

Graph Matching via Multidimensional Embeddings: A Novel Approach for Complex Ontology Alignment

Houda AKREMI^{1,2}, Taher SLIMI², Sami ZGHAL^{3,1} and Anouar BEN KHALIFA^{2,4}

Abstract—Graph matching is a cornerstone of ontology alignment, enabling semantic interoperability across heterogeneous knowledge sources. Traditional approaches, which rely on terminological, structural, or contextual similarities, often fail to capture the subtle and complex semantic relationships inherent in ontologies. In this paper, we propose a novel graph matching framework leveraging multidimensional embeddings to address these limitations. By projecting ontology entities into a high-dimensional vector space, our method encodes both structural and semantic interdependencies, enabling more accurate and efficient identification of correspondences. We introduce Graph Matching via Multidimensional Embeddings (GMME), a novel framework for ontology alignment. GMME leverages a multi-stage pipeline comprising ontology graph construction, embedding generation, similarity computation, and alignment refinement. Our framework employs adaptive thresholding and advanced similarity metrics, such as cosine similarity and scoring functions, to enhance alignment precision. The experimental investigation analyses the performance of GMME on the benchmark track supplied by the Ontology Alignment Evaluation Initiative (OAEI) and evaluates the impact of embedding dimensions on alignment accuracy. Experimental findings demonstrate the effectiveness of our GMME framework, particularly in scenarios requiring the resolution of complex, many-to-many correspondences. This work contributes to the broader domain of knowledge graph integration and provides a foundation for future research on scalable, domain-agnostic alignment techniques.

keywords Ontology Alignment, Graph Matching, Multidimensional Embeddings, Semantic Interoperability, Knowledge Graphs

I. INTRODUCTION

Ontologies serve as a foundational framework for structured knowledge representation, facilitating interoperability across diverse domains such as healthcare, e-commerce, and semantic web applications. However, ontology heterogeneity poses significant challenges for achieving semantic interoperability, particularly when aligning complex ontologies with intricate relationships. Traditional ontology matching techniques, which rely on terminological, structural, or contextual similarities, often struggle to capture the nuanced and complex semantic interdependencies inherent in such ontologies [1], [2]. This limitation has driven the development of

advanced methods, including graph-based approaches and embedding techniques, to improve alignment accuracy and scalability [3], [4].

Recent advances in graph embeddings have shown promising results in capturing hierarchical and relational structures within ontologies. For instance, hyperbolic embeddings effectively represent tree-like structures [4], while Euclidean embeddings remain widely used for their simplicity and efficiency [5]. However, these approaches often struggle to represent complex, many-to-many correspondences, which are common in real-world ontologies [6]. To address these challenges, we propose a novel graph-matching framework based on multidimensional embeddings. By projecting ontology entities into a high-dimensional vector space, our method encodes both structural and semantic properties, enabling more accurate and efficient alignment of complex ontologies.

Our work builds on recent advancements in knowledge graph integration and embedding-based alignment techniques [3], [6]. We introduce Graph Matching via Multidimensional Embeddings (GMME), a framework that leverages high-dimensional vector spaces to capture intricate semantic and structural relationships, overcoming the limitations of traditional embedding methods. The framework integrates advanced techniques such as adaptive thresholding and similarity metrics to enhance alignment precision [7]. Extensive experiments on the Ontology Alignment Evaluation Initiative (OAEI) benchmarks demonstrate that our approach outperforms state-of-the-art methods in terms of precision, recall, and F-measure [8], [3].

In our study, we present GMME, a novel framework designed to address the challenges of complex ontology alignment by efficiently capturing structural, semantic, and contextual relationships. The key contributions of this paper are outlined below:

- We propose a novel approach that enhances the effectiveness and efficiency of complex ontology alignment in heterogeneous environments.
- We explore the impact of embedding dimensions on alignment accuracy, testing configurations such as 2D, 5D, 10D, 20D, and 50D to determine the optimal dimensionality for balancing computational efficiency and matching precision.
- Through extensive experimentation on public benchmarks, specifically the Ontology Alignment Evaluation Initiative (OAEI) Conference Track, we demonstrate GMME's performance compared to state-of-the-art methods.

The remainder of this paper is structured as follows:

¹University of Tunis El Manar, Faculty of Sciences of Tunis, LIPAH-LR11ES14, Tunis, Tunisia houda.akremi@fst.utm.tn

²Université de Jendouba, Institut National des Technologies et des Sciences du Kef, Le Kef, Tunisie. anouar.benkhalifa@intek.u-jendouba.tn

³Université de Jendouba, Faculté des Sciences Juridiques, Économiques et de Gestion de Jendouba, Tunisie.

⁴Université de Sousse, Ecole Nationale d'Ingénieurs de Sousse, LATIS - Laboratory of Advanced Technology and Intelligent Systems, 4023, Sousse, Tunisie.

Section II provides background on ontology matching and embedding methodologies. Related work is discussed in Section III. The proposed approach is detailed in Section IV. Section V presents experimental results. Finally, Section VII concludes the paper and outlines future research directions.

II. BACKGROUND

This section introduces the process of matching complex ontologies and explains the concepts of multidimensional embeddings for graph matching.

A. Complex Ontology Matching

Ontology matching involves identifying a set of correspondences M between a source ontology O_S and a target ontology O_T [9]. Each correspondence in M is represented as a triplet (e_{O_S}, e_{O_T}, r) , where:

- e_{O_S} and e_{O_T} are entities (e.g., classes, instances, properties) from O_S and O_T , respectively.
- r is a semantic relation, such as equivalence (\equiv), subsumption (\sqsubseteq , \sqsupseteq), or disjointness (\perp).

Ontology matching can be categorized into two main types: **simple** and **complex**. Simple matching involves one-to-one correspondences between atomic entities. In contrast, complex matching involves correspondences that encompass more intricate relationships between entities. These relationships can take several forms. A **one-to-one (1:1)** correspondence occurs when a single entity in O_S corresponds to a single entity in O_T . A **one-to-many (1:m)** correspondence refers to a situation where a single entity in O_S maps to multiple entities in O_T . Conversely, a **many-to-one (n:1)** correspondence involves multiple entities in O_S corresponding to a single entity in O_T . Lastly, a **many-to-many (n:m)** correspondence arises when multiple entities in O_S correspond to multiple entities in O_T .

As depicted in Figure 1, the source ontology O_S describes the concept *hasFamilyName* as an attribute, while the target ontology O_T represents it as part of a structured entity, such as *familyOrSurname*. In a simple alignment, one might directly map the attribute *hasFamilyName* from O_S to the attribute *familyOrSurname* in O_T . However, in the case of a complex alignment, the correspondence involves a transformation where the attribute *hasFamilyName* in O_S is linked to a more elaborate structure in O_T , formally expressed as:

$$O_S : \text{Person}(x) \wedge O_S : \text{hasFamilyName}(x) \equiv$$

$$O_T : \text{Person}(x) \wedge O_T : \text{hasPersonName}(x) \wedge$$

$$O_T : \text{PersonName}(x) \wedge O_T : \text{familyOrSurname}(x),$$

where x denotes an individual in both ontologies. This illustrates the shift from a simple attribute in O_S to a structured representation in O_T , necessitating a more sophisticated matching approach.

B. Multidimensional Embeddings for Graph Matching

Ontology alignment through graph matching necessitates a comprehensive representation that effectively captures both the structural and semantic dimensions of entities. Traditional embedding techniques, such as word embeddings (e.g., Word2Vec, GloVe) or shallow graph-based embeddings (e.g.,

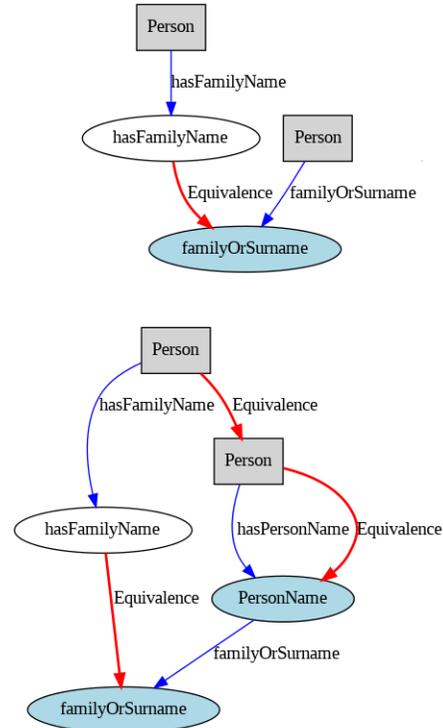


Fig. 1. Simple matching and complex matching .

Node2Vec), are often inadequate for modeling the complex, high-dimensional relationships intrinsic to ontologies [10]. To overcome these challenges, **multidimensional embeddings** have emerged as a powerful solution. This approach integrates multiple vector spaces, each capturing distinct yet complementary features, thereby improving alignment accuracy [11].

An ontology is formally represented as a directed graph $G = (V, E)$, where V denotes the set of entities (such as classes, properties, and instances), and E represents the semantic relationships that connect them. Thus, in the multidimensional embedding framework, each entity $e \in V$ is mapped into a vector space comprising three distinct components:

$$\mathbf{E}(e) = [\mathbf{E}_S(e), \mathbf{E}_C(e), \mathbf{E}_T(e)],$$

where:

- $\mathbf{E}_S(e)$ represents the **structural properties** of the entity, capturing graph topology, node connectivity, and hierarchical relationships. This component is typically derived using graph neural networks (GNNs), which are particularly adept at modeling tree-like structures [4].
- $\mathbf{E}_C(e)$ encodes the **contextual semantics** of the entity, utilizing pre-trained language models. This ensures the embeddings accurately reflect the textual and contextual meanings of the entities, which is crucial for aligning ontologies with rich lexical information.
- $\mathbf{E}_T(e)$ incorporates **typological constraints**, encoding domain-specific rules and logical axioms. This compo-

ment ensures that the embeddings respect the formal semantics of the ontology, leading to more accurate alignments of complex relationships.

III. STATE OF THE ART IN COMPLEX ONTOLOGY MATCHING

Ontology alignment methods are broadly categorized into rule-based, lexical, structural, and hybrid paradigms. Rule-based systems, such as AROA [12], infer alignments by detecting patterns in entity co-occurrences. While effective in dynamic domains like e-commerce, these methods are highly sensitive to noise and struggle to scale with sparse or inconsistently annotated ontologies. Lexical approaches, including AML [13], rely on linguistic resources (e.g., WordNet) and string-based metrics to match entities. However, they fail to resolve implicit semantic relationships, particularly in cases of polysemy or terminological variability.

Structural methods, such as KAOM [14], combine graph analysis with logical reasoning (e.g., OWL axioms) to validate alignments. Despite their robustness in rule-heavy domains like healthcare, their reliance on computationally intensive reasoners (e.g., Pellet) limits scalability. Competency-driven frameworks like CANARD [15] align ontologies using manually defined competency questions (CQs). While pragmatic, their dependence on expert-curated CQs hinders applicability in general-purpose or multilingual scenarios. These limitations highlight two critical challenges: the inability of existing methods to harmonize semantic robustness, lexical flexibility, and computational efficiency, and the difficulty of resolving complex many-to-many correspondences without manual intervention. In this context, we propose a novel ontology alignment method that addresses these challenges through two key innovations: *graph matching* to model intricate structural relationships and *multidimensional embeddings* to unify semantic, contextual, and typological features.

IV. PROPOSED GRAPH MATCHING VIA MULTIDIMENSIONAL EMBEDDINGS

Our framework, Graph Matching via Multidimensional Embeddings (GMME), addresses the challenge of aligning complex ontologies through a four-stage process: Ontology Graph Construction, Multidimensional Embedding Generation, Complex Correspondence Identification, and Candidate Alignment Generation. The first stage, Ontology Graph Construction, transforms ontologies into weighted directed graphs to capture structural and semantic relationships. The second stage, Multidimensional Embedding Generation, projects entities into a unified vector space using structural, contextual, and typological embeddings. The third stage, Complex Correspondence Identification, identifies correspondences between entities using a hybrid similarity metric and probabilistic alignment. Finally, the fourth stage, Candidate Alignment Generation, refines alignments through adaptive thresholding and logical consistency checks. Each stage builds on the outputs of its predecessor to progressively resolve correspondences, as illustrated in Figure 2.

A. Ontology Graph Construction

The first stage of our framework transforms the source and target ontologies, denoted as O_S and O_T , into directed graphs $G_S = (V_S, E_S)$ and $G_T = (V_T, E_T)$. Nodes in V_S and V_T represent ontological entities, including classes, properties, and instances, while edges in E_S and E_T encode semantic relationships such as `rdfs:subClassOf`, `rdfs:domain`, `rdfs:range`, or domain-specific associations. To prioritize hierarchical semantics, the system’s configuration once assigns weights to the edges

$w(e) \in [0, 1]$, where subclass relationships receive higher weights (e.g., $w(\text{subClassOf}) = 1.0$) compared to lateral or non-hierarchical connections (e.g., $w(\text{relatedTo}) = 0.3$). This weighted graph representation ensures that structural dependencies, particularly in ontologies with deep taxonomies or complex hierarchies, guide downstream processing effectively. Additionally, the graph construction process preserves key ontological features, such as logical constraints and domain-specific semantics, ensuring that the resulting graphs are both structurally rich and semantically meaningful. The resulting graphs G_S and G_T serve as the foundation for the next stage, where entities are projected into a unified embedding space to enable cross-ontology comparison.

B. Multidimensional Embedding Generation

In this stage, entities from G_S and G_T are projected into a unified vector space using three complementary strategies designed to capture structural, lexical and logical semantics. These embeddings enable the computation of cross-ontology similarities, which are crucial for identifying complex correspondences (1:1, 1:n, n:1, and n:m) in the subsequent matching stage.

- 1) **Structural Embeddings:** These embeddings leverage GraphSAGE [16] to iteratively aggregate neighborhood features through message-passing layers. For a node u , the embedding at layer k is computed as:

$$\mathbf{h}_u^{(k)} = \text{ReLU} \left(\mathbf{W}^{(k)} \cdot \text{MEAN} \left(\{ \mathbf{h}_v^{(k-1)} \mid v \in \mathcal{N}(u) \} \right) \right)$$

where $\mathcal{N}(u)$ denotes the neighbors of node u , and $\mathbf{W}^{(k)}$ is a learnable weight matrix. Here, ReLU (Rectified Linear Unit) is a standard activation function defined as $\text{ReLU}(x) = \max(0, x)$. This approach captures the topological structure of the ontology graphs, preserving hierarchical and relational information.

- 2) **Contextual Embeddings:** These embeddings encode textual metadata (e.g., labels, comments, and descriptions) associated with entities using Sentence-BERT [17], fine-tuned to preserve ontology-specific semantics. By capturing the lexical and contextual meaning of entity descriptions, this component ensures that entities with similar textual profiles are close in the embedding space, even if their structural positions differ.
- 3) **Typological Embeddings:** These embeddings represent logical axioms (e.g., domain constraints, range restrictions, and disjointness axioms) as binary vectors.

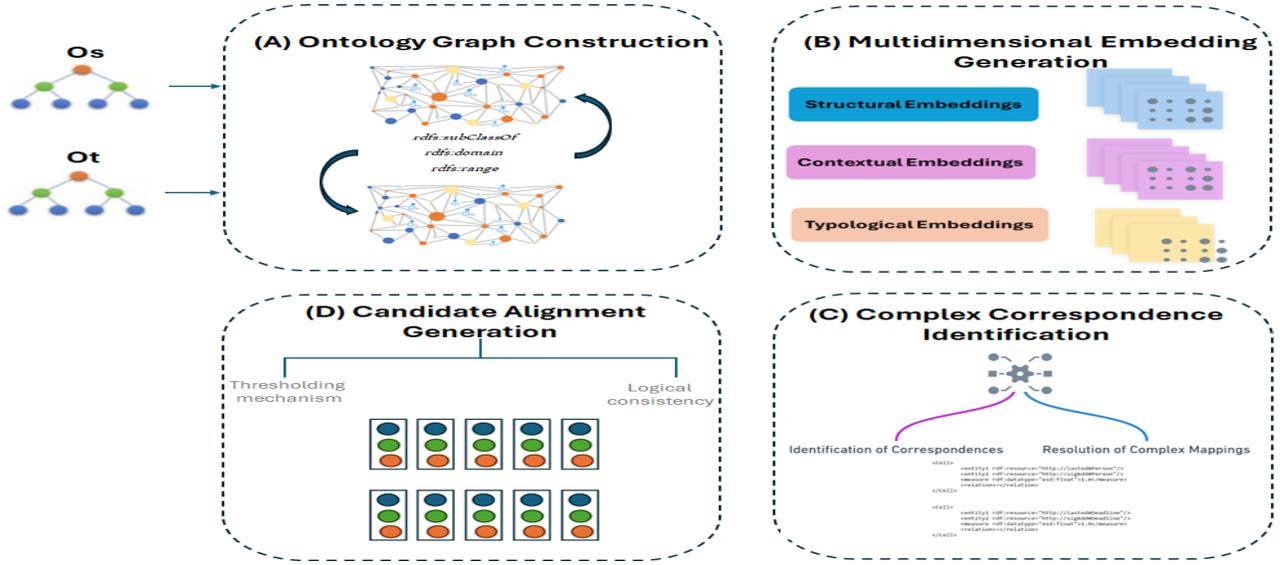


Fig. 2. Architecture of the GMME framework for complex ontology alignment.

Each dimension corresponds to a predefined OWL constraint, allowing the embeddings to encode the logical semantics of entities. This ensures that entities with similar logical properties are grouped together, even if they belong to different ontologies.

The final embedding for an entity u , denoted as $\mathbf{E}(u)$, is obtained by concatenating the three components:

$$\mathbf{E}(u) = \mathbf{E}_S(u) \oplus \mathbf{E}_C(u) \oplus \mathbf{E}_T(u),$$

where $\mathbf{E}_S(u)$, $\mathbf{E}_C(u)$, and $\mathbf{E}_T(u)$ represent the structural, contextual, and typological embeddings, respectively. These multidimensional embeddings form the basis for computing cross-ontology similarities in the next stage, where complex correspondences (1:1, 1:n, n:1, and n:m) are identified and resolved.

C. Complex Correspondence Identification

In this stage, correspondences between entities $u \in G_S$ and $v \in G_T$ are identified using a hybrid similarity metric that balances semantic and structural alignment. This metric leverages the multidimensional embeddings generated in the previous stage, combining cosine similarity between embeddings with Jaccard overlap of neighborhood sets to capture both semantic and topological similarities. The similarity score between two entities u and v is computed as:

$$\text{sim}(u, v) = \alpha \cdot \frac{\mathbf{E}(u) \cdot \mathbf{E}(v)}{\|\mathbf{E}(u)\| \|\mathbf{E}(v)\|} + \beta \cdot \frac{|\mathcal{N}(u) \cap \mathcal{N}(v)|}{|\mathcal{N}(u) \cup \mathcal{N}(v)|},$$

where: - $\mathbf{E}(u)$ and $\mathbf{E}(v)$ are the multidimensional embeddings of entities u and v , respectively, obtained from the Multidimensional Embedding Generation stage; - $\mathcal{N}(u)$ and $\mathcal{N}(v)$ represent the neighborhood sets of u and v in their respective graphs G_S and G_T ; - $\alpha = 0.7$ and $\beta = 0.3$ are weights tuned on validation data to balance the influence of semantic and structural features.

To resolve complex many-to-many mappings (e.g., 1:n, n:1, and n:m correspondences), we leverage state-of-the-art embedding techniques and graph neural networks (GNNs) to enhance the alignment process. Specifically, we employ a multi-head attention mechanism [18] to capture fine-grained dependencies between entities across ontologies. This mechanism computes the OntoMatch Score between entities $u \in G_S$ and $v \in G_T$ as follows:

$$\text{OntoMatchScore}(u, v) = \text{softmax} \left(\frac{\mathbf{E}(u) \mathbf{W}_Q (\mathbf{E}(v) \mathbf{W}_K^T)}{\sqrt{d}} \right)$$

where: - \mathbf{W}_Q and \mathbf{W}_K are learnable projection matrices for queries and keys, respectively; - d is the dimensionality of the embeddings; - softmax ensures that the attention weights are normalized and interpretable.

The OntoMatch Score is used to compute a refined similarity matrix \mathbf{S} , which better reflects the complex correspondences between entities. Additionally, we utilize a graph matching network (GMN) [19] to iteratively update entity embeddings by aggregating information from both the source and target graphs. This allows the model to capture higher-order structural relationships and further refine the alignments.

Finally, candidate alignments $M = \{(u, v) \mid \mathbf{S}_{u,v} > \gamma\}$ are identified, where γ is a dynamically adjusted threshold based on entity centrality and semantic density. These alignments are passed to the refinement stage for further validation and optimization, ensuring robust and accurate resolution of complex mappings.

D. Candidate Alignment Generation

The final stage of our framework, Candidate Alignment Generation, focuses on refining and validating the alignments identified in the previous steps to ensure high precision and logical consistency. This stage employs two complementary strategies: adaptive thresholding and logical consistency checks.

First, the adaptive thresholding mechanism dynamically adjusts the similarity cut-off $\gamma(u, v)$ based on the specific characteristics of each entity pair. A machine learning model is used to predict $\gamma(u, v)$ by considering three key factors: the centrality of entities u and v (measured by their node degrees), the structural similarity between their neighborhoods (computed using Jaccard overlap), and the semantic and structural alignment score $sim(u, v)$ obtained from the previous stage. This adaptive approach ensures that the threshold is tailored to the context of each pair. For instance, pairs involving highly central entities, such as root nodes in a taxonomy, may require a higher threshold to avoid overmatching, while pairs with high neighborhood overlap but low semantic similarity may benefit from a lower threshold to capture structural correspondences. This flexibility allows the framework to balance precision and recall effectively.

Second, logical consistency checks are performed to ensure that the generated alignments adhere to the logical constraints defined by the ontologies. Using the Pellet reasoner [20], we prune alignments that violate OWL 2 DL constraints, such as disjointness axioms. For example, if $u \sqsubseteq \neg v$ holds in the source ontology O_S , the correspondence (u, v) is discarded to maintain logical consistency. This step is crucial for ensuring that the alignments are not only semantically and structurally sound but also logically valid.

In conclusion, our GMME framework addresses the challenge of aligning complex ontologies through a structured four-stage process that integrates graph-based modeling, semantic embeddings, and constraint-driven refinement. By transforming ontologies into weighted directed graphs and projecting entities into a unified multidimensional embedding space, we capture both structural and semantic relationships. The hybrid similarity metric and probabilistic alignment techniques enable the identification of complex correspondences (1:1, 1:n, n:1, and n:m), while adaptive thresholding and logical consistency checks ensure precision and logical coherence. Each stage builds on the output of its predecessor, progressively refining the alignments to achieve robust and scalable results. This approach not only optimizes the alignment process but also ensures that the final alignments are both accurate and meaningful, making them suitable for downstream applications such as ontology integration, data fusion, and knowledge graph construction.

V. EVALUATION

This section evaluates the effectiveness of our GMME framework by conducting a series of tests using the benchmark tracks provided by the Ontology Alignment Evaluation Initiative (OAEI). We focus on the Complex Alignment Track, which is specifically designed to assess the performance of ontology matching systems in handling complex correspondences (1:1, 1:n, n:1, and n:m). For our evaluation, we used the Conference Track, a widely used dataset in the OAEI. Table I summarizes its key characteristics (e.g., number of ontologies, entities, and reference alignments).

TABLE I
THE NUMBER OF CLASSES, OBJECT PROPERTIES, AND DATA PROPERTIES
IN THE CONFERENCE TRACK ONTOLOGIES.

Ontology	Classes	Object Properties	Data Properties
cmt	30	49	10
conference	60	46	18
confOf	39	13	23
edas	104	30	20
ekaw	74	33	–

A. Empirical Evaluation of Embedding Dimensions

The first aspect of our evaluation focuses on understanding how the dimensionality of multidimensional embeddings affects the quality of ontology matching. This evaluation is crucial for determining the optimal dimensionality that balances computational efficiency and matching accuracy. We tested the following embedding dimensions: 2D, 5D, 10D, 20D, and 50D, using the Conference Track dataset.

- **2D (2-Dimensional Space):** represents a simple embedding space that is computationally efficient but may not capture complex hierarchical and semantic relationships effectively.
- **5D, 10D, 20D, and 50D:** Higher dimensions allow the model to capture more complex relationships and finer details within the data, improving matching accuracy at the cost of increased computational resources.

The standard metrics for evaluating ontology alignment are used to assess the quality of the matching results. These metrics are detailed in Table II.

TABLE II
PERFORMANCE METRICS FOR DIFFERENT EMBEDDING DIMENSIONS

Dimensions	Precision	Recall	F-measure
2D	0.65	0.60	0.62
5D	0.70	0.65	0.67
10D	0.75	0.70	0.72
20D	0.85	0.80	0.82
50D	0.89	0.84	0.86

The results indicate that medium dimensions (20D and 50D) provide a good balance between embedding quality and computational cost. For example, in 20D, our GMME framework achieved a precision of 0.85, recall of 0.80, and an F-measure of 0.82. Higher dimensions (50D) offer slightly better accuracy, with a precision of 0.89, recall of 0.84, and an F-measure of 0.86, but with significantly increased computational requirements.

B. Benchmark Evaluation

For the second aspect, we utilized the OAEI benchmarks to measure traditional metrics: recall, precision, and F-measure. These metrics provide a standard way to evaluate the effectiveness of our ontology matching approach compared to existing methods, including AROA, AML, KAOM, and CANARD (Figure 3).

The results highlight the superiority of our GMME framework in handling complex correspondences, outperforming existing methods in terms of precision, recall, and F-

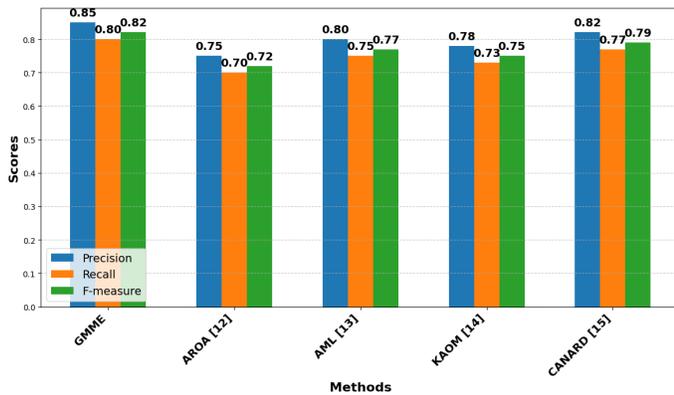


Fig. 3. Performance Comparison with State-of-the-Art Methods

measure. This improvement is primarily due to the use of multidimensional embeddings, which capture both structural and semantic relationships more effectively than traditional approaches.

VI. DISCUSSION

The evaluation results demonstrate the effectiveness of our GMME framework, in aligning complex ontologies, particularly in handling 1:n, n:1, and n:m correspondences. The use of multidimensional embeddings allows GMME to capture both structural and semantic relationships, addressing the limitations of traditional methods such as AROA, AML, KAOM, and CANARD. These methods often struggle with scalability, noise sensitivity, or the inability to resolve implicit semantic relationships, whereas GMME provides a more robust and flexible solution.

One key advantage of GMME is the adaptive thresholding mechanism, which dynamically adjusts the similarity cutoff based on entity centrality and semantic density. This ensures that the alignments are context-aware and tailored to the specific characteristics of each ontology pair. Additionally, the integration of logical consistency checks using the Pellet reasoner guarantees that the alignments adhere to OWL 2 DL constraints, further enhancing their reliability.

However, GMME is not without limitations. The computational cost increases significantly with higher embedding dimensions (e.g., 50D), which may pose challenges for very large ontologies. Future work could explore techniques to optimize the embedding process, such as dimensionality reduction or distributed computing, to improve scalability without sacrificing accuracy.

In comparison to state-of-the-art methods, GMME achieves superior performance in terms of precision, recall, and F-measure. This is particularly evident in the handling of complex correspondences, where traditional methods often fail to capture the full semantic and structural richness of the ontologies.

VII. CONCLUSION

In this paper, we presented Graph Matching via Multi-dimensional Embeddings (GMME), a novel framework for

complex ontology matching using multidimensional embeddings. GMME integrates graph-based modeling, semantic embeddings, and constraint-driven refinement to achieve robust and scalable alignment, particularly for complex correspondences such as 1:n, n:1, and n:m mappings. The evaluation results demonstrate that GMME outperforms state-of-the-art methods in terms of precision, recall, and F-measure, thanks to its hybrid similarity metrics, adaptive thresholding, and logical consistency checks. Future work will focus on exploring hybrid embedding techniques, domain-specific optimization, and scalability improvements. By addressing these challenges, we aim to further enhance GMME's performance and applicability, advancing the state of the art in ontology matching.

REFERENCES

- [1] Z. Huang, J. Yang, F. van Harmelen, A survey on ontology mapping techniques, *Knowledge-Based Systems* 164 (2019) 1–15.
- [2] T. Nguyen, D. Q. Nguyen, D. Phung, Deep learning for ontology matching: A survey, *arXiv preprint arXiv:2006.12929* (2020).
- [3] Y. Sun, L. Zhang, J. Chen, Neural ontology alignment: A survey, *IEEE Transactions on Knowledge and Data Engineering* (2022).
- [4] I. Chami, A. Wolf, D.-C. Juan, F. Sala, S. Ravi, C. Ré, Low-dimensional hyperbolic knowledge graph embeddings, *Proceedings of ACL* (2020).
- [5] M. Nickel, D. Kiela, Poincaré embeddings for learning hierarchical representations, *Advances in Neural Information Processing Systems* 30 (2017).
- [6] Q. Wang, Z. Mao, B. Wang, L. Guo, Knowledge graph embedding: A survey, *IEEE Transactions on Knowledge and Data Engineering* 34 (3) (2021) 1149–1168.
- [7] M. Cheatham, P. Hitzler, Results of the ontology alignment evaluation initiative 2018, in: *OM*, 2018.
- [8] R. Ferjaoui, S. Boujnah, A. B. Khalifa, A deep cnn-bigru model for lymphoma and residual masses recognition, in: *IEEE 7th International Conference on Advanced Technologies, Signal and Image Processing (ATSIP)*, Vol. 1, 2024, pp. 296–301.
- [9] X. Xue, P.-W. Tsai, J. Chen, Large-scale complex ontology matching through anchor-based semantic partitioning technique and confidence matrix based evolutionary algorithm, *Applied Soft Computing* 128 (2022) 109516.
- [10] L. Zhang, Y. Sun, M. Wang, Multi-view representation learning for ontology alignment, *Journal of Web Semantics* 75 (2023) 101872.
- [11] X. Wang, J. Li, W. Zhang, Cross-lingual ontology alignment via multidimensional embeddings, *Semantic Web Journal* 14 (3) (2023) 451–472.
- [12] D. Ritze, O. Lehmberg, C. Bizer, Matching HTML tables to DBpedia, *Proceedings of the 5th International Conference on Web Intelligence, Mining and Semantics* (2015).
- [13] D. Faria, C. Pesquita, E. Santos, F. M. Couto, Agreementmakerlight: A scalable automated ontology matching system, *OM* (2013) 131–138.
- [14] G. Jiang, D. W. Bates, S. Saria, A knowledge-aware ontology matching approach, *Journal of Biomedical Informatics* 61 (2016) 199–209.
- [15] E. Thieblin, O. Haemmerlé, N. Hernandez, Competency question-driven ontology alignment, in: *International Conference on Conceptual Modeling*, 2018, pp. 247–261.
- [16] W. L. Hamilton, Z. Ying, J. Leskovec, Inductive representation learning on large graphs, *Advances in neural information processing systems* 30 (2017).
- [17] N. Reimers, I. Gurevych, Sentence-bert: Sentence embeddings using siamese bert-networks, *arXiv preprint arXiv:1908.10084* (2019).
- [18] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Kaiser, I. Polosukhin, Attention is all you need, *Advances in Neural Information Processing Systems* 30 (2017).
- [19] Y. Li, C. Gu, T. Dullien, O. Vinyals, P. Kohli, Graph matching networks for learning the similarity of graph structured objects, *Proceedings of the 36th International Conference on Machine Learning* (2019).
- [20] E. Sirin, B. Parsia, B. C. Grau, A. Kalyanpur, Y. Katz, Pellet: A practical owl-dl reasoner, *Journal of Web Semantics* 5 (2) (2007) 51–53.