摘　要

随着大数据时代的到来，民情民意调查更多的转向线上。为了更好的相应民众诉求，使用自然语言处理来对留言进行初步分析、反馈，已成为政务办公中的必然趋势。本文使用了Transformer架构的优化BERT模型进行特征提取，在其基础上进行了迁移学习，实现了文本分类、相似度计算、命名实体识别等。

数据预处理阶段，采用WordPiece方法直接对输入进行字典编码和切分，添加必要的[CLS]标签位之后，结合模型自身的位置嵌入信息，通过嵌入层，进入BERT计算。

采用多头自注意力机制使得模型挖掘到更丰富的语言特征信息，并使用BERT层更深的模型结构，增加模型的适应性。解压出整体特征信息后，分为整体特征、字向量特征两部分，进行后续建模。

针对第一问分类问题，我们增加一个全连接层，并通过Softmax激励函数，获得每一类的概率，取argmax获得概率最大的类别序号，对应初始化阶段读取的全部标签信息，进行标注。

针对第二问热点问题挖掘，我们分两步进行：

1.先直接使用所有待排序文本作为全集，以BERT的[CLS]输出向量为依据，进行Me-anShift聚类，带宽约束使用的距离计算方法采用向量余弦相似度的倒数。

2.在BERT上添加一个连接到每个字向量的全连接层，再在其上增加一个条件随机场层，用以分别计算在每一个字的位置到标签的概率，使用Viterbi算法，完成对命名实体的标注。以点赞数在时间段内的比例、相似项个数、反对数在时间段内的占比为依据，加权求和，获得需要热点指数，并使用Pandas生成热点问题表格和热点问题明细表格。

针对第四问留言的回复评价，我们基于客观性、系统性、可操作性三个原则构建二重指标体系四框架模型，分别从答复可解释性、内容完整性、态度优良性、群众满意度四个一级指标综合对答复进行考察。以留言答复相关性、字数充实度、回复及时性、语气真诚度、敬语实用度、答复耐心度、点赞数和留言重现率八个二级指标进一步对回复进行评价。

Key Words: WordPiece Transformer Multi-head-Attention MeanShift 条件随机场

Abstract

With the advent of the era of big data, more public opinion polls have turned online. In order to better respond to the demands of the public, the use of natural language processing to conduct preliminary analysis and feedback of messages has become an inevitable trend in government affairs. In this paper, the optimized BERT model of the Transformer architecture is used for feature extraction, and transfer learning is performed on the basis of it, which implements text classification, similarity calculation, and named entity recognition. In the data preprocessing stage, the WordPiece method is used to directly encode and segment the input. After adding the necessary [CLS] tag bits, embed the information in combination with the position of the model itself. The input information enters the BERT calculation through the embedding layer.

The multi-head self-attention mechanism enables the model to mine richer language feature information, and uses the deeper model structure of the BERT layer to increase the adaptability of the model. After decompressing the overall feature information, it is divided into two parts: overall feature and word vector feature. Then subsequent modeling is performed. For the classification problem, we add a fully connected layer and obtain the probability of each class through the Softmax excitation function. Take the argmax to obtain the category number with the highest probability corresponding to all the label information read in the initialization stage and label it.

There are two steps for mining hot issues:

1.First use all the text to be sorted as the complete set, and perform MeanShift clustering based on the BERT's [CLS] output vector. The distance calculation method used by the bandwidth constraint uses the inverse of the vector cosine similarity.

2.Add a fully connected layer connected to each word vector on BERT, and then add a conditional random field layer on it to calculate the probability of the position of each word to the label. Use Viterbi algorithm to complete the labeling of named entities.

Based on the proportion of likes in the time period, the number of similar items, and the proportion of anti-counts in the time period, weighted summation is used to obtain the required hotspot index. And use Pandas to generate hotspot question form and hotspot question detail form.

For the reply to the message, we constructed a four-frame model of the dual indicator system, and examined the reply from four aspects: the interpretability of the reply, the integrity of the content, the excellent attitude, and the satisfaction of the masses. The response is further evaluated by eight secondary indicators, such as relevance of message reply, word count, timeliness of reply, sincerity of tone, practicability of honorifics, patience of reply, number of likes and message reproduction rate.

Key Words: WordPiece Transformer Multi-head-Attention MeanShrift CRF

目录

1 简介

1.1 题目背景

近年来，随着微信、微博、市长信箱等网络问政平台的普及，各类社情民意相关的文本数据量不断增加。带来的问题是相关部门在面对大量的文本民意信息时无法快速获取需要的信息，对留言进行划分和热点处理，有时甚至会有人工归类群众留言出错的情况。每天群众都会有不同的问题，大量的信息聚集在一起形成海量信息，政府部门若无法在海量文本中找到群众最关注的问题，就无法实现有限资源最大利用化；面对海量的留言和不同部员对其的答复意见，没有合理的评价体系，政府无法知道其职能的实现度，不利于政府的进步优化。这些都会造成政府部门行政效率的降低。

1.2 挖掘目标

建立基于自然语言处理技术的智慧政务系统，对群众问题分类实现部门分工明确化、热点问题筛选实现政府工作针对化、建立留言答复评价体系，实现政府工作透明化。结合题目所给的留言主题表、评价回复表等，实现以下目标：

1. 训练模型使模型实现留言一级标签对应化。
2. 基于地点时间对留言问题进行归类，筛选热点问题并根据热度情况进行排序。
3. 建立留言答复评价体系，从答复的相关性、完整性、可解释性等因素对部门所给的答复意见的质量进行打分。

2 数据预处理

本文基于BERT模型进行数据挖掘。BERT全称为Bidirectional Encoder Representation from Transformer是一种由Google开发的双向预训练语言模型，可以广泛应用于NLP领域。本章节将介绍文本基于BERT系统的预处理部分。

2.1 分词

基于中文词与词之间无间隔的特点，大多数自然语言处理都是基于词。所以我们需要对文本进行分词。常用的中文分词有Python 中的Jieba ，它能将句子拆分成词条的组合。但本文所用的BERT在分词上有所不同。BERT将所有人类语言视为统一种符号系统，因此与其它模型的分词操作在此有所差异。我们认为，足够的预训练规模可以使得模型“掌握”一些字词之间的关系与结构。因此我们直接按照字为单位对输入句进行分词。

2.2 输入表示

在BERT模型中，对于输入句中每一个分词(token)，它的表征由其对应的分词表征(token embedding)，段表征(segment embedding)和位置表征(position embedding)相加产生。基于BERT模型输入形式的特点，在编码过程中，使用了如下的特殊注记：

- **[SEP]**：标记句子的结束。在Q&A类型的任务中，也用于标记两段文本的起始位置。
- **[CLS]**：在文本前加入的标签占位符，用于在输出层叠加出全段文本的特征向量。
- **[MASK]**：在填空、Mask ML预训练过程中，用于标记空缺部位。
- **[PAD]**：在文本长度低于设定的长度时，用于补全文本。

在完成以上注记插入与解析后，编码器将会采用一张体积为30000的含中文中文字表进行WordPiece嵌入，转化为Token Embeddings和Segment Embeddings两个向量，与学习而来的Position Embeddings进行和操作，输入Transformer。输入过程如下图所示。

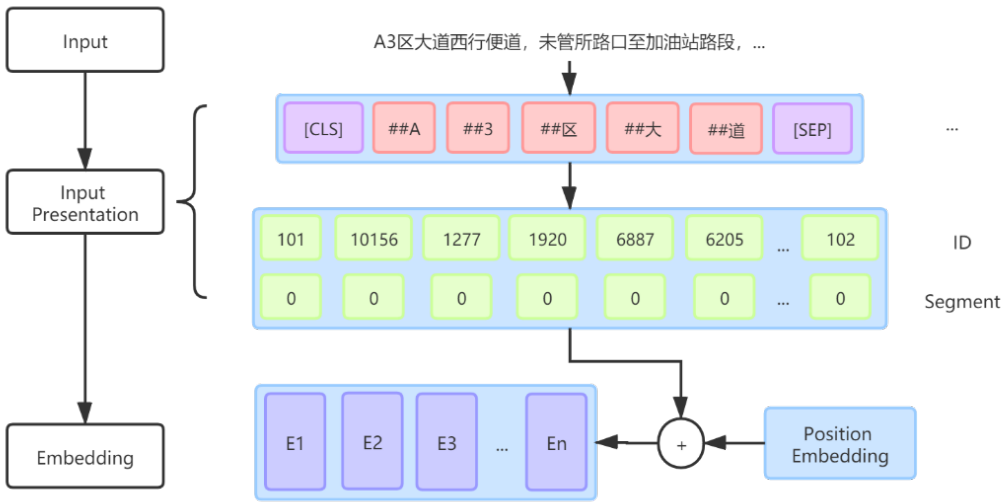


图 1: BERT模型输入表示

3 模型介绍

3.1 Transformer模型

3.1.1 RNN和LSTM模型

传统的长短期记忆网络（LSTM）是一种时间递归神经网络，适合于处理和预测时间序列中间隔和延迟相对较长的重要事件，它的结构是一种循环神经网络（RNN）的变化形式，通过神经元中的几个Sigmoid门来控制每一个个输入信息对于下一个神经元和输出信息的保留程度。双向的LSTM连接（Bi-LSTM）能实现能极大程度保留长程后向和前向依赖特征。但由于它都具有“后一个神经元计算需要前一个神经元的计算结果”这样的特征，导致其并行运算能力较差。对于较长文本、工业中的实时预测来讲，局限性较强。一个更严重的问题在RNN上有所体现，即在误差前向传播的过程中，会产生梯度消失的问题。这对于网络权重训练是极其不利的。

3.1.2 Transformer 模型结构

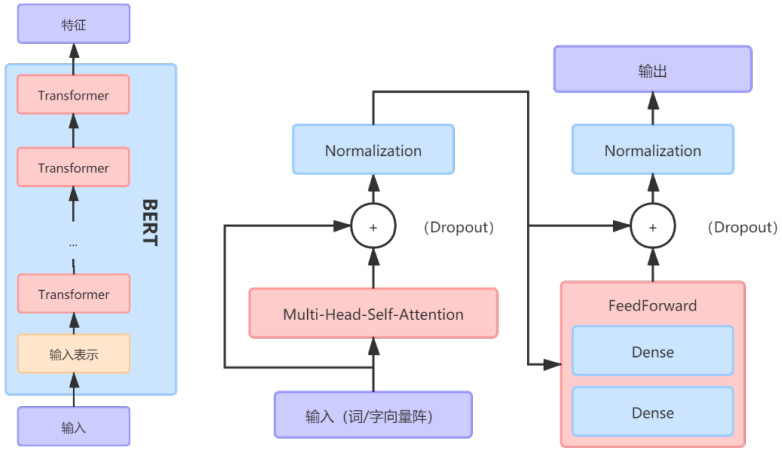
Transformer模型类似于卷积与反卷积的方法，它只依赖于注意力机制，以实现快速并行，极大提高了计算效率。Transformer模型不明确标记单词顺序，只标记单词的绝对位置嵌入，改进了RNN训练慢的缺点，深度增加，使得模型特性被充分挖掘，提升了模型准确度。

Transformer模型采用了encoder-decoder 架构，对输入进行编码，通过多个注意力层之后再进行解码，完成序列到序列、序列到位置、序列到类别的多种预测。encoder负责把自然语言序列映射成为隐藏层，共包含两层，一个self-attention层和一个Feed Forward层。self-attention能帮助当前节点不仅仅只关注当前的词，还能获取到上下文的语义。decoder将隐藏层再映射为自然语言序列，也包含encoder提到的两层网络，但是在这两层中间还有一层attention层，帮助当前节点获取到当前需要关注的重点内容。BERT由多个Transformer单元直接连接构成如图1，每一个Transformer单元的结构如图2。

3.2 注意力机制

3.2.1 attention 层与 self-attention 层

attention是一种能让模型对重要信息重点关注并充分学习吸收的技术。传统的机器翻译基本都是基于Seq2Seq模型来做的，该模型分为encoder层与decoder层。Seq2Seq模型对于短文本的翻译来说效果很好，但是其也存在一定的缺点，如果文本稍长一些，就很容易丢失文本的一些信息。而attention模型很好的弥补了这个不足。



(a) BERT的结构　　(b) Transformer单元的结构

该机制通过对输入采取不同的权重来决定输出对输入的向量所产生的“注意力”大小。这种机制实际上是受到人类阅读理解中的“上下文关联”和“要点信息”启发而被开发出来的。对于一个经过嵌入过程的输入序列而言，对于文本中单个词向量的注意力输出可以表达为以下形式：

$$\hat{a} = Attention(Q, K, V) \tag{1}$$

其中，（query）为要查询注意力的词/字的向量表示，（key）为抽取的要与计算相对注意力权重的词位置，（value）为全文原始输入的副本，用于输出以注意力加权后的增强向量表示。

具体的注意力运算可以表示为一个简单Q与K的乘法和Softmax处理的连接，再与V线性变换求和：

$$\begin{cases} a_k = \sum_{s=0}^{E_{size}} Q_s K_{ks} \\ \hat{a} = softmax(\hat{a}) \end{cases} \tag{2}$$

我们对以上注意力公式做出以下解释：做Q与K的简单点乘输出得到的分数值决定了我们在某个位置encode一个词是，对输入句子的其他部分的关注程度。做softmax处理可得到每个词对当前词的贡献的权重大小，再将Value和softmax得到的值进行相乘，并相加，得到的结果即是self-attetion在当前节点的值。

事实上，可以将提取和的操作视为对原始输出的一次线性变换（为了书写方便，省略了一切偏置，实际的计算层中偏置是不可缺少的），即：

$$Q = X_i, K = W_k X \tag{3}$$

$$O = aX_i \tag{4}$$

其中，X为上一层的输出（可能是词嵌入过程的输出，也可能是上一层Transformer的输出)。

将上述对单个词向量的注意力计算过程扩张到对每一个词进行Attention计算，计算每一个输入向量对于全体输入向量的注意力，即对Q、K、V都取原始输入来进行后续操作操作，可得到self-Attention的计算：

$$a = self - attention(X) \tag{5}$$

$$a_{kq} = \sum_{s=0}^{E_{size}} Q_{qs} K_{ks} \tag{6}$$

$$a_q = softmax(a_q) \tag{7}$$

$$O = aW_K X \tag{8}$$

注意力机制的流程如下图所示，这种通过 Q 和 K的相似性程度来确定 V的权重分布的方法被称为scaled dot-product attention。

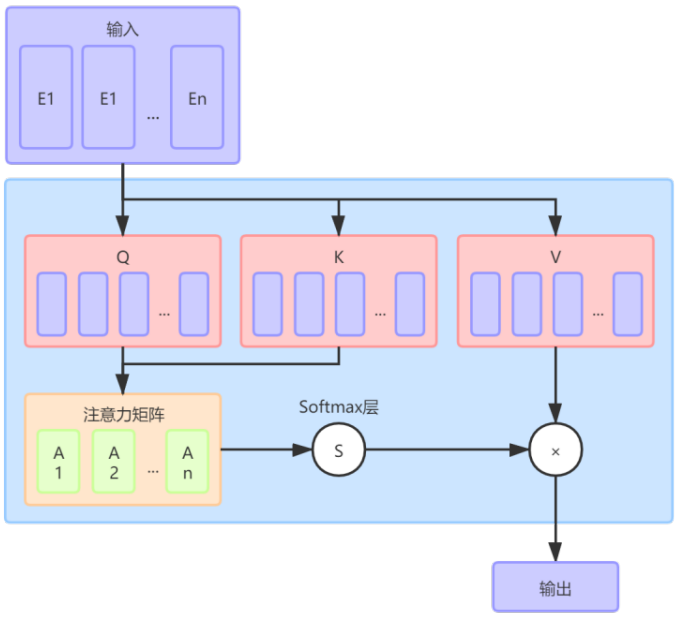


图 2: 注意力机制

3.2.2 Multi-head Self-Attention 层

为了用注意力机制来提取多重语意的含义，获得文本中每个字在不同语义空间下的增强语义向量。我们用一组权重不同的Self-Attention模块，加权线性组合为结果的“多头注意力”。

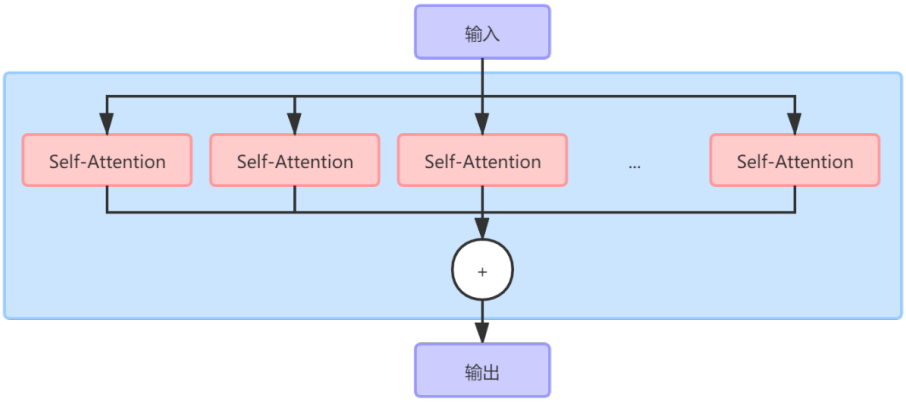


图 3: 多头注意力示意图

使用Pypi上的Bert2Keras库，来将模型加载为Keras标准模型并加以调整。使用Keras实现self-attention模型的构建。此外，Keras模型也可以方便地使用Tensorflow后端来完成训练与测试。

3.3 BERT原始预训练方法

BERT是一种语言模型，本身只是一个特征抽取网络。模型用于其它任务前，需要学习到特定语言的一些语法特征。Transformer结构的开发者认为，一篇文章本身就是天然的训练语料，可以对Position Embedding和Attention产生积极的、通用的影响。
在此情况下一般通过两个任务来使模型获得良好的初始参数：

- **遮罩语言模型（Mask Language Model）**
  对于一段完整的输入文本，在输入模型的时候随机选定15%的字/词进行修改后输入模型。其中80%的情况下直接替换为[MASK]标记，10%的情况下保留原词，剩下10%的情况下替换为其他单词。之后在模型输出端加上解码过程，将向量反向求得对应的字符，进行Sequence2Sequence预测，预测Mask位置的原词，并修正模型。
- **连续句子判定（Next Sentence Predict）:**
  给定一对句子，用[SEP]标记切分点，其中50%的情况下两个句子是连续的，另外50%情况两句话不存在上下句关系。之后添加Dense层进行二分类。

4 实验过程

本节中我们将对上文中我们建立的模型进行实验验证，通过不同模型的比较分析和相互验证以及准确度、召回率等一系列验证，我们说明了我们模型的合理性和有效性。

4.1 实验平台

本章主要用于介绍实验过程用到的硬件、软件用到的库。

硬件：

Intel Core i9-9900K @4.9GHz 8C 16T

DDR4 16GB

NVIDIA GTX 1660Ti 6GB

软件：

Windows 10 1909

Python 3.7

TensorFlow 2.1 with AVX2

TensorFlow GPU 2.1

CUDA 10.1 cuDNN 7.6.5

使用到的python包有：

- Pandas：用于从给定的CSV数据集中筛选重组留言为适用于BERT输入的结构和维度。
- Tensorflow 提供进行计算、误差传播所必须的与CPU、GPU硬件交互的二进制文件及基本接口。使用tensorflow-windows-wheel上提供的适用于windows平台、cuda10和avx2指令集的预编译包。
- Keras 提供进行网络设计的高级类库，并自动选择TensorFlow后端或PyTorch后端
- Bert4keras：内置基本模型层结构，可以加载类BERT模型的不同参数和预训练checkpoint，并输出Keras层，方便编辑。

4.2 一级标签分类模型

题目希望我们建立一级标签分类模型，实现群众留言的分类。通过附件1，我们观察到1级标签总共有15个。

基于上文的优化BERT特征提取及迁移学习的文本特征提取，CLS中含有输出层叠加出的全段文本的特征向量，通过建立全连接层（dense layer）每一个节点与特征节点的连接，通过Softmax激励函数将CLS位置上的特征矩阵转化为1*15的向量$A = (a_1, ...a_{15})$，每个向量元素a_i表示留言属于第i个标签的概率，，取argmax获得概率最大

的类别序号，对应初始化阶段读取的全部标签信息，进行标注。

观察附件2,发现主题一栏除了个别反馈以外，基本上已经体现了要反馈的问题的梗概。但为了使得模型具有更强的长文本预测能力，我们分别以主题、内容作为模型训练预料。剔除两段无效空格后，将每个标签下的语料按1:9的比例分为测试集和训练集来训练一级标签分类模型。

4.2.1 RoBERTa改进

由于预训练所需的数据体积较大，且对硬件条件要求较高。我们直接使用了由第三方提供的预训练模型RoBERTa+Small：训练数据为大小为35G来源于百科和新闻的语料。RoBERTa结构上与BERT相同，唯一的区别在于预训练方法。

1. RoBERTa舍弃了NSP任务，而是每次使用最大长度的文本作为输入，不显式训练网络的句子关系预测能力。而剔除了NSP的模型在MNLI任务上的分数达到了84.7，比BERT的84.3甚至拥有更好的性能。
2. RoBERTa在MLM任务上，用动态Mask取代静态Mask。即对于相同的语段，多次选定不同单词来进行[MASK]操作。动态Mask在SST-2任务上得到了92.9的分数，和静态Mask的92.5相比也略有优势。

表 1: 模型对比

	MNLI	**SST-2**
静态MASK	84.3	92.5
动态MASK	84.0	92.9
应用NSP	84.3	92.8
弃用NSP	84.7	92.5

总体说明在长文本特征抽取与分类上RoBERTa拥有更好的表现，这也是我们选用这组预训练模型的原因。以此具备了语言特征的模型为基础，我们可以使得后续任务拥有更好的训练起始点，得到良好的收敛速度，并对小批量的样本拥有更完备的效果。修改后的模型如图所示：

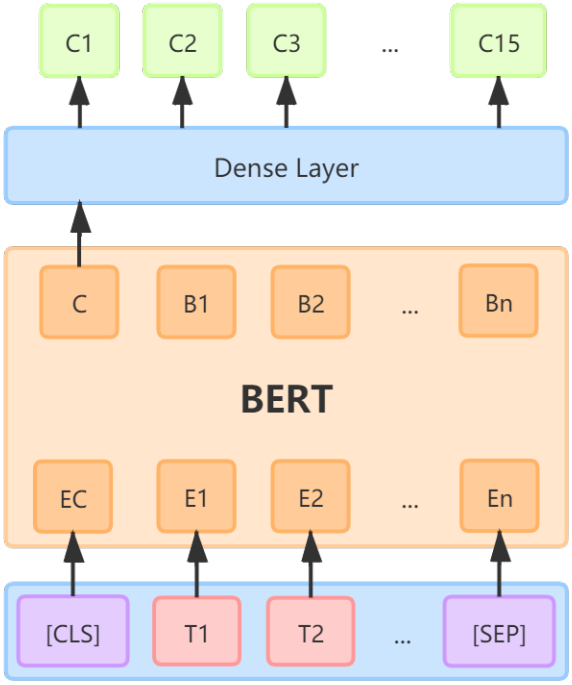


图 4: 修改后的训练模型

4.2.2 模型训练

由于对于序列采用了数字编码形式，损失计算采用交叉熵损失，它刻画的是实际输出（概率）与期望输出（概率）的距离，可以很好的评估当前训练得到的概率分布与真实分布的差异情况：

$$Loss = -\frac{1}{n}\sum_{i=1}^{n}[y^{*}lny + (1-y^{*}ln(1-y))] \tag{9}$$

其中y^{*}为神经元预期输出，y为神经元实际输出，n为类别个数。为了避免出现过拟合和收敛性问题，我们采用RAdam优化器及线性减缓的学习率修正因子：

$$learning\_rate = \begin{cases} 10^{-4}, n = 1000 \\ 10^{-5}, n = 2000 \end{cases} \tag{10}$$

其他参数的情况如下表所示。

表 2: 其他参数

	训练迭代次数	批次体积	批次数量	嵌入维度	截断长度
参数值	10	32	267	128	128

4.2.3 分类模型评价指标

对于本文中的模型，我们采用精确率（Precision）、召回率（Recall）、两者的调和平均数 F1-Score来评价我们的模型。分类准确度评测指标如下图所示。

表 3: 分类准确度评测指标

准确率（查准率）	召回率（查全率）	F1指标
$precesion = \frac{TP}{TP+FP}$	$recall = \frac{TP}{TP+FN}$	$F1 = \frac{2PR}{P+R}$

表 4: 分类准确度指标说明

实际	预测	
	预测到本标签	预测到其他标签
本标签	真正例(TP)	假反例(FN)
其他标签	假正例(FP)	真反例(TN)

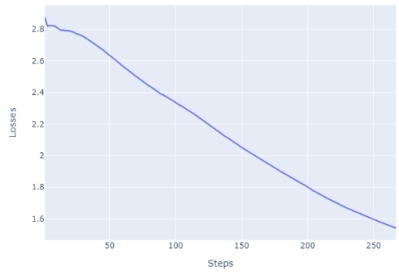
根据以上公式计算出每一个类别i的查准率P_i，查全率R_i。

再根据公式求得F1。

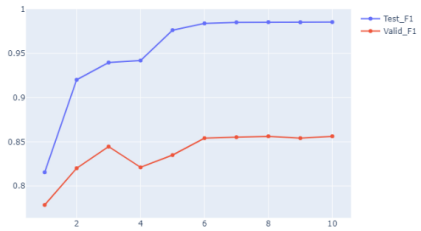
$$F1 = \frac{1}{n}\sum_{i=1}^{n}\frac{2P_iR_i}{P_i+R_i} \tag{11}$$

4.2.4 模型结果

将预处理后的文本输入模型，对训练数据随机抽取，获得了很好的收敛效果：



(a) 迭代1的损失曲线



(b) 10次迭代测试集与验证集的F1值

在10次迭代后：模型在训练集上F1-Score达到0.99，在测试集上F1-Score达到了0.85。基本认为，模型已经可以胜任分类任务。

4.3 热点问题挖掘

在本文中，我们的目标任务是找寻某一时段内群众集中反映的某一问题，定义合理的热度评价指标，对热点问题进行排序，并从热点问题出发给出提出热点问题的具体留言明细。基于之前完成的BERT模型和迁移学习的文本特征提取，我们已获得凝聚每段留言主要信息的特征向量。本题我们的核心思路如下：

1. 对留言的特征向量应用余弦相似度算法判断留言之间的相似度，挖掘相似度高的留言。
2. 根据相似度高的留言的条数以及留言的反对数和点赞数，定义热度评价指标。
3. 将向量余弦相似度的倒数作为留言之间的带宽约束使用的距离，应用MeanShift聚类算法和定义的热度评价指标对热点问题进行排序。
4. 在BERT上添加一个连接到每个字向量的全连接层，再在其上增加一个条件随机场层，用以分别计算在每一个字的位置到标签的概率，使用Viterbi算法，完成对命名实体的标注(NER)，得到热点问题留言明细表。

4.3.1 余弦相似度与MeanShrift

每段留言对应的特征向量都被存在了我们定义的[CLS]这个标签占位符中，由于我们的编码方式保证了不同文本特征向量维数的一致，[CLS]输出的形式均为1*384的向量。通过全连接层，我们将[CLS]位置上的1*384的向量变换成了1*15的向量。两种形式的向量都含有留言文本的特征信息，且维数一致，因此我们采用余弦相似度算法来判断文本的相似度。

余弦相似度算法(Cosine Similarity)是根据两个词向量之间的余弦夹角判断词向量之间的相似性,如果两个向量的夹角为0,此时其夹角 θ 的余弦值cos(θ)=1。余弦值越接近1，则相似度越高。假设词向量的维数为n，则两个词向量可以分别表示为$X = (x_1, ...x_n)$，$Y = (y_1, ..., y_n)$。其具体的计算公式为

$$similarity = cos(\theta) = \frac{XY}{\|X\|\|Y\|} \tag{12}$$

虽然经过全连接层形成的向量的维数比较低，但它的含义是留言属于不同一级分类标签的概率，因而它所包含的信息量远没有[CLS]保存的向量丰富。所以我们对[CLS]中的特征向量应用余弦相似度算法，计算两两留言间的相似度。

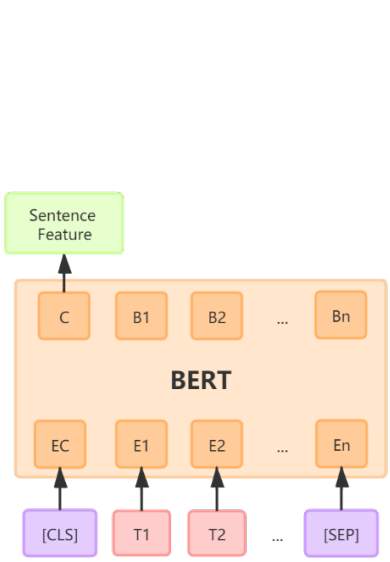
MeanShift是一种基于核密度的非参数聚类算法，其主要的思想是不同类别的数据集符合不同的概率密度分布，任意样本点密度增大最快的方向即均值偏移方向，样本密度高的区域与该分布的最大值对应，样本点最终收敛到局部最大值，且收敛到相同局部最大值的点被认为是同一类的成员。该算法的关键操作是计算中心点的漂移量，从而移动中心进行下一次迭代，直达到达密度最大处。

具体算法流程如下：

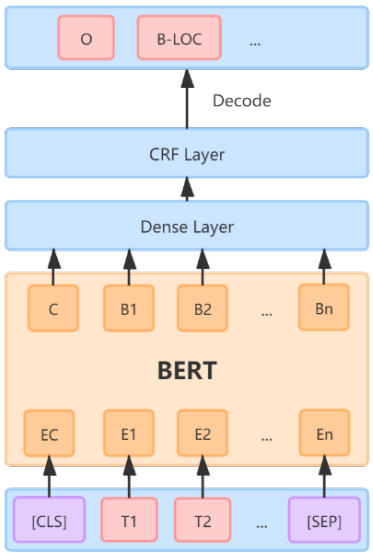
1. 计算样本的均值漂移向量$m(x)$
2. 对每个样本点进行平移$x_i = x_i + m(x)$
3. 重复步骤1和2直到$m(x) = 0$
4. 收敛到同一点的样本被认为是同一类的成员。

在本题中我们采用MeanShift聚类算法而不采用k-means聚类算法，因为MeanShift聚类算法可以自动决定类别的数目而k-means则需要人为指定类别数目。由于我们在进行聚类的时候无法得知其具体的类别数量，所以我们采用了MeanShift聚类算法。

由于需要网络更好地适应聚类，我们移除了分类中的最后一个全连接层，促使模型中的[CLS]标签保留更丰富的特征信息，并使用原始预训练权重，来抽取输入语句的综合特征，网络结构如下：



(c) 模型重组



(d) 模型重构

4.3.2 热度评价指标的建立

当通过MeanShift聚类方法找出热点问题后，需要对问题的热度进行量化。由于问题的热度具有时效性，所以我们需要关注热点问题在某个时间段上的被关注度。

根据题目给出的数据，能够反映热度的指标我们将其归为两类，一类是留言数，一个是点赞数和反对数。前者反映的是当某个问题在某时段内受到的关注越多，那么在该时段与该问题相关的留言在同时段总留言数中应该占有较大的比重。后者则可以表示对该留言的赞同度，如果该问题是急需解决或者争议较多的问题，那么就会有较多的人愿意针对该留言发表自己的意见。此外，评论是群众自发发表的，具有一定主观性，但是点赞数和反对数往往是人们看到了留言才会去思考从而点赞或反对，是被

动的。因此我们认为前者对热度指标的影响应大于后者，故给予较大的权重。

基于以上讨论，热度指标具体定义如下，对于热点问题C_i，它所处的时间范围记为B_i和E_i，其中B_i为起始时间，E_i为结束时间。在B_i至E_i这段时间中，总留言数记为N_i，与该热点问题相关的留言数记为n_i，N_i条留言的总点赞数记为M_i，总反对数记为P_i，每条热点问题相关的留言的点赞数记为$m_{i,j}$,其中$j \in [0, n_i]$，反对数记为$p_{i,j}$,其中$j \in [0, n_i]$,定义热点问题C_i热度指标Q_i为：

$$Q_i = a_1 \frac{n_i}{N_i} + a_2 \frac{\sum_{j=0}^{n_i} m_{i,j}}{M_i} + a_3 \frac{\sum_{j=0}^{n_i} p_{i,j}}{p_i} \tag{13}$$

其中a_1、a_2、a_3表示不同指标的权值，其中$a_1 \geqslant a_2$、$a_1 \geqslant a_3$。

4.3.3 命名实体识别（NER)

命名实体识别也是自然语言处理的一项重要任务，它恰好适合我们所需要的为了获得内容中的实体名称，我们再次对模型的网络进行了调整，使得模型能够输出实体所在的区间信息，并将其抽取。

条件随机场（CRF）层是一种判别式模型，当给定输入随机变量时，可以输出另一组随机变量的条件概率分布。此处，输入随机变量就是BERT在输出全连接后的n个向量，而要计算的条件概率分布的数目，则是该部位对应的分割情形。此时，n个位置的节点所对应的可能的标签为点、其间可能的连接关系，构成一个无向图$G =< V, E >$且有$Y = y_v, v \in V$,此处Y的取值范围是$Y = B, I, O$,分别应识别序列的入点、中间、出点，也可以进行增加，如$Y = B - LOC, I - LOC, O - LOC, B - ORG, I - ORG, O - ORG$,以识别更多的实体类型。定长序列Y的概率可以表示为：

$$exp(\sum_j \lambda_j t_j(y_{i-1}, y_i, x, i) + \sum_k \mu_k s_k(y_i, x, i)) \tag{14}$$

其中μ_k、λ_i是训练而来的参数，t_j为转移概率，s_k为状态函数。输入X和Y如图，每一个向量都是第i个字符通过BERT后的特征表示。CRF层计算以为条件时，Y_i上的条件概率分布。之后使用Viterbi算法进行标注，获得最大概率通路，实现对命名实体的标注。

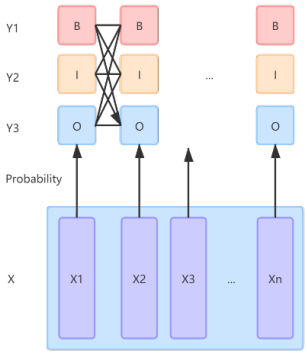


图 5: 条件随机场(CRF)层

4.3.4 模型训练与参数调整

经过多次尝试，聚类模型调整搜索带宽为7.0。

NER模型的训练使用bert4keras提供的脚本进行，训练语料为人民日报的标注命名实体样本（训练集大小约20000个）。

表 5: 其他参数

	训练迭代次数	批次体积	批次数量	嵌入维度	截断长度	学习率	CRF学习倍率
参数值	10	32	652	128	128	10^{-5}	1000

4.3.5 模型结果

聚类模型最终将4325个内容分为4023个类别，寻找到了一些反映相同问题的留言，并用作后续热点指数评定。

用F1-Sore对模型进行评价。NER最终F1-Score为0.92，召回率为0.91，基本可以完成命名实体的提取。

最终热度排名前五的问题如下表所示(具体信息参见附件热点问题表.xls)

表 6: 热度排名前五的问题简表

热度排名	热度指数	时间范围	地点/人群	问题描述
1	106.827	2019/01/15 至 2019/12/24	A市人才	《A市人才购房及购房补贴实施办法...
2	93.782	2019/11/02 至 2020/01/25	丽发新城小区	A市丽发新城小区侧面建设混泥土搅拌...
3	75.577	2019/07/07 至 2019/08/31	伊景园滨河苑	无视职工意愿、职工权益的A市伊景园...
4	44.724	2019/04/11 至 2019/04/11	A市金毛湾	反映A市金毛湾配套入学的问题
5	44.557	2019/01/26 至 2019/09/25	A市地铁1号线	A市地铁1号线延长线到底哪天开工啊？

热点问题留言明细表参见附件热点问题留言明细表.xls。

4.4 答复评价模型

由于留言信息和答复意见具有大篇幅特征，如何对相关部门的答复意见进行评价，实现信息过滤，更高效的对部门工作情况进行评价成为智慧政务的发展方向。

在本章节，我们的目标是对大段的留言文本和相关部门的答复意见进行建模，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建立一套答复评价模型。

我们基于客观性、系统性、可操作性三个原则构建评价指标体系。

1. 客观性。选择的指标不具有主观性，克服因人而异的主观因素的影响。

2. 系统性。能够集合完整形成评价答复质量的体系,各指标间既相互独立又相互联系,共同构成一个有机整体。
3. 可操作性。选取的指标要有实际意义，以便于收集和量化。

基于以上原则，我们构建了“智慧政务”留言答复评价二重指标体系四框架模型，分别从答复可解释性、内容完整性、态度优良性、群众满意度四方面对答复进行考察。

针对答复可解释性的角度，我们用留言和答复之间的文本相似度这个指标来反映。具体实现形式通过对留言和答复基于BERT模型进行命名实体识别，通过比对命名实体的重复率来判断主题的相关性。

针对内容完整性：我们从字数充实度和回复及时性两指标反映。字数充实度：统计留言和答复的字数，用答复的字数除以留言的字数的比率作为字数充实度的指标，比率高说明答复较为充实。回复及时性：对date型数据建模，计算留言和回复的时间差，时间差较短说明留言能够及时得到回复。

针对态度优良性的角度，我们用语气真诚度、敬语使用度、答复耐心度三个指标来反映。语气真诚度通过情感分析，情感分析相当于一个二分分类器，基于Bert进行情感分析模型的训练，给每一个回复的情感积极性进行打分，分数越趋近于1说明情感越积极，越接近0说明回复者的态度较为消极。敬语使用度主要通过调用特定词库，对答复中出现过的敬语做词频统计，通过敬语在答复中所占比率判断回复者态度优良性。按照常理，由于信息可能存在延时性，较长一段时期内，人们会对同一个问题不断留言，回复者可能会对回复失去耐心，表现为对问题回复的字数有所下降或者态度表现出消极倾向，因而对回复做字数统计和情感分析作为判断。

针对群众满意度：我们从点赞数和留言重现率两指标反映。点赞数可以反映群众是否认同该回复，而留言的重现则说明问题并没有被解决，或者对解决方式以及回复不够满意。点赞数：由于不同时期网络流量可能有高有低，所以不直接采用点赞数而是用某个时间段点赞数占同时段所有留言总点数的比例来说明。留言的重现率：利用聚类和NER技术，统计某一留言出现后相似内容和主题的留言再次出现的次数，出现次数多，我们就可以理解为群众对答复不太满意。

具体的答复评价指标体系如下表所示。

表 7: 答复评价指标体系

一级指标	二级指标	指标含义	实现方式
答复可解释性	留言答复相关性	文本相似度（关键词的重复率）	命名实体对应
内容完整性	字数充实度	留言与答复字数的比率	文本字数统计
	回复及时性	留言和回复的时间差	对date型数据建模
态度优良性	语气真诚度	答复情感呈现“positive”的状态	情感分析
	敬语使用度	特定敬语在答复中所占比率	特定词库+词频统计
	答复耐心度	长时期内同一问题答复字数、情感变化	文本字数+情感分析
群众满意度	点赞数	点赞数占留言总赞数的比例	数字统计
	留言重现率	从最早回复时间留言重现比率	聚类+NER

5 模型优化

由以上论述可知，我们的模型都可以出色地完成题目所给的任务。下一步，我们将继续改进模型。由于Trasformer模型的效果极大程度依赖于预训练所使用的语料，因此如果有更多领域相关的文本（如更多的人民群众反馈文本），及更多的运算资源（BERT预训练需要比较高的运算成本），可以对模型进行重新预训练，以对下游任务获得积极的影响。

观察NER的识别结果，可以发现有一些位置的实体抽取存在不完整的情况。原因是模型的训练过程中，几乎没有中英文混杂的实体内容（如“A市，B2县”）。但在实际情况下，该位置应该会有真实的地域名称，NER可以得到更好的结果。

聚类模型已经将大部分同义文本聚集为热点问题，但是依然有很少一部分表述差异较大的同义文本被忽略。在数量级上，当被忽略的文本点赞数较高的时候，这对热点指数会产生不利影响。在有已分类数据的情况下，可以考虑添加全连接层，对模型在特征上进行进一步收束，使其更适合文本聚类的需要。

此外，在工程上，可以对代码进行进一步封装、优化，精简模型参数，使得模型在分类任务上更加适合实时预测。

参考文献

[1] VASWANI A, SHAZEER N, PARMAR N, et al. Attention is all you need[M]//GUYON I, LUXBURG U V, BENGIO S, et al. Advances in Neural Information Processing Systems 30. [S.l.]: Curran Associates, Inc., 2017: 5998-6008.

[2] DEVLIN J, CHANG M W, LEE K, et al. Bert: Pre-training of deep bidirectional transformers for language understanding [J]. 2018.

[3] HEINZERLING B, STRUBE M. Bpemb: Tokenization-free pre-trained subword embeddings in 275 languages[J]. 2017.

[4] LIU Y, OTT M, GOYAL N, et al. Roberta: A robustly optimized bert pretraining approach[J]. arXiv preprint arXiv:1907.11692, 2019.

[5] TSENG H, CHANG P C, ANDREW G, et al. A conditional random field word segmenter for sighan bakeoff 2005[C]// Proceedings of the fourth SIGHAN workshop on Chinese language Processing. [S.l.: s.n.], 2005.

[6] SU J. Bert for keras[EB/OL]. https://github.com/bojone/bert4keras.

[7] SU J. Han language processing[EB/OL]. https://github.com/hankcs/HanLP.

[8] TECHNOLOGY Z. Open source pretrained models for roberta[EB/OL]. https://github.com/ZhuiyiTechnology/pretrained-models.