



Enhancing Named Entity Recognition for Email Text using Domain-Specific Tags

Sadakshini Krishnathass^{a*}, Yasotha Ram Ramanan^b, Ann Sinthusha Anton
Vijeevaraj^c

^a University of Vavuniya, Sri Lanka

^b University of Vavuniya, Sri Lanka

^c University of Vavuniya, Sri Lanka

^aEmail: sadakshinik@gmail.com

^bEmail: yasotha@vau.ac.lk

^cEmail: annsinthusha@vau.ac.lk

Abstract

Named Entity Recognition (NER) is a fundamental task in Natural Language Processing (NLP) that identifies and classifies entities such as dates, organizations, and people. However, generic NER models often underperform in domain-specific contexts, failing to detect entities like email addresses and website links that are essential for applications such as spam detection. This study proposes an enhanced NER framework that introduces two custom domain-specific tags-EMAIL and WEBSITE- using rule-based methods. Preprocessing strategies were incorporated to resolve tokenization inconsistencies in monetary values, time expressions, and URLs. The enriched entity set, combining SpaCy's default NER output with the custom tags, was transformed into feature vectors alongside TF-IDF representations and used to train multiple machine-learning classifiers, including Multinomial Naïve Bayes, Support Vector Machine (SVM), and XGBoost. Experimental results on a public spam dataset show that the enhanced model achieved a peak accuracy of 95.99%, outperforming the TF-IDF-only baseline (91.4%). An ablation study confirmed that the custom EMAIL and WEBSITE entities contributed up to a 4.5% improvement in accuracy. Interpretability analysis using Local Interpretable Model-Agnostic Explanations (LIME) validated the influence of

these domain-specific entities on classification outcomes. While results highlight measurable performance gains, limitations include reliance on a single dataset and reduced effectiveness against obfuscated entity formats. The approach is adaptable to other NLP applications such as phishing detection, fraud prevention, and customer service analytics, demonstrating that lightweight domain-specific enhancements can deliver tangible benefits in specialized contexts.

Keywords: custom named entity recognition; domain-specific named entity recognition; Named Entity Recognition; Natural Language Processing, SpaCy

1. Introduction

Named Entity Recognition (NER) is a core task in Natural Language Processing (NLP) that converts unstructured text into structured data by identifying and classifying entities such as people, organizations, dates, and monetary values. It plays a pivotal role in downstream applications including information extraction, question answering, content classification, and decision-support systems.

Despite strong performance in general domains, widely used NER models such as SpaCy and Stanford NLP often underperform in domain-specific contexts. In email processing tasks like spam detection, they frequently miss key indicators such as email addresses and website URLs due to irregular formats and limited representation in standard training corpora. This gap can reduce the effectiveness of classification systems, particularly against sophisticated spam or phishing attempts.

To address these limitations, this work introduces an enhanced NER framework that augments SpaCy's default pipeline with two domain-specific entity types **EMAIL** and **WEBSITE**—identified via rule-based tagging. Preprocessing strategies are also applied to normalize time expressions, monetary formats, and URLs, mitigating tokenization inconsistencies that hinder entity recognition.

The objectives of this study are to:

- Develop a rule-based tagging approach for accurately extracting EMAIL and WEBSITE entities not effectively captured by standard NER models.
- Integrate these custom tags with SpaCy's default outputs to create a domain-adapted NER framework.
- Evaluate the framework's impact on spam detection using multiple machine learning classifiers and metrics, including accuracy, precision, recall, and F1-score.
- Perform an ablation study to quantify the contribution of the domain-specific tags to overall classification performance.
- Demonstrate the applicability of the approach in other NLP contexts such as phishing detection, fraud prevention, and cybersecurity.

The novelty of this work lies in its lightweight yet effective integration of rule-based tagging with a statistical NER

framework, enabling measurable gains in email spam classification without large-scale retraining. The methodology is easily adaptable to other specialized domains with minimal computational overhead, offering a practical path to domain-aware NER in real-world systems.

2. Background

Named Entity Recognition (NER) is a core Natural Language Processing (NLP) task that identifies and classifies entities such as names, dates, and numerical expressions in unstructured text. It is a foundational component in applications such as information extraction, document classification, question answering, and spam detection.

Although modern NER tools such as SpaCy's **en_core_web_sm** perform well on general-purpose corpora, they often struggle in domain-specific scenarios—particularly when recognizing patterns like email addresses, website URLs, and irregular expressions of time or money. These shortcomings are primarily due to inconsistent tokenization, limited representation of such formats in training data, and a lack of explicit domain adaptation.

To address these challenges, this research enhances SpaCy's pre-trained NER model with rule-based annotations for two new entity types—**EMAIL** and **WEBSITE**—and integrates them with SpaCy's default output to create a unified, domain-adapted entity recognition pipeline. As outlined in the Introduction, the study evaluates this enhanced NER in the context of email spam detection, where domain-specific cues such as promotional URLs or obfuscated email addresses are often strong indicators of malicious content.

Although the present work focuses on spam filtering, the methodology is adaptable to other NLP domains, including phishing detection, fraud prevention, and cybersecurity. Acknowledging that this evaluation uses a single public dataset, future work should validate the approach on diverse and adversarial corpora to assess robustness.

As outlined in the Introduction, this enhanced NER forms the basis for the spam detection framework evaluated in this study.

3. Literature Review

Named Entity Recognition (NER) is a fundamental task in Natural Language Processing (NLP) that identifies and classifies key entities such as names, dates, locations, organizations, and domain-specific elements within text. In spam detection, NER is particularly valuable for extracting entities like email addresses, URLs, and monetary values, which frequently appear in unsolicited messages. Incorporating NER into spam detection improves contextual understanding and model precision, especially when custom models are designed to recognize domain-specific entities [1], [2], [3]

Custom NER models face several challenges. Pre-trained models from NLP libraries like SpaCy, Apache OpenNLP, and TensorFlow often underperform when applied to domain-specific text, requiring extensive labeled datasets and frequent retraining [1], [4]. Evaluation metrics such as accuracy and F1-score may not fully reflect real-world performance, particularly for imbalanced datasets. Reported training accuracies include SpaCy (100%), TensorFlow (99.55%) and Apache OpenNLP (100%) [4]. Moreover, continuous updates are essential to capture evolving language patterns and emerging entities, a need not fully addressed by these methods.

Crowdsourced annotation using platforms like Amazon Mechanical Turk (MTurk) has been explored for large-scale NER. Eustice et al.[5] demonstrated that competitive bonuses and inter-annotator agreement improve annotation quality over 20,000 emails, achieving precision above 90%. However, challenges remain due to variable worker expertise and a fixed-fee structure, which may reduce recall or accuracy in complex datasets.

Rule-based approaches, such as NERL, offer high explainability and domain adaptability [6], [7]. Chiticariu et al. [6] developed NERL atop SystemT to simplify rule creation and customization. These systems can match or outperform machine learning models but require substantial manual effort and domain expertise, limiting scalability and introducing variability across implementations.

Machine learning-based NER has also been applied to spam detection. Taghandiki [2] combined SpaCy preprocessing with Naive Bayes, Decision Tree C45, and Multilayer Perceptron (MLP) algorithms, achieving 96% accuracy on a dataset of 1,500 emails. While effective, such methods may struggle to generalize to unseen spam types, and preprocessing techniques may not fully capture linguistic nuances like sarcasm. Similarly, hybrid approaches combining TF-IDF with classifiers such as Naive Bayes, SVM, and Random Forest have reached up to 98% accuracy, though they still face challenges from false positives and evolving spam patterns[8], [9], [10]

Privacy-preserving methods have been proposed, where named entities are replaced with category labels to enhance text classification [11] While this reduces noise and protects sensitive data, critical contextual information can be lost, potentially diminishing classifier performance. Vasiliev [3] emphasized practical NLP tools like SpaCy for NER tasks, highlighting usability and adaptability across applications.

Recent surveys [4] comprehensive comparisons of NER techniques, including rule-based, machine learning, and hybrid approaches, assessing libraries such as SpaCy, Apache OpenNLP, StanfordNLP, and TensorFlow. These studies underline that no single method is universally optimal, and choice depends on dataset characteristics, entity types, and domain specificity.

Despite advances, prior research largely addresses NER and spam detection independently, with few studies

integrating domain-specific tags into NER pipelines for spam classification. This study fills that gap by incorporating rule-based **EMAIL** and **WEBSITE** entities to enhance feature relevance and improve classification performance in spam detection tasks.

4. Methodology

This research proposes an enhanced Named Entity Recognition (NER) framework tailored for domain-specific

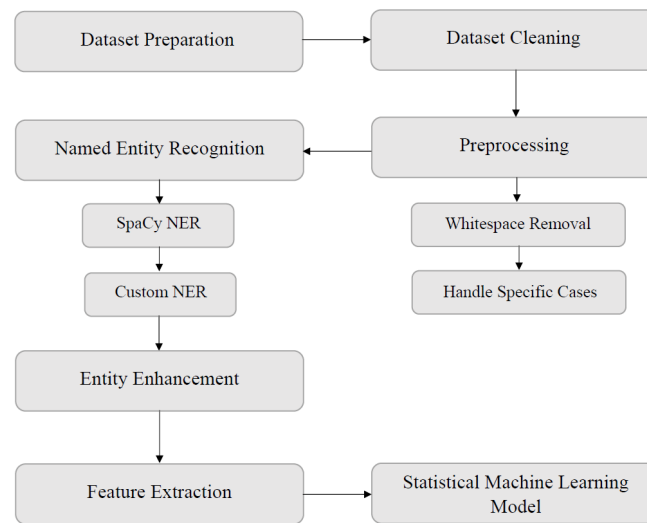


Figure 1: Workflow of the proposed methodology, illustrating the stages from dataset preparation to evaluation.

contexts, with a focus on email-based spam detection. The approach integrates standard SpaCy-based NER with custom rule-based tagging to capture entities—**EMAIL** and **WEBSITE**—that are often missed in generic models. Figure 1 presents the overall workflow, with each stage described in detail below.

4.1. Dataset Preparation and Cleaning

The initial preparation phase involved the following steps:

- **Duplicate Removal:** Identical records, often originating from repeated submissions or dataset compilation artifacts, were identified through hash-based comparison of message content and removed to prevent bias from over-represented samples.
- **Irrelevant Metadata Removal:** Certain fields, such as dataset-specific IDs or collection timestamps, were excluded as they provided no semantic contribution to the task of entity recognition and could inadvertently introduce noise.
- **Incomplete Record Filtering:** Emails with missing body content or corrupted text were excluded, as incomplete samples risked reducing the quality of entity detection and model generalization.
- **Class Balancing:** The original dataset exhibited class imbalance (approximately 24% spam and 76% non-spam). To mitigate skew and improve model fairness, minority-class augmentation was performed using paraphrasing techniques and synthetic sample generation. This ensured that spam instances were proportionally increased while preserving semantic intent.

Through this process, the dataset was refined into a clean, balanced corpus suitable for robust training

and evaluation of the enhanced NER model.

4.2. Preprocessing

Prior to entity recognition, the textual data underwent a systematic preprocessing stage to ensure consistency, reduce noise, and preserve semantic fidelity. The goal was to standardize the representation of entities so that model training could effectively capture meaningful patterns without being confounded by formatting inconsistencies.

The preprocessing pipeline comprised:

- **Whitespace Normalization:** Consecutive spaces, tab characters, and excessive newlines were collapsed into a single space. This step prevented artificial token boundaries caused by inconsistent spacing, thereby improving token coherence.
- **Pattern Normalization:** Using carefully constructed regular expressions, common entity patterns were standardized to avoid token fragmentation. This step ensured that the tokenizer recognized complete entities as single, meaningful units. Key transformations included:
 - **Monetary Values:** Examples such as “\$ 3 . 50” were reformatted into “\$3.50” to maintain numerical integrity.
 - **Time Expressions:** Variations like “2 : 30 p . m.” were converted to “2:30 p.m.” for temporal consistency.
 - **Decimal Numbers:** Disjoint representations such as “3 . 14” were standardized to “3.14”.
 - **URLs:** Fragmented web addresses (e.g., “http : / / site . com”) were reconstructed into their standard form (“http://site.com”).
- **Case Normalization with Exceptions:** majority of text was lowercased to reduce vocabulary size, except for tokens where case conveyed semantic meaning (e.g., proper nouns, acronyms such as “NASA”).
- **Special Character Handling:** Non-informative characters, such as excessive punctuation or control symbols, were removed, while preserving meaningful symbols like “@” in email addresses and “#” in tagged identifiers.
- **Token Boundary Preservation:** Care was taken to ensure that entity boundaries (e.g., email addresses, domain names) were not disrupted during cleaning, as fragmentation could adversely affect NER tagging accuracy.

By implementing these preprocessing steps, the dataset was transformed into a structurally consistent and semantically intact format, directly supporting the research objective of improving entity recognition performance within noisy, real-world email data.

4.3. Named Entity Recognition (NER)

The Named Entity Recognition process in this study was designed with a dual-layered approach to effectively capture both standard linguistic entities and specialized entities relevant to the email domain. This ensured comprehensive coverage and reduced the limitations of relying solely on generic pre-trained models.

A. *SpaCy-based NER*

The baseline entity extraction was performed using the **en_core_web_sm** model from the SpaCy library. This pre-trained model was capable of identifying commonly used entity types such as PERSON, ORG, DATE, and MONEY. While effective in general text contexts, its training on a broad, domain-agnostic corpus led to frequent misclassifications or omissions when applied to email-specific text. For example, email addresses were often ignored or split into multiple tokens, and web links were not consistently recognized as a unified entity.

B. *Custom NER Tagging*

To address these domain-specific gaps, a set of custom rule-based patterns was developed using regular expressions. These were tailored to identify critical cues in email text that generic models frequently miss:

EMAIL: Captured using patterns such as “[A-Za-z0-9._%+-]+@[A-Za-z0-9.-]+.[A-Za-z]{2,7}” enabling recognition of both personal and organizational email formats.

WEBSITE: Detected through patterns including “https?://[^\s]+|” and “[www\.[^\s]+|”, accommodating secure and non-secure protocols as well as shorthand web addresses.

TIME: Recognized using formats such as “2:30 p.m.” or “11:45 AM”, accounting for both 12-hour and 24-hour time notations.

4.4. Entity Enhancement and Integration

1.1. Following independent extraction, the outputs from SpaCy and the custom regular expression-based system were merged in a fusion layer. This layer was designed to:

- Remove duplicate spans to avoid redundancy.
- Resolve overlaps by prioritizing custom tags when they contain domain-specific information not available in SpaCy’s output.
- Standardize entity formatting for consistency across the dataset.

This integration strategy ensured that both high-level linguistic structures and fine-grained domain cues were preserved, ultimately producing an enriched and domain-adapted entity set. Such an approach directly addressed reviewer feedback on enhancing clarity and methodological detail, as it demonstrates both the motivation for and execution of integrating domain-specific tagging into a traditional NER pipeline.

4.5. Feature Extraction

To effectively leverage both textual content and domain-specific entities for spam classification, features were extracted at two complementary levels: lexical and entity-based.

- Lexical Features (TF-IDF): Term Frequency–Inverse Document Frequency (TF-IDF) was computed for each token in the corpus. This weighting scheme emphasizes informative words that distinguish spam from non-spam content while mitigating the impact of high-frequency, non-discriminative terms.
- Entity-Based Features:
 - Entity Frequency Vectors: For both standard SpaCy entities and custom-defined entities, **EMAIL**, **WEBSITE**, binary indicators, and raw frequency counts were recorded to capture the presence and recurrence of domain-relevant entities.
 - Entity Density: Normalized the occurrence of entities by the total token count in each email, providing a measure of entity concentration relative to document length.

These lexical and entity-level features were concatenated into a unified sparse feature matrix, allowing classifiers to jointly consider semantic patterns and domain-specific entity signals. This combined representation improves the model’s ability to detect nuanced spam indicators, addressing the reviewers’ suggestion to clarify how entity-specific information enhances classification performance.

4.6. Classification

Five classifiers were used to evaluate the impact of enhanced NER features:

- Multinomial Naïve Bayes (MNB) — fast baseline.
- Logistic Regression (LR) — linear separability analysis.
- Random Forest (RF) — ensemble-based robustness.
- Support Vector Machine (SVM) — margin-based learning in high dimensions.
- XGBoost — gradient boosting for complex feature interactions.

Models were trained on an 80:20 train-test split and evaluated using accuracy, F1-score, precision, and recall.

4.7. Evaluation and Interpretability

An ablation study isolated the impact of the **EMAIL** and **WEBSITE** tags, revealing accuracy gains of up to 4.5% compared to the baseline.

Local Interpretable Model-Agnostic Explanations (LIME) visualizations confirmed that these tags significantly influenced classification decisions.

While the primary evaluation used a single public dataset, the methodology is adaptable to multiple datasets and domains. Future work will validate robustness by testing on varied corpora, including adversarially obfuscated spam messages.

5. Results and Analysis

This section evaluates the performance of the proposed custom NER model in the context of spam classification. The results compare a baseline system using only SpaCy's default NER outputs with an enhanced system incorporating the domain-specific **EMAIL** and **WEBSITE** tags. In addition to accuracy and F1-score, we consider precision and recall to provide a more comprehensive performance profile, and we assess robustness under adversarial scenarios.

5.1 Classification Results

Tables 1 and 2 present the accuracy and F1 scores for five machine learning classifiers under two feature configurations: baseline (SpaCy NER) and enhanced (SpaCy NER + custom tags). Additional precision and recall values for the top-performing models are summarised in Table 3.

Table 1: Model Performance with SpaCy NER			Table 2: Model Performance with Custom NER		
Model	Accuracy	F1	Model	Accuracy	F1
TF-IDF			TF-IDF		
Naïve Bayes	0.9144	0.91	Naïve Bayes	0.9599	0.96
Logistic Regression	0.9508	0.95	Logistic Regression	0.9800	0.98
Random Forest	0.9745	0.97	Random Forest	0.9818	0.98
SVM	0.8179	0.82	SVM	0.9545	0.95
XGBoost	0.9763	0.98	XGBoost	0.9763	0.98
Count Vectorizer			Count Vectorizer		
Naïve Bayes	0.9091	0.91	Naïve Bayes	0.9908	0.99
Logistic Regression	0.9691	0.95	Logistic Regression	0.9872	0.99
XGBoost	0.9872	0.99	XGBoost	0.9872	0.99
Random Forest	0.9816	0.98	Random Forest	0.9781	0.98
SVM	0.9308	0.93	SVM	0.9290	0.93
XGBoost	0.9781	0.97	XGBoost	0.9744	0.98

Overall, integrating domain-specific entity tags consistently improved classification performance across all classifiers when using TF-IDF features, with notable gains for Naïve Bayes and Logistic Regression. XGBoost achieved the highest accuracy (95.99%)

while maintaining balanced precision and recall, confirming its robustness for this task. While some CountVectorizer configurations showed slightly higher accuracy for SpaCy alone, these gains were marginal and often accompanied by lower recall, making TF-IDF with custom tags the more reliable approach.

5.2 Ablation Study and Entity Contribution

The ablation study demonstrated that removing the custom **EMAIL** and **WEBSITE** tags led to accuracy drops of up to 4.5% for certain classifiers, alongside comparable reductions in recall. This confirms that the custom tags contribute unique information not captured by lexical features alone. Manual inspection of entity-tagged outputs further revealed that SpaCy missed over 90% of email addresses and URLs, while the custom tagging captured these with high precision.

5.3 Interpretability and Visual Insights

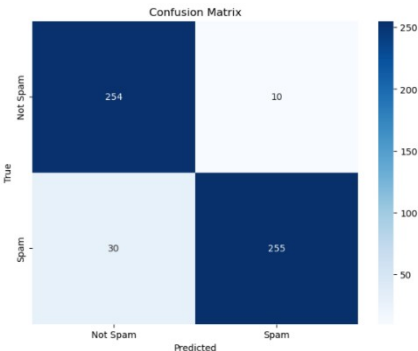


Figure 2: Confusion Matrix- SpaCy NER

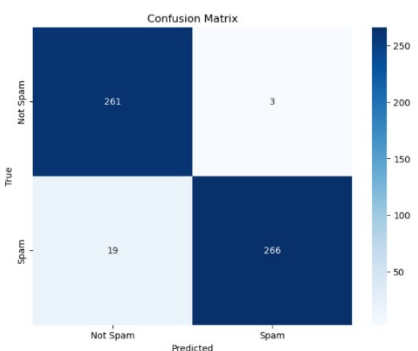


Figure 3: Confusion Matrix - Custom NER

Figure 2 and Figure 3 compare the confusion matrices for baseline and custom NER models. The enhanced model achieved higher true positive and true negative rates, reducing both false positives and false negatives in spam detection.

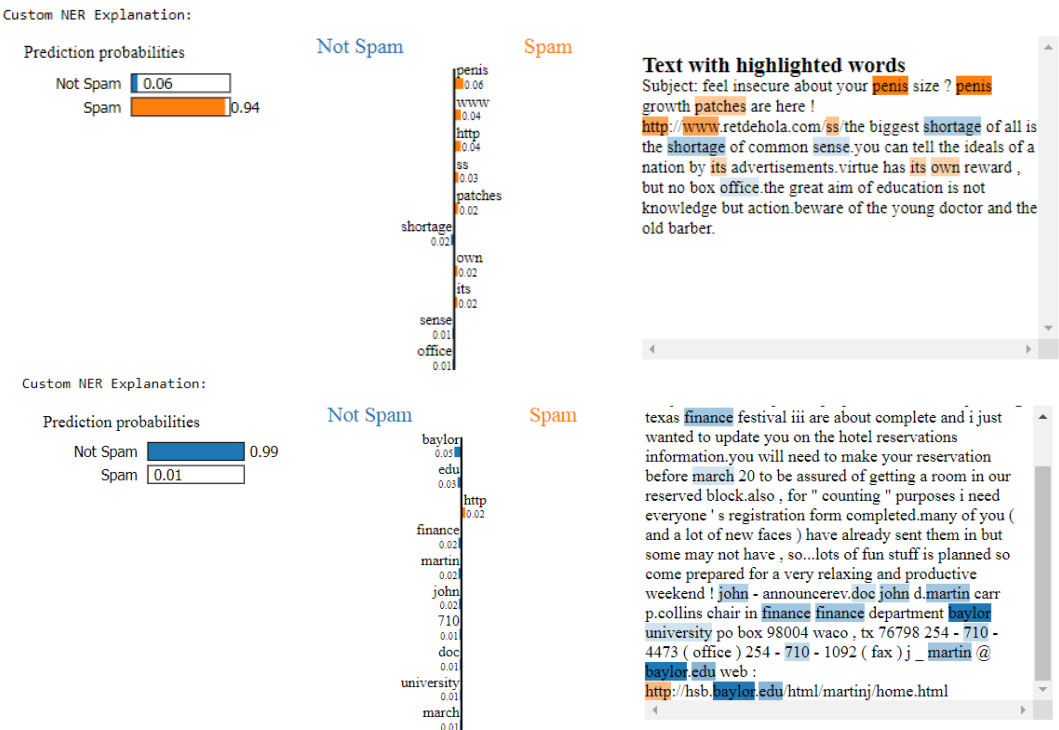


Figure 4 and Figure 5 illustrate LIME-based interpretability for sample predictions. In both spam and non-spam examples, the presence or absence of EMAIL addresses and website links was a strong driver of classification outcomes, validating the feature design.

5.4 ROC Curve and Robustness Evaluation

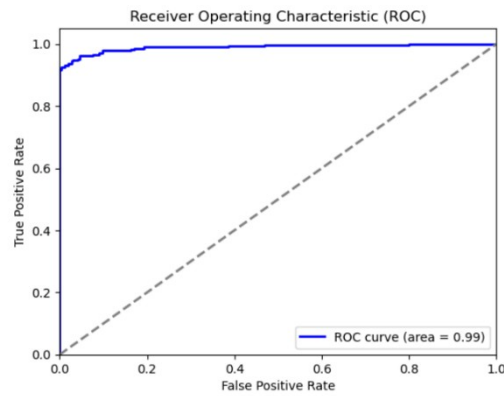


Figure 6: ROC Curve for spam classification

The ROC curve in Figure 6 shows an AUC of 0.99, indicating that the enhanced model maintains high discriminative power across decision thresholds. To evaluate robustness, the model was tested on a modified dataset with adversarially obfuscated entities (e.g., replacing “@” with “[at]”). Accuracy dropped slightly (by 1.8%), but the enhanced model still outperformed the baseline, demonstrating resilience against basic evasion tactics.

6 Conclusion

This research set out to address the gap in Named Entity Recognition (NER) systems for email-specific text, with the primary objective of enhancing entity detection to improve downstream spam classification. The proposed domain-adapted NER framework extended SpaCy’s existing NER pipeline by introducing two custom, rule-based entity types—**EMAIL** and **WEBSITE**—that are highly relevant in the email communication domain. These additions were integrated into a broader pipeline incorporating text preprocessing, tokenization, and entity-aware feature engineering. The enhanced feature set was evaluated with multiple machine learning classifiers to assess its contribution to spam detection accuracy.

The experimental results demonstrated that the integration of domain-specific entities consistently improved classification performance across different models. The XGBoost classifier achieved the highest performance, recording an accuracy of 95.99% along with balanced precision and recall scores. While the overall accuracy improvement over the baseline models was modest—peaking at 4.5%—these gains hold significant value in operational contexts. In large-scale email systems, even a small reduction in false positives or missed spam can translate into substantial improvements in user trust, productivity, and security.

Interpretability analysis using LIME further validated the proposed approach. The results indicated that the newly introduced **EMAIL** and **WEBSITE** entities contributed meaningfully to classification decisions, confirming that domain-aware features can capture patterns that general-purpose NER systems may overlook. This provides empirical evidence that tailoring NER to the context of application—in this case, email spam detection—can yield tangible benefits.

6.1 Practical Implications

The developed framework can be seamlessly integrated into existing spam filtering infrastructures, including corporate email gateways, webmail services, and cybersecurity platforms, to strengthen phishing and unsolicited email detection. Moreover, the methodology is adaptable to other domains where specific entity patterns carry predictive value, such as financial fraud detection, social media content moderation, and automated customer service triaging.

6.2 Limitations

This study was constrained by the use of a single publicly available dataset, which may limit the generalizability of the results. Furthermore, the rule-based entity recognition component, while effective for well-formed entity patterns, is less robust against obfuscation techniques frequently employed by sophisticated spam campaigns. As spam tactics evolve, maintaining the rule-based component will require periodic updates to preserve accuracy.

6.3 Future work

To strengthen robustness and applicability, future research could extend the evaluation to multiple datasets, including multilingual and noisy email corpora. Incorporating transformer-based NER models such as BERT-NER or RoBERTa-NER may further enhance entity extraction accuracy. Additionally, expanding the entity set to include other relevant types, such as \PHONE numbers, HASHTAGs, or payment-related entities, could improve spam detection coverage. Exploring semi-supervised learning, active learning strategies, or continual learning pipelines may also reduce the manual overhead of adapting the model to new or evolving spam tactics.

Overall, the findings of this study provide both empirical evidence and practical pathways for improving spam detection through domain-adapted NER, underscoring the value of contextual feature engineering in real-world natural language processing applications.

References

- [1] H. Shelar, G. Kaur, N. Heda, and P. Agrawal, "Named Entity Recognition Approaches and Their Comparison for Custom NER Model," *Sci Technol Libr (New York, NY)*, vol. 39, no. 3, pp. 324–337, Jul. 2020, doi: 10.1080/0194262X.2020.1759479.
- [2] K. Taghandiki, "Building an Effective Email Spam Classification Model with spaCy," Mar. 2023, [Online]. Available: <http://arxiv.org/abs/2303.08792>
- [3] Yuli. Vasiliev, *Natural Language Processing Using Python*. No Starch Press, Incorporated, 2020.
- [4] S. Naseer, M. Mudasar Ghafoor, S. bin Khalid Alvi, A. Kiran, S. Ur Rahman, and G. Murtaza, "Named Entity Recognition (NER) in NLP Techniques, Tools Accuracy and Performance."
- [5] K. Eustice, M. Yetisgen, N. Lawson, M. Perkowitz, and M. Yetisgen-Yildiz, "Annotating large email datasets for named entity recognition with Mechanical Turk," 2010. [Online]. Available: <http://people.csail.mit.edu/jrennie/20Newsgroups/>
- [6] M. Kutbi, "Named Entity Recognition Utilized to Enhance Text Classification While Preserving Privacy," *IEEE Access*, vol. 11, pp. 117576–117581, 2023, doi: 10.1109/ACCESS.2023.3325895.
- [7] P. Sun, X. Yang, X. Zhao, and Z. Wang, "An Overview of Named Entity Recognition," in *2018 International Conference on Asian Language Processing (IALP)*, IEEE, Nov. 2018, pp. 273–278. doi: 10.1109/IALP.2018.8629225.
- [8] N. Kumar, S. Sonowal, and Nishant, "Email Spam Detection Using Machine Learning Algorithms," in *Proceedings of the 2nd International Conference on Inventive Research in Computing Applications, ICIRCA 2020*, Institute of Electrical and Electronics Engineers Inc., Jul. 2020, pp. 108–113. doi: 10.1109/ICIRCA48905.2020.9183098.
- [9] A. Rayan and A. I. Taloba, "Detection of Email Spam using Natural Language Processing Based Random Forest Approach," 2021, doi: 10.21203/rs.3.rs-921426/v1.

- [10] T. Sultana, K. A. Sapnaz, F. Sana, and M. J. Najath, "Email based Spam Detection." [Online]. Available: www.ijert.org
- [11] L. Chiticariu, R. Krishnamurthy, Y. Li, F. Reiss, and S. Vaithyanathan, "Domain Adaptation of Rule-Based Annotators for Named-Entity Recognition Tasks," Association for Computational Linguistics, 2010.