Generating Referring Expressions from Knowledge Graphs

Armita Khajeh Nassiri[2], Nathalie Pernelle[1], and Fatiha Saïs[1]

[1] LRI, Paris Sud University, CNRS 8623, Paris Saclay University, Orsay F-91405, France
firstname.lastname@lri.fr
[2] École Polytechnique, Palaiseau F-91128, France
armita.khajeh-nassiri@polytechnique.edu

1 Introduction

A *referring expression* (RE) is a description in natural language or a logical formula that can uniquely identify an entity. For instance, the 44th president of the United States unambiguously characterizes Barack Obama. Referring expressions find applications in disambiguation, data anonymization, query answering, or data linking. There may potentially exist many logical expressions for uniquely identifying an entity. Generation of referring expressions is a well-studied task in natural language generation [1]. Hence, various algorithms with different objectives have been proposed to automatically discover REs. These approaches vary depending on the expressivity of the logical formulas they can generate. For instance in [1, 2], REs that are created are conjunctions of atoms. While in [3], more complex REs represented in description logics are discovered that can involve the universal quantifier.
In this work our focus lies on automatically discovering REs for each entity within a class of a knowledge graph. *Keys* of a class are sets of properties whose values can uniquely identify one entity of that class. Hence, if the properties for the keys are instantiated, they can each be considered as a referring expression. What interests us in this work, is to efficiently discover REs by focusing on the ones that cannot be found by instantiating the keys. It should be noted that the quality of REs we discover is very dependent on the dataset. The completeness, correctness and lack of noise in the knowledge graph plays a pivotal role in how good and interpretable REs are.

2 Referring Expression Generation Approach

In this work, we discover **minimal** REs existing in a class. By minimality, we mean that there is no other RE that we discover and that can be logically entailed by the minimal one. The REs we mine always consist of conjunctions that specify the classes the entities belong to.

To generate REs for a given class C, we start by creating the maximal non-keys of C (the set of properties such that addition of a property will make it a key for that class) using SAKey [4]. The algorithm first generates candidate expressions containing one instantiated property (i.e. $p(x, v)$). Whenever an expression E only describes one instance i of C, E is output as a referring expression. Adding more properties to the

description E will still uniquely identify the instance i, just making it more complex. Hence, we remove the REs (e.g. $p(i, v)$) found at the end of this step and reduce the search space. Then, the remaining candidate expressions are taken into account with one more property at each step, until either the search space is empty or there is no more set of non-keys to consider. To increase the depth of subgraph, we have to consider the class of the new individual and obtain its corresponding set of maximal non-keys so that the process can be reiterated. Some pruning techniques can be applied to limit the size and the complexity of the REs discovered by our approach. For instance the depth of the graph pattern and number of allowed variables can be limited.

3 Experimental Evaluation

We chose YAGO as the knowledge graph on which we discover the REs and used 10 different classes such as Actor, City and Book (same data used in VICKEY [5]). We mined REs of depth one and for example, for the class City (with 1.1M triples) we found 1.2M REs in less than 2 minutes. On average, our approach can detect from 1.5 to 14.3 RE per individual depending on the class.
This approach can discover RE such as: *made in heaven* is the album created by Queen in the year 1991. Among the actors, only *George Clooney* has been born in Lexington-Kentucky in the year 1961. When we ran the algorithm with depth 2, we obtained REs like *Alfred Werner* is a scientist who has won the Nobel Prize in Chemistry and has graduated from a university located in Zurich.

4 Conclusion

In this paper, we proposed an approach that can efficiently discover REs by reducing the search space thanks to maximal non-keys. Due to the incompleteness of knowledge graphs, entity linking using keys may be insufficient to link all individuals. We expect that using REs will increase the recall of rule-based data linking methods.

References

1. R. Dale. *Generating referring expressions - constructing descriptions in a domain of objects and processes*. ACL-MIT press series in natural language processing. MIT Press, 1992.
2. E. Krahmer, S. v. Erk, and A. Verleg. Graph-based generation of referring expressions. *Computational Linguistics*, 29(1):53–72, 2003.
3. Y. Ren, J. Z. Pan, and Y. Zhao. Towards soundness preserving approximation for abox reasoning of OWL2. In *Proceedings of the 23rd International Workshop on Description Logics (DL 2010), Waterloo, Ontario, Canada, May 4-7, 2010*, 2010.
4. D. Symeonidou, V. Armant, N. Pernelle, and F. Saïs. SAKey: Scalable Almost Key Discovery in RDF Data. In S. Verlag, editor, *In proceedings of the 13th International Semantic Web Conference, ISWC 2014*, volume Lecture Notes in Computer Science, pages 33–49, Riva del Garda, Italy, Oct. 2014.
5. D. Symeonidou, L. Galàrraga, N. Pernelle, F. Saïs, and F. Suchanek. VICKEY: Mining Conditional Keys on Knowledge Bases. In *International Semantic Web Conference (ISWC)*, volume 10587 of *Lecture Notes in Computer Science*, pages 661–677, Austria, Oct 2017. Springer.