机器学习第三次作业

李沛泽*:1701111586

晁越†:1601110127

2018.01.03

*工学院 ,1701111586@pku.edu.cn

†物理学院 ,litterel@pku.edu.cn

Table des matières

1 基本算法介绍

1.1 FP-growth 算法

本次作业中核心算法是 FP-growth 算法，在团队查找以及最常涉猎主题查找中均主要使用该算法。

FP-growth 算法基于 Apriori 构建，但采用了不同的技术。该算法核心是构建一个 FP 树，然后在 FP 树结构中挖掘频繁项集。FP-growth 算法比 Apriori 算法执行速度快很多，一般性能要好两个数量级以上。

FP-growth 算法发现频繁项集的基本过程如下：

1. 构建 FP 树
2. 从 FP 树中挖掘频繁项集

在具体程序中，创建了一个类 treeNode 来保存 FP 树的每个节点数据，使用 treeNode 类来创建 FP 树。FP 树的构建函数为 createTree()，FP 树构建过程中会扫描数据集两次。主要步骤如下：

1. 遍历扫描数据集并统计每个元素项出现的频度，这些信息被存储在头指针表中
2. 扫描头指针表删掉那些出现次数少于 minSup 的项。如果所有项都不频繁，就不需要进行下一步处理
3. 对头指针表稍加扩展以便可以保存计数值及指向每种类型第一个元素项的指针
4. 然后创建只包含空集合的根节点
5. 再一次遍历数据集，这次只考虑那些频繁项

创建好了 FP 树之后，就可以从中挖掘频繁项集，挖掘算法的实现在 mineTree() 函数中，主要有以下流程：

1. 对头指针表中的元素项按照其出现频率进行排序

2. 将每个频繁项添加到频繁项集列表 freqItemList 中

3. 递归地调用 findPrefixPath() 函数创建条件模式基

4. 以每次得到的条件模式基构建条件 FP 树

5. 对条件 FP 树调用 mineTree() 进行挖掘

算法的具体实现在代码文件中，该算法主要参考自《机器学习实战》，并在此基础上对算法进行了一些改变。

1.2 数据处理

在这次作业中,根据所要处理数据的特点,我们使用了 pandas.DataFrame 型数据结构来储存数据。DataFrame 是类似于 excel 表格的一种数据结构，具有简单直观以及易于检索筛选的特点。调用 loadData.py 文件中的 loadData() 函数，返回一个 DataFrame 类型的 dataset。在实际处理的过程中，由于我们只需要 IJCAI, AAAI, COLT, CVPR, NIPS, KR, SIGIR, KDD 这八个会议的信息，所以其他会议的数据被舍弃。另外考虑到有一些会议有子会议，比如 FCA4AI@IJCAI，MPREF@AAAI，所以我们把这种类型的会议数据都归并到其主会议中。最后需要舍弃掉一些没有作者信息的不完整数据，我们得到了一个包含 26710 条数据的 dataset，表头包括 author，conference，title 和 year 四项，为了便于后续处理，author 项以集合的形式储存作者姓名，其他三项均是字符串，并且将 dataset 按照 year 和 conference 的顺序进行了升序排序，数据输出为 dataset.csv 文件。

2 任务 1

2.1 活跃研究者查找

会议支持者 这一部分的实现比较简单，通过调用 loadData.py 中的 findSuppoters() 函数实现，并且安照会议进行分类。考虑到不同

会议之间文章数目差别很大，我们取 2007-2017 年间每个会议发文章数量前 20 人作为会议的主要支持者，函数返回一个嵌套字典，包括会议的名称，主要支持者及其文章数目。得到的结果保存为为 freqAuthors.txt [1] 文件。

时间变化 为了发现这些主要支持者的活跃程度，我们将 2007-2017 这 11 年分为五个时间段：2007-2009，2010-2011，2012-2013，2014-2015，2016-2017。调用 findSuppotersChange() 函数逐时间段进行分析，不同时间段的文章数储存在一个 list 中，函数返回一个类似于 findSupporters() 的嵌套字典，只是将文章总数换成了不同时间段文章数的 list。得到的结果保存为 authorsChange.txt 文件。

我们发现，有一些会议的主要支持者当中，中国人（或者华裔）的比例很高，比如 KDD，IJCAI，AAAI，而 KR，COLT，SIGIR 中则几乎没有；IJCAI，AAAI 的主要支持者有很大一部分的重叠，比如 Zhi-Hua Zhou，Feiping Nie，Heng Huang 等 13 人，同时是这两个会议的主要支持者；此外，IJCAI，KR 也有小部分重叠的主要主持者，比如 Stefan Woltra，Carsten Lutz 等五人。

观察会议主要支持者的活跃程度可以发现，有一些支持者是在近几年逐渐活跃 , 如表(1) 所示；有一些支持者几年来文章数比较平稳；如表(2)；有一些支持者早年比较活跃，近年来不再活跃，如表(3)。

2.2 研究团队查找

调用 loadData.py 中的 findFreqTeam() 函数可以发现经常合作的团队，这里我们把最小支持度 minSup 设置为 5，并且要求团队人数大于等于 3，函数返回一个字典，得到的结果输出为 freqTeam.txt 文件。表(4) 显示了部分频繁团队。

1. 任务 1 的输出文件都保存在 task_1 文件夹下

Table 1 – 逐年活跃

conference	author	07-09	10-11	12-13	14-15	16-17
AAAI	Feiping Nie	1	3	4	8	17
AAAI	Dacheng Tao	0	1	2	5	15
CVPR	Bernt Schiele	7	8	6	12	20
CVPR	Ming-Hsuan Yang	3	2	8	14	17
CVPR	Stefanos Zafeiriou	1	6	4	10	18
NIPS	Lawrence Carin	4	4	6	11	14

Table 2 – 数量平稳

conference	author	07-09	10-11	12-13	14-15	16-17
CVPR	Shuicheng Yan	14	11	13	14	15
KDD	Jieping Ye	11	7	12	11	11
SIGIR	Iadh Ounis	9	6	9	7	7

Table 3 – 不再活跃

conference	author	07-09	10-11	12-13	14-15	16-17
AAAI	David C. Parkes	8	8	4	1	0
CVPR	Horst Bischof	17	17	9	6	3
SIGIR	Ryen W. White	10	10	14	7	2

Table 4 – 频繁团队

团队核心成员	文章数
Min Zhang , Yiqun Liu, Shaoping Ma	18
Xueqi Cheng, Yanyan Lan, Jiafeng Guo	18
Hua Wang, Heng Huang, Feiping Nie	15
Heng Huang, Feiping Nie, Chris H. Q. Ding	13
Jun Xu, Xueqi Cheng, Jiafeng Guo	12
Jun Xu, Yanyan Lan, Jiafeng Guo	12
Jun Xu, Yanyan Lan, Xueqi Cheng	12
Jun Xu, Yanyan Lan, Xueqi Cheng, Jiafeng Guo	12
Bart Selman, Carla P. Gomes, Stefano Ermon	11
Chang Xu, Dacheng Tao, Chao Xu	11
...	...

3 任务 2

3.1 团队主题查找

在进行团队主题提取的工作上，有下面几点值得考虑

1. 最理想的情况是通过对 title 中的词进行语义分析，从而把相近或相同主题的词分为一类主题词。但由于这项工作已经超出了本课程的范围，难度较大，而且频繁团队的文章数并不多，可以用来学习的样本太少，所以并没有采用
2. 将 title 直接进行分词，去掉一些常见词以及无关紧要的单词，然后进行频数统计，通过频数分布来判断团队的主题，这样做有点类似于朴素贝叶斯法则，不过实际处理当中并没有先验概率以及条件概率分布，所以需要在全部数据集上使用非监督学习方法或者半监督学习方法去生成这些参数，这样的话工作量会非常大，所以我们没有通过求后验概率最大化的方法去提取主题，而是直接使用观察归纳高频词的方法来进行主题提取

3. 由于每个频繁团队的文章数其实很少，由表(4) 可知，最多的也不过 18 篇，而且发表在不同的期刊上，所以我们可以根据期刊类型以及文章的标题直接分析出主题

综合我们要处理数据的特点考虑，我们采用了第 2，3 种方法来进行主题提取。

每个团队的信息都以 .csv 的格式储存在 task_2 文件夹下，主题信息则是将高频词储存为 .txt 文件。为了方便，我们只分析文章数大于 11 的团队[2]，于是我们得到了 10 个团队，包括 9 个三人团队以及一个 4 人团队。

3.2 团队主题及变化情况分析

下面我们将分析团队主题及其变化情况，可以在 task_2 文件夹中找到对应的 .csv 以及 .txt 文件进行检查。

Stefano Ermon, Bart Selman, Carla P. Gomes 该团队一共发表文章 11 篇，发表在 IJCAI，AAAI 以及 NIPS 上；主要涉及 parallel problem decomposition(2015-2017)，markov chain (2011-2012) 以及 partition function(2011-2012) 等主题；在 2011-2014 年，该团队主要与 Ashish Sabharwal 合作，在 2014-2016 年，该团队主要与 Yexiang Xue 合作。

Shaoping Ma, Yiqun Liu, Min Zhang 该团队一共发表文章 18 篇，发表在 IJCAI，SIGIR，AAAI 上；主要涉及 click model(2011-2013)，web search(2015-2017)，information search(2011-2017) 等主题；该团队在 2014 年左右经常与 Yongfeng Zhang 合作，其他时间段的合作者不是很固定，文章作者一般都在 5 人以上。

Jun Xu, Yanyan Lan, Jiafeng Guo, Xueqi Cheng 该团队[3]一共发

2. 要考察的时间段一共是 11 年，如果达不到每年都发一篇文章的话，也没有必要去考察团队主题及其随时间的变化了。

3. 这个四人团队实际上包括了四个三人团队的子集，从数据情况来看，四个三人子团

表文章 12 篇，发表在 IJCAI，NIPS，SIGIR，AAAI 上，并且集中在 2015-2017 年；主要涉及 learning model(2015)，NLP(2016)，markov decision process(2017) 等主题；该团队除了核心四人之外，每年的构成人员都有变化，和 Liang Pang 的合作次数比较多一些。

Feiping Nie, Heng Huang, Hua Wang 该团队一共发表文章 15 篇，发表在 AAAI，IJCAI，NIPS，SIGIR，CVPR 上；主要涉及 multi-instance learning(2011-2012)，semi-unsupervised and unsupervised learning(2013-2017)，以及学习算法的 robust 性能等主题；在 2014 年前该团队和 Chris H. Q. Ding 合作很多，2014 年之后主要是自己独立发表文章。

Feiping Nie, Heng Huang, Chris H. Q. Ding 该团队一共发表文章 13 篇，发表在 IJCAI，NIPS，AAAI，SIGIR，KDD，CVPR 上；主要涉及 matrix theory(2012-2013)，regression(2013-2015)，low-rank(2012-2015)，以及 n-norm maximation and minimization(2010-2012) 等主题；该团队在 2011-2013 年和 Hua Wang 合作比较多，此外和 Xiao Cai 也有一定的合作。

Chao Xu, Dacheng Tao, Chang Xu 该团队一共发表文章 11 篇，发表在 AAAI，IJCAI，KDD，NIPS 上；主要涉及 multi-label(2013-2017)，neural networks(2016-2017) 等主题；该团队在 2013-2016 年期间主要是独立发表文章，2017 的文章主要是和别的作者一起合作完成，文章作者一般是 4 到 6 人。

队和这个四人团队的文章情况大致一样，因此我们只分析这个四人团队的文章，将其子集忽略

4 关于分工和一些细节

这次作业，李沛泽同学完成了 fpGrowth 算法和相关文档的编写工作，晁越同学完成了数据处理，后续分析以及相关文档的编写工作。fpGrowth 算法参考了 “Machine Learning in Action” 第 12 章的内容，数据处理参考了 “Python for Data Analysis” 第 5，6 章的内容。

fpGrowth.py 中包含了 fpGrowth 算法，loadData.py 中包含了所有的数据处理函数，task_1.py 用来输出任务 1 的数据，task_2.py 用来输出任务 2 的数据，两组数据分别输出到 task_1 和 task_2 文件夹中。