# 深度学习模型在互联网公司的实战讲解系列(二)-itme2vec在真实推荐场景中竟然可以这么做！！

原创 DASOU NLP从入门到放弃 2020-06-17

收录于话题
#深度学习模型实战讲解

7个

## 1. 简单介绍

这是一篇对《Embedding技术在民宿中的应用》精读的笔记。

最近知乎或者公众号有很多关于embedding在推荐系统的总结，总结的都很好，不过很多是一些系统性的概述，并没有深入模型细节去讲怎么训练，怎么优化，怎么部署之类的东西。（强推一下腾讯技术那篇，写的真挺好的）

之前看过不少关于此的文章，偏工程实战的，我做了点笔记，慢慢发上来，争取包保证周更。

对原文一句话总结就是讲了如何使用 skip-gram 训练得到 item2vec-embedding ，对相似房源进行推荐，详细阐述了数据获取，模型训练细节，模型优化细节，冷启动问题。

我先在第一部分来个简单概述，然后来个详细介绍，时间不够的大佬们看完第一部分就可以。

所有类似的相关模型实战文章讲解，全都放在了这里,github可能看起来舒服点，在这里外部链接不能跳转真的影响体验：

https://github.com/DA-southampton/Tech_Aarticle

（话说这两天的star让我有点受宠若惊）

## 2.懒人介绍

这个文章简单来说，就是使用用户对房源的点击日志，构造数据，使用item2vec方法来训练房源对应的embedding，上线之后效果不错。其中，我比较关注的点有以下几个：

### 2.1 训练数据是如何构造，

这里细节比较多，比如根据间隔时间确实是一段序列还是两个序列，比如如何构造训练数据中的的正样本，如何构负样本，比如为何把全天的点击房源会当做同个用户所有序列的正样本，而且权重更高（权重就是当做点击五次，很有意思。）

### 2.2 模型训练细节，

使用的800万点击日志，构造了4000万训练数据，训练了一天的时间。

### 2.3 embedding的更新规则

途家并没有简单的对于新的训练数据进行重新训练，而是做了一个类似预训练的东西，使用当前的权重，然后使用近两个月的日志作为新的数据进行训练，不存在的样本放到矩阵模型随机初始化。

### 2.5 房源冷启动的问题

就是对于新样本这个embedding如何确定。这一点上，很好理解，就是找到和当前新的房子比较相似的房子集合，做一个embedding的平均就可以，至于说这个房源相似房源的获取，途家这里采用的就是设定维度比如房间数目比如是否靠海之类的特征表示进行相似度的判定。

总的来说，这个文章可以借鉴的地方还是很多的，精读还是有必要的，所以如果有时间看看我下面的详细介绍更好，没时间看到这里就可以了。

## 3.详细介绍

首先我们需要理解的一个点是，对于一个app来说，最主要是就是两种数据，一种是用户数据，一个是商品数据。所以在做推荐的时候，可以从用户角度出发进行推荐，也可以从商品角度出发进行推荐。

### 3.1 数据特点

对于途家这种app的数据，按照原文的说法，叫做(引自原文)“用户消费频次低，用户兴趣点不好描述”。这两个特点是从用户角度来描述的。

用户消费频次低，其实很好理解，我们出去民宿一般是放假旅游才会去，所以消费频次当然低。

用户的兴趣点不好描述，我是这么理解的，每次出游的目的地不同，同行的人不同等等，就会导致每次选择不同，换句话讲，就是兴趣点不是特别稳定的。

比如说，这次你和女朋友一起去，肯定希望来个情侣大床那种，如果带着孩子，你就需要考虑孩子的感受，带着父母就就又换了另一种要求，等等吧。

说上面这个是什么意思呢？就是说，从用户的角度进行推荐不好做！！今天推荐给你的是按照你半年前出行的兴趣点来推荐，效果能好才有鬼了。

所以基于此，途家是从商品角度的进行推荐，以不变应万变，不管用户怎么变，商品性质是相对固定的（肯定不是不变的，所以我说的是相对）。

### 3.2 推荐方法抉择

从商品方向进行推荐，途家考虑了三种个性化的方案，分别是基于内容，基于item-to-item，基于embedding进行推荐。

首先来说基于内容的推荐。基于内容的推荐本质是在计算内容相似度，所以只需要对商品特征维度确定好，然后做好特征工程就可以，这种方法在各大公司还是有上线的。在途家这里，对应商品特征就是，图片颜色是什么样子的，装修风格是怎么样的，是否适合情侣，是否适合孩子等等特征。缺点就是数据需要大量人工标注。

对于基于item的协同过滤来说，本质上也是在计算商品相似度，只不过使用的是有多少共同的用户喜欢。按照途家的说法，在他们的实践中，在酒店民宿上会陷入以地标相似为主的窘境。这一点我没太理解，我猜测是点击用户的问题？？希望有大佬解惑。

第三种方式就是基于embedding为主的推荐，这是原文分享的重点信息。

对于embedding代表房子信息这个方法，需要注意到两点，一个是内积大的代表相似度高，一个是一个用户在一段时间内（我理解是在一个搜索需求之下的）浏览过的房子是具有内在相似性的。

这个很好理解，我们去打开一个app看旅游民宿，肯定是搜一遍，慢慢的看，肯定是在最后的一段时间点击的概率最大，在这个过程中，你会跳过一些房子（感受一下，这个真的很像一个句子的形式，有的单词重要，有的单词就是停用词，直接跳过就可以）

### 3.3 Skip-gram训练

对于embedding来说，一般做graph-embedding或者item2vec。途家使用的是第二种，采用的item2vec（skip-gram）

Skip-gram模型在房产中如何应用呢?

（下面的9个小点是引自原文，加上了我的理解）

1. 一段时间内的浏览商品行为作为一个无序序列，也就是一个无序的句子。至于说为什么看做无序，我是这样理解的，用户看过的房源序列肯定是和时间有关系的，但是这个关系有多大是个问题，如果很大，那么就是类似语言模型，直接看做有序，如果关系不大，就没必要了。
2. 两个行为超过半个小时，按照两个行为序列来看。很好理解，就是超过一段时间，很有可能用户就去做别的事情了，下一个行为序列可能就发生了变化，所以看做两个序列是正常的。
3. 一个序列中，点击的房子作为正样本，跳过的房子作为负样本。对应到句子上，就是上下文为正样本，非上下文为负样本（非上下文可能是句子内部也可能是非句子内部）
4. 当天下单的房子作为当天所有上下文的正样本，而且权重更大。也就是在训练的时候，当天下单的房子可以作为所有序列的正样本。这样有一个好处就是有的序列正样本较少，可以适当的补充正样本数量。途家是把它当做了五次点击行为。
5. 正样本中任何一个房子可以作为输入

6. 其前后 2 个正样本及下单房子作为输出中的正样本。 （这个没太理解，前后指的是上下文中的吗？还是前后序列，感觉应该是上下文中的正样本，这样下单的样本就保证了2个数字）
7. 采样一个上下文中跳过的 8 个房子作为输出中的负样本
8. 补充采样上下文中目的地 (可能多个) 的若干个房子作为负样本
9. 一个训练样本包括，输入：1 个房子，输出：64 个房子

### 3.4 模型训练细节：

800万个浏览日志行为，对应4000 万训练样本，700 万评估样本。

实际落地技巧（下面6个小点是引自原文加入我的理解）：

1. 过滤掉停留时长太短的点击行为。很好理解，这种行为不足以当做正样本，也不满足负样本，所以过滤掉
2. 过滤掉点击数量太多的用户行为。我的理解是这种行为不能表示商品特征。我们想要的是什么样本呢？是一个用户看了十来个房子，然后点击进入，觉得不错，下单了。如果这个用户频繁点击，说明这个用户不是在一个正常的搜索房源的行为下。
3. 上下文不能太长：+/-2 个点击，行为间隔 <30 分钟（时间这个我上面说过了）
4. 下单参与到用户当天所有上下文中。我的理解是确保每个序列都可以有正样本。
5. 对于负采样，一定要采样用户跳过没点的房子，数量不能比正样本多太多。这种样本能更好的表达负样本信息
6. 一定要采样同目的地的其他房子，数量和用户跳过房子相当

### 3.5 冷启动问题：

1. 找到与新上房子最接近的房子小集合，对小集合中的 Embedding 向量求均值作为该新房子的 Embedding 向量。其实这个方法还是很普遍的，一般来说对于embedding的冷启动或多或少会采用这个方法
2. 小集合确定方法：在距离，房型，价格，图片分，面积，人数等已有数据方面尽量接近。感觉这个方式相当于相当于在一个小范围上做一个相似度的判定

### 3.6 Embedding 的迭代更新

很有意思，不是重新训练，而是加载上一次参数，对2个月内新样本进行训练，相当于预训练。测试表示，这样比重新从初始化训练更快收敛。

打完收工，点个在看！！感谢！！也推荐大家去看原文，有不同的理解欢迎探讨！！感谢原作者写的好文章！！

这个文章放在了开头的github链接里，这是个系列，所有模型实战的文章都在放这里了，涉及到的部署，推理，搜索，推荐等等，大概长这个丑样（哭了）：

## 最近更新文章

因为下面的文章是按照领域划分的，顺序是按照我自己觉得不错的文章在前面，所以担心我最近更新的理解的文章大家看不到，所以单开一个版块，把最近读的文章迭代列出来，保持五篇吧。

| 最近更新文章 | 简单介绍 | 进度(粗度/精读) |
| --- | --- | --- |
| 精读-Embedding技术在民宿推荐中的应用-201907 | 使用item2vec对app内房源进行embding，然后进行推荐，细节比较多，包括训练细节，数据构造细节等等，推荐看一看，我自己有精读，大家可以对照着看一看 | 精读完成 |

## 部署

在我实际工作中，一般来说部署就是Flask+负载均衡，或者Grpc来提供服务。这个模块积累一下我看到不错的模型部署不错的文章

| 部署领域相关文章 | 简单介绍 | 进度(粗读/精读) |
| --- | --- | --- |
| 蘑菇街自研服务框架如何提升在线推理效率? | 使用协程解决并发问题，使用FLask提供Restful接口，进行容器化部署 | |
| 如何解决推荐系统工程难题——深度学习推荐模型线上serving? | 介绍了几种serving方式，值得一看 | |
| 爱奇艺基于CPU的深度学习推理 | 爱奇艺主要是在算法，应用以及系统三个方面对模型的部署进行优化，系统级主要是针对硬件平台上做的一些性能优化的方法，应用级是跟特定应用相关的分析以及 | |



参考：

Embedding技术在民宿推荐中的应用

https://mp.weixin.qq.com/s?__biz=MzU1NTMyOTI4Mw==&mid=2247491647&idx=1&sn=787f20dad8e613c0f72142df6789d91d&chksm=fbd75253cca0db45abed1a13b555f358ce059b4e9477afaeb872cab12d4fc310f119d65f797a&scene=27#wechat_redirect