Trans* and DeepWalk

by Ran Jia (22 Apr 2016)
and Lili Mou (additional reading)
Trans*

by Ran Jia

Relational fact: (head, relation, tail)
TransE:

TransH:

TransR:
TransE

- $h + r \approx t$ when $(h, r, t)$ holds

- score function:
  $$f_r(h, t) = \|h + r - t\|^2_2$$

- 1-to-1 ✓
  1-to-N/N-to-1/N-to-N ×
TransH

- A triple \((h, r, t)\), a hyperplane with normal vector \(\mathbf{v}\),
- Project \(h\) and \(t\) to the hyperplane:
- Score function:
TransR

- $h, t \in R^k \quad r \in R^d \quad k \neq d$

- $M_r \in R^{k \times d}$

- Project entities from entity space to relation space:
  \[ h_r = hM_r \quad t_r = tM_r \]

- Score function:
  \[ f_r(h, t) = \| h_r + r - t_r \|^2 \]

- $\|h\|_2 \leq 1, \|t\|_2 \leq 1, \|r\|_2 \leq 1, \|hM_r\|_2 \leq 1, \|tM_r\|_2 \leq 1$
CTransR: Cluster-based TransR

- TransE/TransH/TransR learn a unique vector for each relation, which may be under-representative to fit all entity pairs under this relation, because these relations are usually rather diverse.

- For a relation r, all entity pairs (h,t) in the training data are clustered into multiple groups, entity pairs in each group are expected to exhibit similar r relation.

- Entity pairs are represented with their vector offsets (h-t), h and t are obtained with TransE.
CTransR: Cluster-based TransR

- For each relation, we learn a separate relation vector $r_c$ for each cluster and $M_r$.

- $h_{r,c} = hM_r \quad t_{r,c} = tM_r$

- Score function:

$$f_r(h, t) = \|h_{r,c} + r_c - t_{r,c}\|^2_2 + \alpha \|r_c - r\|^2_2$$

差别不能太大
Training Objective

\[ L = \sum_{(h,r,t) \in S} \sum_{(h',r,t') \in S'} \max(0, f_r(h, t) + \gamma - f_r(h', t')) \]

\( \gamma \) is the margin

S is the set of correct triples and

S’ is the set of incorrect triples
Data Sets

- WordNet and Freebase
- Freebase
  - (Steve jobs. Founded, Apple Inc.)
- FB15K, FB13
- WordNet
  - Semantic knowledge of words
  - Relations between synsets
- WN11, WN18

<table>
<thead>
<tr>
<th>Dataset</th>
<th>#Rel</th>
<th>#Ent</th>
<th>#Train</th>
<th>#Valid</th>
<th>#Test</th>
</tr>
</thead>
<tbody>
<tr>
<td>WN18</td>
<td>18</td>
<td>40,943</td>
<td>141,442</td>
<td>5,000</td>
<td>5,000</td>
</tr>
<tr>
<td>FB15K</td>
<td>1,345</td>
<td>14,951</td>
<td>483,142</td>
<td>50,000</td>
<td>59,071</td>
</tr>
<tr>
<td>WN11</td>
<td>11</td>
<td>38,696</td>
<td>112,581</td>
<td>2,609</td>
<td>10,544</td>
</tr>
<tr>
<td>FB13</td>
<td>13</td>
<td>75,043</td>
<td>316,232</td>
<td>5,908</td>
<td>23,733</td>
</tr>
<tr>
<td>FB40K</td>
<td>1,336</td>
<td>39528</td>
<td>370,648</td>
<td>67,946</td>
<td>96,678</td>
</tr>
</tbody>
</table>
Experiment Settings

◆ Link Prediction
◆ Triple Classification
◆ Relation Extraction from Text
Link Prediction

◆ For each triple \((h, r, t)\), replace the head/tail entity by all entities in the KG.

◆ Rank these entities in **descending** (ascending?) order of scores

◆ Evaluation Metric:
  (1) mean rank of correct entities
  (2) proportion of correct entities in top-10 ranked entities (Hits@10)

◆ A good link predictor should achieve lower mean rank or higher Hits@10
<table>
<thead>
<tr>
<th>Data Sets</th>
<th>Metric</th>
<th>WN18</th>
<th>FB15K</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Mean Rank</td>
<td>Hits@10 (%)</td>
</tr>
<tr>
<td></td>
<td>Raw</td>
<td>Filter</td>
<td>Raw</td>
</tr>
<tr>
<td>Unstructured (Bordes et al. 2012)</td>
<td>315</td>
<td>304</td>
<td>35.3</td>
</tr>
<tr>
<td>RESCAL (Nickel, Tresp, and Kriegel 2011)</td>
<td>1,180</td>
<td>1,163</td>
<td>37.2</td>
</tr>
<tr>
<td>SE (Bordes et al. 2011)</td>
<td>1,011</td>
<td>985</td>
<td>68.5</td>
</tr>
<tr>
<td>SME (linear) (Bordes et al. 2012)</td>
<td>545</td>
<td>533</td>
<td>65.1</td>
</tr>
<tr>
<td>SME (bilinear) (Bordes et al. 2012)</td>
<td>526</td>
<td>509</td>
<td>54.7</td>
</tr>
<tr>
<td>LFM (Jenatton et al. 2012)</td>
<td>469</td>
<td>456</td>
<td>71.4</td>
</tr>
<tr>
<td>TransE (Bordes et al. 2013)</td>
<td>263</td>
<td>251</td>
<td>75.4</td>
</tr>
<tr>
<td>TransH (unif) (Wang et al. 2014)</td>
<td>318</td>
<td>303</td>
<td>75.4</td>
</tr>
<tr>
<td>TransH (bern) (Wang et al. 2014)</td>
<td>401</td>
<td>388</td>
<td>73.0</td>
</tr>
<tr>
<td>TransR (unif)</td>
<td>232</td>
<td>219</td>
<td>78.3</td>
</tr>
<tr>
<td>TransR (bern)</td>
<td>238</td>
<td>225</td>
<td>79.8</td>
</tr>
<tr>
<td>CTransR (unif)</td>
<td>243</td>
<td>230</td>
<td>78.9</td>
</tr>
<tr>
<td>CTransR (bern)</td>
<td>231</td>
<td>218</td>
<td>79.4</td>
</tr>
</tbody>
</table>
Triple Classification

◆ A binary classification: judge whether a given triple \((h, r, t)\) is correct or not.

◆ A relation-specific threshold \(\delta_r\). For a triple \((h,r,t)\), is \(f_r\) is below \(\delta_r\), the triple will be classified as positive.

◆ \(\delta_r\) is optimized by maximizing classification accuracies on the validation set.
<table>
<thead>
<tr>
<th>Data Sets</th>
<th>WN11</th>
<th>FB13</th>
<th>FB15K</th>
</tr>
</thead>
<tbody>
<tr>
<td>SE</td>
<td>53.0</td>
<td>75.2</td>
<td>-</td>
</tr>
<tr>
<td>SME (bilinear)</td>
<td>70.0</td>
<td>63.7</td>
<td>-</td>
</tr>
<tr>
<td>SLM</td>
<td>69.9</td>
<td>85.3</td>
<td>-</td>
</tr>
<tr>
<td>LFM</td>
<td>73.8</td>
<td>84.3</td>
<td>-</td>
</tr>
<tr>
<td>NTN</td>
<td>70.4</td>
<td>87.1</td>
<td>68.5</td>
</tr>
<tr>
<td>TransE (unif)</td>
<td>75.9</td>
<td>70.1</td>
<td>79.6</td>
</tr>
<tr>
<td>TransE (bern)</td>
<td>75.9</td>
<td>81.5</td>
<td>79.2</td>
</tr>
<tr>
<td>TransH (unif)</td>
<td>77.7</td>
<td>76.5</td>
<td>79.0</td>
</tr>
<tr>
<td>TransH (bern)</td>
<td>78.8</td>
<td>83.3</td>
<td>80.2</td>
</tr>
<tr>
<td>TransR (unif)</td>
<td>85.5</td>
<td>74.7</td>
<td>81.7</td>
</tr>
<tr>
<td>TransR (bern)</td>
<td><strong>85.9</strong></td>
<td>82.5</td>
<td>83.9</td>
</tr>
<tr>
<td>CTransR (bern)</td>
<td>85.7</td>
<td>-</td>
<td><strong>84.5</strong></td>
</tr>
</tbody>
</table>
Relation Extraction from Text

- Combine the scores from text-based relation extraction model with the scores from knowledge graph embeddings to rank test triples.

- Plot a precision-recall curves.
Deep Walk
DeepWalk: Online Learning of Social Representations

Bryan Perozzi  
Stony Brook University  
Department of Computer Science

Rami Al-Rfou  
Stony Brook University  
Department of Computer Science

Steven Skiena  
Stony Brook University  
Department of Computer Science

{bperozzi, ralrfou, skiena}@cs.stonybrook.edu

- A graph (e.g., social networks) defines vertexes (analogous to word in word2vec) and edges (analogous to word co-occurrences)
- The basic idea: Train a “LM” over graph nodes
Step I: A small random walk generates context as a “sentence.”

Note: The random walk is not actually necessary because we can alternatively define context as adjacent nodes. But as some critics point out, random walk increases randomness and is tunable by its weights.

Step II: Standard word2vec (either hierarchical softmax or SkipGram)
An Interesting Phenomenon

- The connections of nodes in social networks, say, typically follow a power law distribution. So does word co-occurrences.