# 做笔记

5.1

-编写基于百度百科中的label抽取上下位关系侯选集的爬虫

5.2

-考虑多义词的label抽取

-考虑重复的上下位词对的去重

5.3

-基于百科label的抽取执行完毕（2days），共抽取到221523个词对(WordPairs-label-v1.txt)，其中，上下位均在词典中的有143004个(WordPairs-label-v2.txt)

-基于百科文本共现度对抽取到的候选集进行筛选，对于两个词a、b，初步定义筛选值F为： F=2\*count(a,b)/min(count(a),count(b))，其中count(word)为词在所有语料中出现的次数，但是筛选的阈值有待商榷。初步选取阈值为0.08在数据集中进行了测试，在测试了10000个词对后发现:

（1）在存在冷门词的场景下表现较好，但是对均为热门词的词对有许多误筛。

（2）下位词是上位词的一部分时，所有的F值均为1.0，无法进行判断。

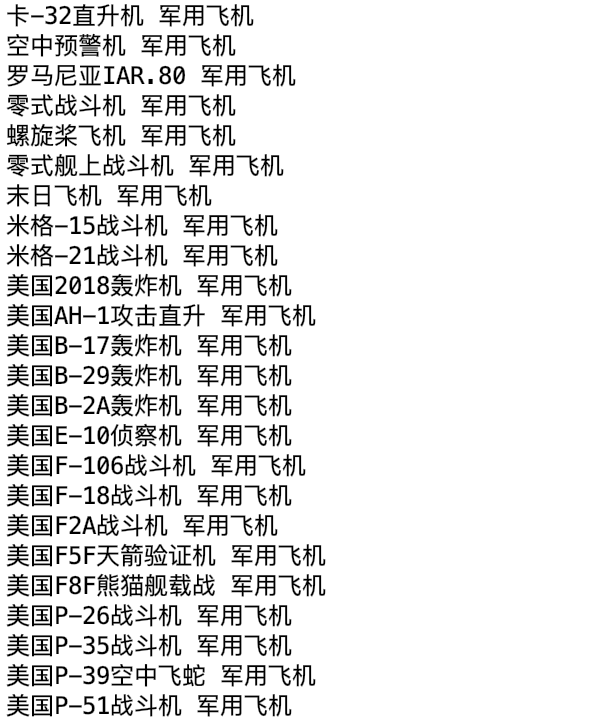
（3）不适合对从结构化文本中抽取的词对进行筛选，因为他们本身已经具有较高的关联性。只适用于筛掉明显有错的。

-计划后期重新设计F值，并根据词的热度分情况处理

5.4

-基于互动百科的分类栏目进行了抽取（4hours），共抽取到1455452个词对(WordPairs-fenlei-v1.txt)，其中，上下位均在词典中的有30134个(WordPairs-fenlei-v2.txt)

-经过对比发现，基于互动百科词条分类抽取出的上下位关系准确度和召回度均表现良好，不足之处在于很多下位词是领域实体名词，不在词典内，如：



5.5

-在百科中的所有句子（9906487句）中，抽取出包含有“是一”的句子（233301句，sentenceWithIsA.txt）

-用jieba分词对上述句子进行处理，得到分词后结果，并进行了词性标注，作为后续基于pattern方法的基础

5.6

-编写了基于有道的翻译爬虫，作为后续基于翻译系统方法的基础

5.7

-考虑到5.3日设计的F值存在许多不足，设计了另一种基于搜索引擎（百度）的筛选方法，对于两个词a、b，初步定义筛选值W为： W=2\*hit(a+b)/hit(a)+hit(b)，其中hit(word)为词在搜索引擎中查询得到的结果数目

-利用百度的高级搜索指令：intitle:xxx，可以将搜索内容限定在网页标题中，发现这样进行搜索，得到的W值往往筛选效果更好

5.8

-用jieba分词对词典中的词进行了分词，发现其中存在后缀的词有16770个(WordPairs-suffix.txt)

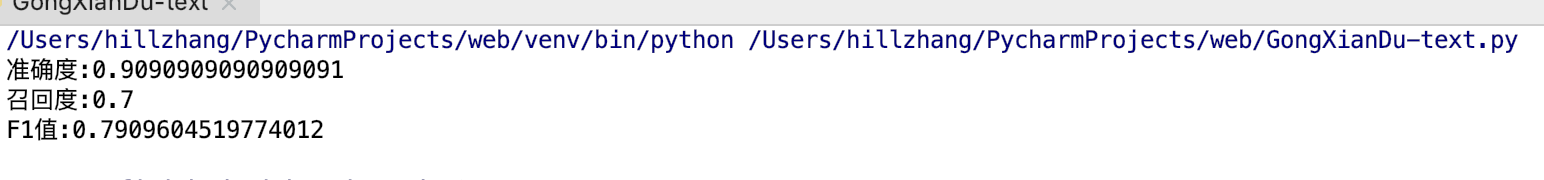
-测试了10000个词对后,发现基于信息检索的筛选方法，缺陷还是很多，比如：

（1）正确性低，因为热门词在信息中出现的频率很高，会相应地提高出错的概率，同时，搜索引擎的搜索结果在不断变化中，得到的值不具有时效性。

（2）覆盖性差，无法判断相似词，不能由一个词对推导更多的词对 ，很多冷门词在搜索引擎中的得分很低

5.9

-选取了100个上下位正例，和100个上下位反例，利用F值和W值相加计算，取阈值为0.01，来筛选结果，最终得到：



-结论：我认为这两种方法得到的模型，在初步筛选时是比较准确的，对一些误差比较大的词还是可以很好地判断，但是对于误差较小的词（比如从百科、词典等结构化文本中爬到的结果）判断的还是不够准确，需要其他的方法。

\*本次测试，正例与反例之间的差异太大，不具有代表性，下次要换更模糊的测试集

下周计划：

-继续在网上找上下位关系的资源，爬下来

-设计基于词向量的评估模型（利用word2vec实现）

-设计基于字面相似度的评估模型（以字为基本单位，考虑相同字的位置，给出权重）

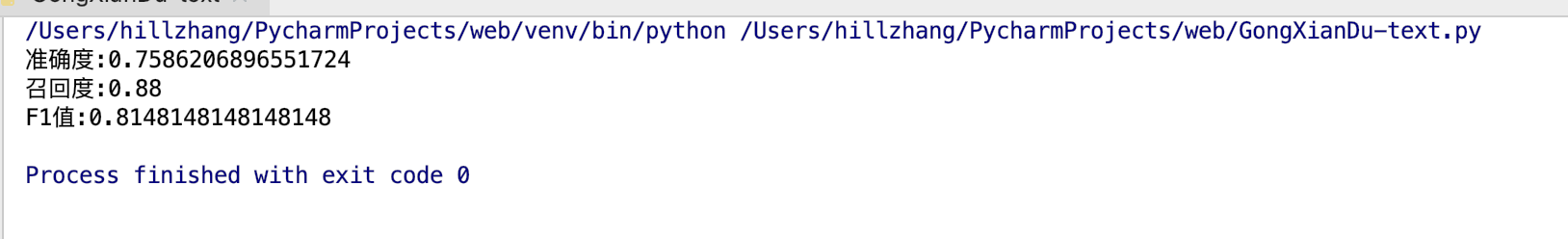
-将字面相似度、文本贡献度、搜索引擎共现度、词向量相似度四种方法结合起来，加权取新的筛选值，测试更具有代表性的数据，看看效果如何

一点思考：

在为pattern方法准备语料时，我发现：当在句子中，符合pattern的两个词中存在一个词被实体名词或代词修饰（即存在范围）时，抽取出的词对很可能不是上下位词对，例如：麝香猫的粪便是一种名贵的咖啡。抽取出来的词对（粪便，咖啡）就很明显不是上下位关系。因为在这种语境下，上下位关系只有在一定范围内成立，不具有普适性。我认为，基于这一原理，可以在抽取词对时减少错误，降低人工标注的负担。

5.10

-更改了W值的计算公式，变更为W=2\*hit(a+b)/min(hit(a),hit(b))，与之前的值（5.9）进行了对比：



-结论：更改后的W值拥有更好的召回度，我想这是因为在存在冷门词的场景中，更改后的公式表现的更好，但与此同时，准确度也大幅下降。总体而言，F1值还是有微小上升的。

-开始基于pattern方法的抽取，初期打算只是用以下四种模板，bootstrapping先放一放：

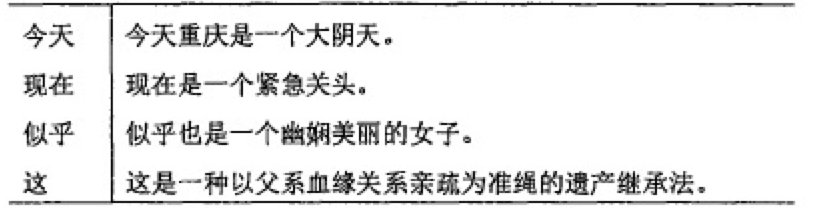


-基于“isA”模式编写了脚本，在去重后，我在233302句话中抽取了89635个词对（WordPairs-isA-v1.txt），其中，上下位均在词典中的有38642个词对（WordPairs-isA-v2.txt）。对结果进行评估，我发现，其中的谬误也很多，主要是因为

（1）中文的句法较为复杂，要想准确的抽取，需要排除修饰语、可过滤词。比如，（他的弟弟是一个佛教徒），我提取的就是（弟弟，佛教徒），但实际上不是这样。再比如，（这个人是一个越南人），我提取出的词对为（人，越南人），与实际相反，这也是因为停用词的影响。

（2）存在可剥离短语的干扰。比如（苏州在这种概念下是一个艺术城市），由于我的算法只考虑到了“的”，所以将“概念”看作是一个下位名词处理，得到的词对（概念，城市）也是有问题的。

5.11

-针对上周的思考，我决定在基于pattern的方法中，尝试建立过滤词表，将含有过滤词的句子过滤掉，比如：



-在采取过滤词典后，去重后，我得到了52216个词对（WordPairs-isA-v3.txt），其中，上下位均在词典中的有26939个词对（WordPairs-isA-v4.txt）。我发现准确率大大提高。。

-采用了剥离词表，也进一步提升了准确度。

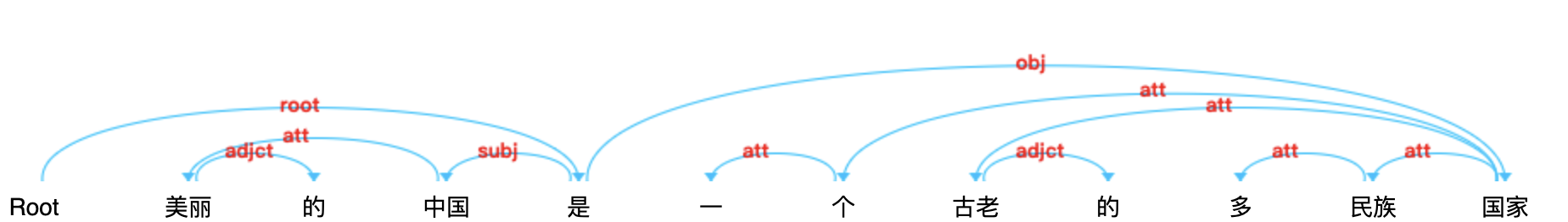
-在基于pattern的过程中，我发现分词也是影响正确性的重要一项，因为我的算法是为了匹配名词，所以对专有名词、领域实体名词的划分有较高的要求。例如（对安德森而言 ， 民族是一种人造体），因为词典中没有人造体这个词，所以最终的结果就变成了（人类，体）。这也会带来很多错误，分类模型也很难甄别，所以可能需要人工标注。

5.12

-目前，学术界对于上下位关系的pattern抽取，全部采用基于词性标注和字符串处理（“的”、剥离词、停用词）的方法。这种方法，我在这几天的应用中发现了许多弊端。对此，我重新设计了一种基于句法分析的词语上下位关系pattern抽取方法。

-具体思路是在句子中寻找“isA”的模式。通过句法分析得到“是”的主语（subj）和宾语（obj）作为下位和上位。

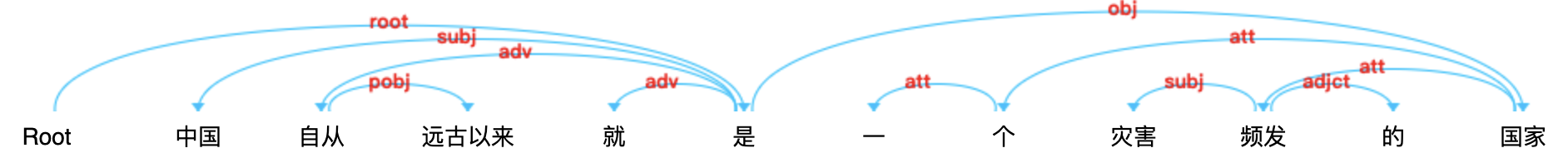
例如：



抽取的结果为：（中国，国家）

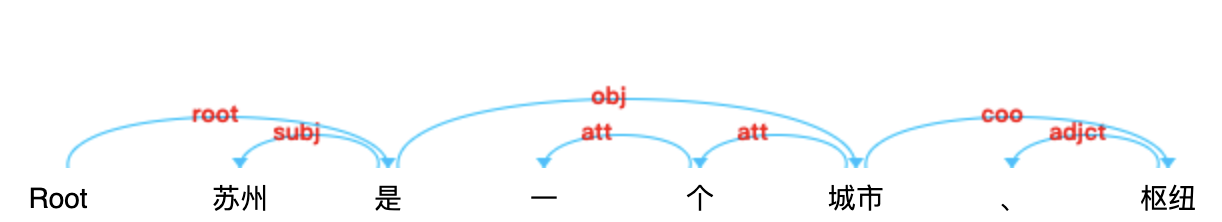
\*本方法的优点：

（1）无需花费大量时间建立剥离词典，即可减少可剥离短语的影响，例如：（中国自从远古以来就是一个多民族的国家），直接采取依存句法，得到的结果为：



（中国，国家）

（2）通过提取同主语（sasubj）的弧和并列（coo）的弧，可以解决之前难以解决的多上位抽取问题。例如：（苏州是一个城市、枢纽）



采用依存句法，通过对coo的抽取，可以提取到（苏州，城市），（苏州，枢纽）

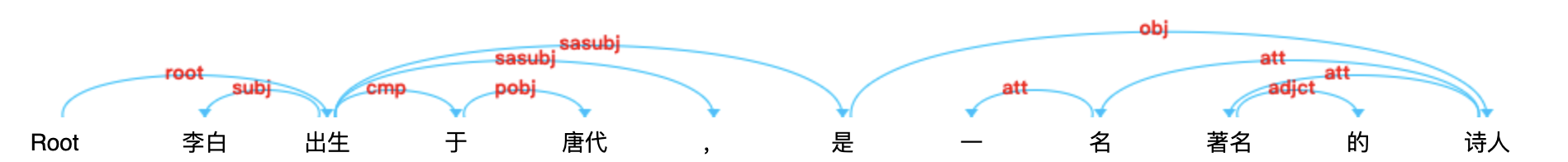
而采取之前的方法，得到的结果为：



（苏州，枢纽），无法得到完整的词对

（3）对错误使用标点的句子更好的判断，比如（中医学是一种起源于中国 ，以古代中国医学实践为主体的传统医学），这里的逗号本应为顿号。采取之前的方法（按逗号分句处理），得到的结果为：（中医学，中国）。而采取依存句法，则可以得到正确结果（中医学，医学）

（4）通过对同主语弧的分析，可以跨句抽取词对。例如（李白出生于唐代，是一名著名的诗人），采用依存句法：



通过分析pattern的主语和宾语，可以跨句抽取到（李白，诗人）。而采用原先的方法是做不到的，因为它没有考虑跨句的联系。

5.13

-花费了两天的时间，编写完了基于依存句法树的pattern抽取脚本（v1）。先在100000个符合IsA的pattern的句子中进行测试，取依存句法置信度大于0.5，并且加载了过滤词典，最终去重后得到了42881个词对（WordPairs-isA-ycjf-v1.txt）

-目前，本方法也存在了一些问题：

（1）有一些分词错误。

（2）百度百科中也存在一些语病，导致了许多错误分割。

（3）对于多pattern的抽取比较麻烦，后期要设计好pattern模板。目前只能抽取“主谓宾”结构的pattern。

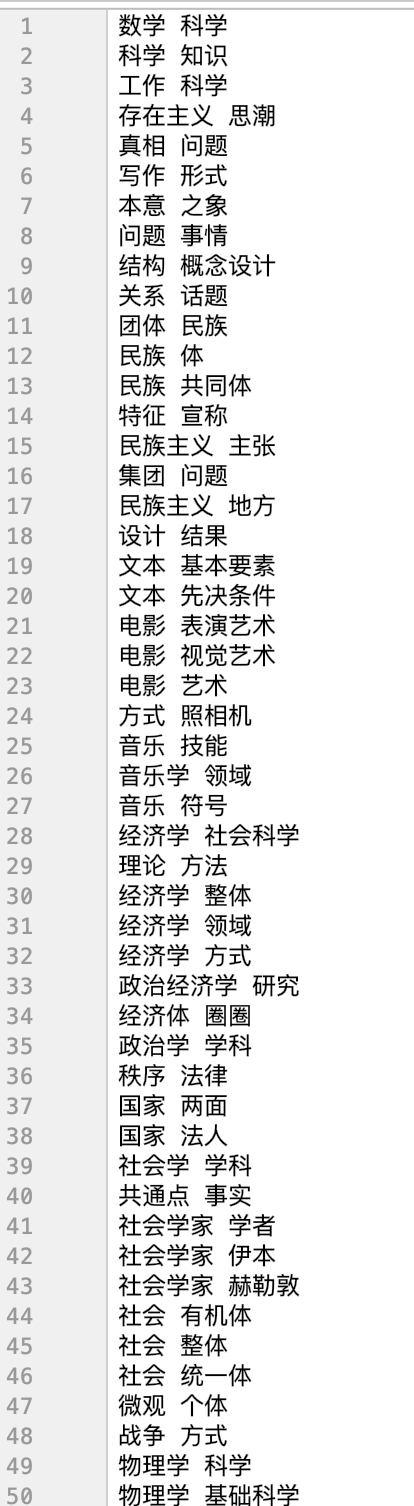
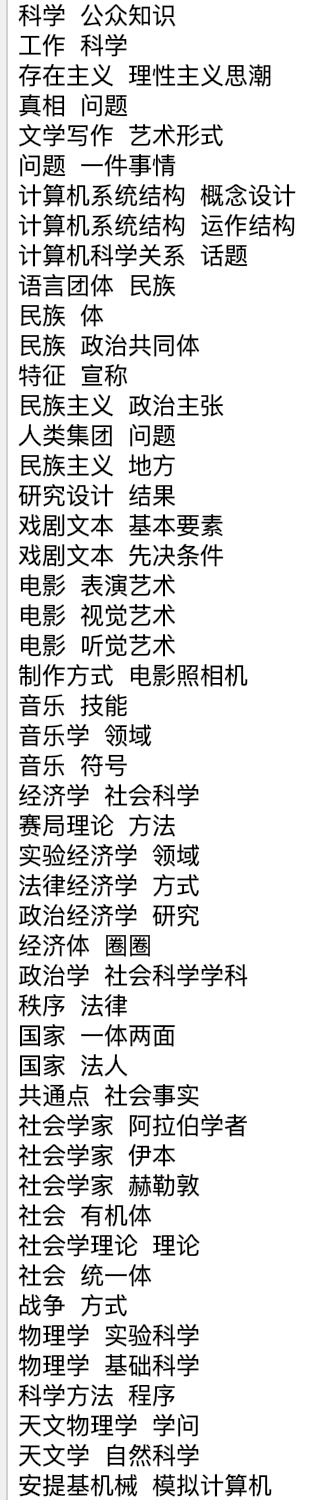
（4）对依存句法分析的准确性有较高要求（演示平台上的demo好像比api调用的更准一些）

5.14

-重写了脚本，解决了分词错误的问题，进一步提高了抽取出的词对的准确率。在100000个测试集中共获得31254个词对（WordPairs-isA-ycjf-v2.txt），无论是召回度和准确度都有了非常大的提升。

5.15

-基于依存句法树重新设计了一种新的获取前缀的方法，使抽取的词更加准确。还加入了HMM-veterbi算法用于识别新词。（WordPairs-isA-ycjf-v4.txt）与上一个版本取前50个结果对比：

**** 

**（前） （后）**

可以发现准确度大大提升，召回度有所下降。

5.16

-为了兼顾词对的准确性、扩大词对范围，我设计了两种获取上下位词对的模式：

（1）标准模式：正常获取词对

（2）后缀模式：取上位词的后缀词，以扩大数据集范围

例子：



具体测试后发现，数据集扩大了约50%

-采取词性、句法树的置信度进行筛选，保留置信度均大于50%的词对，进一步提高了准确度。与此同时，采用手工录入过滤词典的方式保证准确度。（后期可以结合人工标注系统和验证模型大规模构建）

-将所有代码放在了我的github上，地址为：https://github.com/HillZhang1999/nlp