



Research on Text Classification Based on Automatically Extracted Keywords

Pin Ni, University of Auckland, New Zealand

 <https://orcid.org/0000-0003-4516-1249>

Yuming Li, University of Auckland, New Zealand & University of Liverpool, UK

 <https://orcid.org/0000-0003-2219-9033>

Victor Chang, Teesside University, UK

ABSTRACT

Automatic keywords extraction and classification tasks are important research directions in the domains of NLP (natural language processing), information retrieval, and text mining. As the fine granularity abstracted from text data, keywords are also the most important feature of text data, which has great practical and potential value in document classification, topic modeling, information retrieval, and other aspects. The compact representation of documents can be achieved through keywords, which contains massive significant information. Therefore, it may be quite advantageous to realize text classification with high-dimensional feature space. For this reason, this study designed a supervised keyword classification method based on TextRank keyword automatic extraction technology and optimize the model with the genetic algorithm to contribute to modeling the keywords of the topic for text classification.

KEYWORDS

Genetic Algorithm, Keyword Classification, Keyword Extraction, Linear Regression, Naive Bayes, Random Forest, SVM, Text Classification, Textrank

1. INTRODUCTION

Keyword, key sentence and key paragraph as important features of text data, which can reflect the topic of the document to a certain extent (Beliga, Meštrović, & Martinčić-Ipšić, 2015). Automated keyword extraction can extract the most significant key information from specific documents, thus speeding up the abstraction of specific descriptive instances from massive text data. On the other hand, these keyword information as fine-grained text can be more macroscopically divided into different categories. This classification method can be used not only for keyword topic modeling but also for text categorization tasks based on high-dimensional feature space (Onan, Korukoğlu, & Bulut, 2016), (Lautenbacher, Bauer, Sieber, & Cabral, 2010). This could achieve more accurate word-of-mouth text classification (Jansen, Zhang, Sobel, & Chowdury, 2009), (Hung & Lin, 2013), topic feature analysis of social relationships (Hauffa, Lichtenberg, & Groh, 2012), keyword classification (Fernando, 2018),

DOI: 10.4018/IJEIS.2020100101

document classification (Onan et al., 2016), (Hu et al., 2018), (Puri & Singh, 2019), recommendation for user interest features (W. Wu, Zhang, & Ostendorf, 2010), (Meng & Gao, 2019), etc.

Text categorization is a modeling method for categorizing documents according to preset categories (Liu & Wang, 2007), (Schütze, Manning, & Raghavan, 2007), (Y.-C. Wu, 2015), which has widely used in text mining (Al-Thuhli, Al-Badawi, Baghdadi, & Al-Hamdani, 2017), covering information retrieval (Boughareb & Farah, 2013), (Ghneimat & Shaout, 2016), sentiment analysis (Jain, Kumar, & Mahanti, 2018), topic mining, document organization, spam filtering, news classification etc (Aggarwal & Zhai, 2012). However, there are still many difficulties in text categorization in high-dimensional feature space (Joachims, 2002). When entire words in the document served as training features, the computational complexity will be greatly increased, making the task of text categorization transformed into a type of computationally intensive task (Onan et al., 2016). Therefore, as the most relevant feature of documents and a relatively reasonable data dimensionality reduction to a certain extent, keywords can become relatively ideal feature candidates in classification modeling (Liu & Wang, 2007), (Rossi, Marcacini, & Rezende, 2014). From the perspective of classification accuracy, the text classification method based on keywords as features may be an effective approach that worth to be explored, and from the perspective of the actual application from the micro to the macro, the keyword-based approach is also more suitable for the real situation in the information retrieval scenario where the user inputs fine-grained features (e.g. words, character, punctuation character) to more accurately match the corresponding text instance.

For this reason, this study designed a supervised keyword classification method based on TextRank keyword automatic extraction technology and optimize the model with the Genetic Algorithm to contribute to text classification for modeling the feature of the topic. This method improved the accuracy compare with the conventional classification and clustering methods and solve the problem about conventional methods do not have the mechanisms of self-renewal keywords and self-adjusting classification weights, to realize the attributes that keyword topic model can gradually improve with the input of new data. And the study also compared the effect of other commonly used classification methods in keyword classification. Experiments show that the proposed method achieves ideal performance on test datasets of ACM collection (Table 5).

2. LITERATURE REVIEW

There are two approaches to extract text keywords, one is to convert the question of selecting keywords into a binary classification problem of whether a word is a keyword through a supervised machine learning method. The other is use the statistical word frequency method to the calculation of weights in an unsupervised method and select keywords with higher weight (e.g. Term Frequency - Inverse Document Frequency, TF-IDF) (Schütze et al., 2007). In addition, the candidate keywords can be placed in the structure of the graph by the method of the graph model to find the top K words associated with the other words, such as TextRank (Mihalcea & Tarau, 2004).

(Lee & Kim, 2008) proposed an improved TF-IDF method for extracting keywords. By applying the weight calculated by TF-IDF to their designed TTF (Table Term Frequency) model, to eliminate the common cross-domain words to improve the accuracy of keyword extraction.

(Tixier, Malliaros, & Vazirgiannis, 2016) proposed an approach for keywords extraction based on the K-truss algorithm (Cohen, 2008). This method is less than TextRank in the accuracy but greater than the TextRank in the recall rate. And the value of F-measure is higher than TextRank when applied to text in a finer granularity level, but it is lower when applied to text in a more coarse-grained level.

(Zhao, Yu, Lu, Liu, & Li, 2016) proposed an approach for extracting keywords based on FP-Growth. This approach can remove words with similar meanings in alternative keywords to improve the accuracy and practicability of the extracted keywords. In the same conditions, the accuracy, recall rate and F-measure value were 12.1%, 10.1%, and 10.9% higher than TF-IDF, respectively. And the computational complexity is much lower than TF-IDF, that is, the calculation efficiency is higher.

(Abilhoa & De Castro, 2014) uses Graph-based TKG (Twitter Keyword Graph) to find keywords for messages crawled from Twitter. TKG can be implemented without prior training, but due to lacks noise filtering, resulting in operational efficiency is not ideal (Horita, Kimura, & Maeda, 2016), (Horita et al., 2016) extracted keywords from documents in Wikipedia. Among them, they use TSNC (Top Consecutive Nouns Cohesion) on the selection of candidate keywords and uses the method of the Dice Coefficient and the dataset of Keyphraseness on the selected keywords. However, the selection of TCNC of candidate keywords is not as effective as expected.

TextRank does not need to prepare a corpus. And as an unsupervised approach, it can be done without prior training. Therefore, traditional keyword extraction tools (e.g. RAKE) have a better performance at process speed (Rose, Engel, Cramer, & Cowley, 2010). (Balcerzak, Jaworski, & Wierzbicki, 2014) argue that TextRank can accurately define keywords and is quite convenient in an unsupervised manner. (Liu & Wang, 2007) believes that the graph-based keyword extraction algorithm such as TextRank achieves better results than the previous supervised methods. However, some authors on the Internet believe that there is a flaw in the understanding of the relationship between the context, which is needed for improvement.

2.1. TextRank

TextRank applies a graph-based ranking algorithm similar to PageRank of Google (Page, Brin, Motwani, & Winograd, 1999) and HITS algorithm and uses it for the process of text (Kleinberg, 1999). The output of TextRank is a collection of words or sentences that can represent the text separately. Among them, the number of words can be determined by the length of the text or according to the demands. The collection of words will be placed in the graph as vertices, calculating the importance and sorting the order, and select several preset numbers of the most important words as keywords.

2.2. Graph-Based Sorting

TextRank is a graph-based keyword extraction method, the principle is: voting rights are given at each vertex. If there is a vertex connected to another vertex, it will be recorded as a vote of the number of votes of the linked vertex, and the result of each vertex in the whole graph will be obtained by using the recursive method. The number of votes obtained by each vertex related to the importance of the vertices, and voting from the more important vertices, its weight ratio will be higher. In general, in the directed graph $G, G = (V, E)$, V represents a vertex, E represents an edge, and E is a subset of $V \times V$ as well. Given a V_i , where $Out(V_i)$, represents the set of vertices pointing to V_i , and $In(V_i)$ represents the set of vertices pointed to by V_i . Hence, the score of V_i can be calculated by the following formula. Where d represents the damping coefficient and takes a value between $[0, 1]$:

$$S(V_i) = (1 - d) + d \times \sum_{j \in In(V_i)} \frac{1}{|Out(V_j)|} S(V_j) \quad (1)$$

2.3. The Applications of TextRank in the Study

TextRank breaks down all the sentences in the text, filters the sentences according to the stop words as demands, or only retains words with specific part of speech. Through this approach, a collection of sentences and words will be obtained. Setting each word as a vertex and setting its window size to K' , and assuming that a single sentence consists of the set of words $(w_1, w_2, w_3, \dots, w_n)$ are all in the same window. There is an undirected graph without edge weights in each window between the nodes corresponding to any two words. Based on the above constructed graph, the importance of each node in each word will be calculated. The keywords are the words with the highest weight. In

the study, use the weights of keywords extracted by TextRank to sort the significance, and extract the top k number of important words as keywords.

2.4. Optimization

Optimizing the model can improve the accuracy of model effectively. Therefore, this study introduces the Genetic Algorithm as a reward mechanism. After the completed classification stage, adjust the classification results and its weight according to the categories of the original data, and a set of coefficients with the best adjustment effect in a certain number of iterations is selected as the adjustment weight coefficient to improve the overall classification accuracy.

2.5. Genetic Algorithm

Genetic Algorithm (GA), which is an Adaptive Heuristic Search Algorithm based on natural elimination and genetic evolution. GA can obtain the best solution in the condition of a few known conditions. It is a general algorithm and widely used in search technology to solve optimization problems (Sivanandam & Deepa, 2008), (M. Zhang, Wang, & Liu, 2017), (Chang, 2007).

GA is derived from Darwin's concept of "natural selection" in evolution, and the rules are named "evolution" and "survival of the fittest." The GA, applied to optimization problems, was published by J.H. Holland in 1975, and its mode of operation completely mimics the evolution of biology in nature (Sivanandam & Deepa, 2008).

2.6. Comparison of GA and Other Optimization Methods

(Sivanandam & Deepa, 2008) argue that the Greedy Algorithm better than GA in efficiency, but the results of GA are closer to the optimal solution. In this study, the step of adjusting the weight is only used for the adjustment of the weights after the classification step, and the requirement for the correct rate is higher than the efficiency, therefore, the study uses GA as the optimization method.

2.7. Instructions

In the study, we adjusted the classification accuracy of the text used for verification by adjusting the weights of the keywords in each topic. The formula for calculating the weights of the new topic keyword can be found at equation 5.

Therefore, we can consider "maximizing the accuracy of the text used for verification" as an optimization problem. The adaptation equation is $f(w_1, w_2, w_3, \dots, w_n) = Max - F$, and the $Max - F$ is the maximum value of the accuracy.

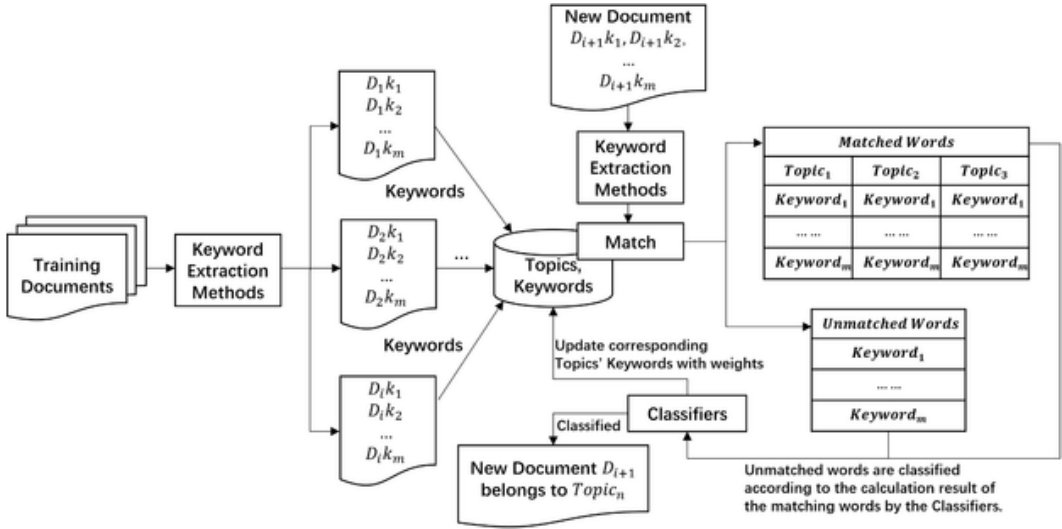
3. METHODOLOGY

3.1. Method Structure

Through TextRank algorithm can rank text words according to their importance to obtain the highest weight of several words, whose weights are used to determine the vertex importance based on the global information recursively obtained in the whole text graph sorting.

To some extent, these keywords can be regarded as representing the most relevant top (K) features described in the document. Therefore, these text keywords can be matched with the keywords co-occurring in the topic models, the sum of correlation degree values (comprehensive weight) of the combination of the respective weights of the co-occurrence keywords in the documents and the topic models can be calculated. And comparing with the results calculated in different topics categories, the topic category with the greatest degree of correlation can be chosen as the final category of attribution. The overall structure and process of the proposed method can be found in Figure 1.

Figure 1. Overall structure and process of the proposed method



3.2. The Value of Average Degree

$(T_n K_m t_n k_1, t_n k_2, \dots, t_n k_m)$ represents a keyword set (k_m) within the topic category (T_n), and (D_i) represents a document. If the document (D_i) of the marked category belongs to a certain topic $n(T_n)$, then each training document keyword set (K_i) and its weight (after converted to a percentage) are classified into the keyword set ($T_n K_m$) in topic (T_n) and its weight table, thus initially forming a training topic model. After extraction of the keywords of the new document, traversing all the words ($T_n K_m$) in each topic model to find out whether the same word appears. If the co-occurrence word is found in one or more topics, the weights of all the keywords in the document are quantified as a percentage, multiplied and summed by the importance values of the keywords co-occurring within each topic model, which could obtain comprehensive weight, and then compared with the comprehensive weight of the co-occurring word in other categories of topic. Through the law of large numbers, the topic with the largest comprehensive weight, all the keywords extracted by the current document and their weights belong to the corresponding category of topic, and updated (added the weight of a single co-occurring word to the importance degree (W_{sum}) and divided by the sum number of occurrence times of the word (tf) can obtain the weights of the related words in the topic category (average degree value, $\bar{W}_{n,j}$).

$$\bar{W}_{n,j} = \frac{W_{sum}}{tf} = \frac{\sum_{z=1}^{TF} W^z}{tf} \quad (2)$$

3.3. Comprehensive Weight

The calculation method of the comprehensive weight is as follows: the words in the document keyword list and all the words co-occurring in the topic categories, the weights (w_j) of keywords in the document and the importance value (W_{sum}) of same keywords in the topic are respectively matched

separately. After the importance value (W_{sum}) is divided by the sum of count in the topic, the average degree value ($\bar{W}_{n,j}$) could be obtained, and all co-occurring words in the current topic are calculated according to the topic category. The comprehensive weight (G_n / G'_n) could be obtained:

$$\begin{aligned}
 \text{Non - Percentile : } G_n &= \sum_{z=1}^{TF} (G_{1,1}, G_{1,2}, \dots, G_{n,j}) \\
 &= \sum_{z=1}^{TF} \bar{W}_{n,j} \times w_j
 \end{aligned} \tag{3}$$

$$\begin{aligned}
 \text{Percentile : } G'_n &= \sum_{z=1}^{TF} (G'_{1,1}, G'_{1,2}, \dots, G'_{n,j}) \\
 &= \sum_{z=1}^{TF} \bar{W}_{n,j} \times \frac{w_j}{\sum_{j=1}^J w_j}
 \end{aligned} \tag{4}$$

Due to the weights extracted by TextRank are not in a uniform range, it is difficult to compare the weights extracted from different documents directly. After quantifying as percentage, the weight proportion of each keyword in the category can be quantified. However, the method without percentage quantification can also be added to the experiment as a comparative test to enhance the reliability of the method. Therefore, based on the above methods, this study also designs another method to test: the weight of keywords in documents and the importance of words in the topic model are not converted by percentage. This is as a separate method participates in the experimental comparison.

3.4. Comparison of Comprehensive Weight

After obtaining the comprehensive weight of each co-occurrence keyword in the topic, compare the comprehensive weights in the given topic category from large to small, and the corresponding topic category with the largest comprehensive weight will be regarded as the result of classification (Table 1).

Through the text classification, the keywords of the new document can be stored in the topic model, so that the features of the topic model can dynamically change. With the continuous expansion of data, this method can replace the traditional supervised text categorization method by using static lexicon or domain knowledge source (e.g. WordNet etc.) as the approach of the knowledge base of classification (Altinel & Ganiz, 2018), (Z. Zhang, Gentile, & Ciravegna, 2011), (Agirre et al., 2009), (Hung & Lin, 2013), (Li, Bandar, & McLean, 2003), (Šuman, Jakupović, & Kuljanac, 2016).

Therefore, the method has the functions of self-updating keywords and self-adjusting classification weights to realize that the keyword topic model can be gradually improved with the input of new data.

3.5. Genetic Algorithm Optimization

As an adaptive search strategy, genetic algorithm is known for its operation similar to the survival of the fittest mechanism in nature. The evolutionary operation of genetic operators (selection, crossover, mutation, etc.) is used to improve the adaptability of individuals and solve various complex problems.

The keyword classification method is a supervised learning method, so optimizing the classification results is great facilitate to improve the performance of the classifier. In this study, genetic algorithm is used to optimize the classification results, and the optimal coefficients are iterated to adjust the comprehensive weight to achieve the global optimal classification performance (Figure 2).

Table 1. The sample result of comprehensive weighted value comparison

Keywords	New Document		Topic	T_1		T_2		T_3		T_4		T_5	
	Degree of Value (w_j) Non-Percentile	Percentile		Non -Percentile /Percentile	Non -Percentile /Percentile	Non -Percentile /Percentile	Non -Percentile /Percentile	Non -Percentile /Percentile	Non -Percentile /Percentile	Non -Percentile /Percentile			
Salah	1.058	17.723%	Composite weighted value $(G_n, j / G'n, j)$	1.052	0.176								
Liverpool	1.058	17.723%		1.087	0.182								
Ramos	1.058	17.723%		1.058	0.177	1.096	0.183			1.127	0.188		
mock	1.058	17.723%						0.147	0.139	1.187	0.210	1.188	0.283
celebration	0.869	14.553%				0.791	0.132			0.887	0.129		
Watford	0.869	14.553%			0.864	0.144							
		100%	Total $(G_n / G'n)$	3.197	0.535	2.751	0.459	0.147	0.139	3.20	0.527	1.188	0.283

The study refers and adopts the parameter setting of the genetic algorithm by (Uğuz, 2011) (Table 2).

In this study, the correctly classified documents are used as training data, and the genetic algorithm is used to iterate the optimal solution as the adjustment coefficient in the specific limited conditions. Among them, a set of solutions with the highest target value of the classification correctness should be selected as the adjustment coefficient (μ_n), and the weights should be adjusted according to the comprehensive weight in each subject category to realize the rearrangement of the comprehensive weight. And the new weight (GA_n) can be obtained after adjusted, which is used to optimize the classification effect:

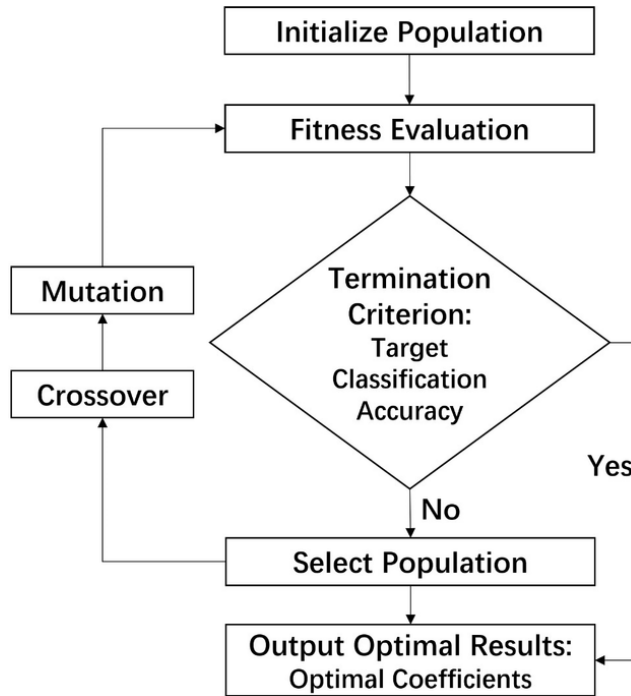
$$GA_n = G_n \times \mu_n \tag{5}$$

Finally, the classification accuracy of the new text could be improved after adjusts the comprehensive weight by the optimal coefficients from GA iteration. And the results of text classification could be evaluated by the F-measure method.

Table 2. Genetic algorithm parameters (Uğuz, 2011)

Parameter Name	Setting
Population size	30
Selection technique	Roulette wheel
Crossover type	2 points crossovers
Crossover rate	0.9
Mutation rate	0.001
Iteration number	500

Figure 2. Genetic algorithm iterative process



4. EXPERIMENTS AND RESULTS ANALYSIS

4.1. Experimental Procedure

In this work, to test the effect of different keyword extraction methods and classifiers, using the scientific literature data to comprehensively evaluate the performance of each method. Eight types of document retrieval in the ACM digital library are adopted, including ACM 1-8 series data files (Table 5). The total number of data files is 401, 411, 424, 394, 471, 439, 471 and 495 respectively, and each file has 5 data sets. The experiment was based on Intel Core i7-4810 MQ Dual 2.80 GHz CPU, 16 GB RAM, 1.5 TB HDD space and Windows 10 OS. Except for the methods mentioned in the study, part of experimental evaluations was conducted using WEKA version 3.7.11 (Witten, Frank, Hall, & Pal, 2016). The toolkit includes a large collection of common machine learning algorithms, and the default parameters are obtained from experience, and they often perform well (Amancio et al., 2014). Therefore, in this study, WEKA default parameters are used as classifier parameters. Meanwhile, the 10-fold cross-validation was used to separate the original dataset into ten mutually exclusive folds. Nine of them are used as training data in turn, and one is used as testing data. Finally, the average correct rate of 10 results is obtained as the accurate estimation of the algorithm. In the study, we also preprocessed the original data set and deleted the noise related to the extraction of keywords (e.g. stop words, stem words etc.).

4.2. Accuracy and Analysis of Category Algorithms and Keyword Extraction Approaches

In order to test the above comprehensive weighted classification method based on genetic algorithm optimization without percentile version (GA-CWC) and its percentile version (GA-CWC(P)), we also use Naive Bayesian, Support Vector Machine, Linear Regression, Random Forest to test

the classification accuracy of ACM document sets using different keyword extraction methods in comparative experiments. These methods include Co-occurrence Statistical Information (Co-SI.), TextRank (TextR.), Most-Frequency (MostFre.), Term Frequency-inverse Document frequency (TF-IDF), Eccentricity-based (Ecc.). In terms of the number of keywords selected, 85 keywords were selected as the number of keywords extracted to test according to the conclusions of (Onan et al., 2016) test.

In the test results list (Table 3), bold is used to represent the best results, while bold and italic are used to represent the second-ranking data. From the analysis of the results, the most accurate method is GA-CWC (P) method based on TextRank (83.91%), followed by Naive Bayesian method based on Most-Frequency (82.67%), the third is GA-CWC method based on TF-IDF (82.58%), the fourth is Random Forest method based on TextRank (82.48%) and the worst is SVM method based on Co-occurrence Information.

4.3. Evaluation and Analysis of the Results of Classification Using Different Keyword Extraction Approaches

F-measure is used as an evaluation index of classification results. From Table 4, F-Measure has the highest GA-CWC method based on TextRank (81.13%), followed by GA-CWC (P) method based on TF-IDF (80.84%) and GA-CWC(P) method based on TextRank (80.37%), The third place is Naive Bayesian method based on Most-Frequency (79.72%) and Linear Regression method based on Co-occurrence Statistical Information (52.86%) is the poorest.

5. DISCUSSION

From the classification accuracy statistics table and the classification F-Measure value statistics table of different methods that the selection of different keyword extraction methods has a greater impact on the classification effect. Such as the Linear Regression classification method, if classified based on the keywords extracted by the Co-occurrence Statistical Information method, the accuracy rate is only 62.80%, but if the keywords are extracted based on TextRank, the classification accuracy rate is increased to 79.98%, and the accuracy rate is increased by about 17.18%. At the same time, in the F-Measure statistics section, the F-Measure value of the Linear Regression algorithm based on the Co-occurrence Statistical Information method for keywords extraction is only 52.86%, but under the keyword extraction based on TextRank, the F-Measure is increased to 77.04%, and the improvement range is about 24.18%. It can be seen that the classification results are influenced by the quality of the keywords extraction through different algorithms, the weight type and distribution of the extracted keywords.

The keyword extraction method based on TextRank achieved the best in the keyword classification task, and the accuracy rate can reach 83.91% by combining with the GA-CWC (P) classification method. Similarly, in the F-Measure statistics, the TextRank-based classification method also reached the highest 81.13% (GA-CWC), obtained the highest value among the comparison methods. In the next place, the keyword extraction method based on Most-Frequency obtained the second-highest accuracy (82.67%) in the classification task combined with the Naive Bayes method is also reached the third-highest F-Measure value (79.72%). It can be seen that in the keyword classification method perspective, the method based on TextRank and Most-Frequent can be used as the most important candidate reference.

Regarding the choice of classification methods, GA-CWC(P) and Naïve Bayes have obtained higher classification accuracy rates based on most keyword extraction methods. However, the average accuracy rate of GA-CWC(P) based on all keyword extraction methods are higher than the other classification methods under the tested keyword extraction methods (average accuracy is about 79.164%), and the GA-CWC(P) method has been obtained the highest average value in the F-Measure evaluation of all classifier under these keyword extraction methods (average F1-Measure is about

Table 3. Classification accuracies of the performance of different keyword extraction algorithms and category approaches

Algorithms	Co-SI.	Ecc.	MostFre.	TF-IDF	TextR.
GA-CWC	63.13	77.35	70.46	74.87	81.13
GA-CWC(P)	62.24	75.79	77.49	80.84	80.37
Naive Bayes	63.67	74.52	79.72	73.68	76.15
SVM	54.41	73.33	72.29	76.35	78.67
Linear Regression	52.86	73.74	74.23	73.51	77.04
Random Forest	55.13	77.91	64.38	63.84	78.19

75.346%). Therefore, GA-CWC(P) has a more accurate classification effect than other comparison methods from a macro perspective.

5.1. Limitation and Further Work

The proposed method has the limitation that it needs a lot of training data to achieve a more accurate classification effect. If the training data is insufficient, it may lead to fewer co-occurrence words and unsatisfactory classification effect. In the follow-up study, we will continue to study how to solve this problem and add more comparative experiments on keyword number selection and model classification effect. Simultaneously, we will also carry out comparative experiments with other machine learning methods.

6. CONCLUSION

This study explores the methods of text categorization using keywords as features and proposes a text categorization method based on automatic keyword extraction with genetic algorithm optimization. Five types of keyword extraction approaches are applied (the most commonly used keyword extraction methods: Co-occurrence Statistical Information, Most-Frequency, Term Frequency-Inverse Sentence Frequency (TF-IDF), Eccentricity-Based Keyword Extraction and TextRank and compared with other four commonly used classification methods, Naive Bayes, Random Forest, SVM, Linear Regression. Experiments show that the proposed methods have obvious advantages in text classification accuracy compared with other common classification methods comparatively, and the highest accuracy and F-Measure of classification has achieved 83.91%, 81.13% respectively in ACM collection. And the

Table 4. F-Measure of the performance of different keyword extraction algorithms and classification approaches

Algorithms	Co-SI.	Ecc.	MostFre.	TF-IDF	TextR.
GA-CWC	64.32	78.42	77.81	82.58	77.96
GA-CWC(P)	69.64	79.48	81.82	80.97	83.91
Naive Bayesian	69.12	80.17	82.67	77.45	81.13
SVM	62.30	76.84	80.43	75.27	78.74
Linear Regression	62.80	78.42	81.39	76.76	79.98
Random Forest	64.68	80.53	82.27	78.12	82.48

method has the attributes of self-updating keywords and self-adjusting weight of classification so that the keyword topic model can be gradually improved with the input of new data. The proposed method has great guiding and practical significance for keyword classification and text classification based on keyword features.

ACKNOWLEDGMENT

This research is supported by VC Research with grant number VCR 0000018.

REFERENCES

- 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. doi:10.1016/j.amc.2014.04.090
- Aggarwal, C. C., & Zhai, C. (2012). A survey of text classification algorithms. In *Mining text data* (pp. 163–222). Springer. doi:10.1007/978-1-4614-3223-4_6
- Agirre, E., Alfonseca, E., Hall, K., Kravalova, J., Paşca, M., & Soroa, A. (2009). *A study on similarity and relatedness using distributional and wordnet-based approaches*. Paper presented at the Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics. doi:10.3115/1620754.1620758
- Al-Thuhli, A., Al-Badawi, M., Baghdadi, Y., & Al-Hamdani, A. (2017). A Framework for Interfacing Unstructured Data Into Business Process From Enterprise Social Networks. *International Journal of Enterprise Information Systems*, 13(4), 15–30. doi:10.4018/IJEIS.2017100102
- Altinel, B., & Ganiz, M. C. (2018). Semantic text classification: A survey of past and recent advances. *Information Processing & Management*, 54(6), 1129–1153. doi:10.1016/j.ipm.2018.08.001
- Amancio, D. R., Comin, C. H., Casanova, D., Travieso, G., Bruno, O. M., Rodrigues, F. A., & da Fontoura Costa, L. (2014). A systematic comparison of supervised classifiers. *PLoS One*, 9(4), e94137. doi:10.1371/journal.pone.0094137 PMID:24763312
- Balcerzak, B., Jaworski, W., & Wierzbicki, A. (2014). Application of TextRank algorithm for credibility assessment. *Proceedings of the 2014 IEEE/WIC/ACM International Joint Conferences on Web Intelligence (WI) and Intelligent Agent Technologies (IAT)*, 1. doi:10.1109/WI-IAT.2014.70
- 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.
- Boughareb, D., & Farah, N. (2013). Identify the User's Information Need Using the Current Search Context. *International Journal of Enterprise Information Systems*, 9(4), 28–42. doi:10.4018/ijeis.2013100103
- Chang, S. E. (2007). an adaptive and voice-enabled Pervasive web System. *International Journal of Enterprise Information Systems*, 3(4), 69–83. doi:10.4018/ijeis.2007100105
- Cohen, J. (2008). Trusses: Cohesive subgraphs for social network analysis. *National Security Agency Technical Report*, 16, 3.1.
- Fernando, A. (2018). *Research paper Classification using Keyword Clustering*. Academic Press.
- Ghnemat, R., & Shaout, A. (2016). Hybrid Fuzzy Neural Search Retrieval System. *International Journal of Enterprise Information Systems*, 12(3), 78–93. doi:10.4018/IJEIS.2016070105
- Hauffa, J., Lichtenberg, T., & Groh, G. (2012). *Towards an NLP-Based Topic Characterization of Social Relations*. Paper presented at the 2012 International Conference on Social Informatics. doi:10.1109/SocialInformatics.2012.80
- Horita, K., Kimura, F., & Maeda, A. (2016). Automatic keyword extraction for wikification of East Asian language documents. *International Journal of Computer Theory and Engineering*, 8(1), 32–35. doi:10.7763/IJCTE.2016.V8.1015
- Hu, J., Li, S., Yao, Y., Yu, L., Yang, G., & Hu, J. (2018). Patent keyword extraction algorithm based on distributed representation for patent classification. *Entropy (Basel, Switzerland)*, 20(2), 104. doi:10.3390/e20020104
- Hung, C., & Lin, H.-K. (2013). Using objective words in SentiWordNet to improve word-of-mouth sentiment classification. *IEEE Intelligent Systems*, 28(2), 47–54. doi:10.1109/MIS.2013.1
- Jain, V. K., Kumar, S., & Mahanti, P. (2018). Sentiment Recognition in Customer Reviews Using Deep Learning. *International Journal of Enterprise Information Systems*, 14(2), 77–86. doi:10.4018/IJEIS.2018040105
- Jansen, B. J., Zhang, M., Sobel, K., & Chowdury, A. (2009). Twitter power: Tweets as electronic word of mouth. *Journal of the American Society for Information Science and Technology*, 60(11), 2169–2188. doi:10.1002/asi.21149

- Joachims, T. (2002). *Learning to classify text using support vector machines* (Vol. 668). Springer Science & Business Media. doi:10.1007/978-1-4615-0907-3
- Kleinberg, J. M. (1999). Authoritative sources in a hyperlinked environment. *Journal of the Association for Computing Machinery*, 46(5), 604–632. doi:10.1145/324133.324140
- Lautenbacher, F., Bauer, B., Sieber, T., & Cabral, A. (2010). linguistics-Based modeling methods and ontologies in requirements engineering. *International Journal of Enterprise Information Systems*, 6(1), 12–28. doi:10.4018/jeis.2010120202
- Lee, S., & Kim, H.-j. (2008). *News keyword extraction for topic tracking*. Paper presented at the 2008 Fourth International Conference on Networked Computing and Advanced Information Management. doi:10.1109/NCM.2008.199
- Li, Y., Bandar, Z. A., & McLean, D. (2003). An approach for measuring semantic similarity between words using multiple information sources. *IEEE Transactions on Knowledge and Data Engineering*, 15(4), 871–882. doi:10.1109/TKDE.2003.1209005
- Liu, J., & Wang, J. (2007). *Keyword extraction using language network*. Paper presented at the 2007 International Conference on Natural Language Processing and Knowledge Engineering. doi:10.1109/NLPKE.2007.4368023
- Meng, Y., & Gao, Y. (2019). Research on Online Reservation Preference of Hotel Consumers Based on Joint Analysis Method. *International Journal of Enterprise Information Systems*, 15(4), 75–86. doi:10.4018/IJEIS.2019100105
- Mihalcea, R., & Tarau, P. (2004). TextRank: Bringing order into text. *Proceedings of the 2004 conference on empirical methods in natural language processing*.
- Onan, A., Korukoğlu, S., & Bulut, H. (2016). Ensemble of keyword extraction methods and classifiers in text classification. *Expert Systems with Applications*, 57, 232–247. doi:10.1016/j.eswa.2016.03.045
- Page, L., Brin, S., Motwani, R., & Winograd, T. (1999). *The PageRank citation ranking: Bringing order to the web*. Academic Press.
- Puri, S., & Singh, S. P. (2019). An Efficient Hindi Text Classification Model Using SVM. In *Computing and Network Sustainability* (pp. 227–237). Springer.
- Rose, S., Engel, D., Cramer, N., & Cowley, W. (2010). Automatic keyword extraction from individual documents. *Text Mining: Applications and Theory*, 1, 1-20.
- 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. doi:10.21528/LNLM-vol12-no1-art2
- Schütze, H., Manning, C. D., & Raghavan, P. (2007). *An introduction to information retrieval*. Cambridge University Press.
- Sivanandam, S., & Deepa, S. (2008). Genetic algorithms. In *Introduction to genetic algorithms* (pp. 15–37). Springer. doi:10.1007/978-3-540-73190-0_2
- Šuman, S., Jakupović, A., & Kuljanac, F. G. (2016). Knowledge-Based Systems for Data Modeling. *International Journal of Enterprise Information Systems*, 12(2), 1–13. doi:10.4018/IJEIS.2016040101
- Tixier, A., Malliaros, F., & Vazirgiannis, M. (2016). A graph degeneracy-based approach to keyword extraction. *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*. doi:10.18653/v1/D16-1191
- Uğuz, H. (2011). A two-stage feature selection method for text categorization by using information gain, principal component analysis and genetic algorithm. *Knowledge-Based Systems*, 24(7), 1024–1032. doi:10.1016/j.knsys.2011.04.014
- Witten, I. H., Frank, E., Hall, M. A., & Pal, C. J. (2016). *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann.

Wu, W., Zhang, B., & Ostendorf, M. (2010). *Automatic generation of personalized annotation tags for twitter users*. Paper presented at the Human language technologies: The 2010 annual conference of the North American chapter of the association for computational linguistics.

Wu, Y.-C. (2015). Chinese Text Categorization via Bottom-Up Weighted Word Clustering. *International Journal of Enterprise Information Systems*, 11(1), 50–61. doi:10.4018/ijeis.2015010104

Zhang, M., Wang, J., & Liu, H. (2017). The probabilistic profitable tour problem. *International Journal of Enterprise Information Systems*, 13(3), 51–64. doi:10.4018/IJEIS.2017070104

Zhang, Z., Gentile, A. L., & Ciravegna, F. (2011). Harnessing different knowledge sources to measure semantic relatedness under a uniform model. *Proceedings of the Conference on Empirical Methods in Natural Language Processing*.

Zhao, M., Yu, W., Lu, W., Liu, Q., & Li, J. (2016). *Chinese Document Keyword Extraction Algorithm Based on FP-growth*. Paper presented at the 2016 International Conference on Smart City and Systems Engineering (ICSCSE). doi:10.1109/ICSCSE.2016.0062

APPENDIX

Table 5. An overview of ACM collection dataset (Rossi et al., 2014)

Doc. Name	Class	Docs	Doc. Name	Class	Docs
ACM(1)	3 dimensional technologies	91	ACM(5)	Tangible and Embedded interaction	81
	Visualization	72		Management of data	96
	Wireless mobile multimedia	82		User interface software and technology	104
	Solid and physical modeling	74		Information technology education	87
	Software engineering	82		Theory of computing	103
ACM(2)	Rationality and knowledge	86	ACM(6)	Computational geometry	89
	Simulation	84		Access control models and technologies	90
	Software reusability	72		Computational molecular biology	71
	Virtual reality	83		Parallel programming	96
	Web intelligence	86		Integrated circuits and system design	93
ACM(3)	Computer architecture education	78	ACM(7)	Database systems	104
	Networking and Communications systems	75		Declarative programming	101
	Privacy in the electronic society	98		Parallel and Distributed simulation	98
	Software and performance	81		Mobile systems Applications and services	95
	Web information and data management	92		Network and system support for games	73
ACM(4)	Embedded networked sensor systems	50	ACM(8)	Mobile ad hoc networking and computing	90
	Information retrieval	71		Knowledge discovery and data mining	105
	Parallel algorithms and architectures	98		Embedded systems	102
	Volume visualization	104		Hypertext and hypermedia	93
	Web accessibility	71		Microarchitecture	105

Pin Ni received the MRes degree from the University of Liverpool, UK in 2020. He is the co-founder of Research Lab for Knowledge and Wisdom (KnoWis), Xi'an Jiaotong-Liverpool University, China. His current research interests include Natural Language Processing, Knowledge Graph, Semantic Web, Data Mining and Knowledge Discovery. He will pursue his PhD at the University of Auckland, New Zealand.

Yuming Li received the MRes degree from the University of Liverpool, UK in 2020. She is the co-founder and member of Research Lab for Knowledge and Wisdom (KnoWis), Xi'an Jiaotong-Liverpool University, China. Her current research interests include Natural Language Processing, Information System, Knowledge Graph, Deep learning in the financial field, etc. She will pursue his PhD at the University of Auckland, New Zealand.

Victor Chang is a Professor of Data Science and Information Systems, SCEdT, Teesside University, UK, since September 2019. He leads the Beneficial Artificial Intelligence (BAI) Research Group at Teesside. Previously he was a Senior Associate Professor, Director of Ph.D. (June 2016–May 2018) and Director of MRes (Sep 2017–Feb 2019) at IBSS, Xi'an Jiaotong-Liverpool University (XJTLU), Suzhou, China. Before that, he worked as a Senior Lecturer at Leeds Beckett University, UK, for 3.5 years. Within 4 years, he completed Ph.D. (CS, Southampton) and PGCert (Higher Education, Fellow, Greenwich) while working for several projects at the same time. Before becoming an academic, he has achieved 97% on average in 27 IT certifications. He won numerous awards since 2012. He is a visiting scholar/Ph.D. examiner at several universities, an Editor-in-Chief of IJOCl & OJBD journals, former Editor of FGCS, Editor of Information Fusion, Associate Editor of TII and founding chair of two international workshops. He is a founding Conference Chair of IoTBDs <http://www.iotbd.org> and COMPLEXIS <http://www.complexis.org> since Year 2016, FEMIB <http://femib.scitevents.org> Year 2019 and IIoTBDSC <http://iiotbdsc.com> since Year 2020. He was involved in different projects worth more than £13 million in Europe and Asia. He has published 3 books as sole authors and the editor of 2 books on Cloud Computing and related technologies. He gave 18 keynotes at international conferences.