Representation Learning in Ontology Matching

Cássia Trojahn, Guilherme Sousa

IRIT & Université de Toulouse 2 Jean Jaurès, Toulouse, France
cassia.trojahn@irit.fr, guilherme.santos-sousa@irit.fr

Séminaire INRAE, SESAME - SEmantic web SeminAr MontpEllier, 11 mars 2024

1 postdoc

- **Khadija Jradeh** *Instance matching*. ANR DACE-LD project (*DAta-CEntric AI-driven Data Linking*), IRIT/LIRMM (04/2022-04/2024).

4 PhD students

- **Julien Breton** *Legal knowledge extraction from text*. CIFRE, UPS/Berger-Levrault (2021-2024).
- **Antoine Dupuy** *Meteorological Observation ontologies and Contextual Knowledge for final User Policies*. ADI/Region, IRIT/CNRM (2022-2025).
- **Guilherme Sousa** *Representation Learning and Complex Ontology Matching*. Ministry of Higher Education grant. IRIT (2022-2025).
- **Soline Felice** *Open memory referential*. ADI/Region. IRIT/CLLE (2023-2026).

Ontology matching

Representation learning: embeddings, language models

Embeddings in OM

Language models in OM

Expressive alignments

Ontology matching

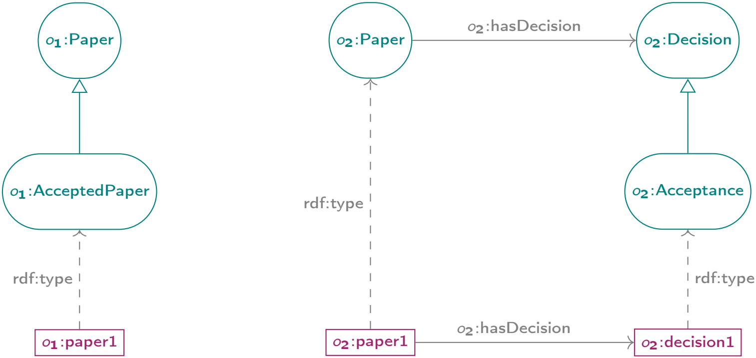
Representation learning: embeddings, language models

Embeddings in OM

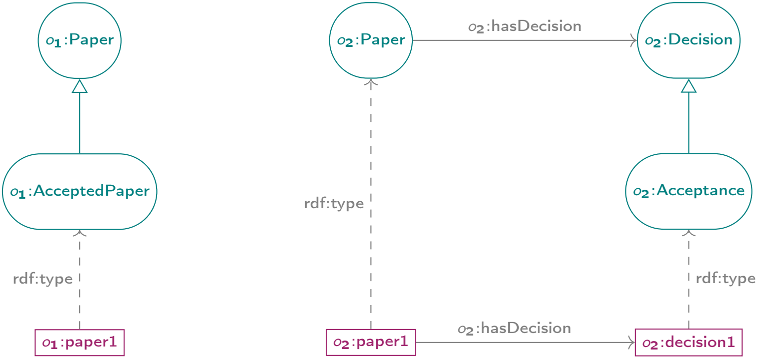
Language models in OM

Expressive alignments

Ontology heterogeneity	Ontology differences in terms of the terminology, coverage, granularity modelling strategies, or still level of generality
Ontology matching	Task of generating a set of correspondences between different ontologies

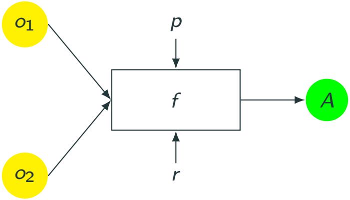


$$o_1\text{:Paper} \quad \equiv \quad o_2\text{:Paper}$$



$$o_1\text{:Paper} \equiv o_2\text{:Paper}$$

$$o_1\text{:AcceptedPaper} \equiv o_2\text{:Paper} \sqcap \exists o_2\text{:hasDecision}.o_2\text{:Acceptance}$$



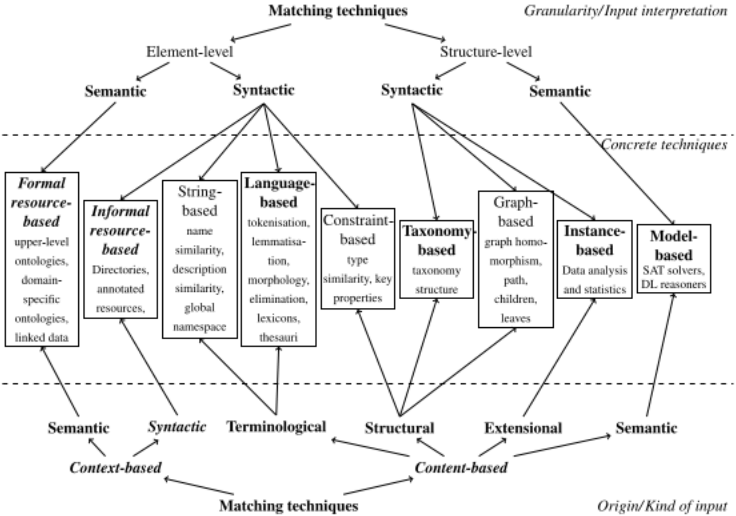
Adapted from [Euzenat and Shvaiko, 2013]

A is a set of **correspondences** $\{c_1, ..., c_n\}$, where c_i is a tuple (e_1, e_2, r)
e_1 and e_2 are the members of the correspondence.

A is a set of **correspondences** $\{c_1, ..., c_n\}$, where c_i is a tuple (e_1, e_2, r)
e_1 and e_2 are the members of the correspondence:

- **simple** correspondence (s:s): e_1 and e_2 are simple expressions
  $(o_1\texttt{:Paper}, o_2\texttt{:Paper}, \equiv)$
- **complex** correspondence (s:c, c:s, c:c): e_1 or/and e_2 is a complex expression
  $(o_1\texttt{:AcceptedPaper},\ \exists\ o_2\texttt{:Paper} \sqcap o_2\texttt{:hasDecision}.o_2\texttt{:Acceptance}, \equiv)$
- r is a relation, e.g., $(\equiv, \sqsupseteq, \sqsubseteq, \perp)$

Classification of matching approaches



Jérôme Euzenat, Pavel Shvaiko: Ontology Matching, Second Edition. Springer 2013, ISBN 978-3-642-38720-3, pp. I-XVII, 1-511

Plan

Ontology matching

Representation learning: embeddings, language models

Embeddings in OM

Language models in OM

Expressive alignments

Exploring the **natural language layer** of ontologies in a better way.

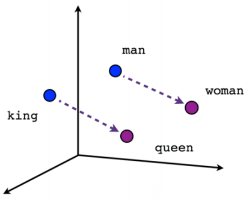
Textual descriptions play more than ever an increasingly important role in OM.

Embeddings and language models: both aim at representing text in a numerical way

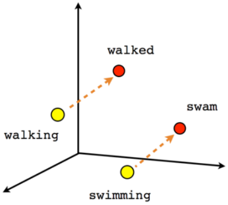
- different purposes
- different stages of the NLP pipeline

Embeddings

- A way to transform words/phrases into numerical vectors
- Capture semantic and contextual information about the text
- Primarily focused on capturing the meaning of singular words or phrases
- Used as feature vectors in downstream NLP tasks

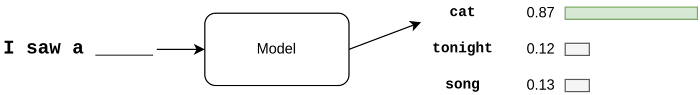


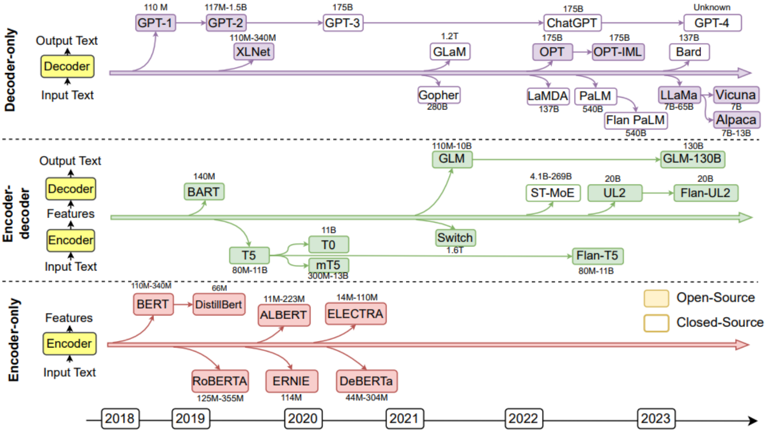
Male-Female



Verb tense

- Predict the next word in a sentence based on the context extracted from the previous words
- Estimate the probability of one or more words given the surrounding words
- Generate coherent and contextually relevant text, pre-trained over a large corpus of data
- Useful in more sophisticated tasks not only requiring contextual semantic meaning but also text generation





Unifying Large Language Models and Knowledge Graphs: A Roadmap Shirui Pan, Linhao Luo, Yufei Wang, Chen Chen, Jiapu Wang, Xindong Wu in IEEE Transactions on Knowledge and Data Engineering, https://github.com/RManLuo/Awesome-LLM-KG

Ontology matching

Representation learning: embeddings, language models

Embeddings in OM

Language models in OM

Expressive alignments

A wave of representation learning systems has appeared in the last few years.

- Word embeddings (Word2Vec, Glove, fastText)
- Sentence embeddings (BERT, SBERT)
- RDF embeddings (RDF2Vec)
- OWL constraints (OWL2Vec)
- Graph embeddings (GNN, GCN, GAT, RGCN)
- Translational embeddings (TransE, TransR, RotatE)

Word2Vec

- Word2Vec [Mikolov et al., 2013] was one of the first embedding methods used in ontology matching.
- Best used when concepts are labeled with only one word.
- It is commonly pre-trained with data outside the ontology and performs better if the trained dataset is in the same topic as the ontology being used.

Word2Vec

Word2Vec embeddings are trained by reading a text corpus with a sliding window and using words and context to create a training dataset.

Example (skip-gram training): The current word (green) is used to predict the context words (yellow) by increasing their similarity while decreasing the similarity of random words.

Window Size	Text	Skip-grams
2	[The **wide** road shimmered] in the hot sun.	wide, the wide, road wide, shimmered
	The [wide road **shimmered** in the] hot sun.	shimmered, wide shimmered, road shimmered, in shimmered, the
	The wide road shimmered in [the hot **sun**].	sun, the sun, hot

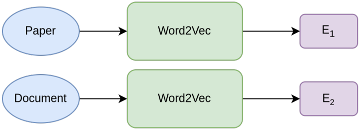
Figure 1: Word2Vec training example. https://www.tensorflow.org/text/tutorials/word2vec

In [Zhang et al., 2014] Word2Vec is applied in ontology matching for the first time.

The matcher uses the model for each entity in the source ontology by using the textual information to retrieve an embedding based on each work in this textual information.

The embeddings are used to compare the similarity between entities using metrics such as cosine similarity.

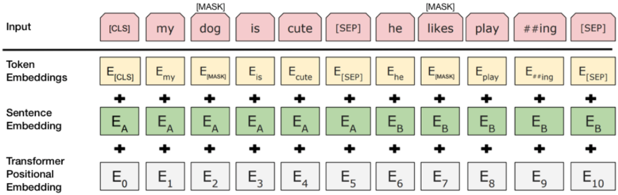
The pairs with higher similarity are selected as a correspondence.



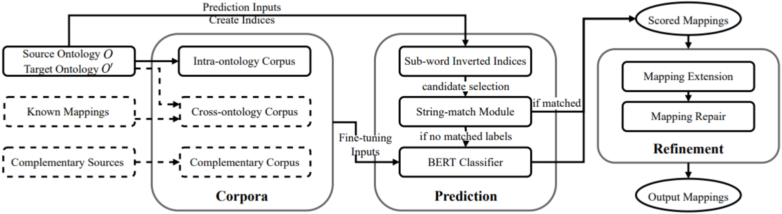
$$f(E_1, E_2) = 0.87$$

Pre-trained transformer encoder model used for language understanding tasks [Devlin et al., 2018].

BERT will read a sequence of words in both directions, from left to right and right to left. This makes it possible to grasp more complex phrases and turn much closer to human language.

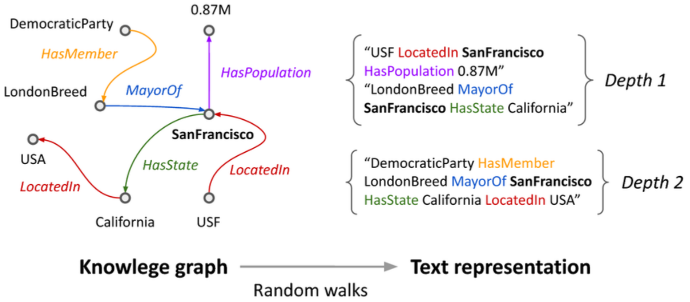


BERT is used as a binary classifier to decide whether two entities are the same using their labels as input [He et al., 2022].



Embedding method used to generate vector representations of RDF graphs: RDF2vec creates a numeric vector for each node in an RDF graph.

First, a random walk strategy to create sequences of RDF nodes is applied, which are then used as input for the word2vec algorithm.

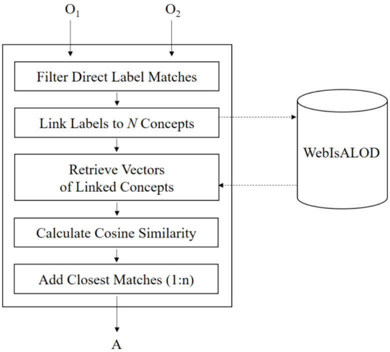


http://rdf2vec.org/

RDF2Vec is used to generate embeddings of an external knowledge graph (WebIsALOD) [Portisch et al., 2020]

Embeddings are then retrieved from the WebIsALOD embeddings at runtime using the concept labels.

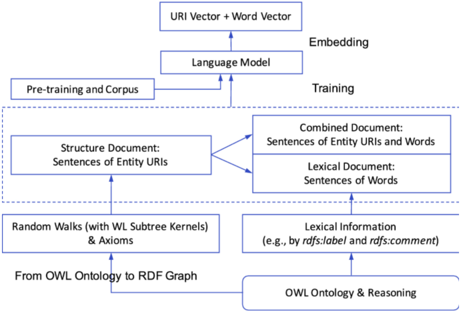
The cosine similarity between entity pair embeddings is computed to find the correspondences between source and target ontologies similar as applied in Word2Vec approaches.



OWL2Vec [Chen et al., 2020] is an extension to RDF2Vec to generate embeddings of OWL ontologies.

It has a set of conversion rules to transform an OWL ontology into an RDF graph and apply random walk strategies to generate the sentences used to train the model.

This model is used in the LogMapML [Chen et al., 2021] matcher.



Graph Neural Networks (GNN) ([Wu et al., 2020] for a survey) are networks capable of generating representations of graph data.

This type of model fits well in the OM since it can be used to generate embeddings that contain contextual information from each node.

This model is often combined with language models to generate the initial embeddings that are then contextualized by the GNN.

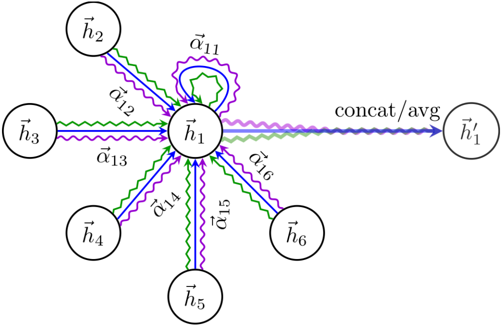
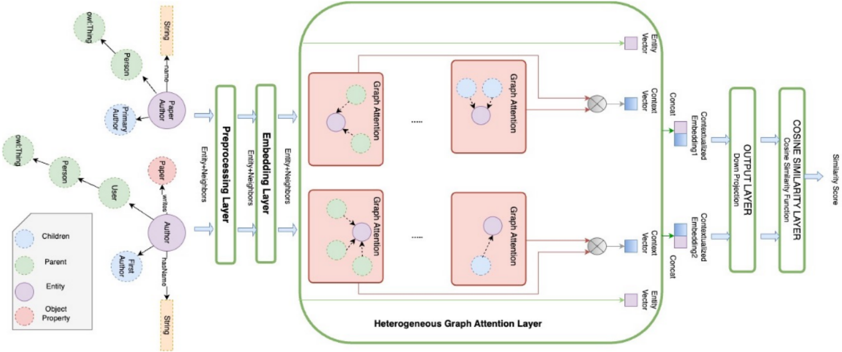
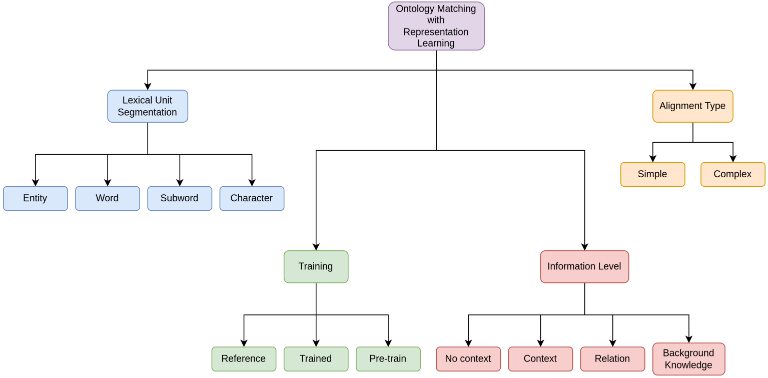


Figure 2: Veličković, Petar, et al. "Graph attention networks." arXiv preprint arXiv:1710.10903 (2017).

Uses a graph attention approach to compute higher-level representation of a class together with its surrounding terms [Efeoglu, 2023].



Works can be grouped based on how the information is used to create the embedding representation [Sousa et al., 2022]:



Lexical Unit Segmentation groups the different types of tokenization used in processing the textual information related to ontology entities to generate their respective embeddings:

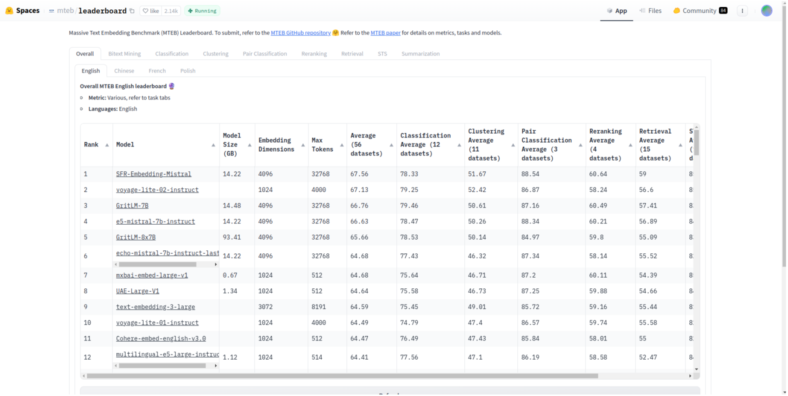
- **Entity-level** segmentation considers the whole sentence as a single entity without requiring aggregation steps to compute the final embeddings. Ex. ['AcceptedPaper']
- **Word-level** segmentation considers the individual words present in the sentence as elements that are aggregated to generate the final entity embedding. Ex. ['Accepted', 'Paper'].
- **Subword-level** segmentation considers subword splits that are also aggregated to generate the final entity embedding. This improves entity generalization as it can generate embeddings for words not present in the training corpus if the subwords are present. Ex. ['Accep', 'ted', 'Pa', 'per'].
- **Character-level** segmentation has one embedding for each character model alphabet resulting in a reduced vocabulary size and a higher generalization compared to the other types of segmentation since embedding for any word can be generated by aggregating the embedings of the characters. Ex. ['A', 'c', 'c', 'e', 'p', 't', 'e', 'd', 'P', 'a', 'p', 'e', 'r'].

Training category groups the different types of data usage related to the system training.

- **Reference** are those approaches that depend on reference alignments to be trained. Reference alignments are the gold standard set of alignments between two ontologies.
- **Trained** are those approaches that adapt their weights for each new ontology it sees and some training happens steps before the matching of each pair of ontologies.
- **Pre-train** are those systems that are trained before the matching and in a different dataset than the others being evaluated.

Information Level category groups the levels of information aggregated in each embedding generated by the matching approach:

- **No Context** are those approaches that only use information present in each entity without considering the nodes near the one being generated. For example, matches that generate embedding only considering entity labels fall in this category.
- **Context** are those approaches that use information present in each entity and also aggregate information of neighbor entities in any depth without considering the relation predicate between them. In this category are grouped the approaches that use labels of the current entity and nearby entities in the embedding generation.
- **Relation** are those systems that also include the information present in the relations between entities, for example, considering that two entities are disjoint or subclasses of each other and inserting that information in the embedding of the current entity.
- **Background Knowledge** are those systems that include information not present in the ontology in the entity embedding generation. For example, fetching data from Wikipedia and inserting the information found there into the entity's final embedding.


Spaces   mteb/leaderboard   like 2.14k   Running   App   Files   Community 84

Massive Text Embedding Benchmark (MTEB) Leaderboard. To submit, refer to the MTEB GitHub repository 🤗 Refer to the MTEB paper for details on metrics, tasks and models.

Overall   Bitext Mining   Classification   Clustering   Pair Classification   Reranking   Retrieval   STS   Summarization

English   Chinese   French   Polish

Overall MTEB English leaderboard

- Metric: Various, refer to task tabs
- Languages: English

Rank	Model	Model Size (GB)	Embedding Dimensions	Max Tokens	Average (56 datasets)	Classification Average (12 datasets)	Clustering Average (11 datasets)	Pair Classification Average (3 datasets)	Reranking Average (4 datasets)	Retrieval Average (15 datasets)	S A (d
1	SFR-Embedding-Mistral	14.22	4096	32768	67.56	78.33	51.67	88.54	60.64	59	8
2	voyage-lite-02-instruct		1024	4000	67.13	79.25	52.42	86.87	58.24	56.6	8
3	GritLM-7B	14.48	4096	32768	66.76	79.46	50.61	87.16	60.49	57.41	8
4	e5-mistral-7b-instruct	14.22	4096	32768	66.63	78.47	50.26	88.34	60.21	56.89	8
5	GritLM-8x7B	93.41	4096	32768	65.66	78.53	50.14	84.97	59.8	55.09	8
6	echo-mistral-7b-instruct-last	14.22	4096	32768	64.68	77.43	46.32	87.34	58.14	55.52	8
7	mxbai-embed-large-v1	0.67	1024	512	64.68	75.64	46.71	87.2	60.11	54.39	8
8	UAE-Large-V1	1.34	1024	512	64.64	75.58	46.73	87.25	59.88	54.66	8
9	text-embedding-3-large		3072	8191	64.59	75.45	49.01	85.72	59.16	55.44	8
10	voyage-lite-01-instruct		1024	4000	64.49	74.79	47.4	86.57	59.74	55.58	8
11	Cohere-embed-english-v3.0		1024	512	64.47	76.49	47.43	85.84	58.01	55	8
12	multilingual-e5-large-instruc	1.12	1024	514	64.41	77.56	47.1	86.19	58.58	52.47	8

Figure 3: https://huggingface.co/spaces/mteb/leaderboard

Where/how we are using embeddings

- PropMatch [Sousa et al., 2023a]: generation of alignments between ontology properties, combining word and sentence embeddings with alignment extension.
- Using BERT Models to Automatically Classify Domain Concepts into DOLCE Top-Level Concepts: A Study of the OAEI Ontologies [Sousa et al., 2023b].
- Extending CANARD (more later).
- Combining BERTInst and Link keys.

Ontology matching

Representation learning: embeddings, language models

Embeddings in OM

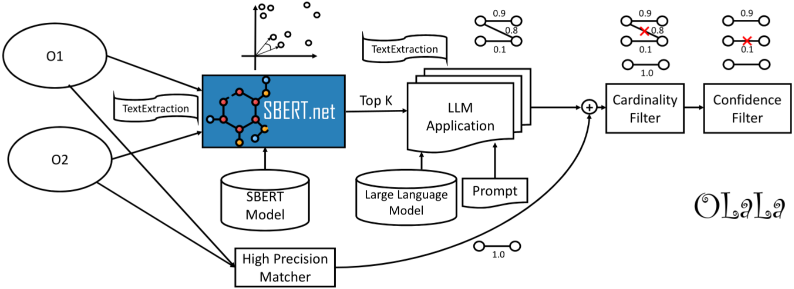
Language models in OM

Expressive alignments

Recent adoption in OM, few papers already adopting them

- OLaLa: Ontology Matching with Large Language Models [Hertling and Paulheim, 2023]
- ChatGPT for entity matching [Peeters and Bizer, 2023]: check if two product descriptions refer to the same product.
- Conversational ontology alignment with [Norouzi et al., 2023]
- Capabilities of large models for biomedical concept linking [Wang et al., 2023]

OLaLa: Ontology Matching with Large Language Models
[Hertling and Paulheim, 2023] is an ontology matching approach applying LLMs in the similarity computation between a pair of entities.



1. Matching candidates are extracted from the two input ontologies O1 and O2.
2. Those selected entities are presented to the LLM with specific prompts to decide their correctness.
3. High-precision matches are also used to include the simple matches in the final alignment.
4. Filters are applied to ensure alignment requirements such as ensuring a 1:1 mapping.

The LLM usage in OLaLa is applied in two different configurations:

- **Binary Decisions** presents the candidate entities to the LLM and uses the generated response to decide about their similarity:
  "Classify if the following two concepts are the same. First concept: (left) Second concept: (right)
  Answer:". To get the final result, the model response is searched for words like true/yes for positive and false/no for negative.
- **Multiple choice decisions** more context is provided to the LLM such as giving a source entity and all possible target entities. Then, the task is to pick the one that represents the same entity or to generate a default answer.
  "The task is ontology matching (find the description that refers to the same real-world entity). Which of the following descriptions fits best to this description: (left)? (candidates)
  Answer with the corresponding letter or "none" if no description fits. Answer:"

LLMs: challenges

Need for training data (fine-tuning/reference alignments)

Validation

Open source issues (hidden data training, full access to the model)

Reproducibility

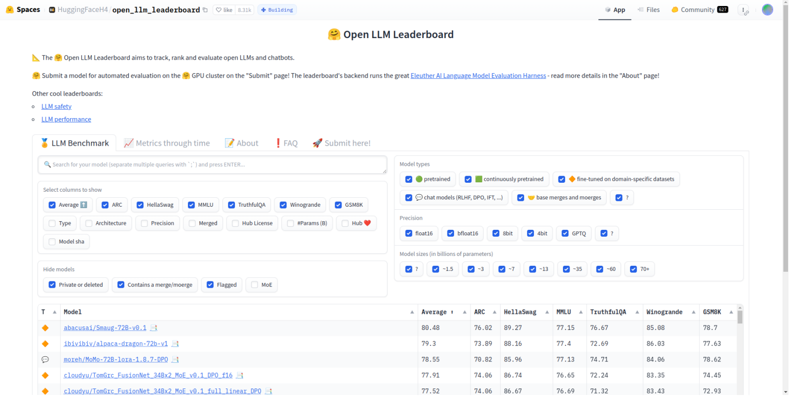


Figure 4: https://huggingface.co/spaces/HuggingFaceH4/open_llm_leaderboard

Plan

Ontology matching

Representation learning: embeddings, language models

Embeddings in OM

Language models in OM

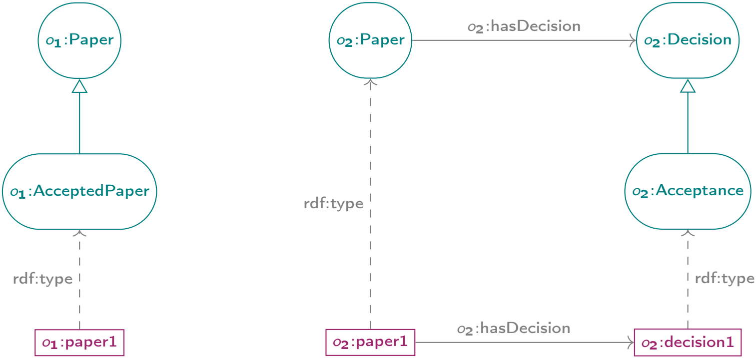
Expressive alignments

Need for complex correspondences

Simple correspondences are not expressive enough
to overcome the different kinds of ontology heterogeneity

Alignments between real-world ontologies contain many
relations uncovered by current systems

Need for more expressiveness in diverse domains and applications



$$o_1\text{:Paper} \equiv o_2\text{:Paper}$$

$$o_1\text{:AcceptedPaper} \equiv o_2\text{:Paper} \sqcap \exists o_2\text{:hasDecision}.o_2\text{:Acceptance}$$

Proposal

- Higher search space for generating complex correspondences
- User needs are neglected in most matching approaches

- Reduce the matching space taking into account user's knowledge needs
  $\rightarrow$ **Competency Questions for Alignment**

Competency questions for alignment (CQAs)

Same as competency questions for ontology authoring [Suárez-Figueroa et al., 2012], but to be answered over two or more ontologies.

Competency questions for alignment (CQAs)

Same as competency questions for ontology authoring [Suárez-Figueroa et al., 2012], but to be answered over two or more ontologies.

Can be a NL question or SPARQL queries.

- "What are the accepted papers?"
- `SELECT ?x WHERE {?x a` o_1`:AcceptedPaper.}`
- `SELECT ?x WHERE {?x` o_2`:hasDecision ?y.  ?y a` o_2`:Acceptance.}`

Competency questions for alignment (CQAs)

Same as competency questions for ontology authoring [Suárez-Figueroa et al., 2012], but to be answered over two or more ontologies.

Can be a NL question or SPARQL queries.

- “What are the accepted papers?”
- `SELECT ?x WHERE {?x a` o_1`:AcceptedPaper.}`
- `SELECT ?x WHERE {?x` o_2`:hasDecision ?y.  ?y a` o_2`:Acceptance.}`

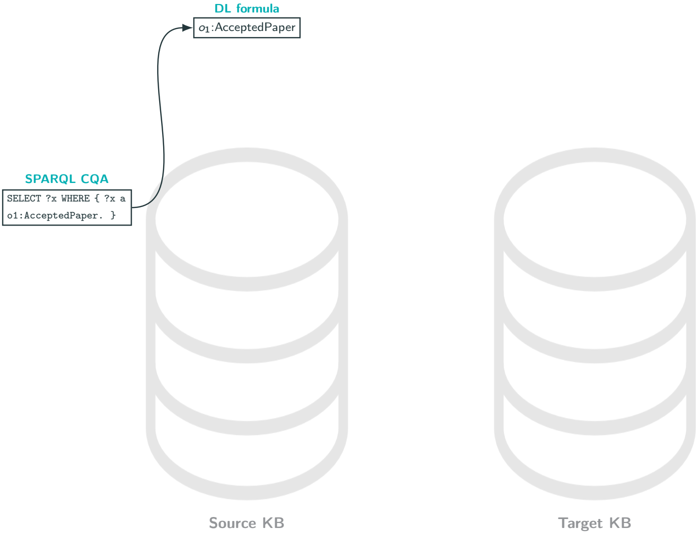
Unary set of instances Which are the accepted papers?
→ {paper1, paper2, ...}

Binary set of pairs of instances Who is the author of which paper?
→ {(paper1, person1), (paper2, person2), ...}

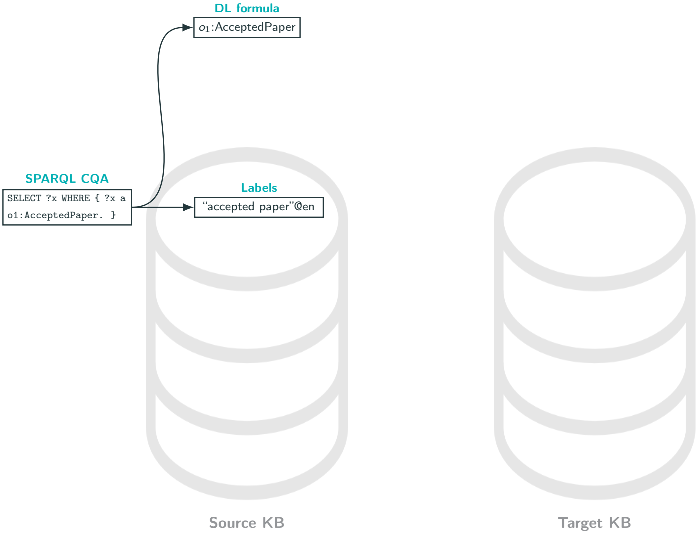
- Takes as input a set of CQAs in the form of SPARQL SELECT queries over o_1
- Requires o_1 and o_2 to have an Abox with at least one common instance for each CQA
    - answer (instances) to each input query are matched with those of a knowledge base described by o_2
    - matching is performed by finding the surroundings of the o_2 instances which are lexically similar to the CQA

Input: CQA, Source KB and Target KB

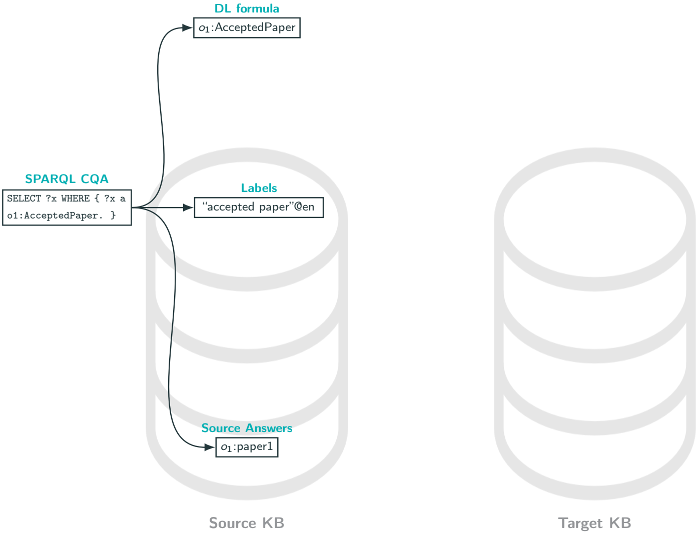
Complex alignment generation based on CQAs



1 Extract formula

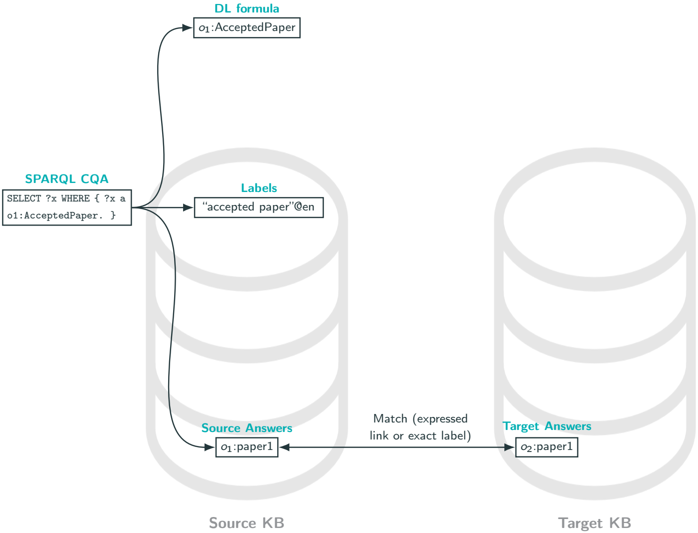


2 Extract CQA labels



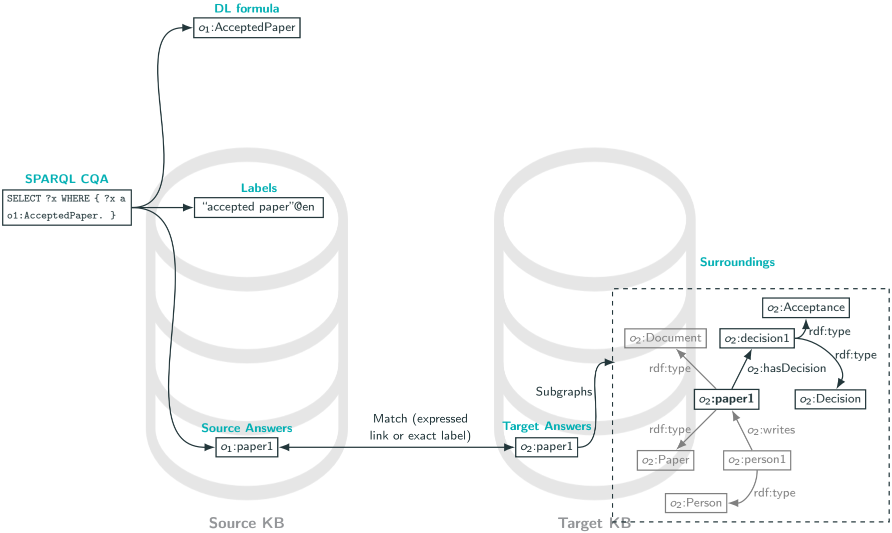
3 Retrieve answers

Complex alignment generation based on CQAs

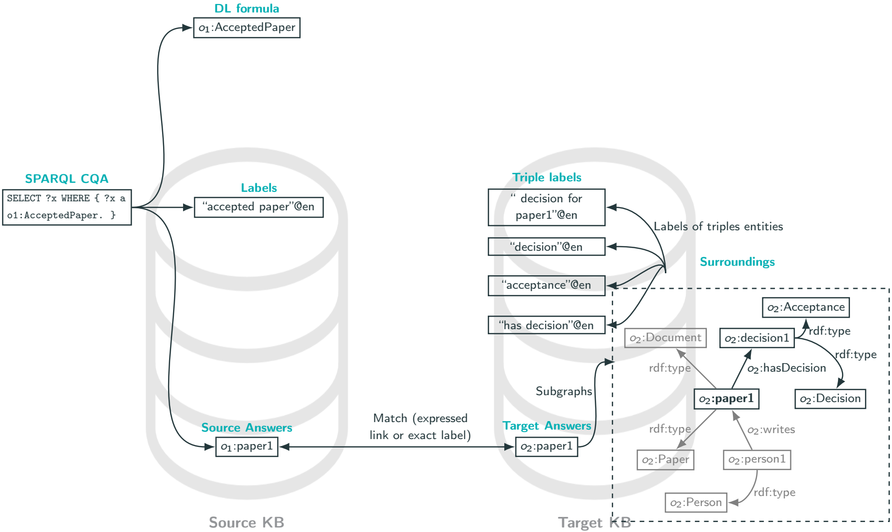


4 Match answers with target instances

Complex alignment generation based on CQAs

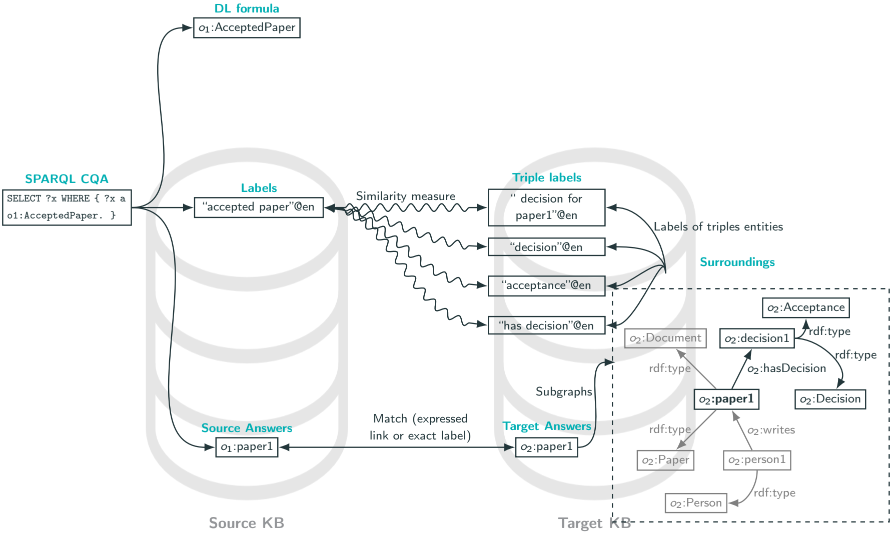


5 Get target subgraphs

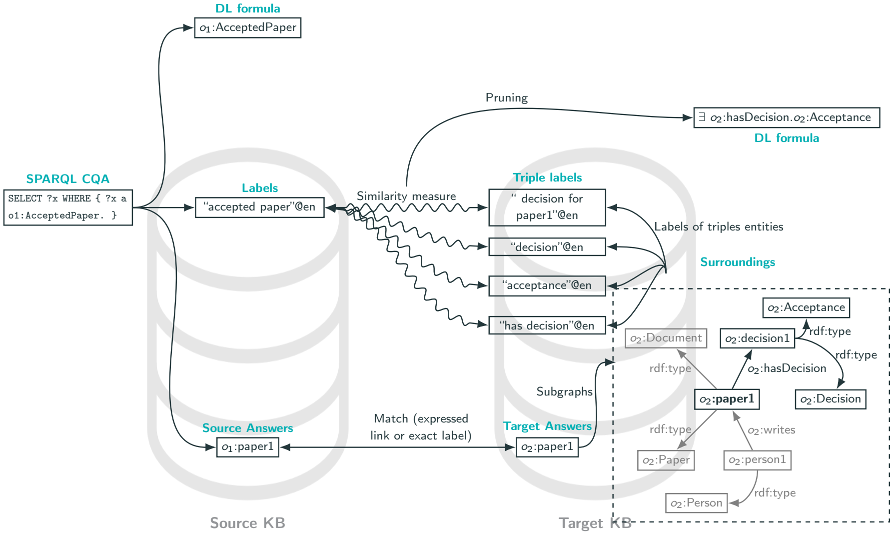


6 For each triple, get entity labels

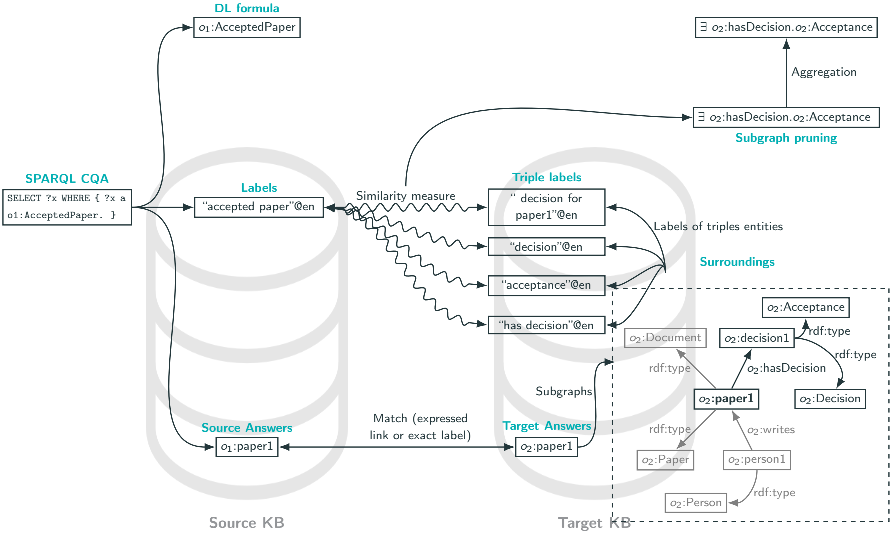
Complex alignment generation based on CQAs



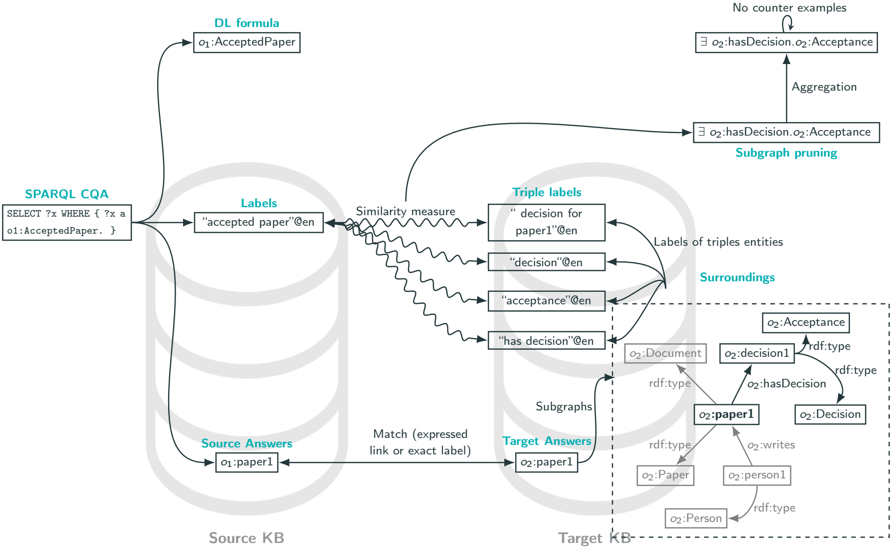
7 Compare the triple entities labels with the CQA labels



8 Prune the subgraph, transform it into a DL formula

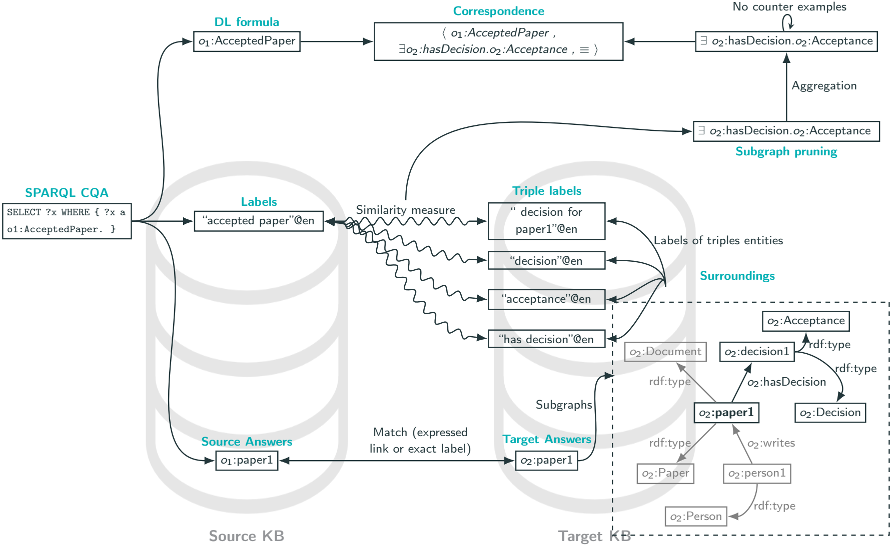


8 Aggregate the formula



9 Look for counter-examples and compute the confidence value

Complex alignment generation based on CQAs



10 Filter the formulae + 11 Generate correspondence

Thank you !

Questions ?

Chen, J., Hu, P., Jiménez-Ruiz, E., Holter, O. M., Antonyrajah, D., and Horrocks, I. (2020).
Owl2vec*: Embedding of OWL ontologies.
CoRR, abs/2009.14654.

Chen, J., Jiménez-Ruiz, E., Horrocks, I., Antonyrajah, D., Hadian, A., and Lee, J. (2021).
Augmenting ontology alignment by semantic embedding and distant supervision.
In Verborgh, R., Hose, K., Paulheim, H., Champin, P., Maleshkova, M., Corcho, Ó., Ristoski, P., and Alam, M., editors, *The Semantic Web - 18th International Conference, ESWC 2021, Virtual Event, June 6-10, 2021, Proceedings*, volume 12731 of *Lecture Notes in Computer Science*, pages 392–408. Springer.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2018).
BERT: Pre-training of deep bidirectional transformers for language understanding.
arXiv preprint arXiv:1810.04805.

Efeoglu, S. (2023).
Graphmatcher system presentation.
In Shvaiko, P., Euzenat, J., Jiménez-Ruiz, E., Hassanzadeh, O., and Trojahn, C., editors, *Proceedings of the 18th International Workshop on Ontology Matching co-located with the 22nd International Semantic Web Conference (ISWC 2023), Athens, Greece, November 7, 2023*, volume 3591 of *CEUR Workshop Proceedings*, pages 154–156. CEUR-WS.org.

Euzenat, J. and Shvaiko, P. (2013).
Ontology Matching, Second edition.
Springer Berlin Heidelberg, Berlin, Heidelberg.

He, Y., Chen, J., Antonyrajah, D., and Horrocks, I. (2022).
Bertmap: A bert-based ontology alignment system.
In *Thirty-Sixth AAAI Conference on Artificial Intelligence, AAAI 2022, Thirty-Fourth Conference on Innovative Applications of Artificial Intelligence, IAAI 2022, The Twelveth Symposium on Educational Advances in Artificial Intelligence, EAAI 2022 Virtual Event, February 22 - March 1, 2022*, pages 5684–5691. AAAI Press.

Hertling, S. and Paulheim, H. (2023).
Olala: Ontology matching with large language models.
In *Proceedings of the 12th Knowledge Capture Conference 2023*, K-CAP '23, page 131–139, New York, NY, USA. Association for Computing Machinery.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013).
Distributed representations of words and phrases and their compositionality.
Advances in neural information processing systems, 26.

Norouzi, S. S., Mahdavinejad, M. S., and Hitzler, P. (2023).
Conversational ontology alignment with chatgpt.
In Shvaiko, P., Euzenat, J., Jiménez-Ruiz, E., Hassanzadeh, O., and Trojahn, C., editors, *Proceedings of the 18th International Workshop on Ontology Matching co-located with the 22nd International Semantic Web Conference (ISWC 2023), Athens, Greece, November 7, 2023*, volume 3591 of *CEUR Workshop Proceedings*, pages 61–66. CEUR-WS.org.

Peeters, R. and Bizer, C. (2023).
Using chatgpt for entity matching.
In Abelló, A., Vassiliadis, P., Romero, O., Wrembel, R., Bugiotti, F., Gamper, J., Vargas-Solar, G., and Zumpano, E., editors, *New Trends in Database and Information Systems - ADBIS 2023 Short Papers, Doctoral Consortium and Workshops: AIDMA, DOING, K-Gals, MADEISD, PeRS, Barcelona, Spain, September 4-7, 2023, Proceedings*, volume 1850 of *Communications in Computer and Information Science*, pages 221–230. Springer.

Portisch, J., Hladik, M., and Paulheim, H. (2020).
ALOD2Vec matcher results for OAEI 2020.
In *Proceedings of the 15th Workshop on Ontology Matching*, pages 147–153.

Sousa, G., Lima, R., and Trojahn, C. (2022).
An eye on representation learning in ontology matching.
In Shvaiko, P., Euzenat, J., Jiménez-Ruiz, E., Hassanzadeh, O., and Trojahn, C., editors, *Proceedings of the 17th International Workshop on Ontology Matching (OM 2022) co-located with the 21th International Semantic Web Conference (ISWC 2022), Hangzhou, China, held as a virtual conference, October 23, 2022*, volume 3324 of *CEUR Workshop Proceedings*, pages 49–60. CEUR-WS.org.

Sousa, G., Lima, R., and Trojahn, C. (2023a).
Results of propmatch in OAEI 2023.
In Shvaiko, P., Euzenat, J., Jiménez-Ruiz, E., Hassanzadeh, O., and Trojahn, C., editors, *Proceedings of the 18th International Workshop on Ontology Matching co-located with the 22nd International Semantic Web Conference (ISWC 2023), Athens, Greece, November 7, 2023*, volume 3591 of *CEUR Workshop Proceedings*, pages 178–183. CEUR-WS.org.

Sousa, G., Lima, R., Vieira, R., and Trojahn, C. (2023b).
Using BERT models to automatically classify domain concepts into DOLCE top-level concepts: A study of the OAEI ontologies.
In *Proceedings of the Joint Ontology Workshops 2023 Episode IX: The Quebec Summer of Ontology co-located with the 13th International Conference on Formal Ontology in Information Systems (FOIS 2023), Sherbrooke, Québec, Canada, July 19-20, 2023*, volume 3637 of *CEUR Workshop Proceedings*. CEUR-WS.org.

Suárez-Figueroa, M. C., Gómez-Pérez, A., and Fernández-López, M. (2012).
The neon methodology for ontology engineering.
In Suárez-Figueroa, M. C., Gómez-Pérez, A., Motta, E., and Gangemi, A., editors, *Ontology Engineering in a Networked World.*, pages 9–34. Springer.

Wang, Q., Gao, Z., and Xu, R. (2023).
Exploring the in-context learning ability of large language model for biomedical concept linking.

Wu, Z., Pan, S., Chen, F., Long, G., Zhang, C., and Philip, S. Y. (2020).
A comprehensive survey on graph neural networks.
IEEE transactions on neural networks and learning systems, 32(1):4–24.

Zhang, Y., Wang, X., Lai, S., He, S., Liu, K., Zhao, J., and Lv, X. (2014).
Ontology matching with word embeddings.
In *Chinese Computational Linguistics and Natural Language Processing Based on Naturally Annotated Big Data: 13th China National Conference, CCL 2014, and Second International Symposium, NLP-NABD 2014, Wuhan, China, October 18-19, 2014. Proceedings*, pages 34–45. Springer.