ALOD2vec Matcher Results for OAEI 2021

Jan Portisch¹,²[0000–0001–5420–0663] and Heiko Paulheim¹[0000–0003–4386–8195]

¹ Data and Web Science Group, University of Mannheim, Germany
{jан, heiko}@informatik.uni-mannheim.de
² SAP SE Business Technology Platform - One Domain Model, Walldorf, Germany
jan.portisch@sap.com

Abstract. This paper presents the results of the ALOD2vec Matcher in the Ontology Alignment Evaluation Initiative (OAEI) 2021. The matching system exploits a Web-scale dataset, i.e. WebIsALOD, as background knowledge source. In order to make use of the dataset, the RDF2vec approach is applied to derive embeddings for each concept available in the dataset. ALOD2vec Matcher participated in the OAEI 2018 and 2020 campaigns before. This is the system’s third participation.

Keywords: Ontology Matching · Ontology Alignment · External Resources · Background Knowledge · Knowledge Graph Embeddings · RDF2vec

1 Presentation of the System

1.1 State, Purpose, General Statement

The ALOD2vec Matcher is an element-level, label-based matcher which uses a large-scale Web-crawled RDF dataset of hypernymy relations as general purpose background knowledge. The dataset contains many tail-entities as well as instance data such as persons or places which cannot be found in common thesauri. In order to exploit the external dataset, a neural language model approach is used to obtain a vector for each concept contained in the dataset. This matching system was initially introduced at the OAEI 2018 [13] and also participated in the 2020 campaign [10]. The implementation is based on the Matching Evaluation Toolkit [6] as well as the KGvec2go [11] REST API to obtain vector representations via a Web API.

1.2 Specific Techniques Used

After the basic concepts of this matcher are introduced (Foundations), the specific techniques applied are presented.

Foundations

³ Copyright © 2021 for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).
WebIsALOD Dataset A frequent problem that occurs when working with external background knowledge is the fact that less common entities are not contained within a knowledge base. The WebIsA [16] database is an attempt to tackle this problem by providing a dataset which is not based on a single source of knowledge – like DBpedia [7] – but instead on the whole Web: The dataset consists of hypernymy relations extracted from the Common Crawl\(^4\), a freely downloadable crawl of a significant portion of the Web. A sample triple from the dataset is `european_union skos:broader international_organization`\(^5\). The dataset is also available via a Linked Open Data (LOD) endpoint\(^6\) under the name WebIsALOD [5]. In the LOD dataset, a machine-learned confidence score \(c \in [0,1]\) is assigned to every hypernymy triple indicating the assumed degree of truth of the statement.

RDF2vec The background dataset can be viewed as a very large knowledge graph; in order to obtain a similarity score for nodes and edges in that graph, the RDF2vec [15] approach is used. It applies the word2vec [8,9] model to RDF data: Random walks are performed for each node and are interpreted as sentences. After the walk generation, the sentences are used as input for the word2vec algorithm. As a result, one obtains a vector for each word, i.e., a concept in the RDF graph. Multiple flavors of RDF2vec have been developed in the past such as biased walks [1] or RDF2Vec Light [12].\(^7\)

KGvec2go Training embeddings on large knowledge graphs can be computationally very expensive. Moreover, the resulting embedding models can be very large since a multidimensional vector needs to be persisted for every node in the knowledge graph. However, most downstream applications require only a small subset of node vectors. The KGvec2go project [11] addresses these problems by providing a free REST API\(^8\) for pre-trained RDF2vec models on various large knowledge graphs (among which WebIsALOD is also available).

Monolingual Matching ALOD2vec Matcher is a monolingual matching system. For the alignment process, the system retrieves the labels of all elements of the ontologies to be matched. A filter adds all simple string matches to the final alignment in order to increase the performance. The remaining labels are linked to concepts in the background dataset, are compared, and the best solution is added to the final alignment. A high-level view of the matching system is provided in Figure 1.

The first step is to link the obtained labels from the ontology to concepts in the WebIsALOD dataset. Therefore, string operations are performed on the label

\(^4\) see http://commoncrawl.org/
\(^5\) see http://webisa.webdatacommons.org/concept/european_union_
\(^6\) see http://webisa.webdatacommons.org/
\(^7\) For a good overview of the RDF2vec approach and its applications, refer to http://www.rdf2vec.org/
\(^8\) see http://kgvec2go.org/api.html
and it is checked whether the label is available in WebIsALOD. If it cannot be found, a token-lookup is performed. Given two entities \( e_1 \) and \( e_2 \), the matcher uses their textual labels to link them to concepts \( e'_1 \) and \( e'_2 \) in the external dataset. Afterwards, the embedding vectors \( v_{e'_1} \) and \( v_{e'_2} \) of the linked concepts \( (e'_1, e'_2) \) are retrieved via a Web request and the cosine similarity between those is calculated. Hence:

\[
\text{sim}(e_1, e_2) = \text{sim}_{\text{cosine}}(v_{e'_1}, v_{e'_2}).
\]

If \( \text{sim}(e_1, e_2) > t \) where \( t \) is a threshold in the range of 0 and 1, a correspondence is added to a temporary alignment. In a last step, a one-to-one arity is enforced by applying a Maximum Weight Bipartite filter on the temporary alignment.

In order to consume the vectors in Java, a client has been implemented and contributed to the MELT-ML module. The KGvec2go REST API can now be accessed through class `KGvec2goClient`. Even though this matcher only uses the WebIsALOD dataset, the implementation supports all datasets accessible on KGvec2go. The extension is available by default in MELT 2.6.

**Instance Matching** After classes and properties have been matched, instances are matched using a string index. The confidence score assigned to instances belonging to matched classes is higher than that of matches between instances belonging to non-matched classes.

**Explainability** ALOD2vec Matcher provides an explanation for every correspondence that is added to the final alignment. Therefore, the extension capabilities of the alignment format \([3]\) are used. Two concrete examples from the Anatomy track for explanations of the matching system are: “Label ‘aqueous
humour’ of ontology 1 and label ‘Aqueous Humor’ of ontology 2 have a very similar writing.” or “The following two label sets have a cosine above the given threshold: [lens|anterior|epithelium] and [anterior|surface|lens]”. In order to explain a correspondence, the description property of the Dublin Core Metadata Initiative is used.

1.3 Extensions to the Matching System for the 2021 Campaign

For the 2021 campaign, the matching system was adapted to use the latest MELT release and was packaged as MELT Web Docker container. The 2021 implementation is publicly available on GitHub.

2 Results

2.1 Anatomy Track

On the anatomy dataset, the system scores a precision of 0.828, a recall of 0.766, and an $F_1$ of 0.796.

2.2 Conference Track

On the conference track, the matcher achieves a recall of 0.49 and a precision of 0.64. The overall $F_1$ score on ra1-M3 was 0.59.

2.3 Multifarm Track

Since the WebIsALOD dataset is only available in English, the focus of the ALOD2vec Matcher is on monolingual matching tasks.

2.4 LargeBio Track

In its current version, the LargeBio track is too large for the matching system’s architecture. There is a tradeoff in package size and runtime performance (a large package with all vectors matches faster than the submitted small package which obtains vectors at runtime from KGvec2go). The current architecture of ALOD2vec Matcher is not intended for large-scale matching – however, the matching algorithm itself could be used for large-scale matching.

2.5 Knowledge Graph Track

The system could complete all matching tasks in time. As in the previous year, this matcher obtains the second best results achieving almost the same score as the Wiktionary Matcher 2021 [14]. The overall $F_1$ score was 0.87 on the complete track.

---

9 see http://purl.org/dc/terms/description
10 see https://dwslab.github.io/melt/matcher-packaging/web
11 see https://github.com/janothan/ALOD2VecMatcher
2.6 Common Knowledge Graph Track

This year, a new track was added to the OAEI: The Common Knowledge Graph Track [4]. Although not optimized for this track, Alod2vec Matcher achieved the second best result with an $F_1$ score of 0.89.

3 Conclusion

In this paper, we presented the newest version of the ALOD2vec Matcher, a matcher utilizing an RDF2vec vector representation of the WebIsALOD dataset, as well as its results in the 2021 OAEI. In the future, the matching system could be improved by using another, potentially larger or newer, hypernymy database, by exploiting other embedding algorithms, and by adding further matching strategies to the overall algorithms such as checking of logical constraints.

References