具体试错和解题思路

摘要

本文主要研究同名消歧冷启动问题,冷启动问题具体是指:对论文同名作者的所有论文，利用论文的信息，如标题，作者，作者机构，摘要，关键词等特征，通过具体方法将论文分配到正确的作者档案中。本文根据待消歧论文中两种类型的特征（语义特征和关系特征），分别运用不同的特征表示方法，得到每篇论文中的两种特征表示向量，结合两种特征表示向量并基于相关的聚类算法和规则匹配将论文划分给具体的作者。最终在验证数据和测试数据上都取得不错的结果。

解题过程

1. 数据分析

通过对训练数据集（train_pub.json，train_author.json)的进一步分析。发现数据存在以下一些待处理的问题：

（1）每篇 paper 都有标题，缺失摘要、关键词、年份、期刊的 paper 数以及两两同时缺失情况存在一定相关性。另外年份信息主要分布在 0~2300 的范围。

（2）在 paper 数据中，整体上英文居多，但仍存在较大一部分数据是中文和日文的。

（3）paper 的作者信息对预测效果具有非常重要的作用。大部分 paper 作者数在 5-12 区间，有很少一部分作者书多于 20 人。

（4）作者名字表达方式多样，比如：

1.'Jianguo Wu', 2. 'Jianguo\xa0Wu', 3. 'WU Jianguo', 4.'Wu Jianguo', 5.'Wu jianguo', 6.'jianguo wu', 7.'wu jianguo'

针对以上问题，对数据进行了预处理，首先把字母小写化，去除各种非字母的符号，接着去掉多余的空格，若文本需要分词，则在分词后去掉停用词，和长度小于 2 的词。

利用构建作者词典的方法，处理作者名字表达方式多样的问题，具体操作是对于每个名字先全部变为小写,然后检查它和它前后调换后的名字是否在作者名词典中,如果某一个在，则把该名字保存成该形式，若都不在，将它加入作者词典。这样作者词典里不会同时存在 wu jianguo 和 jianguo wu 这两个名字。

（以上预处理操作，体现在代码 utils.py 保存论文关系 save_relation 函数中）

2. 特征分析

首先，每篇论文的特征包含 title,abstract,author,venue,organization,year,keyword，这些特征可以概括为两种类型：

（1）语义特征：titile,abstract,keywords，这些文本特征可以使用语义表征学习模型转化为语义向量。

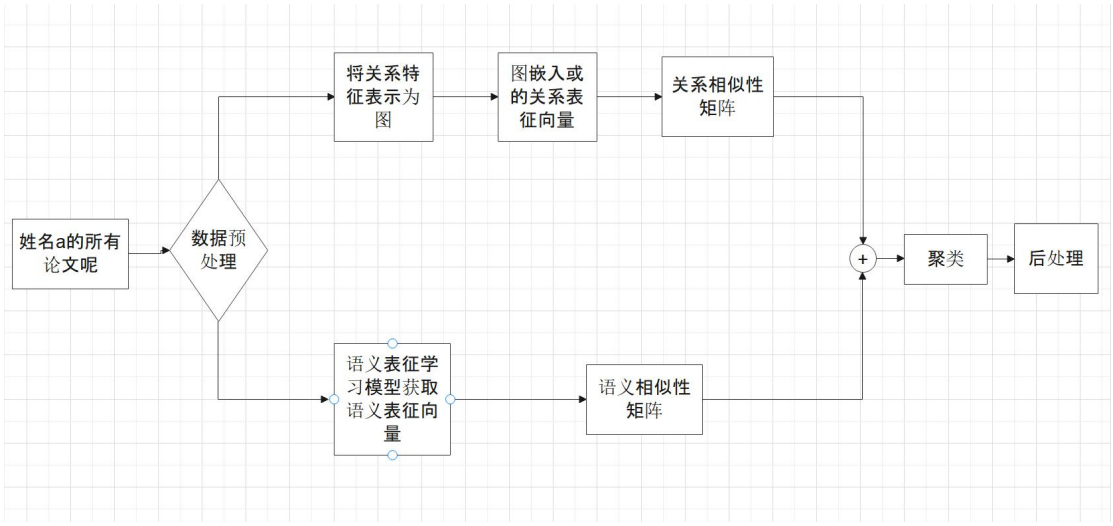
（2）离散特征：离散特征本身的文本信息没有太大价值，例如author,year,venue,organization，但是这些特征能转化为论文之间的关系，最后表征为关系向量。

其次 organization 中包含很多地名和组织名，是一个用来划分不同作者的强影响因素，venue，organization 也具有弱语义关系，既可以加入语义向量的表征中也可以加入关系向量表征中。

3. 整体思路

根据数据分析和特征分析结果：离散特征可以转化为两片论文的关系，比如 A，B 论文有共同作者，共同发表会议等，可以构图表示此关系，并使用图嵌入方式得到论文之间的关系表征向量，对于语义特征就可以用词嵌入方法获得词向量。因此对于每一个需要消歧的名字，可以结合其语义表征向量和关系表征向量，再计算两两相似度得到各论文之间的相似度矩阵，根据此相似度举证再结合聚类算法，据称不同的簇，相同簇中论文之间相似度较高。每个簇就代表一个具体的作者。

整体可概括为以下流程图：



4. 算法设计与实现

4.1 语义表示

语义特征：title,abstract,keyword（venue，organization 弱语义特征，实验证明加入这两个弱语义特征，效果有提升）

使用 paper 语料训练的 OAGBert 预训练语言模型，获取每篇文章的语义表征向量。对于同一人名下的所有 paper，先获语义向量表示然后计算向量之间的 cosine 相似度，得到语义相似度矩阵。表示每篇 paper 与其余 paper 之间的语义特征的相似性关系。

使用相似性矩阵，一方面是为了结合语义和关系特征（语义、关系相似性矩阵可以直接结合，而语义向量表示和关系向量表示不好结合），另一方面，相似性矩阵能体现论文之间的差异，并适用于后面的聚类算法。

具体试错：

1. **使用原 BertBase 进行词向量、句向量表示，效果比 word2vec 训练的词向量好，但是由于训练语料的关系，效果弱于 OAGBert。（官方提供的 cogdl.oagbert 接口，专门适用于此消歧任务，以此训练的词向量质量较高）**

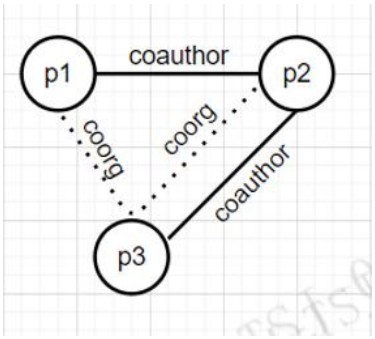
2. **对比学习。用 OAGBert 做了对比学习，最终训练得到对比 OAGBert，但是对比学习过程使模型发生了一些变化，无法使用 cogdl.oagbert 获取词向量，最终词向量质量不如原 OAGBert。（由于对比学习没达到效果，代码中没有体现这一步）**

4.2 关系表示

（1）将关系表示成图.

论文之间的关系表示：（author，year，venue，organization 等离散特征，后面实验证明 year，venue 两个特征会降低模型效果，原因在于相比作者和组织，year，venue 区分度不大）

使用异质网络表示 paper 之间的关系，主要是共同作者关系和共同组织关系[1]。（异质网络是多种节点类型或多种边的类型的特殊图表示）具体形式如下：



（2）图嵌入

根据以上的异质网络，对于每篇论文我们使用基于原路径的随机游走生成由论文 id 组成的路径，把这些路径作为 word2vec 模型的输入，训练得到该论文的关系向量表示，该向量表示与此论文有相同作者和相同组织关系的论文。

基于元路径的随机游走：

定义元路径：p1 > coAuthor >p2 > coOrg > p3，基于元路径随机游走是指：随机游走不是完全随机，而是以定义好的元路径进行[1]，即 p1 随机选择 coauthor 类型的边走到下一个节点 p2，再随机选择

coOrg 类型的边走到下一个节点。

最终对于每篇论文得到如下路径集：

1.p1 > p2 > p5 > p8 ...

2.p1 > p8 > p2 > p6 ...

3.p1 > p8 > p3 > p5 ...

4.p2 > p1 > p5 > p7

……

对于图中的孤立点（即没有边与之相连的节点），这个节点对应的论文 id 不存在于路径集中，无法得到其节点关系表征向量，我们将这些论文 id 加到离群论文集中，后续再做处理。

（3）关系相似性

word2vec 训练得到路径集的向量表示，此为论文关系向量表示，同理计算向量之间的 cosine 相似度，得到语义相似性矩阵，表示每篇论文与其余论文之间的关系相似性。

具体试错：

1. 尝试加入更多特征表示论文间的关系，如 venue 和 year，结果使分数降低了，原因在于：相比作者和组织，venue,year 区分度太小。（相同作者的两篇文章相似性更高，但是相同年份或发表会议的两篇文章相似度较低）

2. 尝试 node2vec 获取图嵌入，但是针对本任务，各节点之间有两类型边连接，每个类型又有多条边，使得 node2vec 计算量太大，导致运行失败。

3. 针对上述问题，增加限制减少边的数量（两篇论文之间有共同作者+共同机构才有边相连），增加限制之后效果不如随机游走。

4.3 聚类

得到两个层面的相似性矩阵之后，使用加权的方式结合两个相似性矩阵，得到最终的相似性矩阵，根据实验最终权重比为 1：1 时效果较好。

聚类算法：DBSCAN

DBSCAN 是基于密度的聚类算法，优点是：自动划分聚类簇数，用距离度量来表示样本之间的远近关系，较适合于本任务中的相似度聚类，该算法参数较少，原理简洁。

最终聚类结果中会生成有份已经划分好的论文簇和 label 为-1 的离群点，将这些离群点加入到离群集中，最后进行后处理。

具体试错：

尝试不同的聚类算法，层次聚类，k_means 聚类等，SMO 聚类等，最终 DBSCAN 聚类算法效果最好。DBSCAN 算法最本任务的优势如上所述。

4.4 后处理

上述聚类过程得到已经划分好的簇。此外还是离群论文集需要处理。我们通过观察这些离群的 case，设置一些规则将其划分到相应的簇中。离群论文产生的原因是这部分论文特征不够明显,或者论文本身属于一个论文数较少的作者,上述的表示向量学习方法效果不好，尝试直接进行匹配。

具体做法：首先对于离群论文集中的每一篇论文，比较它与每个已经划分好的簇中的论文，得到跟它匹配相似度最高的论文，如果他们之间匹配相似度不小于阈值，则将离群论文划到最高匹配相似度论文对应的簇中。

做完上步，对于离群论文集中的每一篇论文，再比较它与离群论文集中其他每个论文匹配相似度，如果两者的相似度不小于阈值 α，则把后者分配到前者所在的簇中；否则不变。

匹配相似度定义如下：

1.s1 = （pi 和 pj 的共同作者数）×1.5

2.s2 = pi.pj 的 venue 之间的 tanimoto 距离

3.s3 = pi.pj 的待消歧名的 organization 之间的 tanimoto 距离

4.s4 = pi 和 pj 中 title 的共词数

后处理之后就得到最终对于某待消歧名字的消歧结果。

参考文献

[1] Dong Y , Chawla N V , Swami A . metapath2vec: Scalable Representation Learning for Heterogeneous Networks[C]// the 23rd ACM SIGKDD International Conference. ACM, 2017.