Scalable Computing: Practice and Experience, ISSN 1895-1767, http://www.scpe.org
© 2025 SCPE. Volume 26, Issues 3, pp. 1539–1550, DOI 10.12694/scpe.v26i3.4326

A STUDY ON FAST ENGLISH SENTENCE RETRIEVAL BASED ON SIMHASH AND VECTOR SPACE MODEL TF-IDF IN AN E-LEARNING ENVIRONMENT

YUEHUA LI*AND XINXIN GUAN†

Abstract. With the rapid development of the digital information age on the Internet, information data on the Internet grows exponentially every day. In today's online learning environment, fast retrieval of English sentences plays a crucial role in the teaching and learning of modern English. The current case-based Machine translation methods can perform in-depth Parsing on sentences, and only use similar instances in the original corpus for matching and replacement processing. However, there are still certain limitations in terms of retrieval speed and similarity calculation. The study proposes an improved Simhash algorithm, which introduces substitution cost for synonym replacement and combines Term Frequency-Inverse Document Frequency (TF-IDF) weights with lexical weights for sentence-to-sentence similarity calculation. The results showed that the performance of the improved Simhash algorithm reached a maximum RI of 98.9%, an improvement of 1.4% compared to the traditional Simhash algorithm. The minimum misclassification rate of the improved algorithm was only 1.1%, a reduction of 1.4% compared to the traditional algorithm. The runtime of the improved Simhash algorithm was only 0.71s per sentence without processing synonyms and 1.82s with processing synonyms, while the runtime of the TF-IDF method alone was 71.82s and 98.11s in these two cases respectively. The improved Simhash algorithm, which combines TF-IDF weight, part of speech weight, and replacement cost, achieved an average accuracy of 92.87%, a recall rate of 88.7%, and an F1 Score of 92.87% in two calculations. This shows that the improved Simhash algorithm has high retrieval accuracy for fast retrieval of English sentences and shows excellent performance, providing a reliable technical support for the current English learning field.

Key words: Simhash algorithm; TF-IDF; Similarity; Synonym replacement; English search

1. Introduction. The massive growth in the amount of information on the Internet has led to data overload, making it difficult for users to get exactly and quickly to the information they most want to wade through, thus increasing the cost of effort and time for users. The field of English language teaching on the Internet has also been greatly affected, and research into the rapid retrieval of English sentences has far-reaching implications for the effective implementation of English language teaching. A great deal of research has been done on sentence retrieval at home and abroad. The traditional method of sentence retrieval is to judge the similarity between sentences based on the matching degree of keywords, with more matching words indicating a higher degree of similarity. However, this method only considers individual words and appears too general [1]. Current retrieval methods divide sentence similarity into three levels: semantic, syntactic and pragmatic. However, this method is extremely difficult to implement and cannot be used in practical retrieval. Commonly used similarity detection techniques include edit distance-based, identical vocabulary-based and vector space model-based methods [2]. The identical vocabulary-based approach is relatively simple, as it only requires the number of identical words between two sentences to be judged. However, the accuracy of the calculation is lower than the other methods. The method based on vector space model is based on constructing each sentence as a high-dimensional vector and judging the semantic similarity by the cosine of the angle between the two vectors [3]. However, this method is only applicable to very few fields and cannot meet the actual large-scale and special occasion measurements, and also suffers from the problem of inaccurate calculation due to information omission. Therefore, an improved Simhash algorithm combining Term Frequency-Inverse Document Frequency (TF-IDF) weights, lexical weights and substitution costs is designed to address the above problems and applied to the fast retrieval of English sentences in order to achieve better results in sentence retrieval. The algorithm is also applied to English sentence fast retrieval with a view to achieving better application results in sentence retrieval.

*Basic Teaching Department, Yantai Vocational College, Yantai, 264670, China (Corresponding author, Yuehua_Li23@outlook.com)
†Basic Teaching Department, Yantai Vocational College, Yantai, 264670, China

2. Literature review. The proliferation of information on the Web has had a negative impact on the current English learning environment, and several researchers have conducted studies on the retrieval of English information. Fu et al. argued that the size of training data for parallel text is still limited and designed an adversarial bidirectional sentence embedding mapping structure. The structure was able to map a limited amount of parallel text data and was shown to exhibit significant advantages in low-resource environments [4]. Boban et al. proposed the use of a reverse sentence frequency method to retrieve English sentences in order to achieve a more intelligent English sentence retrieval goal and to verify the effect of different query lengths on retrieval. The results showed that the method significantly improved sentence querying [5]. Ye et al. designed an intelligent retrieval algorithm based on wireless sensor networks to address the problems of long retrieval time and low accuracy of situational English information. The algorithm uses information filtering and structured documents to achieve intelligent retrieval of English information, and the results show that the method effectively reduces retrieval time and significantly improves accuracy [6]. Kim et al. found that current visual language methods can only support up to two languages, so a modular solution was devised. The method is able to perform more language tasks by means of multimodal language embedding, and results show that it supports up to four languages with an average recall of 20.3% [7]. Khot's team designed a multi-hop inference dataset to optimise a linguistic inference model in order to achieve efficient knowledge combination from multiple texts. The model was able to retrieve and combine valid information from a large corpus of English, and the results showed that the method significantly improved retrieval performance [8].

The Simhash algorithm provides effective technical support for various fields. realizing the importance of medical knowledge graphs for the biomedical field, Wu et al. developed an inference model for reasoning about the realization of paths in combination with the Simhush algorithm and applied it to practical medical detection. The results show that the model exhibits excellent performance for medical applications [9]. Rao et al. found that visual similarity-based techniques could not detect phishing sites in legitimate regions, so Simhash with perceptual hash was introduced to calculate the similarity of phishing points and a random forest model was used to evaluate the effectiveness of the heuristic filter. The results showed that the accuracy of the model was as high as 98.73% [10]. Xiao and other researchers constructed a digital ELT hierarchical retrieval model to address the problems of low search-completeness and accuracy of traditional ELT retrieval models. The model combines the TF-IDF method with the Simhush algorithm to detect the similarity of database documents, and the results show that the model has a high completeness rate of 95% and an accuracy rate of over 96% [11]. Lin et al. considered that the current evaluation methods of network security are too complicated, and designed the Simhash model in a big data environment. The model focuses on dividing the network into multiple modules to obtain security data, and the results show that it is well adapted to large-scale data network evaluation [12].

In summary, many researchers have devised effective solutions for retrieval of English information and have achieved corresponding success in the improvement and application of the Simhash algorithm. However, few scholars have conducted experimental studies on the fusion of the two treatments, so the study introduces an improved Simhash algorithm based on the traditional Simhash algorithm and applies this to the fast retrieval of English sentences in order to obtain better practical application results.

3. Objective of the work. This article mainly explores the rapid retrieval of English sentences in the online learning environment. Traditional online English teaching has problems such as slow English information retrieval and difficulty in obtaining information, which affects students' learning effectiveness and efficiency. To solve this problem, this paper discusses the retrieval algorithm of similar cases in the case Machine translation system, and proposes a retrieval method of similar cases of English sentences suitable for large-scale corpora, in order to achieve good results in modern online English teaching. The research content mainly includes four parts. The first part mainly reviews the retrieval problem of English information and the application of Simhash algorithm. The second part mainly introduces a locally sensitive hash algorithm called Simhash, which can be used for webpage deduplication, and provides a detailed introduction to its working principle. Furthermore, the feasibility of applying it to similar case retrieval in Machine translation is discussed. At the same time, a Vector space model algorithm with high accuracy TF-IDF method is introduced. And apply the combination of the two to the rapid retrieval of English sentences. The third part verifies the retrieval effect of the proposed method on English sentences. The fourth part analyzes the experimental results to demonstrate the superiority of the proposed method. At the same time, propose areas for improvement in the research and

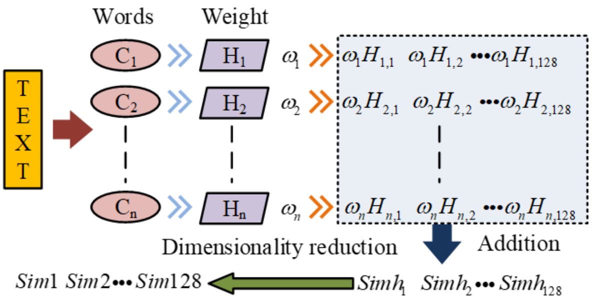


Fig. 4.1: Specific algorithm flow of Simhash

provide prospects for the future work to be done.

4. English sentence search based on TF-IDF.

4.1. Simhash algorithm based on similarity detection. The Simhash algorithm is essentially a locally sensitive hashing algorithm, which was first used in the search engine of a large number of web pages. The Simhash algorithm is used to obtain a Simhash value by dimensionality reduction of web pages, and then compare the Simhash values of different web pages using the Hemming distance to determine their similarity. The Simhash algorithm is widely used in the fields of text similarity detection, page de-duplication and sentence retrieval. Traditionally, text similarity detection is usually performed by word separation and then converted into a feature vector for distance measurement. However, the large number of feature vector words within a single text increases the dimensionality of the algorithm, which leads to an increase in computational cost and cannot be applied in larger scale environments. The Simhash algorithm aims to reduce dimensionality by mapping high-dimensional feature vectors into fixed-dimensional fingerprints through a dimensionality reduction process, and obtain the similarity of web content by comparing the fingerprints of two texts [13]. The Simhash algorithm consists of five main steps, namely word separation, hashing, weighting, merging and dimensionality reduction. The implementation steps are: firstly, the original text is divided into words to obtain the set of words$\{W_1, W_2, ..., W_n\}$ and set different levels of weights for each word in the text. The hash value of each word is then calculated using the hash, and the 0 in the hash value is turned into -1, thus transforming the set of words$\{W_1, W_2, ..., W_n\}$ into the set of$\{H_1, H_2, ..., H_n\}$, whereH_i represents the hash value of the$n$ bits. Next, the weights of each word are weighted into the$\{H_1, H_2, ..., H_n\}$ set and all the hash values in the set are accumulated in turn to obtain an bit text feature value, denoted as$\{Simh_1, Simh_2, ..., Simh_n\}$, which is calculated as shown in equation (4.1).

$$Simh_j = \sum\nolimits_{i=1}^{n} H_{ij}\mu_i \tag{4.1}$$

In equation (4.1),μ_i represents the weight of each word andH_{ij} refers to the$j$ bit of the hash value of the$i$ word. Finally, the Simhash signature is obtained by dimensionality reduction of the text feature values, which is calculated as shown in equation (4.2).

$$Sim_j = redu\left(Simh_j\right) = \begin{cases} 1 & Simh_j > 0 \\ 0 & Simh_j \leq 0 \end{cases} \tag{4.2}$$

The specific algorithm flow of Simhash is shown in Figure 4.1.

The Simhash algorithm uses the Hamming distance to determine the similarity of two pieces of data. The Hamming distance represents the number of different index positions in each of two equal strings and is calculated as shown in equation (4.3).

$$Hammin\left(x, y\right) = \sum\nolimits_{i=1}^{n} y_i \oplus x_i \tag{4.3}$$

In equation (4.3),$x = (x_1, x_2, ..., x_n)$,$y = (y_1, y_2, ..., y_n)$, and$\oplus$ represent heterogeneous operations. The Simhash algorithm converts text into signatures, which facilitates retrieval and also plays a space-saving role. The similarity of two texts can be calculated by the Hemming distance of the signature, as shown in equation (4.4).

$$sim\left(T_1, T_2\right) = \frac{\sum_{k=1}^{128} T_{2k} \oplus T_{1k}}{128} \tag{4.4}$$

In equation (4.4),T_{1k} andT_{2k} refer to the value at the$k$ bit of the two signatures and 128 represents the number of bits in the string. The smaller the Hemming distance of the signatures, the higher the similarity of the two texts. The traditional Simhash algorithm mainly uses the number of occurrences of feature terms as the weight of the weighting, which will result in the Simhash signature not accurately characterising the textual information [14]. Therefore, the study introduces the TF-IDF value and combines it with the lexical properties of words to jointly calculate the weights of feature items. TF-IDF is a numerical weighting calculation method commonly used in natural language processing, which is widely used in information retrieval, text clustering and other fields. The TF refers to word frequency, which represents the number of times a word appears in a text, and IDF means inverse text frequency index, which characterises the text differentiation ability of a word. The weight of a word is calculated as the product of TF and IDF [15-16]. The TF value is calculated as shown in equation (4.5).

$$TF_{ij} = \frac{b_{ij}}{\sum_{l} b_{ij}} \tag{4.5}$$

In equation (4.5),b_{ij} represents the number of occurrences of the$i$ feature word of the$j$ text in the text, and$l$ refers to the set of all words in the text. The IDF value is calculated as shown in equation (4.6).

$$IDF_{ij} = \log \frac{|D|}{|\{d : t_{ij} \in d\}|} \tag{4.6}$$

In equation (4.6),D represents the set of texts, andt_{ij} represents the$i$ th feature word of the$j$ th text. Let the total number of data in the text dataset be N, and the feature wordt_{ij} exists in the$c_{i,j}$ text, the IDF value is calculated as shown in equation (4.7).

$$IDF_{ij} = \log \frac{N}{\beta + c_{i,j}} \tag{4.7}$$

In equation (4.7),β is generally taken as 1, which serves to prevent the denominator from being 0. The weights of the feature terms are calculated as shown in equation (4.8).

$$w_{ij} = IDF_{i,j} \bullet TF_{i,j} \tag{4.8}$$

Assume two text vectors as shown in equation (4.9).

$$\begin{array}{c} \{W_1 - T_1, W_2 - T_2, ..., W_N - T_N\} \\ \{W_1 - T_1', W_2 - T_2', ..., W_N - T_N'\} \end{array} \tag{4.9}$$

The similarity of two texts can be determined by the cosine of the angle between the two vectors, calculated as shown in equation (4.10).

$$Similarity(V, V') = \frac{\sum_{i=1}^{n} T_i \bullet T_i'}{\sqrt{\sum_{i=1}^{n} {T_i}^2 \bullet \sum_{i=1}^{n} T_i'^2}} \tag{4.10}$$

In equation (4.10),V andV' represent two vectors. TF-IDF is feasible for the calculation of feature item weights and text similarity, but it only considers the number of occurrences of feature items, ignoring the influence of different lexicalities on the semantic expression of the text. Therefore, the study uses the lexicality

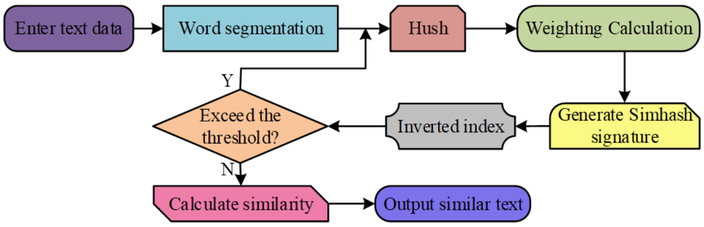


Fig. 4.2: Specific process of improved Simhash algorithm

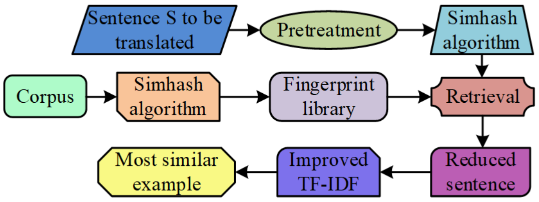


Fig. 4.3: Flow chart of fast English sentence retrieval

of feature words as a measure of word weights as well. The final feature word weights are calculated as shown in equation (4.11).

$$w_{ij} = w_k \bullet IDF_{i,j} \bullet TF_{i,j} \tag{4.11}$$

In Eq. (4.11),w_k refers to the weights corresponding to the specified lexical properties. The noun weight is 4, the verb is 3, the adjective is 2 and the rest of the lexical nature is 1. The combination of TF-IDF value and lexical nature is introduced into the feature weight calculation, which can characterise the text content more comprehensively, thus increasing the effectiveness of the Simhash algorithm for text similarity detection. The specific flow of the improved Simhash algorithm is shown in Figure 4.2.

4.2. Simhash-based Fast English Sentence Retrieval. The study is based on the Simhash algorithm and the vector space model TF-IDF for fast retrieval of English sentences. Among them, the Simhash algorithm focuses on quickly detecting similar texts from a large amount of information data and then returning the text set [17]. The vector space model-based TF-IDF method, on the other hand, focuses more on the accurate representation of the internal information of the text. In a practical translation system, the sentence with the highest similarity to the sentence to be translated needs to be retrieved quickly from a large-scale corpus. Therefore, the study combines the Simhash algorithm with the TF-IDF method to design an algorithm for retrieving the most similar text instances with high accuracy. The algorithm first selects sentences with high similarity by generating a fingerprint library through the Simhash algorithm to form a reduced set of sentence instances. Then a synonym dictionary is applied to all sentences and the replacement cost is calculated. Finally, the improved TF-IDF is used to construct the feature vector and calculate the similarity between each sentence and the sentence to be translated, so as to find the example sentence with the maximum similarity to the sentence to be translated. The flow of the algorithm is shown in Figure 3.

In the Simhash algorithm, the Hemming distance between fingerprints is calculated by first using a heterogeneous operation and then checking the number of ones in the result. Google's web de-duplication algorithm uses the drawer principle to create an inverted index to calculate the Hemming distance. The study refers to this algorithm to calculate the Hemming distance by grouping fingerprints for retrieval. For a 32-bit fingerprint, the threshold is set to 7 and the fingerprint is divided into 8 equal parts of 4 bits each. The specific operation flow is shown in Figure 4.4.

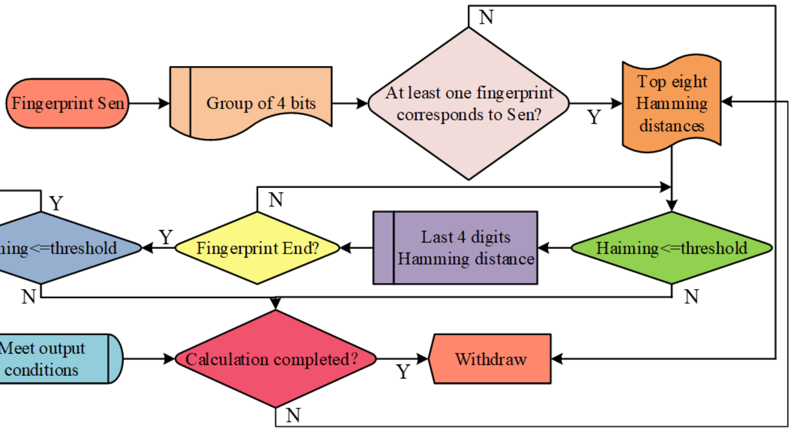


Fig. 4.4: Calculation of Hamming Distance of Fingerprint

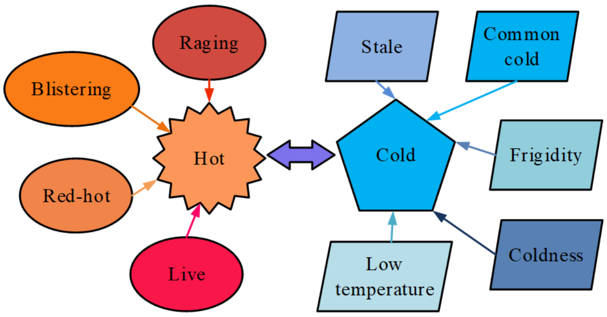


Fig. 4.5: Synonyms of hot and cold

After narrowing down the range of similar sentences using the Simhash algorithm, the similarity of each instance to the sentence to be translated was then calculated. However, the traditional vector method is unable to identify semantic information, resulting in low similarity calculation results. Therefore, the study uses the sentence to be translated, S, as a benchmark and replaces all words that have a synonymous relationship with S with words that exist in [18-19]. This method can make the similarity calculation results more accurate, and at the same time has a good effect of dimensionality reduction, which in turn simplifies the calculation of TF-IDF. However, since there are multiple meanings in natural language, direct substitution after detecting synonyms will lead to substitution errors. For this reason, the study further introduces a parameter to measure the correct rate of substitution between synonyms, namely the substitution cost. When the substitution is correct, the substitution cost is 1, which means that the substitution is possible; when the substitution is incorrect, the substitution cost is 0, and the substitution should not be made. When the replacement is not necessarily correct, then the replacement cost is between 0 and 1. The replacement cost allows the weight of the replaced position to be reduced, thus optimising the results of the calculation. WordNet is currently used to process English vocabulary, not only for lexical purposes but also for semantic word relations, including antonymy, synonymy, subordination and whole-part relations. Antonymic relations are usually found between adjectives, which characterise semantic information through an N-dimensional hyperspace structure, and use antonymic relations to link different clusters of synonyms. For example, the synonymic clusters of hot and cold are represented as shown in Figure 4.5.

When two words are substituted for each other in a linguistic text without affecting the original semantics,

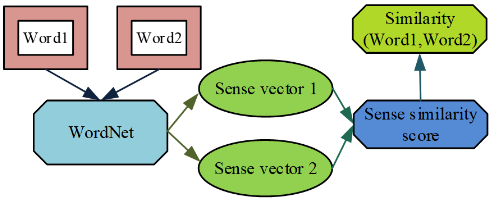


Fig. 4.6: Calculation process of word meaning similarity

the two words are in a synonymic relationship. The superior-subordinate relationship is also referred to as the parent-child relationship or ISA relationship, which is transitive in nature. A whole-part relationship is one in which the meaning of a word is part of another word set [20]. The substitution cost is mainly used to calculate the correctness of the substitution operation, and the lexical similarity of the two words is usually used as the substitution surrogate value. The process of calculating the lexical similarity is shown in Figure 4.6.

When calculating word sense similarity using WordNet, feature extraction of the word sense is first required. The extraction is calculated as shown in equation (4.12).

$$Feature(SW) = \{\{WE\}, \{WC\}, \{WS\}\} \tag{4.12}$$

In equation (4.12),WE represents all real words in the interpretation of$W$,WC refers to all related genera, andWS represents all synonyms of$W$ in WordNet. The similarity of two words can be obtained by calculating their distances in different feature spaces; the further the distance, the smaller the similarity. The similarity is calculated as shown in equation (4.13).

$$Similarity(SW_i, SW_j) = \frac{1}{No(SW)\bullet No(SW_j)} \times \\ \frac{\sum_{W_i\in\{W_{Si}\}\cap\left\{W_{Sj}\right\}} IDF(w_i)^2\bullet K_S+\sum_{W_i\in\{W_{Ci}\}\cap\left\{W_{Cj}\right\}} IDF(w_i)^2\bullet K_C+\sum_{W_i\in\{W_{Ei}\}\cap\left\{W_{Ej}\right\}} IDF(w_i)^2\bullet K_E}{\sqrt{\sum_{i\in Q_o,K\in\left\{K_E,K_C,K_S\right\}} IDF(w_i)^2\bullet K\times\sum_{j\in Q_p,K\in\left\{K_E,K_C,K_S\right\}} IDF(w_j)^2\bullet K}} \tag{4.13}$$

In equation (4.13),$IDF\left(w_i\right)$ represents the countdown of the number of times a text appears when WordNet is created. $No(SW)$represents the order of meaning of the words. $K_E K_C$ andK_S represent the feature weights of sense interpretation, class attributes and synonyms respectively. $Q_O Q_P$ refers to the set of indicators wherew_i appears;w_j refers to the set of indicators where and appear. IfSW_1 andSW_2 are used to denote the number of senses ofW_1 andW_2 , respectively, the word sense similarity is calculated as shown in equation (4.14).

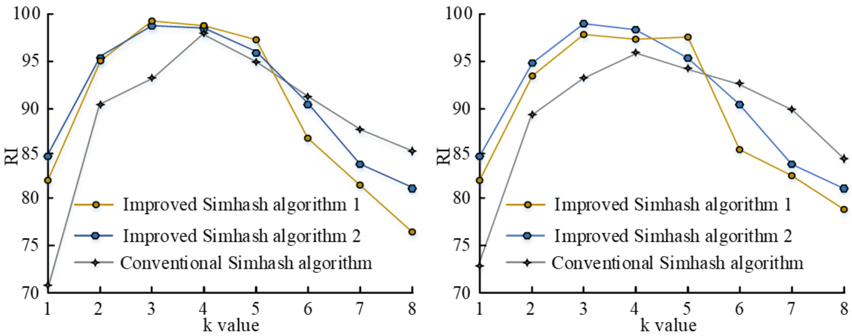
$$Similarity(W_1, W_2) = \frac{\sum_{i\in\{1,\ldots,|SW1|\},j\in\{1,\ldots,|SW2|\}} S(SW1_i,SW2_j)}{|SW1|+|SW2|}+ \\ \frac{\sum_{i\in\{1,\ldots,|SW2|\},j\in\{1,\ldots,|SW1|\}} S(SW2_i,SW1_j)}{|SW1|+|SW2|} \tag{4.14}$$

The research further incorporates the replacement cost into the TF-IDF algorithm to improve the algorithm. The steps of the improved TF-IDF algorithm are as follows: first, WordNet is used to compare the translated sentence S with each example sentence in the narrowed down instance library E, and synonym pairs are found. Calculate the semantic similarity α of the synonym pair, and use this as the substitution value of the synonym pair. Next, replace synonyms with the words appearing in the sentence S to be translated, multiply by α at the replacement position, and construct feature vectors for E and S. And finally the similarity between the two sentences is calculated and the most similar instance S' is obtained, whose similarity is calculated as shown in equation (4.15).

$$Similarity = \frac{\sum_{i=1}^{n} \omega_i \bullet \omega_i'[\alpha]}{\sqrt{\sum_{i=1}^{n} \left(\omega_i[\alpha]\right)^2} \times \sqrt{\sum_{i=1}^{n} \left(\omega_i'\right)^2}} \tag{4.15}$$

Table 5.1: Experimental environment and configuration

Operate	Ubuntu 16.04
Memory	128G
Hard disk	10T
Programming language	Python3.6
GPU	GTX 1080
Deep learning framework	TensorFlow
CPU	Intel Xeon E5-2682 v4



(a) RI value of the first test

(b) RI value of the second test

Fig. 5.1: RI values for different algorithms

5. Experimental analysis of English sentence retrieval based on Simhash algorithm. All experiments were conducted on the server host in the laboratory, using TensorFlow, a deep learning framework developed by Google for data flow programming for multiple tasks, as the implementation tool for the network model. Accelerate training with a single NVIDIA GTX 1080 graphics card. The experimental environment and configuration are shown in Table 5.1.

In order to verify the effectiveness of the improved Simhash algorithm for retrieval of English sentences, the study first conducted performance tests on the clustering effect of the algorithm and chose the traditional Simhash algorithm to compare the results. Due to the fact that online news conforms to the density connected model, the test data was selected from a 200W scale English news dataset, and 400 of them were randomly selected for evaluation experiments.The study selected the RI (Rand Index) value as the performance testing indicator, and the larger the RI value, the better the clustering effect. In general, an RI value of 90% is required to meet practical application requirements. Since the number of segments k has a large impact on the clustering effect, different k values need to be set, and the experiment was conducted twice.

In Figure 5.1, the same clustering evaluation method and manual discriminant method as the traditional Simhash algorithm were used for the improved Simhash algorithm respectively, resulting in two different curves. From the results of the two tests, it can be seen that the improved Simhash algorithm has the highest RI value when the value of k is taken as 3, and it reaches a maximum of 98.9%. When the value of k is greater than 3, the clustering effect will then decrease. The reason for this is that too large a value of k will cause otherwise unrelated clusters to be combined together, thus reducing the clustering accuracy. The traditional Simhash algorithm takes the highest RI value of 97.5% when k is taken to 4, a decrease of 1.4% compared to the improved Simhash algorithm. The misclassification rate is equal to 1 - the RI value, and the minimum misclassification rate of the traditional Simhash algorithm is 2.5%, while the minimum misclassification rate of the improved algorithm is only 1.1%.

Further evaluate the effectiveness of the improved Simhash algorithm in fast English sentence retrieval, using three indicators: accuracy, recall, and F1 Score to measure the algorithm's effectiveness. Among these

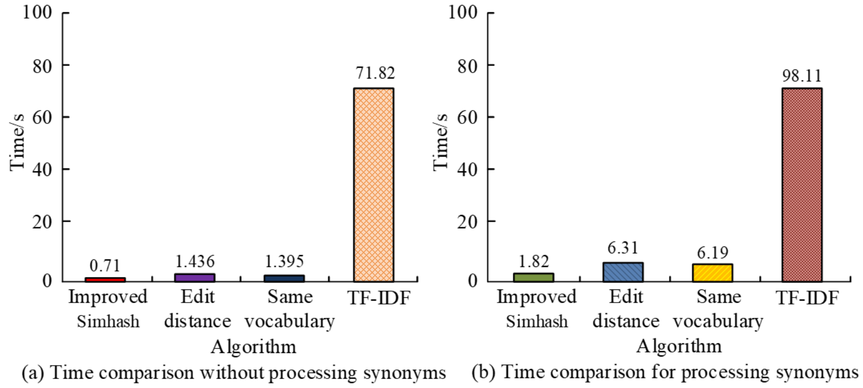


Fig. 5.2: Comparison of running times of different methods

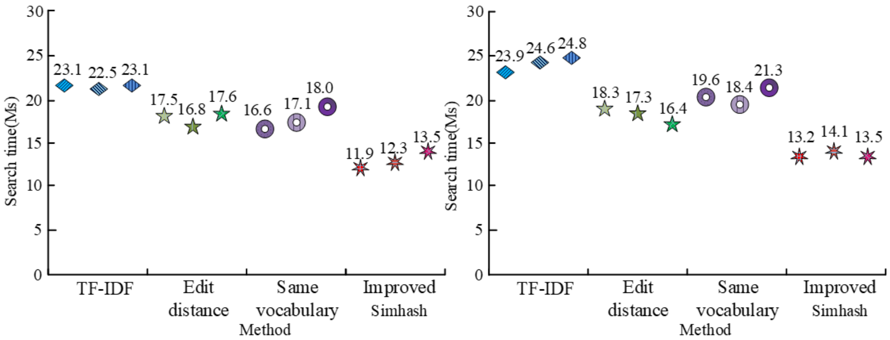
three methods, the edit distance and identical vocabulary-based methods only consider the identical words in two sentences, ignoring the corpus as a whole, which results in lower computational accuracy. The vector-based TF-IDF method, on the other hand, considers different words, identical words and the influence of each word in the corpus on the sentence, and is therefore more accurate. Thirty English sentences from the corpus were used to compare the running time of the four algorithms. was used to record the running time of each sentence instance and to calculate the average time spent by each algorithm. At the same time, the English-Chinese parallel corpus was chosen as the experimental corpus, with a size of 9,948 pairs. The experiments were conducted separately to compare and analyse the case of no synonym processing with the case of introducing synonym processing, and the experimental results are shown in Figure 5.2 8.

As can be seen from Figure 5.2 , the TF-IDF algorithm alone has the longest running time for each sentence, 71.82s and 98.11s respectively, both in the case of no synonyms and in the case of synonyms. the reason for this is that the TF-IDF algorithm requires a high-dimensional vector construction for each sentence instance, which leads to a significant time consumption. The experimental results based on the same vocabulary and edit distance methods are not very different, around 1.4s versus 6.2s in the two cases respectively. In contrast, the improved Simhash algorithm proposed in the study runs in only 0.71s and 1.82s in both cases, which can be seen to have a significant advantage in terms of time performance. In addition, the TF-IDF method has a larger increase in runtime after the introduction of synonym processing compared to the other three methods, mainly because it not only performs synonym queries but also calculates word similarity. In contrast, the method based on edit distance and identical words only performs synonym queries. The improved Simhash algorithm reduces the time consumption for synonym processing as the range of similar instances is reduced, thus reducing the number of queries for synonyms and calculating word similarity. The study continued to measure the retrieval time for different sizes of text fingerprints and repeated the measurement three times to take the average value.

As can be seen from Figure 5.3, the retrieval speed of fingerprints slows down as the size of the fingerprint library increases. The improved Simhash algorithm also has the lowest average retrieval time of 12.6ms and 13.6ms for fingerprints of size 180,000 and 200,000 respectively, which is 10.3ms and 10.8ms respectively compared to the TF-IDF method.

Since the TF-IDF method considers factors such as different words, identical words and the influence of each word on the sentence at the same time, its computational accuracy is relatively high. Therefore, experiments were conducted to analyse its similarity optimisation results with the improved Simhash algorithm. When synonyms were correctly replaced, some of the test results are shown in Table 5.2.

In Table 5.2, a represents the experimental input sentence and b represents the output sentence after synonymous replacement. As can be seen from Table 1, when the synonyms in the sentence instances are correctly replaced, then the Simhash algorithm proposed by the study computes higher similarity results than the TF-IDF method alone. In the tested sentence pairs, the Simhash algorithm achieves a maximum similarity result of 0.9542, which is close to the true value and 0.1217 higher than that of the TF-IDF method, while the



(a) Retrieval time when fingerprint size is 180000 (b) Retrieval time when fingerprint size is 200000

Fig. 5.3: Retrieval time under different scale fingerprints

Table 5.2: Similarity Results of Correct Substitution of Synonyms

Test sentence	Improved Simhash	TF-IDF
a: Word can not depict the wonders of nature	0.8521	0.5075
b: Word can't describe the beauty of the scene		
a: She believes that the present continent once including his name after Ultima of Pangea	0.8369	0.7214
b: She once proposed that the present mainland including a continent he named Pangaea		
a: Please recommend a shoe store to me	0.9542	0.8325
b: I am looking for a less expensive store		

Table 5.3: Similarity results of synonyms being incorrectly replaced

Test sentence	TF-IDF	Improved Simhash
a: I begged Betty to give me some staples	0.5367	03465
b: Betty begged Mary to take a course for him		
a: This is a brightly lit room with high windows	0.2443	0.0136
b: This paper uses substantive cases to popularize it and transparent		
a: The teacher recorded my grades on the form	0.3798	0.0498
b: I will reserve a table for eight		

difference in similarity reaches a maximum of 0.3446. The result will be higher than the true value. Some of the results of the tests when synonyms were replaced incorrectly are shown in Table 5.3.

As can be seen from Table 3, the improved Simhash algorithm, which introduces the cost of synonym substitution, computes significantly lower similarity results when synonyms are replaced incorrectly. The reason for this is that the algorithm reduces the adverse effect of incorrect substitution on the selection of similar sentences and effectively optimises the calculation of similarity. The degree of optimisation mainly depends on the size of the replacement generation value, the smaller the replacement cost the greater the probability of incorrect replacement occurring, thus leading to an improved optimization effect; the larger the replacement cost the smaller the probability of incorrect replacement occurring, then the optimisation effect is not significant.

The study further evaluates the effectiveness of the improved Simhash algorithm in English sentence fast retrieval, using three metrics: accuracy, recall and F1-Score to measure the effectiveness of the algorithm. At the same time, two traditional Simhash algorithms were chosen for comparison experiments with the improved

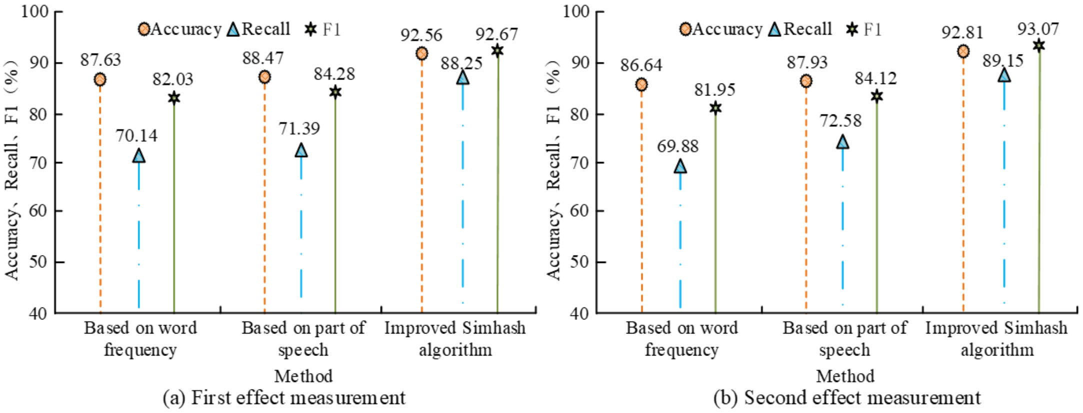


Fig. 5.4: Accuracy, recall, and F1-Score of different algorithms

Simhash algorithm, including the word frequency-based Simhash algorithm and the lexicality-based Simhash algorithm. The results of the two measurements are shown in Figure 5.4.

As can be seen from Figure 5.4, the improved Simhash algorithm that combines TF-IDF weights, lexical weights and replacement cost has an average accuracy of 92.87%, recall of 88.7% and F1-Score of 92.87% for the two measurements. Compared with the traditional Simhash based on word frequency weights, the improvement was 5.735%, 18.69% and 10.88% respectively. It indicates that the fusion of TF-IDF weights, lexical weights and substitution cost can distinguish the influence of different feature words on the text, thus enabling more feature information to be included in the obtained text fingerprint. At the same time, the improved Simhash algorithm effectively enriches the semantic information of the lexicon, which in turn significantly improves the correct substitution rate between synonyms and optimises the similarity calculation results to make them closer to the true value.

6. Conclusion. The complex web-based learning environment has made fast English sentence retrieval an important method for modern English learning. The study combines the TF-IDF method to develop an improved Simhash algorithm, which introduces substitution cost for synonym processing of English sentences and integrates TF-IDF weights with lexical weights for similarity calculation. The results show that the similarity values calculated by the improved Simhash algorithm are very close to the true values when the synonyms are correctly replaced. At the same time, when synonyms were replaced incorrectly, the similarity values calculated by the improved Simhash algorithm were not inflated. The lowest calculated value is 0.0136, which is 0.2307 lower compared to the TF-IDF method. meanwhile, the average retrieval time of the improved Simhash algorithm is 12.6ms and 13.6ms for fingerprints of 180,000 and 200,000 scale respectively. it is 10.3ms and 10.8ms lower compared to the TF-IDF method respectively. in addition, the improved the accuracy, recall and F1-Score of the Simhash algorithm effect reached 92.87%, 88.7% and 92.87%, respectively. Compared with the Simhash based on word frequency weights, they rose by 5.735%, 18.69% and 10.88% respectively. It indicates that the improved Simhash algorithm is highly feasible for fast retrieval of English sentences and has excellent performance qualities.This method can effectively solve the problems of slow retrieval speed and low accuracy in current online English teaching. It effectively improves the learning efficiency and effect of students, optimizes the online teaching effect of Modern English, and promotes the development of modern online English teaching.but there is still some room for improvement in its retrieval accuracy, and thus further improvement is needed.

Acknowledgement. The research is supported by Yantai Vocational Education Research Project in 2023: ”A Study on the Digital Competence of English Teachers in Higher Vocational Colleges under the Background of Educational Digital Transformation” (Higher Vocational Colleges 2023-5); Yantai Vocational College Horizontal Scientific Research Project: ”Development of the English Service Capacity Improvement Project for Yantai

Microspring Youth Affairs Service Center" (HX2024073).

REFERENCES

[1] J. Qin, *An Encrypted Image Retrieval Method Based on SimHash in Cloud Computing*, Computers, Materials and Continua, vol. 62, no. 3, pp. 389–399, 2020.

[2] R. H. Dong, C. Shu, Q. Y. Zhang, *Security Situation Assessment Algorithm for Industrial Control Network Nodes Based on Improved Text SimHash*, International Journal of Network Security, vol. 23, no. 6, pp. 973–984, 2021.

[3] M. J. Lim, Y. M. Kwon, *Efficient algorithm for malware classification: N-gram MCSC*, International Journal of Computing and Digital Systems, vol. 9, no. 2, pp. 179–185, 2020.

[4] Z. Fu, Y. Xian, S. Geng, Y. Ge, Y. Wang, X. Dong, G. De Melo, *ABSent: Cross-lingual sentence representation mapping with bidirectional GANs*, Proceedings of the AAAI conference on artificial intelligence, vol. 34, no. 5, pp. 7756–7763, 2020.

[5] I. Boban, A. Doko, S. Gotovac, *Sentence retrieval using stemming and lemmatization with different length of the queries*, Advances in Science, Technology and Engineering Systems, vol. 5, no. 3, pp. 349–354, 2020.

[6] Q. Ye, *Situational English Language Information Intelligent Retrieval Algorithm Based on Wireless Sensor Network*, International Journal of Wireless Information Networks, vol. 28, no. 3, pp. 287–296, 2021.

[7] D. Kim, K. Saito, K. Saenko, S. Sclaroff, B. Plummer, *Mule: Multimodal universal language embedding*, Proceedings of the AAAI Conference on Artificial Intelligence, vol. 34, no. 7, pp. 11254–11261, 2020.

[8] T. Khot, P. Clark, M. Guerquin, P. Jansen, A. Sabharwal, *Qasc: A dataset for question answering via sentence composition*, Proceedings of the AAAI Conference on Artificial Intelligence, vol. 34, no. 5, pp. 8082–8090, 2020.

[9] X. Wu, J. Duan, Y. Pan, M. Li, *Medical knowledge graph: Data sources, construction, reasoning, and applications*, Big Data Mining and Analytics, vol. 6, no. 2, pp. 201–217, 2023.

[10] R. S. Rao, A. R. Pais, *Two level filtering mechanism to detect phishing sites using lightweight visual similarity approach*, Journal of Ambient Intelligence and Humanized Computing, vol. 11, no. 9, pp. 3853–3872, 2020.

[11] Z. Xiao, *Ontology-based hierarchical retrieval model for digital English teaching information*, International Journal of Continuing Engineering Education and Life Long Learning, vol. 33, no. 2-3, pp. 337–350, 2023.

[12] P. Lin, Y. Chen, *Network Security Situation Assessment Based on Text SimHash in Big Data Environment*, International Journal of Network Security, vol. 21, no. 4, pp. 699–708, 2019.

[13] J. Qin, *An encrypted image retrieval method based on SimHash in cloud computing*, Computers, Materials & Continua, vol. 63, no. 1, pp. 389–399, 2020.

[14] S. Fedushko, *Scientific Content: Language Expansion in Bibliometric Databases*, 2020.

[15] S. Tang, *Identification of Scratch projects' Similarity Using Clustering Algorithms*, International Core Journal of Engineering, vol. 7, no. 12, pp. 158–170, 2021.

[16] S. Fedushko, O. Trach, Z. Kunch, Y. Turchyn, U. Yarka, *Modelling the behavior classification of social news aggregations users*, arXiv preprint arXiv:1909.01677, 2019.

[17] Q. Ye, *RETRACTED ARTICLE: Situational English Language Information Intelligent Retrieval Algorithm Based on Wireless Sensor Network*, International Journal of Wireless Information Networks, vol. 28, no. 3, pp. 287–296, 2021.

[18] R. S. Rao, A. R. Pais, *Two level filtering mechanism to detect phishing sites using lightweight visual similarity approach*, Journal of Ambient Intelligence and Humanized Computing, vol. 11, no. 9, pp. 3853–3872, 2020.

[19] Z. Xiao, *Ontology-based hierarchical retrieval model for digital English teaching information*, International Journal of Continuing Engineering Education and Life Long Learning, vol. 33, no. 2-3, pp. 337–350, 2023.

[20] X. Zhang, P. Li, X. Ma, Y. Liu, *Railway wagon flow routing locus pattern intelligent recognition algorithm based on SST*, Smart and Resilient Transport, vol. 2, no. 1, pp. 3–21, 2020.

Edited by: Sheikh Nasrullah
Special issue on: Transformative Horizons:
The Role of AI and Computers in Shaping Future Trends of Education
Received: Jul 22, 2024
Accepted: Aug 2, 2024