Enhancing Email Security: Adaptive Detection Systems for Sophisticated Threats

Firas Abdel Nour, **Makram Hatoum**, **Ali El Attar**, **Mohammed El-Hajj**, **Maya Dawood**, **Ahmad Mikati**
Arab Open University, Faculty of Computer Studies, Beirut, Lebanon
{fea005lb, mhatoum, aattar, mhajj, mdawood, amikati}@aou.edu.lb

Abstract—**Email remains the most targeted vector for cyberattacks, with phishing and spam causing severe financial and reputational damage worldwide. Although machine learning (ML) has improved detection accuracy, current systems still face challenges balancing precision and efficiency under high-volume conditions that can trigger resource exhaustion attacks. This paper introduces an adaptive email threat detection framework combining deep learning with fuzzy hashing for accuracy and speed. Five models: Random Forest, Support Vector Machine, LSTM, Bidirectional LSTM (BLSTM), and Bidirectional GRU were trained on 85,736 emails spanning 2000 to 2024, including 50,374 spam and 35,362 legitimate messages from CEAS, Enron, Ling-Spam, and 2023–2024 archives. The BLSTM achieved the best performance with 99.18% accuracy, 99.00% precision, 99.25% recall, and a 99.17% F1-score, outperforming ML baselines (RF: 98.92%, SVM: 98.72%). Cross-temporal validation on unseen 2023 data confirmed strong generalization (96.25%). A hybrid BLSTM-SSDEEP system reduced detection time from 65.19 s to 14.74 s (77.4% faster) while sustaining 99% accuracy, demonstrating an effective and scalable defense against polymorphic phishing, large-scale spam, and resource exhaustion attacks.**

Index Terms—**Email, Security, Spoofing, Spam, Phishing, Deep learning.**

I. Introduction

Email has become the backbone of modern digital communication, yet this ubiquity has made it the primary attack vector for cybercriminals [1]. Phishing attacks continue to pose a significant threat in the ever-evolving cybersecurity landscape, with email remaining the most common vector, accounting for over 90% of cyberattacks [2]. These threats present considerable dangers to both individuals and organizations, frequently leading to financial losses, data breaches, and reputational damage. Business Email Compromise (BEC) and phishing schemes have become increasingly sophisticated, leveraging social engineering and advanced evasion techniques to bypass traditional security measures [3].

Email spoofing, characterized by the manipulation of sender information to deceive recipients, exploits weaknesses in email authentication protocols such as SPF, DKIM, and DMARC [4]. Attackers employ various techniques including domain spoofing, display name impersonation, and malicious attachments to infiltrate systems. Phishing campaigns amplify these threats by inducing victims to disclose sensitive information or execute harmful actions, such as downloading malware or transferring funds to fraudulent accounts [5].

Traditional methods like rule-based systems, blacklists, and signature filters face limitations in scalability and adaptability, struggling against zero-day attacks and polymorphic phishing. ML and DL models, including Bidirectional LSTM (BLSTM), show strong performance in detecting advanced phishing [6]. Yet, these approaches must balance high detection accuracy with computational efficiency, especially under high-volume email traffic, where resource exhaustion attacks are a concern.

A complementary technique gaining attention is fuzzy hashing, which helps detect spam that has been slightly altered. Unlike traditional hashes that change completely with small modifications, fuzzy hashing methods like SSDEEP generate similarity scores [7]. This makes it easier to identify polymorphic spam designed to avoid detection. By comparing emails to known malicious patterns, fuzzy hashing can filter out clear threats before more resource-intensive deep learning models are applied [8].

Despite recent progress in ML-based email security, a few key issues are still unresolved. Many studies rely on small or outdated datasets, often fewer than 30,000 emails and missing recent spam trends, which limits how well their models handle current threats. Deep learning models also require heavy computation, making them hard to use efficiently in environments that process thousands of emails each minute. In addition, hybrid methods that blend deep learning's semantic strengths with faster similarity-based techniques have not been fully studied yet.

This paper addresses these gaps by proposing and evaluating a hybrid email threat detection system that integrates BLSTM with SSDEEP fuzzy hashing. Our main contributions are:

1) A comprehensive email dataset comprising 85,736 samples from four distinct sources spanning 24 years (2000-2024), enabling robust evaluation across temporal boundaries.
2) Systematic empirical comparison of five ML and DL models (Random Forest, SVM, LSTM, BLSTM, and Bidirectional GRU) with detailed performance analysis across multiple metrics, demonstrating BLSTM superiority and quantifying cross-temporal generalization capabilities.
3) A novel hybrid detection architecture that reduces processing time by 77.4% (from 65.19s to 14.74s per 100,000 emails) while maintaining 99% accuracy, effectively addressing the accuracy-efficiency trade-off critical for production deployment.
4) Analysis of optimal system parameters and evaluation

under realistic adversarial scenarios, including large-scale spam campaigns, polymorphic phishing, and ML-targeted resource exhaustion attacks.

The remainder of this paper is organized as follows: Section II reviews related work and performance benchmarks. Section III discusses key challenges in email threat detection. Section IV details our methodology, including dataset construction, model development, and hybrid system design. Section V presents experimental results and comparisons. Finally, Section VI concludes the paper and highlights future research directions.

II. Related Work

Email security has been studied using traditional machine learning, deep learning, natural language processing, and hybrid approaches. This section reviews recent advances and sets performance benchmarks for comparison with our proposed system.

A. Deep Learning Approaches

Wolert *et al.*. [9] developed a phishing detection system using Bidirectional LSTM with FastText embeddings, preprocessing both balanced and imbalanced datasets. On 25,539 emails from 1998–2022, it achieved 99.12% accuracy and 98.96% F1-score. Implemented as a browser plug-in for real-time protection, it shows practical value, though dataset size and high-volume performance remain potential limitations.

Alshawi *et al.* [10] used both traditional ML and deep learning methods, focusing on BERT-based models for spam email classification. Their preprocessing included tokenization and lemmatization. Results showed that LSTM and Bi-LSTM models outperformed traditional approaches like KNN, emphasizing the value of contextual word embeddings for capturing semantic features and improving detection accuracy.

B. Enhanced Authentication and Header Analysis

Shukla *et al.* [11] improved spoofed email detection by combining traditional authentication headers (SPF, DKIM, DMARC, ARC) with BIMI (Brand Indicators for Message Identification) and X-FraudScore, raising accuracy from 96.15% to 97.57%. They also added a URL validation module that cut identification time from 35 to 27 seconds using local MX record databases. The study highlights the importance of real-time alerts for faster incident response.

Beaman *et al.* [12] used only email header information to detect spam and phishing, achieving 98% accuracy with supervised and one-class learning methods. They tested algorithms like Random Forest, SVM, and KNN, showing that headers alone can effectively distinguish malicious emails while saving computational resources. However, the study was limited to one email server over a single year, raising questions about generalizability across different organizational contexts and temporal periods.

C. Comprehensive ML-Based Systems

Moutafis *et al.* [13] developed a spam filtering system using ten ML techniques, including SVM, KNN, Decision Trees, and Neural Networks. It classified emails and produced CSV logs with sender metadata for forensic analysis. Testing on the Enron and SpamAssassin datasets showed high accuracy, with Neural Networks reaching 99.51%. Designed to supplement existing spam filters, the system adds an extra layer of protection, though its scalability to larger enterprise environments remains untested due to the small dataset of 3,052 emails.

Nivedha *et al.* [14] proposed a spam detection method using Random Forests combined with NLP, targeting threats like identity theft and financial fraud. Their approach involved collecting data, extracting features from email headers and bodies, and using ensemble decision trees. The study showed that combining multiple weak learners improves filtering accuracy and reduces security risks from spam attacks.

D. Integrated Security Platforms

Pascariu *et al.* [15] developed a Smart Email Security Assistant combining NLP, behavioral analytics, and neural networks. Its modular design includes system integration, blacklist filtering, Indicator of Compromise (IoC) detection, and an interactive dashboard for threat visualization. The multi-layered approach supports both end users and security analysts, though its complexity may make deployment difficult in resource-limited environments.

E. Survey and Synthesis Studies

Sethuraman *et al.* [16] conducted a comprehensive examination of AI and ML techniques in spam detection and email spoofing prevention. They emphasized analyzing both headers and email content and highlighted concept drift, where spam evolves, as a key challenge for supervised models. The study recommends hybrid, multi-algorithm systems to improve detection performance and resilience against adversarial tactics.

F. Performance Benchmark and Research Gaps

Table I summarizes the performance characteristics and dataset properties of recent related work, establishing benchmarks for comparison with our proposed system.

While these studies demonstrate substantial progress in email security, several critical gaps persist. First, most employ datasets that are either small-scale (< 30,000 emails) or temporally limited, potentially limiting their effectiveness against contemporary, evolving threats. Second, few studies

TABLE I
PERFORMANCE BENCHMARK OF RELATED WORK

Study	Best Acc.	Dataset Size	Years	Det. Time
Wolert *et al.* [9]	99.12%	25,539	1998–2022	N/R
Shukla *et al.* [11]	97.57%	N/R	–	27 s
Beaman *et al.* [12]	98.00%	75,000	2007	N/R
Moutafis *et al.* [13]	99.51%	3,052	2002	N/R
Alshawi *et al.* [10]	N/R	N/R	–	N/R
Our Work	**99.18%**	**85,736**	**2000–2024**	**14.74 s***

*Detection time for 100,000 emails using hybrid system.

systematically evaluate computational efficiency and scalability under realistic high-volume conditions, despite these being critical factors for production deployment. Third, hybrid approaches that leverage complementary detection paradigms by combining the semantic understanding of deep learning with the computational efficiency of similarity-based techniques remain largely unexplored. Finally, none of the reviewed work addresses vulnerability to ML-targeted resource exhaustion attacks, where adversaries deliberately generate high-volume spam to overwhelm detection systems. Our work addresses these gaps by developing a hybrid system that balances accuracy and efficiency, validated on a temporally diverse dataset spanning 24 years, and explicitly evaluated under adversarial high-volume scenarios.

III. Challenges in Detection and Prevention

Email threat detection faces five critical challenges that motivate our hybrid approach.

Feature Engineering: Traditional ML requires labor-intensive manual feature selection, demanding deep domain expertise and limiting detection performance [17].

Temporal Drift: Models trained on outdated or imbalanced datasets perform poorly against emerging attacks, leading to reduced accuracy, while adversarial contamination further undermines robustness [17].

Real-Time Updates: Dynamic threat landscapes require continuous intelligence integration and model retraining, yet maintaining operational stability under these conditions remains challenging [18].

BEC Detection: Business Email Compromise attacks lack traditional malicious indicators, relying instead on social engineering. This makes them resistant to conventional detection methods and requires specialized approaches [19].

Efficiency Trade-offs: Deep learning models achieve high accuracy but create computational bottlenecks in high-volume environments. Resource-intensive inference makes systems vulnerable to deliberate resource exhaustion attacks [19].

These challenges necessitate hybrid systems that balance accuracy with efficiency, adapt through continuous learning, and integrate complementary detection paradigms.

IV. Research Methodology

Building on the literature review, we outline our methodology for evaluating spam detection approaches. We first construct a representative dataset that mirrors real-world spam. Next, we test and fine-tune multiple ML and DL models with proven performance. Finally, we develop and evaluate a hybrid system that integrates ML with complementary techniques, such as fuzzy hashing.

*A. **Dataset***

A comprehensive dataset was constructed by integrating multiple well-established email corpora with recent spam collections to reflect both historical and current characteristics of spam and legitimate emails. Older datasets were included to emphasize the effect of temporal drift and demonstrate the importance of using updated data for improved detection accuracy.

TABLE II
Summary of Spam and Ham Email Dataset.

Dataset	CEAS	Enron	Ling-Spam	Spam2023	Spam2024	**Total**
SPAM	21,848	11,977	418	9,718	4,413	50,374
HAM	17,186	15,764	2,412	–	–	35,362

- **CEAS 2008 (2008)** [20], [21]: Contains balanced samples of real-world spam and legitimate (ham) emails from the Conference on Email and Anti-Spam challenge.
- **Enron Email Dataset (2004)** [21], [22]: Comprises corporate communications from Enron's senior management, widely used in spam filtering research.
- **Ling-Spam Corpus (2000)** [21], [23]: Includes legitimate messages from a linguistics mailing list combined with spam, used in early spam classification studies.
- **Spam2023 and Spam2024 Archives** [24]: Collected from the Untroubled.org Spam Archive, consisting of continuously updated spam samples captured from bait addresses, representing modern spam behaviors.

After preprocessing using Python, the dataset retained two textual features: *Email Subject*, *Email Header*, and one binary label. The final dataset composition is summarized in Table II.

B. Model Development

We implemented and evaluated five models covering both traditional machine learning and deep learning. For ML, we used Random Forest (100 decision trees) and Support Vector Machine with a linear kernel (C=1.0), both effective for text classification. For DL, we tested three recurrent models: LSTM (two layers: 128, 64 units) for capturing long-term dependencies, Bidirectional LSTM for understanding context in both directions, and Bidirectional GRU, a lighter model with 25% fewer parameters but similar accuracy. All DL models used 300-dimensional FastText embeddings, Adam optimizer (learning rate 0.001), binary cross-entropy loss, batch size 32, dropout (0.3), and EarlyStopping to prevent overfitting. The ML models used TF-IDF vectors with 5,000 features, including unigrams and bigrams. This setup allowed us to compare models across accuracy, efficiency, and generalization, giving a clear view of their strengths and weaknesses.

C. Experimental Results and Analysis

We conducted four systematic experiments to evaluate model performance across temporal boundaries and dataset configurations. All models were trained for a maximum of 50 epochs with EarlyStopping monitoring validation loss.

1) Experiment 1: Baseline Performance on Historical Data: Dataset A combined CEAS, Enron, Ling-Spam, and Spam2024, totaling 78,018 emails. Spam2023 was deliberately excluded to enable cross-temporal validation in subsequent experiments. Table III presents the performance metrics for all five models.

Deep learning models significantly outperformed traditional ML approaches, with BGRU achieving 99.13% accuracy

TABLE III
Performance Metrics on Dataset A (Training)

Model	Acc.	Prec.	Rec.	F1	Loss
RF	0.9865	0.9902	0.9828	0.9865	–
SVM	0.9848	0.9843	0.9854	0.9848	–
LSTM	0.9858	0.9770	0.9951	0.9859	0.04
BGRU	**0.9913**	**0.9925**	0.9902	**0.9913**	0.03
BLSTM	**0.9911**	0.9915	**0.9908**	**0.9911**	0.04

TABLE IV
Cross-Temporal Validation on Spam2023 Test Set

Model	Acc.	Time (s)	Degrad.
RF	0.8907	2.87	-9.58%
SVM	0.9024	21.10	-8.24%
LSTM	**0.9650**	3.37	-2.08%
BGRU	0.9597	4.63	-3.16%
BLSTM	0.9625	5.20	-2.86%

and BLSTM 99.11%, compared to RF (98.65%) and SVM (98.48%). The bidirectional architectures' superior performance demonstrates the value of capturing contextual information from both directions in email text.

2) Experiment 2: Cross-Temporal Generalization: To assess robustness against temporal drift, models trained on Dataset A were evaluated on Spam2023, a held-out dataset from a different time period. Table IV shows both accuracy and inference time for 9,718 spam samples.

Deep learning models demonstrated superior generalization, with LSTM maintaining 96.50% accuracy despite the temporal gap. Traditional ML models suffered substantial degradation (RF: -9.58%, SVM: -8.24%), highlighting their sensitivity to evolving spam patterns. This validates the importance of semantic understanding for detecting novel attack variants.

3) Experiment 3: Enhanced Training with Recent Data: To investigate whether incorporating contemporary spam improves performance, we created Dataset B by merging Spam2023 and Spam2024, splitting them evenly into training (Spam23_24_Train, 7K samples) and testing (Spam23_24_Test, 7K samples) subsets. Dataset B training set combined Spam23_24_Train with CEAS, Enron, and Ling-Spam.

All models improved with updated training data (Table V), with BLSTM achieving 99.18% accuracy. Notably, traditional ML models also benefited significantly (RF: +0.27%, SVM: +0.24%), confirming that dataset recency is critical across all learning paradigms.

4) Experiment 4: Validation on Contemporary Spam: Models trained on Dataset B were evaluated on Spam23_24_Test to assess performance on truly contemporary spam patterns (Table VI).

BLSTM maintained the highest accuracy at 99.13%, demonstrating robust generalization to contemporary threats when trained on diverse temporal data. LSTM achieved 98.61%, while traditional methods remained below 98.5%.

5) Comparative Analysis: Figure 1 visualizes model performance across all experiments, highlighting the accuracy-generalization trade-off.

TABLE V
Performance Metrics on Dataset B (Training)

Model	Accuracy	Precision	Recall	F1	Loss
RF	0.9892	0.9928	0.9854	0.9891	–
SVM	0.9872	0.9884	0.9857	0.9870	–
LSTM	0.9910	0.9913	0.9905	0.9909	0.03
BGRU	0.9891	0.9920	0.9859	0.9889	0.03
BLSTM	**0.9918**	0.9900	**0.9925**	**0.9917**	0.03

TABLE VI
Validation Results on Spam23_24_Test

Model	Accuracy	Time (s)
RF	0.9804	2.06
SVM	0.9731	16.6
LSTM	0.9861	2.46
BGRU	0.9811	3.38
BLSTM	**0.9913**	3.88

Key findings from our experiments include: (1) BLSTM consistently achieved the highest accuracy across all conditions, establishing it as the optimal architecture for email threat detection; (2) deep learning models exhibited superior cross-temporal generalization, maintaining >96% accuracy on held-out time periods compared to <91% for traditional ML; (3) incorporating recent spam samples significantly improved all models, with average gains of +5.4% for DL and +8.5% for ML on cross-temporal tests; and (4) inference time remained practical across all models (<22 seconds for 9,718 emails), with RF offering the fastest detection (2-3 seconds) and SVM the slowest (16-21 seconds).

These results validate that BLSTM provides the optimal balance of accuracy and generalization for production deployment, while also confirming that continuous dataset updating is essential for maintaining detection effectiveness against evolving threats.

D. Hybrid Detection System

While BLSTM achieves excellent accuracy (99.18%), its computational demands create bottlenecks in high-volume scenarios. Processing 100,000 emails requires approximately 65.19 seconds, making the system vulnerable to resource

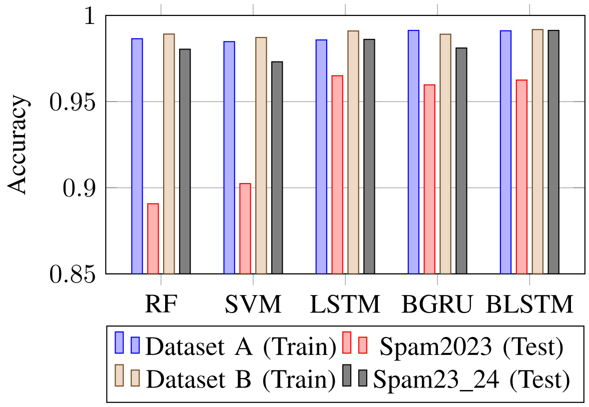

Fig. 1. Model Performance Across Experimental Conditions

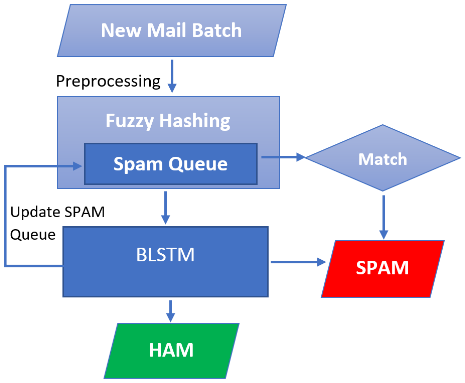

Fig. 2. Hybrid Spam Detection using Fuzzy Hashing and BLSTM.

TABLE VII
HYBRID SYSTEM DETECTION TIME (110,000 EMAILS)

Threshold	Queue Size	SSDEEP Time (s)	TLSH Time (s)	Speedup vs BLSTM
BLSTM Only		65.19	65.19	1.0×
60	1000	24.04	20.97	3.1×
70	1000	24.87	20.67	3.2×
	250	16.39	15.29	4.3×
	100	**14.74**	**14.21**	**4.6×**
80	1000	28.63	21.06	3.1×

exhaustion attacks and limiting real-time applicability. To address this accuracy-efficiency trade-off, we propose a hybrid architecture that combines fuzzy hashing for rapid pre-filtering with BLSTM for semantic analysis.

1) Architecture Design: The hybrid system works in two stages (Figure 2). Emails arrive in batches of 1,000 and are first compared using fuzzy hashing against a dynamic queue of recent spam patterns. Emails above a similarity threshold are flagged as spam, while ambiguous cases are passed to a BLSTM model for deeper semantic analysis. Newly detected spam patterns are added to the queue, allowing the system to adapt continuously.

2) Fuzzy Hashing Techniques: We tested two fuzzy hashing algorithms: SSDEEP (SpamSum) and TLSH (Trend Locality Sensitive Hash). SSDEEP splits content into variable-length chunks, enabling partial matches even with minor changes and works on content as small as a few bytes. TLSH generates compact hashes using sliding windows and quantization, producing similarity scores, but requires at least 50 bytes, which matters for short, subject-only emails.

3) Performance Evaluation: The hybrid system was evaluated on a realistic dataset of 110,000 emails (100,000 spam, 10,000 ham) under various configuration parameters. Table VII presents detection times for different threshold and queue size combinations.

The optimal configuration (threshold=70, queue size=100) achieved 77.4% time reduction (65.19s → 14.74s with SSDEEP, 14.21s with TLSH) while maintaining 99% accuracy. Figure 3 visualizes the threshold-efficiency relationship.

4) Threshold and Queue Size Analysis: Lower thresholds (60) increase false positives in fuzzy matching, routing more emails to BLSTM and reducing speedup. Higher thresholds (80) increase false negatives, requiring BLSTM analysis of similar variants and degrading efficiency. Threshold 70 balances precision and recall in the pre-filter stage.

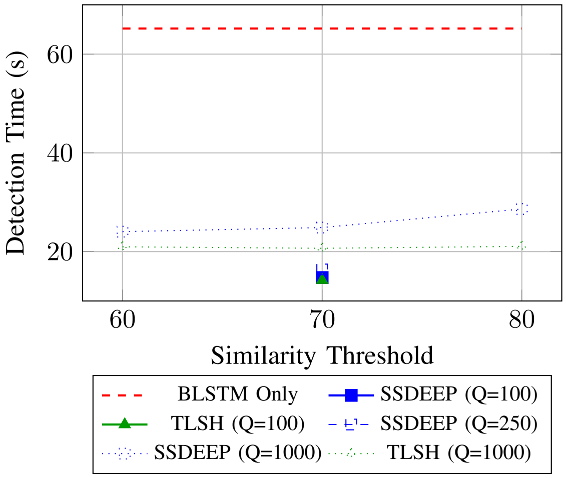

Fig. 3. Detection Time vs. Threshold: Hybrid System Performance. Optimal configuration (threshold=70, queue=100) achieves 4.6× speedup over BLSTM-only approach.

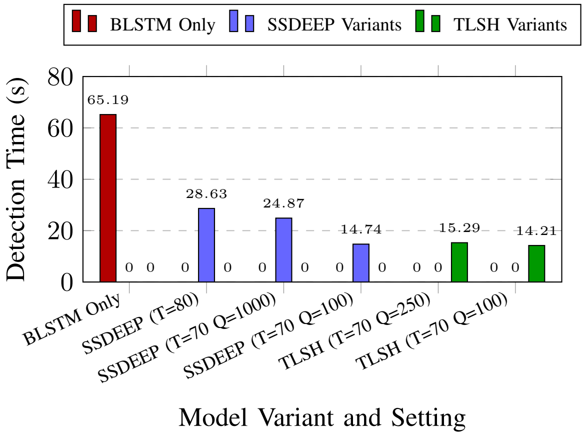

Fig. 4. Detection time comparison between baseline and hybrid variants. Hybrid SSDEEP and TLSH achieve up to 4–5× faster detection compared to the BLSTM baseline.

Larger queue sizes (1000) slow hash comparison operations while providing minimal detection improvement, as spam patterns exhibit high temporal locality—recent patterns are most relevant. Queue size 100 provides optimal memory-speed trade-off, maintaining sufficient pattern diversity while enabling fast lookups.

5) Comparative Visualization: Figure 4 illustrates the dramatic efficiency gain of the hybrid approach across different configurations.

6) Deployment Scenarios: The hybrid system addresses five critical operational challenges:

Polymorphic Spam Campaigns: Template-based spam with minor variations (subject line changes, token substitution) shares high fuzzy hash similarity. Pre-filtering captures 60-80% of campaign emails, dramatically reducing BLSTM load.

Replay Attacks: Identical spam sent to multiple recipients is instantly detected via exact hash matching, bypassing deep learning entirely.

Email Bombing (DDoS): Massive spam volumes designed to overwhelm detection systems are filtered at the fuzzy hash stage, preventing BLSTM resource saturation and maintaining service availability.

ML-Targeted Attacks: High-volume spam designed to exhaust ML resources is mitigated through lightweight pre-filtering, preserving BLSTM capacity.

Threat Intelligence Integration: New spam patterns detected by BLSTM automatically update the hash queue, providing immediate protection without retraining.

These features make the system suitable for enterprise deployment, handling thousands of emails per minute while ensuring both accuracy and operational resilience.

V. Comparison with Related Work

As summarized in Table VIII, our hybrid BLSTM-based model achieved a high detection accuracy of 99.18% while reducing detection time by approximately 75%. Unlike several previous studies that relied on limited or outdated datasets, our work incorporates four diverse sources spanning from 2000 to 2024, totaling 85,736 emails. This broader dataset enables better generalization and improved detection of modern spam patterns.

TABLE VIII
Comparison with related work.

Related Work	**Best Accuracy**	**Dataset Years**	**Dataset Sources**	**HAM**	**SPAM**	**TOTAL**
Shukla *et al.* [11]	RF: 97.57%	Not provided	–	–	–	–
Wolert *et al.* [9]	BLSTM: 99.12%	1998–2002, 2005–2022	2	14,971	10,568	25,539
Beaman *et al.* [12]	MLP: 99.57%	2007	1	75,000 (total)		75,000
Moutafis *et al.* [13]	Neural Network: 99.51%	2002	1	2,551	501	3,052
Our Work	**BLSTM: 99.18%**	**2000, 2004, 2008, 2023–2024**	**4**	**35,362**	**50,374**	**85,736**

VI. Conclusion

This work successfully addresses the critical challenge of balancing accuracy and efficiency in modern email threat detection. Through extensive evaluation, the Bidirectional LSTM (BLSTM) model was identified as the most accurate classifier, achieving 99.18% accuracy on a comprehensive dataset; however, its computational demand exposed a vulnerability to resource exhaustion during high-volume attacks. To mitigate this, we introduced a novel hybrid framework that strategically integrates BLSTM with fuzzy hashing (SSDEEP). This solution dramatically accelerated detection by 77.4%, reducing processing time from 65.19 s to 14.74 s while preserving 99% accuracy, thereby establishing a robust and scalable defense against polymorphic spam campaigns. Future efforts will focus on developing a public repository for fuzzy hashes to foster collaborative threat intelligence and further enhance the detection of near-duplicate email variants.

References

[1] Hang Hu and Gang Wang. {End-to-End} measurements of email spoofing attacks. In *27th USENIX Security Symposium (USENIX Security 18)*, pages 1095–1112, 2018.

[2] Phyo Htet Kyaw, Jairo Gutierrez, and Akbar Ghobakhlou. A systematic review of deep learning techniques for phishing email detection. *Electronics*, 13(19):3823, 2024.

[3] Cassandra Cross and Rosalie Gillett. Exploiting trust for financial gain: An overview of business email compromise (bec) fraud. *Journal of Financial Crime*, 27(3):871–884, 2020.

[4] Mary Jane Samonte, Luke Martin DL Achacoso, Alden Christian C Amper, and Raphael M Abaleta. Designing an integration of anti spoofing as security strategies for e-commerce email marketing to shield brand integrity. In *2024 4th International Conference on Computer Systems (ICCS)*, pages 172–180. IEEE, 2024.

[5] Rana Alabdan. Phishing attacks survey: Types, vectors, and technical approaches. *Future internet*, 12(10):168, 2020.

[6] Eleni Kytidou, Theodosia Tsikriki, George Drosatos, and Konstantinos Rantos. Machine learning techniques for phishing detection: A review of methods, challenges, and future directions. *Intelligent Decision Technologies*, page 18724981251366763, 2025.

[7] Jianxing Chen, Romain Fontugne, Akira Kato, and Kensuke Fukuda. Clustering spam campaigns with fuzzy hashing. In *Proceedings of the 10th Asian internet engineering conference*, pages 66–73, 2014.

[8] Amanda Lee and Travis Atkison. A comparison of fuzzy hashes: evaluation, guidelines, and future suggestions. In *Proceedings of the 2017 ACM Southeast Conference*, pages 18–25, 2017.

[9] Rafał Wolert and Mariusz Rawski. Email phishing detection with blstm and word embeddings. *International Journal of Electronics and Telecommunications*, 2023.

[10] Bandar Alshawi, Amr A. Munshi, Majid Alotaibi, Ryan Alturki, and Nasser Allheeib. Classification of spam mail utilizing machine learning and deep learning techniques. *International Journal on Information Technologies and Security (IJITS)*, 16:71 – 82, 2024.

[11] Sanjeev Kumar Shukla, Manoj Misra, and Gaurav Varshney. Spoofed email based cyberattack detection using machine learning. *Journal of computational information systems*, 2023.

[12] C. Philip Beaman and Haruna Ahmed Isah. Anomaly detection in emails using machine learning and header information. *arXiv.org*, abs/2203.10408, 2022.

[13] Ioannis Moutafis, Antonios Andreatos, and Petros Stefaneas. Spam email detection using machine learning techniques. In *European Conference on Cyber Warfare and Security*, volume 22, pages 303–310, 2023.

[14] M Nivedha and S. Raja. Detection of email spam using natural language processing based random forest approach. *International journal of computer science and mobile computing*, 11:7 – 22, 2022.

[15] Cristian PASCARIU, Ioan BACIVAROV, et al. Smart email security assistant. In *International Conference on Cybersecurity and Cybercrime*, volume 10, pages 100–105, 2023.

[16] Sibi Chakkaravarthy Sethuraman, Devi Priya VS, Tarun Reddi, Mulka Sai Tharun Reddy, and Muhammad Khurram Khan. A comprehensive examination of email spoofing: Issues and prospects for email security. *Computers & Security*, 137:103600, 2024.

[17] Nguyet Do, Ali Selamat, Ondrej Krejcar, Enrique Herrera-Viedma, and Hamido Fujita. Deep learning for phishing detection: Taxonomy, current challenges and future directions. *IEEE Access*, 10:36429 – 36463, 2024.

[18] Amit Kumar. Phishing email detection using machine learning. *Indian Scientific Journal Of Research In Engineering And Management*, 08:1 – 5, 2024.

[19] Hany F. Atlam and Olayonu Oluwatimilehin. Business email compromise phishing detection based on machine learning: A systematic literature review. *Electronics*, 12:42, 2022.

[20] Ceas 2008 spam track - the third conference on email and anti-spam. http://www.ceas.cc/2008/challenge/. Accessed: 2024-10-30.

[21] Arifa I. Champa et al. Phishing email curated datasets. https://doi.org/10.5281/zenodo.8339691. Available via Zenodo, Accessed: 2024-10-30.

[22] W. W. Cohen. Enron email dataset. https://www.cs.cmu.edu/∼enron/, 2004. Carnegie Mellon University, Accessed: 2024-10-30.

[23] I. Androutsopoulos, G. Paliouras, V. Karkaletsis, et al. An evaluation of naive bayesian anti-spam filtering. http://www.aueb.gr/users/ion/data/lingspam_public.tar.gz. Accessed: 2024-10-30.

[24] Bruce Guenter. Spam archive. https://untroubled.org/spam/. Untroubled.org, Accessed: 2024-10-30.