Using Word Embeddings in Linked Data Search

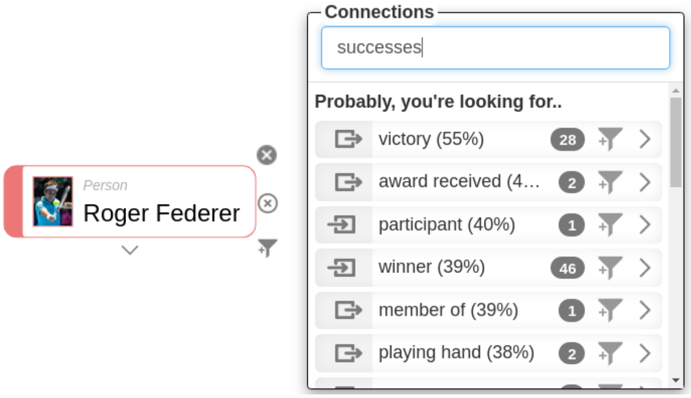
Poster description

Gerhard Wohlgenannt[1], Nikolay Klimov[1], Dmitry Mouromtsev[1], Daniil Razdyakonov[2], Dmitry Pavlov[2], and Yury Emelyanov[2]

[1] Intern. Lab. of Information Science and Semantic Technologies, ITMO University, St. Petersburg, Russia `http://en.ifmo.ru/en`
[2] Vismart Ltd., St. Petersburg, Russia `https://vismart.biz`

1 Introduction

In the last decade, a vast number of linked data (LD) datasets have been created. One of the main challenges in the consumption of linked data is to create visual and natural language interfaces for common users – interfaces which hide the complexity of underlying schemata and the need to write queries in languages such as SPARQL [1]. Ontodia[3] is an open-source library for simple OWL and RDF diagramming and visual exploration. Additionally to the graphical representation, Ontodia provides natural language input fields for example to search for specific entities or within the properties of an entity. Currently, only exact matches to a user query in the labels of properties and entities are found and returned. Giving an example, if a user searches in the Wikidata dataset[4] for the “successes” of entity *Roger Federer*, no exact matches in property labels exist, and therefore no result can be displayed. Using pre-trained word embedding models, which were extended with representations of the Wikidata properties (see below for system details), the improved system ranks the entity properties by similarity to the input query. Figure 1 provides a screenshot of the result for the described scenario.



Connections

successes

Probably, you're looking for..

victory (55%)	28	>
award received (4…	2	>
participant (40%)	1	>
winner (39%)	46	>
member of (39%)	1	>
playing hand (38%)	2	>

Fig. 1. System suggestions for properties related to query term “success”.

[3] `http://www.ontodia.org`
[4] `https://www.wikidata.org`

2 System Description

A prototype of the described system is available online[5]. In a nutshell, the method is a follows: The main ingredient is a pre-trained word embedding model, which is used both to represent the user query words, and the words from the (Wikidata) property labels and description texts. We evaluated (see next section) various word embeddings types and models to find the best candidate. The user input query is split into single words, and stopwords are removed. The query is then represented by the vectorial sum of the query words. For all Wikidata properties, we precompute property vectors using the same strategy, by calculating the sum of the word vectors in the property labels and property description. Finally, cosine similarity between the query vector and property vectors is used to rank the properties for a query.

3 Evaluations

To evaluate the system, we first randomly choose 1150 entities from Wikidata. Many properties have property aliases defined, which can be used as a gold standard to test the system in finding the correct property for the aliases using the described method. The evaluation included the 29255 properties aliases of the properties in the random entity set. We experimented with many word embedding models and settings, due to space restriction we only discuss the best performing here, which was fastText [2]. With fastText, in 68.63% of evaluation cases the first ranked property was the correct one, and the correct suggestion was in 84.75% of Top-3 ranked properties. The mean reciprocal rank (MRR) is 0.78. The applied model was trained on Wikipedia 2016 and is found on github[6]. Query times are typically below $10ms$, well-suited for an interactive system.

4 Conclusions

In this publication we present simple but effective search in linked data (properties) using natural language. The main contributions are the system prototype which is integrated into Ontodia, and an extensive evaluation against a gold standard to demonstrate the effectiveness of the approach.

5 Acknowledgments

This work was supported by the Government of the Russian Federation (Grant 074-U01) through the ITMO Fellowship and Professorship Program.

References

1. Augenstein, I., Gentile, A.L., Norton, B., Zhang, Z., Ciravegna, F.: Mapping keywords to linked data resources for automatic query expansion. In: Cimiano, P.e.a. (ed.) ESWC 2013. pp. 101–112. Springer LNCS, Berlin, Heidelberg (2013)
2. Bojanowski, P., Grave, E., Joulin, A., Mikolov, T.: Enriching word vectors with subword information. arXiv preprint arXiv:1607.04606 (2016)

[5] `http://ontodia-prop-suggest.apps.vismart.biz/wikidata.html`

[6] `https://github.com/facebookresearch/fastText/blob/master/pretrained-vectors.md`