# 对于“智慧政务”中的文本挖掘应用的研究

## 摘要

在新的电子政务建设需求与发展形势下，本着提高政府工作效率、更好的为群众办事的目的等多方面因素促使政府的信息化建设向着智能化、平台化的方向发展。基于大数据、云计算和人工智能等信息技术的发展，建立基于自然语言处理技术的政务系统，创新政务发展方式，即是政府政务工作的发展方向，也是打造智慧型政务平台的基础。本文基于自然语言处理技术对文本挖掘应用进行研究，打造“智慧政务”系统。

问题一针对群众留言的分类问题，本文基于朴素贝叶斯算法建立留言分类模型。因为需要研究的原始样本是文本信息，所以我们采用了多项式朴素贝叶斯算法（*MultinomialNB*），*MultinomialNB* 的特点是在计算时，所需处理的数据大都表示为词向量，这样便可以对文本进行计算等处理操作，*MultinomialNB* 是文本分类中，经典的朴素贝叶斯算法。

本题利用 *sklearn.model_selection.train_test_split* 进行数据集的划分操作，建立 *CountVectorizer* 词袋模型，最后再 *Python* 中定义方法 *text_classifier.predict()* 和 *text_classifier.score(x_test, y_test)*，进行数据运算和测试集评分操作，最后建立出关于留言标签的分类模型。

问题二关于热点问题挖掘的研究，本文以隐马尔可夫模型和最大熵模型为基础进行研究，通过引用语言技术平台 *LTP* 中的分词、词性标注以及命名实体识别技术，对附件 3 中的留言内容进行词法分析操作。*LTP* 中的分词技术以 *CRF* 算法（*Conditional Random Field*，条件随机域）为主要思想，这种方法可以有效的避免在分词时产生歧义的问题；*LTP* 中的词性标注技术基于隐马尔可夫模型并结合最大熵模型算法，*LTP* 采用的这种优化方法，避免了独立特征假设，提高了系统的精确度；*LTP* 中的命名实体识别技术以 *MEMM* 模型为基础，样本信息在经过分词操作后，通过词性标注处理后附注上词性，最后命名实体识别技术来识别词性，以词性特征为条件进行文本信息从抽取和分类。最后再结合 *Python* 中的统计、预测函数，建立了热点问题的统计挖掘模型。

问题三要求针对答复信息，从相关性、完整性等角度对答复意见进行评价。本题根据题目要求，引用了软件工具 *Word2vec*，以 *Skip-gram* 模型和连续词袋（*CBOW*）模型为基础，从文本相似度和词向量的角度进行研究，将原始文本转换成词向量，然后计算词向量之间的距离，通过比较词向量距离值的大小，可以得出文本内容的相似程度，因此可以建立文本相关性评价模型

关键词：*LTP*、w*ord2vec*、词向量、文本相似度、朴素贝叶斯分类器、词袋模型

# 目录

# 1 绪论

## 1.1 问题重述

随着时代社会发展，微信、微博、市长信箱、阳光热线等网络问政平台逐步成为政府了解民意、汇聚民智、凝聚民气的重要渠道，各类社情民意相关的文本数据流不断攀升，给以往主要依靠人工来进行留言划分和热点整理的相关部门的工作带来了极大的挑战。在新的电子政务建设需求与趋势下，以及大数据、云计算、人工智能等信息技术的发展，建立基于自然语言处理技术的智慧政务系统已经是社会治理创新发展的新趋势，有利于政府高效的整合各方面信息资源、深化业务综合能力，对提升政府的管理水平和施政效率具有极大的推动作用。为了更好的处理收集自互联网公开来源的群众问政留言记录，及相关部门对部分群众留言的答复意见，利用自然语言处理和文本挖掘的方法解决下面的问题。

问题一：群众留言分类

在处理网络问政平台的群众留言时，工作人员首先按照一定的划分体系（参考附件 1 提供的内容分类三级标签体系）对留言进行分类，以便后续将群众留言分派至相应的职能部门处理。目前，大部分电子政务系统还是依靠人工根据经验处理，存在工作量大、效率低，且差错率高等问题。请根据附件 2 给出的数据，建立关于留言内容的一级标签分类模型。

通常使用 *F-Score* 对分类方法进行评价：

$$F_1 = \frac{1}{n}\sum_{i=1}^{n}\frac{2P_iR_i}{P_i + R_i},$$

其中 $P_i$ 为第 $i$ 类的查准率，$R_i$ 为第 $i$ 类的查全率。

问题二：热点问题挖掘

某一时段内群众集中反映的某一问题可称为热点问题，如“*XXX* 小区多位业主多次反映入夏以来小区楼下烧烤店深夜经营导致噪音和油烟扰民”。及时发现热点问题，有助于相关部门进行有针对性地处理，提升服务效率。请根据附件 3 将某一时段内反映特定地点或特定人群问题的留言进行归类，定义合理的热度评价指标，并给出评价结果，按表 1 的格式给出排名前 5 的热点问题，并保持为文件“热点问题表.*xls*”。按表 2 的格式给出相应热点问题对应的留言信息，并保持为“热点问题留言明细表.*xls*”。

问题三：答复意见的评价

针对附件 4 相关部门对留言的答复意见，从答复的相关性、完整性、可解释性等角度对答复意见的质量给出一套评价方案，并尝试实现。

## 1.2 本文的主要工作及研究内容

在信息时代如何高效地从海量信息中挖掘有价值的数据，一直是人民持续关注的问题。本文的研究内容主要有以下几个方面：

（1）研究朴素贝叶斯算法中的先验为高斯分布的朴素贝叶斯算法、伯努利贝叶斯算法以及多项式贝叶斯算法。经过甄别三者之间的特征，最终选取多样式贝叶斯算法为理论原理，建立贝叶斯分类模型。

（2）研究中文分词处理的关键技术。中文分词的五大算法，本文着重研究

了基于词典的匹配算法、基于统计原理的 N-gram 和 HMM 模型、最大熵模型。在本文中阐述了他们的理论原理，并详细分析了以这些模型为基础的分词系统。

（3）本文详细分析了 LTP 基于 CRF 算法分词技术、基于 MEMM 为代表模型的词性标注技术、基于统计和规则相结合的命名体识别技术。

（4）针对问题三，本文详细分析了基于 Word2vec 的文本相似度的计算过程，详细解释了词向量的产生过程、原理、 用法等，并大体介绍了 Word2vec 的核心功能，

（5）基于上述理论模型，针对题目要求建立相关计算模型，并通过具体数据和案例进行测试。

最后是对本次学习研究的总结。

## 2 相关理论和技术概述

### 2.1 朴素贝叶斯方法相关理论

贝叶斯分类方法[1]采用了概率推理方法，包括先验概率和后验概率的计算。其原理是通过计算已知样本中各类别下各个特征属性的条件概率，再对每个特征属性计算所有划分的条件概率，对每个类计算 $P(x|y_i)P(y_i)$，以 $P(x|y_i)P(y_i)$最大项为 $x$ 的所属类别。

$$P(B \mid A) = \frac{P(A \mid B)P(B)}{P(A)}$$

贝叶斯是概率框架下实施决策的基本方法。对于文本分类问题，在所有相关概率都已知的情况下，朴素贝叶斯分类器是贝叶斯分类器中应用最为广泛的模型之一。

#### 2.1.1 朴素贝叶斯算法

简单来说，朴素贝叶斯的思想基础对于待分类项，求解各类别下各个特征属性的条件概率，哪个最大，就认为此待分类项属于哪个类别。但是在实际应用中，两个特征值之间很难达到绝对的相互独立。

在给定一个训练集 $Y$ 和待分类项 $x_n$，贝叶斯定理的状态下面的关系：

$$P(y \mid x_1,\ldots,x_n) = \frac{P(y)P(x_1,\ldots,x_n \mid y)}{P(x_1,\ldots,x_n)}$$

由上式可以观察出，分母永远是常数，因此只要实现分子最大化即可，假设各特征属性之间是相互对立的，所以有：

$$P(x_i \mid y) = P(x_i \mid y, x_1,\ldots,x_{i-1},x_{i+1},\ldots,x_n)$$

对于所有属性，上式可以简化为：

$$P(y \mid x_1,\ldots,x_n) = \frac{P(y)\prod_{i=1}^{n} P(x_i \mid y)}{P(x_1,\ldots,x_n)}$$

由于 $P(x_1,...,x_n)$ 是恒为常数，故有：

$$P(y \mid x_1,\ldots,x_n) \propto P(y)\prod_{i=1}^{n} P(x_i \mid y)$$

$$\Downarrow$$

$$\hat{y} = \arg\max_{y} P(y)\prod_{i=1}^{n} P(x_i \mid y)$$

朴素贝叶斯分类算法有多种，是因为对 $P(x_i/y)$做出了不同的假设，而 *scikit-learn* 中朴素贝叶斯类库的使用也比较简单。在 *scikit-learn* 中，有 3 个种朴素贝叶斯的分类算法： *GaussianNB*， *MultinomialNB* 和 *BernoulliNB*。其中 *GaussianNB* 就是先验为高斯分布的朴素贝叶斯，*MultinomialNB* 就是先验为多项式分布的朴素贝叶斯，而 *BernoulliNB* 就是先验为伯努利分布的朴素贝叶斯。

### 2.1.2 高斯朴素贝叶斯算法

对于 *x* 是连续随机变量的分类，可以假设 $P(x_1|y_i),P(x_2|y_i)...P(x_n|y_i)$是彼此独立的，这些值相乘得到 P(x|yi)，“独立”也就是朴素贝叶斯的朴素之处。*GaussianNB* 实现了高斯朴素贝叶斯分类算法。假设特征的似然为高斯分布：

$$P(x_i \mid y) = \frac{1}{\sqrt{2\pi\sigma_y^2}}\exp(-\frac{(x_i-\mu_y)^2}{2\sigma_y^2})$$

参数$\sigma_y$和$\mu_y$采用最大似然估计值。

### 2.1.3 伯努利贝叶斯算法

*BernoulliNB*：当特征属性为连续值时，并且是按多元伯努利分布的数据，那么在计算时可以直接使用伯努利分布的概率公式：

$$P(x_i \mid y) = P(i \mid y)x_i + (1-P(i \mid y))(1-x_i)$$

伯努利贝叶斯是一种离散分布，从上式可以观察出它只有两种可能结果

### 2.1.4 多项式贝叶斯算法

*MultinomialNB* 实现了对多项式分布数据的朴素贝叶斯算法，通俗来讲假设我们有一个固定的类集合 $C=\{c_1,c_2,\ldots,c_j\}$在多项式模型中：

$$\text{先验概率}P(C) = \frac{C\text{中的总数}}{\text{训练样本的总数}}$$

$$\text{类条件概率}P(x_i \mid C) = \frac{\text{类}C\text{下单词出现的次数之和}+1}{\text{类}C\text{的单词总数}+|V|}$$

|V|则表示训练样本包含多少种单词。P(C)可以认为是类别 C 在整体上占多大比例，即多大可能性。

当特征属性服从多项分布时，假设对于分布中的类别 $y_i$，参数为 $\theta_y=(\theta_{y1},\ldots,\theta_{yn})$，其中 *n* 是特征数量，那么 $\theta yi$ 等于概率 $P(xi \mid y)$，是 *i* 在 *y* 的一个样本中出现的概率。

$$\theta_{yi}=\frac{N_{yi}+\alpha}{N_y+\alpha n}$$

$$N_{yi}=\sum_{x\in T}x_i$$

$$N_y=\sum_{i=1}^{|T|}N_{yi}$$

### 2.1.5 朴素贝叶斯分类算法的优缺点

优点： 算法逻辑简单,易于实现；分类过程中时空开销小，只涉及二维存储。

缺点：在属性个数比较多或属性之间相关性较大时，分类效果不好。

这三个类适用的分类场景各不相同，主要根据数据类型来进行模型的选择。一般来说，如果样本特征的分布大部分是连续值，使用 *GaussianNB* 会比较好。如果如果样本特征的分大部分是多元离散值，使用 *MultinomialNB* 比较合适。而如果样本特征是二元离散值或者很稀疏的多元离散值，应该使用 *BernoulliNB*。根据具体实例，以及附件 2 样本，我们选择多项式贝叶斯算法。

## 2.2 中文分词

现有的中文分词算法有五大类：基于词典的分词方法，基于统计的分词方法，基于规则的分词方法，基于字标注的分词方法，基于人工智能技术（基于理解）的分词方法。

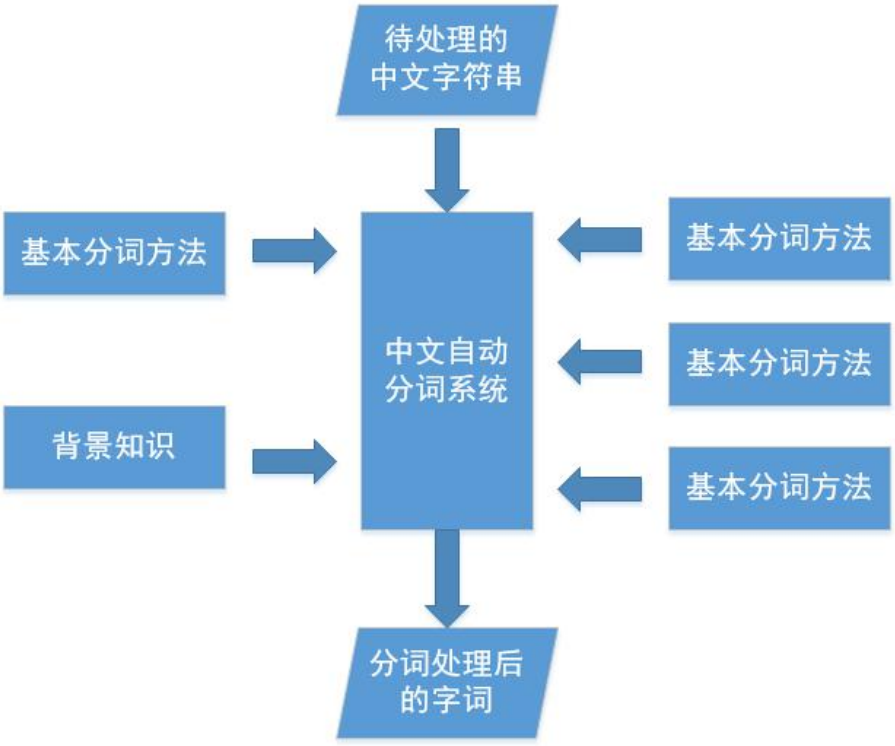


图 1 中文分词系统框架图

### 2.2.1 中文分词概念

问题 2 是针对热点问题的挖掘，这需要对留言内容进行分析、归类、挑选等操作，而在进行智能文本处理中，最主要的问题在于不出差错的情况下，如何顺利完成中文与机器语言之间的转化。因此，如何准确高效地提取文本中的词语，是解决中文语言与机器语言之间转化的问题关键，所以首先需采用中文分词技术。

实现人类语言与计算机语言之间的转化难点在于：计算机是否能够精确的识别人们的语言以及理解语言信息之间的潜在关系。如今，随着时代和社会的发展，人类语言的水平飞速提升，人类语言的内容以及语言之间的逻辑关系也随发展而越发复杂，但是如今计算机提取文字样本的技术，大大限制了计算机对人类语言的处理，经常出现“答非所问”、“上下文内容不相关”之类的现象，而产生这种情况的最根本原因，就是人类语言复杂的逻辑关系组成，使同样发文本，在不同的现实情况中，产生了广泛的多义或歧义现象。

在人类语言中，最复杂的语言之一就是中文。中文，由字可组成词，由词可组合成句，而大量的字、词和句又可组成一段语句。在英语、法语、 德语、等语言中，一个单词便代表一种词意，不同于这些语言的是，在中文中，常常几个字、词联合起来才能表达准确的词义，并且不同的情况下，词意也不相同。比如在英语中，“*Many citizens have reported to the government that urban waste treatment is not timely*.”这句话、每一个单词之间都有“空格”，计算机可以通过识别“空格”这一个分隔符，进而很容易的对语句进行分词处理。而在汉语中，同样语义的句子是“有多位市民向政府反映城市垃圾处理不及时”，相较于英文，中文语句没有类似于“空格”一样天然的分隔符，因此计算机无法对中文语句直接进行分割处理。


图 2 中文分词原理

如图所示中文分词原理，若想使计算机准确的识别中文信息，首先要把句子中的关键词提取出来，这这些关键词能够充分表达原句的含义并不产生其他句意。比如“已经有多位市民向政府反映城市垃圾处理不及时”这句话，正确切分应该是“已经/有/多位/市民/向/政府/反映/城市/垃圾/处理/不及时”,而其中能够充分表达句意的单词是“市民”、“反映”、“垃圾”、“处理”、“不及时”。从而可见，若想实现计算机对中文进行智能自动分词的操作所面临的困难是巨大的，与计算机对英文分词的操作存在天壤之别，所以，我们不能采用传统的分词算法对题目附件中的样本进行数据挖掘操作。

综上所述，所谓中文分词，就是中文信息处理的基本技术，指将一段话切分

成一个个单独的词。分词就是将连续的字序列按照一定的规范重新组合成词序列的过程，分词处理的目的是使计算机能够根据个别词汇检索到该原始文本。

### 2.2.2 基于词典的分词方法——匹配算法

基于词典的分词方法是运用字符串匹配，机械分词的方法。按照一定的规则将待切分的汉字串与匹配词典中的词条进行一一匹配，若在词典中找到某个字符串，则匹配成功，并将其切分成一个词。

这类方法按照扫描方向的不同、长度的不同、是否与词性标注过程相结合等方面，分为包各种形态的最大匹配分词方法、全切分分词算法、基于理解的中文分词算法等。

（1）最大匹配算法

①正向最大匹配算法（*Forward Maximum Matching*，简称 *FMM*）

正向最大匹配算法（*FMM*）[3]是通过在找大机器词典中，从前至后进行扫描匹配的算法，若匹配成功，则将这个匹配字段作为一个词切分出来；若匹配不成功，则将这个匹配字段的最后一个字去掉，剩下的字符串作为自我的匹配字段，进行再次匹配，重复以上过程直到切分出所有词为止。

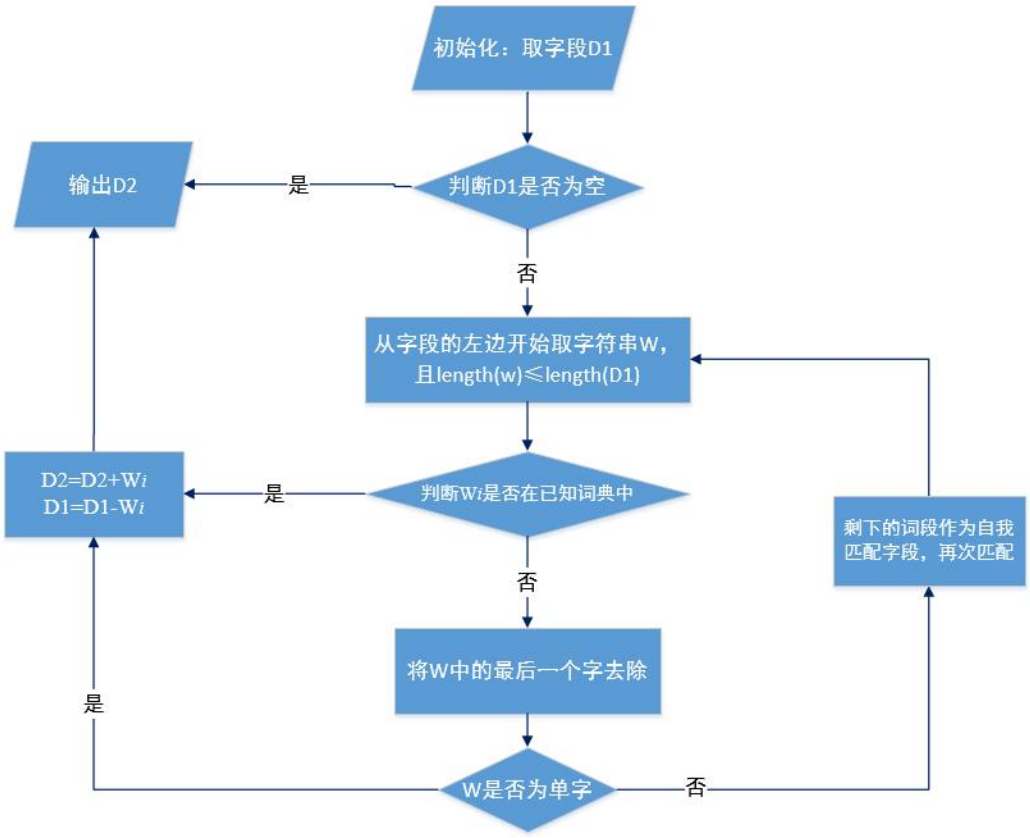


图 3 正向最大匹配算法流程图

*FMM* 算法在精确度上还有待提高，比如“我在家门口等人”这句话，因为中文的多义性，所以对于是否能组成词的标准不统一，在运用 *FMM* 进行分词时，即可以分成“我/在/家门口/等人”，也可以分成“我/在家/门口/等人”，所以在实际使用的过程中，我们还需要配合其他算法来提高 *FMM* 算法分词的精确度。

②逆向最大匹配算法（*Reverse Maximum Matching*，简称 *RMM*）

逆向最大匹配算法的基本原理与正向最大匹配算法相同，也可以说逆向最大匹配算法是正向最大匹配算法的逆向思维，即 *RMM* 的最大匹配顺序是从待匹配字段的末尾开始，由右向左，若匹配不成功，将匹配字段的最前一个字去掉。

已知实践证明，逆向最大匹配算法要优于正向最大匹配算法。

③双向最大匹配算法（*Bi-directction Matching method*，*BMM*）

双向最大匹配算法，是将正向最大匹配法得到的分词结果和逆向最大匹配法的得到的结果进行比较，从而选出正确的分词方法。

（2）全切分分词算法

利用词典匹配，把一语句中所有的词找出来，最终获取所有可能的切分结果。那么切分处理的词如何组成完整的句子，并使之与原句一模一样？由于全切分的结果，导致句子组合使产生的词序数目会随语句的长度而逐渐增加，因此这种方法耗费时间长，且对于歧义较多的句子，切分结果也不够精确。

### 2.2.3 基于统计的分词——*N-gram* 和 *HMM* 模型

这类方法主要是利用从大规模语料库中通过统计得到的各种概率信息，将语料库中的文本输入计算机，根据概率统计模型来统计其中词语的出现频率，以此来作为切分中文字符串的标准。

这种方法不需要进行词条匹配，只需统计文本中词的组合频率，与基于匹配的分词方法相比，所以基于统计的分词方法又叫作无词典的分词方法。这种方法往往不需人工维护规则，也不需复杂的语言学知识，有着较强的系统性和一致性。特别是在进入大数据时代后，是现今分词算法中较常用的做法。目前常用的模型有：*N* 元文法模型、隐 *Markov* 模型和最大熵模型等。

（1）*N* 元语法（*N-gram*）模型

*N-gram* 模型是将需切分的语段看成字符串列，假设取语段中的第 *n* 个词，根据上文提到的马尔可夫理论，那么它的概率与语段中的第 *n-1* 个词有关，若将所有的词组合成一句话，那么会有很多种组合方式，而其中出现某句话的事件概率就是每个词的概率的乘积 。若某句子的词串序列为$(w_1,w_2,w_3,...,w_n)$，则出现这句话的概率 $P(W)$为：

$$\begin{aligned} P(W) &= P(w_1 w_2 w_3 ... w_n) \\ &= P(w_1)P(w_2 \mid w_1)P(w_3 \mid w_1 w_2)...P(w_n \mid w_1 w_2 w_3 ... w_{n-1}) \\ &= \prod_{i=1}^{n} P(w_i \mid w_1 w_2 w_3 ... w_{i-1}) \end{aligned}$$

（2）隐马尔可夫模型（*Hidden Markov Model*,简称 *HMM*）

隐马尔可夫模型（*Hidden Markov Model*,简称 *HMM*）[4]是一种统计模型，它用来描述一个建立在马尔可夫模型基础上含有隐含参数的马尔可夫过程。其难点是根据观测值序列找到真正的隐藏状态值序列，然后利用这些参数来作进一步的分析。

在简单的马尔可夫模型（如马尔科夫链），所述状态是直接可见的，模型的参数就是各状态间的转换概率，而在隐马尔可夫模型中，状态是被隐含的、不直接可见的，无法直接对状态进行分析。

下面用一个简单的例子来阐述：

假设在一个箱子中，有三个颜色不同的袋子，分别是红色、蓝色、黄色。而三个袋子中又装有形状大小相同、数目不同的小球，红色袋子里有 6 个球，蓝色袋子里有 4 个球，黄色袋子里有 8 个球。每个球上都标有阿拉伯数字。


图 4 关于 *HMM* 的示例模型

我们先从箱子中随机盲取一个袋子，挑到每一个袋子的概率都是 1/3.然后再从该袋子中盲取一个球，球上的数字永远是 1、2、3、4、5、6、7、8 中的一个，记下小球上的数字，然后放回。重复上述过程多次后，我们会得到一串数字。如若重复 6 次此过程，我们可能会得到这么一串数字序列：1、6、3、5、2、7。而这串数字叫做可见状态链，即可观测变量组成可观测状态链。

在任意时刻，观测变量（袋子）仅依赖于状态变量（哪种颜色的袋子），同时 *t* 时刻的状态仅依赖于 *t*-1 时刻的状态，这就是马尔科夫链。

但是在隐马尔可夫模型中，除可见状态链外，还有一串由隐含变量组成的隐含状态链，在上述例子中即球的序列。比如隐含状态链可能为：*A*、*C*、*C*、*A*、*B*、*C*。

隐马尔可夫型示意图如下所示：

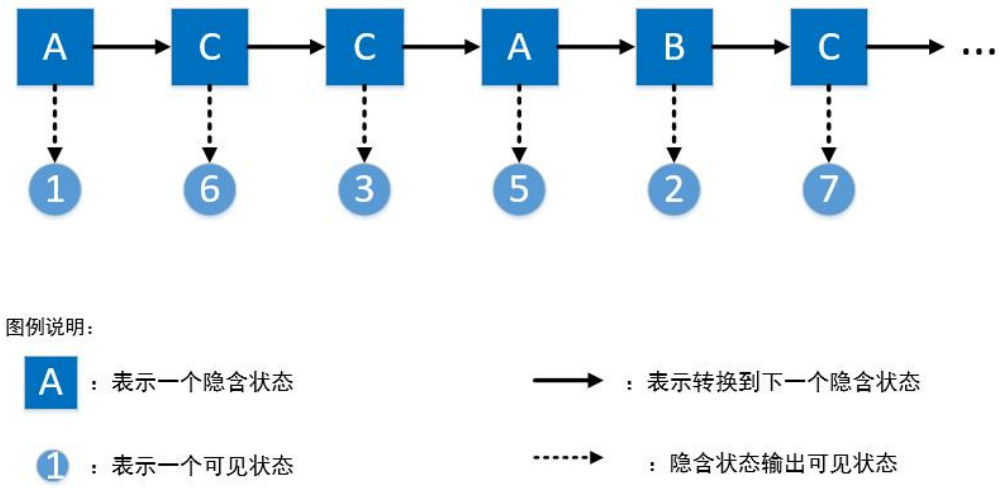


图 5 *HMM* 示意图

在我们这个例子里，若第一次取得红色袋子 *A*，下一次取得红袋子、蓝袋子、黄袋子的概率都是 1/3，即 *A* 的下一个状态是 *A*，*B*，*C* 的概率都是 1/3。同理，若第一次取得黄色袋子 *C*、或蓝色袋子 *B*，则 *C* 或 *B* 的下一个状态是 *A*，*B*，*C* 的概率也是 1/3。

尽管可见状态之间没有转换概率，但是隐含状态和可见状态之间是概率相关的，即输出概率。就上述例子来说，黄色袋子产生 1 的输出概率是 1/8。产生 2，3，4，5，6，7，8 的概率也都是 1/8。

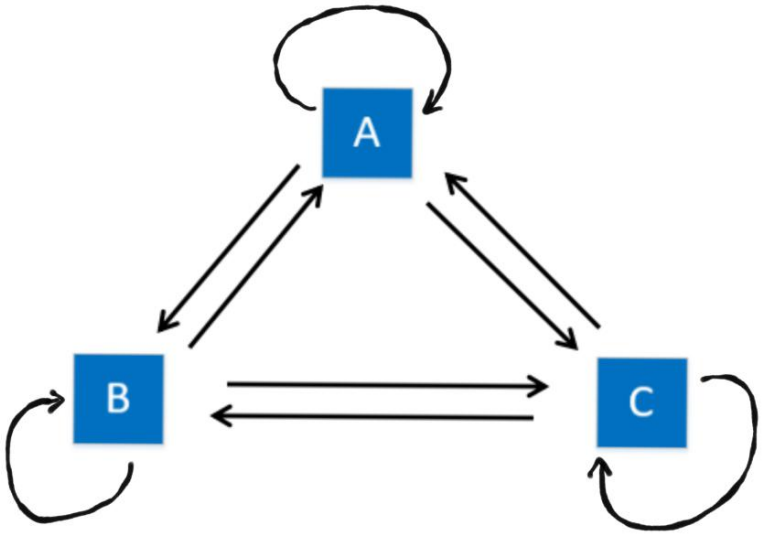


图 6 隐含状态转换关系示意图

如果提前知道所有隐含状态之间的转换概率，以及隐含状态和可见状态之间的输出概率，我们便可以对含隐含参数的马尔可夫过程和与之对应的可观察状态进行建模。

隐马尔可夫模型由两个状态集合和三个概率矩阵组成。

两个状态集合即隐含状态和可观察状态。其中设 *S* 为隐含状态的集合，设 *O* 为可观测状态的集合，*N* 代表隐含状态数目，*M* 代表可观测状态的数目。

$$S = \{s_1, s_2, s_3, ..., s_N\}$$
$$O = \{o_1, o_2, o_3, ..., o_M\}$$

在隐马尔可夫模型中，三个概率矩阵分别是：初始状态概率矩阵 *A*、隐含状态转移概率矩阵 *B*、观测状态转移概率矩阵 *C*。这三个矩阵也被称为隐马尔科夫模型的三要素。

若初始状态概率为：

$$P(S_1) = \mathrm{p}_1, P(S_2) = \mathrm{p}_2, P(S_3) = \mathrm{p}_3, ..., P(S_i) = \mathrm{p}_i$$

其中 *i=1,2,3,...,N*；

则初始状态概率矩阵 *A* 为：

$$A = (A_i), A_i = P(i_1 = s_i), i = 1,2,...,N$$

$A_i$ 表示在时刻 $t$=1 时，隐含状态 $s_i$ 的概率。
隐含状态转移概率矩阵 $B$，描述了状态之间的转移概率：

$$B = [b_{ij}]_{N\times N}$$

其中

$$b_{ij} = p(i_{t+1} = s_j | i_t = s_i), 1 \le i,j \le N$$

表示基于状态 $s_i$ 在 $t$ 时刻时的条件下，在 $t$+1 时刻时，状态为 $s_j$ 的概率。

观测状态转移概率矩阵 $C$：

$$C = [c_{ij}]_{M\times N}$$

$$c_{ij} = p(o_i | s_j), 1 \le i \le M, 1 \le j \le N$$

综上所述，隐马尔可夫模型可以用一个三元组来简洁的表示：

$$\lambda = (A, B, C)$$

### 2.2.4 最大熵模型

“熵”最初是热力学中的一个概念，表示物质系统状态无序程度的一种度量；德国物理学家鲁道夫·克劳修斯提出[5]“熵”的概念，用来表示一个系统空间中分布的均匀程度；香农，最早在信息论中引入了信息熵的概念。简单来说，“熵”是用来表示一个系统状态的的不确定性程度。

设有一个系统 $\xi$，若 $\xi$ 有 $A_1, A_2, A_3, ..., A_n$ 个结果，其中发生 $A_i (1 \le i \le n)$ 的概率为 $p_1, p_2, p_3, ..., p_n$，那么关于 $\xi$ 的信息熵为：

$$H(\xi) = -\sum_{i=1}^{n} p_i \log p_i$$

其中约束条件：

$$\sum_{i=1}^{n} p_i = 1$$

从上述公式中可以看出，不确定值越大，熵值越大。若随机变量是均匀分布的，此时熵值最大；若事件完全确定的，此时的熵值为 0。

$$\mathrm{Max} H(P) : H(p)' = 0$$

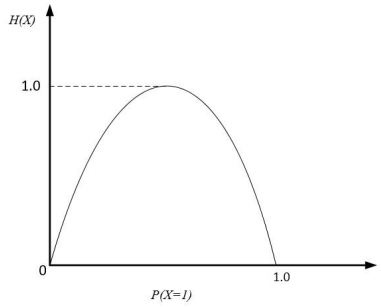


图 7 熵值关于随机变量的分布

为了求出满足最大熵的事件概率 *pi*，我们首先要引入拉格朗日乘子定理：

$$L(p,\lambda,\mu)=H(p)+\sum_{i=1}^{k}\lambda_i(\sum_{x,y}\tilde{p}(x,y)-\tilde{p}(x)p(y|x)f_i(x,y)+\mu(p(y|x)-1)$$

$$\Rightarrow L(p,\lambda)=-\sum_{i=1}^{n}p_i\log p_i+\lambda(\sum_{i=1}^{n}p_i-1)$$

对上式求导得：

$$\frac{\partial L(p,\lambda)}{\partial p_i}=-\log p_i-1+\lambda=0$$

$$p_i=e^{\lambda-1},i-1,2,...,n$$

其中 $p_i$ 含有约束项参数$\lambda$，结合之前求得的约束条件，继续求解得：

$$\sum_{i=1}^{n}\mathrm{p}_i=ne^{\lambda-1}=1$$

$$\Rightarrow e^{\lambda-1}=\frac{1}{n}$$

$$\Rightarrow p_1=p_2=,...,=p_n=\frac{1}{n}$$

最终可得：熵最大时，事件的概率必然满足等概率。

综上所述，最大熵模型，通俗来讲，最大熵模型可以将各种信息综合在一起，即保留事件全部的不确定性，以此尽可能小的来降低风险。最大熵模型的作用是在已知条件下，预测一个随机系统中的分布概率，从而预测出未来可能发生的事件，其本质在只掌握关于未知分布的部分知识时，不对未知信息做假设应选取符合这些知识但熵值最大的概率分布。

最初的最大熵模型的训练方法是迭代算法 *GIS* 和 *IIS* 算法。*GIS*(*Generalized Iterative Scaling*)迭代算法最早是由 *Darroch* 和 *Ratcliff* 在七十年代提出的，八十年代达拉皮垂(*Della Pietra)*对 *GIS* 算法进行了改进，提出了改进迭代算法 *IIS*（*improved iterative scaling*）。

目前 *L-BFGS* 是解无约束非线性规划问题最常用的方法，具有收敛速度快、内存开销少等优点，所以，无论是速度上还是功能上，都优于 *GIS* 和 *IIS* 算法。

### 2.2.5 其他分词方法

（1）基于规则的分词

通过模拟人对句子的理解，达到识别词的效果，基本思想是语义分析，句法分析，利用句法信息和语义信息对文本进行分词。

基于规则的分词方法包括扩充转移网络法、矩阵约束法。

（2）基于字标注的中文分词方法

实质上是构词方法，即把分词过程视作字在字符串中的标注问题。它的一个重要优势在于，它能够平衡地看待词表词和未登录词的识别问题。这使得分词系统的设计大大简化。

通常，基于规则、统计的分词方法的弊端在于都依赖一个事先编制好的词典，

从而通过词表和相关信息来进行分词操作。而基于字标注的分词方法，很好的弥补了这一弊端。

（3）基于人工智能（基于理解）技术的中文分词方法

通俗的讲，这种方法使计算机模拟人脑的工作原理，识别理解文本语义进行分词。这种方法的主要思想是在分词的同时进行句法分析和语义分析，通过对上下文内容所提供的信息进行分词分析，然后对词进行定界，以此来处理分词过程中产生的歧义现象。

该分词方法通常包括三个部分： 分词子系统、句法语义子系统和总控部分。在总控部分的协调下，子系统获得有关文本的句法和语义信息来对分词歧义进行判断，即它模拟了人对句子的理解过程。因此这种方法需要经常更新和维护大量的语言知识信息，且模拟人脑思维的算法复杂收敛速度慢、训练时间长。所以对于基于人工智能的中文自动分词技术还需更深层次的研究。

目前基于人工智能的分词方法主要有：专家系统分词法，神经网络分词法，神经网络专家系统集成式分词法等

## 2.3 词向量

### 2.3.1 词向量的理解

众所周知，计算机是无法直接理解人类自然语言的，所以需要通过词向量[6]对人类自然语言实现数学化。词向量可以用在语言自动识别等其他 *NPL* 任务中，词向量通过把词性相近的词语归到一起，以此来辅助学习算法在处理自然语言。如表，所示给出了部分基于词向量的相似词语。

| | 计算机 | 男生 | 中国 | 学生 | 主席 |
|---|---|---|---|---|---|
| 1 | 计算机系统 | 女生 | 中华人民共和国 | 中学生 | 副主席 |
| 2 | 图形学 | 男学生 | 辽宁大连 | 全国青少年 | 执行主席 |
| 3 | 电子计算机 | 女学生 | 商业联合会 | 全国中学生 | 常务 |
| 4 | 软件工程 | 高年级 | 山东烟台 | 全国大学生 | 委员 |
| 5 | 计算机辅助 | 男女生 | 山东青岛 | 学生 | 秘书长 |
| 6 | 计算机网络 | 校服 | 中国旅游 | 联欢节 | 总干事 |
| 7 | 体系结构 | 班上 | 新疆乌鲁木齐 | 大专生 | 监事会 |
| 8 | 计算机技术 | 年级 | 毽球 | 高中学生 | 理事 |
| 9 | 交互式 | 低年级 | 吉林长春 | 高中毕业生 | 协会主席 |
| 10 | 微型计算机 | 四年级 | 河南郑州 | 青年运动 | 联席 |

表 1　基于词向量的最相似的词语

### 2.3.2 词向量 one-hot representation

*one-hot representation* 是一种很简单的词向量，它用一段很长的向量来表示词。它运用 *One-Hot* 编码的方法，向量的长度为词典的长度，假设长度是 *N*，

*One-Hot* 编码通过寄存器对这 *N* 个状态进行编码，词向量的分量仅含有一个 1，其余全为 0，即只有一位有效。举例如下：
若以下有三个特征：
[“男”，”女”]
[“英国”，”中”，”韩国”]
[“英语”，”俄语”，”汉语”，”韩语”]
为了更好的观测，我们用表格来作比较

| | 学生 *A* | 学生 *B* | 学生 *C* |
|---|---|---|---|
| 性别： | 女 | 男 | 女 |
| 国籍： | 英国 | 中国 | 韩国 |
| 语言： | 英语 | 汉语 | 韩语 |

将上述三个特征换成 *One-Hot* 编码是：
[“01”，”10”]
[“001”，”010”，”100”]
[“0001”，”0010”，”0100”，”1000”]

| | 学生 A | 学生 B | 学生 C |
|---|---|---|---|
| 性别： | 10 | 01 | 10 |
| 国籍： | 001 | 010 | 100 |
| 语言： | 0001 | 0100 | 1000 |

这种词向量在一定程度上可以扩充特征，但是这种方法用在文本上时得到的信息是离散的，在某些情况下过于稀疏，不能很好的体现词语之前的相似性，容易造成维数灾难。

### 2.3.3 词向量 *Distributed Represrntation*

后来 *Hinton* 提出词向量 *Distributed Represrntation*，这种方法可以解决 *one-hot representation* 中词向量维度大的问题。其主要思想是构造一个友词向量组成的向量空间，将每一个向量视为该向量空间中的一个点，这样词与词之间就有了“距离”，因此就可以根据空间上的距离来判断他们之间的词性。这里的向量指的是通过训练模型将所有词都映射成固定长度的短向量。

比如我们将词汇表里的词用"*Strength*","*Masculinity*", "*Femininity*"和"*Age*"4 个维度来表示，假设单词 *Father* 的词向量可能是(0.99,0.99,0.05,0.7)，单词 *Mother* 的词向量可能是(0.85,0.05,0.93,0.6)，单词 *Son* 的词向量可能是(0.45,0.99,0.02,0.5)，单词 *Daughter* 的词向量可能是(0.02,0.01,0.93,0.5)：。

| | Father | Mother | Son | Daughter |
|---|---|---|---|---|
| Strength | 0.99 | 0.85 | 0.45 | 0.2 |
| Masculinity | 0.99 | 0.05 | 0.92 | 0.01 |
| Femininity | 0.05 | 0.99 | 0.02 | 0.93 |
| Age | 0.7 | 0.6 | 0.5 | 0.5 |

图 8 关于示例样本的词向量

经过降维操作，使四维降为三维后，

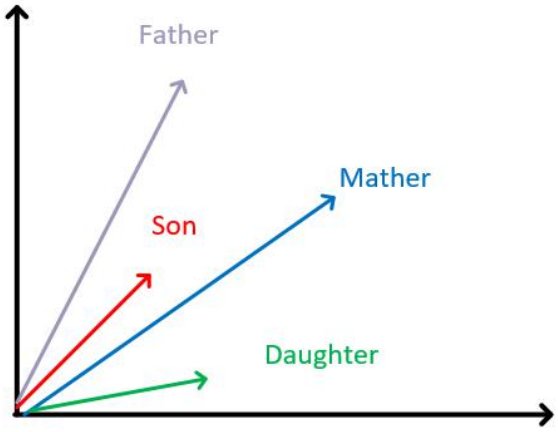


图 9 空间上词向量示意图

通过两个图，从词向量中我们可以很好的观察出，四个单词之间的相关性。

## 2.4 文本相似度

在自然语言处理过程中，经常会涉及到文本之间相似性的问题，上文所述的模型设计也都建立在文本相似度计算的基础上。

若有了文本相似度的计算算法，我们便可以在文本挖掘分析的最后操作中，利用划分法或基于模型的概率方法进行文本之间的聚类分析，以此获取文本中的重点信息。此外，我们还可以利用文本相似度进行模糊匹配操作，即查找某一实体名称的相关名称。文本相似算法的计算流程如下：

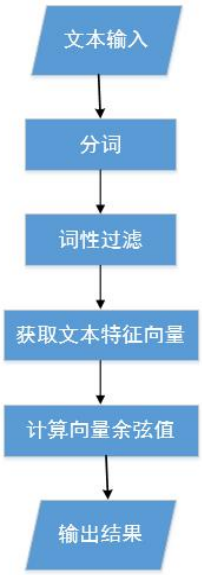


图 9 文本相似度算法计算过程

求文本相似度，即求每个文本特征向量的吻合程度，在文[6]中提供了如下关于文本相似度的算法。

文中认为，词向量在三维空间中，其相似程度度可以通过向量之间的夹角来度量，当夹角为 0 时，则认为相似度最大。

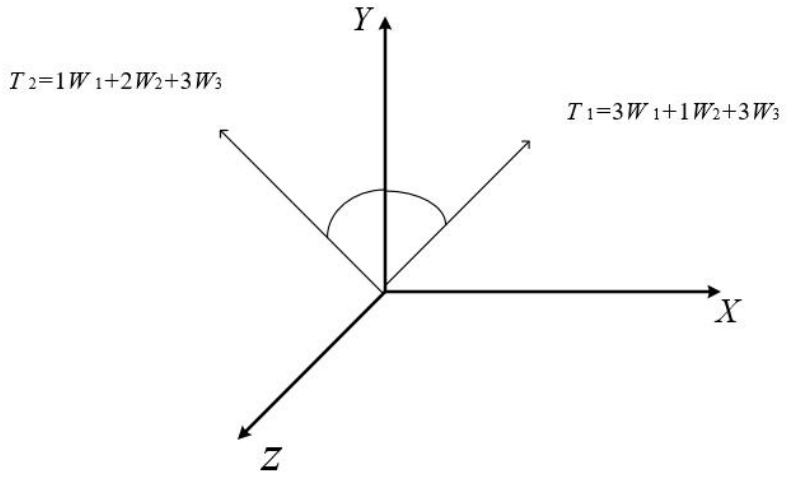


图 10 词向量夹角示意图

因此相似度定义公式如下：

$$\mathrm{sim}=\frac{\sum_{i=1}^{m}\left((a_{1i}\times \mathrm{w}_i\times v_{1i})\times(a_{2i}\times \mathrm{w}_i\times v_{2i})\right)}{\sqrt{\sum_{i=1}^{m}(a_{1i}\times \mathrm{w}_i\times v_{1i}\times v_{2i})^2}\times\sqrt{\sum_{i=1}^{m}(a_{2i}\times \mathrm{w}_i\times v_{1i}\times v_{2i})^2}}$$

其中，$a_{1i}$ 和 $a_{2i}$ 表示词语的个数，$w_i$ 表示词语 $i$ 的权值，$v_{1i}$ 和 $v_{2i}$ 表示两个文本的词向量。

引入 *Word2vec* 词向量后可以提高计算文本相似值的精确度，有效的提高了工作效率，并且拓宽了搜索引擎、文本挖掘、聚类分析等多方面计算机操作应用范围。

## 3 数据预处理

题目附件中所给的数据信息是大量的、杂乱无章的，若直接使用附件中的数据信息，将难以推导出对人们有价值、有意义的信息来作为行动和决策的依据。因此需科学地处理数据，以便能够充分得利用数据资源。数据的预处理过程如下。

### 3.1 数据读取

问题一以附件 2 中的留言内容和一级标签为分类模型

| 留言编号 | 留言用户 | 留言主题 | 留言时间 | 留言详情 | 一级分类 |
|---|---|---|---|---|---|
| 744 | A089211 | 建议增加 A 小区快递柜 | 2019/10/18 14:44 | 我们是 A 小区居民... | 交通运输 |

图 11 附件 2 部分表结构

通过观察表结构，可以看出表内信息含有大量与题目无关的信息，首先需选取有价值的信息并将其转换成便于观察分析、传送或进一步处理的形式。

附件 2 中的一级标签有：城乡建设、党务政务、国土资源、环境保护、纪检监察、交通运输等 15 类标签，根据留言主题的不同，运用 *Python* 将一级标签划分成 15 组，每个标签内对应着相应的留言主题和留言详情。考虑到具体程序可执行性，我们选择留言主题作为训练数据。将每个标签对应的留言主题读取到对应的列表中。一级标签分类对应的 Python 数组如下表。

表 2 级标签分类对应的 Python 数组

| 数组 | 一级标签 |
|---|---|
| *UrbanRuralPlanning* | 城乡建设 |
| *partyAffairs* | 党务政务 |
| *LandrEsources* | 国土资源 |
| *EnvironmentalProtection* | 环境保护 |
| *DisciplineInspection* | 纪检监察 |
| *Transportation* | 交通运输 |
| *EconomicManagement* | 降级管理 |
| *TechnologyInformationIndustry* | 科技与信息产业 |
| *CivilAffairs* | 民政 |
| *RuralAgricultural* | 农村农业 |
| *BusinessTravel* | 商贸旅游 |
| *HealthFamilyPlanning* | 卫生计生 |
| *PoliticalScience* | 政法 |
| *EducationStyle* | 教育文体 |
| *LaborSecurity* | 劳动和社会保障 |

处理结果如下（部分截图）：

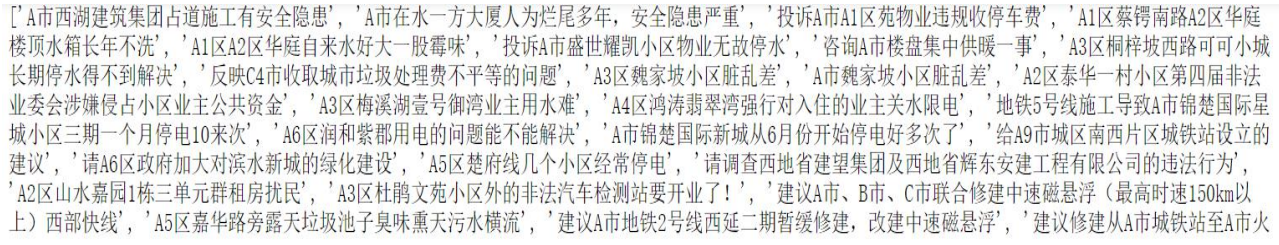
['A市西湖建筑集团占道施工有安全隐患', 'A市在水一方大厦人为烂尾多年，安全隐患严重', '投诉A市A1区苑物业违规收停车费', 'A1区蔡锷南路A2区华庭楼顶水箱长年不洗', 'A1区A2区华庭自来水好大一股霉味', '投诉A市盛世耀凯小区物业无故停水', '咨询A市楼盘集中供暖一事', 'A3区桐梓坡西路可可小城长期停水得不到解决', '反映C4市收取城市垃圾处理费不平等的问题', 'A3区魏家坡小区脏乱差', 'A市魏家坡小区脏乱差', 'A2区泰华一村小区第四届非法业委会涉嫌侵占小区业主公共资金', 'A3区梅溪湖壹号御湾业主用水难', 'A4区鸿涛翡翠湾强行对入住的业主关水限电', '地铁5号线施工导致A市锦楚国际星城小区三期一个月停电10来次', 'A6区润和紫郡用电的问题能不能解决', 'A市锦楚国际新城从6月份开始停电好多次了', '给A9市城区南西片区城铁站设立的建议', '请A6区政府加大对滨水新城的绿化建设', 'A5区楚府线几个小区经常停电', '请调查西地省建望集团及西地省辉东安建工程有限公司的违法行为', 'A2区山水嘉园1栋三单元群租房扰民', 'A3区杜鹃文苑小区外的非法汽车检测站要开业了！', '建议A市、B市、C市联合修建中速磁悬浮（最高时速150km以上）西部快线', 'A5区嘉华路旁露天垃圾池子臭味熏天污水横流', '建议A市地铁2号线西延二期暂缓修建，改建中速磁悬浮', '建议修建从A市城铁站至A市火

图 12 文本预处理结果截图

### 3.2 进行分词和去除停留词

附件 2 所提供的留言信息中，含有大量的语气助词等与一级标签内容无关的字、词。为了更好的精化数据，减少在计算时因数据造成的误差，本文对留言

内容中的语句进行处理，先利用 *Python* 中文分词组件“*jieba*”，将上一过程的到的数据进行遍历分词，再运用“*stopword.txt*”文件去除中文中常见的语气助词等对表达中文语意不起作用的词。

处理结果部分示例如图下（仅对一个语句处理，不代表全部）：

```
#待预测的句子处理
predictstence=[]
def text(line):
            seg = jieba.lcut(line)
            seg = filter(lambda x: len(x) > 1, seg)
            seg = filter(lambda x: x not in stopwords, seg)
            predictstence.append(" ".join(seg))
text('实名举报E市中西医结合医院“科室承包”违规')
print(predictstence[0])
```

```
实名 举报 中西医 医院 科室 承包 违规
```

图 13：关于进行分词和去除停留词的操作结果

由图可得，留言内容为：实名举报 E 市中西医结合医院“科室承包”违规，处理后的结果为：“实名 举报 中西医 医院 科室 承包 违规”，前后对比可见，处理后的样本信息更加精确。

最后，将去除停留词和分词操作结束后的数据加入标签信息，生成训练数据集，并打乱数据。

## 4 机器学习算法构建模型

### 4.1 利用多项式朴素贝叶斯算法建立分类模型

#### 4.1.1 解题思路

首先对样本信息进行预处理，步骤包含读取 *csv* 文件、一级标签分组、分词和去除停用词、生成训练数据、打乱数据。

然后利用 *sklearn* 划分数据集。

最后定义朴素贝叶斯分类模型，其中包括建立词袋模型、将 *x* 转换为词向量、向量化 x、训练模型、预测、 打分。

解题思路流程图如下：

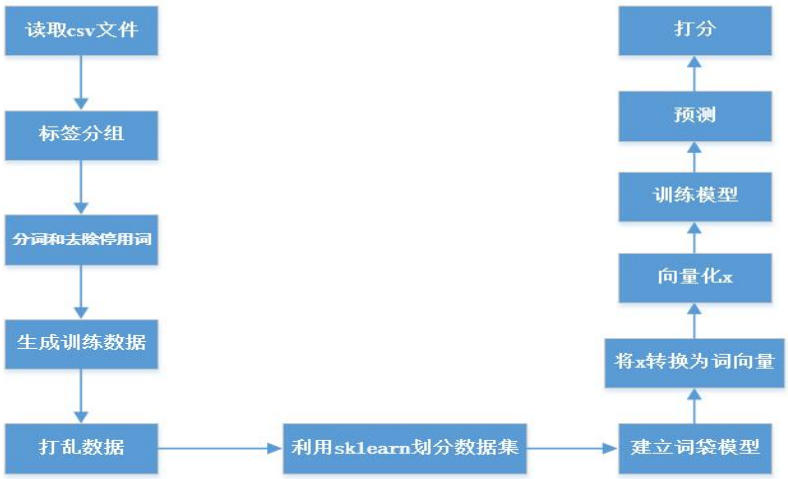


图 14 问题一解题思路流程图

### 4.1.2 模型准备和建立

（1）首先利用 *sklearn.model_selection.train_test_split* 随机划分训练集和测试集。

*sklearn.model_selection* 实现交叉验证，交叉验证用于评估模型的预测性能，尤其是训练好的模型在新数据上的表现。交叉验证本身只能用于评估，但是可以对比不同 *Model* 或者参数对结构准确度的影响。然后可以根据验证得出的数据进行调参，也可以在一定程度上减小过拟合。

*train_test_split* 是交叉验证中常用的函数，功能是从样本中随机的按比例选取 *train_data* 和 *test_data*。

（2）关于建立词袋模型：

*CountVectorizer* 是属于常见的特征数值计算类，是一个文本特征提取方法。对于每一个训练文本，它只考虑每种词汇在该训练文本中出现的频率。 *analyzer* 一般使用默认，可设置为 *string* 类型，如*'word'，'char'，'char_wb'*，还可设置为 *callable* 类型，比如函数是一个 *callable* 类型 *ngram_range* 词组切分的长度范围，待详解 *max_features* 默认为 *None*，可设为 *int*，对所有关键词的 *term frequency* 进行降序排序，只取前 *max_features* 个作为关键词集

假设我们不考虑文本中词与词之间的上下文关系，仅仅只考虑所有词的权重。而权重与词在文本中出现的频率有关。

词袋模型首先会进行分词，在分词之后，统计每个词在文本中出现的次数。

词袋模型有很大的局限性，因为它仅仅考虑了词频，没有考虑上下文的关系，因此会丢失一部分文本的语义。本问题我们不在考虑。

在词袋模型统计词频的时候，我们在本算法中使用 *sklearn* 中的 *CountVectorizer* 来完成。

我们就可以得到该文本基于词的特征，如果将各个文本样本的这些词与对应的词频放在一起，就是我们常说的向量化。定义函数 *fit*，实现向量化，将上步骤中处理好的数据，实现向量化，就可以将实现向量化的数据带入机器学习模型多项式朴素贝叶斯中计算。

（3）处理待预测数据。

1.采用和处理附件 2 中的数据同样的方法，在这不在赘述。

2.利用 *text_classifier.predict()*将处理好的数据代入模型中，进行运算。

3.利用 *text_classifier.score(x_test, y_test)*输出测试集评分。

### 4.1.3 模型求解

选取其中一条留言样本进行测试，结果如图：

```
# 测试
print(text_classifier.predict('市 城镇 职工 养老保险 能 一次 缴费 吗'))
# 测试集评分
print(text_classifier.score(x_test, y_test))
```

```
['LaborSecurity']
0.5241935483870968
```

图 15 程序运行结果部分截图

由测试结果可得，该留言内容属于“*LaborSecurity*”标签，即属于“劳动和社会保障”类内容。且测试集评分为：*0.5241935483870968*。

#### 4.1.4 利用 F-Score 对模型评价

将题目中所给出的 *F-Score* 公式代入模型中，我们综合考虑精确度*(Precision*)和召回率（*Recall*）这 2 个指标，对模型效果进行评估。

$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

$$F - Score = (1 + \beta^2)\frac{Precision \times Recall}{(\beta^2 \times Precision) + Recall}$$

其中β是用来平衡 *Precision,Recall* 在 *F-score* 计算中的权重，通常有以下三种情况：

①如果取 1,表示 *Precision* 与 *Recall* 一样重要；

②如果取小于 1,表示 *Precision* 比 *Recall* 重要；

③如果取大于 1,表示 *Recall* 比 *Precision* 重要。

前面计算的结果得 *F-score*=94.87%.

### 4.2 基于 LTP 的热点问题挖掘模型

#### 4.2.1 解题思路

首先进行导入数据，基于程序可执行性及程序具体执行时间，我们必须选择适合的文本数据，然后进行分词，这里与 *python* 分词库相比，我们选择 *LTP* 中提供的分词模型以便于后继操作匹配，随后进行词性标注，将词性标注后的数据进行命名实体识别，统计名词及地点名词出现个数，依据这个个数选取热点地点及热点人群。

#### 4.2.2 模型的准备——LTP

*LTP*[7]，是哈工大社会计算与信息检索研究中心历时十年于 2006 年 4 月推出了一整套开放中文自然语言处理系统——语言技术平台（*Language Technology Platform*，*LTP*），可以为用户提供高效率精准的中文自然语言处理服务。

语言技术平台(*LTP*) 提制定了基于 *XML* 的语言处理结果表示，并在此基础上提供了一整套自底向上的中文自然语言处理模块，包括中文分词、词性标注、命名实体识别、依存句法分析、语义角色标注 5 项中文处理技术。经过持续研发和推广，如今语言技术平台（*LTP*） 已经成为国内外最具影响力的中文语言处理平台，被多家科研机构和企业使用。

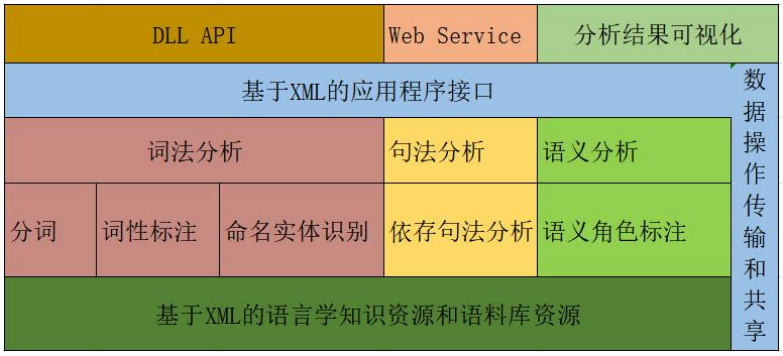


图 16 语言技术平台结构

在解决问题 2 的过程中，本文采用 *LTP* 中的分词技术、词性标注技术和命名实体识别技术来进行中文语言的处理操作。

（1）分词（*Word Segmentation*）

中文语句最基础的处理技术就是中文分词，而汉语中的歧义和未登录词是分词系统所面临的主要难点，因此 *LTP* 采用了基于 *CRF*（*Conditional Random Field*，条件随机域）[8]的分词方法。这种方法能够很好的解决分词歧义问题，也能够解决部分未登录词的问题。该方法是目前分词算法的主流方法。

*CRF* 中的特征函数含有四个参数：需要标注词性的句子 *u*；*i* 用来表示句子中第 *i* 个词；$l_i$ 表示给第 *i* 个单词标注词性的序列；$l_{i-1}$ 表示给第 *i-1* 个单词标注词性的序列。

定义好一组特征函数后，在已知句子 *u* 以及一个标注序列 *l* 后，给每个特征函数 *fi* 赋予一个权重$\lambda_i$，然后对 *l* 评分为：

$$\text{score}(l \mid s) = \sum_{j=1}^{m} \sum_{i=1}^{n} \lambda_j f_j(s, i, l_i, l_{i-1})$$

对上式采用指数化和标准化处理，可以得到关于标注序列的概率值 *P(l|u)*:

$$\text{p}(l \mid s) = \frac{\exp[score(l \mid s)]}{\sum_{l'} \exp[score(l' \mid s)}$$

$$= \frac{\exp[\sum_{j=1}^{m} \sum_{i=1}^{n} \lambda_j f_j(s, i, l_i, l_{i-1})]}{\sum_{l'} \exp[\sum_{j=1}^{m} \sum_{i=1}^{n} \lambda_j f_j(s, i, l_i', l'_{i-1})]}$$

假设 x={x1,x2,...,xn}为观察序列，y={y1,y2,...,y,}为标记序列，那么关于标记序列 y 的联合概率为：

$$\text{p}(y \mid x) = \frac{1}{Z(x)} \exp\left( \sum_i \sum_k \lambda_k t_k(y_{i-1}, y_i, x, i) + \sum_i \sum_k \mu_k s_k(y_i, x, i) \right)$$

又令

$$F_{\text{k}}(y, x) = \sum_{i=1}^{T} f_k(y_{i-1}, y_i, x, i)$$

因此联合概率可以表示为：

$$p(y \mid x)=\frac{1}{Z(x)}\exp\left(\sum_{k}\lambda_k F_k(y,x)\right)$$

相较于其他算法模型，*CRF* 模型定义的数量多，种类丰富；并且在 *CRF* 中，每个特征函数的权重定义灵活，没有限制，可以是任意值。

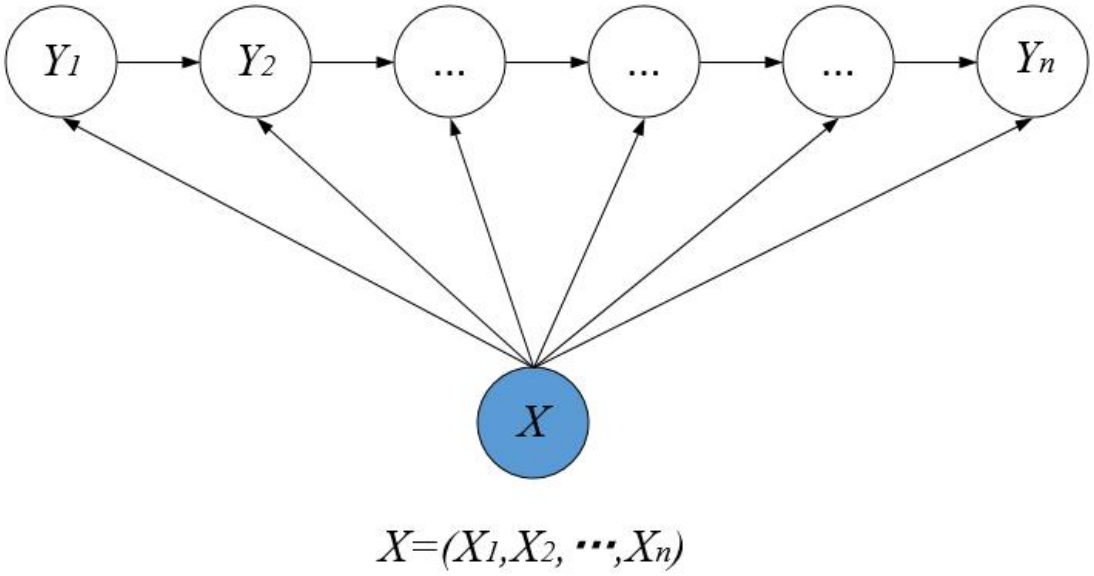


图 17 CRF 模型

（2）词性标注（*POS Tagging*）

词性是词汇最基本的语法属性，词性标注[9]又称为词类标注或语法标注，是指对于句子中的每个词都标注一个正确的词性，也就是确定句子中哪些是名词、动词、形容词或其他词性的过程。经词性标注后的样本信息对于后续操作有很大的便利性，因此词性标注是很多文本语言分析的预处理步骤。同样，词性标注也面临消除语法歧义问题的难点，比如“*book*”，作为名词它的词义是书籍，但作为动词时它的词义是预定。

词性标注的算法基本可以利用分词的方法，其中主流算法是基于最大熵的词性标注和基于隐马尔可夫模型的词性标注。

早期的词性标注算法采用基于隐马尔可夫模型[10]等模型，但隐马尔可夫模型需要做很强的独立假设，这便导致最终结果的精确度不高。在 *LTP* 中，选用最大熵马尔科夫模型为代表的判别模型，使用支持向量机作为基本分类器，不仅解决了假设独立特征的麻烦，也进一步提高了词性标注的精确度。前文已详细讲解隐马尔可夫模型和最大熵模型的原理，此处不再赘述。

同时在 *LTP* 中还引入了汉字的偏旁特征，有效的解决了未登录词的问题，也提升了系统的泛化能力。*LTP* 使用的是 863 词性标注集，其各个词性含义如下。

表 3 LTP 的词性标注集

| *Tag* | *Description* | *Example* | *Tag* | *Description* | *Example* |
|---|---|---|---|---|---|
| *a* | *adjective* | 美丽、漂亮 | *ni* | *organization name* | 保险公司 |
| *b* | *other noun-modifier* | 大型 | *nl* | *location noun* | 城郊 |
| *c* | *conjunction* | 虽然 | *ns* | *geographical name* | 北京 |
| *d* | *adverb* | 很，非常 | *nt* | *temporal noun* | 近日 |
| *e* | *exclamation* | 哎 | *nz* | *other proper noun* | 诺贝尔 |
| *g* | *morpheme* | 甥 | *o* | *onomatopoeia* | 哗啦 |
| *h* | *prefix* | 伪 | *p* | *preposition* | 在 |
| *i* | *idiom* | 百花齐放 | *q* | *quantity* | 个 |
| *j* | *abbreviation* | 公检法 | *r* | *pronoun* | 我们 |
| *k* | *suffix* | 界 | *u* | *auxiliary* | 的 |
| *m* | *number* | 一 | *v* | *verb* | 跑 |
| *n* | *general noun* | 香蕉 | *wp* | *punctuation* | ，。！ |
| *nd* | *direction noun* | 右侧 | *ws* | *foreign words* | CPU |
| *nh* | *person name* | 汤姆 | *x* | *non-lexeme* | 葡 |
| | | | *z* | *descriptive words* | 匆匆 |

（3）命名实体识别（*NE*，*Named Entity Recognition*）

命名实体识别有称作专名识别，是指识别文本中含有特殊意义的单词，例如地名、人名、其他专有名词等。命名实体识别通常是信息挖掘的第一步，所以广泛应用在人类自然语言的处理操作。目前 *NE* 主要有两类算法，分别是基于规则和基于统计的方法。

*LTP* 将基于规则和基于统计这两种方法结合起来，以 *MEMM*[11]统计模型为基础，提出借助英文 *NE* 系统，自动生成中文 NE 训练语料的方法，大大扩展了系统的规模，与此同时也提升了系统的识别能力。

如今的 *NE* 识别，多为人名、地名、日期等有限类别。

### 4.2.3 模型的建立和求解

通过 *Python*3.6，将附件 3 中要进行问题分析的样本信息，根据不同的留言主题添加到字典（数组）中。部分结果如下图所示。

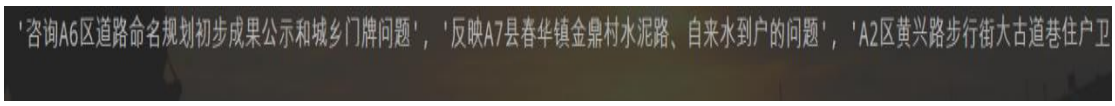


图 18 部分结果截图

问题 2 要求根据地点或特定人群问题的留言进行分类，以此来分析热点问题，所以需要对留言内容进行文本划分、信息处理等操作，本文采用 *LTP* 中的分词、词性标注和命名实体识别技术，并在此基础上建立热点信息提取模型。

对留言内容的分词处理，通过 *Python*3.6 引入 *LTP* 中的分词模型文件："*ltp_data_v3.4.0\cws.model*"，以此来对保存到词典内的留言语句进行切分操作。部分结果如下图所示。

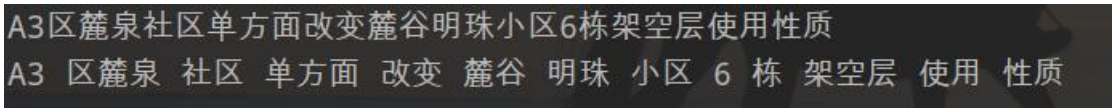


图 19 部分结果截图

分词操作结束后进入词性标注阶段，为上述过程切分出的单词标注词性一遍后续的文本识别操作。

同样，本文通过引入 *LTP* 中的词性标注文件：“*ltp_data_v3.4.0\pos.model*”构建词性标注模型。部分结果如下图所示。


图 20 部分结果截图

文本处理的最后一步命名实体识别部分，引入 *LTP* 中的命名实体识别模型文件：“*ltp_data_v3.4.0\ner.model*”,对标有专有名称或特殊意义的字词进行识别挑选、归类统计。部分统计结果如下图所示。


图 21 部分结果截图

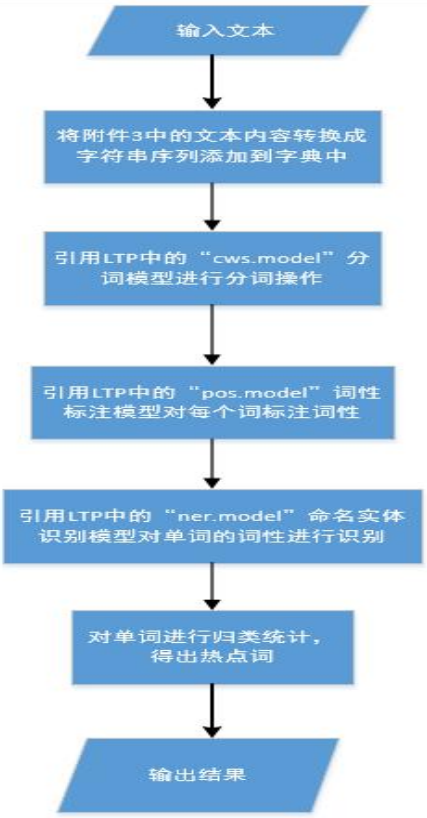


图 22 建立模型流程图

### 4.2.4 模型结果

基于程序可执行性，导入附件 3 中所有留言主题，调用 *LTP* 上述模型，统计出现过的所有名词及地点名词，见附录编写程序，统计各个名词及地点名词出现过的个数，基于统计出的个数，我们选取 业主、公司、幼儿园、新城、梅溪湖这五个中心词作为排名前五的热点人群及地点，选取带有中心词的留言主题得出热点问题留言明细表，其中各个地点名词及人群名词在所有留言主题所占比例作为热度指数。

## 4.3 基于 *Word2vec* 答复意见评价模型

在新的文本挖掘的需求和趋势下，人类自然语言的处理因此受到了高度重视。随着计算机领域的发展，信息检索以及数据挖掘等应用需求越来越大，而计算机的语言处理技术面临越来越困难的挑战。在时代和社会的发展趋势下，为了更好的挖掘文本信息，为自然语言的后续操作提供精确便利的准备，*Google* 公司推出了 *Word2vec*[6]。

*Word2vec* 是 *Google* 公司推出的用于训练词向量的软件工具，将字词转换成多维向量，通俗的讲，就是获取词向量的相关模型。*Word2vec* 可以根据已有的语言资料库，通过训练模型将单词转成向量表达式，以此来表示词语之间的关系。*Word2vec* 主要依赖 *Skip-gram* 模型和连续词袋（*CBOW*）模型。

### 4.3.1 解题思路

对于要计算相似度的两个句子，首先进行与第一问中相似的去除停用词处理，用用 gensim 导入 Word2eve 模型，并引入百度提供的百度百科中文语料。将处理好的两个句子计算词向量，计算相识度。

### 4.3.1 模型准备

（1）神经概率语言模型：

*Word2vec* 模型可以说是简单化的神经网络，如图所示是一个神经网络的结构示意图，它包括输入层（*Input Layer*）、投影层（*Projection Layer*）、隐藏层（*Hidden Layer*）以及输出层（*Output Layer*）,其中 *W* 是投影层和隐藏层之间的权值矩阵，*U* 是隐藏层和输出层之间的权值矩阵。*p* 和 *q* 是偏置向量。

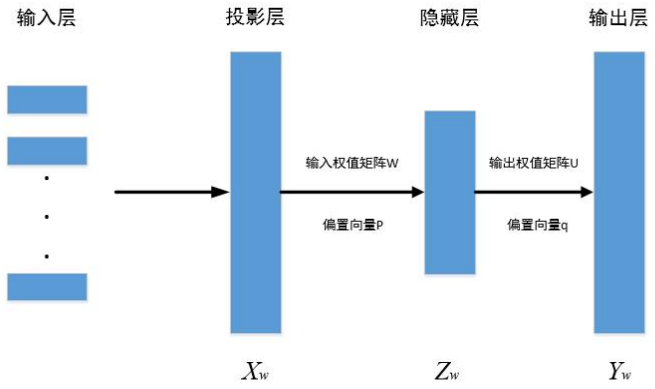


图 23 神经网络结构示意图

隐藏层 *Hidden Layer* 是线性的单元，没有激活函数。输出层 *Output Layer* 维度与输入层 *Input Layer* 的维度一样，用的是 *Softmax* 回归。用 *w* 表示词语，用 *Context(w)*来表示关于词 *w* 周边词的集合，对于语料库中的任意一个词 *w*，*Context*（*w*）的取值词 *w* 前面的 *n-1* 个词，形成是二元对*(Context(w),w)*，就是一个训练样本。那么样本二元对*(Context(w),w)*在通过上图神经网络时的运算如下：

$$\begin{cases} z_w = \tanh(Wx_w + p) \\ y_w = U\mathrm{z}_w + q \end{cases}$$

经过计算可得一个长度为 *N* 的向量：

$$\mathrm{y}_w = (y_{w1}, y_{w2}, \ldots, y_{wN})^T$$

所以，关于 $y_w$ 中的分量 $y_{mi}$ 的概率为：

$$\mathrm{p}(w \mid Context(w)) = \frac{e^{y_{wi}}}{\sum_{i=1}^{N} e^{y_{wi}}}$$

由上述算式和神经网络结构图可以看出，整个模型的大量计算部分，主要集中在隐藏层和输出层之间，主要是向量运算。

神经概率语言模型与 *n-gram* 相比，在神经概率语言模型中，通过词向量来描述词语之间的相似性，因为是基于词向量的模型，有：

$$p(w \mid Context(w)) \in (0,1)$$

所以神经概率语言模型自带平滑化功能。

（2）*CBOW* 模型

在本章开头我们说过 *Word2vec* 主要是依赖 *CBOW*(*Continuous Bag-of-Words* 与 *Skip-Gram* 这两种模型来定义数据的输入和输出

下图是文[5]中的作者 *Tomas Mikolov* 给出的 *CBOW* 模型示意图和 *Skip-gram* 模型示意图。

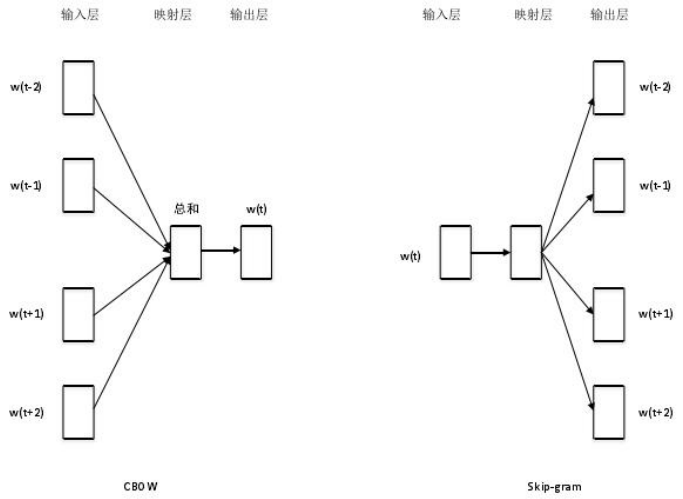


图 24　*CBOW* 和 *Skip-gram* 的模型示意图

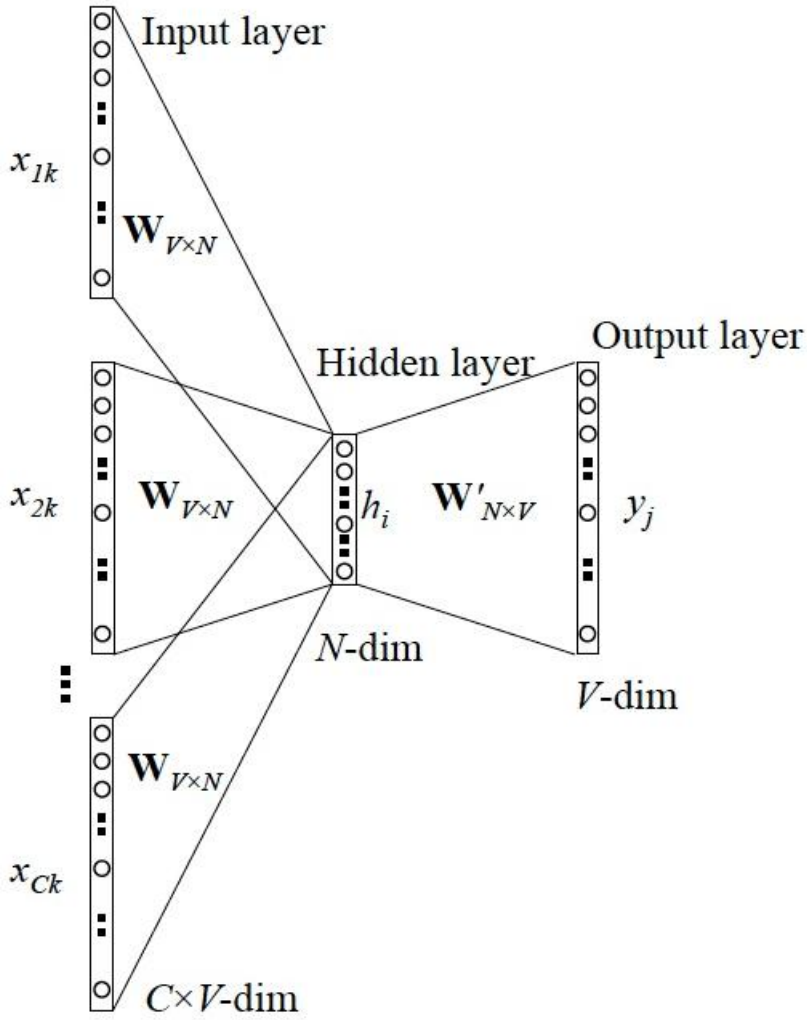


图 25　*CBOW* 模型的训练结构图

在上图中，输入层：这里首先设单词的向量空间 *V*，上下文共有 *C* 个单词，所以输入层包含了上下文词的集合。

每个词都乘以输入初始化权重矩阵 *W*，得到词向量进入投影层。

在投影层将这 *C* 个词向量进行累加求和并求平均操作。

经过投影层处理过的隐层向量乘以输出权重矩阵 *W'*。

在输出层得到的词向量经过激活函数处理得到 *V-dim* 概率分布 。

因为 *CBOW* 模型是基于 *Hierarchical Softmax* 来设计的，而 Hierarchical *Softmax* 是 *Word2vec* 中的一项关键技术，所以 CBOW 模型中的输出层采用 *Huffman* 树结构，可以更准确的对数据进行分类。

（3）Skip-gram 模型

*Skip-gram* 模型与 *CBOW* 模型网络结构图一样，也包括了输入层、投影层以及输出层。

在 *Skip-gram* 的输入层中只含有样本中心词的词向量，不包含上下文单词的集合；*Skip-gram* 的投影层是个恒等投影，输出层和 *CBOW* 模型的输出层一样，采用了 *Huffman* 数结构。

对于 *Skip-gram* 模型，训练此模型的目的是通过获得的词向量来预测该单词上下文中的相邻单词。

假设已知词 *w*，若要以该词为中心，对其上下文进行预测，*Skip-gram* 模型中将其定义为：

$$p(Context(w)\,|\,w)=\prod_{u\in Context(w)}p(u\,|\,w)$$

从直观上理解，*Skip-Gram* 是给定 *input word* 来预测上下文。

假设我们选取“中国”为输入词，我们设定选取词的数量为 2，即在语料库中选取输入词上下文单词的数量为 2。最终模型输出，类似于“英国”、“韩国”这一类相关词的概率远远高于“巾帼”一类的非相关词，以此来训练出我们需要的权重矩阵。

假设选定句子“*The quick brown fox jumps over lazy dog*”为原始文本，设定窗口为“2”（*window_size*=2），也就是说我们仅选输入词前后各两个词，分别与输入词进行组合，下图中，蓝色代表输入词，方框内代表位于窗口内的单词。

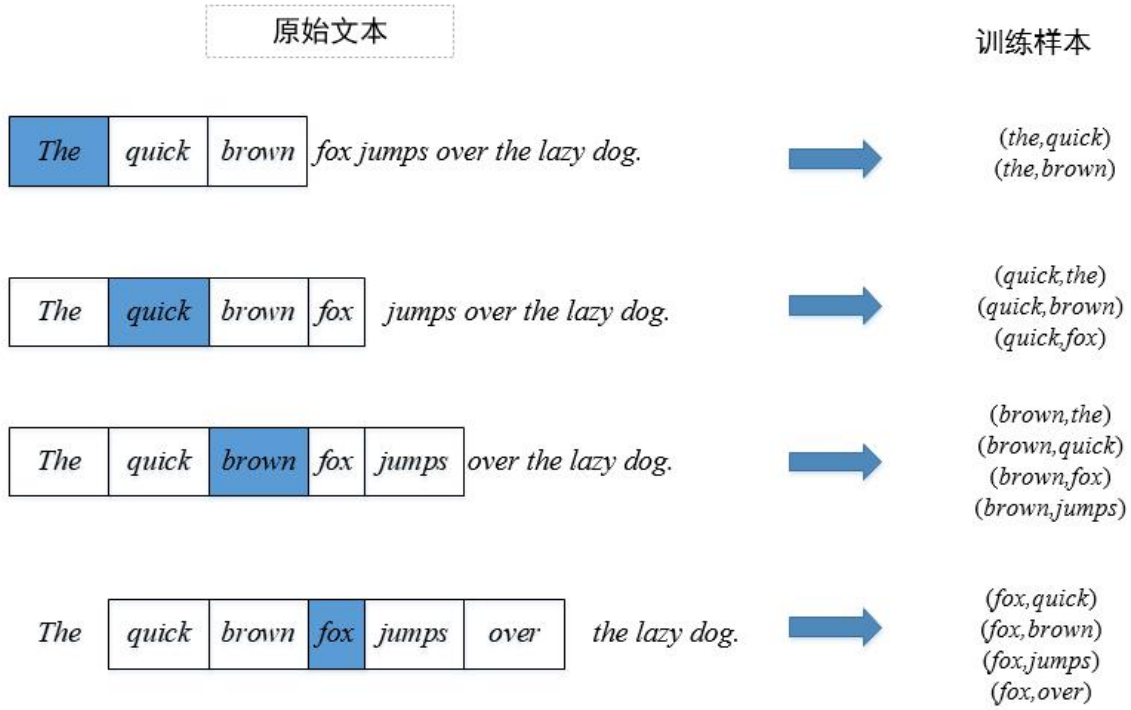


图 26 *Skip-gram* 模型训练样本的过程

上图所示内容，清楚的展示了原始文本生成训练词语集的过程。

### 4.3.2 模型的建立和求解

首先导入 *gensim* 包，*gensim* 开源包为使用 *Word2vec* 提供了一个 *Python* 版的实现。导入 gensim 包后，我们还不可以直接进行 *Word2vec* 的操作，因为 *Word2vec* 默认的分词符是空格，中文中不存在空格，所以我们通过导入 *Python* 的中文分词包 *jieba* 以此来进行中文分词操作。然后导入百度百科中文语料作为训练集。

中文分词操作结束后，接下来便是模型中最重要的算法部分。本文使用

*gensim* 中的 *Word2Vec* 模型进行文本训练的操作。

### 4.3.3 模型结果

例如：

2019 年 4 月以来，位于 A 市 A2 区桂花坪街道的 A2 区公安分局宿舍区出现了一番乱象，该小区的物业公司美顺物业扬言要退出小区，因为小区水电改造造成物业公司的高昂水电费收取不了(原水电在小区买，水 4.23 一吨，电 0.64 一度）所以要通过征收小区停车费增加收入，小区业委会不知处于何种理由对该物业公司一再挽留，而对业主提出的新应聘的物业公司却以交 20 万保证金，不能提高收费的苛刻条件拒之门外，业委会在未召开全体业主大会的情况下，制定了一高昂收费方案要各业主投票，而投票不采用投票箱只制定表格要物业公司人员这一利害关系机构负责组织，对投票业主隐私权没有任何保护，还对投反对票的业主以领导做工作等方式要求改变为同意票，这种投票何来公平公正公开，面对公安干警采用这种方式投票合法性在哪？'

现将网友在平台《问政西地省》栏目向胡华衡书记留言反映“A2 区景蓉花苑物业管理有问题”的调查核实情况向该网友答复如下：您好，首先感谢您对我们工作的信任和支持，关于您在平台栏目给胡华衡书记留言，反映“A2 区景蓉花苑物业管理有问题”的情况已收悉。现将我们调查处理情况答复如下：经调查了解，针对来信所反映的“小区停车收费问题”，景蓉华苑业委会于 2019 年 4 月 10 日至 4 月 27 日以“意见收集方式”召开了业主大会，经业委会统计，超过三分之二的业主同意收取停车管理费，在业主大会结束后业委会也对业主提出的意见和建议进行了认真疏理归纳并进行了反馈，业委会制定的停车收费标准不高于周边小区价格。针对来信所反映的“物业公司去留问题”，5 月 5 日下午，辖区桂花坪街道牵头组织社区、物业公司、业主委员会、业主代表的会议，区住房和城乡建设局也参加了会议。在综合各方面的意见后，辖区桂花坪街道、区住房和城乡建设局已要求业委会依法依规召开业主大会，根据业主大会的表决结果再执行相应的程序。再次感谢您对我区工作的理解和关心。2019 年 5 月 9 日'

将上述两个句子进行中文分词操作及停用词去除后。得出两个句子的相似度为 0.924254989201908。

## 5 结论

### 5.1 总结

本文是对“智慧政务”文本挖掘应用的研究和总结。内容可大体概括为四个方面：文本分类、中文分词、*LTP*、*Word2vec*。

针对问题一，本文以文本分类为研究方向，采用了多项式贝叶斯算法作为建立模型的理论基础，详细介绍了多项式贝叶斯分类模型的理论、特点、运算等。通过实践证明以及 *F-Score* 方法的评分，都表明多项式贝叶斯算法在文本分类方面有着较高的准确率。

针对问题二，本文以语言技术平台 *LTP* 为研究方向，深度研究了 *LTP* 所涵盖的分词技术、词性标注技术以及命名实体识别技术。*LTP* 的分词技术主要采用了 *CRF* 算法，*CRF* 在中文分词技术方面是主流算法；*LTP* 的词性标注技术建立在最大熵马尔科夫模型的基础上，将最大熵模型与马尔可夫理论相结合，大大提高了词性标注的准确率；LTP 的命名实体识别技术采用了统计和规则相结合的技术，以 *MEMM* 为统计模型，该方法即提高了系统识别实体词性的能力，也拓宽

了系统识别的范围。

针对问题三的答复意见评价，该题引用了 *Word2vec*，以空间词向量和文本相似度为研究方向，通过建立向量空间模型，在此基础上分析文本相似度，最后可以得出文本内容之间的相关性。

通过实践证明，本文建立的模型均具有一定的可行性和适用性。

## 5.2 展望

自然语言的处理以及模型的智能型都面临着巨大的挑战。还有以下工作需要完善：

（1）首先本文建立的模型，模块独立性太强，前后不融合，甚至解决一个问题要用到两个软件。

（2）需进一步提高文本相似度。由于汉语存在多意性的的情况、语料库的限制、训练模型的短板，最终得出的词向量精确度不高，会使计算产生误差，因此无法准确的得出关于文本内容相似程度的指数。

（3）热点问题挖掘模型不够灵活，针对不同规模的样本和问题，需要用不同的挖掘技术。后续工作需在模型中加入只能算法，使文本挖掘模型能够针对不同类型提供多样化的服务。

“智慧政务”能够使办公实现数字化、智能化、网络化，既能够为民众提供个性化服务，也可以辅助政府作出精准的计划决策，除了需要不断提高“智慧政务”的性能外，还要考虑数据安全等各方面的问题，总支“智慧政务”是一个值得持续关注并加深研究的问题。

# 参考文献


[1]阿曼，朴素贝叶斯分类算法的研究与应用，大连理工大学，2014 年 6 月
[2]李方，刘琼荪，基于改进属性加权的朴素贝叶斯分类模型[J],计算机工程与应用，2010（4）：132-133
[3]杨淦，基于条件随机场模型的中文分词系统研究与实现，重庆大学计算机学院，20115 年 4 月
[4]Rabiner L R.A tutorial on hidden Markov models and selected applications in speech recognition[J].PROCEEDINGS OF THE IEEE,77(2),P,1989:257-286.
[5]Jaynes E T.Notes on Present Status and Future Prospects[M].Maximum Entrupy and Bayesian Methods,Jr.Grandy W T,Schick L H,Springer Netherlands,1991:43,1-13.
[6]吴多坚，基于 word2vec 的中文文本相似度研究与实现，西安电子科技大学，2015 年 12 月
[7]刘挺、车万翔、李正华，语言技术平台，哈尔滨工业大学社会计算与信息检索研究中心，2011 年 11 月
[8]John Lafferty, Andrew Mccallum, Fernando Pereira Conditional Random Fields. Probabilistic Models for Segmenting and Labeling Sequence Data[C]//Pro-ceedings of ICMI 2001: 282-289
[9]张梅山，邓知龙，车万翔等，统计与词典相结合的领域自适应中文分词[C]//第十一届全国计算语言学学术会议，2011.8
[10]王丽杰，车万翔，刘挺，基于 SVMTool 的中文词性标注[J],中文信息学报，2009,23(4)：16-21
[11]Guodong Zhou, Jian Su. 2002. Named entity recognition using an Hmm-based chunk tagger C//Pro-ceedings of the 40th Annual Meeting of the Association of Comparative Linguistics(ACL): 473-480
[12]Hai Leong Chieu, Hwee Tou Ng. 2002. Named En-tity Recognition: A Maximum Entropy Approach Using Global Information[ C1//Proceedings of the 19th, International Conference on Computational Lingttics (COLING): 190-196
[13]Burr Settles. 2004. Biomedical Named Entity Recognition Using Conditional Random Fields and Rich Feature Sets C1//Proceedings of COLING 2004. the International Joint Workshop on Natural Language Processing in Biomedicine and its Applications (NI PBA). Geneva, Switzerland
[14]Nianwen Xue. Chinese Word Segmentation as Charac-ter Tagging[J], International Journal of Computational Linguistics and Chinese Language Processing, 2003,8(1):29-48
[15]John Lafferty, Andrew Mccallum, Fernando Pereira Conditional Random Fields. Probabilistic Models for Segmenting and Labeling Sequence Data[C]//Pro-ceedings of ICMI 2001: 282-289