ISSN 1330-3651 (Print), ISSN 1848-6339 (Online) https://doi.org/10.17559/TV-20190420161815
Original scientific paper

# Naive Bayesian Automatic Classification of Railway Service Complaint Text Based on Eigenvalue Extraction

Lifeng LI, Wenxing LI, Daqing GONG

**Abstract**: Railways have developed rapidly in China for several decades. The hardware of railways has already reached the world's leading level, but the level of service of these railways still has room for improvement. The railway management department receives a large number of passenger complaints every year and records them in text, which needs to be classified and analyzed. The text of railway complaints includes characteristics spanning wide business coverage, various events, serious colloquialisms, interference and useless information. When using the direct classification via traditional text categorization, the classification accuracy is low. The key to the automatic classification of such text lies in an eigenvalue extraction. The more accurate the eigenvalue extraction, the higher the accuracy of text classification. In this paper, the TF-IDF algorithm, TextRank algorithm and Word2vec algorithm are selected to extract text eigenvalues, and a railway complaint text classification method is constructed with a naive Bayesian classifier. The three types of eigenvalue extraction algorithms are compared. The TF-IDF algorithm, based on eigenvalue extraction, achieves the highest automatic text classification accuracy.

**Keywords:** automatic classification; eigenvalue; naive Bayes; railway complaint text; TextRank; TF-IDF; Word2vec

## 1 INTRODUCTION

The increase of railway construction mileage and the rapid increase of train speed not only brings convenience to people's travel but also greatly contributes to China's national economy and regional economic growth. The railway industry is not only composed of vast infrastructure and equipment but also includes a wide range of services and a large number of customers. The quality of railway service has also attracted increasing attention from the masses and media. The number of complaints has also increased with the increased mileage and number of passengers. The railway management department began to pay attention to customers' complaints and satisfaction with railway services very early on and took improving passenger service satisfaction as an important indicator of the development of the railway industry. The management department has opened a variety of service complaint channels for passengers, including telephone complaints, internet complaints and mail complaints and receives a large number of complaints from passengers every year, recording them in text form. How to classify the existing complaint data automatically has become an urgent problem.

Automatic text classification mainly includes two key technologies: text feature extraction and classifier design. In terms of text feature extraction, a vector space model is often used to express and describe text. The features of each dimension of the vector space model can be directly obtained by word segmentation or word frequency statistical algorithms, and the features of discrimination ability, such as information gain, word frequency, mutual information, TFIDF matrix, etc., can be further extracted on the basis of words. Because the dimension of the vector space model obtained by directly using words as features is too large to be solved for, the vector space model is usually constructed by further extracting features on the basis of words. Among them, the TFIDF feature is most widely used. In terms of classifier design, with the rapid development of machine learning technology, there are many classifiers available. However, to study the characteristics of text classification, the commonly used classifiers include K-Nearest Neighbor, Support Vector Machine (SVM) and Naive Bayes. K-Nearest Neighbor and Support Vector Machine classification methods have better results when dealing with small samples, while the Naive Bayes method has a stronger adaptability for text classification and is suitable for different size samples. Therefore, the Naive Bayes method is widely used in the field of text classification.

## 2 LITERATURE REVIEW

Many studies abroad have applied text mining technology and eigenvalue extraction algorithms to service evaluation or complaint analysis. Chen (2009) conducted cluster analysis on the forms and relationships of keywords of online shopping complaints and verified the effectiveness of text mining [1]. Lee (2011) evaluated the service quality of hotels by mining the contents of hotel messages [2]. Ghazizadeh (2014) used Latent Semantic Analysis (LSA) to mine the text in the US National Highway Safety Accident Complaint Database to extract the main causes and development trends of vehicle failures [3]. Pan Gang (2013) thinks that complaint management is an important part of service quality management. Traditional analysis methods have difficulty analyzing unstructured data and heterogeneous text sets of complaint texts. New technical methods need to be applied, and an improved fuzzy classification algorithm based on statistics is proposed, which improves on the precision of the previous algorithms [4]. Tang Shengtao (2016) analyzed the unstructured data of mobile operator customers' complaints through text mining to provide optimized solutions and improve customers' perception and experience [5]. Xia Haifeng (2013) thinks that customer complaints in the telecommunication industry are special, and strong complaints need to be classified into correct complaint navigation channels in a very short period of time. A complaint text mining model using TF-IDF calculation is proposed, and experiments verify that the model can effectively classify complaint text [6]. Shi Zhifang (2013) proposed an optimized K-means algorithm and realized the mining and analysis of hot topics with

different subjects based on a combination of online analysis processing and association rules in the data warehouse [7]. Liu Xin (2017) obtains passengers' evaluations of railway passenger transport service through social networks and processes the evaluation text by using Chinese word segmentation technology and stop words filtering technology to construct an association of "product characteristics-evaluation viewpoint" to simplify the text and finally realize the evaluation of railway passenger transport service quality [8]. Onan et al. (2016) examined the predictive performance of five statistical keyword extraction methods (most frequent measure-based keyword extraction, term frequency-inverse sentence frequency-based keyword extraction, co occurrence statistical information-based keyword extraction, eccentricity-based keyword extraction and TextRank algorithm) on classification algorithms. Experimental results show that the frequency-based keyword extraction algorithm has the best effect [9]. Sowmya's (2016) main work is to extract the keywords from a conversation using particle swarm optimization [10]. Chen et al. (2019) proposed a new graph-based measure for keyword extraction by leveraging higher-order structural features (e.g., motifs) of a word co occurrence graph [11]. Rossi et al. (2014) thought that keyword extraction methods can be broadly divided into two categories: domain-dependent and domain-independent keyword extraction methods [12]. Mihalcea and Tarau (2004) report on seminal research that introduced a state-of-the-art TextRank model [13]. Mikolov et al. (2013) proposed the word 2 vec model, and some researchers used it to extract keywords [14]. Abilhoa et al. (2014) proposed a keyword extraction method for tweet collections that represents texts as graphs and applies centrality measures to find the relevant vertices [15]. Siddiqiet al. (2015) [16] and Beliga et al. (2015) summarized existing keyword extraction algorithms separately. Jiang et al. (2016) [17] applied the proposed deep feature weighting to some state-of-the-art naive Bayes text classifiers and achieved remarkable improvements. Xie et al. (2017) [18] applied NLP on traditional Chinese medicine prescription to predict the prescriptions and diseases for the treatment of the disease with the trained model.

The existing text classification research is mainly applied to several types of classical datasets. There are few studies on complaint text, especially Chinese complaint text, and the classification accuracy is still low, which is closely related to the characteristics of Chinese text. This paper establishes a complaint text classifier with the goals of extracting the characteristics of the complaint text and improving the accuracy of text classification.

**Table 1** Analysis of railway service characteristics

| The First Level Features | The Second Level Features | The Third Level Features | Feature Expression |
|---|---|---|---|
| Station Services | Ticket Services | Ticket Information | Ticket, Purchase Tickets, Ticket Window, Ticket Price, Refund, etc. |
| | | Purchase Automation | |
| | | Ticket Window Service | |
| | | Ticket Price | |
| | Inbound and Outbound Services | Inbound and Outbound Guidance Information | In and Out Station, Signage and Placard, Ticket Check, etc. |
| | | Inbound and Outbound Ticket Inspection Speed | |
| | Information Inquiry | Arrival Time Information | Departure Time, Arrival time, Stop time, etc. |
| | | Train Stop Information | |
| | Waiting Services | Water Supply | Waiting Room, Drinking Water, Seat, Waiting Environment, etc. |
| | | Seat Comfort | |
| | | Waiting Environment | |
| Carriage Service | In-carriage Information Service | Running Status | Speed, Reminder to the Station, Cabin Temperature, etc. |
| | | Time | |
| | | Arrival Status | |
| | | Inside and Outside Temperature | |
| | Hardware Facility Service | Large Baggage Placement | Luggage Rack, Seat, etc. |
| | | Seat Comfort | |
| | Attendant Service | Attendant Service Attitude and Courtesy | Attendant, Attitude, Dress, Response, Emergency, etc. |
| | | Attendant Instrument | |
| | | Service Response Speed | |
| | | Emergency Response Capability | |

## 3 ANALYSIS OF RAILWAY SERVICE CHARACTERISTICS

By analyzing the passengers' high-speed railway passenger service process and combining the quality characteristics and advantages of high-speed railway passenger transportation services, this paper classifies the key characteristics of high-speed railway passenger transportation service quality as Tab. 1.

Complaints usually involve multiple subjects, and the characteristics of the complainant's association with emotional vocabulary directly indicate the propensity of the complaint. In practical applications, these factors are also an important basis for the analysis of complaint texts. This paper studies the forms of emotional vocabulary and its role in the text, and from the influence of product features and emotional words to the association of part of speech information, distance information, and structural information. Based on this and similar information, the automatic identification method of the association between the complaint feature and the emotional vocabulary is explored.

For topic classification, all words other than the stop words that describe the ability and the discriminating ability are generally used as candidate features. For text topic classification, words with emotional implications should be selected as candidate features. Chinese text classification mostly uses nouns (n), verbs (v) and adjectives (a) as candidate features of text. Some stop words are listed in Tab. 2.

Analysis of complaint content characteristics:

(1) Noun

The main body in a complaint report is mainly the nouns, such as the person, place, equipment and environment in which the incident occurred.

(2) Emotional words

In the complaint text, users use emotional words to express opinions and attitudes.

(3) Verbs

In the text of the complaint, the verb often reflects the specific behavior of the railway service work.

**Table 2** Stop Word List

| Noun | Word Frequency | Adjective | Word Frequency | Verb | Word Frequency |
|---|---|---|---|---|---|
| Ride | 72550 | Bad | 1541 | Complain | 61922 |
| Passenger | 54251 | Good | 1372 | Transform | 24059 |
| Service Quality | 46610 | Serious | 1270 | Said | 17558 |
| Train Times | 24743 | Big | 1144 | Wish | 17272 |
| Name | 24447 | Terrible | 1114 | Reply | 16816 |
| Order Number | 24249 | Convenient | 1092 | Arrive | 16229 |
| Network | 23910 | Successful | 1085 | Stand | 12278 |
| Responsibility | 23697 | High | 891 | Refund | 9277 |
| Unit | 23519 | Fast | 774 | Have | 8831 |
| Type | 23273 | Cold | 1541 | Security Check | 8046 |

## 4 COMPLAINT TEXT CLASSIFICATION PROCESS

The text classification of railway complaints is mainly divided into two processes: the training process and testing process. The main goal of the training process is to obtain text classifiers based on the existing training data. The four steps of classification could be seen in Fig. 1.

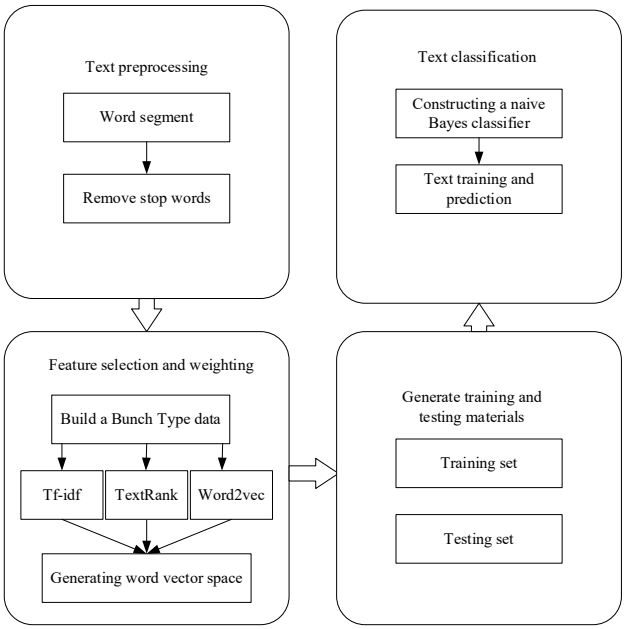


**Figure 1** Complaint text automatic classification process

The main steps include: (1) preprocessing the complaint text, removing punctuation marks and spaces in the text, automatically segmenting the complaint text, and removing stop words; (2) Extracting the eigenvalue of the texts through frequency analysis and part-of-speech analysis of the complaint topic texts; (3) Using Bunch type to describe the complaint text; (4) Assigning values to keywords with TF-IDF values to construct keyword weight matrix; and (5) Using the Naive Bayes algorithm to classify complaint texts. The data from the test process are processed in the same way and filtered according to the characteristics of the training process. Finally, the trained classifier is used to classify the text.

### 4.1 Text Preprocessing

The text uses a jieba word segmentation system for segmentation, and a stop word list is introduced to remove numbers, punctuation marks, stop words and other marks and words that affect the classifier. Due to the long length of the complaint text, there are still too many interfering words after using the stop word list. Therefore, according to the characteristics of the railway text, the keywords of the extracted text are used to reduce noise. Further analysis shows that the core meaning of the complaint text is mainly expressed by verbs, nouns and adjectives. These keywords are closely related to the railway business.

#### 4.1.1 Segmentation

The jieba word segmentation algorithm uses a prefix-based dictionary to achieve efficient word graph scanning, generating a directed acyclic graph (DAG) composed of all possible word-generating words in a sentence, and then uses dynamic programming to find the maximum probability path and the word-based frequency. The maximum segmentation combination for the unregistered words uses the HMM model based on the ability of Chinese characters to form words and uses the Viterbi algorithm.

#### 4.1.2 Stop Words Principle

(1) Words that are too frequent or too small should be set as stop words in all corpora; otherwise, they will affect feature extraction. *TF* (Term Frequency) indicates how often a keyword appears in the entire article. If $TF > 0.8$ and $TF < 0.1$, the word will be set in the list.

**Table 3** Stop Word List

| Number | Stop Words | Frequency |
|---|---|---|
| 1 | Train | 0.93 |
| 2 | Staff | 0.85 |
| 3 | Phone number | 0.82 |
| 4 | Responsibility | 0.76 |
| 5 | Passenger | 0.75 |
| ..... | ..... | ..... |
| *n* | Book | 0.01 |

(2) The phrase that cannot be correctly segmented by the existing word segmentation tool, which has an influence on feature extraction.

The pattern of the stop words list can be seen in Tab. 3.

### 4.2 Eigenvalue Extraction of Complaint Text

To express the evaluation text of the training set as feature vectors, the text needs to be processed by word segmentation to remove the stop words, but the feature space is still as large as in the tens of thousands of dimensions. To train and test the classifier directly on such a high-dimensional vector requires too much computation. Therefore, on the premise of not affecting the classification accuracy, it is necessary to reduce the dimension of the original feature space and compress the feature dimension to suit the number of training texts. In this paper, the Tf-idf algorithm, Word2vec algorithm and TextRank algorithm are used to extract the eigenvalues of the text, and the following complaint text examples are used for keyword extraction experiments.

*"Complaint, 9042 Zheng Lei, Ms. Wang went to Qinhuangdao Ticket Office's bank card self-service ticketing area on January 1 to buy tickets, but the bank card was swallowed by the machine. Now she has been waiting for an hour and no staff has been in charge. No one answered the phone number on the machine: 0335 192222. She asked the staff of the ticket office to find the information desk. No one was at the information desk, so she went to the security guard. The security guard told her to find the verification window. The window said that they were not in charge of this. Now Ms. Wang has called and complained that there was no one in charge. She promised to help report and stay in contact. Contact information: 13292528538."*

#### 4.2.1 TF-IDF Algorithm Eigenvalue Extraction

Tf-IDF (Term Frequency-Inverse Document Frequency) is a weighting technique commonly used in information processing and data mining. This technique uses a statistical method to calculate the importance of a word in the whole body of the text according to where the word appears in the text and the frequency of the word. The advantage of this method is that it can filter out some common but unimportant words while retaining important words that affect the whole text. The calculation method is shown in Eq. (1).

$$fidf_{i,j} = tf_{i,j} \times idf_i \tag{1}$$

In the formula, *tfidf* represents the product of word frequency $tf_{i,j}$ and inverted text word frequency $idf_i$. The greater the TF-IDF value, the more important the feature word is to the text.

IDF (Inverse Document Frequency) indicates the calculation of the inverted text frequency. Text frequency refers to the number of times a certain keyword appears in all articles in the whole body of text. Inverted document frequency is the reciprocal of document frequency and is mainly used to reduce the effect of some common words found in all documents but which have little influence on the documents. Eq. (2) is the calculation formula for TF word frequency.

$$tf_{i,f} = \frac{n_{i,j}}{\sum_k n_{k,j}} \tag{2}$$

In the formula, $n_{i,j}$ is the number of occurrences of feature word $t_i$ in text $d_j$, which is the number of all feature words in text $d_j$. The calculation result is the word frequency of a characteristic word.

Eq. (3) is the calculation formula of IDF.

$$idf_i = \log \frac{|D|}{\{j : t_i \in d_j\}} \tag{3}$$

In the formula, |D| represents the total number of texts considered and $j$ represents the number of feature words $t_i$ contains in these texts.

The extraction results by TF-IDF are listed in Table 4.

**Table 4** Eigenvalues Extracted by TF-IDF Algorithm

| Category | Key Words |
|---|---|
| Ticket Services | Ticket office, information desk, machine, ticket area, ticket purchase, ticket sales, self-service, real name, answer, verification |

#### 4.2.2 Word2vec Algorithm Eigenvalue Extraction

Google used the software tool Word2vec to train word vectors in 2013; this tool can express words as vectors.

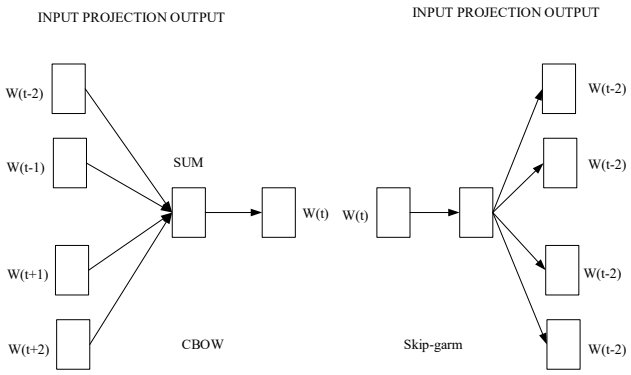


**Figure 2** Outline of the structure of the Word2vec algorithm

The Word2vec model includes two types of word vector learning structure models, which could be seen from Fig. 2: Skip-Gram and CBOW (Continuous Bag of Words Model). Both structures include an input layer, a mapping layer and an output layer. When the number of context words of $w$ is $n$, the Skip-Gram model predicts the context of the current word. The CBOW model predicts the current word by using the last words.

The Skip-Gram model predicts context based on current word. Given the word sequence $W = \{w_1, w_2, \ldots, w_m\}$, the model maximizes the average logarithmic probability as Eq. (4):

$$l(W) = \frac{1}{M} \sum_{m=1}^{M} \sum_{-L \le i \le L} \log p\left(\frac{w_{m+i}}{w_m}\right) \tag{4}$$

In the formula, $L$ is the size of the context window.

The CBOW model predicts the target word by specifying the window word. Given the word sequence $W =$

$\{w_1, w_2,\ldots, w_m\}$, the model maximizes the average logarithmic probability as Eq. (5):

$$l(W)=\frac{1}{M}\sum_{i=L}^{M-L}\log p\left(\frac{w_i}{w_{i-L},\ldots,w_{i+L}}\right) \tag{5}$$

In the formula, $L$ is the size of the context window.

The extraction results by Word2vec are listed in Tab. 5.

**Table 5** Eigenvalues extracted by Word2vec Algorithm

| Category | Key Words |
|---|---|
| Ticket Services | Telephone, ticket sales, contact, self-service, security, Ms. Wang, answer, ticket purchase, ticket office |

### 4.2.3 TextRank Algorithm Eigenvalue Extraction

The TextRank algorithm is a graphic algorithm similar to PageRank. TextRank analogizes words in documents to Internet web pages, such that the links between words are analogous to the links between web pages. In other words, the algorithm thinks that the text is a network composed of words or a graph composed of words that are nodes, and the semantic relations between words form edges. The more important the words in the picture, the more likely they are keywords.

Formally, we let $G(V, E)$ represent a directed graph consisting of words in the text, where $V$ is the word nodes and $E$ is the edges. For each node$V_i$, $In(V_i)$ represents the set of nodes pointing to that node and $Out(V_i)$ represents the set of nodes $V_i$ pointing to that node. $W_{ij}$ represents the weight between $V_i$ and $V_j$. We determine a sliding text window that contains $k$ words. If two words appear in this window at the same time, we can call this a cooccurrence. The number of cooccurrences of word pairs can be used as the weights of the edges connecting them. The score calculation formula for node $V_i$ is Eq. (6):

$$S(V_i)=(1-d)+d\cdot\sum_{V_j\in In(V_j)}\frac{W_{ji}}{\sum_{V_k\in Out(V_j)}W_{jk}}S(V_j) \tag{6}$$

As seen from the formula, the calculation of a TextRank weight is an iterative process, where $d$ is generally set to 0.85. The flow of the algorithm is as follows:

(1) Determine the best representation form of the text: expression or single words or other, which will be used as the nodes in the diagram;

(2) Construct edges between nodes, such as using cooccurrence information as weights;

(3) Iterate until the algorithm converges;

(4) Sort nodes according to scores to obtain keywords.

The extraction results by TextRank is listed in Tab. 6.

**Table 6** Eigenvalue extracted by TextRank Algorithm

| Category | Key Words |
|---|---|
| Ticket Services | Machine, help, self-service, ticket purchase, verification, ticket sales, ticket area, real name, information desk, telephone |

## 4.3 Naive Bayesian Classifier Construction

### 4.3.1 Bunch Object

After preprocessing and feature extraction, the complaint text finally forms the training and test sets. To facilitate the classifier in processing the dataset, we change the structure of the body of text via the Bunch mode. A Bunch mode is a very useful and flexible dataset type that allows us to set any property in its constructor.

"Target Name": Category, which is a collection of all classification categories.

"Filename": Text filename

"Label": Text labels

"Content": Characteristic words extracted from the complaint text

The Bunch mode types are shown in Tab. 7.

**Table 7** Bunch Object

| Target Name | Filename | Label | Contents |
|---|---|---|---|
| Cleaning Service | Cleaning_S21 | Cleaning Service | Content_21 |
| | Cleaning_S23 | Cleaning Service | Content_23 |
| Ticket Service | Ticket_S32 | Ticket Service | Content_32 |
| | Ticket_S34 | Ticket Service | Content_34 |
| Train Service | Train_S45 | Train Service | Content_45 |
| | Train_S48 | Train Service | Content_48 |

### 4.3.2 Distributed Representation

In the previous section, we performed word segmentation on the original dataset and implemented the variable representation of the dataset by binding it to the Bunch data type. However, the problem of natural language understanding is that this problem needs to be translated into machine learning problems. The first step is to find a way to mathematicalize these symbols. The most intuitive and most commonly used word representation in NLP is One-hot Representation, which represents each word as a very long vector. The dimension of this vector is the vocabulary size, where most of the elements are 0, and only one dimension has a value of 1, which represents the current word. The biggest problem with this representation method is the "word gap" phenomenon: any two words are isolated. One does not see whether the two words are related from these two vectors.

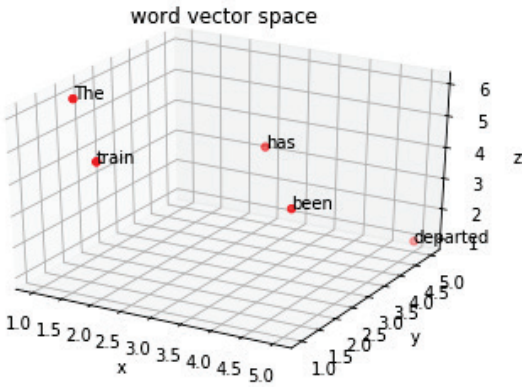


**Figure 3** Word vector space

The distributed representation was first proposed by Hinton in the 1986 paper "Learning distributed representations of concepts" [18]. The biggest contribution of the distributed representation is to make related or similar words closer in distance. The distance of the vector can be

measured by the most traditional Euclidean distance, or by the angle of cos.

For example, the sentence of *"The train has been departed"* is converted to the vector model as Fig. 3:

#### 4.3.3 Naive Bayesian Classifier

The Bayesian network is a classifier learned from a series of sample instances with category marks. For a given instance, $X$ is represented by a feature vector ($a_1$, $a_2$,…, $a_n$). The Bayesian network uses Eq. (7) to further classify this case:

$$c(x) = \arg\max P(c)P(a_1, a_2, ..., a_m|\mathrm{C}) \tag{7}$$

In the formula, $C$ is the set of possible categories $c$, $c(x)$ is the category of Bayesian network classifier prediction $x$. For known categories, it is assumed that all attribute conditions are independent from each other, and the Bayesian network classifier based on the attribute condition independence assumption is called the Naive Bayesian classifier. The Naive Bayesian algorithm is the simplest form of the Bayesian network classifier and is one of the most widely used algorithms in the classifier field. The expression of the Naive Bayesian classifier is Eq. (8):

$$c(x) = \arg\max P(c)\prod_i P(a_i \mid c_k) \tag{8}$$

The aforementioned probability $P(c)$ and conditional probability $P(a_i|c)$ can be calculated by Eq. (9) and Eq. (10):

$$P(c) = \frac{\sum_{j=1}^{n}\delta(c_j, c)+1}{n+l} \tag{9}$$

$$P(a_i \mid c) = \frac{\sum_{j=1}^{n}\delta(a_{ji}, a_i)\delta(c_j, c)+1}{\sum_{j=1}^{n}\delta(c_j, c)+n} \tag{10}$$

In the formula, $n$ is the total number of training instances; $l$ is the total number of categories; $n_i$ is the possible value of attribute $i$; $c_j$ is the category of instance $j$; $a_{ji}$ is property $I$ of instance $j$; and $\delta(\bullet)$ is a binary function wherein if the two parameter values are the same, the function value is 1, but otherwise, it is 0.

#### 4.3.4 Algorithm

```
Input path: the path of text
Output result{<filename, category>:
BEGIN
1: Read text path
2: Read the stop words list
    Stpwrdlst = readfile(path)
3: Load the dictionary
    Jieba.load(path)
4: Segment
    for text in file:
    segs = Jieba.cut(text)
5: Remove stop words
    forseg in segs:
ifseg not in stpwrdlst and seg.isdigit() == False:
            sentence += " " + seg
6: Extract the feature words
6.1: Tf-idf
    Key words = jieba.analyse.extract_tags(sentence)
6.2: TextRank
    Key words= jieba.analyse.textrank(sentence)
6.3: Word2vec
    Key words = pandas. Series (word2vec(sentence))
7: Build the bunch data type
    Bunch(target_name, filename, label, content)
8: Build the words vector and weight matrix
    Tdm = TfidfVectorizer.fit_transform(Bunch)
9: Create the classifier
    clf = MultinomialNB(alpha=0.05).fit(tdm, label)
10:Output
    Print (precision_score)
    Print (recall_score)
    Print (f1_score)
END
```

## 5 EXPERIMENTAL DESIGN AND RESULT ANALYSIS

### 5.1 Experimental Environment

The experiment was conducted in an environment with a CPU that is an Intel(R) Core, 8G of memory, a Windows10 Enterprise Edition operating system, and python version 3.6.

### 5.2 Data Description

The railway complaint texts come from the railway complaint data of the Beijing Railway Administration from 2016, 2017 and 2018. The data include telephone, mail and website messages. The complaint data of 2016 and 2017 were used as the training set, and the complaint data of 2018 were used as the test set, which is listed in Tab. 8.

**Table 8** Training set and test set

| Data Usage | Year | Text Quantity |
|---|---|---|
| Training set | Complaint data of 2016 | 12583 |
| Test set | Complaint data of 2018 | 2068 |

Tab. 9 shows the categories of complaint text, includes cleaning service, catering service, ticket service, public security service, passenger service, train service, sales service, etc.

**Table 9** Type and number of complaints

| Type | Training Set | Test Set |
|---|---|---|
| Cleaning Service | 33 | 9 |
| Catering Service | 261 | 29 |
| Ticket Service | 5185 | 650 |
| Public Security Service | 873 | 190 |
| Passenger Service | 3819 | 619 |
| Train Service | 2149 | 504 |
| Sales Service | 116 | 31 |

### 5.3 Evaluation Indicators

There are three main evaluation indexes of the text classification algorithm: accuracy $P$, recall $R$ and value $F_1$. The formulas are from Eq. (11) to Eq. (13):

$$p = \frac{s_r}{s_a} \tag{11}$$

$$R = \frac{s_r}{s_o} \tag{12}$$

$$F_1 = \frac{2 \times P \times R}{P + R} \tag{13}$$

In the formulas, $s_r$ refers to the number of correctly classified texts; $s_a$ refers to the actual number of classified texts; $s_o$ refers to the number of texts belonging to this category; and $F_1$ is calculated by the accuracy rate $P$ and recall rate $R$.

### 5.4 Experimental Results

On the basis of dataset, Scikit-Learn is used for experiments in this study. The Naive Bayesian (NB) text classification algorithm is used to test the original text and preprocessed text, respectively. The Laplacian smoothing parameter of the Naive Bayesian algorithm is *alpha* = 0.05. The results are shown in Tab. 10.

**Table 10** Experimental results

| Text | $P$ | $R$ | $F_1$ |
|---|---|---|---|
| Eigenvalue not extracted | 0.521 | 0.471 | 0.420 |
| TF-IDF | 0.778 | 0.778 | 0.771 |
| TextRank | 0.737 | 0.731 | 0.720 |
| Word2vec | 0.320 | 0.261 | 0.2142 |

### 5.5 Result Analysis

(1) The texts of Chinese complaints often have complex and intertwined business contents, serious colloquialisms, and no refined expression of a core meaning. Therefore, it is impossible to classify all complaints in the original body of texts or directly extract their core meanings, and the text can only be classified by extracting the key features of the dataset. Therefore, the above experimental results also show that the accuracy of the classification results using the original texts is far lower than the classification accuracy after the key features are extracted.

(2) TF-IDF extracts keywords and is a simple and effective method for extracting keywords. The main idea of this method is to precalculate the frequencies of all words appearing in the test set, calculate the *idf* value, and then find the cutoff for extracting keywords. Thus, each word of the sentence is used to calculate the *tf* value, which is multiplied by the TF-IDF value. The larger the value, the higher the priority as a keyword. The algorithm filters out some common but irrelevant words while preserving important words that affect the entire text. However, the link between the word contexts is lost.

The classic TextRank algorithm does not depend on other training sets, focusing on the internal word structure of the text, and can be used to build a graph model for keyword extraction.

The *idf* value of TF-IDF depends on its environment, which gives this method a statistical advantage; that is, the method can predict the importance of a word in advance, which is an improvement over TextRank. TextRank relies only on the word itself. The importance of each word is the same at the beginning of the analysis. TF-IDF is a pure word-frequency consideration (whether *tf* or *idf*) used to calculate the score of a word. To extract keywords, no single words are used at all. Instead, the relevance of the text is used. TextRank considers the relevance of words (linking adjacent words), which makes it better than TF-IDF.

On the whole, TF-IDF and TextRank have their own advantages and disadvantages. In actual practice, the difference between the two methods is not obvious. You can use either as a reference.

(3) Word2vec uses a shallow neural network model to automatically learn the appearance of words in the text, embedding words into a high-dimensional space, usually in the 100-500 dimensions. In the new high-dimensional space, words are represented as a word vector. Compared with the traditional text representation, the word vector generated by Word2Vec indicates that the semantic relationship between words is better reflected in the high-dimensional space. That is, the words with similar semantics are closer in the high-dimensional space. The use of word vectors avoids the "dimensional disaster" problem of word representation. In terms of practical applications, the extraction of feature word vectors is based on an already trained word vector model. The training of the word vector model requires a large amount of data to achieve better results, while the Chinese wiki dataset is recognized as a large Chinese dataset. The feature word vector of this body of text is extracted from the word vector generated by the Chinese wiki dataset.

## 6 CONCLUSION

In this paper, a structured space vector method is used to construct a naive Bayes classifier to automatically classify railway complaint texts. Because the railway complaint text has the characteristics of wide business, diverse events, serious textual record, serious interference and useless information, the traditional text classification method is adopted directly, and the accuracy of text classification is low. The classification accuracy is only 0.521 by the original text. Therefore, this paper adopts the method of feature value extraction to further improve the classification accuracy of text. In this paper, three feature extraction algorithms are compared, includes TF-IDF, TextRank and Word2vec. TF-IDF and TextRank respectively achieve the accuracy of 0.778 and 0.737, but the Word2vec only achieves the precision of 0.320. Through analysis, Word2vec has a good understanding of the semantics of a single text, and the extracted keywords are also better for a single text, but for a series of texts, the extracted keywords cannot reflect the characteristics of a certain type of text. So the accuracy of text categorization is even lower than the original text.

Even with the TF-IDF and TextRank, the accuracy of text categorization is non-ideal. Through analysis, since the

complaint location and business are relatively similar, the extracted keywords also have more repetitions, thus affecting the accuracy of classification. In the latter research, it is important to remove the duplicated and confusing words between the extracted texts, highlight the characteristics of the text, and further improve the classification accuracy of the text.

## 7 REFERENCES


[1] Chen, K. C. (2009). Text mining e-complaints data from e-auction store with implications for internet marketing research. *Journal of Business & Economics Research, 7*(5), 15-24. https://doi.org/10.19030/jber.v7i5.2286

[2] Lee, M. J., Singh, N., & Chan, E. S. (2011). Service failures and recovery actions in the hotel industry: A text-mining approach. *Journal of Vacation Marketing, 17*(3), 197-207. https://doi.org/10.1177/1356766711409182

[3] Ghazizadeh, M., McDonald, A. D., & Lee, J. D. (2014). Text mining to decipher free-response consumer complaints: Insights from the NHTSA vehicle owner's complaint database. *Human factors, 56*(6), 1189-1203. https://doi.org/10.1177/0018720813519473

[4] Pan Gang. (2013). *Research and Application of Customer Complaint Management in Shanghai Mobile Company*. Shanghai: Shanghai Jiaotong University.

[5] Tang Shengtao. (2013). Research on Analysis Method of Operator Customer Complaint Based on Data Mining. *Internet World*, (3), 53-55.

[6] Xia Haifeng & Chen Junhua. (2013). Intelligent Classification of Complaint Hotspots Based on Text Mining. *Journal of Shanghai Normal University: Natural Science Edition, 42*(5), 470-475.

[7] Shi Zhifang. (2013). *Automatic Discovery and Analysis of Hot Issues in the Mobile Complaint Information*. Beijing University of Posts and Telecommunications.

[8] Liu Xin. (2017). *Study on Evaluation of High-speed Railway Passenger Service Quality Based on Social Network*. Southwest Jiaotong University.

[9] Onan, A., Korukoğlu, S., & Bulut, H. (2016). Ensemble of keyword extraction methods and classifiers in text classification. *Expert Systems with Applications*, 57, 232-247. https://doi.org/10.1016/j.eswa.2016.03.045

[10] Sowmya, D. & Sheeba, J. I. (2016). Keyword extraction using particle swarm optimization. *Procedia Computer Science*, 85, 183-189. https://doi.org/10.1016/j.procs.2016.05.208

[11] Chen, Y., Wang, J., Li, P., & Guo, P. (2019). Single document keyword extraction via quantifying higher-order structural features of word co-occurrence graph. *Computer Speech & Language*. https://doi.org/10.1016/j.csl.2019.01.007

[12] Rossi, R. G., Marcacini, R. M., & Rezende, S. O. (2014). Analysis of domain independent statistical keyword extraction methods for incremental clustering. *Learning and Nonlinear Models, 12*(1), 17-37. https://doi.org/10.21528/LNLM-vol12-no1-art2

[13] Mihalcea, R. & Tarau, P. (2004). *Textrank: Bringing order into text.* In *Proceedings of the 2004 conference on empirical methods in natural language processing*.

[14] Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient estimation of word representations in vector space. *arXiv preprint arXiv*: 1301.3781.

[15] Abilhoa, W. D. & De Castro, L. N. (2014). A keyword extraction method from twitter messages represented as graphs. *Applied Mathematics and Computation*, 240, 308-325. https://doi.org/10.1016/j.amc.2014.04.090

[16] Siddiqi, S. & Sharan, A. (2015). Keyword and keyphrase extraction techniques: a literature review. *International Journal of Computer Applications*, 109(2). https://doi.org/10.5120/19161-0607

[17] Jiang, L., Li, C., Wang, S., & Zhang, L. (2016). Deep feature weighting for naive Bayes and its application to text classification. *Engineering Applications of Artificial Intelligence*, 52, 26-39. https://doi.org/10.1016/j.engappai.2016.02.002

[18] Xie, D., Pei, W., Zhu, W., & Li, X. (2017, October). Traditional Chinese medicine prescription mining based on abstract text. In *2017 IEEE 19th International Conference on e-Health Networking, Applications and Services (Healthcom)* (pp. 1-5). IEEE. https://doi.org/10.1109/HealthCom.2017.8210822

[19] Beliga, S., Meštrović, A., & Martinčić-Ipšić, S. (2015). An overview of graph-based keyword extraction methods and approaches. *Journal of information and organizational sciences, 39*(1), 1-20.

[20] Hinton, G. E. (1986, August). Learning distributed representations of concepts. In *Proceedings of the eighth annual conference of the cognitive science society* (Vol. 1, p. 12).



**Contact information:**

**Lifeng LI**
School of Economics and Management,
Beijing Jiaotong University,
No. 3, Shangyuancun, Haidian District, Beijing, China
E-mail: 98940261@bjtu.edu.cn

**Wenxing LI,** Professor
School of Economics and Management,
Beijing Jiaotong University,
No. 3, Shangyuancun, Haidian District, Beijing, China
E-mail: wxli@bjtu.edu.cn

**Daqing GONG,** Assistant Professor
(Corresponding author)
School of Economics and Management,
Beijing Jiaotong University,
No. 3, Shangyuancun, Haidian District, Beijing, China
E-mail: dqgong@bjtu.edu.cn