NLFOA: Natural Language Focused Ontology Alignment

Florian Schneider
florian.schneider-1@uni-hamburg.de
Universität Hamburg
Hamburg, Germany

Sarthak Dash
Sugato Bagshi
Nananda Mihindukulasooriya
Alfio Gliozzo
sdash@us.ibm.com
bagchi@us.ibm.com
nandana@ibm.com
gliozzo@us.ibm.com
IBM Research
NY, USA

ABSTRACT

For Ontology Alignment (OA), the task is to align semantically equivalent concepts and relations from different ontologies. This task plays a crucial role in many downstream tasks and applications in academia and industry. Since manually aligning ontologies is inefficient and costly, numerous approaches exist to do this automatically. However, most approaches are tailored to specific domains, are rule-based systems or based on feature engineering, and require external knowledge. The most recent advances in the field of OA rely on the widely proven effectiveness of pre-trained language models to represent the human-generated language that describes the entities in an ontology. However, these approaches additionally require sophisticated algorithms or Graph Neural Networks to exploit an ontology's graphical structure to achieve state-of-the-art performance. In this work, we present NLFOA, or Natural Language Focused Ontology Alignment, which purely focuses on the natural language contained in ontologies to process the ontology's semantics as well as graphical structure. An evaluation of our approach on common OA datasets shows superior results when finetuning with only a small number of training samples. Additionally, it demonstrates strong results in a zero-shot setting which could be employed in an active learning setup to reduce human labor when manually aligning ontologies significantly.

ACM Reference Format:
Florian Schneider, Sarthak Dash, Sugato Bagshi, Nananda Mihindukulasooriya, and Alfio Gliozzo. 2023. NLFOA: Natural Language Focused Ontology Alignment. In *Knowledge Capture Conference 2023 (K-CAP '23), December 05–07, 2023, Pensacola, FL, USA.* ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/3587259.3627560

1 INTRODUCTION

An ontology is an efficient, reusable, and machine-readable way to represent knowledge as nodes representing real-world concepts and relations between these nodes in a graph structure. Since the rise of the semantic web, ontologies have been widely utilized in academia and industry for various applications. However, there are often multiple ontologies representing knowledge of the same domain created by different people or organizations. This poses a problem because the ontologies are likely to be different, i.e., the entities in the ontology may have different names and descriptions or are connected via different relationships, or the ontologies may be of different granularity and size. Ontology Alignment (OA) is the task of identifying semantically equivalent concepts or relations from different ontologies. This task is crucial in facilitating semantic interoperability between systems by creating a homogeneous data model, which increases the performance of methods in numerous research areas or applications, such as natural language processing, information retrieval, business intelligence, and analytics, bio-informatics, or cybersecurity [3, 6, 8, 21].

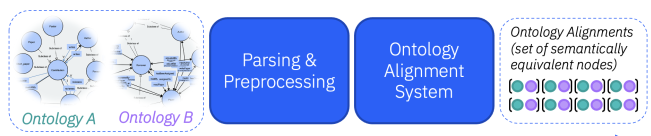

Figure 1: An illustration of the general flow of an Ontology Alignment System.

Since manually aligning ontologies is costly and inefficient, ontology alignment is an active research area with a long history and many different approaches ranging from rule-based systems and machine learning systems that employ manually engineered features [25] to systems based on modern deep learning techniques. The most recent advances in the field of OA rely on the widely proven effectiveness of pre-trained language models [1] to represent the human-generated language that describes the entities in an ontology [10, 14, 27]. However, most of these approaches are tailored to specific domains, utilize external knowledge, and require significant pretraining or finetuning. Further, they additionally require sophisticated algorithms or Graph Neural Networks to exploit an ontology's graphical structure to achieve state-of-the-art performance. The general flow of an Ontology Alignment system is illustrated in Figure 1.

Our proposed method, Natural Language Focused Ontology Alignment (NLFOA), presented in this paper, purely focuses on the

K-CAP '23, December 05–07, 2023, Pensacola, FL, USA

ACM ISBN 979-8-4007-0141-2/23/12…$15.00
https://doi.org/10.1145/3587259.3627560

natural language contained in ontologies to process the ontology's semantics as well as graphical structure. We do so by linearizing a node with its local neighborhood in a pseudo-sentence using a reusable and highly flexible algorithm and computing a representation thereof using a sentence encoder model. In the pseudo-sentence, we use keywords or hints for the encoder model to distinguish between comprised structural and semantical information by extending the tokenizer and embedding layer with special tokens. An evaluation of our approach on common OA datasets from the Ontology Alignment Evaluation Initiative[1] outperforms the current state-of-the-art in a finetuned setting with only a small number of training samples and epochs. Additionally, NLFOA demonstrates strong and competitive results in a zero-shot setting which could be employed in an active learning setup to reduce human labor when manually aligning ontologies significantly.

Moreover, this paper provides an overview of the current state of the art in ontology alignment, presents details on our purely human language-based ontology alignment system, and discusses its limitations.

The main contributions of this paper are:

- We introduce a highly flexible and reusable algorithm to represent semantic and graphical information of nodes and relations in pseudo-sentences.
- We introduce NLFOA, a general ontology alignment system that purely utilizes natural language processing techniques to compute meaningful representations of concepts and relations and can be finetuned with only a small number of training examples.
- We evaluate NLFOA against current state-of-the-art on common datasets and prove the effectiveness of our approach in finetune and zero-shot settings.
- We released the code, all configurations, and pre-trained models for NLFOA on GitHub upon acceptance.

1.1 Problem Statement

Formally, the problem of ontology alignment is defined as follows. There exist two ontologies O_S and O_T, referred to as source and target ontology, each containing a set of concepts and relations $E_S = \{\forall e_S \in O_S\}$ and $E_T = \{\forall e_T \in O_T\}$, respectively. An ontology alignment function $f : E_S \times E_T \mapsto A$ outputs an injective mapping or alignment $A = \{(e_S, e_T) \in E_S \times E_T \mid e_S \equiv e_T\}$ which is the set of pairs of semantically equivalent concepts or relations from the two input ontologies. Note that A is an injective alignment, i.e., it only contains 1:1 mapping between O_S and O_T.

2 RELATED WORK

The following overview of existing ontology alignment systems (OAS) divides the systems into traditional systems that follow a feature engineering or rule-based approach and modern approaches that rely on deep learning techniques that rely on static or contextual word embeddings. Traditional OAS such as LogMap [15] or AML [4] but also recent systems like ATBox [12] or Matcha [5] implement sophisticated multi-stage iterative algorithms based on features from logical reasoning about the ontology structure and lexical string-matching methods for the ontology semantics. Additionally, these systems incorporate background knowledge to increase performance. Auxiliary data structures such as inverted indices, lexica, or optimized graphs are used to improve computational efficiency.

More recent OAS, such as DeepAlignment [19], OntoEmma [30], or Rafcom [22], utilize deep learning methods and replace lexical string-matching with semantic similarity metrics using static word embeddings for the labels and descriptions of nodes contained in the ontologies. Since ontologies often lack descriptive labels or descriptions, all named systems utilize external knowledge to improve performance. Further, they train classifiers on top of the word embeddings or finetune the word embeddings to fit the task better. Finally, the systems rely on graph algorithms incorporating the ontology structure to ensure high-quality and logically possible alignments.

Static word embeddings were broadly replaced by contextualized word embeddings computed by transformer language models [28] due to their significant improvements in various natural language processing tasks. Hence, the most recent ontology alignment systems are based on BERT embeddings [1] to increase performance further [2, 7, 10, 10, 14, 18, 27, 32]. However, these system either require background knowledge, domain-specific language models, or still employ sophisticated multi-stage or iterative algorithms to preprocess or postprocess to find the optimal set of alignments for a specific pair of ontologies. Further, they often rely on graph neural networks to compute knowledge graph embeddings which have to be trained and sometime specially configured to fit the ontology alignment task. Another limitation of these systems is that they do not make full use of the rich semantic and graphical information contained in ontologies described in natural language although they are utilizing BERT-based language models.

Other BERT-based approaches utilize Sentence Transformers [24] to effectively pool multiple word embeddings to compute a single semantically rich vector representation for a node in an ontology [13, 17, 20, 29, 31]. We follow a similar approach in this work. However, as opposed to these system in our NLFOA system, we not only make use of a node's labels, comments, or descriptions but also include its graphical structure, i.e., the local neighborhood as well as properties in a recursive manner. Further, to enrich the nodes' representations, we add and learn semantic indicator tokens to indicate the type of information a particular word embedding holds, e.g., we have keywords to indicate that specific tokens are related to the parent node. Moreover, we do not rely on sophisticated pre- or post-processing methods or subsystems to preselect candidates or repair output mappings.

3 NLFOA

As the name of this work's approach to ontology alignment, Natural Language Focused Ontology Alignment (NLFOA), suggests, we concentrate on human-generated language contained in ontologies and process them using state-of-the-art natural language processing techniques. While the structure or the graph of an ontology is defined in OWL, the semantics, i.e., names and descriptions of concepts, relations, and properties, are typically described using natural language. Hence, current ontology alignment approaches

[1] http://oaei.ontologymatching.org/

have implemented the idea of using pretrained language models in various ways. However, to incorporate the ontology structure, most of these systems rely on traditional graph algorithms, logical reasoning, or graph neural networks.

The main difference in NLFOA is that we process both ontology semantics and graph structure using pre-trained language models to compute dense vector representations of concepts and relations. We do so by linearizing the properties and the local neighborhood of a concept or relation in a pseudo sentence which is forwarded through a Sentence Transformer to compute a dense vector representation. To compute the semantic similarity of two concepts or relations in two different ontologies, we calculate the cosine similarity of their vector representations. An overview of the NLFOA is shown in Figure 2. In the following, the three main steps are explained in more detail.

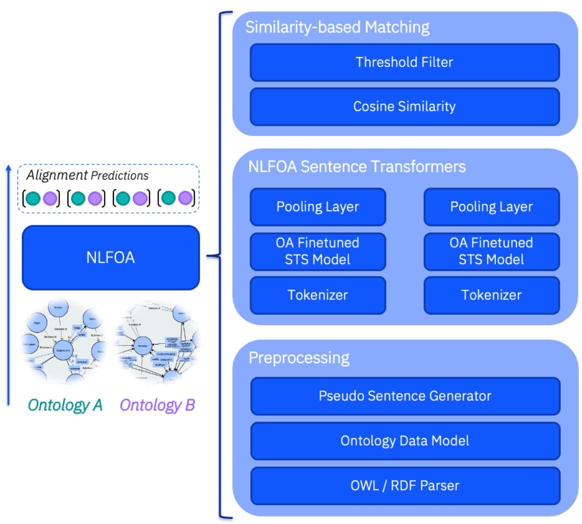

Figure 2: An overview of the Natural Language Focused Ontology Alignment (NLFOA) system introduced in this work.

3.1 Pseudo Sentence Generation

The first step to generate a pseudo sentence $S(e)$ for a concept or relation e in an ontology O is to parse the ontology defined in an OWL file. Therefore we use the Python libraries *owlready2*[2] and *rdflib*[3].

In the second step, from the information extracted during parsing, we build an internal data model of the ontology to better access and transform the information needed in subsequent steps. A concept (`owl:Class`) or relation (`owl:ObjectProperty`, `owl:FunctionalProperty`, `owl:DatatypeProperty`) in our data model is represented by a Python class with its identifier (IRI), names (`rdfs:label`), and descriptions (`rdfs:comment`) as members. Further, the hierarchy of the concept or relation, i.e., the parents and children (`rdfs:subClassOf`), are stored. For relations, we also store the range (`rdfs:range`), the domain (`rdfs:domain`), and the inverse relation (`owl:inverseOf`). For concepts, we store all incoming and outgoing properties and relations, i.e., relations where the target concept is either in the range or domain. An exam-

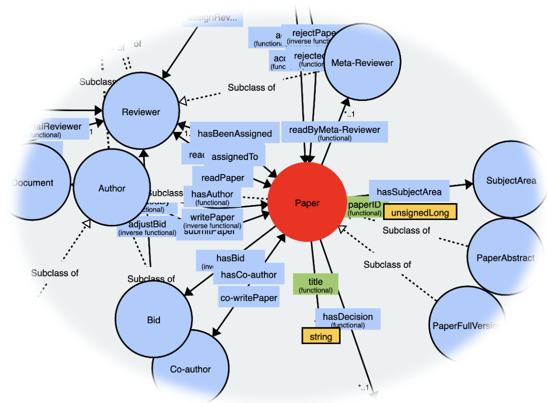

Figure 3: An illustration of an node in an ontology representing the concept of a paper within the domain of scientific conferences, including its properties and relations to neighboring concepts.

ple of an ontology node representing the concept of a paper within the domain of scientific conferences, including its properties and relations to neighboring concepts is illustrated in Figure 3.

In the third step, we generate the pseudo sentence with a highly flexible and configurable algorithm, the Pseudo Sentence Generator (PSG). A pseudo sentence, in our case, is a semi-structured list of keys and values described by words to represent a concept or a relation in an ontology. E.g., the pseudo sentence "`Concept: Name:` *Paper* `. Parents:` *Document* `.`" represents a concept with the name "Paper" that has a parent called "Document". In the example, the monospaced text indicates the keys and delimiters, whereas the italic text represents the values. A more complex example of a pseudo sentence that represents a concept is shown in Figure 4.

The input to the PSG is a target concept or relation e from our internal ontology data model, along with a configuration that specifies what and how information is contained in the output pseudo sentence $S(e)$. Since the detailed explanation of all configurable options is out of the scope of this paper, we briefly describe the most important ones. It is possible to configure:

- which keys to include (e.g., `Name`, `Parents`, `Children`, ...)
- the number of hops of parents and children taken into account
- what information to include recursively for `Parents`, `Children`, or the `Domain` and `Range` of properties
- the structure of the pseudo sentence, i.e., the ordering of the keys (e.g., `Name`, `Parents`, `Children` vs `Parents`, `Name`, `Children`)
- the name of the keys and the delimiter symbols

3.2 Embedding Computation

We interpret the pseudo sentence embedding $\mathfrak{R}(e)$ computed by a Sentence Transformer model as the representation of the respective concept or relation e in the Ontology O. Because we do not want to restrict NLFOA from only working with ontologies from specific

[2]https://owlready2.readthedocs.io/
[3]https://rdflib.readthedocs.io/

```
Target Concept:
Name: Paper .
Parents:
    Document
        Parents: Thing .
Children:
    Paper Abstract
        Children: No Childred .,
    Paper Full Version
        Children: No Childred .
        .
Incoming Properties: co-write Paper , reject Paper ,
read Paper , mark Conflict Of Interest , has Been
Assigned , submit Paper , accept Paper , write Paper
.
Outgoing Properties:  paper ID , title , has Decision
, read By Reviewer , read By Meta-Reviewer , has
Co-author , rejected By , has Bid , assigned To , has
Subject Area , accepted By , has Author .
```

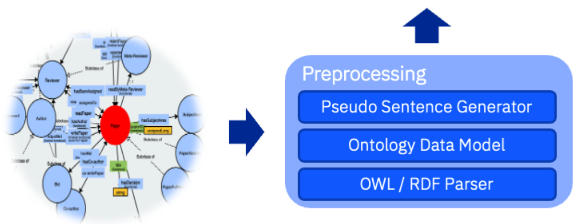

Figure 4: An example of a pseudo sentence generated from a concept in the ontology. The keywords and characters highlighted in blue are treated as special tokens. Note that the sentence is formatted for better readability.

domains but all kinds of different domains, we prefer a general-purpose pretrained language model over a domain-specific used by other BERT-based ontology alignment approaches [10, 32]. Further, since the objective is to compare how semantically similar two concepts or relations are, we employ an S-BERT [24] model pretrained on semantic textual similarity (STS). The *sentence-transformers/all-mpnet-base-v2*[4] fits these requirements and is the best-performing sentence transformer for STS as of the date of writing this, which is why the model was selected as starting point for experiments. The model outputs a 768-dimensional dense vector representation of the pseudo sentence, resulting from a mean pooling operation of all token embeddings produced by the MPNet [26] backbone model.

3.3 Alignment Predictions

To measure the semantic similarity between two concepts or relations e_S and e_t in the Source Ontology O_S and the Target Ontology O_T, we first compute the embeddings of their respective pseudo-sentences $\mathfrak{R}(e_S)$ and $\mathfrak{R}(e_T)$ using the model described above. We then compute the cosine similarity of the two embeddings to obtain a similarity score.

$$S(e_S, e_T) = \frac{\mathfrak{R}(e_S) \cdot \mathfrak{R}(e_T)}{\|\mathfrak{R}(e_S)\| \, \|\mathfrak{R}(e_T)\|}$$

To obtain a candidate set of alignments between O_S and O_T, we first compute the similarity of all elements in the Cartesian product of the elements in the two ontologies.

$$O_S \times O_T = \{(e_S, e_T) \mid \forall e_S \in O_S, \forall e_T \in O_T\}$$

[4] https://huggingface.co/sentence-transformers/all-mpnet-base-v2

We then apply a threshold filter to get the final set of alignments.

4 EXPERIMENTS

In this section the experiments conducted to evaluate our proposed ontology alignment approach.

4.1 Involved Datasets

We evaluate and compare the performance of our system on two commonly known datasets from the Ontology Alignment Evaluation Initiative 2022 (OAEI). The OAEI Conference dataset[5] is a dataset containing multiple ontologies representing knowledge from the domain of scientific conferences [33]. Further, it provides reference alignment between pairs of seven contained ontologies. The OAEI Anatomy dataset[6] contains two ontologies, one describing the adult human anatomy and one describing the mouse anatomy [9], and reference alignments between the two ontologies. Since there are no predefined train, test, and validation splits and explicit negative samples for the OAEI Conference and Anatomy datasets, we describe our data splitting and generation process in the respective experiment sections. Table 1 provides an overview of the dataset statistics and some example alignments.

4.1.1 Experiment Data Generation. Since both datasets do not provide predefined train, test, and validation splits, nor do they provide explicit negative samples, we first describe the data generation process in the following. For the OAEI Conference dataset, we merged the 21 provided reference alignments, which we refer to as positive samples or positives. The total number of positives is 610, which can be thought of as a table with 610 rows and two columns containing the IRIs of the entities of the two aligned ontologies, respectively. The negatives are sampled by shifting the right column by one, i.e., each element is in the $i+1$-th row. Thus, we have the same number of negatives as positives in the resulting dataset referred to as $Conf$ In our experiments, we also use datasets containing twice as many negatives than positives by shifting the right column once more, referred to as $Conf_{2N}$. The train, test, and validation splits are 65%, 20%, and 15% of the shuffled datasets, respectively.

The data generation process for the OAEI Anatomy dataset is the same. However, there is only one pair of reference alignments between the two contained ontologies. The resulting datasets are referred to as Ana and Ana_{2N}. The proportion of samples of the

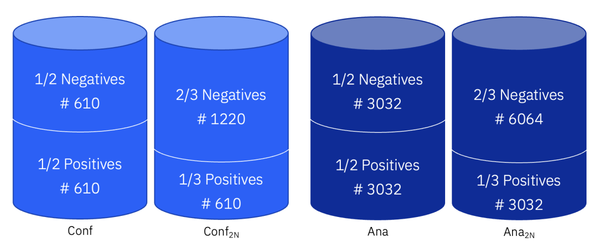

Figure 5: Proportion of negative and positive samples contained in the datasets used in the experiments.

[5] http://oaei.ontologymatching.org/2022/conference/index.html
[6] http://oaei.ontologymatching.org/2022/anatomy/index.html

Name	Domain	#Classes	#Reference Alignments	Reference Alignment Examples
Conference	Scientific Conferences	491	610	• ProgrammComitteeChair – Chair_PC • SubjectArea – Topic • hasAuthor – writtenBy • title – hasTitle • Author – Author • writesPaper – writes
Anatomy	Human and Mouse Anatomy	6048	1516	• posterior limiting lamina – Descemt s Membrane • vaginal hymen – Hymen • coat hair bulb – Hair Bulb • kidney collecting duct – Collecting Tube • patella – Patella • cuboid – Cuboid Bone

Table 1: Statistics and example reference alignments of the OAEI 2022 datasets used in the experiments. The shown examples are the `rdfs:label` of the concepts or relations.

Name	Positives	Negatives	Total
$Conf$	610	610	1220
$Conf_{2N}$	610	1220	1830
Ana	3032	3032	6064
Ana_{2N}	3032	6064	9096

Table 2: Statistics of the generated datasets used in the NLFOA finetuning experiments.

generated datasets are illustrated in Figure 5, statistics are listed in Table 2.

4.2 Experiment 1: Finetuned Ontology Alignment

In this experiment, we evaluate NLFOA on the OAEI Conference and Anatomy datasets in a setting where we finetune the NLFOA core model using the train splits of the generated datasets described in Section 4.1.1. The core model was initialized with the pretrained STS Sentence Transformer described in Section 3. We use the Sentence Transformer framework[7] to finetune the model because it allows rapid development and supports all necessary functionality. During our experiments, we tried several configurations for the Pseudo Sentence Generator (PSG) component of NLFOA. The reported results are achieved using pseudo-sentences, including the name, label, and description of a concept or relation, parents and children in a 2-hop neighborhood, and incoming and outgoing properties. The keywords in the pseudo-sentences are set to proper English terms or phrases. Further, we used a dynamic pseudo-sentence generation strategy to randomize the generation process so that the structure or ordering of a pseudo-sentence (PS) gets shuffled. With this strategy, we force the model to learn the semantics of the special tokens used in a PS to provide hints for the model on how to interpret the contained information. Additionally, we added the keywords contained in a pseudo-sentence as special tokens by extending the tokenizer and embedding layer of the backbone model. During finetuning the model then learns these new embeddings.

[7]https://www.sbert.net/

For all experiments, the NLFOA core model was finetuned for 10 epochs on the Ana and $Conf$ or Ana_{2N} and $Conf_{2N}$ train sets using an A100 GPU with 80GB. However, the required amount of computation, memory, and time is small, so the experiments should easily run on modern consumer hardware. Further, we found that NLFOA finetuned on the datasets containing twice the number of positives than negatives performed better in all experiments.

4.2.1 Results. As reported in Table 3 and Table 4, the finetuned NLFOA achieves perfect and almost perfect scores for the Ana_{2N} and $Conf_{2N}$ test sets, respectively, outperforming all systems evaluated in the OAEI 2022 Conference and Anatomy track. However, it has to be pointed out that blind test sets were used to evaluate the systems reported in Table 3 and Table 4. Thus, the results are not directly comparable, but since the blind test sets are in the same domain, i.e., contain concepts and relations of ontologies of the same domain, the reported results still demonstrate the superior performance of NLFOA.

System	Precision	Recall	F-1 Score	Test Set
Matcha	0.95	0.93	0.941	OAEI
SEBMatcher	0.95	0.87	0.908	OAEI
LogMapBio	0.87	0.92	0.895	OAEI
LogMap	0.92	0.85	0.881	OAEI
AMD	0.95	0.82	0.88	OAEI
ALIN	0.98	0.75	0.852	OAEI
LogMapLite	0.96	0.73	0.828	OAEI
ATMatcher	0.98	0.67	0.794	OAEI
StringEquiv	1.0	0.62	0.766	OAEI
LSMatch	0.95	0.63	0.761	OAEI
ALIOn	0.36	0.93	0.462	OAEI
KERMIT	0.97	0.70	0.81	OAEI
NLFOA (ours)	**1.0**	**1.0**	**1.0**	Ana_{2N}

Table 3: NLFOA and OAEI Systems Results for the Anatomy ontologies. For more details on the OAEI evaluation refer to [23].

System	Precision	Recall	F-1 Score	Test Set
LogMap	0.76	0.56	<u>0.64</u>	OAEI
GraphMatcher	0.75	0.55	0.63	OAEI
SEBMatcher	0.79	0.48	0.60	OAEI
ATMatcher	0.69	0.51	0.59	OAEI
ALIN	0.82	0.44	0.57	OAEI
edna	0.74	0.45	0.56	OAEI
LogMapLt	0.68	0.47	0.56	OAEI
AMD	0.82	0.41	0.55	OAEI
LSMatch	<u>0.83</u>	0.41	0.55	OAEI
StringEquiv	0.76	0.41	0.53	OAEI
KGMatcher+	<u>0.83</u>	0.38	0.52	OAEI
ALIOn	0.66	0.19	0.30	OAEI
TOMATO	0.09	<u>0.60</u>	0.16	OAEI
Matcha	0.37	0.07	0.12	OAEI
NLFOA (ours)	**0.96**	**1.0**	**0.98**	$Conf_{2N}$

Table 4: NLFOA and OAEI Systems Results for the Conference ontologies. For more details on the OAEI evaluation refer to [23].

4.3 Experiment 2: Zero-Shot Ontology Alignment

In this experiment, we test the zero-shot capabilities of our approach. Therefore we initialized the NLFOA core model with the pretrained STS Sentence Transformer described in Section 3. To generate the pseudo-sentences for concepts and relations, we used a similar PSG configuration as described in the finetuned experiment setting (see Section 4.2). However, we did not add the keywords as special tokens but used a more verbose description in natural language. This is because we do not finetune NLFOA for this experiment and, therefore, cannot learn the embeddings of the keywords if we had added them as special tokens. The system was then evaluated on the complete set of provided reference alignments in the OAEI Conference and Anatomy datasets.

4.3.1 Results. The results of the zero-shot experiments are reported in terms of classical metrics for ontology alignment in Table 5. Since we perform a similarity search when predicting the set of alignments, in Table 6, we also report results regarding typical information retrieval metrics. As can be observed, the NLFOA zero-shot setting achieves competitive results compared to the performance of state-of-the-art systems shown in Table 3 and Table 4. The results further confirm the effectiveness of our approach, i.e., the pseudo-sentence generation and embedding strategy to encode both an ontologies semantics and graph structure in dense vector representations for concepts and relations computed by a pretrained sentence transformer. This might be especially interesting for industrial applications since training an ontology alignment system from scratch, defining domain-specific rules, or engineering domain-specific features is costly or even infeasible for smaller companies. However, since we evaluated the experiment only on two different domains, further experiments involving more ontologies of widespread and different domains have to be conducted to finally and confidently asses NLFOA's zero-shot capabilities.

System	Dataset	Precision	Recall	F-1 Score
KERMIT	Anatomy	0.31	0.93	0.46
NLFOA (ours)	Anatomy	0.57	0.61	0.58
KERMIT	Conference	-	-	-
NLFOA (ours)	Conference	0.48	0.48	0.48

Table 5: NLFOA zero-shot experiment results for the OAEI Conference and OAEI Anatomy datasets.

Dataset	H@1	H@5	H@10	MRR
Anatomy	0.71	0.84	0.87	0.77
Conference	0.80	0.91	0.94	0.85

Table 6: NLFOA zero-shot experiment results in terms of information retrieval metrics for the OAEI Conference and OAEI Anatomy datasets.

4.4 Experiment 3: Cross Zero-Shot Ontology Alignment

In this experiment, we test the capabilities of NLFOA when finetuning it on ontologies of one domain and evaluating it on ontologies of a different domain, i.e., in a cross zero-shot setting. This might be interesting in an industrial environment where companies could pretrain NLFOA on already possessed or open-source ontologies and use it in a zero-shot setting for unseen ontologies from new customers. Therefore we finetune NLFOA on the train split of the $Conf_{2N}$ dataset and evaluate it on reference alignments of the Anatomy dataset and vice versa using the Ana_{2N} train split and Conference reference alignments. For this experiment, we use the same finetuning parameters and PSG configuration as in the experiment described in Section 4.2.

4.4.1 Results. The results of the cross zero-shot experiments are reported in terms of classical metrics for ontology alignment in Table 7. From the results, we can observe that the cross-zero shot setting only works in one direction. That is, it improves performance by 0.1 absolute F1 when training on the $Conf_{2N}$ dataset and evaluating on the Anatomy dataset but drastically reduces performance when training on the Ana_{2N} and evaluating on the Conference dataset. This drastic performance decrease might be because the sentence encoder model suffered from catastrophic forgetting when finetuning on the much larger Anatomy dataset due to its contained domain-specific medical language. See Table 1 for some example concept names contained in the dataset. This line of reasoning also explains why performance was increased in the other setting. That is, when finetuning on the $Conf_{2N}$ dataset containing more general terms and language (see Table 1), the model can learn to understand the semantic and structural hints provided by the special tokens, which is beneficial when evaluating on the Anatomy dataset.

5 LIMITATIONS

One central component of the proposed NLFOA approach is the Pseudo Sentence Generator, which linearizes nodes in an ontology graph into semi-structured pseudo-sentences. The length of

Train	Test	Precision	Recall	F-1 Score
$Conf_{2N}$	Anatomy	0.76	0.61	0.68
Ana_{2N}	Conference	0.13	0.04	0.06

Table 7: NLFOA cross zero-shot experiment results for the OAEI Conference and OAEI Anatomy datasets.

the pseudo-sentences, i.e., the number of comprised characters, depends on the semantic information used to describe the nodes that the ontology's human creators express in the form of natural language. Another parameter that influences the pseudo-sentence length is the number of hops considered, i.e., the nodes of the local neighborhood of the target node that are also recursively described in the sentence. The issue that arises with the length of the pseudo-sentence is that the number of tokens that can be forwarded through the sentence encoder language model to compute a dense vector representation is limited. Although most BERT-based models support 512 tokens, which correspond to approximately 400 English words, pseudo-sentences of nodes with a large local neighborhood or with long `rdfs:comment` can be problematic.

Another limitation of NLFOA is that the system computes pairwise similarities for the full Cartesian product of the sets of nodes of two ontologies leading to a the computational complexity $O(N*M)$, where N and M is the number of nodes in the two ontologies, respectively. For large ontologies, e.g., as contained in the Bio-ML dataset [11], this becomes excessively costly regarding the required computational resources and memory consumption. Although there exist approximate nearest neighbor approaches for billion-scale similarity search [16], it would still be infeasible to compute for a pair of two ontologies with each more than 32000 concepts and relations. For this reason, we did not evaluate our system in the OAEI Bio-ML track[8], which is another limitation of this work.

6 CONCLUSION

In this paper, we introduced NLFOA, an effective and general ontology alignment system that does not rely on manually created or domain-specific rules or features and does not rely on background knowledge. Further, NLFOA does not employ complex graph algorithms or graph neural networks but linearizes the semantic information as well as the graphical structure of concepts or relations in an ontology in pseudo-sentences and purely uses natural language processing techniques to compute meaningful representations. In our experiments, we showed that NLFOA outperforms current state-of-the-art systems on common datasets in a finetuned setting by a large margin and achieves competitive results in a zero-shot setting. Finally, we discussed the main limitations of our system in detail.

REFERENCES

[1] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics*. Minneapolis, MN, USA, 4171–4186.

[2] Sefika Efeoglu. 2023. GraphMatcher: A Graph Representation Learning Approach for Ontology Matching: Results for OAEI 2022. In *CEUR Workshop Proceedings*, Vol. 3324. Hangzhou, China (virtual), 174–180.

[8] https://www.cs.ox.ac.uk/isg/projects/ConCur/oaei/index.html

[3] Vadim Ermolayev and Maxim Davidovsky. 2012. Agent-based Ontology Alignment: Basics, Applications, Theoretical foundations, and Demonstration. In *Proceedings of the 2nd International Conference on Web Intelligence, Mining and Semantics*. Craiova, Romania, 1–12.

[4] Daniel Faria, Catia Pesquita, Emanuel Santos, Matteo Palmonari, Isabel F Cruz, and Francisco M Couto. 2013. The AgreementMakerLight Ontology Matching system. In *OTM Confederated International Conferences 'On the Move to Meaningful Internet Systems'*. 527–541.

[5] Daniel Faria, Marta Contreiras Silva, Pedro Cotovio, Patrícia Eugénio, and Catia Pesquita. 2022. Matcha and Matcha-DL Results for OAEI 2022. In *CEUR Workshop Proceedings*.

[6] J Gargantilla and A Gomez-Perez. 2004. OntoWeb: A Survey on Ontology-based Applications. *OntoWeb Consortium IST Project IST-2000-29243, Deliverable* 1 (2004).

[7] Francis Gosselin and Amal Zouaq. 2023. SEBMatcher Results for OAEI 2022. In *CEUR Workshop Proceedings*, Vol. 3324. Hangzhou, China (virtual), 202–209.

[8] Junheng Hao, Chuan Lei, Vasilis Efthymiou, Abdul Quamar, Fatma Özcan, Yizhou Sun, and Wei Wang. 2021. Medto: Medical Data to Ontology Matching using Hybrid Graph Neural Networks. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. Washington DC, USA, 2946–2954.

[9] Terry F Hayamizu, Sherri de Coronado, Gilberto Fragoso, Nicholas Sioutos, James A Kadin, and Martin Ringwald. 2012. The Mouse-Human Anatomy Ontology Mapping Project. *Database: The Journal of Biological Databases and Curation* 2012 (2012), bar066.

[10] Yuan He, Jiaoyan Chen, Denvar Antonyrajah, and Ian Horrocks. 2022. BERTMap: A BERT-based Ontology Alignment System. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 36. 5684–5691.

[11] Yuan He, Jiaoyan Chen, Hang Dong, Ernesto Jiménez-Ruiz, Ali Hadian, and Ian Horrocks. 2022. Machine Learning-Friendly Biomedical Datasets for Equivalence and Subsumption Ontology Matching. In *The Semantic Web – ISWC 2022*. Hangzhou, China (virtual), 575–591.

[12] Sven Hertling and Heiko Paulheim. 2021. ATBox Results for OAEI 2021. In *CEUR Workshop Proceedings*, Vol. 3063. 137–143.

[13] Sven Hertling, Jan Portisch, and Heiko Paulheim. 2022. KERMIT - A Transformer-Based Approach for Knowledge Graph Matching. *CoRR* abs/2204.13931 (2022).

[14] Vivek Iyer, Arvind Agarwal, and Harshit Kumar. 2021. VeeAlign: Multifaceted Context Representation Using Dual Attention for Ontology Alignment. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Online and Punta Cana, Dominican Republic, 10780–10792.

[15] Ernesto Jiménez-Ruiz and Bernardo Cuenca Grau. 2011. LogMap: Logic-based and Scalable Ontology Matching. In *Proceedings of the 10th International Conference on The Semantic Web (ISWC) – Volume Part I*. Bonn, Germany, 273–288.

[16] Jeff Johnson, Matthijs Douze, and Hervé Jégou. 2019. Billion-scale similarity search with GPUs. *IEEE Transactions on Big Data* (2019).

[17] Peter Kardos, Zsolt Szántó, and Richárd Farkas. 2023. WomboCombo Results for OAEI 2022. In *CEUR Workshop Proceedings*, Vol. 3324. Hangzhou, China (virtual), 216–219.

[18] Leon Knorr and Jan Portisch. 2022. Fine-TOM Matcher Results for OAEI 2021. In *CEUR Workshop Proceedings*, Vol. 3063. online, 144–151.

[19] Prodromos Kolyvakis, Alexandros Kalousis, and Dimitris Kiritsis. 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 (NAACL-HLT), Volume 1 (Long Papers)*. New Orleans, LA, USA, 787–798.

[20] Daniel Kossack, Niklas Borg, Leon Knorr, and Jan Portisch. 2022. TOM Matcher Results for OAEI 2021. In *CEUR Workshop Proceedings*, Vol. 3063. online, 193–198.

[21] Weiwei Lin and Reiko Haga. 2021. Matching Cyber Security Ontologies through Genetic Algorithm-Based Ontology Alignment Technique. *Security and Communication Networks* 2021 (2021).

[22] Ikechukwu Nkisi-Orji, Nirmalie Wiratunga, Stewart Massie, Kit-Ying Hui, and Rachel Heaven. 2019. Ontology Alignment based on Word Embedding and Random Forest Classification. In *Joint European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML-PKDD)*. Dublin, Ireland, 557–572.

[23] Mina Abd Nikooie Pour, Alsayed Algergawy, Patrice Buche, Leyla J Castro, Jiaoyan Chen, Hang Dong, Omaima Fallatah, Daniel Faria, Irini Fundulaki, Sven Hertling, Yuan He, Ian Horrocks, Martin Huschka, Liliana Ibanescu, Ernesto Jimenez-Ruiz, Naouel Karam, Amir Laadhar, Patrick Lambrix, Huanyu Li, Ying Li, Franck Michel, Engy Nasr, Heiko Paulheim, Catia Pesquita, Tzanina Saveta, Pavel Shvaiko, Cassia Trojahn, Chantelle Verhey, Mingfang Wu, Beyza Yaman, Ondrej Zamazal, and Lu Zhou. 2023. Results of the Ontology Alignment Evaluation Initiative 2022. In *CEUR Workshop Proceedings*, Vol. 3324. Hangzhou, China (virtual), 84–128.

[24] Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. Hong Kong, China, 3973–3983.

[25] Pavel Shvaiko and Jérôme Euzenat. 2011. Ontology Matching: state of the art and future challenges. *IEEE Transactions on Knowledge and Data Engineering* 25,

1 (2011), 158–176.
[26] Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. 2020. MPNet: Masked and Permuted Pre-Training for Language Understanding. In *Advances in Neural Information Processing Systems (NeurIPS)*, Vol. 33. online, 16857–16867.
[27] Xiaobin Tang, Jing Zhang, Bo Chen, Yang Yang, Hong Chen, and Cuiping Li. 2020. BERT-INT:A BERT-based Interaction Model For Knowledge Graph Alignment. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence (IJCAI)*. Yokohama, Japan, 3174–3180.
[28] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention Is All You Need. In *Advances in Neural Information Processing Systems (NIPS)*, Vol. 30. Long Beach, CA, USA, 5998–6008.
[29] Javier Vela and Jorge Gracia. 2023. Cross-Lingual Ontology Matching with CIDER-LM: Results for OAEI 2022. In *CEUR Workshop Proceedings*, Vol. 3324. Hangzhou, China (virtual), 158–165.
[30] Lucy Lu Wang, Chandra Bhagavatula, Mark Neumann, Kyle Lo, Chris Wilhelm, and Waleed Ammar. 2018. Ontology Alignment in the Biomedical Domain Using Entity Definitions and Context. In *Proceedings of the BioNLP 2018 Workshop*. Melbourne, Australia, 47–55.
[31] Zhu Wang. 2023. AMD Results for OAEI 2022. In *CEUR Workshop Proceedings*, Vol. 3324. Hangzhou, China (virtual), 145–152.
[32] Jifang Wu, Jianghua Lv, Haoming Guo, and Shilong Ma. 2020. DAEOM: A Deep Attentional Embedding Approach for Biomedical Ontology Matching. *Applied Sciences* 10, 21 (2020), 7909.
[33] Ondej Zamazal and Vojtch Svtek. 2017. The Ten-Year OntoFarm and its Fertilization within the Onto-Sphere. *Web Semantics: Science, Services and Agents on the World Wide Web* 43, C (2017), 46–53.