A Deep Learning-based Method for Determining Semantic Similarity of English Translation Keywords

Wu Zhili, Zhang Qian*
Department of International Education and Exchange, Cangzhou Vocational and Technical College,
Cangzhou, Hebei 061001, China

***Abstract*—In the English translation task, the semantics of context play an important role in correctly understanding the subtle differences between keywords. The bidirectional LSTM includes a positive LSTM and a reverse LSTM. When processing sequence data, you can consider the information of the preceding and following text at the same time. Therefore, to capture the subtle semantic differences between English translation keywords and accurately evaluate their similarity, a new semantic similarity determination method for English translation keywords is studied with the bidirectional LSTM neural network in deep learning as the main algorithm. This method introduces an English translation keyword extraction algorithm based on word co-occurrence and uses the co-occurrence relationship between words to identify and extract keywords in English translation. The extracted keywords are input into the bidirectional LSTM neural network keyword semantic similarity judgment model based on deep learning, and the weight of the bidirectional LSTM neural network is set by using the sparrow search algorithm to optimize. After the bidirectional LSTM neural network is trained, the information on keyword word vectors is captured, and the similarity between keyword word vectors is evaluated. The experimental results show that the sentence similarity calculated by the proposed method for English translation is very close to the result of professional manual scoring. The Spearman rank correlation coefficient of the semantic similarity determination result is 1, and the determination result is accurate.**

Keywords*—Deep learning; English translation; keyword; semantic similarity; co-occurrence of words; bidirectional LSTM neural network

I. INTRODUCTION

Under the influence of the expansion of the international economy and trade, English as an international common language has been given more attention; English translation has become an essential part, and all kinds of machine translation systems are developing rapidly [1]. Machine translation is no longer limited to individual grammar and sentence translation but more contextual information of sentence clusters, paragraphs, chapters, and genres within the language [2]. From a semantic point of view, word semantic computation can be defined in the whole text or between individual word meanings; thus, word semantics has a degree of relevance and similarity, that is, reflecting the commonality of two words in the same context and the aggregation of features between two words [3]. To a certain extent, the more similar the semantics of words, the greater the correlation, which can easily lead to misunderstandings in different contexts and bring difficulties to the translation work. At present, word semantic computation is more based on natural language processing to explore the degree of correlation between words, and many words in English have multiple meanings, which may vary according to the context, style, or context [4]. Therefore, it is a challenge to accurately determine the exact meaning of a word in a particular translation [5]. The meaning of certain words and expressions may be influenced by a particular cultural, historical, or social context [6]. For translators who are not familiar with such background information, interpreting the semantics of these terms accurately may be a difficult task [7].

The SI-LSTM model of study [8] captures the complex semantic relationships between keywords through shared inputs and LSTM networks. If the training data set does not cover enough language habits and semantic contexts in different backgrounds, the model may not be able to accurately capture the complex semantic relationships between keywords, resulting in a decline in the accuracy of semantic similarity determination. In addition, the model performance is limited by the size and quality of the training data, and the semantic relationships in some specific domains or specific contexts may not be captured effectively. The study in [9] introduces the network ontology structure and a variety of metrics to evaluate the semantic similarity between concepts. However, for some concepts in network ontology, this method may lack sufficient correlation information, which makes it impossible to accurately evaluate their semantic similarity. At the same time, when the network ontology structure is large and complex, the computational efficiency may be affected, and it is difficult to apply to large-scale data sets. The study in [10] uses RDF triples to evaluate semantic dependencies between entities. If the number of triples available in an RDF data set is limited, semantic correlation analysis based on these triples may suffer from data sparsity, resulting in inaccurate analysis results. In addition, the method cannot effectively evaluate the semantic relevance of emerging entities or relationships without recording the corresponding triples in the RDF data set. Although the non-categorical relational measurement method in study [11] can capture rich semantic information. When dealing with large data sets, the computation of non-categorical relational measures can become complex and time-consuming, requiring efficient algorithms and computational resources. At the same time, the text content, context information and other data that the method relies on May be affected by noise, ambiguity and other factors, resulting in the inaccuracy of semantic relation inference. In addition, the performance of the method is affected by the size and quality of the training data, and the semantic relationships in some specific domains or specific contexts may not be captured effectively. Based on the above analysis, it can be seen that in practical applications, appropriate methods should be selected according to specific scenarios and data characteristics, and a variety of methods should be combined to improve the accuracy and efficiency of analysis.

In order to accurately capture the subtle semantic differences of keywords in context in English translation tasks, this study proposes a deep learn-based semantic similarity determination method for English translation keywords. This method first uses a word co-occurrence based algorithm to identify and extract keywords from English translated texts, and then input these keywords into a bidirectional LSTM neural network model. By considering the contextual information of keywords at the same time, the bidirectional LSTM model can capture the semantic relationship between keywords more accurately. Further, we use the Sparrow search algorithm to optimize the weight setting of the bidirectional LSTM neural network to improve the performance of the model. Finally, by evaluating the similarity between keyword word vectors, this method can accurately determine the semantic similarity of keywords in English translation, provide strong support for improving the quality of English translation, and help promote the development of natural language processing.

II. Methods for Determining the Semantic Similarity of Keywords in English Translation

A. Keyword Extraction Algorithm for English Translation based on Word Co-Occurrence

1) Candidate word selection for English translation: Candidate word selection is the basic part of the English translation keyword extraction algorithm [12]. Due to the existence of a large number of words in English translation papers, if the weights of all English translation words are calculated, the efficiency of the algorithm will be greatly affected [13]. Therefore, the part about candidate word selection for English translation is to avoid the effect of calculating too many word weights. Candidate words are the words that satisfy the basic requirements for becoming keywords in English translation. This step is to select the words that meet the basic requirements of English translation keywords [14]. These basic requirements and how to select candidate words are introduced below.

Firstly, the English translation document is scanned and divided into several clauses according to specific truncation symbols (period, question mark, comma, number, etc.). Then, according to the specified length, scan the English translation clauses to get a fixed-length sequence of consecutive words. Since the number of keywords containing too many words is very small [15], the length of the candidate words also needs to be limited, and the length is set to. Despite this, a very large number of fixed-length sequences of consecutive words will still be produced, so the sequences of consecutive words containing stop words in the beginning or end position are deleted. English translation stops words for papers, adverbs, conjunctions, and other words without practical significance; these words cannot express the meaning of the statement but only play the role of the successive and transitive, so delete these phrases [16].

2) Calculation of weights of candidate words for English translation: Although the translation candidates are selected according to the above steps, since some words in English may have many different meanings, the applicability of the candidate words is determined by utilizing the English translation candidate weighting representation according to the contextual information when translating. To ensure that the context has the same meaning.

The merit of feature selection directly affects the keyword extraction effect of the algorithm [17]. First of all, the first appearance position of the candidate words is taken into account when calculating the weights. Words appearing at the front of the English translation document are more important than those appearing at the back of the document [18], and the algorithm should give more weight to them. This algorithm calculates the value of the first occurrence position of candidate words for English translation as follows:

$$g(Q,C)=\frac{o(Q,C)}{r(C)} \tag{1}$$

Among them, $g(Q,C)$ is the first occurrence of English translation candidate word Q in English translation document$C$;$o(Q,C)$is the first position of candidate word$Q$in English translation document$C$.$r(C)$is the number of all words in the English translation document$C$.

Secondly, the TF value feature is also added when calculating the weight of English translation candidates. The TF value (term frequency) indicates the frequency of an English translation word in the text. The frequency of candidate words in documents is the most important statistical feature of candidate words [19]. Therefore, the probability of words or phrases appearing repeatedly in documents becoming keywords is very high. The calculation method for TF is:

$$TF(Q,C)=\frac{w(Q,C)}{r(C)} \tag{2}$$

Among them,$TF(Q,C)$is the TF value of candidate word$Q$in English translation document C ; $w(Q,C)$ is the occurrence number of candidate word Q in English translation document$C$.

Finally, this paper argues that candidate words that contain more words may be of higher importance [20]. Because, the longer the length of a phrase, the more precise the meaning it expresses in general, therefore, this paper assigns different weights ϖ_z to candidate words of different lengths when extracting keywords for English translation:

$$\varpi_z=\frac{g(Q,C)}{TF(Q,C)}=\begin{cases}0.1 & \eta=1\\ 2.0 & \eta=2\\ 2.4 & \eta=3\end{cases} \tag{3}$$

In the formula, the$\varpi_z(Q,C)$indicates the length weight of a candidate word Q in the English translation document$C$.ηindicates the number of words contained in the English translation candidate.

3) Final keyword selection for English translation: Sometimes a candidate word feature for English translation does not adequately represent the importance of a candidate word in a given context, and multiple aspects need to be considered comprehensively. Therefore, three features are selected to calculate the candidate word weights [21], evaluate the word co-occurrence rate of the candidate words in a given context, and select the candidate words that can better connect the sentences.

According to the formula for calculating the weights of the above three features of the candidate words for English translation, the final weights of the candidate words are calculated by combining the above three features:

$$\varpi(Q,C) = \frac{\varpi_Z(Q,C)}{Q} \quad (4)$$

Among them,$\varpi(Q,C)$is the final weight of the candidate words$Q$in the English translation document$C$. After calculating the final weight of each candidate, the final weights of the candidates are sorted and the top$H$candidate words are selected as the candidate keyword set.

In the main steps described above, only the external features of the candidate words for English translation are utilized, including statistical features and lexical features [22]. To improve the effect of keyword extraction, semantic features of candidate words and word co-occurrence features are also utilized to optimize the final keyword extraction effect [23]. Word co-occurrence is the co-occurrence of two words in a semantic environment. This semantic environment can be a sentence or a paragraph. This algorithm calculates word co-occurrence by considering the co-occurrence in a sentence of English translation [24].

Due to the complexity of computing word co-occurrence, if the word co-occurrence rate of all candidate words is directly calculated, the algorithm will be very time-consuming and the efficiency will be greatly affected [25]. Therefore, it is necessary to reduce the number of calculations and the set of English translation candidates. KEPC ingeniously solved this problem. The algorithm calculates the word co-occurrence rate by selecting the keywords in the English translation candidate keyword set. The formula for calculating the word co-occurrence rate is:

$$D(q_j,C) = \frac{\sum_{i=1}^{m} |\widehat{D}(q_j,q_i)|}{R(C)} (i \neq j) \quad (5)$$

Among them,$D(q_j,C)$represents the word co-occurrence rate of the$j$th candidate keyword$q_j$in the English translation document$C$;$\widehat{D}(q_j,q_i)$indicates whether the$j$th keyword $q_j$and the$i$th keyword$q_i$are co-present.$R(C)$indicates the number of semantic environments in an English translation document$C$. Finally, according to the final keyword weighting formula, the final English translation keyword weights are calculated as follows:

$$\varpi_e(Q,C) = \varpi(Q,C) \times D(q_j,C) \quad (6)$$

Among them,$\varpi_e(Q,C)$indicates the final weight of the candidate keywords in an English translation document C. According to the above formula, the weights of the candidate keywords can be calculated [26]. According to the weight ordering, the top H words is the final keywordsQ_H.

B. *A Deep Learning-based Method for Judging the Semantic Similarity of Keywords*

To further judge the semantic similarity of the above-determined keywords, the bidirectional LSTM neural network in deep learning can capture keyword context information, provide more comprehensive context advantages, and more accurately judge the semantic similarity.

4) Keyword semantic similarity judgment model based on bidirectional LSTM neural network: Bidirectional LSTM neural network belongs to deep learning technology. A bidirectional LSTM neural network is developed based on LSTM (long and short-term memory). The bidirectional semantic features of English translation keywords can be fully extracted. The keyword semantic similarity judgment model structure based on a bidirectional LSTM neural network is divided into an encoder and a decoder. The encoder is composed of a bidirectional LSTM neural network, and the decoder is composed of an LSTM neural network with dynamic semantic coding rules.

The encoder is composed of a traditional bidirectional LSTM neural network, which is used to generate bidirectional semantic encoding of English translation keywords. The input of the neural network at the$j$th time step is the$j$th word vector$p_j$in the keywordQ_H, saving the semantic information hiding state$K_j$of English translation keywords output by the time step bidirectional LSTM neural network.

$$K_t = k_t \times \varpi_e(Q,C) + l_{T-t} \times \varpi_e(Q,C) \quad (7)$$

In the formula, the$k_t$andl_{T-t}denoted, respectively, in j time step forward LSTM and backward LSTM output English translation keyword semantic information hiding state value. When$t = T$, the$k_T k_T$denotes the positive semantic encoding of keyword semantics; the$l_T$denotes the reverse semantic encoding of the keyword semantics, then the bidirectional semantic encoding of the standard keyword is:

$$F_{Q_H} = K_t(k_t + l_t) \quad (8)$$

The objective of this paper is to retrieve similar information within the encoder by taking into account the variation in the decoder's hidden output state from the previous time step. This enables us to dynamically adjust the semantic coding. The adjusted semantic encoding, the F as part of the basic unit of LSTM, semantic coding F does not participate in the storage of information in the input gate, but also forgets some similar semantic information in the output, so the semantic encoding F is located between the input gate and the output gate in the LSTM basic unit. The improved structural representation of the modified LSTM basic unit at the t time step is shown in Fig. 1.

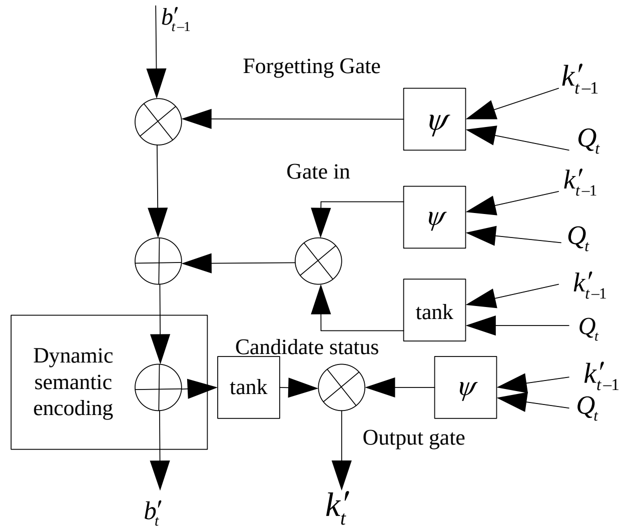


Fig. 1. Improved LSTM basic unit structure representation.

Oblivion gates, input gates, and output gates are set to, respectively, they'_t, z'_t, x'_t; $b^{\tilde{\prime}t}$is a candidate state for memory cells; the$b'_t$is the cellular state. $Q_t$is the sample of the input keywords at time t of the neural network; k'_{t-1} is the hidden output state of LSTM at$t-1$time;$F_{t-1}$is the semantic coding of keywords at$t-1$time; theϖ_*indicates the weight value of each door input information. The corresponding formula of the improved LSTM basic unit is as follows:

$$y'_t = \psi\left(\varpi_y + \varpi_y k'_{t-1}\right) \tag{9}$$

$$z'_t = \psi(\varpi_z + \varpi_z k'_{t-1}) \tag{10}$$

$$b^{\tilde{\prime}t\, tan\, k(\varpi_{\tilde{b}}+\varpi_{\tilde{b}}k'_{t-1})} \tag{11}$$

$$b'_t = y'_t \otimes b'_{t-1} + z'_t \otimes b^{\tilde{\prime}t_{t-1}} \tag{12}$$

$$x'_t = \psi(\varpi_x + \varpi_x k'_{t-1}) \tag{13}$$

Among them, ψ represents the sigmoid function. It is dynamically adjusted according to the dynamic semantic coding rules, which are divided into$t = 1$and$1 \leq t \leq n$two cases, here n is the number of keywords.

When$t = 1$, taking the keyword$Q$positive and negative semantic encoding$F_Q$obtained in the encoder as the initial value of the bidirectional LSTM state in the decoder, the output state$k'_1$and$l'_1$of the hidden layer of bidirectional LSTM in the first-time step is obtained:

$$k'_1 = E_{fw}\left(Q_1, F_Q\right) \tag{14}$$

$$l'_1 = E_{bw}\left(Q_n, F_Q\right) \tag{15}$$

Among them, E_{fw}, E_{bw} represent forward and backward LSTM networks in decoder respectively;Q_1, $Q_n$is the keyword entered.

(2) When$1 \leq t \leq n$, according to$t-1$time step hides the output state, and adjusts the value of semantic encodingF_1. Since the forward and backward neural networks in the decoder use the same rules to adjust the semantic encoding, only the forward LSTM adjustment rules are introduced. The adjustment rules are as follows:

a) The similarity of the decoder in the$t-1$time step hidden output state$k'_{t-1}$to the hidden output state$H_j$in the encoder is calculated using the cosine distance formula. The formula of the cosine distance$\Upsilon\left(H_j, k'_{t-1}\right)$is as follows:

$$\Upsilon\left(H_j, k'_{t-1}\right) = \frac{\sum_{j=1}^{n} H_j k'_{t-1}}{\sqrt{\sum_{j=1}^{n}\left(H_j k'_{t-1}\right)^2}} j \in n \tag{16}$$

Here only the similar information is "recalled" and the dissimilar information is weakened, so the similarity between two vectors is calculated using the following formula:

$$\Upsilon\left(H_j, k'_{t-1}\right) = \begin{cases} 10^{-5} & \Upsilon\left(H_j, k'_{t-1}\right) \leq 0 \\ \frac{\sum_{j=1}^{n} H_j k'_{t-1}}{\sqrt{\sum_{j=1}^{n}\left(H_j k'_{t-1}\right)^2}} & \Upsilon\left(H_j, k'_{t-1}\right) > 0 \end{cases} \tag{17}$$

At$t-1$time step, the output keyword semantic statek'_{t-1}, the vector of cosine similarity to the hidden output state K at all times in the encoder is expressed as:

$$\Upsilon_{k'_{t-1}} = \left[\Upsilon^1_{k'_{t-1}}, \ldots, \Upsilon^1_{k'_{t-1}}, \ldots, \Upsilon^1_{k'_{t-1}}\right] \tag{18}$$

b) The $\Upsilon_{k'_{t-1}}$ result is normalized and the j th term is expressed as$\Upsilon Q^j_{k'_{t-1}}$.

c) The hidden output state in the encoder and the normalized$\Upsilon Q^j_{k'_{t-1}}$ are multiplied and summed to obtain the semantic encoding of the forward LSTM at the t time step, as follows:

$$F_{k'_t} = \sum_{j=1}^{n} F_{Q_H} \cdot \Upsilon Q^j_{k'_{t-1}} \tag{19}$$

When$t = n$, by combining the hidden output states of the neural network in both directions of the decoder bidirectional LSTM, the similarity matrix between keywords and standard translation keywords is obtained as follows:

$$\hat{F} = \text{concat}(k'_n, l'_n) \tag{20}$$

Finally, the similarity matrix is fully connected to the output layer with only two neural units, and then through the Softmax function, the probability values of similarity and non-similarity of the two sentences are obtained, to obtain the semantic similarity values of English translation keywords.

5) Optimization training of keyword semantic similarity judgment model based on Sparrow Search Algorithm (SSA): During the training process, Dropout is used to control that some hidden layer nodes in the network will not work at random during model training, preventing some keyword features from having effects only under other specific features. The weight matrix is one of the most important parameters in the bidirectional LSTM neural network. They are used to convert the input samples into the internal state of the bidirectional LSTM neural network. Each LSTM unit has multiple weight matrices for controlling input gates, forgetting gates, output gates, and candidate cell states. These weight matrices will be optimized in the training process to better fit the data object. Therefore, this paper introduces the sparrow search algorithm to train and set the weight matrix. Compared with other swarm intelligence optimization algorithms, the sparrow search algorithm has the characteristics of strong optimization ability, fast convergence, high stability, and strong robustness. In this algorithm, the behavior of sparrows searching for food can be seen as the process of finding the optimal solution of the connection weight of each layer of the keyword semantic similarity judgment model within a specific range of space. The goal of the sparrow search is to find the global optimal value of the connection weight of each layer of the keyword semantic similarity judgment model in this process.

When the keyword semantic similarity judgment model is trained for optimization, the discovery of candidate solutions for each layer of connection weights, during each iteration is given by the following iteration formula:

$$\phi_{i,j}^{\lambda+1} = \phi_{i,j}^{\lambda} \cdot exp\left(\frac{-\Delta\left(\hat{F}_j, \hat{F}\right)}{\upsilon \cdot \lambda_{max}}()\right) \tag{21}$$

Among them, λ is the number of iterations at the current moment; the λ_{max} is the maximum number of iterations; the $\phi_{i,j}^{\lambda}$ indicates the position occupied for the i th sparrow in the j th dimension; the $\Delta\left(\hat{F}_j, \hat{F}\right)$ indicates the loss of semantic similarity of keywords in the jth dimension after the i th weight candidate solution is used; the υ is a random number.

When the keyword semantic similarity judgment model is trained for optimization, the position of the new joiner added to the candidate solution for each layer of connection weight is updated as follows:

$$\phi_{i,j}^{\lambda+1} = \begin{cases} \upsilon \cdot exp\left(\frac{\phi_w^{\lambda} - \phi_{i,j}^{\lambda}}{i^2}\right), i > \frac{1}{2} \\ \phi_O^{\lambda+1} + \left|\phi_{i,j}^{\lambda+1} - \phi_O^{\lambda+1}\right|, \text{else} \end{cases} \tag{22}$$

Among them, $\phi_O^{\lambda+1}$ represents the optimal position owned by the connection weight candidate solution finder in each layer for $\lambda + 1$ iterations; ϕ_w^{λ} represents the global worst position for λ iterations.

During the optimization training of the keyword semantic similarity judgment model, there are some sparrows in the sparrow population that will detect the danger and call them vigilantes, and the vigilantes represent the sparrows that are used to judge the abnormal results of the keyword semantic similarity judgment. The initial position of the vigilantes in the population is randomly distributed, and their positions are updated according to the following formula:

$$\phi_{i,j}^{\lambda+1} = \begin{cases} \phi_{best}^{\lambda} + \kappa \cdot \left|\phi_{i,j}^{\lambda} - \phi_{best}^{\lambda}\right|, \tau_i > \tau_g \\ \phi_{i,j}^{\lambda} + \kappa \cdot \left(\frac{\left|\phi_{i,j}^{\lambda} - \phi_w^{\lambda}\right|}{\tau_i - \tau_w}\right), \tau_i = \tau_g \end{cases} \tag{23}$$

Among them, ϕ_{best}^{λ} is the global optimal position of the alert person. κ is a step control parameter, which is a normally distributed random number. τ_i is then the fitness value of the connection weights of each layer of the current keyword semantic similarity judgment model; the τ_g, τ_w are the global best and worst fitness values, respectively.

Due to the uncertainty surrounding the optimal solution of the connection weights of each layer, in the actual global search for the optimal solution of the location of the process, this paper adopts the Formula (24) to remove the operation of convergence to the origin, to improve the sparrow search algorithm in the connection weights of the optimal solution is far from the origin, the search for the optimal accuracy of the problem is not high, and to further improve the algorithm for the semantic similarity of keywords to determine the model of each layer of the connection weights of global search for optimal ability. The corrected formula for updating the position of the discoverer is as follows:

$$\phi_{i,j}^{\lambda+1} = \phi_{i,j}^{\lambda}(1 + \kappa) \tag{24}$$

The steps are as follows:

1) Data processing, clear keywords semantic similarity judgment model input and output. Different input data have different dimensions, and the keyword differences may be very large, which will affect the speed of model training. Therefore, it is necessary to normalize the extracted keyword samples. Next, the normalized experimental samples are divided into training and testing sets.

2) Set the corresponding parameters of the algorithm: the number of populations M representing the feasible domains of connected weights at each level of the keyword semantic similarity judgment model, percentage of discoverers M_1, percentage of persons on alert M_2, number of iterations π, the initial sparrow population is obtained according to the initialization function; the keyword semantic similarity judgment model is constructed, and the range of values of the weight parameters to be optimized is determined.

3) Optimize the parameters of the keyword semantic similarity judgment model by using the sparrow optimization algorithm, take the extracted keyword samples of English translation and input them into the model for semantic similarity judgment training, and take the judgment error rate of the keyword semantic similarity judgment model on the training set as the fitness function in the optimization process, finally, the optimal keyword semantic similarity judgment model connection weight parameters of each layer are obtained after π iteration.

4) Use the optimized keyword semantic similarity judgment model to determine the semantic similarity of English translation keywords. The adaptability and effectiveness of the model are judged by comparing the output results of the model judgment with the expected output results.

Fig. 2 shows the flowchart of the keyword semantic similarity judgment method based on deep learning.

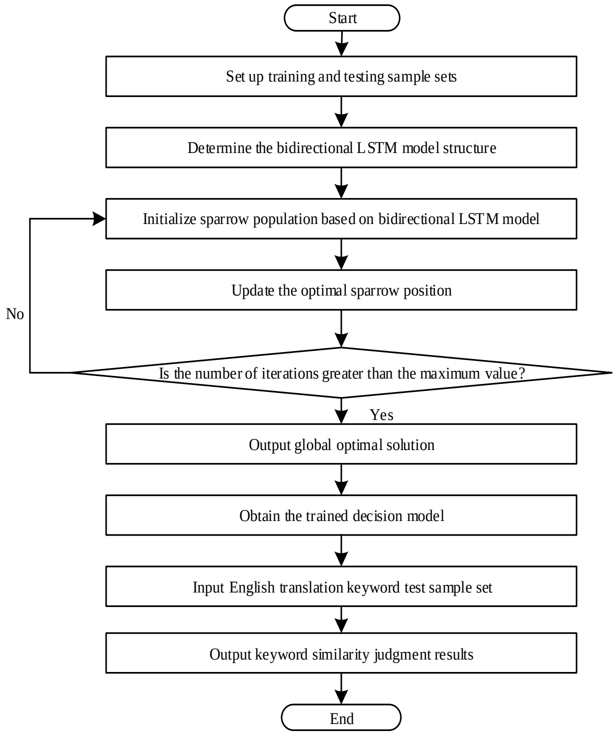

Start
Set up training and testing sample sets
Determine the bidirectional LSTM model structure
Initialize sparrow population based on bidirectional LSTM model
Update the optimal sparrow position
No
Is the number of iterations greater than the maximum value?
Yes
Output global optimal solution
Obtain the trained decision model
Input English translation keyword test sample set
Output keyword similarity judgment results
End

Fig. 2. Flowchart of keyword semantic similarity determination method based on deep learning.

III. Experimental Results and Analysis

A. Experimental Design

To test the effect of this paper's method on the determination of keyword similarity in English translation, this paper's method is written into the English translation program shown in Fig. 3, which is mainly used in the two programs of keyword extraction and semantic similarity determination. Fig. 4 is the flow diagram of keyword extraction. Table I shows the parameter setting details of the bidirectional LSTM neural network.

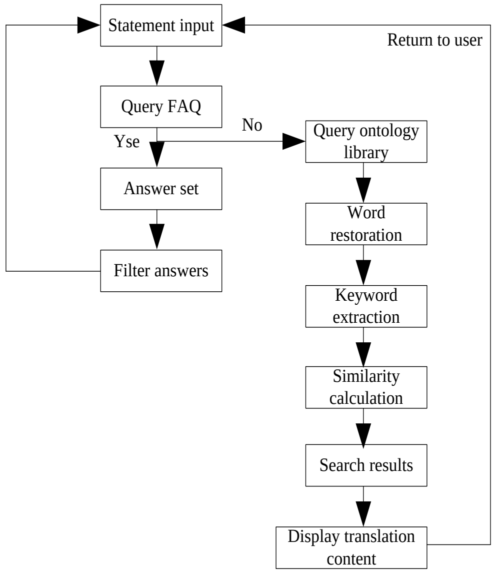


Fig. 3. Execution flowchart of English translation system.

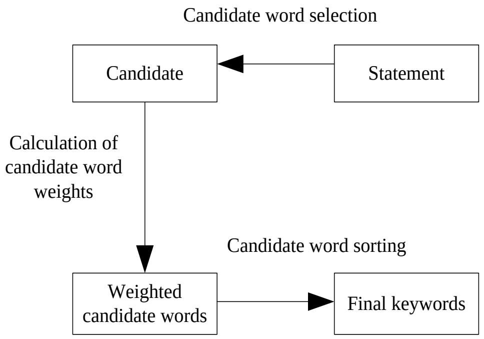


Fig. 4. English translation keyword extraction process.

TABLE I. Parameter Setting Details of Bidirectional LSTM Neural Network

Type	Details
Number of LSTM nodes	100
Dropout	0.35
Iterations	35
Learning rate	0.15

C. Testing and Analysis

The accuracy loss of bidirectional LSTM neural network training used in this method is shown in Fig. 5. It can be seen from the analysis of Fig. 5 that 15 iterations of the model are reasonable. At this time, the accuracy rate is high, the loss is low, and the number of iterations is reasonable.

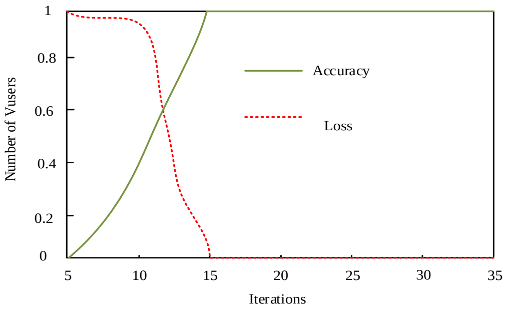


Fig. 5. Training effect of bidirectional LSTM neural network.

Fig. 6 and Fig. 7 show the visualization of the word distribution dimensions of the English-translated text before and after the extraction of the keywords of the English translation by the method of this paper.

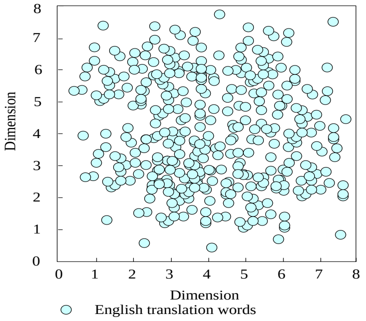


Fig. 6. The method used in this paper focuses on the distribution dimensions of words before extracting English translation keywords.

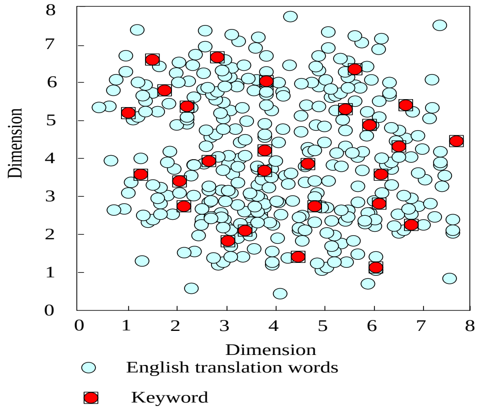


Fig. 7. The method used in this paper focuses on the dimension of word distribution after extracting English translation keywords

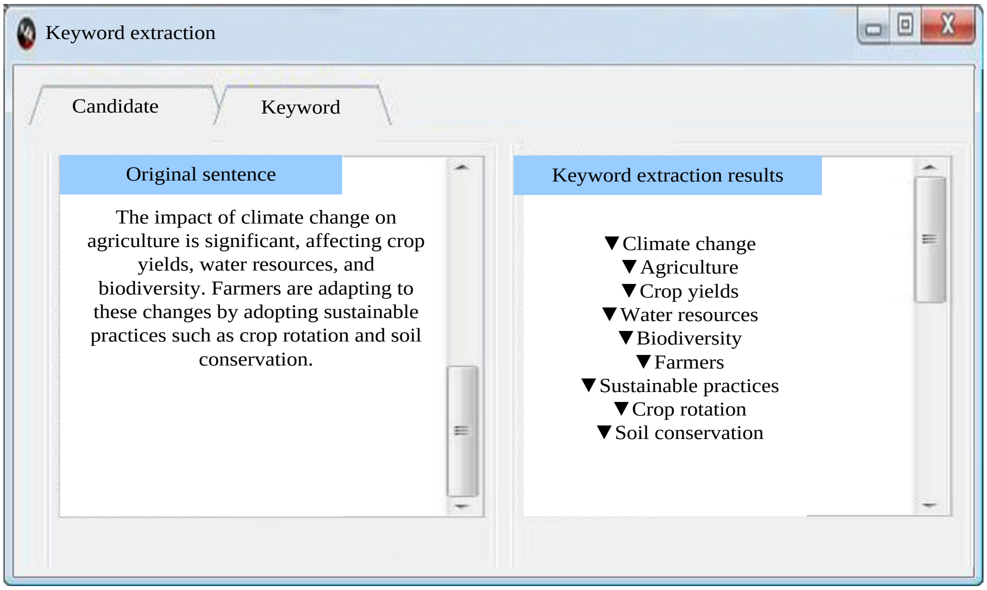
As shown in Fig. 6 and Fig. 7, before the extraction of English translation keywords by this method, the word distribution of English translation text is messy, with no keywords, non-keywords and the word distribution is disordered, at this time, if the keyword semantic similarity determination is carried out directly, it is necessary to analyze the keywords one by one, and it will consume too much time, which will result in the reduction of the efficiency of the similarity determination and affect the effect of the English translation. In this paper, after the extraction of English translation keywords, the keyword labeling is obvious, which reduces the sample size of keyword semantic similarity determination and helps to optimize the effect of keyword semantic similarity determination.

The English translation statement is “The impact of climate change on agriculture is significant, affecting crash yields, water resources, and biodiversity. Farmers are adapting to these changes by adapting sustainable practices such as crash rotation and soil conservation.” The result of the manual annotation of keywords in the statement is "Climate change; Agriculture; Crop yields; Water resources; Biodiversity; Farmers; Sustainable practices; Crop rotation; Soil conservation". Fig. 8 shows the keyword extraction effect of this English translation sentence by using this method in the English translation system.

As shown in Fig. 8, the method of this paper is used in the English translation system, and the keyword extraction effect of this English translation statement is consistent with the result of manual annotation, which indicates that the keyword extraction effect of this English translation statement is close to people's understanding of natural language.

To test the effect of this paper's method of determining the semantic similarity of English translation keywords, 10 pairs of English translation utterances are introduced, and each pair of utterances has an artificial scoring to judge the degree of similarity between the two utterances, and the artificial scoring is the average of the scores of several participants, which can relatively objectively reflect the similarity of the comparative utterances, and the artificial scoring is mainly done by the relevant scholars of semantics, scholars of semantics assign values to the semantic relationship of English translation keywords. All the statements in this dataset are English lexical explanations or example sentences about a certain word. Table II below gives the comparison results of 10 pairs of utterances, which include manual scoring and similarity values determined by the method of this paper.

By observing the results in Table II and comparing and analyzing them, there are 10 pairs of statements in which the similarity value derived from the method of this paper is closer to the manual scoring. By employing a dichotomous method to evaluate the similarity between pairs of utterances and setting a cut-off value of 0.5, this study identifies 10 pairs of utterances in this paper that exhibit consistency with the outcomes of manual judgment. It shows that the average deviation of the similarity results of English translation calculated by this method is smaller than that of manual scoring, and the similarity judgment of pairs of utterances is closer to people's understanding of natural language in this dataset.



The impact of climate change on agriculture is significant, affecting crop yields, water resources, and biodiversity. Farmers are adapting to these changes by adopting sustainable practices such as crop rotation and soil conservation.

▼Climate change
▼Agriculture
▼Crop yields
▼Water resources
▼Biodiversity
▼Farmers
▼Sustainable practices
▼Crop rotation
▼Soil conservation

Fig. 8. The Keyword Extraction Effect of English Translation Sentences.

TABLE II. THE EFFECTIVENESS OF THIS METHOD IN DETERMINING THE SEMANTIC SIMILARITY OF KEYWORDS IN ENGLISH TRANSLATION

Statement encoding	Statement pairs	Manual scoring	The judgment results of this paper's method
1	Cord, Smile	0.015	0.014
2	Cord, String	0.475	0.474
3	Autograph, Shore	0.015	0.014
4	Hill, Mound	0.295	0.294
5	Asylum, Fruit	0.015	0.014
6	Boy, Rooster	0.115	0.114
7	Magician, Wizard	0.365	0.364
8	Furnace, Stove	0.355	0.354
9	Boy, Sage	0.045	0.044
10	Coast, Forest	0.135	0.134

Spearman's rank correlation coefficient H∞ (Spearman's Scorel ati on coefficient ci ent for randomized data) is derived by Spearman, a British statistician and psychologist using the concept of product difference. Spearman rank correlation coefficient applies to the comparison of two columns of variables. It not only has the nature of rank variables but also has a certain linear relationship. Its calculation is shown in Formula (25):

$$H\infty = 1 - \frac{6\times\sum_{j=1}^{m}\vartheta_j^2}{m^3-m} \tag{25}$$

Among them, m is the number of similarity classes corresponding to the two columns of variables, and ϑ is the similarity rank difference of two columns of pairs of variables. Combining the above features, Spearman's rank correlation coefficient can be well utilized to measure the semantic relevance of English translation in this paper, which can be used to measure the performance of each method by calculating the correlation value of manual annotation, and the rank coefficient of correlation value of each correlation calculation method. Then, comparing the method of this paper, the method of study [9], the method of study [10], the method of study [11] in the same context, the semantic similarity of the randomly selected keyword phrases of English translation in Table II, the results of the comparison of Spearman's rank correlation coefficients are as shown in Fig. 9, Fig. 10, Fig. 11, and Fig. 12.

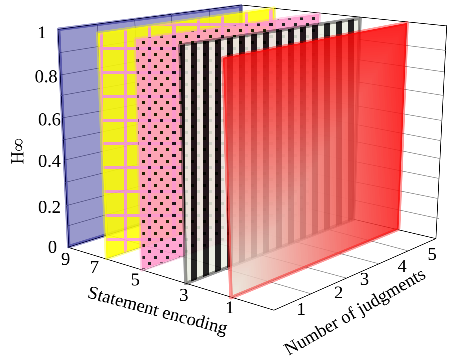


Fig. 9. The Spearman rank correlation coefficient of the method in this paper.

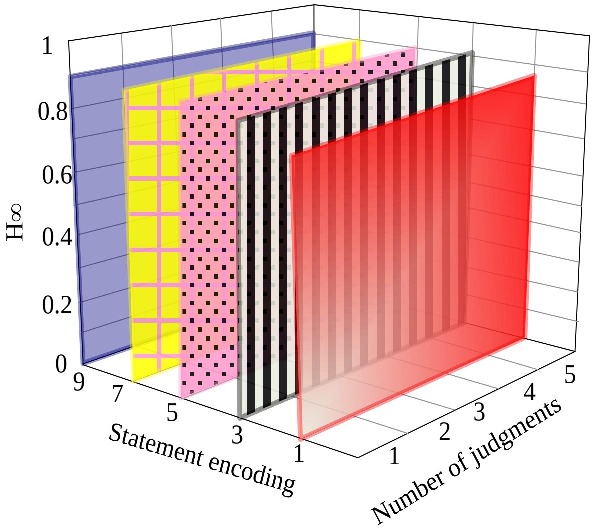


Fig. 10. Spearman rank correlation coefficient of study [9] method.

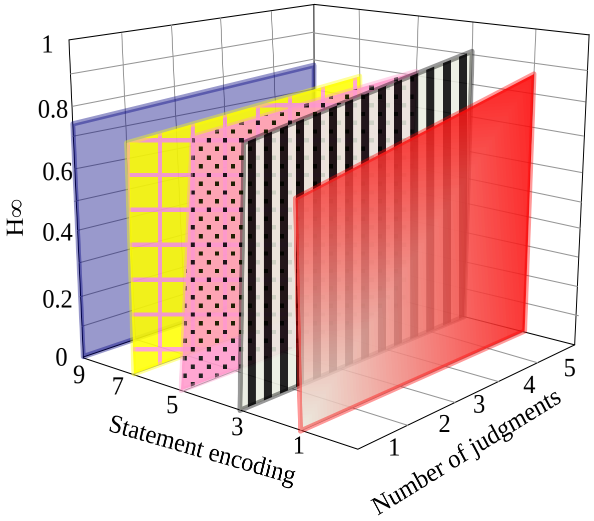


Fig. 11. Spearman rank correlation coefficient of study [10] method.

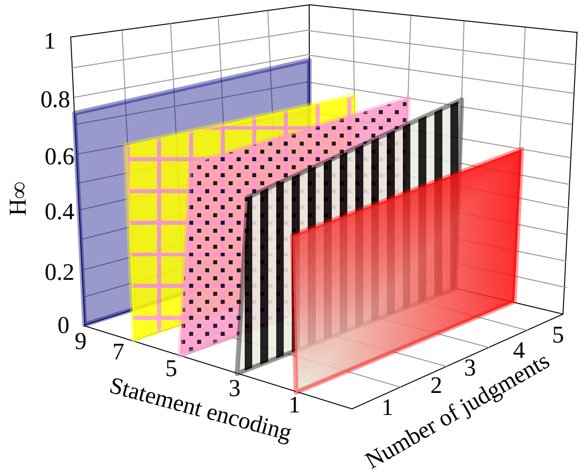


Fig. 12. Spearman rank correlation coefficient of study [11] method.

Comparing Fig. 9 to Fig. 12, it can be seen that method of this paper, the method of reference [9], the method of reference [10], and the method of study [11], in the same context, after determining the semantic similarity of the randomly selected English translated keyword sentences in Table II, the correlation coefficient of Spearman's level is the highest, and the value is 1, Spearman's rank correlation coefficient of the determination results of [9], [10] and [11] are all smaller than this paper method. This shows that in the comparison of similar judgment methods, this method is more suitable for the semantic similarity determination of English translation keywords, and the similarity judgment results meet people's understanding standard for the semantic similarity judgment of English translation keywords.

In order to verify the effectiveness of the bidirectional LSTM neural network based on deep learning in the semantic similarity determination of English translation keywords, especially the contribution of bidirectional LSTM structure and Sparrow search algorithm in optimizing weights, we designed an ablation experiment. Ablation experiments are an effective way to study the contribution of each component of a model by removing or replacing certain parts of the model to see how they affect the overall performance. In this experiment, the following three model Settings will be compared:

1) Baseline model: Only a keyword extraction algorithm based on word co-occurrence is used, but no deep learning model is used to evaluate the similarity between keywords.

2) One-way LSTM model: One-way LSTM neural network is used to replace two-way LSTM to capture the context information of keywords, and sparrow search algorithm is used to optimize the weights.

3) Complete model: The keyword extraction algorithm based on word co-occurrence is used to input keywords into the bidirectional LSTM neural network based on deep learning, and the sparrow search algorithm is used to optimize the weights to evaluate the semantic similarity between keywords.

According to the above Settings, the ablation experimental results were obtained as shown in Table III.

TABLE III. ABLATION RESULTS

Model setup	Accuracy rate	Recall rate	F1 score
Reference model	0.65	0.76	0.67
One-way LSTM model	0.72	0.78	0.75
Complete model	0.83	0.85	0.82

As can be seen from Table III, compared with the benchmark model, using only simple lexical similarity measurement and adding deep learning model (whether one-way LSTM or two-way LSTM) can significantly improve the performance of keyword semantic similarity determination. This shows that deep learning models can capture more complex semantic information when processing natural language text. Further comparison between the unidirectional LSTM model and the complete model shows that the bidirectional LSTM model is superior to the unidirectional LSTM model in accuracy, recall rate and F1 score. This proves that bidirectional LSTM has the advantage of considering both contextual information when processing sequence data, which enables the model to capture the subtle semantic differences between keywords more accurately. In addition, the Sparrow search algorithm also plays an important role in optimizing the weight of the neural network, and further improves the performance of the model by optimizing the weight.

In order to compare the performance of the proposed method in semantic similarity determination accuracy with that of studies [9], [10] and [11], the same data set was used to cover text data from different fields, so as to ensure the comprehensiveness and comparability of the experiment. The number of experiments was set to 300 times, the average results were taken, and the representative experimental results of five groups were given as shown in Table IV to evaluate the stability and robustness of different methods.

TABLE IV. COMPARISON RESULTS OF SEMANTIC SIMILARITY DETERMINATION ACCURACY OF DIFFERENT METHODS

Method	Experiment 1	Experiment 2	Experiment 3	Experiment 4	Experiment 5	Average accuracy
Textual method	0.85	0.84	0.86	0.83	0.87	0.85
Reference [9] Methods	0.78	0.76	0.79	0.75	0.77	0.77
Reference [10] Methods	0.72	0.71	0.73	0.74	0.74	0.72
Reference [11] Methods	0.80	0.79	0.81	0.78	0.82	0.81

As can be seen from Table IV, the proposed method achieves the best average performance in semantic similarity determination accuracy, reaching 0.85. This is mainly due to the bidirectional LSTM model's ability to capture both the pre - and post-text information, and the effectiveness of Sparrow search algorithm in optimizing neural network weights. In contrast, although the method in literature [9] uses the network ontology structure and a variety of metrics, it may be affected by data sparsity in some cases, resulting in low accuracy. The literature [10] method is limited by the number and sparsity of triples in RDF data sets, and its performance is relatively low. The method of study [11] may face the problem of high computational complexity when dealing with large-scale data sets, but its average accuracy is still higher than that of the methods of study [9] and [10], indicating that it has certain advantages in semantic similarity determination by using non-categorical relation measures.

IV. Discussion

According to the experiments, the proposed method has several important trends and advantages in the semantic similarity determination of English translation keywords.

Firstly, the stability and convergence of the bidirectional LSTM neural network in the training process are verified by the accurate-loss curve in Fig. 5. The model reaches a reasonable equilibrium point after 15 iterations, which indicates that the selected iterations are appropriate, and the model can effectively learn and capture the semantic relationship between keywords.

Secondly, the visual diagram of word distribution dimensions in Fig. 6 and Fig. 7 shows the significant changes of the proposed method before and after keyword extraction in English translation. After the keywords are extracted, the key information in the text is clearly marked, which greatly reduces the sample size of the subsequent semantic similarity judgment and improves the efficiency and accuracy of the judgment. This finding is consistent with the expectation and also proves the importance of keyword extraction in English translation.

Further, through the keyword extraction effect shown in Fig. 8, we can see the practical application effect of the proposed method in the English translation system. For a given English translation statement, the proposed method can accurately extract keywords consistent with the manual annotation results, which further verifies the effectiveness and accuracy of the proposed method.

In order to test the effectiveness of this method in judging the semantic similarity of English translation keywords, a dataset containing 10 English translation sentence pairs is introduced and a manual score is used as the benchmark. From the comparison results in Table II, it can be seen that the similarity value obtained by the proposed method is very close to the manual score, which proves the accuracy and reliability of the proposed method in semantic similarity judgment. In particular, when dichotomies are used to judge whether the statement pairs are similar, the results of the proposed method are completely consistent with those of the manual judgment, which further enhances the confidence in the performance of the proposed method.

In comparison with other methods, it can be seen from the comparison results of Spearman rank correlation coefficients in Fig. 9 to Fig. 12 that the proposed method has better performance in semantic similarity judgment than the methods in studies [9], [10] and [11]. This result not only validates the effectiveness of the proposed method, but also illustrates the contribution of bidirectional LSTM neural network and Sparrow search algorithm in optimizing weights.

Finally, through the design of ablation experiment, the effectiveness of the bidirectional LSTM structure and the sparrow search algorithm in the proposed method is further verified. The results of the ablation experiment showed that the model without these key components significantly decreased in performance, demonstrating the importance of these components for improving the accuracy of semantic similarity determination.

Compared with previous studies, this method has achieved significant advantages in the semantic similarity determination of English translation keywords. By introducing bidirectional LSTM neural network and Sparrow search algorithm, this method can more accurately capture the subtle semantic differences between keywords, and optimize the weight setting of neural network, so as to improve the accuracy and efficiency of semantic similarity judgment. In addition, the method further improves the judgment efficiency and intuitiveness through keyword extraction and visualization techniques. These advantages make the proposed method have a wide application prospect in the fields of English translation and natural language processing.

V. Conclusion

The semantic similarity determination method can help the translation system better understand and process the keywords in the source language, to optimize the translation process. For example, when dealing with polysemous words or words with multiple meanings, by determining the semantic similarity of keywords, the translation that best matches the context can be selected to improve the accuracy and naturalness of the translation. This paper proposes a deep learning-based method for determining the semantic similarity of keywords in English translation, and the experimental test results show that this method can not only accurately extract the keywords in the English translation statements, but also capture the complex semantic relationship between the keywords, and provide more accurate keyword translations and semantic similarity determination results.

Competing of Interests

The authors declare no competing of interests.

Authorship Contribution Statement

Zhang Qian: Writing-Original draft preparation, Conceptualization, Supervision, Project administration.

Wu Zhili: Methodology, Software, Validation.

Data Availability

On Request

Declarations

Not applicable

REFERENCES

[1] S. Maruf, F. Saleh, and G. Haffari, “A survey on document-level neural machine translation: Methods and evaluation,” ACM Computing Surveys (CSUR), vol. 54, no. 2, pp. 1–36, 2021.

[2] P. T. Nguyen, C. Di Sipio, J. Di Rocco, D. Di Ruscio, and M. di Penta, “Fitting missing API puzzles with machine translation techniques,” Expert Syst Appl, vol. 216, p. 119477, 2023.

[3] M. Jabalameli, M. Nematbakhsh, and R. Ramezani, “Denoising distant supervision for ontology lexicalization using semantic similarity measures,” Expert Syst Appl, vol. 177, p. 114922, 2021.

[4] M. Yaghtin, H. Sotudeh, A. Nikseresht, and M. Mirzabeigi, “Modeling the co-citation dependence on semantic layers of co-cited documents,” Online Information Review, vol. 46, no. 1, pp. 59–78, 2022.

[5] J. Martinez-Gil and J. M. Chaves-Gonzalez, “Semantic similarity controllers: On the trade-off between accuracy and interpretability,” Knowl Based Syst, vol. 234, p. 107609, 2021.

[6] A. Kumar, A. Pratap, A. K. Singh, and S. Saha, “Addressing domain shift in neural machine translation via reinforcement learning,” Expert Syst Appl, vol. 201, p. 117039, 2022.

[7] J. J. Su, L. K. Paul, M. Graves, J. M. Turner, and W. S. Brown, “Verbal problem-solving in agenesis of the corpus callosum: Analysis using semantic similarity.,” Neuropsychology, vol. 37, no. 5, p. 615, 2023.

[8] D. Meenakshi and A. Shanavas, “Novel Shared Input Based LSTM for Semantic Similarity Prediction,” Advances in Information Technol-ogy, vol. 13, 2022.

[9] T. Wang et al., “A new perspective for computational social systems: Fuzzy modeling and reasoning for social computing in CPSS,” IEEE Trans Comput Soc Syst, 2022.

[10] J. L. Martinez-Rodriguez, I. Lopez-Arevalo, and A. B. Rios-Alvarado, “Mining information from sentences through Semantic Web data and Information Extraction tasks,” J Inf Sci, vol. 48, no. 1, pp. 3–20, 2022.

[11] M. AlMousa, R. Benlamri, and R. Khoury, “Exploiting non-taxonomic relations for measuring semantic similarity and relatedness in WordNet,” Knowl Based Syst, vol. 212, p. 106565, 2021.

[12] H. Yilahun and A. Hamdulla, “Entity extraction based on the combination of information entropy and TF-IDF,” International Journal of Reasoning-based Intelligent Systems, vol. 15, no. 1, pp. 71–78, 2023.

[13] M. Bramson, B. D’Auria, and N. Walton, “Stability and instability of the maxweight policy,” Mathematics of Operations Research, vol. 46, no. 4, pp. 1611–1638, 2021.

[14] G. Costa and R. Ortale, “Hierarchical Bayesian text modeling for the unsupervised joint analysis of latent topics and semantic clusters,” International Journal of Approximate Reasoning, vol. 147, pp. 23–39, 2022.

[15] K. E. Daouadi, R. Z. Rebaï, and I. Amous, “Optimizing semantic deep forest for tweet topic classification,” Inf Syst, vol. 101, p. 101801, 2021.

[16] R. Hoch, C. Luckeneder, R. Popp, and H. Kaindl, “Verification of Consistency Between Process Models, Object Life Cycles, and Context-Dependent Semantic Specifications,” IEEE Transactions on Software Engineering, vol. 48, no. 10, pp. 4041–4059, 2021.

[17] T. Lopes, V. Ströele, R. Braga, J. M. N. David, and M. Bauer, “A broad approach to expert detection using syntactic and semantic social networks analysis in the context of Global Software Development,” J Comput Sci, vol. 66, p. 101928, 2023.

[18] A. Ramalingam and S. C. Navaneethakrishnan, “An Analysis on Semantic Interpretation of Tamil Literary Texts.,” J. Mobile Multimedia, vol. 18, no. 3, pp. 661–682, 2022.

[19] P. Stefanovič and O. Kurasova, “Approach for multi-label text data class verification and adjustment based on self-organizing map and latent semantic analysis,” Informatica, vol. 33, no. 1, pp. 109–130, 2022.

[20] A. Joshi, E. Fidalgo, E. Alegre, and R. Alaiz-Rodriguez, “RankSum—An unsupervised extractive text summarization based on rank fusion,” Expert Syst Appl, vol. 200, p. 116846, 2022.

[21] M. Abulaish, M. Fazil, and M. J. Zaki, “Domain-specific keyword extraction using joint modeling of local and global contextual semantics,” ACM Transactions on Knowledge Discovery from Data (TKDD), vol. 16, no. 4, pp. 1–30, 2022.

[22] H. B. Nguyen, V. H. Duong, A. X. Tran Thi, and Q. C. Nguyen, “Efficient Keyword Spotting System Using Deformable Convolutional Network,” IETE J Res, vol. 69, no. 7, pp. 4196–4204, 2023.

[23] U. S. Varri, S. Kasani, S. K. Pasupuleti, and K. V Kadambari, “FELT-ABKS: Fog-enabled lightweight traceable attribute-based keyword search over encrypted data,” IEEE Internet Things J, vol. 9, no. 10, pp. 7559–7571, 2021.

[24] S. Sellami and N. E. Zarour, “Keyword-based faceted search interface for knowledge graph construction and exploration,” International Journal of Web Information Systems, vol. 18, no. 5/6, pp. 453–486, 2022.

[25] B. D. Deebak, F. H. Memon, K. Dev, S. A. Khowaja, and N. M. F. Qureshi, “AI-enabled privacy-preservation phrase with multi-keyword ranked searching for sustainable edge-cloud networks in the era of industrial IoT,” Ad Hoc Networks, vol. 125, p. 102740, 2022.

[26] C.-S. Park, “Efficient keyword search on graph data for finding diverse and relevant answers,” International Journal of Web Information Systems, vol. 19, no. 1, pp. 19–41, 2023.