GraphMatcher System Presentation

Sefika Efeoglu[1,2]

[1]Technische Universität Berlin, Electrical Engineering and Computer Science Department, Marchstraße 23, 10587 Berlin, Germany
[2]Freie Universität Berlin, Department of Computer Science, Takustraße 9, 14195 Berlin, Germany

Abstract
Ontology matching finds a relationship or correspondence between two or more entities in two or more ontologies. To solve the interoperability of the domain ontologies, semantically correspondence entities in these ontologies must be identified and aligned before merging them. GraphMatcher is an ontology matching system using a graph attention approach to compute higher-level representation of a class together with its surrounding terms.

Keywords
graph attention, graph representation, ontology matching

1. Presentation of the system

GraphMatcher [1] is an ontology matching system based on graph representation learning, using graph attention [2] along with a neighborhood aggregation approach. The graph representation learning approach utilizes graph attention and introduces a neighborhood aggregation algorithm that reveals the contextual meaning of the central class and property.

1.1. Proposal and general statement

Using domain ontologies developed by different domain experts together in an application raises the issue of interoperability. Therefore, before merging them, correspondences between the ontologies have been identified. This interoperability problem is addressed by an ontology matching approach, which identifies correspondences between two or more entities in two or more independent ontologies. To comprehend the actual meaning of a class, contextual information about the class or property is necessary.

GraphMatcher aims to address two weaknesses observed in machine learning-based ontology matching approaches [3, 4]: (i) the lack of contextual information about the property and class, and (ii) how to represent the ontology's data [1]. The former limitation is addressed by aggregating the neighboring terms of the center class or property, while the latter is tackled by representing the ontology's data via an arbitrary graph. Our goal is to develop a graph representation learning model based on a graph attention mechanism [2], utilizing Siamese

OM 2023: The 18th International Workshop on Ontology Matching collocated with the 22nd International Semantic Web Conference ISWC-2023 November 7th, 2023, Athens, Greece
sefika.efeoglu@{fu-berlin.de,tu-berlin.de} (S. Efeoglu)
0000-0002-9232-4840 (S. Efeoglu)

networks [5, 6, 4], to identify semantically corresponding concepts within the ontologies. The graph attention mechanism computes the higher-level representation of a concept and its surrounding concepts, and the model subsequently determines similarity scores between pairs of concepts.

1.2. Specific techniques used

GraphMatcher [1] utilizes a graph representation learning approach employing graph attention [2] and a supervised machine learning algorithm with a network consisting of five layers. The primary contribution is the adaptation of the graph attention mechanism to compute the higher-level representation of the contextual embedding of the center class. It is important to note that this year, we submitted the model that yielded the best performance results in its hyperparameter tuning process. There have been no other improvements or changes to the model.

1.3. Adaptations made for the evaluation

The GraphMatcher framework has been developed in Python using PyTorch and Ontospy, and it is packaged by SEALS using MELT [7]. However, because of the Universal Sentence Encoder's TensorFlow Hub dependency, the model might not run on machines that do not support this library. In the case of this dependency issue, please use another sentence encoder.

1.4. Parameter settings

The model's parameters include a learning rate of 0.01, five epochs, weight decay of 0.01, and a batch size of 32. The threshold is computed from false positive alignments in the validation data, following the approach proposed by the VeeAlign system [4] [1]. These parameters were determined through five-fold cross-validation. The code is available at https://github.com/sefeoglu/gat_ontology_matching.

2. Results

The results of the GraphMatcher in the OAEI 2023 conference tracks are available at https://oaei.ontologymatching.org/2023/results/conference/.

3. General Comments

We plan to change the algorithm to an unsupervised machine learning approach in its next version, as there is no explicit rule regarding the supervised machine learning approach in the OAEI. In light of the study introducing the graph attention approach [2], this algorithm has also been adapted to the unsupervised approach.

[1] The project uses VeeAlign's approach directly to compute the threshold with the permission of the first author.

References

[1] S. Efeoglu, Graphmatcher: A graph representation learning approach for ontology matching (2022).

[2] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Liò, Y. Bengio, Graph attention networks (2018). arXiv:1710.10903.

[3] P. Kolyvakis, A. Kalousis, D. Kiritsis, DeepAlignment: Unsupervised ontology matching with refined word vectors, in: Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers), ACL, 2018, pp. 787–798.

[4] V. Iyer1, A. Agarwal, H. Kumar, Veealign: A supervised deep learning approach to ontology alignment (2020).

[5] A. Bento, A. Zouaq, M. Gagnon, Ontology matching using convolutional neural networks, in: Proceedings of the 12th Language Resources and Evaluation Conference, ELRA, Marseille, France, 2020, pp. 5648–5653. URL: https://www.aclweb.org/anthology/2020.lrec-1.693.

[6] J. Chen, J. Gu, Adol: a novel framework for automatic domain ontology learning, Super Computing (2021).

[7] S. Hertling, J. Portisch, H. Paulheim, Melt - matching evaluation toolkit, in: M. Acosta, P. Cudré-Mauroux, M. Maleshkova, T. Pellegrini, H. Sack, Y. Sure-Vetter (Eds.), Semantic Systems. The Power of AI and Knowledge Graphs, Springer International Publishing, Cham, 2019, pp. 231–245.