Text-enhanced Representation Learning for Knowledge Graph

Reporter: Zhigang WANG
Authors: Zhigang WANG, Juanzi LI, Zhiyuan LIU, Jie TANG
Tsinghua University
To appear in IJCAI 2016
2016-04-17
Outline

- Introduction
- Problem Definition
- Our Proposed Approach
- Experiments and Analysis
- Conclusion
Representation Learning for KG

- **Input**
  
a knowledge graph $\mathcal{KG} = \{(h, r, t)\}$

- **Target**
  
  To learn one embedding (a $k$-dimensional vector) for each entity: $h \to \vec{h}$ and $t \to \vec{t}$, where $\vec{h}, \vec{t} \in \mathbb{R}^k$

$h \quad \quad r \quad \quad t$

$(\text{Avatar, /film/film/directed_by, James Cameron})$

\[ f_r : \mathbb{R}^k \to \mathbb{R}^k \]
Translation-based Methods

- **TransE**
  - For each triple (head, relation, tail), treat *relation* as a **translation** from head to tail
  - Simple, effective, and achieving the state-of-the-art performance

Bordes, et al. (2013). Translating embeddings for modeling multi-relational data. NIPS.
Translation-based Methods

- **TransH and TransR**
  - Build relation-specific entity embeddings

![Diagram of TransH and TransR](image)

Motivation 1. Low performance on 1-to-N, N-to-1 and N-to-N relations

Wang, et al. (2014). Knowledge graph embedding by translating on hyperplanes. AAAI.
Lin, et al. (2015). Learning entity and relation embeddings for knowledge graph completion. AAAI.
Translation-based Methods

- Learn embeddings directly from the graph structure
  - Graph sparseness
  - In domain-specific and non-English situations

Motivation 2. Limited performance by the structure sparseness of KG
Our Idea

- Text-enhanced Representation Learning for KG
  - Go back to traditional relation extraction
  - Inspired by distant supervision

Triple: \( (\text{Avatar}, /\text{film}/\text{film}/\text{directed\_by}, \text{James Cameron}) \)

Context: \( \{\text{film, movie, directed, ...}\} \rightarrow \{\text{direct}\} \leftarrow \{\text{director, ...}\} \)

James Francis Cameron, the famous director of the movie Avatar, is an ...

Text: The fiction film Avatar directed by J. Cameron was nominated by ...
In 1994 director James Cameron wrote an 80-page treatment for Avatar

Contributions:

[Motivation 1]. Enable each relation to own different representations for different head and tail entities.
[Motivation 2]. Incorporate the textual contexts to each entity and relation.
Problem Definition

- **Input**
  - Knowledge Graph
    \[ KG = \{(h, r, t)\} \]
  - Text Corpus
    \[ D = \langle w_1 ... w_i ... w_m \rangle \]

- **Text-enhanced Knowledge Embedding (TEKE)**
  - Learn the entity embeddings \( h \rightarrow \hat{h} \in \mathbb{R}^k \) and \( t \rightarrow \hat{t} \in \mathbb{R}^k \) for each triple \((h, r, t)\) by utilizing the rich text information in \( D \) to deal with
    - low performance on 1-to-N, N-to-1, N-to-N relations
    - knowledge graph sparseness
  - Learn the relation embedding \( r \rightarrow \hat{r} \in \mathbb{R}^k \)
The Proposed Approach

Triple: \((\text{Avatar}, /\text{film/film/directed}_by, \text{James Cameron})\)

1. Entity Annotation
   - James Francis Cameron, the famous director of the movie Avatar, is an …
   - The fiction film Avatar directed by J. Cameron was nominated by …
   - In 1994 director James Cameron wrote an 80-page treatment for Avatar

2. Textual Context Embedding
   - \{film, movie, directed, …\} \rightarrow \{direct\} \leftarrow \{director, …\}

3. Entity/Relation Representation Modelling
   - Representation Training
     \(\mathbf{\hat{h}} + \mathbf{\hat{r}} = \mathbf{\hat{t}}\)

4. Representation Training
The Proposed Approach

- **Entity Annotation**
  - Given the text corpus \( D = \langle w_1 \ldots w_i \ldots w_m \rangle \), use an entity linking tool to automatically label the entities in \( KG \), and get an entity-annotated text corpus:

  \[
  D' = \langle X_1 \ldots X_i \ldots X_m' \rangle
  \]

- **Textual Context Embedding**
  - co-occurrence network \( \mathcal{G} = (X, Y) \)
    - \( x_i \in X \): denotes to the node (a word or an entity)
    - \( y_{ij} \in Y \): co-occurrence frequency between \( x_i \) and \( x_j \)
The Proposed Approach

Textual Context Embedding

- Pointwise textual context

\[ n(x_i) = \{x_j | y_{ij} > \theta \} \]
\[ n(Avatar) = \{film, movie, directed \ldots \} \]
\[ n(James\_Cameron) = \{director \ldots \} \]

- Pairwise textual context

\[ n(x_i, x_j) = \{x_k | x_k \in n(x_i) \cap n(x_j) \} \]
\[ n(Avatar, James\_Cameron) = \{direct \ldots \} \]

<table>
<thead>
<tr>
<th>Triple:</th>
<th>(Avatar, /film/film/directed_by, James Cameron)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Context:</td>
<td>{film, movie, directed, ...} \rightarrow {direct} \leftarrow {director, ...}</td>
</tr>
</tbody>
</table>

James Francis Cameron, the famous director of the movie Avatar, is an ... 

Text: The fiction film Avatar directed by J. Cameron was nominated by ... 

In 1994 director James Cameron wrote an 80-page treatment for Avatar
The Proposed Approach

- **Textual Context Embedding**
  - Word Embedding Learning $x_i \rightarrow x_i$
  - Pointwise textual context embedding of $x_i$:
    \[
    n(x_i) = \frac{1}{\sum_{x_j \in n(x_i)} y_{ij}} \sum_{x_j \in n(x_i)} y_{ij} \cdot x_j
    \]
  - Pairwise textual context embedding of $x_i$ and $x_j$:
    \[
    n(x_i, x_j) = \frac{1}{Z} \sum_{x_k \in n(x_i, x_j)} \min(y_{ik}, y_{jk}) \cdot x_k
    \]

Textual Context Embedding

- \{film, movie, directed, ...\} $\rightarrow$ \{direct\} $\leftarrow$ \{director, ...\}
The Proposed Approach

- **Entity/Relation Representation Modeling**
  - Incorporate the textual context information to the representation learning on knowledge graph

\[
\hat{h} = \mathbf{n}(h)A + \mathbf{h} \\
\hat{t} = \mathbf{n}(t)A + \mathbf{t} \\
\hat{r} = \mathbf{n}(h, t)B + \mathbf{r}
\]

- **Linear transformation of textual context information**
  - given a relation, different textual context embeddings for different pairs of (head, tail) entities
  - to better handle 1-to-N, N-to-1, and N-to-N relations

- **Incorporate textual context information into the KG**
  - more background information
  - to deal with knowledge graph sparseness

\[
f(h, r, t) = \|\hat{h} + \hat{r} - \hat{t}\|^2_2\]
The Proposed Approach

- Representation Training
  - Margin-based score function
  
  \[ L = \sum_{(h,r,t) \in S} \sum_{(h',r,t') \in S'} \max(0, f(h,r,t) + \gamma - f(h',r,t')) \]

  - Stochastic gradient descent (SGD)
Outline

Introduction

Problem Definition

Our Proposed Approach

Experiments and Analysis

Conclusion
Experiments and Analysis

- **Datasets**
  - 4 benchmark knowledge graphs
  - Entity-annotated Wikipedia corpuses

### Table 1: Statistics of the data sets.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>#Relations</th>
<th>#Entities</th>
<th>#Triples (Train/Valid/Test)</th>
</tr>
</thead>
<tbody>
<tr>
<td>WN18</td>
<td>18</td>
<td>40,943</td>
<td>141,442 / 5,000 / 5,000</td>
</tr>
<tr>
<td>FB15K</td>
<td>1,345</td>
<td>14,951</td>
<td>483,142 / 50,000 / 59,071</td>
</tr>
<tr>
<td>WN11</td>
<td>11</td>
<td>38,696</td>
<td>112,581 / 2,609 / 10,544</td>
</tr>
<tr>
<td>FB13</td>
<td>13</td>
<td>75,043</td>
<td>316,232 / 5,908 / 23,733</td>
</tr>
</tbody>
</table>

### Table 2: Statistics of entity-annotated Wikipedia corpuses.

<table>
<thead>
<tr>
<th>KG</th>
<th>#Entities</th>
<th>#Annotated Entities</th>
<th>#Word Stems</th>
</tr>
</thead>
<tbody>
<tr>
<td>WN18</td>
<td>40,943</td>
<td>32,249</td>
<td>1,529,251</td>
</tr>
<tr>
<td>FB15K</td>
<td>14,951</td>
<td>14,405</td>
<td>744,983</td>
</tr>
<tr>
<td>WN11</td>
<td>38,696</td>
<td>30,937</td>
<td>1,526,467</td>
</tr>
<tr>
<td>FB13</td>
<td>75,043</td>
<td>69,208</td>
<td>706,484</td>
</tr>
</tbody>
</table>
Experiments and Analysis

- Evaluation
  - (China, /location/location/adjoin, North_Korea)

- Link Prediction
  - Mean Rank: 11
  - Hits@10: 0%
  - Raw; Filter: 9; 100%

- Triple Classification
  - a binary classification task

<table>
<thead>
<tr>
<th>Head</th>
<th>Relation</th>
<th>China</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>/location</td>
<td>Japan</td>
</tr>
<tr>
<td>2</td>
<td>/location</td>
<td>Taiwan</td>
</tr>
<tr>
<td>3</td>
<td>/location</td>
<td>Israel</td>
</tr>
<tr>
<td>4</td>
<td>/location</td>
<td>South_Korea</td>
</tr>
<tr>
<td>5</td>
<td>/location</td>
<td>Argentina</td>
</tr>
<tr>
<td>6</td>
<td>/location</td>
<td>France</td>
</tr>
<tr>
<td>7</td>
<td>/location</td>
<td>Philippines</td>
</tr>
<tr>
<td>8</td>
<td>/location</td>
<td>Hungary</td>
</tr>
<tr>
<td>9</td>
<td>/location</td>
<td>Germany</td>
</tr>
<tr>
<td>10</td>
<td>/location</td>
<td>USA</td>
</tr>
<tr>
<td>11</td>
<td>/location</td>
<td>North_Korea</td>
</tr>
</tbody>
</table>
Link Prediction

- TEKE compare with baselines

A lower Mean Rank is better while a higher Hits@10 is better

Mean Rank
- TEKE methods perform much better than the baselines on WN18.
- No much improvement is observed on FB15K

Hits@10
- TEKE methods outperform other baselines significantly and consistently
Link Prediction

- **Capability to handle 1-to-N, N-to-1 and N-to-N relations**
  - FB15K: 1-1, 1-N, N-1, N-N → 24.2%, 22.9%, 28.9%, 24.0%

<table>
<thead>
<tr>
<th>Relation Category</th>
<th>Prediction Head (Hits@10)</th>
<th>Prediction Tail (Hits@10)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1-to-1</td>
<td>l-to-N</td>
</tr>
<tr>
<td>TransE/TEKE_E</td>
<td>43.7 / 52.3</td>
<td>18.2 / 76.8</td>
</tr>
<tr>
<td>TransH/TEKE_H unif</td>
<td>66.7 / 66.6</td>
<td>81.7 / 80.9</td>
</tr>
<tr>
<td>TransH/TEKE_H bern</td>
<td>66.8 / 69.3</td>
<td>87.6 / 90.8</td>
</tr>
<tr>
<td>TransR/TEKE_R unif</td>
<td>76.9 / 66.2</td>
<td>82.0 / 82.0</td>
</tr>
<tr>
<td>TransR/TEKE_R bern</td>
<td>78.8 / 70.1</td>
<td>89.2 / 89.3</td>
</tr>
</tbody>
</table>

- TEKE methods significantly outperform the baselines when predicting the entity where multiple entities could be correct.

- TEKE methods have not shown much advantage for predicting the entity where only one entity is correct.
Link Prediction

- Capability to handle knowledge graph sparseness

<table>
<thead>
<tr>
<th>Dataset</th>
<th>#E</th>
<th>#R</th>
<th>#T</th>
<th>#T/#E</th>
<th>#T/#R</th>
</tr>
</thead>
<tbody>
<tr>
<td>FB3K</td>
<td>3,000</td>
<td>613</td>
<td>19,339</td>
<td>6.45</td>
<td>31.55</td>
</tr>
<tr>
<td>FB6K</td>
<td>6,000</td>
<td>913</td>
<td>75,347</td>
<td>12.56</td>
<td>82.53</td>
</tr>
<tr>
<td>FB9K</td>
<td>9,000</td>
<td>1,094</td>
<td>167,191</td>
<td>18.58</td>
<td>152.83</td>
</tr>
</tbody>
</table>

$\mathcal{T}$ represents the training triples.

- Rank 3,000 entities for 2,238 triples for all three datasets

<table>
<thead>
<tr>
<th>Methods</th>
<th>TransE / TEKE_E</th>
</tr>
</thead>
<tbody>
<tr>
<td>Metric</td>
<td>Raw</td>
</tr>
<tr>
<td>FB3K</td>
<td>102.7</td>
</tr>
<tr>
<td>FB6K</td>
<td>81.9</td>
</tr>
<tr>
<td>FB9K</td>
<td>79.5</td>
</tr>
</tbody>
</table>

- As the graph density gets higher, both TransE and TEKE_E perform better.
- TEKE_E achieves the highest improvement on the sparsest FB3K dataset.
Triple Classification

- TEKE compare with baselines

Table 7: Evaluation results of triple classification. (%)

<table>
<thead>
<tr>
<th>Datasets</th>
<th>WN11</th>
<th>FB13</th>
</tr>
</thead>
<tbody>
<tr>
<td>TransE / TEKE_E unif</td>
<td>75.9 / 84.1</td>
<td>70.9 / 75.1</td>
</tr>
<tr>
<td>TransE / TEKE_E bern</td>
<td>75.9 / 84.5</td>
<td>81.5 / 82.1</td>
</tr>
<tr>
<td>TransH / TEKE_H unif</td>
<td>77.7 / 84.3</td>
<td>76.5 / 77.4</td>
</tr>
<tr>
<td>TransH / TEKE_H bern</td>
<td>78.8 / 84.8</td>
<td>83.3 / 84.2</td>
</tr>
<tr>
<td>TransR / TEKE_R unif</td>
<td>85.5 / 85.2</td>
<td>74.7 / 77.1</td>
</tr>
<tr>
<td>TransR / TEKE_R bern</td>
<td>85.9 / 86.1</td>
<td>82.5 / 81.6</td>
</tr>
</tbody>
</table>

- **TEKE_E** and **TEKE_H** consistently outperform the comparison methods, especially on WN11.
- **TEKE_R** (unif) on WN11 and **TEKE_R** (bern) on FB13 perform better than TransR, while others perform a bit worse.
Outline

Introduction

Problem Definition

Our Proposed Approach

Experiments and Analysis

Conclusion
Conclusion and Future Work

- A novel text-enhanced knowledge embedding method named **TEKE** for knowledge graph representation learning to deal with
  - Low performance on 1-to-N, N-to-1 and N-to-N relations
  - Limited performance by structure sparseness of KG

Future Work

- Improve performance on 1-to-1 relations
- Experimentally analyze the influence of entity annotation
- Use different text corpus
- Incorporate knowledge reasoning
- ...
Thanks!

Zhigang WANG

wangzigo@gmail.com

http://xlore.org/
TransE

- **Idea**

\[
\vec{\text{Paris}} - \vec{\text{France}} \cong \vec{\text{Rome}} - \vec{\text{Italy}}
\]

- Treat each relation as one unique vector

\[
\vec{\text{has\_capital}} = \vec{\text{has\_capital}}
\]

- and it would be reasonable that

\[
\vec{\text{Paris}} - \vec{\text{France}} \cong \vec{\text{Rome}} - \vec{\text{Italy}} = \vec{\text{capital\_of}}
\]

Assumption \( \hat{t} - \hat{h} = \hat{r} \)