第八届 “泰迪杯”
全国数据挖掘挑战赛

作品名称:“智慧政务”中的文本挖掘应用研究

摘要

近年来，随着互联网的进一步发展和政府政务工作的完善，“智慧政务”成为处理留言的利器之一随着互联网的介入，电子政务的处理效率越来越快。但是群众的反应量之大，导致人工效率过低，政务人员的处理量大，处理周期长。会导致处理事件不及时，处理周期长的问题。而一般的政务处理系统，往往结构简单，需要大量的人工成本，还需要大量的时间维护。而且智能程度较低，处理效率也不高。为了解决人工效率，时间效率的痛点，借用自然语言处理，以及深度学习模型训练分类问题。加上模型答复意见评价系统 逐渐减少人工的劳动量，进一步智能化处理政务留言。伴随微信平台，支付宝平台，等等的网络平台的发展，这些平台让群众提出意见逐渐便利。便利的收集数据之后，就是面临大基量数据的处理，其中留言分类和热点处理相关的工作需要大量时间，人工成本。而我们抓住痛点，建立智慧政务处理系统即是发展的趋势，而这种智慧智能的处理系统仅需要少部分的人工，和大量的数据即可完成，留言划分，热度问题挖掘，答复意见评价的基本重要步骤。处理系统对于政务的管理水平以及施政效率具有良好的推动，激活作用。

目录

1.问题一分析方法与过程

1.1 主要用到如下步骤。

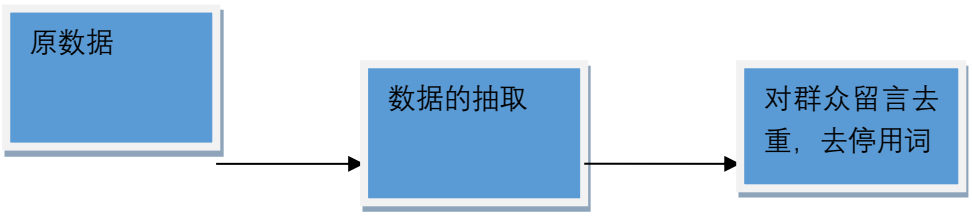
步骤一:对数据的读取，数据的抽取。本数据是 Excel 表，而每列是各种数据，对群众留言内容进行一级标签分类模型。

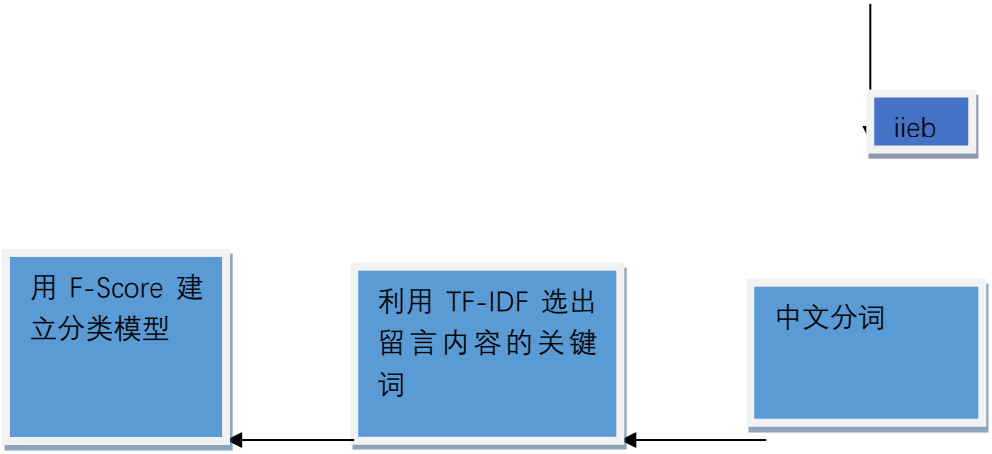
步骤二:数据的预处理，题目给出的数据中，群众留言列中有许多不需要的字符如 x、t 等和一些不需要的标点符号更有一些语气助词，这一类对数据分析是没有任何用处的，我们需要将它们去除掉，我们还需要对数据检查是否有空缺职，如果有对其进行填补，还要对数据进行去重处理，之后我们需要用到中文分词。

步骤三:数据分析，在对数据分析中，首先要对数据进行文本特征选择，用数学方式，选取最是分类的留言特征，如卡方检验，文档频率，之后对得出的特征词进行向量化表示，以提供以挖掘分析使用，在这里中使用 TF－IDF 算法，找出文本中的关键词，把群众留言信息转化为权重向量，采用 F－Score 算法对群众留言进行分类。

步骤四:建立模型，对数据转化为权值向量数据分析建立模型，对模型进行测试与优化。

1.1 流程图





1.2 数据预处理

1.2.1 对群众留言信息去除重，去空

在题目中给出的 Excel 表数据中，在群中留言中，可能有这用户有留言问题存在重复，在某一小区中，多用户遇到某一些问题，会进行反馈留言问题，而其他群众也同时反馈了同样的问题，下都进行了同一类问题的反馈，这就会引起一定的重复数据，所以我们需要用到 pandas 模块中的 drop_duplicates () 方法对文本数据去除重复的数据，在很大的数据中，可能会存在某一处数据为空缺值，则我们需要用 info（）方法，检查数据是否存在空缺值。

1.2.2 对群众留言信息进行中文分词。

在对群众留言进行数据挖掘分析之前，需要对非结构化的文本信息转换为计算机能够识别的结构化信息，在题目提供的附件中，是以中文文本的方式给出了数据，为了便于转换，先要对这些群众留言内容进行中文分词，在分词前，文本数据中都会存在一些不需要的字符如 x、t 等，还有一些不需要的停用词，而我们需要将这些去除掉后再进行中文分词操作，在去除文本中的字符 x、t 等，要用到 re 模块

中 apply 方法，然而在去除停用词中，我们则需要有去除停用词表，而这停用词表，就是自己需要去除的字符和语气词，在编写中导入该表再循环遍历去掉停用词，以上进行后我们采用 Python 的中文分词包，jieba 包进行分词，jieba 分词基于前缀词典实现高效词图，采用基于汉字成词能力的 HMM 模型，能提出文本中词频的最好的分词效果。

TF-IDF 算法步骤

第一步，计算词频：

$$\text{词频(TF)} = \text{某个词在文章中的出现次数}$$

考虑到文章有长短之分，为了便于不同文章的比较，进行"词频"标准化。

$$\text{词频(TF)} = \frac{\text{某个词在文章中的出现次数}}{\text{文章的总词数}}$$

第二步，利用 TF-IDF 来计算权重，计算逆文档频率：

这时，需要一个语料库（corpus），用来模拟语言的使用环境。

$$\text{逆文档频率(IDF)} = \log\left(\frac{\text{语料库的文档总数}}{\text{包含该词的文档数} + 1}\right)$$

如果一个词越常见，那么分母就越大，逆文档频率就越小越接近0。分母之所以要加 1，是为了避免分母为 0（即所有文档都不包含该词）。log 表示对得到的值取对数。

第三步，计算 TF-IDF：

$$\text{TF} - \text{IDF} = \text{词频(TF)} \times \text{逆文档频率（IDF）}$$

可以看到，TF-IDF 与一个词在文档中的出现次数成正比，与该词在整个语言中的出现次数成反比。所以，自动提取关键词的算法就很清楚了，就是计算出文档的每个词的 TF-IDF 权值，然后按降序排列，取排在最前面的几个词。

第三步 计算 TF－IDF 权值 ，后写第四步 利用 F－score 建立模型

F-Score（非模型评价打分，区别与 F1_score）是一种衡量特征在

两类之间分辨能力的方法，通过此方法可以实现最有效的特征选择。

$$F(i) \equiv \frac{\left(\bar{\mathbf{x}}_i^{(+)} - \bar{\mathbf{x}}_i\right)^2 + \left(\bar{\mathbf{x}}_i^{(-)} - \bar{\mathbf{x}}_i\right)^2}{\frac{1}{n_+ - 1}\sum_{k=1}^{n_+}\left(x_{k,i}^{(+)} - \bar{\mathbf{x}}_i^{(+)}\right)^2 + \frac{1}{n_- - 1}\sum_{k=1}^{n_-}\left(x_{k,i}^{(-)} - \bar{\mathbf{x}}_i^{(-)}\right)^2}$$

其中 i 代表第 i 个特征，即每一个特征都会有一个 F-score。x 拔是所有该特征值的平均数，而（+），（-）则分别代表所有阳性样本和阴性样本的特征值（的平均数）。代表 k 是对于具体第 i 个特征的每个实例，分母的两个 sigma 可以理解为阳性样本与阴性样本的特征值的方差。F-score 越大说明该特征的辨别能力越强。

3. 特定热度问题的答复分析

对于特定热度为问题的分析，首先要进行问题分类，在问题分类的基础上通过深度学习中的卷积神经网络来设计答复模型，其关键在于关键词的分析判断，优化模型训练市场。根据评价答复系统的特点，在政务系统中通过留言来收集语料信息，用这些信息构建答复系统的语料库。

基于卷积神经网络的系统设计

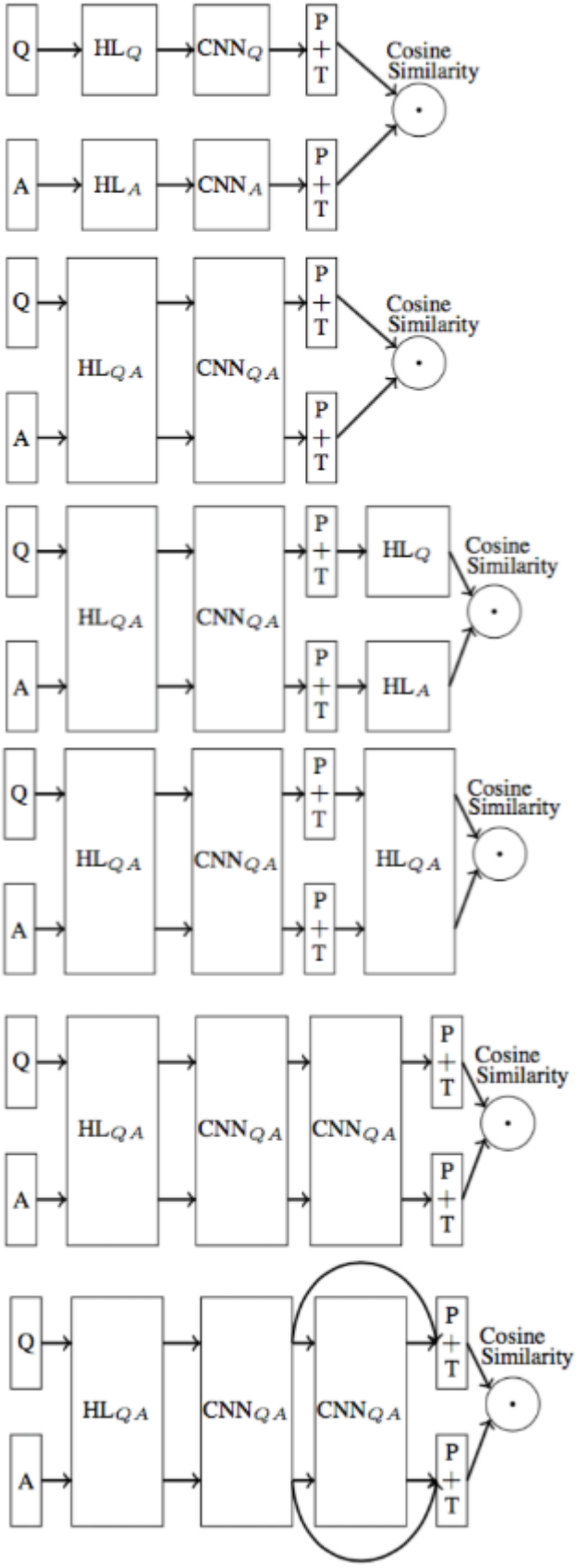
卷积神经网络的三个优点 一、稀疏的交互性，二、参数共享性 三、等价表示性。

所以选择卷积神经网络来进行模型来训练。神经网路的结构设计包括以下六种.

HL 表示 hide layer 隐藏层，它的激活函数设计成

z＝tanh(Wx+B)，CNN 是卷积层，P 是池化层，池化步长为 1，T 是 tanh 层，P+T 的输出是向量表示，最终的输出是两个向量的 cos 相似度

图中 HL 或 CNN 连起来的表示他们共享相同的权重。CNN 的输出是几维的取决于做多少个卷积特征，如果有 4 个卷积，那么结果就是 4*3 的矩阵。


Q
A
HL_Q
HL_A
CNN_Q
CNN_A
HL_{QA}
CNN_{QA}
P + T
Cosine Similarity

2 答复模型构建

第一步，卷积运算在一定的范围内做平移并取平均值

$$\int_{-\infty}^{\infty} f(\tau)g(x-\tau)d\tau$$

总之卷积就是先打乱，再叠加。对 t 积分，而不是对于 x 积分。就是对于固定的 x，找到 x 附近的所有变量，并且求两个函数的乘积，并求和。

第二步，多层卷积和池化

利用一次卷积运算(哪怕是多个卷积核)提取的特征往往是局部的，难以提取出比较全局的特征，因此需要在一层卷积基础上继续做卷积计算 ，这也就是多层卷积

池化是一种降维的方法。按照卷积计算得出的特征向量维度大的惊人，不但会带来非常大的计算量，而且容易出现过拟合，解决过拟合的办法就是让模型尽量“泛化”，也就是再“模糊”一点，那么一种方法就是把图像中局部区域的特征做一个平滑压缩处理，这源于局部图像一些特征的相似性(即局部相关性原理)。

具体做法就是对卷积计算得出的特征在局部范围内算出一个平均值(或者取最大值、或者取随机采样值)作为特征值，那么这个局部范围(假如是 10*10)，就被压缩成了 1*1，压缩了 100 倍，这样虽然更“模糊”了，但是也更“泛化”了。通过取平均值来池化叫做平均池化，通过取最大值来池化叫做最大池化。

第三步，模型构建

卷积核中的因子(×1 或×0)其实就是需要学习的参数，也就是卷积核矩阵元素的值就是参数值。多层神经网络为了方便用链式求导法则更新参数，我们设计 sigmoid 函数作为激活函数，我们同时也发现卷积计算实际上就是多层神经网络中的 Wx 矩阵乘法，同时要加上一个偏执变量 b，那么前向传到的计算过程就是：

$$h_{W,b}(x) = a_1^{(3)} = f(W_{11}^{(2)}a_1^{(2)} + W_{12}^{(2)}a_2^{(2)} + W_{13}^{(2)}a_3^{(2)} + b_1^{(2)})$$

如果有更多层，计算方法相同

因为是有监督学习，所以模型计算出的 y'和观察值 y 之间的偏差用于更新模型参数，参数更新公式是：

$$W_{ij}^{(l)} = W_{ij}^{(l)} - \alpha\frac{\partial}{\partial W_{ij}^{(l)}}J(W,b)$$

$$b_i^{(l)} = b_i^{(l)} - \alpha\frac{\partial}{\partial b_i^{(l)}}J(W,b)$$

偏导计算公式是：

$$\frac{\partial}{\partial W_{ij}^{(l)}}J(W,b;x,y) = a_j^{(l)}\delta_i^{(l+1)}$$

$$\frac{\partial}{\partial b_i^{(l)}}J(W,b;x,y) = \delta_i^{(l+1)}.$$

其中 a 的计算公式是：

$$
\begin{aligned}
a_1^{(2)} &= f(W_{11}^{(1)}x_1 + W_{12}^{(1)}x_2 + W_{13}^{(1)}x_3 + b_1^{(1)}) \\
a_2^{(2)} &= f(W_{21}^{(1)}x_1 + W_{22}^{(1)}x_2 + W_{23}^{(1)}x_3 + b_2^{(1)}) \\
a_3^{(2)} &= f(W_{31}^{(1)}x_1 + W_{32}^{(1)}x_2 + W_{33}^{(1)}x_3 + b_3^{(1)})
\end{aligned}
$$

残差 δ 的计算公式是：

$$
\begin{aligned}
\delta_i^{(n_l)} &= \frac{\partial}{\partial z_i^{n_l}} J(W,b;x,y) = \frac{\partial}{\partial z_i^{n_l}} \frac{1}{2} \left\| y - h_{W,b}(x) \right\|^2 \\
&= \frac{\partial}{\partial z_i^{n_l}} \frac{1}{2} \sum_{j=1}^{S_{n_l}} (y_j - a_j^{(n_l)})^2 = \frac{\partial}{\partial z_i^{n_l}} \frac{1}{2} \sum_{j=1}^{S_{n_l}} (y_j - f(z_j^{(n_l)}))^2 \\
&= -(y_i - f(z_i^{(n_l)})) \cdot f'(z_i^{(n_l)}) = -(y_i - a_i^{(n_l)}) \cdot f'(z_i^{(n_l)})
\end{aligned}
$$

$$
\begin{aligned}
\delta_i^{(n_l-1)} &= \frac{\partial}{\partial z_i^{n_l-1}} J(W,b;x,y) = \frac{\partial}{\partial z_i^{n_l-1}} \frac{1}{2} \left\| y - h_{W,b}(x) \right\|^2 = \frac{\partial}{\partial z_i^{n_l-1}} \frac{1}{2} \sum_{j=1}^{S_{n_l}} (y_j - a_j^{(n_l)})^2 \\
&= \frac{1}{2} \sum_{j=1}^{S_{n_l}} \frac{\partial}{\partial z_i^{n_l-1}} (y_j - a_j^{(n_l)})^2 = \frac{1}{2} \sum_{j=1}^{S_{n_l}} \frac{\partial}{\partial z_i^{n_l-1}} (y_j - f(z_j^{(n_l)}))^2 \\
&= \sum_{j=1}^{S_{n_l}} -(y_j - f(z_j^{(n_l)})) \cdot \frac{\partial}{\partial z_i^{(n_l-1)}} f(z_j^{(n_l)}) = \sum_{j=1}^{S_{n_l}} -(y_j - f(z_j^{(n_l)})) \cdot f'(z_j^{(n_l)}) \cdot \frac{\partial z_j^{(n_l)}}{\partial z_i^{(n_l-1)}} \\
&= \sum_{j=1}^{S_{n_l}} \delta_j^{(n_l)} \cdot \frac{\partial z_j^{(n_l)}}{\partial z_i^{n_l-1}} = \sum_{j=1}^{S_{n_l}} \left(\delta_j^{(n_l)} \cdot \frac{\partial}{\partial z_i^{n_l-1}} \sum_{k=1}^{S_{n_l-1}} f(z_k^{n_l-1}) \cdot W_{jk}^{n_l-1} \right) \\
&= \sum_{j=1}^{S_{n_l}} \delta_j^{(n_l)} \cdot W_{ji}^{n_l-1} \cdot f'(z_i^{n_l-1}) = \left(\sum_{j=1}^{S_{n_l}} W_{ji}^{n_l-1} \delta_j^{(n_l)} \right) f'(z_i^{n_l-1})
\end{aligned}
$$

参考文献

1.赵琳瑛.基于隐马尔科夫模型的中文命名识别研究.西安电子科技大学.2007

2.《Combining SVMs with Various Feature Selection Strategies》

3. 二十七-用深度学习来做自动问答的一般方法 www.shareditor.com