# 限定域文本语料的短语挖掘（Phrase Mining）

夕小瑶的卖萌屋 1月13日

一只小狐狸带你解锁NLP/ML/DL秘籍

正文来源：丁香园大数据

## 前言

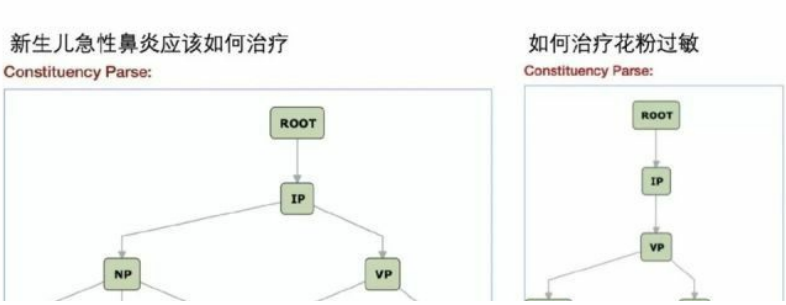
短语挖掘（Phrase Mining）的目的在于从大量的文本语料中提取出高质量的短语，是NLP领域中基础任务之一。短语挖掘主要解决**专业领域（如医疗、科技等）**的专业词典不足的问题，减少人工整理成本。

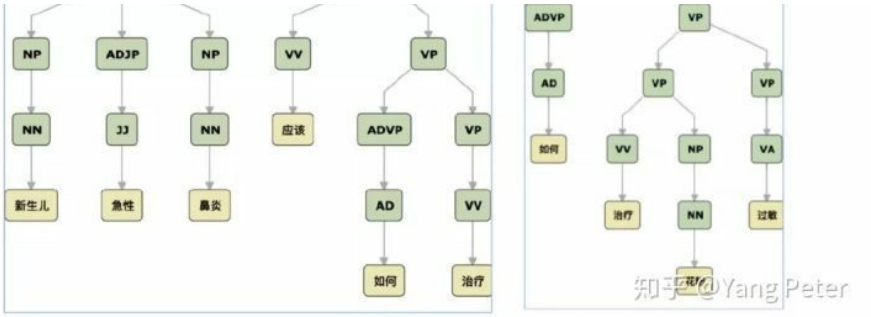
大家都知道，jieba分词是中文分词领域比较好的工具[1,2]，其在分词时使用的方法是根据已统计的词库，利用前缀词典对句子切分，根据所有切分的结果构建有向无环图的方式寻找最优的切分路径。对于存在**未登录词**其使用的方式是根据序列标注的结果，使用Viterbi算法计算最优的状态序列。使用jieba分词可以解决一些**普适化**的需求，但是对于某些特定的专业领域，要达到较好的分词要求，需要整理一批质量较高的**专业领域词典**。但是呢，我们可用到的数据往往是大量无标注的文本，如果人工去整理成本会很高，所以我们可以通过什么方法可以自动提取一些高质量的短语呢(●'◡'●)?

以**医疗领域**为例，丁香园大数据团队是一个处理医疗大数据的团队，每天要处理大量的医疗文本数据，例如论坛文本，医学论文，诊断报告等（里面会不会也有小夕的数据呢😳）。这些专业医疗领域的数据和平时日常的数据有很大的不同，会有大量我们听起来怕怕的专业术语⊙﹏⊙‖，抽取的高质量的短语无疑对**优化检索内容，taxonomy construction构建上下位层次结构、主题模型**等等都非常的重要。

## 无监督抽取方法

根据丁香园log数据汇总发现，很多疾病词和症状词来源于一些特定词的排列组合，比方说牛奶过敏，急性鼻炎，是一些NN和形容词/动词的组合，其实就是**浅层句法分析**的结果，例如："新生儿急性鼻炎应该如何治疗"，coreNLP给出的结果如下图所示：






其中新生儿急性鼻炎是一个**名词短语(NP)**，是由NN + JJ + NN组成的，传统的方式是根据**POS规则模版**[3]对phrase进行提取。但是在实际操作过程中又会存在一些问题，比方说"如何治疗花粉过敏"这句话中的Phrase应该是花粉过敏，但是治疗和花粉合并成了动词短语。但是如果要穷尽所有的**POS pattern**，并不是一件容易的事情，而且pattern之间可能会存在一些冲突，于是pattern之间排序又成了另一个坑¬（￣Д￣）┌。

2012年Matrix67提出了《互联网时代的社会语言学：基于SNS的文本数据挖掘》一种基于统计学角度的**新词挖掘**算法，通过计算凝固度和左右临字信息熵抽取新词，效果灰常不错o(*￣▽￣*)ブ。

《西游记》抽取结果如下所示：

> 行者，八戒，师傅，三藏，大圣，唐僧，沙僧，和尚，菩萨，怎么，长老，老孙，两个，甚么，国王，徒弟...

《资本论》抽取结果：

> 资本，生产，价值，劳动，商品，货币，部分，工人，形式，价格，利润，我们，作为，剩余价值，过程...

可以用在丁香论坛的医患对话日志上却差强人意，这个方法抽取大量**用户俗语**。

> 可以，如果，治疗，需要，医生，情况，建议，检查，什么，这个，问题，现在，症状，目前，或者，医院...

于是考虑去除停用词后再试一试，发现效果确实有所改善，算法找到一些靠谱的词汇比方说肝硬化,肝癌,拉肚子,大便不成型,痔疮出血,红色小疙瘩...。可是呢，这样的操作对出现频率低的短语不是很友好，很有可能被阈值过滤掉，人工给定阈值，没有一个很好的参照标准，数据中可能会存在很多的噪音，无法较好的筛选出有用的短语。

2014年韩家炜团队的学生Ahmed El-Kishky提出一种基于**频繁模式挖掘**和统计的方法TopMine，无监督的对语料进行Phrase Mining。这项工作的主要目的是对文本进行**主题挖掘**。在这篇论文中将主题挖掘分为两个步骤，第一步根据Phrase Mining抽取的结果对文本进行分割，第二部根据分割后的文本约束Topic模型。在Phrase Mining中，根据上下文信息衡量合并后的score，判断是否对token进行合并，伪代码如下所示：

**Algorithm 2:** Bottom-up Construction of Phrases from Ordered Tokens

**Input:** Counter $C$, thresh $\alpha$
**Output:** Partition

1 $H \leftarrow MaxHeap()$

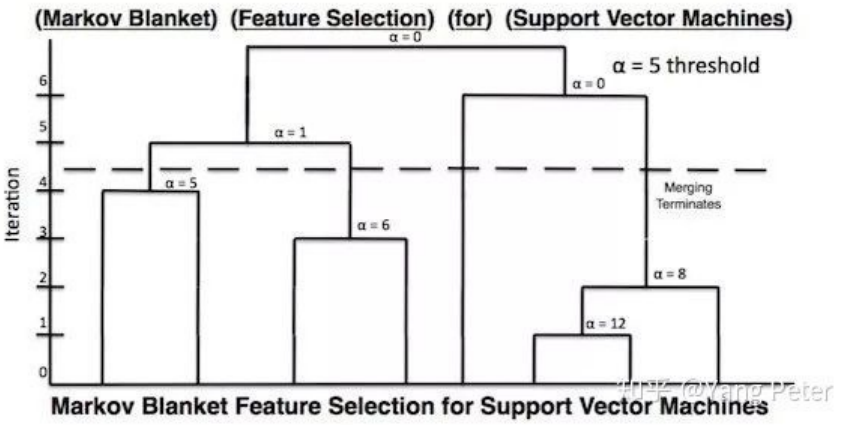
```
2  Place all contiguous token pairs into H with their
   significance score key.
3  while H.size() > 1 do
4  |   Best ← H.getMax()
5  |   if Best.Sig ⩾ α then
6  |   |   New ← Merge(Best)
7  |   |   Remove Best from H
8  |   |   Update significance for New with its left phrase
   |   |   instance and right phrase instance
9  |   else
10 |   |   break
11 |   end
12 end
```



通过给定阈值的方式进行迭代，其中score作为判断合并条件计算公式如下所示：

$$\alpha(P_1, P_2) \approx (f(P_1 \bullet P_2) - \mu_0(P_1, P_2))/\sqrt{f(P_1 \bullet P_2)}$$

其举了一个很有意思的例子，比方说：Markov Blanket Feature Selection for Support Vector Machines这句话来说如果只根据Vector可能只会把文章划分为数学或者物理Topic中，但是显然Support Vector Machines是一个整体,根据支撑向量机可以将其划分为计算机的主题下：




Kavita Ganesan2018年提出《How to incorporate phrases into Word2Vec – a text mining approach》和2019年苏神给出了一个 **无监督挖掘方案**《分享一次专业领域词汇的监督挖掘》有相似之处，只不过苏神再基础上加入一些平行语料，根据停用词确定phrase边界，用PMI等设定阈值方式抽取新词，进行分词，并构建词向量模型。选取一些种子词汇作为初始词，根据抽取新词的词向量计算种子词之间的相似度，设定阈值的方式将相似度高的词加入到候选集中，对于无监督短语挖掘是一种比较新颖的思路，因为词向量包含丰富的上下文语义信息，通过上下文信息计算相似度，将新词进行聚类，这种方式可以较好的筛选出一些高质量的phrase。但是在第一步分词时，根据阈值所发现的新词边界不好控制，会存在大量噪音，比方说，在《西游记》中会抽取出,行者笑道,那妖精,的和尚等词汇，所以在最后一步需要加入了一些规则进行过滤。

## 弱/远程监督抽取方法

韩家炜团队关于Phrase Mining团队的三部曲，刚才已经简单的介绍了其中之一TopMine，其主要目的是对语料库中的文本Topic进行挖掘，其中利用Phrase Mining的方法对文本进行分割。其另外两部SegPhrase和AutoPhrase分别使用弱监督和远程监督的方式对phrase进行抽取并进行质量监测。

### SegPhrase

韩教授的学生刘佳硅认为TopMine的方法是完全无监督的，那么是不是选用一些少量的带标签的数据，可能会在很大程度上提高抽取结果。其认为高质量的短语是可以优化分词结果的，而高质量的分词结果也可以优化phrase抽取的结果，将分词和高质量短语相结合。

Table 1: A hypothetical example of word sequence raw frequency

| sequence | frequency | phrase? | rectified | sequence | frequency | phrase? | rectified |
|---|---|---|---|---|---|---|---|
| relational database system | 100 | yes | 70 | support vector machine | 100 | yes | 80 |
| relational database | 150 | yes | 40 | support vector | 160 | yes | 50 |
| database system | 160 | yes | 35 | vector machine | 150 | no | 6 |
| relational | 500 | N/A | 20 | support | 500 | N/A | 150 |
| database | 1000 | N/A | 200 | vector | 500 | N/A | 200 |
| system | 10000 | N/A | 1000 | machine | 1000 | N/A | 150 |



原始计算频率时，并没有考虑真正分词的结果，只是统计词出现的频率，例如support vector machine出现了在预料中出现了100次，但是根据分词结果进行修正(rectified)后，其只出现了80次，同样的vector machine修正后只出现了6次。那么vector machine不算是一个phrase。

例如：A standard feature vector machine learning setup is used to describe在这句话中，存在vector machine但是根据上下文语义进行分词时，其分词结果应该是feature vector和machine learning

于是接下来的工作中，根据频繁短语监测生成短语候选集，再根据人工筛选出的高质量的短语使用RandomForest构建分类器模型，实验中发现选择200-300个数据就可以满足分类结果。其中特征选取如下表所示：

| feature | function |
|---|---|
| probability *P(u)* | $p(u) = \frac{f[u]}{\sum_{u' \in \mathcal{U}} f[u']}$ |
| minimized PMI *argmin(PMI(v))* | $\langle u_l, u_r \rangle = \arg \min_{u_l \oplus u_r = v} \log \frac{p(v)}{p(u_l)p(u_r)}$ |
| *PMI(l,r)* | $PMI(u_l, u_r) = \log \frac{p(v)}{p(u_l)p(u_r)}$ |
| *PKL(v\|l,r)* | $PKL(v \| \langle u_l, u_r \rangle) = p(v) \log \frac{p(v)}{p(u_l)p(u_r)}$ |
| *HasStopword* | 0/1 |
| *IDF* | $\mathrm{IDF}(w) = \log \frac{\lvert\mathcal{C}\rvert}{\lvert\{d \in [D] : w \in C_d\}\rvert}$ |
| *HasPunctuation* | 0/1 |
| ... | ... |



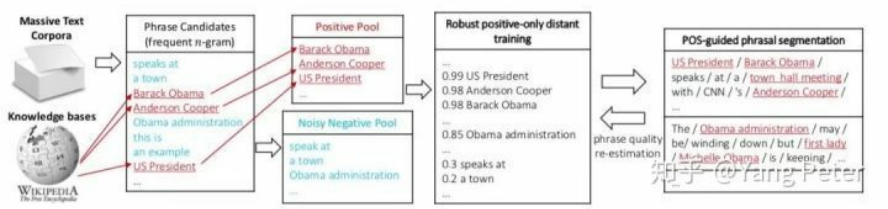
这篇论文在当时的效果不错，但是存在一个缺点，文中说300个标记词汇就够了，那么这300个标记数据应该如何选取？需要人工的去选择一些高质量的短语去构造分类器，如果在一些特定的领域则需要一些专业领域人士对领域内的数据进行筛选。所以韩教授的学生商静波提出了一种远监督方法AutoPhrase自动的对短语进行挖掘。

### AutoPhrase

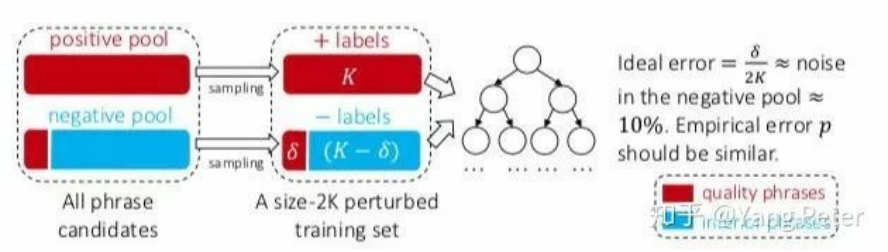
2017年韩教授的学生商静波提出一种远程监督的方法进行Phrase Mining，AutoPhrase使用wiki或Freebase等数据构建高质量词典，代替SegPhrase人工打标签的过程。其在技术上以下两个创新点。

**Robust Positive-Only Distant Training**：使用wiki和freebase作为显眼数据，根据知识库中的相关数据构建Positive Phrases,根据领域内的文本生成Negative Phrases，构建分类器后根据预测的结果减少负标签带来的噪音问题。

**POS-Guided Phrasal Segmentation**：使用POS词性标注的结果，引导短语分词，利用POS的浅层句法分析的结果优化Phrase boundaries。




如上图所示，根据frequent n-gram抽取phrase Candidates根据远程监督的方式，根据wikipedia进行过滤筛选出Positive Pool和Noisy Negative Pool，对于Positive Pool来说，其信源比较准确，于是Positive Pool的抽取结果肯定是置信度极高的，而Negative Pool是存在噪音的，因为可能有一些Phrase不存在WikiPedia中，因此文中提到用一种集成学习的方式降低噪音。




构建多组基本的分类器，分别从Positive Pool和Negative Pool中随机抽取出K个candidates全部，而在负样本中存在*perturbed training set*见图中的δ，为了尽可能低的降低训练误差的基础分类器，构建一颗未进行剪枝的决策树的方式，当在*perturbed training set*中没有两个positive和negative phrase共享相同的特征值时，我们认为此时的决策树可以达到100%的训练准确率。最后通过ranking的方式输出排序的结果。这种方法的结果Segphrase相比有着显著的提升。

| | *EN* | *CN* | |
|---|---|---|---|
| Rank | Phrase | Phrase | Translation (Explanation) |
| 1 | Elf Aquitaine | 江苏 舜 天 | (the name of a soccer team) |
| 2 | Arnold Sommerfeld | 苦 艾 酒 | Absinthe |
| 3 | Eugene Wigner | 白发 魔 女 | (the name of a novel/TV-series) |
| 4 | Tarpon Springs | 笔记 型 电脑 | notebook computer, laptop |
| 5 | Sean Astin | 党委 书记 | Secretary of Party Committee |
| ... | ... | ... | ... |
| 20,001 | ECAC Hockey | 非洲 国家 | African countries |
| 20,002 | Sacramento Bee | 左翼 党 | The Left (German: Die Linke) |
| 20,003 | Bering Strait | 菲 沙 河谷 | Fraser Valley |
| 20,004 | Jacknife Lee | 海马 体 | Hippocampus |
| 20,005 | WXYZ-TV | 斋 贺光希 | Mitsuki Saiga (a voice actress) |
| ... | ... | ... | ... |
| 99,994 | John Gregson | 计算机 科学技术 | Computer Science and Technology |
| 99,995 | white-tailed eagle | 恒 天然 | Fonterra (a company) |
| 99,996 | rhombic dodecahedron | 中国 作家 协会 副 主席 | The Vice President of Writers Association of China |
| 99,997 | great spotted woodpecker | 维他命 b | Vitamin B |
| 99,998 | David Manners | 舆论 导向 | controlled guidance of the media |
| ... | ... | ... | ... |

## 开始搞事情

看了很多的方法，磨拳擦掌开始搞一个属于自己的Phrase mining了，借鉴之前的方法，发现很多都是从统计学角度构建一批先验知识，比方说计算语料中的PMI和一些KL散度等等进行抽取。但是如果获得的数据是一些短文本数据，又将如何提取这些特征呢？我们直接用AutoPhrase的方法套用？仿佛又不是很合适，首先，无法保证分词的效果；其次，stanford POS的浅层句法分析的结果并不适用于所有领域，比如花粉过敏。

根据丁香园本身的业务需求和之前提到的一些方法对特征和分类器模型进行了修改，并没有使用stanford提供的浅层句法分析的结果，而是根据知识库目前现有的一些医疗数据构建了n-gram模型作为特征，再借鉴远监督的方式根据知识库中已有的词库数据，进行词性标注，统计词性标注的结果作为特征，再根据bert构建的字向量作为分割的特征。

同样类似于AutoPhrase的方式构建数据集，Positive Pool中的数据来源于目前知识库已有的医疗数据，Negative Pool中的数据来自N-gram随机选取的非库中的数据，再根据一些规则和N-gram的概率等阈值信息简单的过滤了一些负样本中的脏数据。使用gbdt(Gradient Boosting Decision Tree)构建分类器模型。目前抽取效果如下所示：


在抽取结果中可以看到，一句话中可以抽取多个phrase，对于分词来讲，不同方式组合phrase会生成不同的分词格式，根据score阈值进行过滤，根据phrase抽取结果，查询所有的抽取组合，根据组合结果不同，通过phrase长度和个数进行约束，公式如下所示：

$$f(p) = \frac{1}{n} * \sum_{i=1}^{n} Score_{pi} + \log(len_{pi})$$



下图为最后输出排序的结果：


## 总结

这篇文章调研了关于Phrase Mining构建的方法来解决在特定专业领域中存在未登录词和个性化分词问题。 Phrase Mining只是将非结构化文本转化为半结构化文本的第一步，之后还需要在其基础上进行实体链接和知识图谱的构建。Phrase Mining可以根据数据的不断扩充对模型不断的优化，在其优化的同时，也对后续任务有着决定性的帮助。

可能喜欢

- 跨平台NLP/ML文章索引
- 万万没想到，我的炼丹炉坏了
- 词搜索引擎--项词典与倒排索引优化
- 如何与GPU服务器优雅交互

不要忘了关注小夕~星标☆小夕哦~

参考文献

[1] 结巴分词2--基于前缀词典及动态规划实现分词

[2] 结巴分词3--基于汉字成词能力的HMM模型识别未登录词

[3] http://www.nltk.org/howto/chunk.html

[4] Scalable Topical Phrase Mining from Text Corpora

[5] Mining Quality Phrases from Massive Text Corpora

[6] Automated Phrase Mining from Massive Text Corpora

[7] TruePIE: Discovering Reliable Patterns in Pattern-Based Information Extraction

[8] 中文基本复合名词短语语义关系体系及知识库构建

[9] How to incorporate phrases into Word2Vec – a text mining approach

你的每一个“在看”我都当成了喜欢