TrietNLP at ATE-IT: A Hybrid Pipeline for Italian Waste Management Terminology Analysis

Nguyen Minh Triet[1,2,*], Dang Van Thin[1,2]

[1]University of Information Technology, Ho Chi Minh City, Vietnam
[2]Vietnam National University, Ho Chi Minh City, Vietnam

Abstract

This paper presents the approach of the TrietNLP team for the ATE-IT 2025 Shared Task on Automatic Term Extraction and Clustering in the domain of Municipal Solid Waste Management. The task involves two subtasks: Automatic Term Extraction (Subtask A) and Term Variants Clustering (Subtask B). For Subtask A, we propose a supervised neural architecture leveraging XLM-RoBERTa Large combined with a Conditional Random Field (CRF) layer to handle sequence labeling and boundary detection. For Subtask B, we employ a hybrid methodology that integrates rule-based pre-clustering for morphological variants with a Generative AI approach (Gemini 2.5) for semantic clustering. Our system utilizes incremental context injection to maintain consistency across batches. This report details the system architecture, preprocessing strategies, and the specific optimization techniques applied to address the lexical variability of Italian administrative texts.

Keywords

Term Extraction, Clustering, Natural Language Processing, XLM-RoBERTa, Generative AI, Waste Management

1. Introduction

The automatic identification and standardization of domain-specific terminology constitute a fundamental challenge in Natural Language Processing, particularly when applied to specialized administrative sectors. The ATE-IT 2025 shared task, organized within the EVALITA evaluation campaign [1], focuses on these challenges within the specific domain of Municipal Solid Waste Management. The data provided for this task comprises administrative documents, regulations, and official notices from Italian municipalities. This corpus presents significant linguistic hurdles, including high lexical variability, regional phrasings, abbreviations, and the necessity to distinguish between technical nomenclature and common language usage.

The shared task is divided into two interconnected subtasks designed to facilitate information standardization:

- **Subtask A: Automatic Term Extraction** is a sentence-level information extraction task. The objective is to automatically identify the boundaries of domain-specific terms, which may range from unigrams to complex multi-word expressions. A key challenge in this subtask involves handling nested terms to produce a flat list of technical spans.
- **Subtask B: Term Variants Clustering** functions as an unsupervised or semi-supervised clustering task. The goal is to group unorganized terms referring to the same operational concept into unified clusters. This requires the system to recognize inflectional variants, acronyms expansions, and true synonyms while avoiding the grouping of hierarchical concepts such as hypernyms.

In this paper, we present a comprehensive pipeline approach tailored to the specific constraints of the Italian waste management domain. Our contribution consists of two distinct strategies optimized for each subtask. For the extraction phase, we employ a neural sequence labeling architecture utilizing **XLM-RoBERTa** combined with a **Conditional Random Field** layer, implementing a longest-match-first strategy to resolve nested term conflicts. For the clustering phase, we propose a hybrid methodology

EVALITA 2025: Evaluation Campaign of Natural Language Processing and Speech Tools for Italian
*Corresponding author.
23521652@gm.uit.edu.vn (N. M. Triet); thindv@uit.edu.vn (D. V. Thin)

that integrates rule-based logic with the semantic reasoning capabilities of Generative AI (specifically Gemini 2.5) to achieve high-precision grouping of synonyms and variants.

2. System Description

Our proposed system adopts a pipeline architecture that treats the two subtasks independently, employing a Neural Sequence Labeling approach for term extraction and a Hybrid (Rule-based + LLM) strategy for clustering.

2.1. Subtask A: Automatic Term Extraction

We model Subtask A as a token-level sequence labeling problem. Given the domain-specific nature of the dataset (waste management administrative acts), our approach focuses on robust context embedding and structural output validity.

2.1.1. Data Preprocessing and Labeling Strategy

The input data undergoes standard normalization, including Unicode NFKC normalization and lowercasing. We also remove specific noise characters (e.g., brackets [], ()) while preserving alphanumeric tokens.

A significant challenge in the provided dataset is the presence of "nested terms" (e.g., the term *"rifiuti urbani"* appearing inside *"trattamento rifiuti urbani"*). Standard sequence labeling models typically require a flat label structure. To address this, we implement a **Longest-Match-First** strategy during the creation of the gold-standard BIO (Beginning-Inside-Outside) labels:

- **Sorting:** All gold terms for a sentence are sorted by length in descending order.
- **Greedy Assignment:** We iterate through the sorted terms. If a term's span overlaps with an already labeled span, it is discarded. This prioritizes the longest maximal semantic units (e.g., keeping *"centro di raccolta"* and ignoring the nested *"raccolta"*).
- **Flattening:** This process effectively flattens the nested hierarchy into a single sequence of non-overlapping BIO tags, which is essential for training the CRF layer.

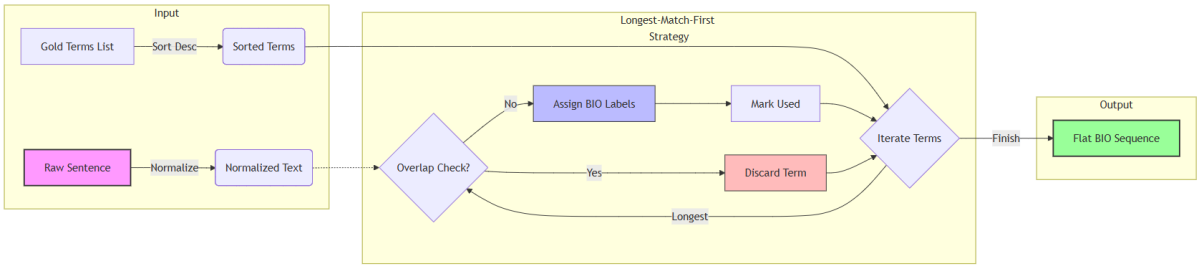

Figure 1: Illustration of the Longest-Match-First Labeling Strategy. The system prioritizes the longer span ("centro di raccolta") over the nested term ("raccolta"), resulting in a flat BIO tag sequence.

2.1.2. Model Architecture

The core architecture is a deep neural network composed of three main blocks:

1. **Backbone Encoder (XLM-RoBERTa):** We utilize `xlm-roberta-large` as the pre-trained encoder [2]. Its multilingual pre-training allows it to capture complex morphological and syntactic features of the Italian language effectively.

2. **Projection Layer (Highway-style):** Instead of feeding the raw encoder embeddings directly into the classifier, we introduce a projection block. This block consists of a Linear layer ($H \rightarrow H$), followed by a Tanh activation and Layer Normalization. This acts as a feature refinement stage, stabilizing the embeddings and reducing the variance of the pre-trained representations before classification.
3. **Decoding Layer (CRF):** To ensure the structural validity of the predicted tags, we employ a Conditional Random Field (CRF) layer [3]. Unlike a simple Softmax layer which predicts labels independently, the CRF models the transition probabilities between tags.
Implementation Detail: We explicitly constrain the transition matrix before training. Transitions that violate the BIO scheme (e.g., 'O' $\rightarrow$ 'I-TERM') are manually set to a large negative value ($-1e^4$) and frozen, forcing the model to learn only valid structural sequences.

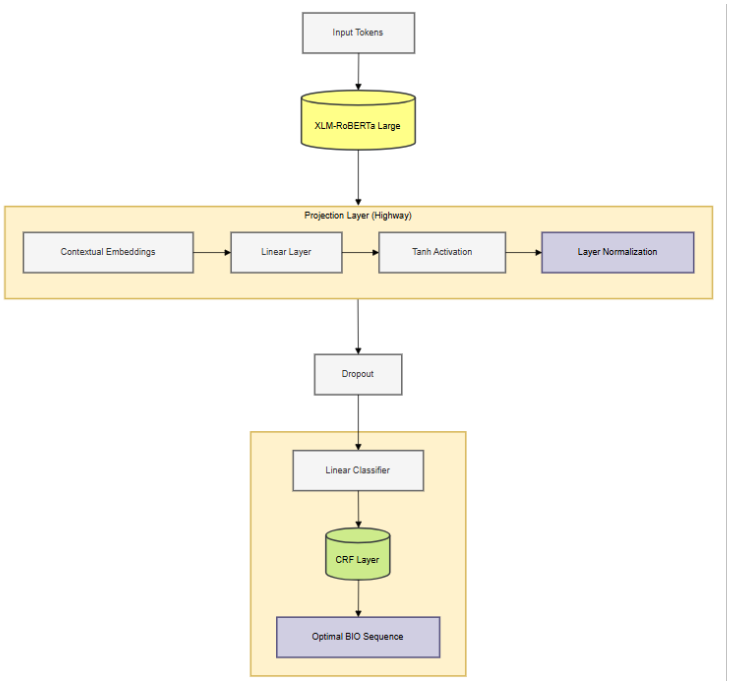

Figure 2: Model Architecture Diagram. The raw embeddings from XLM-R Large are refined via a Projection block (Linear+Tanh+LayerNorm) before passing to the CRF layer for final sequence decoding.

2.1.3. Training and Inference

The model was trained using the *Hugging Face Transformers* library with a batch size of 16 and a learning rate of $1e^{-5}$ for 6 epochs. During inference, we apply Viterbi Decoding via the CRF layer to obtain the globally optimal tag sequence. A final post-processing step extracts terms from the predicted spans, enforces lowercasing, and performs a final check to remove any duplicate or nested terms (keeping only the longest supersets) to strictly adhere to the Subtask A output format.

2.2. Subtask B: Term Variants Clustering

For the clustering of term variants, we developed a Hybrid Approach that combines deterministic rule-based filtering with the semantic reasoning capabilities of Generative AI. This ensures high precision for obvious morphological variants while leveraging LLMs for semantic synonyms.

2.2.1. Phase 1: Rule-based Pre-clustering

The first phase aims to reduce the search space and computational cost by grouping obvious morphological variants.

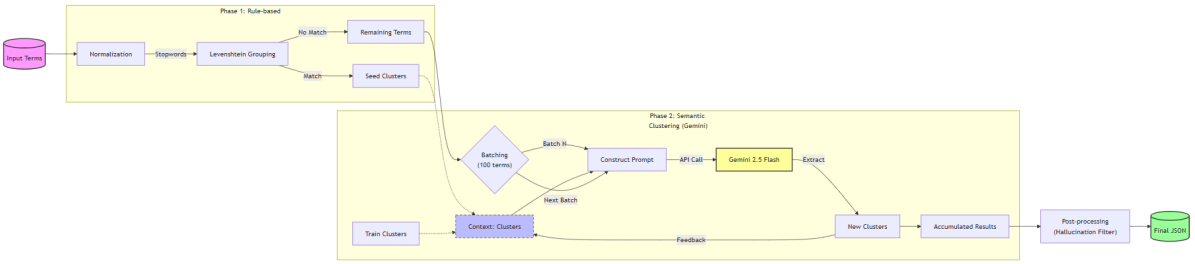

Figure 3: Overview of the Subtask B Pipeline: Input terms first pass through Rule-based Pre-clustering (Stopwords/Stemming) to form Seed Clusters, followed by an incremental Gemini LLM Loop for semantic grouping.

- **Normalization & Light Stemming:** Terms are cleaned by removing accents and punctuation. We apply a custom "light stemming" for Italian, normalizing vowel endings (e.g., substituting ending *-e* with *-a*, *-i* with *-o*) to handle simple gender/number variations.
- **Heuristic Grouping:** We filter out common Italian stopwords (e.g., *di, del, della*). Terms that reduce to the same normalized form and have a length difference ≤ 5 characters are grouped into "Seed Clusters". This effectively captures variants like *"isola ecologica"* vs *"isole ecologiche"* without LLM inference.

2.2.2. Phase 2: Semantic Clustering with Generative AI

Terms that remain unclustered or are only partially clustered are processed by Gemini 2.5 Flash [4]. This model was selected for its cost-efficiency and large context window, which is crucial for maintaining global consistency.

Prompt Engineering We designed a specific "System Prompt" that frames the LLM as a domain expert in Municipal Solid Waste Management. The prompt explicitly defines clustering constraints:

- **Positive Constraints:** Group acronyms with expansions (e.g., *"CCR" = "Centro Comunale di Raccolta"*) and true synonyms (e.g., *"Ecocentro" = "Isola ecologica"*).
- **Negative Constraints:** Strictly forbid grouping hypernyms with hyponyms (e.g., *"rifiuto"* $\neq$ *"rifiuto organico"*).

Incremental Clustering with Context Injection To handle the large volume of terms, we employ an incremental batch processing strategy (Batch size = 100):

1. The system initializes the context with high-confidence clusters derived from the Training set and Phase 1 rules.
2. For each new batch of unclustered terms, the prompt receives the **current state of all existing clusters**.
3. The model is instructed to: *"Insert the new terms into existing clusters when they are true synonyms, otherwise create new clusters."*

This "memory-augmented" loop ensures that a term in Batch N can be correctly clustered with a synonymous term processed in Batch 1, preventing the fragmentation of concepts across batches.

2.2.3. Post-processing

The LLM output is parsed from text to structured JSON. A final validation step filters out any "hallucinated" terms (terms generated by the LLM that were not in the original input list) and assigns any remaining isolated terms to singleton clusters to maximize recall.

3. Data and Experimental Setup

3.1. Dataset

The dataset used in this study is provided by the ATE-IT 2025 Shared Task organizers, specifically focusing on the domain of Italian Municipal Solid Waste Management. The corpus comprises administrative documents, regulations, and official notices, which are characterized by complex bureaucratic language and domain-specific terminology. The data is partitioned into three splits: Training, Development (Dev), and Test.

3.1.1. Data Preprocessing

To handle the linguistic variability inherent in the source text, we applied the following preprocessing pipeline:

- **Normalization:** We normalized all texts using the Unicode NFKC form to standardize characters.
- **Noise Removal:** Non-informative punctuation and noise characters (e.g., brackets, quotes) were removed, while alphanumeric tokens were preserved.
- **Lowercasing:** The entire corpus was converted to lowercase to mitigate vocabulary sparsity issues.

For **Subtask A**, the gold standard contained nested terms. To adapt this for sequence labeling models, we employed a "Longest-Match-First" strategy. This approach prioritizes longer terms over nested sub-terms to generate a flat BIO (Beginning-Inside-Outside) labeling scheme.

Table 1 summarizes the dataset statistics:

Split	Sentences	Annotated Terms
Training	2,308	2,218
Development	577	451
Test	1,142	-

Table 1
Statistics of the ATE-IT 2025 Dataset.

3.2. External Resources

Our system leverages the following external resources and pre-trained models:

- **XLM-RoBERTa Large:** We utilized the `xlm-roberta-large` model from Hugging Face [2] as the backbone for the term extraction task. This multilingual masked language model was pre-trained on 2.5TB of filtered CommonCrawl data.
- **Gemini 2.5 Flash[4]:** For the semantic clustering task, we utilized Google's Gemini 2.5 Flash model accessed via the Generative AI API.
- **Libraries:** The implementation relies on the `Transformers` library for model fine-tuning and `PyTorch-CRF` for the Conditional Random Field layer.

3.3. Experimental Settings

3.3.1. Subtask A: Automatic Term Extraction

We modeled the term extraction task as a token classification problem. The architecture consists of an XLM-RoBERTa Large encoder followed by a Linear layer and a Conditional Random Field (CRF) layer to ensure the structural validity of the predicted BIO sequences.

The model was fine-tuned using the following hyperparameters:

- **Max Sequence Length:** 256 tokens.

- **Batch Size:** 16.
- **Learning Rate:** 1×10^{-5}.
- **Epochs:** 6.
- **Dropout:** 0.2.
- **Seed:** 36.

3.3.2. Subtask B: Term Variants Clustering

For the clustering task, we developed a hybrid pipeline combining rule-based heuristics with Generative AI:

- **Rule-based Pre-clustering:** A module that groups terms based on Levenshtein distance (threshold ≤ 5) and morphological suffix normalization to handle simple inflectional variants.
- **Generative AI Clustering:** The remaining unclustered terms were processed by Gemini 2.5 Flash. We used an incremental prompting strategy with a batch size of 100 terms to maintain context and consistency.

3.4. Evaluation Metrics

System performance is evaluated using the official metrics defined by the ATE-IT 2025 Shared Task organizers.

3.4.1. Subtask A Metrics

Performance for term extraction is measured using two complementary F1-scores: **Micro F1** and **Type F1** [5].

Micro F1 Score This metric evaluates the precision and recall across all sentences in the dataset, accounting for term frequency. Let TP_s, FP_s, and FN_s be the number of True Positives, False Positives, and False Negatives in sentence s, respectively. The micro-averaged Precision and Recall are defined as:

$$Precision_{micro} = \frac{\sum_{s \in D} TP_s}{\sum_{s \in D}(TP_s + FP_s)}, \quad Recall_{micro} = \frac{\sum_{s \in D} TP_s}{\sum_{s \in D}(TP_s + FN_s)}$$

The Micro F1 is the harmonic mean of these values:

$$F1_{micro} = \frac{2 \cdot Precision_{micro} \cdot Recall_{micro}}{Precision_{micro} + Recall_{micro}}$$

Type F1 Score This metric evaluates performance over the set of unique term types, disregarding their frequency in the corpus. Let TP, FP, and FN be the counts calculated on the set of unique extracted terms versus unique gold standard terms.

$$Precision = \frac{TP}{TP + FP}, \quad Recall = \frac{TP}{TP + FN}, \quad F1 = \frac{2 \cdot Precision \cdot Recall}{Precision + Recall}$$

3.4.2. Subtask B Metrics

Clustering quality is evaluated using the **BCubed F1 score** [6, 7]. BCubed calculates precision and recall at the item level and averages them across all items.

Let N_{pred} and N_{gold} be the number of elements in the predicted and gold clustering, respectively. Let $C(x)$ denote the predicted cluster containing element x, and $L(x)$ denote the gold cluster containing element x. For each element x, the item-level precision $P(x)$ and recall $R(x)$ are computed as:

$$P(x) = \frac{|\{y \in C(x) : L(y) = L(x)\}|}{|C(x)|}, \quad R(x) = \frac{|\{y \in L(x) : C(y) = C(x)\}|}{|L(x)|}$$

The global Precision and Recall are averages over all items in the predicted and gold sets, respectively:

$$Precision = \frac{1}{N_{pred}} \sum_{x=1}^{N_{pred}} P(x), \quad Recall = \frac{1}{N_{gold}} \sum_{x=1}^{N_{gold}} R(x)$$

Finally, the BCubed F1 score is calculated as:

$$F1_{BCubed} = \frac{2 \cdot Precision \cdot Recall}{Precision + Recall}$$

4. Results

In this section, we present the experimental results obtained by the **TrietNLP** team for the ATE-IT 2025 Shared Task. We report performance metrics on both the internal Development set and the official Test set as provided by the task organizers.

4.1. Subtask A: Automatic Term Extraction

For the Automatic Term Extraction task, performance is evaluated using Micro-averaged and Type-based Precision, Recall, and F1-scores.

4.1.1. Development Set Performance

During the development phase, we evaluated our proposed architecture (XLM-RoBERTa coupled with a Conditional Random Field layer) on the held-out development set. As shown in Table 2, the system achieved a Micro-F1 score of 0.718, demonstrating strong capability in identifying domain-specific terminology boundaries.

Table 2
Subtask A Results on the **Development Set** (TrietNLP).

Metric	Precision	Recall	F1-Score
Micro-Averaged	0.738	0.689	0.718
Type-Based	0.662	0.632	0.647

4.1.2. Official Test Set Results

Table 3 presents the official results on the blind Test set. **TrietNLP** ranked second overall (Top 2) in this subtask.

Our system significantly outperformed the official Baseline, achieving a Micro-F1 of **0.599** compared to the baseline's 0.526. The inclusion of the projection layer and the CRF module proved essential for maintaining structural consistency in term prediction, allowing our model to remain competitive with the top-ranking system (Micro-F1 0.614).

4.2. Subtask B: Term Variants Clustering

For the Term Variants Clustering task, the evaluation utilizes Bcubed metrics, which are standard for clustering tasks where the number of clusters is not known a priori.

4.2.1. Development Set Performance

Our hybrid approach, combining rule-based pre-clustering with the incremental generative capabilities of Gemini 2.5, yielded exceptionally high results on the development set. Table 4 highlights a Bcubed F1-score of 0.985.

Table 3
Subtask A Official Results on the **Test Set**. Comparison between the Top 1 system, TrietNLP (Ours), and the Baseline.

Metric	System	Precision	Recall	F1-Score
Micro-Averaged	Top 1	**0.656**	**0.577**	**0.614**
	TrietNLP (Ours)	0.634	0.568	0.599
	Baseline	0.497	0.559	0.526
Type-Based	Top 1	**0.645**	0.529	**0.581**
	TrietNLP (Ours)	0.599	**0.545**	0.571
	Baseline	0.435	0.508	0.469

Table 4
Subtask B Results on the **Development Set** (TrietNLP).

Metric	Precision	Recall	F1-Score
Bcubed	0.975	0.944	0.985

4.2.2. Official Test Set Results

On the official Test set, **TrietNLP** achieved the highest performance, ranking as the **Top 1** system.

As detailed in Table 5, our system achieved a Bcubed F1-score of **0.441**, nearly doubling the performance of the Baseline (0.245) and surpassing the second-best system (0.359). This result validates the effectiveness of injecting context into the Large Language Model prompts to handle complex semantic relationships such as synonyms and acronyms in the waste management domain.

Table 5
Subtask B Official Results on the **Test Set**. Comparison between TrietNLP (Ours), the second-best system, and the Baseline.

System	Bcubed Precision	Bcubed Recall	Bcubed F1
TrietNLP (Ours)	**0.528**	0.378	**0.441**
Top 2	0.390	0.333	0.359
Baseline	0.177	**0.396**	0.245

4.3. Discussion

The discrepancy between Development and Test set results, particularly in Subtask B, suggests a significant shift in data distribution or term complexity in the official test data. However, the robust ranking of **TrietNLP** across both subtasks confirms that our methodology—specifically the combination of Transformer-based sequence labeling and LLM-augmented clustering—effectively generalizes better than the baseline approaches.

5. Discussion and Error Analysis

The performance of the TrietNLP system demonstrates the efficacy of separating the extraction and clustering tasks into distinct pipelines. However, an in-depth analysis of the prediction errors reveals specific linguistic and semantic challenges inherent to the domain of waste management in Italian.

5.1. Subtask A: Automatic Term Extraction

Despite achieving the second-best performance in the shared task, our error analysis based on the development set predictions highlights two primary categories of misclassification.

Boundary Detection and Determiner Inclusion. A recurrent error pattern involves the incorrect inclusion of Italian definite articles and prepositions within the term boundaries. As seen in the error report, the model frequently predicts terms such as *l'isola ecologica* or *dell'operatore ecologico*, whereas the Gold Standard expects the normalized forms *isola ecologica* and *operatore ecologico*. Although our preprocessing pipeline included Unicode normalization, the Conditional Random Field layer occasionally over-relied on local syntactic dependencies, failing to exclude the article preceding the noun phrase.

Long-tail and Nested Terminology. The model exhibits reduced recall for exceptionally long, multi-word expressions that resemble full clauses. For instance, the system failed to fully extract complex bureaucratic terms such as *servizio integrato gestione rifiuti – raccolta differenziata e servizi complementari*, often truncating them to shorter spans like *servizio integrato gestione rifiuti*. This suggests that while the XLM-RoBERTa encoder captures semantic meaning well, the fixed maximum sequence length and the rarity of such long n-grams in the training data hinder the model's ability to generalize to extreme cases.

5.2. Subtask B: Term Variants Clustering

Our hybrid approach, combining rule-based heuristics with the Gemini Large Language Model, achieved the top rank in the test phase. Nevertheless, specific challenges remain regarding semantic precision.

Hypernym vs. Hyponym Distinction. The primary challenge in unsupervised clustering is distinguishing between general categories and specific instances. Despite explicit instructions in the system prompt to separate distinct waste streams, the model occasionally struggled with the nuance between a general term (e.g., *rifiuti*) and its specific hyponyms (e.g., *rifiuti ingombranti*). This is a known limitation of current Large Language Models, which tend to associate terms based on semantic relatedness rather than strict ontological equivalence.

Incremental Learning and Semantic Drift. Our methodology employed an incremental batching strategy where previous clusters were fed back into the model context. While this improved consistency across batches, it introduced a risk of error propagation. If the model incorrectly grouped two distinct concepts in an early batch, subsequent batches would reinforce this error, leading to clusters that drifted from their original semantic definition.

6. Conclusion

In this report, we presented the TrietNLP system for the ATE-IT 2025 Shared Task. We approached the problem by decomposing it into two specialized sub-systems: a supervised neural sequence labeling architecture for term extraction and a hybrid neuro-symbolic approach for term clustering.

Our results validate this strategy. For Automatic Term Extraction, the combination of a multilingual encoder (XLM-RoBERTa) with a structured prediction layer (CRF) proved robust, capturing the majority of domain-specific terminology. For Term Variants Clustering, leveraging the reasoning capabilities of Large Language Models allowed us to handle complex synonyms and acronyms that simple string-matching algorithms miss, resulting in state-of-the-art performance on the test set.

Future Work. To address the limitations identified in our analysis, future developments will focus on two key areas:

- **Adversarial Training for Boundary Precision:** To mitigate the inclusion of articles and prepositions in Subtask A, we propose implementing adversarial training techniques. By deliberately exposing the model to perturbed inputs where articles are masked or modified, we aim to force the encoder to focus solely on the semantic core of the term.

- **Fine-tuning Small Language Models:** For Subtask B, replacing the generic API-based Large Language Model with a smaller, domain-finetuned model (e.g., Llama 3 or Mistral) could reduce latency and costs. Fine-tuning on specific ontology datasets would also likely reduce hallucination rates and improve the distinction between hypernyms and hyponyms.

Acknowledgments

We would like to thank the University of Information Technology, Ho Chi Minh City, Vietnam (UIT VNUHCM) for the support and resources provided during this research.

Declaration on Generative AI

During the preparation of this work, the authors used Google Gemini in order to: check for grammar and spelling errors. After using this tool, the authors reviewed and edited the content as needed and takes full responsibility for the publication's content.

References

[1] N. Cirillo, G. M. Di Nunzio, F. Vezzani, ATE-IT at EVALITA 2026: Overview of the Automatic Term Extraction Italian Testbed Task, in: Proceedings of the Ninth Evaluation Campaign of Natural Language Processing and Speech Tools for Italian. Final Workshop (EVALITA 2026), CEUR.org, Bari, Italy, 2026.

[2] A. Conneau, K. Khandelwal, N. Goyal, V. Chaudhary, G. Wenzek, F. Guzmán, E. Grave, M. Ott, L. Zettlemoyer, V. Stoyanov, Unsupervised cross-lingual representation learning at scale, in: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, 2020, pp. 8440–8451. URL: https://aclanthology.org/2020.acl-main.747. doi:10.18653/v1/2020.acl-main.747.

[3] J. Lafferty, A. McCallum, F. C. Pereira, Conditional random fields: Probabilistic models for segmenting and labeling sequence data, in: Proceedings of the eighteenth international conference on machine learning (ICML), 2001, pp. 282–289.

[4] G. Team, R. Anil, S. Borgeaud, Y. Wu, J.-B. Alayrac, J. Yu, R. Soricut, J. Schalkwyk, A. M. Dai, A. Hauth, et al., Gemini: a family of highly capable multimodal models, arXiv preprint arXiv:2312.11805 (2023).

[5] R. Verborgh, M. Röder, R. Usbeck, A.-C. Ngonga Ngomo, GERBIL – Benchmarking Named Entity Recognition and Linking Consistently, Semantic Web 9 (2018) 605–625. URL: https://doi.org/10.3233/SW-170286. doi:10.3233/SW-170286.

[6] A. Bagga, B. Baldwin, Entity-based cross-document coreferencing using the vector space model, in: Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and the 17th International Conference on Computational Linguistics (COLING-ACL'98), Montreal, 1998, pp. 79–85.

[7] E. Amigó, J. Gonzalo, J. Artiles, F. Verdejo, A Comparison of Extrinsic Clustering Evaluation Metrics Based on Formal Constraints, Information Retrieval 12 (2009) 461–486.