Mining Scholarly Publications for Scientific Knowledge Graph Construction

Conference or Workshop Item

Mining Scholarly Publications for Scientific Knowledge Graph Construction

Davide Buscaldi[1], Danilo Dessì[2], Enrico Motta[3], Francesco Osborne[3], and Diego Reforgiato Recupero[2]

[1] davide.buscaldi@lipn.univ-paris13.fr
[2] {danilo_dessi, diego.reforgiato}@unica.it
[3] {enrico.motta, francesco.osborne}@open.ac.uk

Abstract. In this paper, we present a preliminary approach that uses a set of NLP and Deep Learning methods for extracting entities and relationships from research publications and then integrates them in a Knowledge Graph. More specifically, we i) tackle the challenge of knowledge extraction by employing several state-of-the-art Natural Language Processing and Text Mining tools, ii) describe an approach for integrating entities and relationships generated by these tools, and iii) analyse an automatically generated Knowledge Graph including 10,425 entities and 25,655 relationships in the field of Semantic Web.

1 Introduction

Knowledge graphs (KG) are large network of entities and relationships, usually expressed as RDF triples, relevant to a specific domain or an organization [4]. Many of state-of-the-art projects such as DBPedia, Google Knowledge Graph, BabelNet, and YAGO build KGs by harvesting entities and relations from textual resources, such as Wikipedia pages.

Scientific Knowledge Graphs (SKGs) focus on the scholarly domain and typically contains metadata describing research publications such as authors, venues, organizations, research topics, and citations. Good examples are Open Academic Graph[4], Scholarlydata.org [7], and OpenCitations [8]. Their main limitation is that they typically represent the content of the papers as unstructured text (title, abstract, sometimes the full text). Therefore, a significant challenge in this field regards the generation of SKGs that contain also a explicit representation of the knowledge presented in the publications [2], and potentially describes entities such as approaches, application, formats, and so on.

Most of the relevant information for populating such a KG could be derived from the text of research publications. However, integrating this information in a coherent knowledge graph is still an open challenge.

In this paper, we present a preliminary approach that uses a set of NLP techniques for extracting entities and relationships from research publications and then integrate them in a KG. More specifically, we i) tackle the challenge of knowledge extraction by employing several state-of-the-art Natural Language Processing and Text Mining tools, ii) describe an approach for integrating entities and relationships generated by these tools, and iii) analyse an automatically

[4] https://www.openacademic.ai/oag/

generated Knowledge Graph including 10,425 entities and 25,655 relationships in the field of Semantic Web.

2 The Proposed Approach

We collect a dataset composed by 12,007 abstracts of scientific publications about the Semantic Web domain. It was retrieved by selecting all publications from Microsoft Academic Graph dataset which contains the string "Semantic Web" in the "field of science" heading.

For extracting entities and relations, we exploited the following resources:

- An extractor framework designed by [5] which provides tools for detecting entities and relations from scientific literature. It detects six types of entities (*Task, Method, Metric, Material, Other-Scientific-Term, and Generic*) and seven types of relations among a list of predefined choices (*Compare, Part-of, Conjunction, Evaluate-for, Feature-of, Used-for, Hyponym-Of*).
- OpenIE [1], provided with Stanford Core NLP[5]. It detects general entities and relations among them, using verbs as predicates.
- The CSO Classifier [9], a tool for automatically tagging research papers with a set of research topics draw from the Computer Science Ontology (CSO)[6] [10]. CSO is a comprehensive ontology of research areas in the field of Computer Science, which was automatically generated from a dataset of 16 million research publications.

In order to generate the graph, we need to integrate all triples extracted from the abstracts. First we had to clean entities by removing punctuation, stopwords, merging singular and plural forms, splitting entities containing compound expressions, and handling acronyms.

For the entity merging task we exploit two data structures. The first one, labelled $W2LE$, maps each word to a list of entities that share the last token (e.g. *medical ontology*, *biomedical ontology*, *pervasive agent ontology*, and so on.). With $W2LE$ we avoided comparing those entities that syntactically could not refer to the same entity (e.g. the entities *ontology generation* and *ontology adoption* were not compared). The second one, labelled $E2E$, maps each original entity to the entity that will represent it in the KG.

Given an entity e and the list of its tokens $\{t_0, ..., t_n\}$, we chose t_n. If t_n was not present in $W2LE$, a new entry key t_n was added to $W2LE$ and its value is a list with e as its unique element. If t_n was in $W2LE$, then we compute the Levenshtein string similarity[7] between the entity e and all other entities $e'_0, ..., e'_m \in W2LE[t_n]$. If the resulting score met a given threshold t_L (set to 0.9 in the prototype), the entity e was mapped as e'_i in $E2E$. Otherwise e was mapped to itself in $E2E$. At the end, the entity e was added to $W2LE[t_n]$. Finally, the map $E2E$ was used to select the entities for the graph. For each

[5] https://stanfordnlp.github.io/CoreNLP/
[6] http://cso.kmi.open.ac.uk
[7] https://pypi.org/project/python-Levenshtein/

entry key e_x, if its corresponding entity $e_y = E2E[e_x]$ was not in the graph, a new entity with label e_y was added.

In order to merge similar relations and reduce their number, we clustered all verbs labels. To do so we exploited WordNet [6] and a set of Word2Vec word embeddings trained on 9 million research papers from Microsoft Academic Graph[8]. In details, given the set of all verbs $V = \{v_0, ..., v_n\}$, we built a distance matrix M considering as a distance between two verbs v_i and v_j the 1− Wu-Palmer[9] similarity between their synsets. Then, we apply a hierarchical clustering algorithms, cutting the dendrogram where the number of clusters had the highest value of overall silhouette-width [3]. Subsequently, clusters were refined as follows. Given a cluster c, we assigned each verb $v_{i_c} \in c$ with the word embedding w_i in the Word2Vec model, and computed the centroid ce of the cluster as the average of word embeddidngs of its elements. Then, we ordered verbs in ascending order by the distance from ce. All verbs with a distance over a threshold t were discarded. All the other verbs were mapped on the verb nearest to the centroid ce.

3 The Knowledge Graph

In this section, we report our preliminary results about the KG produced from 12,007 papers about the Semantic Web (using $t_L = 0.9$).

The resulting KG contains 10,425 entities and 25,655 relationships. It includes both verb-based relations (from OpenEI) and default relations (from the Extractor Framework). Verbs are usually more informative, but also harder to extract. The Extractor Framework is more flexible and it is able to extract a large number of relationships, but these are usually less specific. Using both systems allows us obtaining a good balance between coverage and specificity.

Table 1. Contribution of Extractor Framework and CSO to the KG entities.

Tools Entities Contribution	Count	Percentage
CSO	1034	9,92%
Extractor Framework	8668	83.15%
Exclusive CSO	117	1.12%
Exclusive Extractor Framework	7751	74.35%
Entities where both tools contribute	917	8.8%
Derived Entities	1640	15.73%

Table 1 reports statistics about entities. To weight the actual contribution of each tool, we counted the number of entities that were extracted by each tool. With the label *Exclusive* we indicate the percentage of entities identified only by a specific tool. The row *Derived Entities* refers to the additional entities that were obtained by merging or splitting the original entities extracted by the tools.

Most entities come from the Extractor Framework tool, which contributes to the 83,15% of all entities, and exclusively contributes to 74,35% of them. The

[8] Avaliable at http://tiny.cc/w0u43y
[9] http://www.nltk.org/howto/wordnet.html

CSO Classifer contributes to 9.92% of them, but only a minority are exclusive. This was expected, since CSO contains fairly established research topics. Conversely, the Extractor Framework is able to identify many entities that appear in very few research papers. On average, each entity was extracted 3.69 ± 32.22 times by one of the tools.

Table 2. Contribution of Extractor Framework and OpenIE to the KG relations.

Tools Relations Contribution	Count	Percentage
Extractor Framework	23,624	92.09%
OpenIE	3,116	12.15%
Exclusive Extractor Framework	22,539	87.85%
Exclusive OpenIE	2,031	7.92%
Contribution of both tools	1,085	4.23%

Similarly to entities, the Extractor Framework produced also the majority of the relationships with a coverage of 92.09%. However, the 12.15% of relationships extracted by OpenIE are usually more informative since they are mapped to specific verbs. On average, each relationship was extracted 1.32 ± 1.41 times.

References

1. Angeli, G., Premkumar, M.J.J., Manning, C.D.: Leveraging linguistic structure for open domain information extraction. In: Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing. vol. 1, pp. 344–354 (2015)
2. Auer, S., Kovtun, V., Prinz, M., Kasprzik, A., Stocker, M., Vidal, M.E.: Towards a knowledge graph for science. In: Proceedings of the 8th International Conference on Web Intelligence, Mining and Semantics. p. 1. ACM (2018)
3. Dessì, D., Recupero, D.R., Fenu, G., Consoli, S.: A recommender system of medical reports leveraging cognitive computing and frame semantics. In: Machine Learning Paradigms, pp. 7–30. Springer (2019)
4. Ehrlinger, L., Wöß, W.: Towards a definition of knowledge graphs. SEMANTiCS (Posters, Demos, SuCCESS) **48** (2016)
5. Luan, Y., He, L., Ostendorf, M., Hajishirzi, H.: Multi-task identification of entities, relations, and coreference for scientific knowledge graph construction. In: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing. pp. 3219–3232 (2018)
6. Miller, G.A.: Wordnet: a lexical database for english. Communications of the ACM **38**(11), 39–41 (1995)
7. Nuzzolese, A.G., Gentile, A.L., Presutti, V., Gangemi, A.: Conference linked data: the scholarlydata project. In: ISWC. pp. 150–158. Springer (2016)
8. Peroni, S., Shotton, D., Vitali, F.: One year of the opencitations corpus. In: ISWC. pp. 184–192. Springer (2017)
9. Salatino, A.A., Thanapalasingam, T., Mannocci, A., Osborne, F., Motta, E.: Classifying research papers with the computer science ontology. In: International Semantic Web Conference (P&D/Industry/BlueSky). CEUR Workshop Proceedings. vol. 2180
10. Salatino, A.A., Thanapalasingam, T., Mannocci, A., Osborne, F., Motta, E.: The computer science ontology: a large-scale taxonomy of research areas. In: ISWC. pp. 187–205 (2018)