ICME2019 Short Video Understanding Model Document

Team: zookeeper

Track 1 Method

1. XDeepFM-based method

该方法基于 XDeepFM 模型，具体特征工程和模型描述如下。

特征工程

1) 计数特征：
   文件：count_feats_series_1.py, count_feats_series_1.py, count_feats_series_1.py
   描述：计算单个类别特征和多个类别特征共现的次数
2) 人脸特征：
   文件：003_face_feats_1.py, 003_face_feats_1_2.py,
   描述：人脸的数目，男性数目和女性数目，人脸位置，高度和宽度，面积，beauty
3) 标题特征：
   文件：004_title_feats.py
   描述：题目长度，题目中包含的词的数目
4) 比例特征：
   文件：005_ratio_feat.py
   描述：在当天和当前小时下 'uid', 'item_id', 'item_city', 'author_id', 'duration_time', 'music_id', 'device' 的出现次数和比例。

模型

文件：model.py, train_fm.py;
描述：模型使用 XDeepFM，其中 'uid', 'item_id', 'author_id', 'item_city', 'channel', 'music_id', 'device' 作为稀疏特征，其余特征作为 dense feature 输入到模型。

2. Result

基于上述模型对 Track1 线上数据进行预测，最优结果为：0.777015024545725

Track 2 Method

1. LGB-based method

该方法基于 LGB 模型，具体特征工程和模型描述如下。

特征工程

1) 基础特征：原始特征
2) 统计特征：我们用的都是常规操作，如 count、ratio、nunique 和 ctr 相关特征。count：一维+二维 count 计数特征 # 对交叉特征求 count
3) ratio：类别偏好的 ratio 比例特征
4) nunique：类别变量的 nunique 特征
5) face 相关的特征：图像的位置（width, height, x, y），beauty 的统计特征(max, avg),男性数量，女性数量，是否有男性或者女性，face 的数量等 ['face_nums', 'x', 'y', 'width', 'height', 'size', 'male_cnt', 'female_cnt', 'avg_beauty', 'max_beauty', 'author_male_cnt', 'author_female_cnt'， 'uid_female_ratio']
6) title 相关的特征：title 中不同词的数量(unique)以及 title 的长度
7) 在该条样本时间前，针对 uid，authorid，musicid 等 组合的正负样本数量统计特征

模型

- 最终使用了 基础特征，count 特征，ratio 特征，face 特征，title 特征，正负样本数量统计特征
- 针对 finish 和 like 采用上述的同一套特征，使用 lgb 模型，对两个任务分别预测
  - clf = lgb.LGBMClassifier(
    boosting_type='gbdt', num_leaves=100, reg_alpha=0.0, reg_lambda=1,
    max_depth=-1, n_estimators=args.num_trees, objective='binary',
    subsample=0.7, colsample_bytree=0.7, subsample_freq=5,
    learning_rate=0.05, min_child_weight=100, random_state=2018, n_jobs=6, verbose=1
)

2. XDeepFM-based methods

该方法基于 XDeepFM 模型，基于不同的特征输入，训练了两个 XDeepFM 模型，该方法主要考虑了行为特征和受众特征，它们起到了协同过滤作用。具体特征工程和模型如下所述。

特征工程

1) 基本特征：uid, user_city, item_id, item_city, author_id, channel, device_id, music_id；
2) 行为特征：（训练集+测试集中）浏览过的视频、音乐、作者、城市列表，计算 TF 值（取前 500 维)；
3) 受众特征：（训练集+测试集中）视频、音乐、作者的用户 uid 列表，计算 TF-IDF 值（取前 400 维)；
4) 标题特征：计算 TF-IDF 值；
5) 脸部特征：{“num_face”：“人脸数目”，“female_ratio”：“女性比例”，“max_beauty”：“beauty 最大值”，“min_beauty”：“beauth 最小值”，“avg_beauty”：“beauty 平均值”，“max_area”：“最大人脸面积”，“avg_area”：“平均人脸面积”}；
6) 时间特征：通过时间戳获取年、月、日、时、分，以及工作日特征，月-日交叉表示节日特征；
7) video 嵌入：128 维原始特征；
8) audio 嵌入：128 维原始特征；

9) count 特征: 计算单个类别特征和多个类别特征共现的次数。

模型 1

model: XDeepFM
输入特征：特征 1-8
模型文件: XDeepFM.py
result:
public finish auc: 0.7366
public like auc: 0.728

模型 2

model: XDeepFM
输入特征：特征 1-9
模型文件: XDeepFM.py
result:
public finish auc: 0.7367
未训练 like 任务

模型 1 和模型 2 超参数是一致的，隐藏单元数: [200,100,75,50,25], CIN 单元数:[50,50,50,50]。
训练超参数为：batch_size=32, learning_rate=0.005, dropout_rate=0.0。
此外，模型 1 与模型 2 的精度不一样，前者是 float64，后者是 float32

模型训练过程

1） 构建统计特征：用户行为特征、物品受众特征
2） 构建标题特征
3） 构建时间特征
4） 调用 DataParser.py 生成特征文件: 对 track2 数据进行分块，并行构造特征，生成 tf_record 记录
5） 调用 Main.py 进行训练
具体运行命令，请参见模型目录下 build_features.sh 和 run_model.sh

3. 模型融合

Track2 线上最优结果是通过模型融合获得的，融合方式是根据经验启发式的设计各模型结果权重，具体计算公式如下：

finish = (0.5 * xdeepfm1_finish + 0.5 * xdeepfm2_finish) * 0.7 + 0.3 * lgb_finsh
like = 0.4 * xdeepfm1_like + 0.6 * lgb_like

根据上述方式融合之后，track2 线上 private 最终得分为 0.799658049326414。