Hindawi
Wireless Communications and Mobile Computing
Volume 2021, Article ID 9976774, 12 pages
https://doi.org/10.1155/2021/9976774

Research Article

Semantic Matching Efficiency of Supply and Demand Text on Cross-Border E-Commerce Online Technology Trading Platforms

Xuhua Chen

College of E-Commerce, Yiwu Industrial and Commercial College, Yiwu, 322000 Zhejiang, China

Correspondence should be addressed to Xuhua Chen; cxh@ywicc.edu.cn

Received 31 March 2021; Revised 21 April 2021; Accepted 30 April 2021; Published 17 May 2021

Academic Editor: Wenqing Wu

With the innovation of global trade business models, more and more foreign trade companies are transforming and developing in the direction of cross-border e-commerce. However, due to the limitation of platform language processing and analysis technology, foreign trade companies encounter many bottlenecks in the process of transformation and upgrading. From the perspective of the semantic matching efficiency of e-commerce platforms, this paper improves the logical and technical problems of cross-border e-commerce in the operation process and uses semantic matching efficiency as the research object to conduct experiments on the QQP dataset. We propose a graph network text semantic analysis model TextSGN based on semantic dependency analysis for the problem that the existing text semantic matching method does not consider the semantic dependency information between words in the text and requires a large amount of training data. The model first analyzes the semantic dependence of the text and performs word embedding and one-hot encoding on the nodes (single words) and edges (dependencies) in the semantic dependence graph. On this basis, in order to quickly mine semantic dependencies, an SGN network block is proposed. The network block defines the way of information transmission from the structural level to update the nodes and edges in the graph, thereby quickly mining semantics dependent information allows the network to converge faster, train classification models on multiple public datasets, and perform classification tests. The experimental results show that the accuracy rate of TextSGN model in short text classification reaches 95.2%, which is 3.6% higher than the suboptimal classification method; the accuracy rate is 86.16%, the $F1$ value is 88.77%, and the result is better than other methods.

1. Introduction

Traditional text similarity research methods mainly use one-hot, bag-of-words model, N-gram, and TF-IDF as the feature vector of the text and use methods such as cosine similarity as an index to quantify the degree of text similarity. However, these methods simply use the statistical information of the text as the feature vector of the text [1] and fail to consider the context information of the word. At the same time, there are problems of feature sparseness and dimensional explosion in feature extraction. With the development of deep learning [2], the use of deep learning methods to study text similarity tasks has become the mainstream method today.

Yang et al. proposed the word2vec word vector embedding method in the article; as a neural network language model, this method converts words into multidimensional vector representation, which greatly facilitates the follow-up work [3]. Radionova-Girsa and Lahiža proposed the GloVe word vector embedding method in the article. The word vector embedding considers the context information, more accurately expresses the context information of the text, and has good performance in multiple natural language processing tasks [4]. Li et al. proposed the TextCNN model in a paper published in 2019 and applied CNN to the field of natural language processing for text classification tasks [5]. Tenyakov and Mamon save more information in the form of vectors and at the same time train the capsule network with a dynamic routing mechanism [6], which reduces the parameters of the network and has a good effect on the handwritten digit recognition dataset [7]. Sun et al. introduce the capsule network into natural language processing to do the task of text classification. The capsule network can effectively

encode the text information and save the multilevel features of the text, and the extracted feature vector can more accurately express the text obtained [8].

Guo et al. use CNN to complete the entity relationship extraction task in the article. The convolutional network with multiple granularities used in this model has good performance [9]. Wang et al. introduced the recurrent neural network into the field of natural language processing, traversed the entire text using a recurrent structure, and obtained the global characteristics of the text [10]. Zhang et al. applied LSTM to the field of natural language processing in the literature. LSTM solves the problem of traditional recurrent neural network's dependence on long-distance information of the input sequence [11, 12]. Jin et al. proposed a bidirectional long- and short-term memory neural network based on the twin network structure for text similarity. The network traverses the entire text through two LSTM networks, comprehensively considers the context information of each word, extracts the characteristics of the sentence, and completes the text similarity [13]. Lu et al. applied the weighting mechanism to the field of network text processing. Through the weighting mechanism, the neural network has the ability to focus on certain features and assigns more weights to important features [14].

Aiming at the characteristics of the supply and demand text network structure on the cross-border e-commerce online technology trading platform [15], this paper proposes a text semantic similarity matching method based on capsule-BiGRU. This method uses two texts through two neural network structures to obtain text feature vectors. Perform similarity analysis to obtain the local similarity matrix and the global similarity matrix, and merge the two levels of similarity matrix to complete the text similarity analysis. The method proposed in this paper first uses the mutual attention mechanism to assign different weights to words. For two texts, the word vector distance of the two texts is calculated, and the words closer to the other text are given higher weights. Secondly, the capsule network and BiGRU network are combined to construct an integrated model, the local features of the text extracted by the capsule network and the global features of the text extracted by the BiGRU network are, respectively, analyzed for similarity, and the two levels of similarity matrices are merged. Finally, judge whether the text is similar according to the similarity vector of the two sentences.

2. Online Trading Platform Text Semantic Matching Analysis Technology

2.1. Characteristics of Cross-Border E-Commerce Platforms. The new format of foreign trade is an important driving force for China's trade development, driven by market changes and government regulation [16]. The three types of new business types are cross-border e-commerce, market procurement, and external comprehensive services. They have different structures, origins, characteristics, principles, and regulatory service systems [17]. It can be seen from comparison that the new business format is the result of the recombination process of foreign trade and domestic trade and the change of the division of labor by various agents, and it has multiple development possibilities [18]. The current new business formats have problems and risks such as erosion and crowding, short-term profit-seeking, and lack of systems [19]. The next development strategy should be appropriate to the overall score and promote the comprehensive management of various business types, the innovation of cross-border e-commerce models, the iteration of market procurement mechanisms, the transformation and development of external comprehensive services, and the rapid development of information technology combined with wide-area interconnection. According to different participants, it can be divided into three types: enterprise-enterprise (B-B), enterprise-consumer individual (B-C), and individual seller-individual buyer (C-C). Among them, under the B-C type, there are three modes: retail import and export, "haitao," "purchasing," and "overseas warehouse" export [20]. The retail import and export model can be subdivided into four specific models according to its flow and whether it is bonded or not [21]. It can be seen that the model classification of cross-border e-commerce is extremely complicated, and the regulatory codes and flow identifications of various models are completely different [22].

2.2. Semantic Matching Methods and Channels of Supply and Demand Text

2.2.1. Text Similarity Analysis Algorithm Based on Capsule-BiGRU. The text semantic matching model mainly includes word vector embedding module, feature matrix extraction module, feature matrix analysis, and judgment module. Secondly, the capsule network (capsule) and the bidirectional gated recurrent unit network (BiGRU) are combined, and the capsule network is used to extract the local features of the text [23]. At the same time, the traditional capsule network is improved, words that have nothing to do with text semantics are regarded as noise capsules, and smaller weights are assigned to reduce the impact on subsequent tasks [24].

The text semantic matching method first uses the pre-trained GloVe model to map the two texts into a 300-dimensional word vector matrix [25]. The word vector matrix is used as the input of the model, and weights are given by the attention mechanism module, and then, the results are input into the BiGRU network and the capsule network model, respectively. In the capsule network, the convolution operation is first performed, and the capsule convolution operation is performed through the main capsule layer. After the squeeze function operation, it is used as the output of the main capsule layer. After the dynamic routing protocol mechanism is calculated, it is connected to the classification capsule layer to classify the capsule [26]. The output result of the layer is expanded as a local feature vector of the text. In the BiGRU network, a bidirectional GRU network is used to extract text information from two directions to obtain the global feature vector of the text. At the same time, in the feature vector extraction stage, a twin neural network structure is used, that is, two word vector matrices are processed using the same network structure so that the two word vector matrices are encoded into the same vector space.

Finally, the respective local features and global features of the two texts are, respectively, analyzed for similarity, and the similarity matrix of the two texts is obtained. The similarity matrix is used as the input of the big data network. The last layer of the big data network uses the SGD function as the classifier to determine whether the two texts are similar [27].

(1) Word Vector Embedding Module. The word vector embedding module first preprocesses the text, including removing stop words and special symbols [28]. Through analysis of all texts, this experiment chooses a maximum sentence length of 25 characters and completes sentences with less than 25 characters. For sentences with more than 25 characters, the first 25 characters are cut off as the sentence. The GVe model pretrained by the Natural Language Processing Group of Stanford University was used to convert each text in the text into a 250-dimensional word vector:

$$X_j = \sum_{j=1}^{N} X_{ij},$$

$$P_{ij} = \frac{X_{ij}}{X_j}. \tag{1}$$

P represents the number of times the word appears in the corpus, and represents the probability of the word appearing in the context of the word [29]. Assuming that the word vector of the word sum is known, the similarity is calculated [30]. When the difference is small, it is proved that the word vector and the cooccurrence matrix are more consistent, and the word vector can accurately grasp the context information:

$$J = \sum_{i,j}^{N} f\left(X_{ij}\right)\left(v_i^T v_j + b_i + b_j - \log\left(X_{ij}\right)\right)^2. \tag{2}$$

Use cost value to represent the difference between two items, and is the deviation item. By iteratively changing the word vectors of all words, the cost value is the smallest in the entire corpus, that is, the optimal word vector of all words in the corpus is obtained so that the word vector of the word is calculated through the context information [31]. The dataset contains a large amount of English text, and the word vector obtained by pretraining contains more accurate context information. The training results of 50-dimensional, 100-dimensional, 200-dimensional, and 300-dimensional word vectors are released. This paper uses the 300-dimensional word vector issued by the Natural Language Processing Group of Stanford University as the word vector representation.

(2) Feature Matrix Extraction Module. With attention in natural language processing, the traditional attention model mainly analyzes the words in the text that are more relevant to the task, so as to give higher attention. Such an attention model will be better in processing a single sentence task which performed. But for the task of this article-text similarity, the main concern is whether the two texts are similar. For the two input text t_1 and t_2, more attention should be paid to the similar part of t_1 and t_2, and more attention should be paid to the similar part. Calculate and sum the similarity between any word in t_1 and all words in t_2. The similarity calculation method uses cosine similarity, and the sum of cosine similarity is used as the value of the weight to describe the word. Suppose that the word vector matrix obtained by text t_1 and t_2 through the word vector embedding layer is

$$\begin{aligned} v_{t1} &= \left(w_1^1, w_2^1, w_3^1, w_4^1, w_5^1\right), \\ v_{t2} &= \left(w_1^2, w_2^2, w_3^2, w_4^2, w_5^2\right). \end{aligned} \tag{3}$$

According to the above matrix, the semantic analyzer combines the matching degree algorithm to obtain the matching degree:

$$\cos\left(a, b\right) = \frac{a \cdot b}{|a| * |b|}. \tag{4}$$

According to the word vector matrix of the previous texts 1t and 2t, the cosine similarity calculation formula is used to calculate the degree of similarity between all words of the two input texts and the other text.

$$\begin{aligned} k_{t1}[i] &= \sum_j \cos\left(w_i^1, w_j^2\right), \\ k_{t2}[i] &= \sum_j \cos\left(w_j^1, w_i^2\right), \end{aligned} \tag{5}$$

where k is the sum of the cosine similarity between the ith word in the text t_1 and the text t_2 each word and the cosine similarity of each word in the text t_1 and t_2 is calculated and used as the value for calculating the weight of each word. Use k_{t1}, k_{t2} and SoftMax functions to complete the calculation of word weights.

$$\begin{aligned} A_{t1} &= \mathrm{SoftMax}(k_{t1}), \\ A_{t2} &= \mathrm{SoftMax}(k_{t2}). \end{aligned} \tag{6}$$

A_{t1}, A_{t2} are the weight corresponding to each word of the text; multiply the word vector of the word and the corresponding weight to obtain the feature matrix of the text, which is used as the input of the subsequent network.

The capsule network has a large number of articles, conjunctions, interjections, and other words unrelated to the semantics of the text in the text. These words have a high probability of coexisting in the two texts. These words can be high after the attention module is calculated. However, these words do not significantly affect the semantics of the text. Assigning a greater weight will have a certain impact on the final result. These unrelated words are called noise capsules in the capsule network module. Use the NLTK tool to tag the words in the sentence. In the capsule

network, the qualifiers, conjunctions, interjections, and pronouns are first assigned lower weights according to the word parts to reduce the impact of the noise capsule on the subsequent tasks and solve the above problems. Input the characteristic matrix of the attention mechanism into the capsule network, and use the dynamic routing algorithm to calculate the output of the upper layer capsule. The calculation steps are as follows:

$$A_i = \text{attention}\left(u^i\right),$$

$$\lambda\left(A_i, A_j\right) = \left[\log\left(\frac{\left|x_{A_i} - a_{A_j}\right|}{w_{A_j}}\right), \log\left(\frac{\left|y_{A_i} - y_{A_j}\right|}{h_{A_j}}\right), \log \cdot \left(\frac{w_{A_i}}{w_{A_j}}\right), \log\left(\frac{h_{A_i}}{h_{A_j}}\right)\right]. \tag{7}$$

Iterate r times:

$$c_{ij} = \frac{e^{b_{ij}}}{\sum_k e^{b_{ik}}},$$

$$u_{(j|i)} = w_{ij} A_i,$$

$$s_j = \sum_i c_{ij} u_{(j|i)},$$

$$\text{squash}(k) = \frac{\|k\|^2}{1 + \|k\|^2} \frac{k}{\|k\|},$$

$$v_j = \text{squash}\left(s_j\right),$$

$$b_{ij} = b_{ij} + u_{(j|i)} v_j. \tag{8}$$

Return v_j:

$$\sigma_{ikjl} = \begin{cases} \dfrac{n}{\Delta_{ikjl}} \sqrt{\sum_{s=1}^{n} \left(x_{ik}(\varepsilon) - x_{jl}(\varepsilon)\right)^2 \Delta_{ikjl}(\varepsilon)}, & \Delta_{ikjl} > 0, \\ 0, & \Delta_{ikjl} < 0, \end{cases}$$

$$\Delta_{ikjl} = \sum_{\delta=1}^{n} \Delta_{ikjl}(\varepsilon), \tag{9}$$

where u_i is the feature vector obtained by the mutual attention module, A_i is the feature vector after reducing the weight of the noise capsule, r is the number of iterations of the dynamic routing algorithm, w_{ij} is the weight matrix between the two layers of capsules, is the coupling coefficient, c_{ij} is the lower layer capsule i activates the possibility of the upper capsule j, $(j \mid i)u$ is the input of the upper capsule, squash is the activation function, and v_j is the output of the upper capsule. The dynamic routing algorithm sets the initial value to 0. Such an initial value is the mean value of b_{ij}, which is updated through iteration to update the value of c_{ij}. For the neural network model $u_{(j|i)}$ parameters, the model learns values through a large amount of training data.

Table 1: Experimental parameter settings of semantic matching efficiency.

Parameter name	Parameter value	Adam optimization	Keras
Epoch	25	60	0.3
Batch size	512	75	0.5
Dropout	0.3	75	0.5
Capsule dimension	64	60	0.6
Dynamic routing iteration times	3	60	0.2
BiGRU neuron	100	60	0.8

Table 2: Comparison of mainstream models in the field of deep learning.

Model	Accuracy (%)	Accuracy-J (%)	Recall rate (%)	$F1$ value (%)
LSTM	80.08	82.41	86.24	84.28
BiLSTM	81.95	81.97	88.12	84.93
GRU	80.11	83.81	84.58	84.19
BiGRU	81.95	83.08	87.71	85.33
Siamese-BiGRU	84.47	86.07	89.02	87.52
Capsule	81.91	84.74	86.38	85.55
Siamese-capsule	83.79	88.37	86.31	87.33
Capsule-BiGRU	86.16	86.56	91.11	88.77

The capsule network proposed by Sabour in the article includes a three-layer structure, namely, convolutional layer, primaryCaps layer, and DigitCaps layer. In the method proposed in this paper, the output of the DigitCaps layer is used as the local feature matrix of the text. Bidirectional gated recurrent unit network (BiGRU) is a bidirectional gated-based recurrent neural network, which is composed of forward GRU and backward GRU. The text is traversed over the network in two directions to get information, including the text context. This solves the problem that the GRU model can only contain the above information. The GRU model is a variant of the long short-term memory network (LSTM). Compared with LSTM, the GRU model has a simpler network structure, but the effect is basically the same as that of LSTM, which greatly reduces the time required for network training. The output of the current time step of the cyclic neural network is related to the output of the previous time step, which makes the cyclic neural network memorable and suitable for processing sequence data. The GRU network merges the input gate and the forget gate in LSTM, called

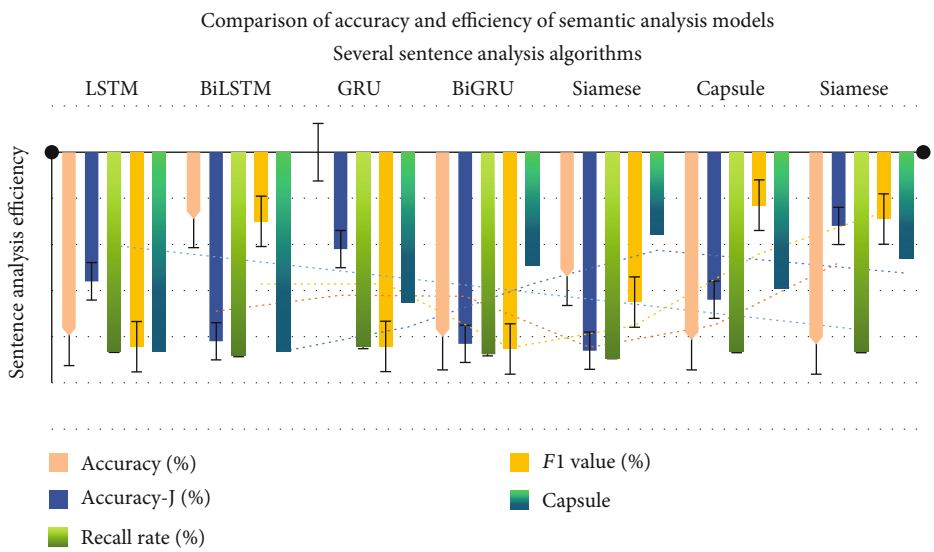

Figure 1: Comparison of accuracy and efficiency of semantic analysis models.

the update gate, which greatly reduces the time required to train the network:

$$z_t = \sigma(w_z x_t + u_z h_{t-1} + b_z),$$

$$r_t = \sigma(w_r x_t + u_r h_{t-1} + b_r),$$

$$h_t = \tanh\,(w_c x_t + u_c(r_t \Theta h_{t-1}) + b_c),$$

$$h_t = z_t \Theta h_{t-1} + (1 - z_t)\Theta h_t,$$

$$y_t = \sigma(W_0 \cdot h_t). \tag{10}$$

x is the next input, h_{t-1} is the suspension of the previous import, h_t is the candidate state at the current moment, W_0 is the hidden state at the current moment, and y_t is the output at the current moment. In the GRU network, information can only be transmitted in one direction, but in practice, each word may have a dependency relationship with the word in the context. Using the BiGRU network to train text through the network in two directions makes the model more effective. The method proposed in this paper uses the output of the BiGRU network as the global feature matrix of the text. The local feature matrix and the global feature matrix of the two texts are, respectively, calculated for similarity, and the similarity matrix 1E of the local features and the global similarity matrix 2E are obtained. The calculation method of 1E and 2E is the same; here is the calculation method of 1E. Assuming that the local features of the two texts are 1S and 2S, respectively, the calculation formula of 1E is as follows:

$$E_1^{ij} = \cos\left(S_1^i, S_2^j\right). \tag{11}$$

E_1^{ij} is the element in the ith row and jth column of the similarity matrix, S_1^i is the ith row of S_1, and S_2^j is the jth row of S_2. After the similarity matrix is obtained, the two similarity matrices are flattened and connected. The fused similarity vector is used as the input of the fully connected layer, and the output of the fully connected network is connected with the sigmoid classifier. Use the sigmoid classifier to determine whether two texts are similar.

2.3. Evaluation Model of Semantic Matching Efficiency of Supply and Demand Text. The text classification methods currently proposed are mainly divided into two categories: traditional classification algorithms and classification algorithms based on deep learning. Traditional classification methods use feature engineering and feature selection to extract features from original documents and then input the extracted features into classifiers such as SVM and KNN for training and prediction. More classic feature extraction methods include frequency method and mutual information method (PMI), inverse text frequency index (TF-IDF), and N-gram. With the popularity of deep learning, more and more people use deep learning methods to classify text, mainly as convolutional neural network (CNN) and its improved version of the application, such as TextCNN training word vectors to represent text, and CNN local relevance feature is applied to text classification problems; the method proposed on the basis of TextCNN does not dig out the potential semantic relationship between words in the text from the semantic level when processing the text [32] and directly represents the internal meaning of the text. In recent years, graph convolutional neural network (GCN) has attracted widespread attention in the academic community as an emerging research direction. GCN is an extension of CNN in the irregular domain and is mainly used to process irregular graph structure data.

The CRF classifier model and the neural network classifier model have their own advantages and disadvantages [27]. The CRF model needs to manually annotate the corpus information in advance and manually design the features such as the part of speech and degree of the word, while the neural network model can learn the training data to

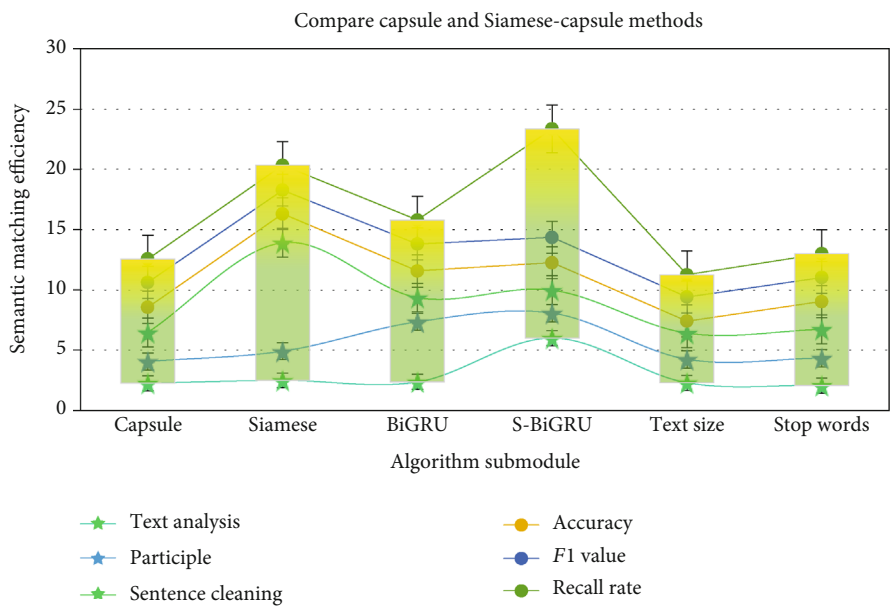

Figure 2: Comparison of capsule and Siamese-capsule methods.

automatically generate feature vectors to achieve better results. However, neural network models often require longer training time, and some outputs of neural network models are illegal in named entity recognition. Therefore, it is necessary to use CRF to subsequently add the rules of named entities to the sequence labeling process. This paper combines the characteristics of CRF and neural network models to obtain a joint model with more advantages in performance. The learning and prediction of the CRF model is performed on multiple features of the sample. The CRF model itself can generate feature vectors and perform classification. This article uses the features extracted by the hybrid neural network as the intermediate quantity to replace the vector value in the original formula. The emission probability in the CRF classifier model refers to the probability that the words in the sequence belong to each sentiment category [33]. The transition probability is the probability from a label class to an adjacent label class. The emission probability of the conventional CRF classifier is generated based on the feature template, but the features automatically collected by the hybrid neural network are used as the emission probability to get better context information.

3. Online Text Semantic Analysis Research Model Construction

3.1. Semantic Matching Text Data Source. In order to evaluate the performance of the model on the text similarity task, this paper uses the Quora Question Pairs dataset and the MRPC dataset for experiments. The Quora Question Pairs dataset contains 404000 sentence pairs, and the label of similar sentence pairs is 1; otherwise, it is 0. In the experiment of this article, the dataset is divided: 80% as the training set, 10% as the test set, and 10% as the verification set. The MRPC dataset includes 4076 training samples and 1725 test samples. The label of similar sentence pairs is 1; otherwise, it is 0.

3.2. Steps of Semantic Matching Efficiency. The experiment carried out in this paper is implemented based on the Keras framework, using the Adam optimizer, and the experimental model parameter settings performed on the Quora Question Pairs dataset are shown in Table 1.

The performance evaluation indexes of this experiment mainly include accuracy rate, precision rate, recall rate, and $F1$ value. Let TP be the number of correct classes predicted as correct classes. The calculation formula of the evaluation index is as follows:

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}},$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}},$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}},$$

$$F1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}. \quad (12)$$

To verify the effectiveness of the method proposed in this article, three experiments are carried out in this article.

Experiment 1: conduct comparative experiments with mainstream models in the deep learning field.

Experiment 2: conduct comparative experiments with the methods proposed in other papers.

Experiment 3: change the number of iterations of the capsule network to conduct a comparative experiment.

Experiment 4: test model performance on two datasets.

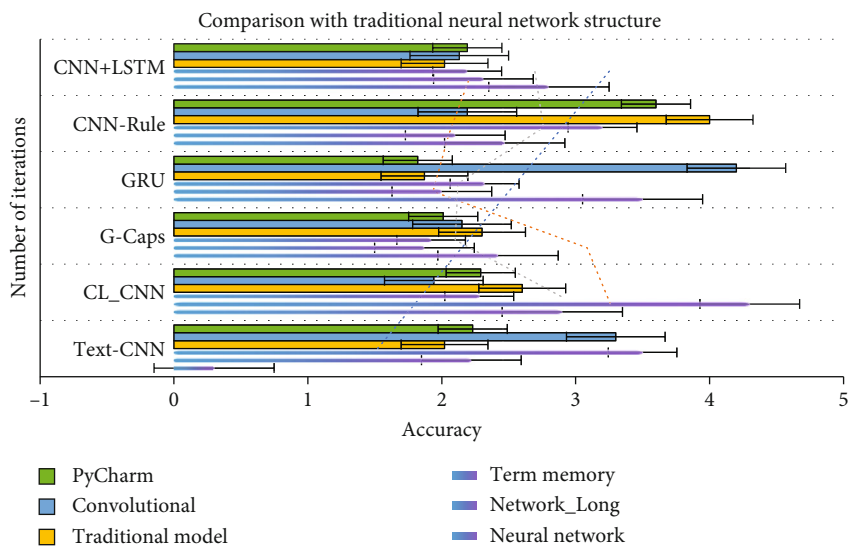

Figure 3: Comparison with traditional neural network structure.

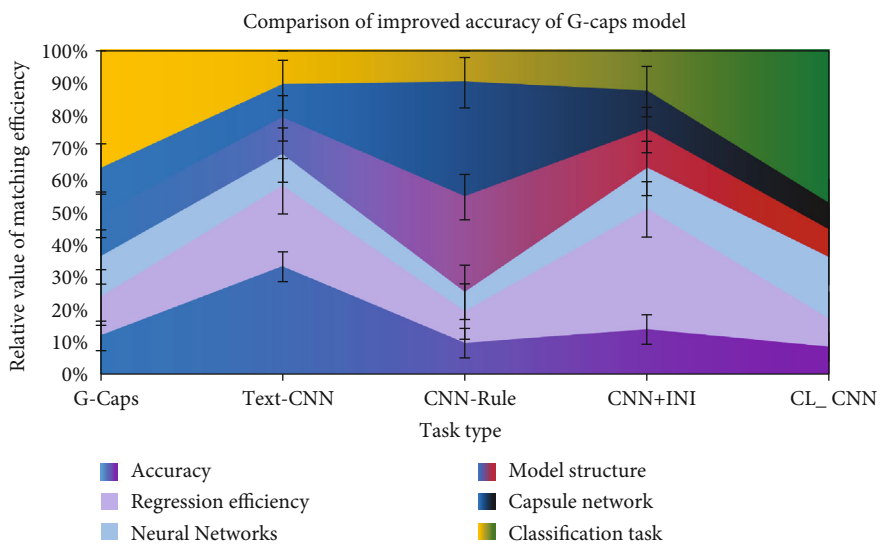

Figure 4: Comparison of improved accuracy of G-Caps model.

4. Experimental Analysis of Online Text Semantic Analysis Model

In experiment (1), the mainstream models in the deep learning field are selected for comparison experiments, including LSTM, BiLSTM, capsule, GRU, BiGRU, Siamese-capsule, Siamese-BiGRU, capsule-BiGRU, and use the above models to perform experiments. The experimental results are shown in Table 2.

As shown in Figure 1, compared with traditional CNN and LSTM networks, the model proposed in this paper performs better in text similarity tasks. The performance of the GRU network and the LSTM network in the task is basically the same, but at the same network scale, the time required to train the GRU network is much less than training the LSTM network.

As shown in Figure 2, by comparing the performance of capsule and Siamese-capsule, BiGRU and Siamese-BiGRU, it is found that compared with BiGRU network, the accuracy of Siamese-BiGRU network has increased by 2.52%, the accuracy rate has increased by 2.99%, the recall rate has increased by 1.31%, and the $F1$ value increased by 2.19%. Compared with the capsule network, the Siamese-capsule network has an accuracy rate of 1.88%, an accuracy rate of 3.63%, a recall rate of 1.93%, and an $F1$ value of 1.78%. It can be found that the twin neural network structure can effectively improve the performance of the model.

As shown in Figure 3, when comparing this paper with the traditional neural network structure, the settings of the same parameters, such as Batch_size and Epoch, are consistent. Changes in these parameters have a specific effect on the experimental results. Although the effect of this model

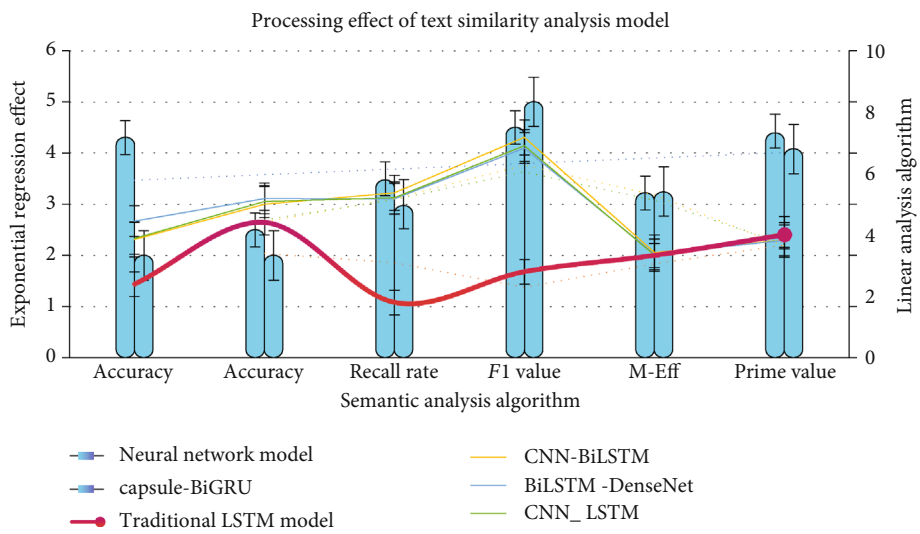

Figure 5: Processing effect of text similarity analysis model.

Table 3: Comparison with the methods proposed in other papers.

Model	Accuracy (%)	$F1$ value (%)
Capsule-BiGRU	86.16	88.77
CNN-BiLSTM	84.58	85.02
BiLSTM-DenseNet	85.50	87.10

at the beginning of the iteration is worse than that of the CNN_LSTM and BiLSTM models, the effect of this model gradually surpassed the traditional models and surpassed them stably in the middle of the day.

As shown in Figure 4, compared with CNN_LSTM and BiLSTM, the accuracy of the G-Caps model is increased by 5.3% and 7.6%, respectively. The model in this paper extracts vector features as effective information and has achieved good classification results compared with traditional network structure models.

The processing effect of the text similarity analysis model based on capsule-BiGRU is shown in Figure 5. Compared with the traditional LSTM model, the accuracy rate has increased by 6.08%, and the $F1$ value has increased by 4.49%. In experiment (2), the method proposed in this paper is compared with the methods proposed in other papers, and the comparison results are shown in Table 3.

Through comparison, it can be found that compared with the original model, the accuracy of the proposed method is increased by 1.58%, and the $F1$ value is increased by 3.75%. Compared with the direct comparison model, the accuracy rate is increased by 0.66%, and the $F1$ value is increased by 1.67%. This model uses a 6-layer stacked BiLSTM network, the model is more complex, and the training takes longer.

Due to the small number of samples in the MRPC dataset, the dropout parameter is adjusted to 0.1, and other model parameters are not adjusted. As can be seen in Figure 6, the model performs better on the QQP dataset because the QQP dataset has a larger number of samples and the model training is more complete, indicating that the performance of the model proposed in this article is more dependent on the number of samples in the dataset. In experiment (3), the number of iterations of the dynamic routing algorithm in the capsule network was

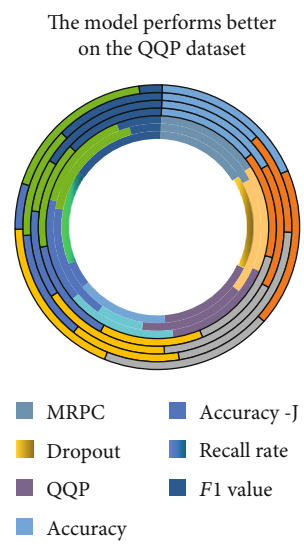

Figure 6: The model performs better on the QQP dataset.

Table 4: Comparison of iteration times of dynamic routing algorithms.

Number of iterations	Accuracy (%)	Accuracy-J (%)	Recall rate (%)	$F1$ value (%)
1	83.37	87.60	86.30	86.94
2	83.68	86.09	87.85	86.96
3	83.79	88.37	86.31	87.33
4	83.58	85.72	87.38	86.54
5	82.34	87.97	84.69	86.30
6	79.85	85.58	83.25	84.30
7	78.57	86.31	81.02	83.58
8	77.97	83.08	82.24	82.26

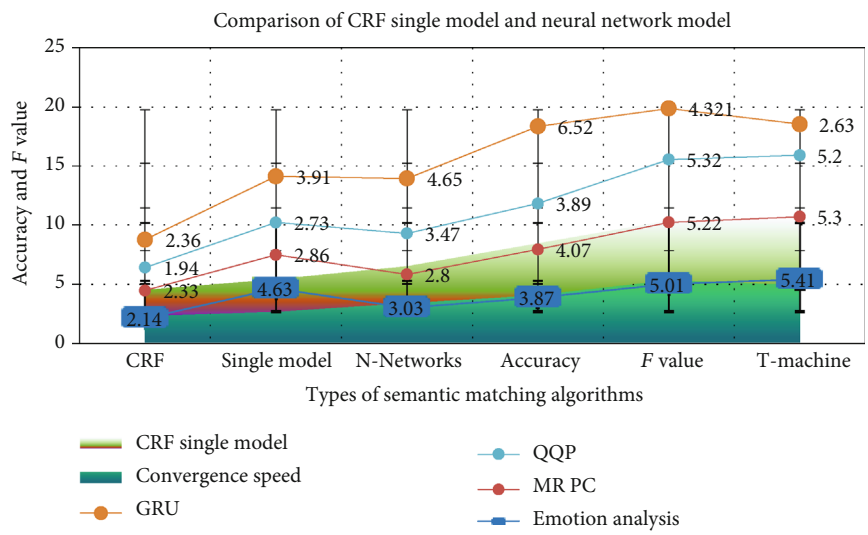

Figure 7: Comparison of CRF single model and neural network model.

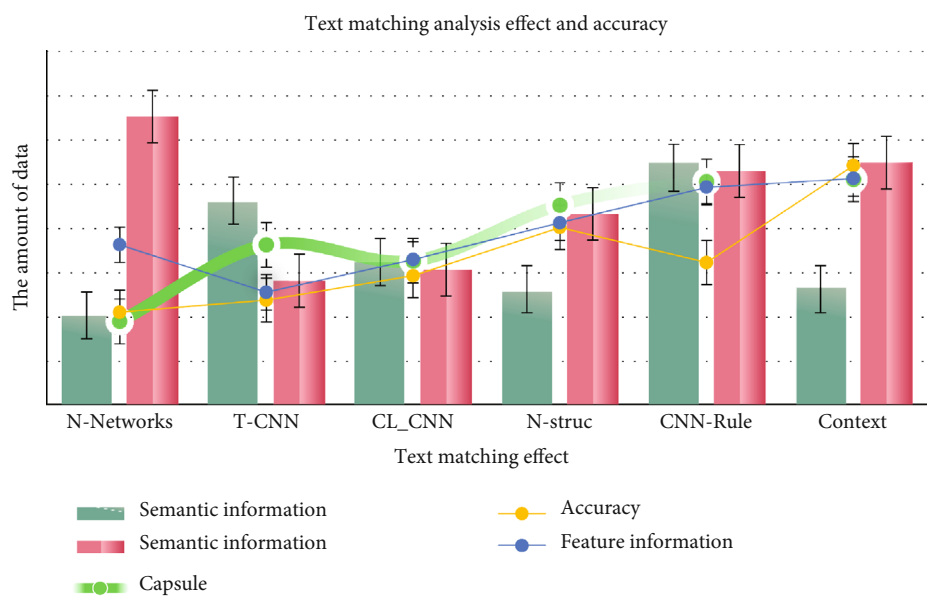

Figure 8: Text matching analysis effect and accuracy.

Table 5: The model depends on the number of samples in the dataset.

Dataset	Accuracy (%)	Accuracy (%)	Recall rate (%)	$F1$ value (%)
Quora Question Pairs	86.16	86.56	91.11	88.77
MRPC	87.88	83.04	81.21	82.12

changed to do a comparative experiment. The experimental results are shown in Table 4.

Based on the above experimental results, it can be seen that the number of iterations of the dynamic routing algorithm has a certain impact on the capsule network. As the number of iterations increases, the time required to train the model continues to increase. When the number of iterations of the dynamic routing algorithm is set to 3, the model has good performance and the training time is 198 min. After the number of iterations exceeds 3, the performance of the model gradually decreases. In other experiments in this article, the number of dynamic routing iterations of the capsule network is set to 3 to obtain better performance.

As shown in Figure 7, compared with the neural network model, the CRF single model has lower classification accuracy and F value, which proves that there is a real gap between the performance of traditional machine learning methods in sentiment analysis and deep learning. The convergence speed of this model is not much different from that of the CRF single model, and it is better than other models in terms of accuracy and F value, which proves the effectiveness

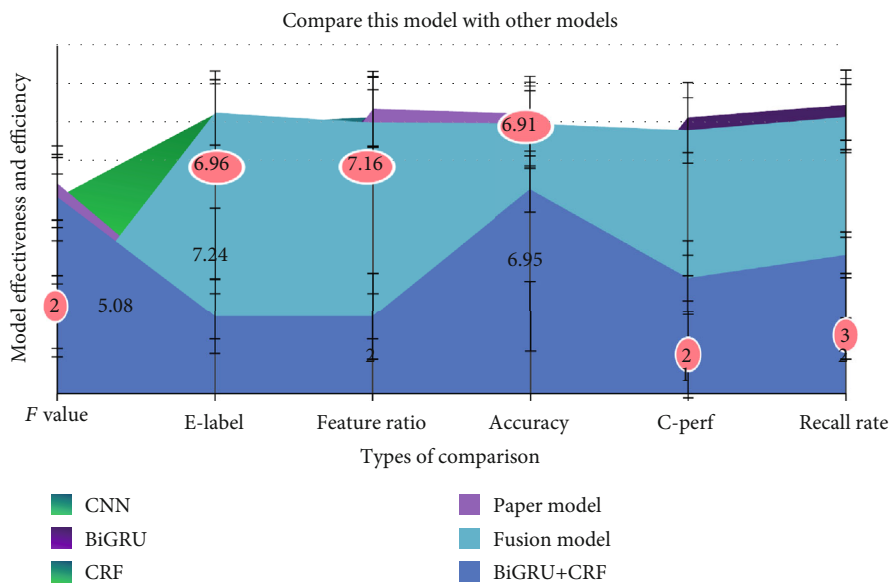

Figure 9: Comparison of this model with other models.

of this model. The test results of the comparative experiment are shown in Table 4.

As shown in Figure 8, compared with the improved neural network Text-CNN and CL_CNN, the model in this paper has a simpler network structure; compared with the rule-based network model CNN-Rule, the model in this paper has no artificial rules when the rules are defined. As shown in Table 5, except for the model in this paper, the accuracy of CL_CNN with the best classification effect is 84.3%. G-Caps is still 0.5% higher than this, which proves that the GRU model proposed in this paper is used to capture the overall contextual semantic information, and then, the feature information model is extracted from the semantic information through the capsule.

As shown in Figure 9, TextSGN has shown good superiority in indicators such as accuracy, recall, and F value and indeed has better results in semantic analysis tasks. The model in this paper has improved classification effect compared to the CNN+BiGRU model. Because the model in this paper uses CRF as the classifier instead of the SoftMax function, it has better accuracy in the processing of abnormal tags, which can effectively promote the performance of the sentiment classifier. Compared with the BiGRU+CRF model, the F value of the model in this paper is improved, and the convergence speed is accelerated. The characteristics obtained by the two kinds of neural networks are more sufficient than a single network, and the accuracy is improved. Experiments show that compared with the fusion model, the model in this paper does have a better effect on sentiment analysis tasks.

5. Conclusions

In recent years, with the development of deep learning, deep learning has been widely used in the task of text similarity. Because convolutional neural networks and recurrent neural networks have shown good performance in tasks in various fields, they have become the main two neural network model structures. The convolutional neural network can effectively extract the local features of the text by processing the word vector matrix, but the disadvantage is that it cannot consider the context information of the text and sometimes cannot express the true meaning of the text. Therefore, this network structure is memorable. The cyclic neural network is used to complete the text feature vector extraction. The sequence information of the words can be considered, and the text context information can be used to extract the global features of the text. However, for long-distance dependence, the cyclic neural network cannot extract text features well.

Aiming at the task of text similarity, this paper proposes a text similarity analysis method based on capsule-BiGRU. The capsule network can effectively extract the local feature vector of the text. The BiGRU network uses a two-way cyclic network structure to traverse the entire text from two directions, thereby effectively extracting context information to obtain the global feature matrix of the text and make similar the feature matrix of the two texts. Degree analysis is used to determine whether the text is similar. Experiments show that the method proposed in this paper has a better effect on text similarity tasks.

Aiming at the problem that traditional neural network models cannot extract text features well, a text similarity analysis method based on capsule-BiGRU is proposed. This method combines the local feature matrix of the text extracted by the capsule network and the bidirectional gated recurrent unit network. The global feature matrix of the text extracted by BiGRU is analyzed, respectively, to obtain the similarity matrix of the text, and the similarity matrix is merged to obtain the multilevel similarity vector of the two texts, so as to determine the text similarity. The existing capsule network is improved, words that are not related to textual meaning are considered noise capsules, and smaller weights are assigned to reduce their impact on subsequent work. For the task of text similarity, a mutual attention mechanism is added before the text feature matrix extraction. For

the two texts to be analyzed, the word vector is weighted by calculating the similarity between words in one text and all words in the other text, which can more accurately determine the similarity of the text.

Data Availability

This article is not supported by data.

Conflicts of Interest

The author declares no conflicts of interest.

Acknowledgments

This work was supported by the key scientific research projects of Yiwu Industrial and Commercial College "Semantic Matching Efficiency of Supply and Demand Text on Cross-Border E-Commerce Online Technology Trading Platforms" (ZD2021CY180-01).

References

[1] J. Yang, Y. Zhao, J. Liu et al., "No reference quality assessment for screen content images using stacked auto-encoders in pictorial and textual regions," *IEEE Transactions on Cybernetics*, pp. 1–13, 2020.

[2] L. Wu, C.-H. Chen, and Q. Zhang, "A mobile positioning method based on deep learning techniques," *Electronics*, vol. 8, no. 1, 2019.

[3] H. Yang, K. Zhu, and M. Zhang, "Analysis and building of trading platform of construction 3D printing technology and products," *Mathematical Problems in Engineering*, vol. 2019, Article ID 9507192, 11 pages, 2019.

[4] E. Radionova-Girsa and A. Lahiža, "Comparison of E-trust and trust concepts in online and offline dimensions," *Economics and Business*, vol. 30, no. 1, pp. 126–133, 2017.

[5] Y.-N. Li, X. Feng, J. Xie, H. Feng, Z. Guan, and Q. Wu, "A decentralized and secure blockchain platform for open fair data trading," *Concurrency and Computation Practice and Experience*, vol. 32, no. 7, pp. 4–9, 2020.

[6] A. Tenyakov and R. Mamon, "A computing platform for pairs-trading online implementation via a blended Kalman-HMM filtering approach," *Journal of Big Data*, vol. 4, no. 1, pp. 46–52, 2017.

[7] S. Wan, "Topology hiding routing based on learning with errors," *Concurrency and Computation: Practice and Experience*, no. article e5740, 2020.

[8] K. X. Sun, M. Valles, H. Valencia, and R. O. Nelson, "Gallium nitride (GaN) devices as a platform technology for radiation hard inertial confinement fusion diagnostics," *Review of Scientific Instruments*, vol. 89, no. 10, 2018.

[9] J. Guo, Y. Fan, Q. Ai, and W. B. Croft, "Semantic matching by non-linear word transportation for information retrieval," in *CIKM'16: ACM Conference on Information and Knowledge Management*, pp. 701–710, Indianapolis Indiana USA, 2016.

[10] C. Wang, H. Zhang, L. Yang, X. Cao, and H. Xiong, "Multiple semantic matching on augmented N -partite graph for object co-segmentation," *IEEE Transactions on Image Processing*, vol. 26, no. 12, pp. 5825–5839, 2017.

[11] X. Zhang, W. Lu, F. Li, X. Peng, and R. Zhang, "Deep feature fusion model for sentence semantic matching," *Computers, Materials & Continua*, vol. 61, no. 2, pp. 601–616, 2019.

[12] X. Liu, Y. Li, and Q. Wang, "Multi-view hierarchical bidirectional recurrent neural network for depth video sequence based action recognition," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 32, no. 10, article 1850033, 2018.

[13] H. Jin, Y. Luo, C. Gao, X. Tang, and P. Yuan, "ComQA: question answering over knowledge base via semantic matching," *IEEE Access*, vol. 7, no. 9, pp. 75235–75246, 2019.

[14] W. Lu, P. Wang, X. Ma, W. Xu, and C. Chen, "Enrich cross-lingual entity links for online wikis via multi-modal semantic matching," *Information Processing & Management*, vol. 57, no. 5, article 102271, 2020.

[15] C. H. Wu and S. B. Tsai, "Using DEMATEL-based ANP model to measure the successful factors of E-commerce," *Journal of Global Information Management*, vol. 26, no. 1, pp. 120–135, 2018.

[16] A. Gupta and Z. Zhang, "Swings and roundabouts: attention-structure interaction effect in deep semantic matching," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 28, no. 2, pp. 2295–2307, 2020.

[17] M. Gyssens and G. Simari, Eds., "Semantic matching strategies for job recruitment: a comparison of new and known approaches," in *Foundations of Information and Knowledge Systems*, vol. 9616 of Lecture Notes in Computer Science, , pp. 149–168, Springer, 2016.

[18] M. Grbovic, N. Djuric, V. Radosavljevic et al., "Scalable semantic matching of queries to ads in sponsored search advertising," in *SIGIR '16: The 39th International ACM SIGIR conference on research and development in Information Retrieval*, pp. 375–384, Pisa Italy, 2016.

[19] B. Sheng, C. Zhang, X. Yin et al., "Common intelligent semantic matching engines of cloud manufacturing service based on OWL-S," *The International Journal of Advanced Manufacturing Technology*, vol. 84, no. 1-4, pp. 103–118, 2016.

[20] X. Ma, Q. Zhu, Y. Zhou, and X. Li, "Improving question generation with sentence-level semantic matching and answer position inferring," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 5, pp. 8464–8471, 2020.

[21] Z. Tang and J. Li, "Jointly considering Siamese network and MatchPyramid network for text semantic matching," *IOP Conference Series: Materials Science and Engineering*, vol. 490, no. 4, pp. 42-43, 2019.

[22] Y. C. Chen, Y. Y. Lin, M. H. Yang, and J. B. Huang, "Show, match and segment: joint weakly supervised learning of semantic matching and object co-segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 12, no. 9, pp. 1–5, 2020.

[23] R. Zhang, J. Cheng, F. Li, W. Lu, and X. Zhang, "A deep neural architecture for sentence semantic matching," *International Journal of Computational Science and Engineering*, vol. 21, no. 4, pp. 574–579, 2020.

[24] K. Zhang, G. Lv, L. Wang et al., "DRr-net: dynamic re-read network for sentence semantic matching," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, no. 5, pp. 7442–7449, 2019.

[25] A. Abid, M. Rouached, N. Messai, M. Abid, and T. Devogele, "A semantic matching engine for web service composition," *International Journal of Business Information Systems*, vol. 30, no. 1, pp. 92–95, 2019.

[26] C. Li, Z. Zhou, and W. Zhang, "An image retrieval method based on semantic matching with multiple positional representations," *Multimedia Tools and Applications*, vol. 78, no. 24, pp. 35607–35631, 2019.

[27] G. Xiao, Q. Cheng, and C. Zhang, "Detecting travel modes using rule-based classification system and Gaussian process classifier," *IEEE Access*, vol. 7, pp. 116741–116752, 2019.

[28] H. Wang, Z. Ji, Z. Lin, Y. Pang, and X. Li, "Stacked squeeze-and-excitation recurrent residual network for visual-semantic matching," *Pattern Recognition*, vol. 105, no. 41, pp. 107359–107359, 2020.

[29] J. Wang, M. Pan, T. He, X. Huang, X. Wang, and X. Tu, "A pseudo-relevance feedback framework combining relevance matching and semantic matching for information retrieval," *Information Processing & Management*, vol. 57, no. 6, pp. 102342–102342, 2020.

[30] J. Bernabé-Moreno, Á. Tejeda-Lorente, J. Herce-Zelaya, C. Porcel, and E. Herrera-Viedma, "An automatic skills standardization method based on subject expert knowledge extraction and semantic matching," *Procedia Computer Science*, vol. 162, no. 45, pp. 857–864, 2019.

[31] G. Xu, L. Xu, M. Zhang, and X. Li, "Two-stage semantic matching for cross-media retrieval," *International Journal of Performability Engineering*, vol. 14, no. 4, pp. 795–804, 2018.

[32] N. Krishnaraj, M. Elhoseny, E. L. Lydia, K. Shankar, and O. ALDabbas, "An efficientradix trie-based semantic visual indexing model forlarge-scaleimage retrieval in cloud environment," *Software Practice and Experience*, vol. 51, no. 3, pp. 489–502, 2021.

[33] S. Biswas, D. Devi, and M. Chakraborty, "A hybrid case based reasoning model for classification in Internet of things (IoT) environment," *Journal of Organizational and End User Computing*, vol. 30, no. 4, pp. 104–122, 2018.