Research on the application of Naive Bayes and Support Vector Machine algorithm on exercises Classification

Qi-na Li¹, Ting-hui Li¹*

¹College of Electronic Engineering, Guangxi Normal University, Guilin, Guangxi 541004, China
*Corresponding author’s e-mail: tinghuili@gxnu.edu.cn

Abstract. In this paper, exercises on electronic major courses of college are classified, further, compared with the accuracy of algorithms based on the Naive Bayes and Support Vector Machine (SVM) in the classification of exercises. Data sets from exercises related to the content on the course of Data Structure are selected. According to knowledge content of course and characteristics of exercises, judgment and multiple-choice questions are used in experimental data, and these exercises are divided into seven categories depending on content of chapters. Firstly, the package of Jieba participle is applied to segment text of exercises, and the proper nouns of knowledge content are identified by importing a custom dictionary. Because characteristic words of short text are quite sparse, weight method about TF-IDF is directly adopted to feature representation of text instead of feature selection. Finally, algorithms based on the Naive Bayes and SVM are employed to train and classify text of exercises. The experimental results show that algorithm based on SVM has lower error rate and higher classification accuracy than the Naive Bayes.

1. Introduction
Exercises are an important part of teaching resources, which not only allow students to consolidate the knowledge points they have mastered, but also enable teachers to discover the points of knowledge undermastered by students. So the classification of the exercises is particularly important. The number of words in the text of exercises range from a dozen to dozens, which belongs to short text and even very short text. So classification of exercises is an application scenario for classification about short text. Commonly, traditional methods on text classification such as KNN [1], Logistic Regression [2], Naive Bayes [3] and SVM algorithm [4], in addition to Latent Dirichlet Allocation (LDA) for classification of short text [5], and the most widely used algorithms are Naive Bayes and SVM, even the combination of LDA theme models and Naive Bayes and SVM [6]. At present, the most popular method on short text classification is deep learning, which has best classification effect [7], but because of the small number of exercises sets and the weak explanation of the model in deep learning, this paper adopts the traditional method on text classification. The algorithms of exercises text classification used in this paper are Naive Bayes and SVM, because they are easy to understand and have less computational complexity.

The application scenarios of Naive Bayes are mainly in text classification and fraud detection. Xiao et al presented the method of Naive Bayes to classify the patent texts, and the result of the classification performance was better than the SVM, and finally improved the security of patent text classification [8]. The main application scenes of SVM are text classification, image recognition and binary classification. By comparing algorithm of SVM with other text classification, Liu et al showed that the classification method based on SVM is superior to KNN and Naive Bayes [9]. Yunlin et al improved the performance...
of classifier based on SVM-KNN algorithm by feeding back and improving probability of classified prediction [10]. Zun et al applied algorithms of Naive Bayes and SVM to classify automatically IT research papers, proving that classifier of SVM is more accurate than Naive Bayes [11].

In this paper, the exercises text is pre-processed by using Jieba participles, the vector space model with weight of TF-IDF is constructed, and the exercises text is classified by adopting algorithms based on Naive Bayes and SVM. Combined with error rate, accuracy rate, recall rate and F1-Score, the accuracy of exercises classification of the two algorithms is evaluated.

2. Text of exercises pre-processing

2.1. Word segmentation
This article adopts the precise pattern about word segmentation of Jieba, which attempts to divide the sentences as accurately as possible. Therefore, it is suitable for the text analysis of exercises. Most of the knowledge points in the textbook of Data Structure are proper nouns, and the original package on word segmentation of Jieba cannot identify these new words, so custom dictionaries must be imported to improve the recognition rate of new words.

2.2. Removal of stop words
Auxiliary word, function words, adverbs, punctuation, letters, and illegal characters and so on from the exercises are removed. Combined with stop words list about Harbin institute of Technology, stop words library of machine intelligence laboratory of Sichuan University and Baidu's stop words list, it was reprocessed, and then added to the list of stop words similar to the "with" "total" "how much" and so on, which are of little significance to the keywords of knowledge points. A more comprehensive list of stop words for the exercises Classification of the Data Structure was rearranged, with a total of 2,503 stop words.

2.3. Merging synonyms
The same knowledge point has different expressions in different exercises, so it is necessary to merge synonyms. For example, "Simple selection sort" and "Direct selection sort" represent the same knowledge point, unified combined into "Simple selection sort". And “Hash Table, HASH Table, and Hash List” are merged into “Hash Table” and so on.

2.4. Feature representation of text
The purpose of the text representation is to convert the pre-processed text into a computer-understandable way, which is the most important part of determining quality of text classification. Traditionally, text representation methods of bag-of-words model and vector space model are commonly used. The bag-of-words model treats a single sentence in a text as a collection of words, and a "the bag of words" represent the text. Based on the bag-of-words model, VSM becomes the most commonly used method for text representation [12]. VSM was proposed by Salon [13], It regards document \textbf{d}_i as an n-dimensional vector in vector space, such as:

\[
\textbf{d}_i = \left( (v_1, w_{i1}), (v_2, w_{i2}), \ldots, (v_j, w_{ij}) \right)
\]

(1)

Where \(v_j\) represents the feature word and \(w_{ij}\) represents the weight of the feature word \(v_j\) in the text \(\textbf{d}_i\), which is generally calculated using the TF-IDF value [14]. Calculation formula of TF-IDF is as follows:

\[
w_{ij} = \text{TFIDF}_{ij} = \text{TF}_{ij} \times \text{IDF}_j
\]

(2)

\[
\text{TF}_{ij} = \frac{n(v_j,d_i)}{\sum_{i=0}^{n(d_i)}}
\]

(3)

\[
\text{IDF}_j = \log \left( \frac{|D|}{n(v_j,d)} + 0.01 \right)
\]

(4)
3. Classification algorithm

After constructing the vector space of TF-IDF weight, we can use machine learning algorithms to classify text of exercises.

3.1. Naive Bayes

Classification of Naive Bayes is a supervised learning classification algorithm based on Bayesian rules. Its basic principle is that when the prior probability of the class is known and the conditional probability of all values of each attribute in the class is known, the conditional probability that the sample to be classified belongs to a certain category can be calculated, and classification of the sample with the greatest conditional probability can be selected. It completes text categorization by assigning probabilistic labels to input documents to be categorized. The Bayesian formula is as follows:

$$P(c_i|d) = \frac{P(d|c_i)P(c_i)}{P(d)}$$  \hspace{1cm} (5)

Among them, the probability that input text \(d\) belongs to category \(c_i\) can be obtained. Where \(d\) denotes text, \(c_i\) denotes classification category. According to the assumption of Bayesian model, each text in the data set is independent of each other, so the category corresponding to the maximum probability value obtained by input text \(d\) is the category of text \(d\). The calculation formula is as follows:

$$Y = \arg \max P(c_j) \prod_j P(d|c_j)$$  \hspace{1cm} (6)

3.2. Support Vector Machine

Support Vector Machine (SVM) was first proposed by Vapnik et al [15]. Linear classifiers use hyperplane boundaries, while non-linear classifiers use hypersurfaces. This hyperplane can be represented by the classification function \(f(x) = W^T + b\), when \(f(x) = 0\), \(x\) is the point on the hyperplane, and the point of \(f(x) > 0\) corresponds to the data point of \(y = 1\), and the point of \(f(x) < 0\) corresponds to the point \(y = -1\). The principle of SVM is to map points in low-dimensional space to high-dimensional space so that they become linear separable, and then use the principle of linear partition to judge the classification boundary. It is a linear partition in high-dimensional space, but it is a nonlinear partition in the original data space. SVM has many unique advantages in solving small sample, non-linear and high-dimensional pattern recognition problems, and can be extended to other machine learning problems such as function fitting.

3.3. Evaluation of classification results

In order to evaluate the experimental results and compare the classification accuracy of Naive Bayes and Support Vector Machine, this paper selects four indicators: error rate, accuracy rate, recall rate and F1-Score. Table 1 shows that parameters of classified evaluation index.

| Actually belong to | Actually does not belong to |
|--------------------|-----------------------------|
| Prediction belongs to | A | B |
| Prediction does not belong to | C | D |

The calculation formula for each indicator is as follows:

Error rate: \(error = \frac{n}{N}\)  
Accuracy rate: \(P = \frac{A}{A+B}\)  
Recall rate: \(R = \frac{A}{A+C}\)  
F1-Score: \(F1 = \frac{2PR}{P+R}\)

For the formulas listed above, \(n\) represents the number of misclassification exercises for the classifier, \(N\) represents the actual total number of exercises, and \(A\) shows the number of exercises that actually belong to the class and are classified by the classifier into the class. \(B\) denotes the number of exercises that are not actually in the class but are classified by the classifier into that class. \(C\) represents the number of exercises that actually belong to the class but are not classified into that class.
4. Analysis results

4.1. Experimental data
There are 1034 exercises in this experiment, including 512 multiple-choice questions and 522 judgement questions. According to the knowledge structure system of Data Structure, exercises are divided into seven categories. Table 2 shows seven categories and their descriptions. This experiment chooses editing tool of Pycharm in the environment of Python language, uses polynomial Bayesian algorithm of Scikit-Learn library and linear SVM algorithm to classify short text. The experimental data is divided into training set and test set, which are divided into two experiments.

| Category | Description                   |
|----------|------------------------------|
| A1       | Linear list                  |
| B1       | Stack and Queue              |
| C1       | String, Array and Generalized list |
| D1       | Tree and Binary Tree         |
| E1       | Graph                        |
| F1       | Search                       |
| G1       | Sorting                      |

4.2. Experimental result

Table 3. The error rate, precision rate, recall rate, F1-Score of the two compared algorithms

| Evaluation index | First | Second | First | Second |
|------------------|-------|--------|-------|--------|
| Naive Bayes      | SVM   | Naive Bayes | SVM   |        |
| error rate       | 12.57% | 8.38% | 10.79 % | 6.16 % |
| Accuracy rate    | 0.880 | 0.918 | 0.893 | 0.940 |
| Recall rate      | 0.874 | 0.916 | 0.892 | 0.938 |
| F1               | 0.873 | 0.914 | 0.891 | 0.938 |

Figure 1. Experimental result

Table 3 and Figure 1 show the results about two experiments of exercises classification for Naive Bayes and SVM. It can be seen from the results of the first and second comparison experiments that due to the number of exercises in the training sample set are increased, the error rate of the predictive exercises of the classifier decreases, and the accuracy, recall rate and F1 value also increase. The results
of each experiment shows that SVM algorithm has a lower error rate in predicting the exercises than the Naive Bayes algorithm. In the second experiment, the error rate of Naive Bayes is 10.79%, while the error rate of SVM is reduced to 6.16%. At the same time, the accuracy rate, recall rate and F1 value of SVM algorithm are higher than the Naive Bayes algorithm. Therefore, the algorithm of SVM is more accurate than the Naive Bayes in exercises classification.

5. Conclusion and Future work
Exercises is an important learning resource in the process of education and teaching, which can test knowledge mastered of students. Therefore, it is particularly important to classify exercises according to chapters, which can consolidate the knowledge learnt by students at different stages, and then check the leaks and fill the gaps. It also prepares for the recommendation of exercises according to the students’ situation of doing exercises in the future. This paper classifies the relevant exercises in the course of Data Structure, deletes stop-words and merges synonyms, and then constructs word vector space of TF-IDF. Finally, we apply machine learning algorithms based on Naive Bayes and Support Vector to classify text of exercises. The results of two experiments show that the training corpus determines the accuracy of classification algorithm of exercises text to a certain extent. The more training data sets and the better the pre-processing of exercises text, the higher the accuracy of exercises text classification based on Naive Bayes and Support Vector Machine algorithms. At the same time, it can be seen from the results of each experiment that the SVM algorithm is higher than Naive Bayes, regardless of error rate or accuracy rate, recall rate and F1 value. Therefore, the classification performance of SVM algorithm is better than Naive Bayes algorithm in text classification of exercises.

More data sets and experiments will be added on the work of future and short text classification algorithms of KNN, Logical Regression and LDA topic models and their improvements will be compared. At the same time, in order to solve the problem of sparse feature in the short text, the next step is to expand the feature words, and use LDA topic model to expand the feature words, combined with SVM and Naive Bayes algorithms to classify exercises, so as to improve the accuracy of exercises classification.

References
[1] Tan, Y.X. (2018) An Improved KNN Text Classification Algorithm Based on K-Medoids and Rough Set. In: 2018 10th International Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC). Hangzhou. pp. 109-113.
[2] Prabhat, A., Khullar, V. (2017) Sentiment classification on big data using Naïve Bayes and logistic regression. In: 2017 International Conference on Computer Communication and Informatics (ICCCI). Coimbatore. pp. 1-5.
[3] Jiang, Q., Wang, W., Han, X., Zhang, S., Wang, X., Wang, C. (2016) Deep feature weighting in Naive Bayes for Chinese text classification. In: 2016 4th International Conference on Cloud Computing and Intelligence Systems (CCIS). Beijing. pp. 160-164.
[4] Dilrukshi, I., Zoysa, K. D and Caldera, A. (2013) Twitter news classification using SVM. In: 2013 8th International Conference on Computer Science & Education. Colombo. pp. 287-291.
[5] Chen, Q., Yao, L., Yang, J. (2016) Short text classification based on LDA topic model. In: 2016 International Conference on Audio, Language and Image Processing. Shanghai. pp. 749-753.
[6] Wang, X., Wang, J., Yang, Y., Duan, J. (2017) Labeled LDA-Kernel SVM: A Short Chinese Text Supervised Classification Based on Sina Weibo. In: 2017 4th International Conference on Information Science and Control Engineering (ICISCE). Changsha. pp. 428-432.
[7] Sotthisopha, N., Vateekul, P. (2018) Improving Short Text Classification Using Fast Semantic Expansion on Multichannel Convolutional Neural Network. In: 2018 19th IEEE/ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD). Busan. pp. 182-187.
[8] Xiao, L., Wang, G., Liu, Y. (2018) Patent Text Classification Based on Naive Bayesian Method. In: 2018 11th International Symposium on Computational Intelligence and Design (ISCID), Hangzhou. China. pp. 57-60.

[9] Liu, Z., Lv, X., Liu, K., Shi, S. (2010) Study on SVM Compared with the other Text Classification Methods. In: 2010 Second International Workshop on Education Technology and Computer Science. Wuhan. pp. 219-222.

[10] Lin, Y., Wang, J. (2014) Research on text classification based on SVM-KNN. In: 2014 IEEE 5th International Conference on Software Engineering and Service Science. Beijing. pp. 842-844.

[11] Moe, Z. H., San, T., Khin, M. M., Tin, H.M. (2018) Comparison Of Naive Bayes And Support Vector Machine Classifiers On Document Classification. In: 2018 IEEE 7th Global Conference on Consumer Electronics (GCCE). Nara. pp. 466-467.

[12] Lauly, S., Boulange, R.A., Larochelle, H. (2014) Learning Multilingual Word Representations using a Bag-of-Words Autoencoder. Computer Science.arXiv:1401. 1803.

[13] Salton, G., Wong, A., Yang, C.S. (1975) A vector space model for automatic indexing. Communication of the ACM. 18: 613-620.

[14] Liu, C., Sheng, Y., Wei, Z., Yang, Y. (2018) Research of Text Classification Based on Improved TF-IDF Algorithm. In: 2018 IEEE International Conference of Intelligent Robotic and Control Engineering (IRCE). Lanzhou. pp. 218-222.

[15] Vapnik, V. N. (1995) The nature of statistical learning theory [M]. New York: Springer-Verlag. 205-208.