Global RDF Vector Space Embeddings
HELLO!

I am Michael Cochez
Fraunhofer FIT and RWTH Aachen University
Aachen, Germany
You can reach me at
michael.cochez@fit.fraunhofer.de
Jointly with

Petar Ristoski
Simone Paolo Ponzetto
Heiko Paulheim
Knowledge graphs are all around us.

- LOD cloud
- Freebase - Wikidata
- DBPedia
- YAGO
- OpenCyc
- Thomson Reuters
- Cyc
- Microsoft Satori
- Yahoo KG
- Google KG
- IBM Watson
- Amazon Product Graph
- Facebook Social Graph
- Springer, ...
What if Knowledge Graphs could be used for Machine Learning and Data Mining?
So, what if Knowledge Graphs could be used for Machine Learning ...

- Classification, Regression, Clustering
- Recommender systems
- Document modeling
  - Entity and Document similarity
  - Entity relatedness
- Alignment of knowledge bases
  - DBpedia and Wikidata
- Link prediction and error detection
- Linking text and semi-structured knowledge to knowledge bases
So, what if Knowledge Graphs could be used for Machine Learning ...
Challenges
Model Mismatch
Our Solution
Embedding Knowledge Graphs in Vector Spaces
Embedding Goals

- One vector for each entity
  - Compatible with traditional data mining algorithms and tools
- Preserve the information
- Unsupervised
  - Task and dataset independent
- Efficient computation
- Low dimensional representation
How?
How?

- Weigh Graph
- Create co-occurrence Matrix
- Train GloVe model
1. Weigh Graph
1. Weigh Graph

Diagram:
- German
- Austria
- Vienna
- Germany
- Berlin
- Language
- Capital
- Type

Relations:
- German to Austria
- German to Vienna
- Austria to Vienna
- Austria to Capital
- Capital to Vienna
- Language to German
- Language to Austria
- Language to Vienna
- Language to Germany
- Language to Berlin
- Type to German
- Type to Austria
- Type to Germany
- Type to Berlin
- In to Vienna
2. Create Co-occurrence Matrix

<table>
<thead>
<tr>
<th>Entities</th>
<th>Austria</th>
<th>Vienna</th>
<th>Germany</th>
<th>Berlin</th>
<th>German</th>
<th>Country</th>
</tr>
</thead>
<tbody>
<tr>
<td>Austria</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Vienna</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Germany</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Berlin</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>German</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Country</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Capital</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Language</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Type</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
2. Create Co-occurrence Matrix (Personalized PageRank)
### 2. Create Co-occurrence Matrix (Personalized PageRank)

<table>
<thead>
<tr>
<th></th>
<th>Austria</th>
<th>Vienna</th>
<th>Germany</th>
<th>Berlin</th>
<th>German Type</th>
<th>Country</th>
<th>Language</th>
<th>Capital</th>
</tr>
</thead>
<tbody>
<tr>
<td>Austria</td>
<td>0</td>
<td>0.174</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Vienna</td>
<td>0</td>
<td>0</td>
<td>0.217</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Germany</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Berlin</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>German</td>
<td>0.152</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Country</td>
<td>0.087</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Capital</td>
<td>0.13</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Language</td>
<td>0.152</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Type</td>
<td>0.087</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
### Create Co-occurrence Matrix (Reversed edges)

<table>
<thead>
<tr>
<th>Table Entry</th>
<th>Vienna</th>
<th>German</th>
<th>Country</th>
</tr>
</thead>
<tbody>
<tr>
<td>Austria</td>
<td>0.174</td>
<td>0.000</td>
<td>0.000</td>
</tr>
<tr>
<td>Vienna</td>
<td>0.217</td>
<td>0.000</td>
<td>0.000</td>
</tr>
<tr>
<td>Germany</td>
<td>0.152</td>
<td>0.000</td>
<td>0.000</td>
</tr>
<tr>
<td>Berlin</td>
<td>0.130</td>
<td>0.000</td>
<td>0.000</td>
</tr>
<tr>
<td>Language</td>
<td>0.152</td>
<td>0.000</td>
<td>0.000</td>
</tr>
<tr>
<td>Type</td>
<td>0.087</td>
<td>0.000</td>
<td>0.000</td>
</tr>
</tbody>
</table>

#### Diagram:

- **Nodes:** Austria, Vienna, Germany, Berlin, German, Country, Language, Type
- **Edges:**
  - Austria → Vienna: 0.174 + 0.308
  - Vienna → Germany: 0.000 + 0.000
  - Germany → Berlin: 0.000 + 0.000
  - Country → Type: 0.087 + 0.000
  - Language → Country: 0.152 + 0.000
  - Capital → In: 0.130 + 0.308
  - Language → Type: 0.152 + 0.000
  - Type → Language: 0.087 + 0.000

The diagram illustrates the co-occurrence relationships between the terms, with edge weights indicating the strength of the relationship.
2. Create Co-occurrence Matrix (Reversed edges)

<table>
<thead>
<tr>
<th></th>
<th>Austria</th>
<th>Vienna</th>
<th>Germany</th>
<th>Berlin</th>
<th>German</th>
<th>Country</th>
<th>Capital</th>
</tr>
</thead>
<tbody>
<tr>
<td>Austria</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Vienna</td>
<td>0.482</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.152</td>
<td>0.087</td>
<td>0.438</td>
</tr>
<tr>
<td>Germany</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Berlin</td>
<td>0</td>
<td>0</td>
<td></td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>German</td>
<td>0.152</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Country</td>
<td>0.087</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Capital</td>
<td>0.438</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Language</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.152</td>
<td>0.087</td>
<td>0.438</td>
</tr>
<tr>
<td>Type</td>
<td>0.087</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
## 2. Create Co-occurrence Matrix (Normalization)

<table>
<thead>
<tr>
<th></th>
<th>Austria</th>
<th>Vienna</th>
<th>Germany</th>
<th>Berlin</th>
<th>German</th>
<th>Country</th>
<th>Language</th>
<th>Capital</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Austria</td>
<td>0</td>
<td>0.482</td>
<td>0</td>
<td>0</td>
<td>0.152</td>
<td>0.087</td>
<td>0</td>
<td>0.438</td>
<td>0.087</td>
</tr>
<tr>
<td>Vienna</td>
<td>0.602</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Germany</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Berlin</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>German</td>
<td>0.152</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Country</td>
<td>0.087</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Language</td>
<td>0.152</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Capital</td>
<td>0.438</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Type</td>
<td>0.087</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
2. Create Co-occurrence Matrix (Normalization)

<table>
<thead>
<tr>
<th></th>
<th>Austria</th>
<th>Vienna</th>
<th>Germany</th>
<th>Berlin</th>
<th>German</th>
<th>Country</th>
<th>Capital</th>
<th>Language</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Austria</td>
<td>0</td>
<td>0.308</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.135</td>
<td>0</td>
<td>0.077</td>
<td>0.135</td>
</tr>
<tr>
<td>Vienna</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.077</td>
<td>0</td>
<td>0.077</td>
<td>0.077</td>
</tr>
<tr>
<td>Germany</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.135</td>
<td>0</td>
<td>0.077</td>
<td>0.077</td>
</tr>
<tr>
<td>Berlin</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.077</td>
<td>0</td>
<td>0.077</td>
<td>0.077</td>
</tr>
<tr>
<td>German</td>
<td>0.308</td>
<td>0</td>
<td>0.135</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.269</td>
<td>0.135</td>
<td>0.077</td>
</tr>
<tr>
<td>Country</td>
<td>0.135</td>
<td>0</td>
<td>0.135</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.077</td>
<td>0.077</td>
<td>0.077</td>
</tr>
<tr>
<td>Capital</td>
<td>0.077</td>
<td>0</td>
<td>0.077</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.269</td>
<td>0.135</td>
<td>0.077</td>
</tr>
<tr>
<td>Language</td>
<td>0.077</td>
<td>0</td>
<td>0.077</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.269</td>
<td>0.135</td>
<td>0.077</td>
</tr>
</tbody>
</table>

\[ \sum = 1 \]
2. Create Co-occurrence Matrix (Optimization)

- Computing PPR for all nodes is slow
- Bookmark Coloring Algorithm (BCA)
- Fast approximation of PPR
- Further optimized for all nodes case
- Determine good order to increase reuse of earlier PPR computation
- Fast approximation all pairs Personalized PageRank
3. Train a GloVe Model

\[ J = \sum_{i,j=1}^{V} f(X_{ij}) \left( w_i^T \tilde{w}_j + b_i + \tilde{b}_j - \log X_{ij} \right)^2 \]
3. Train a GloVe Model

- Similar things will have similar contexts

- Optimizes for analogies
  - King - Man + Woman ≃ Queen
  - Berlin - Germany + Austria ≃ Vienna

- Complete context captured
Evaluation Setup

- Embedded Dataset: DBPedia (English)
  - 4,678,230 instances
  - 1,379 mapping-based properties
  - 12 weighing strategies

- Other methods
  - Baselines
  - TransE/H/R
  - Best RDF2Vec
    - (depth 8 Skipgram - 500 walks and 500 size)
  - RDF2VecGloVe
Embedding time

× Global Embedding
  × 6-48h depending on weighing method
× Other methods
  × RDF2Vec : 1 day
  × TransX : 1 week
× Many methods did not scale enough to compare
Machine learning tasks
- 5 evaluation datasets
- Classification: NB, K-NN (k=3), C4.5 and SVM
- Regression: LR, M5Rules, and K-NN (k=3).

Document Modelling - document similarity
- The LP50 dataset [Lee et al. 2005]
- 7 standard document modeling baselines
- Approach: average maximum link similarity (each document is represented as a set of entities)
Results Summary

Classification and Regression
- RDF2Vec is a strong competitor, but also ours outperforms other baselines
- For some cases similar performance is obtained

Document modelling
- Inverse predicate frequency weighting dominates all baselines
Conclusion

- An approach for generating embeddings of RDF graphs, which exploits global instead of local patterns
- Preserves the graph information
- Compatible with all the traditional machine learning algorithms
- More efficient embedding learning
- No “one-size-fits-all” embeddings
- Task and dataset independent approach
Steps Forward

- Need for better evaluation metrics
- Do vectors contain meaning?
- Do vectors contain King-Man+Woman=Queen
- Other properties
- No “one-size-fits-all” embeddings
- Need for a large scale evaluation
- Include literals in training
- Embed all sorts of graphs
THANKS!

Any questions?

You can find me at

- [http://users.jyu.fi/~miselico](http://users.jyu.fi/~miselico)
- [michael.cochez@fit.fraunhofer.de](mailto:michael.cochez@fit.fraunhofer.de)
- [https://github.com/miselico/globalRDFEmbeddingsISWC](https://github.com/miselico/globalRDFEmbeddingsISWC)
Credits

Special thanks to the many people with whom I had inspiring discussions on embeddings, especially the researchers at Mannheim University.

Besides, we used the following resources:

- Presentation template by SlidesCarnival
- Photographs by Unsplash