TOMATO: results of the 2024 OAEI evaluation campaign

Philippe Roussille[1], Olivier Teste[2]

[1] *École 3iL, Limoges, Institut de Recherche en Informatique de Toulouse, Toulouse, France*
[2] *Université Toulouse 2 Jean Jaurés, Institut de Recherche en Informatique de Toulouse, Toulouse, France*

Abstract
This paper presents the results obtained by TOMATO in the OAEI 2024 evaluation campaign. We describe here the results in the Conference track and our first . We report a general discussion on the results and future improvements of the system.

1. Presentation

1.1. Overview

TOMATO (***TO**olkit for **MAT**ching **O**ntologies*) takes inspiration from previous work on ontology matching systems such as POMAP++ [1]. TOMATO is designed as a pairwise matcher, aligning pairs of input ontologies against each other. At its core, TOMATO utilizes machine learning approaches to learn from element similarities. In earlier versions, it focused mainly on string-based similarity measures of ontology elements [2].

Challenges in Previous Versions. Earlier iterations of TOMATO, such as the 2023 system [3], heavily relied on reference alignments for training machine learning models. This dependency led to issues of overfitting and limited generalizability when applied to new datasets. Furthermore, the lack of external ground truth for most OAEI tracks posed a challenge to developing robust machine learning approaches.

Shift Towards Machine Learning and Scalable Alignment. In 2024, TOMATO made significant strides by shifting towards a more machine learning-centric approach while minimizing dependency on reference alignments. However, the challenge remains: how can machine learning models be trained without a reliable ground truth? To address this, several strategies were explored:

- **Artificial Alignment Generation**. Leveraging large language models (LLMs) such as ChatGPT and XLM-Roberta, we generated hypothetical alignments between ontology entities, which were used to train models without relying on reference alignments.

OM-2024: The 19th International Workshop on Ontology Matching collocated with the 23rd International Semantic Web Conference (ISWC 2024), November 11th, Baltimore, USA.

Philippe.Roussille@irit.fr (P. Roussille); Olivier.Teste@irit.fr (O. Teste)

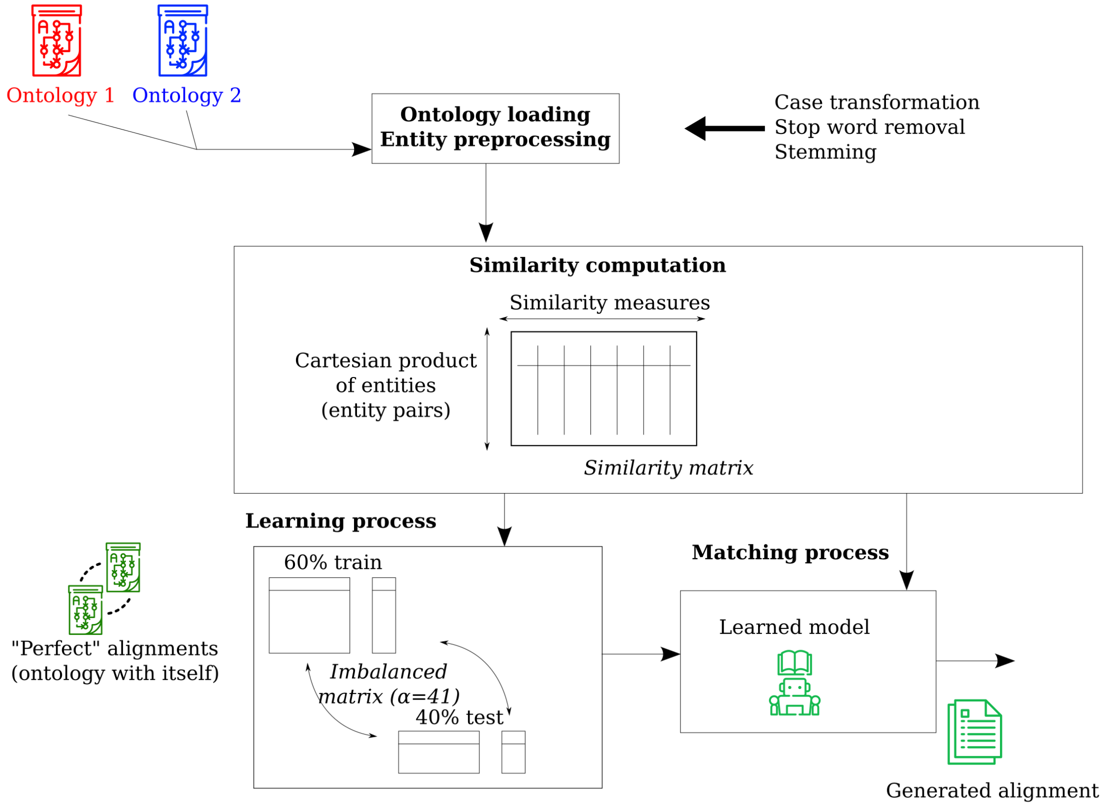

Figure 1: System workflow

- **Exploring Alternative Model Architectures**. To better interpret the learned alignment patterns, we experimented with models such as Decision Trees. These models provided insights into the significance of various similarity metrics, but faced limitations in generalizing across ontologies.
- **Self-Alignment as a Baseline**. A quick and practical solution was self-alignment, where an ontology was aligned with itself. This allowed us to generate accurate correspondences, which improved precision but led to a drop in recall due to a lack of variability.

1.2. Workflow

The workflow in TOMATO begins by taking as input two OWL ontologies to be aligned, along with their associated reference RDF alignment files. These reference alignments provide examples of entity matches that serve as ground truth for the learning process.

TOMATO then prepares the data by combining all ontology entity pairs (both matched and unmatched based on our training data) into a single dataset. This set contains all possible entity couples across the input ontologies, along with their reference match status.

From here, TOMATO can be used in one of two modes:

1. Learning mode: A machine learning model is trained on the full mixed dataset to learn the patterns of matching vs non-matching entities.
2. Matching mode: A pre-trained model is applied to new ontology entity pairs to predict their alignment status.

Unlike previous versions where we explored a wide range of similarity measures, our focus this year has shifted toward finding ways to train models without relying on reference alignments. This change stems from the recognition that reference alignments, while useful, introduce biases that may limit the generalization of the system to unseen ontologies.

In 2024, our approach centers on the development of alternative strategies for generating training data, such as creating artificial alignments using large language models and leveraging self-alignment techniques. These methods provide the system with diverse examples of both trivial and complex matches, allowing it to learn general alignment patterns without depending on explicit reference data. This way, TOMATO is able to explore a broader spectrum of alignment examples, improving its robustness and adaptability to various ontology domains.

1.3. Matching steps

The initial step in TOMATO's workflow is to parse and ingest the input ontologies. We leverage the `owlready2` Python library to build an in-memory representation containing all ontology elements and relationships.

Each entity such as classes, data properties and object properties is indexed via a unique identifier, with the preferred identifier being the element's label if present. As a fallback, the final segment of the entity's URI is used.

Structural relationships between classes are also extracted and stored. This includes superclass-subclass links as well as relations defined through object properties.

By fully populating an internal graph structure in this way, TOMATO is able to consider both lexical properties of entities as well as their positions and connections within the ontology taxonomy during the matching process. This combined view aims to capture more contextual evidence about intended semantic correspondence compared to considering elements in isolation.

The loaded ontologies can then be queried as needed during the various steps of similarity computation, model training and alignment prediction.

1.3.1. Ontology Preprocessing

As in prior iterations, we apply standard text preprocessing techniques to clean and normalize entity labels before computing similarities. This includes:

- Converting CamelCase to snake_case
- Replacing non-alphanumeric symbols with spaces
- Performing English stemming
- Removing stop words

1.3.2. Train and match

Training a Model. We have shifted our focus this year from relying on reference alignments to training our models using alternative methods. Instead of using predefined reference alignments to label entity pairs as matches (1) or non-matches (0), we now generate training data by aligning an ontology with itself. This ensures a perfect match for all entities, providing a reliable ground truth.

In this approach, we construct a similarity matrix for the ontology pair by comparing an ontology to itself. This matrix is then fed into a pre-trained matching model, which is now based on self-alignment data. By using an ontology's internal structure to generate matches, we aim to create robust training examples that improve the model's ability to generalize across different ontologies.

Computing Alignments. A similarity matrix is constructed for the target ontology pair. This matrix is then input to the pre-trained matching model, which outputs the predicted entity alignments.

This year, we successfully broke a significant barrier regarding memory usage by optimizing the overall memory footprint of the system. Through careful optimization of the internal data structures and efficient management of the similarity matrix, we were able to substantially reduce the memory requirements. This allowed TOMATO to handle larger and more complex ontologies without encountering memory overloads, an issue that had previously limited our participation in certain OAEI tracks.

However, despite these improvements, there remain some unresolved issues, likely related to how Docker is used internally for managing the computational processes. These issues occasionally result in suboptimal performance, particularly when handling very dense or highly interconnected ontologies. Despite these challenges, we were still able to successfully participate in the Anatomy Track for the first time, demonstrating the robustness of TOMATO's alignment capabilities.

By optimizing memory usage and tackling system scalability, we have made significant progress in improving TOMATO's overall performance. Nevertheless, addressing the remaining Docker-related issues is a key area for future development.

2. The problem for ground truth

2.1. Generating Artificial Alignments with Large Language Models

One of the first strategies we explored was the generation of artificial alignments using large language models (LLMs) such as ChatGPT and XLM-Roberta. The idea behind this approach was to leverage the capabilities of these advanced models to produce hypothetical alignments between ontology entities, while explicitly excluding reference alignments provided by the OAEI.

As demonstrated by He et al. [4], large language models are capable of identifying semantic relationships between terms, but they struggle with the complexities of specialized terminologies and structures found in ontologies.

The goal was to generate plausible correspondences that could serve as training data for our machine learning models. By excluding the reference alignments, we aimed to address the issue raised in previous OAEI campaigns regarding the use of such resources. Large language models, trained on vast amounts of text data, are capable of identifying semantic relationships between terms, and we hoped to capitalize on this ability to create a diverse set of alignment examples.

Challenges Encountered. However, this approach quickly revealed several significant challenges. First and foremost, the size, structure, and exhaustive nature of ontology descriptions had a direct impact on the performance of the LLMs. Ontologies often consist of highly specialized terminology, complex relationships, and formal structures, which LLMs may struggle to accurately interpret or generate.

The **quantity and form of ontology descriptions** – whether detailed or concise—heavily influenced the quality of the alignments produced. In some cases, the artificial alignments generated by the models were too generic or semantically incorrect, lacking the necessary precision to serve as useful training data. This issue was exacerbated by the complexity of ontological relationships, such as hierarchical structures and role-based associations, which LLMs failed to capture consistently.

As a result, the alignments we produced with this method quickly proved to be **unusable** for training purposes. The generated correspondences either did not align with the real-world complexity of ontology matching or introduced too much noise, rendering them ineffective for model learning.

2.2. Exploring Alternative Model Architectures

In parallel to generating artificial alignments, we explored alternative machine learning model architectures to address the issue of not relying on reference alignments. Our primary approach was to experiment with models such as Decision Trees, which offer a more interpretable representation of the matching process. The goal of this approach was to better understand the importance of different similarity measures and how they contribute to successful alignments.

Decision Trees offer interpretable alignment models, as highlighted in Fleissner et al. [5] and Kokash et al. [6], where the use of Decision Trees for clustering and alignment tasks is explored.

By using models like Decision Trees, we aimed to generate explicit rules that could guide the alignment process. Unlike black-box models, such as neural networks, Decision Trees provide clear decision paths, allowing us to analyze how various similarity metrics—lexical, structural, and semantic—are weighted and applied to different types of correspondences (e.g., class vs. property alignments).

Challenges Encountered. While Decision Trees offered greater interpretability, they also introduced several challenges. First, the complexity of ontology alignment often exceeds the capacity of simple rule-based models. Ontologies involve intricate hierarchical structures, multiple types of relationships, and a high degree of semantic ambiguity, all of which are difficult to capture with rigid, predefined rules.

As we tested this approach, it became clear that Decision Trees struggled to generalize across different ontologies. The alignment rules generated were often specific to the training data and did not perform well when applied to new, unseen ontologies. In addition, the **depth and complexity of the trees** quickly escalated as we introduced more features and similarity metrics, leading to overfitting. This overfitting resulted in poor generalization and a lack of robustness when the model encountered ontologies with different structures or domain-specific terminologies.

Although this approach provided valuable insights into the role of various similarity measures, it ultimately highlighted the need for more flexible models capable of handling the nuanced and complex nature of ontology alignment. Despite the limitations, the experiment underscored the importance of better feature selection and the need for hybrid approaches that combine rule-based reasoning with more adaptive learning techniques.

2.3. Self-alignment as a quick-and-dirty solution, which ultimately worked

As a rapid solution to the training problem, we implemented an approach where the system aligns an ontology with itself. This method was introduced as a quick workaround to avoid using reference alignments, while still providing a baseline for the machine learning model to learn from. The self-alignment ensures that all correspondences between entities are correct by design, as they involve the same ontology being aligned to itself.

Initial Results. During testing, we found that this method allowed us to maintain an acceptable level of precision, although it led to a noticeable drop in recall. Since the system was trained on perfect correspondences (i.e., each entity aligning with itself), it struggled to generalize to more complex inter-ontology alignments, where correspondences are less obvious or more semantically intricate.

Despite this drop in recall, the *precision remained high*, as the system could accurately identify correct matches within the self-alignment framework. The key insight from this experiment is that while recall suffers due to the lack of variability in the training data, precision can be maintained in a controlled environment. This suggests that the model still benefits from understanding simpler correspondences, which may serve as a foundation for more complex learning in future iterations.

Ongoing Use of This Approach. Given the stability of the precision, we decided to retain this approach for the time being, using it as a foundational method while exploring other strategies for improving recall. We maintained the same α *parameter* configuration as last year, which controls the balance between positive and negative examples during training. This allowed us to keep the same level of precision while investigating ways to enrich the model's ability to capture more complex correspondences.

This approach has also proved useful in reinforcing the insights gained from the first two strategies. In particular, it has highlighted how the generation of artificial noise and the exploration of different model architectures can help refine the system's overall alignment capabilities. Moving forward, we plan to combine self-alignment with other techniques, such as artificial alignment generation and feature-driven model exploration, to further enhance TOMATO's performance.

3. Results of the OAEI 2024 Conference Track

TOMATO participated again this year in the Conference Track of the OAEI, where the objective is to align academic domain ontologies. The Conference Track remains a key environment for evaluating matching systems, as it involves dense and diverse ontologies that represent concepts and relationships within academic contexts, such as publications, authors, conferences, and institutions.

3.1. Raw results

Table 1
Performance Results for OAEI 2024 Conference Track

System	Precision	F1-measure	Recall	Threshold
TOMATO	0.57	0.49	0.43	0.0
StringEquiv	0.76	0.52	0.43	0.0
MDMapper	0.67	0.62	0.55	0.0
LogMap	0.77	0.63	0.58	0.0
Matcha	0.66	0.69	0.61	0.0

3.2. Comparison

Table 2
Comparison of TOMATO Performance in OAEI 2023 and 2024

Year	Precision	F1-measure	Recall
2023	0.57	0.52	0.47
2024	0.57	0.49	0.43

3.3. Overall Performance

In 2024, TOMATO showed a slight decrease in performance compared to the previous year. The F-measure dropped from 0.52 in 2023 to 0.49 this year, mainly due to a drop in recall, which went from 0.47 to 0.43. Precision, however, remained stable at 0.57, indicating that TOMATO continues to produce accurate alignments, though with reduced coverage, i.e., fewer relevant correspondences are being found.

3.4. Comparison with 2023

Compared to last year's results, TOMATO's recall decreased, contributing to the overall drop in the F-measure. However, it is worth noting that TOMATO maintained a good precision score, demonstrating that the correspondences it does identify tend to be correct. This stability in precision is encouraging, as it suggests a solid foundation that can be built upon by improving recall in the future.

3.5. Approach and Future Directions

This year, the primary focus for TOMATO was on avoiding the use of reference alignments for model training, addressing concerns raised in the previous OAEI campaign. Despite this shift, TOMATO managed to maintain a consistent precision, although the recall dropped slightly due to the exclusion of reference alignments from the training process.

Looking ahead, we aim to improve TOMATO's recall by incorporating external knowledge sources, such as knowledge graphs, to capture more complex and subtle correspondences. Further experimentation with hybrid strategies combining machine learning with heuristic-based matching could also enhance TOMATO's performance on large-scale, diverse ontology matching tasks.

4. Results of the OAEI 2024 Anatomy Track

TOMATO participated for the first time in the Anatomy Track, which involves aligning the Adult Mouse Anatomy with a section of the NCI Thesaurus representing human anatomy. This track presents unique challenges due to the large size and technical complexity of the ontologies involved. These ontologies are meticulously designed and contain highly specialized terms. Furthermore, they differ from other ontologies in their use of specific annotations and roles, such as the extensive use of the *partOf* relation.

4.1. Performance and Challenges

TOMATO's performance on the Anatomy Track was moderate, placing it behind established systems like LogMap and Matcha in terms of F-measure and recall. The following table summarizes TOMATO's performance compared to other systems:

Table 3
Performance on the Anatomy Track

Matcher	Runtime (s)	Size	Precision	F-Measure	Recall	Recall+
Matcha	42	1485	0.951	0.941	0.931	0.82
LogMapBio	1346	1549	0.888	0.898	0.908	0.757
MDMapper	121	1441	0.926	0.903	0.881	0.703
LogMap	12	1402	0.917	0.881	0.848	0.602
ALIN	370	1156	0.984	0.851	0.75	0.489
LogMapLite	2	1147	0.962	0.828	0.728	0.288
TOMATO	**2154**	**572**	**0.955**	**0.523**	**0.36**	**0.024**
StringEquiv	-	946	0.997	0.766	0.622	0.000

TOMATO achieved a precision of 0.955, which is among the highest in the track. However, its recall was significantly lower at 0.36, leading to a moderate F-measure of 0.523. The low recall indicates that TOMATO struggled to capture a large number of relevant correspondences, especially non-trivial matches. In terms of recall+, which measures non-trivial correspondences (i.e., those that do not have identical labels), TOMATO's performance was the lowest in the track (0.024).

One key limitation observed during this evaluation was related to memory usage and execution speed. The ontologies in the Anatomy Track are not only large but also structurally complex, which increases the computational demands during the alignment process. Despite efforts to optimize TOMATO's performance, we encountered memory bottlenecks and execution times that exceeded acceptable thresholds, particularly when dealing with large datasets.

4.2. Areas for Improvement

The Anatomy Track highlights several areas where TOMATO needs improvement.

- **Memory management.** The large size of the ontologies, coupled with their detailed conceptualization and numerous annotations, requires a more efficient use of memory. Current limitations have led to performance bottlenecks, particularly when processing the full scope of the anatomy ontologies.
- **Execution speed.** Given the complexity of the relationships in these ontologies, such as the extensive use of the *partOf* relation, TOMATO's runtime has proven slower than expected (2154 seconds). We need to further optimize the system to handle these large, hierarchical structures more efficiently.

Despite these challenges, TOMATO has demonstrated its potential in managing simpler correspondences, and we are confident that with improved memory handling and speed optimizations, it can perform more competitively in future Anatomy Track evaluations. Future work will focus on reducing computational overhead while maintaining the precision levels observed during this year's evaluation.

5. Conclusion and Future Directions

This work highlights the key challenges associated with applying machine learning (ML) approaches to ontology matching, particularly in the absence of a reliable ground truth. Unlike traditional matching systems, ML models rely heavily on training data to learn patterns of similarity between entities. The lack of a dedicated ground truth complicates this process, as generating accurate and unbiased training examples becomes a significant hurdle. In the OAEI context, the prohibition against using reference alignments for training further exacerbates this issue, leaving ML-based systems like TOMATO at a disadvantage compared to heuristic-based systems.

However, in certain tracks, such as the medical ontologies, this challenge could be mitigated by leveraging external knowledge sources. Medical ontologies, for instance, benefit from large, well-established resources like SNOMED CT and the Unified Medical Language System (UMLS), which provide robust, validated knowledge bases that could serve as a proxy for ground truth. By incorporating such external resources, ML-based approaches could be trained more effectively, leading to more accurate and generalized matching results in those domains.

We plan to integrate *clustering-based noise generation* with our existing strategies to improve TOMATO's performance. In addition, exploring the use of external knowledge bases in tracks where they are available (e.g., medical ontologies) could provide a more stable ground for training ML models. Finally, we will continue refining the feature selection process and hybrid approaches, combining rule-based reasoning with ML, to address the inherent complexity of ontology alignment tasks.

References

[1] A. Laadhar, F. Ghozzi, I. Megdiche, F. Ravat, O. Teste, F. Gargouri, POMap++ results for OAEI 2019: fully automated machine learning approach for ontology matching, in: 14th International Workshop on Ontology Matching co-located with the International Semantic Web Conference (OM@ISWC 2019), Auckland, New Zealand, 2019, pp. 169–174. URL: https://hal.archives-ouvertes.fr/hal-02942337.

[2] M. Cheatham, P. Hitzler, String similarity metrics for ontology alignment, in: H. Alani, L. Kagal, A. Fokoue, P. Groth, C. Biemann, J. X. Parreira, L. Aroyo, N. Noy, C. Welty, K. Janowicz (Eds.), The Semantic Web – ISWC 2013, Springer Berlin Heidelberg, Berlin, Heidelberg, 2013, pp. 294–309.

[3] P. Roussille, O. Teste, TOMATO : results of the 2023 OAEI evaluation campaign, in: 18th International Workshop on Ontology Matching (OM 2023) co-located with ISWC 2023, volume 3591, Athènes, Greece, 2023, pp. 191–199. URL: https://hal.science/hal-04524356.

[4] Y. He, J. Chen, H. Dong, I. Horrocks, Exploring large language models for ontology alignment, 2023. URL: https://arxiv.org/abs/2309.07172. arXiv:2309.07172.

[5] M. Fleissner, L. C. Vankadara, D. Ghoshdastidar, Explaining kernel clustering via decision trees, 2024. URL: https://arxiv.org/abs/2402.09881. arXiv:2402.09881.

[6] N. Kokash, L. Makhnist, Using decision trees for interpretable supervised clustering, SN Computer Science 5 (2024) 268. URL: https://doi.org/10.1007/s42979-023-02590-7. doi:10.1007/s42979-023-02590-7.