基于 DBLP 数据库的科研合作关系挖掘

李屹 (1501110045), 卢煜腾 (?)

April 9, 2016

1 简介

1.1 开发环境与运行依赖

我们的程序在 python2.7 版本下开发并在 Ubuntu 14.04 LTS, Ubuntu 15.10 中测试通过。由于使用了关系数据库引擎 SQLite，因此运行需要依赖 python 中的 sqlite3 扩展，这个扩展可以通过 `pip`, `easy_install` 等工具进行安装。如不安装这个扩展，则整个程序无法运行。

获取数据的过程依赖 urllib 库来发起 HTTP 请求。如果不需要重新获取数据，那么无需这个库也可运行主程序。

1.2 运行与测试方法

在所有依赖库均已安装完成的情况下，在根目录中直接运行相应的 python 脚本就可以看到结果，如：

```
# 获取数据库并保存到默认位置（这两步可以不做）
python scripts/dbgenerate.py
python scripts/dbprepare.py

# 计算合作关系的权值
python hw1.py 1
# 挖掘师生关系与指导时间
python hw1.py 2
# 挖掘频繁合作团队
pythonw hw1.py 3
```

程序运行的结果见 paper/result.txt。

2 数据的来源与数据库的构建

本文所使用的数据来源于 DBLP 论文数据库[1]. 根据作业要求，我们考虑了 12 个数据挖掘相关会议和 4 种相关期刊，每种会议（期刊）中选取不超过 10000 篇文章作为我们的样本。实际上，由于所考虑的会议与期刊所收录总论文数均未超出该上限，因此可以认为我们抓取了上述会议中的全部论文。

[1] http://dblp.org/search/index.php

我们通过 DBLP 所提供的 RESTful API 获取论文数据，由于挖掘目标较为简单，因此我们仅靠虑文章标题，作者，年份，期刊或会议名称等基本信息。从不同来源获得的论文数如下表所示：

期刊/会议	名称 (DBLP 代码)	论文数
会议	sdm	1161
会议	icdm	2274
会议	ecml-pkdd	689
会议	pakdd	1592
会议	wsdm	612
会议	dmkd	61
会议	kdd	2575
会议	cvpr	7049
会议	icml	3123
会议	nips	5599
会议	colt	1204
会议	sigir	3602
期刊	pattern_recognition_pr	7010
期刊	sigkdd_explorations_sigkdd	477
期刊	tkdd	226
期刊	ieee_trans_knowl_data_eng_tkde	3101

Figure 1: 本文所选取的期刊与会议列表

根据上述条件，我们通过 json 格式接口获得共 40355 篇论文的具体信息，并将它们整理保存在 SQLite3 格式的文本数据库中。SQLite3 是一种十分流行的小型关系数据库，它可以单个普通文件的形式保存，因此在处理小规模数据集时具有方便部署的优势。同时 python 提供了 sqlite 的接口，我们的程序在运行时将通过 SQL 语句动态获取部分数据。数据抓取程序在 `scripts/dbgenerate.py` 中实现，数据默认存储在 `dataset/default.sqlite` 中。如果不需要重新获取数据则不需要再次运行这个脚本。

在这次作业中，我们抓取的数据字段包括：标题，作者，年份，期刊或会议名称。

3 频繁项集挖掘算法与优化

由于 Apriori 算法的效率较低，我们在挖掘频繁项时使用了 fp-growth 算法。算法的过程在此不做详述。在 lib/fptree.py 中我们描述了 FP-Tree 的结构以及 FP-Tree 的构造算法，而在 lib/fpgrowth.py 中我们描述了 FP-Growth 算法。

由于 FP-Growth 算法递归地构造规模更小的条件 FP-Tree，而递归挖掘子树的过程实际上互相之间不存在依赖关系。因此我们用 python 中提供的多线程库 threading 对 FP-Tree 算法进行了并行化。不过由于数据量较小，大量的时间耗费在 FP-Tree 的构建过程中，因此并未有明显的时间优势。

我们所实现的算法在解决作业中三个问题时的时间消耗如下：

	计算合作关系的权值	导师 -学生指导关系挖掘	合作团队挖掘
时间消耗 (秒)	7.06	3.95	3.62

Figure 2: 时间消耗

4 基于先验知识的合作关系分析

在本次作业的数据挖掘过程中，我们仅仅通过 FP-Growth 算法挖掘了频繁项集，而没有进行进一步的关系挖掘。之所以这样做，主要是考虑到各项之间的关系在时间上具有不一致性。举例来说，“如果一个人买了啤酒，那么他也有很大的概率会买尿布” 这条性质将会在很长的一段时间内保持稳定，然而 “如果教授 A 发表了一篇文章，那么这篇文章的合作者中很大几率会出现教授 B” 这种关系却随着时间不断变化。

我们假设有研究人员 A、B、C、D，两个人十年之前曾经紧密合作并发表了 10 篇论文，而之后分别又发表了 90 篇。而 C 与 D 均刚开始工作，两人共同合作发表了 5 篇论文，此外并没有单独发表任何文章。我们很难断言 C 与 D 的合作关系要比 A 与 B 的关系频繁，然而如果通过关系分析便很有可能得到这样的结论。

基于这样的考虑，后续的合作关系分析主要基于对学术领域的先验知识。这部分中描述的所有算法均在 `lib/analysis.py` 中实现。

4.1 合作关系的权值

本次作业中，我们对合作关系的挖掘主要是基于 2 次频繁项的。首先我们从 FP-Growth 算法计算好的频繁项集中提取所有二项集，并认为这些二项集中包含的作者对便是我们所要求的合作关系。

假如两个人之间存在学术上的合作关系，我们一般通过如下规则来衡量其紧密程度：

1. 两人共同合作署名多篇论文（在频繁项挖掘的过程中该条件已经满足）
2. 两人的合作较为稳定（如果两人共同发表多篇论文，则论文前后总时间越长，说明合作关系越不稳定）
3. 合作关系分别在两人的科研工作中所占比重

根据以上规则，我们为科研人员 A, B 之间的合作关系赋予权值:

$$rel_{A,B} = P(B|A) \times P(A|B) \times \frac{(A \cup B).sup}{(A \cup B).scope}$$

其中，$(A \cup B).sup$ 和 $(A \cup B).scope$ 分别表示 A,B 共同合作的文章数目以及这些文章发表时间的范围（最晚年份于最早年份之差）。

4.2 导师 -学生指导关系和指导时间

一般而言，一个比较典型的学生 -导师的关系应当包括下述特征：

1. 两人在一段时期内频繁合作发表文章
2. 这样的合作关系通常时间不短于一年

3. 导师与学生在学术领域的积累通常有较大的稳定差距

4. 一个导师通常指导多个学生

在实际的算法中，我们通过下述的筛选过程来寻找导师 -学生指导关系：首先选取 FP-Growth 中计算得到的所有频繁 2 项集, 并考察每个频繁项中的两个作者。如果这两个作者首次发表论文的时间差不小于某个阈值（本文中设定的阈值为 8）[2]，且资历较老的研究人员发表论文数目明显较多（这里设定为超出 5 篇或以上）那么我们初步认为这是一对学生 -导师的关系。

接下来我们对这些关系按导师归类，如果归类后一个导师旗下有超过两名学生，那么认定这是一组合理的导师 -学生指导关系。反之，则认为指导关系较弱，将其从指导关系中删除。通过上述的算法，我们挖掘到了较为可靠的导师 -学生指导关系，具体结果将在下一章中详细描述。

如果仅靠虑国内高校或研究机构的成果，其实可以考虑作者的顺序。通常而言导师署名排在最末并注明为通讯作者，而学生往往排在第一作者的位置。然而在国外，许多研究人员倾向使用字母序，因此这种方法我们并未采用。

4.3 频繁合作关系与合作团队

在实际的研究工作中，一个较强的合作团队通常具有如下特点:

1. 团队论文发表量较大，且合作人员（署名）较多

2. 团队人员更迭迅速，但存在一个核心小团体。这个小团体将在较长时间内同时署名团队的多篇论文

3. 团队具有共同的工作主题，这也使得团队核心成员较为稳定。（如果团队是由多个小组组成，或者团队中每个人都专心于小方向，那么这样的合作关系是比较松散的，并不在我们考虑之内）

因此，我们采取的团队挖掘方法是：首先寻找核心团队，然后将核心团队进行扩充与合并。

在这个方法中，最重要的是对核心团队的定义或者说阈值。我们认为，如果一个频繁 3 次项的支持度超过了 4，换言之，由三人以上共同署名的文章超过 4 篇，我们便认为这是一个团队的核心团体。事实上，这样的核心团体可能是同一个团队在不同时间的核心人员（一名博士毕业，同时团队招收新的博士生，这样核心团队成员就会变话）。如果在上述挖掘出的核心团队中，有两个团队重合人数超过两名，那么我们便将这两个团队合并。

这种方法有助于缩小团队数目，在时间跨度上保证同一团队不重复出现。但是另一方面，它也有可能导致两个合作较为频繁的团队被合并（比如一个大型实验室，有两名老师共同指导多个小组的情况）。通过核心团队阈值的改变，我们可以对这种误差进行适度纠正。当核心团队阈值设定为 4 篇时，挖掘结果如下：

其中合作指数可以看做是团队紧密度的权值，它的计算规则是：

5 关系验证

根据作业要求，我们对挖掘出来的导师 -学生指导关系进行了人工验证。在此处我们所定义的导师 -学生指导关系包括：Master, PhD, PostDoc 以及其他合作指导关系（由学生在页面上明确说明的）。验证结

[2]根据多位老师同学的意见，10 年是一个更为合适的阈值，但是在特定情况下（老师开发表文章较晚而收学生较早）部分关系可能被错误地过滤掉。因此我们设定了较低的阈值并追加进行后续判断, 希望借此得到更加准确的结果

团队成员	合作指数
Shiguang Shan, Ruiping Wang, Xilin Chen, Zhiwu Huang	12
Dinh Q. Phung, Svetha Venkatesh, Sunil Kumar Gupta, Duc-Son Pham, Budhaditya Saha, Santu Rana	16
Huan Liu, Jiliang Tang, Xia Hu, Huiji Gao	10
Shinjae Yoo, Hong Qin, Dantong Yu, Hao Huang	12
Sethuraman Panchanathan, Wei Fan, Ian Davidson, Jieping Ye	10
Jing Gao, Kang Li, Nan Du, Aidong Zhang	8
Jiafeng Guo, Shuzi Niu, Yanyan Lan, Xueqi Cheng	8

Figure 3: 频繁合作的团队

果见图 4。

经过人工核验，我们找出的大多数师生关系均正确，其中准确率较低的各关系也能够得到解释：

Ron Kohavi 是微软研究院的一名研究人员。由于微软研究院给出的页面上并不包括学生信息，因此我们猜想他的 “学生” 实际上应当是研究院中 Mentor 与实习生或者下属的关系，但是这一层无法得到验证。

Jiawei Han 与 **Philip S. Yu**, **Christos Faloutsos** 等人，由于在业界工作时间过长，已经与很多人建立了疑似指导但并非名义师生的关系，这会造成误判。

6 小结与分工

导师	学生	准确率
Ron Kohavi	Toby Walker , Ya Xu	0%
Nicolò Cesa-Bianchi	Fabio Vitale , Giovanni Zappella	100%
Maarten de Rijke	Wouter Weerkamp , Ilya Markov	100%
Shiguang Shan	Zhiwu Huang , Ruiping Wang	100%
Claudio Gentile	Fabio Vitale , Giovanni Zappella	100%
Michael I. Jordan	Fabian L. Wauthier , Martin J. Wainwright	100%
Huan Liu	Xia Hu , Huiji Gao	100%
Christos Faloutsos	Partha Pratim Talukdar , Shiqiang Yang , Alex Beutel , Meng Jiang , Nicholas D. Sidiropoulos	20%
Philip S. Yu	Philippe Fournier-Viger , Chang-Dong Wang , Bo Liu , Cheng-Wei Wu , Zhifeng Hao , Chaokun Wang , Bokai Cao , Jun Zhang , Hong-Han Shuai , Yanshan Xiao	0%
Xingquan Zhu	Shirui Pan , Li Guo , Jia Wu	
Jiawei Han	Latifur Khan , Xiang Ren , Deng Cai , Xifeng Yan , Jing Gao , Mohammad M. Masud , Xiao Yu	57.14%
Wei Fan	ErHeng Zhong , Sethuraman Panchanathan	50%

Figure 4: 导师 -学生指导关系的准确率