**基于用户画像和文本信息的问题推荐策略**

**MemoryError**

**陈雄君 陈垂泽 黎潇潇**

**华南理工大学**

**【摘要】**

**本文基于用户画像和文本信息对新问题进行用户推荐，旨在高效地将用户新提出的问题邀请其他用户进行解答，以及挖掘用户有能力且感兴趣的问题进行邀请下发，优化邀请回答的准确率，提高问题解答率以及回答生产数。**

**【关键词】文本信息；用户画像；偏好统计。**

**【正文】**

1. 赛题简介

本赛题数据分为用户相关特征数据集(member info)、用户回答相关特征数据集(answer info)、问题相关特征数据集(question info)、邀请情况数据集(invite info)以及单字(single word)、切词(word)和话题(topic)的embedding数据集。赛题旨在根据历史的问题邀请被回答情况，预测问题未来是否会被所邀请用户回答。

从原始数据所包含的信息可知并非纯粹根据文本信息进行问答匹配，进一步地可能涉及到所邀请用户的画像构建，即挖掘用户对问题的偏好情况以及探索用户的自身水平。

1. 数据探索

本次比赛A、B榜数据为训练集之后一周数据，并且官方从其中随机采样出一半作为A榜数据，剩下一半作为B榜数据。从下图可以看出，A榜测试集的每日数据量约为训练集每日数据量的一半，在两份数据间存在较大的落差。

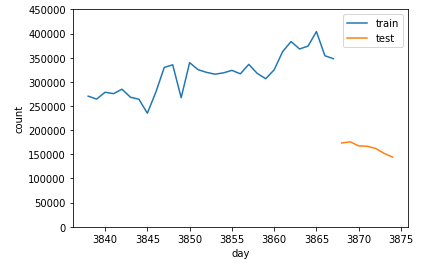


图1. A榜测试集和训练集的每日数据量

因此，当我们在构造全局特征时，就需要注意这种采样带来的影响。例如，当我们统计用户的曝光数的时候，如果仅使用A榜的数据，后一周的用户曝光数其实仅为实际曝光的一半。为此，我们在A榜阶段构造特征时，先对A榜数据进行复制，然后对两倍的A榜数据进行操作。而在B榜阶段，我们将A、B榜的数据合并用于构造特征。

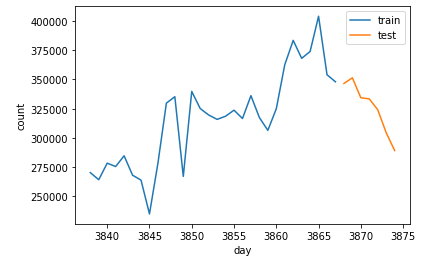
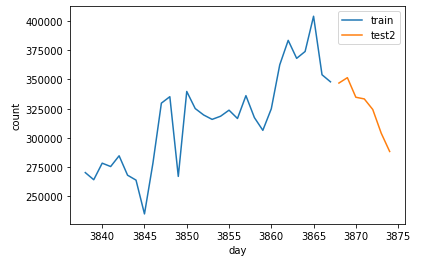


图2. 左图为复制后的A榜测试集和训练集的每日数据量，右图为AB榜合并测试集和训练集的每日数据量。

1. 特征工程

针对本次数据集所构建的特征主要可归类为：横向特征、统计特征、相似度特征和深度提取特征。各类特征说明如下：

（1） 横向特征：本类特征均为对单行数据进行构建，主要集中对member info、invite info以及question info三个数据集进行特征构建，横向特征可概述为原始特征、SVD特征、长度(计数)特征、间隔特征、时间特征和交互特征，如表1所示。

表1. 横向特征及描述

|  |  |
| --- | --- |
| **横向特征** | **特征描述** |
| 原始特征 | 用户二分类/多分类特征、盐值分数、性别、访问频率 |
| SVD特征 | 问题绑定的话题、用户关注的话题、用户感兴趣的话题所对应序列的tf-idf特征，并用svd降维。 |
| 长度特征 | 用户关注/感兴趣的话题数量、问题绑定的话题的数量、问题标题/描述的单字/切词长度 |
| 间隔特征 | 邀请时间和回答时间的间隔、邀请时间和问题创建时间的间隔、问题标题字数和问题描述的字数差 |
| 时间特征 | 问题创建时间(hour)、邀请创建时间(hour) |
| 交互特征 | 问题标题和描述相同的数量、问题标题和描述的编辑距离、问题绑定话题和用户感兴趣/关注话题的编辑距离，等等。 |

（2） 统计特征：本类特征为对合并后的数据集进行的全局/局部统计，统计特征可概述为：盐值分数统计、用户/问题的全局统计、用户/问题的SVD、用户/问题曝光局部统计、时间统计、曾回答问题的状态统计、长度统计。统计特征详情见表2。

表2. 统计特征及描述

|  |  |
| --- | --- |
| **统计特征** | **特征描述** |
| 盐值分数统计特征 | 问题对应的用户盐值分数的mean/max/min |
| 用户/问题全局统计特征 | 用户对应问题的数量、问题对应用户的数量(亦即用户和问题的value-count) |
| 用户/问题的SVD特征 | 用户对应的问题列表和问题对应的用户列表(groupby)的tf-idf接svd |
| 曝光特征(局部统计) | 用户/问题的曾经曝光数、未来曝光数、当天的曝光数 |
| 时间统计特征 | 邀请创建时间、问题创建时间的mean/max/min/std、曾经回答问题的时间间隔的mean/max/min |
| 曾回答问题的状态统计 | 问答是否被标优/被推荐……的count/mean、回答字数/点赞数的sum |
| 长度统计特征 | 用户回答过的问题标题/描述的平均长度 |

（3）相似度特征：本类特征为对各序列进行求取余弦相似度，主要分为对各序列所对应的id的embedding序列求average后求余弦相似度，以及对各序列的每个id的embedding求余弦相似度后再求max/min/mean的统计特征，如表3所示。

表3. 相似度特征及描述

|  |  |
| --- | --- |
| **相似度特征** | **特征描述** |
| 曾经回答过的问题的相似度特征 | 曾经回答过的问题的标题和本次邀请问题标题的余弦相似度、曾经回答过的问题的单字/词描述与本次问题的单字/词描述的余弦相似度 |
| 用户和问题话题的相似度特征 | 用户感兴趣/关注的话题和问题绑定的话题的余弦相似度 |
| 曾经回答过的问题话题的相似度 | 曾经回答过的问题所绑定的话题与本次邀请问题所绑定话题的余弦相似度 |
| 曾经回答过的问题话题相似度的统计 | 曾经回答过的问题所绑定话题与本次邀请问题所绑定话题分别求取余弦相似度再作max/min/mean统计 |

（4）神经网络提取特征：本类特征为通过不同网络提取序列特征，对各编码序列作不同预处理后分别输入不同的网络，根据五折所得的out-of-fold预测作为新特征。深度提取特征详情见表4。

表4. 深度提取特征及描述

|  |  |
| --- | --- |
| **深度提取特征** | **特征描述** |
| wv\_sum\_mlp | 各编码序列求sum后通过一个三层MLP得五折oof作新特征 |
| wv\_lstm | 切词和话题编码序列输入BiLSTM+Attention，根据所给d2v所得vectors作为embedding层权重，得到五折oof作为新特征 |

1. 基于神经网络的序列特征提取

在本次比赛中，我们进一步使用神经网络来对序列数据进行建模，并用于提取序列特征。我们使用序列数据包括：用户关注话题、问题所属话题、用户历史回答问题的标题和描述、问题的标题和描述。对应不同序列的网络结构如下:

1. 用户关注话题、问题所属话题

我们尝试了两种不同结构的网络。首先序列特征经过Embedding层，中间（a）使用卷积层+池化层（Average/Max）、（b）使用Flatten直接拉伸Embedding向量，最后再经过若干层全连接。网络结构如下：

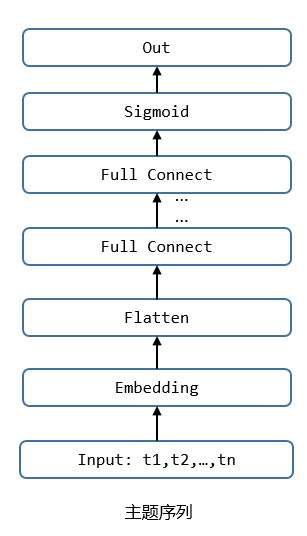
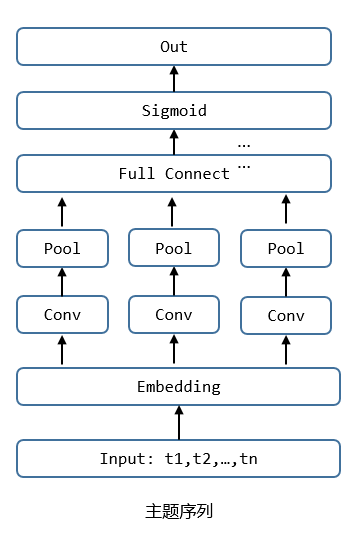


图3. 左图为网络（a），右图为网络（b）

（2）用户历史回答问题的标题和描述、问题的标题和描述

这里我们同样也尝试了不同结构的网络。上述四组序列经过Embedding层后，再使用（a）卷积层+池化层、（b）LSTM+Attention进行特征提取，最后再接全连接层。（c）我们还尝试先对两组标题向量、两组描述向量先分别计算余弦相似度，再接全连接层。网络结构如下：

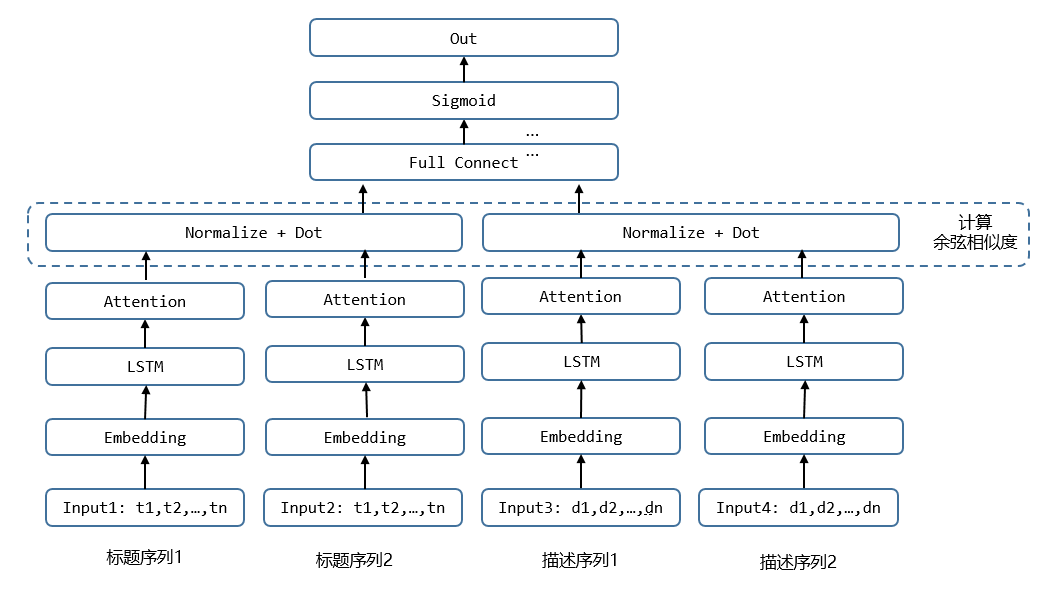


图4. LSTM + Attention (+ cos sim ) 网络

5 实验结果

通过上述的特征构建后，全量特征被使用在树模型以及神经网络模型中，在模型选择方面，因数据较大，选择了可使用并行GPU的Catboost、Xgboost以及Wide& Deep网络，并最终将三个模型作简单的blending，最后再把两个模型组所得结果加权平均。

以CChan的特征和模型在A榜数据为例，仅使用人工构造的特征可达到0.86+的分数，加入神经网络提取的序列特征后，分数达到0.870。对Catboost，Xgboost和Wide & Deep的预测结果进行融合，分数可达到0.873。A榜采用上述特征， Travis的Catboost模型的AUC最高分为0.87559，相同特征通过三个不同模型融合可得0.878左右的分数，与CChan模型所得结果加权融合后可得0.88013左右的分数。B榜根据特征分布情况剔除部分特征并加入在A榜丢弃的特征，我们在B榜得到0.89449的分数。比赛过程的大致得分曲线如下：

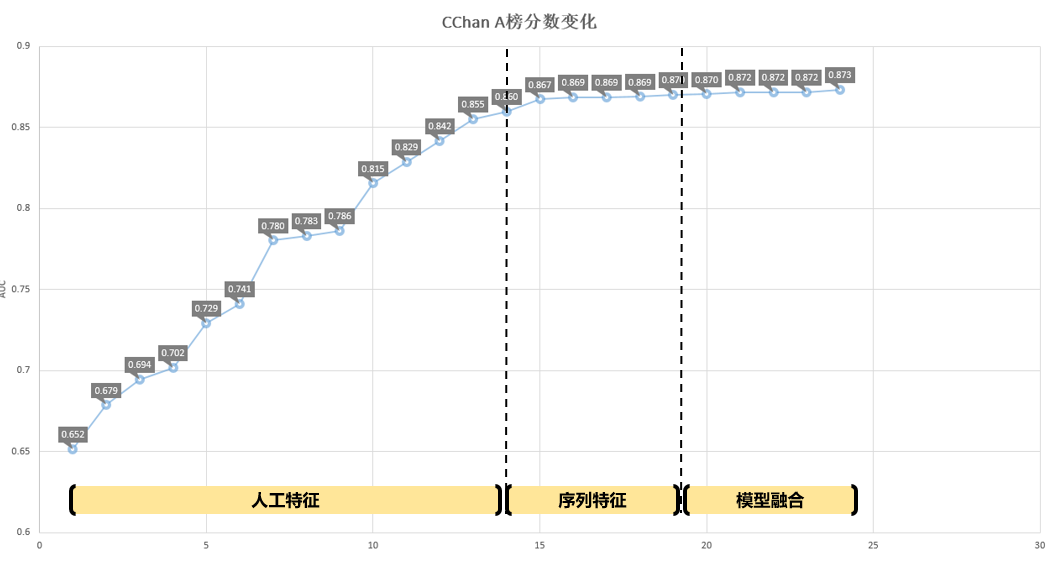


图5. CChan模型特征A榜分数变化