**项目一**

**同名消歧**

**1.简介**

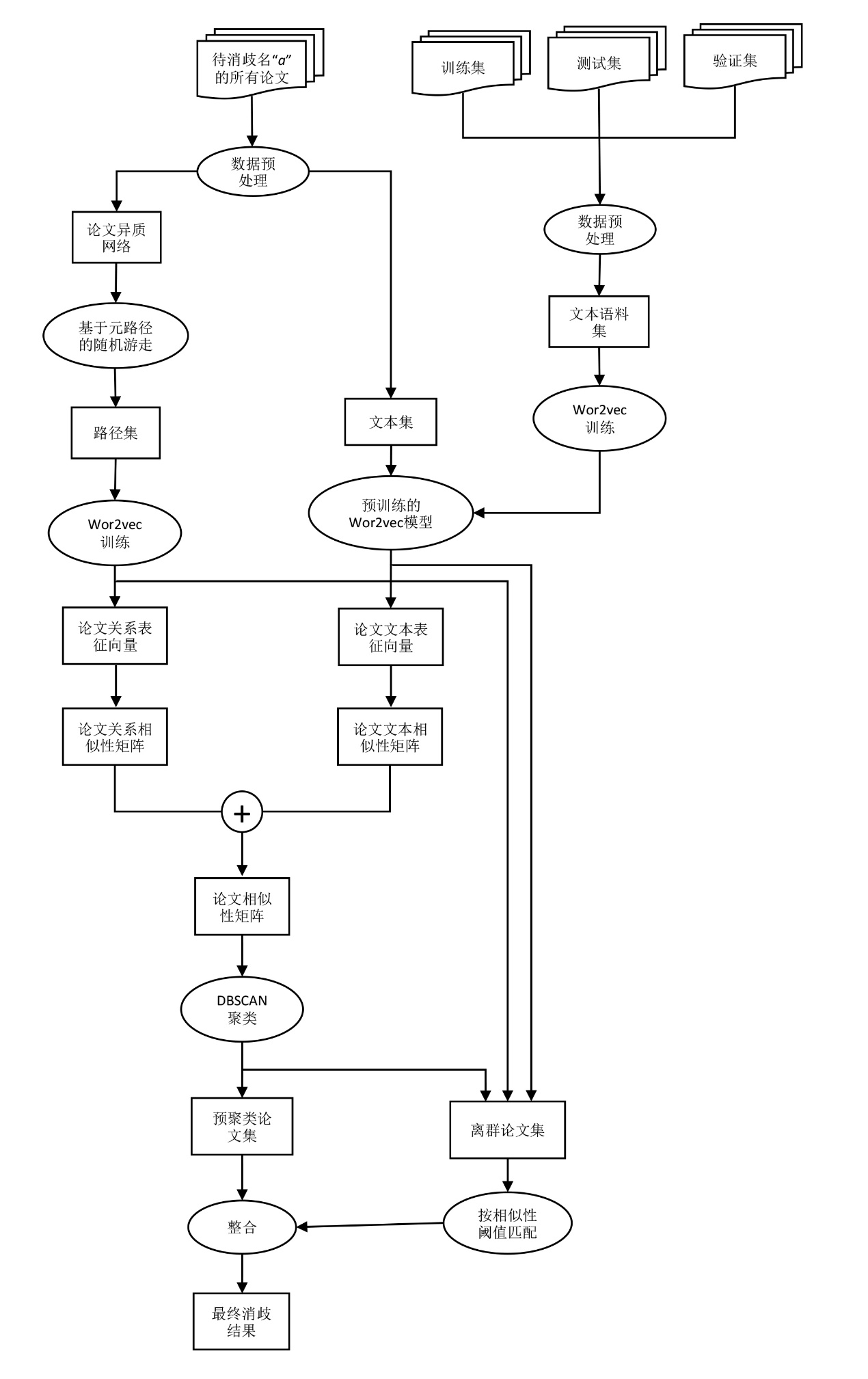
在许多应用中，同名消歧 (Name Disambiguation - aiming at disambiguating WhoIsWho) 一直被视为一个具有挑战性的问题，如科学文献管理、人物搜索、社交网络分析等，同时，随着科学文献的大量增长，使得该问题的解决变得愈加困难与紧迫。尽管同名消歧已经在学术界和工业界被大量研究，但由于数据的杂乱以及同名情景十分复杂，导致该问题仍未能很好解决。

收录各种论文的线上学术搜索系统(例Google Scholar, Dblp和AMiner等)已经成为目前全球学术界重要且最受欢迎的学术交流以及论文搜索平台。然而由于论文分配算法的局限性，现有的学术系统内部存在着大量的论文分配错误；此外，每天都会有大量新论文进入系统。故如何准确快速的将论文分配到系统中已有作者档案以及维护作者档案的一致性，是现有的线上学术系统亟待解决的难题。

本项目前期基于的数据集由OAG-WhoIsWho提供，包含AMiner中大量的作者档案和论文，给出一定的文件包括train\_author.json和train\_pub.json。前者文件中组织成一个字典，dic1的键（key）是作者姓名。 dic1的值（value）是表示同名作者集合的字典（记为dic2）。 dic2的键（key）是作者ID， dic2的值（value）是该作者的论文ID列表；后者文件是关于论文信息的元数据，其键（key）是论文ID，其值是相应的论文信息。

**2.整体思路**

对于某个需要消歧的名字，根据其所有论文之间的关系信息和论文文本信息求出论文表征向量，进而求出论文之间的两两相似度，得到论文相似性矩阵，使用聚类算法和基于相似度阈值的匹配方法将论文划分为不同的簇，则每个簇代表一个特定作者的论文集。

我们的方法的整体框架如下：

**3.研究方法总结**

我们把该问题看作是对一个论文集的聚类任务，且不指定聚类簇个数，即K值。

首先，分析每篇论文的特征，论文的特征包含title, abstract, author, venue, organization, year, keyword. 我们把这些特征划分为两种类型，一种是语义特征，一种是离散特征。

语义特征指的是可以具有语义信息的文本特征，例如title, abstract, keyword，这些文本可以使用语义表征学习模型如word2vec等转化为文本语义向量。在后续的实验中，我们认为venue, organization, year也具有弱语义信息。

离散特征指的是本身的文本信息没有很大的价值，例如author，作者名称本身的语义并没有作用，一个作者只有在两篇文章中同时出现时才有作用，这表示两篇论文有一个共同作者，则它们之间的相似性较大。因此我们把这类特征成为离散特征，只能用来转换为论文间的关系。在后续的实验中，我们发现organization也属于离散特征，因为其中的有些词如地名可以用来搭建论文之间的关系。

具体实施上，我们定义author, organization为离散特征，定义title, venue, organization, year, keyword为语义特征。

基于以上两种特征的认识，我们的思路是使用语义特征学习论文的文本表征向量，利用离散特征来构建论文之间的关系，如有共同作者关系，有机构相似性关系，以此构建论文网络，通过网络嵌入学习方法来学习论文的关系表征向量。再使用聚类算法对论文进行聚类。

3.1论文关系特征提取

此部分的作用是学习到每个论文的关系表征向量。可以看作是先搭建论文异质网络，然后利用网络嵌入的方法学习到每个论文节点的表示向量。本部分用到的特征有author和organization。

网络嵌入（Network Embedding）模型尤其是异质网络嵌入模型已经有了很多研究成果，我们使用的网络嵌入模型主要参考DeepWalk和Metapath2vec。

3.2论文文本特征提取

此部分的作用是生成论文语义表征向量。

首先将训练集，验证集和测试集中的文本提取到一个文本语料集中，具体来说，对于每一篇论文，我们提取了其每个作者的organization，论文的title，论文的abstract，论文的venue这些文本，并按行存储到一个txt文件中。

对于每一篇论文，我们生成它对应的语义表征向量过程如下：将该论文的title, venue, organization, year, keywords 以空格间隔，合成同一段文本，对这个文本进行预处理，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，以空格分词，去掉停用词和长度小于3的词。将这些词都通过上段训练好的word2vec模型生成词向量并求均值，最终得到论文的语义表征向量。

对每个待消歧的名字，得到其所有论文的语义表征向量，当有的论文的所有词都不存在于word2vec模型中时，将其语义表征向量置为全0，并保存到离群论文集中，后续再做处理。最后求得论文两两间的余弦相似度，得到论文语义相似性矩阵。

3.3聚类方法总结

3.3.1利用DBSCAN聚类

论文关系相似性矩阵和论文语义相似性矩阵，首先我们把这两个矩阵相加求均值。得到最终的论文相似性矩阵。我们考虑过加权相加，以测试哪个相似性矩阵具有更高的比重，最终根据训练集的结果得出这两个矩阵的权值比应设置为1:1。

根据这个论文相似性矩阵，将这个矩阵输入DBSCAN中，得到聚类结果。

3.3.2预测簇大小及层次聚类

对于候选集的聚类任务，我们尝试了另一种方法。即通过训练一个模型来预测簇的数目大小再使用层次聚类进行聚类任务。

1. 首先构造伪训练数据集。我们将为每次训练从所有簇中随机选取不同数目的簇，这些簇可以来自所有在训练集中的候选集。选定的数目就是训练的标签，这个数目将在设定的min和max之间，通常min=1，max我们这里设定为300。然后我们从这些簇中再随机选出300篇文章，用来作为信息提取对象。
2. 然后我们将这300篇文章的文本特征向量输入到BiLSTM中，输出一个预测的簇的数目。

然后在训练结束后，应用到我们最后的聚类中时，我们直接从某一个候选集中随机选取300篇文章的特征向量，送入训练好的模型中，预测出簇的数目，就可以调用层次聚类的方法了。

3.4离群论文匹配

我们目前得到了一些已经划分好的簇，里边包含着已经聚类好的论文，我们把这些论文合起来的集合叫做预聚类论文集，还有一个离群论文集，且这两个集合不相交。这一步的操作是将这些离群论文集里的论文用阈值匹配的方法重新分配给已经聚类好的簇或者新的簇中。