Pairwise Tensor Factorization for learning new facts in Knowledge Bases

by

Tanmoy Mukherjee, Vinay Pande, Vasudeva Varma

in

The 19th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)

Chicago

Report No: IIFT/TR/2013/-1

Centre for Search and Information Extraction Lab
International Institute of Information Technology
Hyderabad - 500 032, INDIA
August 2013
Pairwise Tensor Factorization for learning new facts in Knowledge Bases

Tanmoy Mukherjee
IIIT Hyderabad
Hyderabad, India
mukherjee.tanmoy@gmail.com

Vinay Pande
IIIT Hyderabad
Hyderabad, India
vinay.pande@research.iiit.ac.in

Vasudeva Varma
IIIT Hyderabad
Hyderabad, India
vv@iiit.ac.in

ABSTRACT
Knowledge bases provide with the benefit of organizing knowledge in the relational form but suffer from incompleteness of new entities and relationships. Prior work on relation extraction has been focused on supervised learning techniques which are quite expensive. An alternative approach based on distant supervision has been of significant interest where one aligns database records with sentences of these records. A new line of work on embeddings of symbolic representations [2] has shown promise. We introduce a Matrix tri-factorization model which can find missing information in knowledge bases. Experiments show that we are able to query and find missing information from text and shows improvement over existing methods.

Categories and Subject Descriptors
H.4 [Information Systems Applications]: Miscellaneous; D.2.8 [Software Engineering]: Metrics—complexity measures, performance measures

General Terms
Theory

Keywords
ACM proceedings, \LaTeX, text tagging

1. INTRODUCTION
Statistical relational learning analyzes the interaction between entities and their relationships and is an growing area in machine learning. Recent years has seen an increasing interest in analysis pairwise relational data that contains several entity types and multiple relations.

A challenge for AI systems has been to gather, organize and make use of massive amount of collected information. There has been a recent interest in building large scale Knowledge Bases (KB) which are multi-relational graph data whose nodes represent entities and edges corresponds to relations. Multi-Relational data plays a major role in areas such as recommendation systems, computational biology, social networks and has progressed into statistical relational learning [9]. Manually created knowledge bases often lack information about entities and their properties, either due to missing information from the source used to create the knowledge base or because human annotators were unable to add facts due to imprecise knowledge. There has been a great deal of recent research focus on extracting knowledge from text [1, 8, 17] but little work has been done on a problem of drawing inference from this imperfect extracted knowledge.

For example, given the facts PlayForTeam(Lionel Messi, Barcelona) and TeamPlaysInLeague(Barcelona, La Liga) in knowledge base, can we infer and add new fact PlayerPlaysInLeague(Lionel Messi, La Liga) into the knowledge base. While existing systems like NELL[3] try to infer such facts by mining data from their knowledge bases and using limited set of rules, adding new fact takes much efforts in terms of creating new inference rules. While traditional logical inference techniques are too brittle to make inference, probabilistic inference techniques [16] suffer from scalability issues. In this paper we introduce a model that can accurately learn to add additional facts to a database using only that database. We represent each entity by a low dimensional representation which captures local and global semantics [11].

Our model is flexible and allows us to query and find if certain entities that were not originally in the knowledge base are in certain relationships by exploring distributional word vectors. These vectors are learned by training a neural network model [11] using wikipedia and uses local and global properties. The word vectors thus are able to capture semantic and syntactic properties and allows to extend the database without parsing of any additional textual resources.

2. RELATED WORK
There is a vast amount of work done in extending knowledge bases using external corpora [8, 17]. However most techniques use a predefined, finite and fixed schema of relation types. Some text is labeled according to this schema, and this labeling is then used in supervised training for automatic extracting of relations. There has been an increased interest in learning with weak supervision where one aligns a database record with a sentence in which these records appear, effectively labeling the text which is further used to train a machine learning system [7, 13]. However this
method relies on the availability of an existing database with a fixed schema. There has been minimal work done in drawing reliable inference in noisy knowledge bases. The closest work has been in using path ranking algorithm [12] where the authors showed that a soft inference procedure based on combination of constrained, weighted random walk through a knowledge base is able to reliably infer new facts or beliefs.

3. TENSOR FACTORIZATION MODEL

Factorization models for tensors are studied in several fields. The most general model is the Tucker Decomposition [19] which has been studied in problems in tag recommendation [18, 14]. A special case of Tucker Decomposition is Canonical Decomposition (CD) [4] also known as PARAFAC [10]. An issue with CD is that it considers only the three wise interactions between the entities and their relations. We consider the pairwise tensor model (PITF) [15] which considers various interactions however we would like to consider interactions [10]. An issue with CD is that it considers only the three wise interactions. PITF simply models the two way interactions between the entities as well as between the entity and relations, the pairwise model is able to model the interactions [5] this model has better performance compared with the path ranking algorithm (PRA) [12]. Results are as shown in Table 1. From Table 1, our approach performs significantly better than PRA and neural network approaches. Specifically, Our approach performs better for relations where direct inferencing is very difficult.

4. EXPERIMENTS

In our experiments, we follow similar experiments as in [5] for WordNet. There are a total of 38,966 entities and 11 relations. We use 112,581 triples for training and 10,544 for test. We consider the wordnet relationships - has instance, type of, member meronym, member holonym, part of, has part, subordinate instance of, domain region, synset domain region, similar to, and domain topic.

For each triplet (e1, R, e2), we compute the score f(e1, R, e2) for all other entities in the knowledge base. We sort the results in descending order and report the top ranked entity as correct entity. Our pairwise tensor model was able to obtain a score of 25% while the neural models achieve 10.6%.

We also report experiments on NELL knowledge base and compare with the path ranking algorithm (PRA) [12]. Results are as shown in Table 1. From Table 1, our approach performs significantly better than PRA and neural network approaches. Specifically, Our approach performs better for relations where direct inferencing is very difficult.

4.1 Classification

In this experiment we ask the model whether a set of triples is true or not. With the help of a vocabulary of semantic word vectors, we can query whether certain relationships hold or not even for entities that were not originally in Wordnet.

We use a development set to find a threshold for each relation such that \( f(e_1, R, e_2) \geq T_R \) the relation holds or else it is false. As explained in the training stage we randomly switch entities and relations from correctly testing triplets. We observe an accuracy of 78% with semantic word vectors. In contrast the model by Collobert [6] achieved 66.7%. There is decrease in performance if the entities were initialized randomly.

5. CONCLUSION

In this paper we use the Pairwise Tensor Model where we model pairwise relations between the entity and its relations. Similar to [5] this model has better performance for ranking as well as for predicting unseen relationships be-

<table>
<thead>
<tr>
<th>Task</th>
<th>PRA Paths p@100</th>
<th>Pairwise Tensor Model p@100</th>
<th>Neural Network [6] p@100</th>
</tr>
</thead>
<tbody>
<tr>
<td>athletePlaysForTeam</td>
<td>125 0.46</td>
<td>0.42</td>
<td>0.48</td>
</tr>
<tr>
<td>from their knowledge bases athletePlaysInLeague</td>
<td>15 0.84</td>
<td>0.86</td>
<td>0.85</td>
</tr>
<tr>
<td>athletePlaysSport</td>
<td>34 0.78</td>
<td>0.82</td>
<td>0.80</td>
</tr>
<tr>
<td>stadiumLocatedInCity</td>
<td>18 0.62</td>
<td>0.68</td>
<td>0.65</td>
</tr>
<tr>
<td>teamHomeStadium</td>
<td>66 0.48</td>
<td>0.52</td>
<td>0.52</td>
</tr>
<tr>
<td>teamPlaysInCity</td>
<td>29 0.86</td>
<td>0.92</td>
<td>0.9</td>
</tr>
<tr>
<td>teamPlaysInLeague</td>
<td>36 0.70</td>
<td>0.80</td>
<td>0.78</td>
</tr>
<tr>
<td>companiesHeadquarteredIn</td>
<td>42 0.54</td>
<td>0.58</td>
<td>0.56</td>
</tr>
<tr>
<td>publicationJournalist</td>
<td>25 0.70</td>
<td>0.74</td>
<td>0.73</td>
</tr>
<tr>
<td>teamWonTrophy</td>
<td>56 0.50</td>
<td>0.55</td>
<td>0.52</td>
</tr>
<tr>
<td>worksFor</td>
<td>62 0.60</td>
<td>0.68</td>
<td>0.64</td>
</tr>
</tbody>
</table>

Table 1: Evaluation on Nell knowledge base
tween entities. It enables the extension of a knowledge base without external textual corpora.

6. REFERENCES


