REKGC: Learning Re-coupled Representations for Knowledge Graph Completion

Qingyu Zhang✉[1,2], Min Hu[1,2], Wenlong Fei[1,2], Xiaohua Wang[1,2], and Hongbo Li[1,2]

[1] Hefei University of Technology
[2] Anhui Province Key Laboratory of Affective Computing & Advanced Intelligent Machine
zhangqingyu@mail.hfut.edu.cn

Abstract. Knowledge graph completion aims to fill in missing information in knowledge graphs by addressing challenges such as data sparsity and complex relations. The data sparsity can be addressed by introducing richer contextual semantic features, while relational inference features can help infer complex relations. Current knowledge graph completion methods cannot simultaneously capture these two features. In this paper, we propose REKGC, a novel model for learning **RE**-coupled **RE**presentations for **K**nowledge **G**raph **C**ompletion. REKGC separates the learning of contextual semantic features and relational inference features, and then re-couples them to generate new representations. REKGC includes a contextual semantics module, a relational inference module, and a feature re-coupling module. REKGC ensures effective learning of both advantageous features. Experimental results demonstrate that by capturing and re-coupling two types of features, REKGC can significantly improve the performance of knowledge graph completion.

Keywords: Knowledge Graph · Representation Learning · Knowledge Graph Completion.

1 Introduction

Knowledge graphs (KGs) aim to represent facts and events in the real world via structured triples in the form of (h, r, t), where h, t, and r denote the head entity, the tail entity, and the relation between them. Knowledge graph completion utilizes known entities and relations to fill semantic gaps and enhance completeness. Since the inception of knowledge graph completion, it has faced two main challenges: data sparsity and complex relations. Data sparsity refers to the fact that many entities in KGs are typically contained in only a few triples, especially long-tailed entities. Complex relations include symmetric, inverse, and composite relations, among others. Handling these relations requires not only semantic information but also complex relational pattern modeling. To address these two key challenges, it is essential to introduce richer **contextual semantic features**, while also capturing **relational inference features**, both of which are critical for effective knowledge graph completion.

The approaches for knowledge graph completion can be categorized into neural network approaches and mathematical modeling approaches based on the constraints imposed on knowledge embeddings. Neural network approaches leverage deep network architectures to achieve multi-scale feature interactions, such as ConvE [1] and ComplexGCN [2]. However, due to the lack of explicit mathematical support, these approaches often require explicit encoding of relational features when handling complex relational patterns. In contrast, mathematical modeling approaches use metric learning to capture translational or geometric relations between entities, making them particularly suitable for extracting relational inference features from complex relational patterns, such as RotatE [3] and RotatE3D [4]. However, due to the strict geometric constraints imposed on embeddings, these approaches face challenges in capturing sufficient contextual semantic features from a global perspective.

To improve the expressiveness of knowledge embeddings and the inference capability for complex relations, we introduce **REKGC**, learning **RE**-coupled **RE**presentations for **K**nowledge **G**raph **C**ompletion. REKGC includes Contextual Semantics Module (CSM), Relational Inference Module (RIM), and Feature Re-coupling Module (FRM). CSM takes as input a text sequence in the form of "[entity, relation, neighbor]", learning contextual semantic features from triples. RIM models relations as orthogonal matrices and uses multi-head sub-embeddings to ensure sufficient interaction between entities and relations, thus capturing complex **relational inference features**. FRM aggregates contextual semantic features based on confidence, and transfers them to the interaction process of **relational inference features**. By adjusting the weight of prediction loss, REKGC ensures that both features are effectively re-coupled. The experimental results demonstrate that by re-coupling two learned advantageous features, REKGC significantly improves the representation of knowledge graph embeddings. This highlights the effectiveness of fully exploiting these re-coupled advantageous features in knowledge graph completion.

2 Related Work

2.1 Neural Network Approaches

Convolution-based models such as ConvE [1], JointE [5] and ConvHLE [6] extract multi-level semantics via stacked convolutional layers, fully connected layers, and residual modules. DTAE [7] further integrates attention mechanisms with adaptive dilated convolutions, expanding the receptive field for richer local-global feature extraction. Under the graph neural network framework, CompGCN [8] and SHGNet [9] employ graph convolutions for deeper semantic modeling. Hitter [10] leverages transformers to fully exploit contextual information for improved prediction accuracy.

With their flexible architectures, neural network models excel in feature interaction and semantic transfer, enabling multi-level fusion and effective information sharing.

2.2 Mathematical Modeling Approaches

Mathematical modeling approaches utilize high-dimensional space properties to constrain entity and relation embeddings. Translation-based models, including TransE [11] and TransD [12], embed entities in distinct spaces and measure similarity via Euclidean distance.

To enhance relational inference, factorization methods have gained prominence. For example, Rescal [13] scores entity vectors using relation matrices, while ComplEx [14] introduces a complex vector space for positional modeling, and OTE [15] extends it to higher dimensions for richer relational representations. RotatE3D [4] and QuatE [16] represent entity embeddings as 4D vectors, while CompoundE [17] unifies translation, rotation, and scaling to model relation-dependent transformations.

These mathematical approaches offer strong interpretability in KG tasks, excelling in complex relational modeling but often lacking flexibility in contextual semantics. REKGC integrates strengths from both paradigms, re-coupling their advantages for more meaningful representations.

3 Methodology

This section introduces the proposed REKGC, which consists of three modules: CSM, RIM, and FRM, as shown in Figure 1. CSM learns contextual semantic features, RIM captures relational inference features, and FRM re-couples the advantageous features from both modules.

3.1 Contextual Semantics Module

To extract semantic information from KG triples, we propose the Contextual Semantics Module (CSM). Leveraging advancements in natural language processing [18], transformer-based language models have demonstrated remarkable effectiveness in extracting semantic information from text sequences. Given any entity e, the context is defined as a finite set of relation-entity pairs $< r_i, e_i >$. For each pair e, e_i that satisfies the condition $\exists e_i \in E, \exists r_i \in R$, it follows that:

$$(e, r_i, e_i) \in T \vee (e, r_i^{-1}, e_i) \in T. \tag{1}$$

Building on this, we refined an encoder language model to capture contextual information in KGs, enabling comprehensive learning of the semantic content in text sequences. Specifically, we encode entities, relations, and their context into a long text sequence. To encapsulate the overall semantic information of each triple, we append an additional $[CLS]$ token to the beginning of each sequence. After auto-encoder processing, this $[CLS]$ token serves as the CSM representation in our model.

In addition to leveraging the $[CLS]$ token, we also incorporate neighborhood information for enhanced semantic understanding. Given a missing triple $(h, r, ?)$, we retrieve a one-hop adjacent subgraph $\mathcal{G}$ centered on h. The contextual

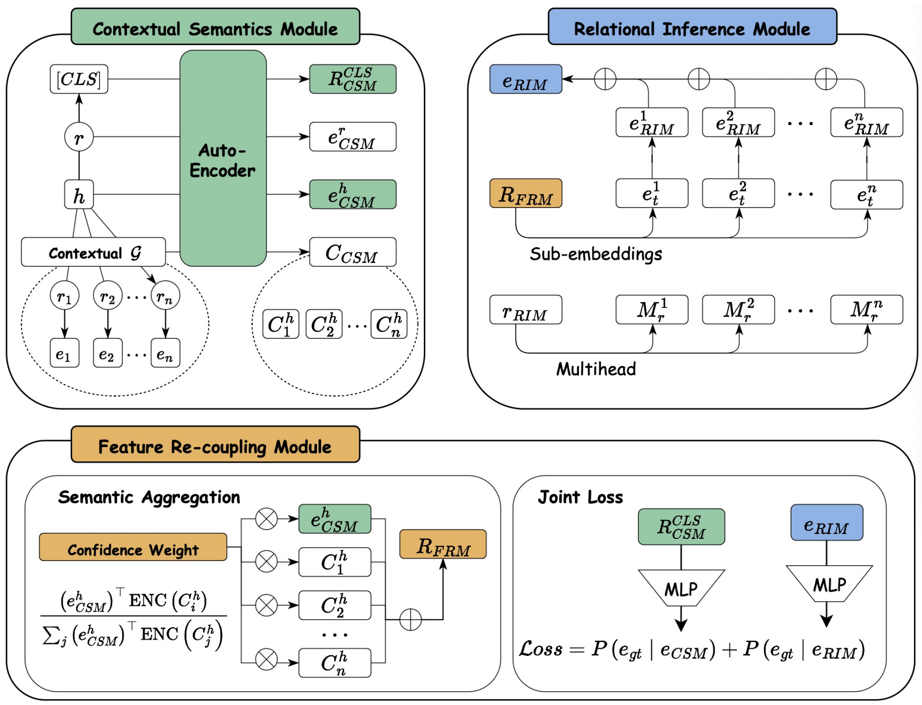

Fig. 1. The overall framework of REKGC.

information of h is then defined by the set of edges and nodes in $\mathcal{G}$, specifically $\mathcal{C} = < r_i, e_i > | \ (h, r_i, e_i) \in \mathcal{G} \cup (e_i, r_i, h) \in \mathcal{G}$, where each pair $< r_i, e_i >$ represents relevant neighboring relations and entities. This additional context helps further refine the representation of incomplete triples. The final triple sequence can be formulated as:

$$\mathcal{S} = [CLS, h, r, \mathcal{G}]. \tag{2}$$

Contextual information illustrates the role of the entity in the predicted triple from multiple perspectives, aiding the model in capturing behavioral patterns more effectively. The triple sequence is then processed by a semantic extraction model, ensuring both contextual semantic aggregation and clearer differentiation of entity- and relation-related features. These contextual semantics, derived through feature disentanglement, constitute the first type of advantageous features in the model.

3.2 Relational Inference Module

Considering that KGs are complex multi-relational graphs with rich patterns [3], we propose a Relational Inference Module (RIM). We extract entity representations from the CSM output and refine their inference features in the RIM.

Following [3], relational inference is modeled using orthogonal matrices, which offer two key advantages over standard representations. First, they inherently encode symmetric, inverse, and composite relations without explicit feature encoding, enabling deeper semantic connections. Second, they preserve the L2 norm during entity transformation, maintaining geometric and semantic stability. To further enhance semantic extraction for complex relations, a multi-head mechanism is introduced. It splits an entity representation $e_{RIM} \in \mathbb{R}^{1\times d}$ into N sub-embeddings, each processed by its respective orthogonal relation matrix. By independently computing each sub-embedding, this design effectively captures complex relational structures across different dimensions. Specifically, given an entity representation $e_{RIM} \in \mathbb{R}^{1\times d}$, it is split into N sub-embeddings according to the multi-head mechanism, where each sub-embedding is represented as:

$$e'_{RIM} = \{e_i \mid e_i \in \mathbb{R}^{1\times d'}, i = 1, 2, \ldots, N\}, \tag{3}$$

where e_i denotes the i-th sub-embedding of the entity representation e_{RIM}. e'_{RIM} denotes the set of sub-embeddings of e_{RIM}. If e_{RIM} has dimension d and there are N heads, then each sub-embedding has dimension $d' = \frac{d}{N}$, using the following equation:

$$r_{RIM} = \{M_r^i \mid M_r \in \mathbb{R}^{d'\times d'}, i = 1, 2, \ldots, N\}, \tag{4}$$

where r_{RIM} denotes the inference representation, and M_r^i denotes the i-th orthogonal matrix of the relational operation r.

Next, we model the interaction between entities and relations as the projection of entities through an orthogonal transformation. Given a matrix M_r with column vectors $v_1, v_2, \ldots, v_{d'}$, the orthogonalized matrix M_r^o is formulated as:

$$u_i = v_i - \sum_{j=1}^{i-1} \mathrm{Proj}_{q_j}(v_i), \tag{5}$$

$$M_r^o = \phi(M_r) = [q_1, q_2, \cdots, q_{d'}], \tag{6}$$

where v_i is the i-th column vector of M_r, and u_i is the i-th orthogonal vector obtained by the orthogonal mapping ϕ. The normalized vector $q_i = \frac{u_i}{|u_i|}$ is used to construct M_r^o. $\mathrm{Proj}_{q_j}(v_i)$ denotes the projection of v_i onto q_j, ensuring orthogonality during transformation.

Ultimately, we can model the interaction between entities and relations using the following equation:

$$e_{RIM} = \phi(r_{RIM})\psi(e'_{RIM}), \tag{7}$$

where r_{RIM} represents the relational inference, e'_{RIM} represents the entity embedding, which will be introduced in the next section, and is referred to as R_{FRM}. Here, ϕ refers to the orthogonal mapping, and ψ refers to the multi-head splitting of the entity embedding.

3.3 Feature Re-coupling Module

We propose the Feature Re-coupling Module (FRM), which integrates advantageous features through feature transfer and feature balancing. To quantify their impact on knowledge inference, we define a confidence score Con_i for each neighbor. The confidence score is the similarity between the neighbor and the entity, normalized by the total similarity between the entity and all neighbors:

$$Con_i = \frac{(e^h_{CSM})^\top \mathrm{ENC}(C^h_i)}{\sum_j (e^h_{CSM})^\top \mathrm{ENC}(C^h_j)}, \tag{8}$$

where e^h_{CSM} and $\mathrm{ENC}(C^h_i)$ are the entity embedding and the encoded representation of C^h_i, respectively. C^h_i represents the i-th neighbor of the entity h. Next, $\mathcal{R}_{FRM}$ is computed as:

$$\mathcal{R}_{FRM} = \sum_i Con_i \times \mathrm{ENC}(C^h_i) + e^h_{CSM}. \tag{9}$$

When generating $\mathcal{R}_{FRM}$, we process each context's contribution differently based on its association with the query entity. The $\mathcal{R}_{FRM}$ serves as an enhanced representation of the query entity, replacing the original entity representation. This can be used in missing entity prediction:

$$e_{RIM} = \phi(r_{RIM})\psi(\mathcal{R}_{FRM}), \tag{10}$$

where e_{RIM} represents the predicted tail entity embedding derived from the context fusion representation $\mathcal{R}^h_{FRM}$ of the head entity in the query $(h, r, ?)$. $\mathcal{R}_{FRM}$ integrates rich context information, significantly improving prediction accuracy and feature generalisation ability.

The FRM uses a joint scoring function to effectively balance the learning of semantic and inference features, avoiding over-reliance on one type of feature. The scoring function computes the $L2$ norm between the predicted entity embedding e and the ground truth entity embedding e_{gt}, dynamically adjusting the emphasis between the two types of features. Specifically, we use a cross-entropy loss function to compute the loss during training:

$$P(e_{gt}|e_{CSM}) = \Phi\left(\sum_{i=1}^{N} \left(|\mathcal{R}^{CLS}_{CSM} - e^i_{cdd}|^2\right)\right), \tag{11}$$

$$P(e_{gt}|e_{RIM}) = \Phi\left(\sum_{i=1}^{N} \left(|e_{RIM} - e^i_{cdd}|^2\right)\right), \tag{12}$$

$$\mathcal{L} = -\sum_{(h,r,t)\in T} \log(P(e_{gt}|e_{CSM})) + \lambda \log(P(e_{gt}|e_{RIM})), \tag{13}$$

where $\mathcal{R}^{CLS}_{CSM}$ represents the global semantic representation extracted from the triple sequence, Φ represents the softmax, and e_{RIM} is the relational inference representation. e_{cdd} represents the candidate entities. We introduce a tuning factor λ to balance semantic and inference features, preventing irrelevant context from affecting model learning.

4 Experiment

In this section, we evaluate the effectiveness of REKGC on two popular benchmarks. First, we introduce the experimental setup, including the datasets and implementation details. Then, we analyze the results of REKGC compared to other methods for the link prediction task. Furthermore, we conduct ablation studies to investigate the contribution of each component in REKGC.

4.1 Experimental Setup

Table 1. Statistics of the experimental datasets.

Datasets	**Entities**	**Relations**	**Train**	**Valid**	**Test**
FB15K-237	14,541	237	272,115	17,535	20,446
WN18RR	40,943	11	86,835	3,034	3,314

Datasets We evaluate the effectiveness of REKGC through link prediction experiments on two benchmark datasets: FB15K-237 [19] and WN18RR [1]. Detailed information about these datasets is presented in Table 1.

Implementation Details The key hyperparameters such as the embedding dimensions for $\mathcal{R}_{FRM}$, $\mathcal{S}$ and r_{RIM} are set to 320. The number of sub-embeddings N ranges from {1, 2, 4, 8, 10}. The context length L_{cxt} ranges from {5, 10, 15, 20, 25} for FB15K-237, and ranges from {0, 2, 4, 6, 8} for WN18RR. Other hyperparameters include the training period of 500 epochs, the label smoothing rate of 0.1, the feature dropout rate of 0.3, and the batch size is set to 512. The learning rate for FB15K-237 is set to 0.001, and for WN18RR is set to 0.00125.

4.2 Experimental Results

As shown in Table 2, we present a comparison of REKGC with 10 other models. Based on the experimental results, we can draw the following main conclusions:

Neural network models excel in capturing **contextual semantic features**. Models such as SHGNet, ConvHLE, and DTAE achieve superior performance on FB15k-237, largely due to dataset characteristics. FB15k-237 associates each entity with an average of 18 triples, providing rich contextual semantics, whereas WN18RR has only 3 triples per entity on average, leading to sparser context. This suggests that neural networks inherently facilitate high-dimensional information interactions, making them particularly effective when contextual information is abundant. Their flexible architectures also enhance feature fusion, allowing for better semantic extraction.

In contrast, mathematical modeling approaches focus on **complex relational patterns**. On WN18RR, methods like RotatE3D and CompoundE significantly outperform neural networks in the Hits@10 metric, highlighting their

Table 2. The performance of link prediction on the FB15K-237 and WN18RR.

Methods	**FB15K-237**				**WN18RR**			
	Hits@1	Hits@3	Hits@10	MRR	Hits@1	Hits@3	Hits@10	MRR
Neural Network Approaches								
ConvE[1]	.237	.356	.501	.325	.400	.440	.520	.430
MRAN[20]	.264	.392	.540	.356	.446	.493	.543	.478
SHGNet[9]	.268	.395	.544	.355	.448	.496	.549	.476
ConvHLE[6]	.268	.395	.544	.360	.437	.482	.532	.469
DTAE[7]	.269	.395	.547	.360	.440	.486	.531	.472
Mathematical Modeling Approaches								
RotatE[3]	.241	.375	.533	.338	.428	.492	.571	.476
RotatE3D[4]	.250	.385	.543	.347	.442	.505	<u>.579</u>	.489
GIE[21]	.271	.401	.552	.362	**.452**	.505	.575	.491
CompoundE[17]	<u>.275</u>	<u>.402</u>	<u>.555</u>	<u>.367</u>	<u>.451</u>	<u>.507</u>	.578	<u>.493</u>
GreenKGC[22]	.265	.369	.507	.345	.367	.430	.491	.411
REKGC	**.282**	**.413**	**.565**	**.377**	.449	**.521**	**.598**	**.501**

strength in relational inference. Although FB15k-237 contains 237 relations while WN18RR has only 11, the latter exhibits hierarchical and compositional structures, requiring deeper relational reasoning. Despite the larger relation set in FB15k-237, its relations are relatively uniform, whereas WN18RR contains structured dependencies, favoring algebraic modeling approaches.

REKGC enhances feature extraction by re-coupling the strengths of neural networks and mathematical modeling. On FB15k-237, REKGC achieves competitive results across multiple evaluation metrics, validating its capability in contextual semantic learning. On WN18RR, REKGC maintains an advantage in relational inference, particularly in Hits@10 and MRR, proving its ability to encode complex relational structures. These results demonstrate REKGC's strong performance and adaptability across datasets with different structural properties, confirming the robustness of its modeling approach.

4.3 Ablation Study

Table 3. Ablation study on FB15K-237 and WN18RR.

Components	**FB15K-237**		**WN18RR**	
	Hits@10	MRR	Hits@10	MRR
CSM	.553	.369	.519	.457
RIM	.534	.357	.581	.490
CSM+RIM	**.565**	**.377**	**.598**	**.501**

In order to analyze the contribution of each component in REKGC, we conducted ablation experiments in the link prediction task, as detailed in Table 3. The experimental results validate the effectiveness of each component. REKGC achieves a significant improvement in capturing contextual semantic features and relational inference features by re-coupling the advantageous features of each module. This suggests that the synergy between modules effectively overcomes the limitations of single features and further improves the embedding quality, and significantly enhancing the knowledge graph completion task.

5 Conclusion

In this paper, we propose REKGC, an innovative knowledge graph embedding method that enhances representation quality by re-coupling two key features: **contextual semantic features** and **relational inference features**. REKGC consists of three modules: the Contextual Semantics Module (CSM), which captures **contextual semantic features** using a knowledge graph-specific autoencoder; the Relational Inference Module (RIM), which leverages orthogonal transformations to learn **relational inference features**; and the Feature Recoupling Module (FRM), which integrates both features to refine representations. Experimental results demonstrate that re-coupling these features significantly improves knowledge graph completion, underscoring their importance in knowledge graph embeddings.

References

1. Dettmers, T., Minervini, P., Stenetorp, P., Riedel, S.: Convolutional 2D Knowledge Graph Embeddings. In: Proceedings of the AAAI Conference on Artificial Intelligence, pp. 1811–1818. AAAI Press (2018)
2. Zeb, A., Saif, S., Chen, J., Haq, A.U., Gong, Z., Zhang, D.: Complex Graph Convolutional Network for Link Prediction in Knowledge Graphs. Expert Systems with Applications **200**, 116796 (2022)
3. Sun, Z., Deng, Z.-H., Nie, J.-Y., Tang, J.: RotatE: Knowledge Graph Embedding by Relational Rotation in Complex Space. In: Proceedings of the International Conference on Learning Representations. OpenReview (2019)
4. Gao, C., Sun, C., Shan, L., Lin, L., Wang, M.: Rotate3D: Representing Relations as Rotations in Three-Dimensional Space for Knowledge Graph Embedding. In: Proceedings of the ACM International Conference on Information and Knowledge Management (CIKM), pp. 385–394. ACM (2020)
5. Zhou, Z., Wang, C., Feng, Y., Chen, D.: JointE: Jointly Utilizing 1D and 2D Convolution for Knowledge Graph Embedding. Knowledge-Based Systems **240**, 108100 (2022)
6. Wang, J., Zhang, Q., Shi, F., Li, D., Cai, Y., Wang, J., Li, B., Wang, X., Zhang, Z., Zheng, C.: Knowledge Graph Embedding Model with Attention-Based High-Low Level Features Interaction Convolutional Network. Information Processing & Management **60**(4), 103350 (2023)

7. Deng, W., Zhang, Y., Yu, H., Li, H.: Knowledge graph embedding based on dynamic adaptive atrous convolution and attention mechanism for link prediction. Information Processing & Management **61**(3), 103642 (2024)
8. Vashishth, S., Sanyal, S., Nitin, V., Talukdar, P.P.: Composition-based Multi-Relational Graph Convolutional Networks. In: Proceedings of the International Conference on Learning Representations. OpenReview (2020)
9. Li, Z., Zhang, Q., Zhu, F., Li, D., Zheng, C., Zhang, Y.: Knowledge Graph Representation Learning with Simplifying Hierarchical Feature Propagation. Information Processing & Management **60**(4), 103348 (2023)
10. Chen, S., Liu, X., Gao, J., Jiao, J., Zhang, R., Ji, Y.: HittER: Hierarchical Transformers for Knowledge Graph Embeddings. In: Proceedings of EMNLP, pp. 10395–10407. Association for Computational Linguistics (2021)
11. Bordes, A., Usunier, N., Garcia-Duran, A., Weston, J., Yakhnenko, O.: Translating embeddings for modeling multi-relational data. Advances in Neural Information Processing Systems **26**, 2787–2795 (2013)
12. Ji, G., He, S., Xu, L., Liu, K., Zhao, J.: Knowledge Graph Embedding via Dynamic Mapping Matrix. In: Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL), pp. 687–696. Association for Computational Linguistics (2015)
13. Nickel, M., Tresp, V., Kriegel, H.-P.: A Three-Way Model for Collective Learning on Multi-Relational Data. In: Proceedings of the International Conference on Machine Learning (ICML), pp. 809–816. PMLR (2011)
14. Trouillon, T., Welbl, J., Riedel, S., Gaussier, É., Bouchard, G.: Complex Embeddings for Simple Link Prediction. In: Proceedings of the International Conference on Machine Learning (ICML), pp. 2071–2080. PMLR (2016)
15. Tang, Y., Huang, J., Wang, G., He, X., Zhou, B.: Orthogonal Relation Transforms with Graph Context Modeling for Knowledge Graph Embedding. In: Proceedings of the Annual Meeting of the Association for Computational Linguistics, pp. 2713–2722 (2020)
16. Zhang, S., Tay, Y., Yao, L., Liu, Q.: Quaternion Knowledge Graph Embeddings. In: Advances in Neural Information Processing Systems (NeurIPS), pp. 2731–2741. Curran Associates (2019)
17. Ge, X., Wang, Y., Wang, B., Kuo, C.-C.J.: CompoundE: Knowledge graph embedding with translation, rotation and scaling compound operations. CoRR **abs/2207.05324** (2022)
18. Brown, T.B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., et al.: Language Models are Few-Shot Learners. In: NeurIPS 2020. Advances in Neural Information Processing Systems **33**, 1877–1901 (2020)
19. Toutanova, K., Chen, D.: Observed versus Latent Features for Knowledge Base and Text Inference. In: Proceedings of the 3rd Workshop on Continuous Vector Space Models and Their Compositionality, pp. 57–66 (2015)
20. Xue, Z., Zhang, Z., Liu, H., Yang, S., Han, S.: Learning Knowledge Graph Embedding with Multi-Granularity Relational Augmentation Network. Expert Systems with Applications **233**, 120953 (2023)
21. Cao, Z., Xu, Q., Yang, Z., Cao, X., Huang, Q.: Geometry interaction knowledge graph embeddings. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 36, no. 5, pp. 5521–5529 (2022)
22. Wang, Y., Ge, X., Wang, B., Kuo, C.-C. J.: GreenKGC: A Lightweight Knowledge Graph Completion Method. In: Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics vol. 1, pp. 10596–10613. (2023)