OntoAligner Meets Knowledge Graph Embedding Aligners

Hamed Babaei Giglou[1], Jennifer D'Souza[1], Sören Auer[1] and Mahsa Sanaei[2]

[1]*TIB – Leibniz Information Centre for Science and Technology, Hannover, Germany*
[2]*University of Tabriz, Tabriz, Iran*

Abstract

Ontology Alignment (OA) is essential for enabling semantic interoperability across heterogeneous knowledge systems. While recent advances have focused on large language models (LLMs) for capturing contextual semantics, this work revisits the underexplored potential of Knowledge Graph Embedding (KGE) models, which offer scalable, structure-aware representations well-suited to ontology-based tasks. Despite their effectiveness in link prediction, KGE methods remain underutilized in OA, with most prior work focusing narrowly on a few models. To address this gap, we reformulate OA as a link prediction problem over merged ontologies represented as RDF-style triples and develop a modular framework—integrated into the OntoAligner library—that supports 17 diverse KGE models. The system learns embeddings from a combined ontology and aligns entities by computing cosine similarity between their representations. We evaluate our approach using standard metrics across seven benchmark datasets spanning five domains: Anatomy, Biodiversity, Circular Economy, Material Science and Engineering, and Biomedical Machine Learning. Two key findings emerge: first, KGE models like ConvE and TransF consistently produce high-precision alignments, outperforming traditional systems in structure-rich and multi-relational domains; second, while their recall is moderate, this conservatism makes KGEs well-suited for scenarios demanding high-confidence mappings. Unlike LLM-based methods that excel at contextual reasoning, KGEs directly preserve and exploit ontology structure, offering a complementary and computationally efficient strategy. These results highlight the promise of embedding-based OA and open pathways for further work on hybrid models and adaptive strategies.

Keywords

OntoAligner, Knowledge Graph Embeddings, Ontology Alignment, Ontology

1. Introduction

Ontologies serve as formal, structured representations of knowledge within a specific domain. By axiomatizing concepts, relations, and properties, ontologies provide semantic richness that facilitates knowledge sharing, reasoning, and interoperability. Over the past decades, they have become a fundamental component of the Semantic Web, powering knowledge-intensive applications across domains ranging from E-commerce to Biomedical and Material Science. With the rise of artificial intelligence (AI), particularly in natural language processing (NLP), ontologies have increasingly been integrated into symbolic AI systems. The advent of large-scale transformer models, such as BERT [1] and its successors, has accelerated this trend. These models have enabled new methods for semantic understanding, prompting researchers to revisit ontology-based approaches through the lens of deep learning. One such area that has gained substantial attention is Ontology Alignment (OA) [2] — the task of identifying correspondences between semantically equivalent entities across different ontologies.

Modern OA techniques predominantly leverage machine learning and embedding-based strategies to compute alignments. Among these, large language model (LLM) techniques have become especially popular due to their ability to encode contextual semantics [3, 4, 5, 6]. Despite this momentum, an important class of embedding methods — Knowledge Graph Embeddings (KGEs) [7] — remains relatively underexplored in the context of OA. KGEs are techniques that convert the entities and relations in a knowledge graph into continuous low-dimensional vectors, allowing the graph's semantic and structural

information to be represented numerically. This transformation offers several advantages: it simplifies large and complex graphs, enables the use of machine learning algorithms, enhances search and recommendation systems by capturing semantic similarity, and helps uncover hidden patterns and relationships that are not easily detectable in the original symbolic form.

KGEs, such as TransE [8], DistMult [9], and ComplEx [10], are typically designed for single knowledge graphs, focusing on link prediction or taxonomy enrichment tasks within a single ontology. As a result, they are often perceived as less directly applicable to the cross-ontology matching objective of OA. However, advancements such as RDF2Vec [11] and OWL2Vec [12] have demonstrated the potential of adapting KGE techniques for ontology-related tasks, including alignment, as is evident with OWL2VecOA [13]. These models learn embeddings from RDF graphs or OWL ontologies by generating graph-based sequences, allowing them to capture semantic structure in a way that is more compatible with OA needs. Still, the majority of current work either focuses narrowly on a few KGE-based models or fails to harness the full spectrum of available KGE techniques in the alignment setting. This gap highlights an important research opportunity: *to systematically explore and adapt KGE models for OA, evaluate their comparative performance, and investigate hybrid models that integrate the strengths of both LLMs and KGEs.* By doing so, this work aims to advance the state of OA through KGE methodologies that are both semantically grounded and computationally scalable.

OA is the process of finding correspondences between semantically related entities from two ontologies. Formally, given two ontologies $O_{source} = (C_s, R_s, I_s)$ and $O_{target} = (C_t, R_t, I_t)$, where C, R, and I denote sets of concepts, relations, and instances respectively, an alignment A is a set of mappings $m = \langle e_s, e_t, r, \theta \rangle$ such that $e_s \in O_s, e_t \in O_t, r \in \{=, \subseteq, \supseteq, \equiv, \approx\}$ is a semantic relation, and $\theta \in [0, 1]$ represents the confidence score. A similarity function $\text{sim}(e_s, e_t) : O_{source} \times O_{target} \rightarrow [0, 1]$ is typically used to compute this score, and mappings are included in the alignment if $\text{sim}(e_s, e_t) \geq \tau$, where τ is a predefined similarity threshold.

Additionally, an ontology can be represented as a set of triples in the form (h, r, t), where h denotes the head entity, r the relation, and t the tail entity. In this work, to enable KGE models to learn meaningful representations from both O_{source} and O_{target}, we combined their respective triples to construct a unified triple repository referred to as the triple factory FA. This integration serves as a foundational step toward a systematic framework for analyzing and exploring the behavior of KGEs within the context of OA. By formulating OA as a link prediction task over the merged ontologies, we implemented and evaluated a collection of KGE models specifically tailored for alignment objectives. This formulation allows the models to jointly learn latent representations of entities and relations across both ontologies, thereby capturing structural and semantic correspondences more effectively. The learned representations were subsequently utilized for equivalence-based alignment, enabling the identification of semantically equivalent concepts across ontologies. Moreover, to make the collection entirely available for researchers and practitioners, we integrated our systematic approach to the *OntoAligner* [14] – a comprehensive and modular Python library dedicated to OA.

The remainder of this paper is organized as follows: section 2 reviews related work; section 3 details our proposed methodology with integration with the OntoAligner library; section 4 presents the experimental results and analysis; and section 5 concludes the paper with discussions and future research directions.

2. Related Works

Early work on graph alignment employs a graph embedding algorithm, and studies have shown that the great capability of the KGE method is effective for aligning structurally similar ontologies and is more robust against alignment noise when dealing with graphs of different sizes and architectures [15].

[16] proposed a multi-view embedding model for biomedical OA using TransE and ConvE, demonstrating the utility of combining structural and semantic perspectives. Similarly, [17] conducted a systematic evaluation of KGEs for gene-disease association prediction, benchmarking models including TransD, TransE, TransH, DistMult, HolE, and ComplEx. More recent approaches have explored deeper

models and alignment-specific enhancements. [18] introduced a security-aware, deep model-based entity alignment method incorporating MTransE, TransD, RotatE, ConvE, AlignE, AttrE, and GCN-a, tailored for edge-specific knowledge graphs. [19] tackled cross-lingual OA by leveraging both structural and semantic similarity via node2vec, GCN, RGCN, and TransE.

Contextual embeddings have also gained traction. [20] combined traditional KGEs such as TransE, TransR, and DistMult with semantic embeddings like Word2Vec, Onto2Vec, OPA2Vec, and OWL2Vec to predict subsumption relations. Likewise, [21] presented LaKERMap, a contextualized structural self-supervised learning approach for ontology matching that employed TransE for inference tasks. Additionally, [22] used TransE, RotatE, and CompGCN for deep active alignment of knowledge graph entities and schemata. [23] introduced an assertion and alignment correction framework using RDF2Vec, TransE, TransR, TransH, DistMult, and ComplEx.

Additional contributions include A-LIOn by [24], which used TransR to align ontologies through inconsistency-based negative sampling, and the AMD matcher [25], which also adopted TransR for large-scale alignment scenarios. [26] examined context-enriched models for aligning biomedical vocabularies using a range of methods, including TransE, TransR, RESCAL, DistMult, HolE, ComplEx, and ConvKB. In the scholarly knowledge domain, [27] refined the SemOpenAlex concept ontology using TransE, DistMult, and QuatE with SKOS-based constraints. [28] introduced NELLIE, an open data linking system leveraging ComplEx embeddings for scalable entity linking. Finally, [29] investigated fine-grained semantics in knowledge graph relations, further underscoring the breadth of KGE applications in ontology understanding and alignment.

3. Methodology

The OntoAligner framework is composed of three key components: the *parser*, *encoder*, and *aligner*. This modular design enables OntoAligner to serve as a hub for integrating diverse OA approaches. Building upon these design principles, we developed a framework that adheres to these foundational components, as illustrated in Figure 1. In the following sections, we first introduce the overall alignment strategy, followed by an overview of the collection of KGE models integrated within this strategy.

3.1. Graph Embedding Aligner

The architecture of the *Graph Embedding Aligner*, as illustrated in Figure 1, follows a modular three-stage pipeline: Parser, Encoder, and Aligner. This design reflects the core components implemented in the code and provides a precise data flow from input ontologies to alignment predictions.

1) Parser.   The pipeline begins with parsing the source O_{source} and target O_{target} ontologies. Each ontology is decomposed into RDF-style metadata, consisting of Subject, Predicate, and Object, each annotated with its respective IRI and label. Moreover, a class membership metadata — indicating whether each entity plays a subject or object role in a class assertion is also attained. In the end, each ontology is represented by its own metadata.

2) Encoder.   In the encoding stage, the extracted triples from both ontologies are unified into a single triplet representation: a set of $O_{source}(h, r, t)$ and $O_{target}(h, r, t)$ triples are obtained, where h, r, and t are represented using natural language text rather than IRIs. Next, as a triplet representation, we unified both triples to form a triple factory $FA := O_{source}(h, r, t) \cup O_{target}(h, r, t)$. One of the key advantages of this unified representation is that it enables the embedding model to automatically identify and learn shared structural and semantic patterns across both ontologies with high precision, thereby enhancing the quality and effectiveness of the alignment process.

3) Aligner.   The aligner component consists of two submodules. *1) Representation Learning*, where a PyKEEN model embeds and trains each entity into a continuous vector space where semantically
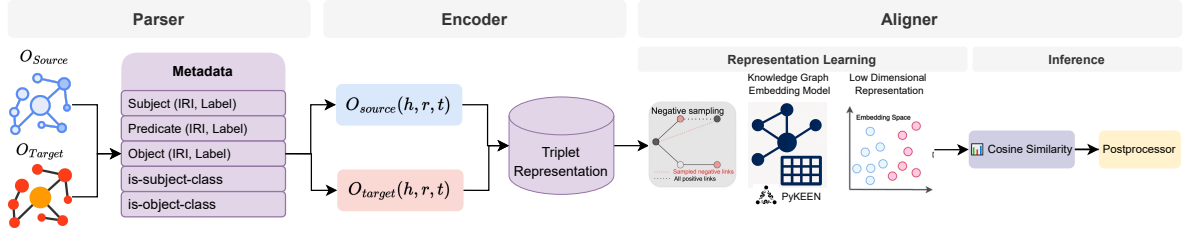
Figure 1: The architecture of the proposed framework, comprising three main stages: Parser, Encoder, and Aligner. The model ingests source and target ontologies, encodes them as unified triples, learns low-dimensional embeddings via a KGE model (trained in PyKEEN), and finally computes alignment predictions through cosine similarity.

or structurally similar entities are positioned closely together. *2) Inference*, which uses the learned embeddings to calculate cosine similarity between every $\langle e_s \in O_{source}, e_t \in O_{target} \rangle$ pair for ranking, and postprocessing.

- **REPRESENTATION LEARNING**. In this submodule, a KGE model is trained using a link prediction objective. The process begins with negative sampling, which augments the dataset by generating plausible but incorrect triples. This step helps the model learn to distinguish between valid and invalid relationships. The KGE model is then fine-tuned, and its resulting low-dimensional embeddings are used for alignment. Although negative sampling may seem to introduce noise into the OA process, it plays a critical role in improving the embedding model's ability to distinguish between valid and invalid relations. This results in more robust and generalizable representations, which ultimately lead to more accurate alignment across heterogeneous ontologies. We utilize PyKEEN [30] to support each stage of representation learning, as it offers a comprehensive suite of KGE models and tools.

- **INFERENCE**. In inference, once embeddings are learned, the embeddings $\forall e \in O_{source}$ and $\forall e \in O_{target}$ are extracted. Let $\mathbf{E}_{source} = [\mathbf{e}_s^{(1)}, \mathbf{e}_s^{(2)}, \ldots, \mathbf{e}_s^{(n)}] \in \mathbb{R}^{n \times d}$ denote the matrix of L2-normalized embeddings for entities from the source ontology O_{source}, and let $\mathbf{E}_{target} = [\mathbf{e}_t^{(1)}, \mathbf{e}_t^{(2)}, \ldots, \mathbf{e}_t^{(m)}] \in \mathbb{R}^{m \times d}$ represent the normalized embeddings from the target ontology O_{target}, where d is the embedding dimension. Each vector is normalized as $\|\mathbf{e}\|_2 = 1$. The similarity between entities is computed using cosine similarity. The similarity matrix $\mathbf{S} \in \mathbb{R}^{n \times m}$ is defined as $\mathbf{S} = \mathbf{E}_{source} \cdot \mathbf{E}_{target}^{\top}$. Where, each entry S_{ij} corresponds to the cosine similarity between the i-th source and j-th target entities: $S_{ij} = \cos(\theta) = \mathbf{e}_s^{(i)} \cdot \mathbf{e}_t^{(j)} = \sum_{k=1}^{d} e_{s,k}^{(i)} \cdot e_{t,k}^{(j)}$. In the final step, for each source entity i, the target entity with the highest similarity score is selected: $j^* = \arg\max_j S_{ij}$. The alignment result is then given by: $A_i = \langle e_s^{(i)}, e_t^{(j^*)}, \theta \rangle$. Where $\theta := S_{ij^*}$ is the confidence score of the alignment.

  Once the alignment pairs are extracted, a post-processing step is applied to refine the results. First, a one-to-one cardinality constraint is enforced to ensure that each $e_s \in O_{source}$ aligns with at most one $e_t \in O_{target}$, and vice versa. Then, a confidence-based filtering is performed by applying a similarity threshold $\theta \geq \tau$, where $\tau \in [0, 1]$ is a predefined cutoff. Alignment pairs with scores below this threshold are discarded to retain only the most confident and unambiguous matches.

3.2. Knowledge Graph Embedding Collections

Based on a review of related works, we identified the **10** most frequently used KGE models that have been applied to OA from various perspectives—whether as standalone baseline models or as part of hybrid frameworks. These models consistently appear across numerous OA studies and knowledge engineering benchmarks. In addition to these top 10, we incorporated seven additional models that, while not as widely adopted in OA specifically, have shown strong performance and versatility in

Table 1
Overview of 17 KGE models supported in OntoAligner.

KGE Model	Overview	Related Works
ConvE [31]	ConvE is a deep convolutional embedding model for link prediction that uses 2D convolutions over reshaped entity and relation embeddings to capture complex interaction patterns between them.	[32, 33, 34, 16, 18]
TransD [35]	TransD (Translation on Dynamic Mapping Matrices) is a knowledge graph embedding model designed for link prediction and triple classification. It improves upon earlier translation-based models like TransE, TransH, and TransR by dynamically constructing relation-specific projection matrices using both entity and relation projection vectors.	[17, 36, 18]
TransE [8]	TransE is a simple and scalable model for embedding knowledge graphs in low-dimensional vector spaces. It represents relationships as vector translations between head and tail entity embeddings. For a valid triple (h, r, t), TransE enforces that $h + r \approx t$. The model is easy to train, has few parameters, and can handle very large datasets.	[16, 17, 19, 20, 21, 22, 23, 26, 27]
TransF [37]	TransF is a flexible translation-based model for knowledge graph embedding. It improves on previous methods by allowing more adaptable translations to better handle complex relation types, like one-to-many or symmetric relations. Without increasing model complexity, TransF introduces a new scoring function and shows strong performance improvements in experiments.	-
TransH [38]	TransH improves knowledge graph embeddings by projecting entities onto relation-specific hyperplanes before translation. This allows it to handle complex relation types (e.g., one-to-many) better than TransE, while maintaining similar efficiency and scalability.	[17, 23]
TransR [39]	TransR enhances knowledge graph embeddings by projecting entities into relation-specific spaces before applying translations, allowing better modeling of diverse relational semantics than TransE and TransH.	[20, 23, 24, 25, 26]
DistMult [9]	DistMult is a simple bilinear embedding model for knowledge graphs that represents entities and relations as vectors, using matrix multiplication to capture relational semantics. It outperforms previous models like TransE in link prediction and enables effective logical rule mining.	[17, 20, 23, 26, 27, 29]
ComplEx [10]	ComplEx is a link prediction model that uses complex-valued embeddings to effectively capture both symmetric and antisymmetric relations. It relies on the Hermitian dot product, offering a simple yet powerful and scalable approach that outperforms existing models on standard benchmarks.	[17, 23, 26, 28]
HolE [40]	HolE (Holographic Embeddings) is a knowledge graph embedding model that uses circular correlation to create compositional vector representations of entities and relations. It efficiently captures complex interactions while remaining scalable and easy to train.	[17, 26]
RotatE [41]	RotatE is a knowledge graph embedding model that represents relations as rotations in complex space, enabling it to capture patterns like symmetry, inversion, and composition. It uses a novel self-adversarial negative sampling for efficient training and outperforms prior models on link prediction tasks.	[18, 22]
SimplE [42]	SimplE improves tensor factorization for knowledge graph link prediction by learning dependent embeddings for each entity, overcoming limitations of traditional methods. It offers interpretable, efficient embeddings, supports background knowledge, and achieves strong performance with proven full expressiveness.	-
CrossE [43]	CrossE is a knowledge graph embedding method that models bi-directional interactions between entities and relations by creating both general and triple-specific embeddings. It achieves state-of-the-art link prediction results on complex datasets and improves explainability by generating reliable paths to support its predictions.	-
BoxE [44]	BoxE is a knowledge base completion model that represents entities as points and relations as hyper-rectangles (boxes) in a spatial embedding. It overcomes key limitations of previous models by supporting logical rules, hierarchies, and higher-arity relations.	-
CompGCN [45]	CompGCN is a graph convolutional network designed for multi-relational graphs that jointly learns embeddings for both nodes and relations. It uses entity-relation composition methods to efficiently handle many relations, generalizes existing multi-relational GCNs, and achieves strong results on tasks like node classification and link prediction.	[22]
MuRE [46]	MuRE proposes embedding multi-relational knowledge graphs in hyperbolic space using relation-specific transformations, better capturing multiple hierarchies. It outperforms Euclidean and other methods on link prediction, especially in low-dimensional settings.	-
QuatE [47]	QuatE uses quaternion embeddings to model entities and relations in knowledge graphs, enabling expressive 4D rotations and compact interactions via the Hamilton product. It generalizes ComplEx with better geometric properties and effectively captures key relational patterns, achieving strong results on benchmark datasets.	[27]
SE [48]	SE proposes a neural network-based approach to embed symbolic knowledge from diverse Knowledge Bases into a continuous vector space, preserving and enriching their structure.	-

broader knowledge graph tasks such as link prediction, completion, and entity classification. Table 1 summarizes all 17 models, presenting their key characteristics and citations from recent literature.

The selection of these models was guided by several criteria. First, we prioritized diversity in modeling approaches, ensuring inclusion of translation-based (e.g., TransE, TransH), convolutional (e.g., ConvE), bilinear (e.g., DistMult), and neural graph-based methods (e.g., CompGCN). Second, we considered theoretical expressiveness and scalability, selecting models that are capable of handling

Table 2
OAEI tracks and tasks statistics across source, target, and alignments.

Track	Task	O_{source} **Triplets**	O_{target} **Triplets**	**References**
ANATOMY	Mouse-Human	5,957	13,196	1,516
BIODIV – Biodiversity & Ecology	ENVO-SWEET	16,237	17,182	805
	FISH-ZOOPLANKTON	440	173	15
CE–Circular Economy	CEON - BiOnto	1,896	1,307	18
MSE–Material Science & Engineering	MI - MatOnto	2,152	2,740	302
BIO-ML–Biomedical Machine Learning	NCIT-DOID	22,559	19,276	3,280
	OMIM-ORDO	23,189	11,429	2,605

large-scale ontologies with complex relational patterns. Third, we looked at empirical evidence from prior evaluations that demonstrated the effectiveness of these models across a range of domains [15]. Finally, compatibility with PyKEEN was an important practical consideration, allowing for unified implementation and experimentation within the OntoAligner framework. These selected models provide a representative and comprehensive foundation for experimenting with embedding-based OA strategies.

3.3. Integration with OntoAligner

To enable embedding-based OA, we developed a specialized module called *GraphEmbeddingAligner*. Built on top of the PyKEEN framework, this aligner harnesses KGE models to learn vector representations of entities from both source and target ontologies. OntoAligner currently supports **17** KGE models, all of which can be easily integrated through this module. A comprehensive usage guide is available at http://ontoaligner.readthedocs.io/aligner/kge.html. With a modular and extensible design, `GraphEmbeddingAligner` allows users to flexibly experiment with different KGE models and customize training configurations to suit various alignment tasks. An example demonstrating how to use KGE-based aligners within OntoAligner can be found at https://github.com/sciknoworg/OntoAligner/blob/main/examples/kge.py.

4. Evaluations

This section delves into empirically validating KGE models by employing precision, recall, and F1-score metrics. Experimental datasets and results are presented in the following.

4.1. Experimental Setups

Evaluation Datasets OAEI Tracks and Tasks.   We carefully chose five tracks from the OAEI-2024 campaign [49] spanning diverse domains for our experimental configurations. The statistics for seven datasets in five tracks are outlined in Table 2. The chosen tracks include: ANATOMY [50] (Mouse-Human), BIODIV – Biodiversity and Ecology [51] (two tasks), CE – Circular Economy [52] (CEON-BiOnto), MSE – Material Science and Engineering [53](MI-MatOnto), and BIO-ML – Biomedical Machine Learning [54] (two tasks). These tracks were chosen to represent a range of dataset sizes (according to the number of triplets) and complexity levels, including: Small-scale tasks (CE track, MSE track, and FISH-ZOOPLANKTON from BIODIV track), Medium-scale tasks (ANATOMY), and Large-scale tasks (BIO-ML track and the ENVO–SWEET task from the BIODIV track). This selection ensures the evaluation covers both domain and size diversity, providing insights into the performance of KGEs.

KGE Hyperparameters and OS.   For fair comparison, we used CPU-based experimentation with an embedding dimension of 200, training epoch number of 20, train batch size of 64, evaluation batch size of 128, and a number of negative samples per positive sample of 5. Moreover, we used 10 core CPUs with a maximum memory of 80 GB for experimentation.

Table 3

Best-performing KGE models on seven OAEI-2024 benchmark tasks. The table reports performance metrics for the top KGE model per task, including threshold τ, overlap between alignments and gold ($\cap$), alignment size (A), execution time (T in seconds), precision (Prec), recall (Rec), and F-Measurement (F). The final column lists the best-performing OA system for each task (with F-Measurement score in parentheses) for comparison.

Task	KGE	τ	$\cap$	A	T	Prec	Rec	F	Best Performer
Mouse-Human	DistMult	0.34	1047	1069	70.3	97.9	69.0	81.0	Matcha [55] (94.1)
FISH-ZOOPLANKTON	TransF	0.25	9	9	4	100	60.0	**74.9**	LogMapLt [56] (64.4)
ENVO-SWEET	ConvE	0.6	327	367	855	89.1	40.6	55.8	LogMap [56] (71.3)
CEON-BiOnto	ConvE	0.35	10	17	80	58.8	55.5	**57.1**	Matcha [55] (47.8)
MI-MatOnto	TransD	0.39	32	37	29	86.4	10.5	18.8	LogMap [56] (32.0)
OMIM-ORDO	ConvE	0.43	538	772	871	69.6	20.6	31.8	BERTMap‡ [56] (64.6)
NCIT-DOID	SE	0.43	1751	2537	1584	69.0	53.3	60.2	HybridOM* [57] (91.8)

4.2. Results

The Table 3 presents the best-performing KGE model for each of the seven OAEI benchmark tasks for all seven tasks of this study.

4.2.1. Domain Specific Analysis

We have explored KGE models across five domains, and the results presented in Table 3 present interesting findings per domain, such as:

- **Anatomy Track.** In the Mouse-Human task, the *DistMult* model reached a high precision of 97.9—ranking second among all methods of OAEI-2024[1]—though its recall (69.0%) brought the overall F-Measure to 81.0, behind Matcha aligner [55] with F-Measure of 94.1.

- **Biodiversity and Ecology Track.** In the *FISH-ZOOPLANKTON* task of Biodiv, KGE models – *TrasnF* aligner – showed notable superiority. The TransF model achieved a perfect precision of 100 and an F-Measure of **74.9%**, significantly outperforming LogMapLt [56]—the best OAEI-2024 system at this task[2]—with an F1 of 64.4. Despite LogMapLt's nearly perfect execution time, the KGE method required only 4 seconds, maintaining efficiency while outperforming the state-of-the-art system. In the *SWEET-ENVO* task, the LogMap [56] remains the best OAEI system by F-Measure of 71.3%. However, *ConvE* aligner exhibited a higher precision of 89.1% than all systems within OAEI 2024, including LogMapLt with a precision of 80.3%.

- **Circular Economy Track.** For the *CEON-BiOnto* task from this track, the *ConvE* aligner achieved the best overall F-Measure of **57.1%**, outperforming Matcha [55] with an F-Measure of 47.8%, the top performer in the official OAEI-2024[3] results.

- **Material Science and Engineering Track.** The *MI-MatOnto* task within this track, showed a poor F-Measure of 18.8% using *TransD* aligner. However, the maintained precision of 86.4% places this model's performance close to the LogMap system (from OAEI-2023[4]) with a precision of 88.1%.

- **Biomedical Machine Learning Track.** For this track, the KGE models did not perform well in terms of precision and F-Measure. For the *OMIM-ORDO* task, *ConvE* aligner reached only F-Measure of 31.8%, far behind BERTMap‡ [56] with F-Measure of 64.6% (OAEI-2024[5] system performance). Even the precision of 69.6% fell short, indicating limitations of unsupervised KGE methods for complex disease-related alignments. For another task of this track, specifically *NCIT-DOID* task, it showed *SE* aligner performance of 60.2% in terms of F-Measure, significantly
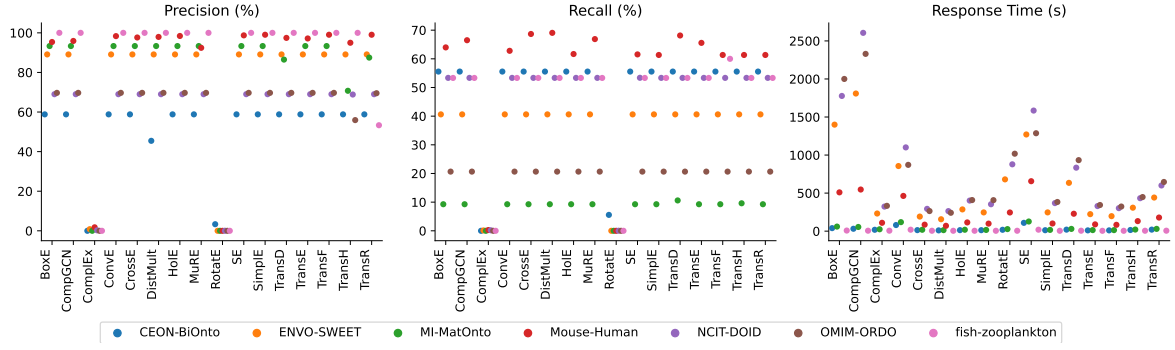

Figure 2: Column 1 and 2: Precision and recall analysis across aligners and tasks. Column 3: Representation learning and inference time response analysis.

underperforming compared to HybridOM* [57] with F-Measure of 91.8%. Precision of 69.0% was still promising, but the gap in recall of 53.3% limited its overall effectiveness.

KGE models, particularly *ConvE* and *TransF*, demonstrated competitive or even superior performance on two tasks—*FISH-ZOOPLANKTON* and *CEON-BiOnto*—in terms of F-Measure. In other tasks, particularly OMIM-ORDO and NCIT-DOID, performance lagged behind traditional or supervised OA systems. Overall, KGE methods tend to produce **high-precision** alignments with lower recall, suggesting their suitability for applications requiring conservative, high-confidence mappings.

4.2.2. Empirical Trends in KGE Aligners

The analysis of the summary of results presented in Table 3 across seven benchmark tasks and five domains reveals consistent empirical trends in KGE aligners' behavior. While performance varies across tasks and domains, certain trends emerge regarding precision, recall, alignment size, execution time, and model-task interactions. These observations provide insight into the operational characteristics and efficiency of KGE aligners and highlight considerations for their application in ontology alignment tasks

Behavioral Characteristics. Across all tasks, the top-performing KGE models exhibited notably high precision scores—often exceeding 85%—even in tasks where recall and F-Measure were relatively poor, e.g., TransD in MI-MatOnto (Precision 86.4% but Recall 10.5%), ConvE in ENVO-SWEET (Precision 89.1% vs. Recall 40.6%), or DistMult in Mouse-Human: (Precision 97.9% vs. Recall 69.0%). More detailed analysis is presented in Figure 2 (Precision and Recall columns). This shows that KGE aligners are conservative aligners that prioritize correctness over completeness, making them well-suited for high-confidence, low-risk integration tasks where false positives are costly.

Moreover, tasks like OMIM-ORDO and NCIT-DOID, which have large-scale sizes in terms of references, also had large-scale alignments ($A = 772$ and $A = 2537$ respectively). Yet, their performance (F-Measures of 31.8% and 60.2%, respectively) remained moderate compared to tasks with smaller alignments (e.g., FISH-ZOOPLANKTON with only nine alignments and F-Measures = 74.9%, where total references is 9 – see Table 2 for total references). This means that predicted alignment volume does not directly translate to quality, especially in semantically dense or noisy domains. KGE models may overgenerate candidates in large ontologies unless appropriately constrained.

Operational Efficiency. The majority of KGE aligners completed their tasks—including representation learning and inference—within 100 seconds for most benchmark tasks (Table 3), even when handling hundreds or thousands of candidate alignments, such as in Mouse-Human. Figure 3 illustrates the CPU versus memory usage of the models, averaged across all tasks. Most aligners utilize 80−90% of available CPU cores and consume over 6GB of memory, highlighting that KGE-based methods
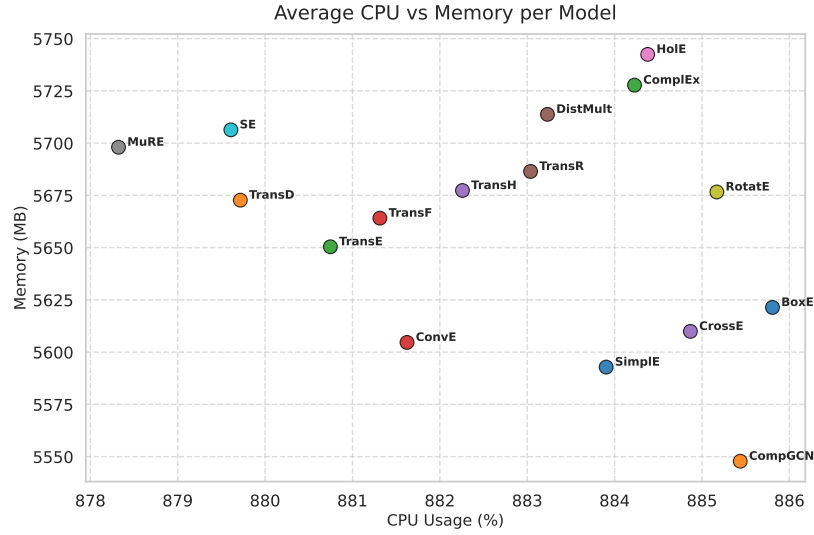
Figure 3: Scatter plot showing the average CPU utilization (%) against memory consumption (MB) for each KGE aligner. Each color represents a distinct model. This visualization highlights the relative computational efficiency and resource demands of the models.

are computationally efficient and capable of supporting scalable or on-demand ontology alignment, including in dynamic or real-time systems.

Model-Task Interactions. The optimal similarity threshold τ for alignment significantly varies across tasks, suggesting that no universal threshold works across domains. This indicates that KGE aligners require task-specific or domain-specific calibration, particularly around the similarity threshold. Auto-tuning or adaptive thresholding could significantly improve F-Measure in future iterations.

Nevertheless, certain KGE aligners seem particularly effective in specific domains. ConvE aligner performs best in comparison to its own companion KGE aligners in multi-relational tasks (ENVO-SWEET, CEON-BiOnto, OMIM-ORDO). TransF aligner excels in structure-rich, less ambiguous domains (FISH-ZOOPLANKTON). DistMult works well in clean, hierarchical taxonomies (Mouse-Human). SE aligner provides a better balance of precision and recall in larger biomedical terminologies (NCIT-DOID). Future systems could auto-select the KGE aligner based on ontology metadata (e.g., size, depth, domain) to optimize performance per task or do ensemble learning.

5. Conclusions and Future Directions

In this paper, we have systematically explored the application of Knowledge Graph Embeddings within Ontology Alignment tasks. Our comprehensive framework, integrated within the OntoAligner toolkit, leverages a collection of 17 prominent KGE models. Through empirical evaluations on seven benchmark tasks across diverse domains such as Anatomy, Biodiversity, Circular Economy, Material Science, and Biomedical Machine Learning, we identified several key findings. KGE-based aligners generally produce alignments characterized by high precision but moderate recall, indicating their suitability for conservative, high-confidence ontology matching scenarios.

Notably, ConvE and TransF emerged as particularly effective models, demonstrating superior performance in multi-relational and structure-rich tasks, respectively. Nevertheless, our analysis also revealed limitations of KGE methods in complex biomedical alignments, highlighting the need for improved techniques or hybrid approaches in these domains.

Future research could address several promising directions:

- *Adaptive Thresholding and Calibration:* Our results indicated no universal similarity threshold applicable across diverse ontology alignment tasks. Developing adaptive thresholding strategies

that dynamically calibrate based on ontology metadata could enhance model flexibility and performance.

- *Hybrid Models:* Integrating KGEs with LLMs or other contextual embedding approaches could leverage complementary strengths, potentially improving alignment accuracy in complex, context-rich domains.
- *Domain-specific Enhancements:* Given the domain-dependent effectiveness observed, tailoring KGE methodologies to specific ontological structures or leveraging metadata-driven model selection and ensemble strategies may provide meaningful gains.

In conclusion, embedding-based ontology alignment presents a powerful yet still evolving paradigm. Addressing these future directions will not only advance the state-of-the-art in ontology alignment but also extend the practical utility of KGEs across a broader range of semantic web and knowledge-intensive applications.

Acknowledgments

Declaration on Generative AI

In preparing this manuscript, generative AI tools—specifically ChatGPT—were used solely for: grammar checking, spelling check, and the readability of some sentences. All suggested changes were carefully reviewed and adapted by the authors to ensure accuracy and appropriateness. The scientific content, research design, analysis, and conclusions were developed and verified exclusively by the authors without AI involvement. The use of ChatGPT was limited to enhancing the presentation of the work.

References

[1] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, BERT: Pre-training of deep bidirectional transformers for language understanding, in: J. Burstein, C. Doran, T. Solorio (Eds.), Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), Association for Computational Linguistics, Minneapolis, Minnesota, 2019, pp. 4171–4186. URL: https://aclanthology.org/N19-1423/. doi:10.18653/v1/N19-1423.

[2] J. Euzenat, P. Shvaiko, Ontology matching, Springer, 2007.

[3] Y. He, J. Chen, H. Dong, I. Horrocks, Exploring large language models for ontology alignment, arXiv preprint arXiv:2309.07172 (2023).

[4] Z. Qiang, W. Wang, K. Taylor, Agent-om: Leveraging llm agents for ontology matching, arXiv preprint arXiv:2312.00326 (2023).

[5] S. Hertling, H. Paulheim, Olala: Ontology matching with large language models, in: Proceedings of the 12th knowledge capture conference 2023, 2023, pp. 131–139.

[6] H. Babaei Giglou, J. D'Souza, F. Engel, S. Auer, Llms4om: matching ontologies with large language models, in: European Semantic Web Conference, Springer, 2024, pp. 25–35.

[7] Q. Wang, Z. Mao, B. Wang, L. Guo, Knowledge graph embedding: A survey of approaches and applications, IEEE transactions on knowledge and data engineering 29 (2017) 2724–2743.

[8] A. Bordes, N. Usunier, A. Garcia-Duran, J. Weston, O. Yakhnenko, Translating embeddings for modeling multi-relational data, Advances in neural information processing systems 26 (2013).

[9] B. Yang, W.-t. Yih, X. He, J. Gao, L. Deng, Embedding entities and relations for learning and inference in knowledge bases, arXiv preprint arXiv:1412.6575 (2014).

[10] T. Trouillon, J. Welbl, S. Riedel, É. Gaussier, G. Bouchard, Complex embeddings for simple link prediction, in: International conference on machine learning, PMLR, 2016, pp. 2071–2080.

[11] P. Ristoski, H. Paulheim, Rdf2vec: Rdf graph embeddings for data mining, in: International semantic web conference, Springer, 2016, pp. 498–514.

[12] J. Chen, P. Hu, E. Jimenez-Ruiz, O. M. Holter, D. Antonyrajah, I. Horrocks, Owl2vec*: embedding of owl ontologies, Machine Learning 110 (2021) 1813–1845.

[13] S. Teymurova, E. Jiménez-Ruiz, T. Weyde, J. Chen, Owl2vec4oa: Tailoring knowledge graph embeddings for ontology alignment, in: International Knowledge Graph and Semantic Web Conference, Springer, 2024, pp. 168–182.

[14] H. Babaei Giglou, J. D'Souza, O. Karras, S. Auer, Ontoaligner: A comprehensive modular and robust python toolkit for ontology alignment, in: European Semantic Web Conference, Springer, 2025, pp. 174–191.

[15] J. Chen, O. Mashkova, F. Zhapa-Camacho, R. Hoehndorf, Y. He, I. Horrocks, Ontology embedding: a survey of methods, applications and resources, IEEE Transactions on Knowledge and Data Engineering (2025).

[16] W. Li, X. Duan, M. Wang, X. Zhang, G. Qi, Multi-view embedding for biomedical ontology matching., OM@ ISWC 2536 (2019) 13–24.

[17] C. Canastra, C. Pesquita, A systematic evaluation of knowledge graph embeddings for gene-disease association prediction, arXiv preprint arXiv:2504.08445 (2025).

[18] J. Kim, K. Kim, M. Sohn, G. Park, Deep model-based security-aware entity alignment method for edge-specific knowledge graphs, Sustainability 14 (2022) 8877.

[19] S. Banerjee, B. R. Chakravarthi, J. P. McCrae, Cross-lingual ontology matching using structural and semantic similarity, in: Proceedings of the 9th Workshop on Linked Data in Linguistics@ LREC-COLING 2024, 2024, pp. 11–21.

[20] J. Chen, Y. He, Y. Geng, E. Jiménez-Ruiz, H. Dong, I. Horrocks, Contextual semantic embeddings for ontology subsumption prediction, World Wide Web 26 (2023) 2569–2591.

[21] Z. Wang, Contextualized structural self-supervised learning for ontology matching, arXiv preprint arXiv:2310.03840 (2023).

[22] J. Huang, Z. Sun, Q. Chen, X. Xu, W. Ren, W. Hu, Deep active alignment of knowledge graph entities and schemata, Proceedings of the ACM on Management of Data 1 (2023) 1–26.

[23] J. Chen, E. Jiménez-Ruiz, I. Horrocks, X. Chen, E. B. Myklebust, An assertion and alignment correction framework for large scale knowledge bases, Semantic Web 14 (2022) 29–53.

[24] S. M. Alghamdi, F. Zhapa-Camacho, R. Hoehndorf, A-lion-alignment learning through inconsistency negatives of the aligned ontologies (2022).

[25] Z. Wang, Amd results for oaei 2022., in: OM@ ISWC, 2022, pp. 145–152.

[26] V. Nguyen, H. Y. Yip, G. Bajaj, T. Wijesiriwardene, V. Javangula, S. Parthasarathy, A. Sheth, O. Bodenreider, Context-enriched learning models for aligning biomedical vocabularies at scale in the umls metathesaurus, in: Proceedings of the ACM Web Conference 2022, 2022, pp. 1037–1046.

[27] Ö. Erten, S. Mehryar, B. Xiong, R. Çelebi, C. Brewster, Refining semopenalex concept ontology: A constraint-aware approach via knowledge graph embeddings and skos constraints, in: 15th International Conference on Semantic Web Applications and Tools for Health Care and Life Sciences, SWAT4HCLS 2024, CEUR-WS, 2024, pp. 56–65.

[28] A. F. Ahmed, M. A. Sherif, A.-C. N. Ngomo, Nellie: Never-ending linking for linked open data, IEEE Access 11 (2023) 84957–84973.

[29] N. Jain, R. Krestel, Discovering fine-grained semantics in knowledge graph relations, in: Proceedings of the 31st ACM International Conference on Information & Knowledge Management, 2022, pp. 822–831.

[30] M. Ali, M. Berrendorf, C. T. Hoyt, L. Vermue, S. Sharifzadeh, V. Tresp, J. Lehmann, PyKEEN 1.0: A Python Library for Training and Evaluating Knowledge Graph Embeddings, Journal of Machine Learning Research 22 (2021) 1–6. URL: http://jmlr.org/papers/v22/20-825.html.

[31] T. Dettmers, P. Minervini, P. Stenetorp, S. Riedel, Convolutional 2d knowledge graph embeddings, in: Proceedings of the AAAI conference on artificial intelligence, volume 32, 2018.

[32] Z. Sun, W. Hu, C. Wang, Y. Wang, Y. Qu, Revisiting embedding-based entity alignment: A robust and adaptive method, IEEE Transactions on Knowledge and Data Engineering 35 (2022) 8461–8475.

[33] T. Wu, C. Gao, L. Li, Y. Wang, Leveraging multi-modal information for cross-lingual entity matching across knowledge graphs, Applied Sciences 12 (2022) 10107.

[34] A. Kalinowski, D. Datta, Y. An, A scalable approach to aligning natural language and knowledge graph representations: Batched information guided optimal transport, in: 2023 IEEE International Conference on Big Data (BigData), IEEE, 2023, pp. 383–392.

[35] G. Ji, S. He, L. Xu, K. Liu, J. Zhao, Knowledge graph embedding via dynamic mapping matrix, in: C. Zong, M. Strube (Eds.), Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), Association for Computational Linguistics, Beijing, China, 2015, pp. 687–696. URL: https://aclanthology.org/P15-1067/. doi:10.3115/v1/P15-1067.

[36] P. Qiu, L. Pang, Y. Luo, Y. Liu, H. Xing, K. Liu, G. Zhuang, Earthquake event knowledge graph construction and reasoning, Geomatics, Natural Hazards and Risk 15 (2024) 2383768.

[37] J. Feng, M. Huang, M. Wang, M. Zhou, Y. Hao, X. Zhu, Knowledge graph embedding by flexible translation., in: KR, 2016, pp. 557–560.

[38] Z. Wang, J. Zhang, J. Feng, Z. Chen, Knowledge graph embedding by translating on hyperplanes, in: Proceedings of the AAAI conference on artificial intelligence, volume 28, 2014.

[39] Y. Lin, Z. Liu, M. Sun, Y. Liu, X. Zhu, Learning entity and relation embeddings for knowledge graph completion, in: Proceedings of the AAAI conference on artificial intelligence, volume 29, 2015.

[40] M. Nickel, L. Rosasco, T. Poggio, Holographic embeddings of knowledge graphs, in: Proceedings of the AAAI conference on artificial intelligence, volume 30, 2016.

[41] Z. Sun, Z.-H. Deng, J.-Y. Nie, J. Tang, Rotate: Knowledge graph embedding by relational rotation in complex space, arXiv preprint arXiv:1902.10197 (2019).

[42] S. M. Kazemi, D. Poole, Simple embedding for link prediction in knowledge graphs, Advances in neural information processing systems 31 (2018).

[43] W. Zhang, B. Paudel, W. Zhang, A. Bernstein, H. Chen, Interaction embeddings for prediction and explanation in knowledge graphs, in: Proceedings of the twelfth ACM international conference on web search and data mining, 2019, pp. 96–104.

[44] R. Abboud, I. Ceylan, T. Lukasiewicz, T. Salvatori, Boxe: A box embedding model for knowledge base completion, Advances in Neural Information Processing Systems 33 (2020) 9649–9661.

[45] S. Vashishth, S. Sanyal, V. Nitin, P. Talukdar, Composition-based multi-relational graph convolutional networks, arXiv preprint arXiv:1911.03082 (2019).

[46] I. Balazevic, C. Allen, T. Hospedales, Multi-relational poincaré graph embeddings, Advances in neural information processing systems 32 (2019).

[47] S. Zhang, Y. Tay, L. Yao, Q. Liu, Quaternion knowledge graph embeddings, Advances in neural information processing systems 32 (2019).

[48] A. Bordes, J. Weston, R. Collobert, Y. Bengio, Learning structured embeddings of knowledge bases, in: Proceedings of the AAAI conference on artificial intelligence, volume 25, 2011, pp. 301–306.

[49] E. Jiménez-Ruiz, O. Hassanzadeh, C. Trojahn, S. Hertling, H. Li, P. Shvaiko, J. Euzenat, Proceedings of the 19th international workshop on ontology matchingco-located with the 23rd international semantic web conference (iswc 2024), in: The 19th International Workshop on Ontology Matching co-located with the 23rd International Semantic Web Conference (ISWC 2024), CEUR-WS. org, 2025.

[50] Z. Dragisic, V. Ivanova, H. Li, P. Lambrix, Experiences from the anatomy track in the ontology alignment evaluation initiative, J Biomed Semant 8 (2017) 56. URL: https://doi.org/10.1186/s13326-017-0166-5. doi:10.1186/s13326-017-0166-5.

[51] N. Karam, A. Khiat, A. Algergawy, M. Sattler, C. Weiland, M. Schmidt, Matching biodiversity and ecology ontologies: challenges and evaluation results, The Knowledge Engineering Review 35

(2020) e9. URL: https://doi.org/10.1017/S0269888920000132. doi:10.1017/S0269888920000132.

[52] E. Blomqvist, H. Li, R. Keskisärkkä, M. Lindecrantz, M. A. N. Pour, Y. Li, P. Lambrix, Cross-domain modelling-a network of core ontologies for the circular economy., in: WOP@ ISWC, 2023, pp. 1–12.

[53] E. Nas, M. Huschka, MSE Benchmark, https://github.com/EngyNasr/MSE-Benchmark, 2023.

[54] Y. He, J. Chen, H. Dong, E. Jiménez-Ruiz, A. Hadian, I. Horrocks, Machine learning-friendly biomedical datasets for equivalence and subsumption ontology matching, in: U. Sattler, A. Hogan, M. Keet, V. Presutti, J. P. A. Almeida, H. Takeda, P. Monnin, G. Pirrò, C. d'Amato (Eds.), The Semantic Web – ISWC 2022, Springer International Publishing, Cham, 2022, pp. 575–591.

[55] P. G. Cotovio, L. Ferraz, D. Faria, L. Balbi, M. C. Silva, C. Pesquita, Matcha-dl a tool for supervised ontology alignment, preprint (2024).

[56] E. Jiménez-Ruiz, B. Cuenca Grau, Logmap: Logic-based and scalable ontology matching, in: International Semantic Web Conference, Springer, 2011, pp. 273–288.

[57] M.-H. Totoian, A. Marginean, P. Blohm, M. N. Hussain, Hybridom: Ontology matching using hybrid search., in: OM@ ISWC, 2024, pp. 138–145.