Multifaceted Context Representation using Dual Attention for Ontology Alignment

Vivek Iyer¹, Arvind Agarwal², Harshit Kumar²
¹IIIT Hyderabad, India, vivek.iyer@research.iiit.ac.in
²IBM Research, India, email address must be in roman text type, not monospace or sans serif
{arvagarw,harshit}@in.ibm.com

Abstract

Ontology Alignment is an important research problem that finds application in various fields such as data integration, data transfer, data preparation etc. State-of-the-art (SOTA) architectures in Ontology Alignment typically use naive domain-dependent approaches with handcrafted rules and manually assigned values, making them unsuitable and inefficient. Deep Learning approaches for ontology alignment use domain-specific architectures that are not only inextendible to other datasets and domains, but also typically perform worse than rule-based approaches due to various limitations including over-fitting of models, sparsity of datasets etc. In this work, we propose VeeAlign, a Deep Learning based model that uses a dual-attention mechanism to compute the contextualized representation of a concept in order to learn alignments. By doing so, not only does our approach exploit both syntactic and semantic structure of ontologies, it is also, by design, flexible and scalable to different domains with minimal effort. We validate our approach on various datasets from different domains and in multilingual settings, and show its superior performance over SOTA methods.

Introduction

Ontologies form an integral part of information organization and management. An ontology (Gruber et al. 1993) is a formal description of knowledge defined using a set of concepts and relations between them. Different organizations have different information requirements, and therefore, they follow different nomenclatures (objects and properties) for defining their requirements, resulting in different ontologies for the same underlying data. As a consequence, to integrate and migrate data among applications, it is crucial to first establish correspondences (or mappings) between the vocabularies of their respective ontologies. Ontology Alignment constitutes the task of establishing correspondences between semantically related elements (i.e. classes and properties) from different ontologies.

Ontology alignment task has been extensively studied in the last several years, and the solutions have ranged from simple rule based systems (Faria et al. 2013; Jiang et al. 2016) to ones incorporating external knowledge (Hertling and Paulheim 2012; Alregwawy, Massmann, and Rahmi 2011), and the most recent ones use sophisticated deep learning based systems (Huang et al. 2007; Kolyvakis, Kalousis, and Kiritsis 2018; Wang et al. 2018; Jiménez-Ruiz et al. 2020a). Among all the methods, AgreementMarkerLight (AML) (Faria et al. 2013) has been a consistent top performer on several ontology alignment tasks organized by OAEI. The Ontology Alignment Evaluation Initiative (OAEI) (Euzenat et al. 2011) has played a key role in the benchmarking of different ontological alignment systems by facilitating their comparison on the same basis and the reproducibility of the results. While AML is one of the best performing systems across different tracks in OAEI, it uses handcrafted rules with manually assigned weights and string similarity algorithms along with domain specific knowledge to discover concept alignments. This kind of approach, while useful, has some obvious limitations. Firstly, using string similarity algorithms with minimal focus on context does not address semantic relatedness. Secondly, for every pair of ontologies, a new set of rules and weights may need to be defined, which is often a laborious and time consuming process, thus adversely affecting scalability. Deep Learning (DL) based systems have also been used for the ontology alignment task (Kolyvakis, Kalousis, and Kiritsis 2018; Wang et al. 2018). However, not only do these approaches typically perform worse than rule-based systems, they are also very domain-dependent, with extensive dependability on background knowledge, which, in turn, affects scalability. One of the primary reasons DL architectures use external background knowledge is because of there being a lack of usable training data when it comes to the ontology alignment task. Classification datasets for ontology alignment typically suffer from severe class imbalance and data sparsity, since the number of ground truth alignments is usually several orders smaller than the number of non-alignments. For example, the conference dataset (Zamazal and Svátek 2017) used for experimentation in this paper has 305 similar concept pairs and 122588 dissimilar concept pairs. This data sparsity issue also leads to most standard DL architectures over-fitting and performing poorly. The challenge, therefore, is two fold: a) to use a generic, domain-independent approach to build a training dataset, based solely on the intrinsic semantic and structural information encoded in on-

Copyright © 2021, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.
tologies with no requirement of external knowledge, and b) to train a model on this dataset that strikes the right balance between model expressivity (which is minimal in rigid rule-based systems) and model complexity (which leads to overfitting).

Despite the significant research, ontology alignment still remains a challenging task. Figure 1 provides an illustration highlighting this challenge. The task is to determine the alignment between the concept Attendee in Ontology-1 and the concept Listener in Ontology-2. Current approaches which work on finding the concept similarity or the neighborhood similarity will fail to capture the alignment between these two concepts since neither of them have high similarity. While there is a common term (i.e. Person) between the contexts of the two concepts, there are several terms which are not similar. This example shows that not only is it important to consider the context, it is also important to model the context in such a way such that the relevant portions of the context have higher weights than the irrelevant portions. In this particular example, ancestor nodes should be given higher weights than children and neighbours connected by object and datatype properties. Modelling a concept’s context in a principled manner is one of the limitations of the existing methods that we address in this work. Note that, a concept in an ontology comes along with rich and diverse context (parent, children, properties); and, it is important that an alignment method is able to make use of it in an effective manner. Among existing methods, AML uses hand crafted structural similarity to include contextual significance, while both the deep learning systems, DeepAlign [Kolyvakis, Kalousis, and Kiritsis 2018] and OntoEmma [Wang et al. 2018], do not use ontological context at all.

In this paper, we propose an ontology alignment method, referred to as VeeAlign, that computes mapping between two concepts driven by its context. Our method includes a novel way of modelling context, where the context is split into multiple facets based on the type of neighborhood. More specifically, we divide the context based on its relationship with the central concept such as ancestors, children, object properties and data properties. Such a multi-faceted context, however, poses a new challenge i.e., some of these facets context shall include paths while others will have only neighbouring nodes. In order to deal with this challenge, we propose a dual attention mechanism that comprises of path level attention and node level attention. The path level attention helps find the most important path among all the available paths, whereas the node level attention finds nodes in the path that have the most influence on the central concept alignment. The main contributions of this paper are as follows:

- We model the task of ontology alignment to determine the similarity between two concepts driven by their context. We introduce the notion of multi-faceted context, and model it using a novel attention mechanism, i.e. dual attention.
- We show through an ablation study the effect of dual attention over single attention and no attention, and the effect of different facet types on model performance
- We evaluate the proposed model on four datasets, Conference, Lebensmittel, Freizeit, and Web Directory; and show that using the context improves the performance of ontology alignment task, in particular recall of positive alignments. We choose these datasets to demonstrate the applicability of our approach to diverse data sources in terms of language, domain, and numbers of concepts and concept alignments.

**Approach**

This section presents details of VeeAlign, a Deep Learning based ontology alignment system, that computes contextualized representation of concepts as a function of not just its label, but the multi-faceted neighbours that surround it. In other words, the context is divided into multiple facets based on the relationship between the concept and its neighbours, and then a contextual vector is computed using a dual attention mechanism. This helps the model compute a contextualized representation of a concept, which is later used to compute alignments.

**Preliminaries**

VeeAlign is a Deep Learning based supervised model trained on positive and negative alignments. Let $O^s$ and $O^t$ be the respective source and target ontologies with $\{e_1^s, e_2^s, \ldots, e_n^s\}$ and $\{e_1^t, e_2^t, \ldots, e_n^t\}$ be the corresponding elements in the respective ontologies. An ontology consists of different kind of elements such as classes, subclasses, datatype properties, and object properties. In our problem formulation, we consider all these elements i.e., we discover correspondences between elements from one ontology to the respective elements in another ontology. Ontology alignment in its most general form involves finding different kinds of relationships between elements, including complex relationships such as transformation [Thieblin et al. 2019] or inference [Zhou 2018]. The focus of this work is to discover the equivalence relationship between elements, primarily because equivalence relations is of the most interest to the community. A formal definition of the ontology alignment task is as following:
**Definition 1.** Given a source ontology \( O^s \) and a target ontology \( O^t \), each consisting of a set of elements, the goal of the ontology alignment task is to find all semantically equivalent pairs of elements, i.e. \( \{(e^s, e^t) \in O^s \times O^t : e^s \equiv e^t\} \), where \( \equiv \) indicates semantic equivalence.

We now describe the methodology for finding semantically equivalent concepts i.e. classes and subclasses from the given source and target ontologies.

**Concept Representation**

The deep learning architecture for VeeAlign is described in Figure 2. Since VeeAlign is a supervised model, it requires training data in the form of positive and negative alignment pairs. In other words, given a source and target ontology pair as input, we are given ground truth alignment pairs with their labels, i.e. for each \( (e^s, e^t) \in O^s \times O^t \), we have \( L(e^s, e^t) = 1 \) when \( e^s \equiv e^t \), and 0 otherwise. For concept alignment, the input to the VeeAlign model are candidate concept pairs \( (c^s, c^t) \) along with their labels. Given such a pair as input, a naïve approach could be to find the distributed representations of the elements in the pair and use them for similarity computation. Furthermore, one can use the additional information associated with the concept such as synonym information, description etc for similarity computation. VeeAlign does use label embeddings, but the key difference lies in its method of capturing the context and computing contextualized concept representation, which not only exploits the semantic but also the syntactic and structural nature of ontologies.

**Context Representation**

We believe that context plays a critical role in alignment, therefore, it is important to model the context in a principled manner. Note that, an ontology consists of concepts along with the relationships among concepts, such as "parent-child" subclass-of relationships, datatype properties, object properties etc. VeeAlign is based on computing the distributed representations of both the concept and its context, which are then concatenated and used to compute the probability score of alignment. For a concept \( c_i \), let \( u_i \) be the distributed representation obtained using Universal Sentence Encoder (Cer et al. 2018). In VeeAlign, the neighboring concepts connected to a concept forms its context. Each neighboring concept has a role and has its own influence on the concept alignment, therefore we categorize neighboring concepts into four categories: ancestor nodes, child nodes, nodes connected through a datatype property and nodes connected through an object property.

Sifting through several ontologies and their reference alignments, we observed that two concepts align not just based on their one-hop neighbours, but also on the basis of similarity of "ancestral nodes". In other words, while comparing two concepts, we consider not just their immediate parents, but also the ancestral nodes that lie all the way from the current node to the root node, i.e in the "lineage paths". We thus enumerate all the lineage paths from the current concept to the root and use them for alignments. Let \( a_1, a_2, \ldots \) be all the lineage paths for a given concept.

Each lineage path consists of multiple nodes arranged in a sequence, i.e. \( a_j = [c_{a_j, 1}, c_{a_j, 2}, \ldots, c_{a_j, \ell}] \).

The child nodes consist of concepts which are connected to the current concept using a direct SUBCLASS-OF relationship. In order to follow a consistent terminology, we also represent them as a path, however all the paths will have only one node. The contextual concepts linked to the current node through datatype and object properties are represented in the same manner as child concepts, i.e. we only consider immediate one hop concepts linked to the current concept through either of the properties.

![Figure 2: VeeAlign Architecture](image-url)
Dual Attention

Attention (Bahdanau, Cho, and Bengio 2015) in deep learning can be broadly interpreted as a vector of weights denoting relative importance. For this task, attention computes weight for neighboring concepts that influence the central concept’s alignment. The higher/lower the weight of a concept in a weight vector, the higher/lower the influence in the central concept alignment.

Path-level Attention

The path level attention aims to find the most important paths for each category. This involves performing the aforementioned computations for both source and target ontologies, we do not use context and simply take the representation of the names associated with the property. For a given property \( p \), we denote by \( g(p) \) the representation provide by the embedding layer. Since the context representation \( v_i \) is concatenated with the central concept representation \( u_i \), and the combined representation is input to a linear layer for dimensionality reduction in a lower dimension space, as follows:

\[
f(c_t) = W \ast [u_i, v_i]
\]
aforementioned attention layers) and then computing the confidence score of the alignment by taking a cosine similarity between the two contextualized representations, i.e.

\[
H(e^s, e^t) = \cos(f(e^s), f(e^t)) \\
H(p^s, p^t) = \cos(g(p^s), g(p^t))
\]

(8)

where \((e^s, e^t)\) and \((p^s, p^t)\) are the concept and properties pairs respectively, which we will now denote by elements pair. Finally, an element pair \((e^s, e^t)\) is considered a positive alignment when the similarity score is more than a threshold, i.e. \(L(e^s, e^t) = 1\) when \(H(e^s, e^t) > \Theta\) and 0 otherwise. For the training, we use mean squared error computed as following:

\[
L = \frac{1}{N} \sum_{(e^s, e^t) \in O^s \times O^t} \left( H(e^s, e^t) - L(e^s, e^t) \right)^2
\]

(9)

where \(H(e^s, e^t)\) is obtained using equation [5], and \(N\) is total number of training examples. \(L(e^s, e^t)\) denotes the ground truth label which is 1 if \(e^s \equiv e^t\) and 0 otherwise.

Experiments
In this section, we provide details of the experiments, i.e. the datasets used, baseline models, experimental setup, results and their analysis including ablation study.

Datasets
We evaluate the performance of our model on four benchmark datasets used in several prior studies for the ontology alignment task ((Euzenat et al. 2011) (Peukert, Massmann, and Koenig 2010)). Table 1 shows the number of concepts in each ontology along with the total number of ground truth positive alignments for entire dataset.

| Dataset          | Ontology        | #Concepts | #Ground Truth Alignments |
|------------------|-----------------|-----------|--------------------------|
| conference       | cmt             | 29        | 305                      |
|                  | conference      | 39        |                          |
|                  | connaIF         | 38        |                          |
|                  | edas            | 103       |                          |
|                  | ekaw            | 13        |                          |
|                  | lacted          | 140       |                          |
|                  | sigkdd          | 49        |                          |
| Lebensmittel     | Google          | 58        | 32                       |
|                  | web             | 52        |                          |
| Freizeit         | dmoz            | 70        | 67                       |
|                  | Google          | 66        |                          |
| Web directory    | dmoz            | 745       | 2051                     |
|                  | Google          | 727       |                          |
|                  | web             | 417       |                          |
|                  | Yahoo           | 1132      |                          |

Table 1: Concept and alignment distribution for each dataset

Alignments are meaningful only for an ontology pair, not for a single ontology, therefore we provide the total number of positive alignments available in the entire dataset.

• Conference (Zamazal and Svátek 2017): The OAEI Conference dataset consists of 16 ontologies from the conference organization domain with ground truth alignments provided for 7 of them, resulting in 21 ontology pairs.

• Lebensmittel (Peukert, Massmann, and Koenig 2010): This dataset consists of ontologies from Food domain. The concepts and ground truth alignments are extracted from web directories of Google and the web. Both ontologies are in the German language.

• Freizeit (Peukert, Massmann, and Koenig 2010): Similar to Lebensmittel, this dataset is also in German, and consists of concepts and ground truth alignments pairs extracted from web directories of dmoz and Google, related to online shopping in the Leisure domain.

• Web directory (Massmann and Rahm 2008): This dataset, in German, contains relatively larger ontologies. The four ontologies in this dataset consist of concepts related to online shopping websites that are extracted from dmoz, Google, web and Yahoo web directories.

The datasets were selected in order to be able to fairly evaluate a general purpose domain-independent ontology alignment system that does not use any background knowledge and is also suitable for different languages. Language is an important consideration in ontology alignment, as several applications of the ontology alignment problem such as data integration, data transfer require system to be operable for multiple languages.

Hyperparameters
In our implementation, we use the following hyperparameters optimized through grid-search. The word vectors for each concept were initialized with 512-dimension Universal Sentence Encoder (USE) (Cer et al. 2018) for the confer-

cence dataset and its multilingual variant (Yang et al. 2019) for the 3 German-language datasets respectively. The model was converged using MSE loss and Adam optimizer with a learning rate of 0.001, after training for 50 epochs with a batch size of 32. We experimented with another variation of the model where one-hop properties and children neighbors are represented as one path of length \(L\) created with random sequence, as opposed to \(L\) paths of length 1. For obtaining a unified path representation, we experiment with weighted sum and max pooling, and report the best results. Finally, the dimension of the down-projecting output layer was set to 300. All randomizations including PyTorch, Numpy is done using 0 as the seed. More details on the experimental setup including computing infrastructure is provided in Appendix A.

Data Preprocessing and Evaluation Methodology
Our model takes the positive and negative alignment pairs as input, and since we are only given positive alignment pairs, we construct negative alignment pairs by first creating all the possible pairs from input ontologies and selecting the ones which are not part of the ground truth alignments. The data consisting of the positive and negative pairs is split into training, validation and test sets using the K-fold
Table 2: Performance comparison of VeeAlign with the baseline methods. “-” means that the results could not be obtained either due to inapplicability of the method on the dataset or it not being able to finish under a reasonable time limit. \( P = 0 \) and \( R = 0 \) mean that the algorithm did not output any alignments.

| Dataset          | VeeAlign | AML | LogMap2 | DeepAlign |
|------------------|---------|-----|---------|-----------|
|                  | \( P \) | \( R \) | \( F \) | \( P \) | \( R \) | \( F \) | \( P \) | \( R \) | \( F \) | \( P \) | \( R \) | \( F \) |
| Conference       | 0.774   | 0.741 | 0.748   | 0.802   | 0.651 | 0.7    | 0.821   | 0.654 | 0.701   | 0.631   | 0.586 | 0.567   |
| Lebensmittel     | 0.641   | 0.562 | 0.596   | 0   | 0 | nan    | 0.925   | 0.747 | 0.821   | -       | -    | -       |
| Freizeit         | 0.788   | 0.985 | 0.874   | 0 | 0 | nan    | -       | -    | -       | -       | -    | -       |
| Web directory    | 0.468   | 0.695 | 0.559   | - | - | -      | 0.778   | 0.542 | 0.638   | -       | -    | -       |

Table 3: F-score on individual ontology pairs from Conference dataset

| Pair          | VeeAlign | AML | LogMap2 | DeepAlign |
|---------------|----------|-----|---------|-----------|
| cmt-ekaw      | 0.545    | 0.6 | 0.632   | 0.666     |
| cmt-sigkidd   | 0.8      | 0.818 | 0.957 | 0.78      |
| conf-sigkidd  | 0.69     | 0.692 | 0.759 | 0.615     |
| edas-tasted   | 0.667    | 0.667 | 0.519 | 0.47      |
| ekaw-tasted   | 1        | 0.737 | 0.7   | 0.46      |
| cmt-confOf    | 0.643    | 0.583 | 0.455 | 0.364     |
| cmt-tasted    | 0.889    | 0.8   | 0.889 | 0.615     |

Table 4: Effect of incorporating context using single and dual attention

| Context Type    | \( P \) | \( R \) | \( F \) |
|-----------------|--------|--------|--------|
| No Context      | 0.775  | 0.608  | 0.670  |
| Context + single attention | 0.678  | 0.728  | 0.697  |
| Context + dual attention     | 0.774  | 0.741  | 0.748  |
| Context           | P    | R    | F    |
|------------------|------|------|------|
| Parents          | 0.747| 0.707| 0.719|
| Children         | 0.634| 0.75  | 0.678|
| Object Properties| 0.647| 0.74  | 0.681|
| Data properties  | 0.64  | 0.75  | 0.681|
| All combined     | 0.774| 0.741 | 0.748|

Table 5: Effect of different types of context

Effect of Context and Dual Self Attention  We perform an ablation study and analyze the effect of different layers on model performance, in particular when there is no context, using context with single attention (not using path level information) and and using context with dual attention i.e., using both path and node level information. The results are shown in Table 4. These results indicate that the dual attention is an improved model over single attention which is an improved model over when there is no context, demonstrating the efficacy of context and of modeling it using dual attention. It is worth noting that modeling the context using dual attention gives a significant improvement in recall and F-score, primarily because the model considers a richer set of information for alignment, but at the same time does not reduce precision because information is used based on its importance for the alignment task.

Effect of Context Type  Another dimension to analyze the model performance is along the lines of using different types of context information i.e., parent, children, data properties and object properties. The results from this analysis are show in Table 5. These results indicate that parents are the most useful type of context information whereas the children are of the least. However the best alignment results are obtained when we combined all four types of context, this combination gives us a performance improvement in both recall and precision.

Related Work

There has been a large body of work on the ontology alignment problem (Euzenat, Shvaiko et al. 2007; Otero-Cerdeira, Rodríguez-Martínez, and Gómez-Rodríguez 2015; Niepert, Melícke, and Stuckenschmidt 2010; Schumann and Lécue 2015), primarily driven by the OAEI (Ontology Alignment Evaluation Initiative). OAEI has been conducting ontology alignment challenges since 2004 where multiple datasets belonging to different domains are released along with a public evaluation platform to evaluate different systems. Among all the systems submitted to the challenge, two systems have consistently outperformed the others.

The first is AgreementMakerLight (AML) (Faria et al. 2013) which uses a combination of various matching algorithms called matchers such as lexical matcher reflecting the lexical similarities between the entities, structural matcher which compares ontology concepts or their instances based on their relationships with other concepts or instances. The recent AML system (Santos et al. 2015) also include a repair algorithm that minimizes the incoherence of the resulting alignment and the number of matches removed from the input alignment. The second best performing system is LogMap2 (Jiménez-Ruiz et al. 2020b) that is specifically designed to align large scale ontologies. The system works in an iterative manner, starting from the initial anchors, alternates mapping repair and mapping discovery steps. Both of these systems have been heavily engineered over the years to give the best performance on the datasets and domains available in OAEI. However when applied on the datasets other than OAEI, their performance is rather less impressive. These observations have been supported by the experimental study as well. Furthermore these systems are highly domain specific requiring specialized knowledge in terms of lexicons.

The ontology alignment community has only recently started to look into statistical methods, in particular, Deep Learning based methods where models are trained on the given input data. However, these systems are typically even more domain-specific, and require background knowledge in order to train. For instance, a recent work by Wang et al. (2018) presents a neural network based ontology alignment system for the Biomedical domain. The idea is to enrich entities in an ontology with aliases from the ontology, definitions from Wikipedia and context from background knowledge sources, and use this additional information for ontology alignment. Similarly, DeepAlign (Kolyvakis, Kalousis, and Kiritsis 2018) too requires synonyms and antonyms extracted from external sources such as WordNet and PPDB in order to refine the word vectors using synonymy and antonymy constraints, which are subsequently used for alignment. Such dependencies can have severe limitations when dealing with, say, multilingual ontologies, or small generic ontologies where no background knowledge is available. In addition, they typically perform worse than rule-based systems, and even more so when tested on other domains.

In contrast to these methods, VeeAlign does not require any external background knowledge. It completely relies on the semantic and structural information encoded in an ontology, in particular the contextual information available with entities to learn better representations.

Conclusion

In this paper we have presented a general purpose ontology alignment method that does not require any external or background knowledge. The method is based on a deep learning architecture where context is modeled explicitly, by first dividing it into different categories based on its relationship with the concept, and then applying a novel dual attention method. The dual attention helps focus on the parts of the context which are most important for the alignment. Our experiments on several datasets from two languages show that the method outperforms the state of the art method by a significant margin. Our ablation study examining the effect of context splitting and dual attention show that these are indeed the factors behind the performance improvement.
References

Algergawy, A.; Massmann, S.; and Rahm, E. 2011. A clustering-based approach for large-scale ontology matching. In East European Conference on Advances in Databases and Information Systems, 415–428. Springer.

Bahdanau, D.; Cho, K.; and Bengio, Y. 2015. Neural Machine Translation By Jointly Learning To Align And Translate. ICLR.

Bromley, J.; Guyon, I.; LeCun, Y.; Säckinger, E.; and Shah, R. 1994. Signature verification using a ”siamese” time delay neural network. In Advances in neural information processing systems, 737–744.

Cer, D.; Yang, Y.; Kong, S.-y.; Hua, N.; Limtiaco, N.; John, R. S.; Constant, N.; Guajardo-Cespedes, M.; Yuan, S.; Tar, C.; et al. 2018. Universal sentence encoder. arXiv preprint arXiv:1803.11175.

Euzenat, J.; Meilicke, C.; Stuckenschmidt, H.; Shvaiko, P.; and Trojahn, C. 2011. Ontology alignment evaluation initiative: six years of experience. In Journal on data semantics XV, 158–192. Springer.

Euzenat, J.; Shvaiko, P.; et al. 2007. Ontology matching, volume 18. Springer.

Faria, D.; Pesquita, C.; Santos, E.; Palmonari, M.; Cruz, I. F.; and Couto, F. M. 2013. The agreementmakerlight ontology matching system. In OTM Confederated International Conferences “On the Move to Meaningful Internet Systems”, 527–541. Springer.

Gruber, T. R.; et al. 1993. A translation approach to portable ontology specifications. Knowledge acquisition 5(2): 199–220.

Hertling, S.; and Paulheim, H. 2012. WikiMatch: using wikipedia for ontology matching. Ontology Matching 946.

Huang, J.; Dang, J.; Vidal, J. M.; and Huhns, M. N. 2007. Ontology matching using an artificial neural network to learn weights. In IJCAI workshop on semantic Web for collaborative knowledge acquisition, volume 106.

Jiang, S.; Lowd, D.; Kafle, S.; and Dou, D. 2016. Ontology matching with knowledge rules. In Transactions on Large-Scale Data-and Knowledge-Centered Systems XXVIII, 75–95. Springer.

Jiménez-Ruiz, E.; Agibetov, A.; Chen, J.; Samwald, M.; and Cross, V. 2020a. Dividing the Ontology Alignment Task with Semantic Embeddings and Logic-based Modules. arXiv preprint arXiv:2003.05370.

Jiménez-Ruiz, E.; Agibetov, A.; Chen, J.; Samwald, M.; and Cross, V. 2020b. Dividing the Ontology Alignment Task with Semantic Embeddings and Logic-based Modules. arXiv preprint arXiv:2003.05370.

Jiménez-Ruiz, E.; Grau, B. C.; Zhou, Y.; and Horrocks, I. 2012. Large-scale Interactive Ontology Matching: Algorithms and Implementation. In ECAI, volume 242, 444–449.

Kolyvakis, P.; Kalousis, A.; and Kiritsis, D. 2018. 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), 787–798.

Massmann, S.; and Rahm, E. 2008. Evaluating Instance-based Matching of Web Directories. In WebDB.

Niepert, M.; Meilicke, C.; and Stuckenschmidt, H. 2010. A Probabilistic-Logical Framework for Ontology Matching. In AAAI, 1413–1418. Citeseer.

Otero-Cerdeira, L.; Rodríguez-Martínez, F. J.; and Gómez-Rodríguez, A. 2015. Ontology matching: A literature review. Expert Systems with Applications 42(2): 949–971.

Paulus, R.; Xiong, C.; and Socher, R. 2017. A Deep Reinforced Model for Abstractive Summarization. arXiv preprint arXiv:1705.04304.

Peukert, E.; Massmann, S.; and Koenig, K. 2010. Comparing similarity combination methods for schema matching. INFORMATIK 2010. Service Science–Neue Perspektiven für die Informatik. Band 1.

Santos, E.; Faria, D.; Pesquita, C.; and Couto, F. M. 2015. Ontology alignment repair through modularization and confidence-based heuristics. PloS one 10(12): e0144807.

Schumann, A.; and Lécué, F. 2015. Minimizing user involvement for accurate ontology matching problems. In Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence, 1576–1582.

Thiéblin, E.; Haemmerlé, O.; Hernandez, N.; and Trojahn, C. 2019. Survey on complex ontology matching. Semantic Web (Preprint): 1–39.

Zamazal, O.; and Svátek, V. 2017. The ten-year ontofarm and its fertilization within the onto-sphere. Journal of Web Semantics 43: 46–53.

Zhou, L. 2018. A Journey From Simple to Complex Alignment on Real-World Ontologies. In DC@ ISWC, 93–101.