Generalized Embedding Model for Knowledge Graph Mining

Qiao Liu, Rui Wan, Xiaohui Yang, Yifu Zeng, Haibin Zhang
School of Information and Software Engineering
University of Electronic Science and Technology of China
Chengdu, China
qliu@uestc.edu.cn,{rwan,yangxhui,ifz,herb.zhang}@std.uestc.edu.cn

ABSTRACT

Many types of relations in physical, biological, social and information systems can be modeled as homogeneous or heterogeneous concept graphs. Hence, learning from and with graph embeddings has drawn a great deal of research interest recently, but developing an embedding learning method that is flexible enough to accommodate variations in physical networks is still a challenging problem. In this paper, we conjecture that the one-shot supervised learning mechanism is a bottleneck in improving the performance of the graph embedding learning, and propose to extend this by introducing a multi-shot "unsupervised" learning framework where a 2-layer MLP network for every shot .The framework can be extended to accommodate a variety of homogeneous and heterogeneous networks. Empirical results on several real-world data set show that the proposed model consistently and significantly outperforms existing state-of-the-art approaches on knowledge base completion and graph based multi-label classification tasks.

CCS CONCEPTS

• **Computing methodologies** → **Knowledge representation and reasoning**; *Neural networks*; Statistical relational learning;

KEYWORDS

representation learning, graph embedding learning, reasoning, link prediction, multi-label classification, knowledge graphs

1 INTRODUCTION

Recent studies have highlighted the importance of learning distributed representations for symbolic data in a wide variety of artificial intelligence tasks [2]. Research on word embeddings [13] has led to breakthroughs in many related areas, such as machine translation [1], question answering [24] and visual-semantic alignments [10]. However, learning to predict for large-scale knowledge graphs (KGs) is still a challenging problem left, this is largely due to the *diversity* of the ontologies, and the *semantic richness* of the concepts which makes it really hard to generate proper and universally applicable graph embeddings based on word-level embeddings [5].

Being able to generate reasonable and accurate distributed representations for large-scale KGs would be particularly valuable, in that it may help predict unobserved facts from limited concepts, uncover gaps in our knowledge, suggest new downstream applications, which clearly reflects the central concerns of the artificial intelligence [9, 16]. Therefore, massive attention has been devoted to the potential of embedding entities and relationships of multi-relational data in low-dimensional vector spaces in recent years [22].

In this paper, we consider the problem of developing simple and efficient model for learning neural representation of ***generalized knowledge graphs***, including the multi-relational *heterogeneous* graphs, and more specifically defined *homogeneous* graphs (such as social and biological networks).

Following the pioneer work[4, 18], although almost all of the state-of-the-art approaches have proven success in numerous applications by modeling the graph embedding learning problem as supervised binary classification problems, their modeling goals only consider a **one-shot** (single purpose) mapping from the embedding space to the criterion space, which we conjecture, would be vulnerable to loss considerable amount of the structured semantic information and prevents the formulation of a methodology that is objective enough to cope with the highly sparse knowledge graphs. For instance, if given a fact (*Elvis Presley, profession, singer*), one could immediately learn the following queries, which we call these queries as **multi-shot** briefly:

- Q1: What is the *profession* of *Elvis Presley*? A1: *singer*.
- Q2: Can you name a person whose *profession* is *singer*? A2: *Elvis Presley*.
- Q3: What is the possible relationship in between *Elvis Presley* and *singer*? A3: *profession*.

This is the Natural way we humans learn the meaning of concepts expressed by a statement. These self-labeled queries reflect the following modeling philosophy: (1) $(Subject, Predicate) \Rightarrow Object$; (2) $(Object, Predicate) \Rightarrow Subject$; (3) $(Subject, Object) \Rightarrow Predicate$, respectively. This has been exclusively adopted by the previous research. However, none of them have systematically investigated the effect of combining such information, so we propose to handle the embedded learning problem of KGs with a novel **multi-shot** unsupervised neural network model, called the *Graph Embedding Network* (**GEN**).The primary motivation of this paper is to develop a representation learning method that is suitable and flexible enough for modeling different types of KGs from a universal perspective. To achieve this objective, the most important problem is associated with: *how to define the optimization problem and how to solve it*.We consider modeling the facts conceptually instead of concretely (or syntactically), which means that we will focus on the semantic meanings of the embeddings, rather than their syntactic features. Meanwhile, we call GEN unsupervised for we won't give model

MLG'18, August 2018, London, United Kingdom

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM.
https://doi.org/10.1145/nnnnnnn.nnnnnnn

the label of the data, but each of the triple is decomposed into three self-labeled queries, so the model can be seen as supervised.

The major contributions are: (1) We propose GEN, a novel and efficient embedding learning framework for generalized KGs. (2)We consider multi-shot information for embedding learning simultaneously.(3) We show how GEN accords with established principles in cognitive science and provides flexibility in learning representations that works on graphs conforming to different domains.

2 RELATED WORK

During the last few years, an increasing amount of research attention has been devoted to the challenge of representation learning on KGs, especially focused on the potential benefits for the knowledge base completion (KBC) tasks, including the *entity prediction* and *relation prediction* problem. Among which, the relation translating model TransE [4], the tensor factorization based semantic matching model RESCAL [18], and the neural network based semantic matching model ER-MLP [7, 17], are probably the most heavily studied from the methodology perspective. [16], [22], and [5] have provided a good survey on such embedding learning algorithms.

Broadly speaking, related works can be divided into two categories: linear and non-linear, according to whether the output embedding has a reasonable linear interpretation. State-of-the-art linear models include the TransE, RESCAL, TranH [23], DistMult [25], and ANALOGY [11], while the popular non-linear models include the ER-MLP, ComplEx [21], HoIE [17], ProjE [20] and ConvE [6]. The proposed GEN model is also a non-linear model. Actually, the ComplEx model can be seen as an extension of DistMult in the complex space, albeit there is no nonlinear transformations applied, we treat it as a non-linear model here.

The graph embedding learning models that is most closely related to this work is probably the ProjE model, which makes use of an embedding projection function defined as:

$$\mathbf{h}(\mathbf{r},\mathbf{t}) = g(\mathbf{w}_0 \cdot f(\mathbf{w}_1^r \mathbf{r} + \mathbf{w}_1^t \mathbf{t} + \mathbf{b}_1) + \mathbf{b}_0)$$

where $\mathbf{h}, \mathbf{r}, \mathbf{t}$ denote the embedding vectors, $f(\cdot)$ and $g(\cdot)$ are non-linear activation functions, $\mathbf{w}_0$, $\mathbf{w}_1^r$ and $\mathbf{w}_1^t$ are learnable weight matrices, $\mathbf{b}_0$ and $\mathbf{b}_1$ are bias vectors. ProjE model built upon the query $(?, r, t)$ and the output ranking scores of entity h with regard to the given query $(?, r, t)$ can be obtained through a softmax function. ProjE is a one-shot solution which is distinctly different from our GEN model. In order to save the computation cost, ProjE introduced a negative sampling process, this could cause potential risks for introducing additional bias. Besides, its candidate sampling process is time-consuming and hard to work in parallel.

Another model that is closely related to the GEN model is the ER-MLP model, which can be interpreted as creating representation for each element of triples and deriving their existence from this representation [16]. This model is built upon $(h, r, t) \Rightarrow T/F$ and it is a supervised solution, which is quite different from ours. One well-known disadvantage of the ER-MLP is that, even properly regularized, it is still easily prone to over-fitting on knowledge graph datasets [17], therefore we do not compare with it in this work, but instead with the ProjE model.

Simultaneously, in order to verify the validity of our solution on heterogeneous networks, we further test it on *multi-label classification* tasks with two state-of-the-art techniques: DeepWalk [19] and node2vec [8]. Both of them are derived directly from the word2vec model [13], which embeds nodes based on the skip-gram framework, and trains the model with *corpus* generated through random walking on that graph. However, it is shown that the random walk sampling can be insufficient for supervised learning tasks in the sparse network environment [12]. Our results support this conjecture, the experimental results on benchmark tests provide strong evidence that our model performs much better.

3 METHODS

3.1 The multi-shot Learning Framework

Most of the prevalent semantic knowledge databases are built upon the Resource Description Framework (RDF) , in which the *facts* are represented and stored in the form of SPO (Subject, Predicate, Object) ***triples***. Following the convention, we will use the symbol (h, r, t) to represent a unit of *facts*, in which h, r and t denote the *head* entity, the *relation*, and the *tail* entity, respectively.

The proposed model (GEN) is designed to process data in sequential form. As shown in Fig.1, GEN consists of three components (cells), each corresponding to an individual query with regard to the given input triple. In this study, we propose to use a 2-layer MLP network to deal with the parameter estimation problem for each query individually, although it can be substituted by any other one-shot models, we only report the test results on MLP cells for simplicity. In training mode, the training set is fed into the system sequentially, each of the triple is decomposed into three **self-labeled queries**: $(h, r, ?) \Rightarrow t$, $(?, r, t) \Rightarrow h$, and $(h, ?, t) \Rightarrow r$. Each of the queries is fetched into the corresponding cell in order to update the parameters. Since for any given triple, our model would read it from three different perspective, we call it "multi-shot model" to distinguish it from other related works.

We propose to use unsupervised learning techniques for graph embedding learning tasks, because: (1) almost all of the large-scale knowledge graphs are extremely sparse, which would unavoidably degrade the quality and reliability of the supervised learning algorithms. (2) Selecting negative examples for pair-wise training would be tricky and expensive, since in practice, it is very hard to generate a "proper and informative" negative sample responsive to each of the positive examples. In order to avoid the sampling bias due to the selection of uninformative entities, we use softmax cross-entropy loss as a measure of the predictive discrepancy for model training, because its probability interpretation is more objective than those squared or logistic errors conventionally used in this area, and, it has been proven to be convex for the MLP we used in this paper [3].

3.2 Definition of the GEN cells

The network structure of the E_CELLs and the R_CELLs are quite similar, the only difference is that they have different number of neurons in the hidden layer and the output layer, which are defined as hyper-parameters as shown in Fig.1. For simplicity, we only present the implementation details of the E_CELLs here. In order to answer query $(h, r, ?) \Rightarrow t$, the hidden layer of the E_CELL takes input from the embedding dictionary according to label h and r, the hidden layer is defined as:

$$\mathbf{x}_1 = f(\mathbf{W}_o^e \cdot \mathbf{x}_0 + \mathbf{b}_0) \tag{1}$$

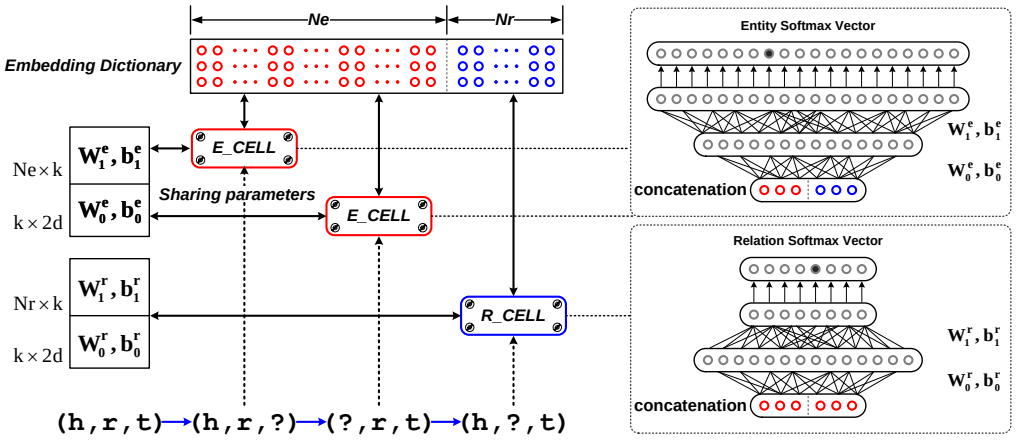

Figure 1: GEN: A Graph Embedding Model

where $\mathbf{x}_0 = [\mathbf{h} \oplus \mathbf{r}]$, denotes the concatenation of the embedding vectors, hence the $\mathbf{x}_0$ is a $2d \times 1$ real-value vector. $\mathbf{W}_o^e$ is a $k \times 2d$ weights matrix, $\mathbf{b}_0$ is a $k \times 1$ bias vector, k denotes the number of neurons in the hidden layer, and $f(\cdot)$ is a non-linear activation function, in this work, we use the rectified linear unit (ReLU) function for all the experiments [15]. The output layer takes the hidden state vector $\mathbf{x}_1$ as input, mapping it to the target label space:

$$\hat{\mathbf{y}} = g(\mathbf{W}_1^e \cdot \mathbf{x}_1 + \mathbf{b}_1) \tag{2}$$

where $\mathbf{W}_1^e$ is a $N_e \times k$ weights matrix, $\mathbf{b}_1$ is a $N_e \times 1$ bias vector, N_e denotes the number of entities in the dictionary, $g(\cdot)$ denotes the softmax function. Hence, $\hat{\mathbf{y}}$ is a $N_e \times 1$ probability vector, when training the model with a given fact (h, r, t) to answer the query $(h, r, ?)$, the predictive results output by the model is a probabilistic distribution over all of the possible candidate entities.

3.3 Model training

Parameters of the model can be logically divided into two parts. Firstly, the distribution representation of the entities and the relations are defined in the same d-dimensional space, which, as shown in Fig.1, is organized together as a *learnable* dictionary of embeddings. Secondly, there exist two types of MLP cells in the model, one deals with the entity prediction tasks, the other is responsible for the relation prediction tasks, which are marked as "E_CELL" and "R_CELL" respectively. Each individual cell has its own parameter set $\{\mathbf{W}_0, \mathbf{b}_0; \mathbf{W}_1, \mathbf{b}_1\}$ representing certain network structures. Please note that two E_CELLs are introduced to learn from the labeled entities, based on query $(h, r, ?)$ and $(?, r, t)$. According to our modeling hypothesis, which claims that all of the *relations* should be treated conceptually instead of syntactically, we propose to share parameters between the E_CELLs, the intuition behind is to let them share their *memory* of each known *facts* from both side of the *relation*, so that after training with enough knowledge, the E_CELLs will eventually be able to learn how to correctly distinguish valid entities from invalid entities for the given queries.

In training, we update parameters by calculating the cross-entropy loss with regard to each prediction result, which is defined as:

$$\mathcal{L}(\hat{\mathbf{y}}) = -\sum_{i=1}^{N_e} \mathbf{y}[i] log(\hat{\mathbf{y}}[i]) + (1 - \mathbf{y}[i]) log(1 - \hat{\mathbf{y}}[i]) \tag{3}$$

where $\mathbf{y}$ denotes the ground truth, which is a one-hot vector exclusively activated by t. To speed-up the stochastic convex optimization process, we use a mini-batch setting, and rewrite the averaged cross-entropy loss over a batch of multiple samples of size N as following simplified form:

$$\mathcal{L}(\mathbf{y}) = -\frac{1}{N} \sum_{i=1}^{N} log(\hat{\mathbf{y}}_i[\ t_i\ ]) \tag{4}$$

ÆŠwhere i denotes the i-th sample of the batch, t_i represent the index of label t in the ground truth vector of that sample. Eq.4 is computationally efficient, however, it tend to ignores the existing knowledge for query $(h, r, ?)$ other than the current fact (h, r, t), which has been proven to be useful for improving performance [20]. But, our experimental results show that the impact of the problem can be controlled by means of collaborative correction with related facts under our framework, which further demonstrate the validity of our modelling assumptions. Hopefully, the lessons learned for designing reasonable and computationally efficient cost functions in this study can serve as exemplars for future work.

4 EXPERIMENTAL RESULTS

4.1 Datasets

We evaluate the proposed model on two distinctly different types of graph embedding learning tasks. The statistics of the data sets are summarized in Table 1 and 2 .

Firstly, we evaluate GEN on **knowledge base completion** tasks with two benchmark datasets: FB15K and WN18[1] and their upgrade version FB15k-237 and WN18RR[2]. **FB15K** is a subset of Freebase, which contains facts gathered from Wikipedia, mostly focused on the topic of movies. **WN18** is a subset of WordNet which is one of

[1] Available online at: https://everest.hds.utc.fr/doku.php?id=en:transe
[2] Available online at: https://github.com/TimDettmers/ConvE

Table 1: Statistics of the complex network data sets

Dataset	# nodes	# edges	# categories	# labels
BlogCatalog	10,312	333,983	39	14,476
PPI	3,890	38,292	50	6,640
Wikipedia	4,777	184,812	40	6,770

Table 2: Statistics of the knowledge graph data sets

Dataset	WN18	WN18RR	FB15K	FB15K-327
# entities	40,943	40,943	14,951	14,541
# relations	18	11	1,345	237
# training	141,442	86,835	483,142	272,115
# validation	5,000	3,034	50,000	17,535
# test	5,000	3,134	59,071	20,466

the largest online English lexical databases, in which each distinct concept(synset) is interlinked by means of rigidly defined (hence limited) conceptual-semantic or lexical relations. Since the test set of FB15K and WN18 contains a lot of reversed triples that have been presented in the training set which can lead to biased estimation of model parameters [19]. Therefore we provide results on FB15K237 and WN18RR where the reversing relations are removed.

Secondly, we evaluate GEN on graph based multi-label classification tasks with two benchmark datasets from the complex network research area: BlogCatalog and Protein-Protein Interaction (PPI)[3]. In the training set of these networks, every node is assigned one or more labels from a finite set. *BlogCatalog* is a social network sampled from the BlogCatalog website, which consists of the social connection between the blog authors. The labels represent the interested topic categories provided by the users. PPI is a biological network sampled from the PPI network for Homo Sapiens, which consists of the existence of interactions between the proteins, where the labels represent the biological states of the proteins.

4.2 Evaluation Protocol

For multi-relational inference task, we optimized the hyper-parameters of all the datasets via extensive grid search and selected the model with the best filtered Hits@10 score on the validation set. Hyper-parameter ranges for the grid search were the following: embedding dimension d in $\{50, 100, 200, 300\}$, hidden layer dimension k in $\{256, 512, 1024, 2048\}$, MLP dropout rate p in $\{0.0, 0.1, 0.2, 0.3\}$, learning rate η in $\{0.001, 0.01, 0.1, 1, 5, 10\}$, learning rate decay λ in $\{0.75, 0.8, 0.85, 0.9, 0.95\}$. In this study, we use the following combination of parameters for all graph embedding learning tasks :

- E_CELLS: $\{d : 200, k : 2048, p : 0.2, \eta : 5, \lambda : 0.9\}$.
- R_CELLS: $\{d : 200, k : 512, p : 0.2, \eta : 5, \lambda : 0.9\}$.
- Mini-batch Settings: {batch_size : 512, epoch : 50}

For multi-label classification tasks, we implemented a single layer perceptron model with: $\{k : 128, \eta : 0.1, \lambda : 0.9\}$, which is selected through grid search with the best averaged Macro-F1 score on randomly sampled validation set from the labeled nodes. The source codes have been released on GitHub[4].

[3]Available online at: https://snap.stanford.edu/node2vec/
[4]Available at: https://github.com/uestcnlp/GEN

4.3 Knowledge Base Completion Tasks

The first evaluation was to assess the performance of GEN in link prediction tasks, by comparing it with other state-of-the-art approaches. We report the *filtered Hits@N* scores following the protocols proposed by [4] and the numerical results are presented in Table 3, where the highest scores in each column are presented in bold.

We reproduced the results of the existing studies (mostly with the released code), whereas some of which are below the reported record. For a fair comparison of the models, we cite those numbers from the original publications (marked with ⋆ symbols). Also, it seems that results reported by [6] only consider the *tail* entity prediction scenario (without averaging with the *head* entity prediction results), hence we report two version of the test results of our model, the averaged version is named as **GEN(avg.)**, and the *tail* entity prediction results are reported with model named **GEN(tail)**. Besides, we found that our model tends to remember the reverse facts with regard to the triples that has been processed during the training phase. We argue that this is an inherent characteristic of our modeling methodology, since multi-shot would treat such reverse facts as *conceptually correct*. Therefore, we also report Hits@N scores after screening out such reverse facts, this model is named as **GEN(opt)**. We consider that under certain practical circumstances, it is reasonable to care about such results, because the *reverse facts* are direct reflections of the known facts, and in many scenarios, they themselves are useful and effective facts.

From Table 3 one could see that the performance of ComplEX seems much more competitive than other models on both of the WordNet subset, however, according to our tests, TransE and HoIE perform (generalized) more stable than others for all of the subtasks. Also please note that, after filtering out the *reverse facts* from the ranking list, we recorded a significant increase in Hits@1 score on WN18, which was not observed in other models. Since most of the semantic relations defined in WordNet are reflexive [14], we believe that these results help verify the efficacy of our model framework. Further evidence can be found by looking at evaluation results on FB15K and FB15K-237, in which our model consistently and significantly outperforms others for all settings.

The goal of the second evaluation was three-folded: (1) To assess the entity prediction performance of our model. (2) To verify the validity of the multi-shot learning framework. (3) To evaluate the quality(representability) of different embedding schemes. To achieve this goal, we carried out a group of experiments depicted in Table 4, where the model name shown in the parentheses indicate that the test is based on the embeddings generated by that model, but being re-trained with our framework for fair comparison. For example, GEN(TransE) means training a GEN model with TransE embeddings, but the pre-trained embeddings will not be updated during the training process, such that the quality of the different embedding schemes can be assessed more objectively. The pre-trained word2vec embedding[5] and GloVe embedding[6] are obtained from the publicly available dictionaries released respectively by Google and Stanford NLP Group, which are also heavily studied by recent researches. For entities and relations consisting of many words, we

[5]Available at: https://code.google.com/archive/p/word2vec; version: GoogleNews-vectors-negative300.
[6]Available at: https://nlp.stanford.edu/projects/glove/; file version: glove.42B.300d.

Table 3: Link prediction results on WN18, FB15K and WN18RR, FB15K-237 (symbols: ⋆ denotes the value is cited from the original source, † denotes the result comes from [6])

Datasets	WN18			WN18RR			FB15K			FB15K-237		
Measures	Hits@1	Hits@3	Hits@10	Hits@1	Hits@3	Hits@10	Hits@1	Hits@3	Hits@10	Hits@1	Hits@3	Hits@10
TransE	44.5	85.9	93.8	2.7	33.1	42.7	36.1	59.0	76.2	17.6	29.6	44.6
TransH	33.7	79.3	87.4	1.9	33.7	40.4	33.0	59.1	70.7	19.3	34.0	44.7
HoIE	93.0*	94.5*	94.9*	35.6	37.8	39.3	40.2*	61.3*	73.9*	8.2	15.2	26.1
Analogy	**93.9***	94.4*	94.7*	37.9	39.2	41.0	64.6*	78.5*	85.4*	13.2	22.8	37.2
DistMult	72.8*	91.4*	93.6*	38.9†	43.9†	49.1†	54.6*	73.3*	82.4*	15.5†	26.3†	41.9†
ComplEX	93.6*	94.5*	94.7*	**41.1**†	**45.8**†	**50.7**†	59.9*	75.9*	84.0*	15.2†	26.3†	41.9†
ER-MLP	86.3	91.8	94.2	28.0	34.2	41.9	42.6	64.9	80.1	23.3	36.3	54.0
ConvE	93.5†	**94.7**†	**95.5**†	30.6†	36.0†	41.1†	67.0†	80.1†	87.3†	22.0†	33.0†	45.8†
ProjE	75.7	87.8	95.1	31.8	41.7	46.0	57.5	66.32	88.4*	17.3	28.0	43.0
GEN(avg.)	64.2	91.8	94.1	37.8	40.2	43.0	76.4	84.1	88.8	20.4	31.3	45.8
GEN(opt)	90.6	94.1	94.5	38.3	40.5	43.1	77.7	84.7	89.0	20.8	32.1	46.2
GEN(tail)	65.0	91.8	94.2	39.0	41.7	44.5	**78.9**	**86.9**	**91.6**	**29.5**	**42.3**	**57.7**

Table 4: Empirical comparison of the embedding schemes on FB15K dataset

Tasks	Predict h		Predict t		Predict r	
Measures(Hits)	@1	@10	@1	@10	@1	@10
GEN(GloVe)	39.79	68.80	44.64	74.72	85.24	98.57
GEN(word2vec)	48.05	75.81	52.09	81.34	86.50	98.77
GEN(HoIE)	30.55	58.86	35.66	64.84	92.28	99.68
GEN(TransE)	47.91	77.58	52.25	82.75	93.15	99.71
GEN	**73.85**	**86.01**	**78.86**	**91.64**	93.99	**99.75**
GEN($h, r \Rightarrow t$)	36.18	62.88	36.85	63.38	86.61	98.49
GEN($t, r \Rightarrow h$)	32.47	58.11	40.40	67.72	86.44	98.41
GEN($h, t \Rightarrow r$)	26.34	49.42	30.11	54.41	**94.11**	99.75

use the weighted sum of the word embeddings as their distributed representation for the test. The three models listed in the bottom of Table 4 demonstrate the one-shot learning capability of GEN, for instance, the results of GEN($h, r \Rightarrow t$) were obtained by only considering the query $(h, r, ?)$ during the training stage.

4.4 Multi-label Classification Tasks

In previous section, the term "knowledge graph" was used to refer to a multi-relational database, in which the entities were engaged in one or more heterogeneous relations, which means the relations related with a entity may range over different domains. In this section, we consider the problem of embedding learning on another type of graph — the homogeneous graphs, in which the entities were engaged in a specific relationship, which is a natural structure people use to model the physical world, such as the various social network and the biological information systems. In this study, we consider it as a generalized form of the KGs, and attempt to come up with a general-purpose framework that could be used for embedding learning on different graphs.

To verify the validity of the proposed model, we evaluate GEN by comparing its performance on multi-label classification tasks with the state-of-the-art DeepWalk and Node2vec models. Besides, we report results on TransE, HoIE and ER-MLP embeddings for comparison purpose, the supervised model used for multi-label classification are identical to each other (but differ from the embeddings). For fair comparison, the results of the DeepWalk [19] and Node2vec [8] are cited from their original sources.

Following the convention of previous authors, we randomly sample a portion of the labeled nodes as training set (and the rest are used for test), we repeat this process 9 times (with the training ratio increased from 10% to 90%), and report two of the averaged measures (w.r.t. recall, precision, and F1-measure) on each of the test, namely, macro-average and micro-average. The Macro-F1 weights equally all the *categories* regardless of how many *labels* belong to it, while the Micro-F1 weights equally all the *labels*, thus favouring the performance on common *categories*.

Numerical results are presented in Table 5 and 6, the highest scores in each column are presented in bold face. From Table 5 one could see that the performance of DeepWalk proves much more competitive than other models when labeled data is sparse, but GEN still consistently outperforms when given 50% of the data, which demonstrates the validity of the proposed embedding learning framework for modeling author connections on social networks. Next, we investigate the performance of our model on even more sparse graphs, i.e. the Protein-Protein Interactions network. Table 6 shows that GEN performs consistently and significantly better than other baselines. In fact, when trained with only 20% of the labeled proteins, GEN performs significantly better than other approaches when they are given 90% of the data. We argue that this strong performance not only indicates our model is flexible enough to the biological networks, but also provides new insights into their underlying biological mechanisms. Also please note that Macro-F1 scores in Table 5 and 6 demonstrate that, comparing with other embedding schemes, GEN performs more stable (and better) in both common and rare categories, which indicates that the embeddings generated by GEN are probably more representative and informative than other solutions, thus the supervised model built on top of it is less vulnerable to global under-fitting and local over-fitting.

Table 5: Multi-label classification results on BlogCatalog dataset

Measures	Models	10%	20%	30%	40%	50%	60%	70%	80%	90%
Micro-F1	DeepWalk	**36.00**	**38.20**	**39.60**	**40.30**	41.00	41.30	41.50	41.50	42.00
	Node2vec	34.64	36.15	36.63	37.01	37.20	37.38	38.05	38.27	40.91
	TransE	16.71	17.10	17.44	17.64	17.77	18.50	19.13	19.62	20.50
	HoIE	30.88	33.31	34.63	35.70	36.17	37.31	40.21	38.79	40.69
	ER-MLP	23.39	29.53	32.44	35.76	39.42	42.49	45.70	47.84	49.73
	GEN	27.61	31.38	35.02	38.55	**41.19**	**44.40**	**45.78**	**48.87**	**51.84**
Macro-F1	DeepWalk	**21.30**	**23.80**	25.30	26.30	27.30	27.60	27.90	28.20	28.90
	Node2vec	16.52	18.81	19.81	20.09	20.97	21.50	22.37	23.16	24.60
	TransE	2.69	3.09	3.33	3.52	3.41	3.85	4.14	4.63	5.33
	HoIE	13.86	17.10	18.98	20.84	20.77	22.65	25.64	23.06	27.79
	ER-MLP	15.86	21.32	24.67	28.46	31.64	34.66	37.42	39.74	42.47
	GEN	19.32	23.26	**26.74**	**31.06**	**33.53**	**36.57**	**38.83**	**40.27**	**44.60**

Table 6: Multi-label classification results on PPI dataset

Measures	Models	10%	20%	30%	40%	50%	60%	70%	80%	90%
Micro-F1	DeepWalk	15.36	17.40	18.26	19.41	19.75	20.23	20.46	21.52	21.79
	Node2vec	16.32	17.94	19.14	19.68	20.32	21.80	21.76	22.50	22.88
	TransE	12.80	17.69	20.94	23.57	24.58	27.32	30.42	31.84	35.20
	HoIE	14.85	18.95	21.52	24.58	27.55	29.34	31.03	33.56	35.71
	ER-MLP	9.49	14.90	19.03	23.76	25.71	32.60	35.16	36.70	46.06
	GEN	**16.36**	**27.31**	**27.97**	**32.73**	**38.10**	**42.85**	**46.43**	**51.09**	**55.16**
Macro-F1	DeepWalk	12.93	14.46	15.94	17.05	17.74	18.05	18.41	18.52	20.03
	Node2vec	13.00	15.56	16.82	17.28	17.92	18.37	19.60	20.72	21.28
	TransE	8.71	11.45	16.43	19.00	20.37	22.69	25.42	27.35	30.53
	HoIE	9.36	16.10	17.55	20.76	23.96	24.92	26.82	30.26	32.45
	ER-MLP	7.25	12.46	16.53	20.70	21.70	28.75	30.10	34.25	40.81
	GEN	**14.74**	**25.83**	**27.04**	**31.27**	**35.98**	**40.82**	**45.02**	**50.35**	**52.92**

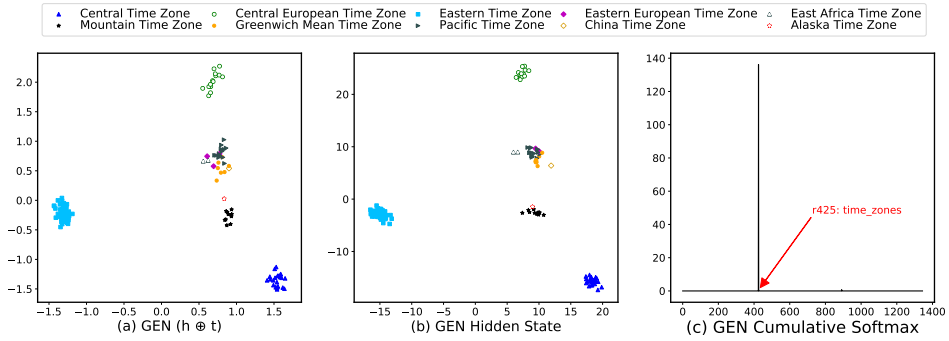

Figure 2: Visualization analysis of the GEN embedding space by using of the principal component analysis on embedding of the entities for relation prediction tasks. The case is taken from the FB15K test set, with all of the triples related to relation #425: /location/location/time_zones.

4.5 Investigating and Visualizing

In this section, we provide qualitative analysis on four typical embedding schemes (GEN, HoIE, TransE and word2vec) with the intention of better understanding the connection between the existing graph embedding schemes, and highlighting areas that remain poorly understood for further investigation. The reason we choose GEN, HoIE and TransE is because, according to our tests, they have demonstrated to be efficient and scalable to large-scale problems and are also exhibiting good generalization ability on real data sets. We also consider the word2vec embeddings because with the help of our multi-shot learning model, it achieves state-of-the-art performance on most of the knowledge base completion tasks (see

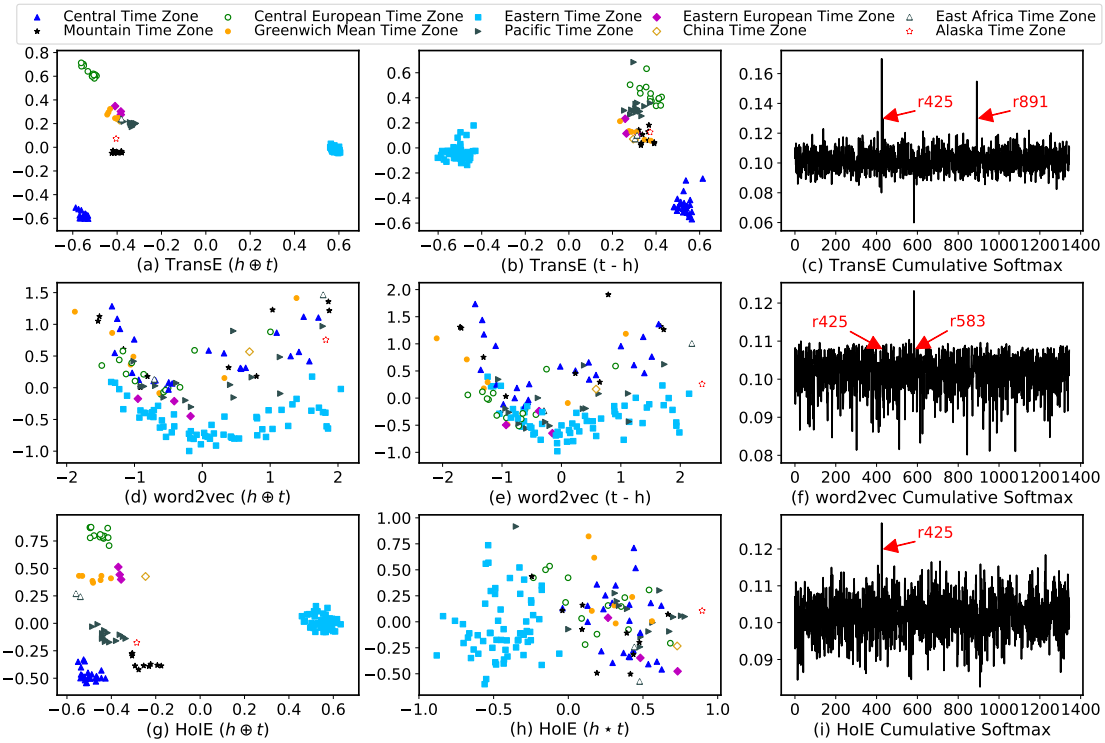

Figure 3: Visualization analysis of the TransE, HoIE and word2vec embedding schems by using of the principal component analysis on embedding of the entities for relation prediction tasks. The case is taken from the FB15K test set, relation #425: /location/location/time_zones.

Section 4.3), which is interesting and worth some consideration (probably indicates a promising potential for transfer learning).

To verify the claim that the embeddings generated by GEN are more representative and informative than other embedding schemes, we provide a case study on a randomly selected relation from FB15K, namely "/location/location/time_zones". There are 137 triples related to this relation in the test set, all of the head entities are countries or regions, and the tail entities are the corresponding time zones. The heads are uniquely different from each other, while there are only 10 different *time zones* existed in the tails.

We plot all of the 137 triples in Fig.2, in which (Fig.2a and Fig2b) the *input* multi-dimensional vectors are projected to a 2-dimensional subspace spanned by x and y, by using of the principal component analysis (PCA), then we choose the first two principal components as the principal axes. In Fig.2a, the *input* is the concatenation of the head and tail entity of each triple, i.e. $(h \oplus t)$, with the intention of investigating the patterns of such *feature vectors* for relation prediction tasks. Hence, we choose the name of the tails as legend labels. As can be seen from Fig.2a, the *feature vectors* of the 137 triples show clear clustering tendencies with regard to the categories in their tail entities. Based on this observation, we further plot the hidden layer of the R_CELL (which is a 512-dimensional vector in this case) located before the output layer in our GEN model, as depicted in Fig.2b. From Fig.2b one could see that the distance between the data points is amplified, and the distinction becomes more prominent. We plot the cumulative softmax in Fig.2c, in which the X-axis represents the 1,345 type of relations in FB15K, Y-axis denotes the cumulative softmax values. The curve is obtained by adding all of the softmax vectors output by GEN with regard to the 137 triples. Obviously, the only peak observed in Fig.2c clearly exhibit that GEN can make good use of these (concatenated) features to identify the corresponding relations correctly.

For comparison purpose, we also visualize the other three embedding schemes with the same protocol, as illustrated in Fig.3. Since the corresponding models do not use MLP for relation prediction, we can not plot their "hidden state" and "accumulate softmax" for the second and the third subplots, hence we choose to visualize their predictive criterion vectors and output ranking list instead. The processing practice is consistent with the protocol of the original literature. Specifically, for TransE, we plot $(t - h)$ as the *hidden state* for relation prediction, and calculate the $\ell 1$-norm distance $|r_i - (t - h)|_1$ w.r.t each of the relation r_i in FB15K, then we process the distance vector with the softmax function for calculation of the *accumulate softmax*. While for HoIE, we plot the circular correlation vector $(h \star t)$ as the *hidden state*, and calculate the cosine similarity of $(h \star t) \cdot r_i$ w.r.t each of the relation r_i, then we use the obtained (cosine) vector to calculate the *accumulate softmax*. For word2vec embeddings, we use the same protocol as dealing with TransE.

In Fig.3, the concatenated embedding vectors of TranE and HoIE shows similar clustering pattern as the GEN case, which help explaining the reason that under our multi-shot learning framework, the embeddings generated by these models perform similar in entity prediction tasks (see Table 4). It also provides evidence for our conjecture that these two embedding schemes could be inherently similar to each other. Their criterion vectors (the second subplot for each models) show that their clustering pattern is not as clear as the case of GEN, which help explain their performance on relation prediction tasks (as shown in the third subplot. The alternative peaks appeared in subplot Fig.3c and Fig.3f are: #891: "/base /schemastaging/phone_open_times/time_zone", and #583: "/time/time_zone/locations_in_this_time_zone"). We consider this as a solid support for the validity of the proposed multi-shot learning framework.

5 CONCLUSION

Representation learning of KGs is a key concern for artificial intelligence and cognitive science. Many types of relations in physical, biological, social and information systems can be modeled with concept (knowledge) graphs. In this paper, we present an efficient scalable framework for learning conceptual embeddings of entities and relations in generalized KGs, including the homogeneous and heterogeneous graphs. We give evidence that the proposed model learns good representations of all these graphs for knowledge inference and supervised learning. For future work, we plan to investigate more thoroughly the efficacy of the proposed modeling framework, with respect to the decomposition of the semantic information conveyed by the linked concepts into elementary information, i.e. the four Q&A pairs. Also, we seek to enhance the quality of scientific investigations and theoretical conceptualizations on graph embedding learning in the context of *semantic interoperability*, for there is usually no possibility to interpret the *embedded* information meaningfully and accurately in order to produce useful results as defined by existing algorithms.

6 ACKNOWLEDGMENTS

We thank the anonymous reviewers for taking time to read and make valuable comments on this paper. This work was supported by NSFC under grant 61133016 and 61772117, the General Equipment Department Foundation (61403120102), and the Sichuan Hi-Tech industrialization program (2017GZ0308).

REFERENCES

[1] Dzmitry Bahdanau, KyungHyun Cho, and Yoshua Bengio. 2015. Neural Machine Translation by Jointly Learning to Align and Translate. In *Proceedings of the 3rd International Conference on Learning Representations* (May 7 - 9). CoRR, San Diego, CA, USA.

[2] Yoshua Bengio, Aaron Courville, and Pascal Vincent. 2013. Representation Learning: A Review and New Perspectives. *IEEE Trans. on Pattern Analysis and Machine Intelligence* 35, 8 (2013), 1798–1828.

[3] Yoshua Bengio, Nicolas Le Roux, Pascal Vincent, Olivier Delalleau, and Patrice Marcotte. 2005. Convex neural networks. In *Proceedings of the 18th International Conference on Neural Information Processing Systems* (Dec. 05 - 08). MIT Press, Vancouver, British Columbia, Canada, 123–130.

[4] Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. 2013. Translating Embeddings for Modeling Multi-relational Data. In *Proceedings of the 26th International Conference on Neural Information Processing Systems* (Dec. 5-10). Curran Associates, Inc., Nevada, USA, 2787–2795.

[5] Hongyun Cai, Vincent W. Zheng, and Kevin Chen-Chuan Chang. 2017. A Comprehensive Survey of Graph Embedding: Problems, Techniques and Applications. *CoRR* arXiv:1709.07604 (2017).

[6] Tim Dettmers, Pasquale Minervini, Pontus Stenetorp, and Sebastian Riedel. 2017. Convolutional 2D Knowledge Graph Embeddings. *CoRR* arXiv:1409.0473 (2017), 1–14.

[7] Xin Dong, Evgeniy Gabrilovich, Geremy Heitz, Wilko Horn, Ni Lao, Kevin Murphy, Thomas Strohmann, Shaohua Sun, and Wei Zhang. 2014. Knowledge Vault: A Web-scale Approach to Probabilistic Knowledge Fusion. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (Aug. 24 - 27). ACM, New York, USA, 601–610.

[8] Aditya Grover and Jure Leskovec. 2016. Node2Vec: Scalable Feature Learning for Networks. In *Proceedings of the 22Nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (Aug. 13 - 17). ACM, San Francisco, CA, USA, 855–864.

[9] Mikael Henaff, Jason Weston, Arthur Szlam, Antoine Bordes, and Yann LeCun. 2017. Tracking the World State with Recurrent Entity Networks. In *Proceedings of the 5th International Conference on Learning Representations* (Apr. 24 - 26). CoRR, Toulon, France.

[10] Andrej Karpathy and Li Fei-Fei. 2017. Deep Visual-Semantic Alignments for Generating Image Descriptions. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 39, 4 (2017), 664–676.

[11] Hanxiao Liu, Yuexin Wu, and Yiming Yang. 2017. Analogical Inference for Multi-relational Embeddings. In *Proc. of the 34th International Conf. on Machine Learning* (Aug. 6 - 11). PMLR, 2168–2178.

[12] Qiao Liu, Liuyi Jiang, Minghao Han, Yao Liu, and Zhiguang Qin. 2016. Hierarchical Random Walk Inference in Knowledge Graphs. In *Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval* (July 17-21). ACM, 445–454.

[13] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Distributed Representations of Words and Phrases and their Compositionality. In *Proceedings of the 26th International Conference on Neural Information Processing Systems* (December 05-10). Nevada, USA, 3111–3119.

[14] George A. Miller. 1995. WordNet: A Lexical Database for English. *Comm. of the ACM* 38, 11 (1995), 39–41.

[15] Vinod Nair and Geoffrey E. Hinton. 2010. Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th International Conference on International Conference on Machine Learning* (June 21 - 24). Omnipress, Haifa, Israel, 807–814.

[16] Maximilian Nickel, Kevin Murphy, Volker Tresp, and Evgeniy Gabrilovich. 2016. A Review of Relational Machine Learning for Knowledge Graphs. *Proc. IEEE* 104, 1 (2016), 11–33.

[17] Maximilian Nickel, Lorenzo Rosasco, and Tomaso Poggio. 2016. Holographic Embeddings of Knowledge Graphs. In *Proceedings of the 30th AAAI Conference on Artificial Intelligence* (February 12 - 17). AAAI Press, Phoenix, AZ, USA, 1955–1961.

[18] Maximilian Nickel, Volker Tresp, and Hans-Peter Kriegel. 2011. A three-way model for collective learning on multi-relational data. In *Proceedings of the 28th International Conference on Machine Learning* (June 28-July 2). Omnipress, Washington, USA, 809–816.

[19] Bryan Perozzi, Rami Al-Rfou, and Steven Skiena. 2014. DeepWalk: Online Learning of Social Representations. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (Aug. 24 - 27). ACM, New York, NY, USA, 701–710.

[20] Baoxu Shi and Tim Weninger. 2017. ProjE: Embedding Projection for Knowledge Graph Completion. In *Proceedings of the 31st AAAI Conference on Artificial Intelligence* (Feb. 4-9). AAAI Press, San Francisco, CA, USA, 1236–1242.

[21] Theo Trouillon, Johannes Welbl, Sebastian Riedel, Eric Gaussier, and Guillaume Bouchard. 2016. Complex Embeddings for Simple Link Prediction. In *Proceedings of the 33rd International Conference on Machine Learning* (June 19 - 24). JMLR, New York, NY, USA, 2071–2080.

[22] Quan Wang, Zhendong Mao, Bin Wang, and Li Guo. 2017. Knowledge Graph Embedding: A Survey of Approaches and Applications. *IEEE Trans. on Knowledge and Data Engineering* PP, 99 (2017).

[23] Zhen Wang, Jianwen Zhang, Jianlin Feng, and Zheng Chen. 2014. Knowledge Graph Embedding by Translating on Hyperplanes. In *Proceedings of the 28th AAAI Conference on Artificial Intelligence* (July 27 - 31). AAAI Press, QuÃlbec City, QuÃlbec, Canada, 1112–1119.

[24] Caiming Xiong, Stephen Merity, and Richard Socher. 2016. *Dynamic memory networks for visual and textual question answering*. Proceedings of the 33rd International Conference on Machine Learning, Vol. 48. JMLR.org, New York, NY, USA.

[25] Bishan Yang, Wen-tau Yih, Xiaodong He, Jianfeng Gao, and Li Deng. 2015. Embedding Entities and Relations for Learning and Inference in Knowledge Bases. In *Proceedings of the 3rd International Conference on Learning Representations* (May 7 - 9), Vol. arXiv:1412.6575. CoRR, San Diego, CA, USA, 12 pages.