

Introduction

随着近年来互联网的飞速发展，个性化推荐已成为各大主流网站的一项必不可少服务。同一个用户浏览的不同新闻的内容之间会存在一定的相似性和关联，物理世界完全不相关的用户也有可能拥有类似的新闻浏览兴趣。此外，用户浏览新闻的兴趣也会随着时间变化，这给推荐系统带来了新的机会和挑战。因此，在这次大作业中，我们主要根据用户浏览新闻的兴趣，用户之间的相关性，以及结合新闻的热度和时效性来进行新闻的推荐。

Related work

数据处理

- 来源

一万名国内某著名财经新闻网站得用户一个月的全部浏览记录

- 数据格式

共有五个域：用户编号、新闻编号、访问页面的时间(Unix时间戳)、新闻标题、新闻正文，例如：

user_id	news_id	read_time	news_title	news_content
5218791	100648598	1394463264	消失前的 马航370	【财新网】 (实习记者葛 菁) 据新华社消息，马来西 亚航空公司表示...
5218791	100648802	1394463205	马航代表 与乘客家 属见面	3月9日，马来西亚航空公司 代表在北京与马航客机失联 事件的乘客家属见面。

- 数据统计

- 浏览记录数目: **116,224条**
- 数据集大小: **201M**
- 用户数: **10,000**
- 出现新闻数: **6183条**

- 训练集测试集划分

以3月20号为界限，前20天的数据(**83209条**)作为训练数据，后10天的数据(**18995条**)作为测试数据。

- **数据集分隔、存储与读取**

源数据大小达到了200M，如果直接在这些数据集上处理的话效率将相当低下。因此采用下列措施以提高算法的效率：

1. 数据读取等处理采用[pandas](#)(Python Data Analysis Library), pandas是基于[numpy](#)的开源数据处理工具，提供了高效地操作大型数据集所需的工具；
2. 由于数据集主要部分是新闻内容(200M存储空间中由180M是新闻内容占用)，所以可以将数据集按各个域分隔，并将有关联的数据结合形成表，做处理时只对相应数据表处理；
3. 将抽取到的特征集合存储成文件，供下一步分析使用，减少重复处理数据造成的性能影响。

具体分隔成的数据子表如下：

表名	内容	作用
same_user	在测试集和训练集中同时出现过的用户的id	判断是新用户还是老用户以采取不同的推荐方案
hot_user	阅读新闻最频繁的用户的id	按阅读频繁程度给用户权重
new_id_time_table	将新闻id和对应的内容单独抽取出来	减少计算量和简化操作
tran_user_id	测试集中出现过的用户	简化操作
test_user_id	测试集中出现过的用户	简化操作
testing_data_freq_dict	测试集中出现的新闻的关键词	
U2U_tags	用户相似度	
user_click_data	原始数据集	
user_feature	用户特征	计算用户间的相似度
user_news	每个用户看过的新闻集	通过用户id快速查找其看过的新闻集
user_time_test_table	测试集中用户id与点击时间	将点击时间作为用户登录时间，以此时间推荐最近热点新闻

他们存储在工程的data文件夹下

分词

1. TF-IDF

TF-IDF是一种统计方法，用以评估一词对于一个文件集或一个语料库中的其中一份文件的重要程度。词的重要性随着它在文件中出现的次数成正比增加，但同时会随着它在语料库中出现的频率成反比下降。其计算公式如下：

词频率：

$$tf_{i,j} = \frac{n_{ij}}{\sum_k n_{kj}}$$

表示词条在文档中出现的频率

逆文档频率：

$$idf_i = \log \frac{|D|}{j : t_i \in d_j}$$

其中 $|D|$ 为语料库中文件总数

$$tfidf_{i,j} = tf_{i,j} \times idf_i$$

2. 文档特征提取

求出每个文档tf-idf值最高的前十个词作为该文档的特征，比较文档间相同特征词数目多少来量化任意两篇文章的相似度。

3. 用户特征提取

将测试集中用户浏览过的新闻的特征集中在一起排序，保留排名靠前的特征关键词作为用户的特征。

Approach

基于内容推荐

关键步骤

- 通过结巴分词计算出前20天用户所看新闻的关键词，作为每个用户的喜好特征 (profile)
- 通过结巴分词计算后十天新闻的关键词，来表示每个新闻的特征
- 如果后十天的用户在前二十天看过新闻（即已经知道此用户的喜好特征），根据用户的喜好特征与新闻特征相关性来推荐新闻给用户
- 如果后十天的用户是新用户（冷启动问题），根据一天之类（时效性）新闻的点击量来推荐给新用户，综合了新闻的时效性及点击量来对新用户进行推荐

具体实现方法：

- 把用户看过的新闻连接在一起，通过结巴分词选出tf-idf最大的十个关键词作为用户的属性，例如：

user_id	属性
3506171	李云成 市局 分局 工作 北京市公安局 两会 党建 民警 安保 天安门
60073	互殴 两名 长沙 王锋 乘务 长沙市 行凶 现场 微博 歹徒

- 得到每个新闻的属性，也是取tf-idf权重最大的十个关键词
例如：

news_id	属性
100655099	龙滩 一辆 21 由西向东 娄底市 相撞 越野车 湘阳 街江 2014
100646909	基本 文化 社会 发展 农村 改革 贫困地区 加强 城乡居民 教育

- 如何计算用户与新闻之间的相似度

Jaccard coefficient:

$$J(A, B) = \frac{|A \cap B|}{|A \cup B|}$$

在上式中A为用户属性，B为新闻属性，计算用户属性和所用新闻属性的相似度，推荐前K个相似度最高的新闻

- 对新用户的推荐

根据用户的在线时间，统计统计在线时间前一天的新闻点击次数，根据点击量推荐前K个新闻。

协同过滤推荐

关键步骤

- 计算每个用户的关键词属性
- 计算后10天中老用户与前20天中其他用户间的相似度
- 对相似度高的用户，把相似用户之前的关键词合并
- 利用基于内容的方法对用户推荐前K个相似的新闻
- 对于新用户同样采用基于新闻和时效性和点击量进行推荐

具体方法

- 计算用户之间相似度时，由于用户数量太多，不太可能计算后十天每个用户与前二十天所有用户之间的相关性，因此我们采取的方案是：

* 首先计算出活跃度最高的100位用户

用户活跃度的定义为：用户阅读新闻的数量

- 计算后十天中的老用户与这100位活跃用户之间的相似度，还是使用Jaccard相似度公式
- 如果这一用户与某用户的相似度大于一定阈值，把他们的关键字属性集的并集（即推荐这个用户不曾阅读过的新闻类型）作为新的关键字属性给这一用户
- 基于新计算得到的用户关键词属性进行内容推荐

Experiments

推荐正确的衡量标准

- 后十天的测试集里一共有2070条不同的新闻，如果把新闻随机推荐给每个用户的话概率是1/2070，为了衡量推荐的准确度，我们定义下面两条规则来衡量一个成功的推荐：
 - 用户点击的新闻集如果一半或一半以上在推荐的新闻集里
 - 推荐的新闻集如果一半或一半以上在用户点击的新闻集里

基于内容推荐的实验结果

	用户人数	推荐正确	所占比例
总用户	2915	1109	0.38
老用户	2131	753	0.35
新用户	784	356	0.45

协同过滤推荐的实验结果

	用户人数	推荐正确	所占比例
总用户	2915	1078	0.37
老用户	2131	722	0.34
新用户	784	356	0.45

实验结果分析

- 从上述结果来看推荐结果还是挺好的，随机推荐的概率仅为0.000483，推荐结果的成功率增加了约700多倍。
- 上述结果还可以看出基于新闻的点击率和时效性的推荐比基于内容和协同过滤的推荐准确率要高，这也和新闻推荐的特殊性有关，和其它推荐任务(如电影推荐)相比，新闻具指数上升，幂率下降的特性，所以基于新闻的点击率和时效性的推荐是十分有效的。

优缺点和改进方案

- 基于内容和协同过滤的推荐优点是容易理解，计算简单，但没有综合考虑多种特征。
- 在上述方案中，还有一些改进的地方：
 - 1. 没有考虑用户和新闻关键词的权重，用户和新闻属性集里的关键词被同等对待
 - 1. 在考虑新闻的时效性时，没有量化时间的作用因子，只是把一天以内的热点新闻推荐给新用户
 - 1. 在对老用户进行推荐时，没有考虑新闻的热度和时效性，只是根据用户的推兴趣和相似用户进行推荐