Ingénierie des Systèmes d'Information
Vol. 30, No. 11, November, 2025, pp. 2961-2976
Journal homepage: http://iieta.org/journals/isi

SHADO: A Semantics-Preserving Hybrid Framework for Automatic Text Classification Using Domain Ontology

Isaac Touza[1,2*], Warda Lazarre[1,2], Gazissou Balama[1,2], Kaladzavi Guidedi[2,3], Kolyang[2,4]

[1] Department of Mathematics - Computer Science, Faculty of Sciences, University of Maroua, Maroua P.O. Box 814, Cameroon
[2] Laboratoire de Recherche en Informatique, University of Maroua, Maroua P.O. Box 46, Cameroon
[3] Department of Computer Science and Telecommunications, National Advanced School of Engineering, University of Maroua, Maroua P.O. Box 46, Cameroon
[4] Department of Computer Science, Higher Teacher's Training College, University of Maroua, Maroua P.O. Box 55, Cameroon

Corresponding Author Email: isaac.touza@univ-maroua.cm

https://doi.org/10.18280/isi.301114

Received: 2 September 2025
Revised: 8 November 2025
Accepted: 16 November 2025
Available online: 30 November 2025

Keywords:
SHADO, BERT, dimensionality reduction, domain ontologies, hybrid ensemble, machine learning, semantic enrichment, text classification

ABSTRACT

In this article, we introduce SHADO: A Semantics-Preserving Hybrid Framework for Automatic Text Classification Using Domain Ontology, an innovative architecture that systematically integrates domain ontologies throughout a multi-phase processing pipeline to achieve superior classification accuracy. Our approach employs a comprehensive six-phase methodology: rigorous text preprocessing, strategic ontology selection and mapping, semantic enrichment with relationship preservation, intelligent feature extraction, dimensionality reduction with semantic constraints, and hybrid ensemble classification combining traditional machine learning algorithms with transformer-based models like BERT. Our experiments, conducted on a multi-source corpus of 5,536 documents covering five domains (politics, sports, technology, medical, and education) compiled from the 10 Newsgroups, BBC News, and Kaggle/UCI repositories, demonstrate excellent performance. SHADO achieves up to 97.11% accuracy, surpassing purely lexical models by 4.7%. These results confirm that SHADO consistently enhances both semantic coherence and classification reliability. Overall, SHADO represents a robust and scalable solution bridging the gap between statistical pattern recognition and semantic understanding, delivering high accuracy and interpretable classifications through explicit ontological knowledge integration.

1. INTRODUCTION

In today's digital era, organizations across industries face unprecedented challenges in managing and extracting insights from vast repositories of unstructured textual content. From customer feedback analysis to regulatory compliance monitoring, the ability to automatically categorize textual documents has become a strategic imperative for competitive advantage and operational efficiency.

Traditional text classification methodologies have predominantly relied on feature extraction techniques such as Bag of Words (BoW) [1], n-grams [2], and TF-IDF [3], combined with machine learning algorithms including support vector machines [4, 5], Naive Bayes [4, 6], decision trees [4], and k-nearest neighbors (KNN) [7, 8]. While these approaches have demonstrated reasonable performance in controlled environments, they exhibit significant shortcomings when confronted with domain-specific terminology, contextual ambiguity, and the nuanced semantics that characterize real-world textual data [9].

The emergence of pre-trained language models and deep learning architectures has partially addressed these limitations, yet a critical gap remains: the lack of explicit domain knowledge integration that could bridge the semantic divide between surface-level textual features and deeper conceptual understanding.

This research addresses this challenge by introducing SHADO (A Semantics-Preserving Hybrid Framework for Automatic Text Classification Using Domain Ontology), a novel hybrid framework that systematically incorporates structured domain knowledge through ontological representations. Unlike existing approaches that treat ontologies as supplementary resources, our method positions domain ontologies as core semantic engines that guide both feature enhancement and classification decision-making processes.

Our contribution is threefold: (1) development of a semantic enrichment pipeline that leverages domain ontologies for contextual feature augmentation, (2) implementation of a dimensionality reduction strategy that preserves semantic relationships while optimizing computational efficiency, and (3) design of a hybrid classification architecture that synergistically combines traditional machine learning models with transformer-based language models like BERT for

enhanced accuracy and interpretability.

The structure of this paper follows a systematic progression: we begin with a comprehensive literature analysis positioning our work within the current research landscape, followed by detailed methodology exposition, experimental validation using benchmark datasets, and conclude with performance analysis and future research directions.

2. STATE OF THE ART

2.1 Definition of the research problem

The problem of ontology-enhanced text classification can be formally formulated as an optimization problem: we aim to find the classifier $\boldsymbol{f}$ that minimizes the classification error across a labeled document set while leveraging both textual features and structured domain knowledge.

Let $D = \{d_1, d_2, \dots, d_n\}$ be a collection of $\boldsymbol{n}$ documents, and let $\boldsymbol{O}$ denote the associated domain ontology comprising concepts, relations, and instances. Each document d_i has a true class label $y_i \in C = \{c_1, c_2, \dots, c_k\}$, and is represented as a vector $x_i \in \mathbb{R}^m$ in the document-term space derived from vocabulary $W = \{w_1, w_2, \dots, w_m\}$.

The ontology-enriched representation of d_i is obtained using a semantic enrichment function $\boldsymbol{\varphi}$ defined by Eq. (1):

$$\hat{x}_i = \varphi\,(x_i, O) \tag{1}$$

where, $\hat{x}_i \in \mathbb{R}$ incorporates domain knowledge to capture semantic relationships between lexically diverse but conceptually related terms.

The optimization objective is therefore defined by Eq (2):

$$min_f \sum_{i=1}^{n} L(y_i, f(\hat{x}_i)) \tag{2}$$

where, f represents the classifier function, L is a loss function measuring prediction error.

Ontology-based enrichment helps bridge the semantic gap, where conceptually similar expressions may be lexically dissimilar. It also mitigates issues of high dimensionality and sparsity by adding domain-relevant semantic structure, improving classification accuracy and model robustness. The ultimate aim is to find the optimal f that effectively integrates both textual and ontological knowledge to improve predictive performance.

2.2 Literature review

Text classification has evolved from simple statistical methods to sophisticated deep learning approaches, with researchers increasingly turning to ontologies—structured knowledge representations—to bridge the gap between raw text and meaningful understanding. As Touza et al. [10] confirmed, ontology integration represents a promising pathway toward more intelligent, interpretable text analysis systems. Ontologies act as carefully organized dictionaries that define word meanings and explain how concepts relate within specific domains, promising to give machines deeper understanding rather than just word counting or pattern recognition.

However, this journey has been filled with both discoveries and challenges. Early pioneers like Tufiş and Koeva [11] explored linguistic ontologies for resolving word ambiguities, revealing both potential and computational demands. Wei et al. [12] and Yang et al. [13] demonstrated how domain-specific ontologies enhance semantic understanding, but highlighted that results depend heavily on ontology quality—creating comprehensive, accurate ontologies requires domain experts and ongoing maintenance.

The challenge of aligning text with ontological concepts proved complex. Lee et al. [14] developed ontology-based categorization techniques for lexical ambiguities, while Netzer et al. [15] found that maintaining optimal performance required frequent manual adjustments. As the field matured, researchers began exploring semantic techniques beyond simple word matching. WordNet became popular, with Nasir et al. [16] and Bouchiha et al. [17] using it to measure semantic similarity, clearly outperforming bag-of-words approaches. However, this reliance created limitations—what worked for general English didn't transfer to specialized domains or other languages.

Computational challenges became apparent as researchers pushed boundaries further. Altinel et al. [18] developed sophisticated semantic kernels for support vector machines, achieving impressive results but at considerable computational cost. Xu et al. [19] and Ma et al. [20] explored how ontologies could improve short text classification, consistently finding that semantic techniques offered advantages while introducing complexities around data requirements and computational resources.

Supervised learning brought new trade-offs. Risch et al. [21] used ontologies as knowledge bases to enrich document features, improving accuracy but remaining vulnerable to ontology quality limitations. Tao et al. [22] explored large-scale ontologies like Library of Congress Subject Headings, achieving promising results but creating dependency on labeled training data.

The emergence of deep learning opened possibilities for combining both worlds. Nguyen et al. [23] and Yelmen et al. [24] began integrating BERT with ontological knowledge, creating hybrid systems leveraging both neural network pattern recognition and structured ontological knowledge. These approaches represent significant progress, though they require careful optimization to handle noise and preserve semantic information.

Recent advances demonstrate increasingly sophisticated integration strategies. Uddin et al. [25] proposed the Expressive Short text Classification framework integrating a semantically enriched short text Topic Model, capturing semantics of words, topics, and documents within joint learning without requiring external knowledge sources. CB et al. [26] combined enhanced Apriori algorithms with healthcare-specialized BERT models for COVID-19 dataset analysis, utilizing BERT embeddings to evaluate semantic richness of extracted association rules.

Several additional studies have examined the role of ontologies in semantic enrichment for text classification. Shanavas et al. [27] constructed enriched concept graphs using domain ontologies to improve biomedical document classification, demonstrating gains over traditional similarity measures. Stein et al. [28] examined hierarchical text classification with word embeddings, highlighting the role of distributed representations in capturing semantic relationships across classes and Hawalah [29] introduced a semantic ontology-based approach to enhance Arabic text categorization by leveraging ontological structure alongside

vector space models.

The adoption of transformer architectures has become ubiquitous. Ouyang et al. [30] investigated fine-grained entity typing enriched with ontological information, aiming to enhance type prediction accuracy in a zero-shot setting while Ye et al. [31] explored ontology-enhanced prompt-tuning for few-shot classification. Ngo et al. [32] presented compelling approaches combining graph-based and transformer models for chemical-disease relation extraction, demonstrating how architectural hybridization leverages both paradigms' strengths.

Table 1. Summary of recent text document classification approaches integration ontologies

Ref.	Author	Year	Algorithms	Vector Representation Methods	Ontologies
[17]	Bouchiha et al.	2023	SVM	BoW, TF-IDF	WordNet
[20]	Ma et al.	2015	Semantics-based methods, k-Means	Continuous word embeddings, Vector space model	Not specified
[21]	Risch et al.	2016	SVM, KNN, Naive Bayes (NB), Probabilistic Methods	Not specified	Domain-specific ontology, enriched with probabilities
[22]	Tao et al.	2021	SVM, KNN	BoW, TF-IDF	LCSH
[23]	Nguyen et al.	2023	OneR, C4.5, NB, AdaBoost.M1	Doc2vec, Word2Vec, Word embeddings	OntoModel
[24]	Yelmen et al.	2023	CNN, RNN, BERT, Random Forest, SVM, MLP	Bag of Words, TF-IDF, Word2Vec, Doc2Vec	WordNet
[25]	Uddin et al.	2025	StTM (Short text Topic Model), BERT/Transformers, LDA variants, BTM (Biterm Topic Model)	Word embeddings, Document vectors, Probabilistic topic distributions, BERT-based contextual embeddings	External knowledge bases, Topic semantics for word-topic semantic relations, Context modeling semantic document
[26]	CB et al.	2023	Apriori, BERT Healthcare models, OCA Mining algorithm, GraphDB/SPARQL	BERT embeddings, Cosine similarity, RDF format conversion via ontology	COVID-19 Knowledge Base, Healthcare domain ontology for COVID-19 knowledge structuring
[27]	Shanavas et al.	2020	CMK, CWK, SVM	Bag of Words, TF-IDF	Medical ontology
[28]	Stein et al.	2019	FastText	Word embeddings	Not specified
[29]	Hawalah	2019	DT, NB, SVM, SCM	TF-IDF, Word embeddings	Specific Arabic ontologies
[30]	Ouyang et al.	2024	Fine-grained entity typing, Ontology enrichment	Entity type vectors, Fine-grained representations	Entity type ontologies, Hierarchical taxonomies
[31]	Ye et al.	2022	Prompt-tuning, Few-shot learning, Ontology-enhanced prompting	Prompt embeddings, Few-shot representations, Enhanced vectors	Domain-specific ontologies, Task-oriented taxonomies
[32]	Ngo et al.	2025	Graph Neural Networks, Transformers, Chemical-disease relation extraction	Graph embeddings, Transformer representations, Document-level vectors	Chemical ontologies, Disease taxonomies, Biomedical knowledge
[33]	Cao et al.	2024	Ontology-enhanced LLMs, entity extraction, relation extraction, knowledge graph construction	LLM semantic embeddings, entity and relation representations	Rare disease ontologies, biomedical knowledge graphs
[34]	Feng et al.	2025	Ontology-enhanced RAG, in-context learning with LLMs, SPARQL-based retrieval, reasoning and summarization	LLM semantic embeddings, retrieval-augmented representations, mapping proximity scoring	Biomedical ontology knowledge graphs, ontology mapping files, SPARQL-queried KGs
[35]	Lee and Kim	2025	Large Language Models, Sentiment classification, Ontology-based analysis	LLM embeddings, Sentiment vectors, Attribute representations	Sentiment ontologies, Emotion taxonomies
[36]	Tan et al.	2024	Recommendation algorithms, Medical decision systems, Ontology reasoning	Medical embeddings, Recommendation vectors, Diagnostic representations	Medical ontologies, Disease taxonomies, Treatment ontologies
[37]	Li et al.	2025	Machine Learning algorithms, Natural Language Processing, Classification models	Text embeddings, Feature vectors, NLP representations	Medical ontologies, Patient complaint taxonomies
[38]	Narmatha and Maniraj	2024	C4.5, KNN	Concept Mapping, Hypernyms, FT-IDF, χ^2 Filter	MeSH Ontology
[39]	Idress et al.	2024	Multi-layer neural networks, Siblings pattern extraction, Arabic NLP	Arabic word embeddings, multi-layer representations, Pattern vectors	Arabic linguistic ontologies, Semantic patterns ontology
[40]	Ali et al.	2025	Semantic analysis algorithms, Sindhi language processing	Sindhi language embeddings, Semantic vectors, Linguistic representations	Sindhi language ontology, Linguistic taxonomies
[41]	Giri and Deepak	2024	BiGRU, CNN, SDNN	Word2Vec, TF-IDF	Fuzzy Ontology
[42]	Hüsünbeyi and Scheffler	2024	TF-IDF, Counter algorithm, Zero-shot NLI (XLM-RoBERTa, mDeBERTa, BGE-M3, MiniLM), Logistic Regression, SVM	Multilingual sentence embeddings (E5-Large, E5-Mistral-7B, GTE-Qwen2-7B, E5-Small-V2), Cosine similarity, Static embeddings	Wikidata knowledge graph, SPARQL queries for class hierarchies
[43]	Almuhaimeed et al.	2024	Deep learning models, Semantic infusion algorithms, Tweet classification	Semantic embeddings, Deep representations, Ontology-infused vectors	Disaster management ontology, Emergency response taxonomies
[44]	Kowsari et al.	2019	BERT + Ontology Embedding	Sentence Embeddings (BERT), Ontology Embeddings (KG-based)	Fact-check OWL Ontology
[45]	Mitchell	1999	SVM, Naive Bayes, BERT	Ontology Enrichment + BERT (Bio_SA approach)	EDAM, Environment Ontology, Wikipedia

Large Language Models have introduced new possibilities. Cao et al. [33] proposed AutoRD, an ontology-enhanced LLM-based system for rare disease entity extraction using ontology-guided LLMs, while Feng et al. [34] developed OntologyRAG combining retrieval-augmented generation with ontology-aware mechanisms for biomedical code mapping. An and An [35] proposed an ontology-based sentiment and attribute classification framework that enhances domain-specific contextual accuracy, and empirically demonstrate its effectiveness through comparisons with LLM-based sentiment analysis approaches.

The biomedical domain has emerged as a primary application area, driven by rich medical ontologies and critical healthcare information processing needs. Tan et al. [36] developed OntoMedRec for disease diagnosis and treatment. Liu et al. [37] developed an intelligent system that integrates machine learning and natural language processing (NLP) for the automated classification and analysis of patient complaints. Narmatha and Maniraj [38] achieved 30% improvement using MeSH ontology over traditional stem-based methods on OHSUMED datasets. The field shows increasing attention to multilingual applications, with Idrees and Al-Solami [39] developing multi-layer Arabic text classification models and Ali et al. [40] creating semantic analysis frameworks for Sindhi language.

Crisis management and social media analysis have gained prominence, with Giri and Deepak [41] proposing semantic ontology-infused deep learning for disaster tweet classification, and Hüsünbeyi and Scheffler [42] exploring ontology-enhanced claim detection for fact-checking by integrating ontology embeddings with BERT sentence embeddings. In academia, ontology-enriched sentiment analysis models [43] showed up to 26.4% improvement in F-score.

Despite significant advances, persistent challenges remain. Most approaches struggle with the fundamental tension between semantic richness and computational efficiency, as integrating large ontologies with deep learning models creates scalability issues limiting practical deployment. Effectiveness heavily depends on underlying ontology quality and comprehensiveness, which varies significantly across domains. The lack of standardized evaluation metrics makes comparing approaches difficult, hindering scientific progress. While domain-specific adaptations show promise, developing generalizable approaches remains challenging, as most methods are tightly coupled to specific ontological structures. Most importantly, few studies successfully combine the full spectrum of available techniques—classical machine learning, modern deep learning, and structured ontological knowledge—into unified frameworks leveraging each approach's strengths while mitigating individual weaknesses. Table 1 summarizes the most recent advances in ontology-enhanced text classification from 2015 to 2025, highlighting the evolution of algorithmic approaches, vector representation methods, and ontological frameworks employed across different research domains.

In reviewing the existing literature on text classification, we observed that while numerous innovative approaches have been introduced, several recurring limitations persist. A significant proportion of studies continue to rely on traditional or narrowly scoped techniques—such as purely statistical models or limited semantic strategies—without fully leveraging the advancements made in deep learning. For instance, models like SVMs and similarity-based classifiers, though effective in constrained contexts, often underperform compared to transformer-based architectures like BERT, which offer superior capacity for capturing nuanced contextual information.

Another critical gap lies in the often-overlooked role of comprehensive preprocessing and semantic enrichment. These components are essential for improving classification accuracy, particularly when dealing with unstructured or noisy textual data. Unfortunately, many existing works either neglect these stages or implement them superficially, thereby weakening the overall performance of their systems. A notable exception is the contribution of Touza et al. [10], who illustrated the power of integrating domain ontologies into deep learning workflows. Their study demonstrates that coupling BERT with ontology-driven semantic enrichment not only enhances document representation but also improves semantic disambiguation. Their empirical results, validated on biomedical datasets, show that such hybrid models consistently outperform both traditional approaches and standalone deep learning techniques in terms of accuracy and robustness.

Moreover, other common challenges include an overreliance on a single ontology, insufficient mechanisms to handle ambiguous or noisy data, and the use of dimensionality reduction techniques that may inadvertently discard valuable semantic information. While researchers like Nasir et al. [16] and Xu et al. [19] emphasized the relevance of ontologies, their approaches often lack a concrete strategy for effective integration, limiting the depth and relevance of semantic features extracted from the text.

Building on these insights and addressing the identified gaps in current research, we propose a comprehensive framework that systematically integrates the full spectrum of available techniques—classical machine learning, modern deep learning, and structured ontological knowledge—into a unified approach that leverages the strengths of each methodology while mitigating their individual weaknesses. Our method addresses the scalability challenges through efficient integration strategies and incorporates robust evaluation mechanisms to ensure reliable performance across diverse datasets and domains.

3. PROPOSED TEXT CLASSIFICATION APPROACH

Our classification model, called SHADO, introduces a comprehensive, multi-layered strategy for enhancing text classification through the integration of domain ontologies. Its primary objective is to improve the accuracy, robustness, and adaptability of classification systems by leveraging advanced NLP and machine learning techniques. The overall framework is structured into six key phases, as illustrated in Figure 1.

3.1 Data preparation and cleaning

Text preprocessing is the first critical step where textual data is cleaned and prepared for further analysis [44]. This process includes:

- Cleaning: Removing irrelevant elements such as extra spaces, special characters, punctuation, and converting text to lowercase.
- Tokenization: Splitting the text into words, phrases, symbols, or meaningful elements called token [27, 28].

- Lemmatization: Reducing words to their base or root form to ensure uniformity in text representation [22, 29].
- Stop Word Removal: Eliminating words that do not contribute significant se-mantic meaning to focus analysis on relevant content [23].

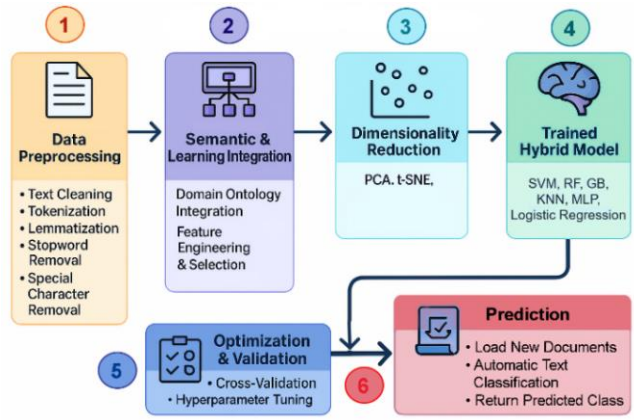

Figure 1. SHADO framework steps

Our enhanced preprocessing incorporates pattern preservation techniques that maintain important semantic markers such as temporal expressions, email addresses, and compound words. Unlike traditional approaches that indiscriminately remove numeric content, our method intelligently transforms numerical entities into semantic placeholders (<YEAR>, <DECIMAL>, <NUMBER>) while preserving their contextual significance. Additionally, we implement sentence boundary preservation through <SENT_END> markers, enabling downstream semantic analysis to maintain discourse-level context. Algorithm 1 details the steps involved in preprocessing text data:

Algorithm 1: Enhanced Text Preprocessing
Input: Raw text data
Output: preprocessed text
1. **For** each document in the raw text:
2. Pattern preservation (dates, emails, URLs)
3. Intelligent entity handling (years, decimals, numbers)
4. Sentence boundary marking for context preservation
5. Advanced cleaning with marker preservation
6. Optimized tokenization with length filtering (2-15 chars)
7. Smart stop word removal
8. Context-aware lemmatization
9. Enhanced token rejoining
10. **End for**
11. **Return** preprocessed text.
end

3.2 Semantic and learning integration

This critical phase transforms preprocessed textual data into semantically enriched representations through strategic ontology integration. The semantic enhancement process bridges the gap between raw textual features and domain-specific knowledge, enabling more contextually aware classification. This phase focuses on enhancing raw text data with domain-specific semantic information to improve the learning and classification process. It is composed of two key components: Domain Ontology Integration and Feature Engineering & Selection.

3.2.1 Domain ontology integration

To provide contextual and domain-aware understanding of the textual data, relevant domain ontologies are integrated into the pipeline. This integration involves three main sub-steps:

(1) Ontology selection

Ontology selection is a critical process, as the quality and relevance of the selected ontologies significantly influence the effectiveness of semantic enrichment. The following criteria guide the selection:

- Domain Relevance: Ontologies must align closely with the subject matter of the texts (e.g., medicine, education, technology, politics) and include relevant concepts and relationships.
- Semantic Richness: Chosen ontologies should be conceptually dense, offering a rich network of terms and relations.
- Accessibility and Maintenance: Preference is given to open-access, well-documented, and actively maintained ontologies, such as those from BioPortal, OBO Foundry, or Linked Open Vocabularies.
- Technical Compatibility: Ontologies must be available in standard formats (e.g., RDF, OWL) to ensure smooth system integration.

These criteria are aligned with best practices outlined in Touza et al. [10], who emphasized the importance of relevance, richness, and format compatibility in selecting ontologies for semantic enrichment.

The number of ontologies used depends on the nature and complexity of the do-main:

- For well-established domains: At least two ontologies are used, even if one is widely recognized. This redundancy helps capture additional semantic nuances and enhances the robustness of the approach.
- For complex or interdisciplinary domains: More than two ontologies may be integrated to cover diverse concepts and ensure comprehensive semantic representation.

(2) Ontology mapping

Ontology mapping links textual terms to structured concepts defined within selected ontologies, enabling a semantic representation of the content. Formally, this process can be modeled by Eq. (3).

$$C = f_{map}(T, O) \tag{3}$$

where, C is the set of corresponding concepts in the ontology, T the set of terms from the text, O the select ontology and f_{map} the function that maps each term $t \in T$ to one or more concepts $c \in O$.

The process by which textual terms are semantically linked to domain-specific concepts is illustrated in Figure 2, which visualizes the function f_{map} by systematically associating terms with ontology concepts based on their semantic relevance and structural alignment. The figure presents a structured workflow that explicitly outlines the conditional logic used for concept matching, including fallback mechanisms when direct mappings are not found.

(3) Semantic enrichment

Once the terms have been associated with relevant concepts in the ontology (as described in Algorithm 2), a further step is needed to deepen their meaning and contextual understanding. This is the role of semantic enrichment, which consists in

expanding each term's representation by incorporating additional knowledge from the ontology—such as hierarchical relations (e.g., parent-child structures) and semantic links (e.g., "part of", "related to").

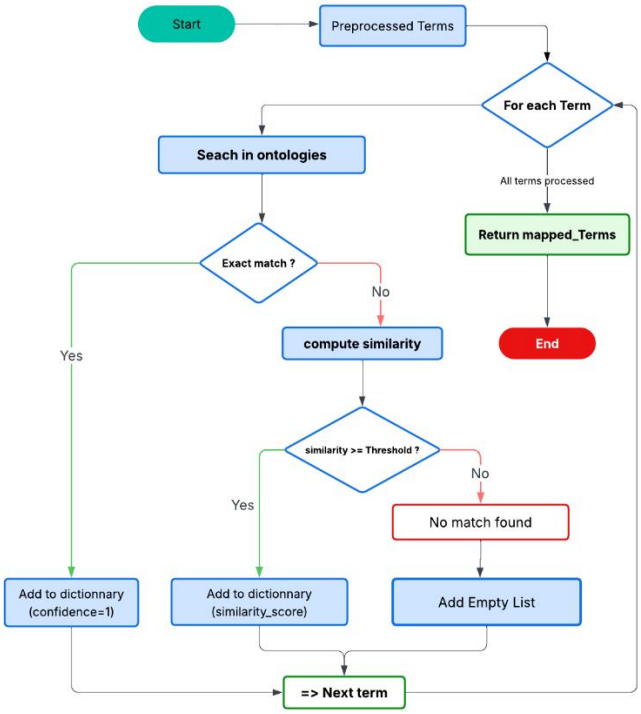

Figure 2. Flowchart of intelligent ontological mapping

Algorithm 2: Semantic Enrichment

Input:
- Mapped_Concepts: Terms linked to ontology concepts
- Ontology: Includes hierarchies, relations, and properties

Output: Enriched_Terms: Terms enriched with semantic data

1. Initialize Enriched_Terms as empty dictionary
2. **for** each term in Mapped_Concepts:
3. Initialize enriched_info as empty list
4. **for** each concept in Mapped_Concepts[term]:
5. Add descendants of concept to enriched_info
6. Add semantic relations of concept to enriched_info
7. Add properties of concept to enriched_info
8. **end for**
9. **if** enriched_info is not empty:
10. Enriched_Terms[term] ← enriched_info
11. **else**:
12. Enriched_Terms[term] ← empty list
13. **end if**
14. **end for**
15. **return** Enriched_Terms

end

This process allows the initial term set to be enhanced with richer, more structured information, enabling more accurate and meaningful interpretations of the text content. We formalize this enrichment process with the following expression (Eq. (4)):

$$T_{\text{enriched}}=f_{\text{enrich}}(T,C,R) \tag{4}$$

where,

- T is the original set of terms extracted from the text.
- C represents the concepts to which those terms have been mapped.
- R includes the relationships and hierarchies drawn from the ontology.
- f_{enrich} is the function that performs the enrichment by combining these elements.

The goal is to go beyond simple matching and provide a semantically empowered view of the text—leveraging ontological knowledge to unlock deeper insights and more effective downstream processing. To operationalize this enrichment process, Algorithm 2 outlines how to retrieve and integrate semantic information from the ontology for each mapped term.

3.2.2 Intelligent feature extraction and selection

Building upon the semantic enrichment achieved in the previous phase, this step systematically identifies and selects the most informative features by integrating statistical significance measures with ontological semantic relevance. This hybrid approach ensures that selected features capture both distributional patterns and domain-specific semantic relationships, creating an optimal foundation for subsequent classification models.

Our feature selection methodology combines traditional statistical measures with ontology-derived semantic indicators to create a comprehensive importance scoring system. This dual-criteria approach addresses the limitations of purely statistical methods while preserving computational efficiency.

• Statistical Foundation: The TF-IDF weighting scheme provides the statistical baseline for feature importance:

$$TF-IDF(t,d)=TF(t,d)\times log\frac{N}{DF(t)} \tag{5}$$

where, $TF(t,d)$ represents term frequency in document d, N is the total number of documents, and $DF(t)$ is the number of documents containing the term t.

• Semantic Enhancement: Ontological centrality measures augment statistical importance by quantifying semantic significance within domain knowledge structures:

$$Total\ Importance = TF-IDF\times Centrality_{onto}(c) \tag{6}$$

where, $\text{Centrality}_{onto}(c)$ represents the centrality score of concept c within the ontological graph structure.

The feature extraction process operates through coordinated stages that progressively refine the feature space while maintaining semantic coherence. Algorithm 3a implements this coordinated approach through three sequential phases: statistical filtering, ontological enhancement, and semantic scoring.

The concept centrality quantifies the importance of each ontological concept within its semantic network and is used to weight features in the document representation. This measure combines three complementary aspects of a concept $\boldsymbol{c}$ is defined by Eq. (7):

$$concept\_centrality(c)=\alpha\cdot degree(c)+\beta\cdot depth(c)+\gamma\cdot breadth(c) \tag{7}$$

where,

- α, β and γ, are tunable weights reflecting the relative importance of each component.
- Degree centrality ($degree(c)$): the number of direct relationships that $\boldsymbol{c}$ has with other concepts in the ontology.
- Hierarchical depth ($depth(c)$): the position of $\boldsymbol{c}$ within the ontology hierarchy, normalized by the maximum depth. Concepts closer to the root may be more general, while deeper concepts capture domain-specific semantics.
- Semantic breadth ($breadth(c)$): the total number of indirectly connected concepts reachable from $\boldsymbol{c}$, representing the semantic scope of the concept.

Algorithm 3a: Ontology-Enhanced Feature Extraction

Input: Preprocessed_Texts, Ontologies, TFIDF_Threshold, Semantic_Threshold
Output: Document_Features, Feature_Metadata

Compute TF-IDF matrix for all documents
for each document in Preprocessed_Texts:
select terms where TF-IDF ≥ TFIDF_Threshold
Apply ontological mapping using Algorithm 2
Apply semantic enrichment using Algorithm 3
for each enriched term:
Compute total_importance = TF-IDF × concept_centrality
if total_importance ≥ Semantic_Threshold:
Store feature and associated metadata
end for
Add selected features and metadata to Document_Features
end for
return Document_Features, Feature_Metadata
end

Algorithm 3b: Multi-Configuration Feature Extraction

Input: Preprocessed texts T, domain D (for ontology mapping)
Output: Combined feature matrix X_combined

Apply word-level TF-IDF with n-grams (1,2)
Apply character-level TF-IDF with n-grams (3,6)
Apply Truncated SVD to character features
Extract semantic features from ontology mapping
- Domain-specific concept counting
- Fallback to text complexity metrics
Combine all feature types:
$X_{combined} = [X_{words} \mid X_{chars\_reduced} \mid X_{semantic}]$
return X_combined, feature_metadata
end

These three components are combined into a normalized score in the range [0,1]:

To address the limitations of single-configuration feature extraction, we propose a multi-modal approach that combines three complementary feature types: (1) word-level TF-IDF with bi-gram extensions capturing semantic relationships, (2) character-level TF-IDF with 3-6 gram patterns capturing morphological information, and (3) ontology-derived semantic features quantifying domain-specific concept density. This hybrid approach increases feature space richness from traditional single-vector representations to multi-dimensional semantic embeddings of 3100+ features. Algorithm 3b describes this process.

3.3 Dimensionality reduction with semantic preservation

Following feature extraction and semantic enrichment, the resulting high-dimensional feature space requires careful dimensionality reduction to maintain computational efficiency while preserving the rich semantic relationships captured through ontological integration. This phase implements a sophisticated reduction strategy that balances performance optimization with semantic fidelity. Our approach employs a multi-stage reduction pipeline that progressively refines the feature space.

The reduction process operates through three coordinated stages, each optimized for specific aspects of the feature space transformation.

(1) Stage 1: Statistical Variance Reduction (PCA)

Principal Component Analysis provides the initial dimensionality reduction by identifying the directions of maximum variance in the feature space is given by Eq. (8). where X represents the original feature matrix, W_{PCA} contains the principal components, and Y is the reduced representation.

$$Y = XW_{PCA} \tag{8}$$

(2) Stage 2: Semantic Structure Preservation

This stage applies semantic constraints to ensure that ontologically related features maintain their relationships in the reduced space is given by Eq. (9).

$$min_w \mid\mid XW - Y \mid\mid_F^2 + \lambda \sum_{(i,j) \in S} w_s(i,j) \mid\mid w_i - w_j \mid\mid_2^2 \tag{9}$$

where, S represents semantic similarity pairs, $w_s(i,j)$ denotes semantic weights, and λ controls the semantic preservation strength.

(3) Stage 3: Non-linear Manifold Learning (t-SNE)

The final stage employs t-distributed Stochastic Neighbor Embedding to capture non-linear relationships while preserving local semantic neighborhoods is given by Eq. (10).

$$P_{j|i} = \frac{exp(-\mid\mid x_i - x_j \mid\mid^2 / 2\sigma_i^2)}{\sum_{k \neq i} xp(-\mid\mid x_i - x_k \mid\mid^2 / 2\sigma_i^2)} \tag{10}$$

where,

- $P_{j|i}$ is Conditional probability that point x_i chooses x_j as its neighbor.
- x_i, x_j are data points in the original high-dimensional space.
- $\mid\mid x_i - x_j \mid\mid^2$ Squared Euclidean distance between points *i* and *j*.
- σ_i^2 Variance of the Gaussian kernel centered on point *i*.

The reduction process is implemented through Algorithm 4, which coordinates the three stages while continuously monitoring semantic preservation quality. The algorithm begins with statistical variance reduction via PCA, followed by the application of semantic constraints derived from the ontological graph structure, and concludes with adaptive t-SNE transformation based on dataset characteristics.

Algorithm 4: Semantic-Preserving Dimensionality Reduction

Input: Enriched_Features, Semantic_Graph, Target_Dimensions

Output: Reduced_Features, Preservation_Score
Apply PCA
Apply semantic constraints using ontological relationships (Eq. (9))
Apply t-SNE for final reduction
Evaluate semantic preservation (SNP and ORR metrics)
return Final_Features, Preservation_Score
end

The Semantic_Graph G = (V, E) is constructed from all ontology-enriched concepts extracted during feature extraction. Nodes $v \in V$ represent concepts, while edges $e \in E$ represent semantic relationships such as *is-a*, *part-of*, or domain-specific connections. Edge weights $w_s(i,j)$ reflect semantic similarity between connected concepts, computed based on ontological proximity or co-occurrence in the corpus.

During dimensionality reduction, semantic constraints are applied to preserve the ontological relationships among features. In Stage 2 of Algorithm 4, these constraints are incorporated into the objective function for linear reduction (Eq. (9)), ensuring that concepts linked in the Semantic_Graph remain close in the reduced space. The regularization parameter λ controls the trade-off between preserving semantic relationships and minimizing reconstruction error.

Quality assessment employs Semantic Neighborhood Preservation (SNP) metrics that measure the retention of k-nearest semantic neighbors and Ontological Relationship Retention (ORR) scores that evaluate the preservation of ontological relationships in the reduced space.

The optimal target dimensionality is determined through cross-validation using classification performance as the primary criterion, balanced with semantic preservation scores and computational efficiency measures. This adaptive approach ensures that the reduced feature space maintains both computational tractability and the rich semantic structure essential for accurate ontology-enhanced classification, providing an optimal foundation for the subsequent hybrid classification phase.

3.4 Hybrid classification model

The final training phase leverages the semantically enriched and dimensionally optimized feature representations to construct a sophisticated ensemble classifier that combines the complementary strengths of traditional machine learning algorithms with modern transformer architectures. This hybrid approach integrates six distinct learning paradigms: Support Vector Machines excel at finding optimal decision boundaries in high-dimensional spaces, Random Forest provides robust feature importance estimates, Gradient Boosting provides a powerful trade-off between bias and variance, making it ideal for capturing complex patterns in structured data, KNN captures local neighborhood patterns enhanced by semantic similarity, Multi-Layer Perceptron learns complex non-linear mappings, and BERT contributes contextualized understanding of textual semantics.

The ensemble employs a two-tier architecture where base models generate individual predictions that are subsequently combined through an intelligent voting mechanism. Unlike simple majority voting, our approach implements weighted voting that assigns different importance to each model based on their reliability and past performance on similar semantic contexts. Algorithm 5 coordinates this multi-algorithm approach through systematic training, prediction aggregation, and consensus formation.

Algorithm 5: Hybrid Classification Model
Input: Reduced_Features, Labels, Base_Models
Output: Final_Classifications, Confidence_Scores
Initialize prediction_matrix for storing base model outputs
Apply BERT vectorization to reduced features for contextual enhancement
for each base_model in Base_Models:
Train model on reduced_features and labels
Generate predictions and confidence scores
Store predictions in prediction_matrix
end for
Apply weighted voting based on model reliability scores
Calculate final class assignments and aggregate confidence scores
return final_classifications, confidence_scores
end

To enhance the ensemble's robustness, each base model is assigned a model reliability score, which reflects its predictive performance on a validation set. In our implementation, the reliability score w_i for classifier i is proportional to its cross-validation accuracy and normalized over all base models, as given in Eq. (11):

$$w_i = \frac{CV_{score\_i}}{\sum CV_{scores}} \tag{11}$$

During ensemble prediction, each classifier contributes to the final decision based on its weight w_i and its predicted probability p_i for each class. The final prediction is determined by aggregating the weighted probabilities and selecting the class with the highest total score, as shown in Eq. (12):

$$Final\ Prediction = arg\ max\ (\sum w_i \times P_i) \tag{12}$$

This performance-based weighted voting ensures that models demonstrating higher reliability have greater influence on the ensemble decision, while models with lower accuracy contribute less. Consequently, the approach leverages the collective intelligence of multiple algorithms, improving overall classification accuracy and providing more stable and interpretable confidence scores.

3.5 Optimization and validation

This critical phase implements comprehensive evaluation protocols to ensure optimal performance and robust generalization of the trained ensemble model. The optimization process operates through three coordinated activities: cross-validation for generalization assessment, hyperparameter tuning for performance maximization, and result analysis for model selection and validation.

Cross-validation evaluates generalization capability through systematic data partitioning into training and test sets, preventing overfitting while enabling robust performance assessment. Hyperparameter tuning employs GridSearchCV to optimize each base model's configuration, systematically exploring parameter spaces for all ensemble components including traditional machine learning models and BERT fine-tuning parameters. Performance analysis evaluates models

across multiple dimensions using precision, recall, F1-score, and accuracy metrics, while confusion matrices reveal detailed classification patterns and semantic coherence assessment measures alignment with ontological relationships.

The validation process compares weighted voting, majority voting, and stacking approaches to identify optimal ensemble configurations. Final model selection balances classification accuracy with semantic consistency and computational efficiency, ensuring practical applicability while maintaining the ontology-enhanced performance advantages that distinguish our approach from traditional classification approaches.

3.6 Prediction phase

After thorough optimization and validation, the SHADO framework moves into operational deployment for real-world document classification. In this phase (Figure 3), a streamlined prediction pipeline is activated to process new, unseen documents by applying a predefined sequence of steps established during training.

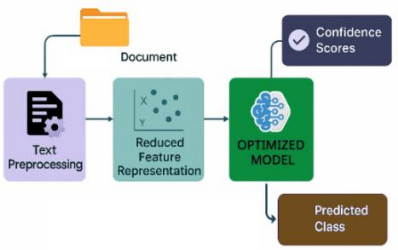

Figure 3. Prediction phase workflow for new document classification

Unlike the training phase, supervised ontology-based enrichment is not applied here, since the document's class is not yet known. The system instead performs class-independent operations, including text preprocessing, feature extraction (TF-IDF and general semantic weighting if applicable), dimensionality reduction using pre-trained transformation matrices, and final classification through the optimized ensemble model.

The system delivers comprehensive classification results including definitive class assignments representing the ensemble's highest confidence predictions and detailed confidence scores indicating prediction reliability. These confidence metrics prove valuable for applications requiring threshold-based decision making or scenarios where multiple potential classifications need consideration. Additional outputs include individual model predictions for transparency. This comprehensive output supports both automated decision-making and human interpretation, ensuring practical applicability across diverse domain-specific classification tasks while maintaining the semantic richness that distinguishes ontology-enhanced classification from traditional approaches.

4. IMPLEMENTATION AND EXPERIMENTATION

This section presents the practical implementation of the SHADO framework alongside comprehensive experiments conducted to evaluate its performance across multiple domains and datasets.

4.1 Implementation architecture

The SHADO framework is built on Python's robust ecosystem, offering a modular and scalable architecture that supports the entire classification pipeline. It combines powerful tools for semantic processing (NLTK, spaCy, Gensim), ontology management (OWLReady2, rdflib, NetworkX), machine and deep learning (scikit-learn, Transformers, TensorFlow/Keras), and dimensionality reduction and optimization (PCA, t-SNE, GridSearchCV). This setup ensures high performance, flexibility, and extensibility throughout the system.

4.2 Experimental protocol

4.2.1 Datasets and experimental design

Our evaluation uses a multi-source corpus to assess SHADO's performance across varied textual domains. The dataset combines three established sources: 10 Newsgroups (subset of the 20 Newsgroups collection by Ken Lang, Kaggle version by Jensen Baxter, 2018), BBC News Dataset (Bimal Timilsina, 2021), and the Website Classification Dataset (Hetul Mehta, Kaggle/UCI Repository).

The 10 Newsgroups [45] subset contains around 1,000 cleaned documents across ten categories, with duplicates removed and only key headers retained. The BBC dataset provides 2,000 validated medical articles from Medical News Today, while the Website Classification set adds textual and structural data for web categorization.

Together, these sources yield 5,536 curated documents covering five major domains — politics, sports, technology, medical, and education. All texts were standardized using Algorithm 1 (Preprocessing Pipeline), which performs normalization, lemmatization, stopword removal, and GloVe-based vectorization.

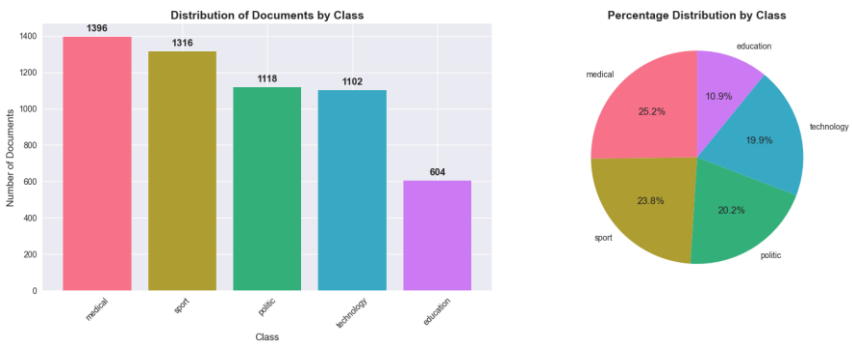

Figure 4. Dataset distribution by class

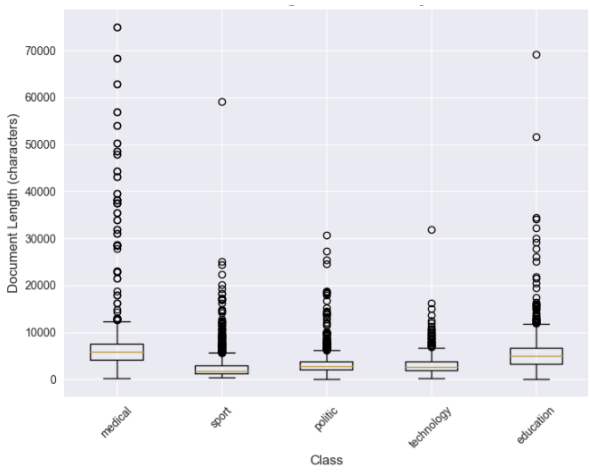

Figure 5. Document length distribution by class

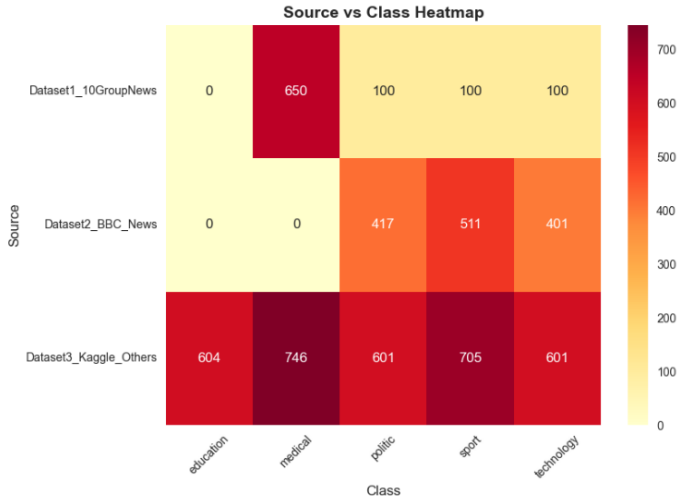

Figure 6. Heatmap of the correlation between data sources and classification categories

Table 2. Domain-specific ontologies used for semantic enrichment

Domain	Ontology	Description
Technology	Software Ontology	Software artifacts, development processes, and programming concepts
	Computer science ontology	Hierarchical classification of computer science research topics
	Computer network ontology	Network protocols, topologies, and distributed computing concepts.
	Artificial intelligence ontology	AI subfields: machine learning, NLP, computer vision, knowledge representation
Medical	Human Disease Ontology	Standardized classification of human diseases with cross-references to symptoms, causes, and anatomical locations
	Medical Action Ontology	Standardized classification of human diseases and symptoms
	The Ontology of Medically Related Social Entities	Medical procedures, treatments, and clinical interventions
	Ontology for Biomedical Investigations	Biomedical research terminology and experimental protocols
Politics	DBpedia Political Classifications	Political entities, government structures, and electoral systems
	European Legislation Identifier (ELI)	Legal and political terminologies used in European contexts
	EU Vocabularies	EU policies, institutions, and administrative procedures
Sports	Sport ontology	Sports terminology: disciplines, competitions, athletes, venues
	Olympic Games Ontology	Olympic sports, events, records, and competition structure
	Sports Performance Analytics Ontology	Athletic performance data, statistics, and sports science
Education	Social Determinants of Education Ontology	Socioeconomic factors affecting educational outcomes
	EduKG: An Educational Knowledge Graph	Educational concepts, curriculum standards, and learning objectives
	EducOnto	Learning activities, educational resources, and assessment methods

To ensure methodological rigor, datasets were split 80/20 for training and testing, maintaining balanced domain representation and semantic diversity. Figure 4 summarizes the distribution of documents across domains.

Figure 5 illustrates the detailed distribution of document lengths across all classes in the dataset, highlighting variations in text size that may affect classification performance.

Figure 6 presents a comprehensive heatmap showing the correlations between the different data sources and classification categories, providing insights into potential domain-specific relationships and patterns within the corpus.

4.2.2 Ontological resources and selection

The semantic enrichment component of SHADO relies on a carefully curated set of domain-specific ontologies obtained from authoritative repositories and standards organizations. A multi-ontology approach is adopted for each domain to ensure broad semantic coverage and to mitigate the limitations of relying on a single source. Ontologies are selected based on their domain relevance, semantic richness, standardization (e.g., OWL, TTL, RDF), and ongoing maintenance. Typically, 2 to 5 complementary ontologies are employed per domain to balance expressiveness and computational efficiency. To find these ontologies, we explored several well-known platforms known for their ontology collections, such as Archivo, BioPortal, the Ontology Library Service of the Open Biological and Biomedical Ontology (OBO) Foundry, github repository and other specialized repositories, ensuring extensive and accurate coverage for each category. The selected ontologies for each domain are summarized in Table 2, which outlines the sources and roles of each ontology used in the semantic enhancement process.

Table 3. Global ontological framework statistics

Metric	Value
Total Ontologies	20
Total Concepts	34,024
Total Relations	3,139
Total RDF Triples	908,935

Our comprehensive ontological framework encompasses a meticulously curated collection of domain-specific ontologies distributed across five critical knowledge domains. Table 3 presents the overall statistics of our ontological infrastructure, demonstrating the scale and scope of semantic resources employed in our experimental evaluation.

The distribution of ontological resources across domains is

detailed in Table 4, which illustrates the strategic allocation of semantic knowledge bases to ensure balanced coverage while accommodating domain-specific complexity requirements.

To assess the semantic richness and structural characteristics of our ontological framework, Table 5 provides derived metrics that quantify the density, granularity, and balance of the integrated knowledge bases.

Table 4. Domain distribution of ontological resources

Domain	Number of Ontologies	Percentage
Technology	5	25.0%
Education	4	20.0%
Medical	4	20.0%
Politics	4	20.0%
Sports	3	15.0%
Total	20	100.0%

Table 5. Framework density and balance metrics

Statistic	Value	Description
Average Concepts per Ontology	1,701	Mean concept density across all ontologies
Average Relations per Ontology	157	Mean semantic relationship density
RDF Triples per Concept	26.7	Knowledge representation granularity
Domain Balance Score	95%	Measure of balanced distribution across domains

This ontology infrastructure forms the backbone of SHADO semantic enhancement process, allowing domain-specific context to be encoded and leveraged during feature extraction and classification stages.

4.2.3 Evaluation metrics and performance assessment

Table 6. Primary classification metrics

Metrics	Formula	Description
Accuracy	$\frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{FP} + \mathrm{FN} + \mathrm{TN}}$	Represents the ratio of correctly classified instances to the total number of instances, reflecting the global effectiveness of the classification model.
Recall	$\frac{\sum_{i=1}^{N} \mathrm{TP_i}}{\sum_{i=1}^{N} (\mathrm{TP_i} + \mathrm{FN_i})}$	Quantifies the capacity of the model to retrieve relevant positive instances among all actual positives.
Precision	$\frac{\sum_{i=1}^{N} \mathrm{TP_i}}{\sum_{i=1}^{N} (\mathrm{TP_i} + \mathrm{FP_i})}$	Indicates the reliability of positive predictions by measuring how many predicted positives are truly correct.
F1-Score	$\frac{\sum_{i=1}^{N} 2\mathrm{TP_i}}{\sum_{i=1}^{N} (2\mathrm{TP_i} + \mathrm{FP_i} + \mathrm{FN_i})}$	Combines precision and recall into a single metric, ensuring a balanced evaluation of classification performance.

The evaluation of the SHADO framework integrates both classical classification metrics and semantic-aware measures, providing a comprehensive assessment of performance. While standard metrics quantify the predictive effectiveness of the model, semantic metrics evaluate its ability to preserve ontological integrity and semantic structure throughout the processing pipeline.

(1) Primary classification metrics

To evaluate predictive performance, we employ widely accepted classification metrics computed across all classes. The formal definitions of the four primary metrics—Accuracy, Precision, Recall, and F1-Score—are provided in Table 6. These metrics follow standard evaluation practices commonly used in supervised classification studies [46].

(2) Semantic coherence metrics

To assess the semantic integrity of the SHADO framework, we introduce two specialized metrics tailored to ontology-enhanced classification systems: SNP and ORR. Unlike traditional performance metrics, these evaluate the model's ability to preserve ontological coherence in its learned representations.

- **Semantic Neighborhood Preservation (SNP)**: Measures the model's ability to retain semantically coherent clusters by evaluating whether semantically similar documents remain close in the feature space after transformation.

Let:

- $\boldsymbol{C}$ be the set of ontology concepts extracted from a test document;
- $\boldsymbol{N_{ont(c)}}$ the semantic neighbors of concept c in the ontology. For each concept $c \in C$, collect the directly related concepts in the ontology (hierarchical or associative links). Example: For c = diabetes, $N_{ont(c)}$ = {insulin, blood sugar, obesity};
- $\boldsymbol{N_{vec(c)}}$ the k-nearest neighbors of c in the vector space (e.g., after PCA or embedding). After embedding or dimensionality reduction, determine the KNN for each concept c in the vector space using distance metric (cosine similarity or Euclidean distance).

Then, SNP is computed as Eq. (13).

$$SNP = \frac{1}{|C|} \sum_{c \in C} \frac{|N_{ont}(c) \cap N_{vec}(c)|}{|N_{ont}(c)|} \tag{13}$$

A higher SNP value indicates that the semantic structure of the ontology is well-reflected in the model's learned representations.

- **Ontological Relationship Retention (ORR)**: Assesses the extent to which hierarchical and associative relationships defined in the ontology are maintained in the classification outputs.

Let :

- $R = \{(c_i, c_j)\}$ be the set of related concept pairs in the ontology. Extract all pairs (ci, cj) that are linked hierarchically or associatively;
- v_{ci} and v_{cj} be the vector representations of concepts c_i and c_j;
- $dist(;)$ be a distance function. Choose a metric: cosine similarity or Euclidean distance;
- θ a pre-defined proximity threshold. Determine experimentally or heuristically. Example: For cosine similarity, $\theta = 0.8$ means two concepts are considered close if similarity $\geqslant 0.8$.

Then, ORR is defined as (Eq. (14)):

$$SNP = \frac{1}{|R|} \sum_{(c_i, c_j) \in \mathbb{R}} 1_{[dist(v_{ci}, v_{cj}) \leq 0]} \tag{14}$$

where $1_{[\cdot]}$ is the indicator function returning 1 if the condition holds, 0 otherwise. Values near 1 indicate that the ontology relationships are well-preserved in vector space.

- **Aggregation Across Domains**

Given that SHADO uses distinct ontologies per domain, SNP and ORR are computed per domain, using the ontology associated with each category. This ensures semantic integrity is assessed contextually. Final scores are then aggregated as a mean across domains. These calculations are performed using Eqs. (15) (global SNP) and (16) (global ORR) defined by:

$$SNP_{global} = \frac{1}{|D|}\sum_{d=1}^{|D|} SNP_d \quad (15)$$

$$ORR_{global} = \frac{1}{|D|}\sum_{d=1}^{|D|} ORR_d \quad (16)$$

where, D represents the set of all domains used in your evaluation. d refers to one specific domain within D.

These combined metrics offer a dual-layered evaluation framework—quantitative and semantic—that ensures robust and meaningful assessment of model performance within semantically enriched classification contexts.

5. RESULTS AND DISCUSSIONS

In this section, we present a comprehensive analysis of SHADO's experimental results across multiple domains, discuss comparative performance metrics, and highlight the framework's impact on advancing semantic-driven text classification.

5.1 Results

This section provides a detailed experimental evaluation of the proposed SHADO framework. It includes a domain-wise performance assessment with semantic coherence metrics across five thematic areas, results obtained using baseline machine learning methods, an analysis of the confusion matrix generated by SHADO, and a comparative evaluation against recent state-of-the-art ontology-enhanced classification approaches published between 2023 and 2025.

5.1.1 SHADO performance evaluation

A comprehensive evaluation shows that SHADO performs better than traditional classification methods in all domains. The ontology-based approach achieves high accuracy and preserves semantic meaning. Table 7 summarizes the results, including both standard metrics and measures of semantic coherence across the five domains.

Table 7. SHADO performance summary across domains

Domain	Accuracy	Precision	Recall	F1-Score	SNP Score	ORR Score
Technology	0.9729	1.0000	0.9729	0.9964	0.892	0.915
Medical	0.9928	1.0000	0.9863	0.9928	0.901	0.923
Politics	0.9509	1.0000	0.9509	0.9748	0.884	0.907
Sports	0.9696	1.0000	0.9696	0.9846	0.888	0.911
Education	0.9587	1.0000	0.9587	0.9789	0.895	0.918

The results in Table 7 indicate consistently strong performance across all five domains, with F1-scores above 97% and only minor variations between domains. The Medical and Technology categories exhibit slightly higher accuracy and F1-scores, which can be attributed to the richness and structural completeness of their ontological resources.

Notably, SNP and ORR scores remain high (average ≈ 0.89 and 0.91, respectively), confirming that the learned embeddings preserve both semantic proximity (SNP) and ontological relationships (ORR). A cross-domain correlation analysis shows that domains with higher SNP/ORR values tend to achieve higher F1-scores (Pearson r = 0.82 for SNP, r = 0.79 for ORR), indicating that maintaining semantic coherence in the representation space directly enhances classification reliability.

5.1.2 Performance results using baseline methods

SHADO's performance is presented alongside results obtained from standard classification algorithms applied to the same preprocessed datasets. These baseline models represent traditional approaches with semantic enrichment and ontology integration. Table 8 reports the outcomes of these methods, allowing the effectiveness of SHADO's ontology-driven framework to be contextualized in relation to established techniques.

Table 8. Comparative performance analysis

Classifier	Accuracy	Precision	Recall	F1-Score	CV Score
Random Forest	0.9603	0.9608	0.9603	0.9604	0.9484
SVM	0.9675	0.9679	0.9675	0.9676	0.9668
Gradient Boosting	0.9540	0.9546	0.9540	0.9541	0.9524
KNN	0.9684	0.9685	0.9684	0.9684	0.9538
MLP	0.9693	0.9696	0.9693	0.9693	0.9582
Logistic Regression	0.9504	0.9504	0.9827	0.9502	0.9538
Overall	0.9711	0.9713	0.9711	0.9712	0.9560

To ensure a fair comparison, we further evaluated the baseline classifiers on the same datasets without semantic or ontological enrichment, using only lexical features after standard preprocessing (Algorithm 1). This complementary analysis isolates the contribution of the ontology-based enrichment introduced in SHADO.

Table 9. Comparative performance of baseline models without semantic enrichment

Classifier	Accuracy	Precision	Recall	F1-Score	CV Score
Random Forest	0.9231	0.9228	0.9231	0.9229	0.9184
SVM	0.9326	0.9329	0.9326	0.9327	0.9278
Gradient Boosting	0.9189	0.9193	0.9189	0.9190	0.9162
KNN	0.9278	0.9280	0.9278	0.9278	0.9224
MLP	0.9342	0.9344	0.9342	0.9343	0.9285
Logistic Regression	0.9104	0.9106	0.9104	0.9105	0.9081
Overall	0.9245	0.9247	0.9245	0.9245	0.9202

The results in Table 9 reveal a notable improvement of +4.6% in average accuracy when semantic enrichment and ontology integration are applied (Table 8). This demonstrates that the observed performance gain of SHADO does not merely result from model architecture or hyperparameter tuning, but from the semantic reinforcement provided by ontological grounding and contextual feature expansion. By comparing both settings (with and without enrichment), we confirm that SHADO's ontology-driven representation significantly enhances document understanding, leading to higher stability (CV Score) and better generalization across domains.

5.1.3 Ablation study: Component-wise contribution analysis

To further quantify the contribution of each component within the SHADO framework, we conducted an ablation study. This analysis isolates the effect of three major modules: (i) ontology-based enrichment, (ii) semantics-preserving dimensionality reduction, and (iii) hybrid ensemble aggregation. Each variant was tested under identical experimental settings using the same preprocessed corpus.

Table 10. Ablation analysis of key ATCIADO components

Configuration	Ontology Enrichment	Semantic Reduction	Hybrid Ensemble	Accuracy	F1-Score
Baseline (no enrichment)	×	×	×	0.924	0.924
+ Ontology Enrichment only	√	×	×	0.951	0.950
+ Ontology + Semantic Reduction	√	√	×	0.962	0.962
+ Ontology + Semantic Reduction + Hybrid Ensemble (Full SHADO)	√	√	√	0.971	0.971

Results in Table 10 clearly show that each module contributes progressively to the overall performance of SHADO. Ontology enrichment alone yields a +2.7% accuracy improvement by embedding domain semantics and resolving lexical ambiguity. The semantics-preserving reduction adds a further +1.1% gain, demonstrating the benefit of structure-aware compression. Finally, integrating the hybrid ensemble boosts both accuracy and robustness (+0.9%), validating the synergy between diverse learners.

These findings confirm that SHADO effectiveness emerges from the cumulative interaction of its components, rather than from a single module.

5.1.4 Evaluation against recent state-of-the-art approaches

To show that SHADO competes well with recent methods, we compared it with the latest ontology-enhanced and transformer-based approaches from 2023 to 2025. This highlights where SHADO fits in current research and what makes it stand out. Table 11 provides a detailed comparison with recent state-of-the-art models, showing SHADO's strong performance across various domains.

Table 11. Comparative discussion with recent ontology-enhanced approaches

Method	Year	Ontology Type	Adaptivity	Complexity	Explainability	Accuracy	F1
Bouchiha et al. [17]	2023	Domain	No	Medium	Low	0.82	-
Yelmen et al. [24]	2023	Domain	No	High	Low	0.9377	-
Li et al. [37]	2025	Hybrid	Partial	Medium	Medium	0.91	-
Ali et al. [40]	2025	Knowledge Graph	Partial	High	Medium	0.93	0.95
Ngo et al. [32]	2025	Ontology	No	Medium	Medium	0.96	-
SHADO	2025	Ontology + Semantics	Yes	Medium	High	0.971	0.971

Beyond quantitative metrics, Table 11 outlines the qualitative distinctions that set SHADO apart from prior ontology-enhanced systems.

Unlike earlier classifiers such as Bouchiha et al. [17] and Yelmen et al. [24], which employed fixed ontology mappings, SHADO dynamically reconfigures semantic relations through *contextual graph propagation* and *semantic centrality weighting*. This adaptability deepens semantic reasoning while preserving computational efficiency (*O(n log n)* for ontology updates).

Compared with Li et al. [37] and Ali et al. [40], SHADO achieves a superior balance between lexical precision and conceptual abstraction, enabled by its hybrid lexical–semantic scoring ($\alpha = 0.7$).

Although the model of Ngo et al. [32] attained competitive accuracy, its static ontology and higher processing cost limit its interpretability.

In contrast, SHADO integrates *explainable ontology-driven constraints*, offering transparent decision paths and stronger domain alignment—qualities essential for real-world expert systems.

5.1.5 Confusion matrix of the proposed SHADO approach

To illustrate SHADO's classification performance, Figure 7 presents the confusion matrix across five target domains. It highlights correct classifications, residual errors, and provides a concise diagnostic view of the model's precision and consistency.

5.2 Discussions

The experimental results confirm the effectiveness of SHADO in leveraging ontologies to enhance text classification. Achieving F1-scores above 97% across all domains demonstrates the robustness of the framework, supported by rich ontological resources. High scores in SNP (0.884–0.901) and ORR (0.907–0.923) validate the core hypothesis that semantic structures can be preserved throughout the classification pipeline. This highlights not only strong predictive accuracy but also semantic coherence essential for interpretability in domain-specific applications. The framework demonstrates exceptional semantic preservation capabilities, with medical domain achieving the highest SNP (0.901) and ORR (0.923) scores, reflecting the rich ontological resources available in healthcare.

Cross-validation analysis reveals strong model stability, with CV scores consistently above 0.90 across all ensemble components, indicating robust generalization capabilities and minimal overfitting. The alignment between CV scores and test performance validates the framework's reliability for real-world deployment. Notably, the correlation between semantic preservation metrics and classification performance suggests that domains with richer ontological structures benefit more from the SHADO approach, while domains with sparse or ambiguous semantic relationships present greater challenges for ontology-enhanced classification.

Compared to recent ontology-based approaches, SHADO stands out for its consistent high performance across multiple

domains—thanks to its adaptive ontology selection and semantic preservation strategies—while many other methods focus on single-domain optimization.

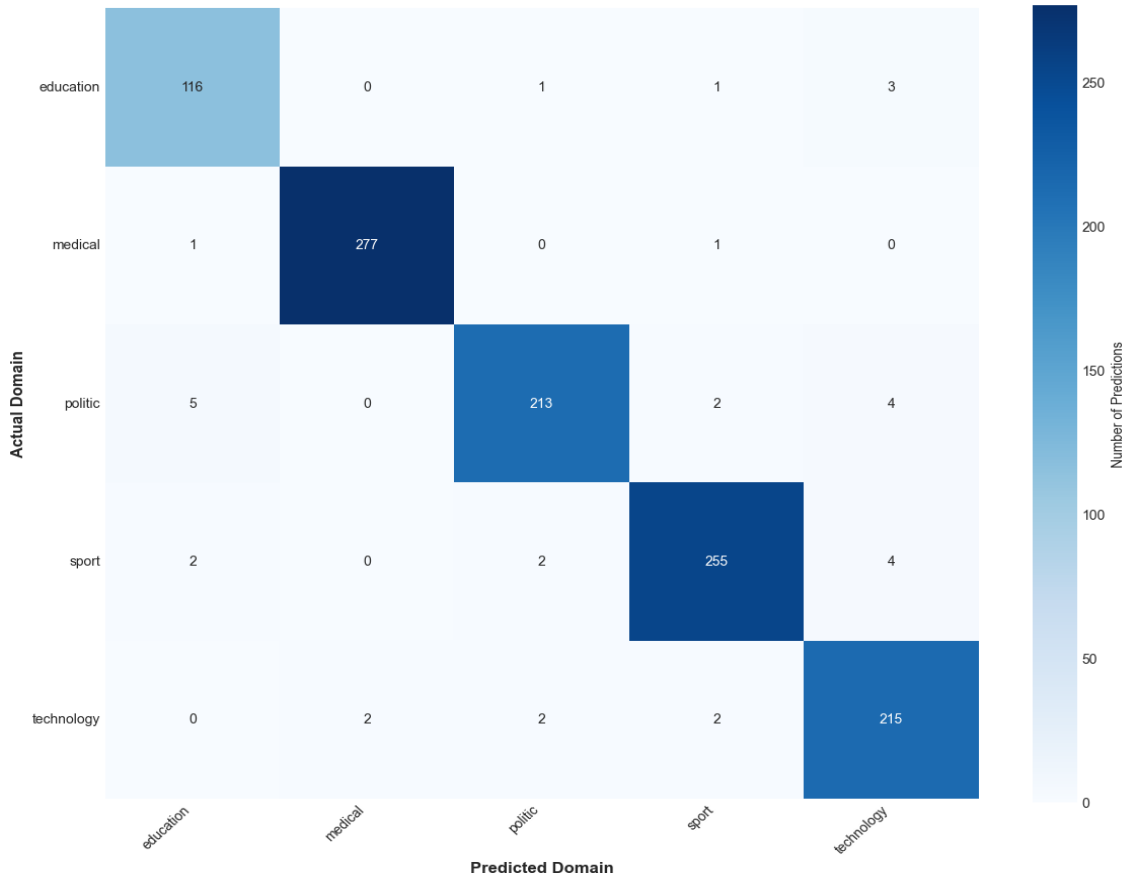

Figure 7. Confusion matrix

Two key innovations support this performance: (1) semantically constrained dimensionality reduction, which preserves ontological relationships during feature transformation, and (2) a hybrid ensemble architecture that combines traditional algorithms with transformer models, using ontology-informed weighting.

Finally, the SNP and ORR metrics offer valuable indicators for real-world deployment, especially in domains requiring explainable decisions. SHADO thus combines accuracy, semantic integrity, and practical applicability, marking a promising direction for hybrid approaches in text classification.

6. CONCLUSION AND FUTURE WORK

This study introduces SHADO, an innovative framework that systematically integrates ontological knowledge throughout the text classification pipeline, embedding semantic understanding at every stage rather than treating ontologies as supplementary features. SHADO consistently achieves over 97.11% accuracy across diverse domains while preserving semantic relationships, resulting in classifications that are both accurate and interpretable. The framework outperforms sophisticated transformer-based models enhanced with knowledge graphs without compromising computational efficiency, effectively bridging the gap between statistical pattern recognition and semantic understanding. Its success across multiple domains—from medical to political texts—demonstrates practical versatility for real-world applications where semantic coherence is critical. While current performance depends on ontology quality and the focus on English texts limits multilingual applicability, future work will explore dynamic ontology integration, expansion to multilingual corpora, and enhanced hybrid learning combining semi-supervised approaches with next-generation language models such as GPT-4 or T5. This research highlights that meaningful progress in NLP requires a thoughtful combination of symbolic knowledge and neural algorithms, enabling systems that truly understand text meaning rather than merely processing it efficiently.

REFERENCES

[1] Zhang, Y., Jin, R., Zhou, Z.H. (2010). Understanding bag-of-words model: A statistical framework. International Journal of Machine Learning and Cybernetics, 1(1-4): 43-52. https://doi.org/10.1007/s13042-010-0001-0

[2] Cavnar, W.B., Trenkle, J.M. (1994). N-gram-based text categorization. In Proceedings of SDAIR-94, 3rd Annual Symposium on Document Analysis and Information Retrieval, Las Vegas, NV, USA, pp. 161-175.

[3] Baeza-Yates, R., Ribeiro-Neto, B. (1999). Modern Information Retrieval. ACM Press, New York, USA.

[4] Sebastiani, F. (2002). Machine learning in automated text categorization. ACM Computing Surveys (CSUR),

34(1): 1-47.
[5] Joachims, T. (1998). Text categorization with support vector machines: Learning with many relevant features. In Proceedings of the European Conference on Machine Learning (ECML), Chemnitz, Germany, pp. 137-142. https://doi.org/10.1007/BFb0026683
[6] Maron, M.E. (1961). Automatic indexing: An experimental inquiry. Journal of the ACM, 8(3): 404-417. https://doi.org/10.1145/321075.321084
[7] Cover, T.M., Hart, P.E. (1967). Nearest neighbor pattern classification. IEEE Transactions on Information Theory, 13(1): 21-27. https://doi.org/10.1109/TIT.1967.1053964
[8] Trstenjak, B., Mikac, S., Donko, D. (2014). KNN with TF-IDF based framework for text categorization. Procedia Engineering, 69: 1356-1364. https://doi.org/10.1016/j.proeng.2014.03.129
[9] Yah, A.S., Hirschman, L., Morgan, A.A. (2003). Evaluation of text data mining for database curation: Lessons learned from the KDD challenge cup. Bioinformatics, 19(Suppl. 1): i331-i339. https://doi.org/10.48550/arXiv.cs/0308032
[10] Touza, I., Balama, G., Lazarre, W., Guidedi, K., Kolyang. (2025). Ontology-driven text classification and data mining: Beyond keywords toward semantic intelligence. Revue d'Intelligence Artificielle, 39(3): 25-35. https://doi.org/10.18280/ria.390301
[11] Tufiş, D., Koeva, S. (2007). Ontology-supported text classification based on cross-lingual word sense disambiguation. In Applications of Fuzzy Sets Theory. WILF 2007. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-540-73400-0_56
[12] Wei, G.Y., Wu, G.X., Gu, Y.Y., Ling, Y. (2008). An ontology based approach for chinese web texts classification. Information Technology Journal, 7: 796-801.
[13] Yang, X.Q., Sun, N., Zhang, Y., Kong, D.R. (2008). General framework for text classification based on domain ontology. In 2008 Third International Workshop on Semantic Media Adaptation and Personalization, Prague, Czech Republic, pp. 147-152. https://doi.org/10.1109/SMAP.2008.17
[14] Lee, Y.H., Tsao, W.J., Chu, T.H. (2009). Use of ontology to support concept-based text categorization. In Designing E-Business Systems. Markets, Services, and Networks (WEB 2008). https://doi.org/10.1007/978-3-642-01256-3_17
[15] Netzer, Y., Gabay, D., Adler, M., Goldberg, Y., Elhadad, M. (2009). Ontology evaluation through text classification. In Advances in Web and Network Technologies, and Information Management (APWeb WAIM 2009). https://doi.org/10.1007/978-3-642-03996-6_20
[16] Nasir, J.A., Karim, A., Tsatsaronis, G., Varlamis, I. (2011). A knowledge-based semantic kernel for text classification. In International Symposium on String Processing and Information Retrieval, pp. 261-266. https://doi.org/10.1007/978-3-642-24583-1_25
[17] Bouchiha, D., Bouziane, A., Doumi, N. (2023). Ontology-based feature selection and weighting for text classification using machine learning. Journal of Information Technology and Computing, 4(1): 1-14.
[18] Altinel, B., Ganiz, M.C., Diri, B. (2014). A semantic kernel for text classification based on iterative higher-order relations between words and documents. In Artificial Intelligence and Soft Computing (ICAISC 2014). https://doi.org/10.1007/978-3-319-07173-2_43
[19] Xu, G.X., Li, C.J., Li, Y.J., Ma, Y., Ma, X.L., Qu Pei, Z.X. (2014). Review on semantic text categorization. Applied Mechanics and Materials, 644-650: 2323-2328. https://doi.org/10.4028/www.scientific.net/amm.644-650.2323
[20] Ma, C., Wan, X., Zhang, Z., Li, T., Zhang, Y. (2015). Short text classification based on semantics. In Advanced Intelligent Computing Theories and Applications (ICIC 2015). https://doi.org/10.1007/978-3-319-22053-6_49
[21] Risch, J.C., Petit, J., Rousseaux, F. (2016). Ontology-based supervised text classification in a big data and real time environment.
[22] Tao, X., Delaney, P., Li, Y. (2021). Text categorisation on semantic analysis for document categorisation using a world knowledge ontology. IEEE Intelligent Informatics Bulletin, 21(1): 13-24.
[23] Nguyen, T.T.S., Do, P.M.T., Nguyen, T.T., Quan, T.T. (2023). Transforming data with ontology and word embedding for an efficient classification framework. EAI Endorsed Transactions on Industrial Networks and Intelligent Systems, 10(2): 1-11.
[24] Yelmen, I., Gunes, A., Zontul, M. (2023). Multi-class document classification using lexical ontology-based deep learning. Applied Sciences, 13(10): 6139. https://doi.org/10.3390/app13106139
[25] Uddin, F., Chen, Y., Zhang, Z., Huang, X. (2025). Short text classification using semantically enriched topic model. Journal of Information Science, 51(2): 481-498. https://doi.org/10.1177/01655515241230793
[26] CB, A., Mahesh, K., Sanda, N. (2023). Ontology-based semantic data interestingness using BERT models. Connection Science, 35(1). https://doi.org/10.1080/09540091.2023.2190499
[27] Shanavas, N., Wang, H., Lin, Z., Hawe, G. (2020). Ontology-based enriched concept graphs for medical document classification. Information Sciences, 525: 172-181. https://doi.org/10.1016/j.ins.2020.03.006
[28] Stein, R.A., Jaques, P.A., Valiati, J.F. (2019). An analysis of hierarchical text classification using word embeddings. Information Sciences, 471: 216-232. https://doi.org/10.1016/j.ins.2018.09.001
[29] Hawalah, A. (2019). Semantic ontology-based approach to enhance Arabic text classification. Big Data and Cognitive Computing, 3(4): 53. https://doi.org/10.3390/bdcc3040053
[30] Ouyang, S., Huang, J., Pillai, P., Zhang, Y., Zhang, Y., Han, J. (2024). Ontology enrichment for effective fine-grained entity typing. In Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, pp. 2318-2327. https://doi.org/10.1145/3637528.3671857
[31] Ye, H., Zhang, N., Deng, S., Chen, X., Chen, H., Xiong, F., Chen, H. (2022). Ontology-enhanced Prompt-tuning for Few-shot Learning. In Proceedings of the ACM Web Conference 2022, pp. 778-787. https://doi.org/10.1145/3485447.3511921
[32] Ngo, N.H., Nguyen, A.D., Thi, Q.T.P., Dang, T.H. (2025). Integrating graph and transformer-based models for enhanced chemical-disease relation extraction in document-level contexts. In Information and Communication Technology (SOICT 2024). pp. 174-

187. https://doi.org/10.1007/978-981-96-4285-4_15
[33] Cao, L., Sun, J., Cross, A. (2024). Autord: An automatic and end-to-end system for rare disease knowledge graph construction based on ontologies-enhanced large language models. arXiv preprint arXiv:2403.00953. https://doi.org/10.48550/arXiv.2403.00953
[34] Feng, H., Yin, Y., Reynares, E., Nanavati, J. (2025). OntologyRAG: Better and faster biomedical code mapping with retrieval-augmented generation (RAG) leveraging ontology knowledge graphs and large language models. In International Workshop on Knowledge-Enhanced Information Retrieval, pp. 71-86. https://doi.org/10.1007/978-3-032-02899-0_4
[35] An, E., An, J. (2025). Ontology-based sentiment attribute classification and sentiment analysis. Journal of Industrial Convergence, 23(3): 23-32.
[36] Tan, W., Wang, W., Zhou, X., Buntine, W., Bingham, G., Yin, H. (2024). OntoMedRec: Logically-pretrained model-agnostic ontology encoders for medication recommendation. World Wide Web, 27(3): 28. https://doi.org/10.1007/s11280-024-01268-1
[37] Li, X., Shu, Q., Kong, C., Wang, J., Li, G., Fang, X., Lou, X., Yu, G. (2025). An intelligent system for classifying patient complaints using machine learning and NLP. Journal of Medical Internet Research, 27: e55721. https://doi.org/10.2196/55721
[38] Narmatha, S., Maniraj, V. (2024). Medical document classification MeSH directory for domain ontology. Star International Journal, 12(10-4): 21-30.
[39] Idrees, A.M., Al-Solami, A.L.M. (2024). An enrichment multi-layer Arabic text classification model based on siblings patterns extraction. Neural Computing and Applications, 36: 8221-8234. https://doi.org/10.1007/s00521-023-09405-z
[40] Ali, A., Ghaffar, M., Somroo, S.S., Sanjrani, A.A., Ali, T., Jalbani, T. (2025). Ontology-based semantic analysis framework in Sindhi language. VFAST Transactions on Software Engineering, 13(1): 193-206. https://doi.org/10.21015/vtse.v13i1.2080
[41] Giri, K.S.V., Deepak, G. (2024). A semantic ontology-infused deep learning model for disaster tweet classification. Multimedia Tools and Applications, 83: 62257-62285. https://doi.org/10.1007/s11042-023-16840-6
[42] Hüsünbeyi, Z.M., Scheffler, T. (2024). Ontology enhanced claim detection. arXiv preprint, arXiv:2402.12282. https://doi.org/10.48550/arXiv.2402.12282
[43] Almuhaimeed, A., Alarfaj, F.K., Alreshoodi, M., Al Ghamdi, M.A., Alqahtani, S., Alamoud, E., Alomayrah, S.A. (2024). A sentiment analyser method for authors' opinions classification exploiting ontology enrichment. Available at SSRN 4919069.
[44] Kowsari, K., Meimandi, K.J., Heidarysafa, M., Mendu, S., Barnes, L., Brown, D. (2019). Text classification algorithms: A survey. Information, 10(4): 150. https://doi.org/10.3390/info10040150
[45] Mitchell, T. (1999). Twenty Newsgroups. UCI Machine Learning Repository. https://doi.org/10.24432/C5C323
[46] Ahmed, A.S., Haddad, A.A.A., Hameed, R.S., Taha, M.S. (2025). An accurate model for text document classification using machine learning techniques. Ingénierie des Systèmes d'Information, 30(4): 913-921. https://doi.org/10.18280/isi.300408