影视领域实体关系抽取实验报告v0.1.0

马敬超

目 录

[一、数据抓取以及预处理 2](#_Toc444712410)

[1.1 数据抓取 2](#_Toc444712411)

[1.2 数据预处理 3](#_Toc444712412)

[二、领域命名实体识别 3](#_Toc444712413)

[2.1 条件随机场进行命名实体识别 3](#_Toc444712414)

[三、语料库的构建 4](#_Toc444712415)

[3.1 影视领域实体关系体系构建 5](#_Toc444712416)

[3.2 影视领域实体关系语料库构建 5](#_Toc444712417)

[四、关系抽取 6](#_Toc444712418)

[4.1 方法1——基于规则的关系抽取 6](#_Toc444712419)

[4.2 方法2——基于无监督的关系抽取 6](#_Toc444712420)

[4.3 方法3——基于有监督的关系抽取 6](#_Toc444712421)

[4.4 方法4——基于半监督的关系抽取 9](#_Toc444712422)

[4.5 方法5——基于CNN的关系抽取 10](#_Toc444712423)

[五、实体关系抽取算法的封装(关系抽取平台开发) 12](#_Toc444712424)

[六、参考文献 13](#_Toc444712425)

## 一、数据抓取以及预处理

主要抓取新浪新闻上面电影一栏的新闻数据,如图1-1所示，数据来源: <http://ent.sina.com.cn/film/>



图1-1 数据来源

### 数据抓取

Step 1:抓取新闻的链接

Step 2:解析新闻页面，获取其标题、日期以及正文，保存为json格式。

完整代码见：<https://github.com/ecnumjc/crawl>

该过程用流程图描述如图1-2数据抓取流程图所示：



图1-2 数据抓取流程图

### 数据预处理

数据预处理主要包括去除多余空格、分句、词性标注（采用jieba分词）等。

完整代码见github: <https://github.com/ecnumjc/data_preprocessing>

## 二、领域命名实体识别

### 2.1 条件随机场进行命名实体识别

使用条件随机场进行命名实体识别实验，主要步骤包括语料标注、特征选择、模板制定、使用CRF学习特征、实体识别测试以及实体识别效果的评测。详细实验步骤如下所述。

#### 2.1.1 语料标注

使用条件随机场进行领域命名实体识别，需要进行大量语料的标注，该项工作在实验室开发的NerTools（[**http://192.168.10.12:8000/**）标注工具上由10](http://192.168.10.12:8000/）标注工具上由10)位实验室同学共同完成,如图2-1所示。考虑到由于标注个体的差异，导致标注数据分布不一致，为了解决这个问题，需要分别考量每位标注者标注数据对训练命名实体识别模型的有效性，另外，还需要人工的介入进行数据的过滤。最后将将标注好的语料按8:2比例分为训练集和测试集。

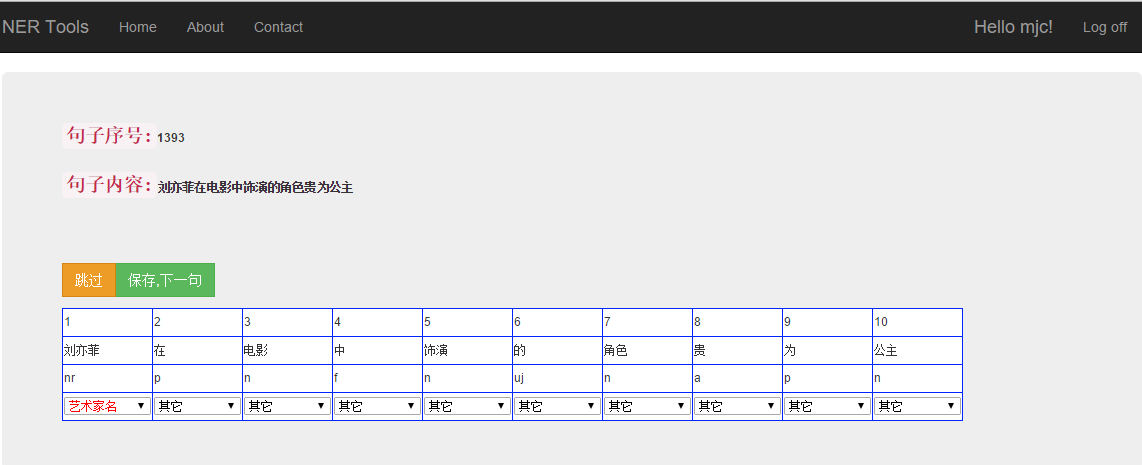


图 2-1 数据标注

#### 2.1.2 特征选择

目前实验用到的特征主要有两类，词性特征和Word2Vec特征。

1. 词性特征

词性特征，即标注语料中词的词性标签；

1. Word2vec特征

根据语义相似度选用与当前标注词最相似的1、3、5、7、9、10个词语作为特征。

#### 2.1.3 模板制定

根据特征选用不同，制定不同的训练模板：

（1）使用词性特征，对应的模板文件为template\_pos

（2）使用最相似的1个词语，对应的模板文件为template\_vec1

（3）使用最相似的3个词语，对应的模板文件为template\_vec3

（4）使用最相似的5个词语，对应的模板文件为template\_vec5

（5）使用最相似的7个词语，对应的模板文件为template\_vec7

（6）使用最相似的9个词语，对应的模板文件为template\_vec9

（7）使用最相似的7个词语，对应的模板文件为template\_vec10

（8）使用所有特征，对应的模板为template\_pos\_vec

#### 2.1.4 使用CRF学习特征

命令：crf\_learn template\_file train\_file model

得到使用CRF进行命名实体识别的模型model文件。

#### 2.1.5 实体识别测试

使用step3训练得到的模型进行测试。

命令：crf\_test -m model\_file test\_file > result\_file

#### 2.1.6 实体识别效果评测

评测脚本conlleval.pl见github

评测脚本使用perl编写，使用之前要保证所用计算机已经安装了perl

评测命令：perl conlleval.pl < output.txt

其中conlleval.pl为评测脚本，output.txt为待评测的文本文件。

## 三、语料库的构建

在使用有监督的学习算法进行实体关系抽取时需要一个大的语料库的支持，另外，无监督的学习算法做关系抽取任务时也需要使用到自己标注的语料。目前，实体关系抽取可用的中文语料资源来自于ACE RDR任务。

参考ACE语料构建经验，构建影视领域实体关系语料，定义语料的格式和标注规范。

### 3.1 影视领域实体关系体系构建

再定义影视领域实体关系体系之前，首先需要确定待抽取的实体的类型，根据实验需求，需要识别的实体类型如表3-1所示。

|  |  |  |
| --- | --- | --- |
| 序号 | 命名实体名 | 命名实体标签 |
| 1 | 艺术家名 | Na |
| 2 | 艺术家别名 | Nab |
| 3 | 影视名 | Nf |
| 4 | 出品公司 | Nc |
| 5 | 日期时间 | Nd |
| 6 | 地点 | Nl |
| 7 | 影视类型 | Nt |

表3-1 实体类型表

实体类型确定后，我们在此基础上结合针对影视领域新闻数据的特点，定义9种影视领域实体关系，如表3-2所示。

|  |  |
| --- | --- |
| 序号 | 实体关系类型 |
| 1 | <导演，影视名> |
| 2 | <演员，影视名> |
| 3 | <编剧，影视名> |
| 4 | <监制，影视名> |
| 5 | <影视名，出品公司> |
| 6 | <影视名，上映时间> |
| 7 | <影视名，上映地点> |
| 8 | <影视名，影视类型> |
| 9 | <艺术家，艺术家别名> |
| 0 | 其他 |

表 3-2 实体关系类型定义

### 3.2 影视领域实体关系语料库构建

在前两章的基础上，可以进行关系类型的标注。该步骤需要大量人工参与。目前已经完成500句实体关系类型的标注。（考虑后续工作，为提高实体关系分类的准确性，需要标注更多数据）

## 四、关系抽取

### 4.1 方法1——基于规则的关系抽取

对于关系抽取，研究者首先利用知识工程解决该问题，通过人工定义抽取规则，从文中抽取与规则匹配的关系实例。例如抽取Is-a这样的关系，可以通过定义规则“Y such as X ((, X)\* (, and|or) X)”、“such Y as X”、“X or other Y”、“X and other Y”、“Y including X”、“Y, especially X”等，其中X和Y就是满足Is-a关系的实体对。像这种知识工程领域的基于规则的关系抽取方法在特定领域可以根据抽取关系的特点设计针对性的规则，而且人工编写规则，准确率高。但是，也避免不了一些缺陷：

（1）人工编写规则，召回率低；

（2）基于手工规则的方法需要领域专家构筑大规模的知识库，这不但需要有专业技能的专家，也需要付出大量劳动；

（3）知识库构建完成后，对于特定的领域的抽取具有较好的准确率，但移植到其它领域十分困难，效果往往较差。

基于规则的实体关系抽取的完整代码见

<https://github.com/ecnumjc/RE/tree/master/RelationExtraction_algorithms>

### 4.2 方法2——基于无监督的关系抽取

在无监督关系抽取方面的代表研究是美国华盛顿大学图灵中心提出的开放式信息抽取（Open Information Extraction，OIE），被称为“新型抽取范式”的一种知识抽取方法[4]。OIE作为一种新型的Web信息抽取，任务的目标在于促进领域无关的知识的抽取应用。另外，由于实体关系抽取已经深入到了语义的层面，传统的基于提取特征和训练分类器的方法遇到了特征提取以及取舍困难的问题。面对这个问题可以借助于核（Kernel）方法。针对实体关系类型体系的局限性，有研究者将研究重点放在从大规模预料出发，通过聚类方法发现实体关系，通过聚类的方法来自动发现实体关系[2]。这类方法的优点是可以不依赖于当前的实体关系类型体系，可方便地进行领域移植。缺点在于聚类的结果依赖于语料的质量，而且很多聚类的结果并没有实际意义，往往还是需要进行人工筛选，其聚类准确性和完备性都有待于进一步研究。

### 4.3 方法3——基于有监督的关系抽取

采用有监督机器学习方法将二元关系的抽取看作是一个分类问题：



S为包含实体对的文本片段，T（S）表示从S上抽取到的特征集合，FR表示关系R的分类器。通常使用的模型有感知机（Perception）、表决感知机（Voted Perception）和支持向量机（Support Vector Machines，SVMs）等。首先通过标注语料库训练得到一个关系抽取模型，然后再利用该模型对自由文本中出现的关系实例进行识别。根据关系实例的表达方式不同，有监督机器学习方法主要分为两种，即基于特征向量的方法和基于核函数的方法。

#### 4.3.1特征工程

特征主要分为四大类：

1. 实体对本身特征；
2. 实体对上下文特征；
3. 句法特征
4. 句子中其他实体特征

#### 4.3.2特征抽取

具体抽取的特征包括以下这些：

F1：ARG1的实体类型，注意F1还可能是代词。

F2：ARG2的实体类型，注意F2还可能是代词。

F3：ARG1和ARG2两个实体类型的组合。

F4：ARG1词本身。

F5：ARG2词本身。

F6：ARG1和ARG2的位置关系，由于实体对不存在包含关系，所以位置关系只有两种：ARG1在前，ARG2在后，F6取真；ARG2在前，ARG1在后，F6取假。

F7：ARG1之前的第一个词。

F8：ARG1之前的第一个词的词性。

F9：ARG1之前的第二个词。

F10：ARG1之前的第二个词的词性。

F11：ARG2之后的第一个词。

F12：ARG2之后的第一个词的词性。

F13：ARG2之后的第二个词。

F14：ARG2之后的第二个词的词性。

F15：ARG1的实体类型和ARG1之前的第一个词的词性的组合。

F16：ARG2的实体类型和ARG2之后的第一个词的词性的组合。

F17：ARG1和ARG2之间是否存在其他词。当两个实体之间没有单词时F17为真，否则F17为假。

F18：当ARG1和ARG2之间只有一个词时，F18为该词，否则为空。

F19：当ARG1和ARG2之间只有一个词时，F19为该词的词性，否则为空。

F20：当ARG1和ARG2之间只有一个词时，F20为ARG1实体类型和该词的词性的组合。

F21：当ARG1和ARG2之间只有一个词时，F21为ARG2实体类型和该词的词性的组合。

F22：当ARG1和ARG2之间至少有两个词时，F22为其中的第一个词。

F23：当ARG1和ARG2之间至少有两个词时，F23为其中的第一个词的词性。

F24：当ARG1和ARG2之间至少有两个词时，F24为其中的第一个词的词性和ARG1实体类型的组合。

F25：当ARG1和ARG2之间至少有两个词时，F25为其中的第一个词的词性和ARG2实体类型的组合。

F26：当ARG1和ARG2之间至少有两个词时，F26为其中的最后一个词。

F27：当ARG1和ARG2之间至少有两个词时，F27为其中的最后一个词的词性。

F28：当ARG1和ARG2之间至少有两个词时，F28为其中的最后一个词的词性和ARG1实体类型的组合。

F29：当ARG1和ARG2之间至少有两个词时，F29为其中的最后一个词的词性和ARG2实体类型的组合。

F30：当ARG1和ARG2之间只有三个词时，取中间的词。

F31：当ARG1和ARG2之间只有三个词时，取中间的词的词性。

F32：当ARG1和ARG2之间至少有四个词时，取ARG1之后的第二个词。

F33：当ARG1和ARG2之间至少有四个词时，取ARG1之后的第二个词的词性。

F34：当ARG1和ARG2之间至少有四个词时，取ARG2之前的第二个词。

F35：当ARG1和ARG2之间至少有四个词时，取ARG2之前的第二个词的词性。

F36：ARG1和ARG2之间靠近ARG1的第一个动词。

F37：ARG1和ARG2之间靠近ARG2的第一个动词。

F38：ARG1和ARG2之间词的个数。

F39：ARG1和ARG2之间其他实体的个数。

F40：ARG1和ARG2之间动词的个数。

F41：ARG1和ARG2之间代词的个数。

F42：在依存句法分析树中，ARG1和ARG2之间的最短依存句法路径（Dependency-tree paths）。

F43：在依存句法分析树中，ARG1和ARG2之间的树距离（Tree distance）,即最短句法路径长度。

特征抽取的完整代码见：

<https://github.com/ecnumjc/feature_extraction>

#### 4.3.3 选用不同分类器

##### 4.3.3.1最大熵分类器

使用最大熵模型需要使用标注训练集进行训练，训练集和测试集格式如图4-1和图4-2所示。

其中，第一列数字代表关系的类别，如表3-2所述。

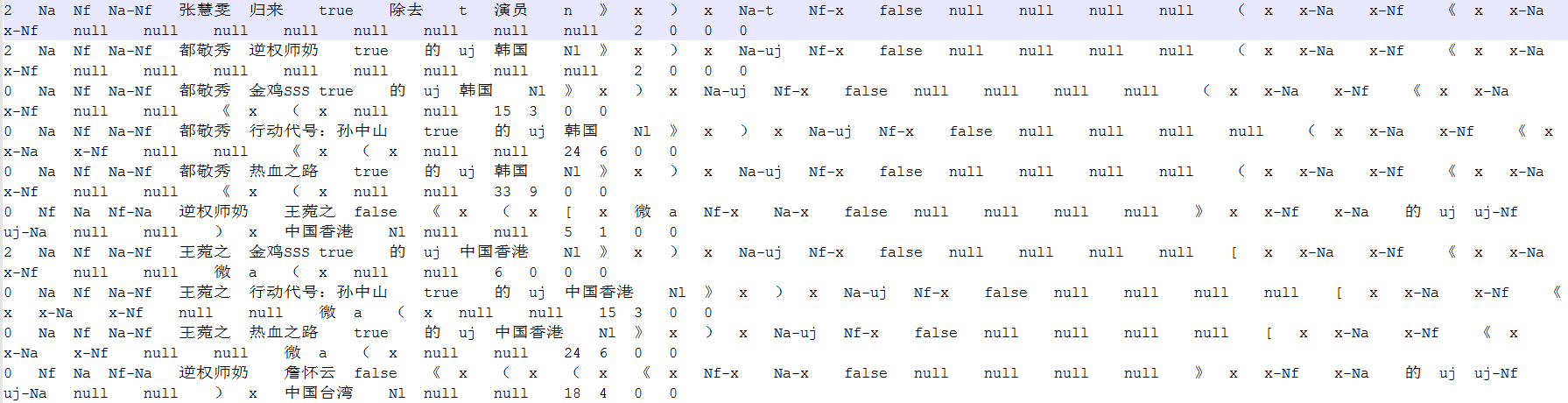


图 4-1 最大熵模型关系分类训练集

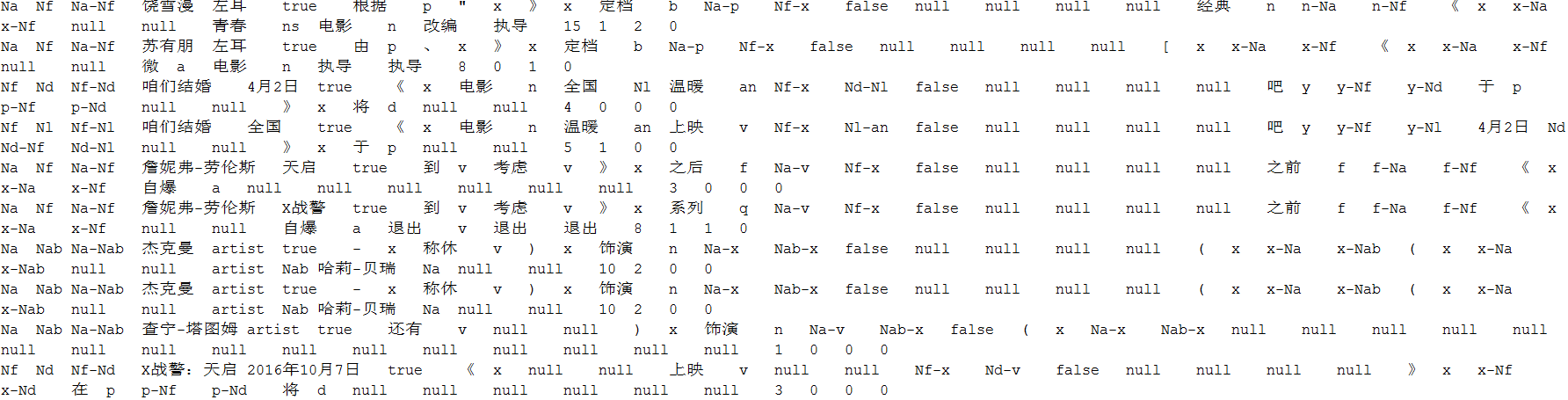


图 4-2 最大熵模型关系分类测试集

最大熵关系分类的完整代码见：<https://github.com/ecnumjc/maxnet>

**4.3.3.3 CRF分类**

使用条件随机场进行关系分类，训练集和测试集与使用最大熵模型一致。需要注意的是，使用条件随机场进行关系分类，需要针对抽取的特征维数和特点编写模板文件，即template文件。该文件详见<https://github.com/ecnumjc/CRF_RE>

##### 4.3.3.2 SVM分类器

本文采用台湾大学林智仁教授等人开发的LibSVM作为svm分类器。该SVM分类器要求特征值必须为整数值，而本文提取的特征，如图4-1和图4-2所示，皆为字符串型，无法直接使用libsvm。修改方案为，对于原始字符串特征fi，统计它所有的取值，然后对每个取值vi建立一个新的特征fi-vi,如果fi取值恰为vi,则新特征fi-vi取值为1，否则为0，如此把原始的一维字符串型特征转换为高维的二值特征。

#### 4.3.4 实验结果

### 4.4 方法4——基于半监督的关系抽取

对于缺乏训练集，利用一些种子关系和一些高精度的模板，也可以做些游泳的关系抽取工作。Bootstrapping就是其中一种代表方法。该方法的主要思想是，首先利用种子关系生成关系模板，然后基于关系模板抽取新的关系，得到新关系后，从中选择可信度高的关系作为种子，再寻找新的关系和新的模板，如此不断迭代, 直到没有新的关系或新的模式产生。该方法的代表性系统Snowball[5] 和DIPRE[6]。

基于Bootstrapping的关系抽取方法流程图如图4.1所示

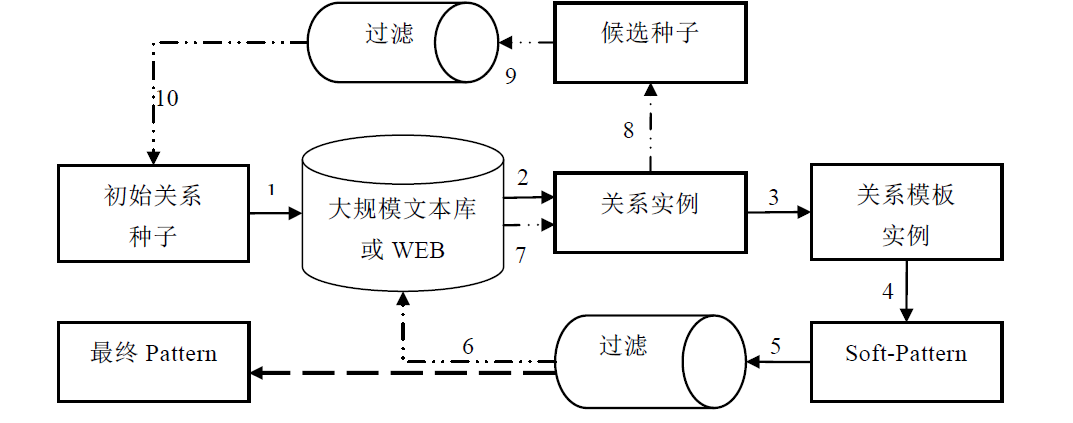


图4.1 Bootstrapping关系抽取流程图

伪代码描述如下：

BootStrapping Sequential Pattern Mining(pattern P’,Seed S’)

1. S←S’
2. P←P’
3. While not Done
4. Do Occt←Match-Seeds(S)
5. P←P∪Learn-Patterns(Occt)
6. Evaluate-Pattern(P)
7. P←{p∈P|Pattern-Filter-Condition(p)}
8. Occp←Match-Pattern(P)
9. S←S+Extract-Seeds(Occp)
10. Evaluate-Seeds(S)
11. S←{t∈S|Seed-Filter-Condition(t)}

### 4.5 方法5——基于CNN的关系抽取

#### 4.5.1 训练word embeddings

**Step 1:**下载中文维基数据：<https://dumps.wikimedia.org/zhwiki/latest/zhwiki-latest-pages-articles.xml.bz2>

**Step2:** process\_wiki.py处理这个XML压缩文件，该process\_wiki.py文件见github

执行大约10 mins,命令：python process\_wiki.py zhwiki-latest-pages-articles.xml.bz2 wiki.zh.text

**Step3:**将繁体字转化为简体字

opencc -i wiki.zh.text -o wiki.zh.text.jian -c zht2zhs.ini

**step4:**中文分词

python -m jieba -d" " wiki.zh.text.jian > wiki.zh.text.jian.seg  #将已将转换为简体字的文件使用jieba命令行进行分词

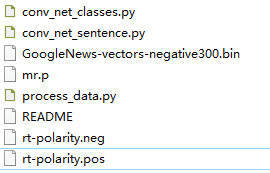
**step5：**将不是utf-8编码部分的文本转化为utf-8编码

iconv -c -t UTF-8 < wiki.zh.text.jian.seg > wiki.zh.text.jian.seg.utf-8

**step6:**使用word2vec训练word embeddings

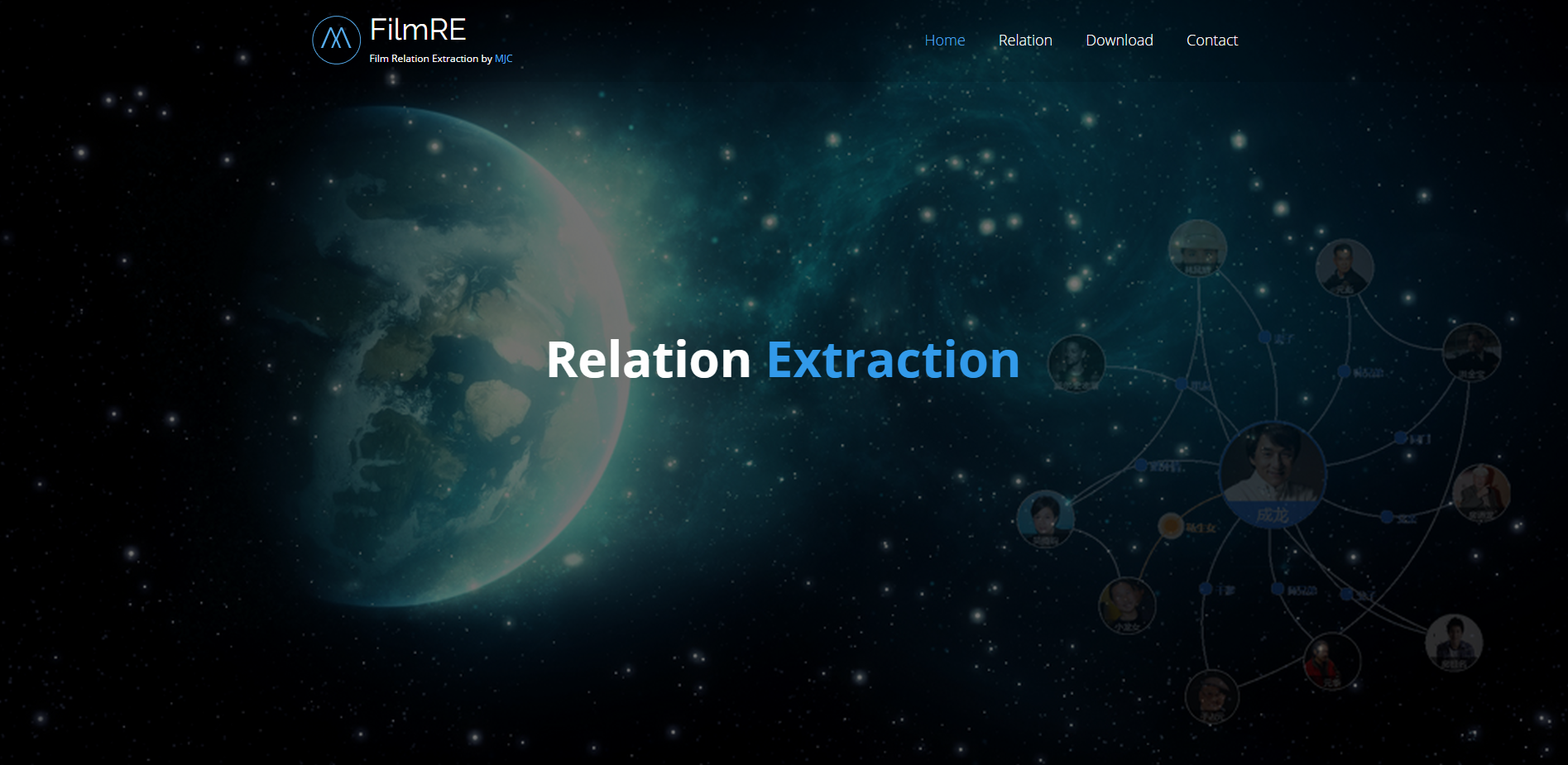
python train\_word2vec\_model.py wiki.zh.text.jian.seg wiki.zh.text.model wiki.zh.text.vector

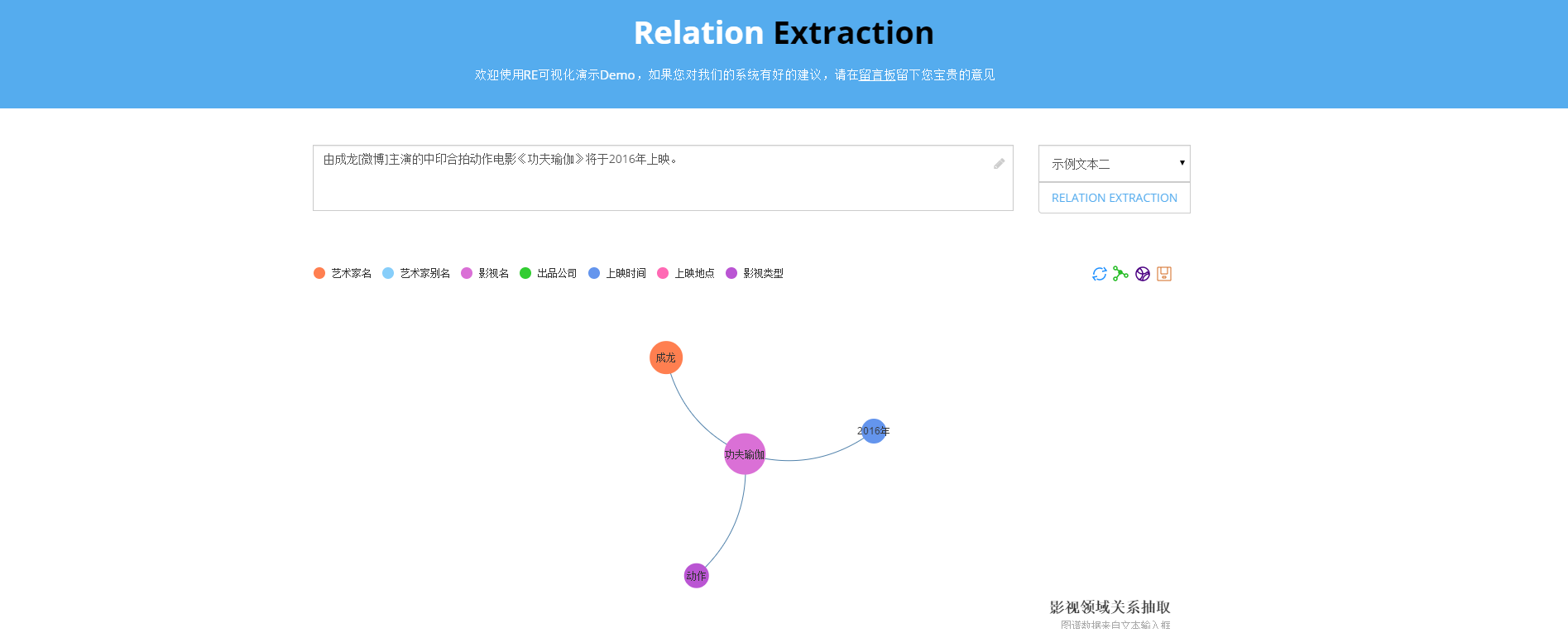
#### 4.5.2 基于CNN关系分类实验

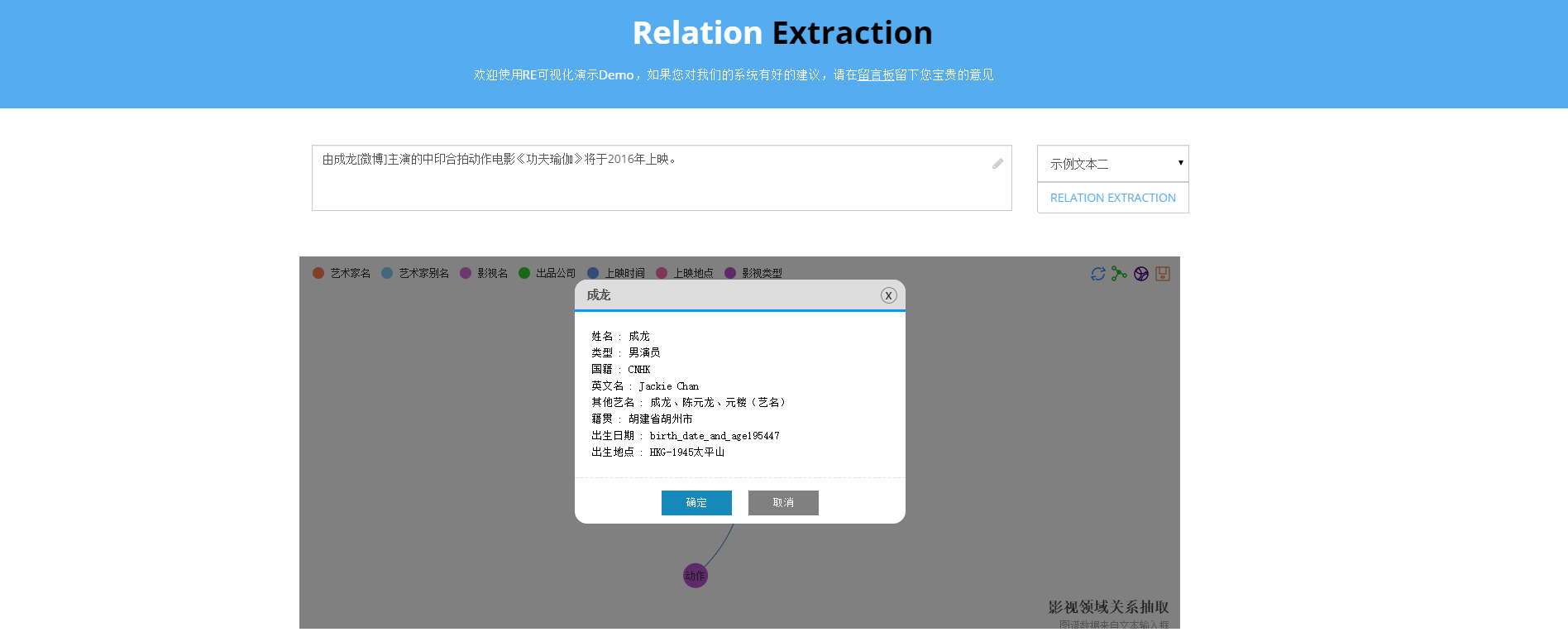


#### 4.5.3 实验结果

## 五、实体关系抽取算法的封装(关系抽取平台开发)







## 六、参考文献

1. Davidov D, Rappoport A. Classification of Semantic Relationships between Nominals Using Pattern Clusters.[J]. Acl, 2008:227-235.
2. Davidov D, Rappoport A. Unsupervised Discovery of Generic Relationships Using Pattern Clusters and its Evaluation by Automatically Generated SAT Analogy Questions.[J]. Proc of the Annual Meeting of the Acl, 2008.
3. Zhou G, Qian L, Fan J. Tree kernel-based semantic relation extraction with rich syntactic and semantic information[C]. //Information Sciences. 2010:1313-1325.
4. Banko, M., Cafarella, M. J., Soderland, S., Broadhead, M., & Etzioni, O. (2007). Open information  extraction from the web. IJCAI ’07: Proceedings of the 20th International Joint Conference on Artificial Intelligence. Hyderabad, India.
5. Agichtein E, Gravano L. Snowball: Extracting Relations from Large Plain-Text Collections[J]. In Proc. of the 5 th ACM International Conference on Digital Libraries (ACMDL’00, 2000:85-94).
6. Brin S． Extracting Patterns and Ｒelations from the World Wide Web［C］．In: Proceedings of International Workshop on the World Wide Web and Databases． London，UK: Springer － Verlag，1998:172 － 183．
7. HEARST M A. Automatic acquisition of hyponyms from large text corpora[C]. //Proceedings of International Conference on Computational Linguistics. Association for Computational Linguistics, 1992:539--545.
8. Kambhatla N. Combining lexical, syntactic, and semantic features with maximum entropy models for extracting relations[J]. Proceedings of Acl’, 2004.
9. Qian L, Zhou G, Kong F, et al. Exploiting constituent dependencies for tree kernel-based semantic relation extraction[J]. International Conference on Computational Linguistics, 2008, 1.
10. Yao X, Durme B V. Information Extraction over Structured Data: Question Answering with Freebase[J]. In Association for Computational Linguistics , 2014:956-966.
11. Fader A, Zettlemoyer L, Etzioni O. Open question answering over curated and extracted knowledge bases[J]. Homes.cs.washington.edu, 2014:1156-1165.