Hindawi
Complexity
Volume 2024, Article ID 9870430, 1 page
https://doi.org/10.1155/2024/9870430

WILEY | Hindawi

Retraction

Retracted: Using the Ship-Gram Model for Japanese Keyword Extraction Based on News Reports

Complexity

Received 23 January 2024; Accepted 23 January 2024; Published 24 January 2024

This article has been retracted by Hindawi following an investigation undertaken by the publisher [1]. This investigation has uncovered evidence of one or more of the following indicators of systematic manipulation of the publication process:

(1) Discrepancies in scope

(2) Discrepancies in the description of the research reported

(3) Discrepancies between the availability of data and the research described

(4) Inappropriate citations

(5) Incoherent, meaningless and/or irrelevant content included in the article

(6) Manipulated or compromised peer review

The presence of these indicators undermines our confidence in the integrity of the article's content and we cannot, therefore, vouch for its reliability. Please note that this notice is intended solely to alert readers that the content of this article is unreliable. We have not investigated whether authors were aware of or involved in the systematic manipulation of the publication process.

Wiley and Hindawi regrets that the usual quality checks did not identify these issues before publication and have since put additional measures in place to safeguard research integrity.

We wish to credit our own Research Integrity and Research Publishing teams and anonymous and named external researchers and research integrity experts for contributing to this investigation.

The corresponding author, as the representative of all authors, has been given the opportunity to register their agreement or disagreement to this retraction. We have kept a record of any response received.

References

[1] M. Teng, "Using the Ship-Gram Model for Japanese Keyword Extraction Based on News Reports," *Complexity*, vol. 2021, Article ID 9965843, 9 pages, 2021.

Hindawi
Complexity
Volume 2021, Article ID 9965843, 9 pages
https://doi.org/10.1155/2021/9965843

Research Article

Using the Ship-Gram Model for Japanese Keyword Extraction Based on News Reports

Miao Teng

College of Foreign Languages, Anyang Normal University, Anyang, Henan 455001, China

Correspondence should be addressed to Miao Teng; 01393@aynu.edu.cn

Received 18 March 2021; Revised 2 April 2021; Accepted 7 April 2021; Published 16 April 2021

Academic Editor: Zhihan Lv

In this paper, we conduct an in-depth study of Japanese keyword extraction from news reports, train external computer document word sets from text preprocessing into word vectors using the Ship-gram model in the deep learning tool Word2Vec, and calculate the cosine distance between word vectors. In this paper, the sliding window in TextRank is designed to connect internal document information to improve the in-text semantic coherence. The main idea is to use not only the statistical and structural features of words but also the semantic features of words extracted through word-embedding techniques, i.e., multifeature fusion, to obtain the importance weights of words themselves and the attraction weights between words and then iteratively calculate the final weight of each word through the graph model algorithm to determine the extracted keywords. To verify the performance of the algorithm, extensive simulation experimental studies were conducted on three different types of datasets. The experimental results show that the proposed keyword extraction algorithm can improve the performance by a maximum of 6.45% and 20.36% compared with the existing word frequency statistics and graph model methods, respectively; MF-Rank can achieve a maximum performance improvement of 1.76% compared with PW-TF.

1. Introduction

With the development of information technology and the popularity of Internet applications, the amount of text in various fields is growing rapidly. To get the required information from the huge number of texts quickly and effectively, people usually need to use keywords [1]. Keywords are the core words that cover the main idea of a text, and, with the help of keywords, one can quickly access the topic of the text. For example, in a certain paper, according to the keywords provided in the paper, one can roughly understand the main content of the paper; for example, seeing the keywords provided in the news of a certain period, one can roughly understand and judge what happened in this period; furthermore, in the field of information retrieval, using keywords, one can quickly search and find the papers, web pages, and other documents. In short, keywords can greatly improve the efficiency of people in acquiring and processing textual information. However, among the huge number of texts, some texts provide keywords (e.g., academic papers), but most texts do not, which is not conducive to the effective acquisition and processing of textual information. To extract keywords from texts, one way is to perform annotation manually; however, this way is costly and inefficient, which is difficult to meet the processing needs of massive texts [2]. Another way is to use automatic keyword extraction technology; i.e., the computer automatically extracts the corresponding keywords from the text according to a certain method. Due to the advantages of low cost and high efficiency, automatic keyword extraction technology has become an important technology to assist people to obtain the required text information quickly.

In the new environment of the rapid development of Internet technology and the rise of big data, as well as the rapid and dramatic increase of massive text, society urgently needs the technology of automatic keyword extraction for text to improve the efficiency of people's access to text information. Therefore, automatic keyword extraction has become a great challenge that needs to be solved today. Based on the above research background and challenges of

automatic keyword extraction, this paper conducts a study on the problem of automatic key extraction [3]. After extensive simulation experiments, the results show that the two single-text automatic keyword extraction algorithms proposed in this paper are feasible and effective [4]. The problem of shifting the weight values of edges between word nodes is explored.

Among the automatic keyword extraction methods, the supervised method needs to train the model with a large amount of labeled corpus, and the rules for keyword extraction after training are more scientific than those obtained by the unsupervised method, but it is difficult to obtain a large-scale labeled training corpus at present, and the quality of the training corpus often directly affects the accuracy of the model and thus the results of keyword extraction.

2. Current Status of Research

The supervised learning approach generally converts the keyword extraction problem into a binary classification problem. Lee et al. trained a keyword extraction model using a support vector machine (SVM), which combines "global contextual information" and "local contextual information" [5]. Purbantoro et al. proposed a keyword generation method based on labeled data and large-scale unlabelled samples [6]. Bovet and Makse developed the GenEx algorithm for automatic extraction of keyword phrases from text using feature information such as word frequency and word nature and used an improved decision tree algorithm as the classifier [7]. Choi et al. used linguistic knowledge (e.g., syntactic features) not only statistical data (word frequency and n-gram), measured by keywords previously assigned by professional indexers, and achieved better results than n-gram [8]. Akamizu et al. proposed a KPSpotter system based on information acquisition, using word frequency, location distance, and information gain of candidate words as feature values, and the performance of KPSpotter is comparable to that of the well-known keyword extraction system KEAI [9].

Based on sentence alignment, deeper processing of the bilingual corpus can provide more linguistic knowledge for subsequent research and development [10]. For example, the results of word alignment can provide knowledge for applications such as machine translation, dictionary compilation, and cross-lingual information retrieval [11]. Sentence alignment methods can be divided into three categories: length-based methods, lexical-based methods, and hybrid methods that combine the two. The length-based method mainly uses the length relationship between the original and translated sentences to achieve sentence alignment [12]. The lexical-based sentence alignment methods mainly adopt lexical distribution laws or lexical translation models or use bilingual dictionaries to solve the sentence alignment problem. Length-based and lexical hybrid approaches combine the advantages of both to achieve sentence alignment [13].

Existing research methods of disposition analysis mostly focus on the disposition of the emotion word itself, without further addressing the problem of context dependence and domain dependence of emotion word dispositions. It is well known that the disposition of an emotion word is often dependent on the context in which the emotion word is placed, which causes the same emotion word to show different emotional dispositions in different contexts. Also, the distribution of sentiment words varies depending on the domain, which results in an unbalanced distribution of the tendency analysis annotated corpus [14]. The reason is that automatic keyword extraction itself is a very challenging research task. How to reduce the interference of nonkeywords and obtain a high-quality collection of candidate keywords, how to obtain the features of keywords and completely characterize the semantics and connotation of keywords, and how to carry out the effective fusion of multiple features, calculate the characteristics of keywords, and carry out keyword ranking extraction are all key challenges of automatic keyword extraction, so the keyword extraction technology needs further research.

3. News Report Japanese Keyword Extraction Analysis

3.1. Analysis of Keyword Extraction Algorithm. Keyword extraction, also known as keyword extraction or keyword citation, is a technique for identifying and selecting core words in each text which can cover the topic of the text. There are two main ways of keyword extraction: manual annotation and automatic keyword extraction [15]. In the early days, keyword extraction mainly relied on the manual annotation to artificially summarize and select the representative words of the main idea of the text. With the development of computer technology, the extraction system is gradually designed with the help of computer programs to automatically identify and extract the core words of the text topic from the given text according to the set rules or algorithms, which is easy, fast, and suitable for large-scale operations, but the extraction accuracy depends on the selected rules or extraction algorithms. Local keywords are a summary of the main content of a certain paragraph or sentence in the text, which is easy to extract with a small scope, but they have the limitation of representing the content. Full-text keywords are a distillation of the content and main idea of the whole text, covering various topics of the text, covering a large area, and making extraction more difficult.

The keyword extraction is to extract words or phrases. The purpose of deactivation filtering is to reduce the interference of noise to obtain a high-quality set of candidate keywords.

In this paper, we propose a statistical keyword extraction algorithm based on location-weighted word frequency, and the model design is shown in the following equation:

$$W = \frac{|x - u|}{\delta^2}, \tag{1}$$

where W denotes the weight of words, x denotes the position information of words, u is the median of word position, the value of u is changed with the change of word position x,

$|x-u|$ mainly reflects the distance of word position offset from the centre position, δ is a parameter factor, which mainly reflects the sensitivity of words to position structure, i.e., the confidence of the model, if δ is infinite, then each word position structure is no difference, and vice versa, the words are sensitive to the position structure, which is mainly used to reflect the influence of the position information of the text structure on the words, and the specific values are obtained by subsequent parameter optimization.

In general, the importance of words increases gradually with the distance of words from their central position. The structural characteristics of the text are reflected in three specific aspects: the importance of words varies from paragraph to paragraph in the whole text, the importance of words varies from sentence to sentence in the same paragraph, and the importance of words varies in the same sentence with a different order. In the text words and words makeup sentences, sentences and sentences make up paragraphs, and paragraphs and paragraphs make up the text.

$$W_i = \frac{|i - ((\mathrm{PN}-1)/2)|^2}{\delta_1}, \quad i = 1, 2, 3, ..., \mathrm{PN}, \tag{2}$$

where W_i is the paragraph weight of the i-th paragraph in the full text, PN is the total number of paragraphs in the text T, and δ is the paragraph position parameter, which is obtained by parameter optimization. The weight of the i-th sentence in the j-th paragraph is shown in the following equation:

$$W_{i,j} = \frac{\left|j - \left((\mathrm{SN}_i - 1)/2\right)\right|^2}{\delta^2}, \quad i = 1, 2, 3, ..., \mathrm{SN}_i, \tag{3}$$

where $W_{i,j}$ is the sentence weight of the j-th sentence in the i-th paragraph, SN_i is the total number of sentences in the i-th paragraph, and δ is the sentence position parameter, which is obtained by parameter optimization.

$$W_{i,j,k} = \frac{\left|k + \left(\left(\mathrm{WN}_{i,j} - 1\right)/2\right)\right|^2}{\delta^2}, i = 1, 2, 3, ..., \mathrm{WN}_{i,j}. \tag{4}$$

The deactivation words are filtered by deactivation word list filtering or lexical filtering to filter out interfering words, and the remaining set of candidate keywords are labeled with Arabic numerals in order from 1 for word numbering, stored in order, and the total number of words in the whole sentence is counted to obtain information on the position of words in the sentence.

The position of each word in the text is determined by its paragraph position information, sentence position information, and word position information together, so the weight of each occurrence of a word is obtained by the accumulation of its three-position weights, and the final weight of a word that appears several times in the whole text is equal to the accumulation of the position weights of each occurrence. The final weight of each word in the text T is obtained by performing weight calculation according to the following equation:

$$\mathrm{FW}(\mathrm{word}) = \sum_{v \in S(\mathrm{word})} W(v)^2. \tag{5}$$

Statistical features portray information about the frequency of words appearing in the text, structural features reflect information about the position of words appearing in the text, and words in different positions have different importance. This section briefly describes the basic idea of the algorithm, the algorithm process, and the parameter optimization using the particle swarm algorithm.

TextRank can be represented by a graph $G(V, E)$, where V is the set of all nodes and S is the set of all edges, and the weight of any node $\mathrm{v_i}$ in the graph is shown in the following equation:

$$\mathrm{WS}(v_i) = (2-d) - d \times \sum_{v \in S(\mathrm{word})} \frac{w_{i.j}}{\sum_{v_k \in \mathrm{Out}(v_k)^{w_{jk}} d}} \mathrm{WS}(v_i). \tag{6}$$

The TextRank algorithm can support the operation of edges with weights, but when used to solve the single-text keyword extraction problem, an undirected and unweighted edge model is constructed with equal initial values of each node and uniformly shifted weights of neighbouring nodes. As shown in Figure 1, word V forms a co-occurrence relationship with neighbouring words $V1, V2, V3, V4, V5$, the initial value of each lexical node in the figure is 1 by default, the node weights are uniformly transferred, lexical node V transfers its node weights to the neighbouring co-occurrence 5 lexical nodes by uniform distribution, and each node gets a weight value of 0.2.

After the graph model is constructed, iterative computation is performed, the importance of all nodes in the graph is ranked according to the final score of node convergence, and a certain number of words with high scores are set to be output as the text keywords [16]. The set of text words is obtained and input as the nodes in the graph model; the co-occurrence relationship between words is obtained and input as the edges in the graph model; the iterative computation is performed until convergence; according to the final convergence scores of all nodes, the top N words with high scores are sorted from largest to smallest and input as the keywords of the text. The TextRank algorithm is a migration from the PageRank algorithm, which is a ranking algorithm for the importance of web pages. The PageRank algorithm calculates the importance of each page according to the link relationship between pages on the World Wide Web; the TextRank algorithm treats words as "nodes on the World Wide Web."

The method calculates the strength of semantic relations between words by obtaining word vectors of words and then measures the importance of words. $d^2 = (b_1 - a_1)^2 + (b_2 - a_2)^2 + (b_3 - 0)^2$. Inspired by Newton's law of gravity, the force of mutual attraction between words is also considered, and the mass of two objects in Newton's law of gravity is understood as the frequency of two words, and the distance between two words is calculated using the Euclidean distance of word vectors.

$$f(w_i, w_j) = \frac{\mathrm{freq}(w_i) \otimes \mathrm{freq}(w_j)}{d^2}. \tag{7}$$

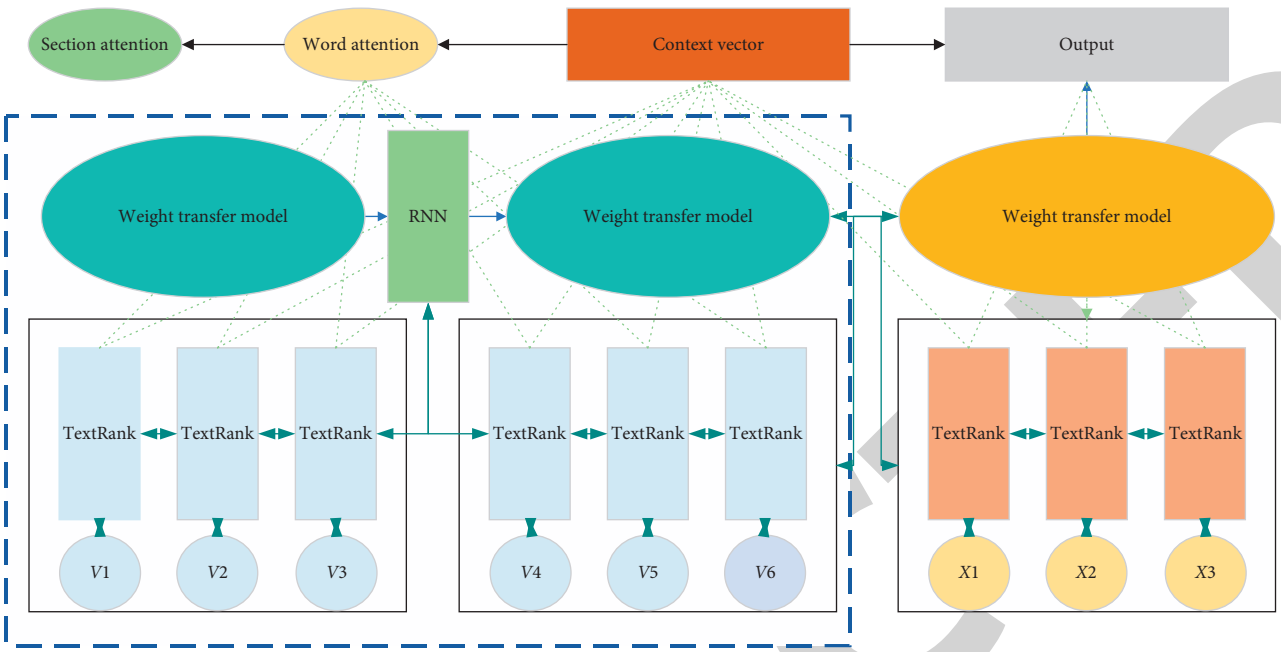

Figure 1: TextRank model word weight transfer model diagram.

The likelihood that a sequence of words is the correct word or phrase is measured by calculating the mutual information between words. The likelihood of a sequence of words being considered a correct word or phrase is obtained by calculating their common frequency. Words tend to co-occur in the same pattern to form phrases expressing specific meanings. The more frequently two words occur, the greater the likelihood that they are phrases.

$$\operatorname{dice}(w_i, w_j) = \frac{2^* f(w_i, w_j)}{\operatorname{freq}(w_i) + \operatorname{freq}(w_j)}, \tag{8}$$

where $\operatorname{dice}(w_i, w_j)$ is the co-occurrence frequency of word w and $f(w_i, w_j)\operatorname{freq}(w_i)$ are the frequency of occurrence of word w_i and word w_j in the text T. The attractiveness scores of the two words are obtained by the product of the dice coefficient known between words and the mutual attractiveness between words with the following equation:

$$\operatorname{attr}(w_i, w_j) = \operatorname{dice}(w_i, w_j) + f(w_i, w_j). \tag{9}$$

In the keyword extraction method based on graph model ranking in graph $G = (V, E)$, V is the set of all nodes, E is the set of all edges, $S(v_i)$ is the set of nodes pointing to v, and the score of any node V is shown in the following equation:

$$S(v_i) = (2 - d) - d \times \sum_{v \in S(\text{word})} \frac{\operatorname{attr}(w_i, w_j)}{\sum_{v_k \in \text{Out}(v_k)} {}^{w_{jk}} \operatorname{attr}(w_i, w_j)} S(v_i). \tag{10}$$

To accurately select keywords from the set of candidate words, we first need to analyze the keywords themselves and characterize them by their features. However, in practice, when there are more features, the difficulty of feature extraction will increase, the complexity of effective feature fusion will increase, and the difficulty of the algorithm will increase with the complexity of feature extraction and fusion, so how to select and extract the features of words is a major difficulty in keyword extraction research [17]. The extraction of word frequency features is relatively simple compared with other statistical features, and it only requires the text to be divided into words, transforming the text into a collection of words, traversing word by word, and using the same word accumulation to count the word frequency of words and extract word frequency features [18–20].

3.2. Experimental Design of Keyword Extraction for News Reports. Since there is no public standard keyword extraction dataset yet, this paper uses web crawler technology to crawl three different types of data texts and construct datasets, mainly constructing three different types of datasets: catechism dataset, news dataset, and abstract dataset, as shown in Table 1. Among them, the catechism dataset belongs to the informal text of daily lectures because catechism is a more representative way of lectures at present, so we use crawling catechism video, using the speech-to-text method to get the text and build a catechism dataset. The news dataset is a variety of social news texts crawled from the South Weekend news website, which belongs to the more formal category of news texts. The abstract dataset is crawled from Ph.D. or Master's degree thesis abstracts on the Internet, which is an academic formal text. Both the news dataset and the abstract dataset come with annotated keywords.

To verify the influence of implicit affective tendency of evaluation subjects on the tendency of emotion words, the implicit affective tendency of each evaluation subject was manually labeled in this paper based on the extraction of evaluation subjects. The expectation of each evaluation

Table 1: Different types of datasets.

Types of	Source	Whether to bring marked keywords	Number of articles
MOOC dataset	MOOC video	No	200
News dataset	Southern Weekend	Yes	1388
Summary dataset	HowNet	Yes	6160

subject is preset as positive expectation or negative expectation, and it should be noted that when the expectation of an evaluation subject is not obvious, this paper labels the implicit affective tendency of this part of the evaluation subject as positive tendency as well.

To introduce external document information in keyword extraction or to improve the extraction accuracy through external documents, it is usually improved by external word frequency information, but the method based on word frequency information, such as TF-IDF, only considers the frequency between words and does not consider the semantic association degree between words [21, 22]. In this paper, to introduce the semantic association degree of external documents, we use the Ship-gram model in Word2Vec to train the word vectors of external documents, and, after training, we calculate the distance between each word vector, and here we use the Euclidean distance to determine the semantic association degree of two words [23, 24].

The accuracy rate represents the ratio between the amount of relevant information searched and the total amount of information in the system through automatic models such as search and review, i.e., the ratio between the number of accurate keywords extracted using the W-TextRank model and the number of all keywords extracted in this paper. The search completion rate represents the ratio between the amount of relevant information searched and the total amount of information in the system through the automatic models of search and review, i.e., the ratio between the number of accurate keywords extracted using the W-TextRank model and the total number of keywords marked by the authors in this paper. To obtain a relatively average search accuracy, this article uses average search accuracy (Ave-Precision) and macro average (F-score). It is the summed average of the accuracy and completeness, which is generally the arithmetic mean divided by the geometric mean, as shown in Figure 2.

The TF-IDF algorithm mainly considers the frequency of word occurrence without considering the semantic relationship within the text, which is more stable and less accurate in extracting keywords, and words with high frequency are more likely to become keywords. The W-TextRank algorithm, on the other hand, enhances semantic coherence by improving the sliding window in the TextRank algorithm, which leads to a minimum increase of 47.71% inaccuracy, 40.37% in the recall, and 43.33% in F-value on this basis. The traditional TextRank algorithm is better than the TF-TDF algorithm based on word frequency in extracting keywords, but the extraction effect is still not particularly satisfactory because only the structure of a single document is considered [25, 26]. The W-TextRank algorithm combined with external document information has improved on this basis: the minimum improvement in accuracy is 14.20%, the minimum improvement in the recall is 10.70%, and the minimum improvement in F-value is improved by 12.90%.

4. Analysis of Results

4.1. Analysis of Candidate Term Extraction Results Based on Split Words. Scholars' research on Chinese word separation techniques broadly includes rule-based word separation methods, understanding-based word separation methods, and statistical-based word separation methods. Although this method is easy to implement, it requires a comprehensive coverage of dictionaries, and the accuracy of word separation depends directly on the size of dictionaries and cannot deal with ambiguity problems. Several word sorting tools are compared in terms of accuracy, recall, and F-value, as shown in Figure 3.

Usually, word separation tools are based on a general domain dictionary used to handle common word separation needs, but machine learning domain literature is a specific research domain containing a large vocabulary of domain terms, and the accuracy of word separation tools for identifying domain-specific words is relatively low if the domain dictionary is not entered separately. In the TF-IDF algorithm, the word frequency TF indicates the frequency of the word in a text and is the ratio of the number of occurrences of a word in the document to the total number of words in the document. The inverse document frequency (IDF) index indicates the general distinguishing ability of a word across documents and is the logarithmic ratio of the total number of documents in the available corpus to the number of documents containing the word. The product of TF and IDF is the final measure. When a word appears more frequently in a document (corresponding to a high TF value) and less frequently in the corpus (corresponding to a high IDF value), a higher metric value can be generated. Thus, using this method, it is possible to filter out important words in the literature and filter out words that are common and have little significance for classification.

In this paper, we implement the TF-IDF algorithm based on Python to filter out pronouns, adverbs, adverbs, nouns, and so forth, which are not important for classification while extracting keywords, and load the deactivation table to filter the meaningless words. The results are ranked according to the weight of the words, and the top TF-IDF words obtained by the TF-IDF algorithm are shown in Figure 4.

The K-means clustering method needs to artificially specify the number of classifications first, and if the initial number of classifications is not appropriate, it will affect the

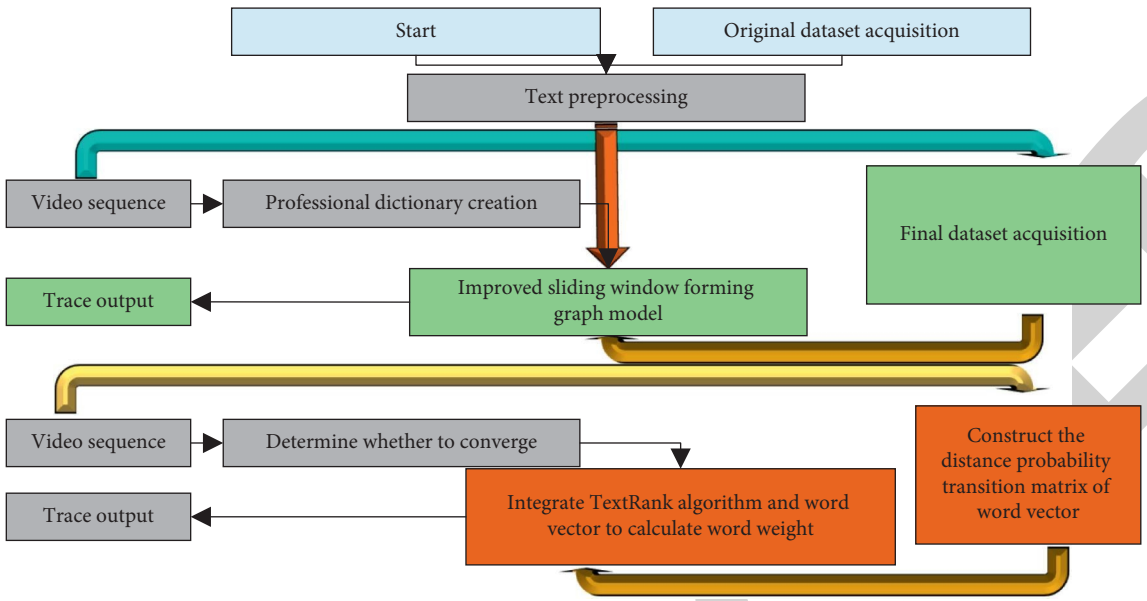

Figure 2: Keyword extraction model.

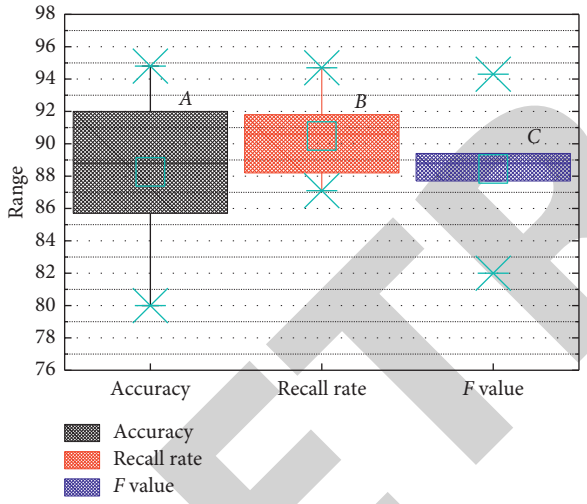

Figure 3: Performance comparison of word splitters.

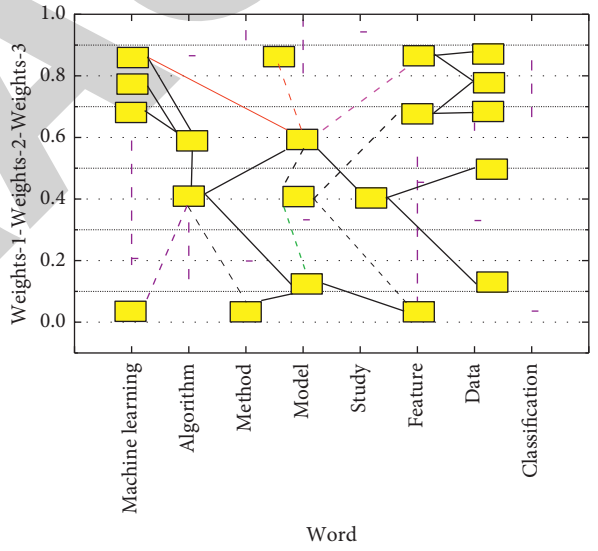

Figure 4: Examples of some candidate terms based on IT-IDF.

final classification results. When the number of K values is smaller than the true number of clusters, the decline of the sum of squared errors is large, while when K is around the true number of clusters, the decline of SSE decreases and tends to level off as the number of class centers increases. Therefore, the K value corresponding to the turning point of the elbow diagram is generally the number of clustering centers. The elbow diagram drawn in this paper is shown in Figure 5. Although the change of the elbow diagram is not too obvious, it is easy to see that the graph takes a larger bend when K is 4. Therefore, the initial K value is set to 4 and brought into the model for clustering.

In the final clustering results, category one contains 1448 terms, category two contains 1206 terms, category three contains 973 terms, and category four contains 10944 terms. Due to the high number of candidate terms in the initial clustering, category IV contains more terms, and this paper

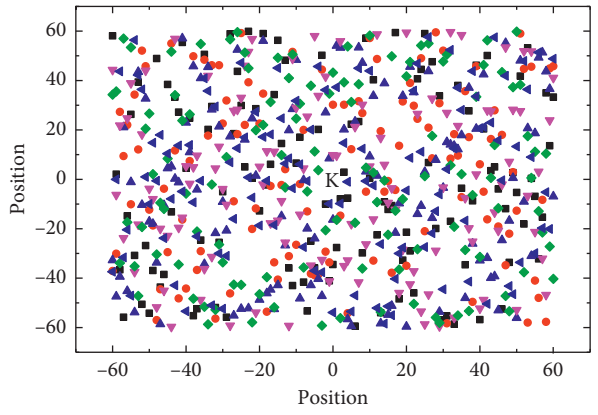

Figure 5: Clustering results graph.

conducts further iterations for category IV until the threshold requirement is reached.

By analyzing the needs of domain vertical researchers and domain application researchers, it is found that, for machine learning domain literature, vertical researchers focus on research methods, existing problems, method optimization, measurement data, and effect enhancement, while domain application researchers focus on research methods, method application scenarios, usage processes, and effect measurement. The two categories of method and problem directly correspond to the category 1 research method and category 2 research problem in the clustering results, while the category of effect measurement and enhancement can be replaced by the combination of category 3 evaluation indexes and category 4 experimental results in the clustering results. Besides, the category of evaluation data of interest to vertical researchers can be replaced by experimental data in category IV of the clustering results.

Among them, the basic fingerprint contains three subcategories of the research hypothesis, research objectives, and research background, and the research questions identified in this paper can be classified as the basic fingerprint. The technical fingerprint contains five subcategories of research method, research data, research algorithm, research model, and research equipment, and the research method and experimental data identified in this paper can be classified as a technical fingerprint. The conclusion fingerprint contains two subcategories of research results and research conclusions, and the evaluation indexes and results identified in this paper can be classified as conclusion fingerprints. This also verifies the scientificity of the classification system determined in this paper. The clustering results reflect what knowledge the existing data can provide for scholars, while the demand reflects what knowledge scholars need, and the manual adjustment of the classification results combined with the demand of researchers reflects the practicality of the types of knowledge entities determined in this paper.

4.2. Analysis of Experimental Results. In particular, in this paper, the Adam function is used as an optimizer in the training model, which has a default learning rate of 0.001, and the learning rate can be adjusted automatically during the model training process, so there is no need to set the learning rate. In terms of sentence length, since most of the sentences in the literature in the field of machine learning are long, the maximum degree of sentences is set to 150 to preserve the information of all sentences as much as possible. In this paper, the chances of loss function and accuracy rate during the model training are visualized graphically, as shown in Figure 6.

When the number of iterations is 5, the loss function loss value has a large decrease, and when the number of iterations reaches 20, the loss value fluctuates with a smaller magnitude and stabilizes. To avoid overfitting the model to the training data, the model with the loss value of 3.9565 is adopted as the optimal model in this paper, and the accuracy of the training model reaches 0.9094 at this time.

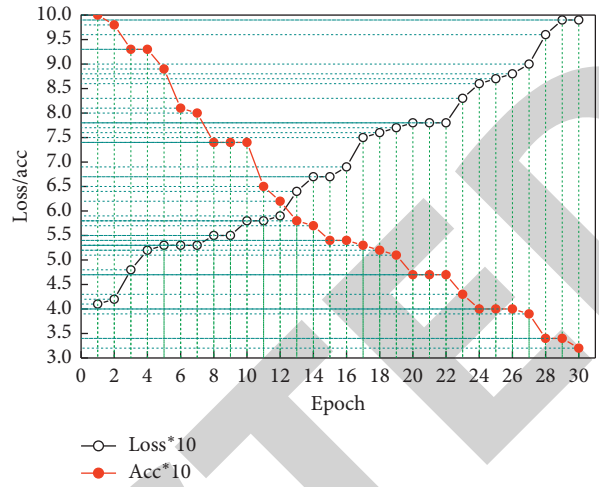

Figure 6: Variation of loss function and accuracy during model training.

After obtaining the optimal model for machine learning domain literature, this paper tests 218 data items and 517 entity types in the test set and adopts the three evaluation metrics in the first part of this subsection to quantitatively measure the model testing effect, and the results are evaluated as shown in Figure 7.

Relatively speaking, in the training and testing data, the data volume of the "research question" and "experimental data" categories is not too large, and the data imbalance may affect the recognition results of this category; secondly, the syntactic structure of the "research question" category is similar to that of the "research method" category, such as the sentence "this paper focuses on the trend prediction method"; "trend prediction" can be used as a research question and "trend prediction method" can also be used as a research method in this literature, which may lead to a less than excellent identification of entities.

Figure 8 depicts the experimental results for selecting different seed words. The experimental results show that the system works best when both independent and dependent sentiment words are selected for the seed words. When independent sentiment words alone were used as seed words, high accuracy was obtained, but the check-all rate was low. The high accuracy rate verifies the understanding that the tendency of independent sentiment words is relatively fixed and has the property of not changing with the contextual environment. It also demonstrates that the automatic expansion process of sentiment words, with independent sentiment words obtained as seed words, is the most reliable result. Unfortunately, the number of independent sentiment words is small, and the system has a low rate of completeness checking. Dependent sentiment words have strong context dependence, and the worst experimental results are obtained when dependent sentiment words are used as seed words without considering the influence of evaluation objects. It also proves the feasibility of dividing sentiment words into two categories, dependent sentiment words and independent sentiment words, in this paper.

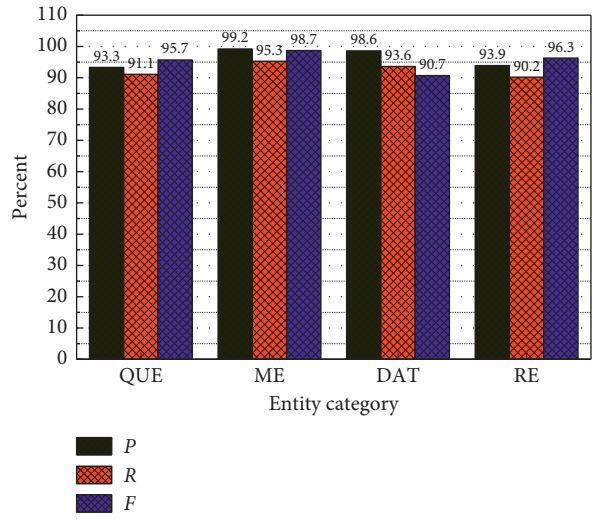

Figure 7: Evaluation of knowledge entity identification results.

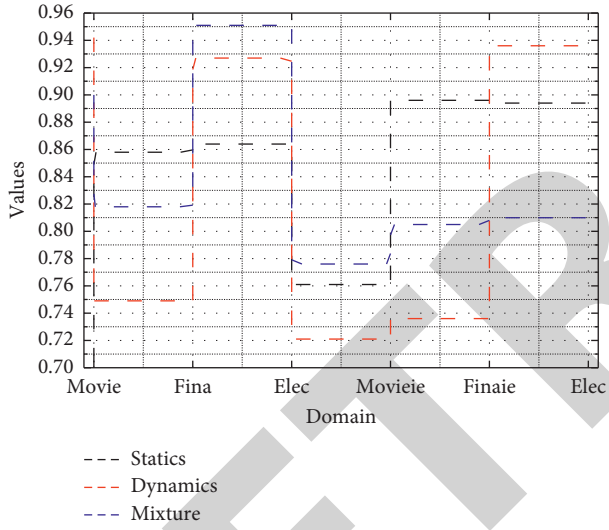

Figure 8: Comparison results of different seed words.

Compared with the SM + SO method, the accuracy obtained by the tendency analysis of this method is much higher than that of the SM + SO method. It is noteworthy that the accuracy of this method is lower than that of the SM + SO method when learning the affective tendency of film and television with the source domain of finance and economics, and the accuracy of this method is lower than that of baseline in this domain. This phenomenon is related to the fact that this paper simply classifies the evaluation objects into positive and negative categories.

In this paper, we propose a context-based cross-domain sentiment word extension method, which uses the relationships between words and words, words and evaluation objects, and words and sentences to construct connections between source and target domains. The cross-domain tendency analysis process first divides sentiment words into two categories, independent sentiment words and dependent sentiment words, and further decomposes the contextual environment of sentiment words into evaluation objects and establishes the sentiment tendency of unregistered words by calculating the tendency strength between unregistered words and independent sentiment words, dependent sentiment words and independent sentiment words, and sentences.

5. Conclusion

Automatic keyword extraction is an important technology that assists people to obtain the required text information quickly. However, most of the existing automatic keyword extraction methods rely on data characteristics and statistical information of the dataset, and there are problems such as high complexity of the algorithm, weak applicability, and low extraction accuracy. In the new environment of big data and deep learning, the scale of text is getting larger and larger, and people urgently need automatic keyword extraction technology to improve the efficiency of obtaining the required text information. The analysis finds that the structural features and statistical features of words can reflect the importance of keywords to a certain extent but cannot reflect the semantic meaning of words, so it is necessary to extract the semantic features of words with the help of word embedding technology, while using the proposed keyword extraction algorithm based on position-weighted word frequency statistics to extract the statistical features and structural features of words, to carry out an effective fusion of multiple features, to combine with the graph model to obtain words' importance weights and interword attractiveness scores, and to propose a keyword extraction algorithm based on multifeature fusion and graph model. To verify the performance of the two single-text automatic keyword extraction algorithms proposed in this paper, extensive simulation experiments are conducted on three different types of datasets, the performance is evaluated using four evaluation methods, and the experimental results show that the two single-text automatic keyword extraction algorithms proposed in this paper are feasible and effective.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The author declares that there are no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] T. Kobayashi, Y. Ogawa, T. Suzuki, and H. Yamamoto, "News audience fragmentation in the Japanese Twittersphere," *Asian Journal of Communication*, vol. 29, no. 3, pp. 274–290, 2019.

[2] S. Sato and F. Imamura, "An analysis of web coverage on the 2018 west Japan heavy rain disaster," *Journal of Disaster Research*, vol. 14, no. 3, pp. 531–538, 2019.

[3] Y. Xu, C. Yang, S. Peng, and Y. Nojima, "A hybrid two-stage financial stock forecasting algorithm based on clustering and

ensemble learning," *Applied Intelligence*, vol. 50, no. 11, pp. 3852–3867, 2020.

[4] I. Mele, S. A. Bahrainian, and F. Crestani, "Event mining and timeliness analysis from heterogeneous news streams," *Information Processing & Management*, vol. 56, no. 3, pp. 969–993, 2019.

[5] S. S. Lee, "Conceptual extraction of compound Korean keywords," *Journal of Information Processing Systems*, vol. 16, no. 2, pp. 447–459, 2020.

[6] B. Purbantoro, J. Aminuddin, N. Manago et al., "Comparison of cloud type classification with split window algorithm based on different infrared band combinations of Himawari-8 satellite," *Advances in Remote Sensing*, vol. 07, no. 03, pp. 218–234, 2018.

[7] A. Bovet and H. A. Makse, "Influence of fake news in Twitter during the 2016 US presidential election," *Nature Communications*, vol. 10, no. 1, pp. 1–14, 2019.

[8] S. Choi, H. Shin, and S.-S. Kang, "Predicting audience-rated news quality: using survey, text mining, and neural network methods," *Digital Journalism*, vol. 9, no. 1, pp. 84–105, 2021.

[9] T. Akamizu, "Thyroid storm: a Japanese perspective," *Thyroid*, vol. 28, no. 1, pp. 32–40, 2018.

[10] W. Zhang and D. Zinoviev, "How North Korea views China: quantitative analysis of Korean central news agency reports," *Korean Journal of Defense Analysis*, vol. 30, no. 3, pp. 377–396, 2018.

[11] J. M. Mintal and R. Vancel, "(Un) trendy Japan: twitter bots and the 2017 Japanese general election," *Politics in Central Europe*, vol. 15, no. 3, pp. 497–514, 2019.

[12] J. Dou, A. P. Yunus, D. T. Bui et al., "Improved landslide assessment using support vector machine with bagging, boosting, and stacking ensemble machine learning framework in a mountainous watershed, Japan," *Landslides*, vol. 17, no. 3, pp. 641–658, 2020.

[13] H. Li and Z. Jin, "Related research on "the belt and road" initiative based on big data text mining: taking the domestic area and the Korean Peninsula as an example," *OALib*, vol. 06, no. 09, pp. 1–15, 2019.

[14] W. Souma, I. Vodenska, and H. Aoyama, "Enhanced news sentiment analysis using deep learning methods," *Journal of Computational Social Science*, vol. 2, no. 1, pp. 33–46, 2019.

[15] A. Joshi, S. Karimi, R. Sparks et al., "Survey of text-based epidemic intelligence: a computational linguistics perspective," *ACM Computing Surveys (CSUR)*, vol. 52, no. 6, pp. 1–19, 2019.

[16] S. Lahiri, R. Mihalcea, and P.-H. Lai, "Keyword extraction from emails," *Natural Language Engineering*, vol. 23, no. 2, pp. 295–317, 2017.

[17] Y. Cochrane, "An investigation into the roles of the Katakana syllabary in Japanese discourse: from the perspective of discourse producers' motivation," *Contemporary Japan*, vol. 31, no. 2, pp. 215–233, 2019.

[18] M. Karimi, D. Jannach, and M. Jugovac, "News recommender systems-survey and roads ahead," *Information Processing & Management*, vol. 54, no. 6, pp. 1203–1227, 2018.

[19] Y. Wu, "Is automated journalistic writing less biased? an experimental test of auto-written and human-written news stories," *Journalism Practice*, vol. 14, no. 8, pp. 1008–1028, 2020.

[20] L. Zheng, F. Wang, X. Zheng, and B. Liu, "Discovering the relationship of disasters from big scholar and social media news datasets," *International Journal of Digital Earth*, vol. 12, no. 11, pp. 1341–1363, 2019.

[21] J. Yang, J. Wen, Y. Wang et al., "Fog-based marine environmental information monitoring toward ocean of things," *IEEE Internet of Things Journal*, vol. 7, no. 5, pp. 4238–4247, 2019.

[22] J. Yang, J. Zhang, and H. Wang, "Urban traffic control in software defined Internet of things via a multi-agent deep reinforcement learning approach," *IEEE Transactions on Intelligent Transportation Systems*, pp. 1–13, 2020.

[23] Y. Sun, H. Song, A. J. Jara, and R. Bie, "Internet of things and big data analytics for smart and connected communities," *IEEE Access*, vol. 4, pp. 766–773, 2016.

[24] X. Liu, H. Song, and A. Liu, "Intelligent UAVs trajectory optimization from space-time for data collection in social networks," *IEEE Transactions on Network Science and Engineering*, p. 1, 2020.

[25] A. Onan, S. Korukoğlu, and H. Bulut, "A multiobjective weighted voting ensemble classifier based on differential evolution algorithm for text sentiment classification," *Expert Systems with Applications*, vol. 62, pp. 1–16, 2016.

[26] O. Aytuğ, "Sentiment analysis on Twitter based on ensemble of psychological and linguistic feature sets," *Balkan Journal of Electrical and Computer Engineering*, vol. 6, no. 2, pp. 69–77, 2018.