基于深度学习的智能政务留言处理模型

摘　要

人工智能深刻影响了人们的日常生活和工作场景。将深度学习与政务服务相结合形成的智能政务系统，可以极大的提升工作效率。本文构建了一基于 BERT 模型、CNN 模型和聚类算法的智能文本挖掘模型，可以将大量的留言文本进行分类，筛选出热点问题，并且对留言的回复进行评价。

在预处理阶段，我们将源文本进行清洗，进行去停用词和分词处理。通过逐词、逐字符编码，对字/词计数，并建立字典，得到词的向量表示。于此同时，通过初步提取出文本的关键词。

在留言分类阶段，我们为了得到留言的分类，我们同时利用 BERT-Fine tune 模型和 CNN 模型，在两个网络下分别得到留言分类的概率，然后将二者的概率分布按权重相加，最终得到留言分类。

在热点挖掘阶段，我们将留言主题编码成 768 维的向量，利用 Hanlp 的 NER 得到留言详情的关键词（地名，组织名，人名）所产生的向量，使用层次聚类的方法，选取容量前 30 的簇，根据距离矩阵对簇进行适当的补充和评价之后，根据评分得出热点问题。

在答复意见的评价阶段，我们从相关性、完整性、可解释性和细致程度四个方面来评分，每个方面各分配一分，总分为 4 分。相关性通过关键词出现的的次数来评价。为了评价完整性，我们将设置一个要素表，包括引用文献、礼貌用语、联系电话等要素，将回复中出现要素的种类多少作为评价标准。对回复进行可解释性评价时，我们通过留言中是否出现序号来进行评判，出现序号则可以体现解释性。细致程度我们将通过留言的字数多少来进行评价。

关键词：BERT-Fine-tune　CNN　NER　层次聚类 智能政务 文本挖掘

目录

一、简介 3
1.1 挖掘意义 3
1.2 挖掘目标 3
1.3 挖掘流程 3
二、预处理 4
2.1 分词 4
2.2 去停用词 4
2.3 建字典和编码 4
三、群众留言分类 5
3.1 BERT-Fine-tune 模型 5
3.1.1 Transform Encoder 5
3.1.2 Positional Encoding 6
3.1.3 Fine-tune 6
3.2 CNN 7
3.2.1 嵌入层 7
3.2.2 卷积层 7
3.2.3 最大池化层 8
3.2.4 全连接层 8
3.3 分类 8
四、热点问题提取 8
4.1 Encode 部分 8
4.2 Hanlp——中文自然语言处理分词方法 8
4.3 NER——自然语言处理技术 9
4.3.1 结合 NER 的 BiLSTM 9
4.4 层次聚类 10
4.4.1 两个留言之间的距离计算 11
五、答复意见的评价 11
六、实验结果部分 11
6.1 实验环境配置 11
6.2 相关参数设置 12
6.3 热点问题 12
七、总结 13
参考文献 13

一、简介

1.1 挖掘意义

近年来，互联网高速发展，人工智能在人们的日常生活和工作场景中发挥着越来越重要的作用。随着政府公共管理服务与互联网的融合，民意的反应渠道越来越多，如微信、微博、市长信箱、阳光热线等，这给大众反映意见，给政府了解民意提供了巨大的便利。但是，随着各类民意相关文本数据量的极速上升，给传统的人工分类带来了巨大的工作量，给政府的工作人员带来了极大的挑战。因此，我们希望能通过人工智能，可以缓解政务服务人员人力不足的难题。

为了构建对大量文本数据的挖掘处理模型，学界对大数据和自然语言处理的步伐一直在进行。文本分类处理作为机器学习中自然语言处理任务的一部分，其目的是将大量的文本数据，按照一定的主题进行分类。在智能政务系统中，大量的留言需要进行分类，以便于后续将群众留言分配至相应的职能部门处理。

1.2 挖掘目标

我们要构建一个智能的文本挖掘模型。模型可以起到分类、热点问题挖掘和答复意见评价的作用，帮助人们显著提高工作效率。首先将大量留言信息按一定的标签分类，如城乡建设、保护环境、交通运输等，可以解决人工分类工作量大、效率低、差错率高的问题；其次，按照一定的热度评价指标，挖掘热点问题，可以有重点、有针对性的解决人民群众的问题；最后，针对相关部门对留言的答复意见，给出评价，可以作为留言回复考核的一部分。

1.3 挖掘流程

挖掘主要分为四部分，预处理部分、分类部分、热点提取部分和留言回复评价部分。其中预处理包括清洗数据，去停用词，建字典和编码。解决第一问分类问题，我们对留言主题部分采用 BERT+Fine-tune 模型，对留言详情部分采用 CNN，对两者的概率分布按权重相加，得到最终的留言分类结果。解决第二问热点问题

挖掘问题，我们对每个留言都构建一个 representation，分别是用 BERT 将留言主题 encode 成 768 维的向量，用 Hanlp 的 NER 得到留言详情的关键词（地名，组织名，人名），日期所产生的向量。用层次聚类，选取容量前 30 的簇，再对簇进行补充和评价，最终根据评分得出热点问题。

二、预处理

2.1 分词

在自然语言处理中，分词是指把句子拆分成一个一个的词语，这样能更好的处理句子，更好的分析句子的特性。但是，由于中文的句子并不像英文一样，天生自带分割，并且存在各种各样的词组，从而给中文分词带来了难度。

本文采用 Jieba[1]进行中文分词。Jieba 是一个 python 中实现中文分词的组件，它基于前缀词典实现高效的词图扫描，生成句子中汉字所有可能成词的情况所构成的有向无环图（DAG），用动态规划查找最大概率路径，找出基于词频的最大切分组合。

除此之外，Jieba 还有基于 TF-IDF 算法的关键词抽取功能，可以返回各个分词在一段文本中的权重比例。

2.2 去停用词

在文本处理中，去停用词是关键的一步。停用词是指语气助词、副词、介词、连接词等，通常本身并没有明确意义的词，比如说“了”“在”“呢”“的”等。去停用词，可以使我们的分析更多的关注定义文本含义的词，可以有效的提高关键词密度。在删除停用词时，数据集大小减小，训练模型的时间也减少，可以提高分类的准确性。

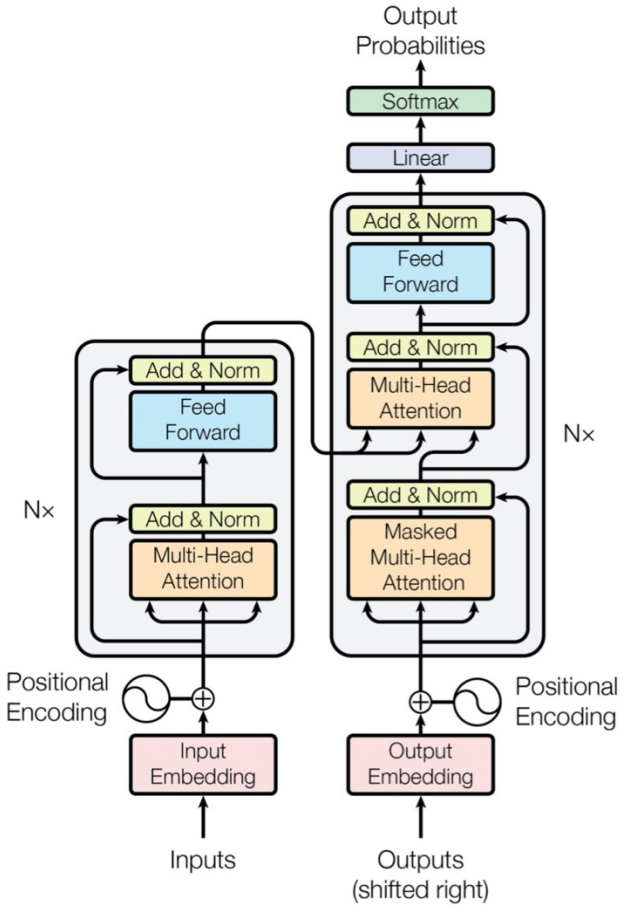
2.3 建字典和编码

用 Jieba 分词后，将一个个词放在字典中进行储存。为了将语料输入神经网络进行训练，我们要将自然语言文字表示称计算机能够理解的数字形式，即进行编码。我们采用的是逐词、逐字符编码。

三、群众留言分类

3.1 BERT-Fine-tune 模型

BERT 模型在 Google 发表的论文《Pre-training of Deep Bidirectional Transformers for Language Understanding》[2] 中被提出。BERT 采用 Transform Encoder 作为语言模型，完全抛弃了 RNN 和 CNN 的模型结构，完全采用 Attention 机制来进行 input-output 之间的关系计算。BERT 模型包括两个阶段，一个是训练语言模型的与训练阶段，另一个是训练具体任务的 Fine-tune 部分。



3.1.1 Transform Encoder

Transformer 模型是 Google 在论文《Attention Is All You Need》[3] 中提出的全新模型。与 Attention 模型类似的是，在 Transformer 模型中一样采用了 Encoder-Decoder 的模型架构，但是与 Attention 相比更加复杂。

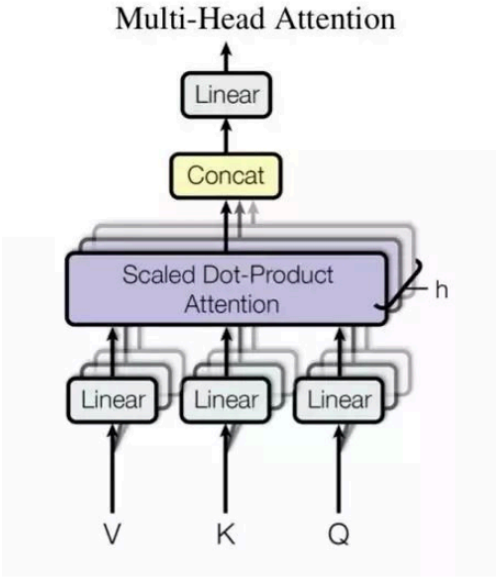
Transformer 中厉害的部分，是给原本的 Self-Attention 机制中，加入了一个

新的机制，即 Multi-Head Attention。当我们输入进一个句子时，这个机制将初始化多组 Query（Q）、Key（K）、Value（V）矩阵，对 Q、K、V 做线性变换，再对应计算出新生成的 Q’，K’，V’的 Attention 结果。重复这样的操作 h 次之后，h 次操作后的结果连接在一起，在做一次线性变换，就得到了一个 Multi-Head Attention 单元的输出。这个模型形成多个子空间，可以更好的让模型去关注不同方面的信息。

相应的计算公式为：

$$head_i = Attention\left(QW_i^Q, KW_i^K, VW_i^V\right)$$

$$MultiHead(Q,K,V) = concat(head_1, head_2, \dots, head_h)W^O$$



3.1.2 Positional Encoding

除此以外，在图中我们可以发现，Transformer 还在 encoder 层和 decoder 层中加入了一个额外的向量，即 Positional Encoding，这个向量的维度与 Embedding 的维度一样，它用来决定当前词的位置。

3.1.3 Fine-tune

Fine-tune 模型是指在已经训练好的语言模型的基础上，加入少量的具体任务的参数，在新的语料上重新训练。我们将在 BERT 上训练好的语言模型，进行 Fine-tune。

3.2 CNN

文本分类的中心思想，是抽取文档或者句子的关键词作为特征，然后基于这些特征去训练分类器并且实现分类。由于 CNN 模型中的卷积层和池化层就是一个抽取特征的过程，所以我们可以通过 CNN，在获得关键词的特征后，准确的提炼出文档或者句子的中心思想。

CNN 首次用于文本分类是在《Convolutional Neural Networks for Sentence Classification》[4] 一文中。模型主要包括四层，包括嵌入层、卷积层、最大池化层、全连接层。

3.2.1 嵌入层

嵌入层的本质上是一个特征提取器，它通过一个隐藏层，将编码的词投影到一个维数较低的空间当中，在指定的维度中编码语义特征。这样处理后，语义相近的词，它们之间的距离也会比较接近。

3.2.2 卷积层

在处理文本数据是，由于我们输入的每一行向量代表一个词，所以词是文本的最小细粒度，所以卷积核的宽度与词向量的维度是一致的。当我们在输入一个句子是，句子中相邻的词与词之间有很高的关联性，所以，当我们使用卷积核进行卷积的时候，不仅将词义考虑了进去，而且也可以包含到词序以及上下文之间的关系。

具体的操作如下[5]：

当我们输入一个表示句子的矩阵时，维度为$n \times d$, 即每句话共有n个词，每个词用一个d维的词向量表示。假设$X_{i:\ i+j}$表示X_i到X_{i+j}个词，使用一个宽度为d，高度为h的卷积核W与$X_{i:i+h-1}$（h个词）进行卷积操作后再使用激活函数激活得到相应的特征c_i，使用点乘来表示卷积操作，则卷积操作可以表示为：

$$c_i = f(w \cdot x_{i:i+h-1} + b)$$

经过卷积操作后，可以得到一个$n + h - 1$维的向量$c = [c_1, c_2, \dots, c_{n-h+1}]$。

同样的，我们使用更多高度不同的卷积核，从而得到更多不同的特征。

3.2.3 最大池化层

池化层的特点是它可以输出一个大小固定的矩阵，并且可以达到降低结维度的效果，同时可以保留原有文本的显著特征。最大池化只会输出最大层，并且对输入中的补 0 做过滤。

3.2.4 全连接层

最后一层全连接层，用 softmax 作为激活函数，输出每个类别的概率。

3.3 分类

通过 BERT-fine-tune 模型和 CNN 模型，我们可以分别得到留言分类的概率。将两个模型的概率分布按权重相加，我们可以将概率最大的类别，作为该留言的分类类别。

四、热点问题提取

4.1 Encode 部分

我们在进入模型前，需要先将自然语言通过 embedding 的方式进行编码，从而将高维语义空间的自然语言转化到低维的向量空间，这基本上成了 NLP 的通用模式。我们首先为每个留言构建一个 representation，由三部分组成，利用 3.1 中的方法，分别是用 BERT 将留言主题 encode 成 768 维的向量。

4.2 Hanlp——中文自然语言处理分词方法

Hanlp 的图词储存方法采用的是快速 offset 法。Hanlp 是一种自然语言处理技术，使用 Hanlp 可以很高效的进行自然语言的处理工作，比如进行文章摘要，语义判别以及提高内容检索的精确度和有效性等。

Hanlp 提供了以下功能：最短路分词，N-最短分词、CEF 分词、索引分词、极速词典分词等中文分词方法。索引分词是面向搜索引擎的分词器，能够对长词全切分，另外通过可以获取单词在文本中的偏移量；N-最短路分词器比最短路分词器慢，但是效果稍微好一些，对命名实体识别能力更强，但一般来说，最短路分词的精度已经足够，而且速度比 N-最短路分词器快几倍；CRF 对新词有很好的识别能力，但是无法利用自定义词典；极速分词是词典最长分词，速度极其快，但是精度一般[6]。

4.3 NER——自然语言处理技术

NER 的任务就是要将这些包含关键信息（地名，组织名，人名）的实体给识别出来。NER 的算法可以分为三个部分 CNN+BiLSTM+CRF：

（1）CNN 做字符级别的编码，主要解决词典以外的词 ：测试数据中出现了训练数据中未出现过的词的问题。

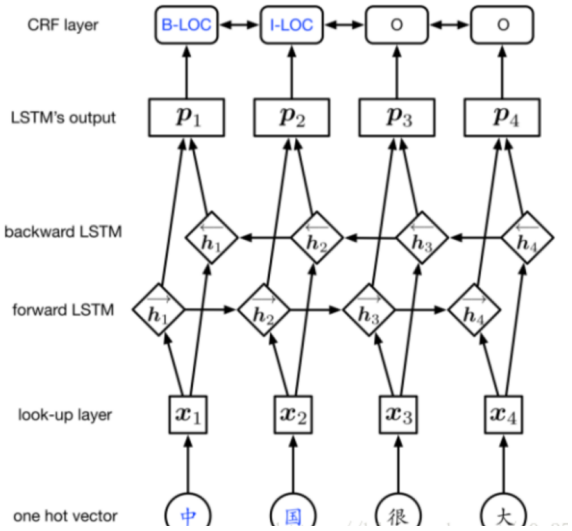
（2）LSTM 具有强大的拟合能力，可以很好的完成序列标注问题。

（3）CRF 能记住实体序列的规则，可以纠正 LSTM 的一些低级错误。

以此提取出留言详情中的关键词[7]。

4.3.1 结合 NER 的 BiLSTM

LSTM 解决了 RNN 中长期依赖的问题，但是其只考虑了语料的上文信息，而命名实体识别不仅需要考虑上文信息，还需考虑下文信息，故 BiLSTM 是一个非常好的选择。



BiLSTM 的架构如上图所示，x_1、x_2等表示文本的向量输入，$h \rightarrow t$为前向 LSTM 在t时刻的输出表 示,$h \leftarrow t$ 为反向 LSTM 在t时刻的输出表示，则 BiLSTM 在t时刻的输出表示定义为$h_t = [h \rightarrow t : h \leftarrow t]$,即直接拼接$h \rightarrow t$与$h \leftarrow t$。这种表示同时包含了上文信息和下文信息,适用于标签种类较多的命名实体识别任务【8】。

4.4 层次聚类

把留言详情所产生的向量作为输入进行层次聚类。凝聚型层次聚类的策略是先将每个对象作为一个簇，然后合并这些原子簇为越来越大的簇，直到所有对象都在一个簇中，或者某个终结条件被满足。

先计算样本之间的距离。每次将距离最近的点合并到同一个类。然后，再计算类与类之间的距离，将距离最近的类合并为一个大类。不停的合并，直到合成了一个类。其中类与类的距离的计算方法有：最短距离法，最长距离法，中间距离法，类平均法等。本文中采取的是计算类平均距离

算法流程：

(1) 将每个对象看作一类，计算两两之间的平均距离；

(2) 将距离最小的两个类合并成一个新类；

(3) 重新计算新类与所有类之间的距离；

(4) 重复(2)、(3)，直到所有类最后合并成一类。

(5) 选取容量前 30 的簇，再对簇进行补充和评价，最终根据评分得出热点问题

热度评价指标的计算方法为：

（1） len是簇内元素个数；

（2） 长度指标=$sigmoid\left(\frac{len}{10}\right)+\frac{len}{50}$

（3） 紧密指标=簇内各个元素间的平均距离/5

（4） 二者相加即为热度指标

层次聚类的优势：距离和规则的相似度容易定义，限制少；不需要预先制定聚类数；可以发现类的层次关系；且用层次聚类可以一次性得到所有结果【9】

4.4.1 两个留言之间的距离计算

在进行层次聚类时需要计算两个留言之间的距离，每条留言分为三个部分分别计算之间的欧氏距离，然后按照权重相加。关键词的距离也是三类分开算然后按权重相加，具体方法是如果两个关键词列中有相等或者包含关系则将会缩短距离，如果既不是相等也不是包含，则会将两个词 embed 后算其向量的夹角(这里解决的是相同地点的不同叫法问题)，适当按照词列的长度有正则化。这里的 embed，用的是基于字的，词的 embed 是其平均。

日期向量用的是 cos 夹角来衡量距离(想的是要捕捉不同年但季节相近的留言)

五、答复意见的评价

针对相关部门对留言的答复意见，我们建立了如下的评价标准：

内容	标准	得分
相关性	关键词出现的词数	$\frac{\text{在回答中出现的关键词数}}{\text{总关键词数}}$
完整性	设置一个要素表, 如引用文献、礼貌用语、联系电话等要素	$\frac{\text{出现的要素数}}{\text{总要素数}}$
可解释性	是否有序号	有序号得一分，无序号为 0 分
细致性	留言回复的文本长度	$\text{sigmoid}\left(\frac{\text{文本长度}}{200}\right)$
总分		4

如上图所示，我们建立了凭借留言答复意见的一套评价机制，从相关性、完整性、可解释性和细致程度进行评价，总分为 4 分。

六、实验结果部分

6.1 实验环境配置

我们的实验环境配置如下图所示。

CPU	i7
RAM	16G
OS	win10
python	3.7.0
torch	1.3.1
Tensorflow	1.15.0&2.1.0

6.2 相关参数设置

我们训练时的参数设置如下表所示。

参数	CNN	BERT	第二问
学习速率	1×10^{-3}到1×10^{-4}	5.0×10^{-5}	
优化算法	Adam		
一阶矩衰减率	0.9		
二阶矩衰减率	0.999		
Epoch	30	3	
Batch size	32	32	
词向量维数	200		300
Kernel size	10		

6.3 热点问题

经过上述的算法，我们评选出如下五个热点问题，热点问题明细可见《附件一：热点问题留言明细表》。

热度排名	问题ID	热度指数	时间范围	地点/人群	问题描述
1	1	3.4314	2019/06/19 至 2020/01/26	A 市丽发新城小区	搅拌厂噪音和扬尘扰民
2	2	3.1007	2019/07/11 至 2019/09/01	A 市伊景园滨河苑	强制捆绑车位销售
3	3	2.0189	2019/01/26 至 2019/06/04	星沙中茂城商铺购买者	企业老板邱浪杰欠债不还
4	4	2.0142	2019/07/21 至 2019/09/25	A5 区劳动东路魅力之城小区	烧烤夜宵摊油烟直排扰民
5	5	1.9876	2019/01/09 至 2019/09/08	A 市各个小区	自来水公司无故停水，且有时水质较差

七、总结

随着大数据、云计算等现代信息技术的发展，传统的纸质文档快速向电子化、数字化转变。面对大量的数据和信息，人们越来越倾向于利用计算机对数据和信息进行处理，不但可以提高相关操作的效率，还可以在一定程度上提高相关操作的准确度。信息挖掘和检索、自然语言处理是目前数据管理的关键技术，而文本分类则是这些技术进行操作的重要基础，是目前研究的一个热点，也是一个难点。本文构建了一基于遇 BERT 模型、CNN 模型和聚类算法的智能文本挖掘模型，可以将大量的留言文本进行分类，筛选出热点问题，并且对留言的回复进行评价。我们还建立了凭借留言答复意见的一套评价机制，从相关性、完整性、可解释性和细致程度等角度进行评价。

参考文献

【1】 https://blog.csdn.net/bozhanggu2239/article/details/80157305

【2】 《 Pre-training of Deep Bidirectional Transformers for Language Understanding》

【3】 《Attenion Is All You Need》

【4】 《Convolutional Neural Networks for Sentence Classification》

【5】 https://blog.csdn.net/vivian_ll/article/details/80831509

【6】 https://blog.csdn.net/adnb34g/article/details/80104119

【7】 https://www.jianshu.com/p/6668b965583e

【8】 https://blog.csdn.net/m0_37565948/java/article/details/80936596

【9】 https://blog.csdn.net/andy_shenzl/java/article/details/83783469