The Concept Information of Graph Granule with Application to Embedding Learning of Knowledge Graph

Jiaojiao Niu (✉ [njiao_92@163.com](mailto:njjiao_92@163.com))

North China Electric Power University

Degang Chen

North China Electric Power University - Beijing Campus: North China Electric Power University

Yinglong Ma

North China Electric Power University - Beijing Campus: North China Electric Power University

Jinhai Li

Kunming University of Science and Technology - Lianhua Campus: Kunming University of Science and Technology

Research Article

Keywords: Knowledge graph, Knowledge graph embedding, Formal concept analysis, Graph granule, Link prediction

Posted Date: August 9th, 2021

DOI: <https://doi.org/10.21203/rs.3.rs-453545/v1>

License: This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

The concept information of graph granule with application to embedding learning of knowledge graph

Jiaojiao Niu · Degang Chen* · Yinglong Ma · Jinhai Li

Received: date / Accepted: date

Abstract Knowledge graph embedding (KGE) are routinely used to represent entities and their relations in knowledge bases with a quantitative measure, and the triples usually play the role of basic units in KGE learning. Considering that triples are sometimes far from adequate and knowledge graph itself contains a lot of information, this paper employs FCA-based technology to mine the deterministic knowledge from knowledge graph, that is, the formal concept, and attempts to establish the relationship between knowledge graph (KG) and formal concept analysis (FCA). Specifically, each set of triples sharing the same head entity are grouped as a graph granule and the concepts of each graph granule are mined. By further exploration, the maximal concepts are integrated into embedding learning to develop a novel KGE model named TransGr for knowledge graph completion. This model learns a matrix for each maximal concept in graph granule as well

as a vector for each entity and relation. The performed experiments on link prediction and triple classification tasks demonstrate that the proposed TransGr model is effective on the datasets with relatively complete graph granules.

Keywords Knowledge graph · Knowledge graph embedding · Formal concept analysis · Graph granule · Link prediction

1 Introduction

Knowledge graph (KG) is a multi-relational graph database, using nodes to represent entities, and directed edges to depict the relations between the two connected entities. It has become an important infrastructure for many research fields and various KGs Auer et al. (2007); Bollacker et al. (2008); Miller (1995); Suchanek et al. (2007) have been established for different tasks. However, the symbolism of knowledge graph makes it face the challenge of efficiency in specific application. At this time, many knowledge reasoning methods came into being.

Knowledge graph embedding (KGE), which aims at learning vector representation for the symbolic entities and relations, is one of the most explored knowledge reasoning methods. The existing KGE models can be roughly divided into two categories. One category includes translational distance models Bordes et al. (2013); Feng et al. (2015); He et al. (2015); Ji et al. (2015); Lin et al. (2015b); Wang et al. (2014), such as TransE Bordes et al. (2013), which measure the plausibility of the facts as the distance between two entities. Simplicity is the biggest advantage of such models. However, because of the complexity of the relations in the knowledge graph, they sometimes may lack efficiency. Therefore, some researchers try to embed addition

Jiaojiao Niu
School of Control and Computer Engineering, North China
Electric Power University, Beijing 102206, China
E-mail: njjiao_92@163.com

Degang Chen
School of Mathematics and Physics, North China Electric
Power University, Beijing 102206, China
E-mail: wscz973@163.com

Yinglong Ma
School of Control and Computer Engineering, North China
Electric Power University, Beijing 102206, China
E-mail: yinglongma@ncepu.edu.cn

Jinhai Li
Data Science Research Center, Kunming University of Science
and Technology, Kunming 650500, Yunnan, China
E-mail: jhlixjtu@163.com

information to improve the robustness of the KGE models and the semantic matching models Lacroix et al. (2018); Lin et al. (2015a); Nickel et al. (2011, 2016); Trouillon et al. (2016); Xiao et al. (2016); Yang et al. (2015); Zhang et al. (2019) which taking certain semantic information into account is one of them. For instance, RESCAL Nickel et al. (2011) associates each entity with a vector and each relation with a matrix to capture their latent semantics while another work Lin et al. (2015a) considers relational path as semantic compositions of their relation embedding to benefit for embedded learning. These methods regard triple or relational path as the basic unit of embedding, and thus lack the mechanism of using additional theories to mine structured information from knowledge graph, which may affect the practical applications of knowledge graph with obvious structured information.

As a mathematical tool for data analysis and processing, formal concept analysis (FCA) Wille (1982) has been widely used to deal with several variants of the knowledge graph, such as graph and relational data. In an existing report Wille and Fachbereich (1997), conceptual graphs and FCA have been connected through the conceptual structure to obtain a formalization of elementary logic. Relational concept analysis (RCA) Rouane-Hacene et al. (2013) extended the concept to multi-relational datasets. Concretely, it constructed a set of concept lattices, one per object sort, through an iterative analysis process that is bound towards a fixed-point. Graph-FCA Ferré (2015); Ferré and Cellier (2019) is a direct application of FCA in the knowledge graph. It introduced projected graph patterns (PGP) and the object relations to define graph concept. Based on the theory introduced by Graph-FCA, Ferré (2017) proposed a symbolic form of k-NN (k Nearest Neighbors) named C-NN (Concepts of Nearest Neighbors) where numerical distances were replaced by graph patterns that provide an intelligible representation of how similar two entities are. Later, Ferré (2019, 2020) applied C-NN to knowledge graph for link prediction and achieved state-of-the-art results. In addition to link prediction, some scholars applied FCA for information retrieval Balasubramaniam (2015); Cigarrán et al. (2005); Fkih and Omri (2016), ontology learning Jabbari and Stoffel (2018, 2019) and semantic annotation González and Hogan (2018), from text data or semantic network. For instance, Balasubramaniam (2015) proposed a hybrid FOGA framework using FCA for information retrieval based on clustering while Jabbari and Stoffel (2019) introduced a pipeline that incorporates Natural Language Processing (NLP), FCA and Ontology Engineering techniques to build an ontology from textual data. Although these FCA-based studies involve the

concept information of the knowledge graph, there is no discussion on how to use this information to develop specific knowledge graph embedding models.

In FCA, concept is used to represent the generalization and specialization relationship between entities, usually formed as (X, B) , indicating that objects in X have at least the attributes in B . Therefore, concepts usually imply the deterministic information and contain rich semantic information. This fact motive our idea to establish the relationship between KG and FCA to develop a novel KEG model named TranGr by taking the concept information into account. Note that only some relevant information rather than the whole knowledge graph is needed to infer an entity, we first group each set of triples with the same head entity as a graph granule. Then, concept mining is carried out in graph granule instead of the whole knowledge graph to reduce the search space and improve the efficiency of concept computing. By further exploration, a novel KGE model named TransGr is developed to learn a matrix for each concept as well as a vector for each entity and relation by capturing the interaction between entities through concepts obtained from graph granules. Considering that there are a large number of concepts in the whole knowledge graph and the maximal concepts contains more tail entities, this paper uses the maximal concepts for embedding learning.

The paper is organized as follows. Section 2 reviews some related pieces of knowledge with KG and FCA. The TransGr model is presented in Section 3 and the numerical experiments conducted on two standard datasets Wordnet and Freebase follow immediately in Section 4. A summary and future work are presented in Section 5.

2 Preliminaries

In this section, we first recall some related definitions about knowledge graph and formal concept analysis, and then introduce the notion of graph granule.

2.1 Knowledge graph and knowledge graph embedding

Knowledge graph was formally proposed by Google in 2012, aiming at improving the ability of search engines. It encodes structured information of entities and their rich relations, and triples formed as (head entity, relation, tail entity), also called facts, usually play the role of basic units in a knowledge graph. In the literatures, it can be defined as follows:

Definition 2.1 Cai et al. (2018) A knowledge graph $G = (V, E)$ is a directed graph whose nodes are enti-

ties and edges are subject-property-object triple facts. Each edge of the form (head entity, relation, tail entity) (denoted as $\langle h, r, t \rangle$) indicates a relation of r from entity h to entity t .

Although the triples are efficient in representing structured data, the potential symbolic property often makes them difficult to application. Knowledge graph embedding is a promising approach to tackle this issue.

Definition 2.2 Given a knowledge graph $G = (V, E)$. Knowledge graph embedding is to convert the entities and relations in G into a continuous low dimensional vector or matrix space in which the plausibility of a triple is preserved as much as possible.

The plausibility of the triple (h, r, t) is typically reflected in the definition of score function $f_r(h, t)$. For instance, if triple (h, r, t) holds, TransE Bordes et al. (2013) requires the embedded vectors to satisfy $h + r = t$. Then, $f_r(h, t) = \|h + r - t\|_{1/2}^2$ is employed as a score function to make (h, r, t) have a lower score than an incorrect triple in the vector space.

In order to obtain the vectors of entities and relations, a corrupted triple is generated with either the head entity or tail entity replaced by a random entity for each triple in knowledge graph to construct a margin-based score function:

$$L = \sum_{(h,r,t) \in S} \sum_{(h',r,t') \in S'} \max(0, f_r(h, t) + \gamma - f_r(h', t'))$$

where $\gamma > 0$ is a margin hyperparameter and S, S' respectively denote a triple set in the knowledge graph and corresponding corrupted triple set.

2.2 The graph granule and its concept lattice

Generally speaking, the triples in a knowledge graph are extremely large. To infer an entity accurately, we only need some relevant information rather than the whole knowledge graph. Thus, in this section, we group each set of triples with the same head entity as a graph granule for deterministic information mining.

Definition 2.3 Given a knowledge graph $G = (V, E)$. A graph granule $G_h = (h, T, R)$ is a triple, in which $h \in V$ is the head entity, $T \in V$ is a set of tail entities related to h , and R from the relation part of E , point to all the relations from head entity h to tail entity set T .

It is obvious that a graph granule is the subgraph of a knowledge graph with one head entity and a certain number of tail entities related to it. When a graph granule contains only one tail entity, it will degenerate into a triple. Since the head entity in a graph granule is unique, it will be used to represent a graph granule in this paper when there is no confusion.

Table 1 The formal context (U, R, I)

$O \backslash R$	75	30	61	167	37	9
(26,467)	1	0	0	0	0	0
(26,651)	0	1	0	0	0	0
(26,10332)	0	0	1	0	0	1
(26,4138)	0	0	0	1	0	0
(26,6309)	0	0	0	1	0	1
(26,65)	0	0	0	0	1	0
(26,230)	0	1	0	0	0	0
(26,6864)	0	0	0	1	0	0

Example 1: Fig.1 presents a graph granule chosen from the dataset FB15K used in Section 4, in which the middle ball represents the head entity, and the surrounding balls represent the tail entities. The strings below the tail entities represent the relation between all entities from the head entity to the tail. Each edge denotes a triple: such as $(/m/0q9kd, /people/person /profession, /m/02hrhlq)$, which means there is a relation “/people/person/profession” from entity “/m/0q9kd” to entity “/m/02hrhlq”.

A graph granule contains a lot of information, and some of which may be uncertain. The FCA-based method can be used to mine the deterministic information in graph granule. Therefore, we transform the graph granule into formal context.

Definition 2.4 Let $G_h = (h, T, R)$ be a graph granule, the formal context of G_h can be defined as $F_h = (O, R, I)$, where $O = \{(h, t_1), (h, t_2), \dots, (h, t_{|T|})\}$ is a nonempty, finite set of entity pairs, $R = \{r_1, r_2, \dots, r_m\}$ is a set of relations between the entity pairs, while I is the binary relation on $O \times R$. If triple (h, r, t) be a fact, then the value of I is 1, otherwise 0.

Example 2: Table 1 presents the formal context (O, R, I) of the graph granule shown in Fig. 1, where $O = \{(26, 467), (26, 651), (26, 10332), (26, 4138), (26, 6309), (26, 65), (26, 230), (26, 6864)\}$ and $R = \{75, 30, 61, 167, 37, 9\}$. All these numbers in the formal context are the entity (relation) ids in the dataset.

Definition 2.5 Let the triple (O, R, I) be a formal context, $P(O)$ be the power set of O , $P(R)$ be the power set of R , $f : P(O) \rightarrow P(R)$, $g : P(R) \rightarrow P(O)$. For $X \subseteq P(O)$, $L \subseteq P(R)$, we define

$$f(X) = \{r \in R | \forall (h, t) \in X, I(h, r, t) = 1\};$$

$$g(L) = \{(h, t) \in X | \forall r \in L, I(h, r, t) = 1\}.$$

Similar to the classical concept lattice theory, we can define concept in graph granule based on the operations defined in Definition 2.5.

Definition 2.6 Let the triple (O, R, I) be a formal context. A tuple (X, L) , with $X \in P(O)$ and $L \in P(R)$, is called the concept of the formal context (O, R, I) if $f(X) = L$ and $g(L) = X$. The set of entity pair set

Fig. 1 A graph granule in dataset FB15K

Fig. 2 The concept lattice $L(U, R, I)$

X and relation set L are, respectively, labeled as the extension and the intension of the concept (X, L) .

For two concepts (X_1, B_1) and (X_2, B_2) , the order relation between them can be defined as follows:

$$(X_1, B_1) \subseteq (X_2, B_2) \Leftrightarrow X_1 \subseteq X_2 (\Leftrightarrow B_2 \subseteq B_1).$$

The infimum and supremum between them can be defined as follows:

$$\begin{aligned} (X_1, B_1) \wedge (X_2, B_2) &= (X_1 \cap X_2, f(g(B_1 \cup B_2))); \\ (X_1, B_1) \vee (X_2, B_2) &= (g(f(X_1 \cup X_2)), B_1 \cap B_2). \end{aligned}$$

If we use $L(O, R, I)$ to denote the poset of all the concepts in the formal context (O, R, I) , it can be verified that $L(O, R, I)$ constitutes a complete lattice, which is called the concept lattice of the graph granule.

In what follows, (h, T^*) with $T^* = \{t_1, t_2, \dots, t_n\}$ is used to represent the set of entity pairs $\{(h, t_1), \dots, (h, t_n)\}$ for convenience.

Example 3: According to the Definition 2.6, the formal context (O, R, I) in Table 1 has a total of nine concepts: $(\{(26, 467)\}, \{75\})$, $(\{(26, 65)\}, \{37\})$, $(\{(26, 10332)\}, \{61, 9\})$, $(\{(26, 6309)\}, \{167, 9\})$, $(\{(26, \{10332, 6309\})\}, \{9\})$, $(\{(26, \{651, 230\})\}, \{30\})$, $(\{(26, \{4138, 6309, 6864\})\}, \{167\})$, $\emptyset, U$. Fig.2 shows its concept lattice in which the head entity “26” is omitted.

Obviously, there are many concepts in a graph granule and it is unnecessary to use all the concepts for embedded learning. According to the definitions of supremum and infimum between concepts, it is easy to find that the maximal concepts connect more tail entities, so they contain more extensive information. Therefore, in this paper, we only focus on the maximum concepts and develop Algorithm 1 to compute the maximal concepts from each graph granule for KGE learning.

Algorithm 1 The computing algorithm for finding the maximal concepts

Input: Knowledge graph K

Output: The maximal concept set C_{max}

```

1: Set  $C_{max} = \emptyset$ 
2: Get the graph granule set  $G_K$ 
3: For each  $G_h$  in  $G_K$ 
4:   For each  $r$  in  $G_h$ 
5:     Get the frequency  $fre(r)$  of  $r$ 
6:   End
7:   Arrange  $fre$  in descending order
8:   For each  $fre(r)$  in  $fre$ 
9:     If there is no  $(A, B) \in C_{max}$  satisfies  $g(r) \subseteq A$ 
10:       $C_{max} = C_{max} \leftarrow (g(r), f(g(r)))$ 
11:     End
12:   End
13: End
14: Return  $C_{max}$ 

```

It is easy to verify that the time complexity of Algorithm 1 in worst case is $O(2|G_K||R|)$.

3 Knowledge graph embedding learning via maximum concept

From the results obtained in the previous sections, we know that a graph granule shares a mathematical structure as a concept lattice. Since the concepts in the concept lattice describe the generalization and specialization relationship between entities, they can be used to

Fig. 3 The diagrammatic sketch of TransGr

capture the connection between entities in knowledge graph. Thus, in this section, we develop a novel KGE model TransGr by considering the concept information in graph granule. Considering that there are a large number of concepts in the whole graph granule and the maximal concepts contains more tail entities, TransGr uses the maximal concepts for embedding learning.

If we use bold e to represent the vector of e , then, given a graph granule G_h , TransGr learns a series of matrices $C_i (i = 1, \dots, n)$ for its maximal concepts (X_i, B_i) and represents these information with $C_h = \sum_{i=1}^n C_i$. Then, if there is a relation r between entities h and t , (h, t) must be part of some concepts. Since these concepts reflect the relationship between two entities, TransGr argues that they can project some information of one entity from another entity. Thus, it forces

$$\begin{aligned} C_h * t &= h_c \\ C_t * h &= t_c \end{aligned}$$

in the vector space hoping that the concepts cover an entity can gather the related tail entities together and employs

$$L_r(h, t) = \|C_h * t + r - C_t * h\|_{1/2}^2$$

as the score function of triple (h, r, t) . In addition, in order to capture the connection between e_c and e , TransGr defines the following two functions to make h_c, t_c not too far away from h, t :

$$\begin{aligned} L_h(h_c, h) &= \|h_c - h\|_{1/2}^2, \\ L_t(t_c, t) &= \|t_c - t\|_{1/2}^2. \end{aligned}$$

Fig.3 displays a simple diagrammatic sketch of TransGr. As shown in Fig.3, the entities h and t are firstly projected onto the relation space by the corresponding concept planes. Then, it requires $h_c + r = t_c$ in the relation space. Finally, it defines the following score function:

$$f_r(h, t) = L_r(h, t) + L_h(h_c, h) + L_t(t_c, t).$$

The score function $f_r(h, t)$ is expected to be lower for a correct triple and higher for an incorrect triple. For better convergence, the following restrictions are applied on entities and relations:

$$\begin{aligned} \|e\|_2 &= 1, \forall e \in V \\ \|e_c\|_2 &= 1, \\ \|r\|_2 &= 1, \forall r \in R. \end{aligned}$$

In order to obtain the embeddings of entities and relations, we only need to optimize the following margin-based score function:

$$L = \sum_{(h,r,t) \in S} \sum_{(h',r,t') \in S'} \max(0, f_r(h, t) + \gamma - f_r(h', t'))$$

where $\gamma > 0$ is a margin hyperparameter and S, S' respectively denote a triple set in the knowledge graph and corresponding corrupted triple set. To generate reasonable corrupted triples, we consider to randomly corrupt each triple by replacing either the head or the tail entity with every entity that do not violate the semantics of the corresponding relation-types Krompa et al. (2015) in a knowledge graph.

For solving the optimization problem, the mini-batched stochastic gradient descent is used to optimize the objective function, see details in Algorithm 2. During training, the data will first be divided into minibatches. Then, for each iteration, a minibatch is used to calculate the loss function and update the vectors.

Algorithm 2 The learning algorithm of TransGr

Input: Training set S , entities and relations set E and R , margin γ , embedding dimension k

Output: The embeddings of entities and relations and matrices of the maximal concepts

- 1: Get the maximal concept set C_{max} by Algorithm 1
 - 2: **Initialize**
 - 3: Randomly initialize e for each $e \in E$
 - 4: Randomly initialize r for each $r \in R$
 - 5: Randomly initialize C_e for each maximal concept
 - 6: Calculate e_c and normalize e_c, e and r
 - 7: **Loop**
 - 8: $S_{batch} \leftarrow \text{sample}(S, b)$ //sample a minibatch of size b
 - 9: $T_{batch} \leftarrow \emptyset$ //initialize the set of pairs of triples
 - 10: **For** $(h, l, t) \in S_{batch}$
 - 11: $(h', r, t') \leftarrow \text{sample a corrupted triple}$
 - 12: $T_{batch} \leftarrow T_{batch} \cup \{(h, r, t), (h', r, t')\}$
 - 13: **End**
 - 14: **Update embeddings**
w.r.t. $\sum_{(h,r,t) \in S} \sum_{(h',r,t') \in S'} \nabla [f_r(h, t) + \gamma - f_r(h', t')]_+$
 - 15: **End**
-

It is easy to verify that TransGr totally has $k^2 n_e + k n_e + k n_r$ parameters, and its time complexity is $O(k^2)$.

In the literatures, there also exist models that consider to learn vector or matrix representation for a non-single triple, such as CTransR Lin et al. (2015b) learns a discrete relation vector for each entity pair cluster and path-based model Lin et al. (2015a) learns a vector or matrix for every relational path. Still, neither entity pair cluster nor relational path contains concept

information of knowledge graph. Although Guan et al. (2019) proposed a common-sense concept based KGE model, they need to carry out multiple training processes due to the dependence on additional language database. Unlike these, TranGr groups multiple triples with the same head entity into a graph granule to extract the maximum concepts, and uses its own concept information to learn vector of knowledge graph.

4 Experiments

In this section, we conduct some numerical experiments on two popular tasks involving knowledge graph completion to evaluate the proposed TransGr model, i.e., link prediction and triple classification. For comparison, we chose two typical knowledge graphs: Wordnet and Freebase. Wordnet is a large lexical database of English, in which each entity represents a synset consisting of several words, and a word can also belong to different synsets, while Freebase is a sharing website that stores general facts about the world. Six sub datasets were generated from those two knowledge graphs. Table 2 lists the details of these datasets. Among these six datasets used, WN18, WN18RR and WN11 are subsets of Wordnet while FB15K, FB15K-237 and FB13 are subsets of Freebase. Besides, there have no inverse relations in datasets WN18RR and FB15K-237.

Table 2 The statistics of the used datasets

Dataset	#Relation	#Entity	#Train	#Valid	#Test
WN18	18	40943	141442	5000	5000
WN18RR	11	40559	86835	3034	3134
FB15K	1345	14951	483142	50000	59071
FB15K-237	237	15541	272115	17535	20466
WN11	11	38696	112581	2609	10544
FB13	13	75043	316212	5908	23733

Link prediction aims to predict the unknown entity or relation in a triple, such as predicting the missing head entity Beijing in triple (*, capital of, China) or predicting the missing relation “Capital of” in triple (Beijing, *, China). This task is conducted on four datasets: WN18, WN18RR, FB15K and FB15K-237. For comparison, we chose several non-FCA based models, that is RESCAL Nickel et al. (2011), TransE Bordes et al. (2013), TransH Wang et al. (2014), TransR Lin et al. (2015b), DistMult Yang et al. (2015), ComplEx Trouillon et al. (2016), KG2E He et al. (2015), TransG Xiao et al. (2016), ComplEx-N3 Lacroix et al. (2018) and a FCA-based model named C-NN Ferré (2020) as our baselines.

To generate reasonable corrupted triples, we randomly corrupted each triple by replacing either the head or the tail entity with every entity that do not violate the semantics of the corresponding relation-types Krompa et al. (2015) in a knowledge graph. Then the score of each corrupted triple was calculated by the score function $f_r(h, t)$ introduced in Section 3 and ranked these entities in descending order of possibility. Considering that the correct entity should rank before the incorrect ones, we expect a high value for two indicators: the average of the reciprocal ranks (MRR) of correct entities and the proportion of testing triple whose rank is not larger than N (Hits@N). Since false-negative triples may be generated in the process of generating negative triples that can influence the experimental results, the corrupted triples that exist in the training, validation or test datasets were filtered out for a more reasonable result. And this is called the “Filter” setting while the original one is called the “Raw” setting.

For comparison, the experimental results of several baselines were directly reproduced from the existing literatures, in which the MRR are from Kazemi and Poole (2018) and Tan et al. (2018), and the results of TransG come from Jia et al. (2018). We ran TransGr with learning rate $\alpha = 0.3$, $k = 50$, $\gamma = 10$, $N = 100$ on WN18; $\gamma = 0.7$, $k = 50$, $\gamma = 7$, $N = 100$ on FB15K.

Table 3 lists the comparison of the MRR and HITS@10 on WN18 and FB15K, in which “-” implies that we didn’t find the corresponding results from the existing literatures. It is easily identifiable from Table 4 that TransGr ranks first with 82.2%, 28.9% and 53.9% in HITS@10 on WN18 and in MRR and HITS@10 on FB15K with the “Raw” setting. It also outperforms the baselines significantly when compared to translation-based models (TransE, TransR, TransH, and KG2E), demonstrating that capturing concept information can benefit for embedding learning. When comparing TransGr to the remaining baselines, it produces better results than DistMult and ComplEx in several aspects, such as MRR and HITS@10 on FB15K with “Filter” setting, but worse results than ComplEx-N3 and C-NN in all respects. Since FB15K is a relatively complex knowledge graph with 1345 relations, this may indicate that the complex model can capture complex datasets better.

To verify the impact of the inverse relation on the proposed model, we also compare TransGr with the four most effective models on two more challenging datasets: WN18RR and FB15K-237 with inverse relations removed. We ran TranGr with learning rate $\alpha = 0.1$, $k = 50$, $\gamma = 27$, $N = 100$ on the former dataset and $\alpha = 0.1$, $k = 50$, $\gamma = 25$, $N = 30$ on the latter dataset. The evaluation results are listed in Table 4, in

Table 3 Evaluation results of link prediction on WN18 and FB15K

Dataset	WN18				FB15K			
	MRR		HITS@10(%)		MRR		HITS@10(%)	
	Raw	Filter	Raw	Filter	Raw	Filter	Raw	Filter
RESCAL	60.3	80.0	37.2	52.8	18.9	35.4	28.4	44.1
TransE	33.5	45.4	75.4	89.2	22.1	38.0	34.9	47.1
TransH	26.8	36.1	75.4	86.7	18.3	28.3	42.5	58.5
TransR	42.7	60.5	78.3	91.7	19.8	34.6	43.8	65.5
DistMult	53.2	82.2	-	93.6	24.2	65.4	-	82.4
ComplEx	58.7	94.1	-	94.9	23.2	69.2	-	84.0
KG2E	-	-	74.8	87.8	-	-	47.5	71.5
TransG	-	-	81.4	93.3	-	-	53.1	79.8
ComplEx-N3	-	95.0	-	96.0	-	86.0	-	91.0
C-NN	-	96.9	-	97.2	-	84.9	-	89.0
TransGr	57.6	87.4	82.2	94.5	28.9	71.3	53.9	85.6

Table 5 Evaluation results on FB15K by mapping properties of relations

Task	Predicting Head				Predicting Tail			
	Hits@10				Hits@10			
	1-1	1-n	n-1	n-n	1-1	1-n	n-1	n-n
TransE	43.7	65.7	18.2	47.2	43.7	19.7	66.7	50.0
TransH	66.8	87.6	28.7	64.5	65.5	39.8	83.3	67.2
TransR	78.8	89.2	34.1	9.2	79.2	37.4	90.4	72.1
TransD	86.1	95.5	39.8	78.5	85.4	50.6	94.4	81.2
KG2E	92.3	93.7	66.0	69.6	92.6	67.9	94.4	73.4
TransG	93.0	96.0	62.5	86.8	92.8	68.1	94.5	88.8
TransGr	93.2	96.0	65.6	85.2	93.3	72.3	94.6	88.2

Table 4 Evaluation results of link prediction on WN18RR and FB15K-237

Dataset	WN18RR		FB15K-237	
	HITS@3	HITS@10	HITS@3	HITS@10
DistMult	44.0	49.0	26.3	41.9
ComplEx	46.0	51.0	27.5	42.8
ComplEx-N3	-	57.0	-	56.0
C-NN	-	51.9	-	44.6
TransGr	38.9	44.5	32.4	46.8

which the results of DistMult and ComplEx are from Dettmers et al. (2018) while the other two come from Ferré (2020). We can conclude that: (1) TransGr got the worst results on the two selected datasets except for HITS@3 on FB15K-237; (2) Compared with Table 3, the performance of TransGr is greatly reduced on these two datasets; (3) The inverse relation has smaller influence on FB15K than WN18. These facts all indicate that TransGr is sensitive to the relation types and has more stable performance on more complex datasets.

For further evaluation, we separately show evaluation results by mapping the properties of relations on FB15K in Table 5, in which the results of TransD Ji et al. (2015) are from the original paper and the others are from TransG Xiao et al. (2016). It demonstrates that TransGr, which takes concept into consideration, outperforms all the chosen baselines except for TransG and KG2E. Although TransGr shows weaker perfor-

mance in some aspects than TransG when facing 1-n and n-n relations, it recovers up to 3.1% when predicting head entity of n-1 relations and 4.2% when predicting tail entity of 1-n relations. Besides, TransGr has 0.2% and 0.5% increase respectively when predicting the head entity and tail entity of 1-1 relation, which may be due to the fact that taking structured information into account can make better use of information in the knowledge graph.

Triple classification aims at labeling triples with 1 or -1, where 1 implies that a given triple (h, r, t) is correct and -1 implies that it is incorrect. In this task, we compare TransGr with several KGE models covering TransE, TransH, TransR, TransF, NTN Socher et al. (2013) and Transpare Ji et al. (2016) on three datasets: WN11, FB13 and FB15K.

For this task, we set a threshold δ_r for each relation r in a knowledge graph maximizing the classification accuracies on the validation set. Then, for a triple (h, r, t) , if its score $f_r(h, t)$ is below δ_r , this triple is judged to be correct; otherwise, it is incorrect. During training, we searched for dimension k from $\{30, 50, 70, 100\}$, learning rate α from $\{0.5, 0.05, 0.005\}$, margin γ from $\{20, 22, 25, 27\}$ and batch size N from $\{50, 100, 200\}$. The optimal configurations are: learning rate $\alpha = 0.5$, $k = 50, \gamma = 22, N = 200$ on WN11; $\alpha = 0.1$, $k = 50$,

$\gamma = 27$, $N = 100$ on FB13; $\alpha = 0.5, k = 50$, $\gamma = 7$, $N = 100$ on FB15K.

Table 6 Evaluation results of triple classification

Dataset	WN11(%)	FB13(%)	FB15K(%)
TransR	85.9	82.5	83.9
TransSpare	86.8	87.5	88.5
NTN	70.4	87.1	68.5
TransE	75.9	81.5	79.2
TransH	78.8	83.3	82.0
TransRF	86.6	82.9	88.8
TransF	86.4	82.1	90.5
TransGr	67.9	79.3	91.9

Table 6 lists the results, in which the results of baselines are from Xiao et al. (2016). We can find that TransGr ranks first with 91.9% precision in FB15K, but ranks last with 67.9% and 79.3% precision in WN11 and FB13, respectively. For further investigation on these results, we made statistics on the graph granule information of these three datasets used. By comparison, we find that the datasets WN11 and FB13 with more entities have smaller graph granules and quite a few missing graph granules, even up to 10% in FB13. **Since we need enough graph granules to obtain the maximal concepts, the missing graph granules make the training insufficient.** In one word, there is no sufficient training data for those graph granules.

5 Conclusion and Future Work

In this paper, a set of triples sharing the same head entity was grouped as a graph granule and **the maximal concepts were obtained by a technique obtained from formal concept analysis.** By further exploration, a novel model called TransGr was developed to learn a matrix for each **maximal concept** obtained in addition to a vector for each entity and relation, respectively. The performed experiments demonstrate that the proposed TransGr model is effective on the datasets with relatively complete graph granules.

In this paper, the graph granule only includes the tail entity of a head entity and ignores the head entities of this head entity. Since these head entities also contain rich information, in future work, our effort will be focused on how to balance these two parts of information to improve the learning ability of KGE model.

6 Compliance with Ethical Standards

Funding: This work is partially supported by the National Key R&D Program of China under granted

(2018YFC0831404, 2018YFC0830605) and the National Natural Science Foundation of China (No. 12071131 and 11971211).

Conflict of Interest: The authors declare that they have no conflict of interest.

Ethical approval: This article does not contain any studies with animals performed by any of the authors.

References

- Auer S, Bizer C, Kobilarov G, Lehmann J, Cyganiak R, Ives Z (2007) Dbpedia: A nucleus for a web of open data. In: The semantic web, Springer, pp 722–735
- Balasubramaniam K (2015) Hybrid fuzzy-ontology design using fca based clustering for information retrieval in semantic web. *Procedia Computer Science* 50:135–142
- Bollacker K, Evans C, Paritosh P, Sturge T, Taylor J (2008) 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
- Bordes A, Usunier N, Garcia-Duran A, Weston J, Yakhnenko O (2013) Translating embeddings for modeling multi-relational data. In: *Proceedings of the 26th International Conference on Neural Information Processing Systems*, vol 2, pp 2787–2795
- Cai H, Zheng VW, Chang KCC (2018) A comprehensive survey of graph embedding: Problems, techniques, and applications. *IEEE Transactions on Knowledge and Data Engineering* 30(9):1616–1637
- Cigarrán JM, Penas A, Gonzalo J, Verdejo F (2005) Automatic selection of noun phrases as document descriptors in an fca-based information retrieval system. In: *International Conference on Formal Concept Analysis*, Springer, pp 49–63
- Dettmers T, Minervini P, Stenetorp P, Riedel S (2018) Convolutional 2d knowledge graph embeddings. In: *Proceedings of the AAAI Conference on Artificial Intelligence*, vol 32, pp 1811–1818
- Feng J, Huang M, Wang M, Zhou M, Hao Y, Zhu X (2015) Knowledge graph embedding by flexible translation. In: *Proceedings of the Fifteenth International Conference on Principles of Knowledge Representation and Reasoning*, pp 557–560
- Ferré S (2015) A proposal for extending formal concept analysis to knowledge graphs. In: *International Conference on Formal Concept Analysis*, Springer, Cham
- Ferré S (2017) Concepts de plus proches voisins dans des graphes de connaissances. In: *28es Journées francophones d'Ingénierie des Connaissances IC 2017*, pp 163–174

- Ferré S (2019) Link prediction in knowledge graphs with concepts of nearest neighbours. In: European Semantic Web Conference, Springer, pp 84–100
- Ferré S (2020) Application of concepts of neighbours to knowledge graph completion. *Data Science (Preprint)*:1–28
- Ferré S, Cellier P (2019) Graph-FCA: An extension of formal concept analysis to knowledge graphs. *Discrete Applied Mathematics* 273:81–102
- Fkih F, Omri MN (2016) IRAFCA: an $O(n)$ information retrieval algorithm based on formal concept analysis. *Knowledge and Information Systems* 48(2):465–491
- González L, Hogan A (2018) Modelling dynamics in semantic web knowledge graphs with formal concept analysis. In: Proceedings of the 2018 World Wide Web Conference, pp 1175–1184
- Guan N, Song D, Liao L (2019) Knowledge graph embedding with concepts. *Knowledge-Based Systems* 164:38–44
- He S, Liu K, Ji G, Zhao J (2015) Learning to represent knowledge graphs with gaussian embedding. In: Proceedings of the 24th ACM international on conference on information and knowledge management, pp 623–632
- Jabbari S, Stoffel K (2018) Fca-based ontology learning from unstructured textual data. In: International Conference on Mining Intelligence and Knowledge Exploration, Springer, pp 1–10
- Jabbari S, Stoffel K (2019) A methodology for extracting knowledge about controlled vocabularies from textual data using fca-based ontology engineering. In: 2018 IEEE International Conference on Bioinformatics and Biomedicine (BIBM), IEEE, pp 1657–1661
- Ji G, He S, Xu L, Liu K, Zhao J (2015) Knowledge graph embedding via dynamic mapping matrix. In: Proceedings of the 53rd annual meeting of the association for computational linguistics and the 7th international joint conference on natural language processing, pp 687–696
- Ji G, Liu K, He S, Zhao J (2016) Knowledge graph completion with adaptive sparse transfer matrix. In: Proceedings of the 30 AAAI Conference on Artificial Intelligence, vol 30, pp 985–991
- Jia Y, Wang Y, Jin X, Cheng X (2018) Path-specific knowledge graph embedding. *Knowledge-Based Systems* 151:37–44
- Kazemi SM, Poole D (2018) Simple embedding for link prediction in knowledge graphs. In: Proceedings of the 32nd International Conference on Neural Information Processing Systems, pp 4289–4300
- Krompa D, Baier S, Tresp V (2015) Type-constrained representation learning in knowledge graphs. In: International semantic web conference, Springer, pp 640–655
- Lacroix T, Usunier N, Obozinski G (2018) Canonical tensor decomposition for knowledge base completion. In: International Conference on Machine Learning, PMLR, pp 2863–2872
- Lin Y, Liu Z, Luan H, Sun M, Rao S, Liu S (2015a) Modeling relation paths for representation learning of knowledge bases. In: Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, pp 705–714
- Lin Y, Liu Z, Sun M, Liu Y, Zhu X (2015b) Learning entity and relation embeddings for knowledge graph completion. In: Proceedings of the 29th AAAI Conference on Artificial Intelligence, vol 29, pp 2181–2187
- Miller GA (1995) Wordnet: a lexical database for english. *Communications of the ACM* 38(11):39–41
- Nickel M, Tresp V, Kriegel HP (2011) A three-way model for collective learning on multi-relational data. In: Proceedings of the 28th International Conference on Machine Learning, pp 809–816
- Nickel M, Rosasco L, Poggio T (2016) Holographic embeddings of knowledge graphs. In: Proceedings of the 30th AAAI Conference on Artificial Intelligence, vol 30, pp 1955–1961
- Rouane-Hacene M, Huchard M, Napoli A, Valtchev P (2013) Relational concept analysis: mining concept lattices from multi-relational data. *Annals of Mathematics and Artificial Intelligence* 67(1):81–108
- Socher R, Chen D, Manning CD, Ng A (2013) Reasoning with neural tensor networks for knowledge base completion. In: Proceedings of the 26th International Conference on Neural Information Processing Systems, Citeseer, pp 926–934
- Suchanek FM, Kasneci G, Weikum G (2007) Yago: a core of semantic knowledge. In: Proceedings of the 16th international conference on World Wide Web, pp 697–706
- Tan Z, Zhao X, Fang Y, Ge B, Xiao W (2018) Knowledge graph representation via similarity-based embedding. *Scientific Programming* 2018(PT.2):1–12
- Trouillon T, Welbl J, Riedel S, Gaussier É, Bouchard G (2016) Complex embeddings for simple link prediction. In: Proceedings of the 33rd International Conference on International Conference on Machine Learning, PMLR, vol 48, pp 2071–2080
- Wang Z, Zhang J, Feng J, Chen Z (2014) Knowledge graph embedding by translating on hyperplanes. In: Proceedings of the 28th AAAI Conference on Artificial Intelligence, vol 28, pp 1112–1119
- Wille R (1982) Restructuring lattice theory: an approach based on hierarchies of concepts. In: Rival I (ed) *Ordered sets*, pp 445–470

- Wille R, Fachbereich THD (1997) Conceptual graphs and formal concept analysis. *Lecture Notes in Computer Science* pp 290–303
- Xiao H, Huang M, Hao Y, Zhu X (2016) Transg: a generative mixture model for knowledge graph embedding. In: *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, pp 2316–2325
- Yang B, Yih Wt, He X, Gao J, Deng L (2015) Embedding entities and relations for learning and inference in knowledge bases. In: *Proceedings of the International Conference on Learning Representations (ICLR) 2015*
- Zhang W, Paudel B, Wang L, Chen J, Zhu H, Zhang W, Bernstein A, Chen H (2019) Iteratively learning embeddings and rules for knowledge graph reasoning. In: *The World Wide Web Conference*, pp 2366–2377

Supplementary Files

This is a list of supplementary files associated with this preprint. Click to download.

- [Commentsreply.docx](#)