**ROTATE: KNOWLEDGE GRAPH EMBEDDING BY RELATIONAL ROTATION IN COMPLEX SPACE**

**Zhiqing Sun** 1*, **Zhi-Hong Deng** 1, **Jian-Yun Nie** 3, **Jian Tang** 2, 4, 5

1 Peking University, China
2 Mila-Quebec Institute for Learning Algorithms, Canada
3 Université de Montréal, Canada
4 HEC Montréal, Canada
5 CIFAR AI Research Chair

{1500012783, zhdeng}@pku.edu.cn
nie@iro.umontreal.ca
jian.tang@hec.ca

**ABSTRACT**

We study the problem of learning representations of entities and relations in knowledge graphs for predicting missing links. The success of such a task heavily relies on the ability of modeling and inferring the patterns of (or between) the relations. In this paper, we present a new approach for knowledge graph embedding called RotatE, which is able to model and infer various relation patterns including: symmetry/antisymmetry, inversion, and composition. Specifically, the RotatE model defines each relation as a rotation from the source entity to the target entity in the complex vector space. In addition, we propose a novel self-adversarial negative sampling technique for efficiently and effectively training the RotatE model. Experimental results on multiple benchmark knowledge graphs show that the proposed RotatE model is not only scalable, but also able to infer and model various relation patterns and significantly outperform existing state-of-the-art models for link prediction.

**1 INTRODUCTION**

Knowledge graphs are collections of factual triplets, where each triplet \((h, r, t)\) represents a relation \(r\) between a head entity \(h\) and a tail entity \(t\). Examples of real-world knowledge graphs include Freebase (Bollacker et al., 2008), Yago (Suchanek et al., 2007), and WordNet (Miller, 1995). Knowledge graphs are potentially useful to a variety of applications such as question-answering (Hao et al., 2017), information retrieval (Xiong et al., 2017), recommender systems (Zhang et al., 2016), and natural language processing (Yang & Mitchell, 2017). Research on knowledge graphs is attracting growing interests in both academia and industry communities.

Since knowledge graphs are usually incomplete, a fundamental problem for knowledge graph is predicting the missing links. Recently, extensive studies have been done on learning low-dimensional representations of entities and relations for missing link prediction (a.k.a., knowledge graph embedding) (Bordes et al., 2013, Trouillon et al., 2016, Dettmers et al., 2017). These methods have been shown to be scalable and effective. The general intuition of these methods is to model and infer the connectivity patterns in knowledge graphs according to the observed knowledge facts. For example, some relations are symmetric (e.g., marriage) while others are antisymmetric (e.g., filiation); some relations are the inverse of other relations (e.g., hypernym and hyponym); and some relations may be composed by others (e.g., my mother’s husband is my father). It is critical to find ways to model and infer these patterns, i.e., symmetry/antisymmetry, inversion, and composition, from the observed facts in order to predict missing links.

Indeed, many existing approaches have been trying to either implicitly or explicitly model one or a few of the above relation patterns (Bordes et al., 2013, Wang et al., 2014, Lin et al., 2015b).

*This work was done when the first author was visiting Mila and Université de Montréal.
KnowledgeGraphEmbedding (Nguyen et al., 2016), where TransE’s variants, such as TransH (Wang et al., 2014), TransR (Lin et al., 2015b), and STransE vectors, and 2D reshaping for real vectors in ConvE model. TransX represents a wide range of motivation is from Euler’s identity $e^{i	heta} = \cos \theta + i \sin \theta$, which indicates that a unitary complex number can be regarded as a rotation in the complex plane. Specifically, the RotatE model maps the entities and relations to the complex vector space and defines each relation as a rotation from the source entity to the target entity. Given a triplet $(h, r, t)$, we expect that $t = h \circ r$, where $h, r, t \in \mathbb{C}^k$ are the embeddings, the modulus $|r_i| = 1$ and $\circ$ denotes the Hadamard (element-wise) product. Specifically, for each dimension in the complex space, we expect that:

$$t_i = h_i r_i, \text{ where } h_i, r_i, t_i \in \mathbb{C} \text{ and } |r_i| = 1. \quad (1)$$

It turns out that such a simple operation can effectively model all the three relation patterns: symmetric/antisymmetric, inversion, and composition. For example, a relation $r$ is symmetric if and only if each element of its embedding $r$, i.e. $r_1$, satisfies $r_1 = e^{0/1\pi} = \pm 1$; two relations $r_1$ and $r_2$ are inverse if and only if their embeddings are conjugates: $r_2 = r_1^\dagger$; a relation $r_3 = e^{i\theta}$ is a combination of other two relations $r_1 = e^{i\theta_1}$ and $r_2 = e^{i\theta_2}$ if and only if $r_3 = r_1 \circ r_2$ i.e. $\theta_3 = \theta_1 + \theta_2$. Moreover, the RotatE model is scalable to large knowledge graphs as it remains linear in both time and memory.

To effectively optimizing the RotatE, we further propose a novel self-adversarial negative sampling technique, which generates negative samples according to the current entity and relation embeddings. The proposed technique is very general and can be applied to many existing knowledge graph embedding models. We evaluate the RotatE on four large knowledge graph benchmark datasets including FB15k (Bordes et al. 2013), WN18 (Bordes et al. 2013), FB15k-237 (Toutanova & Chen, 2015) and WN18RR (Dettmers et al. 2017). Experimental results show that the RotatE model significantly outperforms existing state-of-the-art approaches. In addition, RotatE also outperforms state-of-the-art models on Countries (Bouchard et al. 2015), a benchmark explicitly designed for composition pattern inference and modeling. To the best of our knowledge, RotatE is the first model that achieves state-of-the-art performance on all the benchmarks.

Table 1: The score functions $f_r(h, t)$ of several knowledge graph embedding models, where $\langle \cdot \rangle$ denotes the generalized dot product, $\circ$ denotes the Hadamard product, $\otimes$ denotes circular correlation, $\sigma$ denotes activation function and $\ast$ denotes 2D convolution. $\dagger$ denotes conjugate for complex vectors, and 2D reshaping for real vectors in ConvE model. TransX represents a wide range of TransE’s variants, such as TransH (Wang et al., 2014), TransR (Lin et al., 2015b), and STransE (Nguyen et al., 2016), where $g_{r, \cdot}(\cdot)$ denotes a matrix multiplication with respect to relation $r$.

### Related Work

1 The codes of our paper are available online: https://github.com/DeepGraphLearning/KnowledgeGraphEmbedding

2 The $p$-norm of a complex vector $v$ is defined as $\|v\|_p = \sqrt[p]{\sum |v_i|^p}$. We use L1-norm for all distance-based models in this paper and drop the subscript of $\| \cdot \|_1$ for brevity.
Predicting missing links with knowledge graph embedding (KGE) methods has been extensively investigated in recent years. The general methodology is to define a score function for the triplets. Formally, let \( \mathcal{E} \) denote the set of entities and \( \mathcal{R} \) denote the set of relations, then a knowledge graph is a collection of factual triplets \((h, r, t)\), where \( h, t \in \mathcal{E} \) and \( r \in \mathcal{R} \). Since entity embeddings are usually represented as vectors, the score function usually takes the form \( f_r(h, t) \), where \( h \) and \( t \) are head and tail entity embeddings. The score function \( f_r(h, t) \) measures the salience of a candidate triplet \((h, r, t)\). The goal of the optimization is usually to score true triplet \((h, r, t)\) higher than the corrupted false triplets \((h', r, t)\) or \((h, r, t')\). Table 1 summarizes different score functions \( f_r(h, t) \) in previous state-of-the-art methods as well as the model proposed in this paper. These models generally capture only a portion of the relation patterns. For example, TransE represents each relation as a bijection between source entities and target entities, and thus implicitly models inversion and composition of relations, but it cannot model symmetric relations; ComplEx extends DistMult by introducing complex embeddings so as to better model asymmetric relations, but it cannot infer the composition pattern. The proposed RotatE model leverages the advantages of both.

A relevant and concurrent work to our work is the TorusE (Ebisu & Ichise, 2018) model, which defines knowledge graph embedding as translations on a compact Lie group. The TorusE model can be regarded as a special case of RotatE, where the modulus of embeddings are set fixed; our RotatE is defined on the entire complex space, which has much more representation capacity. Our experiments show that this is very critical for modeling and inferring the composition patterns. Moreover, TorusE focuses on the problem of regularization in TransE while this paper focuses on modeling and inferring multiple types of relation patterns.

There are also a large body of relational approaches for modeling the relational patterns on knowledge graphs (Lao et al., 2011; Neelakantan et al., 2015; Das et al., 2016; Rocktäschel & Riedel, 2017; Yang et al., 2017). However, these approaches mainly focus on explicitly modeling the relational paths while our proposed RotatE model implicitly learns the relation patterns, which is not only much more scalable but also provides meaningful embeddings for both entities and relations.

Another related problem is how to effectively draw negative samples for training knowledge graph embeddings. This problem has been explicitly studied by Cai & Wang (2017), which proposed a generative adversarial learning framework to draw negative samples. However, such a framework requires simultaneously training the embedding model and a discrete negative sample generator, which are difficult to optimize and also computationally expensive. We propose a self-adversarial sampling scheme which only relies on the current model. It does require any additional optimization component, which make it much more efficient.

### 3 RotatE: Relational Rotation in Complex Vector Space

In this section, we introduce our proposed RotatE model. We first introduce three important relation patterns that are widely studied in the literature of link prediction on knowledge graphs. Afterwards, we introduce our proposed RotatE model, which defines relations as rotations in complex vector space. We also show that the RotatE model is able to model and infer all three relation patterns.

#### 3.1 Modeling and Inferring Relation Patterns

The key of link prediction in knowledge graph is to infer the connection patterns, e.g., relation patterns, with observed facts. According to the existing literature (Trouillon et al., 2016; Toutanova & Chen, 2015; Guu et al., 2015; Lin et al., 2015a), three types of relation patterns are very important...
and widely spread in knowledge graphs: symmetry, inversion and composition. We give their formal definition here:

**Definition 1.** A relation $r$ is symmetric (antisymmetric) if $\forall x, y$

$$r(x, y) \Rightarrow r(y, x) \quad (r(x, y) \Rightarrow \neg r(y, x))$$

A clause with such form is a 

**symmetry (antisymmetry)** pattern.

**Definition 2.** Relation $r_1$ is inverse to relation $r_2$ if $\forall x, y$

$$r_2(x, y) \Rightarrow r_1(y, x)$$

A clause with such form is a 

**inversion** pattern.

**Definition 3.** Relation $r_1$ is composed of relation $r_2$ and relation $r_3$ if $\forall x, y, z$

$$r_2(x, y) \land r_3(y, z) \Rightarrow r_1(x, z)$$

A clause with such form is a 

**composition** pattern.

According to the definition of the above three types of relation patterns, we provide an analysis of existing models on their abilities in inferring and modeling these patterns. Specifically, we provide an analysis on TransE, TransX, DistMult, and ComplEx. We did not include the analysis on HolE and ConvE since HolE is equivalent to ComplEx [Hayashi & Shimbo, 2017], and ConvE is a black box that involves two-layer neural networks and convolution operations, which are hard to analyze. The results are summarized into Table 2. We can see that no existing approaches are capable of modeling all the three relation patterns.

### 3.2 Modeling Relations as Rotations in Complex Vector Space

In this part, we introduce our proposed model that is able to model and infer all the three types of relation patterns. Inspired by Euler’s identity, we map the head and tail entities $h, t$ to the complex embeddings, i.e., $h, t \in \mathbb{C}^k$; then we define the functional mapping induced by each relation $r$ as an element-wise rotation from the head entity $h$ to the tail entity $t$. In other words, given a triple $(h, r, t)$, we expect that:

$$t = h \circ r, \quad \text{where } |r_i| = 1,$$

and $\circ$ is the Hadamard (or element-wise) product. Specifically, for each element in the embeddings, we have $t_i = h_i r_i$. Here, we constrain the modulus of each element of $r \in \mathbb{C}^k$, i.e., $r_i \in \mathbb{C}$, to be $|r_i| = 1$. By doing this, $r_i$ is of the form $e^{i \theta_{r,i}}$, which corresponds to a counterclockwise rotation by $\theta_{r,i}$ radians about the origin of the complex plane, and only affects the phases of the entity embeddings in the complex vector space. We refer to the proposed model as RotatE due to its rotational nature. According to the above definition, for each triple $(h, r, t)$, we define the distance function of RotatE as:

$$d_r(h, t) = ||h \circ r - t||$$

By defining each relation as a rotation in the complex vector spaces, RotatE can model and infer all the three types of relation patterns introduced above. Formally, we have following result:

---

3See discussion at Appendix A

4We relegate all proofs to the appendix.
Lemma 1. RotatE can infer the symmetry/antisymmetry pattern. (See proof in Appendix B)

Lemma 2. RotatE can infer the inversion pattern. (See proof in Appendix C)

Lemma 3. RotatE can infer the composition pattern. (See proof in Appendix D)

These results are also summarized into Table 2. We can see that the RotatE model is the only model that can model and infer all the three types of relation patterns.

Connection to TransE. From Table 2, we can see that TransE is able to infer and model all the other relation patterns except the symmetry pattern. The reason is that in TransE, any symmetric relation will be represented by a 0 translation vector. As a result, this will push the entities with symmetric relations to be close to each other in the embedding space. RotatE solves this problem and is able to model and infer the symmetry pattern. An arbitrary vector \( r \) that satisfies \( r_i = \pm 1 \) can be used for representing a symmetric relation in RotatE, and thus the entities having symmetric relations can be distinguished. Different symmetric relations can be also represented with different embedding vectors. Figure 1 provides illustrations of TransE and RotatE with only 1-dimensional embedding and shows how RotatE models a symmetric relation.

### 3.3 Optimization

Negative sampling has been proved quite effective for both learning knowledge graph embedding (Trouillon et al., 2016) and word embedding (Mikolov et al., 2013). Here we use a loss function similar to the negative sampling loss (Mikolov et al., 2013) for effectively optimizing distance-based models:

\[
L = - \log \sigma(\gamma - d_r(h, t)) - \sum_{i=1}^{n} \frac{1}{k} \log \sigma(d_r(h'_i, t'_i) - \gamma),
\]

(4)

where \( \gamma \) is a fixed margin, \( \sigma \) is the sigmoid function, and \((h'_i, r, t'_i)\) is the \(i\)-th negative triplet.

We also propose a new approach for drawing negative samples. The negative sampling loss samples the negative triplets in a uniform way. Such a uniform negative sampling suffers the problem of inefficiency since many samples are obviously false as training goes on, which does not provide any meaningful information. Therefore, we propose an approach called self-adversarial negative sampling, which samples negative triplets according to the current embedding model. Specifically, we sample negative triplets from the following distribution:

\[
p(h'_j, r, t'_j | \{(h_i, r_i, t_i)\}) = \frac{\exp \alpha f_r(h'_j, t'_j)}{\sum_{i} \exp \alpha f_r(h'_i, t'_i)}
\]

(5)

where \( \alpha \) is the temperature of sampling. Moreover, since the sampling procedure may be costly, we treat the above probability as the weight of the negative sample. Therefore, the final negative sampling loss with self-adversarial training takes the following form:

\[
L = - \log \sigma(\gamma - d_r(h, t)) - \sum_{i=1}^{n} p(h'_i, r, t'_i) \log \sigma(d_r(h'_i, t'_i) - \gamma)
\]

(6)

In the experiments, we will compare different approaches for negative sampling.
4 EXPERIMENTS

4.1 EXPERIMENTAL SETTING

We evaluate our proposed model on four widely used knowledge graphs. The statistics of these knowledge graphs are summarized into Table 3.

- FB15k (Bordes et al., 2013) is a subset of Freebase (Bollacker et al., 2008), a large-scale knowledge graph containing general knowledge facts. Toutanova & Chen (2015) showed that almost 81% of the test triplets (x, r, y) can be inferred via a directly linked triplet (x, r′, y) or (y, r′, x). Therefore, the key of link prediction on FB15k is to model and infer the symmetry/antisymmetry and inversion patterns.

- WN18 (Bordes et al., 2013) is a subset of WordNet (Miller, 1995), a database featuring lexical relations between words. This dataset also has many inverse relations. So the main relation patterns in WN18 are also symmetry/antisymmetry and inversion.

- FB15k-237 (Toutanova & Chen, 2015) is a subset of FB15k, where inverse relations are deleted. Therefore, the key of link prediction on FB15k-237 boils down to model and infer the symmetry/antisymmetry and composition patterns.

- WN18RR (Dettmers et al., 2017) is a subset of WN18. The inverse relations are deleted, and the main relation patterns are symmetry/antisymmetry and composition.

Hyperparameter Settings. We use Adam (Kingma & Ba, 2014) as the optimizer and fine-tune the hyperparameters on the validation dataset. The ranges of the hyperparameters for the grid search are set as follows: embedding dimension \(k \in \{125, 250, 500, 1000\}\), batch size \(b \in \{512, 1024, 2048\}\), self-adversarial sampling temperature \(\alpha \in \{0.5, 1.0\}\), and fixed margin \(\gamma \in \{3, 6, 9, 12, 18, 24, 30\}\). Both the real and imaginary parts of the entity embeddings are uniformly initialized, and the phases of the relation embeddings are uniformly initialized between 0 and 2\(\pi\). No regularization is used since we find that the fixed margin \(\gamma\) could prevent our model from over-fitting.

Evaluation Settings. We evaluate the performance of link prediction in the filtered setting: we rank test triples against all other candidate triples not appearing in the training, validation, or test set, where candidates are generated by corrupting subjects or objects: \((h′, r, t)\) or \((h, r, t′)\). Mean Rank (MR), Mean Reciprocal Rank (MRR) and Hits at N (H@N) are standard evaluation measures for these datasets and are evaluated in our experiments.

Baseline. Apart from RotatE, we propose a variant of RotatE as baseline, where the modulus of the entity embeddings are also constrained: \(|h_i| = |t_i| = C\), and the distance function is thus \(2C \| \sin \frac{\theta_i - \theta_j}{2} \|\) (See Equation 17 at Appendix F for a detailed derivation). In this way, we can investigate how RotatE works without modulus information and with only phase information. We refer to the baseline as pRotatE. It is obvious to see that pRotatE can also model and infer all the three relation patterns.

4.2 MAIN RESULTS

We compare RotatE to several state-of-the-art models, including TransE (Bordes et al., 2013), DistMult (Yao et al., 2014), ComplEx (Trouillon et al., 2016), HolE (Nickel et al., 2016), and ConvE (Dettmers et al., 2017), as well as our baseline model pRotatE, to empirically show the importance of modeling and inferring the relation patterns for the task of predicting missing links.

Table 4 summarizes our results on FB15k and WN18. We can see that RotatE outperforms all the state-of-the-art models. The performance of pRotatE and RotatE are similar on these two datasets. Table 5 summarizes our results on FB15k-237 and WN18RR, where the improvement is much more significant. The difference between RotatE and pRotatE is much larger on FB15k-237 and
Table 4: Results of several models evaluated on the FB15K and WN18 datasets. Results of [♥] are taken from (Nickel et al., 2016) and results of [♦] are taken from (Kadlec et al., 2017). Other results are taken from the corresponding original papers.

<table>
<thead>
<tr>
<th></th>
<th>FB15k</th>
<th></th>
<th></th>
<th></th>
<th>WN18</th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>MR</td>
<td>MRR</td>
<td>H@1</td>
<td>H@3</td>
<td>H@10</td>
<td>MR</td>
<td>MRR</td>
<td>H@1</td>
</tr>
<tr>
<td>TransE</td>
<td>-</td>
<td>.463</td>
<td>.297</td>
<td>.578</td>
<td>.749</td>
<td>-</td>
<td>.495</td>
<td>.113</td>
</tr>
<tr>
<td>DistMult</td>
<td>42</td>
<td>.798</td>
<td>-</td>
<td>-</td>
<td>.893</td>
<td>655</td>
<td>.797</td>
<td>-</td>
</tr>
<tr>
<td>HolE</td>
<td>-</td>
<td>.524</td>
<td>.402</td>
<td>.613</td>
<td>.739</td>
<td>-</td>
<td>.938</td>
<td>.930</td>
</tr>
<tr>
<td>ComplEx</td>
<td>-</td>
<td>.692</td>
<td>.599</td>
<td>.759</td>
<td>.840</td>
<td>-</td>
<td>.941</td>
<td>.936</td>
</tr>
<tr>
<td>ConvE</td>
<td>51</td>
<td>.657</td>
<td>.558</td>
<td>.723</td>
<td>.831</td>
<td>374</td>
<td>.943</td>
<td>.935</td>
</tr>
<tr>
<td>pRotatE</td>
<td>43</td>
<td>.799</td>
<td>.750</td>
<td>.829</td>
<td>.884</td>
<td>254</td>
<td>.947</td>
<td>.942</td>
</tr>
<tr>
<td>RotatE</td>
<td>40</td>
<td>.797</td>
<td>.746</td>
<td>.830</td>
<td>.884</td>
<td>309</td>
<td>.949</td>
<td>.944</td>
</tr>
</tbody>
</table>

Table 5: Results of several models evaluated on the FB15k-237 and WN18RR datasets. Results of [♥] are taken from (Nguyen et al., 2017). Other results are taken from (Dettmers et al., 2017).

<table>
<thead>
<tr>
<th></th>
<th>FB15k-237</th>
<th></th>
<th></th>
<th></th>
<th>WN18RR</th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>MR</td>
<td>MRR</td>
<td>H@1</td>
<td>H@3</td>
<td>H@10</td>
<td>MR</td>
<td>MRR</td>
<td>H@1</td>
</tr>
<tr>
<td>TransE</td>
<td>357</td>
<td>.294</td>
<td>-</td>
<td>-</td>
<td>.465</td>
<td>3384</td>
<td>.226</td>
<td>-</td>
</tr>
<tr>
<td>DistMult</td>
<td>254</td>
<td>.241</td>
<td>.155</td>
<td>.263</td>
<td>.419</td>
<td>5110</td>
<td>.43</td>
<td>.39</td>
</tr>
<tr>
<td>ComplEx</td>
<td>339</td>
<td>.247</td>
<td>.158</td>
<td>.275</td>
<td>.428</td>
<td>5261</td>
<td>.44</td>
<td>.41</td>
</tr>
<tr>
<td>ConvE</td>
<td>244</td>
<td>.325</td>
<td>.237</td>
<td>.356</td>
<td>.501</td>
<td>4187</td>
<td>.43</td>
<td>.40</td>
</tr>
<tr>
<td>RotatE</td>
<td>177</td>
<td>.338</td>
<td>.241</td>
<td>.375</td>
<td>.533</td>
<td>3340</td>
<td>.476</td>
<td>.428</td>
</tr>
</tbody>
</table>

WN18RR, where there are a lot of composition patterns. This indicates that modulus is very important for modeling and inferring the composition pattern.

Moreover, the performance of these models on different datasets is consistent with our analysis on the three relation patterns (Table 2):

- On FB15K, the main relation patterns are symmetry/antisymmetry and inversion. We can see that ComplEx performs well while TransE does not perform well since ComplEx can infer both symmetry/antisymmetry and inversion patterns while TransE cannot infer symmetry pattern. Surprisingly, DistMult achieves good performance on this dataset although it cannot model the antisymmetry and inversion patterns. The reason is that for most of the relations in FB15K, the types of head entities and tail entities are different. Although DistMult gives the same score to a true triplet (h, r, t) and its opposition triplet (t, r, h), (t, r, h) is usually impossible to be valid since the entity type of t does not match the head entity type of h. For example, DistMult assigns the same score to (Obama, nationality, USA) and (USA, nationality, Obama). But (USA, nationality, Obama) can be simply predicted as false since USA cannot be the head entity of the relation nationality.

- On WN18, the main relation patterns are also symmetry/antisymmetry and inversion. As expected, ComplEx still performs very well on this dataset. However, different from the results on FB15K, the performance of DistMult significantly decreases on WN18. The reason is that DistMult cannot model antisymmetry and inversion patterns, and almost all the entities in WN18 are words and belong to the same entity type, which do not have the same problem as FB15K.

- On FB15k-237, the main relation pattern is composition. We can see that TransE performs really well while ComplEx does not perform well. The reason is that, as discussed before, TransE is able to infer the composition pattern while ComplEx cannot infer the composition pattern.

- On WN18RR, one of the main relation patterns is the symmetry pattern since almost each word has a symmetric relation in WN18RR, e.g., also see and similar to. TransE does not well on this dataset since it is not able to model the symmetric relations.
countries are of the form regions and 23 subregions). Unlike link prediction on general knowledge graphs, the queries in position pattern modeling and inferring. It contains 2 relations and 272 entities (244 countries, 5 which is carefully designed to explicitly test the capabilities of the link prediction models for com-

We also evaluate our model on the Countries dataset (Bouchard et al., 2015; Nickel et al., 2016),

tries dataset has 3 tasks, each requiring inferring a composition pattern with increasing length and
to category

to category

to category

to category

respectively.

The symmetry, inversion and composition pattern is represented in Figure 2a, 2c and 2g,

Figure 2: Histograms of relation embedding phases \( \{\theta_{r_i}\} \) \( (r_i = e^{i\theta_{r_i}}) \), where \( \text{for}_1 \) represents relation \text{award\_nominee}/\text{award\_nominations}/\text{award}/\text{award\_nomination}/\text{nominated\_for},

winner represents relation \text{award\_category}/\text{winners}/\text{award}/\text{award\_honor}/\text{award\_winner}

and \( \text{for}_2 \) represents \text{award\_category}/\text{nominees}/\text{award}/\text{award\_nomination}/\text{nominated\_for}.

The symmetry, inversion and composition pattern is represented in Figure 2a, 2c and 2g, respectively.

4.3 Inferring Relation Patterns on Countries DataSet

We also evaluate our model on the Countries dataset (Bouchard et al., 2015; Nickel et al., 2016),

which is carefully designed to explicitly test the capabilities of the link prediction models for composition pattern modeling and inferring. It contains 2 relations and 272 entities (244 countries, 5 regions and 23 subregions). Unlike link prediction on general knowledge graphs, the queries in Countries are of the form \text{locatedIn}(c,?)

and the answer is one of the five regions. The Countries dataset has 3 tasks, each requiring inferring a composition pattern with increasing length and difficulty. For example, task S2 requires inferring a relatively simpler composition pattern:

\[
\text{neighborOf}(c_1, c_2) \land \text{locatedIn}(c_2, r) \Rightarrow \text{locatedIn}(c_1, r),
\]

while task S3 requires inferring the most complex composition pattern:

\[
\text{neighborOf}(c_1, c_2) \land \text{locatedIn}(c_2, s) \land \text{locatedIn}(s, r) \Rightarrow \text{locatedIn}(c_1, r).
\]

In Table 6, we report the results with respect to the AUC-PR metric, which is commonly used in the literature. We can see that RotatE outperforms all the previous models. The performance of RotatE is significantly better than other methods on S3, which is the most difficult task.

4.4 Implicit Relation Pattern Inference

In this section, we verify whether the relation patterns are implicitly represented by RotatE relation embeddings. We ignore the specific positions in the relation embedding \( \theta_r \) and plot the histogram of the phase of each element in the relation embedding, i.e., \( \{\theta_{r_i}\} \).

\textbf{Symmetry} pattern requires the symmetric relations to have property \( r \circ r = 1 \), and the solution is \( r_i = \pm 1 \). We investigate the relation embeddings from a 500-dimensional RotatE trained on WN18.

Figure 2a gives the histogram of the embedding phases of a symmetric relation \text{similar\_to}. We can find that the embedding phases are either \( \pi (r_i = -1) \) or \( 0, 2\pi (r_i = 1) \). It indicates that the RotatE
Table 7: TransE with different negative sampling techniques. The results in first 2 rows are taken from (Cai & Wang, 2017), where KBGAN uses a ComplEx negative sample generator.

<table>
<thead>
<tr>
<th></th>
<th>FB15k-237</th>
<th>WN18RR</th>
<th>WN18</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>MRR</td>
<td>H@10</td>
<td>MRR</td>
</tr>
<tr>
<td>uniform</td>
<td>.242</td>
<td>.422</td>
<td>.186</td>
</tr>
<tr>
<td>KBGAN (Cai &amp; Wang [2017])</td>
<td>.278</td>
<td>.453</td>
<td>.210</td>
</tr>
<tr>
<td>self-adversarial</td>
<td>.298</td>
<td>.475</td>
<td>.223</td>
</tr>
</tbody>
</table>

Table 8: Results of TransE and ComplEx with self-adversarial sampling and negative sampling loss on FB15k, FB15k-237 and Countries datasets.

<table>
<thead>
<tr>
<th></th>
<th>FB15k</th>
<th>FB15k-237</th>
<th>Countries (AUC-ROC)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>MRR</td>
<td>H@10</td>
<td>MRR</td>
</tr>
<tr>
<td>TransE</td>
<td>.735</td>
<td>.871</td>
<td>.332</td>
</tr>
<tr>
<td>ComplEx</td>
<td>.780</td>
<td>.890</td>
<td>.319</td>
</tr>
<tr>
<td>RotatE</td>
<td>.797</td>
<td>.884</td>
<td>.338</td>
</tr>
</tbody>
</table>

model does infer and model the symmetry pattern. Figure 2b is the histogram of relation hypernym, which shows that the embedding of a general relation does not have such a ±1 pattern.

Inversion pattern requires the embeddings of a pair of inverse relations to be conjugate. We use the same RotatE model trained on WN18 for an analysis. Figure 2c illustrates the element-wise addition of the embedding phases from relation \( r_1 = \text{hypernym} \) and its inverted relation \( r_2 = \text{hyponym} \). All the additive embedding phases are 0 or 2π, which represents that \( r_1 = r_2^{-1} \). This case shows that the inversion pattern is also inferred and modeled in the RotatE model.

Composition pattern requires the embedding phases of the composed relation to be the addition of the other two relations. Since there is no significant composition pattern in WN18, we study the inference of the composition patterns on FB15k-237, where a 1000-dimensional RotatE is trained. Figure 2d-2g illustrate such a \( r_1 = r_2 \circ r_3 \) case, where \( \theta_{2,i} + \theta_{3,i} = \theta_{1,i} \) or \( \theta_{2,i} + \theta_{3,i} = \theta_{1,i} + 2\pi \).

More results of implicitly inferring basic patterns are presented in the appendix.

4.5 Comparing Different Negative Sampling Techniques

In this part, we compare different negative sampling techniques including uniform sampling, our proposed self-adversarial technique, and the KBGAN model (Cai & Wang, 2017), which aims to optimize a generative adversarial network to generate the negative samples. We re-implement a 50-dimension TransE model with the margin-based ranking criterion that was used in (Cai & Wang, 2017), and evaluate its performance on FB15k-237, WN18RR and WN18 with self-adversarial negative sampling. Table 7 summarizes our results. We can see that self-adversarial sampling is the most effective negative sampling technique.

4.6 Further Experiments on TransE and ComplEx

One may argue that the contribution of RotatE comes from the self-adversarial negative sampling technique. In this part, we conduct further experiments on TransE and ComplEx in the same setting as RotatE to make a fair comparison among the three models. Table 8 shows the results of TransE and ComplEx trained with the self-adversarial negative sampling technique on FB15k and FB15k-237 datasets, where a large number of relations are available. In addition, we evaluate these three models on the Countries dataset, which explicitly requires inferring the composition pattern. We also provide a detailed ablation study on TransE and RotatE in the appendix.

From Table 8, we can see that similar results are observed as Table 4 and 5. The RotatE model achieves the best performance on both FB15k and FB15k-237, as it is able to model all the three relation patterns. The TransE model does not work well on the FB15k datasets, which requires modeling the symmetric pattern; the ComplEx model does not work well on FB15k-237, which requires modeling the composition pattern. The results on the Countries dataset are a little bit different, where the TransE model slightly outperforms RotatE on the S3 task. The reason is that
4.7 Experimental results on FB15k by relation category

We also did some further investigation on the performance of RotatE on different relation categories: one-to-many, many-to-one, and many-to-many relations. The results of RotatE on different relation categories on the data set FB15k are summarized into Table 9. We also compare an additional approach KG2E_KL (He et al., 2015), which is a probabilistic framework for knowledge graph embedding methods and aims to model the uncertainties of the entities and relations in knowledge graphs with the TransE model. We also summarize the statistics of different relation categories into Table 10 in the appendix.

We can see that besides the one-to-one relation, the RotatE model also performs quite well on the non-injective relations, especially on many-to-many relations. We also notice that the probabilistic framework KG2E_KL (He et al., 2015) is quite powerful, which consistently outperforms its corresponding knowledge graph embedding model, showing the importance of modeling the uncertainties in knowledge graphs. We leave the work of modeling the uncertainties in knowledge graphs with RotatE as our future work.

5 Conclusion

We have proposed a new knowledge graph embedding method called RotatE, which represents entities as complex vectors and relations as rotations in complex vector space. In addition, we propose a novel self-adversarial negative sampling technique for efficiently and effectively training the RotatE model. Our experimental results show that the RotatE model outperforms all existing state-of-the-art models on four large-scale benchmarks. Moreover, RotatE also achieves state-of-the-art results on a benchmark that is explicitly designed for composition pattern inference and modeling. A deep investigation into RotatE relation embeddings shows that the three relation patterns are implicitly represented in the relation embeddings. In the future, we plan to evaluate the RotatE model on more datasets and leverage a probabilistic framework to model the uncertainties of entities and relations.

---

Table 9: Experimental results on FB15k by relation category. The first three rows are taken from (He et al., 2015). The rest of the results are from RotatE trained with the self-adversarial negative sampling technique.

<table>
<thead>
<tr>
<th>Relation Category</th>
<th>1-to-1</th>
<th>1-to-N</th>
<th>N-to-1</th>
<th>N-to-N</th>
<th>1-to-1</th>
<th>1-to-N</th>
<th>N-to-1</th>
<th>N-to-N</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tasks</td>
<td>Prediction Head (Hits@10)</td>
<td>Prediction Tail (Hits@10)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TransE</td>
<td>.437</td>
<td>.657</td>
<td>.182</td>
<td>.472</td>
<td>.437</td>
<td>.197</td>
<td>.667</td>
<td>.500</td>
</tr>
<tr>
<td>TransH (bern)</td>
<td>.668</td>
<td>.876</td>
<td>.287</td>
<td>.645</td>
<td>.655</td>
<td>.398</td>
<td>.833</td>
<td>.672</td>
</tr>
<tr>
<td>KG2E_KL (bern)</td>
<td>.922</td>
<td>.946</td>
<td>.660</td>
<td>.696</td>
<td>.926</td>
<td>.679</td>
<td>.944</td>
<td>.754</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Tasks</th>
<th>Prediction Head (MRR)</th>
<th>Prediction Tail (MRR)</th>
</tr>
</thead>
<tbody>
<tr>
<td>TransE</td>
<td>.701</td>
<td>.912</td>
</tr>
<tr>
<td>ComplEx</td>
<td>.832</td>
<td>.914</td>
</tr>
<tr>
<td>RotatE</td>
<td>.878</td>
<td>.934</td>
</tr>
</tbody>
</table>

---

Following Wang et al. (2014), for each relation $r$, we compute the average number of tails per head ($tphr$) and the average number of head per tail ($hptr$). If $tphr < 1.5$ and $hptr < 1.5$, $r$ is treated as one-to-one; if $tphr \geq 1.5$ and $hptr \geq 1.5$, $r$ is treated as a many-to-many; if $tphr < 1.5$ and $hptr \geq 1.5$, $r$ is treated as one-to-many.
REFERENCES


APPENDIX

A DISCUSSION ON THE ABILITY OF PATTERN MODELING AND INFERENCE

No existing models are capable of modeling all the three relation patterns. For example, TransE cannot model the symmetry pattern because it would yield $r = 0$ for symmetric relations; TransX can infer and model the symmetry/antisymmetry pattern when $g_{r,1} = g_{r,2}$, e.g. in TransH (Wang et al. [2014]), but cannot infer inversion and composition as $g_{r,1}$ and $g_{r,2}$ are invertible matrix multiplications; due to its symmetric nature, DistMult is difficult to model the asymmetric and inversion pattern; ComplEx addresses the problem of DisMult and is able to infer both the symmetry and asymmetric patterns with complex embeddings. Moreover, it can infer inversion rules because the complex conjugate of the solution to $\arg\max_r \text{Re}(\langle y, r, x \rangle)$ is exactly the solution to $\arg\max_r \text{Re}(\langle y, r, x \rangle)$. However, ComplEx cannot infer composition rules, since it does not model a bijection mapping from $h$ to $t$ via relation $r$. These concerns are summarized in Table 2.

B PROOF OF LEMMA 1

Proof. if $r(x, y)$ and $r(y, x)$ hold, we have
$$y = r \circ x \land x = r \circ y \Rightarrow r \circ r = 1$$
Otherwise, if $r(x, y)$ and $\neg r(y, x)$ hold, we have
$$y = r \circ x \land x \neq r \circ y \Rightarrow r \circ r \neq 1$$

C PROOF OF LEMMA 2

Proof. if $r_1(x, y)$ and $r_2(y, x)$ hold, we have
$$y = r_1 \circ x \land x = r_2 \circ y \Rightarrow r_1 = r_2^{-1}$$

D PROOF OF LEMMA 3

Proof. if $r_1(x, z)$, $r_2(x, y)$ and $r_3(y, z)$ hold, we have
$$z = r_1 \circ x \land y = r_2 \circ x \land z = r_3 \circ y \Rightarrow r_1 = r_2 \circ r_3$$

E PROPERTIES OF ROTATE

A useful property for RotatE is that the inverse of a relation can be easily acquired by complex conjugate. In this way, the RotatE model treats head and tail entities in a uniform way, which is potentially useful for efficient 1-N scoring (Dettmers et al. [2017]):
$$\|h \circ r - t\| = \|(h \circ r - t) \circ \overline{r}\| = \|t \circ \overline{r} - h\|$$

Moreover, considering the embeddings in the polar form, i.e., $h_i = m_{h,i}e^{i\theta_{h,i}}, r_i = e^{i\theta_{r,i}}, t_i = m_{t,i}e^{i\theta_{t,i}}$, we can rewrite the RotatE distance function as:
$$\|h \circ r - t\| = \sum_{i=1}^{k} \sqrt{(m_{h,i} - m_{t,i})^2 + 4m_{h,i}m_{t,i}\sin^2 \frac{\theta_{h,i} + \theta_{r,i} - \theta_{t,i}}{2}}$$

This equation provides two interesting views of the model:

(1) When we constrain the modulus $m_{h,i} = m_{t,i} = C$, the distance function is reduced to $2C\|\sin \frac{\theta_{h,i} + \theta_{r,i} - \theta_{t,i}}{2}\|$. We can see that this is very similar to the distance function of TransE: $\|h + r - t\|$. Based on this intuition, we can show that:

Theorem 4. RotatE can degenerate into TransE. (See proof at Appendix F)

which indicates that RotatE is able to simulate TransE.

(2) The modulus provides the lower bound of the distance function, which is $\|m_h - m_t\|$.
Table 10: Statistics of FB15k by mapping properties of relations.

<table>
<thead>
<tr>
<th>Relation Category</th>
<th>1-to-1</th>
<th>1-to-N</th>
<th>N-to-1</th>
<th>N-to-N</th>
</tr>
</thead>
<tbody>
<tr>
<td>#relation</td>
<td>326</td>
<td>308</td>
<td>388</td>
<td>323</td>
</tr>
<tr>
<td>#triplet (train)</td>
<td>6827</td>
<td>42509</td>
<td>70727</td>
<td>363079</td>
</tr>
<tr>
<td>#triplet (test)</td>
<td>832</td>
<td>5259</td>
<td>8637</td>
<td>44343</td>
</tr>
</tbody>
</table>

Table 11: Results of several models evaluated on the YAGO3-10 datasets. Other results are taken from [Dettmers et al., 2017].

<table>
<thead>
<tr>
<th></th>
<th>MR</th>
<th>MRR</th>
<th>H@1</th>
<th>H@3</th>
<th>H@10</th>
</tr>
</thead>
<tbody>
<tr>
<td>DistMult</td>
<td>5926</td>
<td>.34</td>
<td>.24</td>
<td>.38</td>
<td>.54</td>
</tr>
<tr>
<td>ComplEx</td>
<td>6351</td>
<td>.36</td>
<td>.26</td>
<td>.40</td>
<td>.55</td>
</tr>
<tr>
<td>ConvE</td>
<td>1671</td>
<td>.44</td>
<td>.35</td>
<td>.49</td>
<td>.62</td>
</tr>
<tr>
<td>RotatE</td>
<td>1767</td>
<td>.495</td>
<td>.402</td>
<td>.550</td>
<td>.670</td>
</tr>
</tbody>
</table>

Table 11: Results of several models evaluated on the YAGO3-10 datasets. Other results are taken from [Dettmers et al., 2017].

F PROOF OF THEOREM

Proof. By further restricting \(|h_i| = |t_i| = C\), we can rewrite \(h, r, t\) by:

\[ h = C e^{i \theta_h} = C \cos \theta_h + i C \sin \theta_h \]
(9)

\[ r = e^{i \theta_r} = \cos \theta_r + i \sin \theta_r \]
(10)

\[ t = C e^{i \theta_t} = C \cos \theta_t + i C \sin \theta_t \]
(11)

Therefore, we have

\[ \|h \circ r - t\| = C \|e^{i(\theta_h + \theta_r)} - e^{i \theta_t}\| = C \|e^{i(\theta_h + \theta_r - \theta_t)} - 1\| \]
(13)

\[ = C \|\cos(\theta_h + \theta_r - \theta_t) - 1 + i \sin(\theta_h + \theta_r - \theta_t)\| \]
(14)

\[ = C \|\sqrt{(\cos(\theta_h + \theta_r - \theta_t) - 1)^2 + \sin^2(\theta_h + \theta_r - \theta_t)}\| \]
(15)

\[ = C \|\sqrt{2 - 2 \cos(\theta_h + \theta_r - \theta_t)}\| \]
(16)

\[ = 2C \|\sin(\theta_h + \theta_r - \theta_t) 2\| \]
(17)

If the embedding of \((h, r, t)\) in TransE is \(h', r', t'\), let \(\theta_h = ch', \theta_r = cr', \theta_t = ct'\) and \(C = 1/c\), we have

\[ \lim_{c \to 0} \|h \circ r - t\| = \|h' + r' - t'\| \]

G LINK PREDICTION ON YAGO3-10

YAGO3-10 is a subset of YAGO3 [Mahdisoltani et al., 2013], which consists of entities that have a minimum of 10 relations each. It has 123,182 entities and 37 relations. Most of the triples deal with descriptive attributes of people, such as citizenship, gender, profession and marital status.

Table 11 shows that the RotatE model also outperforms state-of-the-art models on YAGO3-10.

H HYPERPARAMETERS

We list the best hyperparameter setting of RotatE w.r.t the validation dataset on several benchmarks in Table 12.

14
Table 12: The best hyperparameter setting of RotatE on several benchmarks.

<table>
<thead>
<tr>
<th>Benchmark</th>
<th>embedding dimension $k$</th>
<th>batch size $b$</th>
<th>negative samples $n$</th>
<th>$\alpha$</th>
<th>$\gamma$</th>
</tr>
</thead>
<tbody>
<tr>
<td>FB15k</td>
<td>1000</td>
<td>2048</td>
<td>128</td>
<td>1.0</td>
<td>24</td>
</tr>
<tr>
<td>WN18</td>
<td>500</td>
<td>512</td>
<td>1024</td>
<td>0.5</td>
<td>12</td>
</tr>
<tr>
<td>FB15k-237</td>
<td>1000</td>
<td>1024</td>
<td>256</td>
<td>1.0</td>
<td>9</td>
</tr>
<tr>
<td>WN18RR</td>
<td>500</td>
<td>512</td>
<td>1024</td>
<td>0.5</td>
<td>6</td>
</tr>
<tr>
<td>Countries S1</td>
<td>500</td>
<td>512</td>
<td>64</td>
<td>1.0</td>
<td>0.1</td>
</tr>
<tr>
<td>Countries S2</td>
<td>500</td>
<td>512</td>
<td>64</td>
<td>1.0</td>
<td>0.1</td>
</tr>
<tr>
<td>Countries S3</td>
<td>500</td>
<td>512</td>
<td>64</td>
<td>1.0</td>
<td>0.1</td>
</tr>
<tr>
<td>YAGO3-10</td>
<td>500</td>
<td>1024</td>
<td>400</td>
<td>1.0</td>
<td>24</td>
</tr>
</tbody>
</table>

Table 13: Results of ablation study on FB15k-237, where “adv” represents “self-adversarial”.

<table>
<thead>
<tr>
<th></th>
<th>RotatE</th>
<th>TransE</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>MR</td>
<td>MRR</td>
</tr>
<tr>
<td>negative sampling loss</td>
<td></td>
<td></td>
</tr>
<tr>
<td>w/ adv</td>
<td>177</td>
<td>.338</td>
</tr>
<tr>
<td>w/o adv</td>
<td>185</td>
<td>.297</td>
</tr>
<tr>
<td>margin-based ranking criterion</td>
<td></td>
<td></td>
</tr>
<tr>
<td>w/ adv</td>
<td>225</td>
<td>.322</td>
</tr>
<tr>
<td>w/o adv</td>
<td>199</td>
<td>.293</td>
</tr>
</tbody>
</table>

I ABLATION STUDY

Table 13 shows our ablation study of self-adversarial sampling and negative sampling loss on FB15k-237. We also re-implement a 1000-dimension TransE and do ablation study on it. From the table, We can find that self-adversarial sampling boosts the performance for both models, while negative sampling loss is only effective on RotatE; in addition, our re-implementation of TransE also outperforms all the state-of-the-art models on FB15k-237.

J VARIANCE OF THE RESULTS

In Table 14 We provide the average and variance of the MRR results on FB15k, WN18, FB15k-237 and WN18RR. Both the average and the variance is calculated by three runs of RotatE with difference random seeds. We can find that the performance of RotatE is quite stable for different random initialization.

K MORE RESULTS OF IMPLICIT BASIC PATTERN INFERENCE

We provide more histograms of embedding phases in Figure 3-5.
Table 14: The average and variance of the MRR results of RotatE on FB15k, WN18, FB15k-237 and WN18RR.

<table>
<thead>
<tr>
<th></th>
<th>FB15k</th>
<th>WN18</th>
<th>FB15k-237</th>
<th>WN18RR</th>
</tr>
</thead>
<tbody>
<tr>
<td>MRR</td>
<td>.797 ± .001</td>
<td>.949 ± .000</td>
<td>.337 ± .001</td>
<td>.477 ± .001</td>
</tr>
</tbody>
</table>

Figure 3: Histograms of embedding phases from two general relations and four symmetric relations on WN18. ( k = 500 )
Figure 4: Histograms of embedding phases from six symmetric relations on FB15k-237. ($k = 1000$)
Figure 5: Histograms of element-wise additions of inversed relation embedding phases on WN18. 
($k = 500$)