
电影风向标
制作人：醉舟
项目仓库：
https://github.com/Hansoluo/Movie_learning

项目需求来源

电影院即将上映一部电影，要不要去看呢？打开豆瓣，没有评分。

底下评论有人说好看，有人说难看，该如何选择？

电影风向标为你推荐：摔跤吧！爸爸

功能设想

1. 输入豆瓣电影页面链接，返回电影制作信息
2. 制作信息输入到预测模型，调用模型进行预测
3. 输出预测结果（推荐 |不推荐）

电影风向标为你推荐：帕丁顿熊2

实现方案

X1： 导演
X2： 演员
X3： 电影类型
X4： 制作地区
X5： 上映时间
X6： 电影片长
Y ： 豆瓣评分

电影风向标为你推荐：看不见的客人

具体过程分解

1. 问题定义
2. 获取 & 分析数据集
3. 数据清洗 & 预处理
4. 数据分析 & 探索
5. 构建模型 & 训练参数
6. 验证 & 预测结果

电影风向标为你推荐：请以你的名字呼唤我

获取 & 分析数据集

1. 有哪些特征？哪些是有用的？
2. 哪些特征是类别型？
3. 哪些特征是数据型？
4. 哪些特征是混合类型?
5. 哪些特征是脏数据？
6. 哪些特征数据存在缺失？

电影风向标为你推荐：银翼杀手2049

数据清洗 & 预处理

1. Completing
2. Correlating
3. Correcting
4. Creating
5. Classifying

电影风向标为你推荐：敦刻尔克

数据清洗 & 预处理

1. Completing
2. Correlating

```
df = df.dropna() # 去除缺失的行数
# 筛选数据，选择上映时间在2000年以上，电影时间大于60的
df = df[(df['showtime'] >= 2000) & (df['length'] >= 60)]
# 电影的评分转换为标准字段 1（7.0以上评分推荐） 和 0（7.0以下不推荐）
df['rm'] = df['rate'].apply(lambda s :1 if s >= 7 else 0)
# 做出上映时间的趋势图
g = sns.FacetGrid(df, col='rm')
g.map(plt.hist, 'showtime', bins=20)
```

电影风向标为你推荐：至暗时刻

数据清洗 & 预处理

3. Correcting
4. Creating
5. Classifying

```
df['district'] = df['district'].apply(lambda s: get_district(s))
df['director'] = df['director'].apply(lambda s: s.split("/")[0])
# 演员拆分成2个特征，演员0 和演员1
df['actor0'] = df['actor'].apply(lambda s: s.split("/")[0])
df['actor1'] = df['actor'].apply(lambda s: s.split("/")[1] if
len(s.split("/"))> 1 else "UNK")
# 电影类型拆分成2个特征，类型0、类型1、类型2
df['genre0'] = df['category'].apply(lambda s: s.split("/")[0])
df['genre1'] = df['category'].apply(lambda s: s.split("/")[1] if
len(s.split("/"))> 1 else "UNK")
# 清洗完毕
df =
df[['title','director','actor0','actor1','genre0','genre1','district
','rate','rm']]
df.to_csv('movie_clean_20180109.csv', encoding='utf-8')
```

电影风向标为你推荐：寻梦环游记

数据分析 & 探索

1. Correlating
2. Converting
3. Charting

```
df_data =
pd.read_csv('movie_clean_20180109.csv')
# 特征 导演 文本转数字
director_list =
df_data['director'].drop_duplicates().tolist()
director_dic = {c: i for i, c in
enumerate(director_list)}
df_data = df_data.replace({'director':
director_dic})
```

电影风向标为你推荐：逃出绝命镇

建模 & 训练参数

1. 拆分数据集
2. 构建TensorFlow计算图
3. 训练参数

```
# 构造前向传播计算图
y_pred = tf.nn.softmax(tf.matmul(x, w) + b)
# 构造代价函数
cross_entropy = -tf.reduce_sum(y_ *
tf.log(y_pred + 1e-10), reduction_indices=1)
cost = tf.reduce_mean(cross_entropy)
# 梯度下降算法
train_op =
tf.train.AdagradOptimizer(0.03).minimize(cost)
```

电影风向标为你推荐：芳华

验证 & 预测结果

1. 验证数据集上的预测正确率
   62%~65%
2. 预测电影结果

```
with tf.Session() as sess:
    saver.restore(sess, 'model_movie.ckpt')
    predictions = np.argmax(sess.run(y_pred,
feed_dict={x:movie_test}), 1)
testdata0 = pd.read_csv('movie_test.csv')
movie_pred = pd.DataFrame({
    "title": testdata0['title'],
    "rm": predictions
})
testdata0['rm'] = movie_pred['rm']
testdata0['rm'] = testdata0['rm'].apply(lambda s:"推荐"
if s == 1 else "不推荐")
testdata0
```

电影风向标为你推荐：爱乐之城

近期已上映预测

	title	director	actor0	actor1	genre0	genre1	district1	rate	电影风向标
0	妖猫传	陈凯歌	黄轩	染谷将太	剧情	悬疑	中国大陆	7.0	不推荐
1	星球大战8最后的绝地武士	莱恩·约翰逊	马克·哈米尔	凯丽·费雪	动作	科幻	美国	7.4	推荐
2	追龙	王晶	甄子丹	刘德华	动作	犯罪	中国大陆	7.3	推荐
3	芳华	冯小刚	黄轩	苗苗	剧情	历史	中国大陆	7.8	推荐
4	第一夫人Jackie	帕布罗·拉雷恩	娜塔莉·波特曼	彼得·萨斯加德	剧情	传记	美国	6.6	推荐
5	悟空传	郭子健	彭于晏	倪妮	剧情	动作	中国大陆	5.1	不推荐
6	合约男女	刘国楠	郑秀文	张孝全	喜剧	爱情	中国大陆	4.4	推荐
7	妖铃铃	吴君如	吴君如	沈腾	喜剧	恐怖	中国大陆	4.8	推荐
8	心理罪之城市之光	徐纪周	邓超	阮经天	动作	悬疑	中国大陆	6.1	推荐
9	解忧杂货店	韩杰	王俊凯	迪丽热巴	剧情	奇幻	中国大陆	5.3	不推荐

电影风向标为你推荐：天才枪手

近期未上映预测

	title	director	actor0	actor1	genre0	genre1	district1	rate	电影风向标
10	无问西东	李芳芳	章子怡	黄晓明	剧情	爱情	中国大陆	NaN	推荐
11	卧底巨星	谷德昭	陈奕迅	李荣浩	喜剧	动作	中国大陆	NaN	推荐
12	英雄本色2018	丁晟	王凯	马天宇	剧情	动作	中国大陆	NaN	不推荐
13	纯洁心灵·逐梦演艺圈	毕志飞	朱哲健	李彦漫	剧情	喜剧	中国大陆	NaN	不推荐
14	移动迷宫3：死亡解药 The Maze Runner: The Death Cure	韦斯·鲍尔	迪伦·欧布莱恩	卡雅·斯考达里奥	动作	科幻	美国	NaN	不推荐
15	唐人街探案2	陈思诚	王宝强	刘昊然	喜剧	动作	中国大陆	NaN	推荐
16	捉妖记2	许诚毅	梁朝伟	白百何	喜剧	奇幻	中国大陆	NaN	推荐
17	西游记女儿国	郑保瑞	郭富城	冯绍峰	喜剧	爱情	中国大陆	NaN	推荐
18	狄仁杰之四大天王	徐克	赵又廷	冯绍峰	动作	悬疑	中国大陆	NaN	不推荐
19	祖宗十九代	郭德纲	岳云鹏	吴京	喜剧	奇幻	中国大陆	NaN	推荐

电影风向标为你推荐：弗兰兹

后续改进

1. 提升预测正确率
2. 增加更多分类，辨别烂片
3. 整合成WEB版应用

电影风向标为你推荐：至爱梵高.星空之谜

致谢

童牧玄晨
冯睿博
吴阳平
卓璇
Hugo 沥川
Hysic 张亮
Vwan 从从
DL102全体同学

电影风向标为你推荐：降临