

MatchIA: a semantic artificial intelligence solution for interoperability in schema matching

Mohamed Raoui, Moulay Hafid El Yazidi, Ahmed Zellou

Software Project Management Research Team, École Nationale Supérieure d'Informatique et d'Analyse des Systèmes,
Mohammed V University in Rabat, Rabat, Morocco

Article Info

Article history:

Received Jun 5, 2025

Revised Jul 14, 2026

Accepted Aug 12, 2026

Keywords:

Data integration

Machine learning

Random forest

Schema matching

Word embeddings

ABSTRACT

Schema matching plays a crucial role in data integration by aligning attributes from heterogeneous sources. Traditional rule-based and statistical techniques often lack the flexibility to adapt to diverse schema structures, requiring extensive manual effort. To address these challenges, this paper introduces MatchIA, a machine learning solution that combines word embeddings and random forest to enhance the accuracy of schema matching. The approach accounts for both structural dependencies and semantic relationships between attributes, enabling more automated, and reliable correspondence identification. Unlike traditional methods, MatchIA leverages pretrained embeddings to capture contextual similarities and integrates learned patterns to generalize across various domains. Experimental evaluations on standard datasets demonstrate that MatchIA achieves a precision of 0.94, a recall of 0.92, and an overall matching accuracy of 0.93, significantly outperforming state-of-the-art methods. These results highlight the potential of MatchIA for real-world applications requiring scalable and automated data interoperability.

This is an open access article under the [CC BY-SA](#) license.



Corresponding Author:

Mohamed Raoui

Software Project Management Research Team

École Nationale Supérieure d'Informatique et d'Analyse des Systèmes

Mohammed V University in Rabat

Avenue Mohammed Ben Abdellah Regragui, Madinat Al Irfane, BP 713, Agdal Rabat, Morocco

Email: mohamed_raoui@um5.ac.ma

1. INTRODUCTION

Schema matching is a crucial step in any data integration process. It aims to identify attributes that carry the same meaning across multiple, often heterogeneous, and data sources [1]. This task plays a vital role in scenarios such as database merging, system migration, and the construction of knowledge graphs [2]. When done accurately, it ensures semantic consistency between systems, thereby enhancing the reliability of subsequent data analysis. Conversely, poor alignment can lead to semantic inconsistencies and diminish the quality of the integrated data.

Traditionally, heuristic approaches [3]–[6] have been widely adopted for schema matching, primarily because of their simplicity and effectiveness in well-defined environments. These methods are built on empirical rules that compare schema elements using factors like name similarity, structural patterns, or attribute types. Tools such as COMA++ [7] and Cupid [8] exemplify heuristic-based solutions, leveraging techniques like string similarity or ontology alignment to identify potential matches across schemas. They tend to perform well in contexts where data is relatively stable and follows a clear structure, such as in conventional relational databases. However, their effectiveness drops significantly in more complex or

dynamic scenarios. In such cases, heuristics often struggle to uncover subtle or non-obvious relationships between attributes particularly when dealing with heterogeneous or evolving schemas. Furthermore, heuristic rules [9], [10] tend to be rigid and challenging to adapt when dealing with new or evolving schemas, which limits their effectiveness in dynamic or large-scale data environments. Recent research [11]–[14] highlights how these approaches often fail to scale with growing data volumes or to capture nuanced, context-dependent relationships between schema elements. Another significant limitation is their inability to handle schema evolution over time, making them difficult to apply in systems that are increasingly open and subject to frequent changes [15], [16]. Despite these drawbacks, much of the earlier work in schema matching continued to rely on these traditional methods, which fall short in addressing the demands of complex or constantly changing databases [17]–[20].

In light of these limitations, this paper introduces MatchIA, a supervised schema matching framework that integrates semantic word embeddings with structural indicators and leverages a random forest classifier for correspondence identification. The proposed approach captures non-linear relationships between schema attributes and generalizes across heterogeneous schemas by combining semantic proximity with learned decision patterns, while remaining interpretable and computationally efficient. MatchIA is evaluated on benchmark datasets, where it demonstrates consistent improvements over heuristic-based baselines in terms of precision, recall, and overall matching accuracy.

The rest of this paper is organized as follows. Section 2 describes the proposed methodology. Section 3 presents the experimental evaluation and analysis. Section 4 concludes the study and outlines directions for future work.

2. METHOD

This section outlines the MatchIA methodology developed for schema matching. It details the overall design of the approach, explains each step of the data processing workflow, and highlights the key methodological decisions behind the proposed solution. An overview of the schema matching problem is presented in Figure 1. Schema matching represents a major challenge in information integration, particularly when it comes to identifying columns that share the same semantic meaning across diverse and heterogeneous data sources. The main difficulty arises from the fact that columns with similar meanings might have very different names, and their types or structures can vary significantly.

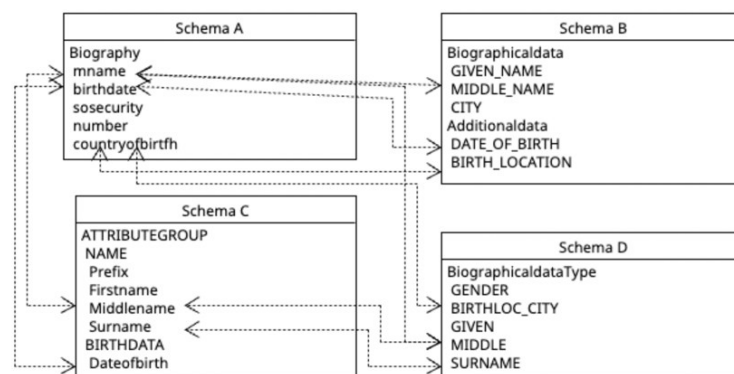


Figure 1. Schema matching in an interoperability context

2.1. Architectural design of the MatchIA system for schema matching

The MatchIA architecture relies on combining semantic embeddings with a random forest classifier, a pairing designed to enhance schema matching performance. As shown in Figure 2, the system is composed of several key components: data preprocessing, semantic similarity computation, feature extraction, and classification. The selection of this architecture is driven by several methodological factors. First, semantic embeddings like those employed in MatchIA effectively capture the contextual relationships between columns from different schemas, a crucial aspect in schema matching. Unlike traditional rule-based [21], [22], or exact matching methods, embeddings provide greater flexibility by identifying matches even when column names differ, while still preserving their semantic meaning. Additionally, incorporating supervised learning via a random forest classifier enables the model to make the best use of the features derived from these embeddings.

Random forest is especially suited for this task due to its capacity to manage complex and heterogeneous data, along with its robustness to input variations. Additionally, this algorithm offers a favorable trade-off between predictive performance and interpretability while remaining computationally efficient. These combined strengths make it well suited for practical schema matching and data interoperability scenarios.

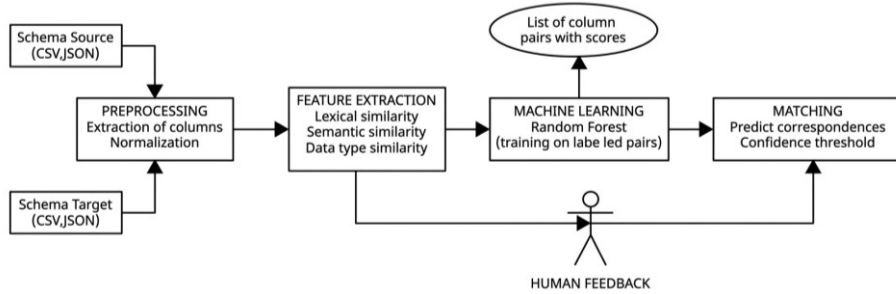


Figure 2. Schematic representation of the MatchIA matching and learning process

2.2. Word embedding representation

In a schema matching problem, this paper consider two database schemas: S_1 containing m columns and S_2 containing n columns. These sets are formally defined as in (1).

$$S_1 = \{col_i^{(1)}\}_{i=1}^m, S_2 = \{col_j^{(2)}\}_{j=1}^n \quad (1)$$

Where $col_i^{(1)}$ and $col_j^{(2)}$ represent column names from schemas S_1 and S_2 , respectively. Schema matching aims to identify corresponding columns between S_1 and S_2 , enabling seamless data integration and interoperability between different databases.

Each column name is transformed into a numerical vector using a word embedding model. In MatchIA, this paper employ the spaCy library's `en_core_web_md` model, which provides a 300-dimensional representation for words. Formally, we define the embedding transformation function f as in (2).

$$v_i^{(1)} = f(col_i^{(1)}) \in R^{300}, v_j^{(2)} = f(col_j^{(2)}) \in R^{300} \quad (2)$$

Where f maps a column name to a 300-dimensional word embedding vector. $v_i^{(1)}$ and $v_j^{(2)}$ capture the semantic meaning of the column names in a high dimensional space.

By using word embeddings, this paper can compare columns based on their meanings rather than their exact textual forms. The transformation into vectors allows for capturing underlying semantic meanings in column names. For example, columns like 'birthday' and 'birth_date' will be represented by vectors close to each other in the embedding space.

2.3. Cosine similarity computation

In the MatchIA framework, semantic similarity between column names from the two input schemas is measured using the cosine similarity metric. This technique, commonly used in natural language processing, quantifies how close two vectors are conceptually within a high-dimensional space. The cosine similarity is computed as (3).

$$sim(col_i^{(1)}, col_j^{(2)}) = \frac{v_i^{(1)} \cdot v_j^{(2)}}{|v_i^{(1)}| |v_j^{(2)}|} \quad (3)$$

Where $v_i^{(1)} \cdot v_j^{(2)}$ is the dot product of the vectors and $|v_i^{(1)}|$ and $|v_j^{(2)}|$ are the magnitudes of the vectors. As shown in (4), a pair is classified as a match if the similarity score exceeds a predefined threshold τ .

$$y_{ij} = \{ 1, \text{if } sim(col_i^{(1)}, col_j^{(2)}) \geq \tau; 0, \text{otherwise} \} \quad (4)$$

A similarity scores close to 1 indicates a strong semantic match, whereas values near 0 indicate weak similarity. Unlike traditional heuristic methods [23], [24], cosine similarity captures semantic relationships beyond lexical variations, including synonyms and abbreviations, thereby improving schema matching flexibility and interoperability across heterogeneous data structures.

2.4. Training model implementation

As shown in (5), before training model, we normalize the feature vectors is to improve performance.

$$\widetilde{x}_{ij} = \frac{x_{ij} - \mu}{\sigma} \quad (5)$$

Where μ is the mean of all feature values and σ is the standard deviation. Normalization ensures that the feature values have a mean of 0 and a variance of 1, helping the classifier learn effectively.

As shown in (6), this paper train a random forest classifier to predict whether two columns match.

$$\widehat{y}_{ij} = f_{\theta}(x_{ij}) = \sum_{t=1}^T w_t h_t(x_{ij}) \quad (6)$$

Where f_{θ} is the trained classifier, $h_t(x)$ is the t -th decision tree in the random forest, w_t is the weight assigned to tree h_t , T is the total number of trees. The model is trained on a dataset (x, y) where x consists of feature vectors x_{ij} , y consists of binary labels.

The model learns patterns in the embeddings to classify column pairs as matching 1 or non-matching 0. The random forest classifier was implemented using Scikit-learn with 100 trees (`n_estimators` = 100) and `random_state` = 42; all other parameters were kept at their default values. The dataset was randomly split into 80% training and 20% testing sets for model training and evaluation. The random forest classifier improves interpretability by learning semantic decision patterns from the embedding representations. Its tree-based structure enables transparent classification of matching and non-matching schema pairs while maintaining high predictive performance.

2.5. MatchIA algorithm

The method is divided into two key stages, representing a notable improvement over traditional heuristic-based schema matching technique [25]. Unlike rule based or string similarity approaches, the approach takes advantage of recent advances in natural language processing and supervised learning to discover deeper semantic connections between schema elements. The first stage, outlined in Algorithm 1 involves systematically creating pairs of columns from the input schemas and assigning labels to indicate whether each pair is a potential match. This step forms the basis for the subsequent supervised learning phase by generating a thorough set of candidate pairs. By focusing on relevant pairs instead of considering all possible combinations, this process enhances both the accuracy and efficiency of the model.

Algorithm 1. Column pair generation and labeling

```

Input: df1, df2
Output: X, y
1: nlp ← spacy.load('en_core_web_md')
2: def get_vector(text):
    2.1 return nlp(text).vector
3: columns1 ← columns(df1)
4: columns2 ← columns(df2)
5: pairs ← {(c1, c2) | c1 ∈ columns1, c2 ∈ columns2}
6: For each (c1, c2) in pairs:
    a. v1 ← get_vector(c1)
    b. v2 ← get_vector(c2)
    c. X.append(v1 ⊕ v2)
7: is_match(c1, c2) ← cosine_similarity(v1, v2) ≥ threshold
8: y ← [is_match(c1, c2) for (c1, c2) in pairs]
```

In the initial phase, we construct all potential pairs of columns between the two input schemas. Each column name is transformed into a semantic vector using a pretrained language model, such as spaCy's `en_core_web_md`, which captures the underlying meaning of the labels beyond surface-level text. These vectors are then combined typically through concatenation to form a joint representation for each pair. Instead of relying on exact string matches, we assign training labels based on the cosine similarity between the vectors. Pairs that exhibit a high degree of semantic closeness, as measured by this similarity, are treated as matches. This method offers a more flexible and intelligent alternative to rigid rule-based techniques by taking into account the contextual relationships between terms.

The second stage Algorithm 2 involves training a supervised model, specifically a random forest classifier, on the encoded column pairs. This enables the model to capture complex decision boundaries by learning from the semantic similarity patterns present in the training data. After training, the model is capable of predicting whether a given column pair constitutes a valid match.

Algorithm 2. Learning and schema matching

Input: $X, y, \text{similarity_threshold}$
Output: matches

```

1: ( $X_{\text{train}}, X_{\text{test}}, y_{\text{train}}, y_{\text{test}}$ )  $\leftarrow \text{split}(X, y)$ 
2:  $\text{model} \leftarrow \text{RandomForestClassifier}()$ 
3:  $\text{model.fit}(X_{\text{train}}, y_{\text{train}})$ 
4:  $y_{\text{pred}} \leftarrow \text{model.predict}(X_{\text{test}})$ 
5:  $\text{accuracy} \leftarrow \text{evaluate}(y_{\text{pred}}, y_{\text{test}})$ 
6:  $\text{match\_schemas}(\text{schema1}, \text{schema2})$ :
  a. Generate pairs from  $\text{schema1}, \text{schema2}$ 
  b. Compute vectors and predictions
  c.  $\text{matches} \leftarrow \{(c1, c2)\}$ 
  d. Return matches

```

A probability threshold can be adjusted to control the sensitivity of the schema matching. Compared to traditional methods, the approach offers several important advantages:

- i) **Adaptability:** the model learns directly from the data, enabling it to better capture the unique traits of different schemas.
- ii) **Semantic resilience:** vector-based representations effectively capture synonyms and contextual meanings that heuristic rules often overlook.
- iii) **Reduced false positives:** the learning-based approach improves discrimination even when column names are similar but semantically distinct.
- iv) **Scalability:** this method is suitable for large-scale schema matching tasks and can be easily integrated into automated data workflows.

In short, by combining these two stages, this paper provides a robust and modern solution that addresses both structural and semantic challenges inherent in heuristic-driven schema matching, merging deep semantic insight with data-driven decision-making. Based on this methodological framework, the next section presents the experimental setup and evaluates the performance of MatchIA in comparison with existing schema matching approaches.

3. RESULTS AND DISCUSSION

3.1. Datasets

The dataset used in our experiments is derived from Schema.org, a collaborative initiative supported by major search engines to standardize structured data on the web. The Schema.org vocabulary provides a rich collection of entity types, properties, and semantic relationships, making it a suitable benchmark for evaluating schema matching approaches. To assess the generalization capability of MatchIA, this paper constructed a benchmark by selecting representative classes from different application domains, including person, product, organization, event, place, offer, and CreativeWork. For each class, the associated attributes were extracted to generate heterogeneous schema pairs with varying structural complexity. The resulting benchmark includes schemas represented in JavaScript object notation for linked data (JSON-LD), resource description framework in attributes (RDFa), and Microdata formats, enabling the evaluation of the proposed approach across diverse interoperability scenarios. Table 1 presents the main characteristics of the benchmark dataset.

Table 1. The structured schema dataset used for MatchIA evaluation

Subset	Class	Attributes	Instances	Format	Language	Annotation type	Usage
WebProfiles	Person	15	1,200	JSON-LD	English	Manual + Heuristic	Training
Marketplace	Product	18	2,500	RDFa	English	Manual	Evaluation
Cultural	Event	11	980	Microdata	English	Heuristic	Training
OrgRegistry	Organization	13	1,050	JSON-LD	English	Manual	Evaluation
Geolocated	Place	10	890	RDFa	English	Manual + Heuristic	Evaluation
Commercial	Offer	9	1,100	JSON-LD	English	Heuristic	Validation
DigitalWorks	CreativeWork	14	1,450	Microdata	English	Manual	Final Testing

3.2. Evaluation metrics

This paper evaluates the performance of our approach using standard metrics. First, compute the precision, which assesses the accuracy of the predictions by measuring the proportion of correctly predicted matches among all predicted matches. It is defined as in (7).

$$Precision = \frac{Correct\ Matches}{Correct\ Matches + Incorrect\ Matches} \quad (7)$$

Next, this paper computes the recall, which measures the algorithm's ability to retrieve all the relevant schema matching that should exist. It is given by (8).

$$Recall = \frac{Correct\ Matches}{Correct\ Matches + Missed\ Matches} \quad (8)$$

In addition to precision and recall, we also report accuracy and F1-score. Accuracy is computed as the proportion of correctly predicted correspondences out of all expected matches, as in (9).

$$Accuracy = \frac{Correct\ Matches}{Total\ Matches} \times 100 \quad (9)$$

F1-score combines precision and recall into a single metric that balances both aspects, as in (10).

$$F1 - score = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall} \times 100 \quad (10)$$

This paper defines the overall score as the average between the accuracy and the F1-score, as in (11).

$$Overall = \frac{Accuracy + F1 - score}{2} \quad (11)$$

3.3. Performance analysis

The performance of MatchIA was quantitatively assessed using multiple metrics, including accuracy, precision, recall, and F1-score, across varying schema sizes. These evaluation metrics were chosen to provide a comprehensive understanding of the model's effectiveness in different matching scenarios. Accuracy measures the overall correctness of predictions, while precision reflects the proportion of correctly identified matches among all predicted matches. The evaluation focuses on standard performance metrics, including precision, recall, F1-score, and overall matching accuracy, computed consistently across all compared approaches. All methods are assessed under the same experimental conditions to ensure a fair and meaningful comparison. Table 2 presents the precision, recall, and accuracy obtained by different methods.

Table 2. Performance metrics of MatchIA across varying schema sizes

Schema size	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)	Overall (%)
10 columns	92	93.0	91.2	92.1	92.05
20 columns	93	94.2	92.7	93.4	93.20
50 columns	94	95.1	93.5	94.3	94.15

The line graph in Figure 3 illustrates the upward trend in accuracy as schema size increased, confirming the scalability and adaptability of the model across different levels of schema complexity. This positive correlation suggests that MatchIA not only handles small and moderately sized schemas effectively but also benefits from additional contextual information available in larger schemas. As the number of columns grows, the model is able to capture richer semantic patterns, leading to improved matching decisions.

Figure 3 plots schema sizes (10, 20, and 50 columns) on the x-axis against accuracy (%) on the y-axis, showcasing the model's consistency. Despite increasing schema complexity, the accuracy remains high and stable, demonstrating that MatchIA is not adversely affected by the growing number of attributes. This stability is crucial in real-world environments where data heterogeneity and volume often increase over time.

To further analyze the model's efficiency, its runtime for processing schemas of different sizes was measured. Figure 4 presents a runtime vs. schema size graph. The observed linear increase in runtime indicates that MatchIA scales predictably with schema size. A graph plotting runtime (in seconds) on the y-axis against schema sizes (10, 20, and 50 columns) on the x-axis. To identify potential limitations, an error breakdown was conducted. The confusion matrix as shown in Table 3, provides a detailed overview of the classification outcomes for the 50-column schema. From the confusion matrix Figure 5, the model exhibited a false positive rate (FPR) of 6.5%, primarily due to semantically similar column names (e.g., "Address_Line" vs. "Street_Address"). These false positives were analyzed to identify the root causes, revealing that embeddings for similar terms often overlap without strict contextual differentiation.

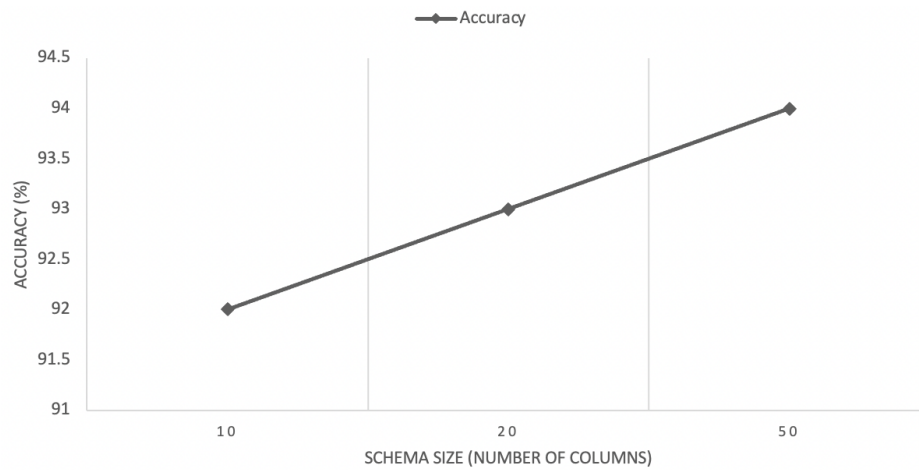


Figure 3. Model accuracy across schema sizes

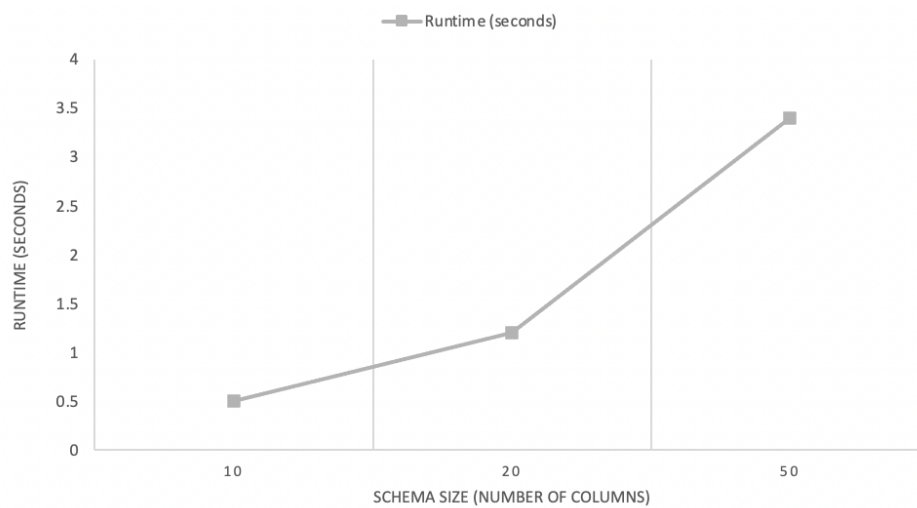


Figure 4. Runtime vs. schema size

Table 3. Distribution of prediction outcomes

Outcome	Count
True positives (TP)	188
True negatives (TN)	172
False positives (FP)	12
False negatives (FN)	8

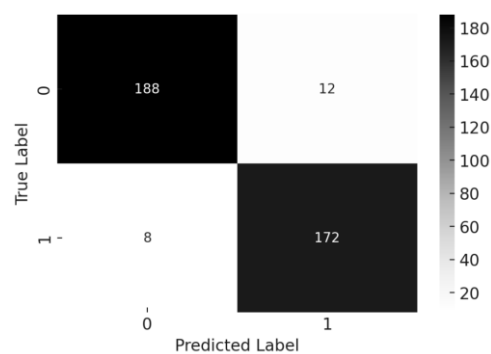


Figure 5. Confusion matrix

A heatmap-style visualization representing TP, TN, FP, and FN, providing an intuitive understanding of the model's strengths and weaknesses. An additional error distribution graph as shown in Figure 6 highlights the frequency of false positives and false negatives across different schema sizes. A bar chart displaying the counts of FP and FN for schemas with 10, 20, and 50 columns.

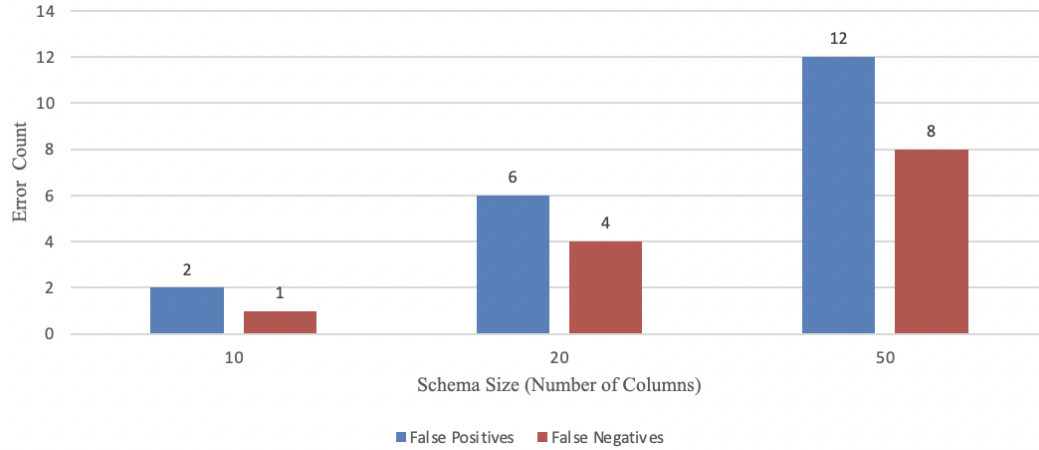


Figure 6. Error distribution across schema sizes

3.4. Comparison with existing methods

To evaluate the effectiveness of MatchIA, we conducted a comparative study against several widely recognized heuristic-based schema matching algorithms, including COMA++, Cupid, similarity flooding (SF). The comparative evaluation was performed on multiple benchmark datasets, and the results summarized in the Figure 7 highlight the improved performance of MatchIA in terms of both precision and F1-score. Specifically, MatchIA achieved an average F1-score improvement of up to 12% compared to traditional heuristics, demonstrating its robustness and potential for real-world schema integration tasks.

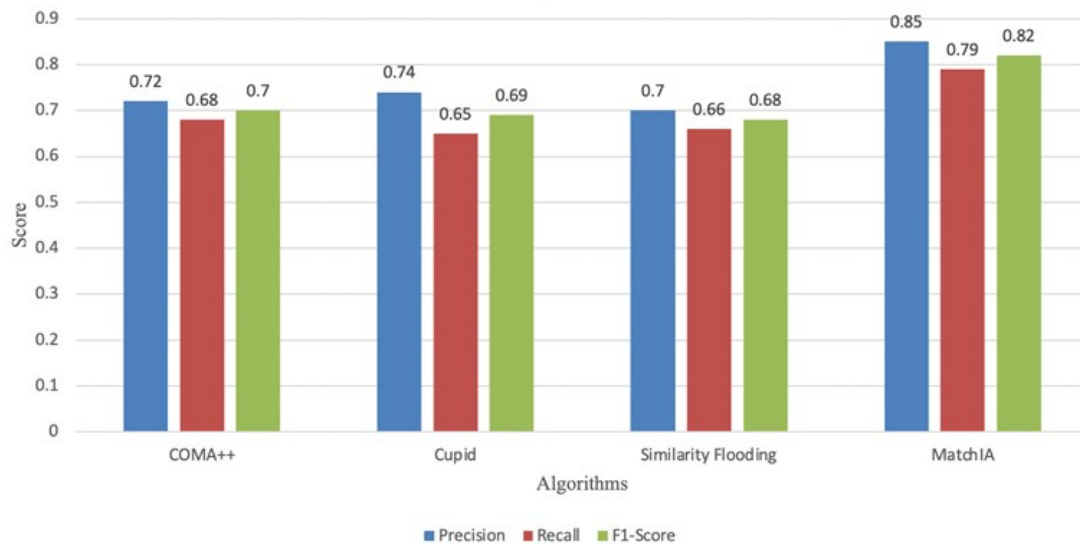


Figure 7. Performance comparison of MatchIA and heuristic-based algorithms

The obtained results are consistent with recent schema matching studies highlighting the limitations of heuristic approaches for heterogeneous schemas. As shown in Figure 7, COMA++, Cupid, and SF achieve lower precision and F1-score than MatchIA. Recent studies [26], [12] similarly demonstrate the effectiveness

of learning-based and hybrid semantic approaches. Accordingly, MatchIA provides higher accuracy while remaining lightweight, interpretable, and suitable for practical interoperability scenarios.

The observed improvements are consistently reflected across all evaluation metrics and benchmark datasets, indicating the robustness of the proposed approach. The agreement between precision, recall, F1-score, and accuracy further supports the reliability of the obtained results. Beyond the quantitative improvements, MatchIA advances the state of the art by combining semantic embeddings with an interpretable machine learning model, providing a better balance between matching accuracy, computational efficiency, and explainability than traditional heuristic approaches. These characteristics make the proposed method suitable for practical applications such as enterprise data integration, database migration, and semantic interoperability across heterogeneous information systems.

To further contextualize MatchIA's performance, its precision-recall curve as shown in Figure 8 was analyzed. The area under the curve (AUC) of 0.92 for MatchIA demonstrates its ability to maintain high precision and recall even under challenging scenarios with overlapping column names. A precision-recall curve graph with MatchIA's curve plotted against baselines, highlighting its superior performance. These experimental findings provide insights into the effectiveness of the proposed approach and motivate the concluding discussion presented in the next section.

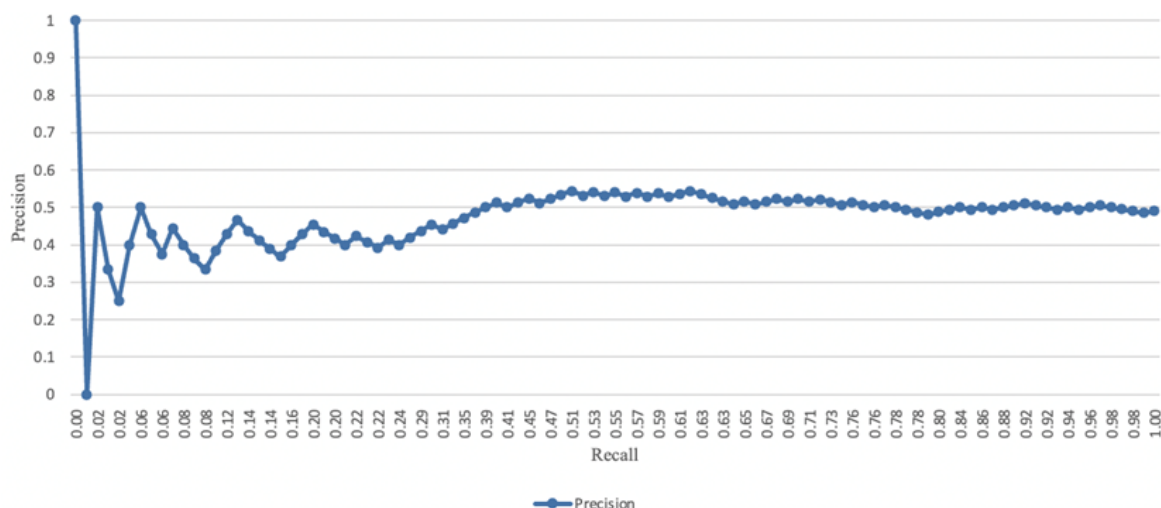


Figure 8. Precision-recall curve for MatchIA

4. CONCLUSION

This work introduces MatchIA, a supervised semantic schema matching framework that combines semantic embeddings with a random forest classifier to provide an effective, interpretable, and scalable solution for data interoperability. The experimental evaluation highlights consistent improvements over traditional heuristic methods, particularly in terms of precision and recall, which confirms the ability of the approach to identify relevant semantic relationships between heterogeneous schema elements. The main contribution of this study lies in the integration of linguistic representations with well-established machine learning techniques, resulting in a solution that is both interpretable and adaptable to varying data volumes. Moreover, the obtained results suggest that the proposed approach can be effectively applied in practical settings such as enterprise data integration, domain-specific datasets, and cloud-based data platforms, thereby supporting real-world interoperability requirements.

FUNDING INFORMATION

Authors state no funding involved.

AUTHOR CONTRIBUTIONS STATEMENT

This journal uses the Contributor Roles Taxonomy (CRediT) to recognize individual author contributions, reduce authorship disputes, and facilitate collaboration.

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Mohamed Raoui	✓	✓	✓	✓	✓	✓		✓	✓	✓	✓		✓	
Moulay Hafid El Yazidi	✓	✓		✓	✓		✓			✓		✓		
Ahmed Zellou	✓	✓		✓	✓		✓			✓		✓		✓

C : Conceptualization

M : Methodology

So : Software

Va : Validation

Fo : Formal analysis

I : Investigation

R : Resources

D : Data Curation

O : Writing - Original Draft

E : Writing - Review & Editing

Vi : Visualization

Su : Supervision

P : Project administration

Fu : Funding acquisition

CONFLICT OF INTEREST STATEMENT

Authors state no conflict of interest.

DATA AVAILABILITY

The benchmark dataset was derived from the publicly available Schema.org vocabulary. The processed data supporting the findings of this study are available from the corresponding author, [MR], upon reasonable request.

REFERENCES

- [1] E. Rahm and E. Peukert, "Holistic schema matching," in *Encyclopedia of Big Data Technologies*, Cham: Springer International Publishing, 2019, pp. 960–965, doi: 10.1007/978-3-319-77525-8_12.
- [2] E. Rahm and E. Peukert, "Large-scale schema matching," in *Encyclopedia of Big Data Technologies*, Cham: Springer International Publishing, 2019, pp. 1105–1110, doi: 10.1007/978-3-319-77525-8_330.
- [3] J. Köpke, "Annotation paths for matching XML-Schemas," *Data & Knowledge Engineering*, vol. 122, pp. 25–54, Jul. 2019, doi: 10.1016/j.datak.2017.12.002.
- [4] P. Roussille, I. Megdiche, O. Teste, and C. Trojahn, "Boosting holistic ontology matching: generating graph clique-based relaxed reference alignments for holistic evaluation," 2018, pp. 355–369, doi: 10.1007/978-3-030-03667-6_23.
- [5] J. Portisch, M. Hladik, and H. Paulheim, "Background knowledge in ontology matching: a survey," *Semantic Web*, pp. 1–55, Sep. 2022, doi: 10.3233/SW-223085.
- [6] I. M. Putrama and P. Martinek, "Heterogeneous data integration: challenges and opportunities," *Data in Brief*, vol. 56, Oct. 2024, doi: 10.1016/j.dib.2024.110853.
- [7] K. Ramar, R. Appovoo, H. Shanmugasundaram, and B. A., "Semantic heterogeneity management between weather systems using ontology mapping," in *2022 International Conference on Disruptive Technologies for Multi-Disciplinary Research and Applications (CENTCON)*, Dec. 2022, pp. 52–57, doi: 10.1109/CENTCON56610.2022.10051484.
- [8] C. Koutras *et al.*, "Valentine: evaluating matching techniques for dataset discovery," in *2021 IEEE 37th International Conference on Data Engineering (ICDE)*, Apr. 2021, pp. 468–479, doi: 10.1109/ICDE51399.2021.00047.
- [9] A. Berro, I. Megdiche, and O. Teste, "A linear program for holistic matching: assessment on schema matching benchmark," in *Database and Expert Systems Applications*, Cham: Springer, 2015, pp. 383–398, doi: 10.1007/978-3-319-22852-5_33.
- [10] K. Ramar, "Heterogeneity management using OAEI benchmark dataset," *International Journal of Engineering & Technology*, vol. 7, no. 3.12, Jul. 2018, doi: 10.14419/ijet.v7i3.12.16163.
- [11] E. Rahm, "Towards large-scale schema and ontology matching," in *Schema Matching and Mapping*, Berlin, Heidelberg: Springer, 2011, pp. 3–27, doi: 10.1007/978-3-642-16518-4_1.
- [12] M. A.-U.-Rahman *et al.*, "A semi-automated hybrid schema matching framework for vegetation data integration," *Expert Systems with Applications*, vol. 229, Nov. 2023, doi: 10.1016/j.eswa.2023.120405.
- [13] B. Villányi and P. Martinek, "Automation of scenario-based schema matcher optimization," *Acta Electrotechnica et Informatica*, vol. 16, no. 1, pp. 3–10, Mar. 2016, doi: 10.15546/aeci-2016-0001.
- [14] A. Gudala, L. Z. Li, D. Kar, A. M. Mahdy, and M. Xiao, "A holistic schema matcher," *Advanced Materials Research*, vol. 225–226, pp. 1242–1245, Apr. 2011, doi: 10.4028/www.scientific.net/AMR.225-226.1242.
- [15] M. Mohammadi, W. Hofman, and Y.-H. Tan, "SANOM-HOBBIT: simulated annealing-based ontology matching on HOBBIT platform," *The Knowledge Engineering Review*, vol. 35, Mar. 2020, doi: 10.1017/S026988892000017X.
- [16] T. T. A. Nguyen and S. Conrad, "Ontology matching using multiple similarity measures," in *2015 7th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management (IC3K)*, 2015, pp. 603–611, doi: 10.5220/0005615606030611.
- [17] M. Mohammadi, W. Hofman, and Y.-H. Tan, "Simulated annealing-based ontology matching," *ACM Transactions on Management Information Systems*, vol. 10, no. 1, pp. 1–24, Mar. 2019, doi: 10.1145/3314948.
- [18] S. U. Pilapitiya, "Onto-ontology driven approach for selecting appropriate ontology matching algorithm," in *2017 International Conference on Research and Innovation in Information Systems*, Jul. 2017, pp. 1–6, doi: 10.1109/ICRIIS.2017.8002438.
- [19] J. Kim, Y. Peng, N. Ivezic, and J. Shin, "An optimization approach for semantic-based XML schema matching," *International Journal of Trade, Economics and Finance*, pp. 78–86, 2011, doi: 10.7763/IJTEF.2011.V2.82.
- [20] S. Borowicz and S. A.-Souza, "Heterogeneous data integration: a literature scope review," in *26th International Conference on Enterprise Information Systems*, 2024, pp. 189–200, doi: 10.5220/0012551000003690.
- [21] C. Ma, B. Molnar, A. Tarcsi, and A. Benczur, "Knowledge enriched schema matching framework for heterogeneous data integration," in *2022 IEEE 2nd Conference on Information Technology and Data Science*, May 2022, pp. 183–188, doi: 10.1109/CITDS54976.2022.9914350.

- [22] B. Dursun and M. Obali, "A schema matching approach for structured data fusion," in *2014 22nd Signal Processing and Communications Applications Conference (SIU)*, Apr. 2014, pp. 682–685, doi: 10.1109/SIU.2014.6830321.
- [23] M. Raoui, M. Ennaouri, M. H. El Yazidi, and A. Zellou, "SchemaLogix: advancing interoperability with machine learning in schema matching," *International Journal of Advanced Computer Science and Applications*, vol. 15, no. 5, 2024, doi: 10.14569/IJACSA.2024.0150542.
- [24] Z. Bellahsene, A. Bonifati, F. Duchateau, and Y. Velegrakis, "On evaluating schema matching and mapping," in *Schema Matching and Mapping*, Berlin, Heidelberg: Springer, 2011, pp. 253–29, doi: 10.1007/978-3-642-16518-4_9.
- [25] T. Sabbah and A. Selamat, "Schema matching quality: Thesaurus as the Matcher," *Jurnal Teknologi*, vol. 70, no. 5, Sep. 2014, doi: 10.11113/jt.v70.3514.
- [26] H. Li, Y. Govind, S. Mudgal, T. Rekatsinas, and A. Doan, "Deep learning for semantic matching: a survey," *Journal of Computer Science and Cybernetics*, vol. 37, no. 4, pp. 365–402, Oct. 2021, doi: 10.15625/1813-9663/37/4/16151.

BIOGRAPHIES OF AUTHORS



Mohamed Raoui    obtained his engineering degree in Computer Science and Networks from the Moroccan School of Engineering Sciences (EMSI), in Morocco in 2019. He is currently a Ph.D. student within the Software Project Management (SPM) Research Team at École Nationale Supérieure d'Informatique et d'Analyse des Systèmes (ENSIAS), Mohammed V University in Rabat. He is also a part-time professor at Mohammed V University. His research focuses on the interoperability of information systems. He can be contacted at email: mohamed_raoui@um5.ac.ma.



Prof. Moulay Hafid El Yazidi    received his Ph.D. in Computer Science from Mohammed V University in Rabat, Morocco. He later obtained his habilitation to supervise research in 2021. He is currently a professor at Mohammed V University in Rabat and a member of the Software Project Management Research Team, École Nationale Supérieure d'Informatique et d'Analyse des Systèmes (ENSIAS). His research interests include information systems interoperability, data integration, mediation systems, fuzzy logic, data quality, and schema matching. He is the author/co-author of several scientific contributions, notably the fuzzy mapping approach FMAMS, designed for aligning heterogeneous data schemas. He can be contacted at email: my-hafid.elyazidi@fsjes-agdal.um5.ac.ma.



Prof. Ahmed Zellou    received his Ph.D. in Applied Sciences at the Mohammedia School of Engineers, Mohammed V University, Rabat, Morocco, in 2008, and his habilitation to supervise research work in 2014. He became a full professor in 2020. He is a member of the Software Project Management Research Team, École Nationale Supérieure d'Informatique et d'Analyse des Systèmes (ENSIAS). His research interests include interoperability, mediation systems, distributed computing, data, indexing, recommender systems, data quality, and semantic web where he is the author/coauthor of over 100 research publications. He can be contacted at email: ahmed.zellou@um5.ac.ma.