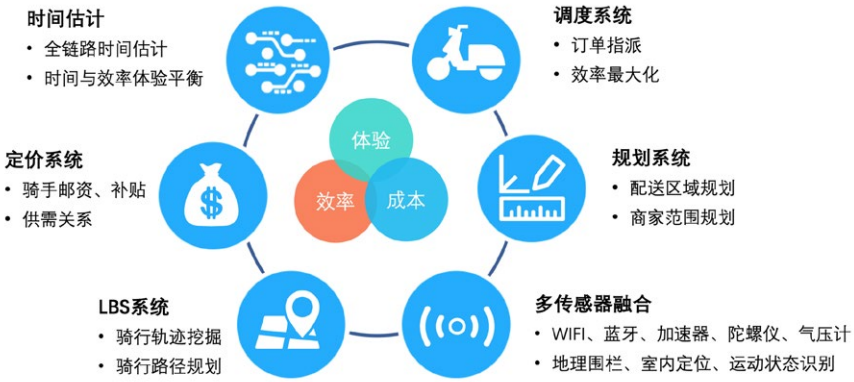
AI 是目前互联网行业炙手可热的“明星”，无论是老牌巨头，还是流量新贵，都在大力研发 AI 技术，为自家的业务赋能。配送作为外卖平台闭环链条上重要的一环，配送效率和用户体验是配送业务的核心竞争力。随着单量上涨、骑手增多、配送场景复杂化，配送场景的各种算法在更快（算法需要快速迭代、快速上线）、更好（业务越来越依赖机器学习算法产生正向的效果）、更准（算法的各种预测如预计送达时间等，需要准确逼近真实值）的目标下也面临日益增大的挑战。算法从调研到最终上线发挥作用，需要有一系列的工程开发和对接，由此引发了新的问题：如何界定算法和工程的边界，各司其职，各善其长？如何提升算法迭代上线的速度和效率？如何快速准确评估算法的效果？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：

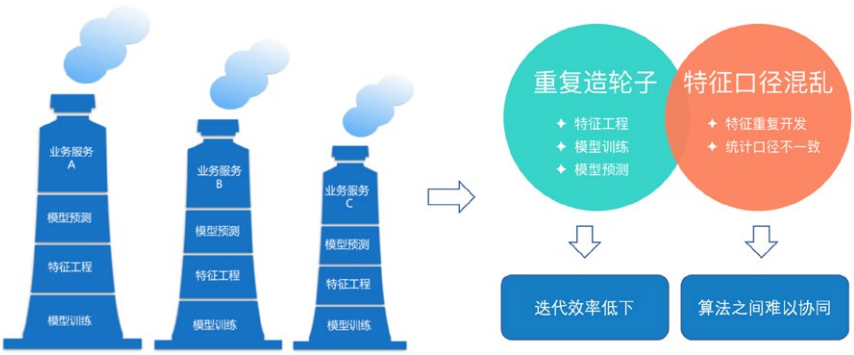
**MVP 阶段**：灵活，快速试错，具备快速迭代能力。

**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

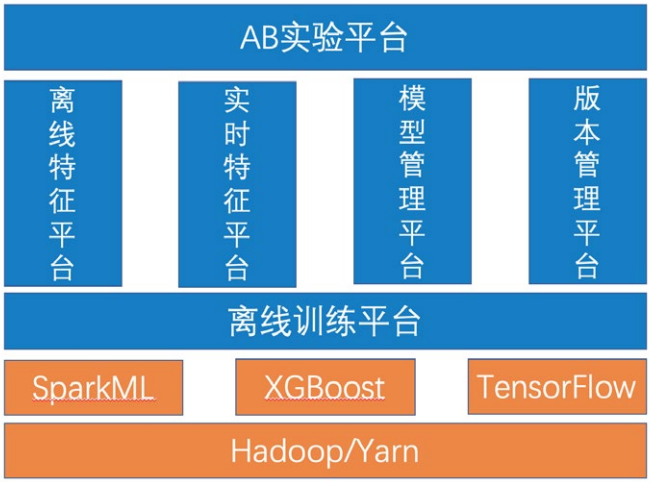
初始阶段，大家对机器学习平台要发展成什么样子并不明确，很多事情也想不清楚。但是为了支撑业务的发展，必须快速上线、快速试错。因此，在此阶段，各个业务线独自建设自己的机器学习工具集，按照各自业务的特殊需求进行各自迭代，快速支持机器学习算法上线落地应用到具体的业务场景，也就是我们所熟知的“烟囱模式”。此种模式各自为战，非常灵活，能够快速支持业务的个性化需求，为业务抢占市场赢得了先机。但随着业务规模的逐渐扩大，这种“烟囱模式”的缺点就凸显了出来，主要表现在以下两个方面：

- **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。

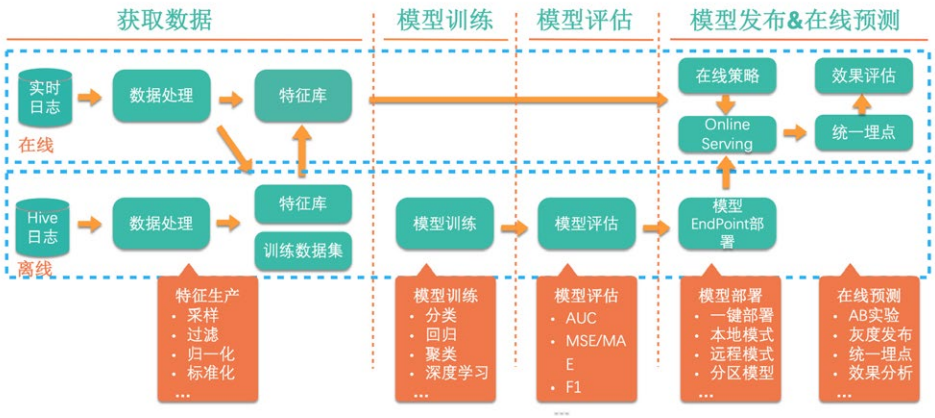
1）首先在获取数据阶段，支持在线和离线两个层面的处理，分别通过采样、过滤、归一化、标准化等手段生产实时和离线特征，并推送到在线的特征库，供线上服务使用。

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。



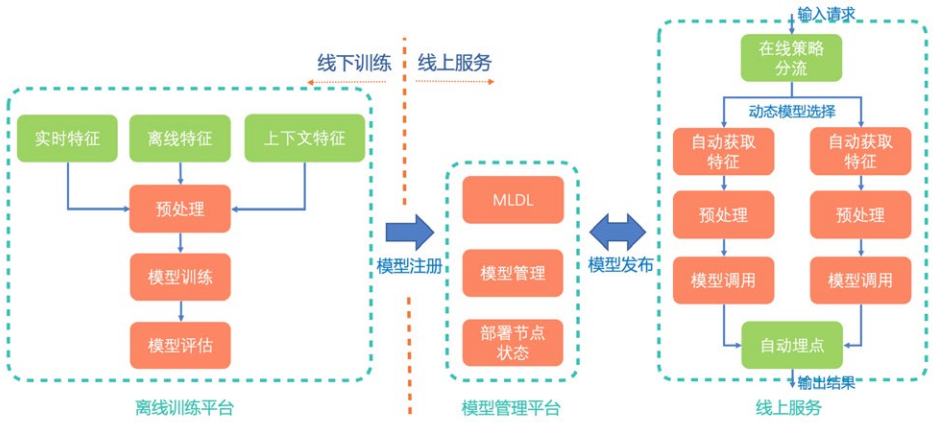

## 3.1 离线训练平台

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。

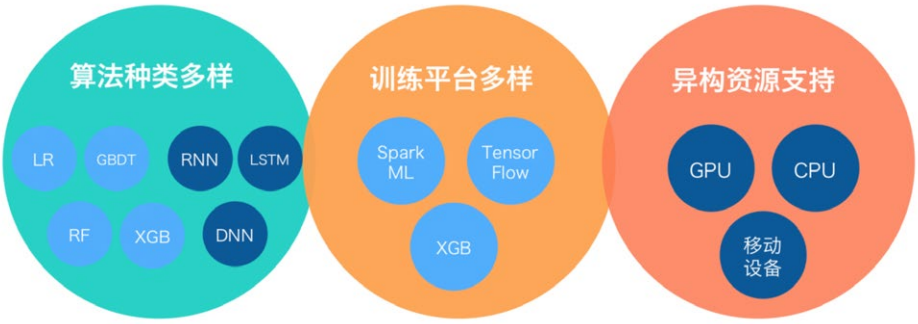

我们的离线训练平台在产出模型时，除了产出模型文件之外，还产出了一个 MLDL（Machine Learning Definition Language）文件，将各模型的所有预处理模块信息写入 MLDL 文件中，与模型保存在同一目录中。当模型发布时，模型文件连带 MLDL 文件作为一个整体共同发布到线上。在线计算时，先自动执行 MLDL 中的预处理逻辑，然后再执行模型计算逻辑。通过 MLDL 打通了离线训练和在线预测，贯穿整个机器学习平台，使得线下和线上使用同一套特征预处理框架代码，保证了线下和线上处理的一致性。



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

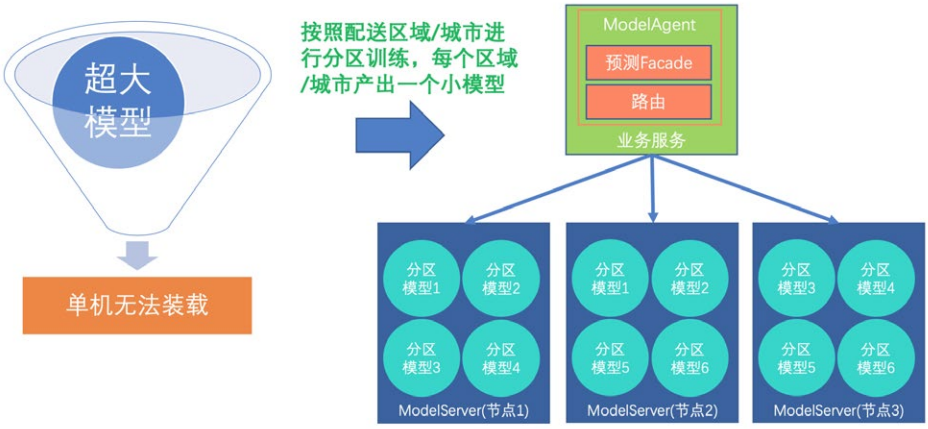
前面介绍了，我们的图灵平台集成了 Spark ML、XGBoost、TensorFlow 三种底层训练框架，基于此，我们的训练平台产出的机器学习模型种类也非常多，简单的有 LR、SVM，树模型有 GBDT、RF、XGB 等，深度学习模型有 RNN、DNN、LSTM、DeepFM 等等。而我们的模型管理平台的目标就是提供统一的模型注册、发现、部署、切换、降级等解决方案，并为机器学习和深度学习模型提供高可用的线上预测服务。




模型管理平台支持本地和远程两种部署模式：

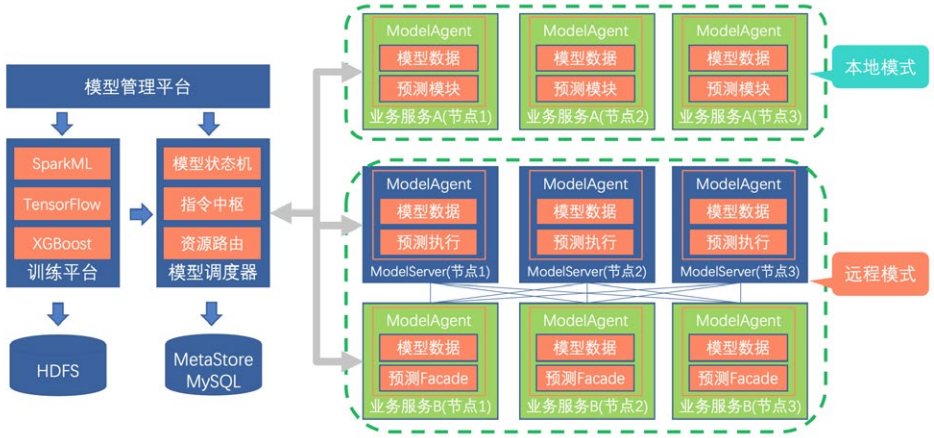
- 本地：模型和 MLDL 统一推送到业务方服务节点上，同时图灵平台提供一个 Java 的 Lib 包，嵌入到业务方应用中，业务方通过本地接口的方式调用模型计算。
- 远程：图灵平台维护了一个专用的在线计算集群，模型和 MLDL 统一部署到在线计算集群中，业务方应用通过 RPC 接口调用在线计算服务进行模型计算。

| 部署模式 | 优势 | 劣势 | 适用的业务场景 |
| --- | --- | --- | --- |
| 远程模式 | · 高度并行化<br>· 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<br>· 无网络开销 | · 占用业务方服务器资源<br>· 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 |



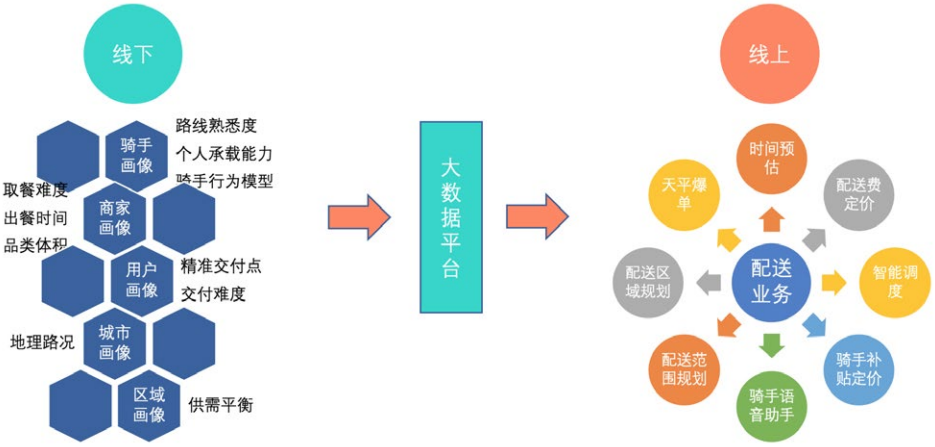
对于超大规模模型，单机无法装载，需要对模型进行 Sharding。鉴于美团配送的业务特性，可以按照配送城市 / 区域进行分区训练，每个城市或区域产出一个小模型，多个分区模型分散部署到多个节点上，解决单节点无法装载大模型的问题。分区模型要求我们必须提供模型的路由功能，以便业务方精准地找到部署相应分区模型的节点。

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。

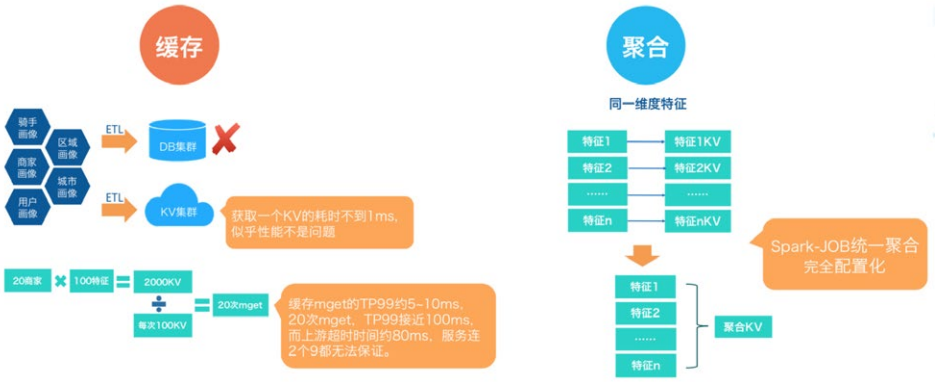



最简单的方案，直接把离线特征存储到 DB 中，线上服务直接读取 DB 获取特征 Value。读取 DB 是个很重的操作，这种方案明显不能满足互联网大并发的场景，直接被 Pass 掉。

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。

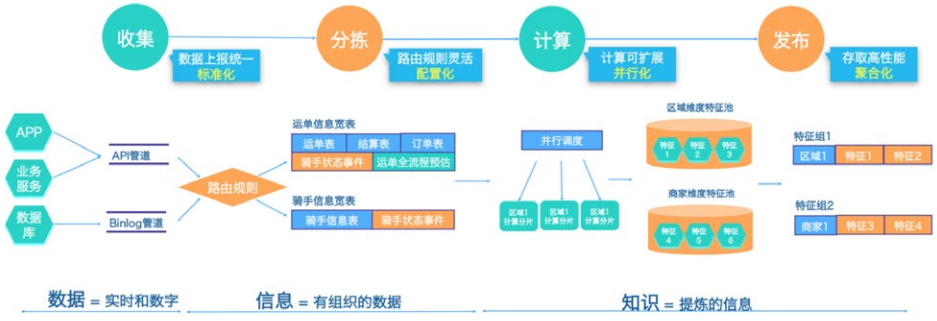
典型的业务场景：比如我们要预测 20 个商家的配送时长，假设每个商家需要 100 个特征，则我们就需要 20*100=2000 个特征进行模型计算，2000 个 KV。如果直接单个获取，满足不了业务方的性能需求；如果使用 Redis 提供的批量接口 Mget，如果每次获取 100 个 KV，则需要 20 次 Mget。缓存 mget 的耗时 TP99 约 5ms，20 次 Mget，TP99 接近 100ms，也无法满足业务方的性能需求（上游服务超时时间约 50ms）。

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。

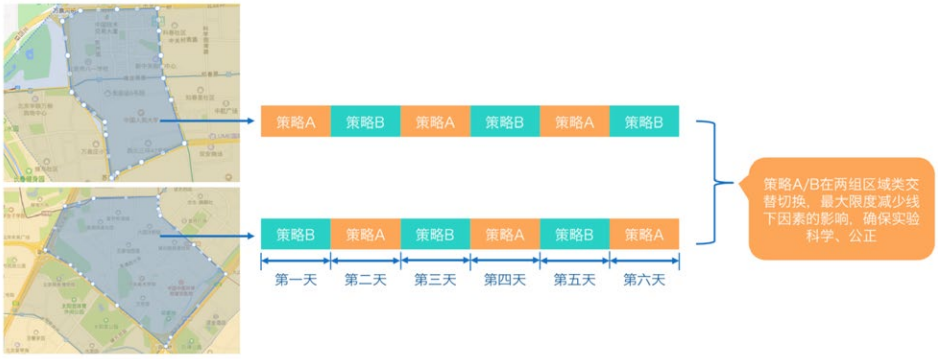
即时配送领域的 AB 实验是围绕用户、商户、骑手三者进行，用户 / 商户 / 骑手之间不再是相互独立的，而是相互影响相互制约的。针对此类场景，现有的分流方案会造成不同策略的互相干扰，无法有效地评估各个流量各个策略的优劣。


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。

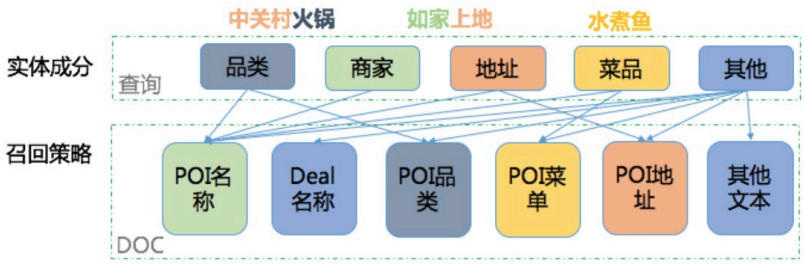



图 1　实体识别与召回策略

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。

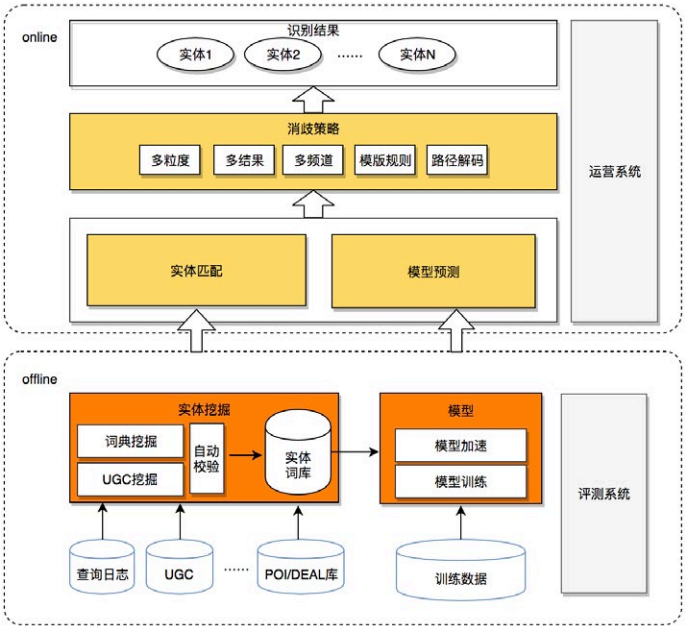



图 2　实体识别整体架构

## 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：

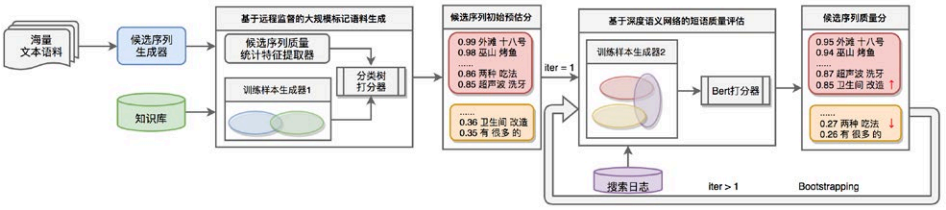



图 3　一种适用于垂直领域的新词自动挖掘方法

**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。

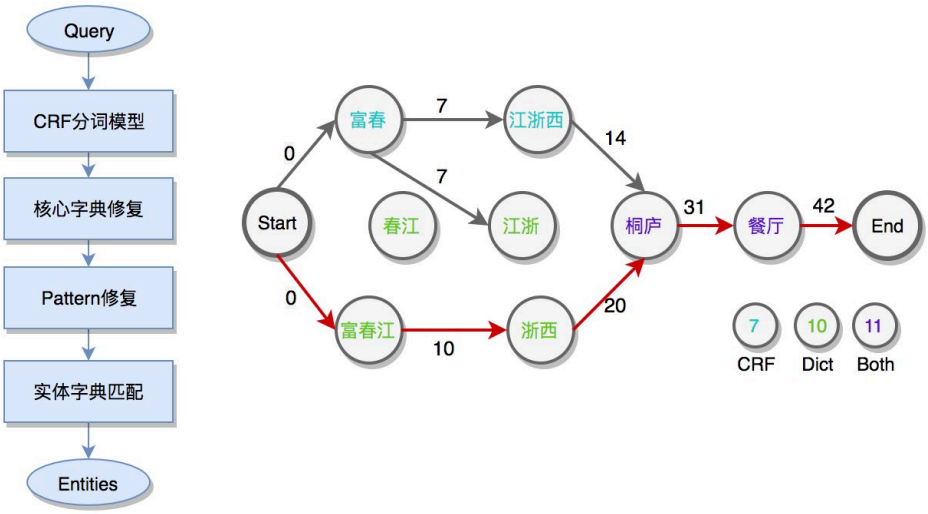



图 4　实体在线匹配

## 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。

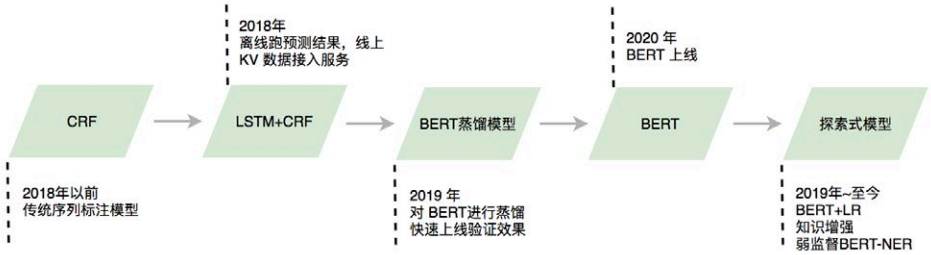



图 5　NER 模型演进

## 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。

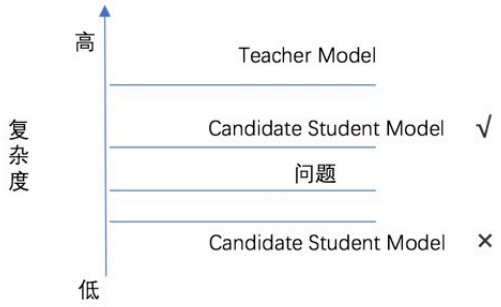



图6 Student Model复杂度选择方法

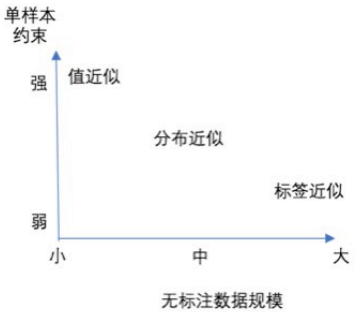



图7 标注数据量级与蒸馏约束关系

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。

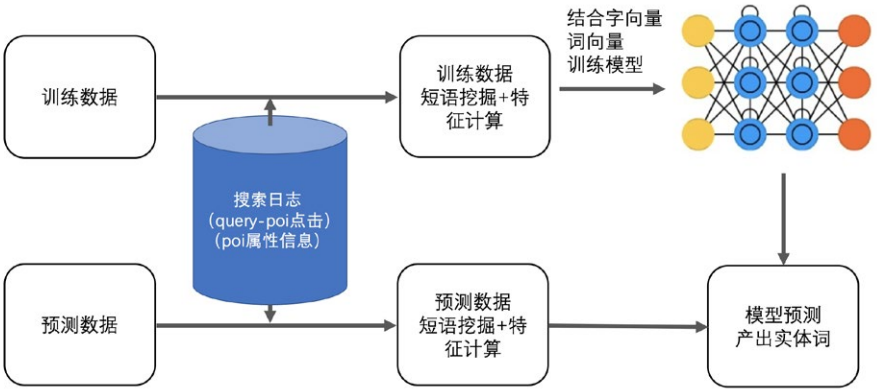



图 8　融合搜索日志特征的 Lattice-LSTM 构建流程

#### 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，$\alpha^{j-i}$ |
| | $\alpha<1$，答案倾向于选择短的子串 |
| | $\alpha>1$，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

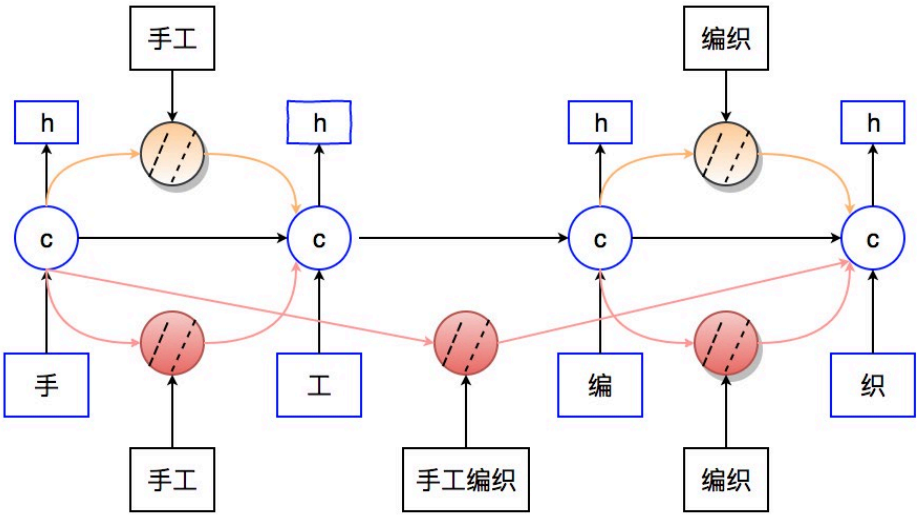



图 11　融合搜索日志特征的 Lattice-LSTM 模型结构

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。

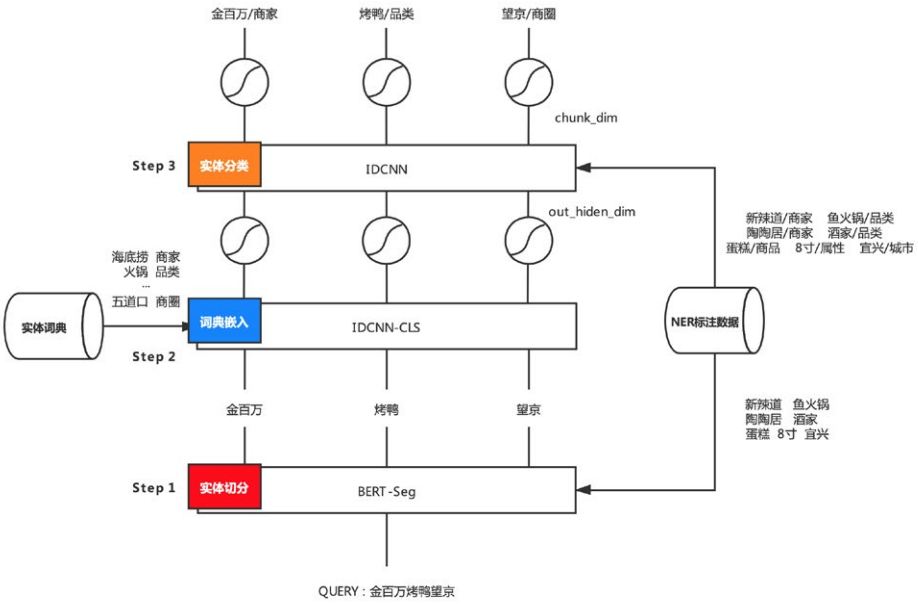



图 12　融合实体词典的两阶段 NER

## 4.3 弱监督 NER

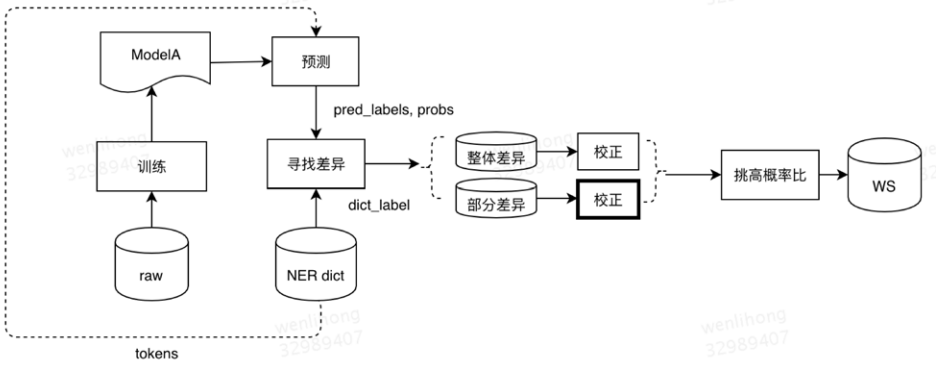



图 13　弱监督标注数据生成流程

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。

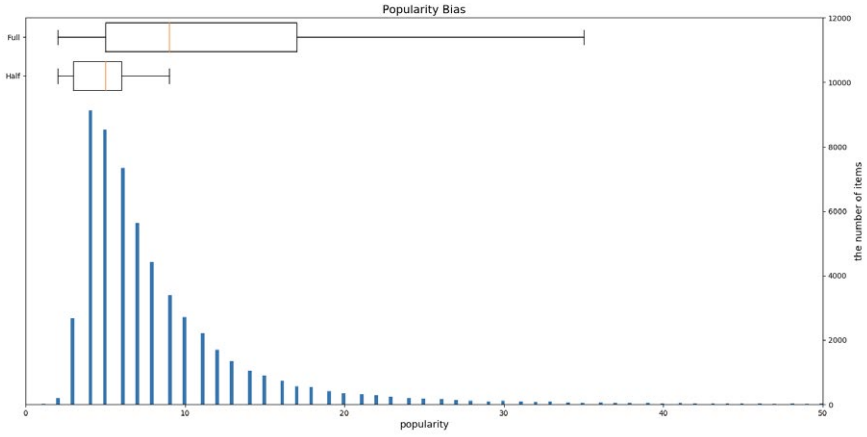



图 3　商品的流行度偏差

### 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。

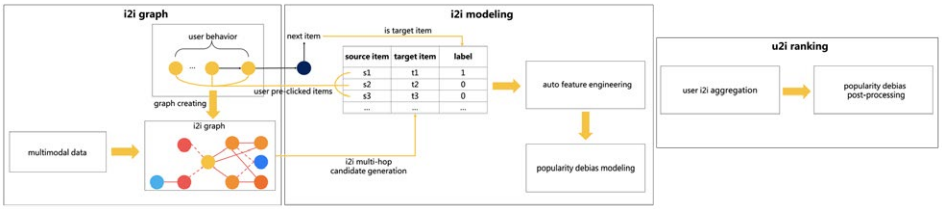



图 4　基于 i2i 建模的排序框架

## 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。

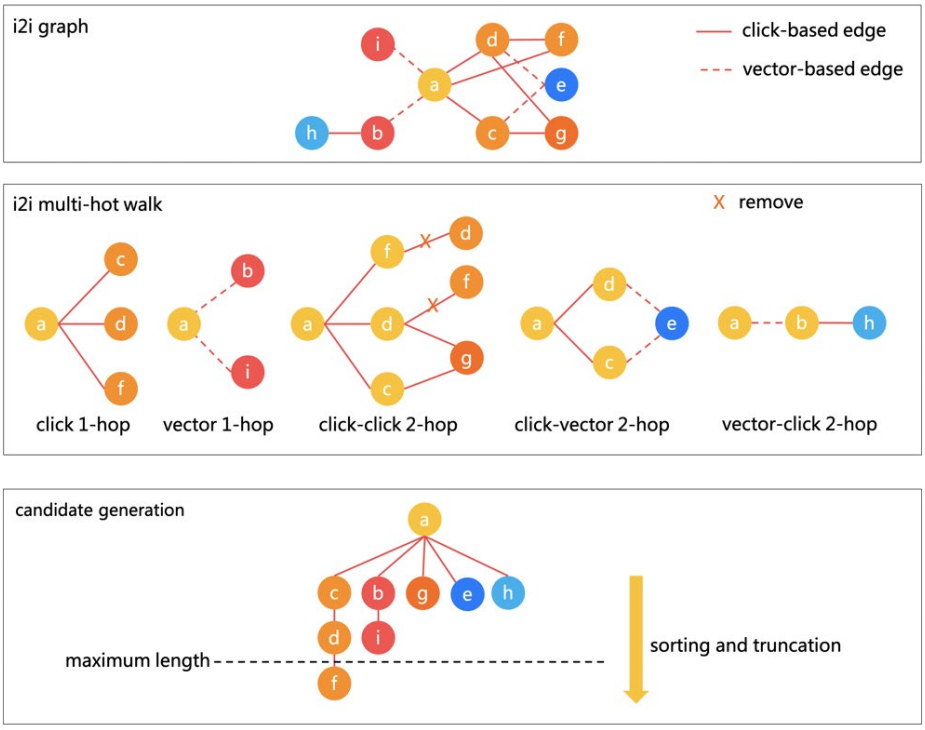
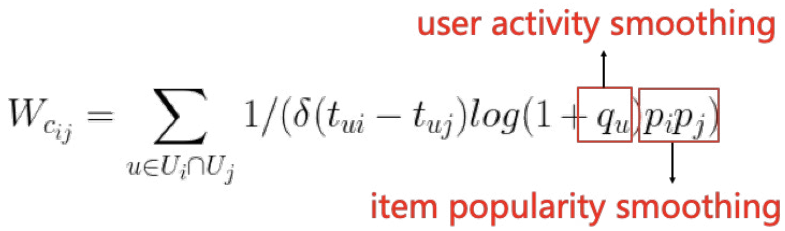



图 5　基于多跳游走的 i2i 候选样本生成

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。




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。

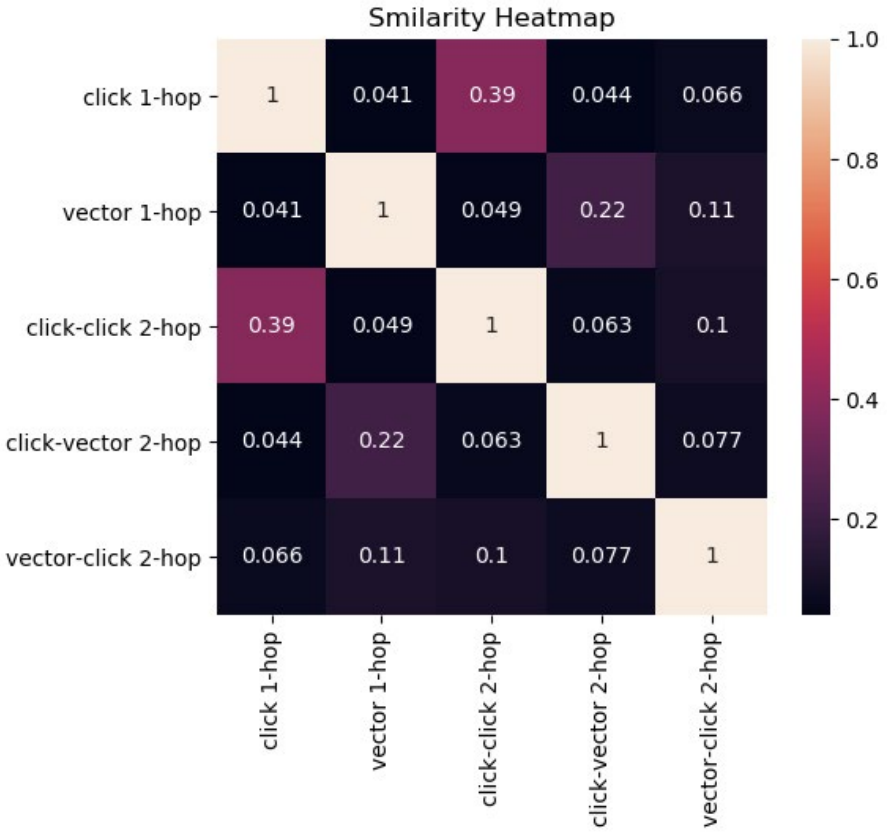


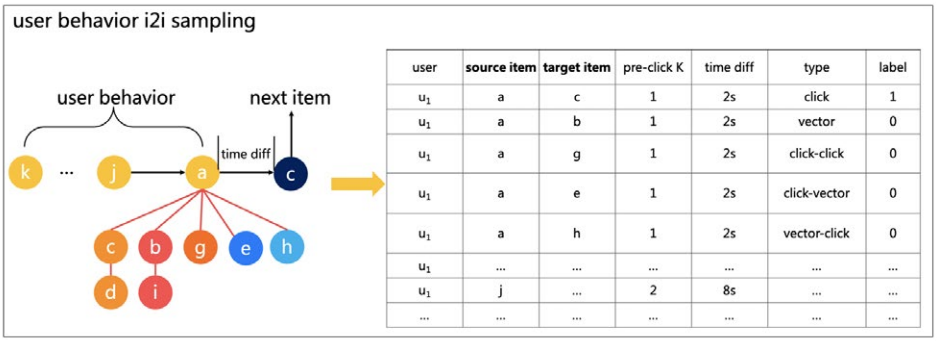

图 6　i2i 关系相似度热图

## 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。

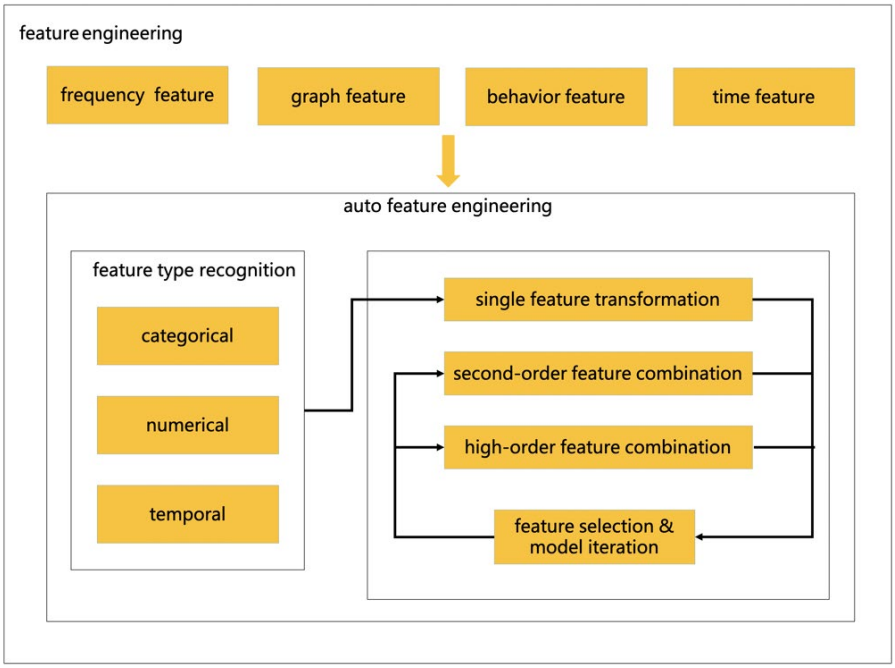



图 8　自动化特征工程

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”。

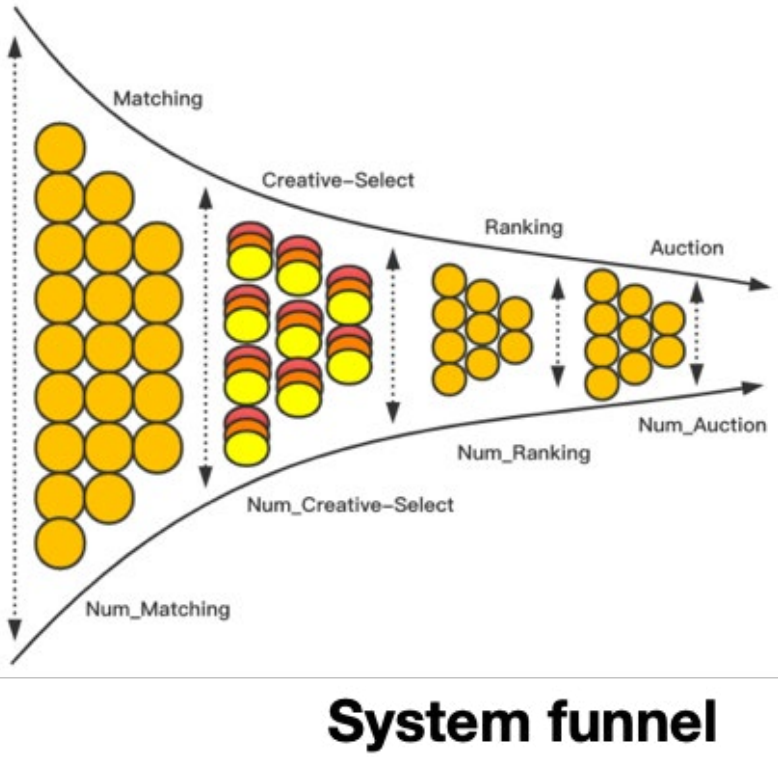



图 9　广告系统的漏斗图

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。

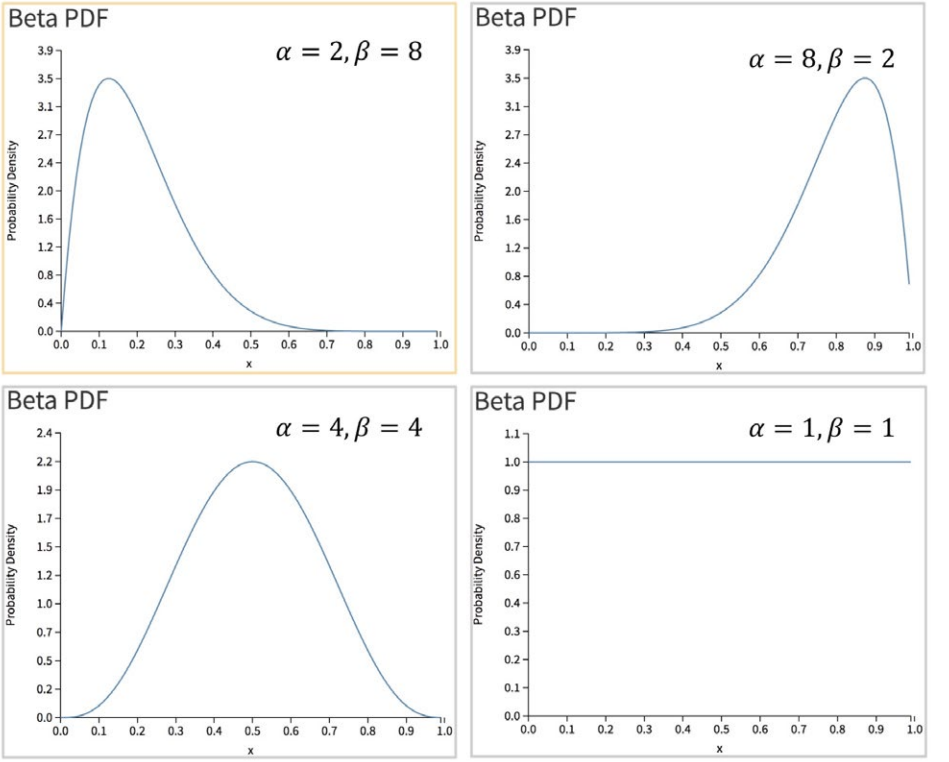



图 10　不同参数的 Beta 分布

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。

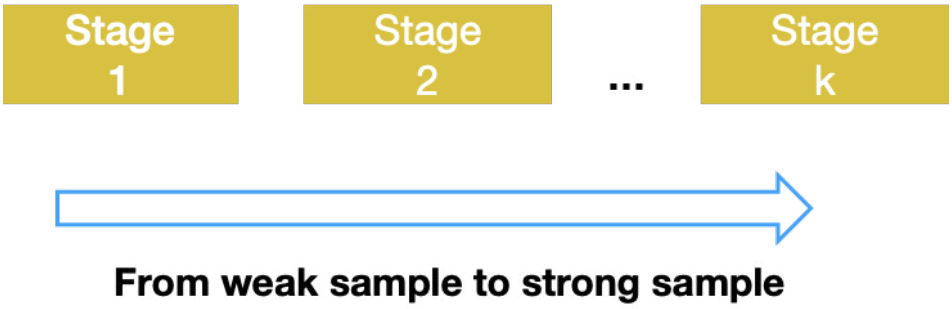



图 11　基于样本强度的多阶段训练

## 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。

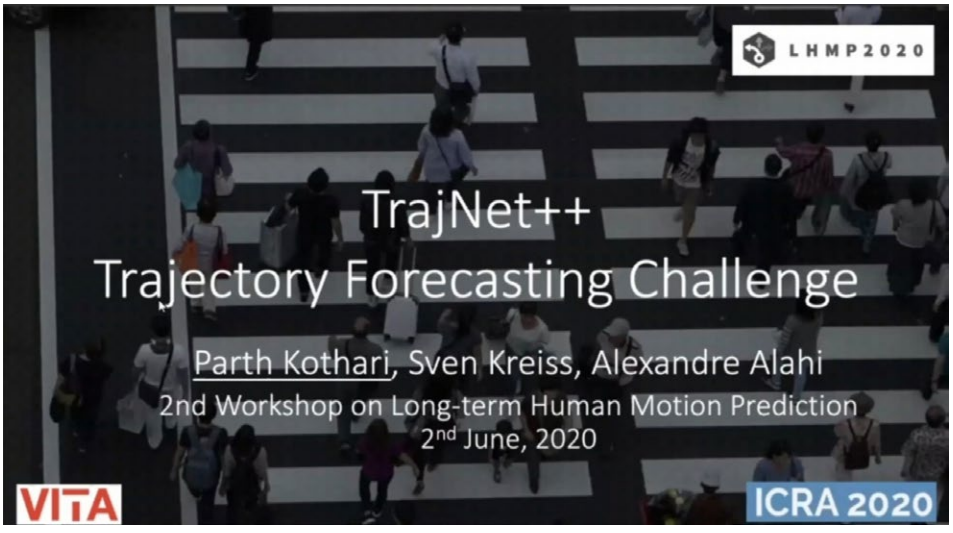



图 1　ICRA 2020 TrajNet++ 轨迹预测竞赛

## 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。

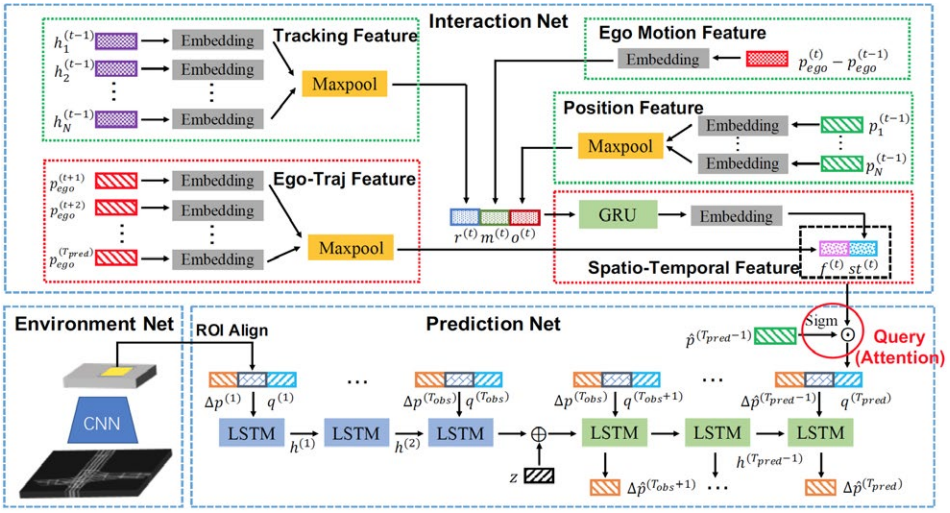



图 2　基于世界模型的预测算法

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：

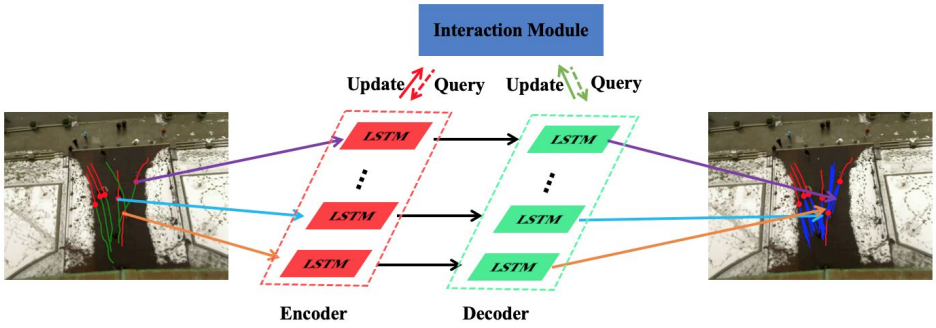



图 3　竞赛使用的基于世界模型的预测算法

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。

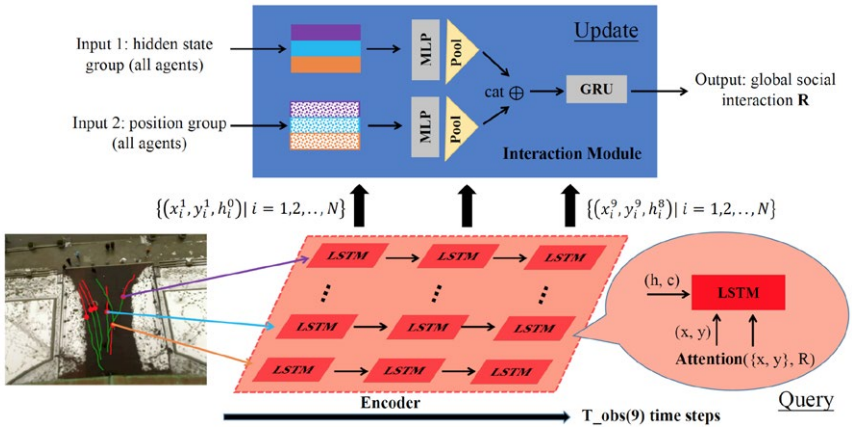



图 4　编码阶段

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。

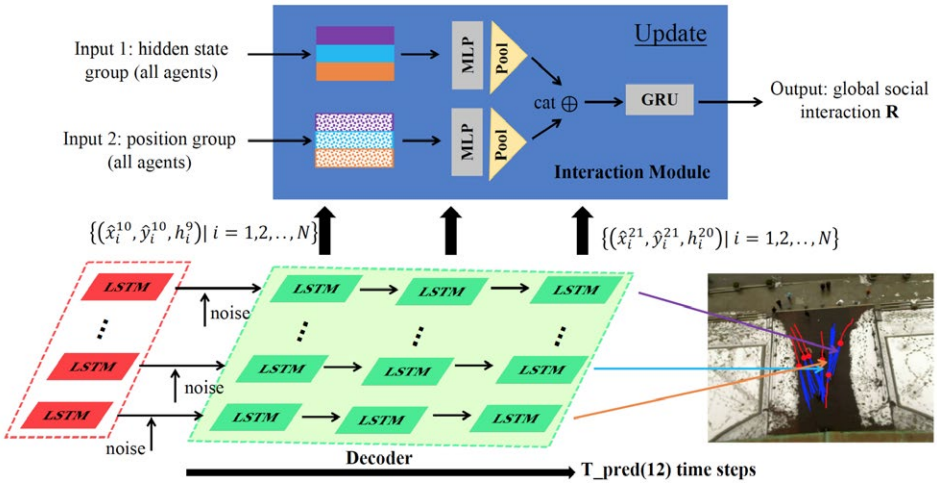



图 5　解码预测阶段

## 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。

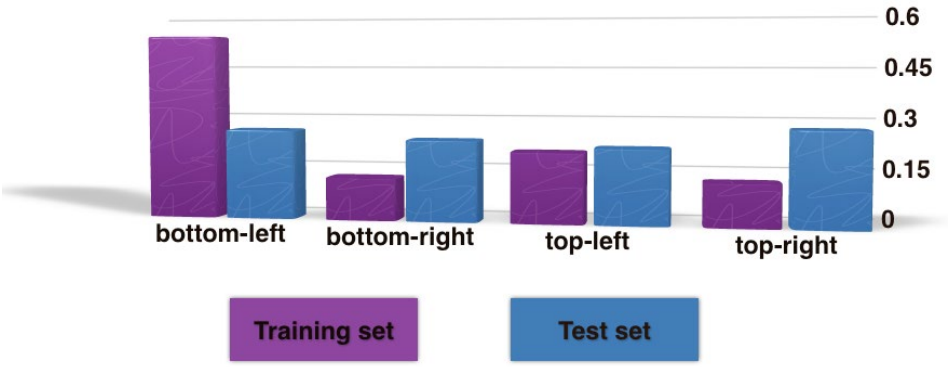



图 6　训练集与测试集历史观测轨迹中行人运动方向分布

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。

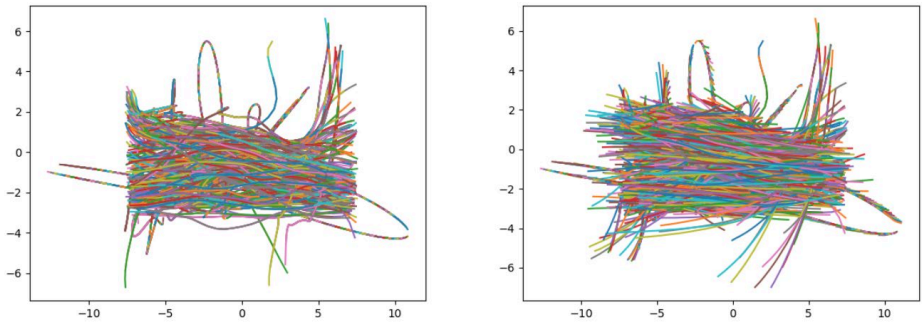



图 7　训练轨迹的可视化

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。

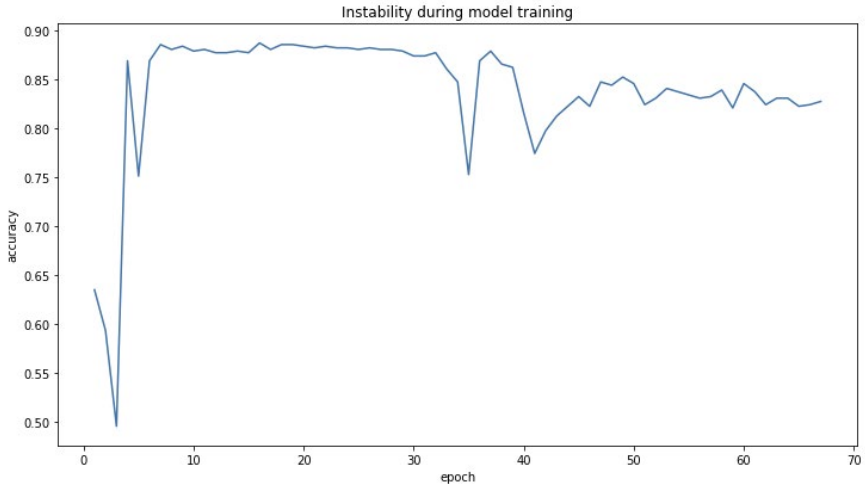



图 3　模型在训练过程中的不稳定性

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。

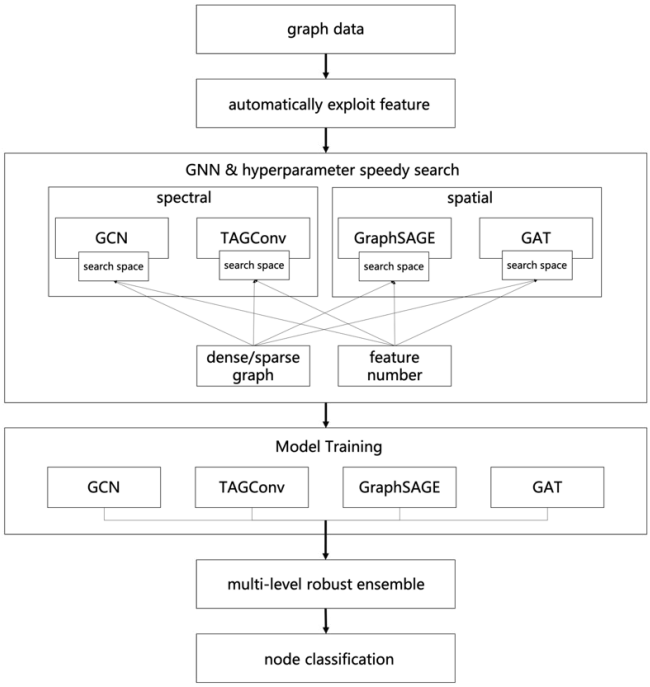



图 5　自动化图学习框架

## 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。

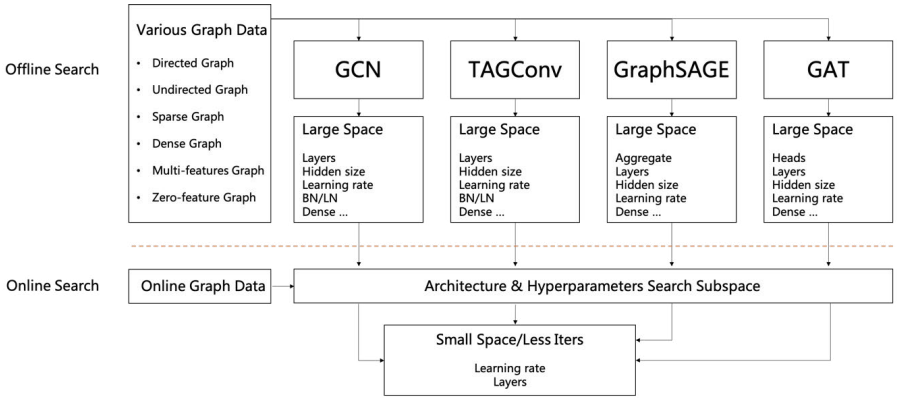



图 6　超参快速搜索

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。

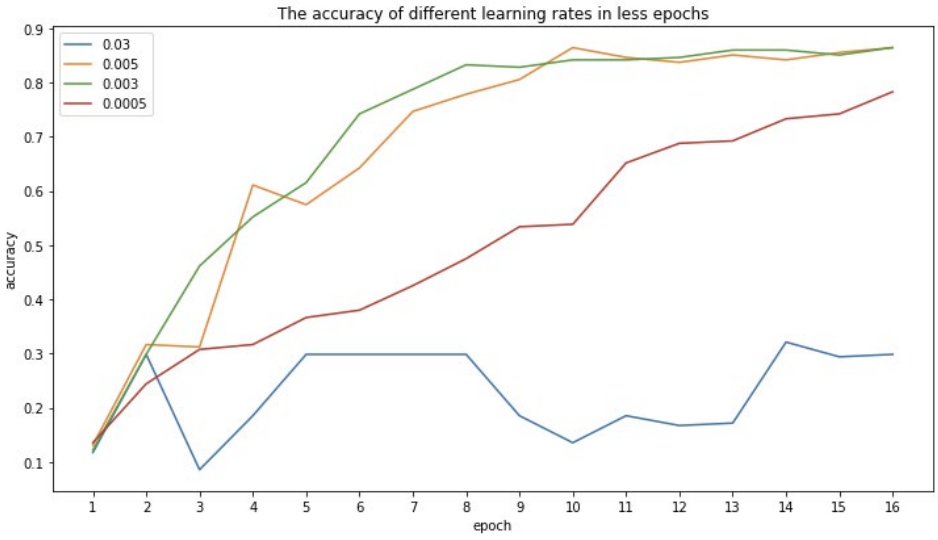



图 7　少量 epochs 模型训练下不同学习率的验证集精度

## 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。

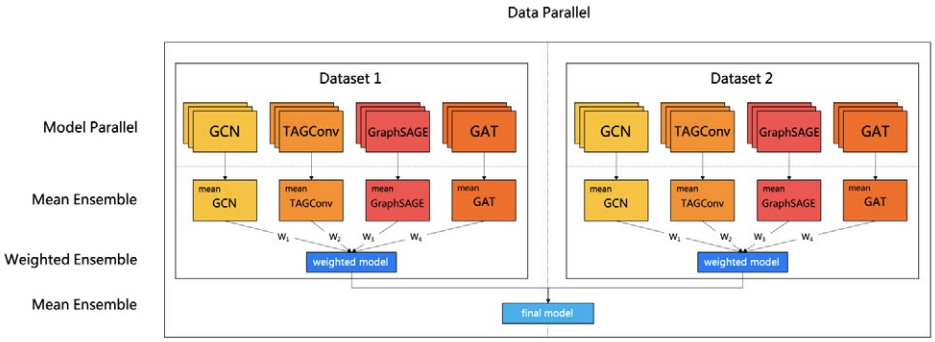



图 8　多级鲁棒模型融合

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。

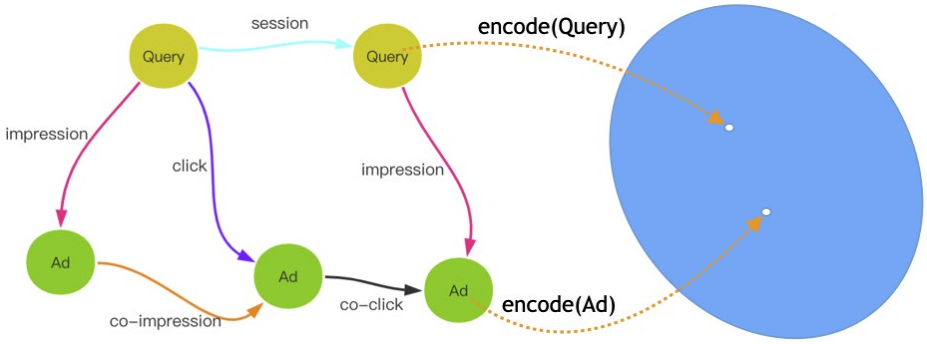



图 9　异构图的构建

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。

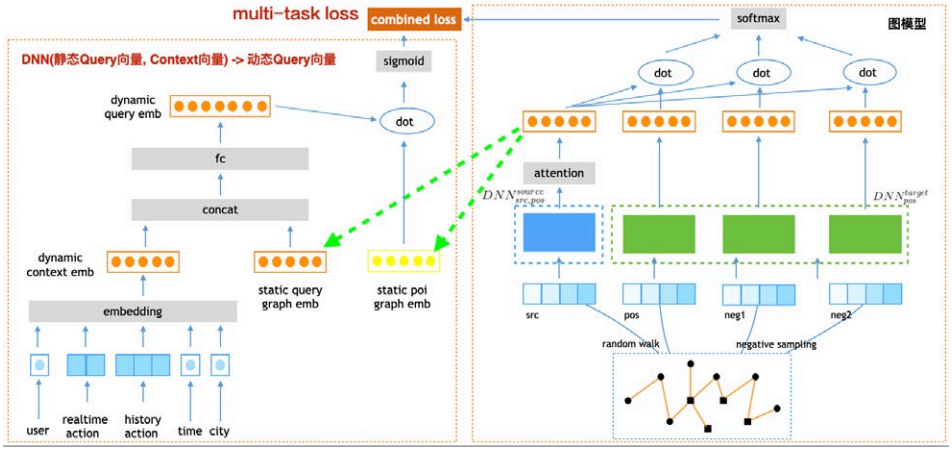



图 10　基于图表示学习的触发网络

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。

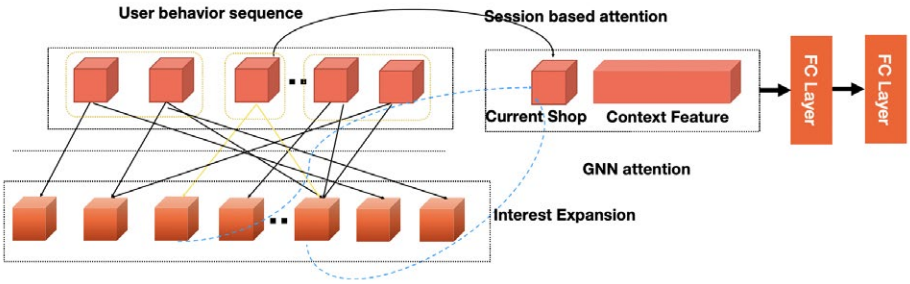



图 11　基于图神经网络的个性化预估网络

## 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

| 数据集 | | 数据量 | 包含的字段信息 |
|---|---|---|---|
| 训练集（Train） | | 3,000,000 | • 图片ID<br>• 图片的长宽<br>• 目标框的数量<br>• 目标框的2048维特征<br>• 目标框的标签（Label）<br>• 目标框的坐标信息和面积信息<br>• 用户查询Query<br>• 查询Query的ID |
| 验证集（Val） | | 14720 | |
| 测试集（Test） | test A | 28830 | |
| | test B | 29005 | |

数据示例


- 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
- 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 0.30830625][0.85375 0.0025 0.94875 0.1225 0.0114 ]]
- 用户查询Query：students printed schoolbag valid
- 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。

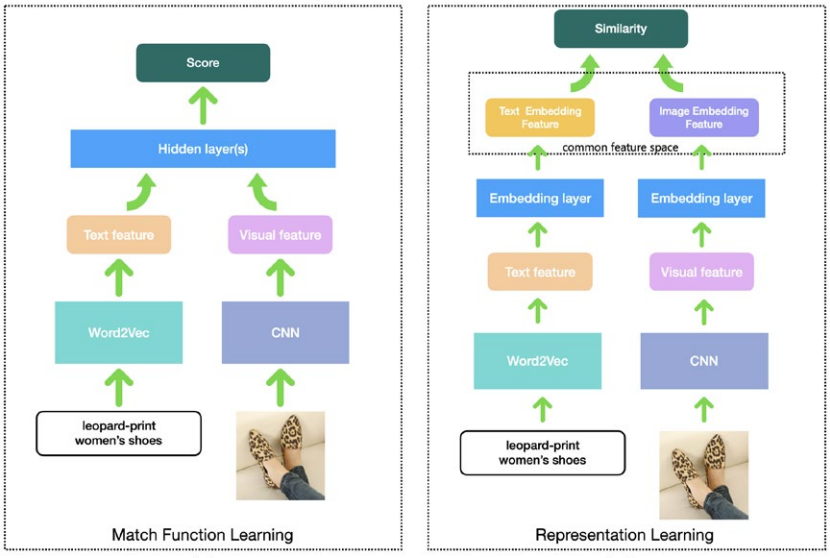



图 3　常用的多模态匹配解决思路

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：

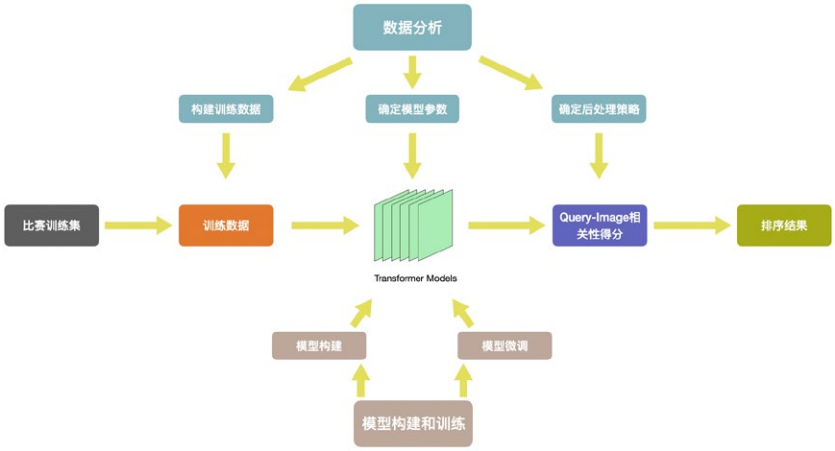



图 4　算法流程图

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。

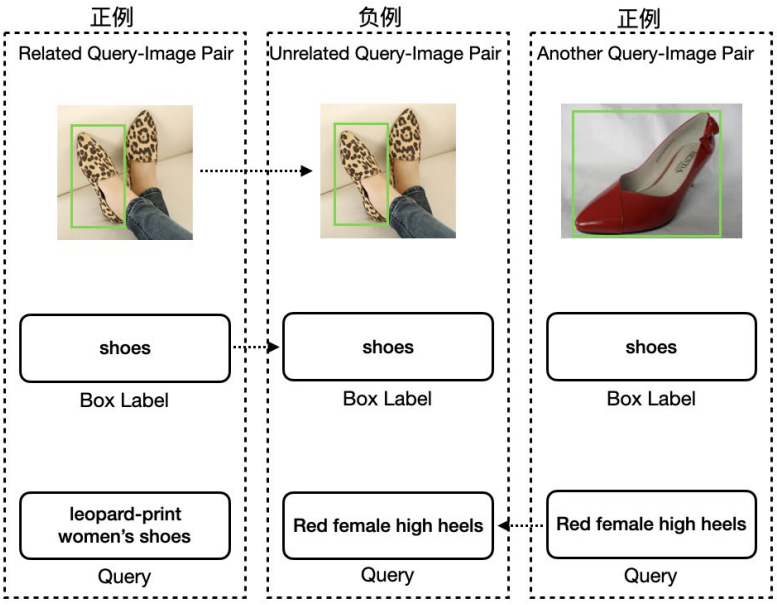



图 5　难例挖掘过程示意图

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：

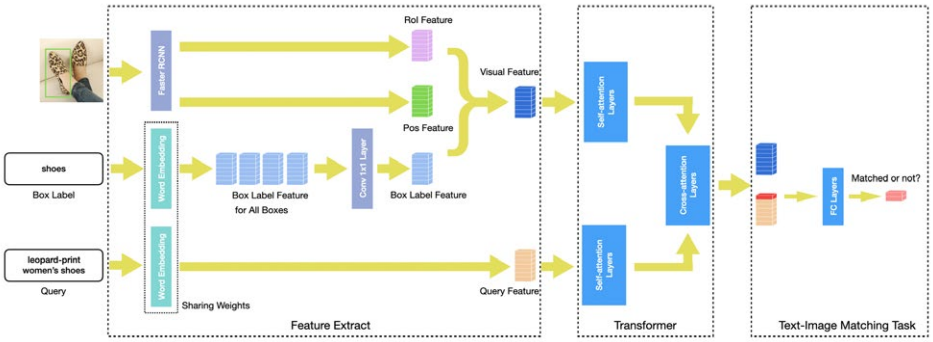



图 6　比赛中使用的 LXMERT 模型结构

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：

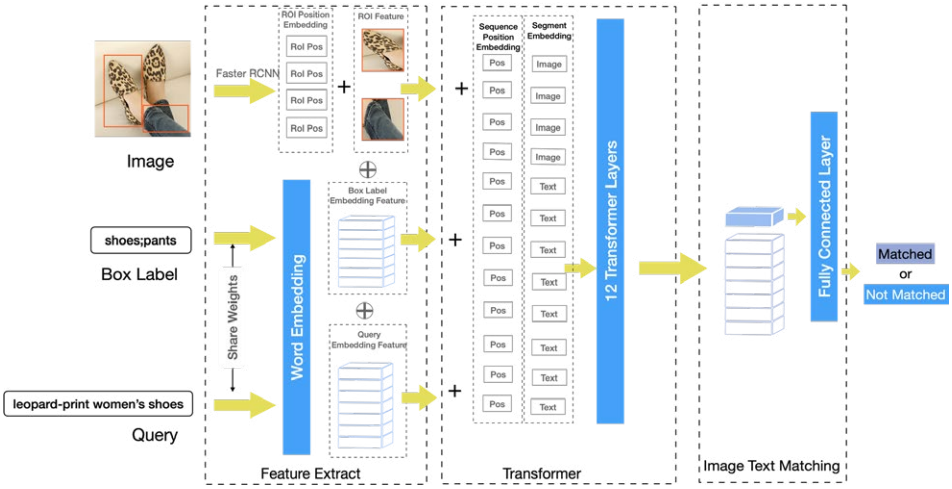



图 7　比赛中使用的 ImageBert 模型结构

**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。

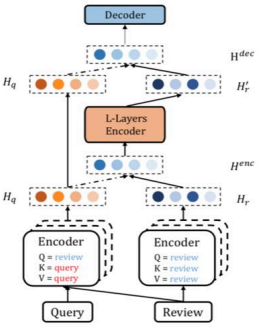



Figure 2: The Transformer-based query-aware tip generation framework. The left encoder encodes review-aware query while the right encoder maintains a self-attention manner to encode review. The two dotted arrows indicate the query information flows to the encoder and the decoder respectively.

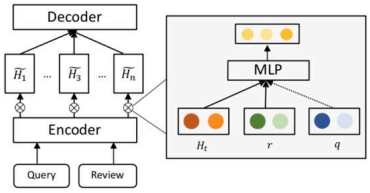



Figure 3: The RNN-based query-aware tip generation framework.

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》

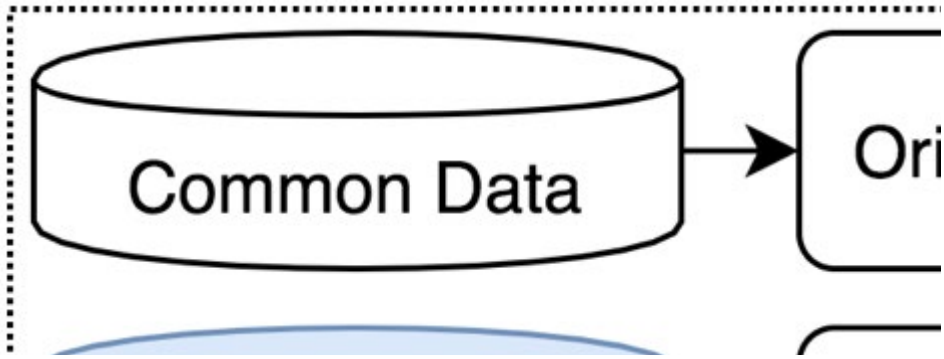
| 本论文系美团知识图谱组与中国科学院软件研究所唐弘胤同学、金蓓弘老师的合作论文。

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。

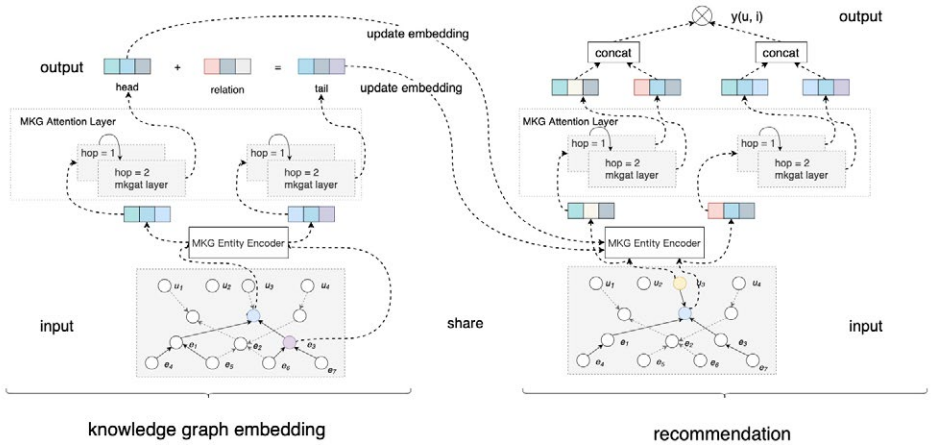
论文下载

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。

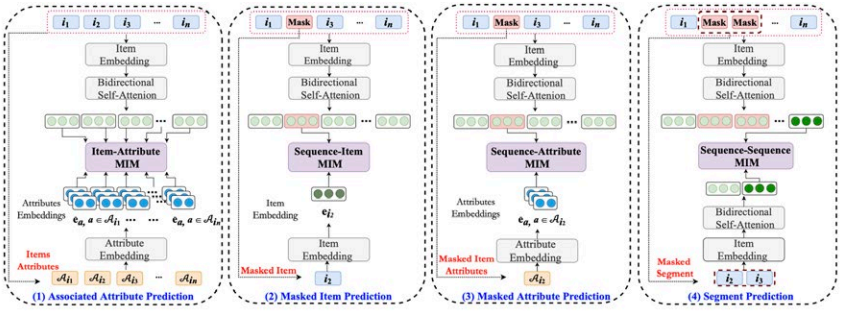



**Figure 1: The overview of $S^3$-Rec in the pre-training stage. We assume that the user sequence is $\{i_1, \cdots, i_n\}$ and each item $i$ is associated with several attributes $\mathcal{A}_i = \{a_1, \cdots, a_m\}$. We incorporate four self-supervised learning objectives: (1) Associated Attribute 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。

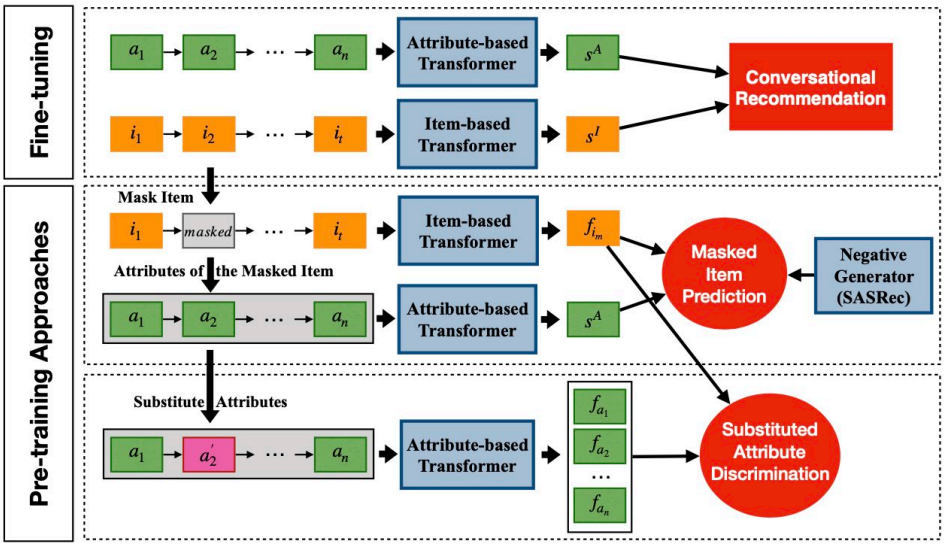
| Datasets | Metric | PopRec | FM | AutoInt | GRU4Rec | Caser | SASRec | BERT4Rec | HGN | GRU4Rec$_F$ | SASRec$_F$ | FDSA | $S^3$-Rec | Improv. |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Meituan | HR@1 | 0.0946 | 0.1084 | 0.0804 | 0.1194 | 0.1368 | 0.1797 | 0.1381 | 0.1603 | 0.1436 | 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：

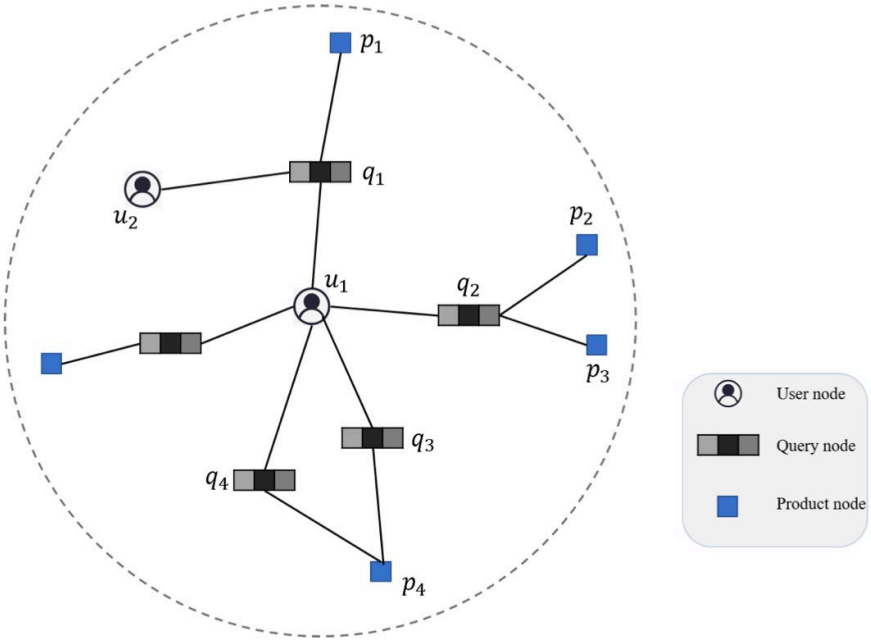



**Figure 1: Example of graph construction.**

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。

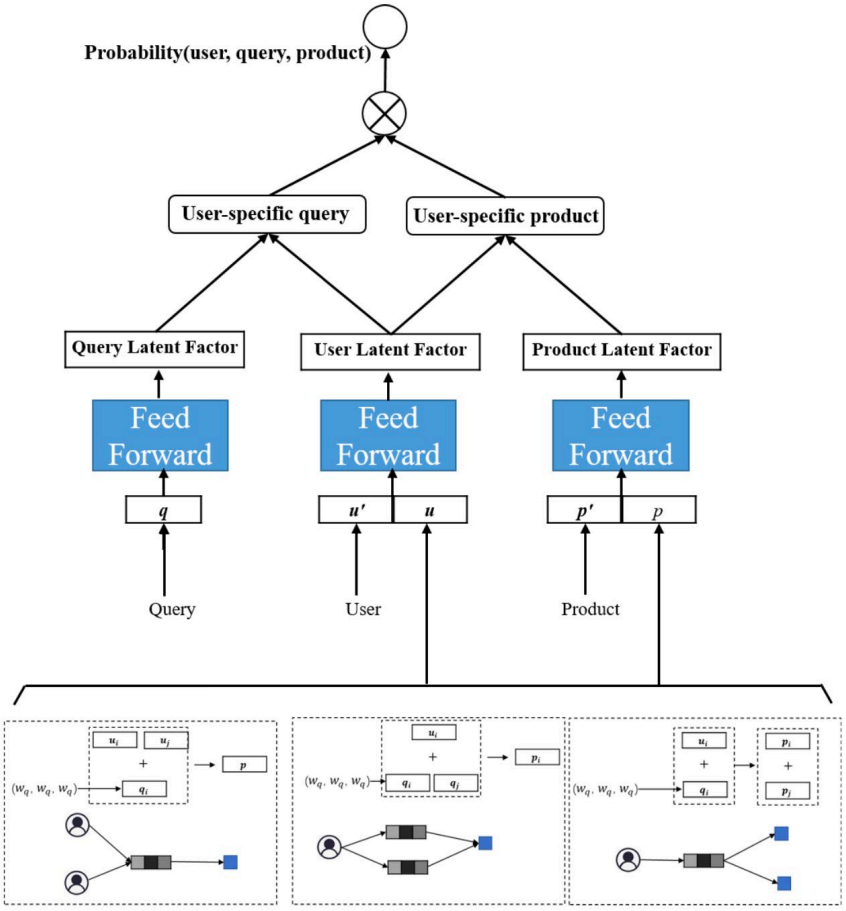



**Figure 2: The framework of GraphLSR. The bottom half and top half part illustrate the graph embedding and personalized search module, respectively, which are connected by the transmitting of nodes embedding representation**

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。


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。

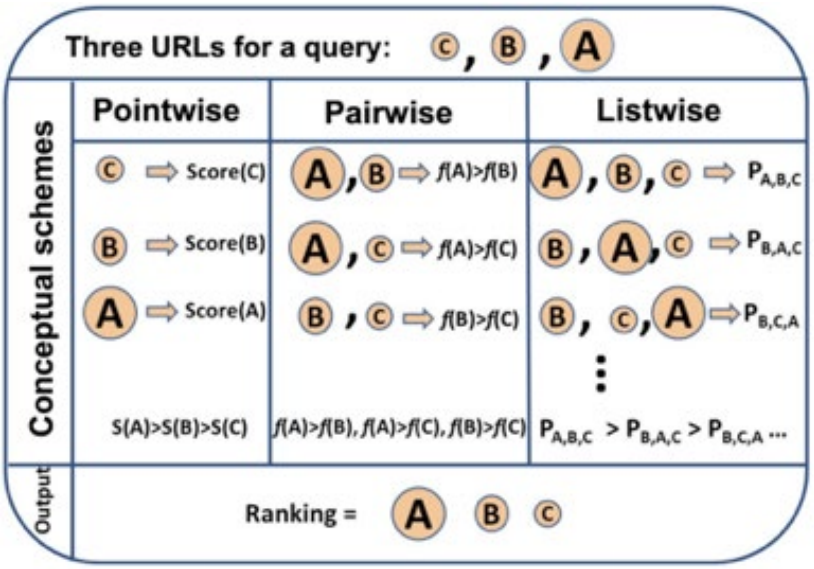



图 2　Pointwise、Pairwise、Listwise 训练的目标

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）。

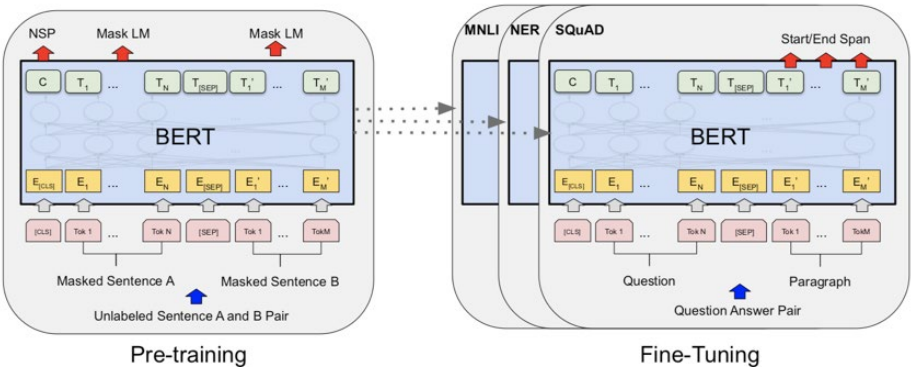



图 3　BERT 的结构和训练模式

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。

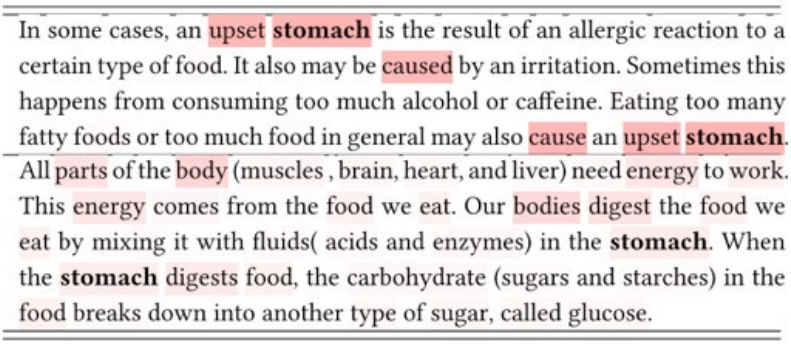
## 模型介绍

### 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

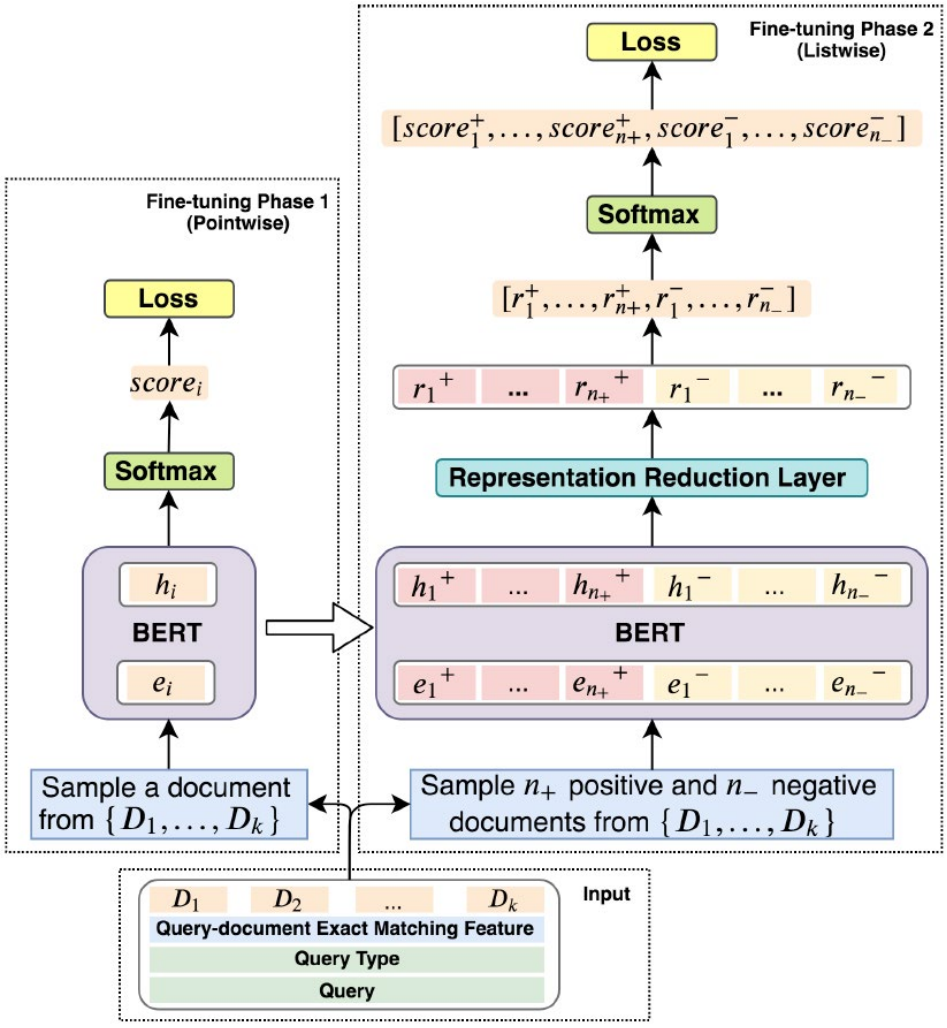



图 5　模型结构

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。

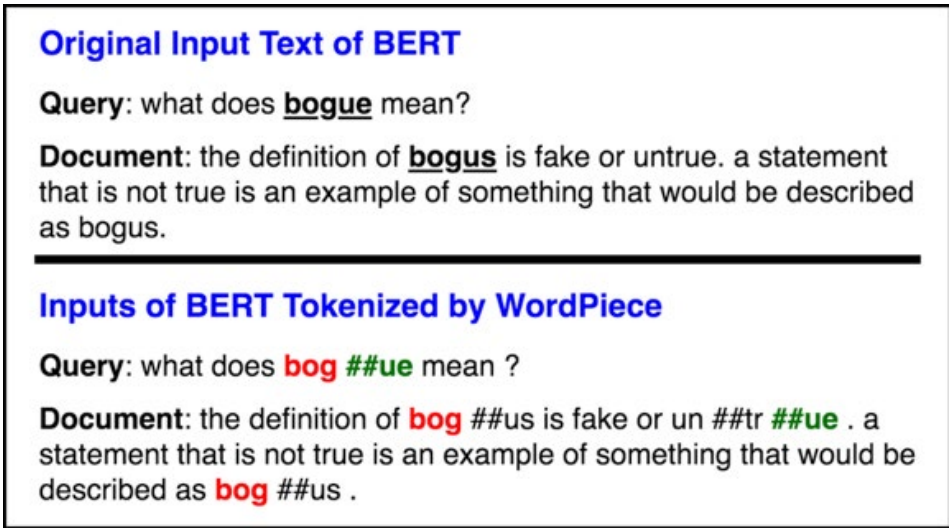



图 6　BERT WordPiece 处理前 / 后的文本

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，$\alpha$表示该特征的重要性，作为一个超参数使用。

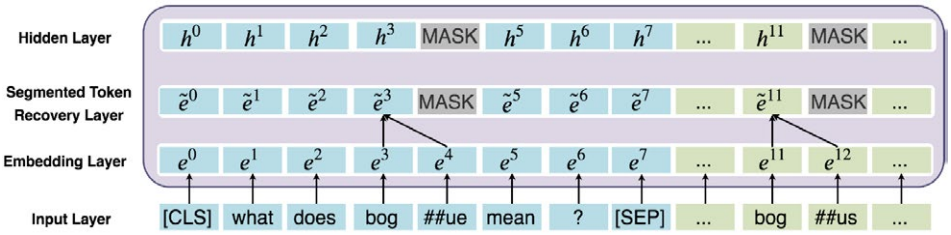



图 7　词还原机制的工作原理

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 Word-Piece 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。

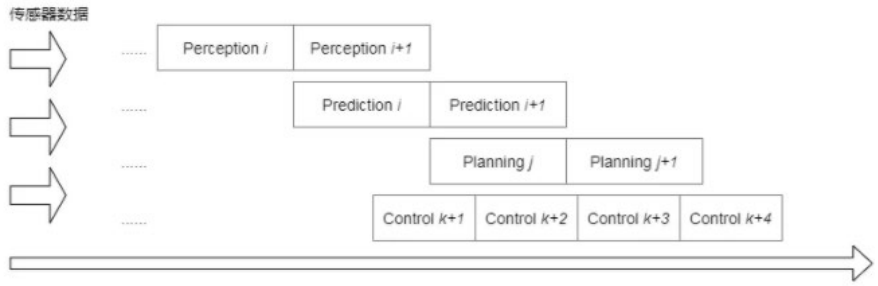



图 2　无人车系统理想时序

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。

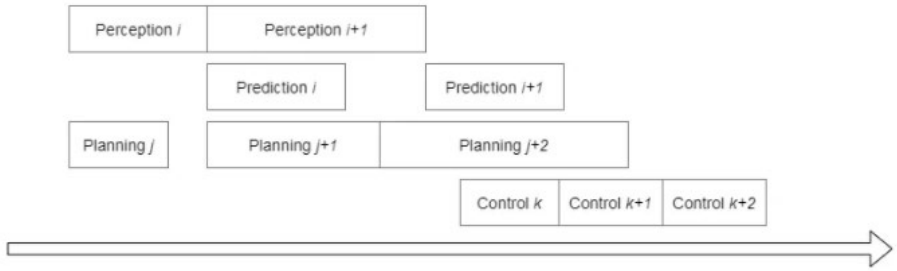



图 3　无人车系统实际时序

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。

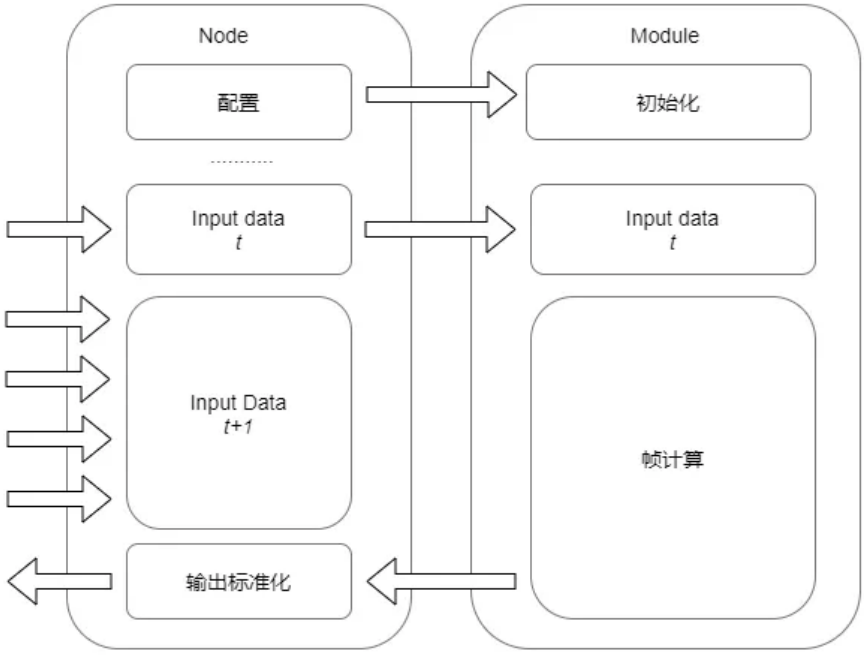



图 5　功能模块的标准模型

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。

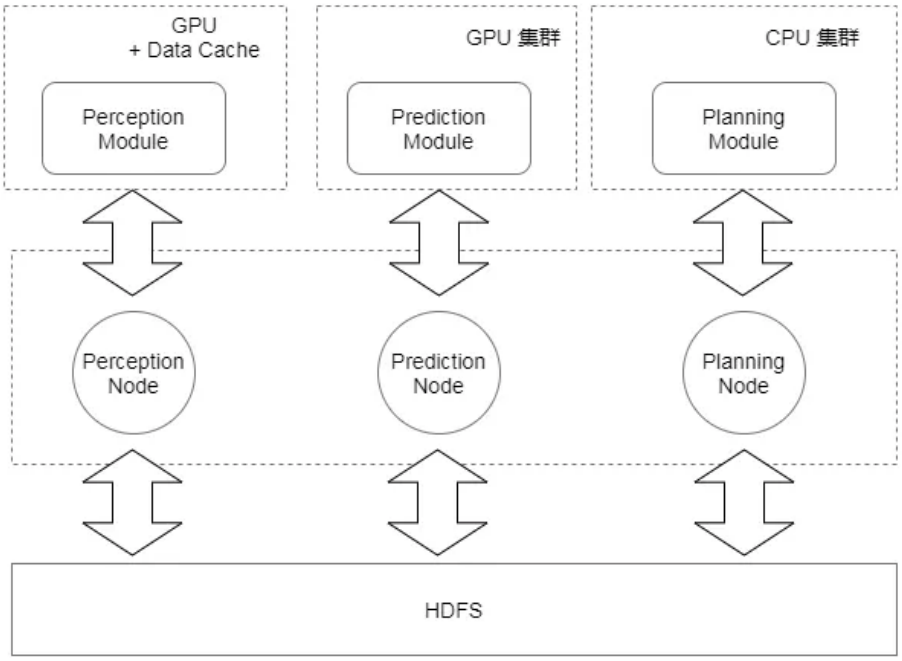
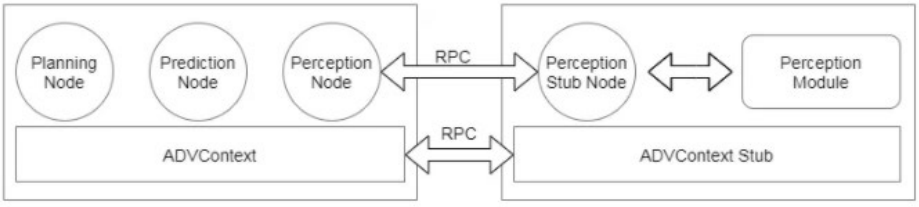



图 6　分布式部署

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。

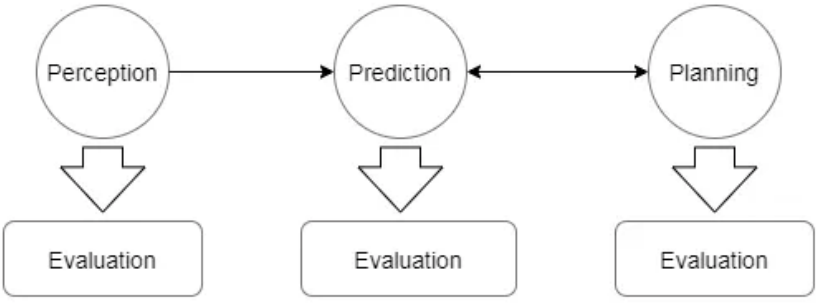



图 8　分模块 Evaluation

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。

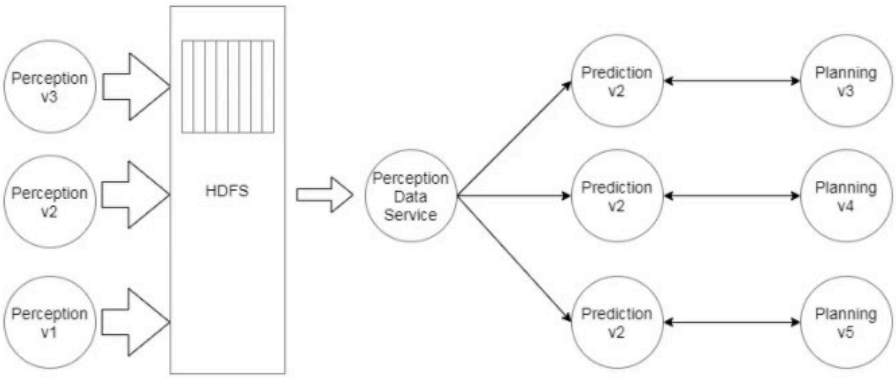



图 9　数据复用

### 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。

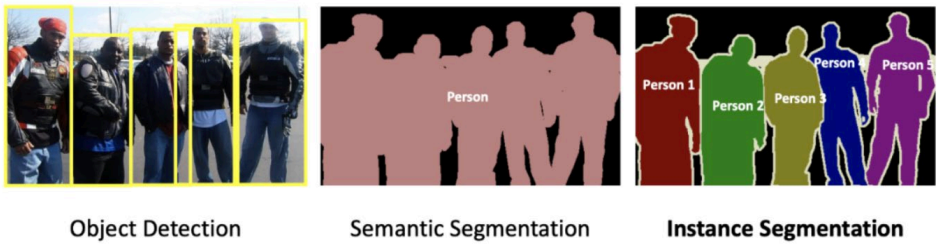



图 1　目标检测，语义分割和实例分割的区别

### 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

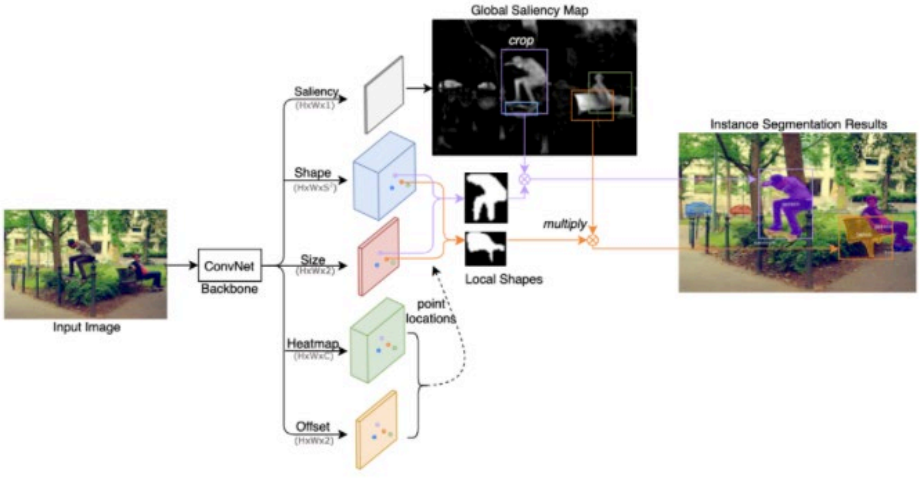



图 2　CenterMask 网络结构图

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。

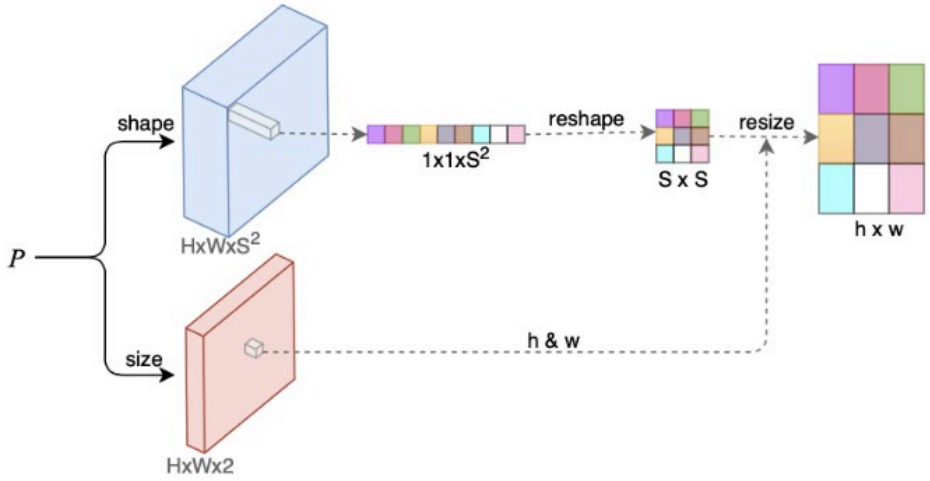


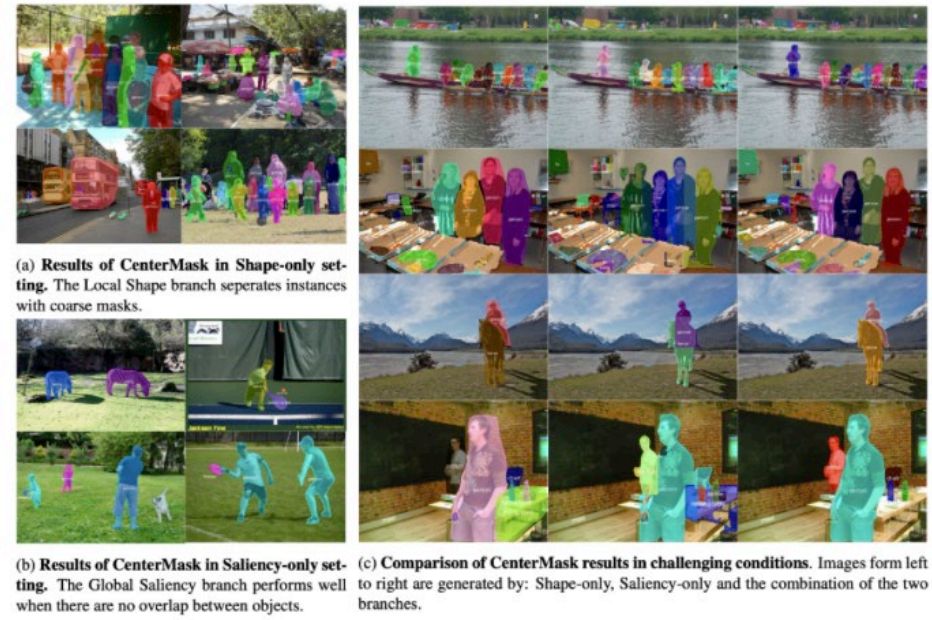

图 3　Local Shape 预测分支

## 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。

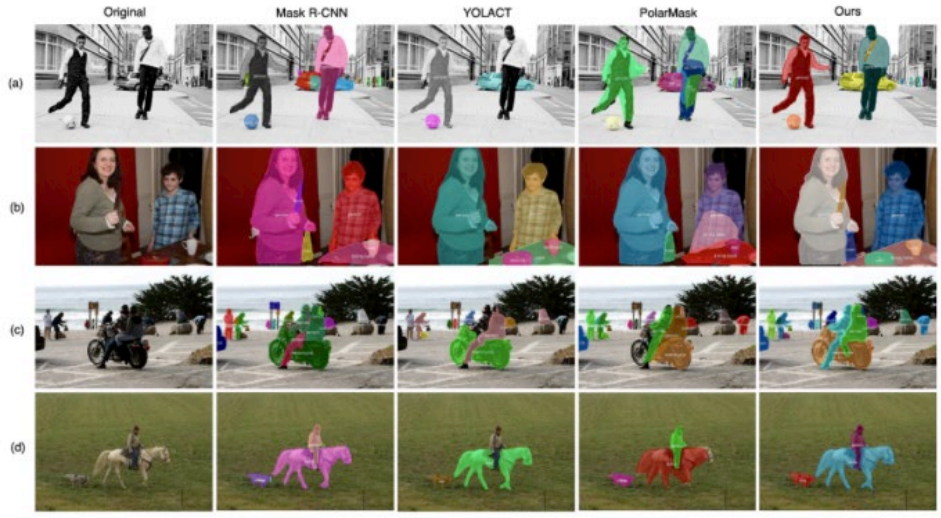



图 6　CenterMask 与其他方法在 COCO 数据集上的可视化对比

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

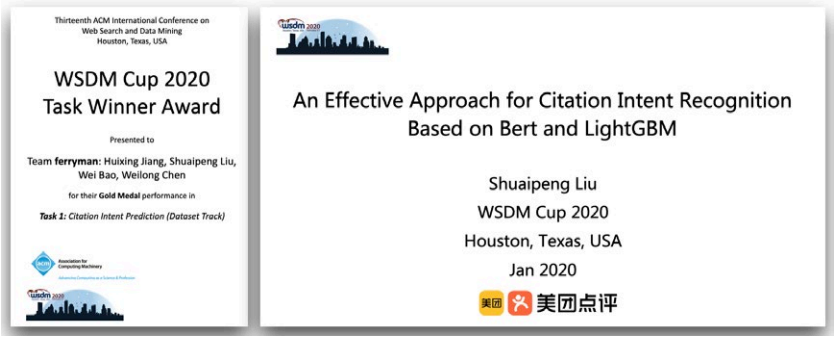
作者：帅朋

## 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

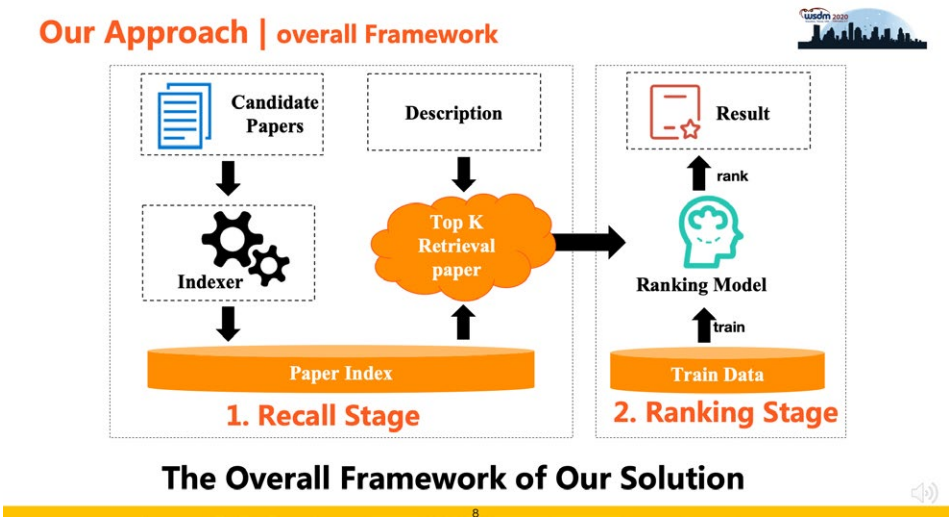
评测使用的评价指标为 Mean Average Precision @3 (MAP@3), 形式如下：

$$MAP@3 = \frac{1}{|U|}\sum_{u=1}^{|U|}\sum_{k=1}^{min(3,n))} P(k))$$

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。

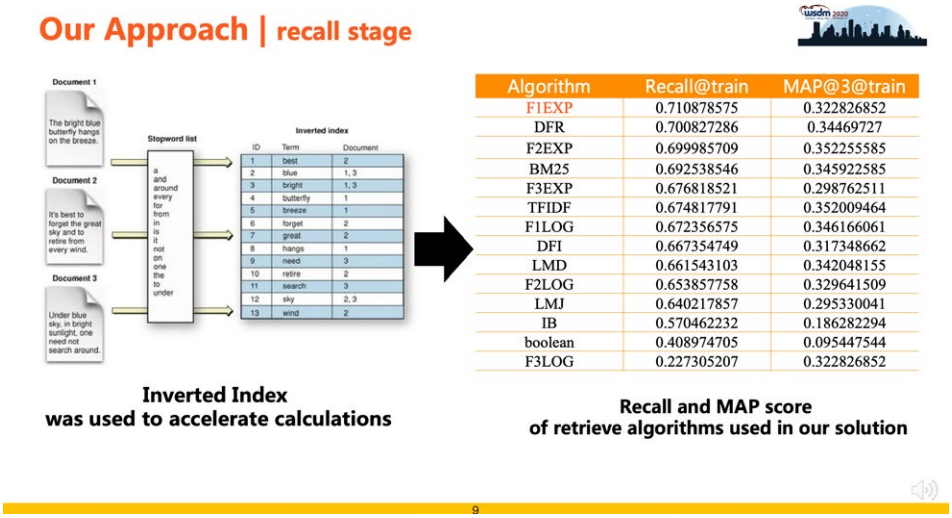
**检索召回方案**：比赛过程中对比实验了两种召回方案，基于“文本语义向量表征“和“基于空间向量模型 + Bag-of-Ngram”。由于本任务文本普遍较长且专有名词较多等数据特点，实验表明“基于空间向量模型 + Bag-of-Ngram”的召回方案效果更好，下表中列出了使用的相关模型及其实验结果（recall@200）。可以看到相比于传统的 BM25 和 TFIDF 等算法，F1EXP、F2EXP 等公理检索模型（Axiomatic Retrieval Models）可以取得更高的召回覆盖率，该类模型增加了一些公理约束条件，例如基本术语频率约束，术语区分约束和文档长度归一化约束等等。

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

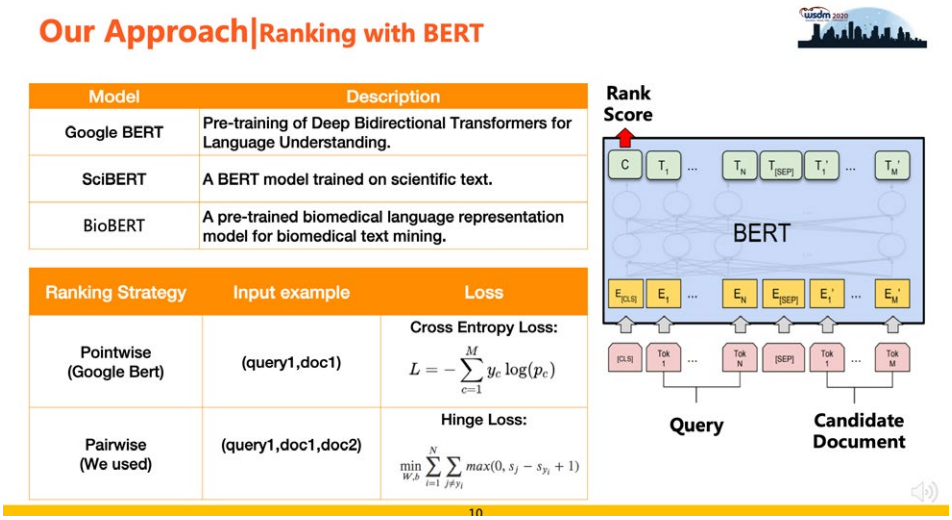
精排阶段基于 Learning to Rank 的思想进行方案设计，提出了两种解决方案，一种是基于 Pairwise-BERT 的方案，另一种是基于 LightGBM 的方案，下面分别进行介绍：

### 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：

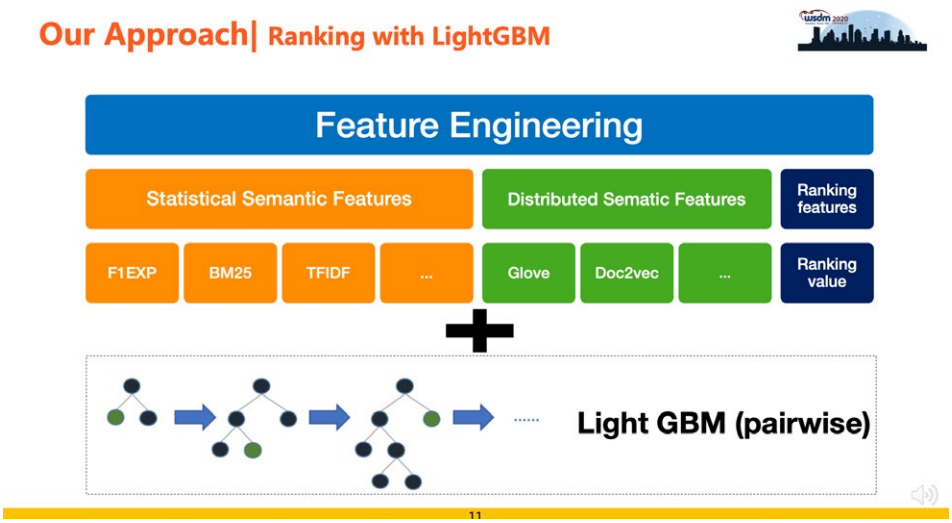
采用 Gradient-based One-Side Sampling(GOSS) 技术去掉很大部分梯度很小的数据，只使用剩下的去估计信息增益，避免低梯度长尾部分的影响；

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。

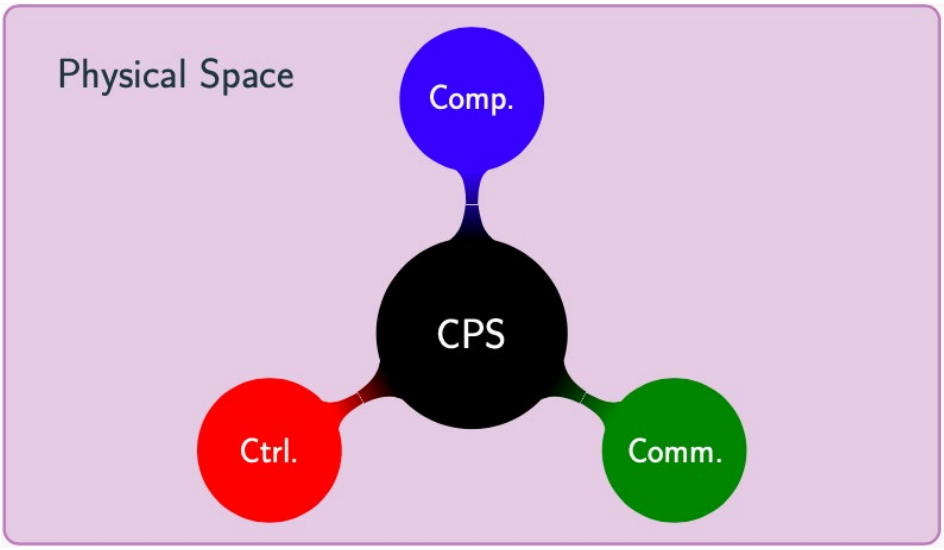
2020 年 9 月 10 日，Top Talk 邀请到了清华大学自动化系莫一林副教授，请他带来题为《信息物理系统中的安全控制算法设计》的分享。本文系本次分享内容的文字实录，希望能对大家有所帮助或者启发。

**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。

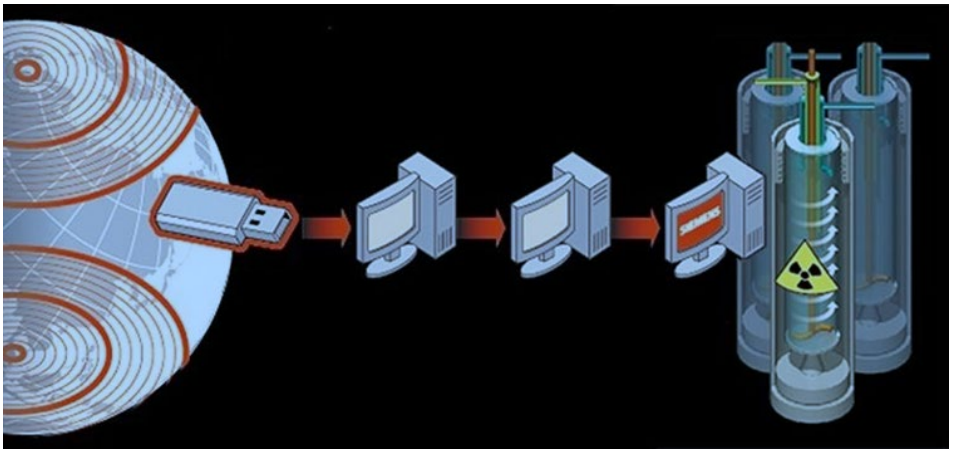

上图是我们之前说的智能家居的一个示意图。智能家居希望人们可以通过比如手机 App 去控制家里的电器，但是一旦家里的电器都上网之后，智能家居控制系统的安全就会变得非常重要，如果当一个攻击者可以操纵成千上万个家庭电器的话，这就可能带来非常严重的问题。

## 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

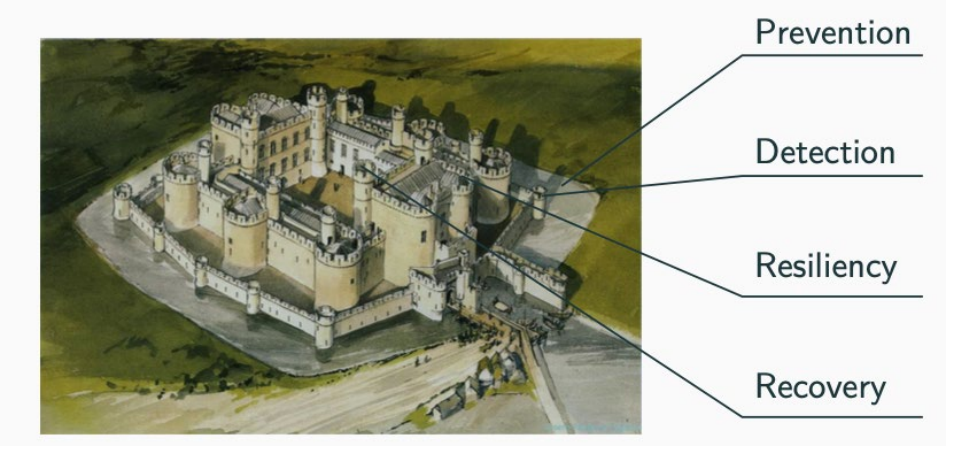
较著名的事件，有两个黑客（这两个人应该只是研究者，并不是真的想要搞破坏）通过攻击克莱斯勒吉普的一款车，入侵了这辆车的显示信息，就是所谓的 implementation，还包括一些娱乐系统，然后通过这个系统，进入车内部的网络来控制这个车，比如说方向、控制刹车，甚至包括安全气囊等等。

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

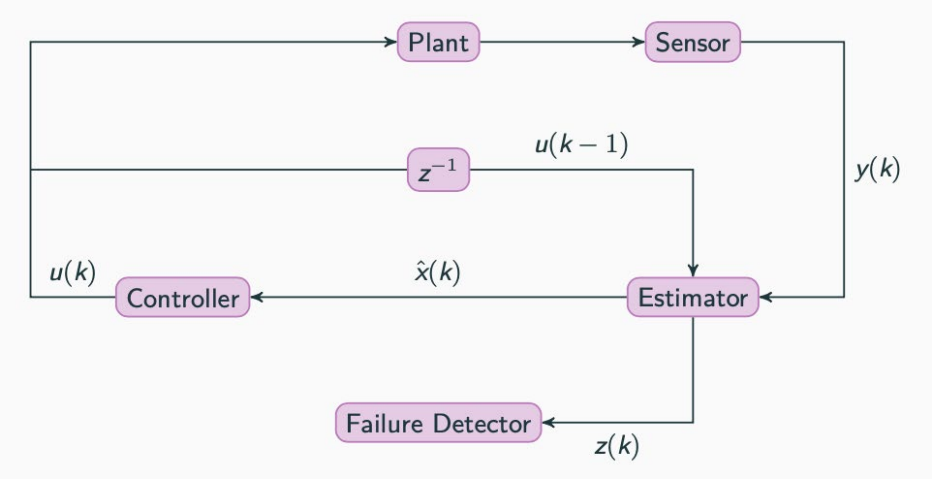
控制，到底能给我们带来什么？以离线设计系统为例，首先我们可以通过控制找到这个系统关键的部件，然后对这些部件做额外的冗余设计。在这个系统没有上线之前，我们可以对系统做一些可控、可观的分析，进而提升系统本身的韧性；上线之后，我们可以利用传统的方法如故障诊断，当然这里就变成了入侵诊断和入侵定位的问题。此外，我们还可以做一些鲁棒控制，保证系统的控制器在有攻击情况下依然能够容错。最后还有一个问题，一个大的系统想要让它更安全，我们到底应该先加固哪个地方？这都是控制可以提供给我们的一些东西。

### 检测

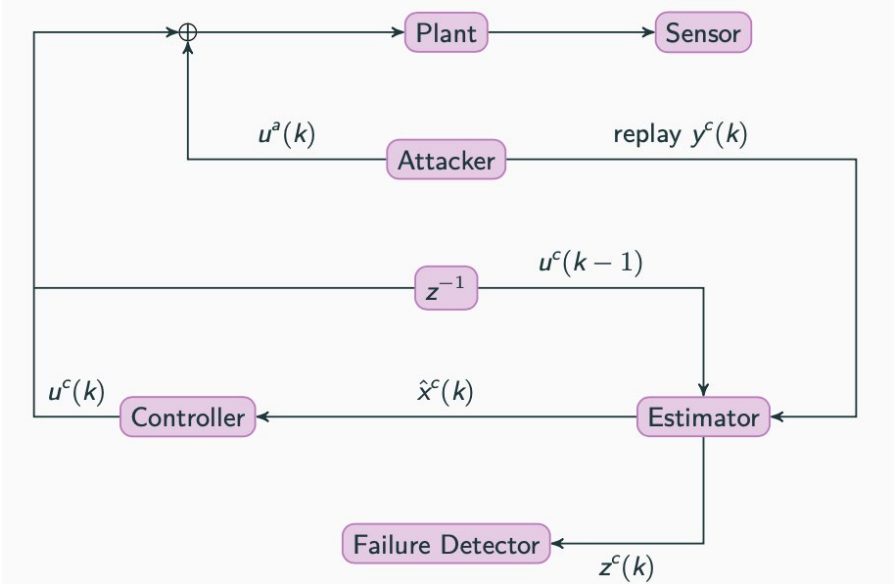
接下来，我主要讲一下关于检测的问题，我们也是受到“震网病毒”例子的启发之后开始做了相关的研究。“震网病毒”是 2010 年被发现的，从计算机的角度来说，它是一个很复杂的、很难防御的病毒。但是从控制角度来说，它的策略其实相当简单。离心机本身是一个类似于快速旋转东西，如果想要毁坏离心机，那就需要让它转的比原本设定的转速快很多，但是如果只是让它转的特别快的话，由于整个系统有传感器，传感器就会发现离心机转速太快，从而触发报警，报警之后就会有技术人员前来检查，整个过程并不能对这个系统造成很大破坏。

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

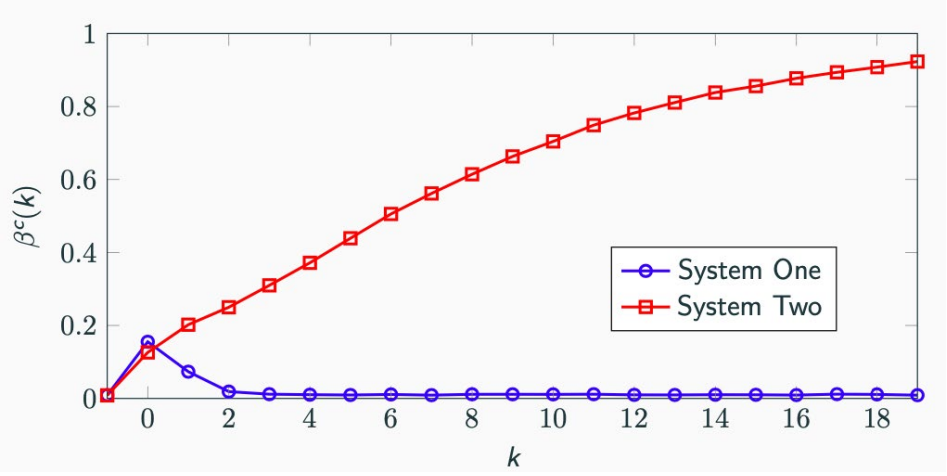
说把整个系统的转速调到比如 2000 转的时候，会使用之前正常的数据去替换异常数据，那么系统操作员看到依然是正常的转速，就不会察觉这个系统已经出现问题了。比如在一些警匪片中，匪徒会去把他们想要抢劫的地方的监控视频的影像做一个重放，比如用前一天没有异常的影像去覆盖掉抢劫的影像，跟“震网病毒”的策略很类似，在信息安全领域也是比较常见的“重放”攻击。



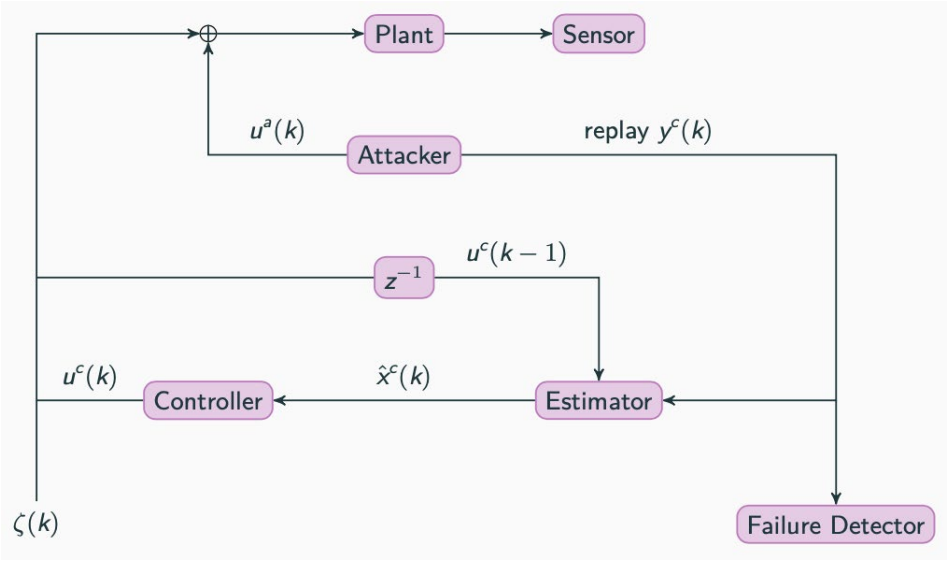

大家可以简单地看一下这个系统的框图，比如说我们做一个离心机或者一个其他的物理系统，需要做控制。那么我们使用传感器去监测这个系统，传感器的输出会给一个估计器，对于无人驾驶来说，可能这个应该叫感知，而不能简单地叫估计，因为它的功能可能会更复杂。但不管怎样的系统，我们都需要通过一个这样的东西，对收集到的信息做一个处理，然后得到系统的状态，再根据状态来设计系统的控制，最后反馈给物理系统。估计器可能还会输出一些信息传递到故障检测器里面，然后故障检测器会检测收到的信息 y(k) 是不是有问题，大概就是这样的一个系统。




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。



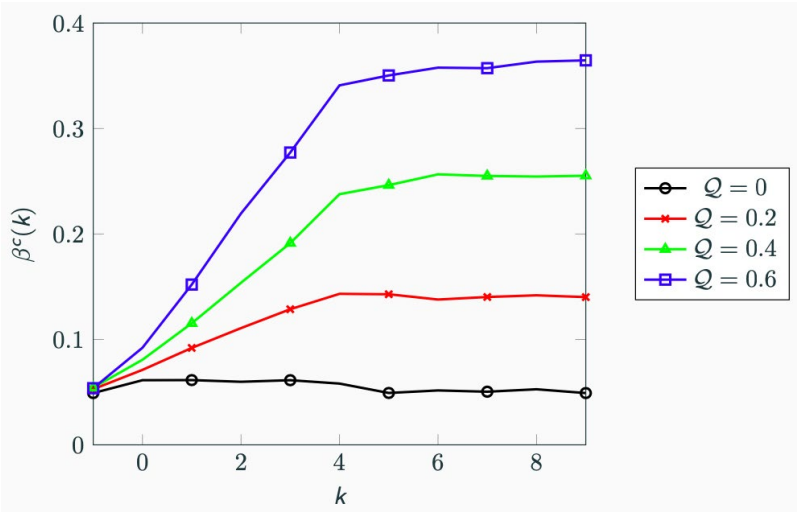
由于系统本身是有故障检测器的，所以最原始的系统在设计的时候根本没有考虑安全性，没有考虑是否能检测出来所谓的“重放”攻击。事实上，“重放”攻击并不是总是有效的，我们发现有一些系统可以检测到“重放”攻击，而有一些系统是不能检测的。如上图所示，Y 轴表示检测器报警的概率，报警概率最大值是 1。整个系统的重放是在时刻零开始，那么作为第一个系统，也就是蓝线标记的系统，我们可以看到在重放开始的时候，它有一个比较短暂的损害过程，就是说这个时候报警的概率有一些，但是也不是特别高，然后很快报警概率就缩减到一个接近于 0 的值；而第二个系统，也就是红线标记的系统，我们发现它的报警概率随着“重放”变得越来越高，最后会趋向于 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

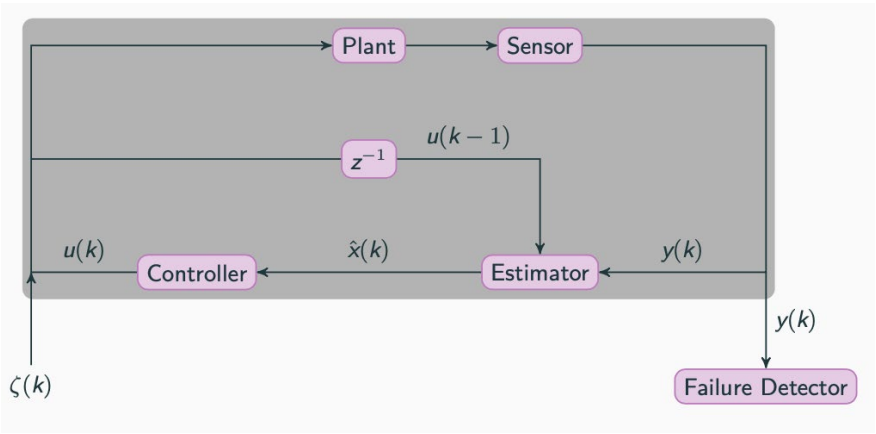
们任何信息，因为系统控制的很好。

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。

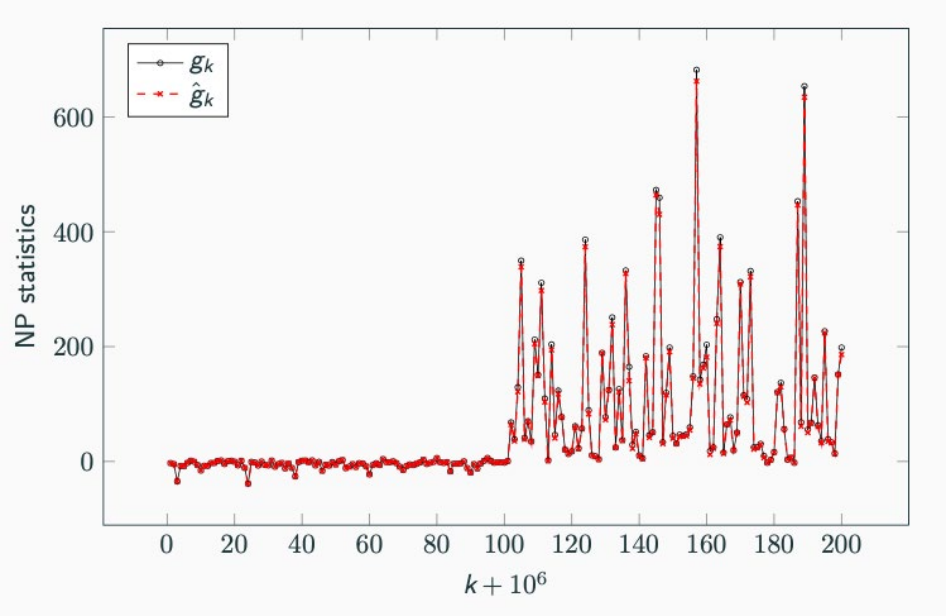



上图是我们做的一些实验的结果。针对的是一个非常简单的系统，大家可以认为这个是噪声的能力，我们发现随着加的能量越高，检测的概率会变得越高。我们大概的处理思路就是这样的，后面会有一些技术的问题，比如说加个扰动信号进去之后，系统

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。

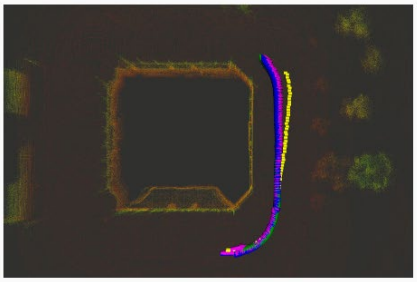


**Figure 7:** EKF

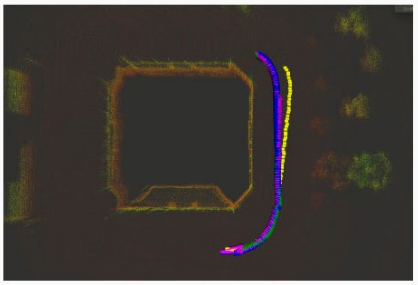


**Figure 8:** Secure EKF

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。

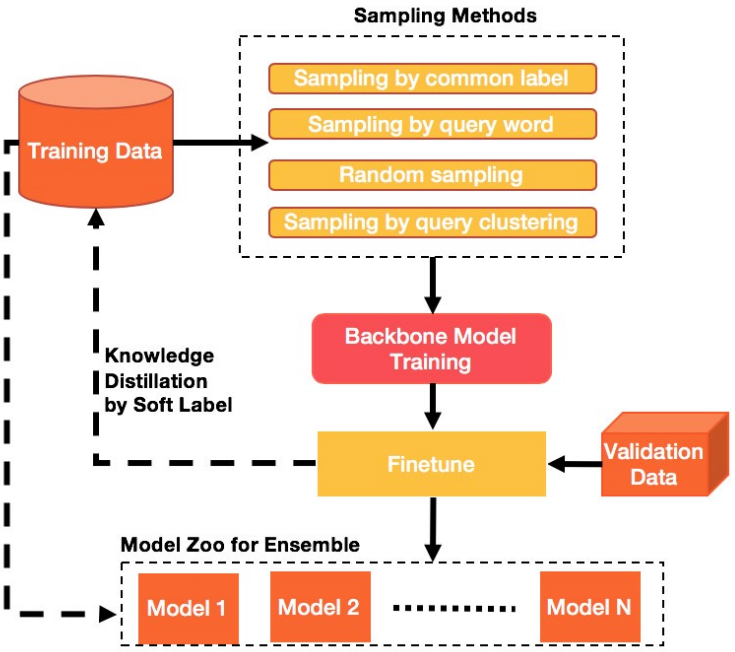



图 4 基于多样化负采样的多阶段蒸馏学习框架

## 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。

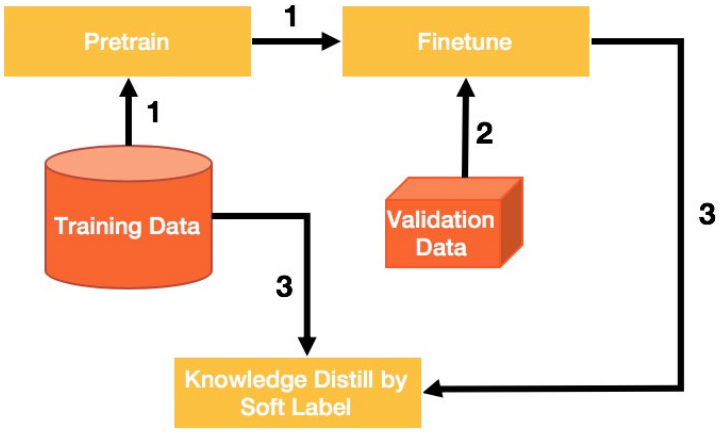



图 5　多阶段蒸馏学习

## 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。

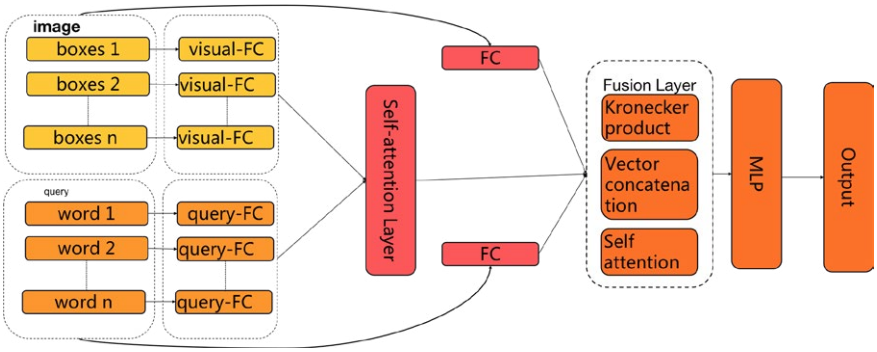



图 6　细粒度匹配网络

## 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。

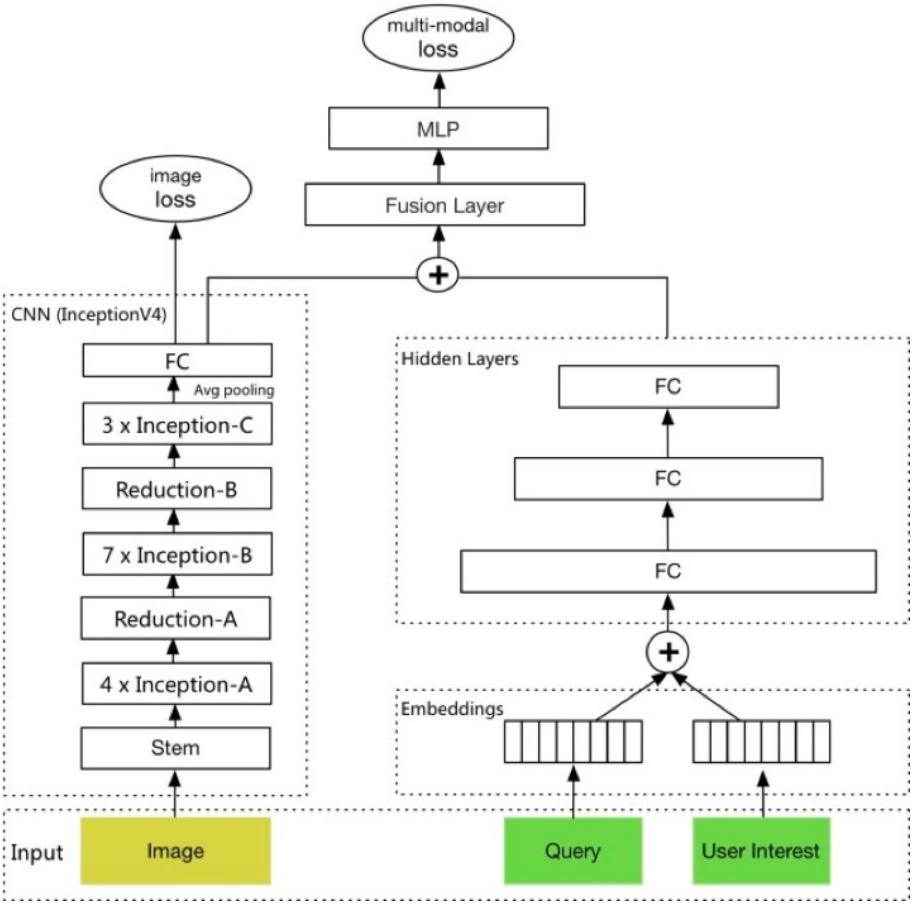



图 7　广告创意业务中的多模态学习

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。

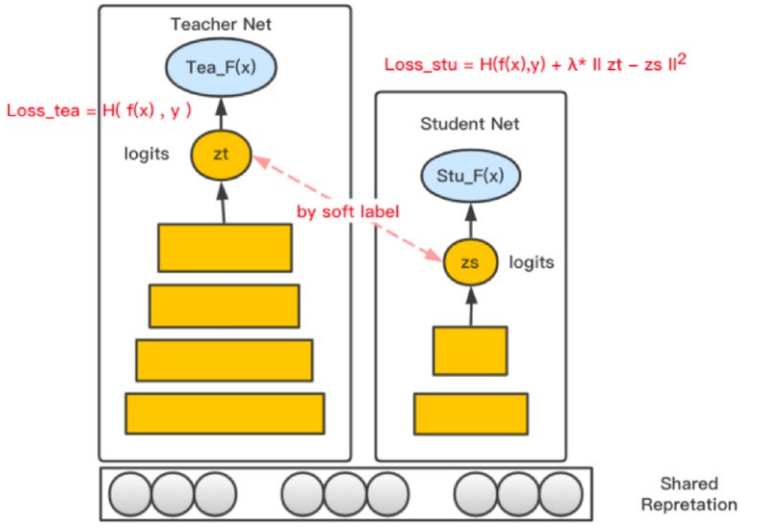



图 8　广告创意业务中的蒸馏学习

## 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。

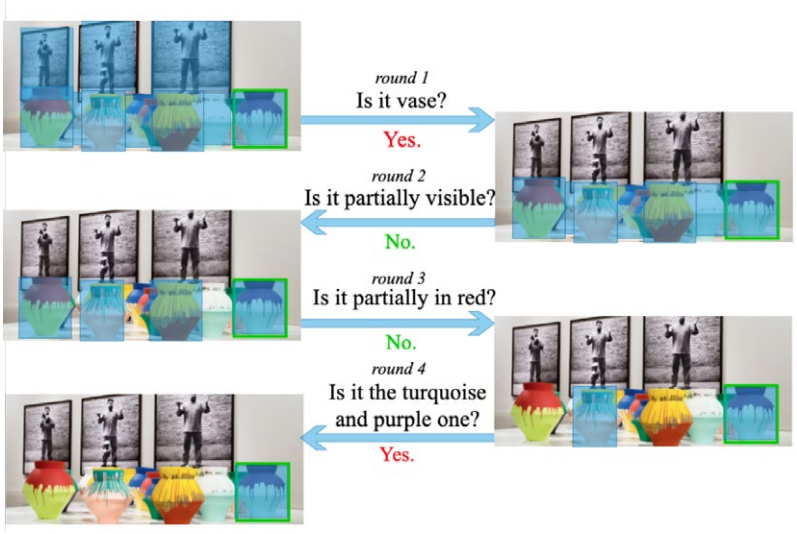



图 1　目标导向的视觉对话

## 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。

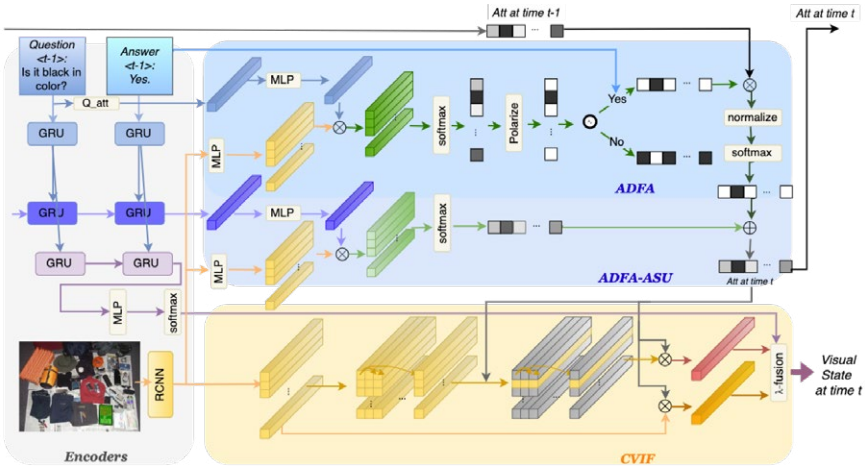



图 2　响应驱动的视觉状态估计器框架图

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$\alpha_t^q$。
- 第二步，对$\alpha_t^q$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$\alpha_t^q$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$\lambda_t$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。

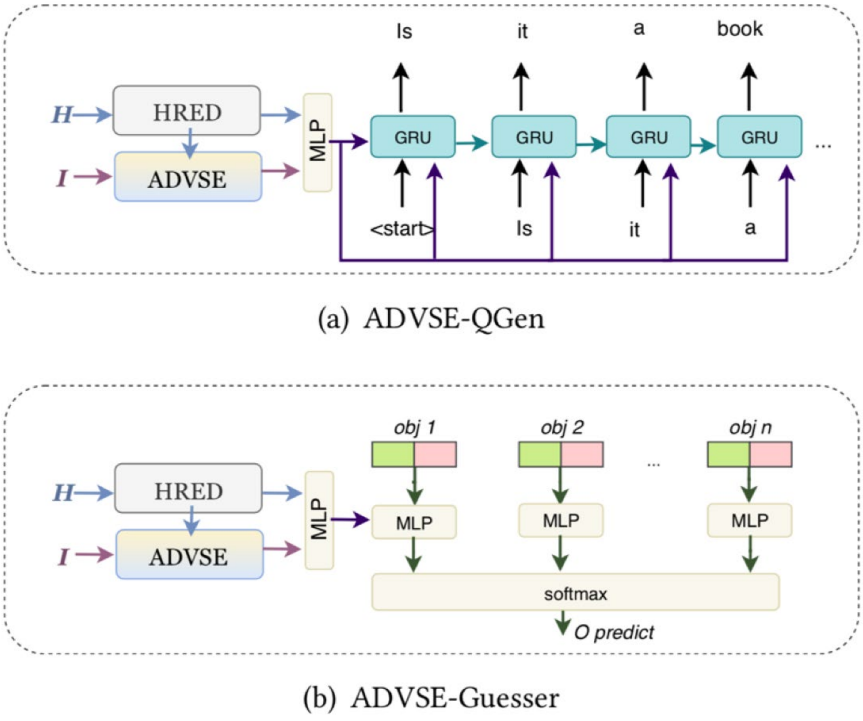



图 3　响应驱动的视觉状态估计器用于问题生成和回答示意图

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

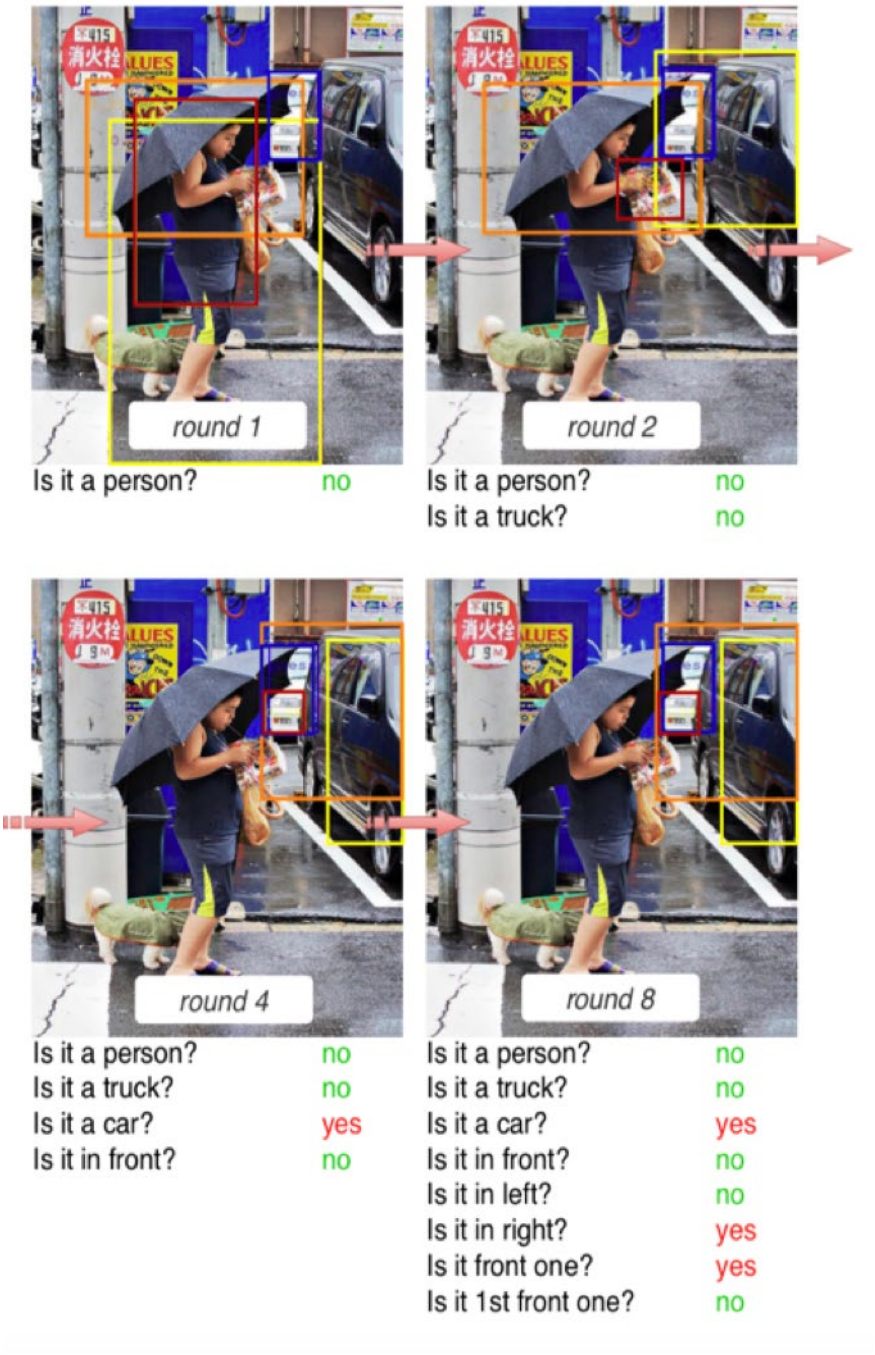



图 4　问题生成过程中响应驱动的注意力转移样例分析


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。

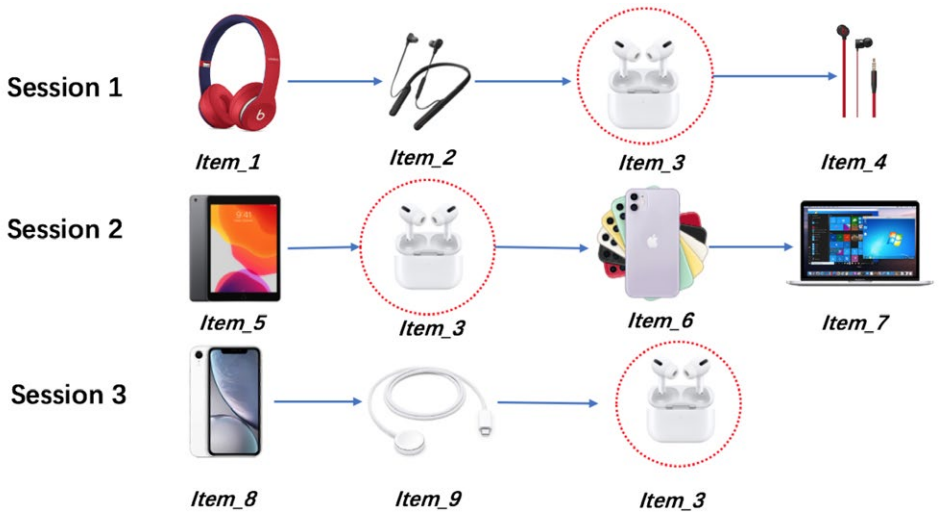



图 1　跨会话 item 的 Toy 样例

- 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。

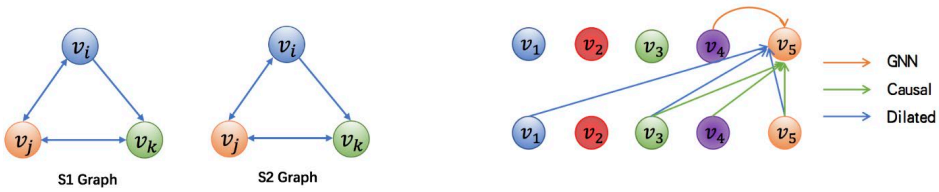



图 2　图结构对于序列数据的建模示意图

## 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。

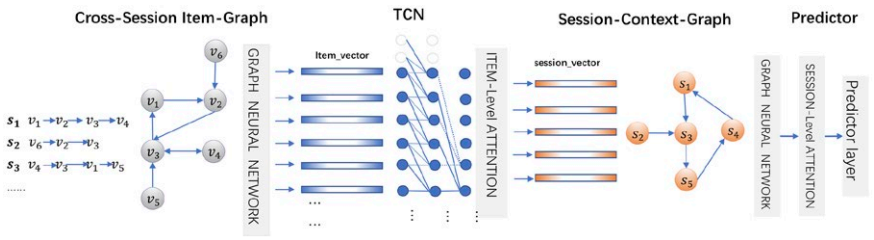



图 3　CA-TCN 模型总体框架图

## 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。

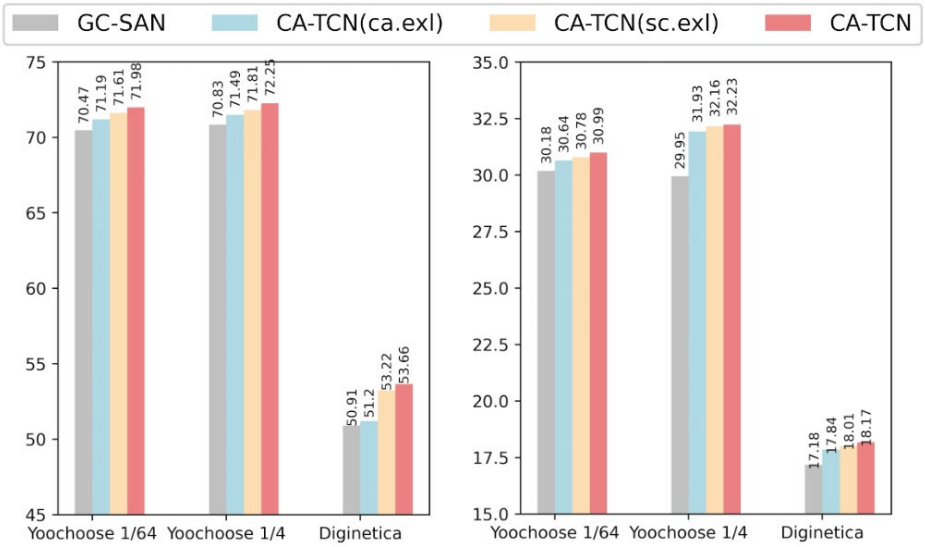



图 4　CA-TCN 模型不同组成部分的实验结果

## 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

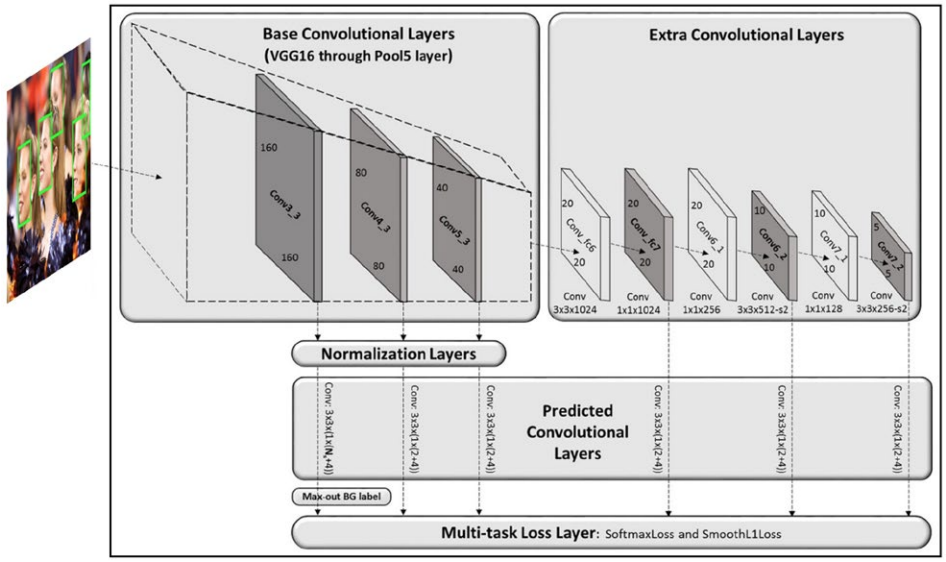
人脸检测模型结构设计主要包括检测框架、主干网络、预测模块、Anchor 设置与正负样本划分等四个部分，是单阶段人脸检测方法优化的核心。

- 检测框架

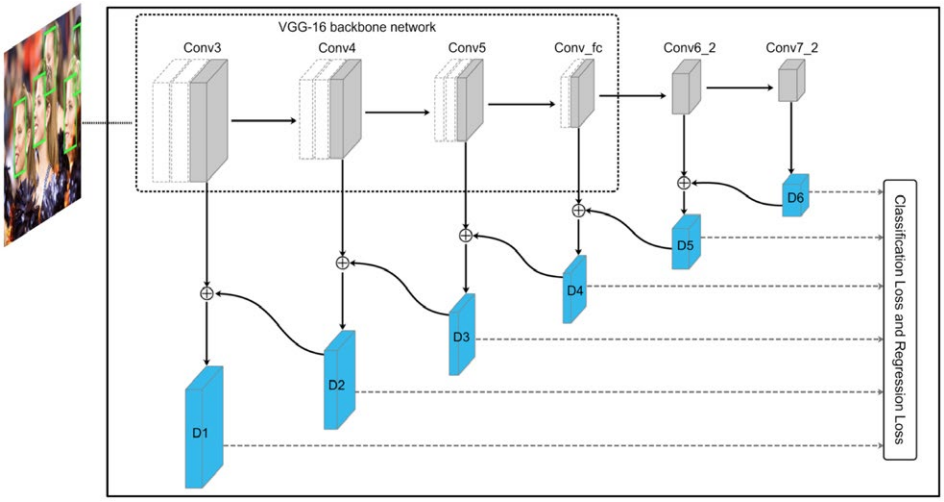
近年来单阶段人脸检测框架取得了重要的发展，代表性的结构有 S3FD[17] 中使用的

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。

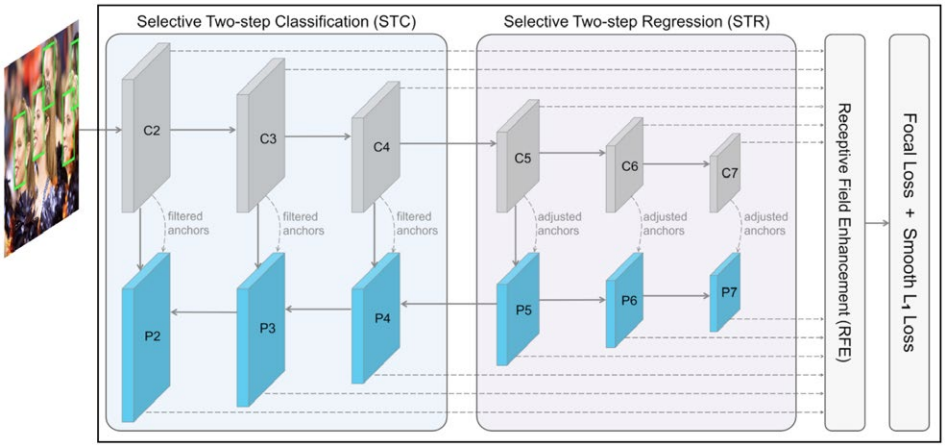
S3FD:



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:

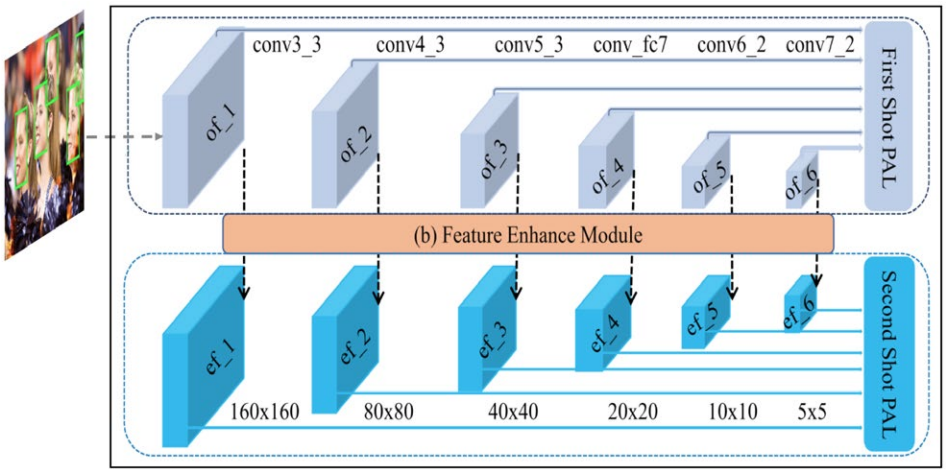



图 2　四种检测结构

| 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。

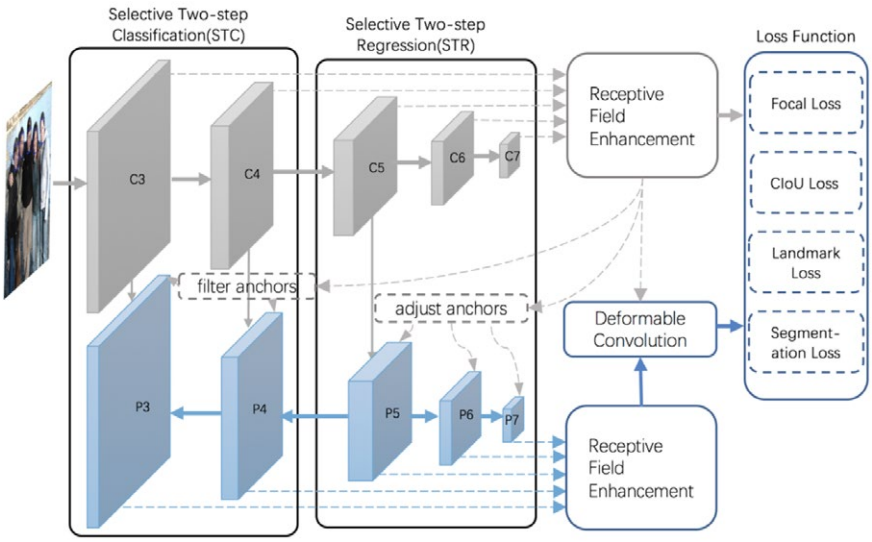



图 3　视觉智能中心 VICFace 网络整体结构图

- 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。

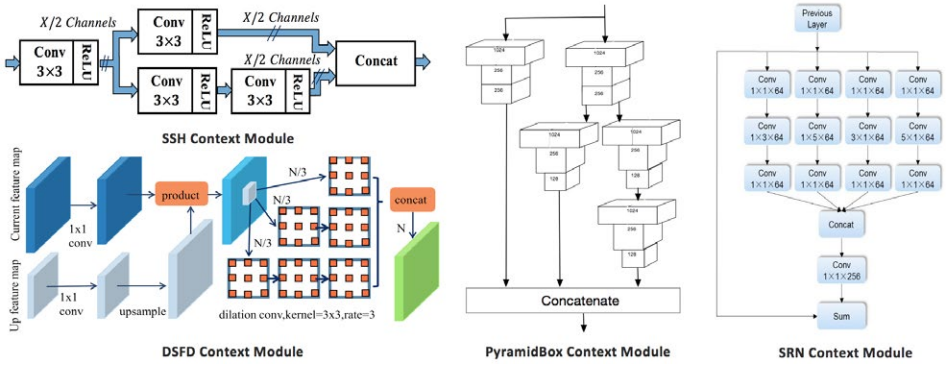



图 4　不同网络结构中的 Context Module

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。

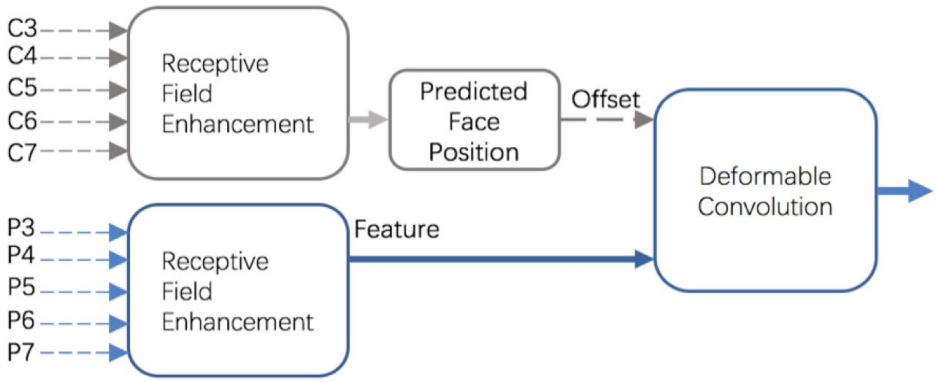



图 5　自研检测模型结构中的预测模块

- 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。

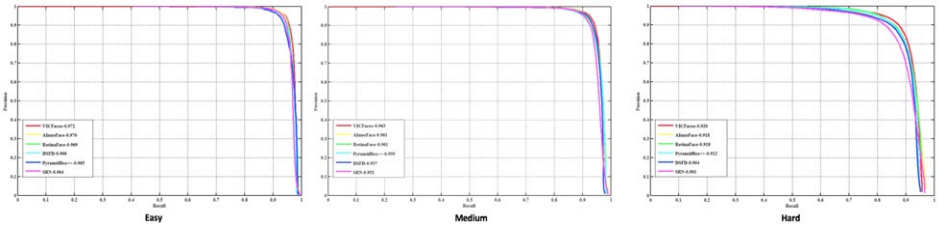



图 6　VICFace 以及当前主流人脸检测方法在 WIDER FACE 上的测评结果

| 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）。

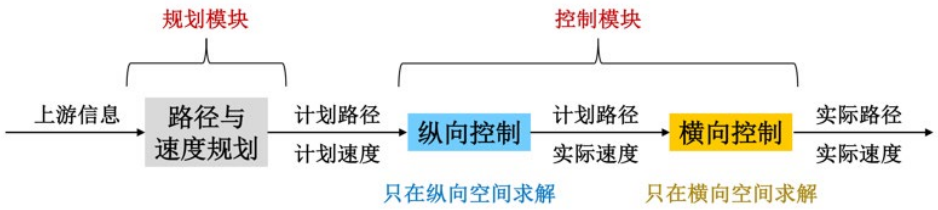



图 1　经典的横纵分离控制方案

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：

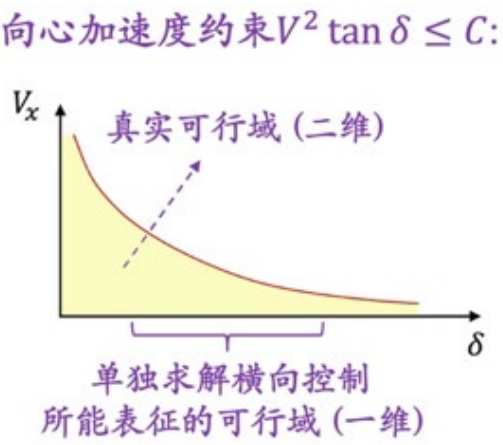



图 2　向心加速度约束所对应可行域的示意图

(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：

$$\begin{cases} V_x^2 \tan\delta \cdot 1/L \le a_{\mathrm{n,max}} \\ -V_x^2 \tan\delta \cdot 1/L \le a_{\mathrm{n,max}} \end{cases}$$

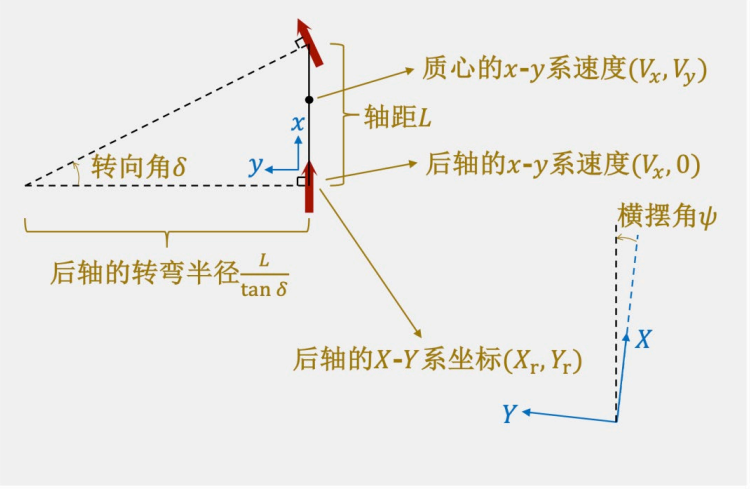



图 4　横纵耦合二轮运动学模型的示意图

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 $\tau$（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 $q_V$ 、$q_X$ 、$q_Y$ 、$q_{\psi}$ 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]：

$$\begin{gathered}\min_{\{\mathbf{u}_k\}} \sum_{k=1}^{N} \left[(\mathbf{x}-\mathbf{x}^{\mathrm{ref}})^{\mathrm{T}}\mathbf{Q}(\mathbf{x}-\mathbf{x}^{\mathrm{ref}})+\Delta\mathbf{u}_k^{\mathrm{T}}\mathbf{R}\Delta\mathbf{u}_k\right] \\ \text{s.t.} \quad (\text{for } \mathrm{k}=0,1,\ldots,\mathrm{N}-1) \\ \mathbf{x}_{k+1}-\bar{\mathbf{x}}=\mathbf{A}_{\mathrm{d}}(\mathbf{x}_k-\bar{\mathbf{x}})+\mathbf{B}_{\mathrm{d}}(\mathbf{u}_k-\bar{\mathbf{u}})+\bar{\mathbf{d}}_{\mathrm{d}} \\ \mathbf{C}(\mathbf{x}_k-\bar{\mathbf{x}})+\mathbf{D}(\mathbf{u}_k-\bar{\mathbf{u}})+\bar{\mathbf{e}}\leq\mathbf{m} \\ \mathbf{u}_{k+1}=\mathbf{u}_k+\Delta\mathbf{u}_{k+1} \\ \mathbf{u}_{\min}\leq\mathbf{u}_{k+1}\leq\mathbf{u}_{\max},\ \Delta\mathbf{u}_{\min}\leq\Delta\mathbf{u}_{k+1}\leq\Delta\mathbf{u}_{\max}\end{gathered}$$

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.


CODE A BETTER LIFE
一行代码 亿万生活
美团
技术团队
长按二维码关注我们