Knowledge Graph Embedding

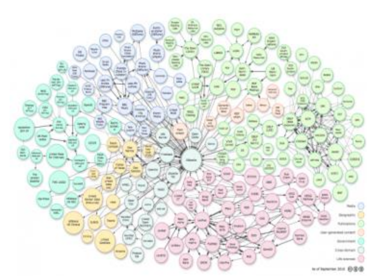
Invited Lecture (Graph7) at CMU 11441/11741

Zhiqing Sun
Email: zhiqings@cs.cmu.edu

Carnegie Mellon University
Language Technologies Institute

1

Knowledge Graphs



- A set of facts represented as triplets
  - (head entity, relation, tail entity)

- A variety of applications
  - Question answering
  - Search
  - Recommender Systems
  - Natural language understanding
  - …

Google Knowledge Graph

Microsoft SATORI

NELL: Never-Ending Language Learning

Freebase

DBpedia

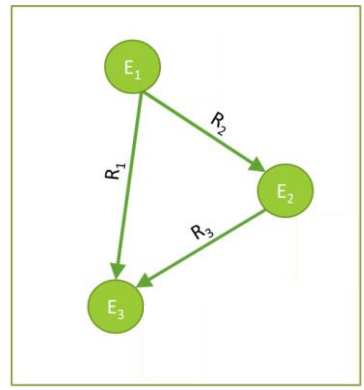
yago select knowledge

OpenIE
(Reverb, OLLIE)

2

Knowledge Graphs: Knowledge in graph form

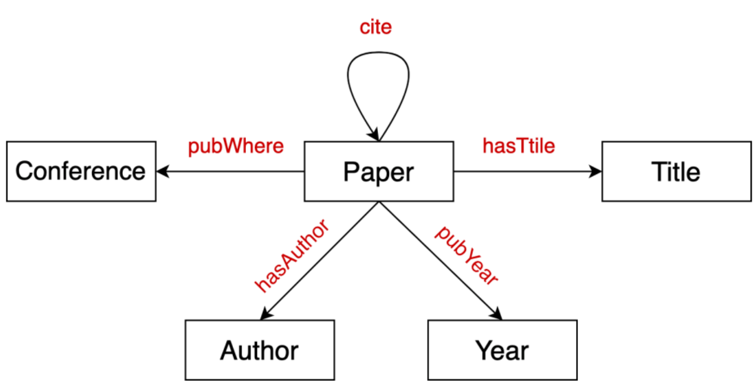
- A set of facts represented as triplets
  - with (h, r, t) for head entity, relation, tail entity
- Nodes are **entities**
- Nodes are labeled with their **types**
- Edges between two nodes capture **relationships** between entities
- KG is an example of a **heterogeneous** graph



3

Example: Bibliographic Networks

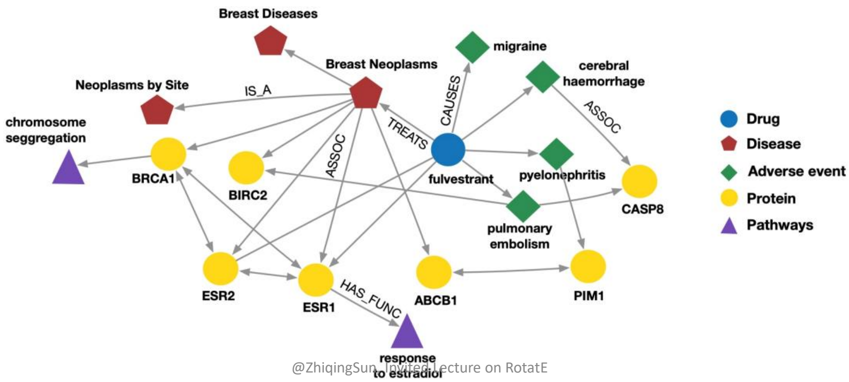
- **Node types**: paper, title, author, conference, year
- **Relationships**: pubWhere, pubYear, hasTitle, hasAuthor, cite



4

Example: Biological Knowledge Graphs

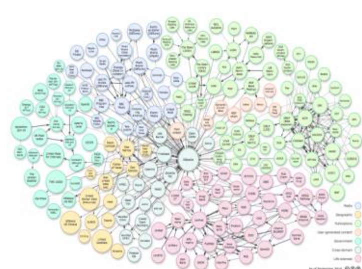
- **Node types**: drug, disease, adverse event, protein, pathways
- **Relationships**: has_func, causes, assoc, treats, is_a



5

Knowledge Graph Datasets

- Publicly available KGs:
  - FreeBase, Wikidata, Dbpedia, YAGO, NELL, etc.
- Common characteristics:
  - **Massive**: Millions of nodes and edges
  - **Incomplete**: Many true edges are missing
- Examples: Freebase
  - 93.8% of persons from Freebase have no place of birth and 78.5% have no nationality!



6

Task: Knowledge Graph Completion

- A fundamental task: **predicting missing links**
- The Key Idea: model and infer the **relation patterns** in knowledge graphs according to observed knowledge facts.
  - The relationship between relations
- Example:

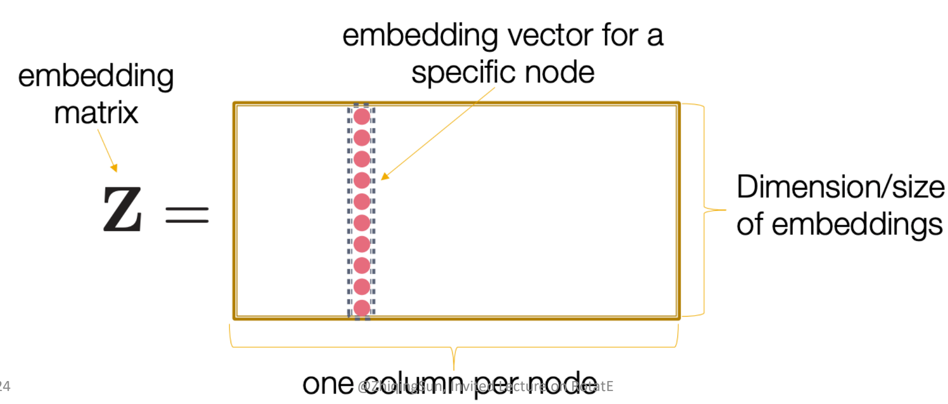
Obama_Barack **Wife** Michelle_Obama

Michelle_Obama **Husband** Obama_Barack

Parents of Parents are Grandparents

Knowledge Graph Embedding

- Representing entities as **embeddings**

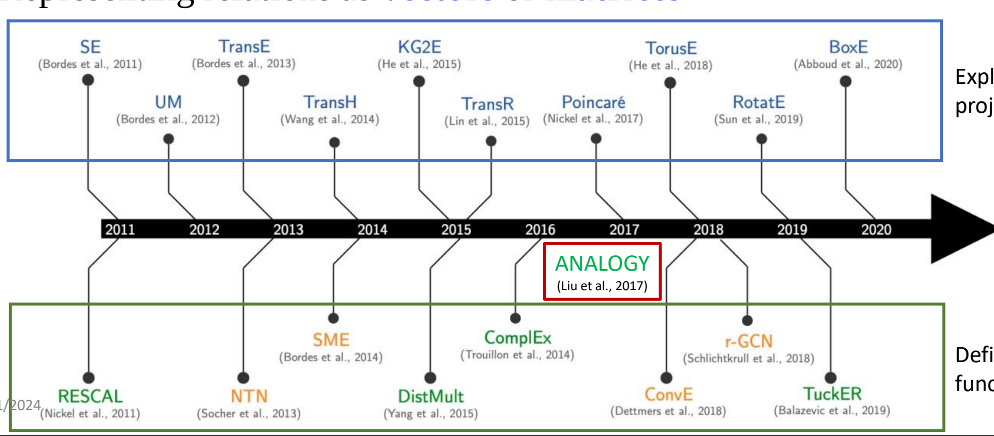


Knowledge Graph Embedding

- Representing entities as **vectors**
- Representing relations as **vectors** or **matrices**
- For each semantically valid triple (h, r, t), our goal is to establish projection $f: (\boldsymbol{h}, \boldsymbol{r}) \rightarrow \boldsymbol{t}'$ such that $\boldsymbol{t}'$ is close to the true $\boldsymbol{t}$ (here the boldfaced $\boldsymbol{h}, \boldsymbol{r}, \boldsymbol{t} \in \mathbb{R}^{\boldsymbol{d}}$ are the embeddings of the head, relation and tail in a triplet, respectively).
- Also, we want to define scoring function $\phi(\boldsymbol{h}, \boldsymbol{r}, \boldsymbol{t}) \in \mathbb{R}$ which takes the embeddings $\boldsymbol{h}, \boldsymbol{r}$ *and* $\boldsymbol{t}$ as its input, and returns a high score if (h, r, t) is semantically valid, and a low score otherwise.

Related Work in Knowledge Graph Embedding

- Representing entities as **vectors**
- Representing relations as **vectors** or **matrices**



Explicitly define the projection $f(h, r)$

Define the scoring function $\phi(h, r, t)$

Related Work on Knowledge Graph Embedding

- Representing entities as **vectors**
- Representing relations as **vectors** or **matrices**

Model	Score Function	
SE (Bordes et al., 2011)	$-\|\boldsymbol{W}_{r,1}\mathbf{h} - \boldsymbol{W}_{r,2}\mathbf{t}\|$	$\mathbf{h}, \mathbf{t} \in \mathbb{R}^k, \boldsymbol{W}_{r,\cdot} \in \mathbb{R}^{k \times k}$
TransE (Bordes et al., 2013)	$-\|\mathbf{h} + \mathbf{r} - \mathbf{t}\|$	$\mathbf{h}, \mathbf{r}, \mathbf{t} \in \mathbb{R}^k$
TransX	$-\|g_{r,1}(\mathbf{h}) + \mathbf{r} - g_{r,2}(\mathbf{t})\|$	$\mathbf{h}, \mathbf{r}, \mathbf{t} \in \mathbb{R}^k$
DistMult (Yang et al., 2014)	$\langle \mathbf{r}, \mathbf{h}, \mathbf{t} \rangle$	$\mathbf{h}, \mathbf{r}, \mathbf{t} \in \mathbb{R}^k$
ComplEx (Trouillon et al., 2016)	$\mathrm{Re}(\langle \mathbf{r}, \mathbf{h}, \overline{\mathbf{t}} \rangle)$	$\mathbf{h}, \mathbf{r}, \mathbf{t} \in \mathbb{C}^k$
HolE (Nickel et al., 2016)	$\langle \mathbf{r}, \mathbf{h} \otimes \mathbf{t} \rangle$	$\mathbf{h}, \mathbf{r}, \mathbf{t} \in \mathbb{R}^k$
ConvE (Dettmers et al., 2017)	$\langle \sigma(\mathrm{vec}(\sigma([\overline{\mathbf{r}}, \overline{\mathbf{h}}] * \boldsymbol{\Omega}))\boldsymbol{W}), \mathbf{t} \rangle$	$\mathbf{h}, \mathbf{r}, \mathbf{t} \in \mathbb{R}^k$
RotatE	$-\|\mathbf{h} \circ \mathbf{r} - \mathbf{t}\|$	$\mathbf{h}, \mathbf{r}, \mathbf{t} \in \mathbb{C}^k, \|r_i\| = 1$

Relational Patterns

- **Symmetric vs. Antisymmetric** Relations
  - Symmetric: e.g., Marriage ("A is married B" means that "B is marred A").
  - Antisymmetric: e.g., Filiation ("A is a son of B" means that "B is not a son of A").
- Formally
  - For relation r to be symmetric, we have

$$r(x, y) \text{ is true} \Rightarrow r(y, x) \text{ is true}, \ \forall x, y$$

  - For relation r to be antisymmetric, we have

$$r(x, y) \text{ is true} \Rightarrow \neg r(y, x) \text{ is true}, \forall x, y$$

Relational Patterns

- **Inverse** Relations
  - E.g., "A is the husband of B" and "B is the wife of A"
- Formally
  - For two relations r_1 and r_2 to be inversely related, we have

$$r_2(x, y) \text{ is true} \Rightarrow r_1(y, x) \text{ is true}, \forall x, y$$

Relational Patterns

- **Composition** Relations
  - My mother's husband is my father (i.e., "A is the husband of B" and " B is the mother of C" implies that "A is the father of C")
- Formally
  - For relation r_3 to be a composition of r_1 and r_2, we have

$$r_1(x, y) \wedge r_2(y, z) \Rightarrow r_3(x, z), \forall\ x, y, \mathrm{z}$$

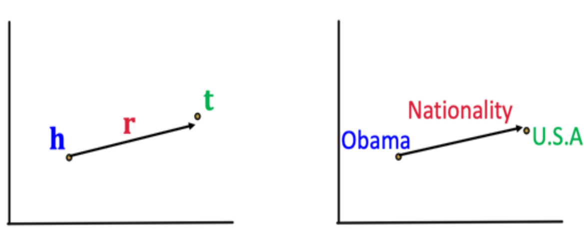
Abilities in Inferring the Relation Patterns

- None of existing methods can model and infer all the four types of relation patterns except RotatE

Model	Score Function	Symmetry	Antisymmetry	Inversion	Composition
SE	$-\|\boldsymbol{W}_{r,1}\mathbf{h} - \boldsymbol{W}_{r,2}\mathbf{t}\|$	✗	✗	✗	✗
TransE	$-\|\mathbf{h} + \mathbf{r} - \mathbf{t}\|$	✗	✓	✓	✓
TransX	$-\|g_{r,1}(\mathbf{h}) + \mathbf{r} - g_{r,2}(\mathbf{t})\|$	✓	✓	✗	✗
DistMult	$\langle \mathbf{h}, \mathbf{r}, \mathbf{t} \rangle$	✓	✗	✗	✗
ComplEx	$\mathrm{Re}(\langle \mathbf{h}, \mathbf{r}, \overline{\mathbf{t}} \rangle)$	✓	✓	✓	✗
RotatE	$-\|\mathbf{h} \circ \mathbf{r} - \mathbf{t}\|$	✓	✓	✓	✓

TransE (Bordes et al., NIPS 2013)

- Denote by boldfaced $\boldsymbol{h}, \boldsymbol{r}, \boldsymbol{t} \in \mathbb{R}^{\boldsymbol{d}}$ as the embeddings of the head (h), relation (r) and tail (t) in triplet (h, r, t), respectively.
- TransE's Objective: Find the embeddings such that $\mathbf{h} + \mathbf{r} \approx \mathbf{t}$ if triplet (h, r, t) exists in the KG, and otherwise $\mathbf{h} + \mathbf{r} \neq \mathbf{t}$.
- Scoring Function $\phi(\boldsymbol{h}, \boldsymbol{r}, \boldsymbol{t}) = -\|\boldsymbol{h} + \boldsymbol{r} - \boldsymbol{t}\|$



Supervised Learning of Embeddings

- Training set $\mathrm{S} = S^{+} \cup S^{-}$
  - $S^{+} \coloneqq \{sampled\ triplets\ (h, r, t) \in KG\}$ as the possible instances
  - $S^{-} \coloneqq \{(h', r, t) \cup (h, r, t') \notin KG\}$ as the negative instances

- Scoring function (higher is better) for any triplets

$$\phi(h, r, t) = -\|h + r - t\| = -\delta(t, \hat{t})$$

- Optimize entity/relation embeddings as

$$\max_{(h,r,t)} \sum_{x \in S^{+}} \sum_{x' \in S^{-}} [\phi(x) - \phi(x') - \gamma]_{+}$$

where x and x' are two triplets; γ is a margin hyperparameter; $[.]_{+}$ is the hinge loss.

3/21/2024 @ZhiqingSun, Invited Lecture on RotatE 17

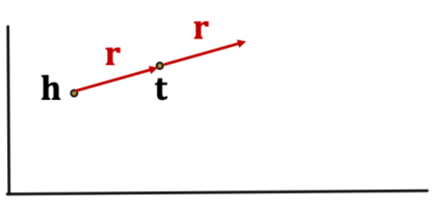
17

Analysis of TransE (Bordes et al., NIPS 2013)

- **Antisymmetric Relations:**

$$r(h, t) \Rightarrow \neg r(t, h) \quad \forall h, t$$

  - **Example**: Hypernym (a word with a broader meaning: dog v.s. poodle)
- **TransE** can model antisymmetric relations ✓
  - $\mathbf{h} + \mathbf{r} = \mathbf{t}$, but $\mathbf{t} + \mathbf{r} \neq \mathbf{h}$



3/21/2024 @ZhiqingSun, Invited Lecture on RotatE 18

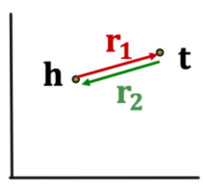
18

Analysis of TransE (Bordes et al., NIPS 2013)

- **Inverse Relations:**

$$r_2(h,t) \Rightarrow r_1(t,h)$$

  - **Example** : (Advisor, Advisee)
- TransE can model inverse relations ✓
  - $\mathbf{h} + \mathbf{r}_2 = \mathbf{t}$, we can set $\mathbf{r}_1 = -\mathbf{r}_2$



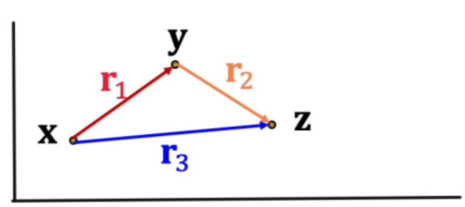
Analysis of TransE (Bordes et al., NIPS 2013)

- **Composition (Transitive) Relations:**

$$r_1(x,y) \wedge r_2(y,z) \Rightarrow r_3(x,z) \quad \forall x,y,z$$

  - **Example**: My mother's husband is my father.
- TransE can model composition relations✓

$$\mathbf{r}_3 = \mathbf{r}_1 + \mathbf{r}_2$$



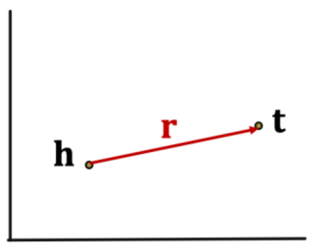
Analysis of TransE (Bordes et al., NIPS 2013)

- **Symmetric Relations:**

$$r(h,t) \Rightarrow r(t,h) \quad \forall h,t$$

  - **Example**: Family, Roommate
- TransE **cannot** model symmetric relations

  only if $\mathbf{r} = 0, \ \mathbf{h} = \mathbf{t}$



For all h,t that satisfy $r(h,t)$, $r(t,h)$ is also True, which means $||\mathbf{h}+\mathbf{r}-\mathbf{t}|| = 0$ and $||\mathbf{t}+\mathbf{r}-\mathbf{h}|| = 0$. Then $\mathbf{r} = 0$ and $\mathbf{h} = \mathbf{t}$, however h and t are two different entities and should be mapped to different locations.

21

RotatE (Sun et al., ICLR'2019)

- RotatE treats each relation as an operator of elementwise rotation from the source entity vector to the target entity vector in the complex vector space.
- RotatE can model and infer all the four types of relation patterns.
- RotatE offers an efficient and effective negative sampling algorithm for optimization.
- RotatE achieved SOTA results (at the time) on all the evaluation benchmarks for link prediction over knowledge graphs

Zhiqing Sun, Zhihong Deng, Jian-Yun Nie, and Jian Tang. "RotatE: Knowledge Graph Embedding by Relational Rotation in Complex Space." ICLR'19.

22

Elementwise Rotation in Complex Space

- Representing head and tail entities as $\mathbf{h}, \mathbf{t} \in \mathbb{C}^{\boldsymbol{k}}$ in a complex vector space.
- Define each relation **r** as an element-wise rotation from the head entity **h** to the tail entity **t**, such that

$$t_i = h_i r_i \quad \text{where } |r_i|=1.$$

- Each $\mathrm{r_i}$ can also be represented as:

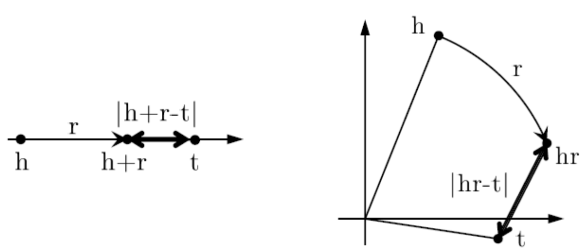
$$r_i = e^{i\theta_{r,i}} \quad \text{where } \theta_r = \left(\theta_{r,1}, \theta_{r,2}, \dots, \theta_{r,d}\right)$$

- Each $\theta_{r,i}$ in vector θ_r is the angle of **r** in the i-th dimension.

Geometric Interpretation

- Define the distance function of RotatE as

$$d_{\boldsymbol{r}}(\boldsymbol{h}, \boldsymbol{t}) = ||\mathbf{h} \circ \mathbf{r} - \mathbf{t}||$$



(a) TransE models $\mathbf{r}$ as translation in real line.

(b) RotatE models $\mathbf{r}$ as rotation in complex plane.

Modeling Symmetric and Antisymmetric

$r_i = e^{i\theta_{r,i}}$

$d_r(\boldsymbol{h}, \boldsymbol{t}) = ||\mathbf{h} \circ \mathbf{r} - \mathbf{t}||$

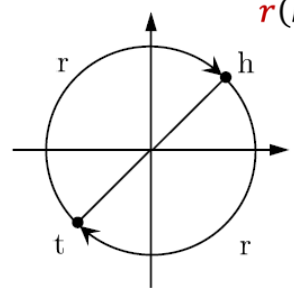
- A relation **r** is **symmetric** if and only if $r_i^2 = 1$ or $r_i = \pm 1$, i.e.,

$$\theta_{r,i} = 0 \; or \; \pi$$

- An example in the space of $\mathbb{C}$

$r(h, t) \Rightarrow r(t, h) \quad \forall h, t$

$r_i = -1$ or $\theta_{r,i} = \pi$



Modeling Symmetric and Antisymmetric

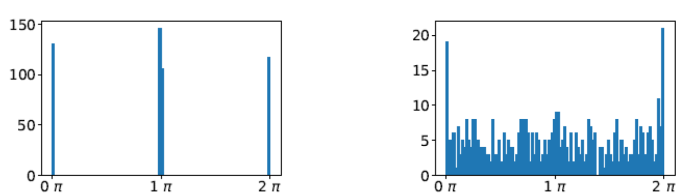
$r_i = e^{i\theta_{r,i}}$

$d_r(\boldsymbol{h}, \boldsymbol{t}) = ||\mathbf{h} \circ \mathbf{r} - \mathbf{t}||$

- A relation r is **antisymmetric** if and only if $\mathbf{r} \circ \mathbf{r} \neq \mathbf{1}$

$r(h, t) \Rightarrow r(t, h) \quad \forall h, t$

$r(h, t) \Rightarrow \neg r(t, h) \quad \forall h, t$



(a) similar_to (b) hypernym

A **symmetric** relation An **antisymmetric** relation

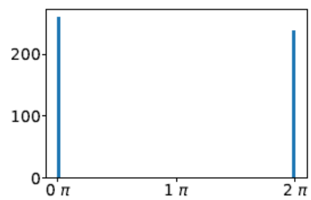
Figure: The histogram of θ_i

Modeling the Inverse Relations

$r_i = e^{i\theta_{r,i}}$
$d_r(\boldsymbol{h}, \boldsymbol{t}) = ||\mathbf{h} \circ \mathbf{r} - \mathbf{t}||$

- Two relations r_1 and r_2 are **inverse** if and only if $\mathbf{r}_1 \circ \boldsymbol{r}_2 = \mathbf{1}$ or $\mathbf{r}_2 = \bar{\mathbf{r}}_1$, i.e.,

$$\theta_{2,i} = -\theta_{1,i}$$



hypernym is the **inverse** relation of hyponym

(c) hypernym ∘ hyponym

Figure: The histogram of $\theta_{1,i} + \theta_{2,i}$

27

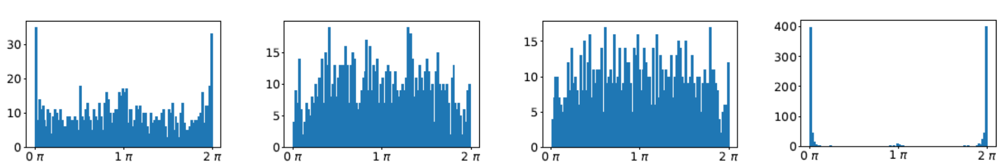
Modeling the Composition Relations

$r_i = e^{i\theta_{r,i}}$
$d_r(\boldsymbol{h}, \boldsymbol{t}) = ||\mathbf{h} \circ \mathbf{r} - \mathbf{t}||$

- d relation $\boldsymbol{r_3} = e^{i\boldsymbol{\theta_3}}$ is a **composition** of two relations $\boldsymbol{r_1} = e^{i\boldsymbol{\theta_1}}$ and $\boldsymbol{r_2} = e^{i\boldsymbol{\theta_2}}$ if only if $\boldsymbol{r_3} = \boldsymbol{r_1} \circ \boldsymbol{r_2}$, i.e.,

$$\boldsymbol{\theta_3} = \boldsymbol{\theta_1} + \boldsymbol{\theta_2}$$

Figure 2: Histograms of relation embedding phases $\{\theta_{r,i}\}$ $(r_i = e^{i\theta_{r,i}})$, where $\texttt{for}_1$ represents relation award_nominee/award_nominations./award/award_nomination/nominated_for, winner represents relation award_category/winners./award/award_honor/award_winner and $\texttt{for}_2$ represents award_category/nominees./award/award_nomination/nominated_for.



(d) $\texttt{for}_1$ (e) winner (f) $\texttt{for}_2$ (g) $\texttt{for}_2^{-1} \circ \texttt{winner} \circ \texttt{for}_1$

Figure: The histogram of $\theta_{1,i} + \theta_{2,i} - \theta_{3,i}$

28

Optimization

- Negative sampling loss

$$L = -\log \sigma\big(\gamma - d_r(\boldsymbol{h}, \boldsymbol{t})\big) - \sum_{i=1}^{k} \frac{1}{k} \log \sigma(d_r(\boldsymbol{h}'_i, \boldsymbol{t}'_i) - \gamma)$$

- γ is a fixed margin, σ is the sigmoid function, and $(\boldsymbol{h}'_i, \boldsymbol{r}, \boldsymbol{t}'_i)$ is the i-th negative triplet.

Self-adversarial Negative Sampling

- Traditionally, the negative samples are drawn in a uniform way
  - Inefficient as training goes on since many samples are obviously false
  - Does not provide useful information
- A self-adversarial negative sampling
  - Sample negative triplets according to the current embedding model
  - Starts from easier samples to more and more difficult samples
  - Curriculum Learning

$$p(h'_j, r, t'_j | \{(h_i, r_i, t_i)\}) = \frac{\exp \alpha f_r(\mathbf{h}'_j, \mathbf{t}'_j)}{\sum_i \exp \alpha f_r(\mathbf{h}'_i, \mathbf{t}'_i)}$$

- α is the temperature of sampling. $f_r(h'_j, t'_j)$ measures the salience of the triplet

The Final Objective

- Instead of sampling, treating the sampling probabilities as weights.

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

Connections to Rotation in Real Space

- Rotation in real space corresponds to the orthogonal matrices with the determinant as +1, known as special orthogonal group SO(n)
  - n =1 : [1]
  - n = 2 : $\begin{bmatrix} \cos\theta, -\sin\theta \\ \sin\theta\,, \ \cos\theta \end{bmatrix}$ θ is the rotation angle
  - n > 2: there exists an orthogonal matrix P that brings the rotation matrix Q into block diagonal:

$$P^{\mathrm{T}}QP = \begin{bmatrix} R_1 & & \\ & \ddots & \\ & & R_k \end{bmatrix} (n \text{ even}),\ P^{\mathrm{T}}QP = \begin{bmatrix} R_1 & & & \\ & \ddots & & \\ & & R_k & \\ & & & 1 \end{bmatrix} (n \text{ odd}).$$

    - Each R_i is a 2 x 2 rotation matrix
  - This is the same as RotatE!!

Experimental Results on FB15K and WN18

- Task: Link prediction, (h, r, ?) or (?, r, t)
- RotatE achieves state-of-the-art performance

	FB15k					WN18				
	MR	MRR	H@1	H@3	H@10	MR	MRR	H@1	H@3	H@10
TransE [♥]	-	.463	.297	.578	.749	-	.495	.113	.888	.943
DistMult [♦]	42	.798	-	-	**.893**	655	.797	-	-	.946
HolE	-	.524	.402	.613	.739	-	.938	.930	.945	.949
ComplEx	-	.692	.599	.759	.840	-	.941	.936	.945	.947
ConvE	51	.657	.558	.723	.831	374	.943	.935	.946	.956
pRotatE	43	**.799**	**.750**	.829	.884	**254**	.947	.942	.950	.957
RotatE	**40**	.797	.746	**.830**	.884	309	**.949**	**.944**	**.952**	**.959**

Results on FB15k-237 and WN18RR

- RotatE achieves state-of-the-art performance

	FB15k-237					WN18RR				
	MR	MRR	H@1	H@3	H@10	MR	MRR	H@1	H@3	H@10
TransE [♥]	357	.294	-	-	.465	3384	.226	-	-	.501
DistMult	254	.241	.155	.263	.419	5110	.43	.39	.44	.49
ComplEx	339	.247	.158	.275	.428	5261	.44	.41	.46	.51
ConvE	244	.325	.237	.356	.501	4187	.43	.40	.44	.52
pRotatE	178	.328	.230	.365	.524	**2923**	.462	.417	.479	.552
RotatE	**177**	**.338**	**.241**	**.375**	**.533**	3340	**.476**	**.428**	**.492**	**.571**

Further Experiments

- Comparing with the other models trained with self-adversarial
- Similar results are observed

	FB15k		**FB15k-237**	
	MRR	H@10	MRR	H@10
TransE	.735	.871	.332	.531
ComplEx	.780	**.890**	.319	.509
RotatE	**.797**	.884	**.338**	**.533**

35

Limitations

- Many relations are generally non-commutative
  - You father's wife does not equal to your wife's father
    - $\boldsymbol{r_1 \circ r_2 \neq r_2 \circ r_1}$

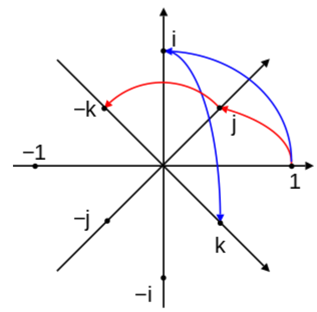
Model	**Score Function**	**Symmetry**	**Antisymmetry**	**Inversion**	**Composition**	**commutative**	**non-commutative**
SE	$-\|\boldsymbol{W}_{r,1}\mathbf{h} - \boldsymbol{W}_{r,2}\mathbf{t}\|$	✗	✗	✗	✗	✗	✓
TransE	$-\|\mathbf{h} + \mathbf{r} - \mathbf{t}\|$	✗	✓	✓	✓	✓	✗
TransX	$-\|g_{r,1}(\mathbf{h}) + \mathbf{r} - g_{r,2}(\mathbf{t})\|$	✓	✓	✗	✗	✗	✓
DistMult	$\langle \mathbf{h}, \mathbf{r}, \mathbf{t} \rangle$	✓	✗	✗	✗	✓	✗
ComplEx	$\mathrm{Re}(\langle \mathbf{h}, \mathbf{r}, \overline{\mathbf{t}} \rangle)$	✓	✓	✓	✗	✓	✗
RotatE	$-\|\mathbf{h} \circ \mathbf{r} - \mathbf{t}\|$	✓	✓	✓	✓	✓	✗

36

Limitations

- The relations are generally non-commutative
  - You father's wife does not equal to your wife's father
    - $\boldsymbol{r_1} \circ \boldsymbol{r_2} \neq \boldsymbol{r_2} \circ \boldsymbol{r_1}$
- Solution: rotation in four-dimensional space through Quaternion

$$a + b\,\mathbf{i} + c\,\mathbf{j} + d\,\mathbf{k}$$



ij = k
ji = −k
ij = −ji

Zhang S, Tay Y, Yao L, Liu Q. Quaternion knowledge graph embeddings. Advances in neural information processing systems. 2019;32.

Summary

- Modeling relation patterns is critical for knowledge base completion
  - Symmetric/Antisymmetric, Inverse, and composition
- RotatE: define each relation as an **element-wise rotation** from the head entity to the tail entity in the complex vector space
  - Capable of modeling and inferring all the four types of relation patterns
- A new self-adversarial negative sampling approach
  - Sampling the negative samples according to current embeddings
  - Curriculum learning
- State-of-the-art results on all existing benchmark data sets