ingegneria sismica
International Journal of Earthquake Engineering
Anno XLIII - Num. 1 - 2026

Knowledge Graph Completion Algorithms for Fusing Semantic Information

Lihua Duan[1,*]

[1] School of Cybersecurity, Liaoning Police College, Dalian, Liaoning, 116036, China

SUMMARY: *Although knowledge graph can enhance the efficiency of knowledge understanding and application, its inherent accuracy and completeness problems make knowledge graph complementation a current research focus. In this paper, the Trans H algorithm is improved by fusing semantic information, simplifying the triad by constructing an information hyperplane, and introducing BERT word vectors, which effectively improves the training efficiency. Adopting the attention mechanism proposed by previous researchers, the semantic information and the parameter vectors of the original model are fused and inputted into the Attention structure, and the attention scores of the semantic information of the triad are calculated. In order to evaluate the performance of the joint model SI-KGC, the dataset and experimental environment are constructed, hyperparameters are set, and the evaluation results are obtained after experiments such as ternary classification. The SI-KGC model constructed in this paper improves the accuracy by 3.12% compared to TransD. Meanwhile, it has high accuracy in the ternary group classification task, with an average accuracy of 83.8% and 74.75% on the two datasets, respectively, with higher classification accuracy and superior performance. Comparing the performance of TransE and SI-KGC on the FB15K dataset, the performance gap between the two models is between 0.003 and 0.011, and SI-KGC has a complete modeling capability, which has the effect of improving the model performance.*

KEYWORDS: *trans H algorithm; BERT; semantic word vector; semantic information; SI-KGC; knowledge graph complementation*

1 Introduction

With the rapid development of the Internet, people's production and life have generated a huge amount of information [1]. Processing and utilizing this information can improve people's production and life style, and enhance production efficiency and living standards [2]. In order to make full use of this information, it is necessary to first extract this information to form a machine-understandable formal representation, and then construct a knowledge system. For this reason, the concept of knowledge graph has been proposed. Knowledge graph is a technology for storing, managing and retrieving knowledge, and as a structured knowledge representation, it has been widely used in various fields, including clinical decision-making, intelligent search, and recommender systems [3, 4]. By constructing and utilizing knowledge graphs, knowledge can be better organized and understood to provide more accurate, comprehensive and intelligent information services [5, 6]. However, due to the dynamic and incomplete nature of knowledge graphs, how to accurately and comprehensively complete the missing entities, relationships and attributes in the graphs has become one of the hot issues in the current knowledge graph

---

*daliandlh@163.com
https://doi.org/10.65102/is2026108

research field [7, 8].

Traditional knowledge graph completion algorithms mainly rely on graph structure information, for example, utilizing topological relationships between entities [9, 10]. However, these approaches usually ignore the semantic information of entities and relations, limiting their applications in real-world scenarios. Meanwhile, with the development of the field of natural language processing, pre-trained language models have achieved great success in providing rich semantic information, and fusing semantic information provides new possibilities for complementing knowledge graphs [11-13]. By employing natural language processing techniques to extract semantic information about entities and relationships from text, the accuracy of knowledge graph complementation can be effectively improved [14, 15]. The remarkable performance of capturing factual knowledge from massive text by utilizing pre-trained language models, such as BERT and GPT, has also been gradually applied to knowledge graph complementation tasks [16].

For knowledge graph complementation algorithms incorporating semantic information and their applications in related fields, literature [17] proposes a semantically-enhanced complementation model SE-KGC for knowledge graph sparsity, which effectively captures the potential semantic associations among entities by integrating predefined semantic schemas and with the help of a multirelational graph convolutional network encoder, and experimentally confirms that this model significantly improves the performance of link prediction on multiple datasets, which emphasizes the the key role of semantic information in the complementation process. Literature [18] for the problem of insufficient semantic modeling in knowledge graph complementation, proposed enhancement strategies that integrate external knowledge, deep representation and causal inference, and experiments show that these methods can effectively improve the prediction accuracy of long-tailed entities, which emphasizes the key impact of semantic information on the complementation effect. Literature [19] proposes a semantic information complementation algorithm that combines pre-trained language models with hyperbolic spatial distances, which enhances the robustness of Knowledge Graph prediction of unseen entities and reduces the inference overhead by fusing semantic descriptions of ternary groups and graph structure information, and experiments confirm that it outperforms the existing methods on standard datasets. Literature [20] proposes FuST-KGC, a knowledge graph complementation method that fuses subgraph structure and text semantics, integrating multi-hop topology information through dynamic subgraph sampling and serialization, and combining it with entity descriptions to form a unified input, realizing synergistic structural and semantic reasoning, and experimentally verifying that it outperforms the existing baselines on the WN18RR and FB15k-237 datasets. Literature [21] proposes ISA-KGC, a knowledge graph complementation method that fuses semantics and structure, which effectively combines graph structural information and textual semantics by integrating graph neural network and Transformer, and experiments show that it significantly improves the performance of complementation on the baseline dataset. Literature [22] introduces a knowledge graph complementation method that fuses semantic and structural information, extracts entity description semantics through TinyBERT encoder, and combines it with convolutional neural network to capture ternary structural features, which reduces the parameter scale while realizing the performance better than the single-information method, and verifies the effectiveness of the semantic information complementation. Literature [23] proposes an enhanced relational representation model fusing external graph structure and internal entity semantics for less-sample knowledge graph complementation, extracting structural features through graph convolutional network with attention mechanism and constructing semantic mapping by combining entity information, and finally utilizing a prototype network for classification, and experimentally verifies its excellent performance on two datasets.

In addition, literature [24] proposed the SR-GNN model for fusing semantic and relational information for the task of knowledge graph completion, which captures the semantic similarity between entities through recurrent neural networks and distinguishes the relational semantic differences using GRUs, and the experiments show that it outperforms the existing baseline methods on two benchmark datasets. Literature [25] proposes a deep relational graph information maximization model DRGI, which utilizes an adaptive relational graph attention network to capture semantic information and complete structural information respectively, and achieves effective fusion through mutual information maximization, and achieves superior complementation performance and faster convergence on standard datasets. Literature [26] introduces MgHiSal, an MLLM-guided hierarchical semantic alignment framework, which solves the semantic fragmentation problem in multimodal knowledge graph complementation by generating contextual visual descriptions and optimizing the cross-modal features using gated attention, which significantly improves the link prediction performance. Literature [27] adopts a complementation framework of dynamic semantic sampling and associative embedding, which effectively solves the problems of gradient vanishing and insufficient semantic expression of traditional methods and improves the performance of knowledge graph complementation through quality-oriented negative sampling and sequential information learning. Literature [28] designed a Tc-MLS complementation model based on the newly constructed carbon neutral time-series knowledge graph dataset, which acquires text semantics by pre-training the language model and mines potential semantic associations by combining topological and logical rules, and the experiments show that the model significantly improves the performance of the complementation on multiple datasets. Literature [29] explored the application value of semantic knowledge graph in the news field, and by analyzing how it integrates heterogeneous data to meet the industry needs, it pointed out the key role of this technology in the production, distribution and consumption links, and emphasized its importance to the future development of the news industry. Literature [30] analyzes the constraints of medical knowledge graph sparsity on the completion task, points out that the performance of traditional path inference methods is limited due to ignoring text semantics, and emphasizes the necessity of fusing semantic information to enhance the completion effect of medical knowledge graph.

In this paper, the TransH translation model under the ternary structure representation learning model is modified to construct a semantic information hyperplane using the improved Trans H algorithm to accelerate the efficiency of DSI extraction from ternary and extract relevant semantic information. BERT is utilized to capture contextual information, which makes the performance of NLP tasks better. The attention mechanism is introduced, and the attention output is obtained after the operation of scoring function to calculate the similarity of the biased semantic information vectors. The FB15K, WN18 & WN11 datasets are constructed and hyperparameter settings are set to evaluate the joint model SI-KGC from four aspects, namely, ternary classification, linkage test, ablation experiments, and supplementary experiments.

2 Methodology

2.1 Ternary structure to represent the learning model

2.1.1 Translation model

(1) TransE

After Word2Vec was proposed, the peculiar phenomenon that word vectors have a kind of translation invariance, such as shown in Eq. (1), has been discovered in the course of research:

$$V("King") - V("Man") = V("Queen") - V("Women") \tag{1}$$

From the above equation, it can be found that the word vectors can capture certain semantic information implied between King and Man, and between Queen and Women, which indicates that after the words are trained by Word2Vec to obtain the vector representations, the addition and subtraction operations between the word vectors can indicate their semantic correlations. Inspired by this phenomenon, some scholars proposed the TransE model in 2013.

TransE regards the relationships in the knowledge graph as some kind of translation between entities, which needs to satisfy Eq. (2) for the triple (h, r, t) and should not be satisfied for the wrong triple, as shown in Fig. 1:

$$\vec{h} + \vec{r} = \vec{t} \tag{2}$$

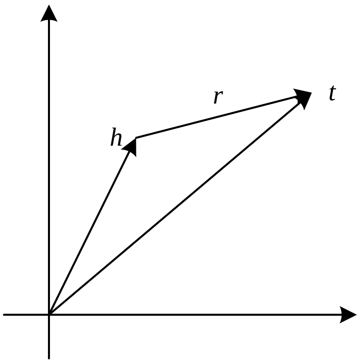


Figure 1: TransE model

TransE constructs the scoring function by calculating the Euclidean distance between the sum of the vectors of the head entity and the relation and the tail entity. For the triples (h, r, t) existing in the knowledge graph, it is desirable that the distance between $\vec{h} + \vec{r}$ and $\vec{t}$ be as small as possible, and the score be as high as possible. Correspondingly, for incorrect triples, the distance should be as large as possible, and the score should be as low as possible.

(2) Trans H model

TransE has received a lot of attention because of its simplicity and efficiency, but it suffers from the major drawback that it can only handle one-to-one relationships. There are usually four types of relationships in the knowledge graph: one-to-one, one-to-many, many-to-one, and many-to-many.

Assume that relation r is a many-to-one relationship, and there exists a relation r between the head entity h and two tail entities t_1 and t_2, i.e., there exist two triples (h_1, r, t) and (h_2, r, t). According to the TransE model, the vector representations of both triples satisfy $\overrightarrow{head} + \overrightarrow{relation} = \overrightarrow{tail}$, i.e. $\vec{h_1} + \vec{r} = \vec{t}$ and $\vec{h_2} + \vec{r} = \vec{t}$ hold at the same time, and it can be introduced that $\vec{h_1} = \vec{h_2}$, which is obviously wrong. For example, for the triples (Shanghai, Located_in, China) and (Beijing, Located_in, China), the representation of entities and relations using TransE would lead to the erroneous conclusion that $V(Beiging) = V(Shanghai)$. The same is true for one-to-many and many-to-many relationships.

In practical scenarios, complex relationships are common in knowledge graphs, and one-to-one relationships only account for a small proportion, so TransE has strong limitations. In order to be able to model the complex relationships in the knowledge graph, researchers have proposed the TransH model after improving TransE.

Unlike TransE, which directly performs ternary distance calculation, the TransH model models on a relationship plane, as shown in Figure 2.

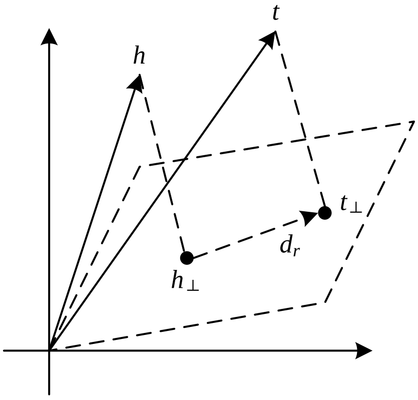


Figure 2: TransH model

TransH introduces a hyperplane for the relation r and projects the head and tail entities onto that plane before translating. The hyperplane is represented by the normal vector w_r and the translation vector d_r, and for a given triple $(h,r,t),h$ and t are projected onto this hyperplane to obtain $h_\perp$ and $t_\perp$, respectively, which are represented as shown in equations (3) and (4):

$$h_\perp = h - w_r^T h w_r \tag{3}$$

$$t_\perp = t - w_r^T t w_r \tag{4}$$

Similar to TransE, TransH treats the translation vector d_r of the relation in the hyperplane as a translation of the projections $h_\perp$ and $t_\perp$ of the entity in the hyperplane, which satisfies $h_\perp + d_r = t_\perp$ if the triple (h,r,t) exists and the scoring function is as in equation (5)) is shown:

$$f_r(h,t) = \left\| h_\perp + d_r - t_\perp \right\|_2^2 = \left\| \left(h - w_r^T h w_r\right) + d_r - \left(t - w_r^T t w_r\right) \right\|_2^2 \tag{5}$$

TransH maps entities onto hyperplanes corresponding to relations, and different entities may have the same projection on the hyperplane of a relation, solving the defect that TransE cannot represent complex relations well, e.g., the triples (h,r,t) and $\left(h,r,t'\right)$ both hold, and different vectors t and t' have the same projection on the hyperplane corresponding to the relation r.

2.1.2 Improved Trans H algorithm for semantic information extraction

The traditional Trans E algorithm is inefficient in extracting the fugitive semantic information of the ternary when dealing with complex knowledge graphs such as N-1, 1-N, etc. The Trans R algorithm has large parameters, which makes it difficult to be applied to the information extraction of large-scale knowledge graphs. While the improved Trans H algorithm has fewer parameters, it accelerates the extraction efficiency of the biased semantic information vector (referred to as DSI) of the ternary by constructing the semantic information hyperplane, effectively simplifies the ternary semantic information, and dramatically improves the efficiency of the later training.

By constructing an improved semantic information hyperplane S_i, the head entity H and the tail entity T in the triplet are mapped onto S_i (mapping: $H \to H_i; T \to T_i$), such that the head entity information H_i is mapped onto the tail entity information T_i through the semantic relationship D_i, that is: $H_i + D_i = T_i$, so that the triplets after being processed by the improved Trans H have semantic information.

Taking the unit normal vector $\vec{s}$ for the semantic information hyperplane S_i, the projection length vector $\vec{h_i}$ of the head entity vector $\vec{h}$ on the surface of S_i is computed as shown in equation (6):

$$\vec{h_i} = \vec{h} * \vec{s} \times \vec{s} \tag{6}$$

It is possible to derive the projection vector $\vec{h_s}$ of the head solid vector $\vec{h}$ on the S_i-plane as shown in equation (7):

$$\vec{h_s} = \vec{s}^T \vec{h} \vec{s} \tag{7}$$

Then the head entity projection vector $\overrightarrow{h_s}$ can be expressed as equation (8):

$$\vec{h_i} = \vec{h} - \vec{h_s} = \vec{h} - \vec{s}^T \vec{h} \vec{s} \tag{8}$$

Similarly the tail entity projection vector $\vec{t_s}$ can be expressed as equation (9):

$$\vec{t_s} = \vec{t} - \vec{s}^T \vec{t} \vec{s} \tag{9}$$

The distance difference L of the triples on the semantic information hyperplane S_i is computed using the Euclidean distance calculation method, which is shown in Equation (10):

$$L = \left[\gamma + \left(h_i + d_i - t_i \right)^2 - \left(h_i{}' + d_i - t_i{}' \right)^2 \right] + \tag{10}$$

In Eq. (10), $[x]+$ denotes $\max(x, 0)$, and by combining Eq. (8) with Eq. (9) shown in Eq. (10), the objective function of the improved Trans H is obtained as:

$$\min \sum_{h,t \in S} \sum_{h', t' \in S} \left[\gamma + \left(\vec{h} - \vec{s}^T \vec{h} \vec{s} + \vec{d_i} - \vec{t} + \vec{s}^T \vec{t} \vec{s} \right)^2 - \left(\vec{h}' - \vec{s}^T \vec{h}' \vec{s} + \vec{d_i} - \vec{t}' + \vec{s}^T \vec{t}' \vec{s} \right)^2 \right]_+ \tag{11}$$

Based on the objective function of the improved Trans H it is known that the loss function loss is shown in equation (12):

$$loss = \sum\sum_{h,t\in Sh',\, t'\in S}\left[\gamma + \left(\vec{h} - \vec{s}^T\vec{h}\vec{s} + \vec{d}_i - \vec{t} + \vec{s}^T\vec{t}\vec{s}\right)^2 - \left(\vec{h}' - \vec{s}^T\vec{h}'\vec{s} + \vec{d}_i - \vec{t}' + \vec{s}^T\vec{t}'\vec{s}\right)^2\right]_+ \tag{12}$$

For Eq. (12) for an entity information h_k its biased derivative semantic information is obtained:

$$\overrightarrow{h_d} = \frac{\partial loss}{\partial h_k} = \begin{cases} 2\left(\overrightarrow{h_k} - \vec{s}^T\vec{h}\vec{s} + \overrightarrow{d_i} - \vec{t} + \vec{s}^T\vec{h}\vec{s}\right) \\ *\left(-\vec{s}i^2\right), \quad \text{When } \Delta > 0 \\ 0, \quad \text{When } \Delta \leqslant 0 \end{cases} \tag{13}$$

Format:

$$\begin{aligned} \Delta = \gamma &+ \left(\vec{h} - \vec{s}^T\vec{h}\vec{s} + d_i - \vec{t} + \vec{s}^T\vec{t}\vec{s}\right)^2 \\ &- \left(\vec{h}' - \vec{s}^T\vec{h}'\vec{s} + d_i - \vec{t}' + \vec{s}^T\vec{t}'\vec{s}\right)^2 \end{aligned} \tag{14}$$

2.2 Pre-trained language model BERT

Encoding natural language text into vector representations, these vectors can be used for various NLP tasks such as text categorization, question and answer, named entity recognition, etc. Compared to traditional NLP methods based on bag-of-words models or sequence models, BERT can better capture contextual information, resulting in better performance of NLP tasks.

2.3 Attention mechanisms

The attention mechanism is a method used to enhance the attention of a deep learning model to the input data, which can help the model better understand the relationships and importance in the input data, and improve the performance and robustness of the model. The implementation principle of the attention mechanism can be simply summarized as follows: the representation vector of each element (e.g., word vector) in the input data is weighted and summed to generate a global context vector. Among them, the weight values are usually calculated based on some kind of similarity metric, such as dot product, cosine similarity, etc. Fig. 3 shows the model of attention score calculation, whose attention performance score is shown in equation (15):

$$f(x) = \sum_i a\left(x, x_i\right) y_i = \sum_{i=1}^{n} sotf \ \max\left(-\frac{1}{2}\left(x - x_i^2\right)\right) y_i \tag{15}$$

where α denotes the attention scoring function, through the Keys and Query input into the scoring function by obtaining the attention weights, and finally through the summing operation with Values to produce the attention output results.

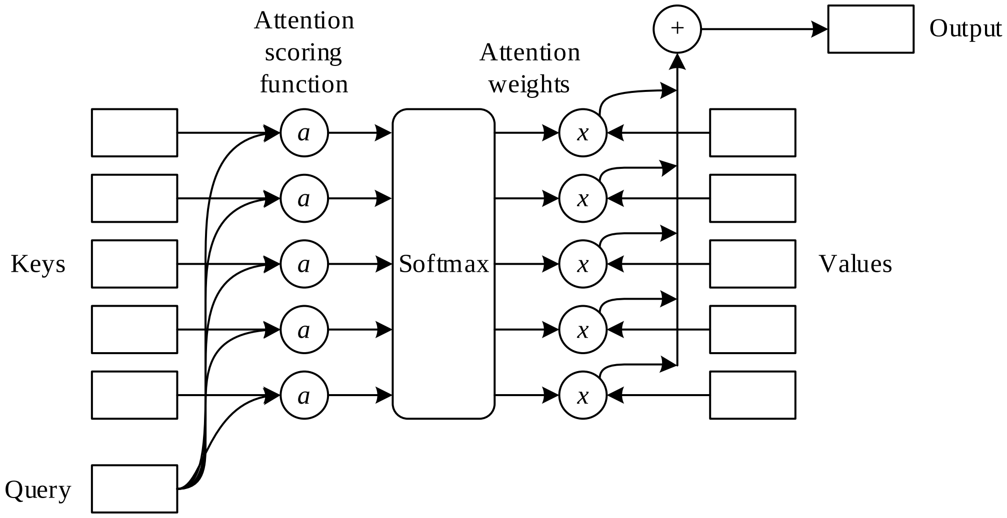


Figure 3: Calculation model of attention score

Specifically, the attention mechanism can usually be divided into the following steps:

(1) Calculate the attention weights: first, according to the representation vector of each element in the input data, calculate its similarity to the biased semantic information vector. This similarity calculation can be done using dot product, cosine similarity or other similarity measures. Then, these similarities are normalized to get the attention weight of each element.

(2) Weighted summation: multiply the representation vector of each element by its attentional weight, and then weight and sum them to get the global biased semantic information vector. This biased semantic information vector can be regarded as a summary representation of the input data, which contains a comprehensive consideration of the importance and relevance of all elements.

(3) Embedding model: the biased semantic information vector is connected or spliced with the representation vector of the original input data, and then fed into the subsequent deep learning model for training and inference.

2.4 Knowledge Graph Completion Algorithm for Fusing Linguistic Information

This paper proposes a knowledge graph completion method (SI-KGC) that incorporates semantic information. The method is improved on the basis of the original TransH model, increasing the extraction and utilization of the ternary semantic information by the translation model, improving the accuracy of the original translation model in knowledge graph completion, and obtaining a certain improvement in the evaluation indexes in knowledge graph completion.

2.4.1 Ternary Semantic Information Extraction

BERT is a pre-trained natural language processing model to extract semantic information from text data. In terms of the extraction of ternary semantic information, BERT model can be used to realize it. First, the ternary semantic information needs to be input into the BERT model according to a certain format. Entities and relations can be taken as inputs to the BERT model respectively, and some special markers, such as [CLS] and [SEP], can be added to distinguish the start and end positions of each input.The BERT model will preprocess, disambiguate and encode the input text, and then output a vector representation containing the extracted semantic information. The method of using the BERT model for ternary semantic information extraction can effectively combine natural language processing and deep learning techniques, providing another efficient and accurate way for text data analysis and processing. The input and output of the BERT model used in this paper are shown in Figure 4.

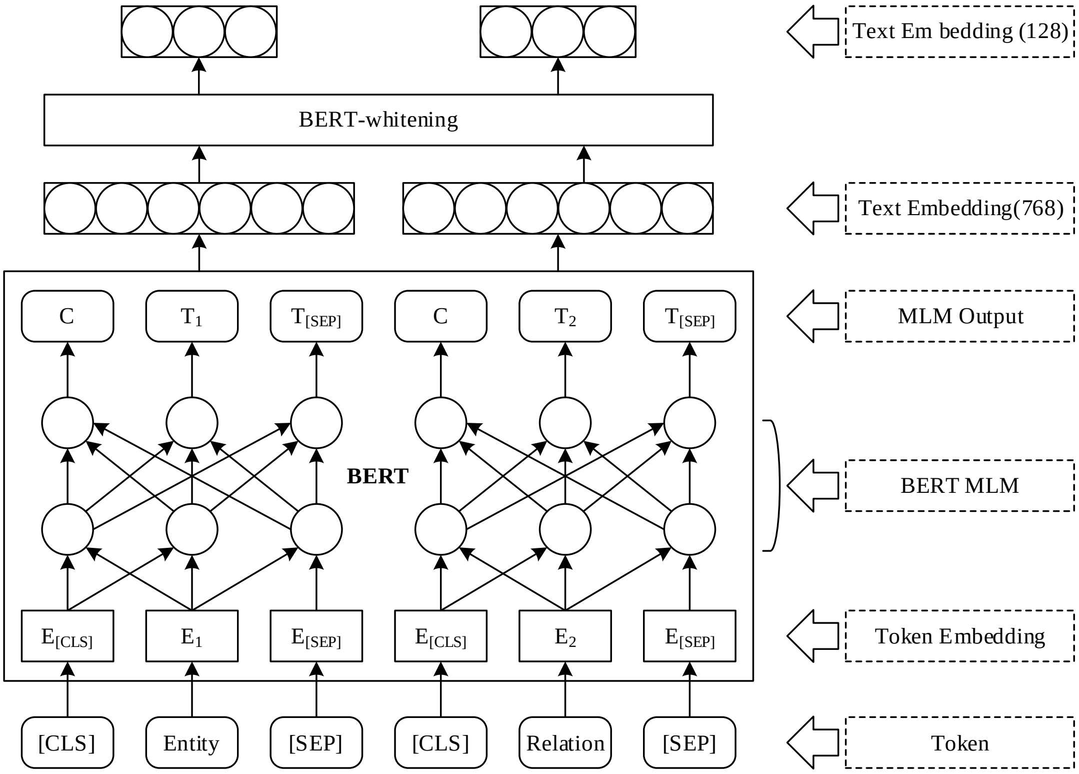


Figure 4: triplet semantic information extraction

The sentence $S=(t_1,t_2,\cdots t_n)$ consists of n subwords, and t_i denotes subwords, which are independent of each other. The language model likelihood value of a sentence S is equivalent to the product of the probabilities of all subwords, as shown in Equation (16):

$$\log P(S)=\sum_{i=1}^{n}\log P(t_i) \tag{16}$$

Merging two subwords x and y in neighboring quantitative positions, the merger produces a subword noted as z. The change in the likelihood value of the sentence S can be represented by Equation (17):

$$\log P(t_z)-\left(\log P(t_x)+\log P(t_y)\right)=\log\left(\frac{P(t_z)}{P(t_x)\,p(t_y)}\right) \tag{17}$$

From the above formulas (16) and (17), it can be concluded that the two subwords are strongly correlated on the language model and they often occur together in an expected neighboring manner.

The advantages of the WordPiece algorithm are that it can recognize out-of-the-way words and affixes, and it is highly flexible and scalable. Therefore, it is widely used in many natural language processing tasks, such as machine translation, named entity recognition, and text categorization. In addition, the WordPiece algorithm is one of the key components of deep learning models such as BERT, which plays an important role in improving the effectiveness of text processing.

2.4.2 Semantic feature dimensionality reduction

After the above ternary semantic information extraction method, the semantic word vectors generated by BERT were obtained. However, many studies have shown that the semantic vectors obtained using BERT are ineffective in similarity computation and 768-dimensional word vectors can suffer from problems such as memory challenges. Therefore, a simple vector whitening technique with reference to the BERT-whitening model is considered to improve the quality of word vectors, increase the effectiveness of semantic similarity computation, and at the same time reduce the dimensionality of the semantic vectors, so as to improve the efficiency of the subsequent computation of the semantic attention among ternary groups. Referring to the computation rules of BERT-whitening model, this paper defines a dimension transformation method, as shown in Equation (18):

$$\tilde{x}_i = \left(x_i - \mu\right)W \tag{18}$$

Let the set of (row) vectors be $\left\{x_i\right\}_{i=1}^{N}$, and perform the transformation shown in Eq. (18) such that $\left\{x_i\right\}_{i=1}^{N}$ has a mean of 0 and a covariance of the unit matrix. There are two parameters μ and W in Eq. Taking $\mu = \frac{1}{N}\sum_{i=1}^{x} x_i$ allows its $\left\{x_i\right\}_{i=1}^{N}$ mean to be 0. The parameter W is computed by noting the covariance matrix for the original data as Σ, and the covariance of the transformed data as $\tilde{\Sigma}$, it can be obtained that the transformed data U is an orthogonal matrix. A is a bi-diagonal matrix, and the diagonal matrix elements are all positive. The parameter $W = U\sqrt{A^{-1}}$ can be found according to the matrix transformation algorithm.

2.4.3 Attentional Information Fusion

Attention information fusion adopts the attention mechanism proposed by previous authors, by fusing the semantic information of the triad with the parameter vectors of the original model to input into the Attention structure to calculate the attention score of the triad's semantic information, and the specific operation method is shown in Fig. 5.

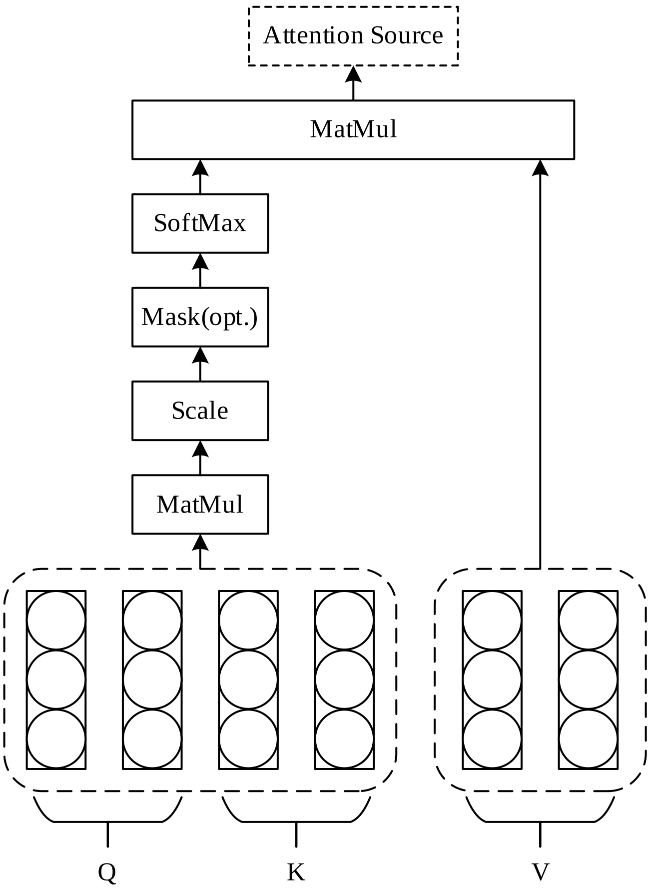


Figure 5: Attention score of semantic fusion

The semantic information vector and the parameter vector are summed to get the semantic information representation about the parameter. Q is the fusion of head entity vectors with semantic information vectors, K is the fusion of semantic information of tail entity vectors, and V is the fusion of relation vectors with semantic information. Q and K are multiplied to get the relation weight between each word vector, multiplied with the scaling factor $\frac{1}{\sqrt{d_2}}$ by softmax normalization, and then multiplied with V to finally get the attention output. The specific calculation method is shown in Equation (19):

$$Attention(Q,K,V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{19}$$

Positive and negative ternary get the attention score about the semantic information after the above operation of attention, after obtaining the scores of the two, the method adopts a new semantic information comparison loss function as shown in Eq. (20) (21):

$$l_{contr}^{x_i x_j}(W) = \frac{e^{sim\left(SiTcontr\left(x_i\right)SiTcontr\left(x_j\right)\right)/\tau}}{\sum_{k=1,k\neq i}^{2N} e^{sim\left(SiTcontr\left(x_i\right)SiTcontr\left(x_k\right)\right)/\tau}} \tag{20}$$

$$L = -\frac{1}{N}\sum_{j=1}^{N} \log l_{contr}^{x_j x_{\tilde{j}}}(W) \tag{21}$$

where x_i, x_j denote the positive and negative ternary semantic scores of the inputs, respectively, and $sim(\cdot,\cdot)$ denotes the cosine similarity, i.e. the dot product after normalization. τ is a hyper-temperature parameter used to regulate the degree of attention given to difficult samples. x_j, x_j denotes the input vectors from two different data augmentations under the same semantics. $S_iTcontr(\cdot)$ denotes the semantic information representation from the comparison.

The semantic information of positive and negative samples is optimized by the above loss function for contrast loss to get the final loss function of the model as shown in Eq. (22):

$$\begin{aligned} L = & \sum_{(h,r,t)\in\Delta}\ \sum_{(h',r',t')\in\Delta'_{(h,r,t)}} \left[f_r(h,t) + \gamma - f_{r'}\left(h',t'\right) \right]_+ \\ & + C\left\{ \sum_{e\in E}\left[\| e\|_2^2 - 1 \right]_+ + \sum_{r\in R}\left[\frac{\left(w_r^T d_r\right)^2}{\|d_r\|_2^2} - \varepsilon^2 \right]_+ + -\frac{1}{N}\sum_{j=1}^{N} \log l_{contr}^{x_x x_{\tilde{j}}}(W) \right\} \end{aligned} \tag{22}$$

By this method, the attention constraints of semantic information are added under the rules of the original soft constraints, so that different ternary groups can get their own semantic information, which is added to the original model to form an adaptive semantic information enhancement to the ternary group information.

3 Experimental design and analysis of results

3.1 Data set and experimental environment construction

3.1.1 Data sets

In order to better evaluate the joint model SI-KGC, it is usually validated on large knowledge graphs, such as Freebase, WordNet, etc. This section focuses on the FB15K dataset, WN18 dataset, and WN11 dataset used for the experiments.

FB15K is a commonly used knowledge graph dataset, which is a subset of the Freebase library.FB15K contains 14965 entities, 592,234 triples, and 1342 relationship types. Compared to other knowledge graph datasets, FB15K has higher density and noise, and also contains a large number of unknown entities and relationship types. Due to its richness and complexity, FB15K has become one of the standard datasets for many knowledge representation learning studies, and is also widely used to evaluate and compare the performance of different knowledge representation learning methods.

WN18 and WN11 are also commonly used to evaluate the performance of knowledge representation learning methods. This dataset consists of entities, relations extracted from WordNet. Compared to FB15K, WN18 and WN11 are smaller, denser and more regular datasets with a smaller number of triples, but still contain a variety of complex semantic relations. Due to their size, structure and semantic diversity, WN18 and WN11 are also widely used for experiments and research in the field of knowledge representation learning. The details of the datasets used for the experiments are shown in Table 1.

Table 1: Dataset

Dataset	Number of entities	Relationship number	Training set	Verification set	Test set
FB15K	14965	1342	472869	48966	57836
WN18	40936	20	141236	5012	5012
WN11	38648	10	112536	2600	10536

3.1.2 Experimental environment

In order to validate the effectiveness of the PTransD-HRS method, we use the commonly used knowledge graph complementary assessment method to conduct experiments and compare it with the benchmark model. The benchmark models mainly include the following categories:

TransE: The classical translation-based representation learning model, which converts entities and relations into low-dimensional vectors, and realizes representation learning by having the head entity plus relation vectors as close as possible to the tail entity vectors.

TransH: In order to overcome the limitation of the TransE method in dealing with complex relations, this model adopts the method of relation-specific hyperplane, which makes the same entity have different representations on different relation hyperplanes.

TransR: maps entities and relations into different spaces, imposing translation restrictions in the corresponding relation spaces.

TransD: proposed on the basis of TransR model, it maps entities and relations into new low-dimensional vector spaces for embedding learning, which is more suitable for dealing with the case of polysemy and ambiguous relations.

Improved TransH: This model adds hyperplanes to the TransH model as a way to uncover semantic correlations between entities.

The experiments were all written under the operating system: Ubuntu 18.04, programming environment: Python 3.6, Pytorch 1.7.0, GPU NVIDIA RXT3060, CPU AMD Ryzen 7 6800H, and 32G RAM.

3.2 Hyperparameter settings

The range of hyperparameters is set as follows:

(1) Vector dimensions of entities, relations $d \in \{50,80,100,150\}$

(2) Set the learning rate $\lambda \in \{0.001,0.002,0.005,0.01,0.1\}$ for SGDs

(3) Negative sampling parameter $n \in \{10,15,20,30\}$

(4) Interval $\gamma \in \{0.5,1,2,4\}$

(5) batch size $b \in \{40,120,240,480,1000,1500,4000\}$ in SGD

It is found through experiments that the optimal parameters of SI-KGC are $d=100$, $\lambda=0.001$, $n=20$, $\gamma=1$, and $b=480$. In terms of relational paths, a path with a step size of 3 is chosen in the experiments, and for the reliability of the paths, the same value is chosen here as that for the improved TransH: model, which is set to 0.01. The paradigm for the scoring function is adopted as the L1 paradigm, and the HRS structure with hyperparameters $\lambda_1 = 0.00001$, $\lambda_2 = 0.0001$, and $\lambda_3 = 0.001$, and the number of iterations of the experiment are all carried out for 1000 rounds.

3.3 Experimental tasks

The knowledge graph complementation task mainly consists of ternary group classification and link prediction.

3.3.1 Classification of ternary groups

Table 2 shows the ternary group classification results. Experimental results of ternary group classification on FB15K and WN11 datasets.SI-KGC indicates better performance of the model compared to TransE, with a combined improvement of 13.12% in accuracy on both datasets WN11 and FB15K. The accuracy is also improved by 3.12% compared to TransD. SI-KGC also shows an improvement in the performance of the model over the improved TransH model, which suggests that the inclusion of hyperplane computation in the relational paths enhances the inference ability of the model. On the other hand utilizing the hierarchical relational structure enables better vector representation of entities and relationships. This again validates the effectiveness of the proposed model.

Table 2: Classification results of spoon triples

Model	WN11(%)	FB15K (%)	Avg (%)
TransE	75.96	78.72	77.34
TransH	78.85	81.23	80.04
TransR	84.33	83.15	83.74
TransD	86.45	88.23	87.34
Improvement TransH	86.32	87.45	86.885
SI-KGC	88.56	92.36	90.46

In order to be able to more intuitively analyze the impact of parameter settings on model performance, the effect of vector representation dimensions on ternary classification accuracy is observed on the datasets FB15K and WN11, respectively, with a fixed learning rate $\lambda = 0.001$. Figure 6 shows the vector dimension debugging. It can be intuitively seen that on the one hand, the SI-KGC model has high accuracy in the ternary group classification task, with average accuracies of 83.8% and 74.75% on FB15K and WN11, respectively, when the parametric models are all the same. On the other hand, the highest accuracy of all three models is achieved when the embedding dimension of the model is d = 100.

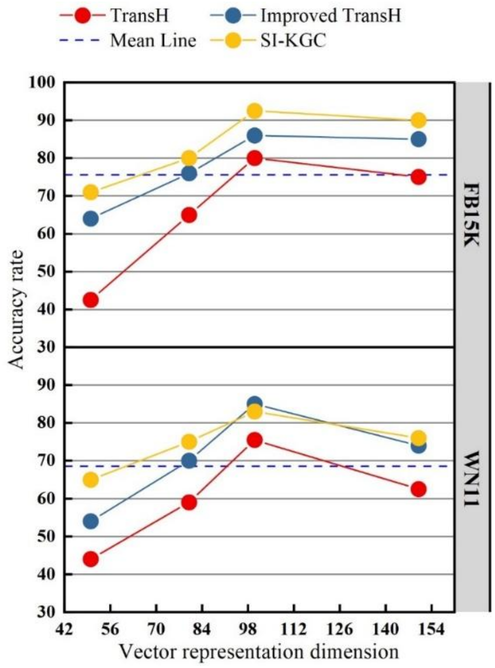


Figure 6: Vector dimensional debugging

3.3.2 Link projections

It is now common to use the link prediction task in Knowledge Graph Completion to evaluate the merits of an embedding model. The commonly used evaluation metrics are Hits@N (N=1,3,10), MR, and MRR. Hits@N indicates the proportion of correct entities that rank in the top N places in each prediction result. The larger its value, the better the model performance. MR indicates the average ranking of correct entities in all prediction results. The smaller the value, the higher the predicted rank of the correct result and the better the model performance. MRR denotes the inverse of the average rank of the correct entity in all the predicted results. The larger its value, the better the model performance. Their calculation formulas are as follows:

$$MR = \frac{1}{|M|}\sum_{j=1}^{|M|} rank\left(m_j\right) \tag{23}$$

$$MRR = \frac{1}{|M|}\sum_{j=1}^{|M|} \frac{1}{rank\left(m_j\right)} \tag{24}$$

$$Hits@N = \frac{|\{m \in M : m \leq N\}|}{|M|} \tag{25}$$

Compared to MR with Hits@N (N=1,3,10), the MRR metric is more resistant to the effect of anomalous data, so in this paper, the best value on the MRR metric is selected as the best experimental result for all datasets, and Hits@N (N=1,3,10) is selected as the second most important metric to evaluate the model performance.

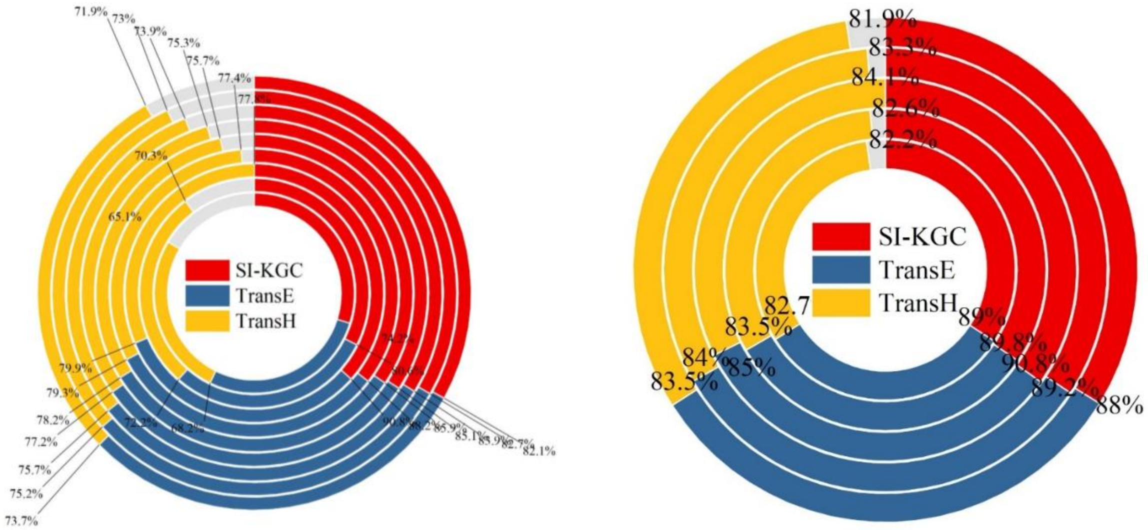
Table 3 shows the link prediction results, in the link prediction task, the datasets used are FB15K and WN18, in particular, two additional models are added: TransH-tm and TransH-mb. TransH-tm is the model that combines the top and middle layers of the hierarchical relational structure which is the relational cluster layer and the relational layer, and TransH-mb is the model that combines the middle layer of the hierarchical relational structure and the bottom layer also known as the relationship layer and the sub-relationship layer.

SI-KGC improves on all evaluation metrics with respect to other benchmark models.The Raw and Filter of MR are 185 and 41 respectively, which are the smallest among all methods.The Raw and Filter of HITS@10 are 56.15 and 91.26 respectively. The importance of including hyperplanes in relational paths for the model's reasoning ability is illustrated, as well as the value of the information learned from the BERT structure.

Table 3: Link prediction results

Dataset	FB15K				WN18			
Method	MR		HITS@10		MR		HITS@10	
	Raw	Filter	Raw	Filter	Raw	Filter	Raw	Filter
TransE	245	124	34.96	47.59	265	250	75.36	89.24
TransH	212	85	45.15	64.25	320	302	75.25	86.25
TransR	196	79	48.23	68.78	234	213	78.48	91.48
TransD	213	68	49.12	74.26	225	218	79.63	92.36
Improvement TransH	208	54	51.42	84.63	215	200	82.48	93.48
TransH-tm	203	53	53.23	87.86	185	212	83.24	93.25
TransH-mb	196	48	53.69	88.36	180	203	84.26	93.55
SI-KGC	**185**	**41**	**56.15**	**91.26**	212	205	81.25	94.35

In addition, DSI information is also highly scalable and very easy to combine with other models, here we extend DSI information to TransE and TransH models for comparison. Figure 7 shows the effect of the number of relationship clusters R1 and subrelationships R2 on the model performance, Fig. (a) shows the number of relationship clusters R1, and Fig. (b) shows the effect of the number of subrelationships R2 on the model performance, which represents 100~900 as well as 1~5 from the inside to the outside, respectively. In the FB15K dataset, it is clearly shown that when the number of relation clusters R1 reaches 300 and the number of sub-relations R2 reaches 3, the performance of the model is optimal. At this point, the SI-KGC HITS@10 values are 90.8% and 90.84% respectively. This is because: when the number of relationship clusters, R1, has a small value, semantically dissimilar relationships may be grouped together, introducing more noise. When the value of R1 is large, there is less information that can be used for each relation, resulting in worse performance, and when the number of subrelations, R2, is small, it also introduces noise and does not provide an adequate representation for each relation. When the R2 value is large, it results in a lack of training data for the subrelations, which also reduces the performance of the model.



(a)HITS@10 varies with the number of relational clusters

(b)HITS@10 varies with the number of subrelations

Figure 7: The effect of R1 and R2 quantity on model performance

3.3.3 Ablation experiments

In order to examine the role of each constituent module of the SI-KGC model, in this section, ablation experiments are conducted on the SI-KGC.The SI-KGC model mainly consists of two parts, the vertical axis part including the entity semantic mapping matrix and the horizontal axis part including the shared semantic mapping matrix, so ablation experiments are conducted on the overall model consisting of the horizontal axis and the vertical axis.

Table 4 shows the results of the ablation experiments for the SI-KGC model, where Group A represents the SI-KGC that utilizes only the vertical axis part for knowledge modeling, Group B represents the SI-KGC that utilizes only the horizontal axis part for knowledge modeling and the SI-KGC that participates in the modeling of the knowledge representation in both the horizontal axis parts.

The experiments have shown that the results obtained by modeling solely with the horizontal axis or the vertical axis are not as good as those of the SI-KGC model where both parts are involved. The SI-KGC model performs almost optimally on both datasets, except for H@1 on FB15K, which is 29.85. All other performances are the best. This is mainly due to the weak learning of entity-specific semantics in the horizontal axis part and the weak learning of shared semantics in the vertical axis part, and since different datasets are characterized by different features, there is a significant differentiation between the effects of the horizontal axis part and the vertical axis part.

Table 4: The results of the ablation experiment of the SI-KGC model

Grouping	Vertical axis	Horizontal axis	WN18			
			MRR	H@1	H@3	H@10
A	√		42.36	37.26	29.48	26.04
B		√	45.26	43.24	36.21	34.36
SI-KGC	√	√	56.48	53.36	54.36	66.23
Grouping	Vertical axis	Horizontal axis	FB15K			
			MRR	H@1	H@3	H@10
A	√		28.36	21.26	33.01	27.48
B		√	37.05	31.52	37.03	27.95
SI-KGC	√	√	41.36	29.85	42.21	62.38

3.3.4 Supplementary experiments

Table 5 shows the performance of TransE and SI-KGC on the FB15K dataset, and the TransE model and the SI-KGC model are excellent in all metrics. The performance gap between the two is not large, and the performance gap between different metrics ranges from 0.003 to 0.011, with the TransE model slightly outperforming the SI-KGC model.The excellent performance of the TransE model on FB15K verifies the effectiveness of relations as a vector of biased semantic information, and that of the SI-KGC model on FB15K verifies the effectiveness of the complete relational modeling capability to improve the effectiveness of model performance.

Table 5: Transe and SI-KGC performance on the FB15K data set

Model	MRR	HITS@N		
		10	3	1
TransE	0.349	0.539	0.385	0.245
SI-KGC	0.345	0.533	0.374	0.248

4 Conclusion

In this paper, the study of knowledge map complementation for fusing semantic information is based on the knowledge map of teaching and learning research. Based on the TransH model, entities are mapped onto the hyperplane corresponding to relations, which solves the problem that TransE cannot represent complex relations. Input the ternary semantic information into the BERT model according to a certain format, and then output a vector representation, introduce the attention mechanism, input the Keys and Query into the scoring function, obtain the attention weights, and derive the attention output results by summing with Values. The knowledge graph training set is constructed, the experimental environment and hyperparameters are set, and the ternary group classification, link prediction, and ablation experiments are performed on the SI-KGC model, respectively.

(1) In ternary group classification, the accuracy of SI-KGC model is improved by 13.12% compared with TransE on two datasets, WN11 and FB15K. It shows that the inclusion of hyperplane computation in the relational path can enhance the inference ability of the model.

(2) Evaluate the model's effectiveness in completing the task of link prediction in knowledge graph complementation by MR, MMR, and Hits@N metrics.The Raw and Filter of SI-KGC in MR are 185 and 41 respectively, which are the smallest among all the models, suggesting that the information learned from the BERT structure is valuable.

(3) The roles of each component module in the SI-KGC model were examined. The performance of the SI-KGC model on the two datasets was nearly optimal. The H@1 score on FB15K was 29.85, and the performance indicators on other datasets were all the highest, indicating the effectiveness of the performance of the SI-KGC model.

Funding

This work was supported by Joint application project of Liaoning Provincial Science and Technology Department (2024-MSLH-203).

About the Author

Lihua Duan was born in Yingkou, Liaoning, P. R. China, in 1975. She obtained a master's degree from Dalian Maritime University. I am currently an associate professor at Liaoning Police College. My main research directions are information science and management science, as well as complex system analysis.

References

[1] Storm, B. C., Stone, S. M., & Benjamin, A. S. (2017). Using the Internet to access information inflates future use of the Internet to access other information. Memory, 25(6), 717-723.

[2] Gubanova, O. A., Ivanov, A. Y., Krutolapov, A. S., Pelekh, M. T., & Gubanova, J. A. (2018). Reasons for the information processes utilization in the era of the platform industry 4.0. International Journal of Civil Engineering and Technology, 9(12), 43-47.

[3] Ji, S., Pan, S., Cambria, E., Marttinen, P., & Yu, P. S. (2021). A survey on knowledge graphs: Representation, acquisition, and applications. IEEE transactions on neural networks and learning systems, 33(2), 494-514.

[4] Peng, C., Xia, F., Naseriparsa, M., & Osborne, F. (2023). Knowledge graphs: Opportunities and challenges. Artificial intelligence review, 56(11), 13071-13102.

[5] Zhu, X., Li, Z., Wang, X., Jiang, X., Sun, P., Wang, X., ... & Yuan, N. J. (2022). Multi-modal knowledge graph construction and application: A survey. IEEE Transactions on Knowledge and Data Engineering, 36(2), 715-735.

[6] Hao, X., Ji, Z., Li, X., Yin, L., Liu, L., Sun, M., ... & Yang, R. (2021). Construction and application of a knowledge graph. Remote Sensing, 13(13), 2511.

[7] Zamini, M., Reza, H., & Rabiei, M. (2022). A review of knowledge graph completion. Information, 13(8), 396.

[8] Chen, Z., Wang, Y., Zhao, B., Cheng, J., Zhao, X., & Duan, Z. (2020). Knowledge graph

completion: A review. Ieee Access, 8, 192435-192456.

[9] Yin, P., Zhao, E., BianBaDroMa, & Ngodrup. (2025). A knowledge graph completion model based on weighted fusion description information and transform of the dimension and the scale. Applied Intelligence, 55(5), 336.

[10] Zhai, H., Lv, X., Hou, Z., Tong, X., & Bu, F. (2023). MLSFF: Multi-level structural features fusion for multi-modal knowledge graph completion. Mathematical Biosciences and Engineering, 20(8), 14096-14116.

[11] Wang, Y., Ning, B., Wang, X., Liu, C., & Li, G. (2025, July). Segmentation Similarity Enhanced Semantic Related Entity Fusion for Multi-modal Knowledge Graph Completion. In Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval (pp. 1176-1185).

[12] Wang, Y., Zhao, E., & Wang, W. (2022). A knowledge graph completion method based on fusing association information. IEEE Access, 10, 50500-50507.

[13] Akrami, F., Saeef, M. S., Zhang, Q., Hu, W., & Li, C. (2020, June). Realistic re-evaluation of knowledge graph completion methods: An experimental study. In Proceedings of the 2020 ACM SIGMOD International Conference on Management of Data (pp. 1995-2010).

[14] Xiong, H., Wang, S., Tang, M., Wang, L., & Lin, X. (2021). Knowledge graph question answering with semantic oriented fusion model. Knowledge-Based Systems, 221, 106954.

[15] Liu, C., Tang, J., Zeng, W., Wu, J., & Huang, H. (2023, September). Knowledge Graph Completion with Fused Factual and Commonsense Information. In International Conference on Web Information Systems and Applications (pp. 137-148). Singapore: Springer Nature Singapore.

[16] Zhu, H., Xu, D., Huang, Y., Jin, Z., Ding, W., Tong, J., & Chong, G. (2024). Graph structure enhanced pre-training language model for knowledge graph completion. IEEE Transactions on Emerging Topics in Computational Intelligence, 8(4), 2697-2708.

[17] Yuan, X., Chen, J., Wang, Y., Chen, A., Huang, Y., Zhao, W., & Yu, S. (2024). Semantic-enhanced knowledge graph completion. Mathematics, 12(3), 450.

[18] Sun, J., Xu, Y., & Li, Y. (2025). Semantic Enhancement Strategy for Knowledge Graph Completion. Future Trends in AI Research, 2(1).

[19] Hu, B., Zhang, J., & Chen, H. (2024). Knowledge Graph Completion Method of Combining Structural Information with Semantic Information. Chinese Journal of Electronics, 33(6), 1412-1420.

[20] Han, D., Jin, M., Zhang, J., Fan, L., Su, Q., & Qiao, B. (2026). FuST-KGC: Fusing sub-graph structures and textual semantics for knowledge graph completion. Neurocomputing, 132690.

[21] Liu, X., Wang, Z., Sun, Y., Han, J., Xiao, G., & Jiang, J. (2024). ISA-KGC: Integrated Semantics-Structure Analysis in Knowledge Graph Completion. IEEE Access, 12, 57250-57260.

[22] Li, Z., He, M., Chen, L., & He, T. (2024, September). A knowledge graph completion research combining structure and semantic information. In International Conference on Automation Control, Algorithm, and Intelligent Bionics (ACAIB 2024) (Vol. 13259, pp. 570-575). SPIE.

[23] He, H., Niu, H., & Feng, J. (2022). A Semantic Mapping Method of Relation Representation Enhancement for Few-Shot Knowledge Graph Completion. Electronics, 11(22), 3783.

[24] Li, X., Tian, Y., & Ji, S. (2024). Semantic and Relational-Based Graph Neural Network for Knowledge Graph Completion. Available at SSRN 4455564.

[25] Liang, S., Shao, J., Zhang, D., Zhang, J., & Cui, B. (2021). DRGI: Deep relational graph infomax for knowledge graph completion. IEEE Transactions on knowledge and data engineering, 35(3), 2486-2499.

[26] Chen, J., Zhang, W., Zhao, S., & Yin, Y. (2025). MgHiSal: MLLM-guided hierarchical semantic alignment for multimodal knowledge graph completion. Knowledge-Based Systems, 114552.

[27] Nie, H., Zhao, X., Bi, X., Ma, Y., & Yuan, G. Y. (2023). Correlation embedding learning with dynamic semantic enhanced sampling for knowledge graph completion. World Wide Web, 26(5), 2887-2907.

[28] Zhu, X., Xiao, L., Gao, A., Guo, L., & Yang, H. (2025). Temporal carbon neutral Chinese knowledge graph completion method based on latent semantic mining. Journal of Computational Science, 102744.

[29] Opdahl, A. L., Al-Moslmi, T., Dang-Nguyen, D. T., Gallofré Ocaña, M., Tessem, B., & Veres, C. (2022). Semantic knowledge graphs for the news: A review. ACM Computing Surveys, 55(7), 1-38.

[30] Lan, Y., He, S., Liu, K., Zeng, X., Liu, S., & Zhao, J. (2021). Path-based knowledge reasoning with textual semantic information for medical knowledge graph completion. BMC medical informatics and decision making, 21(Suppl 9), 335.