Spam Message Classification Based on the Naïve Bayes Classification Algorithm

Bin Ning, Wu Junwei, Hu Feng

***Abstract*—A classification model based on the naïve Bayes algorithm is proposed to classify spam messages more effectively. Spam message classification models based on the naïve Bayes algorithm are constructed both for multi-classification and multi-two-classification through steps involving text preprocessing based on regular expression and feature extraction based on Jieba segmentation and the TF-IDF (term frequency–inverse document frequency) algorithm. By further comparing the classification performance against the support vector machine and random forest algorithms, the naïve Bayes algorithm based on multi-two-classification is shown to be the best.**

***Index Terms*—Naïve Bayesian, spam message, classification**

I. Introduction

As a convenient communication method with good mobility and low cost, the short message service has gradually affected more people's lives in the modern information era. However, with the increasing popularity of short message service, the problem of spam messages has become increasingly more serious, which has severely affected not only people's normal lives but also social stability and public security [1]. Therefore, filtering spam message has become an important task that must be solved urgently, and research on technology for the intelligent classification of spam messages is of great significance.

The technology of filtering spam message currently used generally includes black-and-white list technology [2], the rules of matching [3] and so on. When implementing black-and-white list technology, the black-and-white list is maintained by a third party. This method is dynamically querying whether a certain IP address is in the list by ways of DNS. However, the method will be limited if a dynamic or hidden IP is used by the other side. The fundamental principle of the rules of matching is to determine whether it is a spam message based on the comparison result with the presupposed rules. However, these presupposed rules generally are set statically without a credible knowledge learning strategy, so they have poor filtration efficiency and low filtration accuracy in application fields, without obvious rules [4].

As a kind of content-based filtering technology, the naïve Bayes algorithm is regarded highly for its simple and easily understood theoretical rules, rapid classification speed and high classification accuracy; thus, it is widely applied in text filtering [5].

A type of message classification model based on Naïve Bayes algorithm is presented in this paper. Focusing on spam message data from a mobile operator, the model first classifies the original data set into seven types of message data; then, the model analyses and studies the classification performance of the naïve Bayes algorithm with multi-classification and multi-two-classification after text preprocessing and feature extraction based on the TF-IDF (term frequency–inverse document frequency) algorithm; and the model further studies various latitudinal features with various TF-IDF weights. Moreover, the model compares the classification efficiency of the multi-two-classification naïve Bayes algorithm with the support vector machine and random forest algorithms, and the results show that the multi-two-classification naïve Bayes algorithm has the best classification efficiency.

II. Naïve Bayes Algorithm

As a kind of classification method based on Bayes theorem with the assumption of characteristic conditional independence [6], the naïve Bayes algorithm is a highly applied method of Bayes learning. Its performance can be compared with those of the decision tree and the neural network algorithm in some specific applications, but its computation complexity is much less than that of other algorithms.

1. The classification principle and process of the naïve Bayes algorithm

The naïve Bayes classification is defined as follows [7]:

① Providing $x=\{a_1,a_2,...,a_m\}$ is an item needed to be classified, and each a is one of feature attributes of x;

②Classification set $C=\{y_1,y_2,...,y_n\}$ is available;

③Calculate $P(y_1|x),P(y_2|x),...,P(y_n|x)$;

④If $P(y_k|x)=\max\{P(y_1|X),P(y_2|X),...,P(y_n|X)\}$, $X\in y_k$.

The key now is how to calculate the conditional probability in step 3. We can perform the calculation as follows:

Manuscript received December 5th, 2017; revised November 9th, 2018. This work is supported by Guangzhou philosophy and social science "13th Five-Year" project planning (2017GZYB13, 2017GZYB98), Guangdong philosophy and social science "12th Five-Year" project planning (GD14YGL01), Guangdong supporting key discipline construction project of philosophy and social science (GDXK201716), the National Natural Science Fund of China (71571053).

Bin Ning is with the school of Management, Guangdong University of Technology, Guangzhou, 510520, Guangdong, China (corresponding author:+8613580523714; e-mail: bn_gdut@163.com).

Wu Junwei is with the school of Management, Guangdong University of Technology, Guangzhou, 510520, Guangdong, China (e-mail:117944634@qq.com).

Hu Feng is with the school of Management, Guangdong University of Technology, Guangzhou, 510520, Guangdong, China (e-mail: phoenin@163.com).

①Find a collection set to be classified with known classification, which is called the training sample set.

② Obtain the conditional probability of various characteristic attributes of various categories, that is,

$$P(a_1 \mid y_1), P(a_2 \mid y_1),..., P(a_m \mid y_1);$$
$$P(a_1 \mid y_2), P(a_2 \mid y_2),..., P(a_m \mid y_2);$$
$$.....$$
$$P(a_1 \mid y_n), P(a_2 \mid y_n),..., P(a_m \mid y_n) \quad (1)$$

③ If each characteristic attribute is conditionally independent, the following deduction can be obtained according to Bayes' theorem:

$$P(y_i \mid x) = \frac{P(x \mid y_i) \times P(y_i)}{P(x)} \quad (2)$$

As the denominator is a constant for all classifications, only the numerator must be maximized. Furthermore, as each characteristic attribute is conditionally independent, we have the following:

$$P(x \mid y_i)P(y_i) = P(a_1 \mid y_i)P(a_2 \mid y_i)...P(a_m \mid y_i) = P(y_i)\prod_{j=1}^{m} P(a_j \mid y_i) \quad (3)$$

2. The application of the naïve Bayes algorithm for text classification

As a type of classification method based on Bayes theorem, the naïve Bayes algorithm can be used in text classification [8].

Providing the training set is $D_t = \{dt_1,..., dt_n\}$, the classification set is $C = \{c_1,..., c_p\}$, the feature set is $\mathrm{T} = \{\mathrm{t}_1,..., \mathrm{t}_n\}$, the test text is $\mathrm{d}_e = \{(t_1, w_1),..., (t_n, w_n)\}$, and the weight set in the test text is $\mathrm{W} = \{w_1,..., w_n\}$, the basic principle of the naïve Bayes algorithm in text classification is as follows [9]:

①For each classification c_k in the classification set C, the posterior probability of test text d_e against c_k can be calculated according to the formula as follows:

$$P(d_e \mid c_k) = \sum_{w_i \in W} p(w_i \mid c_k) \quad (4)$$

In the training set D_t, providing the number of text of feature w_i with weight c_k is $|D_t(w_i, c_k)|$, and the text number of train set is $|\mathrm{D}_t|$, the formula (4) can be calculated by maximum likelihood estimation:

$$P(w_i \mid c_k) = \frac{|D_t(w_i, c_k)|}{|D_t|} \quad (5)$$

②For each classification c_k in the classification set C, using the calculation result of formula (4), the posterior probability of test text d_e can be calculated by the Bayes formula:

$$P(c_k \mid d_e) = \frac{P(d_e \mid c_k)P(c_k)}{P(d_e)} \quad (6)$$

Providing the number of text with classification c_k in the classification set D_t is $|D_t(c_k)|$, and the text number of training set is $|D_t|$, $P(c_k)$ in formula (6) can be calculated by the maximum likelihood estimation:

$$P(c_k) = \frac{|D_t(c_k)|}{|D_t|} \quad (7)$$

③Use the posterior probabilities calculated by formula (6) in step 2 to form a set $bP = \{bp_1,..., bp_n\}$. Providing the subscript of the maximum value in bP is $MaxIndex$, the classification of test text d_e is $c_{MaxIndex}$.

In formula (7), $P(d_e \mid c_k)$ and $P(c_k)$ are nonnegative, and $P(d_e)$ is a positive value, so for the sake of convenience, we can use the numerator of formula (8) as the statistical parameters in practical applications.

III. Spam Message Data Classification From a Mobile Operator

The naïve Bayes algorithm is adopted in this paper to address the classification of spam message data from a mobile operator. These data have been tagged and classified manually, which can be divided into two types: one includes the reported data, and the other the arbitral data. The reported data are being reported to the operator as spam messages by users, which have a basic form of "reported mobile phone number + message content + classification label". The classification tags contained in the reported data are SP decoy information, commercial advertisements, prostitution information, gambling information, propaganda information from the mobile company, mafia information, fraud information and reactionary information. The arbitral data refer to those judged as spam messages by the operator according to current spam message filtration technology, which has a basic form of "message content + classification tag". The classification tag includes customized SP decoy information, commercial advertisement information, prostitution information, political information, and crime information such as that regarding gambling, the mafia, and fraud. Their data forms are shown in Table I and II (the data comes from a mobile communication corporation).

TABLE I
THE REPORTED MESSAGE DATA FORM

Fieldname	Field attribute	Field description
Mobile phone number reported	String	Type of text character
Message content	String	Type of big text character
Classification tag	String	Spam message classification tag

TABLE II
THE ARBITRAL MESSAGE DATA FORM

Fieldname	Field attribute	Field description
Message content	String	Type of text character
Classification tag	String	Spam message classification tag

As these two types of data are both spam messages with similar corresponding classification tags, we can combine their tags for the convenience of later classification. The combination process is shown in Table III:

TABLE III
SPAM MESSAGE DATA INTRODUCTION

Combined tag	Reported data tag	Arbitral data tag
Political information	Reactionary information	Political information
Commercial information	Commercial advertisement, propaganda information from the mobile company	Commercial advertisement information
Prostitution information	Prostitution information	Prostitution information
Mafia information	SP decoy information, gambling information, mafia information	Customized SP decoy information, crime information such as gambling and mafia
Fraud information	Fraud information	Crime-fraud
Other spam messages	—	Other spam messages
Other information	Other information	—

Other information from the reported data can be used as normal messages for tests, while other spam messages from the reported data can be used as spam messages for tests. After combining those two types of spam messages, we will perform the process of duplicate removal. The duplicate removal process should be performed since the spam message often includes considerable repeated information sent to other users from many fraudulent users, which is useless for our classification model. The comparison of the quantity of message data after duplicate removal is shown in Table IV.

TABLE IV
INTRODUCTION OF THE DATA AFTER COMBINING THE SPAM MESSAGE CLASSIFICATION TAGS

Combined tag	Total	Quantity after duplicate removal	Percentage (%)
Political information	5382	784	14.57
Commercial information	3174844	96778	3.05
Prostitution information	132975	19807	14.90
Mafia information	1513822	109351	7.22
Fraud information	507890	28319	5.58
Other spam messages	34188	3576	10.46
Other information	169686	67216	39.61
Total	5538787	325831	5.88

The number of messages after duplicate removal is greatly reduced and is only 6% of the original data number. Regarding the number of messages, there is a large amount of information with the commercial and mafia tags but much less information with the political and other spam messages tags. Regarding the duplicate removal percentage, the information for the commercial, mafia and fraud information tags contains many repeated messages, and these tags have a larger percentage of removed duplicates, while the amount of repeated messages with the political, prostitution and other information tags is relatively less. It can be seen that the spam messages are mainly tagged as commercial and mafia information, which generally contain a great amount of repeated messages.

For the operation efficiency and accuracy of classification model, we should further carry out text preprocessing on the spam messages to obtain the ideal text data form for the model classification.

3.1 Text Preprocessing

Text preprocessing is a necessary and essential process. During preprocessing, we should not only clean the data but also extract and save some important features [10], such as mobile phone number, telephone number, URL, bank card number, WeChat, and QQ. These electronic addresses play an important role, screening the electronic address of the message when it is entered it into the classification model in the very beginning, identifying the suspicious message and then further improving the detection efficiency. Therefore, each stage of preprocessing should be carried out precisely. The following cleaning process is finally obtained after continuous adjustment through experiments (Fig. 1):

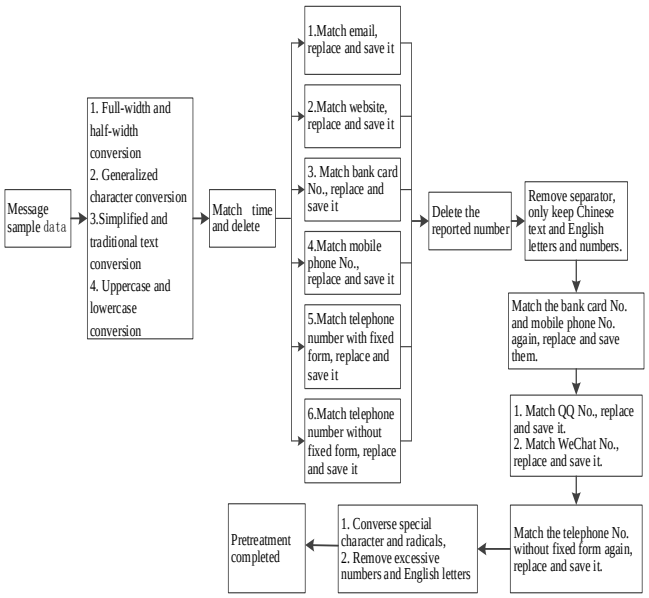

Fig. 1 The process structure of text preprocessing

3.2 Text Feature Extraction

Clean Chinese message texts suitable for analysis are obtained after the above text preprocessing. We then will extract the text feature. As the data are Chinese text instead of ordinary digital data, it is necessary to perform word segmentation first to remove the stopped words and finally deal with them in a computable form [11]. Jieba word segmentation, widely used in dealing with Chinese text and python programming, is applied to accomplish the word segmentation process. The feature extraction is achieved by adopting a TF-IDF algorithm, which is generally used for text, as well as python programming. The detailed text feature extraction process is shown in Fig. 2:

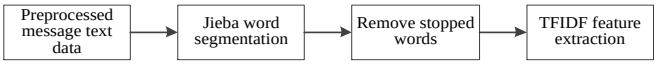

Fig. 2 The process structure of text feature extraction

1. Jieba word segmentation

Jieba word segmentation is an effective algorithm for word segmentation of Chinese text with high accuracy and fast speed, which is quite appropriate for text analysis [12]. Jieba word segmentation involves some algorithms as follows [13]:

①Achieve efficient word and figure scanning based on the Trie tree structure, and generate a directed acyclic graph (DAG) constituted by all possible word formations of Chinese characters in sentences;

②Adopt dynamic programming to find the maximum probability path and maximum segmentation combination based on word frequency;

③Adopt an HMM mode for unregistered words based on Chinese word formation and the Viterbi algorithm.

2. TF-IDF calculation steps [14]

①Calculate word frequency

Word frequency = Total times a certain word appears in the paper

②Calculate the inverse document frequency

Inverse document frequency (IDF) = log (total number of documents in the corpus/number of documents including a certain word + 1)

(The denominator is 0, plus 1 to the denominator)

③Calculate TF-IDF value

TF-IDF value= TF * IDF

(TF-IDF value is in proportion to the appearance frequency of a certain word and is in inverse proportion to the times of such word appearing in the whole corpus, which accords with the previous analysis.)

④Find out the key words

After calculating the TF-IDF values of each word in the paper, sort them and select several words with the highest value as the keywords.

3. The statistical analysis of features with different weights

Various types of feature words have various characteristics, representative and distinctive from other types, which satisfies the result of the TF-IDF algorithm and is suitable for text analysis [15]. However, because there is too much text content in each classification, the calculation weight values are generally small. Next, we will make a statistical analysis of features with different weights.

TABLE V
TOTAL NUMBER OF FEATURES EXTRACTED BY THE TF-IDF ALGORITHM WITH DIFFERENT WEIGHT

	0.01	0.02	0.03	0.05
Commercial	566	207	109	46
Other message	181	82	50	32
Other spam message	151	70	31	14
Mafia	288	107	45	19
Prostitution	357	181	104	43
Fraud	347	193	125	64
Political	315	130	76	43

Table V shows that if the threshold value of the weight is reduced by one percentage point, the total number of features will decline rapidly. When the weight is more than 0.05, the number of features of each classification is less than 100. When the weight is more than 0.01, the number of features of each classification is more than 100. Clearly, our classification problem is a multiple classification problem, and each classification has its own obvious feature. Therefore, we prefer to establish multiple two-classification models for such problems. Multiple two-classification models not only have obvious features but can also optimize the performance. One hundred dimensional features can be extracted for each classification, and the computation dimension can be simplified greatly, which is an optimization for our classification process.

3.3 Spam Message Classification Model

Text classification for spam messages can be performed after a series of text preprocessing and feature extraction steps.

3.3.1 Multi-classification Naïve Bayes Spam Message Classification Model

1. Model framework

The naïve Bayes algorithm has good performance regarding the problems of classification and high latitude. Focusing on the problem of the multiple classification of spam messages, a multi-classification naïve Bayes spam message model is presented in this paper, which is shown in Fig. 3:

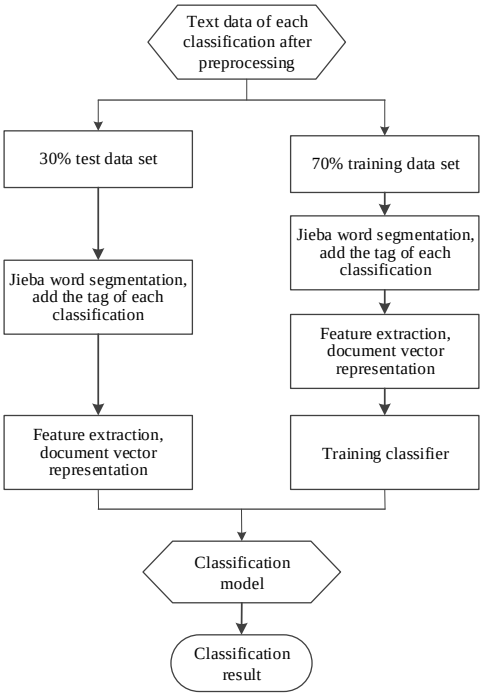

Fig. 3 Multi-classification naïve Bayes spam message classification model

2. Realization process

The multi-classification naïve Bayes model focuses on multi-classification problems. The data set has seven classification tags; 30% of the data set is divided into the test data set, and the other 70% is the training data set. Each message will be addressed with Jieba word segmentation and feature extraction technology and will finally be processed into a document vector form for model training and testing. Python programming technology is adopted to realize our model. The pseudocodes of the model are given as follows:

```
test_data = []      // Test data set
train_data = []     //  Training data set
      forі = 1:N
//  Number of spam message classification
  fobj = file.open(file(i))
// Read text preprocessed data of each classification
   while True:
      raw = fobj.readline()      // Read each message
      if raw:
         word_cut = jieba.cut(raw)
// Jieba word segmentation
         iftest_data.length>0.3*fobj.length:
//Judgment, test set 30%, Training set 70%

            train_data.append(word_cut,i)
//Add data for Training set, word segmentation
+classification

         else:
            test_data.append(word_cut,i)
//Add data for test set, word segmentation +classification

      else:
         break

word_features = get_features()
// Read TF-IDF feature value
test_data = document_features(test_data)
// Test set document vector process
train_data = document_features(train_data)
// Training set document vector process

classify = NaïveBayesClassifier.train(train_data)
// Carry out classification training for Training data set
classify.test(test_data)
// Carry out test inspection for classification model
```

3. Experiment result and analysis

After the seven classifications are combined together, the classification effects of the naïve Bayes algorithm with different TF-IDF weights and different feature dimensions is used, as shown in Table VI.

TABLE VI
EXPERIMENT RESULT OF MULTI-CLASSIFICATION NAÏVE BAYES ALGORITHM MODEL

Weight	Feature dimension	Accuracy (%)	Time consumed (s)
0.05	254	63.72	3.98
0.03	533	69.74	9.89
0.02	963	73.80	15.97
0.01	2198	78.78	29.16

In the multi-classification algorithm shown in Table 6, the proportional sampling method is used, and 35,290 data are sampled for algorithm evaluation. The "time consumed" is the training time for the algorithm model, and the accuracy refers to the precision of the calculations in the test set. From the above data in Table 6, the accuracy tends to increase as the weight decreases, i.e., the feature dimension increases. However, the time consumed doubles, and the computational complexity increases greatly. Among the results, the feature accuracy at 254 dimensions with a weight less than 0.05 is only 63%, which is far from the classification result we expect. Although the feature accuracy at 2198 dimensions with a weight less than 0.01 increases to 78%, the time consumed is nearly half a minute. The reason for the lower accuracy is that features of more than 2000 dimensions cannot separate into these seven classifications. The more types of classification there are, the more interference and noise each classification will receive. Therefore, the efficiency of the multi-classification naïve Bayes model with the seven-classification combination is far from satisfactory.

3.3.2 Multi-two-classification Naïve Bayes Spam Message Classification Model

1. Model framework

From the research on text feature extractions, we know that relatively obvious features can be obtained by calculating TF-IDF feature extraction. Various types of spam messages have various feature words representing their characteristics. Moreover, different weight values will lead to different total numbers of features. Obviously, the problem of spam message classification is a multi-classification problem involving seven types of data. Therefore, it is necessary to consider these seven types of feature words together as a classification feature. As a result, the feature dimensions will become dramatically large, which leads to slow computational operation and low classification accuracy. According to the experiment results of the multi-classification naïve Bayes model in the above section, the model can be improved by dividing the multi-classification problem into a multi-two-classification problem, which not only reduces the computational complexity but also improves the classification accuracy.

A multi-two-classification naïve Bayes spam message classification model is constructed in this paper, which requires seven naïve Bayes two-classification models. The detailed process of each model is shown in Fig. 4:

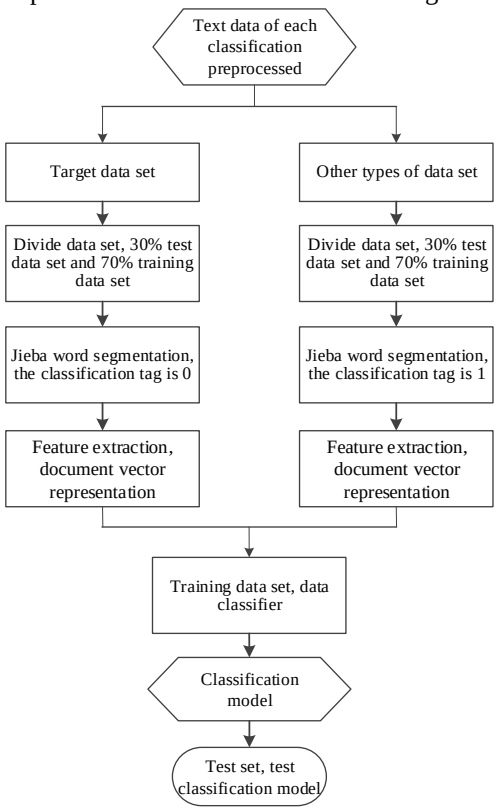

Fig. 4 Naïve Bayes two-classification model

2. Realization process

To construct a multi-two-classification naïve Bayes classification model, seven naïve Bayes two-classification models for the seven types of data should be built first. The classification tag for one type of data is 0, and the classification tag for other types of data is 1. The feature value of the type of data with tag 0 is used for feature extraction, which helps reduce the feature dimension of each classification model and improve the classification of this model. Each naïve Bayes model should go through processes such as data set division, Jieba word segmentation, feature extraction and document vector representation, and the models should finally undergo model training and testing.

The following pseudocodes indicate the realization process of the naïve Bayes two-classification model.

```
test_data = []      // Test data set
train_data = []     // Training data set
      fori = 1:N
      //  Number of spam message classification
   fobj = file.open(file(i))
// Read data of each classification preprocessed
   if fobj.name = 'xxx'
// If it is the target class, the classification tag is 0
      while True:
         raw = fobj.readline()      // Read each message
         if raw:
            word_cut = jieba.cut(raw)
// Jieba word segmentation
            iftest_data.length>0.3*fobj.length:
//Judge, test set 30%, Training set 70%

               train_data.append(word_cut,0)
// Add data to Train set, word segmentation + type

            else:
               test_data.append(word_cut,0)
// Add data to test set, word segmentation + type

         else:
            break
   else:
// Other types of data, the classification tag is 1
      while True:
         raw = fobj.readline()      // Read each message
         if raw:
            word_cut = jieba.cut(raw)
// Jieba word segmentation
            iftest_data.length>0.3*fobj.length:
//Judge, test set 30%, Training set 70%

               train_data.append(word_cut,1)
// Add data to Train set, word segmentation + type

            else:
               test_data.append(word_cut,1)
// Add data to test set, word segmentation + type

         else:
            break
word_features = get_features('xxx')
// Read a certain type of TF-IDF feature value
test_data = document_features(test_data)
// Test set document vector treatment
train_data = document_features(train_data)
// Training set document vector treatment
classify = NaïveBayesClassifier.train(train_data)
// Classify the training data set
classify.test(test_data)    // Test the classification model
```

3. Experiment results and analysis

The experiment results of the multi-two-classification naïve Bayes model are shown in Table VII:

TABLE VII
EXPERIMENTAL RESULTS OF THE MULTI-TWO-CLASSIFICATION NAÏVE BAYES CLASSIFICATION MODELS

Classification model	TF-IDF weight	Feature dimension	Accuracy (%)	Time consumed (s)
Commercial model	0.05	46	76.57	0.86
	0.03	109	80.43	2.07
	0.02	207	83.68	3.94
	0.01	566	88.25	12.61
Other message model	0.05	32	86.04	0.61
	0.03	50	85.82	0.93
	0.02	82	86.32	1.59
	0.01	181	87.09	3.44
Other spam message model	0.05	14	98.10	0.27
	0.03	31	98.23	0.60
	0.02	70	97.86	1.34
	0.01	151	97.87	2.96
Mafia model	0.05	19	73.72	0.36
	0.03	45	76.79	0.88
	0.02	107	80.36	2.95
	0.01	288	84.63	5.56
Prostitution model	0.05	43	97.24	0.84
	0.03	104	97.58	1.98
	0.02	181	97.73	3.52
	0.01	357	97.88	6.99
Fraud model	0.05	64	92.79	1.22
	0.03	125	93.86	2.44
	0.02	193	93.85	3.72
	0.01	347	94.44	7.01
Political model	0.05	43	99.94	0.86
	0.03	76	99.94	1.50
	0.02	130	99.93	2.50
	0.01	315	99.94	6.24

The above multi-class and proportional sampled 35,290 data points is adopted for the multi-two-classification naïve Bayes experiment. The experiment results show that the classification accuracy tends to increase as the weight decreases, i.e., the feature dimension increases. Although the rising trend is not significant, the accuracy obtained is quite satisfactory, and the time consumed by the algorithm training model is also close to our ideals. Among these seven classifications, commercial, other message and mafia classification have lower accuracy, which is lower than 90% but generally higher than 80%. The other four classifications, i.e., the prostitution, fraud, political and other spam message tags, have higher accuracy, which is more than 90%. The accuracy of political classification is nearly 100%, which has excellent efficiency for classification. Thus, it can be seen that the classification performance of multi-two-classification is far higher than that of multi-classification. Regarding the time consumption of the algorithm, if the feature dimension is low, the algorithm is the fastest when the weight is 0.05, but the overall accuracy rate is low. If the feature dimension is higher, the accuracy rate

will become higher when the weight is 0.01, but the time consumed will increase by almost 3 s or even more than 5 s. On the whole, when the weight is below 0.02, the comprehensive classification performance is the best. Meanwhile, the feature dimension of each classification at that weight remains at approximately 100, and the time consumed is approximately 2 s. Therefore, the multi-two-classification naïve Bayes algorithm has the best efficiency for these seven classifications of messages when the feature extraction TF-IDF algorithm uses a threshold of 0.02.

4. Classification scheme

The above is the experimental research of each classification model in the two-classification model. Seven classification results will be obtained for every unknown message. In most cases, a unique classification tag will be obtained through statistical analysis of the conditional probability of each classification model. However, when the conditional probability reaches a maximum, two or more classification tags will appear, and the message then will be classified manually. The following diagram is the final process of the classification model (Fig. 5).

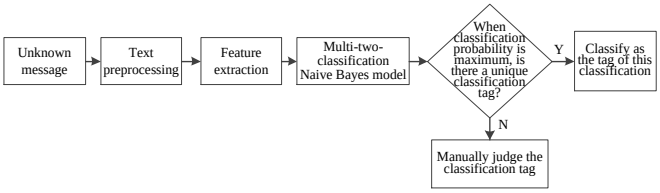

Fig. 5. The classifying solution of the multi-two-classification model

3.4 Performance Comparison of Different Classification Algorithms with Multi-two-classifications

In this section, the naïve Bayes algorithm is compared with two other classification algorithms that are used often; one is the support vector machine, and the other is the random forest algorithm.

① The support vector machine: The basic concept of the support vector machine is to achieve a compromise between the accuracy (for a given training set) and machine capacity (the capability of machine to learn any training set without mistake) for a specified learning task with limited training samples to thus obtain the best promotion performance [16].

② The random forest: Considering the decision tree as the basic classifier, the random forest algorithm adopts a sampling method without a replacement used for the bagging algorithm for training set sampling and only extracts a part of the features for training using the referring random subspace method. Finally, the classification result will be determined by the vote of the trained decision tree [9].

The classification efficiency comparison between the naïve Bayes algorithm, the support vector machine, and the random forest algorithm is shown in Table 8. The proportional sampling of 35,290 data points is adopted by taking the feature value of TF-IDF 0.02 as the feature dimension.

TABLE VIII
EXPERIMENTAL RESULTS OF DIFFERENT CLASSIFICATION ALGORITHMS WITH MULTI-TWO-CLASSIFICATION

	Naïve Bayes		Support vector machine		Random forest (10)	
Classification	Accuracy (%)	Time consumed(s)	Accuracy (%)	Time consumed (s)	Accuracy (%)	Time consumed (s)
Commercial	83.68	3.94	81.5	19.69	77.97	7.57
Other message	86.32	1.59	86.76	10.07	82.26	2.86
Other spam message	97.86	1.34	99.16	2.32	99.18	1.89
Mafia	80.36	2.95	79.30	15.39	74.63	3.44
Prostitution	97.73	3.52	97.23	6.25	97.58	4.86
Fraud	93.85	3.72	93.36	9.97	93.45	5.75
Political	99.93	2.50	99.86	2.95	99.94	3.02

The experiment results in Table VIII show that the accuracies of the naïve Bayes and support vector machine algorithms are higher and more similar to each other than the accuracies of the random forest algorithm, which are slightly lower for some classifications. In general, the naïve Bayes classification has high and stable accuracy values. For the time consumed of the three algorithm models, it is clear that the support vector machine and random forest are not as fast as the naïve Bayes model, and their performance is unstable. On the whole, after comparing different algorithms, the naïve Bayes model has the best classification performance, which is not only feasible but also achieves the ideal result.

IV. CONCLUSIONS

This paper puts forward a spam message classification model based on the naïve Bayes algorithm and estimates the performance of the naïve Bayes algorithm model based on multi-classification and multi-two-classification in spam message classification using spam message text preprocessing based on Java regular expression and spam message feature extraction based on Jieba word segmentation and the TF-IDF algorithm. This paper further compares the classification performance of the naïve Bayes, support vector machine, and random forest algorithms with multi-two-classification. The experiment results show that the multi-two-classification naïve Bayes algorithm has the best efficiency of the three models.

However, the data used in the process of spam message classification are only a part of the sampled data. With the development of big data technology, how to perform feature extraction and text classification for bulk data on the basis of big data computation will be our future research direction.

REFERENCES

[1] C. Huang, "Research on SMS filtering technology on intelligent mobile phone," M.S. thesis, Huazhong University of Science and Technology, Wuhan, China, 2012.

[2] J. Ma, Y. Zhang and Z. Wang, "A message topic model for multi-grain SMS spam filtering," *International Journal of Technology and Human Interaction (IJTHI)*, vol. 12, no. 2, pp. 83-95, 2016.

[3] S. J. Delany, M. Buckley and D. Greene, "Review: SMS spam filtering: Methods and data," *Expert Systems with Applications*, vol. 39, no. 10, pp. 9899-9908, 2012.

[4] J. Fdez-Glez, D. Ruano-Ordas and J. R. Méndez, "A dynamic model for integrating simple web spam classification techniques," *Expert Systems with Applications*, vol. 42, no. 21, pp. 7969-7978, 2015.
[5] A. Harisinghaney, A. Dixit and S. Gupta, "Text and image based spam email classification using KNN, Naïve Bayes and Reverse DBSCAN algorithm," in *Optimization, Reliabilty, and Information Technology (ICROIT), 2014 International Conference on. IEEE,* 2014, pp. 153-155.
[6] A. Gelman, J. B. Carlin and H. S. Stern, *Bayesian data analysis*. Boca Raton, FL: CRC press, 2014.
[7] S. Ajaz, M. Nafis and V. Sharma, "Spam Mail Detection Using Hybrid Secure Hash Based Naïve Classifier," *International Journal of Advanced Research in Computer Science*, vol. 8, no. 5, 2017.
[8] G. Feng, B. An and F. Yang, "Relevance popularity: A term event model based feature selection scheme for text classification," *Plos One*, vol. 12, no. 4, pp. e0174341, 2017.
[9] D. M. Diab, K. M. El Hindi, "Using differential evolution for fine tuning naïve Bayesian classifiers and its application for text classification," *Applied Soft Computing*, vol. 54, pp. 183-199, 2017.
[10] P. Chandrasekar, K. Qian, "The Impact of Data Preprocessing on the Performance of a Naïve Bayes Classifier," in *Computer Software and Applications Conference (COMPSAC), 2016 IEEE 40th Annual. IEEE,* 2016, pp. 618-619.
[11] X. Y. Wang, "A study on the Chinese text summarization method based on concept lattice," M.S. thesis, Beijing Institute of Technology, Beijing, China, 2015.
[12] L. Liu, Y. Lu and Y, Luo, "Detecting "Smart" Spammers On Social Network: A Topic Model Approach," *arXiv preprint arXiv*, pp.1604.08504, 2016.
[13] D. Ye, P. Huang and K, Hong, "Chinese Microblogs Sentiment Classification using Maximum Entropy," *Proceedings of the Eighth SIGHAN Workshop on Chinese Language Processing*, pp. 171-179, 2015.
[14] M. H. Arif, J. Li and Iqbal. M, "Sentiment analysis and spam detection in short informal text using learning classifier systems;" *Soft Computing*, pp. 1-11, 2017.
[15] J. K. Yi, L. K. Tian. A text feature selection algorithm based on class discrimination;" *Journal of Beijing University of Chemical Technology (Natural Science)*, vol. 40, pp. 72-75, 2013.
[16] M. Diale, W. C. Van Der and T. Celik, "Feature selection and support vector machine hyper-parameter optimisation for spam detection;" in *Pattern Recognition Association of South Africa and Robotics and Mechatronics International Conference (PRASA-RobMech) 2016. IEEE,* 2016, pp. 1-7, 2016.