Improving Knowledge Graph Embedding Using Simple Constraints

Boyang Ding, Quan Wang, Bin Wang, Li Guo
Institute of Information Engineering, Chinese Academy of Sciences
School of Cyber Security, University of Chinese Academy of Sciences

Code and data available at https://github.com/iieir-km/ComplEx-NNE_AER
Outline

1. Intro
2. Approach
3. Experiments
4. Summary
Knowledge graph

A directed graph composed of entities (nodes) and relations (edges)

- (Cristiano Ronaldo, bornIn, Funchal)
- (Cristiano Ronaldo, playsFor, Real Madrid)
- (Cristiano Ronaldo, teammates, Sergio Ramos)
- (Sergio Ramos, bornIn, Camas)
- (Sergio Ramos, playsFor, Real Madrid)
- (Funchal, locatedIn, Portugal)
- (Real Madrid, locatedIn, Spain)
- (Camas, locatedIn, Spain)
Knowledge graph embedding

- Learn to represent entities and relations in continuous vector spaces

**Entities as points in vector spaces (vectors)**
- Cristiano Ronaldo
- Sergio Ramos
- Funchal
- Real Madrid
- Camas
- Portugal
- Spain

**Relations as operations between entities (vectors/matrices/tensors)**
- teammates
- bornIn
- playsFor
- locatedIn
Knowledge graph embedding (cont.)

Easy computation and inference on knowledge graphs

- Is Spain more similar to Camas (a municipality located in Spain) or Portugal (both Portugal and Spain are European countries)?

  \[
  \text{Spain} \lessgtr \text{Camas} \lessgtr \text{Portugal}
  \]

- What is the relationship between Cristiano Ronaldo and Portugal?

  \[
  \text{argmax} \ f(\text{C. Ronaldo}, ? , \text{Portugal})
  \]
Previous approaches

- Early works
  - Simple models developed over RDF triples, e.g., TransE, RESCAL, DistMult, ComplEx, etc.

- Recent trends
  - Designing more complicated triple scoring models
    Usually with higher computational complexity
  - Incorporating extra information beyond RDF triples
    Not always applicable to all knowledge graphs
This work

- Using simple constraints to improve knowledge graph embedding
  - Non-negativity constraints on entity representations
  - Approximate entailment constraints on relation representations

- Benefits
  - More predictive embeddings
  - More interpretable embeddings
  - Low computational complexity
Basic embedding model: ComplEx

- Entity and relation representations: complex-valued vectors

\[
\text{Entity: } \mathbf{e} = \text{Re}(\mathbf{e}) + i \text{Im}(\mathbf{e}) \\
\text{Relation: } \mathbf{r} = \text{Re}(\mathbf{r}) + i \text{Im}(\mathbf{r})
\]

- Triple scoring function: multi-linear dot product

\[
\phi(e_i, r_k, e_j) \triangleq \text{Re}(\langle e_i, r_k, \bar{e}_j \rangle) \\
\triangleq \text{Re}(\sum_\ell [e_i]_\ell [r_k]_\ell [ar{e}_j]_\ell)
\]

- Triples with higher scores are more likely to be true
Non-negativity of entity representations

- Intuition
  - Uneconomical to store negative properties of an entity/concept

- Positive properties of cats
  - Cats are mammals
  - Cats eat fishes
  - Cats have four legs

- Negative properties of cats
  - Cats are not vehicles
  - Cats do not have wheels
  - Cats are not used for communication

- Non-negativity constraints

\[ 0 \leq \text{Re}(e), \text{Im}(e) \leq 1, \quad \forall e \in \mathcal{E}. \]
Approximate entailment for relations

- **Approximate entailment**
  - $r_p \overset{\lambda}{\rightarrow} r_q$: relation $r_p$ approximately entails relation $r_q$ with confidence level $\lambda$
  - **bornIn** $\overset{0.8}{\rightarrow} \text{nationality}$: a person born in a country is very likely, but not necessarily, to have a nationality of that country
  - Can be derived automatically by modern rule mining systems
Approximate entailment for relations (cont.)

- Approximate entailment constraints
  - Strict entailment $r_p \rightarrow r_q \ (\lambda = +\infty)$
    \[ \phi(e_i, r_p, e_j) \leq \phi(e_i, r_q, e_j), \quad \forall e_i, e_j \in \mathcal{E} \quad (*) \]
  - A sufficient condition for (*)
    \[ \text{Re}(r_p) \leq \text{Re}(r_q), \quad \text{Im}(r_p) = \text{Im}(r_q) \quad (**) \]
  - Introducing confidence $\lambda$ and allowing slackness in (**)
Overall model

- Basic embedding model of ComplEx + non-negativity constraints + approximate entailment constraints

\[
\min_{\Theta,\{\alpha,\beta\}} \sum_{D+\cup D-} \log \left( 1 + \exp(-y_{ijk} \phi(e_i, r_k, e_j)) \right) \\
+ \mu \sum_{\mathcal{T}} 1^T (\alpha + \beta) + \eta \|\Theta\|_2^2,
\]

s.t. \( \lambda (\text{Re}(r_p) - \text{Re}(r_q)) \leq \alpha \),
\( \lambda (\text{Im}(r_p) - \text{Im}(r_q))^2 \leq \beta \),
\( \alpha, \beta \geq 0, \quad \forall r_p \xrightarrow{\lambda} r_q \in \mathcal{T} \),
\( 0 \leq \text{Re}(e), \text{Im}(e) \leq 1, \quad \forall e \in \mathcal{E} \).

logistic loss for ComplEx

approximate entailment constraints on relation representations

non-negativity constraints on entity representations
Complexity analysis

- **Space complexity:** $O(nd + md)$  
  - $n$ is the number of entities  
  - $m$ is the number of relations  
  - $d$ is the dimensionality of the embedding space

- **Time complexity per iteration:** $O(sd + \bar{n}d + td) \sim O(sd)$  
  - $s$ is the average number of triples in a mini-batch  
  - $\bar{n}$ is the average number of entities in a mini-batch  
  - $t$ is the total number of approximate entailments
1. Intro
2. Approach
3. Experiments
4. Summary
Experimental setups

- **Datasets**
  - WN18: subset of WordNet
  - FB15k: subset of Freebase
  - DB100k: subset of DBpedia
  - Training/validation/test split

- **Approximate entailment**
  - Automatically extracted by AMIE+ with confidence level higher than 0.8

<table>
<thead>
<tr>
<th>Dataset</th>
<th># Ent</th>
<th># Rel</th>
<th># Train/Valid/Test</th>
<th># Cons</th>
</tr>
</thead>
<tbody>
<tr>
<td>WN18</td>
<td>40,943</td>
<td>18</td>
<td>141,442</td>
<td>5,000</td>
</tr>
<tr>
<td>FB15K</td>
<td>14,951</td>
<td>1,345</td>
<td>483,142</td>
<td>50,000</td>
</tr>
<tr>
<td>DB100K</td>
<td>99,604</td>
<td>470</td>
<td>597,572</td>
<td>50,000</td>
</tr>
</tbody>
</table>

Approximate entailment:
- `hyponym`:
  - `synset_domain_topic_of` with confidence 0.99
  - `member_of_domain_topic`
  - `instance_hyponym` with confidence 0.98

- `/people/place_of_birth`:
  - `location/people_born_here` with confidence 1.00

- `/film/directed_by`:
  - `director/film` with confidence 0.98

- `/country/admin_divisions`:
  - `country/1st_level_divisions` with confidence 0.91

- `owner`:
  - `owning_company` with confidence 0.95

- `child`:
  - `parent` with confidence 0.92

- `distributing_company`:
  - `distributing_label` with confidence 0.92
Experimental setups (cont.)

- **Link prediction**
  - To complete a triple \((e_i, r_k, e_j)\) with \(e_i\) or \(e_j\) missing

- **Baselines**
  - Simple embedding models based on RDF triples
  - Other extensions of ComplEx incorporating logic rules
  - Recently developed neural network architectures

- **Our approaches**
  - ComplEx-NNE: only with non-negativity constraints
  - ComplEx-NNE+AER: also with approximate entailment constraints
## Link prediction results

<table>
<thead>
<tr>
<th></th>
<th>WN18</th>
<th></th>
<th></th>
<th>FB15K</th>
<th></th>
<th></th>
<th>DB100K</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>MRR</td>
<td>HITS@1</td>
<td>HITS@3</td>
<td>MRR</td>
<td>HITS@1</td>
<td>HITS@3</td>
<td>MRR</td>
<td>HITS@1</td>
<td>HITS@3</td>
</tr>
<tr>
<td>TransE(2013)</td>
<td>0.454</td>
<td>0.089</td>
<td>0.823</td>
<td>0.380</td>
<td>0.231</td>
<td>0.472</td>
<td>0.111</td>
<td>0.016</td>
<td>0.164</td>
</tr>
<tr>
<td>DistMult(2015)</td>
<td>0.822</td>
<td>0.728</td>
<td>0.914</td>
<td>0.654</td>
<td>0.546</td>
<td>0.733</td>
<td>0.233</td>
<td>0.115</td>
<td>0.301</td>
</tr>
<tr>
<td>HolE(2016)</td>
<td>0.938</td>
<td>0.930</td>
<td>0.945</td>
<td>0.524</td>
<td>0.402</td>
<td>0.613</td>
<td>0.260</td>
<td>0.182</td>
<td>0.309</td>
</tr>
<tr>
<td>ComplEx(2016)</td>
<td>0.941</td>
<td>0.936</td>
<td>0.945</td>
<td>0.692</td>
<td>0.599</td>
<td>0.759</td>
<td>0.242</td>
<td>0.126</td>
<td>0.312</td>
</tr>
<tr>
<td>ANALOGY(2017)</td>
<td>0.942</td>
<td>0.939</td>
<td>0.944</td>
<td>0.725</td>
<td>0.646</td>
<td>0.785</td>
<td>0.252</td>
<td>0.143</td>
<td>0.323</td>
</tr>
<tr>
<td>RUGE(2018)</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>0.768</td>
<td>0.703</td>
<td>0.815</td>
<td>0.246</td>
<td>0.129</td>
<td>0.325</td>
</tr>
<tr>
<td>ComplEx$^R$(2017)</td>
<td>0.940</td>
<td>—</td>
<td>0.943</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>0.253</td>
<td>0.167</td>
<td>0.294</td>
</tr>
<tr>
<td>R-GCN(2017)</td>
<td>0.814</td>
<td>0.686</td>
<td>0.928</td>
<td>0.651</td>
<td>0.541</td>
<td>0.736</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>R-GCN+(2017)</td>
<td>0.819</td>
<td>0.697</td>
<td>0.929</td>
<td>0.696</td>
<td>0.601</td>
<td>0.760</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>ConvE(2018)</td>
<td>0.942</td>
<td>0.935</td>
<td>0.947</td>
<td>0.745</td>
<td>0.670</td>
<td>0.801</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>Single DistMult(2017)</td>
<td>0.797</td>
<td>—</td>
<td>—</td>
<td>0.798</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>ComplEx-NNE</td>
<td>0.941</td>
<td>0.937</td>
<td>0.944</td>
<td>0.727</td>
<td>0.659</td>
<td>0.772</td>
<td>0.298</td>
<td>0.229</td>
<td>0.330</td>
</tr>
<tr>
<td>ComplEx-NNE+AER</td>
<td><strong>0.943</strong></td>
<td><strong>0.940</strong></td>
<td>0.945</td>
<td><strong>0.803</strong></td>
<td><strong>0.761</strong></td>
<td><strong>0.831</strong></td>
<td><strong>0.306</strong></td>
<td><strong>0.244</strong></td>
<td><strong>0.334</strong></td>
</tr>
</tbody>
</table>

- **Simple embedding models**
- **Incorporating logic rules**
- **Neural network architectures**

ComplEx-NNE+AER can beat very strong baselines just by introducing the simple constraints.
Analysis on entity representations

- Visualization of entity representations
  - Pick 4 types: reptile/wine region/species/programming language, and randomly select 30 entities from each type.
  - Visualize the representations of these entities learned by ComplEx and ComplEx-NNE+AER.

Compact and interpretable entity representations

- Each entity is represented by only a relatively small number of “active” dimensions.
- Entities with the same type tend to activate the same set of dimensions.
Analysis on entity representations (cont.)

- Semantic purity of latent dimensions
  - For each latent dimension, pick top K percent of entities with the highest activation values on this dimension
  - Calculate the entropy of the type distribution of these entities

Latent dimensions with higher semantic purity
- A lower entropy means entities along this dimension tend to have the same type (higher semantic purity)
Analysis on relation representations

- Visualization of relation representations

- Encode logical regularities quite well
  - Equivalence $r_p \leftrightarrow r_q$
    \[
    \text{Re}(r_p) = \text{Re}(r_q) \quad \text{Im}(r_p) = \text{Im}(r_q)
    \]
  - Inversion $r_p \leftrightarrow r_q^{-1}$
    \[
    \text{Re}(r_p) = \text{Re}(r_q) \quad \text{Im}(r_p) = -\text{Im}(r_q)
    \]
  - Ordinary entailment
    \[
    \text{Re}(r_p) \leq \text{Re}(r_q) \quad \text{Im}(r_p) = \text{Im}(r_q)
    \]
This work

- Using simple constraints to improve knowledge graph embedding
  - Non-negativity constraints on entity representations
  - Approximate entailment constraints on relation representations

- Experimental results
  - Effective
  - Efficient
  - Interpretable embeddings

Code and data available at https://github.com/iieir-km/ComplEx-NNE_AER
Thank you!

Q&A

wangquan@iie.ac.cn