Knowledge Graphs Embeddings to Support Knowledge Exploration

Federico Bianchi

University of Milan - Bicocca, Viale Sarca 336, Milan, Italy,
federico.bianchi@disco.unimib.it

Abstract. Knowledge Graphs (KGs) have become a standard model for knowledge representation used both in research and industry. In a KG, real world entities are represented and connected by relations. The huge amount of information that is contained inside KGs requires the development of techniques for the exploration of data. One problem with KGs is that they provide a sparse representation that is computationally engaging to manage and that hides some of the global knowledge of both entities and relations. To solve this problem, high dimensional graphs can be embedded in a lower dimensional vector space in which both local and global properties of KG's elements are more visible. The information that is contained inside the embeddings can be also used for KGs exploration. In this work we introduce the problem and the state of the art. We also present some of the initial results and our next steps.

1 Introduction

Knowledge Graphs (KG) have become a standard model to represent information and knowledge. In a KG real world entities are represented as nodes while the relations between them are represented by edges that connect entities. Fundamental brick of knowledge inside a KG is the triple <subject, relation, object>, where subject and object are generally two entities that are connected by a relation. A KG consists of many triples. One example of triple of a KG is <Barack Obama, born in, Hawaii>. Chains of triples, that are semi-walk inside the graph, are also called Semantic Associations (SAs) [1]. KG can also have an ontology associated, that connects each entity to a type; for example Barack Obama is a Person. Types are structured in a hierarchy to allow inheritance of properties from type to type (e.g. the type Politician inherits properties from the type Person, since a politician is a person). One example of KG is DBpedia[1] that constructs its KG from information stored inside Wikipedia. Even in the industry KGs have started to gain attention: companies like Google, Facebook and Microsoft have all started to build and use their own KG. For instance, Google extends the results of a search query on the Google search engine with additional information extracted from its own KG. A KG contains huge amount of information and developing techniques for the exploration of data has rapidly

[1] http://dbpedia.org

become a crucial task.

KGs are often characterized by a sparse representation that is difficult to manage and to explore: to reduce the dimensionality the KG entities and relations can be embedded in a lower dimensional space. One of the main advantage of the embedded vectors that we want to exploit is the fact that they aggregate much of the information that comes from the KG and this information can be used to explore the graphs and finding relevant information inside of them. Entities that are not connected in the KG might be similar in the embedded space because they share latent factors hardily visible in the original representation of the KG. In particular, this PhD work will be focused on the generation of embeddings that can be used for exploration of KGs. In the next section we present an overview of the related work for both KGs exploration and KG embeddings. We will then describe our initial results and in the end we will analyze our next steps. Advisor of this PhD project will be Matteo Palmonari of University of Milan - Bicocca.

2 Related Work

KGs Exploration In a recent survey [2] several methods for the exploration of KG are summarized and it is stated that different techniques for the exploration use a mix of navigation, filtering, sampling and visualization methods to help users explore large data sets. RelFinder [3] is an online application that can be used to retrieve SAs of different lengths between two specific entities given in input. In the literature other applications have been defined to explore SAs and they also incorporate measures to evaluate and explain the SAs between two specific entities [4,5,6]. Refer [7] is a Wordpress Plugin that helps a user by enriching an article with additional information extracted from KBs like Wikipedia. The plugin is used to find entities in an article and it recommends to the user entities that are estimated to be unknown to her. An other approach uses genetic programming to find strong relationships in linked data [8].

KGs Embeddings Recent approaches to embed KGs [9,10,11] rely on finding a representation where entities and relations are projected in a vector space using some constraints to define the position of the elements in the space. TransE [11] is an algorithm that, given a set of triples (h, l, t), generate embedding such that the embedding of entity $\mathbf{h}$ plus the embedding of the relationship $\mathbf{l}$ gives a point in the space that is close to the embedding of the entity $\mathbf{t}$. In this way, entities and relations are embedded in the same vector space. TransE applies well to relations that are 1-to-1, but does not work well with other kinds of relations. This work has been extended allowing the possibility of representing 1-to-N, N-to-1 and N-to-N relationships between entities [9]. This is done by using different vector spaces for entities and relations data using a projection matrix that projects entities from their own space into the relation space. Other approaches consider KGs has a 3-dimensional matrix (tensor) and use tensor factorization methods

to reduce the dimensionality and to find latent components that can be used for different tasks such as link prediction tasks [12].

Natural language processing techniques can be used on text to generate vector representations of words. Word2vec [13] is an algorithm that generates vector representations of a center word using by considering as a context other words near to the center word. The basic idea is derived from the distributional hypothesis which states that "words that occur in the same contexts tend to have similar meanings" [14]. The same idea can be applied to text that contains entities and can be thus used to generate representations of entities that are based on the relative frequency of entities inside a text. One example of these is presented in the context of a learning to rank framework where entity embeddings are used as a feature for entity relatedness [15].

3 Preliminary Results

First experiments on KG exploration we did were reported in a recent work [1]. We defined a model that allows users to view interesting SAs while they are reading a news online. SAs are extracted form the entities that are found inside the text using a semantic annotator [16]. Users can iteratively rate SAs and we use an active learning to rank system that provides new SAs to the users at each iteration. The objective is to ask to the user the less number of ratings as possible on small samples of SAs and to quickly find the SAs that might interest her most.

Another step in the direction of knowledge exploration was done with the use of KGs embeddings [17]. Basing our approach on the distributional hypothesis we generated representations for entities (ER) and representations for types (TR) starting from text. The idea is that the vector representation of each entity can be concatenated to the representation of its most specific type and thus generating a joint representation of entity and types (TER). This joint representation allows to keep explicit types that comes from the KG, while generally in literature embeddings do not have explicit information related to types [11]. This representation is interesting because in the joint embedded space entity of the same (or similar) type are nearer to each other. Our current approach takes in input texts that are semantically annotated, i.e., text where mentions of entities in the KG are recognized. To generate a corpus of text that contains types, we replace each entity occurrence inside the annotated text with its own most specific type. Figure 1 shows an example of annotation, we took text from the Wikipedia abstract of the city Rome and we used a semantic annotation tool to find the DBpedia entities in the text, in a second steps entities are replaced by types (dbr and dbo, are DBpedia's prefixes).

We use word2vec [13] on the corpus that contains entities and on the corpus that contains types. In this way we are able to learn embeddings for both entities and types. Since we now have the representation of the entities and the representation of types we can generate TER as explained before. One major

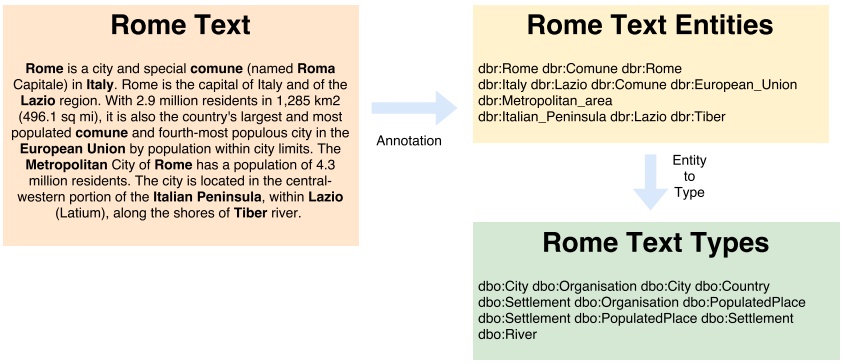

Fig. 1. Example of the annotation process starting from text

advantage of embeddings generated using this approach is that knowledge encoded in the vector representations can change with the corpus even if the KG does not change.

This model has been tested on an analogical reasoning task that are also tackled with word embeddings. An example of analogical reasoning task could be: "Rome" is to "Italy" as ? is to "France". To solve the task, the model should be able to find "Paris", as answer. This can be done in the vector space using vector operations like $v(\text{“}Rome\text{”}) - v(\text{“}Italy\text{”}) + v(\text{“}France\text{”}) \approx v(\text{“}Rome\text{”})$. Analogical reasoning with entities is an example of knowledge exploration task in which the analogical relation about two entities (e.g. Rome and Italy) is used to infer the relation of other two entities (e.g. France and Paris). Words are ambiguous (e.g., with Paris is the name of a city in France, but also of a city in Texas) and this is way the TER model, that uses entities, reaches an high accuracy in the analogical reasoning task [17].

4 Conclusion and Future Work

The initial results given by TER are promising and we plan to extend the TER model and to evaluate its performance on different tasks related to analogical reasoning with entities. We have started our work by extending the entity representations with a representation that comes from types, nevertheless other extensions are possible and we would soon like to extend the representation of the entities using temporal information (e.g. extending the representation of people using information related to the time in which they have lived). During our first steps in KG exploration [1] we want use the embeddings as features inside the active learning to rank model we developed.

References

1. Federico Bianchi, Matteo Palmonari, Marco Cremaschi, and Elisabetta Fersini. Actively learning to rank semantic associations for personalized contextual exploration of knowledge graphs. In *ESWC*, 2017.
2. Nikos Bikakis and Timos Sellis. Exploration and visualization in the web of big linked data: A survey of the state of the art. *preprint arXiv:1601.08059*, 2016.
3. Philipp Heim, Sebastian Hellmann, Jens Lehmann, Steffen Lohmann, and Timo Stegemann. Relfinder: Revealing relationships in rdf knowledge bases. In *SAMT*, pages 182–187. Springer, 2009.
4. Gong Cheng, Yanan Zhang, and Yuzhong Qu. Explass: exploring associations between entities via top-k ontological patterns and facets. In *ISWC*, pages 422–437. Springer, 2014.
5. Giuseppe Pirrò. Explaining and suggesting relatedness in knowledge graphs. In *ISWC*, pages 622–639. Springer, 2015.
6. Lujun Fang, Anish Das Sarma, Cong Yu, and Philip Bohannon. Rex: explaining relationships between entity pairs. *Proceedings VLDB*, 5(3):241–252, 2011.
7. Tabea Tietz, Joscha Jäger, Jörg Waitelonis, and Harald Sack. Semantic annotation and information visualization for blogposts with Refer. In *VOILA '16*, volume 1704, pages 28 – 40, 2016.
8. Ilaria Tiddi, Mathieu d'Aquin, and Enrico Motta. Learning to assess linked data relationships using genetic programming. In *ISWC*, pages 581–597. Springer, 2016.
9. Yankai Lin, Zhiyuan Liu, Maosong Sun, Yang Liu, and Xuan Zhu. Learning entity and relation embeddings for knowledge graph completion. In *AAAI*, pages 2181–2187, 2015.
10. Zhen Wang, Jianwen Zhang, Jianlin Feng, and Zheng Chen. Knowledge graph embedding by translating on hyperplanes. In *AAAI*, pages 1112–1119, 2014.
11. Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. Translating embeddings for modeling multi-relational data. In *NIPS*, pages 2787–2795, 2013.
12. Maximilian Nickel, Volker Tresp, and Hans-Peter Kriegel. A three-way model for collective learning on multi-relational data. In *ICML-11*, pages 809–816, 2011.
13. Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In *NIPS*, pages 3111–3119, 2013.
14. Patrick Pantel. Inducing ontological co-occurrence vectors. ACL '05, pages 125–132, Stroudsburg, PA, USA, 2005. Association for Computational Linguistics.
15. Pierpaolo Basile, Annalina Caputo, Gaetano Rossiello, and Giovanni Semeraro. Learning to rank entity relatedness through embedding-based features. In *NLDB*, pages 471–477. Springer, 2016.
16. Pablo N Mendes, Max Jakob, Andrés García-Silva, and Christian Bizer. Dbpedia spotlight: shedding light on the web of documents. In *7th international conference on semantic systems*, pages 1–8. ACM, 2011.
17. Federico Bianchi and Matteo Palmonari. Joint learning of entity and type embeddings for analogical reasoning with entities. In *In Proceedings of the NL4AI Workshop, co-located with the International Conference of the Italian Association for Artificial Intelligence (AI*IA)*, 2017.