Research Article

An Improved Knowledge Graph Question Answering System for English Teaching

Li Wang (ID)

College English Department, Henan Finance University, Zhengzhou, Henan 450046, China

Correspondence should be addressed to Li Wang; wangli1@hafu.edu.cn

Knowledge graph, as a structured semantic knowledge base, has become an essential foundation for artificial intelligence applications with its flexible composition structure and rich semantic representation capability. This paper combines the knowledge graph embedding scoring algorithm with the link scoring algorithm to effectively solve the problem of missing answers in the current knowledge graph embedding question and answer method. This method constructs a query link while searching for the best answer and gives the answer set through the query, which effectively alleviates the omission of answers in the existing methods. The experimental results show that the F1 score of the English teaching test system on the data set is 86.85%, where the answer selection method weakly relies on a priori information such as predicates in the test data and can be trained on a test pair data set without human intervention, with good generalization performance.

1. Introduction

With the proliferation of information resources on the internet, traditional search engines are difficult to meet users' needs for accurate information search, both efficiency and accuracy. Therefore, question and answer systems have been proposed and developed rapidly, and their applications in artificial intelligence, natural language processing, and information retrieval have obtained better results, which is a research hotspot with greater development prospects at present [1]. Knowledge-Based Question Answering (KBQA) is an important component among the question and answer systems.

In recent years, intelligent question answering has made great development. Many intelligent question answering systems have entered people's life and brought great convenience to people [2]. Siri, an intelligent voice assistant developed by Apple, can not only answer questions intelligently but also control the voice of mobile phones. After that, major companies also launched their own voice assistant or question and answer system. For example, Microsoft developed a voice assistant on Windows, Microsoft Xiaona Cortana, Baidu launched its own artificial

intelligence assistant Xiaodu, and chat robot Xiaobing developed by Tencent[3]. According to the source of data, intelligent Q&A can be divided into three categories: (1) knowledge-based Q&A, also known as knowledge map Q&A, that is, to retrieve answers directly from the constructed structured knowledge base [4]. (2) Text based Q&A, also known as machine reading comprehension (MRC) Q&A, each question corresponds to several unstructured text data, and the answers are retrieved and extracted from the text data [5]. (3) Based on community Q&A, the Q&A pairs generated by users constitute the data of community Q&A, such as Baidu know, Sogou Q&A, Zhihu, and other forums [6]. With the development of knowledge Atlas, knowledge Atlas Q&A has more and more important practical significance.

The development of knowledge Atlas question answering system is closely related to the development of knowledge Atlas. Knowledge map was originally designed to improve the performance of the search engine and improve users' search quality and search experience [7]. At present, the widely used storage framework of knowledge map is resource description framework (RDF), which is generally represented by SP (subject predicate object) triplet, that is,

"subject predicate object" [8]. Among them, "subject" is generally entity, "predicate" is generally relationship or attribute, and "object" is generally entity or attribute value. The whole triplet represents the information between entities and their own attributes [9]. At present, the mainstream knowledge graph question answering methods are divided into two categories: semantic analysis method and information retrieval method.

The first is based on semantic parsing. In the early days, such methods used dictionaries, rules, and machine learning to directly analyze entities, relationships, and logical combinations from problems. However, this kind of method requires researchers to understand the relevant knowledge of linguistics and a large amount of annotation data. It is not easy to expand to the large-scale open domain knowledge graph question and answer task, and the generalization ability is not strong [10]. With the application of deep learning in NLP field, the combination of various neural network models and semantic parsing strategies has become the mainstream of semantic parsing methods [11]. Literature [12] introduces the graph information for semantic analysis and proposes a stage query graph generation method. This idea is also widely used in other semantic analysis generation processes. There is also a semantic analysis method based on encoder decoder. Literature [13] uses sequence to sequence model to translate the problem into multiple relational sequences. Reference [14] proposes to use the atomic operation of state transition to improve the result of problem semantic analysis. Semantic parsing based methods usually use classification models to predict relationships. However, because the knowledge graph contains hundreds of thousands of relationships, the training set is difficult to cover such large-scale relationships, so the semantic parsing based methods are limited in the question answering of the knowledge graph.

The second type is the method based on information retrieval. This kind of method first obtains several candidate entities according to the question, extracts the relationship connected with the candidate entities from the knowledge graph as the candidate query path, and then uses the text matching model to select the candidate query path with the highest similarity to the question to retrieve the answer. In the early stage, it was mainly based on the method of feature engineering. Literature [15] first analyzed the questions and extracted the candidate answers, and then generated the combined ranking of question features and candidate answer features. This method needs to customize the construction features and has a poor processing effect on complex problems. In recent years, representation based learning methods have been proposed and achieved good performance. Representation learning is to graph the candidate entities in the problem and knowledge graph to a unified semantic space for comparison. Literature [16] uses multi column convolution network to represent the semantic information of different aspects of the answer. Literature [17] proposed a translation distance model, Transe, to learn the translation invariance of entities and relations in low dimensional space and proved the effectiveness of embedding human in the knowledge graph on some related

problems. Literature [18] proposed a rescal semantic scoring model, which models the latent semantics of triple facts and completes the embedded representation of the knowledge graph from the perspective of semantics. The method of information retrieval transforms the complex semantic analysis problem into a large-scale learnable problem. It focuses on calculating the similarity between the problem and the candidate relationship and has better generalization ability in relationship selection. In addition, there are also some new methods, such as complex problem decomposition, the combination of neural computing and symbolic reasoning, the use of memory network to realize question answering, and so on.

The current mainstream knowledge graph-based Q&A methods only use a single scoring mechanism to score and rank the candidate entities and then output the single entity with the highest score as the answer, which may lead to missing answers when facing multiple answer entities. Although such methods can utilize the semantic information learned during knowledge graph embedding, they do not explicitly construct knowledge graph queries. In this paper, we propose an improved multi-hop ELT knowledge test method based on the knowledge graph embedding, which introduces a relational link scoring mechanism in the answer scoring part and outputs all candidate entities on the same relational link when the best answer entity is obtained; thus, effectively solving the answer omission problem and improving the robustness of the knowledge graph embedding test method. In the question embedding model, this paper improves the embedding of sentence vectors for the ELT knowledge test domain so that the model can better understand the English semantics.

2. Methodology

2.1. English Teaching Knowledge Graph Question Answering Method. Common knowledge graph testing tasks are classified into single-hop questions and multi-hop questions. Single-hop questions apply a single fact from the knowledge base to an answer. Multi-hop questions require two or more facts to be used together as the basis for an answer. Figure 1 shows the difference between these questions. Some current research methods have achieved high accuracy in answering single-hop questions, but there are still many difficulties and challenges in answering multi-hop questions [19], which are as follows: (1) Since multi-hop questions have higher semantic complexity compared to single-hop questions, it is difficult for the model to accurately separate multiple semantic relations from the question sentences. (2) The real knowledge graph is sparse and often misses. For example, in problem 3 in Figure 1, if the relationship between (Lu Xun, Chinese name, Zhou Shuren) and if the relationship link of the entity (Lu Xun, Chinese name, Zhou Shuren) is missing in the knowledge base, it is difficult to search for the true answer by the existing methods. Some current knowledge graph embedding methods can effectively capture the potential semantic relations in the knowledge base and have better prediction ability for the missing links. Thus, this paper proposes a multi-hop ELT knowledge question and
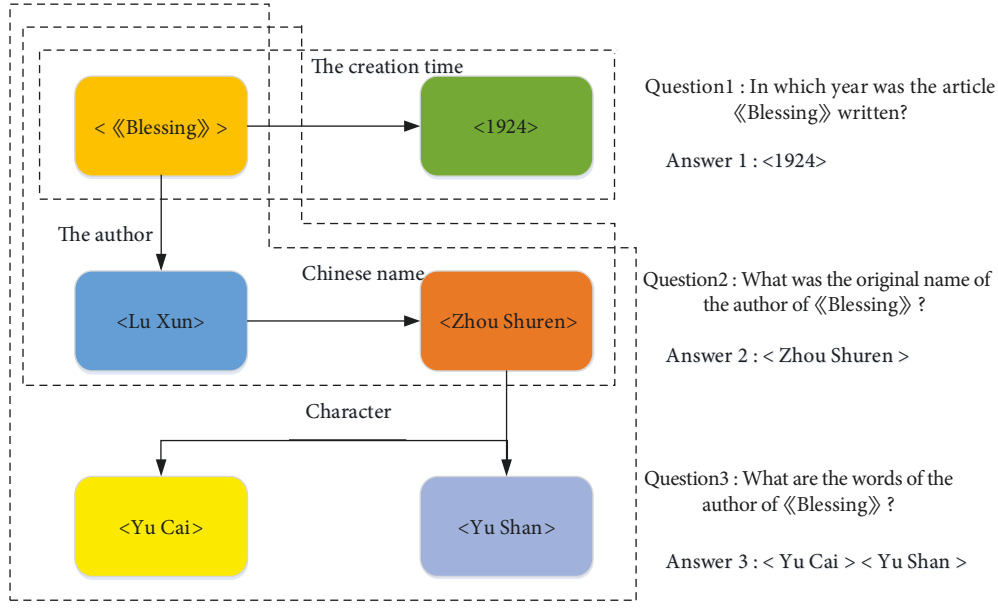
FIGURE 1: Examples of single-hop and multi-hop knowledge question answering.

answer method with the knowledge graph embedding model as the main body.

In this paper, a knowledge graph is defined as, given a set of entities E and a set of relations R, a knowledge graph ZA can be represented as a set of triples, i.e., $ZA \subseteq E \times R \times E$. For any triple n in ZA, it can be represented as an ordered pair (b, r, n), where $b, n \in E$ and $r \in R$, usually called b as the head entity and n as the tail entity. On this basis, the task is defined as the known knowledge graph $ZA \subseteq E \times R \times E$. Given a natural language question v_x, the central entity of the question is b, where $b \in E$. The role of the question and answer method is to give the set of answer entities $G_x \subseteq E$, for which $\forall g \in G_x$ is a reasonable answer entity for the question v_x.

The overall process can be divided into two parts as shown in Figure 2, which are the answer scoring part based on knowledge graph embedding and the link scoring and answer filtering part. The process steps of the multi-hop ELT knowledge test method based on knowledge graph embedding are as follows.

(1) For the infant question v and its head entity b_v, $\mathbf{e}_{b_v}$ is obtained by querying the embedding vector table obtained by pre-training.

(2) The embedding vector m_v of question v is computed by the sentence vector embedding model.

(3) Compute the knowledge graph embedding score for all candidate answers g, the formula is as follows:

$$S_g(g, b_v, v) = \text{Re}\left(\langle m_v, e_{b_v}, \overline{e}_g \rangle\right), \quad (1)$$

where e_g can be obtained by querying the embedding vector table.

(4) For the further narrowed candidate answer g, the link score is calculated as shown in formula:

$$S_R\left(R_{(b_v, g)}, v\right). \quad (2)$$

(5) Select the best answer entity by matching the score, the formula is as follows:

$$g_{\text{best}} = \arg\max_g S_g(g, b_v, v) + \beta \times S_R\left(R_{(b_v, g)}, v\right). \quad (3)$$

(6) The query link is constructed based on g_{best}, and the answer set G is obtained, and the following query methods are available according to whether b_v and g_{best} are connected.

(a) When b_v and g_{best} are connected, $G = \text{SPARQL}(b, R_{(b_v, g_{\text{best}})})$.

(b) When b_v and g_{best} are not connected, $G = \{g_{\text{best}}\}$.

(7) Return the set G as the result of the method calculation.

2.2. Knowledge Graph Embedding. A typical approach is divided into two parts. One defines the representation of entities and relations in the vector space (usually $\mathbb{R}^d$ or $\mathbb{C}^d$), and the other is to give the scoring function $f_{\text{triple}}(b, r, n)$ of a triple under this representation. The main role of the scoring function is to evaluate the rationality of the triple. In this premise, embedding models can be classified into distance transfer models and semantic scoring models according to the type of scoring functions. The former treats the plausibility of facts as the distance between vectors, while the latter evaluates the potential semantic relationships between entities.

In the knowledge graph embedding-based answer scoring, the scoring process is considered as a link prediction problem. It is hoped that the question-answer rationality
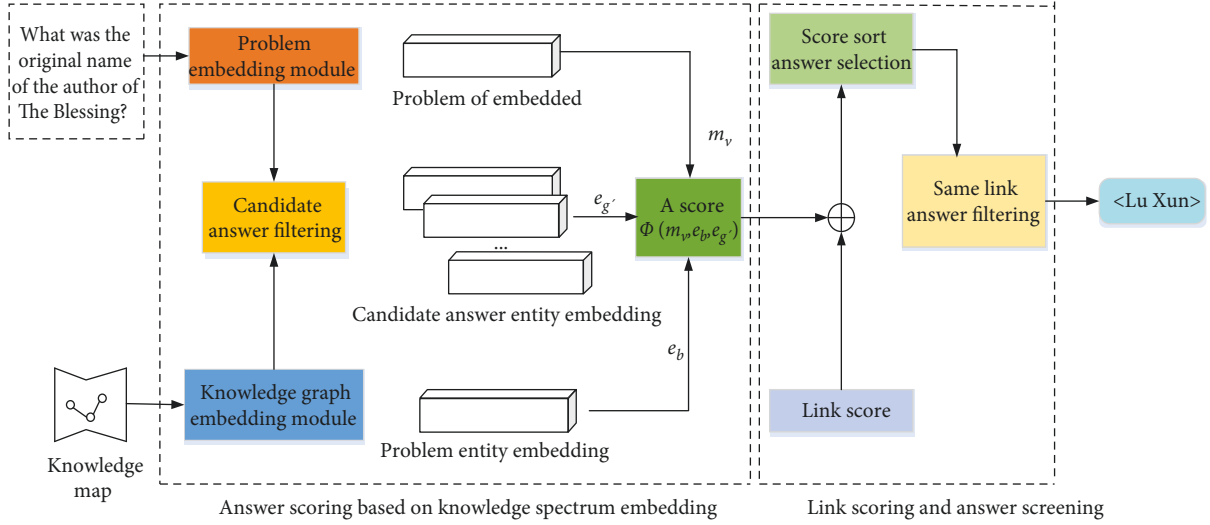
Figure 2: Overall process of multi-hop Chinese knowledge graph question answering.

evaluation can be combined with the triadic scoring in the embedding model by using the question embedding module to learn the semantics of the multi-hop relations contained in the question sentences. Literature [20] shows that the ComplEx method has excellent modeling capability for complex latent semantic relations, and the time complexity of the model pre-training algorithm is low. Therefore, the ComplEx embedding method is chosen in this paper.

2.2.1. ComplEx Embedding Method. The ComplEx embedding method extends the semantic embedding in the real space to the imaginary space, given a head entity with a tail entity b, $n \in E$ and a relation $r \in R$. ComplEx learns the vector representation $\mathbf{e}_b, \mathbf{e}_n, \mathbf{m}_r \in \mathbb{C}^d$ in the imaginary space based on the scoring function. In the embedding model, all triples considered reasonable have $\Phi(r, s, n) > 0$, all triples considered unreasonable have $\Phi(r, s, n) < 0$, and satisfies

$$\Phi(r, s, n) = \text{Re}\left(m_r, e_b, \overline{e}_n\right) = \text{Re}\left(\sum_{x=1}^{d} m_{rx} e_{bx} \overline{e}_{nx}\right), \quad (4)$$

where Re denotes the real part of the element and $\overline{\mathbf{e}}_n$ denotes the conjugate vector of $\overline{\mathbf{e}}_n$. This property allows the head and tail entity to get different fractional values when exchanging positions. Therefore, ComplEx can learn asymmetric relations, which is also more consistent with the nature of relations between entities in real knowledge graphs, making the embedding model more expressive in question and answer tasks.

2.3. Answer Scoring based on the Knowledge Graph Embedding. The answer selection module is dominated by the scoring of answers based on the knowledge graph embedding. In this paper, we use an improved question embedding model to obtain the sentence vector m_v and combine the embedding vectors e_{b_v} and e_g of related entities to obtain the answer scoring of this part by the scoring

function, which is improved from the scoring function based on the ComplEx method.

2.3.1. Question Embedding. The main task of the question embedding module is to embed a question sentence v composed of natural language into a complex space to obtain a sentence vector $m_v \in \mathbb{C}^d$. In some ELT question and answer methods, the typical way to obtain the sentence vector is to fine-tune it for the downstream task based on the BERT pre-trained language model units. In this paper, the pre-trained language model is used to process the interrogative sentences, and the model incorporates $N - $ gram vocabulary for representation enhancement, which preserves the semantic information of the sentences to a greater extent. The overall structure of the question embedding model is shown in Figure 3.

As seen in Figure 3, the network uses the ZEN model to embed the problem v into a 768-dimensional vector, which then passes through four fully connected layers and finally maps into the complex space $\mathbb{C}^d$. The learning is based on using the semantic scoring function in ComplEx. The composite semantic representation of multi-hop relations is obtained by approximating the relational vector m_r with the sentence vector $\mathbf{m}_v$. Given a question v and its head entity b_v, with a reference answer set C_v. The network learns based on the criteria defined in formula (5), and the loss function is the cross-entropy loss function.

$$\begin{aligned}
\text{Re}\left(\langle m_v, e_{b_v}, \overline{e}_g \rangle\right) > 0, \quad &\forall g \in C_v, \\
\text{Re}\left(\langle m_v, e_{b_v}, \overline{e}_{\bar{g}} \rangle\right) < 0, \quad &\forall \bar{g} \in C_v.
\end{aligned} \quad (5)$$

2.3.2. Scoring Function. When performing answer inference, the sentence vector w_q is obtained by question embedding, and the embedding vectors $\mathbf{e}_{b_v}$ and $\mathbf{e}_g$ of the head entity and candidate answer entities are known, the model
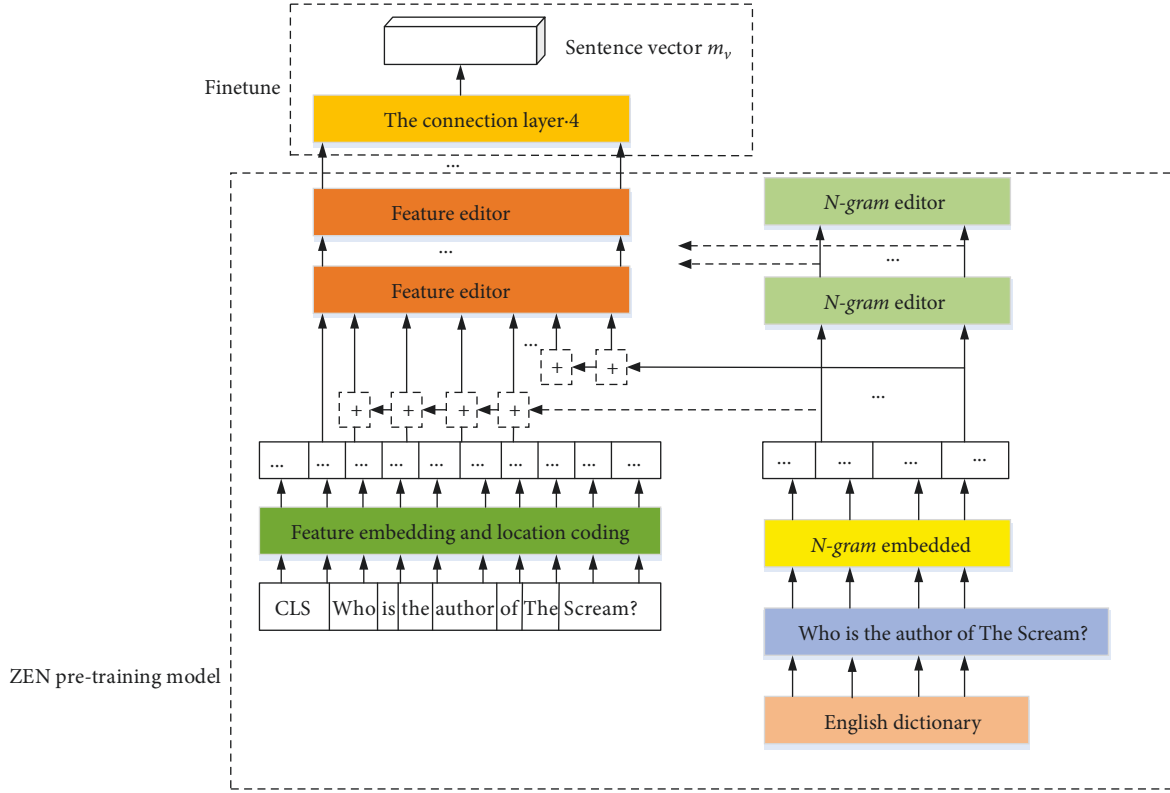
FIGURE 3: Question embedding based on ZEN.

can calculate the answer score $S_g(g, b_v, v)$ for each candidate answer, which is calculated as follows:

$$S_g(g, b_v, v) = \text{Re}\left(\langle m_v, e_{b_v}, \bar{e}_g \rangle\right) = \text{Re}\left(\sum_{x=1}^{d} m_{vx} e_{b_v} \bar{e}_{gx}\right). \quad (6)$$

The answer scoring function in formula (6) is formally consistent with the triadic scoring function, which is the main reason why the model can make full use of the unsupervised information in the knowledge graph embedding. In order to reduce the range of candidate answers, a scoring threshold S_{nb} can be given, and only the answers with scores above the threshold are selected as the set of candidate answers for link scoring.

2.4. Link Scoring and Answer Filtering. Unlike the query construction method, the answers obtained by the knowledge graph-based embedding method do not depend on the query path. The advantage of this method is that it has link prediction capability for sparse knowledge graphs, and can achieve a high score for the correct answer even if the query path is missing, but the disadvantage is that the robustness of the model is poor. This disadvantage is mainly reflected in the following two aspects.

(1) One of the advantages of knowledge graph Q&A is that the relational paths in the knowledge graph have higher credibility and interpretability, and simply using the knowledge graph embedding for answer scoring will weaken this advantage.

(2) The real answer set usually consists of multiple entities on the same query link, and it is difficult to obtain an accurate answer set by scoring with knowledge graph embedding.

To overcome these drawbacks, this paper introduces a link scoring mechanism based on knowledge graph embedding scoring to enhance the method's robustness.

2.4.1. Link Scoring. The query link $R_{(b_v, g)}$ between the head entity b_v of question v and the answer entity g consists of a sequence of relations $(r_1, r_2, \ldots, r_n)$. To measure the relevance between the relations and the question. In this paper, we construct a metric score $S_r(r_x, v)$ for network learning relevance based on formula (7), and use it to give a link score $S_R(R_{(b_v, g)}, v)$.

$$S_r(r_x, v) = \sigma\left(\text{Re}\left(m_{r_x}^N\right) \cdot \text{ZEN}(v)\right), \quad (7)$$

where m_{r_x} is the relationship vector in the knowledge graph embedding, $\text{ZEN}(\cdot)$ is the English pre-trained language model, and $\sigma(\cdot)$ is the activation function, which is the Sigmoid function in this paper. Thus, different link scoring functions can be given for the two other cases of whether the head entity b_v and the answer entity g are connected or not.

When there is an optimal connected link $R_{(b_v, g)}$, the link scoring function is as follows:

$$S_R\left(R_{(b_v, g)}, v\right) = \sum_{r_x \in R_{(b_v:g)}} S_r(r_x, v). \quad (8)$$

When there is no connected link, the link scoring function is as follows:

$$S_R\left(R_{(b_v,g)}, v\right) = \arg\max_{r_x \in \phi_g} S_r\left(r_x, v\right), \tag{9}$$

where ϕ_g is the set of relations of single-hop reachable answer entities g. The meaning of formula (8) gives a complementary scoring term to the answer entities that may have missing links, which is used to eliminate the bias, and the term can be set to 0 when the knowledge graph is dense.

2.4.2. Answer Filtering. To deal with the case of multiple answer entities, the same link query mechanism is used in the answer filtering. The formula is as follows:

$$g_{\text{best}} = \arg\max_g S_g\left(g, b_v, v\right) + \beta \times S_R\left(R_{(b_v,g)}, v\right). \tag{10}$$

Firstly, the answer entities are scored based on the knowledge graph embedding score and link score, and the best answer entity is given after ranking, where β is a hyperparameter.

When the optimal link $R_{(b_v, g_{\text{best}})}$ exists, the answer set G is as follows:

$$G = \text{SPARQL}\left(b, R_{(b_v, g_{\text{best}})}\right), \tag{11}$$

where $\text{SPARQL}(b, R)$ denotes the set of entities obtained from one graph database query based on the head entity b through the relational sequence R. When there is no connected link, the answer set G is as follows:

$$G = \{g_{\text{best}}\}. \tag{12}$$

The answer set G is obtained for a given question v and head entity b_v.

3. Result Analysis and Discussion

3.1. Experimental Data Set and Environment. This paper uses the knowledge base and question and answer pair data published by the DBpedia Neural Question Answering data set and selects 15050 training question and answer pairs and 9360 test question and answer pairs. In order to make the evaluation results more objective, the training Q&A pair is further randomly divided into training set and development set, and the test Q&A pair is used as the test set. The data set division is shown in Table 1.

The experiments are run on a computer with Inter15-4590 CPU and 12 GB RAM, Nvidia GTX1080Ti graphics card with 11 GB video memory, CUDA10.0 and Tensorflow1.14 deep learning framework, 64-bit Windows 10 operating system, and Mysql5.6.46 for knowledge base data storage and retrieval using Mysql5.6.46.

3.2. Analysis of Named Entity Recognition Results. The training process iterates 8122 times in total, and Adam optimizer with weight attenuation is used to optimize the loss function. After completing the training on the training

TABLE 1: Division of data set.

Name	Number
Training set	13000
Development set	2050
Test set	9360

set, test the performance of the model on the training set, development set, and test set, respectively. The results are shown in Figure 4. Due to the size of the corpus, there is a slight overfitting phenomenon in the test results, which is basically accurate on the training set and has some errors on the development and test sets, and the overall performance is good.

3.3. Analysis of Answer Matching Results. To train the answer matching network model, we need to create the answer matching data set based on the existing question-answer pairs. Creating negative samples is similar to that of Q&A. The named entity is used as the keyword to search the knowledge base to get the set of answers related to the entity, and the nouns that are not the answer to the question are connected to the question in the same way. For entities with only one triad in the knowledge base, five answers are randomly selected as negative samples from the triads with other entities as keywords and added to the data set to obtain. The size of the answer matching data set is shown in Table 2.

The training set data are fed into the answer matching network for training. Since the feature extraction part of the network also uses BERT, the hyperparameter selection is consistent with the named entity recognition except for the absence of LSTM. The optimizer for model training also used Adam with weight decay, and the network was iterated 12120 times. Since the similarity score is a value between 0 and 1, which cannot be exactly equal to the label, the output of the network is modified as a category when calculating the test metrics, i.e., it is treated as a dichotomous problem and can only output "0" or "1." In addition to the accuracy, AUC is also an important performance index when calculating the performance index. It can more objectively measure the classification effect of the model on the answer matching data set. The test results of the answer matching model on the training set, development set, and test set are shown in Table 3. Due to the limited size of the data, the performance of the model in the development and test sets is poor but the AUC values are above 86%, which guarantees the quality of the final automatic test.

3.4. Analysis of Hyperparameter Selection Results. After completing the training of named entity recognition and answer matching model, the knowledge base Q & A can be carried out. When the super parameter selection mechanism is not added, the answer with the highest super parameter score in the triplet set containing entities in the knowledge base is directly selected as the output, and the Q & A results are shown in Table 4. Since there is only one standard answer
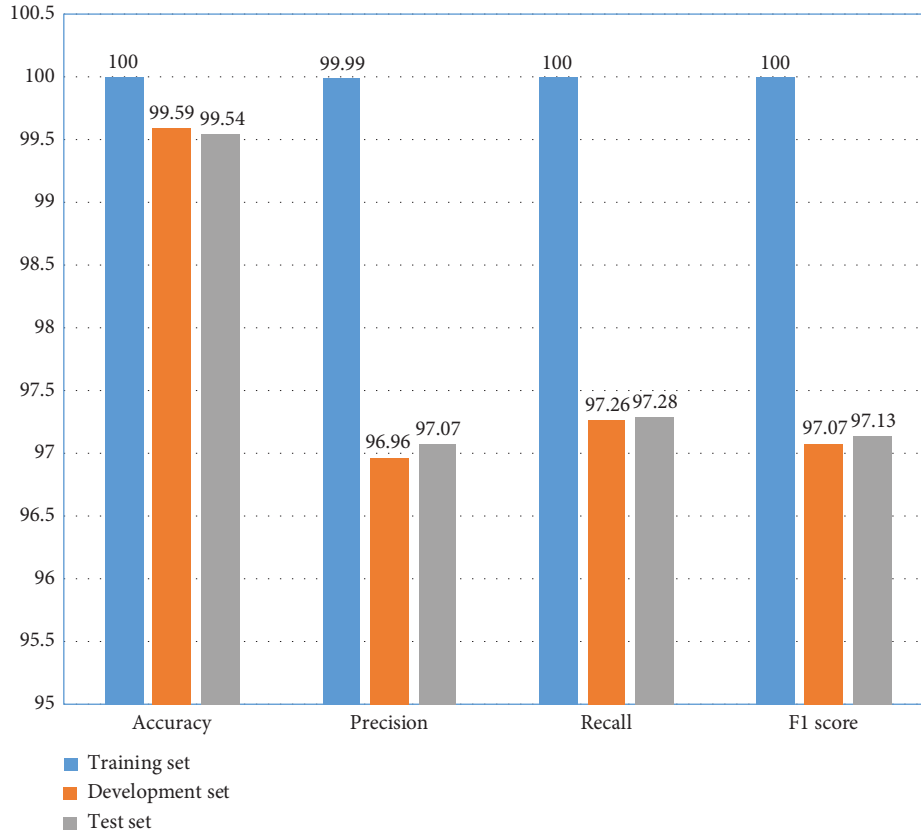
FIGURE 4: Test results of named entity recognition %.

TABLE 2: Data set size of answer matching.

Data set	Number of positive samples	Number of negative samples	Total number
Training set	13000	64225	77225
Development set	2050	31866	33916
Test set	9360	40452	49812

TABLE 3: Test results of answer matching %.

Data set	Accuracy	AUC
Training set	99.44	98.63
Development set	96.47	87.58
Test set	92.56	86.25

TABLE 4: Question answering results without threshold selection mechanism %.

Data set	F1 score
Training set	95.23
Development set	87.71
Test set	86.73

TABLE 5: Question answering results of development set with different similarity thresholds %.

β value	Precision	Recall	F1 score
0.1	83.89	91.21	87.37
0.3	84.26	91.12	87.52
0.5	85.62	90.91	88.20
0.7	85.85	90.11	87.89
0.9	86.39	89.64	87.89

and one predicted answer and the scores are the same, only F1 scores are listed.

By recording the incorrectly answered questions and observing their hyperparameters. It was found that in addition to the noise in the data set itself, the hyperparameters among the top few of the more ambiguous answers were mainly around 0.1 to 0.9. In order to determine the best super parameters under the data set in this paper, select five super parameters of 0.1, 0.3, 0.5, 0.7, and 0.9, respectively, and call the super parameter selection mechanism on the development set for testing as shown in Table 5.

From the results of hyperparameter selection, we can see that the precision rate gradually becomes smaller and the recall rate gradually becomes larger as the selected hyperparameter decreases, which is the inevitable result of more alternative answers. When the hyperparameter is 0.5, the F1 score of the test on the development set is the highest.
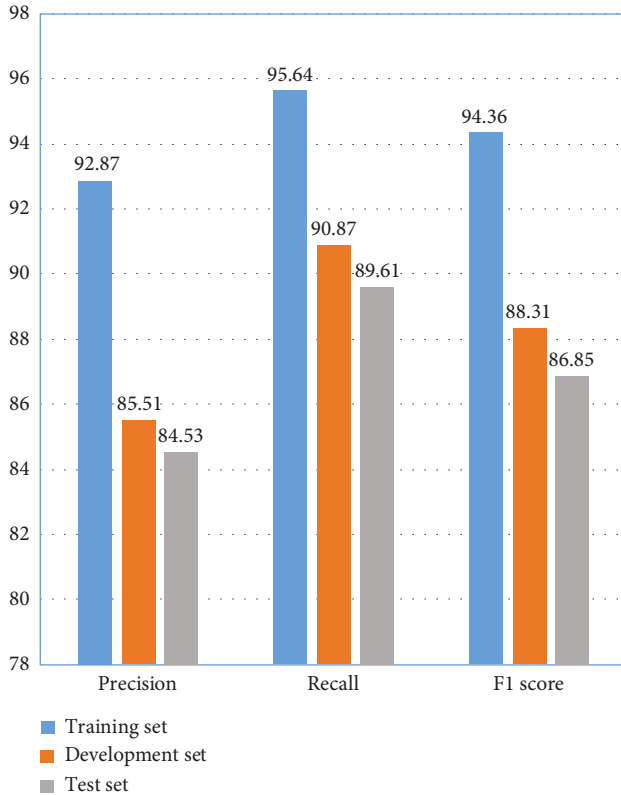
FIGURE 5: Final results of the knowledge base question answering.

TABLE 6: Automatic question answering results of ten methods %.

Automatic question and answer method	Test set F1 scores
Literature [21]	71.12
Literature [22]	72.65
Literature [23]	79.18
Literature [24]	79.52
Literature [25]	81.30
Literature [26]	81.51
Literature [27]	82.43
Literature [28]	82.87
Literature [29]	84.55
Proposed	86.85

the evaluation results of Arts & human cities index tasks, respectively. They mainly rely on some manual rules to ensure the question and answer performance. For example, literature [27] constructs regular expressions to remove redundant information from the interrogative sentences. Literature [26] uses combinatorial features of lexicality to achieve named entity recognition, etc. Literature [25] is an automatic question and answer method constructed based on attribute mapping of predicates in knowledge base triples with a small number of artificial features. Literature [28] is an automatic question and answer method implemented by syntactic analysis, etc. Literature [29] first applied BERT for feature extraction on the DBpedia Neural Question Answering data set and achieved the best results published so far. In addition to applying BERT, the method in this paper also improves the answer selection method by decomposing it into two steps, answer matching and hyperparameter selection, which reduces the need for manual annotation and preprocessing and obtains a test set F1 score of 86.85% with the best performance.

When the hyperparameter decreases further, the precision rate decreases more due to the number of selected answers, so the F1 score decreases as well.

3.5. Comparison of Automatic Test Results. Through the experimental results of hyperparameter selection, 0.5 was selected as the β value of the knowledge base test in this paper, and it was applied to the final ELT test system as shown in Figure 5. Both the training and development sets are derived from the training set of the original test pairs of DBpedia Neural Question Answering task and the score of test set F1 is used in the public evaluation index.

In this paper, the English teaching test system uses hyperparameter selection as an optional switch in practical applications. In many application scenarios where the test task requires a single answer to be returned, the hyperparameter selection switch is turned off and the most accurate answer is presented to the user. If the user has doubts about the answer or if some scenarios allow multiple answers to be returned, the hyperparameter selection can be turned on and the set of candidate answers will be presented in order of similarity from lowest to highest.

This paper selects literature [21–28]and literature [29] as the comparison method. The automatic question and answer results are shown in Table 6. Literature [21] is based on the idea of dynamic programming. Its unsupervised idea has reference significance, but the effect of question and answer is relatively limited. Literature [22–24, 26] and literature [27] are the top five automatic question and answer methods for

4. Conclusion

In this paper, the multi-hop ELT knowledge test method based on knowledge graph embedding innovatively combines the knowledge graph embedding scoring algorithm and link scoring algorithm in scoring answers, and constructs a query path with high confidence through composite scoring, which effectively solves the phenomenon of missing answers in the current knowledge graph embedding-based test methods. The experimental results show that the F1 score of the English teaching test system on the DBpedia Neural Question Answering data set is 86.58%, in which the improved vector embedding model based on ZEN fits complex semantic relations and provides more accurate semantic understanding for multi-hop test tasks. At the same time, because of the introduction of knowledge graph embedding, the question-answer method has some link prediction ability, and the method still has strong inference ability on the incomplete knowledge graphs. Through the experiments, it is found that the accuracy of the English teaching test system for number type answers in this paper needs to be improved, and the subsequent methods such as representation learning will be used to filter the optimal solutions from the set of candidate answers to improve the test quality further.

Data Availability

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

Conflicts of Interest

The author declares that there are no conflicts of interest.

References

[1] K. P. Kalyanathaya, D. Akila, and P. Rajesh, "Advances in natural language processing–a survey of current research trends, development tools and industry applications," *International Journal of Recent Technology and Engineering*, vol. 7, no. 5C, pp. 199–202, 2019.

[2] Q. Zheng, B. Dong, B. Qian, and F. Tian, "The state of the art and future tendency of smart education," *Journal of Computer Research and Development*, vol. 56, no. 1, p. 209, 2019.

[3] V. A. Kotkar, "Interactive robot for automated question and answer system," *Turkish Journal of Computer and Mathematics Education (TURCOMAT)*, vol. 12, no. 1S, pp. 90–95, 2021.

[4] S. Hu, L. Zou, J. X. Yu, and W Haixun, "Answering natural language questions by subgraph matching over knowledge graphs," *IEEE Transactions on Knowledge and Data Engineering*, vol. 30, no. 5, pp. 824–837, 2017.

[5] S. Liu, X. Zhang, S. Zhang, H Wang, and W Zhang, "Neural machine reading comprehension: methods and trends," *Applied Sciences*, vol. 9, no. 18, p. 3698, 2019.

[6] S. Momtazi, "Unsupervised latent dirichlet allocation for supervised question classification," *Information Processing & Management*, vol. 54, no. 3, pp. 380–393, 2018.

[7] S. K. Mohamed, A. Nounu, and V. Novacek, "Biological applications of knowledge graph embedding models," *Briefings in Bioinformatics*, vol. 22, no. 2, pp. 1679–1693, 2021.

[8] J. Yuan, Z. Jin, H Guo et al., "Constructing biomedical domain-specific knowledge graph with minimum supervision," *Knowledge and Information Systems*, vol. 62, no. 1, pp. 317–336, 2020.

[9] S. Liu, H. Yang, J Li, and S Kolmanic, "Preliminary study on the knowledge graph construction of Chinese ancient history and culture," *Information*, vol. 11, no. 4, p. 186, 2020.

[10] T. Ruan, Y. Huang, and X. Liu, "QAnalysis: a question-answer driven analytic tool on knowledge graphs for leveraging electronic medical records for clinical research," *BMC Medical Informatics and Decision Making*, vol. 19, no. 1, pp. 1–13, 2019.

[11] X. Li, H. Wang, H. He, D Jixiang, C Jian, and W Jinzhun, "Intelligent diagnosis with Chinese electronic medical records based on convolutional neural networks," *BMC Bioinformatics*, vol. 20, no. 1, pp. 1–12, 2019.

[12] A. M. Rinaldi and C. Russo, "Using a multimedia semantic graph for web document visualization and summarization," *Multimedia Tools and Applications*, vol. 80, no. 3, pp. 3885–3925, 2021.

[13] H. Strobelt, S. Gehrmann, and M. Behrisch, "S eq 2s eq-v is: a visual debugging tool for sequence-to-sequence models," *IEEE Transactions on Visualization and Computer Graphics*, vol. 25, no. 1, pp. 353–363, 2018.

[14] R. Nagamanjula and A. Pethalakshmi, "A novel framework based on bi-objective optimization and LAN2FIS for Twitter sentiment analysis," *Social Network Analysis and Mining*, vol. 10, no. 1, pp. 1–16, 2020.

[15] B. Abu-Nasser, "Medical expert systems survey," *International Journal of Engineering and Information Systems (IJEAIS)*, vol. 1, no. 7, pp. 218–224, 2017.

[16] J. Zhang, J. Sun, J Wang, and X. G Yue, "Visual object tracking based on residual network and cascaded correlation filters," *Journal of Ambient Intelligence and Humanized Computing*, vol. 12, no. 8, pp. 8427–8440, 2021.

[17] Z. Cao, X. Qiao, S Jiang, and X Zhang, "An efficient knowledge-graph-based web service recommendation algorithm," *Symmetry*, vol. 11, no. 3, p. 392, 2019.

[18] G. A. Gesese, R. Biswas, M Alam, and H Sack, "A survey on knowledge graph embeddings with literals: which model links better literal-ly?" *Semantic Web*, vol. 12, no. 4, pp. 617–647, 2021.

[19] Y. Fang, H. Wang, L. Zhao, F Yu, and C Wang, "Dynamic knowledge graph based fake-review detection," *Applied Intelligence*, vol. 50, no. 12, pp. 4281–4295, 2020.

[20] K. Watanabe, "Latent semantic scaling: a semisupervised text analysis technique for new domains and languages," *Communication Methods and Measures*, vol. 15, no. 2, pp. 81–102, 2021.

[21] E. Dimitrakis, K. Sgontzos, and Y. Tzitzikas, "A survey on question answering systems over linked data and documents," *Journal of Intelligent Information Systems*, vol. 55, no. 2, pp. 233–259, 2020.

[22] A. Thomas and S. Sangeetha, "Semi supervised, knowledge integrated pattern learning approach for fact extraction from judicial text," *Expert Systems*, vol. 38, no. 3, Article ID 12656, 2021.

[23] Q. Lu, Z. Zhu, F Xu, D Zhang, W Wu, and Q Guo, "Bi-GRU sentiment classification for Chinese based on grammar rules and BERT," *International Journal of Computational Intelligence Systems*, vol. 13, no. 1, p. 538, 2020.

[24] R. Zheng, H. Ma, Q. Wang, J Fu, and Z Jiang, "Assessing the security of campus networks: the case of seven universities," *Sensors*, vol. 21, no. 1, p. 306, 2021.

[25] P. Wang, Q. Wu, C. Shen, A V D Hengel, and D Anthony, "Fvqa: fact-based visual question answering," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 10, pp. 2413–2427, 2017.

[26] J. Li, A. Sun, J. Han, and C Li, "A survey on deep learning for named entity recognition," *IEEE Transactions on Knowledge and Data Engineering*, vol. 34, no. 1, pp. 50–70, 2020.

[27] L. Li, M. Zhang, Z Chao, and J Xiang, "Using context information to enhance simple question answering," *World Wide Web*, vol. 24, no. 1, pp. 249–277, 2021.

[28] D. Diefenbach, V. Lopez, K. Singh, and P Maret, "Core techniques of question answering systems over knowledge bases: a survey," *Knowledge and Information Systems*, vol. 55, no. 3, pp. 529–569, 2018.

[29] F. A. Acheampong, H. Nunoo-Mensah, and W. Chen, "Transformer models for text-based emotion detection: a review of BERT-based approaches," *Artificial Intelligence Review*, vol. 54, no. 8, pp. 5789–5829, 2021.