A Contextual Information-augmented Probabilistic Case-based Reasoning Model for Knowledge Graph Reasoning

Yuejia Wu and Jian-tao Zhou ✉

College of Computer Science, Inner Mongolia University
National & Local Joint Engineering Research Center of Intelligent Information Processing Technology for Mongolian
Engineering Research Center of Ecological Big Data, Ministry of Education
Inner Mongolia Engineering Laboratory for Cloud Computing and Service Software
Inner Mongolia Key Laboratory of Social Computing and Data Processing
Inner Mongolia Key Laboratory of Discipline Inspection and Supervision Big Data
Inner Mongolia Engineering Laboratory for Big Data Analysis Technology
Hohhot, China
`wuyuejia@imudges.com, cszjtao@imu.edu.cn`

Abstract. Knowledge Graph Reasoning (KGR) is one effective method to improve incompleteness and sparsity problems, which infers new knowledge based on existing knowledge. Although the probabilistic case-based reasoning (CBR) model can predict attributes for an entity and outperform other rule-based and embedding-based methods by gathering reasoning paths from similar entities in KG, it still suffers from some problems such as insufficient graph feature acquisition and omission of contextual relation information. This paper proposes a contextual information-augmented probabilistic CBR model for KGR, namely CICBR. The proposed model frame the reasoning task as the query answering and evaluates the likelihood that a path is valuable at answering a query about the given entity and relation by designing a joint contextual information-obtaining algorithm with entity and relation features. What's more, to obtain a more fine-grained representation of entity features and relation features, the CICBR introduces Graph Transformer for KG's representation and learning. Extensive experimental results on various benchmarks prominently demonstrate that the proposed CICBR model can obtain the state-of-the-art results of current CBR-based methods.

Keywords: Knowledge Graph Reasoning, Case-based Reasoning, Graph Neural Network, Graph Transformer, Query Answering

1 Introduction

A Knowledge Graph (KG) is essentially a semantic network that reveals the relations between entities. Existing KGs such as NELL [1], Freebase [2], and WordNet [3] have been widely used for Knowledge-Based Question Answering [4], Recommendation System [5], Anomaly Detection [6], etc. However, the KG is usually incomplete and sparse, which still suffers from some limitations in the above applications.

Knowledge Graph Reasoning (KGR) is one of the main methods to improve those problems by inferring new knowledge based on the existing knowledge in the KG. Recent studies on KGR have shown that case-based reasoning approaches [7–10] solve a new problem by retrieving "cases" that are similar to the given problem, which can achieve advanced results more than rule-based and embedding-based reasoning models. Although the approach proposed in literature [10] can outperform state-of-the-art (SOTA) methods and nearly match the best offline method, there are still two kinds of problems that influence its performance. Firstly, the learning of graph features is inadequate, which makes the model insensitive to finer-grained differences between entities and relations. Secondly, the effects of relations between entities are ignored when obtaining and inferring paths, which leads to a lack of effective reasoning paths and may affect the accuracy of prediction.

In this paper, we present a KGR model based on the contextual information-augmented probabilistic CBR algorithm, referred to as CICBR. Our approach first introduces Graph Transformer [11] architecture, which is a generalization of transformer neural network architecture for arbitrary graphs, to obtain more finer-grained entity and relation characteristics to reasoning, and then joint finds similar entities and relations through feature representations and generate contextual information to augment the probabilistic case-based reasoning model. The main contributions of this paper are as follows:

- We utilize the Graph Transformer architecture for learning and acquiring more detailed entity and relation feature representations. To the best of our current knowledge, this is the first approach that introduces the Graph Transformer architecture into the field of CBR-based KGR.
- We present a joint contextual information-obtaining algorithm with entity and relation features, namely J_{ER}-CIO. Compared to the previous contextual information acquiring algorithm, J_{ER}-CIO is the first method that considers the contextual relations between entities, which can obtain more effective inference paths by combining with contextual entity extraction to enhance inference performance.
- We conduct extensive experiments on three benchmark datasets and the results demonstrate that the proposed KGR model CICBR can effectively outperform existing CBR-based models and obtain optimal experimental results.

2 Related Work

In this section, we mainly state KG's incompleteness and sparsity [12] and briefly summarize the solution to these problems based on KGR technology. As shown in Fig. 1, we give the ubiquitous manifestations of incompleteness and sparsity in KG: (i) incompleteness of entities and relationships: entity incompleteness is represented by missing knowledge about people related to “Lionel Messi” and others; relationship incompleteness is instantiated in the absence of the “father & son” relationship of “Lionel Messi” and “Ciro Messi”. (ii) Sparsity of entities and relations: it can be intuitionistically found from Fig. 1 that the ratio of entities’ kind (7) and relations’ kind (3) to the number of facts (13) is relatively small.

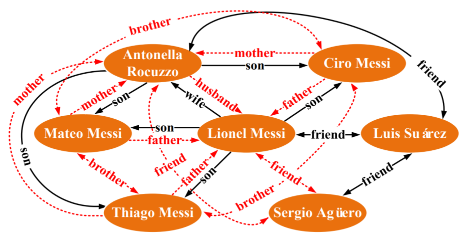

Fig. 1. An example of KG. The solid black line is existing relations, and the dashed red line is relations obtained by KGR.

KGR is one of the main methods to improve the above problems and recent studies have shown that CBR-based methods can obtain the most advanced reasoning results than rule-based and embedding-based models. In detail, the solutions in these methods [9, 10] first regard the reasoning task as a query answering, i.e. answering questions of the form (“*Antonella Rocuzzo*”, “*husband*”, ?). Then retrieve k similar entities (cases) to the query entity and find multiple KG paths, which are the solution to retrieved cases, to the entity they are connected by the query relation.

Although the advanced CBR-based KGR approach proposed in the literature [10] can gather reasoning paths from entities that are similar to the query entity and estimate parameters of the model efficiently using simple count statistics, there are still two problems. Firstly, the lack of learning ability of entity features and relation features in knowledge graphs leads to the insufficient perception of fine-grained differences between entities and relations when calculating the similarity of contextual information. Secondly, only entities that are similar to the query entity are considered when obtaining the contextual information, while the contextual information of the relations that are similar to the query relation is ignored, resulting in the loss of a large number of useful paths affecting the prediction results. This paper focuses on those problems and proposes a contextual information-augmented probabilistic CBR model CICBR that jointly obtains contextual entities and relations information from KG’s entities and relations feature representations, which are learned and generated from Graph Transformer architecture.

3 Methodology

In this section, we first introduce the preliminaries used in this paper, and then establish our proposed model in detail.

3.1 Preliminaries

In order to facilitate the understanding of the subsequent formulae, the general definition of KG is defined as follows:

Definition 1 (**Knowledge Graph**). A Knowledge Graph $\mathcal{G} = (V, E, \mathcal{R})$, where V represents the set of entities, $\mathcal{R}$ represents the set of binary relations, $E \subseteq V \times \mathcal{R} \times V$ represents the edges of the KG, and a KG is a collection of facts stored as triplets (e_1, r, e_2) where $e_1, e_2 \in V, r \in \mathcal{R}$. Also, following previous approaches[**?**], we add the inverse relation of every fact, i.e., for a fact $(e_1, r, e_2) \in E$, we add the fact (e_2, r^{-1}, e_1) to the KG. (If the set of binary relations $\mathcal{R}$ does not contain the inverse relation r^{-1}, it is added to $\mathcal{R}$ as well.)

This paper frames the reasoning as a query answering task on KG, i.e., answering questions of the form $(e_{1q}, r_q, ?)$, where the answer is an entity in the KG. Its definition is as follows:

Definition 2 (**Query Answering Task**). Given an input query of the form $(e_{1q}, r_q, ?)$, starting from vertex corresponding to e_{1q} in $\mathcal{G}$, the query answering model follows a path in the graph stopping at a node that it predicts as the answer.

The paths used in the query answering task in the KG are defined as follows [10]:

Definition 3 (**KG Paths**). A path $p = (e_1, r_1, e_2, r_2, ..., r_n, e_{n+1})$ with $st(p) = e_1$, $en(p) = e_{n+1}$, $len(p) = n$, and $type(p) = (r_1, r_2, ..., r_n)$ in a KG between two entities e_1 and e_{n+1} is defined as a sequence of alternating entities and relations that connect them. Let $\mathcal{P}$ represents the set of all paths in $\mathcal{G}$ and $\mathcal{P}_n \subseteq \mathcal{P} = \{p \mid len(p) \leq n\}$ be the set of all paths of length up to n. Let $P_n = \{type(p) \mid p \in \mathcal{P}_n\}$ denotes the set of all path types with length up to n and $P_n(e_1, r) \subseteq P_n$ represents all path types of length up to n that originate at e_1 by a direct edge of type r, i.e., if $S_{e_1 r} = \{e_2 \mid (e_1, r, e_2) \in \mathcal{G}\}$ is the set of entities that are connected to e_1 via a direct edge r, then $P_n(e_1, r)$ denotes the set of all path types of length up to n that start from e_1 and end at entities in $S_{e_1 r}$. Also, $\mathcal{P}_n(e_1, r)$ is defined to represent paths instead of path types.

3.2 The proposed Model ICCBR

In this section, we first present the overall architecture of the model, and then elaborate on the architecture in the next two subsections.

Overview

To improve the problems mentioned above, we propose the CICBR model to approach KGR. As shown in Fig. 2, the proposed CICBR architecture can be equipped with two stages: (a) joint contextual entities and relations information obtaining and (b) augmented probabilistic cased-based reasoning. Specifically, given an input KG and a query, CICBR's first stage, which will be presented in the next subsection, is to learn the graph feature representations from the Graph Transformer architecture and then utilize them to jointly acquire contextual entities and relations information by finding the entities and relations that are similar to the query entity and relation respectively. Next, CICBR's second stage, which will be described in the next subsection, aims to generate reasoning paths from contextual information, compute the score of each answer candidate and weigh paths with an estimate of their frequency and precision.

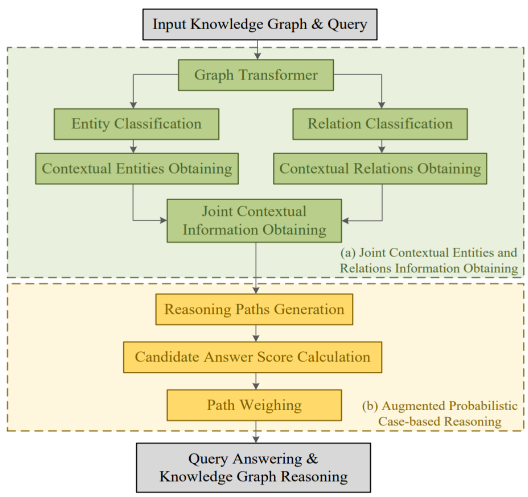

Fig. 2. Overview of the Proposed CICBR model.

Joint Contextual Entities and Relations Information Obtaining

In the previous CBR-based KGR model [10], each entity is represented as a sparse vector of its outgoing edge types, which is an extremely simple way of representing entities. However, this setting leads to a lack of ability to perceive entity characteristics and to gain fine-grained differences between entities, especially when clustering the entity and acquiring contextual entities information. To improve this problem, we present to utilize the Graph Transformer architecture [11] into the CBR field for better utilization of rich feature information available in the knowledge graph in the form of entity and relation attributes.

In addition, the previous work only considered the contextual entity information in the knowledge graph and ignored the importance of the relationship between entities, i.e. the contextual relations information, which may lead to the loss of a large number of valid inference paths and the failure to obtain the correct answer. To improve this problem, we propose a joint contextual information-obtaining algorithm named J_{ER}-CIO, which can effectively obtain the contextual entities and contextual relations information, and contribute to the generation of candidate paths in the next stage.

Specifically, following previous approaches [11], we pre-compute the Laplacian eigenvectors of knowledge graphs, which are defined as follows:

$$\Delta = I - D^{-\frac{1}{2}} A D^{-\frac{1}{2}} = U^T \Lambda U \tag{1}$$

where A is a $n \times n$ adjacency matrix, D is a degree matrix, and Λ, U correspond to the eigenvalues and eigenvectors respectively.

After pre-computing the Laplacian eigenvectors of KGs, we prepare the input entity and relation embeddings passed to the Graph Transformer. For a Graph $\mathcal{G}$ with entity features $\alpha_i \in \mathbb{R}^{d_n \times 1}$ for each entity i and relation features $\beta_{ij} \in \mathbb{R}^{d_e \times 1}$ for each relation between entities i and j, the input entity features α_i and relation features β_{ij} are passed through a linear projection to embed those to d-dimensional hidden features h_i^0 and e_{ij}^0.

$$\begin{aligned} \hat{h}_i^0 &= A^0 \alpha_i + a^0 \\ e_{ij}^0 &= B^0 \beta_{ij} + b^0 \end{aligned} \tag{2}$$

where $A^0 \in \mathbb{R}^{d \times k}$, $B^0 \in \mathbb{R}^{d \times d_e}$ and a^0, $b^0 \in \mathbb{R}^d$ are the parameters of the linear projection layers. Then, we embed the pre-computed entity positional encoding of dim k via a linear projection and add to the entity features $\hat{h}_i^0$.

$$\begin{aligned} \lambda_i^0 &= C^0 \lambda_i + c^0 \\ h_i^0 &= \hat{h}_i^0 + \lambda_i^0 \end{aligned} \tag{3}$$

where $C^0 \in \mathbb{R}^{d \times k}$ and $c^0 \in \mathbb{R}^d$.

Then, as shown in Fig. 3, we define the layer update equations for a layer ℓ as follows:

$$\hat{h}_i^{\ell+1} = O_h^{\ell} \mathop{\Big\|}_{k=1}^{H} \Big(\sum_{j \in \mathcal{N}_i} w_{ij}^{k,\ell} V^{k,\ell} h_j^{\ell} \Big) \tag{4}$$

$$\hat{e}_{ij}^{\ell+1} = O_e^{\ell} \mathop{\Big\|}_{k=1}^{H} (\hat{w}_{ij}^{k,\ell}) \tag{5}$$

$$w_{ij}^{k,\ell} = Softmax_j(\hat{w}_{ij}^{k,\ell}) \tag{6}$$

$$\hat{w}_{ij}^{k,\ell} = \Big(\frac{Q^{k,\ell} h_i^{\ell} \cdot K^{k,\ell} h_j^{\ell}}{\sqrt{d_k}} \Big) \cdot E^{k,\ell} e_{ij}^{\ell} \tag{7}$$

where $Q^{k,\ell}$, $K^{k,\ell}$, $V^{k,\ell}$, $E^{k,\ell} \in \mathbb{R}^{d_k \times d}$, O_h^ℓ, $O_e^\ell \in \mathbb{R}^{d \times d}$, $k = 1$ to H represents the number of attention heads, and $\|$ denotes a concatenation operation.

The attention outputs $\hat{h}_i^{\ell+1}$ and $\hat{e}_{ij}^{\ell+1}$ are then passed to separate Feed Forward Network (FFN) preceded and succeeded by residual connections and normalization layers as follows:

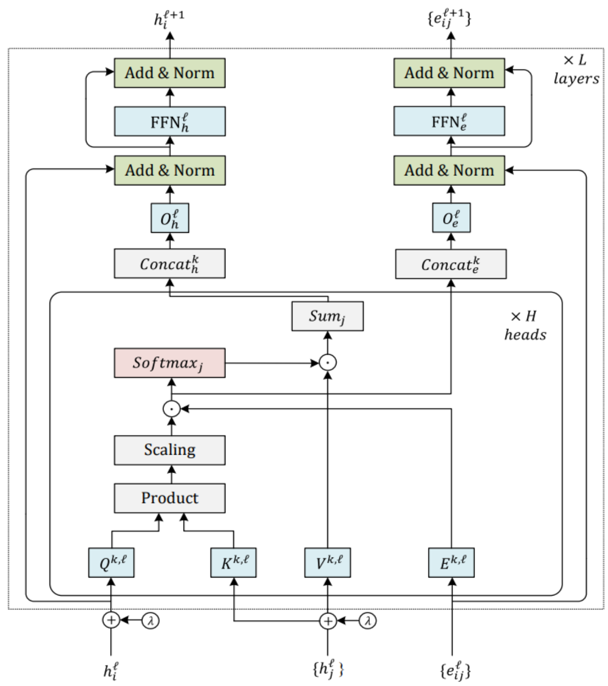

Fig. 3. Overview of the Graph Transformer architecture.

$$\hat{\hat{h}}_i^{\ell+1} = Norm(h_i^\ell + \hat{h}_i^{\ell+1}) \tag{8}$$

$$\hat{\hat{\hat{h}}}_i^{\ell+1} = W_{h,2}^\ell ReLU(W_{h,1}^\ell \hat{\hat{h}}_i^{\ell+1}) \tag{9}$$

$$h_i^{\ell+1} = Norm(\hat{\hat{h}}_i^{\ell+1} + \hat{\hat{\hat{h}}}_i^{\ell+1}) \tag{10}$$

$$\hat{\hat{e}}_{ij}^{\ell+1} = Norm(e_{i,j}^\ell + \hat{e}_{ij}^{\ell+1}) \tag{11}$$

$$\hat{\hat{\hat{e}}}_{ij}^{\ell+1} = W_{e,2}^\ell ReLU(W_{e,1}^\ell \hat{\hat{e}}_{ij}^{\ell+1}) \tag{12}$$

$$e_{ij}^{\ell+1} = Norm(\hat{\hat{e}}_{ij}^{\ell+1} + \hat{\hat{\hat{e}}}_{ij}^{\ell+1}) \tag{13}$$

where $W_{h,1}^{\ell} \in \mathbb{R}^{2d\times d}$, $W_{h,2}^{\ell} \in \mathbb{R}^{d\times 2d}$, $\hat{\hat{h}}_i^{\ell+1}$, $\hat{\hat{\hat{h}}}_i^{\ell+1}$, $W_{e,1}^{\ell} \in \mathbb{R}^{2d\times d}$, $W_{e,2}^{\ell} \in \mathbb{R}^{d\times 2d}$, $\hat{\hat{e}}_{ij}^{\ell+1}$, $\hat{\hat{\hat{e}}}_{ij}^{\ell+1}$ represent intermediate representations.

After obtaining entity and relation features representation, we first find similar entities to the query entity that has at least a relation r_q, i.e., for a query ("$Lionel\ Messi$", "$works\_for\_country$", ?), if there is ("$Emiliano\ Martínez$", "$works\_for\_country$", "$Argentina$"), the "$Emiliano\ Martínez$" may be considered and those entities are regarded as 'contextual entities'.

Different from previous approaches, then we added additional 'contextual relations' to get more useful reasoning paths by finding similar relations linked to the query entity e_{1_q}, i.e., we would consider "$lives\_in\_country$" if we obverse ("$Lionel\ Messi$", "$lives\_in\_country$", "$Argentina$"). Therefore, we let $E_{c_e,q}$ and $E_{c_r,q}$ denote the set of contextual entities and contextual relations for the query q respectively.

To compute $E_{c_e,q}$ and $E_{c_r,q}$, we propose a joint contextual information-obtaining algorithm with entity and relation features namely J_{ER}-CIO, which is shown in Algorithm. 1. Lines 3-18 are to acquire the input knowledge graph representation with entity features and relation features for contextual information obtaining in the next step. Lines 19-26 sort entities with respect to their cosine distance with respect to query entity and select the K_1 entities with the least distance and which have the query relation r_q. For each contextual entity e_c, we gather the path types that connect e_c to the entities it is connected by the relation r_q. Lines 27-34 sort relations with respect to their cosine distance with respect to query relation and select the K_2 relations with the least distance and which connect the query entity e_{1_q}. For each contextual relation r_c, we aggregate the path types starting from e_{1_q} and containing similarity relation r_c. These extracted path will be used to reason about the query entity.

Augmented Probabilistic Case-based Reasoning

After jointly obtaining the contextual entities and relations, we give the representation of the entity retrieved according to the context information as follows:

$$P_n(E_{c_e,q}, r_q) = \bigcup_{e_c \in E_{c_e,q}} P_n(e_c, r_q) \tag{14}$$

$$P_n(e_{1_q}, E_{c_r,q}) = \bigcup_{r_c \in E_{c_r,q}} P_n(e_{1_q}, r_c) \tag{15}$$

Then, the probability of finding the answer entity e_2 given the query is given by:

Algorithm 1 Joint contextual information-obtaining algorithm with entity and relation features (J_{ER}-CIO)

Input: Knowledge Graph $\mathcal{G} = (V, E, \mathcal{R})$ with entity features α_i for each entity i, positional encoding λ_i for each entity i, and relation features β_{ij} for each relation between entities i and j, query q with entity feature $h_{e_{1_q}}$ and relation feature e_{r_q}, and hyper-parameters K_1 and K_2.

Output: A set of contextual entities $E_{c_e,q}$ of query q and a set of contextual relations $E_{c_r,q}$ of query q.

$E_{c_e,q} \leftarrow []$
$E_{c_r,q} \leftarrow []$
% Graph feature acquiring
for i in V **do**
    $h_i^0 \leftarrow A^0\alpha_i + C^0\lambda_i + a^0 + c^0$
    for $j.isNeighbor(i)$ **do**
        $e_{ij}^0 \leftarrow B^0\beta_{ij} + b^0$
    end for
    for ℓ in L **do**
        $\hat{h}_i^{\ell+1} \leftarrow O_h^\ell \Big\Vert_{k=1}^{H} (\sum_{j \in \mathcal{N}_i} Softmax_j((\frac{Q^{k,\ell}h_i^\ell \cdot K^{k,\ell}h_j^\ell}{\sqrt{d_k}}) \cdot E^{k,\ell}e_{ij}^\ell)V^{k,\ell}h_j^\ell)$
        $\hat{e}_{ij}^{\ell+1} \leftarrow O_e^\ell \Big\Vert_{k=1}^{H} ((\frac{Q^{k,\ell}h_i^\ell \cdot K^{k,\ell}h_j^\ell}{\sqrt{d_k}}) \cdot E^{k,\ell}e_{ij}^\ell)$
        $h_i^{\ell+1} \leftarrow Norm(Norm(h_i^\ell + \hat{h}_i^{\ell+1}) + W_{h,2}^\ell ReLU(W_{h,1}^\ell \hat{\hat{h}}_i^{\ell+1}))$
        $e_{ij}^{\ell+1} \leftarrow Norm(Norm(e_{i,j}^\ell + \hat{e}_{ij}^{\ell+1}) + W_{e,2}^\ell ReLU(W_{e,1}^\ell \hat{\hat{e}}_{ij}^{\ell+1}))$
    end for
    if $i.equal(e_{1_q})$ **then**
        $h_{e_{1_q}} \leftarrow h_i^L$
    end if
end for
% Contextual entities obtaining
$similarity_e \leftarrow []$
for i in V **do**
    $similarity_e.add(\frac{h_i^L \cdot h_{e_{1_q}}}{|h_i^L||h_{e_{1_q}}|})$
end for
for k in K_1 **do**
    $E_{c_e,q}.add(reverseSort(similarity_e)[k])$
end for
% Contextual relations obtaining
$similarity_r \leftarrow []$
for r in $\mathcal{R}$ **do**
    $similarity_r.add(\frac{e_r^L \cdot e_{r_q}}{|e_r^L||e_{r_q}|})$
end for
for k in K_2 **do**
    $E_{c_r,q}.add(reverseSort(similarity_r)[k])$
end for
return $E_{c_e,q}, E_{c_r,q}$

$$
\begin{aligned}
P(e_2 \mid e_{1q}, r_q) &= \sum_{p \in P_n(E_{c_e,q}, r_q) \cup P_n(e_{1q}, E_{c_r,q})} P(e_2, p \mid e_{1q}, r_q) \\
&= \sum_p P(p \mid e_{1q}, r_q) P(e_2 \mid p, e_{1q}, r_q)
\end{aligned} \tag{16}
$$

Next, we marginalize the random variable representing the path types obtained from $E_{c_e,q}$ and $E_{c_r,q}$. $P(p \mid e_{1q}, r_q)$ denotes the probability of finding a path type given the query, which captures how frequently each path type co-occurs with a query and represents the prior probability for a path type. $P(e_2 \mid p, e_{1q}, r_q)$ captures the proportion of times, when a path type p is traversed starting from the query entity, we reach the correct answer instead of some other entity, which can be understood as capturing the likelihood of reaching the right answer or the "precision" of a reasoning path type.

Follow the settings in the previous approaches, we let c be a random variable representing the cluster assignment of the query entity. Then for the path-prior term, we have

$$
P(p \mid e_{1q}, r_q) = \sum_c P(c \mid e_{1q}, r_q) P(p \mid c, e_{1q}, r_q) \tag{17}
$$

where $P(c \mid e_{1q}, r_q)$ is zero for all clusters except the cluster in which the query entity belongs to. And if $c_{e_{1q}}$ is the cluster in which the e_{1q} has been assigned, then $P(p \mid c_{e_{1q}}, e_{1q}, r_q) = P(p \mid c_{e_{1q}}, r_q)$. Instead of per-entity parameters, we now aggregate statistics over entities in the same cluster and have per-cluster parameters. To perform clustering, we use hierarchical agglomerative clustering with average linkage mentioned in the literature [10] with the entity-entity similarity defined in Section 3.2.2.

Then, we estimate parameters by simple count statistics from the KG. i.e., the path prior $P(p \mid c, r_q)$ is estimated as follows:

$$
\frac{\sum_{e_c \in c} \sum_{p' \in \mathcal{P}_n(e_c, r_q) \cup \mathcal{P}_n(e_{1q}, r_c)} \mathbb{1}[type(p') = p]}{\sum_{e_c \in c} \sum_{p' \in \mathcal{P}_n(e_c, r_q) \cup \mathcal{P}_n(e_{1q}, r_c)} \mathbb{1}} \tag{18}
$$

For each entity in cluster c, we consider the paths that connect e_c to entities it is directly connected to via edge type r_q and its contextual relations E_{c_r}. The path prior for a path type p is calculated as the proportion of times the type of paths in $\mathcal{P}_n(e_c, r_q)$ is equal to p. Note that in Eq.18, if a path type occurs multiple times, all instances are counted. Similarly, the path-precision probability $(P(e_2 \mid p, c, r_q))$ can be estimated as follows:

$$
\frac{\sum_{e_c \in c} \sum_{p' \in \mathcal{P}_n(e_c)} \mathbb{1}[type(p') = p] \cdot \mathbb{1}[en(p') \in S_{e_c r_q}]}{\sum_{e_c \in c} \sum_{p' \in \mathcal{P}_n(e_c)} \mathbb{1}[type(p') = p]} \tag{19}
$$

where $\mathcal{P}_n(e_c)$ represents the paths of up to length n starting from the entity e_c, $en(p)$ represents the end entity for a path p and $S_{e_c r_q}$ represents the set of entities that are connected to e_c through a direct relation of type r_q. Then, given

r_q, the Eq.19 estimates the proportion of times the path p successfully ends at one of the answer entities when starting from e_c.

In general, given a query (e_{1_q}, r_q), our proposed CBR-based KGR model CICBR first learns the representations of entities and relations features through Graph Transformer architecture and then proposes a joint contextual information obtaining algorithm to gather reasoning paths from K_1 similar entities to e_{1_q} and K_2 similar relations to r_q and then traverse those generated reasoning paths in the KG starting from e_{1_q}, which can obtain a set of candidate answer entities. Then, the score of each answer entity candidate is computed as a weighted sum of the reasoning paths, which is weighed with an estimate of its frequency and precision given the query relation.

4 Experiments and Results

In this section, we conduct extensive comparison experiments to verify the performance of the proposed CICBR model.

4.1 Experimental Datasets

To sufficiently verify the effectiveness of the proposed CBR-based KGR model CICBR, we use three different knowledge graph reasoning standard datasets: NELL-995 [13], FB122 [14], and WN18RR [15] in the experiment, which as shown in Table 1. Among them, NELL-995 is a subset of the NELL derived from the 995th iteration of the system. FB122 is a subset of the dataset derived from Freebase, FB15K, which contains 122 relations regarding people, locations, and sports. WN18RR is created from WN18 by removing inverse relation test-leakage.

Table 1. Overview of the experimental datasets.

Dataset	#Ent	#Rel	#Train	#Valid	#Test-I	#Test-II	#Test-ALL
NELL-995	75,492	200	149,678	543	-	-	3,992
FB122	9,738	122	91,638	9,595	5,057	6,186	11,243
WN18RR	40,943	11	86,835	3,034	-	-	3,134

4.2 Baselines and Evaluation Metrics

The experiment represents the rule-based, embedding-based, and case-based reasoning methods as baselines comparing our proposed model CICBR to prove the model's validity thoroughly. The rule-based baselines include KALE [14], ASR [16], and KG_{LR} [17]. The embedding-based baselines contain TransE [18], DistMult [19], ComplEx [20], ConvE [15], RotatE [21], GNTP [22], and MINERVA [23]. The CBR-based baselines consist of CBR [9] and PCBR [10]. Notability, because the research of combining CBR into the field of KGR is still in the development stage, there are few baselines for comparison.

In the link prediction task, two kinds of standard metrics were used to evaluate the experimental performance, including Mean Reciprocal Ranking (MRR) and Hits@K. For each metric, a higher score indicates a better effect. The MRR is calculated as follows:

$$MRR = \frac{1}{|N|}\sum_{i=1}^{|N|}\frac{1}{rank_i} = \frac{1}{|N|}(\frac{1}{rank_1} + \frac{1}{rank_2} + ... + \frac{1}{rank_{|N|}}) \quad (20)$$

N is the set of triples and $rank_i$ is the link prediction ranking (which is a triple ranks by its score in the overall link prediction task results) of the i-th triple.

In addition, $Hits@K(K = 1, 3, 5, 10)$ is described as follows:

$$Hits@K = \frac{1}{|N|}\sum_{i=1}^{|N|}\mathbb{I}(rank_i \leq K) \quad (21)$$

where $\mathbb{I}(\cdot)$ is an indicator function that the value sets to 1 if the condition is true, otherwise it sets to 0.

4.3 Results and Analysis

The link prediction overall results for the three standard datasets are shown in Table 2, Table 3, and Table 4, where "†" indicates the results taken from literature [10]. It can be seen that the proposed CICBR model is explicitly improved compared with CBR-based KGR approaches that the CICBR model can improve the {Hits@3, Hits@5, Hits@10, MRR} average prediction accuracy compared with CBR-based baselines by {17.604%, 14.989%, 12.866%, 20.392%} under the FB122 dataset and improve the {Hits@1, Hits@3, Hits@10, MRR} average prediction accuracy compared with CBR-based baselines under the NELL-995 and WN18RR datasets by {9.091%, 4.777%, 1.149%, 7.664%} and {14.015%, 5.368%, 11.480%, 12.427%} respectively.

Table 2. Overall results of link prediction task on FB122 datasets (Part 1).

	Test-I				Test-II			
	Hits@K				Hits@K			
Models	3	5	10	MRR	3	5	10	MRR
KALE-Pre (Guo et al., 2016)†	0.358	0.419	0.498	0.291	0.829	0.861	0.899	0.713
KALE-Joint (Guo et al., 2016)†	**0.384**	**0.447**	**0.522**	0.325	0.797	0.841	0.896	0.684
ASR-DistMult (Minervini et al., 2017)†	0.363	0.403	0.449	0.330	0.980	0.990	0.992	0.948
ASR-ComplEx (Minervini et al., 2017)†	0.373	0.410	0.459	**0.338**	**0.992**	**0.993**	**0.994**	**0.984**
TransE (Bordes et al., 2013)†	0.360	0.415	0.481	0.296	0.775	0.828	0.884	0.630
DistMult (Yang et al., 2015)†	0.360	0.403	0.453	0.313	0.923	0.938	0.947	0.874
ComplEx (Trouillon et al., 2016)†	0.370	0.413	0.462	0.329	0.914	0.919	0.924	0.887
RotatE (Sun et al., 2019)†	**0.511**	**0.551**	**0.603**	**0.471**	0.868	0.886	0.907	0.846
GNTPs (Minervini et al., 2020)†	0.337	0.369	0.412	0.313	**0.982**	**0.990**	**0.993**	**0.977**
CBR (Das et al., 2020)†	0.400	0.445	0.488	0.359	0.678	0.718	0.759	0.636
PCBR (Das et al., 2020)†	0.490	0.527	0.571	0.457	0.948	0.950	0.953	0.948
CICBR (Ours)	**0.508**	**0.543**	**0.592**	**0.465**	**0.959**	**0.971**	**0.975**	**0.956**

Table 3. Overall results of link prediction task on FB122 datasets (Part 2).

	Test-ALL			
	Hits@K			
Models	3	5	10	MRR
KALE-Pre (Guo et al., 2016)†	0.617	0.662	0.718	0.523
KALE-Joint (Guo et al., 2016)†	0.612	0.664	0.728	0.523
ASR-DistMult (Minervini et al., 2017)†	0.707	0.731	0.752	0.675
ASR-ComplEx (Minervini et al., 2017)†	0.717	0.736	0.757	0.698
KG_{LR} (Garcia-Duran and Niepert, 2017)†	**0.740**	**0.770**	**0.797**	**0.702**
TransE (Bordes et al., 2013)†	0.589	0.642	0.702	0.480
DistMult (Yang et al., 2015)†	0.674	0.702	0.729	0.628
ComplEx (Trouillon et al., 2016)†	0.673	0.695	0.719	0.641
RotatE (Sun et al., 2019)†	0.708	**0.736**	**0.770**	**0.678**
GNTPs (Minervini et al., 2020)†	**0.692**	0.711	0.732	**0.678**
CBR (Das et al., 2020)†	0.570	0.612	0.653	0.527
PCBR (Das et al., 2020)†	0.742	0.760	0.782	0.727
CICBR (Ours)	**0.749**	**0.771**	**0.788**	**0.733**

Table 4. Overall results of link prediction task on NELL-995 and WN18RR datasets.

	NELL-995				WN18RR			
	Hits@K				Hits@K			
Models	1	3	10	MRR	1	3	10	MRR
TransE (Bordes et al., 2013)†	0.53	0.79	**0.87**	0.67	-	-	0.50	0.23
DistMult (Yang et al., 2015)†	0.61	0.73	0.79	0.68	0.39	0.44	0.49	0.43
ComplEx (Trouillon et al., 2016)†	0.61	0.76	0.83	0.69	0.41	0.46	0.51	0.44
ConvE (Dettmers et al., 2018)†	**0.67**	0.81	0.86	**0.75**	0.40	0.44	0.52	0.43
RotatE (Sun et al., 2019)†	0.65	**0.82**	**0.87**	0.74	**0.43**	**0.49**	**0.57**	**0.48**
GNTP (Minervini et al., 2020)†	-	-	-	-	0.41	0.44	0.48	0.43
MINERVA (Das et al., 2017)†	0.66	0.77	0.83	0.72	0.40	0.43	0.49	0.43
CBR (Das et al., 2020)†	0.70	0.83	0.87	0.77	0.38	0.46	0.51	0.43
PCBR (Das et al., 2020)†	0.77	0.85	**0.89**	0.81	0.43	0.49	0.55	0.48
CICBR (Ours)	**0.80**	**0.88**	**0.89**	**0.85**	**0.46**	**0.50**	**0.59**	**0.51**

Compared with CBR-based baselines, the link prediction results of our proposed CICBR model on all experimental datasets are obviously improved. What's more, under the FB122 (Test-ALL), NELL-995 and WN18RR datasets, our CICBR model also can obtain the SOTA results against the rule-based and embedding-based methods. This is because: (i) by utlizing the Graph Transformer architecture for earning and representating entities and relations in the KG, richer feature information can be obtained to enhance inference and the ability of perceptual feature difference between entities and relations; (ii) the joint contextual entities and contextual relations information obtaining algorithm J_{ER}-CIO is proposed to acquire more effective and roundly contextual information for generating candidate paths in the reasoning stage, which can as fully as possible generate and calculate candidate entities so as to improving the accuracy of model's resoning precision.

In conclusion, based on the above comparative experiments, it is indicated that by using the CICBR model proposed in this paper, more plentiful graph features can be acquired, especially the relations information, and more effective contextual information can be obtained to generate more paths conducive to the reasoning for improving model's reasoning accuracy and gain SOTA results on all experimental datasets.

5 Conclusion

In this paper, we proposed a contextual information-augmented probabilistic case-based reasoning model for KGR named CICBR. First, CICBR enhanced the ability to extract and learn graph features, especially the easily neglected relation features, by utilizing the Graph Transformer architecture. Secondly, CICBR proposed a contextual information acquisition algorithm combining contextual entities and contextual relations, which can obtain candidate paths through similarity calculation and processing of the obtained entity and relation features for further reasoning. Third, the probabilistic case-based reasoning method is adopted to reason from the augmented acquired contextual information, which can not only enhance the sensitivity to entity correlation but also provide more attention to the relations between entities. Finally, extensive comparison experiments on three benchmarks demonstrated that our proposed CICBR model can achieve the SOTA reasoning performance against current CBR-based baselines.

Although our work can make improvements to current CBR-based approaches, there still exists the problem of how to obtain more effective contextual information. In the future, we are interested in presenting CBR methods that are more in line with the KGR to improve the reasoning ability of the model.

Acknowledgments

This work is supported by the National Natural Science Foundation of China under Grant No.62162046, the Inner Mongolia Science and Technology Project under Grant No.2021GG0155, the Natural Science Foundation of Major Research Plan of Inner Mongolia under Grant No.2019ZD15, and the Inner Mongolia Natural Science Foundation under Grant No.2019GG372.

References

1. Carlson, A., Betteridge, J., Kisiel, B., Settles, B., Hruschka, E., & Mitchell, T. (2010, July). Toward an architecture for never-ending language learning. In Proceedings of the AAAI conference on artificial intelligence (Vol. 24, No. 1, pp. 1306-1313).
2. Bollacker, K., Evans, C., Paritosh, P., Sturge, T., & Taylor, J. (2008, June). Freebase: a collaboratively created graph database for structuring human knowledge. In Proceedings of the 2008 ACM SIGMOD international conference on Management of data (pp. 1247-1250).

3. Miller, G. A. (1995). WordNet: a lexical database for English. Communications of the ACM, 38(11), 39-41.
4. Behmanesh, S., Talebpour, A., Shamsfard, M., & Jafari, M. M. (2023). Improved Relation Span Detection in Question Answering Systems over Extracted Knowledge Bases. Expert Systems with Applications, 119973.
5. Lin, R., Tang, F., He, C., Wu, Z., Yuan, C., & Tang, Y. (2023). DIRS-KG: a KG-enhanced interactive recommender system based on deep reinforcement learning. World Wide Web, 1-23.
6. Tailhardat, L., Chabot, Y., & Troncy, R. (2023). Designing NORIA: a Knowledge Graph-based Platform for Anomaly Detection and Incident Management in ICT Systems.
7. Dubitzky, W., Büchner, A. G., & Azuaje, F. J. (1999). Viewing knowledge management as a case-based reasoning application. In AAAI Workshop Technical Report WS-99-10 (pp. 23-27).
8. Bartlmae, K., & Riemenschneider, M. (2000, October). Case Based Reasoning for Knowledge Management in KDD Projects. In PAKM.
9. Das, R., Godbole, A., Dhuliawala, S., Zaheer, M., & McCallum, A. (2020). A simple approach to case-based reasoning in knowledge bases. arXiv preprint arXiv:2006.14198.
10. Das, R., Godbole, A., Monath, N., Zaheer, M., & McCallum, A. (2020). Probabilistic case-based reasoning for open-world knowledge graph completion. arXiv preprint arXiv:2010.03548.
11. Dwivedi, V. P., & Bresson, X. (2020). A generalization of transformer networks to graphs. arXiv preprint arXiv:2012.09699.
12. Pujara, J., Augustine, E., & Getoor, L. (2017, September). Sparsity and noise: Where knowledge graph embeddings fall short. In Proceedings of the 2017 conference on empirical methods in natural language processing (pp. 1751-1756).
13. Xiong, W., Hoang, T., & Wang, W. Y. (2017). Deeppath: A reinforcement learning method for knowledge graph reasoning. arXiv preprint arXiv:1707.06690.
14. Guo, S., Wang, Q., Wang, L., Wang, B., & Guo, L. (2016, November). Jointly embedding knowledge graphs and logical rules. In Proceedings of the 2016 conference on empirical methods in natural language processing (pp. 192-202).
15. Dettmers, T., Minervini, P., Stenetorp, P., & Riedel, S. (2018, April). Convolutional 2d knowledge graph embeddings. In Proceedings of the AAAI conference on artificial intelligence (Vol. 32, No. 1).
16. Minervini, P., Demeester, T., Rocktäschel, T., & Riedel, S. (2017). Adversarial sets for regularising neural link predictors. arXiv preprint arXiv:1707.07596.
17. García-Durán, A., & Niepert, M. (2017). Kblrn: End-to-end learning of knowledge base representations with latent, relational, and numerical features. arXiv preprint arXiv:1709.04676.
18. Bordes, A., Usunier, N., Garcia-Duran, A., Weston, J., & Yakhnenko, O. (2013). Translating embeddings for modeling multi-relational data. Advances in neural information processing systems, 26.
19. Yang, B., Yih, W. T., He, X., Gao, J., & Deng, L. (2015). Embedding entities and relations for learning and inference in knowledge bases. In ICLR.
20. Trouillon, T., Welbl, J., Riedel, S., Gaussier, É., & Bouchard, G. (2016, June). Complex embeddings for simple link prediction. In International conference on machine learning (pp. 2071-2080). PMLR.
21. Sun, Z., Deng, Z. H., Nie, J. Y., & Tang, J. (2019). Rotate: Knowledge graph embedding by relational rotation in complex space. arXiv preprint arXiv:1902.10197.

22. Minervini, P., Bošnjak, M., Rocktäschel, T., Riedel, S., & Grefenstette, E. (2020, April). Differentiable reasoning on large knowledge bases and natural language. In Proceedings of the AAAI conference on artificial intelligence (Vol. 34, No. 04, pp. 5182-5190).
23. Das, R., Dhuliawala, S., Zaheer, M., Vilnis, L., Durugkar, I., Krishnamurthy, A., ... & McCallum, A. (2017). Go for a walk and arrive at the answer: Reasoning over paths in knowledge bases using reinforcement learning. arXiv preprint arXiv:1711.05851.