“智慧政务”中的文本挖掘应用

摘要

文本的数据挖掘就是指从文本数据中抽取有价值的信息和知识的计算机处理技术。随着近些年科技水平的发展，各种社交和评论的软件如：微信、微博、市长信箱、阳光热线等出现，为政府能进一步了解民意、汇聚民智、凝聚民气提供了重要的渠道，并配合云计算、大数据、人工智能等技术，为格相关部门更好的处理众多民意产生的文本数据，本文介绍了通过神经网络及程序软件对数据的预处理等方法，将各民意留言进行划分，分类，筛选，并给出一套完整的方案，这可以为相关部门对民情民意的了解及答复提供参考依据。

关键词： 神经网络 Python 标签分类

Text Mining Application in "Smart Government Affairs"

Abstract

Text data mining refers to computer processing technology that extracts valuable information and knowledge from text data. With the development of science and technology in recent years, various social and commenting software such as WeChat, Weibo, mayor's mailbox, and sunshine hotline have emerged, providing an important channel for the government to further understand public opinion, gather people's wisdom, and gather people's popularity. , And cooperate with cloud computing, big data, artificial intelligence and other technologies, in order to better handle the text data generated by many public opinion departments, this article introduces the method of preprocessing the data through neural networks and program software, etc. Divide, classify, screen, and give a complete set of plans, which can provide a reference for the relevant departments to understand the public opinion and reply.

Keywords: Neural Network Python Label classification

目录

绪论

近些年来，微信、微博、市长信箱等聊天平台已经成为政府了解民情民意的重要渠道，也因如此，随着使用者的增多，有这类软件而产生的各类设情民意相关的文本数据不断攀升，给靠着人工来进行留言划分和热点整理得各部门人员带来巨大的挑战。

本着这个问题的背景，本文旨在通过建立数学模型和文本挖掘的方法对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进行整理划分和归类。

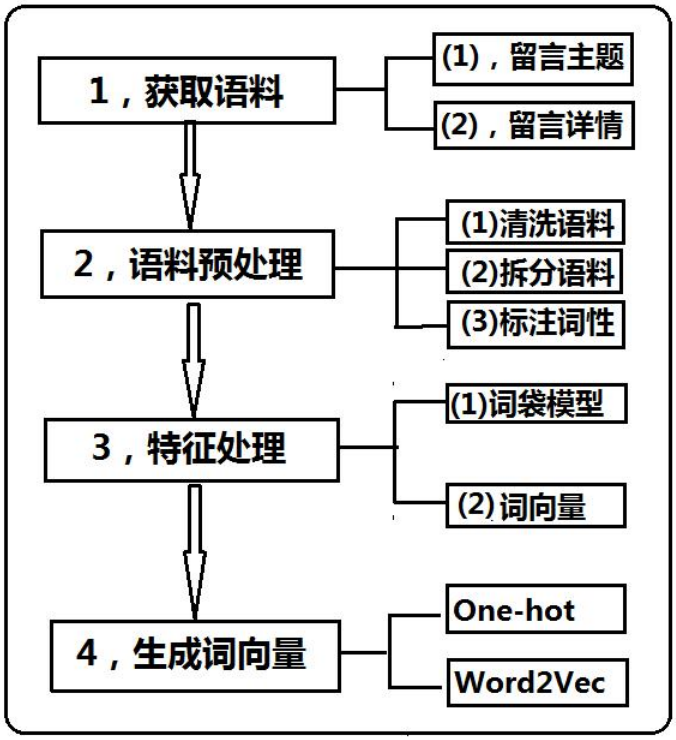
一．群众留言分类

1.1 建立关于留言内容的一级标签分类模型

首先根据表格附件二中的留言详情，对其归类，预测划分出一级标签，对照已划分好的一级标签，然后得出其准确率。关于处理附件二中的留言详情，则可用到自然语言处理，将其中的文本文字向量化。

1.1.1 自然语言处理介绍及问题处理

自然语言处理(Natural Language Processing,NLP)，研究在人与人交互中以及在人与计算机交互中的语言问题的一门学科。NLP 算法的一般步骤如下：



此为自然语言处理(NLP)算法流程图，本文采用数字类别 1、2、3、4、5、6、7，分别表示附件 2 的一级标签“城乡建设”、“环境保护”、“交通运输”、“教育文体”、“劳动和社会保障”、“商贸旅游”、“卫生计生”7 个类别标签。

然后将每个留言详情的每个字符转化为一个向量，根据此方法，将每条留言详情进行向量化，由此可得出稀疏矩阵，在此我们选择附件二中的前 2500 行进行训练，具体结果如附件 2-1-0-0，提取其中几行如图 1.1：

	0	1	2	3	4	5	6	7	8	9
0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0	0	0

图 1.1

该图为前几行留言详情对应稀疏矩阵的一部分。

用此方法就，我们将所有的留言详情的字符型转化为了数值型，接着对此进行分类处理。在此我们便用到了神经网络，神经网络是由具有适应性的简单单元组成的广泛并行互连的网络，它的组织能够模拟生物神经系统对真实世界物体所做出的交互反应。在此，我们主要用到 BP 神经网络对此进行分类。

1.1.2 BP 神经网络介绍及问题处理

BP 神经网络是一种按照误差逆向传播算法训练的多层前馈神经网络，是应用最广泛的神经网络。可应用于函数逼近、模式识别/分类、数据压缩等，其主要是根据输出，从后向前逐层传播输出层的误差。其计算过程分为两个部分：第一部分，正向传播过程，即输入信息从输出层经隐含层逐层计算各个神经元的输出值；第二部分，反向传播过程，输出层误差逐层向前计算隐含层各个单元的误差，并修正权值。

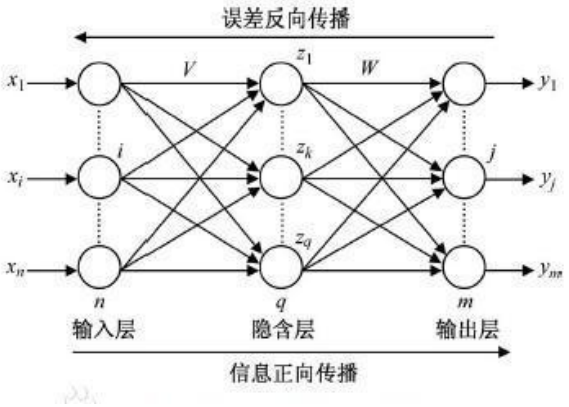


图 1.2BP 神经网络拓扑结构图

给定训练集 $\{(x_1,y_1),(x_2,y_2),\cdots,(x_n,y_n)\}$，其中 $x\in R^d, y\in R^l$，即神经网络的输入是 d 维向量，输出是 l 维。输入神经元的输出传播给隐含层。

1，前向传播

BP 神经网络的第二层为隐含层，输入层和隐含层之间为连接权值 $\{v_1,v_2,\cdots,v_m\}$，隐含层和输出层的链接权值为 $\{w_1,w_2,\cdots,w_o\}$；其中，m 为隐含层神经元的数量。则神经元的输入：

$$Net_{in}=\sum_{i=1}^{n}v_i x_i$$

神经元的输出为：

$$Net_{in}^{j} = f(Net_{in} - \theta_j)$$

式中，θ表示神经元的阈值，只有当神经元接收到的信息达到阈值神经元才会被激活。$f(x)$表示激活函数，本文采用 sigmoid 激活函数，计算公式如下。

$$f(x) = 1/(1+e^{-x})$$

2，反向传播

BP 神经网络采用梯度下降法更新网络参数，神经网络的输出值为y_i，期望值为t_i，则神经网络的误差值为

$$E_i = \frac{1}{2}\sum_{i=1}^{l}(y_i - t_i)^2$$

则输出层和隐含层、隐含层和输入层的权值和阈值采用下式更新。

$$w_{ij} = w_{ij} + \alpha\frac{\partial E(w,\theta)}{\partial w_{ij}}x_i$$

$$\theta_j = \theta_j + \alpha\frac{\partial E(w,\theta)}{\partial \theta}$$

由于附件 2 的数据样本量较大，若全部使用则计算用时较长，本文随机抽取 2000 个样本验证所建模型。从 2000 个样本中随机抽取 10%的样本作为测试集，剩下的样本作为训练集。则本文 BP 神经网络的输入是 NLP 生成的词向量，输出为附件 2 的 1、2、3、4、5、6、7 的数字类别标签。

模型验证

经过多次试验，最终本文 BP 神经网络的隐含层神经元个数$n = 200$，学习率$\alpha = 0.08$。根据所给的数据集，得到 BP 神经网络的训练结果如图 1.3

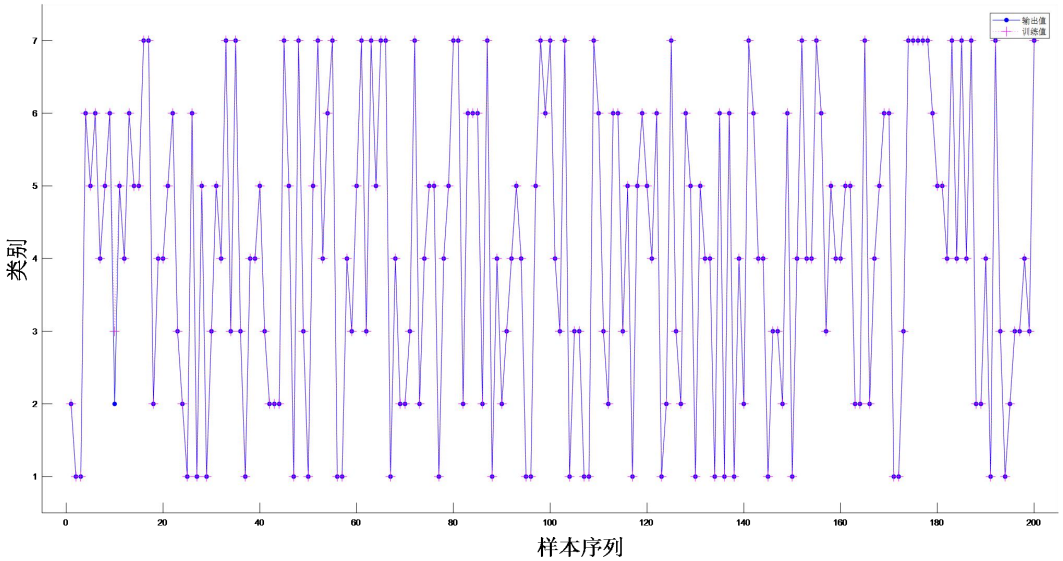


图 1.3 训练结果图

根据训练好的 BP 神经网络，对预测集进行预测，预测结果如表一和图所示。

表 1　NLP 结合 BP 神经网络的预测结果

实际类别	预测类别	实际类别	预测类别	实际类别	预测类别
1	4	1	1	1	1
1	1	1	1	1	1
1	1	1	1	5	5
1	6	1	1	1	1
1	1	4	4	4	4
4	4	1	1	1	1
1	1	1	1	1	1
1	1	4	4	4	4
1	1	1	1	1	1
6	6	1	1	1	1
1	1	1	1	1	1
1	1	1	1	1	1
1	1	1	1	1	1
6	6	1	1	1	1
1	1	1	1	1	1
1	1	1	1	1	1
1	1	1	1	1	1
1	1	1	1	1	1
1	1	1	1	1	1
1	1	1	1	1	1
4	4	1	1	1	5
1	1	1	1	1	1
1	1	2	2	1	1
6	6	2	2	6	6
1	2	1	6	1	1
1	1	1	1	1	1
1	1	1	1	1	1
1	1	1	1	1	1
7	7	1	1	1	1
1	1	1	6	1	1
6	6	1	1	4	4
1	1	1	1	1	1
1	1	1	1	1	1
1	1	1	1	1	1
1	1	1	1	1	1
1	1	4	4	1	1
4	4	1	1	1	1
1	1	1	1	4	4
1	1	4	4	1	1
1	1	1	6	2	2

1	1	1	1	1	1
1	1	1	1	1	1
1	1	1	1	1	1
1	1	1	1	1	1
1	1	4	4	1	1
1	1	1	1	1	1
1	1	1	1	4	4
1	1	1	1	1	1
1	1	1	1	1	1
1	1	1	1	1	1
1	1	6	6	1	6
1	1	1	1	1	1
1	1	1	1	2	2
1	1	1	1	1	1
1	1	4	4	1	1
1	1	1	1	4	4
1	1	1	1	1	1
1	1	1	1	1	1
1	1	1	1	1	1
1	1	3	3	1	1
1	1	1	1	1	1
1	1	1	1	1	1
1	1	4	4	1	6
1	1	1	1	1	1
1	1	1	6	1	1
1	1	1	1	1	1
1	1	1	1	--	--

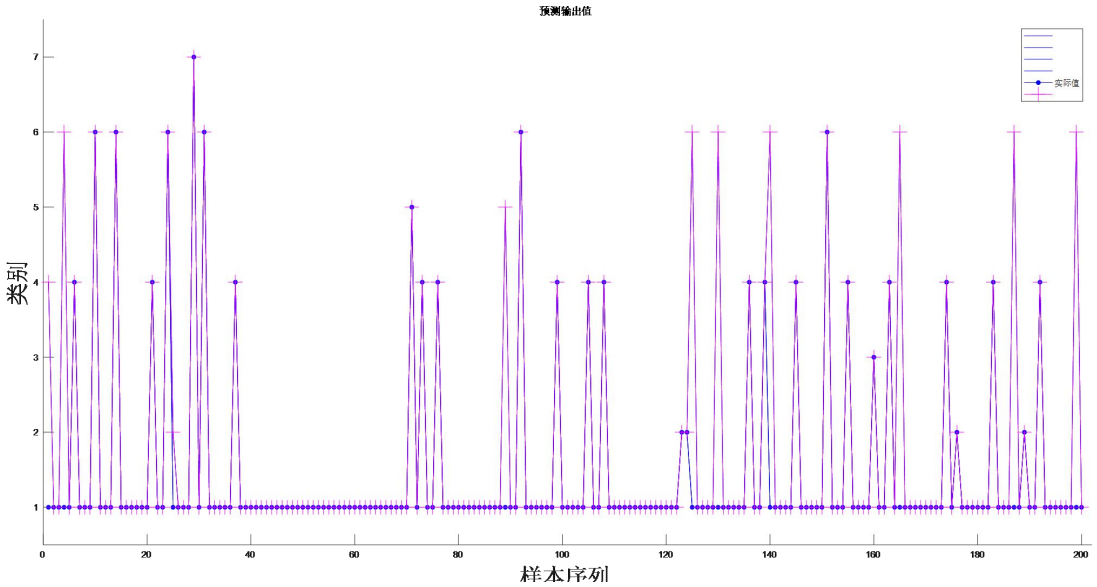


图 1.4BP 神经网络的预测效果图

表 1、图 2 表明，训练结果与统计值吻合程度很高；其中分类错误的样本个数有 4 个，即 NLP 结合 BP 神经网络的预测正确率为 98%，这表明可以采用所建立的模型进行类别划分。

1.2 使用 F-Score 对分类方法进行评价

根据此算法所的得到的结果，我们应用 F-Score 对此方法进行分类评价，首先，给出函数：

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

其中 P_i 为第 i 类查准率， R_i 为第 i 类查全率，结合上面的预测类别与实际类别可得，求出其每一类的真正例 TP，假反例 FN，假正例 FP，真反例 TN，由公式 $P = \frac{TP}{TP + FP}$ 与公式 $R = \frac{TP}{TP + FN}$，求出每一类的查准率与查全率，在代入上式中求出 F_1，根据上述表述得出经计算得出表 2：

i	P_i	R_i
1	0.9470588235294117	1.0
2	1.0	0.8
3	1.0	1.0
4	1.0	1.0
5	1.0	0.5
6	1.0	0.46153846153846156

表 2

再用此数据求出 $F_1=\frac{1}{n}\sum_{i=1}^{n}\frac{2P_iR_i}{P_i+R_i}=0.8599906950996153$，综合上述所有结论，通过 BP 神经网络得出预测的正确率为 98%，而用 F-Score 方法得出的查全率，查准率及 F_1 的值除了第 6 类均较高。

分析：第 6 类的查全率方面精准度不是太高，由于表示查全率 $R_i=\frac{TP}{TP+FN}$ 中，假反率 FN 通过自然语言处理对留言详情进行处理，将其变为一个稀疏矩阵，根据稀疏矩阵的特征再将其进行分类，而第六类中的一些特征与前几类的一些留言详情相似，因此导致假反率较低，但总体而言该分类方法效果较好，可以应用于关于标签分类模型。

二．热点问题挖掘

2.1 进行归类及评价

根据问题一的解决过程，对附件 3 作出同样的处理方式，类似于之前，首先应用自然语言处理将其进行分类，具体的分类情况见附件“热点问题表”，截取其中几行为见图 2.1：

留言编号	留言用户	留言主题	留言时间	留言详情	反对数	点赞数	类别归类	点赞数/反对数之和
188006	A000102948	米阳光婚纱艺术摄影是否合法纳	2019/2/28 11:25:05	，因为地处居民楼内	0	0	7	0
188007	A00074795	路命名规划初步成果公示和城	2019/2/14 20:00:00	10年都未曾更换过，	0	1	7	1
188031	A00040066	春华镇金鼎村水泥路、自来水到	2019/7/19 18:19:54	了，且天还没黑就开灯	0	1	7	1
188039	A00081379	路步行街大古道巷住户卫生间	2019/8/19 11:48:23	进行清扫。没有解决相	0	1	2	1
188059	A00028571	际社区三期与四期中间空地夜门	2019/11/22 16:54:42	给投诉业主，态度强	0	0	7	0
188073	A909164	单方面改变麓谷明珠小区6栋架	2019/3/11 11:40:42	何政府调规、改建的	0	0	7	0
188074	A909092	区富绿新村房产的性质是什么	2019/1/31 20:17:32	让给业主了，然而因	0	0	7	0
188119	A00035029	对A市地铁违规用工问题的质疑	2019/5/27 16:04:44	加班还扣钱，扣身份	0	0	2	0
188170	A88011323	A市6路公交车随意变道通行	2019/12/23 8:50:24	该司机并未按地面车	0	0	7	0
188249	A00084085	梧桐梓坡路与麓松路交汇处地铁	2019/9/17 4:25:00	边邻居也是苦不堪言	0	0	7	0
188251	A00013092	东四路口晚高峰太堵，建议调	2019/10/19 11:02:40	下至少两到三个信号	0	0	7	0
188260	A00053484	小区乐果果零食炒货公共通道	2019/5/31 17:06:13	零食炒货公共通道摆放	0	0	7	0
188396	A00047580	禁在西地省商学院宿舍旁安装	2019/4/15 16:23:09	《中小学校校园环境	2	1	5	3

图 2.1

该图为运用自然语言处理对附件 3 进行分类后所得的结果的一部分，该类问题的解决方法与上述问题相似，从留言的详情为依据，建立稀疏矩阵，根据其特征性进行划分与归类，其中的 1 到 7 的类别分别为政法，城乡建设，卫生生计，教育，商业，生活保障，交通这些类别，将每一个问题的留言详情分到相应的标签下。

然后再对该表格数据进行热度的排名，具体情况见附件“热点问题表”，截取其中一部分如图 2.2：

留言编号	留言用户	留言主题	留言时间	留言详情	反对数	点赞数	NLP+BP 类别归类	点赞数/反对数之和
208636	A00077171	区汇金路五矿万境K9县存在一系	2019/8/19 11:34:04	狗咬人，请问有人对	0	2097	7	2097
223297	A00087522	反映A市金毛湾配套入学的问题	2019/4/11 21:02:44	益纳入配套入学，A3区	5	1762	7	1767
220711	A00031682	请书记关注A市A4区58车贷案	2019/2/21 18:45:14	案情消息总是失望，	0	821	7	821
217032	A00056543	A市58车贷特大集资诈骗案保	2019/2/25 9:58:37	小股东、苏纳弟弟苏	0	790	7	790
194343	A00010616	A市58车贷案警官应跟进关注	2019/3/1 22:12:30	经侦并没有跟进市领	0	733	7	733

图 2.2

该图为排名前五的热点问题。

对于该问题而言，对其进行排序的依据为各类问题的关注度（即反对数与点赞数之和），然后将其按照此依据进行排列。

三．答复意见的评价

根据附件 4 给出的数据，可以看出，对于每个问题相关部门都给出了解决的手段，例如从留言编号为 35492 的问题，他是说在路口没有装上红绿灯摄像头，导致车辆的拥堵，人们生活的不便以及生命安全问题，对于给问题给出的意见为在路段装上摄像头及红绿灯，并随时监控现场，保证人们生活便利与人生安全，这与提出的问题的关键点非常接近。从这点看出对于相关性比较强。

从完整性方面来看，有些问题的答复比较合理长远，但也有些答复并不是非常满意，从长远方面来看它只解决了近期的问题，若时间长久的话，它便不适用了，会产生许多漏洞，因此从完整性来看需要对一些问题采取新的方法，达到长治久安的效果。

关于解决这些问题，我们应该先根据一些问题的关注度来说，先分类问题的标签，然后在各标签对应的问题，按照关注度从大到小的顺序，逐一实行，但需要间隔，在某一问题方案实行后一段时间，观察其反馈效果，若是良好则可以提供给下一问题，若反馈结果不理想，则需采用新的手段制作新的解决方法。

结束语

该模型将实现政务大数据技术与理论研究在我国政务服务领域处理，为市区智慧政务发展与管理水平提升提供更加科学的决策支持，促进市区政务服务的技术进步和创新发展。该模型的建设思想也可以应用于其他行业，能够促进我国经济发展和社会进步。

在信息技术快速发展的时代，“智慧政务”模型为现代化的政府治理打开了新世界的大门，“互联网 + 行政服务解决方案”以进一步转变政府职能、改进行政审批方式、提高行政效率和公共服务水平、方便企业及群众办事为目标，应用“互联网 + 政务信息技术”强化行政审批监管，积极推进网上审批及标准管理措施，进一步加强行政服务中心的规范建设，努力构建务实、高效、便民的服务型政府。

“智慧政务”模型将走进人民的生活，是提升政府人性化服务的重要模型，扩大了政府的服务对象与服务范围，公民参与到政务中来，及时了解国家发展态势与政务，增加公众的政务意识，有利于共同维护社会稳定，推动整个社会有序发展。

最后要实时根据当地政务发展状况不断进行改进，听取用户反馈，政府主动加强与用户之间的联系将“智慧政务”模型进行衔接和规划建设。

参考文献

[1]刘洁.智慧政务 APP 应用问题与对策[J].合作经济与科技,2020(08):169-171.

[2]阳敏辉.智慧城市信息智能服务模式的构建[J].山西建筑,2020,46(07):195-197.

[3]高永强. 以新金融理念推进智慧政务建设[N]. 重庆日报,2020-03-20(009).

[4].安徽省智慧政务新模式及典型应用[J].大数据,2020,6(02):107-112.

[5]周君,王显强.新型智慧城市下政务数据安全管理的研究[J].信息通信技术与政策,2020(03):29-33.

[6]李思思.基于政府治理能力现代化的智慧政务建设探析[J].中国管理信息化,2019,22(24):142-143.

[7]梁丹蕾. 智慧政务平台绩效评价信息集成模型构建及实证研究[D].湘潭大学,2019.

[8]王颖. 基于 SERVQUAL 模型的智慧政务服务质量评价研究[D].华中科技大学,2019.

[9]李军,乔立民,王加强,高杰.智慧政务框架下大数据共享的实现与应用研究[J].电子政务,2019(02):34-44.

[10]杨琴,陈银. 一种大数据智慧政务平台信息推送方法[P]. CN109165255A,2019-01-08.

[11]本刊编辑部.智慧政务:创新社会治理[J].中国建设信息化,2018(23):6-7.

[12]杨慧珍. 电子政务助推“智慧政务”转型若干建议[N]. 山西经济日报,2018-11-06(007).

[13].智慧政务新模式下的公共服务供给改革研究[J].机构与行政,2018(06):46-52.

[14]张娟. 面向智慧政务的地市级政务管理系统的设计与实现[D].天津大学,2018.

[15]张建光. 智慧政务信息生态协同演化机制研究[D].中央财经大学,2016.

附录：

```
#F-Score 评价代码
import numpy as np
import pandas as pd
y_true = pd.read_excel('D:\\文件\\实际结果.xlsx')
y_predict = pd.read_excel('D:\\文件\\预测结果.xlsx')
y_true=y_true.values
y_predict=y_predict.values
F1=0
for i in range(1,7):
```

```
    TNi=np.sum((y_true!=i)&(y_predict!=i))
    FPi=np.sum((y_true==i)&(y_predict!=i))
    FNi=np.sum((y_true!=i)&(y_predict==i))
    TPi=np.sum((y_true==i)&(y_predict==i))
    print(TPi,FPi)
    Pi= TPi/(TPi+FPi)
    Ri= TPi/(FNi+TPi)
    F1=(2*Pi*Ri/(Pi+Ri))/6+F1
    print(Pi,Ri)
print(F1)

#BP 神经网络
import jieba
import jieba.analyse
import pandas as pd
from gensim.models import word2vec
import numpy as np

from sklearn.metrics import classification_report
from sklearn.neural_network import MLPClassifier
###

from sklearn import preprocessing
data = pd.read_csv('留言主题-0.txt','r','UTF-8')
data0 = pd.read_csv('结果 1-0.txt','r','UTF-8')
data1 = data.values

enc = preprocessing.OneHotEncoder()
enc.fit(data1)
aa = np.array(data1)
b = enc.transform(aa).toarray()
y = data0.values
data_in = np.column_stack((b,y))

np.random.shuffle(data_in)
x_train = data_in[0:1751,0:-1]
y_train = data_in[0:1751,-1]
x_test = data_in[1752:-1,0:-1]
y_test = data_in[1752:-1,-1]

bp                                                                    =
MLPClassifier(hidden_layer_sizes=(1000, ),activation='relu',solver='lbfgs',alph
a=0.0001,batch_size='auto',learning_rate='constant')
bp.fit(x_train,y_train)
```

```
y_predict = bp.predict(x_test)
from sklearn.metrics import accuracy_score

bp                                                                    =
MLPClassifier(hidden_layer_sizes=(200, ),activation='logistic',solver='lbfgs',a
lpha=0.0001,batch_size='auto',learning_rate='constant')
bp.fit(x_train,y_train)
y_predict = bp.predict(x_test)
print('正确率: \n',str(accuracy_score(y_test,y_predict)))
```