复杂结构数据挖掘大作业报告*

荀琪[1+]

[1](南京大学 人工智能学院, 江苏 南京 210023)

Data Mining for Complex Data Objects*

Gou Qi[1+]

[1](School of Artificial Intelligence, Nanjing University, Nanjing 210023, China)

Abstract: In the era of big data, data mining is extremely important. It can help us find the principles in the data, understand the nature of data, and then draw some scientific and regular conclusions. The final task of this Data Mining for Complex Data Objects is to find the principles from the given data, and finally infer whether the given paper matches the author. The data given include paper data, conference data, author data, journal data, noisy paperAuthor data, train data and test data. In this problem, I first carefully observe the data and extract some features, then use the traditional machine learning algorithm for classification. Then I get the model results and analyze the importance of its features, and finally I predict the given test data .

Key words: data mining; feature extraction; machine learning

摘 要: 在大数据横行的时代，数据挖掘显得极为重要，它能够帮助我们发现数据中的规律，看透数据的本质进而得出一些科学性，规律性的结论。本次复杂结构数据挖掘的大作业是从给定的数据中，发现规律，最后推断给定论文和作者是否匹配。所给的数据包含论文数据，会议数据，作者数据，期刊数据，带噪声的论文-作者数据，已经训练数据和测试数据。在这个问题中，我首先仔细地观察了数据，提取了一些特征，然后使用传统机器学习算法进行分类，最终得到了模型结果，分析了其特征的重要性，最后再预测所给的测试数据。

关键词: 数据挖掘;特征提取;机器学习

本次大作业是一个真实场景中的数据挖掘任务，要求预测给定论文是否为给定作者所著。对每个领域的科研工作者而言，搜索和阅读论文是他们的重要工作之一。当他们初入某领域或想深入了解某一领域时，通常会搜索该领域的研究者，并跟随这些研究者的论文进行学习。通常来说，这是可行的，但不幸的是，由于各行各业的科研人员数量众多，可能会出现同名或其缩写名相同的情况，因此导致了很多论文划分到其他错误的同名作者中去。这一现象在中国研究者内更为普遍，因为中国科研人员的英文名就是其姓名拼音。如张三和章三的英文名都是 San Zhang。这跟论文收集和数据整理都带了很大的麻烦，因此我们希望从已有数据中挖掘到一些数据的特征或规律，它能帮助我们预测给定论文和作者之前的关系，判断该论文是否属于该作者。

本文第 1 节介绍所给数据，包括数据的种类以及数据的特点。第 2 节介绍如何从数据中提取出有用的特征。第 3 节介绍我此次任务使用的模型.第 4 节介绍实验结果.最后总结全文.

1 数据总览

本次任务给了很多数据，包括论文数据，会议数据，期刊数据，作者数据，带噪声的论文-作者数据以及训练数据和测试数据。其中各数据所包含的信息如表 1 所示：

表 1 数据集

数据文件	大小	包含信息					
Paper.csv	235.34MB(2259021 条)	Id	Title	Year	ConferenceId	JournalId	Keyword
Author.csv	10.89MB(2293830 条)	Id	Name	Affiliation(作者所属单位)			
Conference.csv	513KB(5222 条)	Id	ShortName	FullName			HomePage
Journal.csv	1.42 MB (22228 条)	Id	ShortName	FullName			HomePage
PaperAuthor.csv	605 MB	PaperId	AuthorId	Name			Affiliation
Train.csv	1.50MB(3739 条)	AuthorId	ConfirmedPaperIds(已经确认是该作者所发论文的 id 列表)	DeletedPaperIds（已经确认不是该作者所发论文的 id 列表）			
Test.csv	626 KB(29675 条)	Id	AuthorId	PaperId			
Test2.csv(track 2)	1.87 MB(90088 条)	Id	AuthorId	PaperId			

2 特征提取

2.1 读取数据

由于本次数据过大，且后续需要经常遍历不同的对象，因此我这里将数据都统一存为字典（dict）类型。此外，如果每次都重新从 csv 文件读取数据再进行整理会格外耗时，因此我这里仅仅第一次读取并整理数据文件，在读取成功后，将字典用 pickle 库保存下来，下一次再调试或运行时，只读取字典文件而不用再读 csv 文件，这样也为数据读取省了一定的时间。

具体实现上，我设计了四个二重字典(defaultdict(dict))来存储数据，Paper,Journal,Conference,Author,分别以键值对的方式来存储论文信息，期刊信息，会议信息，作者信息，key 是其对应 id，value 则是 csv 文件中除了 id 之外的其它信息。

在读取完上述四个 csv 文件之后，我们还剩一个 PaperAuthor.csv 文件，在这个文件中存储了带噪声的 PaperId-AuhtorId 对。此处，我也选择将其存入上述实体中。我将每个 authorId 对应的 paperId 加入 author 字典对应 AuthorId 下对应的 paper 列表中,代表该作者可能发表的论文 list。同理,将每个 paperId 对应的 authorId 加入 paper 实体所对应 author 列表中，同时还要将作者的其它信息（Name,Affiliation）保存在其中。

最后需要注意的是，在读取数据时，需要考虑某些数据缺失的情况，因此最好在读取的时候都进行非空判断一下，防止出现 nan 值。

2.2 提取特征

我一共提取了 7 个特征，下面我会解释为什么要提取这些特征。

从直观上来讲，要判断一篇论文是否属于某个作者可以从以下几个方面考虑。

1. 作者发过的论文数量。

对于一个领域的研究者们而言，很显然，如果其它数据我们都不知道的情况下，那么给定一个领域的某篇论文，按照极大似然估计的思想，直接找这个领域论文数量最多的作者总没错。才进入这个领域的科研工作者论文数量肯定较少，是他的论文的概率偏低。

2. 该篇论文的作者总数。

这是关于科研工作者的个人习惯或领域习惯。某些科研人员爱好科研合作，因此一篇论文可能会有很多作者，但相反有的论文作者就比较少。如果一个作者之前所发表的论文都是有很多作者，现在给定一篇论文仅

仅一，两个作者，那么可以怀疑这就不是他的论文。

3. 作者在该会议或期刊上的论文总数。

每个研究领域偏向的会议或期刊都不太相同，都有侧重点。如计算机视觉科研人员肯定更关注 cvpr 而不是 acl，因此给定一篇 acl 论文和一个计算机视觉方向作者，可以显然地遇见该作者几乎不会在此刊上发表论文。而且对于具体的科研人员来讲，他也会有着自己偏好，如高水平科研人员可能更偏向于投递高水平会议或期刊，而不屑于投递低水平会议或期刊。而初入该领域的科研人员可能会尝试从一些水平一般的会议或期刊入手。因此，在某一个会议或期刊上的论文总数是一个辨别论文和作者之间关系的很重要特征。

4. 作者与该论文的其它作者的合作总次数。

科研工作者们很擅长于合作，这样也能激起思维的碰撞，因此，一篇论文往往有很多作者。这也为我们辨别论文和作者之间的关系提供了一个切入点，如果该作者从来未与该篇论文的其它作者合作过，那么很有可能搞错了，出现了同名的作者。

5. 计数 paperAuthor 中<authorid, paperid>的数目

PaperAuthor 中是带有噪声的 authorid-paperid 对，那么如果给定一个 authorid-paperid，它在 paperAuthor 中出现的次数越多说明 paperAuthor 中的数据应该是越有可信力的，换句话说，出现的次数越多，就应该越能认为该 paper 是该作者所著。

6. 论文所发表的时间

论文所发表的时间是一个重要的因素，如一个科研人员可能已经年暮，那么他的黄金科研时期可能是在 30-40 年前。现在给定一篇近年来发表的论文，那么我们可以大概率认为这不是他的论文。

7. 计算作者对应的信息与 PaperAuthor 文件中 edit distance（编辑距离）。

对于给定作者，我们可以通过查询 author 实体得到该作者的真实信息（包括 Name 和 Affiliation）,然后通过<authored,paperid>对去查询 PaperAuthor 文件所记录的 Name 和 Afiiliation，在这两个信息之间，可以用一个距离来衡量其差异，由于都是字符串，这里选择用编辑距离。如果编辑距离越小，说明 author 和 paperAuthor 中作者信息的一致度就越高，paperAuthor 的可信度就越高。

上述 7 个特征均可在代码文件中的 construct_features 函数找到，并附以注释说明。

3 模型选择

在提取到上述特征后，我选择了两种模型，一种是 Xgboost，一种是 Randomforest,这两种都是在数据挖掘领域常用的算法。

对于 Xgboost 来说，xgboost 中的基学习器除了可以是 CART（gbtree）也可以是线性分类器（gblinear），它有以下优点：

1. 正则化项防止过拟合。
2. xgboost 不仅使用到了一阶导数，还使用二阶导数，损失更精确，还可以自定义损失。
3. XGBoost 的并行优化，XGBoost 的并行是在特征粒度上的。
4. 考虑了训练数据为稀疏值的情况，可以为缺失值或者指定的值指定分支的默认方向，这能大大提升算法的效率。
5. 支持列抽样，不仅能降低过拟合，还能减少计算。

对于 Randomforest 来说，其有以下优点：

1. 训练可以高度并行化，可以有效运行在大数据集上。
2. 由于对决策树候选划分属性的采样，这样在样本特征维度较高的时候，仍然可以高效的训练模型。
3. 由于有了样本和属性的采样，最终训练出来的模型泛化能力强。
4. 可以输出各特征对预测目标的重要性。
5. 对部分特征的缺失容忍度高。
6. 袋外数据可用作验证集来检验模型的有效性，不用额外划分数据集。

我对比了两种模型的实验结果，在下一节中予以说明。

4　实验结果

4.1 Xgboost

对于 Xgboost 模型,我的最优参数如下所示：

```
self.params = {
        'booster': 'gbtree',
        'objective': 'multi:softmax',
        'num_class': 2,
        'max_depth': 10,
        'eta': 0.1,
        'silent': 1,
        'gamma': 0.1,
        'min_child_weight': 2,
        'subsample': 1,
        'colsample_bytree': 1,
        'colsample_bylevel': 1,
        'lambda': 1,
        'alpha': 0,
        'nthread': -1,
        'eval_metric': 'merror',
        'seed': 40
    }
```

调整到最好的参数后，运行模型，得到其特征的重要性关系图如图 1 所示

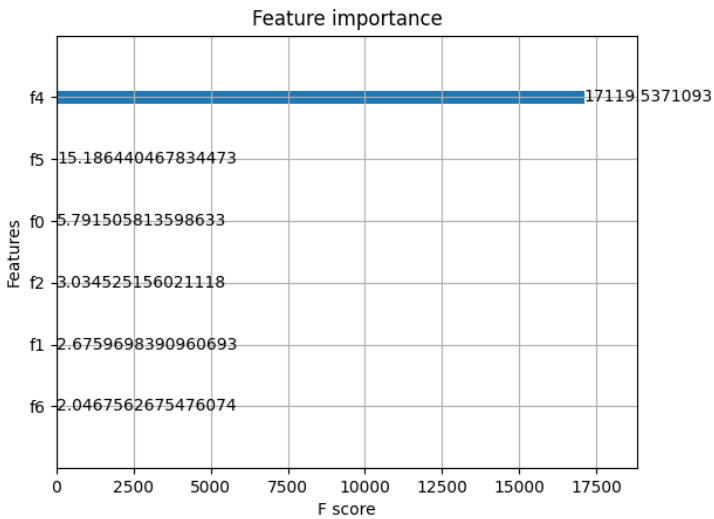

图 1 Xgboost 特征重要性

从图 1 中可以看出来，第五个特征极为重要，即 paperAuthor 中<authorid, paperid>的数目，其重要性程度显著超过了其余所有特征之和。然后调用 xgboost 模型的预测函数，得到其得分为：

Track1	0.96064
Track2	0.93107

4.2 RandomForest

对于 RandomForestClassifier，我的最优参数为 (n_estimators= 200,max_depth=10, random_state=47)。

然后训练模型，最终得到其特征重要性得分如下：

[0.03442095 0.02227491 0.02339308 0.00124159 0.80462333 0.10349326 0.01179447]

从上述数组也可以看出来，第 5 个特征是最重要的，占据了所有特征的 80%。其在测试数据上的得分如下所示：

Track1	0.95887
Track2	0.94214

4.3 结果分析

从实验结果来看，两个模型都显示了第五个特征，也就是 paperAuthor 中的 paperId-authorId 对出现的数目极为重要。然后我尝试对它进行单独分析，发现就一个特征也能达到 95%的正确率，而没加这个特征时，用其它再多特征也不过才 80 多点。我猜想这个特征有效的原因是 PaperAuthor 中的数据已经较为精准或者所含噪声数据较少，或是这道题是人为向文件中添加了噪声数据，该特征恰好将这些噪声给拎了出来。

其次，从模型结果来看，分类器对结果得分的重要性没那么大，其实验结果差异不大。而且 Xgboost 在 Track1 中比 RandomForest 高，而在 Track2 中比 RandomForest 低。

5 小结

本次复杂结构数据挖掘大作业内容很充实，数据很大，光是读取数据，提取特征就需要运行 20 分钟左右。其给出的数据较多，可以提取的特征也特别多，特征提取出来以后可以选择的模型也特别多。我发现在事先你并不能确定那个特征好，那个特征不好，都需要你去尝试。在这个问题中，我感觉分类器并不是影响得分的关键，得分的关键在你特征工程做得如何。

其次开始看到这道题时也想了其它很多方法，如基于知识图谱的分类，初步想法是将实体表示成图中节点，其关系表示成边，而最终需要判别的论文和作者直接在图上找对应的节点然后预测其中有没有论文属于作者的关系即可，但是实际上做的时候发现这样做节点和边太多了，根本无法实施，故否决。还有一种方法是用 NLP 的方法，我开始认为所给论文 title 和该作者以前发表的论文 title 或关键字肯定有一定的相关性，因为科研人员一般是围绕着相关的方向开展科研，因此我想可以将他们输入进 bert，然后取 bert 的输出做分类任务，但是这样会有一个缺点是没有很好利用 PaperAuthor 文件的信息，而且后续其它特征不好扩展，故否决。最后发现提取出来的特征用传统机器学习模型已经能达到 96%的正确率，个人觉得不需要再强行换成深度学习模型，重在提取特征。

致谢 在此,感谢本课程的授课老师黎明老师，黎明课上生动形象，课后精心为大家挑选有质量的作业，再次感谢黎明老师和本课程的助教们!