“智慧政务”中的文本挖掘应用

摘要

近年来，随着微信、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量不断攀升，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挑战。因此，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，对提升政府的管理水平和施政效率具有极大的推动作用。

对于问题1，首先对数据进行去重预处理，接着利用北大开源的一款基于Python的PKUSeg 中文分词工具对留言主题、留言详情以及标签进行分词；然后进行分段分词匹配，即各级标签分别与留言进行匹配；最后，考虑到有的留言无法与标签进行匹配，会将该类留言加入人工检查步骤。需要注意的是有的留言涉及到多个部门，会将该类分类到多个以及标签。

对于问题2，根据附件三的信息特征，将留言类别定义为两大类：特定地点和特定人群，定义合理的热度评价指标，同时给出分词热度，用Word2vec生成热点词汇的词向量，用于词典的构建。利用发散性思维，再分别对筛选出来的结果按照时间、点赞数、反对数三个方面对其进行多方面系统地统计，结合图表对附件三中的留言信息进行归类总结。

对于问题3，根据问题1和问题2的研究结果，首先分析答复与留言的相关性，主要是通过答复和留言的分词的相关性，来的出答复的相关性、完整性和可解释性；除了要考虑答复的相关性、完整性和可解释性，本文还将留言与答复之间的时间差纳入到答复意见的评价标准中。

关键字：中文分词；标签分类；PKUSeg；Python；Word2vec

Application of Text Mining in "Smart Government Affairs"

Abstract

In recent years, with the online inquiry platforms such as WeChat, Weibo, Mayor's Mailbox, and Sunshine Hotline gradually becoming important channels for the government to understand public opinion, gather public wisdom, and consolidate public opinion, the amount of text data related to various social conditions and public opinion has continued to rise, giving The work of related departments that mainly rely on manual work to divide messages and organize hot spots has brought great challenges. Therefore, the establishment of a smart government affairs system based on natural language processing technology has become a new trend in the innovation and development of social governance, which has greatly promoted the improvement of government management and governance efficiency.

For question 1, first remove duplicate data, use Python's open source Python-based PKUSeg Chinese word segmentation tool to segment the message subject, message details, and tags; then perform segmentation word matching, that is, the tags at all levels are matched with the message respectively; finally, considering Some messages cannot be matched with tags, and they will be added to the manual inspection step. It should be noted that some messages involve multiple departments, which will be classified into multiple categories and tags.

For question 2, according to the information characteristics in Annex III, the message categories are defined into two categories: specific locations and specific populations, and reasonable thermal evaluation indicators are defined. At the same time, word segmentation heat is given. Word2vec is used to generate word vectors for dictionary construction. in. Using divergent thinking, the screened results are systematically and statistically summarized in three aspects, including time, number of likes, and number of oppositions, and the message information in Annex III is summarized in combination with the chart.

For question 3, based on the research results of question 1 and question 2, first analyze the relevance of the response and the message, mainly through the relevance of the response and the participle of the message, the relevance, completeness and

interpretability of the response; In addition to considering the relevance, completeness, and interpretability of the response, this article also incorporates the time difference between the message and the response into the evaluation criteria of the response.
Keywords: Chinese word segmentation; PKUSeg; Python; Word2vec

目 录

1.挖掘目标

本次建模目标是利用问政平台逐系统发布的各类社情民意相关文本数据，利用中文分词工具PKUSeg对相关数据进行分词、利用Word2vec生成相关热点关键词，达到以下三个目标：

(1)通过PKUSeg对留言主题、留言详情以及标签进行分词，利用分段分词匹配算法将留言进行标签匹配，以便后续将群众留言分派至相应的职能部门处理。

(2)利用Words2vec生成热点词汇的词向量，对筛选出来的结果按照时间、点赞数、反对数三个方面对其进行多方面系统地统计，对热点问题进行挖掘，以便利于相关部门进有针对性的处理。

(3)根据上述理论提出对答复意见进行相关性、完整性和可解释性等角度的评价方案，将回复速度纳入到评价方案中。

总而言之，上述三个目标是为了将自然语言处理技术融合到只会政务系统中，符合社会治理创新发展的新趋势，将依靠人工进行留言划分和热点整理的相关部门减轻工作压力提升，政府的管理水平和施政效率。

2.分析方法与过程

该课题的分析方法和过程大致上分为以下几步：

(1)数据预处理，在题目给出的数据中，主要是对标签、留言主题、留言详情和答复意见进行中文分词，同时依据不同需求建立相关字典；

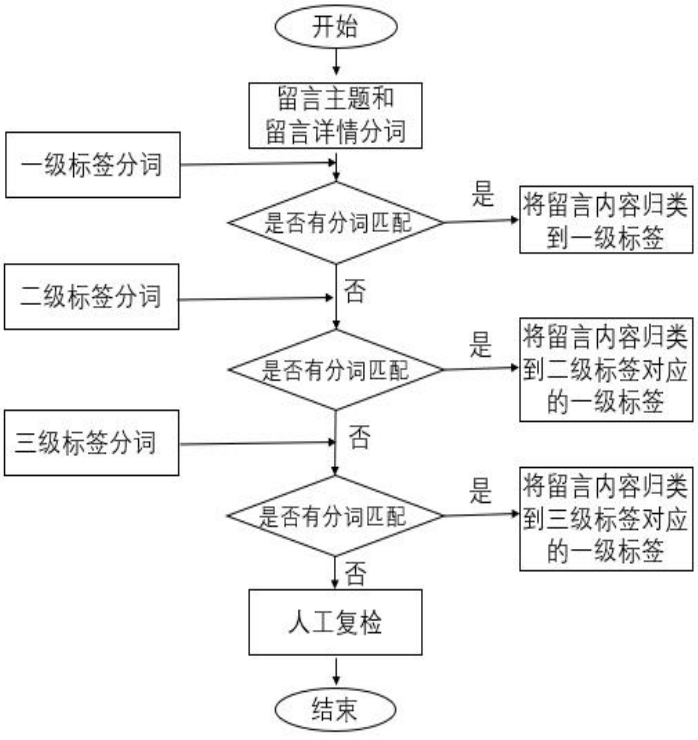
(2)数据处理，在数据预处理之后，需要将这些词语转换为词向量，以便于进行挖掘分析使用，使用Word2vec生成相应的热点词汇，将热点词汇进行分类、匹配、归纳等操作；

(3)利用上述步骤的结果建立出来的字典，将留言和回复与其进行比对，总结出热点事件、热点人群、热点时间，为群众民意的表达提供更好的表现。

本课题依据上述方法与过程来实现“智慧政务”中的文本挖掘应用。

2.1 问题 1 分析方法与过程

2.1.1 数据去重后流程图



2.1.2 中文分词

在中文分词中使用PKUSeg进行分词，这一工具包有如下三个特点：

(1)高分词准确率。相比于其他的分词工具包，当使用相同的训练数据和测试数据，PKUSeg 可以取得更高的分词准确率；

(2)多领域分词。不同于以往的通用中文分词工具，此工具包同时致力于为不同领域的数据提供个性化的预训练模型。根据待分词文本的领域特点，用户可以自由地选择不同的模型。而其他现有分词工具包，一般仅提供通用领域模型；

(3)支持用户自训练模型。支持用户使用全新的标注数据进行训练。

首先利用PKUSeg将各级标签进行分词，并依据该结果建立字典，之后基于字典的方式将留言主题和留言详情待与词典进行匹配，当发现与词典中登录的词语一致时，匹配成功，进行切割。该方法使用广泛、易于实现、耗时短、切分速度快。但这种方法对词典的依赖程度较高，并且不能实现更深层次的操作。目前分词系统一般将该方法用于对文本的初分，之后利用其它手段提升分词的质量。

为了解决上述问题，将上述结果进行基于统计的方法进行计数。即将留言主题和留言详情中的分词进行统计，分词出现的次数越高与字典中的词汇越匹配，以此解决基于词典的分词方法中无法统计词数的问题。

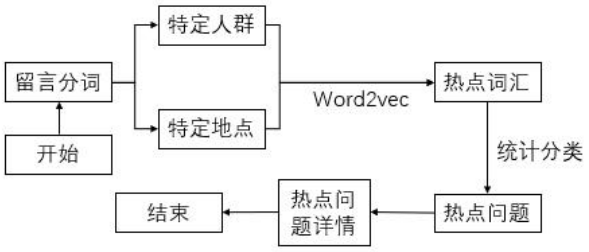
2.1.3 多段匹配原则

在本课题中采用多段匹配原则，在对留言进行分词后会分别与一级标签、二级标签和三级标签进行匹配，将匹配度相加，如果匹配度相似或超过30%的相似度将会输出对应的一级标签，如果不匹配将会进入人工审核。这样做会尽量避免将留言发送到错误的部门。

需要注意的是本案例中会出现一个留言涉及多个部门的问题，所以会出现一个留言有多个标签的问题，如果留言含有正确的一级标签，本留言会被视为正确分类。其次是案例中可能出现留言无法判定标签，会将该类留言纳入到人工核查通道中。

2.2 问题 2 分析方法与过程

根据附件三的信息特征，将留言反应的内容分为两大类：特定地点和特定人群，定义合理的热度评价指标，同时给出分词热度，用Word2vec生成热点词汇的词向量，用于词典的构建。构建字典之后按照时间、点赞数、反对数三个维度对其进行多方面系统地统计，得出相应的热点问题，加大政务平台的效率。流程图如下所示。



2.2.1 Word2vec 的原理

2013年，Google团队发表了word2vec工具。word2vec工具主要包含两个模型：跳字模型（skip-gram）和连续词袋模型（continuous bag of words，简称CBOW），以及两种高效训练的方法：负采样（negative sampling）和层序softmax（hierarchical softmax）。Word2vec词向量可以较好地表达不同词之间的相似和类比关系。近年来，词向量已逐渐成为自然语言处理的基础知识。

NLP（自然语言处理）里面，最细粒度的是词语，词语组成句子，句子再组成段落、篇章、文档。词语，是符号形式的（比如中文、英文、拉丁文等等），所以需要把他们转换成数值形式，就叫词嵌入（word embedding)，而 Word2vec，就是词嵌入(word embedding)）的一种。即把一个词语转换成对应向量的表达形式，来让机器读取数据。

统计语言模型为计算一段文本序列在某种语言下的词向量问题提供了一个基本的解决框架，对于一段文本序列S=w1,w2,…,wT，它的概率可以表示为：

$$p(S)=p(w1,w2,w3,w4,w5,\ldots,wt)$$

$$=p(w1)p(w2|w1)p(w3|w1,w2)\ldots p(wt|w1,w2,\ldots,wt-1)$$

即将序列的联合概率转化为一系列条件概率的乘积。常见的统计语言模型有马尔科夫假设

2.2.2 马尔科夫假设

基于马尔科夫假设(Markov Assumption)：下一个词的出现仅依赖于它前面的一个或几个词。

假设下一个词的出现依赖它前面的一个词，则有：

p(S)=p(w1)p(w2|w1)p(w3|w1,w2)…p(wn|w1,w2,…,wn-1)

=p(w1)p(w2|w1)p(w3|w2)…p(wn|wn-1) // bigram

假设下一个词的出现依赖它前面的两个词，则有：

p(S)=p(w1)p(w2|w1)p(w3|w1,w2)…p(wn|w1,w2,…,wn-1)

=p(w1)p(w2|w1)p(w3|w1,w2)…p(wn|wn-1,wn-2) // trigram

那么，如何选择依赖词的个数，即n。

更大的n：对下一个词出现的约束信息更多，具有更大的辨别力；

更小的n：在训练语料库中出现的次数更多，具有更可靠的统计信息，具有更高的可靠性。

2.2.3 One Hot Representation

最早的词向量是很冗长的，它使用是词向量维度大小为整个词汇表的大小，对于每个具体的词汇表中的词，将对应的位置置为1。比如我们有5个词组成的词汇表，词"Queen"在词汇表中的序号为2， 那么它的词向量就是(0,1,0,0,0)。同样的道理，词"Woman"是序号3，词向量就是(0,0,1,0,0)。这种词向量的编码方式我们一般叫做One Hot Representation。

One Hot Representation用来表示词向量非常简单，但是却有很多问题。

（1）任意两个词之间都是孤立的，根本无法表示出在语义层面上词语词之间的相关信息，而这一点是致命的；

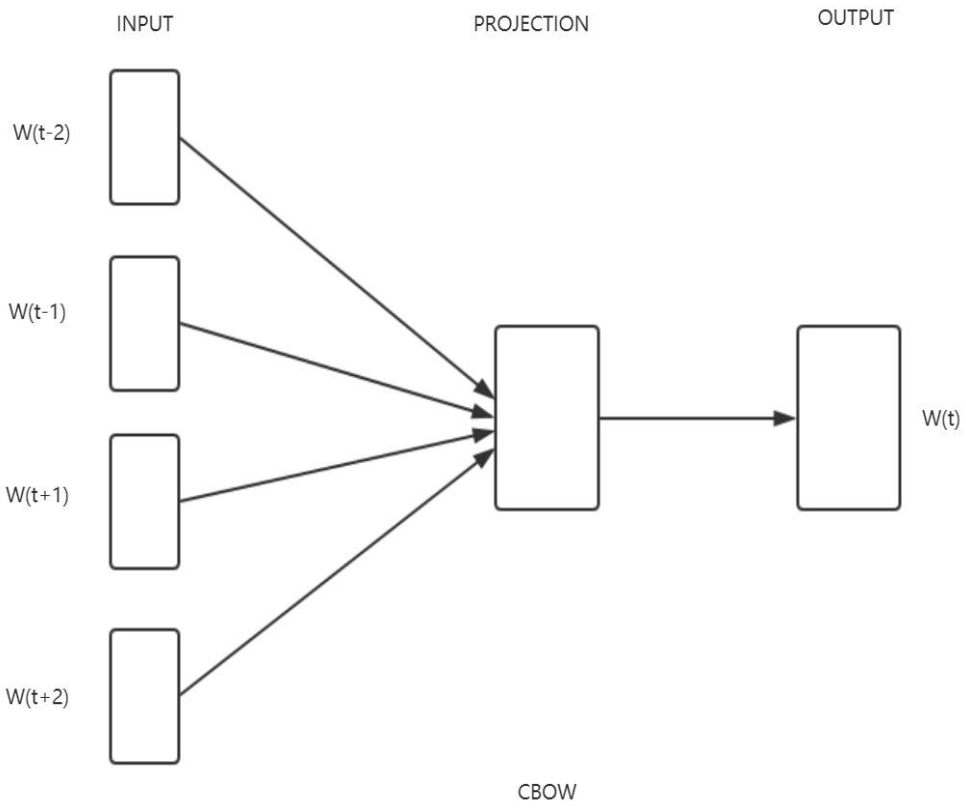
（2）我们的词汇表一般都非常大，比如达到百万级别，这样每个词都用百万维的向量来表示简直是内存的灾难。

Dristributed representation可以解决One hot representation的问题，它的思路是通过训练，将每个词都映射到一个较短的词向量上来。所有的这些词向量就构成

了向量空间，进而可以用普通的统计学的方法来研究词与词之间的关系。这个较短的词向量维度是多大呢？这个一般需要我们在训练时自己来指定，本案例中对筛选出来的结果按照时间、点赞数、反对数三个维度对其进行多方面系统地统计。词的分布式表示主要可以分为三类：基于矩阵的分布表示、基于聚类的分布表示和基于神经网络的分布表示。

2.2.4 连续词袋模型

连续词袋模型(Continuous Bag-of-Word Model)，是一个三层神经网络。如下图所示，该模型的特点是输入已知上下文，输出对当前单词的预测。



网络计算的步骤：

(1)上下文单词的onehot；（假设单词向量空间dim为V，上下文单词个数为C）

(2)所有onehot分别乘以共享的输入权重矩阵W；（V*N矩阵，N为自己设定的数，初始化权重矩阵W）

(3)所得的向量（注意onehot向量乘以矩阵的结果）相加求平均作为隐层向量, size为1*N；

(4)乘以输出权重矩阵W’ {N*V}；

(5)得到向量 {1*V} 激活函数处理得到V-dim概率分布 {PS: 因为是onehot嘛，其中的每一维都代表着一个单词}，概率最大的index所指示的单词为预测出的中间词（target word）；

(6)与true label的onehot做比较，误差越小越好。loss function（一般为交叉熵代价函数）

我们并不关心输出的内容，预测的结果不重要，重要的是训练完成后第一个全连接层的参数就是我们要的word embedding矩阵。通过该矩阵可以得出热点词汇的词向量，之后通过统计词向量得出热点词汇的时间跨度、事件热度等信息。

2.3 问题 3 分析方法与过程

本题的目的是在于如何更好地评价答复的相关质量，本案例中主要考虑的是答复的相关性、完整性，同时在此基础上添加了留言与答复之间的时间差来评价执政效率。具体如下：

2.3.1 回复的相关性

根据问题1的原理，利用PKUSeg将留言主题和详细留言进行分词与回复内容进行分词，之后进行两者之间的匹配，该匹配度越高说明回复的相关性越好，匹配的结果是有92.4%的回复相关性匹配度在60%以上，说明回复的相关度已经比较优秀。

2.3.2 回复的完整性

根据问题2的原理，对留言和回复进行分词后，将留言中存在的词汇没有出现在回复中进行统计后，匹配度越高说明完整性越差，即留言的词汇没有出现在回复中越少，完整性越高。为了在数值上更清晰地体现完整性，会用1减去留言的词汇没有出现在回复中的概率，即该结果越高，完整性越高。经过程序运行后得出的结果是有75.6%的回复匹配度在70%以上，说明回复的完整性稍有欠缺。

2.3.3 执政效率

对每条回复进行时间差的计算，得出的结果是有86%的留言后一个星期到一个月内进行回复，最长的时间差达到了三个月之久，回复速度还需要进行提升。

根据问题1和问题2的研究结果，首先分析答复与留言的相关性，主要是通过答复和留言的分词的相关性，来的出答复的相关性、完整性和可解释性；除了要考虑答复的相关性、完整性和可解释性，本文还将留言与答复之间的时间差纳入到答复意见的评价标准中。

3.结果分析

3.1 问题 1 结果分析

群众留言分类中，采用通过 PKUSeg 对留言主题、留言详情以及标签进行分词，利用分段分词匹配算法将留言进行标签匹配，建立了关于留言的一级标签分类模型，一个留言有多个标签的问题，如果留言含有正确的一级标签，本留言会被视为正确分类。为了检测模型分类的准确度，采用了题目中给出了 F-Score 对分类方法进行了评估：

第一步：随机抽选出 30 条留言；

第二步：运行程序，为留言设置一级标签；

第三步：采用 F-Score 将程序结果和材料结果进行比对，得出结果。

结论：采用通过 PKUSeg 对留言主题、留言详情以及标签进行分词，利用分段分词匹配算法将留言进行标签匹配，F 的平均数值为 87.3%。通过问题 1 给出结论可以看出通过 PKUSeg 进行分词以及利用分段匹配原则为留言进行分派取得了较好的效果，可以为工作人员节省相当一部分工作量。

标签正确		标签错误	没有标签
一个标签	多个标签		
67.9%	19.4%	9.8	2.9%

3.2 问题 2 结果分析

利用 Words2vec 生成热点词汇的词向量，对筛选出来的结果按照时间、点赞数、反对数三个方面对其进行多方面系统地统计，对热点问题进行挖掘。结果如下所示：

热度排名	问题 ID	时间范围	地点人群	问题描述
1	1	2019.7.21 至 2019.9.25	A5 区劳动东路魅力之城小区	非法经营餐饮油烟
2	2	2019.2.26 至	A 市百姓	省城 A 市医疗福利

3	3	2020.1.6 2017.6.8 至 2019.11.22	西地省经济学院	学生管理问题
4	4	2019.5.14 至 2019.10.18	移动公司用户	移动通信问题
5	5	2019.4.26 至 2019.10.31	A 市百姓	A 市物业管理问题

3.3 问题 3 结果分析

根据2.3节，政务系统回复性的匹配的结果是有92.4%的回复相关性匹配度在60%以上，说明回复的相关度已经比较优秀。具体概率如下表所示：

相关性	0%~20%	20%~40%	40%~60%	60%~80%	80%~100%
回复所占比例	1.4%	1.2%	4.9%	52.7%	39.8%

政务系统完整性的匹配的结果是有75.6%的回复完整性在70%以上，说明回复的完整性稍有欠缺。具体概率如下表所示：

完整性	0%~20%	20%~40%	40%~60%	60%~80%	80%~100%
回复所占比例	2.6%	7.8%	7.2%	42.6%	39.8%

同时对每条回复进行时间差的计算，得出的结果是有86%的留言后一个星期到一个月内进行回复，最长的时间差达到了三个月之久，回复速度还需要进行提升。

时间差（天）	0~10	11~20	21~30	31~40	41~50	51~60	61天以后
回复所占比例	42.2%	36.4%	7.4%	4.7	5.1%	2.6%	1.6%

我们可以看出回复的相关性比较好，但是完整性和实时性存在问题，建议政府尽快将“智慧政务”纳入到政务平台中。

4.结论

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，对提升政府的管理水平和施政效率具有极大的推动作用。这也是文本分析的一个创新点。传统的方法不能够满足如此巨量的数据。所以本文采用了文本解读已经不能满足数据量庞大的网络招聘信息。本文采用北大开源的一款基于Python的PKUSeg 中文分词工具对留言主题、留言详情以及标签进行分词；然后进行分段分词匹配，即各级标签分别与留言进行匹配；最后，考虑到有的留言无法与标签进行匹配，会将该类留言加入人工检查步骤。并且将留言类别定义为两大类：特定地点和特定人群，定义合理的热度评价指标，同时给出分词热度，用Word2vec生成热点词汇的词向量，用于词典的构建。利用发散性思维，再分别对筛选出来的结果按照时间、点赞数、反对数三个方面对其进行多方面系统地统计，结合图表对附件三中的留言信息进行归类总结。

通过问题1给出结论可以看出通过PKUSeg进行分词以及利用分段匹配原则为留言进行分派取得了较好的效果，可以为工作人员节省相当一部分工作量。为了追求效率，将字典的训练模型进行了简化，为了取得更好的效果，可以对字典进行多轮训练，以解决标签之间的冲突问题。

在问题2中，通过Word2vec得出的热点词汇，我们可以看出多个热点留言之间存在着相当高的重复，这为我们进行热点汇总提供了很好的经验，为了使Word2vec更好得使用，可以为热点词汇赋予动态权值，在一段时间内多次重复出现的词汇，权值会变高，表明政府需要注意，这是一个该方法的改进方向。

在问题3中，我们可以看出回复的相关性比较好，但是完整性和实时性存在问题，建议政府尽快将“智慧政务”纳入到政务平台中，同时为了确保回复的及时性和完整性，建议政务平台为回复也设置上点赞等功能。

总的来说，鉴于低下的人工效率以及快速发展的大数据、人工智能和NPL，政务平台应用新技术提升效率已经成为趋势和潮流，希望政务平台更好地服务大众。

参考文献

[1] 钟晓，马少平，张钹, et al. 数据挖掘综述[J]. 模式识别与人工智能, 2001(01):50-57.

[2] 邵峰晶. 数据挖掘原理与算法[M]. 水利水电出版社, 2003.

[3] 陈安, 陈宁, 周龙骧. 数据挖掘技术及应用[M]. 科学出版社, 2006.

[4] 贺玲, 吴玲达, 蔡益朝. 数据挖掘中的聚类算法综述[J]. 计算机应用研究, 2007, 24(1):10-13.

[5] 穆瑞辉, 付欢. 浅析数据挖掘概念与技术[J]. 管理学刊, 2008, 21(3):105-106.

[6] 王伟强，高文，段立娟. Internet 上的文本数据挖掘[J]. 计算机科学, 2000, 27(4).