Combining Text Embedding and Knowledge Graph Embedding Techniques for Academic Search Engines

SemDeep-4, Oct. 2018

Gengchen Mai    Krzysztof Janowicz    Bo Yan

STKO Lab, University of California, Santa Barbara
The past decades have witnessed a rapid increase in the global scientific output as measured by publish papers. Exploring a scientific field and searching for relevant papers and authors seems like a needle-in-a-haystack problem.
Several academic search engines have been established to facilitate this process such as Google Scholar, Microsoft Academic Search, Semantic Scholar, DBLP, and so forth.

They provide paper-level (and sometimes author-level) recommendations based on: textual content, authors, publication year, and citation information.
Score question: how to define and measure similarity and relatedness among research papers, authors, potential funding sources, and so forth.

Conventional way: using feature engineering which extracts features from textual content, citation networks, and co-author networks.
### Introduction

- Semantic Web technologies play an increasing role in the field of academic publishing for easing publishing, retrieving, interlinking, and integrating datasets across outlets and publishers.
  - Springer Nature SciGraph
  - DBLP SPAQRL endpoint
  - IOS Press LD Connect

The availability of these bibliography knowledge graphs makes it possible to bring entity retrieval and content-based paper recommendations together.
Our contribution

- We present an entity retrieval prototype on top of IOS LD Connect which utilizes both textual information and structure information.
  - An entity retrieval system based on paragraph vectors and knowledge graph embeddings.
  - A paper similarity benchmark dataset from Semantic Scholar which is used to empirically evaluate the learned embedding models.
  - Another benchmark dataset from DBLP is constructed and used to evaluate the performance of the learned knowledge graph embedding model.
This knowledge graph encodes the information about all the papers published by IOS Press until now.

All metadata about papers are serialized and published as Linked Data by following the bibliographic ontology.

■ a SPARQL endpoint:  
http://ld.iospress.nl:3030

■ a dereference interface:  

**Table:** An overview of LD Connect as of 05/2018

<table>
<thead>
<tr>
<th>Class Name</th>
<th># of Instances</th>
</tr>
</thead>
<tbody>
<tr>
<td>prov:Publisher</td>
<td>1</td>
</tr>
<tr>
<td>bibo:Journal</td>
<td>125</td>
</tr>
<tr>
<td>bibo:Series</td>
<td>41</td>
</tr>
<tr>
<td>bibo:Periodical</td>
<td>2255</td>
</tr>
<tr>
<td>bibo:Issue</td>
<td>8891</td>
</tr>
<tr>
<td>bibo:Chapter</td>
<td>46915</td>
</tr>
<tr>
<td>bibo:AcademicArticle</td>
<td>80891</td>
</tr>
<tr>
<td>foaf:Person</td>
<td>385272</td>
</tr>
<tr>
<td>foaf:Organization</td>
<td>168360</td>
</tr>
<tr>
<td>rdf:Seq</td>
<td>109309</td>
</tr>
</tbody>
</table>
Distributed Bag of Words version of Paragraph Vector (PV-DBOW), is used to encode all textual information of each paper into low dimensional vectors.

PV-DBOW aims to maximize the average log probability of predicting a word given the paper.

The learned vectors preserve the semantics of the text.
**Textual Embedding**

- PV-DBOW calculates average log probability for a sequence of training words $w_1, w_2, \ldots, w_T$ in paper $pg_i$.

\[
\frac{1}{T} \sum_{t=1}^{T} \log p(w_t|pg_i) \tag{1}
\]

- The prediction is done by means of a softmax classifier shown in Equation 2.

\[
p(w_t|pg_i) = \frac{\exp(y_{w_t})}{\sum_j \exp(y_j)} \tag{2}
\]
Textual Embedding

■ PV-DBOW assumed that cosine similarity between two paragraph vectors represents the semantic similarity between the corresponding texts.

■ all 117,835 PDF documents are parsed and mapped to entities in the knowledge graph.

■ After some text preprocessing steps such as tokenization and lemmatization, the preprocessed texts of each paper are fed into PV-DBOW model.
Structure Embedding

- An **entity retrieval system** for a bibliographic dataset should go beyond simple similar paper search.
  - finding similar researchers
  - searching similar organizations
  - reviewer recommendations

**Challenge:** The symbolic representations of KGs prohibit the usage of probabilistic models which are widely used in many kinds of ML applications.

**Core problem:** how to *transform* the components of these heterogeneous networks into numerical representations such that they can be easily utilized in an entity retrieval system.
Structure Embedding

- **KG Embedding**: learning distributional representations for components of a KG while preserving the inherent structure of the original KG.
  - *Translation-based models* (e.g. TransE, TransH, and TransR)
  - *Semantic matching models* (e.g. RESCAL, HoIIE, and DisMult).
Structure Embedding

- Given a knowledge graph $G$ which contains a collection of triples/statements $(h_i, r_i, t_i)$
- TransE embeds the entities and relations in a KG into the same low-dimensional space
- TransE treats each relation $r_i$ as a transformation operation from the head entity $h_i$ to the tail entity $r_i$.
- A plausibility scoring function $d(h_i, r_i, t_i)$ is defined on each triple which measures the accuracy of the translation operation:

$$d(h_i, r_i, t_i) = || h_i + r_i - t_i ||$$  \hspace{1cm} (3)
Structure Embedding

- A margin-based loss function $\mathcal{L}$ is defined to set up an optimization problem

$$\mathcal{L} = \sum_{(h_i, r_i, t_i) \in G^+} \sum_{(h'_i, r'_i, t'_i) \in G^-} \left[ \gamma + d(h_i, r_i, t_i) - d(h'_i, r'_i, t'_i) \right]_+$$

- TransE has been applied to the entire LD Connect graph to learn the embeddings for all entities and relations.
We choose **TransE**:  
- Efficient to run on a large knowledge graph;  
- A very intuitive geometric interpretation;  
- TransE embeds all entities and relations in the same low-dimensional vector space which is important for property path reasoning.
Paper similarity search interface

A similar paper search interface based on the learned PV-DBOW model.

Figure: Paper similarity search interface

1http://stko-testing.geog.ucsb.edu:3000/ios/qe/paper
**Entity similarity search interface**

- An entity similarity search interface\(^2\) is developed based on the TransE model for searching different types of entities like papers, authors, journals, and organizations.

---

**Figure:** Entity similarity search interface

\(^2\)http://stko-testing.geog.ucsb.edu:3000/ios/qe/entity
<table>
<thead>
<tr>
<th>Paper Similarity Evaluation</th>
</tr>
</thead>
</table>

- **Similar paper binary classification task:** Given a paper \( q_i \) as the query paper and \( K \) papers \( d_k \) where \( k \in 1, 2, \ldots, K \) within the IOS Press corpus, we classify each pair \( (q_i, d_k) \) for \( k \in 1, 2, \ldots, K \) as similar or dissimilar.

- **Features:** Combine textual and structure embeddings for a similar paper search task.
Establish a paper similarity benchmark dataset:

- Use the title of all paper (106705) in the IOS Press corpus to search for the top 500 similar papers in Semantic Scholar;
- Co-reference papers in the search results to the papers in IOS Press document corpus by the DOIs and the titles and treat them as positive samples;
- The same number of papers are randomly selected from the rest of the corpus and labeled as negative samples.
Paper Similarity Evaluation

- 33871 paper search results left and on average 4.96 relevant papers for each search paper.
- Given a query paper \(q_i\) and a list of papers \(d_k\) (\(k \in 1, 2, \ldots, 2K\)) where \(d_1, d_2, \ldots, d_K\) are positive samples and \(d_{K+1}, d_{K+2}, \ldots, d_{2K}\) are negative samples:
  - Cosine similarity \(PV_{ik}\) between the textual embeddings of \(q_i\) and \(d_k\)
  - Cosine similarity \(KG_{ik}\) between the structure embeddings of \(q_i\) and \(d_k\)
  - Train a logistic regression model based on \(PV_{ik}\) and \(KG_{ik}\) and compare with the baseline models which use only one feature \(PV_{ik}\) or \(KG_{ik}\) in the logistic regression

<table>
<thead>
<tr>
<th></th>
<th>Precision</th>
<th>Recall</th>
<th>F1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Combined Model</td>
<td>0.8790</td>
<td>0.8372</td>
<td>0.8576</td>
</tr>
<tr>
<td>PV-DBOW</td>
<td>0.8770</td>
<td>0.8345</td>
<td>0.8552</td>
</tr>
<tr>
<td>TransE</td>
<td>0.6747</td>
<td>0.6817</td>
<td>0.6782</td>
</tr>
</tbody>
</table>
**Co-author Inference Evaluation**

- Is TransE model seem useless?
- Node A, B, C, and D refer to four authors in LD Connect and DBLP.
- The links between nodes represent the co-author relationship.
- **Hypothesis**: a similarity search on the trained TransE model for author A will likely also yield author D even though their co-author relationship is missing in IOS Press LD Connect

**Figure**: An illustration of co-author inference evaluation
Co-author Inference Evaluation

Build a co-author dataset from DBLP:

- Randomly select 10,000 authors from LD Connect corpus;
- Based on the TransE embeddings, for each selected author \( p_i \), obtain the top 10 similar authors \( p_{ik} \) where \( k \in 1, 2, \ldots, 10 \) who have not co-authored any paper with \( p_i \) according to LD Connect;
- For each pair of authors \( (p_i, p_{ik}) \), search for # of co-authored papers they have in DBLP KG which forms author pair dataset \( C \);
- For each selected author \( p_i \), randomly select 10 authors \( p'_{ik} \) where \( k \in 1, 2, \ldots, 10 \) from the conflated LD Connect;
- For each pair of authors \( (p_i, p'_{ik}) \), search for # of their co-authored papers in DBLP KG which forms author pair dataset \( C' \);
- Compute the ratio of co-author relationship for these person pairs in \( C \) and \( C' \) and compare them.
Co-author Inference Evaluation

Result:

- 5.511 percent of author pairs in C which have co-author relationships in DBLP KG.
- Only 1.537 percent for the randomly selected author pair dataset C'.
- This validates our assumption that the TransE model can help predict the missing co-author relationship between authors based on the observed graph structure.
We presented an entity retrieval system utilizing LD Connect based on textual embedding and structure embedding techniques.

The retrieval model is evaluated by two benchmark datasets collected from Semantic Scholar and DBLP.

TransE does not have a huge impact on improving the performance of paper similarity classification.

TransE is able to do co-author inference based on the observed triples in a bibliographic dataset.
Future Work

- More advanced sequence models like LSTM can be used instead of PV-DBOW to capture richer information from text content.
- Build a joint learning model which will help both of the embedding learning processes.
- Instead of using a generic knowledge graph embedding model such as TransE, explore ways to build a structure embedding model which specifically focuses on bibliographic knowledge graphs.