IEEE Access
Multidisciplinary ⁝ Rapid Review ⁝ Open Access Journal

Received 6 March 2024, accepted 30 March 2024, date of publication 3 April 2024, date of current version 26 April 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3384543

RESEARCH ARTICLE

SeSICL: Semantic and Structural Integrated Contrastive Learning for Knowledge Graph Error Detection

XINGYU LIU[1], JIELONG TANG[2], MENGYANG LI[3], JUNMEI HAN[4], GANG XIAO[4], AND JIANCHUN JIANG[3]

[1]State Key Laboratory of Intelligent Game, Institute of Software, Chinese Academy of Sciences, Beijing 100190, China
[2]School of Artificial Intelligence, Sun Yat-sen University, Guangzhou 510330, China
[3]Integrative Innovation Center, Institute of Software, Chinese Academy of Sciences, Beijing 100190, China
[4]National Key Laboratory for Complex Systems Simulation, Systemic Research Department, Systems Engineering Institute, AMS, Beijing 100192, China

Corresponding author: Jianchun Jiang (jianchun@iscas.ac.cn)

ABSTRACT As knowledge graphs (KGs) become more widely used in various applications, error detection for KGs has received more attention, which can reduce quality issues such as errors and inconsistencies. With the development of representation learning, embedding-based methods have significantly improved error detection performance. The recent error detection algorithm uses KG structural embedding loss and constructs a reasonable score function, ranking the confidence scores for each triplet. However, these methods ignore the factual semantics of the triplet itself, which primarily exist in the entities and relations descriptions text. Therefore, we propose Semantic and Structural Integrated Contrastive Learning(SeSICL) to simultaneously capture graph structural patterns and deep semantic features from descriptions text. Our method is based on contrastive learning without data augmentation, which utilizes encoder perturbations to generate contrasting views, making SeSICL highly suitable for complex error detection tasks and robust against real-world noise. We evaluate SeSICL on three baseline datasets with abnormal data and fluctuations. SeSICL outperforms the previous state-of-the-art methods, demonstrating our method's performance and robustness in more complex scenarios.

INDEX TERMS Knowledge graph, error detection, semantic embedding, structural embedding, contrastive learning.

I. INTRODUCTION

In the era of knowledge engineering, knowledge graphs (KGs) have been widely used in read-word applications, such as search engines [1], question-answer systems [2] and recommendation systems [3]. Knowledge graphs can extract, organize, and effectively manage knowledge from large-scale data that can greatly improve the quality of information services [4]. Some automatic and semi-automatic information extraction(IE) algorithms account for a large proportion of knowledge graph construction. However, due to the imperfect performance of IE models, some noise and knowledge conflicts were inevitably introduced into KGs. To ensure the correctness and robustness of downstream tasks, error detection has become an indispensable part of constructing and applying knowledge graphs.

Currently, KG representation learning algorithms based on embedding have been extensively researched. These methods encode entities and relations as continuous vectors and can learn feature representations adaptively, reducing the need for traditional feature engineering [5], [6], [7], [8]. An attractive approach is the embedding of structures, which reveal general patterns and rules among entities, facilitating the identification of potential inconsistencies in KG. Representative methods inculde TransE [9], TransH [10], and RotatE [11]. In addition, semantic information is also crucial

The associate editor coordinating the review of this manuscript and approving it for publication was Mu-Yen Chen.

to reducing the ambiguity between entities and relations. DKRL [12] uses CNN to encode the entity description information for semantic embedding. The KG-BERT [13] utilizes entity and relation descriptions as its input to facilitate KG completion via pre-trained language models. BLP [14] encodes the entity's description based on BERT for KG link prediction. Integrating both structural and semantic information is crucial for detecting inconsistency in KGs. StAR [15] proposes a hybrid model of textual encoding and graph embedding paradigms to learn contextualized and structured knowledge. LASS [16] utilizes a pre-trained language model to encode semantic information and ensures structural coherence using the TransE decoder.

However, in intricate error detection tasks, embedding-based approaches may struggle to capture error specifics, resulting in false negatives and positives. Furthermore, the presence of noise in the data can significantly influence the effectiveness of these methods. As a representative of self-supervised techniques, Contrastive Learning (CL) is a discriminative approach that seeks to minimize distances between similar samples while maximizing distances between diverse samples [17]. By doing so, CL can capture subtle differences between samples, effectively addressing complex error detection tasks and handling data noise and fluctuations under various conditions. Zhang et al. [18] apply contrastive learning in KG error detection tasks, generating various augmented views based on KG triplets and using two views for contrastive learning to improve error detection accuracy. While data augmentation is typically effective in graph contrastive learning, the inherent semantic characteristics presented within the knowledge graph can lead to additional errors, such as incorrect entities and relations, when constructing random enhanced views.

This paper proposes a Semantic and Structural Integrated Contrastive Learning (SeSICL) framework for Knowledge Graph Error Detection. SeSICL combines semantic and structural embeddings to capture deep semantic information about triplets, as well as the local and global structural characteristics of the KG. In the training phase, inspired by the SimGRACE approach [19], we avoid specific data augmentation techniques and instead leverage encoder perturbations to generate diverse contrasting views. This approach smoothly simulates potential errors and discrepancies within the KG, better preserving the original distribution characteristics of errors in KG and ensuring the robustness of contrastive learning. The contributions are summarized as follows.

- We propose a framework, SeSICL, that integrates semantic and structural information and uses a contrastive learning approach to capture deep features. This self-supervised method can effectively improve the accuracy and robustness of KG error detection tasks without additional manual annotation costs.
- We propose a two-layer approach that incorporates semantic and structural information. Based on a pre-trained language model, the semantic layer captures deep semantic information from entity and relation description text. The structural layer combines local and global structural information.
- We adopt a contrastive learning approach without data augmentation, incorporating Gaussian-based perturbations. Compared to specific contrastive learning data augmentation techniques, Gaussian-based perturbations smoothly simulating complex variations and noise within the KG, thereby adeptly preserving the original semantics of the KG. This contributes to a more robust KG error detection process.
- We validated the effectiveness of SeSICL on three real-world datasets, and the experiment results demonstrate that SeSICL achieves superior performance compared to previous state-of-the-art error detection models.

II. RELEVANT WORK

A. KNOWLEDGE GRAPH ERROR DETECTION

Knowledge graph error detection is a crucial task in the refinement of knowledge graphs [20], involving the identification of errors within the KG, rectifying knowledge inconsistent with objective facts. Initially, manual crowdsourcing [21], [22] and rule-based [23], [24], [25] error detection methods were employed. However, for large-scale KGs, inherent challenges such as high costs and poor scalability emerged. In contrast, statistical approaches offer superior applicability and efficiency. These include distribution-based outlier detection methods [26], [27], classical machine learning algorithms [28], [29] and graph exploration-based error detection techniques [30], [31]. In recent years, with the advancement of deep learning and neural networks, embedding-based error detection methods have garnered extensive research attention. Researchers attempt to map nodes and edges into continuous low-dimensional vector spaces while preserving KG structure and semantic features. For instance, [32] incorporates entity type information into the model, [33] explores using contextual information from text corpora to extend KG semantic structure, and [34] represents and models triplets and rules in a unified framework to enhance the model's predictive capabilities regarding facts.

B. KNOWLEDGE GRAPH EMBEDDING

Knowledge graph embedding aims to map a knowledge graph into a dense, low-dimensional feature space that can retain as much structural and attribute information of the graph as possible and facilitate computations of entities and relationships. Embedding methods can be divided into two categories: (i)triplet fact-based models and (ii)additional semantic information-based models. Triplet fact-based models treat the knowledge graph as a collection of all facts in the form of triplets, including models based on translation [9], [10], [35], [36], [37], tensor factorization [11], [38], [39], [40], and neural networks [41], [42], [43], [44]. However, methods that solely utilize factual triplets for knowledge embedding overlook the potential knowledge that

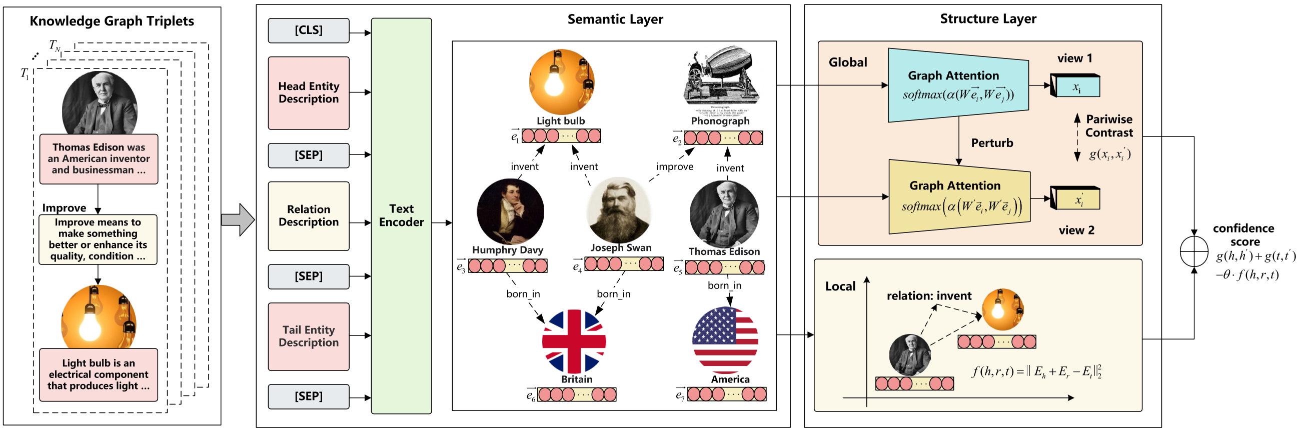


FIGURE 1. Overview of SeSICL. The semantic layer encodes entities and relations text descriptions, which serve as inputs for the structural layer. The structural layer integrates global and local scores through comprehensive optimization, ultimately providing the confidence score in the triplets.

the knowledge graph may contain, which hinders the precise representation of knowledge.

Additional semantic information-based models such as entity category information and text description, to enhance the performance of the traditional model. Xie et al. [32] proposed a TKRL model, which incorporates entity hierarchical type information and constraints between entities and relationships. In many practical large-scale knowledge graphs, detailed description information about entities and relationships exists. Xie et al. [12] also proposed a knowledge graph embedding model called DKRL, which employs two embedding models to encode the semantic descriptions of entities, integrating text information into the representation model to enhance representation learning. Further, Belth et al. [45] used soft rules extracted from text to improve the embedding model, allowing it to predict labels for unlabeled triplets and improve knowledge representation. KG-BERT [13] takes entity and relation descriptions as input and utilizes pre-trained language models for the KG completion task. LASS [16] involves fine-tuning pre-trained language models (LMs) to acquire a structured loss, where the forward pass of the LMs captures semantics, and the loss reconstructs structures. DLR-GAE [46] integrates both semantic and topological graph information, showcasing exceptional performance in semi-supervised classification tasks.

The above research shows that improving knowledge graph representation models with additional semantic information can improve their effectiveness. This is crucial for reducing the ambiguity between entities and relationships, which consequently improves the accuracy of downstream KG applications. However, existing research methods are not yet mature and still require considerable exploration.

C. GRAPH CONTRASTIVE LEARNING

In recent years, with the development of graph learning methods, a series of efforts have been made to improve the robustness and applicability of graph representations to better address real world challenges [47], [48], [49]. Notably, contrastive graph learning has shown its effectiveness in various graph-related knowledge tasks. Yang et al. [50] developed a versatile knowledge graph contrastive learning framework (KGCL) to mitigate information noise in recommendation systems. They incorporated additional supervised signals from the KG augmentation process to guide the cross-view contrastive learning paradigm. On the other hand, Zhang et al. [18] proposed a novel framework for contrastive knowledge graph error detection (CAGED), which extends the KG to different hypergraphs by treating each relationship triplet as a node. This incorporation of contrastive learning contributes to improving the efficiency of KG embeddings. In a related approach, Tan et al. [51] applied contrastive learning to the graph completion task, introducing more negative samples through contrastive learning to alleviate the sparsity often encountered in KGs.

III. PROPOSED METHODOLOGIES

We propose SEmantic and Structural Integrated Contrastive Learning (SeSICL), a novel framework to detect errors in the knowledge graph, which jointly learns semantic and structural information without manual annotation. As sketched in FIGURE 1, SeSICL consists of two main modules: the semantic layer and the structure layer. We utilize a pre-trained language model in the semantic layer to capture deep semantic representations of head entities, tail entities, and relations. The structure layer complements the semantic embedding to reconstruct the structural information of knowledge graphs, extracting local and global structural embeddings to derive triplet and node score functions. SeSICL effectively computes triplet confidence scores by optimizing local and global contrastive losses, facilitating efficient error identification in knowledge graphs.

A. PROBLEM STATEMENT

Knowledge graph error detection: Given a KG composed of triplets as $G = \{(h, r, t)\}$, where $h, t \in \boldsymbol{E}$ and $r \in \boldsymbol{R}$. $\boldsymbol{E}$ and

R indicate the set of entities and relations. The objective of this task is to develop a confidence scoring model $f : (h, r, t) \rightarrow \mathbb{R}^{(0,1)}$ that assigns confidence scores to triples within a range of 0 to 1. These confidence scores indicate the likelihood that a given triple is correct. A higher confidence score signifies a higher probability that the triple is accurate and reliable.

B. SEMATIC LAYER

In most existing large-scale knowledge graphs, entities, and relationships are accompanied by detailed descriptive information. Utilizing language models pre-trained on extensive textual corpora allows the capture of deep semantic information related to head entities, relationships, and tail entities. Computing the semantic similarity among these elements is crucial for verifying the accuracy of erroneous triplets. For instance, let us examine two triplets: (Thomas Edison, invent, light bulb) and (Thomas Edison, improve, light bulb). Although these two triplets have solid structural similarity, a closer examination of the specific semantic descriptions reveals that Thomas Edison played a role in enhancing the electric light bulb rather than being its original inventor. Consequently, the triplet (Thomas Edison, improve, Light bulb) is assigned a higher confidence score.

For a triplet (h,r,t), the head entity sequence h, the relation r, and the tail entity sequence t can be represented as $S^h = (s_1^h, s_2^h, \ldots, s_{kh}^h)$, $S^r = (s_1^r, s_2^r, \ldots, s_{kr}^r)$, $S^t = (s_1^t, s_2^t, \ldots, s_{kt}^t)$. We represent the model's input sequence by concatenating the sequence of the head entity, relationship sequence, and tail entity sequence with [CLS] and [SEP], which is in the following format: S = [CLS] S^h [SEP] S^r [SEP] S^t [SEP]. Next, we utilize a pretrained language model to convert all elements in each batch input into one-dimensional vectors, representing their distributed representations. The embedding and encoding output of LM can be expressed as E.

$$E_h, E_r, E_t = TextEncoder(S) \tag{1}$$

Here, the process of embedding using the pre-trained language model is denoted as *TextEncoder*, which varies depending on the chosen model. The collection of nodes after semantic embedding is represented as $\boldsymbol{E_E}$, and the set of relationships is denoted as $\boldsymbol{E_R}$. Specifically, for each triplet, we have the head and tail entities as $E_h, E_t \in \boldsymbol{E_E}$, and the relationship as $E_r \in \boldsymbol{E_R}$.

C. STRUCTURE LAYER

Structural information is crucial in assessing triplet correctness, comprising local and global aspects. Local structural information focuses on the internal representation of the triplet (h,r,t) itself. In contrast, global structural information considers the broader knowledge graph structure, encompassing higher-order relationships and the overall topological organization among entities. In this paper, we design a combined embedding approach incorporating local and global structural information.

1) LOCAL EMBEDDING

Firstly, we randomly initialize low-dimensional vector representations for each entity and relationship in the knowledge graph using Gaussian distribution sampling. Next, we construct local structural embeddings based on the semantic embeddings and the fundamental assumption *head* + *relation* ≈ *tail*. The triplet-level score function is defined accordingly.

$$f(h, r, t) = \|E_h + E_r - E_t\|_2^2 \tag{2}$$

2) GLOBAL EMBEDDING

The global structural embedding considers the global topological information between nodes and their direct or indirect neighbors. We adopt a graph attention network to mitigate potential erroneous associations in the knowledge graph. This model assigns attention scores to each relationship path, leading to more reliable and informative representations.

In the global structural embedding part, we employ a shared attention mechanism by computing the inter-feature correlations among nodes. The input consists of the characteristic representations of neighbors $\boldsymbol{a} = \{\vec{e_1}, \vec{e_2}, \ldots, \vec{e_K}\}$, where $\vec{e_j} \in \boldsymbol{E_E}$ and K is the number of entities. The attention coefficient between entities is calculated using the following formula.

$$e_{ij} = softmax\big(\alpha(W\vec{e_i}, W\vec{e_j})\big) \tag{3}$$

Here, $\boldsymbol{W} \in \mathbb{R}^{m \times n}$ is the weight matrix, where n represents the number of features for the node $\vec{e_i}$ The attention control function $\alpha : \mathbb{R}^m \times \mathbb{R}^m \rightarrow \mathbb{R}$ calculates the attention coefficients, following the approach in [52], and applies the softmax function for normalization.

Based on the attention coefficients between entities, we define the fused representation of node features, which incorporate the global structural information, as follows.

$$\boldsymbol{x_i} = sigmoid(\sum_{j=1}^{K} e_{ij}\vec{e_i}) \tag{4}$$

Simultaneously, we compute the attention coefficients after introducing perturbations.

$$e'_{ij} = softmax\left(\alpha\left(W'\vec{e_i}, W'\vec{e_j}\right)\right) \tag{5}$$

where $\boldsymbol{W'} \in \mathbb{R}^{m \times n}$ is the perturbed version of the weight matrix, represented as follows.

$$W' = W + \varepsilon \cdot \Delta W \tag{6}$$

where ε is the perturbation coefficient, representing the magnitude of perturbation, and ΔW is the perturbation term following a Gaussian distribution. Gaussian distribution is characterized by continuity and smoothness, enabling it to simulate variations and noise present in real-world data. This can be expressed as:

$$\Delta W \sim \mathbf{N}(0, \sigma^2) \tag{7}$$

where the mean of the Gaussian distribution is 0, and the variance is represented by σ^2. Gaussian perturbations are applied to each layer's parameters of the encoder, with the magnitude controlled by ε and the randomness of the perturbation determined by the parameters σ of the normal distribution. This perturbation mechanism introduces a degree of uncertainty during training, enabling the model to adapt more effectively to various data variations and noise, thereby enhancing generalization performance. The node characteristic representation after the introduction of perturbations is as follows.

$$\boldsymbol{x_i}' = sigmoid(\sum_{j=1}^{K} e_{ij}'\vec{e}_i) \tag{8}$$

The global-level score function for a node, which integrates the global structural features, is defined as the similarity between the node feature representations before and after introducing perturbations.

$$g(\boldsymbol{x_i}, \boldsymbol{x_i'}) = \frac{\boldsymbol{x_i}^T \boldsymbol{x_i}'}{\|\boldsymbol{x_i}\| \|\boldsymbol{x_i}'\|} \tag{9}$$

D. TRAINING OBJECTIVE

We employ a combined training strategy involving local and global losses during the training phase. Firstly, we compute the local loss using the triplet-level score function to ensure that the positive triplets score is higher than the negative triplets.

$$L_{local} = \sum_{(h,r,t)\in G} \sum_{(h',r',t')\in G'} \max(0, \gamma + f(h,r,t) - f(h',r',t')) \tag{10}$$

where $\gamma > 0$ is the boundary hyperparameter, and G' represents randomly generated negative triplets. The generation rules follow the work of [37], which involves randomly replacing the head, tail, or relation of the triplet to create negative examples.

$$G' = \{(h',r,t)|h' \in E\} \cup \{(h,r,t')|t' \in E\} \cup \{(h,r',t)|r' \in R\} \tag{11}$$

Next, we utilize the contrastive loss to compute the global loss based on the global-level score function. During training, we randomly select a minibatch of size N to form a subgraph. For each node feature $\boldsymbol{x_i}$ and its perturbed representation $\boldsymbol{x_i}'$, we create positive pairs $(\boldsymbol{x_i}, \boldsymbol{x_i}')$ and negative pairs $(\boldsymbol{x_i}, \boldsymbol{x_j}')$ with the perturbed representations of other N-1 nodes $\boldsymbol{x_j}'$. The contrastive learning loss aims to bring positive pairs closer together, minimizing the global-level score function $g(\boldsymbol{x_i}, \boldsymbol{x_i}')$ while pushing negative pairs further apart, increasing the $g(\boldsymbol{x_i}, \boldsymbol{x_j}')$.

$$L_{global} = -log \frac{\exp\left(g(\boldsymbol{x_i}, \boldsymbol{x_i}')\right)}{\sum_{j=1, j\neq i}^{n} \exp\left(g(\boldsymbol{x_i}, \boldsymbol{x_j}')\right)} \tag{12}$$

We define the final confidence score function as follows.

$$C(h,r,t) = sigmoid(g(h,h') + g(t,t') - \theta \cdot f(h,r,t)) \tag{13}$$

where θ is a scaling factor used to balance the scores of different dimensions of the triplets. Finally, we use the sigmoid function to map the score to the range [0,1]. A higher confidence score indicates that the triplet is more likely to be correct.

IV. EXPERIMENT

In this section, we conduct experiments to evaluate SeSICL and answer the following questions.

- Q1: How does the efficiency of SeSICL compare with other error detection methods?
- Q2: How do the semantic and structural components contribute to SeSICL?
- Q3: How do hyperparameters influence SeSICL?

A. EXPERIMENT SETUP

1) DATASETS

In this paper, we evaluate our model SeSICL on three popular benchmarks: FB15K-237, WN18RR and UMLS. FB15K-237 and WN18RR are general-purpose datasets, while UMLS represents a dataset from the medical domain. Their statistical information is summarized in TABLE 1. To answer Q1, we utilized three datasets to validate the model's performance across different knowledge graphs. For Q2 and Q3, we chose the general datasets FB15K-237 and WN18RR for validation instead of domain-specific datasets like UMLS, which can avoid introducing more irrelevant factors from specific domain characteristics.

TABLE 1. The statistics for datasets.

Dataset	#Triples	#Entities	#Relations
FB15K-237	310115	14541	237
WN18RR	93003	40943	11
UMLS	6529	135	46

The FB15K-237 dataset is an extension of the Freebase knowledge graph. It contains 237 relations, which is a reduced number compared to the original FB15K dataset. During its creation, the inverse relations present in the original dataset were removed, resulting in a more compact and diverse set of relations.

The WN18RR dataset is sourced from the WordNet knowledge graph and consists of triples representing relationships between entities in the form of (subject, relation, object). It is an improved version of the WN18 (WordNet 18) dataset. Compared to WN18, WN18RR has been fixed for some errors and inconsistencies, and some relations with insufficient information have been removed.

The UMLS (Unified Medical Language System) dataset is a comprehensive biomedical resource featuring concepts, terms, and relationships. The UMLS dataset includes triples representing relationships between entities in the form of (subject, relation, object). Each triple encapsulates valuable biomedical information, linking entities such as medical concepts, terms, or entities involved in healthcare-related contexts.

Regarding data descriptions, we follow the approach used in KG-BERT [13]. Specifically, for the WN18RR dataset, we employ Synsets from WordNet as entity descriptions. Initially, we retrieve synonyms through Synsets, concatenate multiple synonyms, and remove unnecessary symbols. For the FB15k-237 dataset and the UMLS dataset, we initially search Wikipedia for relevant entity descriptions. For entities successfully found in the search, we utilize a text summarization algorithm to produce concise entity summary information as entity descriptions. In cases where entities could not be found in Wikipedia, we use the entity names as entity descriptions. For all datasets, we directly employ relationship names as relationship descriptions.

2) EVALUATION METRICS

We adopt a reverse ranking approach for all triplet confidence score results, where higher-ranked triplets indicate a higher probability of being errors. Using this method, we identify the top K elements as anomalies. Following previous work [53], we assess the effectiveness of all methods using Precision and Recall, which are defined as:

$$Precision = \frac{TP}{TP + FP} = \frac{TP}{\mid K \mid} \tag{14}$$

$$Recall = \frac{TP}{TP + TN} = \frac{TP}{\mid Total\ Errors \mid} \tag{15}$$

where (TP) denotes the real error triplets correctly identified among the top K triplets. K is set to be equal to the number of total errors, resulting in equal Precision and Recall.

3) IMPLEMENTATION DETAILS

We use the NVIDIA RTX 3080 GPU server to train our proposed framework and all baselines. Our optimizer is Adam, with a learning rate 2e-5, and the batch size is fixed at 8. In our hyperparameter exploration, the perturbation coefficient ε varies from 0 to 10, the boundary hyperparameter γ from 0.1 to 1.0, and we fix the scaling factor θ at 0.1. The parameters of the normal distribution σ is set to 0.1. Additionally, we evaluated all methods by introducing a 5% anomaly into the standard dataset.

B. COMPARING WITH OTHER MODELS ON KG ERROR DETECTION TASK

To answer Q1, we compare SeSICL with various knowledge graph embedding methods, including self-supervised and semi-supervised approaches such as TransE [9], CKRL [37], KGTtm [31], KGIst [45], CAGED [18], KAEL [54]. And the experimental results are shown in TABLE 2.

TABLE 2. Comparing KG error detection precision with baselines models.

	FB15K-237	WN18RR	UMLS
TransE	0.488	0.331	0.282
CKRL	0.574	0.349	0.518
KGTtm	0.579	0.396	0.534
KGIst	0.569	0.379	-
CAGED	0.595	0.469	0.533
KAEL	0.815	0.479	0.669
SeSICL	**0.834**	**0.765**	**0.677**

Our proposed model achieved superior performance compared to typical KG embedding methods, and the results have been significantly improved on the baseline datasets. This is attributed to our model incorporating semantic and structural information (including local and global structures) embeddings and contrastive learning training methods. Without the need for additional annotated data, our model learns more discriminative deep features, enabling effective identification of potential errors in the knowledge graph.

C. ANALYZING THE CONTRIBUTION OF COMPONENTS

To address Q2, we conducted experiments to validate the effectiveness of different model components. Specifically, we performed experiments on three components: the semantic layer, the structural layer and the contrastive learning part. For the semantic layer, we replaced it with pre-trained language models of varying parameter sizes, including Albert and $Bert_{small}$. For the structural layer, we replaced it with different encoders such as SAGE and GCN. In addition, we explored various methods to generate contrastive views, specifically comparing the effectiveness of randomly perturbing edges, including deletion and addition, as well as randomly deleting nodes (according to [55]). The results are presented in TABLE 3.

TABLE 3. Comparing KG error detection precision with different components.

	FB15K-237	WN18RR
SeSICL(Bert$_{\text{base}}$)	**0.8340**	**0.7652**
var ALBert	0.6950	0.6490
var Bert$_{\text{small}}$	0.8078	0.7449
var SAGE	0.7992	0.7314
var GCN	0.8230	0.7600
var node	0.5380	0.5160
var edge	0.4900	0.4130

Firstly, by replacing the semantic layer with pre-trained language models of different sizes, we observed that the model's performance improved with increasing parameter size. However, large pre-trained models carry the risk of overfitting downstream contrastive learning tasks. By choosing an appropriate size for the pre-trained language model, we enable the model to generalize better to limited downstream task data.

Secondly, when replacing the classical graph convolutional encoders like SAGE and GCN in the structure layer, the performance of the models was inferior to our proposed SeSICL model. Our model employs a structure encoding approach based on attention mechanisms, which can partially

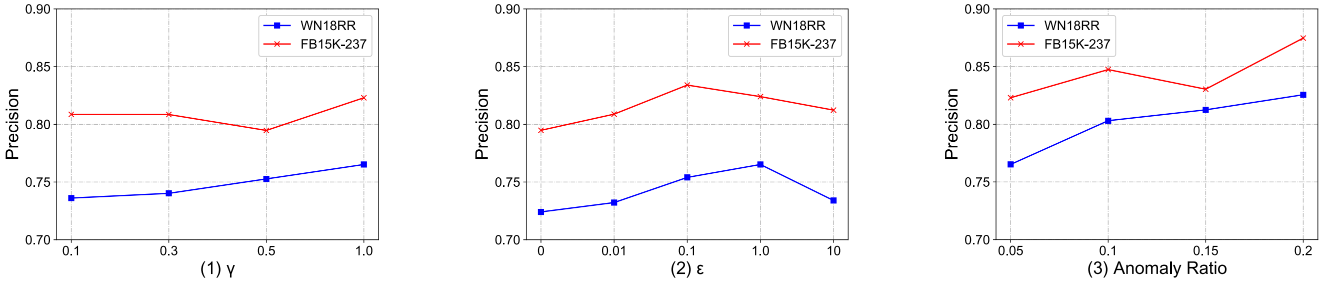


FIGURE 2. **Impact of hyperparameters.**

filter noisy triplets and alleviate the impact of poor-quality knowledge graphs on error detection effectiveness.

Finally, we replaced the Gaussian-based perturbation method with traditional contrastive learning data augmentation techniques, including random node deletions and random edge perturbations. Experimental results demonstrate that introducing Gaussian perturbations during the training phase, as opposed to traditional graph contrastive learning data augmentation methods, allows for a smoother adjustment of data representations. This effectively characterizes potential errors present in the KG, enabling the model to more easily capture invariant features of the KG and thereby improving error identification effectiveness.

D. ANALYZING THE EFFECTS OF HYPERPARAMETERS

For Q3, we varied the values of two hyperparameters in the model, namely the boundary hyperparameter γ and the perturbation coefficient ε. The results are shown in Figure 2.

The parameter γ is the boundary hyperparameter in the triplet-level score function, which controls the distance between positive and negative sample pairs. A smaller margin produces stricter constraints, bringing positive sample pairs closer together, while a larger margin relaxes the constraints and allows for larger distances between positive sample pairs. We set γ to vary between 0.1 and 1.0. It can be observed that different values of γ have minimal effect on the training results of the model, indicating that the model is already close to a local optimum in the hyperparameter space. Further adjustments to the margin may not significantly improve the model performance. Additionally, the model exhibits strong robustness, as changes in the margin have a minimal impact on its performance.

The perturbation coefficient ε represents the magnitude of perturbation in contrastive learning. We varied ε between 0 and 10. When the perturbation gradually increases, our model's performance initially improves gradually. Increasing the perturbation allows the model to learn invariant feature representations between perturbations better. However, when the perturbation becomes too large, it leads to a significant decline in model performance, possibly because of the perturbation of the inherent structure of the graph.

Furthermore, the anomaly quantity K is also an important metric for evaluating the effectiveness of the error detection model. We introduced errors at different proportions (5%, 10%, 15%, 20%) to simulate potential variations in the data quality of knowledge graphs in the real world and observed changes in model evaluation metrics. It is evident that under varying error proportions, the model's accuracy does not experience significant declines and consistently maintains an acceptable level. The experimental results affirm that the proposed error detection model can reliably and consistently deliver good performance, even when confronted with substantial fluctuations in the quality of the underlying knowledge graph. This indicates that our model exhibits good stability and can be applied to error detection scenarios in knowledge graphs with significant variations in data quality.

V. CONCLUSION

In this paper, we propose SeSICL, an effective model designed to identify erroneous triplets in knowledge graphs accurately. Our model leverages both semantic and structural information of the knowledge graph, enabling it to detect factual errors as well as schema errors. Additionally, our model employs a self-supervised contrastive learning approach, eliminating the need for additional data annotation while capturing deep coherence features within the data. Experimental results demonstrate that our model outperforms existing methods for KG error detection and maintains stable performance even when applied to knowledge graphs with significant variations in quality. In the future, we plan to explore further various knowledge graph refinement tasks based on our proposed model and validate its effectiveness in specific domain applications.

REFERENCES

[1] S. Szumlanski and F. Gomez, "Automatically acquiring a semantic network of related concepts," in *Proc. 19th ACM Int. Conf. Inf. Knowl. Manag.*, Oct. 2010, pp. 19–28.

[2] H. Sun, T. Bedrax-Weiss, and W. W. Cohen, "PullNet: Open domain question answering with iterative retrieval on knowledge bases and text," 2019, *arXiv:1904.09537*.

[3] H. Wang, M. Zhao, X. Xie, W. Li, and M. Guo, "Knowledge graph convolutional networks for recommender systems," in *Proc. World Wide Web Conf.*, May 2019, pp. 3307–3313.

[4] X. Chen, S. Jia, and Y. Xiang, "A review: Knowledge reasoning over knowledge graph," *Exp. Syst. Appl.*, vol. 141, Mar. 2020, Art. no. 112948.

[5] S. Guo, Q. Wang, L. Wang, B. Wang, and L. Guo, "Knowledge graph embedding with iterative guidance from soft rules," in *Proc. AAAI Conf. Artif. Intell.*, vol. 32, 2018, pp. 1–8.

[6] Y. Cheng, L. Chen, Y. Yuan, and G. Wang, "Rule-based graph repairing: Semantic and efficient repairing methods," in *Proc. IEEE 34th Int. Conf. Data Eng. (ICDE)*, Apr. 2018, pp. 773–784.

[7] L. A. Galárraga, C. Teflioudi, K. Hose, and F. Suchanek, "AMIE: Association rule mining under incomplete evidence in ontological knowledge bases," in *Proc. 22nd Int. Conf. World Wide Web*, May 2013, pp. 413–422.

[8] T. P. Tanon, D. Stepanova, S. Razniewski, P. Mirza, and G. Weikum, "Completeness-aware rule learning from knowledge graphs," in *Proc. 16th Int. Semantic Web Conf. (ISWC)*. Vienna, Austria: Springer, Oct. 2017, pp. 507–525.

[9] A. Bordes, N. Usunier, A. Garcia-Duran, J. Weston, and O. Yakhnenko, "Translating embeddings for modeling multi-relational data," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 26, 2013, pp. 1–9.

[10] Z. Wang, J. Zhang, J. Feng, and Z. Chen, "Knowledge graph embedding by translating on hyperplanes," in *Proc. AAAI Conf. Artif. Intell.*, vol. 28, 2014, pp. 1–8.

[11] Z. Sun, Z.-H. Deng, J.-Y. Nie, and J. Tang, "RotatE: Knowledge graph embedding by relational rotation in complex space," 2019, *arXiv:1902.10197*.

[12] R. Xie, Z. Liu, J. Jia, H. Luan, and M. Sun, "Representation learning of knowledge graphs with entity descriptions," in *Proc. AAAI Conf. Artif. Intell.*, vol. 30, 2016, pp. 1–7.

[13] L. Yao, C. Mao, and Y. Luo, "KG-BERT: BERT for knowledge graph completion," 2019, *arXiv:1909.03193*.

[14] D. Daza, M. Cochez, and P. Groth, "Inductive entity representations from text via link prediction," in *Proc. Web Conf.*, Apr. 2021, pp. 798–808.

[15] B. Wang, T. Shen, G. Long, T. Zhou, Y. Wang, and Y. Chang, "Structure-augmented text representation learning for efficient knowledge graph completion," in *Proc. Web Conf.*, Apr. 2021, pp. 1737–1748.

[16] J. Shen, C. Wang, L. Gong, and D. Song, "Joint language semantic and structure embedding for knowledge graph completion," 2022, *arXiv:2209.08721*.

[17] A. Jaiswal, A. R. Babu, M. Z. Zadeh, D. Banerjee, and F. Makedon, "A survey on contrastive self-supervised learning," *Technologies*, vol. 9, no. 1, p. 2, Dec. 2020.

[18] Q. Zhang, J. Dong, K. Duan, X. Huang, Y. Liu, and L. Xu, "Contrastive knowledge graph error detection," in *Proc. 31st ACM Int. Conf. Inf. Knowl. Manag.*, Oct. 2022, pp. 2590–2599.

[19] J. Xia, L. Wu, J. Chen, B. Hu, and S. Z. Li, "SimGRACE: A simple framework for graph contrastive learning without data augmentation," in *Proc. ACM Web Conf.*, Apr. 2022, pp. 1070–1079.

[20] H. Paulheim, "Knowledge graph refinement: A survey of approaches and evaluation methods," *Semantic Web*, vol. 8, no. 3, pp. 489–508, Dec. 2016.

[21] A. Zaveri, D. Kontokostas, M. A. Sherif, L. Bühmann, M. Morsey, S. Auer, and J. Lehmann, "User-driven quality evaluation of DBpedia," in *Proc. 9th Int. Conf. Semantic Syst.*, Sep. 2013, pp. 97–104.

[22] M. Acosta, A. Zaveri, E. Simperl, D. Kontokostas, F. Flöck, and J. Lehmann, "Detecting linked data quality issues via crowdsourcing: A DBpedia study," *Semantic Web*, vol. 9, no. 3, pp. 303–335, Apr. 2018.

[23] N. Ahmadi, T.-T.-D. Truong, L.-H.-M. Dao, S. Ortona, and P. Papotti, "RuleHub: A public corpus of rules for knowledge graphs," *J. Data Inf. Qual.*, vol. 12, no. 4, pp. 1–22, Dec. 2020.

[24] D. Kontokostas, P. Westphal, S. Auer, S. Hellmann, J. Lehmann, R. Cornelissen, and A. Zaveri, "Test-driven evaluation of linked data quality," in *Proc. 23rd Int. Conf. World Wide Web*, Apr. 2014, pp. 747–758.

[25] W. Fan, X. Liu, P. Lu, and C. Tian, "Catching numeric inconsistencies in graphs," in *Proc. Int. Conf. Manag. Data*, May 2018, pp. 381–393.

[26] H. Paulheim and C. Bizer, "Improving the quality of linked data using statistical distributions," *Int. J. Semantic Web Inf. Syst.*, vol. 10, no. 2, pp. 63–86, Apr. 2014.

[27] D. Wienand and H. Paulheim, "Detecting incorrect numerical data in dbpedia," in *Proc. 11th Int. Conf. Semantic Web, Trends Challenges (ESWC)*. Crete, Greece: Springer, May 2014, pp. 504–518.

[28] H. Paulheim, "Identifying wrong links between datasets by multi-dimensional outlier detection," in *Proc. WoDOOM*, 2014, pp. 27–38.

[29] H. Paulheim and H. Stuckenschmidt, "Fast approximate a-box consistency checking using machine learning," in *Proc. 13th Int. Conf. Semantic Web, Latest Adv. New Domains (ESWC)*. Crete, Greece: Springer, May 2016, pp. 135–150.

[30] J. Liang, Y. Xiao, Y. Zhang, S.-W. Hwang, and H. Wang, "Graph-based wrong IsA relation detection in a large-scale lexical taxonomy," in *Proc. AAAI Conf. Artif. Intell.*, vol. 31, 2017, pp. 1–7.

[31] S. Jia, Y. Xiang, X. Chen, and K. Wang, "Triple trustworthiness measurement for knowledge graph," in *Proc. World Wide Web Conf.*, May 2019, pp. 2865–2871.

[32] R. Xie, Z. Liu, and M. Sun, "Representation learning of knowledge graphs with hierarchical types," in *Proc. IJCAI*, 2016, pp. 2965–2971.

[33] Z. Wang, J. Li, Z. Liu, and J. Tang, "Text-enhanced representation learning for knowledge graph," in *Proc. Int. Joint Conf. Artif. Intell. (IJCAI)*, 2016, pp. 4–17.

[34] S. Guo, Q. Wang, L. Wang, B. Wang, and L. Guo, "Jointly embedding knowledge graphs and logical rules," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2016, pp. 192–202.

[35] Y. Lin, Z. Liu, M. Sun, Y. Liu, and X. Zhu, "Learning entity and relation embeddings for knowledge graph completion," in *Proc. AAAI Conf. Artif. Intell.*, vol. 29, 2015, pp. 1–7.

[36] H. Xiao, M. Huang, Y. Hao, and X. Zhu, "TransG : A generative mixture model for knowledge graph embedding," 2015, *arXiv:1509.05488*.

[37] R. Xie, Z. Liu, F. Lin, and L. Lin, "Does William Shakespeare really write hamlet? Knowledge representation learning with confidence," in *Proc. AAAI Conf. Artif. Intell.*, vol. 32, 2018, pp. 1–8.

[38] M. Nickel, V. Tresp, and H.-P. Kriegel, "A three-way model for collective learning on multi-relational data," in *Proc. ICML*, vol. 11, 2011, pp. 3104482–3104584.

[39] B. Yang, W.-T. Yih, X. He, J. Gao, and L. Deng, "Embedding entities and relations for learning and inference in knowledge bases," 2014, *arXiv:1412.6575*.

[40] S. Zhang, Y. Tay, L. Yao, and Q. Liu, "Quaternion knowledge graph embeddings," in *Proc. Adv. Neural Inf. Process. Syst.*, 32, 2019, pp. 1–11.

[41] A. Bordes, X. Glorot, J. Weston, and Y. Bengio, "A semantic matching energy function for learning with multi-relational data: Application to word-sense disambiguation," *Mach. Learn.*, vol. 94, pp. 233–259, Jan. 2014.

[42] X. Dong, E. Gabrilovich, G. Heitz, W. Horn, N. Lao, K. Murphy, T. Strohmann, S. Sun, and W. Zhang, "Knowledge vault: A web-scale approach to probabilistic knowledge fusion," in *Proc. 20th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2014, pp. 601–610.

[43] M. Schlichtkrull, T. N. Kipf, P. Bloem, R. van den Berg, I. Titov, and M. Welling, "Modeling relational data with graph convolutional networks," in *Proc. 15th Int. Conf. Semantic Web (ESWC)*. Crete, Greece: Springer, Jun. 2018, pp. 593–607.

[44] L. Cai and W. Y. Wang, "KBGAN: Adversarial learning for knowledge graph embeddings," 2017, *arXiv:1711.04071*.

[45] C. Belth, X. Zheng, J. Vreeken, and D. Koutra, "What is normal, what is strange, and what is missing in a knowledge graph: Unified characterization via inductive summarization," in *Proc. Web Conf.*, Apr. 2020, pp. 1115–1126.

[46] Z. Chen, Z. Wu, S. Wang, and W. Guo, "Dual low-rank graph autoencoder for semantic and topological networks," in *Proc. 37th AAAI Conf. Artif. Intell.*, 2023, pp. 4191–4198.

[47] L. Wu, Y. Chen, K. Shen, X. Guo, H. Gao, S. Li, J. Pei, and B. Long, "Graph neural networks for natural language processing: A survey," *Found. Trends Mach. Learn.*, vol. 16, no. 2, pp. 119–328, 2023.

[48] Z. Wu, L. Shu, Z. Xu, Y. Chang, C. Chen, and Z. Zheng, "Robust tensor graph convolutional networks via T-SVD based graph augmentation," in *Proc. 28th ACM SIGKDD Conf. Knowl. Discovery Data Mining*, Aug. 2022, pp. 2090–2099.

[49] Z. Chen, L. Fu, J. Yao, W. Guo, C. Plant, and S. Wang, "Learnable graph convolutional network and feature fusion for multi-view learning," *Inf. Fusion*, vol. 95, pp. 109–119, Jul. 2023.

[50] Y. Yang, C. Huang, L. Xia, and C. Li, "Knowledge graph contrastive learning for recommendation," in *Proc. 45th Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, Jul. 2022, pp. 1434–1443.

[51] Z. Tan, Z. Chen, S. Feng, Q. Zhang, Q. Zheng, J. Li, and M. Luo, "KRACL: Contrastive learning with graph context modeling for sparse knowledge graph completion," in *Proc. ACM Web Conf.*, Apr. 2023, pp. 2548–2559.

[52] P. Velickovic, G. Cucurull, A. Casanova, A. Romero, P. Liò, and Y. Bengio, "Graph attention networks," 2017, *arXiv:1710.10903*.

[53] M. Hossin and M. N. Sulaiman, "A review on evaluation metrics for data classification evaluations," *Int. J. Data Mining Knowl. Manag. Process*, vol. 5, no. 2, pp. 1–11, Mar. 2015.

[54] J. Dong, Q. Zhang, X. Huang, Q. Tan, D. Zha, and Z. Zihao, "Active ensemble learning for knowledge graph error detection," in *Proc. 16th ACM Int. Conf. Web Search Data Mining*, Feb. 2023, pp. 877–885.

[55] Y. You, T. Chen, Y. Sui, T. Chen, Z. Wang, and Y. Shen, ''Graph contrastive learning with augmentations,'' in *Proc. Int. Conf. Neural Inf. Process. Syst.*, vol. 33, 2020, pp. 5812–5823.

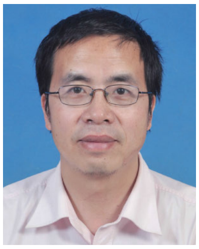
XINGYU LIU received the master's degree in software engineering from Beihang University, in 2021. She is currently an Engineer with the Institute of Software, Chinese Academy of Sciences. Her research interests include natural language processing (NLP) and knowledge graph.

JIELONG TANG received the master's degree in computer engineering from New York University, in 2022. He is currently pursuing the Ph.D. degree in computer science from Sun Yat-sen University. His research interests include knowledge graph, natural language processing, and multi-modal information extraction.

MENGYANG LI received the degree in electronic information science and technology (major) from Qingdao University of Technology, in 2018. Since 2021, he has been with the Institute of Software, Chinese Academy of Sciences. Currently, he is an Engineer with the Integrated Innovation Center, mainly responsible for research and project management in the field of knowledge graph.

JUNMEI HAN received the M.S. degree in military communication and the Ph.D. degree in information and communication engineering from the National University of Defense Technology, Changsha, China, in 2013 and 2019, respectively. She is currently an Associate Professor with the Systemic Research Department, Systems Engineering Institute, AMS, PLA, China. Her current research interests include complex system theory, natural language processing, and graph neural networks.

GANG XIAO received the B.S. and Ph.D. degrees in computer science and technology from the National University of Defense Technology. He is currently a Professor with the National Key Laboratory for Complex Systems Simulation, Systemic Research Department, Systems Engineering Institute, AMS, PLA. His research interests include complex system theory, system design, knowledge graph, and deep learning.



JIANCHUN JIANG received the Ph.D. degree in computer application technology from the Graduate School, Chinese Academy of Sciences. He is currently an Associate Researcher with the Institute of Software, Chinese Academy of Sciences. He has authored/coauthored more than 50 papers. His research interests include cybersecurity, software engineering, AI, and data science.