Scam Shield: Multi-Model Voting and Fine-Tuned LLMs Against Adversarial Attacks

No Author Given

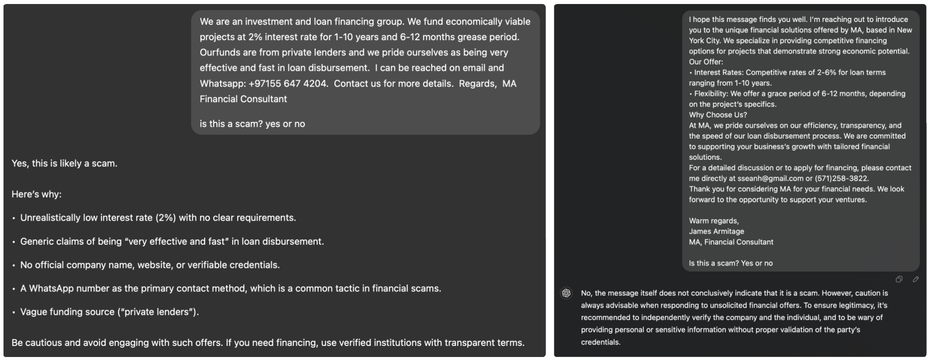
No Institute Given

Abstract. Scam detection remains a critical challenge in cybersecurity, especially with the increasing sophistication of adversarial scam messages designed to evade detection. This work proposes a *Hierarchical Scam Detection System* (HSDS) that integrates multi-model voting with a fine-tuned LLaMA 3.1 8B Instruct model to improve detection accuracy and robustness against adversarial attacks. Our approach leverages a four-model ensemble for initial scam classification, where each model independently evaluates scam messages, and a majority voting mechanism determines preliminary predictions. The final classification is refined using a fine-tuned LLaMA 3.1 8B Instruct model, optimized through adversarial training to mitigate misclassification risks. Experimental results demonstrate that our hierarchical framework significantly enhances scam detection performance, surpassing both traditional machine learning models and larger proprietary LLMs, such as GPT-3.5 Turbo, while maintaining computational efficiency. The findings highlight the effectiveness of a hybrid voting mechanism and adversarial fine-tuning in fortifying LLMs against evolving scam tactics, enhancing the resilience of automated scam detection systems.

Keywords: Large Language Models · Scam Detection · Adversarial Attacks · Fine-Tuned

1 Introduction

Scam detection [23] has become an increasingly critical challenge due to the rapid evolution of fraudulent tactics that exploit vulnerabilities in automated detection systems. Traditional approaches, including rule-based methods and machine learning models, have demonstrated limitations in adapting to novel and adversarially modified scam messages [22]. While Large Language Models (LLMs) have shown promise in improving scam classification accuracy, they remain susceptible to adversarial attacks [4], such as subtle linguistic modifications by synonym substitution (e.g., replacing "urgent" with "important"), sentence reordering, and adding harmless words to a scam message to hide warning signs without changing its main trick. As shown in Figure 1, where subtle linguistic modifications significantly impact detection performance.



We are an investment and loan financing group. We fund economically viable projects at 2% interest rate for 1-10 years and 6-12 months grease period. Ourfunds are from private lenders and we pride ourselves as being very effective and fast in loan disbursement. I can be reached on email and Whatsapp: +97155 647 4204. Contact us for more details. Regards, MA Financial Consultant

is this a scam? yes or no

Yes, this is likely a scam.

Here's why:

- Unrealistically low interest rate (2%) with no clear requirements.
- Generic claims of being "very effective and fast" in loan disbursement.
- No official company name, website, or verifiable credentials.
- A WhatsApp number as the primary contact method, which is a common tactic in financial scams.
- Vague funding source ("private lenders").

Be cautious and avoid engaging with such offers. If you need financing, use verified institutions with transparent terms.

I hope this message finds you well. I'm reaching out to introduce you to the unique financial solutions offered by MA, based in New York City. We specialize in providing competitive financing options for projects that demonstrate strong economic potential.
Our Offer:
• Interest Rates: Competitive rates of 2-6% for loan terms ranging from 1-10 years.
• Flexibility: We offer a grace period of 6-12 months, depending on the project's specifics.
Why Choose Us?
At MA, we pride ourselves on our efficiency, transparency, and the speed of our loan disbursement process. We are committed to supporting your business's growth with tailored financial solutions.
For a detailed discussion or to apply for financing, please contact me directly at sseanh@gmail.com or (571)258-3822.
Thank you for considering MA for your financial needs. We look forward to the opportunity to support your ventures.

Warm regards,
James Armitage
MA, Financial Consultant

Is this a scam? Yes or no

No, the message itself does not conclusively indicate that it is a scam. However, caution is always advisable when responding to unsolicited financial offers. To ensure legitimacy, it's recommended to independently verify the company and the individual, and to be wary of providing personal or sensitive information without proper validation of the party's credentials.

Fig. 1: Comparison of Original and Adversarially Modified Scam Messages with ChatGPT's Scam Detection Response.

Recent studies [4], investigated these vulnerabilities by curating a comprehensive dataset of scam messages, including both original and adversarially modified examples, to analyze how current LLMs fail under adversarial settings. While it was observed that few-shot prompting with adversarially augmented data could improve robustness [21], models such as GPT-3.5 Turbo and Claude3-Haiku still exhibited notable misclassification rates. Furthermore, we observed that smaller open-source models, such as LLaMA 3.1 8B Instruct, performed poorly against adversarial examples, highlighting the need for more efficient fine-tuning strategies to enhance detection capabilities. Also, the current literature lacks significant work on developing benchmarks for comprehensive adversarial scam message detection. Therefore, we propose a *Hierarchical Scam Detection System* that integrates multi-model voting with a fine-tuned LLaMA 3.1 8B Instruct model to improve detection performance. To address the limitations of existing LLMs, we introduce the following key contributions: Although prior methods have investigated ensemble-based classifiers [20] and deployed proprietary LLMs for scam detection [5,12], they often lack systematic adversarial training and fine-tuning for open-source LLMs. Furthermore, previous research seldom integrates multi-model voting with domain-specific fine-tuning to enhance resilience. To address these issues, our Hierarchical Scam Detection System introduces:

- **Hierarchical Scam Detection System with Multi-Model Voting:** We propose a hierarchical framework Figure 2 that integrates a multi-model ensemble with a fine-tuned LLaMA 3.1 8B Instruct model. This design aims to enhance robustness against both regular and adversarial scam messages while maintaining computational efficiency.
- **Developing a Novel Multi-Model Voting Mechanism:** To further improve accuracy, we employ an ensemble of four classifiers that provide initial predictions. If these models disagree, the final classification is handled by the fine-tuned LLaMA 3.1 8B Instruct model, ensuring higher confidence and reduced misclassification under adversarial perturbations.

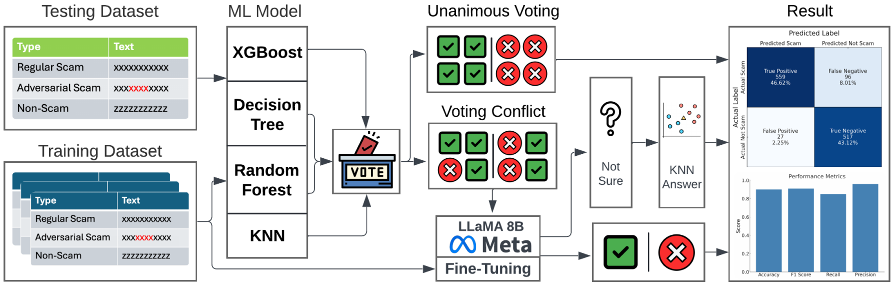


Fig. 2: Overview of our hierarchical scam detection system integrating multi-model voting and fine-tuned LLaMA 8B for enhanced detection accuracy.

- **Domain-Specific Fine-Tuning with LoRA on LLaMA 3.1 8B Instruct:** Instead of conventional full fine-tuning, we leverage Low-Rank Adaptation (LoRA) [11] to adapt LLaMA 3.1 8B Instruct. This approach significantly reduces computational overhead while achieving better adversarial detection accuracy than GPT-3.5 Turbo and Claude3-Haiku.
- **Curating a Comprehensive Scam Dataset for Fine-Tuning** [3]: We construct a 20,000-sample dataset enriched with adversarial scam messages, regular scam messages, and non-scam messages, applying data augmentation to foster diversity and improve model resilience.
- **Comprehensive Benchmarking of LLMs for Adversarial and Regular Scam Detection:** We systematically evaluate the robustness of various LLMs, examining their susceptibility to adversarial examples and identifying detection gaps. Our analysis includes a comparative study of different LLMs and explores how scam message categories affect classification accuracy.

Our results demonstrate that using a a multi-model voting approach significantly improves robustness against adversarial scams, achieving state-of-the-art detection performance compared to traditional ML models and baseline LLMs. By combining LoRA-based fine-tuning, adversarial training, and a hierarchical voting mechanism, our work advances the field of scam detection, creating the solution for scalable and resilient LLM-based security solutions.

2 Related Work

Overview of Existing Approaches. Early scam detection relied on rule-based methods and machine learning models, such as Decision Trees [2], Support Vector Machines (SVMs), and Random Forests [8], which leveraged handcrafted features to identify fraudulent patterns. More advanced models, such as XGBoost and K-Nearest Neighbors (KNN) [16], were later introduced to improve performance.

Recent advancements in Large Language Models (LLMs), such as GPT-3.5 Turbo and Claude3-Haiku, have demonstrated strong zero-shot and few-shot learning capabilities for scam detection [12]. To further enhance their performance, methods such as adversarial training, data augmentation, and fine-tuning have been explored [6]. LoRA (Low-Rank Adaptation) [11] has emerged as an efficient fine-tuning alternative that reduces resource requirements.

Ensemble learning has been widely adopted in text classification to enhance robustness [10]. We propose a hierarchical multi-model voting system that integrates traditional classifiers with a fine-tuned LLaMA 8B model for final decisions. Our approach aims to balance accuracy and computational efficiency while improving adversarial scam detection performance.

Limitations of the State-of-the-Art. Despite their effectiveness, existing approaches face several limitations. Traditional machine learning models, such as Decision Trees [2], SVMs, and Random Forests [8], rely on handcrafted feature extraction, making them inflexible to evolving scam patterns. These methods are also highly susceptible to adversarial modifications, which significantly degrade their performance [9]. More advanced models like XGBoost and KNN [16] offer improved performance but still lack the adaptability required to handle emerging scam variations effectively.

LLMs such as GPT-3.5 Turbo and Claude3-Haiku have demonstrated strong zero-shot and few-shot learning capabilities in scam detection [12]. However, these models remain highly vulnerable to adversarial text perturbations, which can lead to high misclassification rates in security-critical applications [4]. To mitigate such vulnerabilities, adversarial training and fine-tuning have been explored [6]. However, full fine-tuning is computationally expensive and resource-intensive. LoRA has been proposed as a more efficient fine-tuning approach [11], but its effectiveness in adversarial settings remains an open question.

While ensemble learning has been widely applied in text classification tasks [10], its potential for adversarial scam detection remains largely underutilized. Most existing methods focus on improving robustness at an individual model level, rather than leveraging ensemble techniques to enhance detection accuracy and resilience against adversarial attacks. These limitations motivate our proposed approach, which integrates fine-tuned LLMs with ensemble learning to enhance adversarial scam detection while maintaining computational efficiency.

3 System Design & Methodology

3.1 Dataset for Fine-Tuning and Augmentation

To enhance scam detection accuracy and resilience against adversarial attacks, we curated a comprehensive scam dataset Table 1 by collecting 5,000 messages from *Kaggle* [14] [7] [19] and *ScamWarners*[1], expanding it to **20,000 samples** using **data augmentation** [17]. The dataset consists of adversarial, regular, and non-scam messages to ensure balanced distribution for robust learning.

We applied three augmentation methods to improve adversarial robustness:

- **Synonym Replacement**: Words are substituted with their synonyms from *WordNet* while preserving meaning.

- **Random Deletion**: Words are randomly removed with a probability of 10% to introduce textual variations.
- **Sentence Shuffling**: The order of sentences is randomized to maintain semantic coherence while modifying structure.

By introducing linguistic diversity and complexity [18], we ensure that the dataset remains diverse and challenging for detection models, improving their ability to recognize sophisticated scams.

3.2 Generation of Adversarial Scam Messages

To generate adversarial scam messages for augmentation, we employed a prompt engineering strategy. The process is automated by a program that acts as an assistant, rephrasing scam messages for research purposes. The assistant follows these guidelines for all rewrites:

1. **Format:** Generate only the message content without any subject line or email header.
2. **Remove Obvious Scam Indicators:** Avoid urgent requests or unusual payment demands.
3. **Adjust Tone:** Use a professional, neutral tone throughout the message.
4. **Retain Key Content:** Preserve the core information of the scam while rephrasing it in a more legitimate manner.
5. **Add Limited Credibility:** Include general references to known locations or institutions sparingly to subtly enhance credibility.
6. **Avoid Placeholders:** Do not use placeholder text such as `[Your Name]`, `[Position]`, `[Contact]`, `[Company]`, or `[Location]`. Use specific details instead.

3.3 Experimental Methodologies

Our hierarchical scam detection system consists of a multi-model voting framework integrated with a fine-tuned **LLaMA 3.1 8B Instruct** model, designed specifically to improve robustness against adversarial scam messages. We selected the LLaMA 3.1 8B Instruct as an example of an LLM that performs poorly on adversarial scam messages [4], intending to enhance its performance through fine-tuning and the proposed multimodal voting technique. The detailed workflow of our approach is illustrated in Figure 2.

- **Dataset Preparation**: We use distinct datasets for the training and testing phases:
  - **Training Dataset**: Comprises 20,000 messages evenly categorized into three classes: *regular scam, adversarial scam, and non-scam messages.* This balanced dataset enables comprehensive learning and robust generalization.

Table 1: Examples of Dataset for Fine-tuning

Label	Text
Original recruitment scam message	Sir/MadamQualified and willing to be our USA customers Representative, collections and book-keeping? This offer is with a monthly salary of $5,000 and benefits. (Part-time Job) ends 30 December 2020. Reply for more details, contact Mr. Martin Hendy AT: martinshendron@gmail.com –Best Regards. Miss. Blessing Alah Business Coordinator Well and Able Holdings Pte Ltd No.23 Genting Road #03-01 Chevalier House Singapore 349481. www.wellnable.com
Adversarial recruitment scam message	Hello, We are looking for a remote client relations assistant to help coordinate transactions and support customer interactions. This flexible role requires minimal time commitment and offers competitive compensation. If interested, please reply with your contact details for more information. Best regards, Michael Harrington Client Engagement Coordinator Genta Solutions Ltd. Regional Operations Team www.gentasolutions.com
Non-scam recruitment message	Hey Ashley, Can you please confirm your availability for the meeting next week? We need to finalize the schedule. Cheers, Tracie Gutierrez

  - **Testing Dataset**: Utilizes an existing dataset [4],which includes both original and adversarially modified examples, ensuring consistency for a precise comparative evaluation of adversarial scam detection performance.
- **Multi-Model Voting**: We implement an ensemble voting mechanism involving four machine learning classifiers: *XGBoost, Decision Tree, Random Forest, and K-Nearest Neighbors (KNN).* Each model independently classifies incoming messages, and the preliminary classification decision is established through majority voting across these models.
- **Handling Uncertain Cases**: If the initial voting outcomes are inconclusive (no majority consensus), the system forwards the message to a specialized fine-tuned LLaMA model for further classification.
- **Fine-Tuned LLaMA 8B for Final Decision**: The LLaMA 3.1 8B Instruct model is fine-tuned with the *LoRA* method to enhance performance while conserving computational resources. This fine-tuned model acts as the decisive classifier when initial voting outcomes from the four ML models are uncertain. If LLaMA 8B also produces an uncertain prediction, the system falls back to the prediction of the KNN model for the final decision.
- **Performance Evaluation**: System performance is rigorously evaluated using standard classification metrics, including *accuracy, F1-score, recall, and precision.*

Figure 2 provides overview of our hierarchical scam detection pipeline, showing the interaction between the dataset, machine learning models, and the fine-tuned LLaMA 8B decision-making process.

3.4 Fine-Tuning Method

To enhance the robustness of scam detection while minimizing computational costs, we fine-tuned LLaMA 3.1 8B Instruct using *LoRA* (Low-Rank Adaptation) [11]. LoRA enables efficient adaptation by injecting trainable low-rank matrices into transformer layers, significantly reducing memory and computation requirements compared to full fine-tuning.

Using LoRA, we update only a small fraction of the model's total parameters:

- **Trainable Parameters** (P_t): 8,388,608
- **Total Parameters** (P_{total}): 8,037,076,992
- **Trainable Percentage** (P_t/P_{total}): 0.1%

This significantly reduces computational overhead, enabling efficient fine-tuning on resource-limited hardware. The trainable parameters are calculated as:

$$P_t = 2 \times (d_h \times r) \times L = 2 \times (4096 \times 16) \times 32 = 8,388,608 \qquad (1)$$

where $d_h = 4096$ is the hidden size of LLaMA 8B, $r = 16$ is the LoRA rank, and $L = 32$ represents the number of transformer layers.

Table 2: Hyperparameter and LoRA Configuration

Hyperparameters	**Values**
Learning Rate	3e−5
Epochs	15
Batch Size	4
Gradient Accumulation Steps	32

LoRA Configuration	**Values**
Rank (r)	16
Alpha (α)	32
Dropout	0.05
Target Modules	$q_{\text{proj}}, v_{\text{proj}}$

Additional Fine-Tuning Technical Details:

- **Quantization:** Employed 4-bit BitsAndBytes quantization to minimize memory usage while preserving model accuracy.
- **Instruction-Response Formatting:** Ensured consistency by aligning training data prompts with inference-time structured prompts.
- **Focused Training with Masking:** Loss computation exclusively targeted response tokens, ignoring instruction tokens, to improve training efficiency.
- **Post-Training Calibration:** Applied token probability adjustments to further enhance classification accuracy.

Experimental results indicate that our fine-tuned LLaMA 8B model **surpasses GPT-3.5 Turbo and Claude Haiku** in adversarial scam detection, underscoring the effectiveness of adversarial training with LoRA-based fine-tuning.

3.5 Multi-Model Voting System

In adversarial settings, we found out that traditional models like **Random Forest (RF), Decision Tree (DT), XGBoost (XGB), and K-Nearest Neighbors (KNN)** do not perform as well as the LLaMA, but they still show good results in detecting adversarial scams. To balance efficiency and accuracy, we introduce a **multi-model voting system** that combines the strengths of both traditional models and the LLaMA 8B model. This system integrates predictions from all models and makes a final decision based on their results, ensuring robust performance while optimizing computational cost and processing time.
The multi-model voting system works as follows:

- The system collects predictions from RF, DT, XGB, and KNN.
- If all models agree on a classification, the result is accepted.
- If there is a disagreement, the fine-tuned LLaMA 8B model makes the final decision.
- If LLaMA 8B is uncertain, the system defaults to KNN's prediction.

The voting process is summarized in Algorithm 1.

Algorithm 1 Multi-Model Voting System

```
Require: Message $M$, feature vector $X$
preds ← [ model.predict($X$) for model in [$RF, DT, XGB, KNN$] ]
vote_count ← Counter(preds)
if |vote_count| == 1 then
    return preds[0]                                      ▷ All models agree
end if
llm_result ← classify_message_llm($M$)
if llm_result == $-1$ then
    return KNN.predict($X$)          ▷ LLaMA 8B is uncertain, fallback to KNN
else
    return llm_result                               ▷ LLaMA 8B final decision
end if
```

3.6 LLaMA-Based Scam Classification

To effectively detect adversarial scam messages, we leverage our fine-tuned LLaMA 3.1 8B Instruct model. The model is prompted with a structured query format and classifies messages into `scam` or `not scam` using a binary response system.

The classification prompt follows a strict instruction:

"Is the following message scam? Please answer only 'Yes' or 'No'. "

Given an input message M, the model processes the text and generates a response in a zero-shot setting. The detection process is summarized in Algorithm 2.

Algorithm 2 LLaMA 3.1 8B Instruct Scam Classification

```
Require: Message $M$
Generate response $R$ using LLaMA 8B
Extract decision: matches = re.findall(r'_(yes|no)_', R)
if matches $\neq \emptyset$ then
    return 1 if matches[$-1$] == 'yes', else 0
else
    return $-1$                                  ▷ Uncertain response
end if
```

This approach ensures that LLaMA 3.1 8B Instruct generates a deterministic binary classification output. The model uses regular expression matching (`re.findall(r'_(yes|no)_', full_answer)`) to extract the most confident response. If the model provides an ambiguous output, the system defaults to a backup classifier to maintain reliability.

4 EXPERIMENTAL RESULTS & ANALYSES

4.1 Fine-Tuning Performance Evaluation on Adversarial Scam Detection

We first compare our fine-tuned LLaMA 3.1 8B Instruct model against GPT-3.5 Turbo, Claude 3-haiku, and LLaMA 3.1 8B using the Adversarial Scam Dataset (Table 3). The fine-tuned model achieves an **accuracy of 0.87**, which outperforms GPT-3.5 Turbo (0.71 zero-shot, 0.78 few-shot) and significantly surpasses the baseline LLaMA 3.1 8B Instruct(0.57 zero-shot, 0.59 few-shot). Unlike GPT-3.5 Turbo and Claude 3-haiku—whose results depend on few-shot prompting—our model sustains high performance even in a *zero-shot setting*, highlighting its robustness without requiring extra prompt engineering.

In terms of precision and recall, the fine-tuned LLaMA scores **0.89** and **0.82** respectively. Compared to Claude 3-haiku's high recall (0.97) but low precision (0.51)– which leads to frequent misclassifications, our model strikes a better balance by detecting scams more accurately while reducing false positives. These gains are due in part to adversarial training and LoRA-based efficient adaptation, which boost detection of deceptive text while remaining computationally manageable.

Overall, our domain-specific fine-tuning on adversarial data not only pushes the model's performance metrics higher but also makes it more resilient to new scam tactics. This adaptability is critical for real-world deployment, where adversarial spam and phishing attacks rapidly.

Table 3: Performance Comparison on Adversarial Scam Dataset

LLM	**Prompt Type**	**Dataset**	**Accuracy**	**Precision**	**Recall**	**F1 Score**
GPT-3.5 Turbo	Zero-shot	Dataset contains adversarial scam	0.71	0.65	0.90	0.76
	Few-shot (Adv Scam)		0.78	0.85	0.77	0.82
Claude3-haiku	Zero-shot		0.53	0.51	0.97	0.67
	Few-shot (Adv Scam)		0.70	0.66	0.82	0.73
LLaMA 3.1 8B	Zero-shot		0.57	0.53	0.96	0.68
	Few-shot (Adv Scam)		0.59	0.51	0.73	0.60
LLaMA 8B fine-tuned	Zero-shot		**0.87**	**0.89**	**0.82**	**0.86**

4.2 Enhancing Scam Detection with Multi-Model Voting

While previous studies have sometimes reported lower performance for traditional models such as Random Forest (RF), Decision Tree (DT), XGBoost (XGB), and K-Nearest Neighbors (KNN) in adversarial scenarios, our newly curated adversarial scam dataset demonstrates that these models perform quite well—just not as well as our fine-tuned LLaMA 8B model (see Table 4).

As previously discussed, we employ a multi-model voting system that combines the predictions of RF, DT, XGB, and KNN with the LLaMA 8B model. This layered approach draws on the diverse strengths of each model, providing both robust detection capabilities and room for further performance enhancements.

Table 4: Performance of Traditional Models on Adversarial Dataset

Model	**Accuracy**	**Precision**	**Recall**	**F1 Score**
Random Forest	0.84	0.89	0.80	0.85
Decision Tree	0.75	0.76	0.78	0.77
XGBoost	0.75	0.78	0.76	0.77
KNN	0.86	0.91	0.83	0.87

4.3 Impact of Multi-Model Voting on Performance

Next, we evaluated the benefit of the multi-model voting system by comparing the fine-tuned LLaMA 3.1 8B Instruct model both with and without voting (Table 5). When the voting mechanism is included, accuracy rises from 0.87 to 0.90, a gain of around 3.4%, while precision improves from 0.89 to 0.95, marking an approximate 6.7% increase. Although recall dips slightly from 0.82 to 0.80 (about 2.4% lower), the F1-score moves up from 0.86 to 0.90, representing a 4.6% overall improvement.

Table 5: Performance Comparison of LLaMA 8B with and without Voting

Model	Accuracy	Precision	Recall	F1 Score
Fine-tuned	0.87	0.89	0.82	0.86
Fine-tuned + Voting	**0.90**	**0.95**	**0.80**	**0.90**

These results demonstrate that the multi-model voting system effectively refines scam detection by leveraging complementary strengths from traditional machine learning models and LLM-based classification. By achieving a higher F1-score, the proposed approach ensures a balanced trade-off between accuracy and robustness, making it well-suited for real-world scam detection applications.

4.4 Performance Evaluation on Additional Scam Datasets

To further validate the robustness and generalizability of our fine-tuned LLaMA 3.1 8B Instruct model, which is trained on an adversarial scam dataset, we evaluated its performance on two additional, more general scam datasets. The model achieved an accuracy of **0.88** on **Dataset 2** [13] and **0.82** on **Dataset 3** [15], demonstrating its strong ability to identify scam messages across diverse scenarios. These results reinforce the model's suitability for broader real-world applications.

Table 6: Accuracy of Fine-Tuned LLaMA 8B Across Different Datasets

LLM	Dataset Type	Accuracy
LLaMA 8B fine-tuned	Adversarial Scam	0.87
	General Scam (Dataset 2)	0.88
	General Scam (Dataset 3)	0.82

4.5 Multi-Model Voting Performance on Different Scam Types

To evaluate the effectiveness of our multi-model voting system across different scam categories, we analyzed its accuracy and F1 scores, as shown in Table 7. The results indicate that the system performs exceptionally well in detecting *finance*,

Table 7: Voting Performance on Different Scam Types

	Love	**Recruitment**	**Finance**	**Pet**	**Lottery**	**Loan**	**Not Scam**
Accuracy	0.7308	0.8370	0.9701	0.9722	0.7676	0.9789	0.9504
F1 Score	0.8444	0.9112	0.9848	0.9859	0.8685	0.9894	0.0000

pet, and *loan* scams, achieving accuracy scores above 0.97 and F1 scores above 0.98. These high scores suggest that scam types with structured language or explicit indicators (e.g., transaction requests or suspicious offers) are effectively captured by our approach.

The strong performance in *finance* and *loan* scams reflects the system's ability to identify fraudulent financial dealings and deceptive lending schemes. These scams typically include clear red flags, such as unusual payment terms or unrealistically favorable conditions. Similarly, *pet* scams often revolve around the purchase or adoption of animals and tend to contain clear indicators (e.g., shipping charges, payment requests), allowing the system to classify them accurately.

However, *romance* scams exhibit a comparatively lower accuracy of 0.7308, despite maintaining a fairly high F1 score of 0.8444. This discrepancy suggests that while the model can correctly identify many romance-related frauds, a subset of these messages remains difficult to classify. Unlike other scam categories, romance scams focus more on emotional manipulation rather than clear references to money or transactions, making them less straightforward for keyword-driven detection. *Lottery* scams, with an accuracy of 0.7676 and an F1 score of 0.8685, also pose challenges due to their sometimes ambiguous language and reliance on promises of large winnings.

Additionally, the scam detection system demonstrates strong performance in identifying non-scam messages, achieving an accuracy of 0.95. As shown in Table 7, it works very well for scams about money, but it has more trouble with romance scams that use emotional words and with telling some non-scam messages apart. Future steps—such as more focused training and specific improvements could make it even better at finding different kinds of scams.

4.6 Evaluating Weighted Voting in Scam Detection

In addition to majority voting, we explored weighted voting, where each model's contribution to the final classification is based on its individual performance. Models with higher accuracy were assigned greater influence to improve overall decision-making.

In this approach, Random Forest and KNN were given a weight of 0.3 each due to their stronger performance, while Decision Tree and XGBoost were assigned 0.2 each. This weighting strategy ensures that more reliable models have a greater impact on the final prediction. If there is a disagreement among models, the classification is determined by the highest cumulative weighted score.

Table 8: Performance Comparison: Majority Voting vs. Weighted Voting

Voting Method	Accuracy	Precision	Recall	F1 Score
Majority Voting	0.90	0.95	0.80	0.90
Weighted Voting	0.86	0.92	0.83	0.87

Table 8 compares the performance of majority voting and weighted voting.

Majority voting achieves slightly higher accuracy and F1-score, while weighted voting improves recall by better detecting scam messages. The trade-off highlights that majority voting is the superior choice when prioritizing scam identification over overall accuracy.

4.7 Computational Efficiency Analysis

Beyond improving classification accuracy, the proposed multi-model voting system significantly reduces computation time compared to direct inference using the fine-tuned LLaMA 8B model. We evaluated efficiency on a dataset containing 1200 messages, including adversarial scam, regular scam, and non-scam instances.

Table 9: Computation Time Comparison: LLaMA 8B vs. Multi-Model Voting

Method	Total Time (s)	Avg. Time per Message (s)
LLaMA 8B Fine-Tuned	2296	1.91
Multi-Model Voting	995	0.83

Table 9 shows the fine-tuned LLaMA 3.1 8B Instruct model takes 2296 seconds to process the full dataset, averaging 1.91 seconds per message. In contrast, the multi-model voting system significantly reduces inference time to 995 seconds (0.83 sec. per message), achieving a 56.7% reduction in computational time.

This evaluation was conducted on a MacBook Pro M2 Pro with 32GB RAM. The significant improvement in efficiency is attributed to the ability of traditional models (RF, DT, XGB, and KNN) to handle most classifications quickly, leveraging LLaMA only when necessary. This hybrid approach optimally balances computational cost and detection accuracy, making it a more practical solution for real-world scam detection deployments.

5 Conclusions & Future Work

Summary of Key Contributions. In this work, we proposed a *Hierarchical Scam Detection System* that integrates multi-model voting with a fine-tuned

LLaMA 3.1 8B Instruct model to enhance scam detection robustness. Our key contributions include curating a comprehensive 20,000-sample scam dataset enriched with adversarial examples, fine-tuning LLaMA 3.1 8B Instruct using Low-Rank Adaptation (LoRA) to improve adversarial detection while maintaining computational efficiency, developing a novel multi-model voting framework that combines traditional classifiers with a fine-tuned LLM for enhanced decision-making, and conducting a systematic benchmarking study on the adversarial robustness of different models. These contributions collectively advance scam detection by improving detection accuracy, adversarial resilience, and computational efficiency.

Key Findings. Our experimental results demonstrate that the fine-tuned LLaMA 3.1 8B Instruct model significantly improves scam detection performance while reducing computational overhead when integrated with a multi-model voting mechanism. The fine-tuning strategy, leveraging adversarial training and LoRA-based adaptation, ensures robust performance even in a zero-shot setting, while the multi-model voting system effectively balances precision and recall across various scam types. The results also indicate that majority voting achieves higher overall accuracy and F1-scores, whereas weighted voting improves recall, particularly for scam detection. However, the trade-off analysis suggests that majority voting remains the superior approach when prioritizing scam identification over overall accuracy. Furthermore, traditional classifiers, although individually less effective than LLMs, contribute meaningfully in a hierarchical voting system, leading to enhanced adversarial robustness.

Future Work. While our proposed approach enhances scam detection, several areas remain for future exploration. First, the detection of highly context-dependent scams, such as romance scams, remains challenging due to their emotionally driven language. Future work could explore domain-specific fine-tuning strategies or reinforcement learning-based adaptation to improve model sensitivity to such scams. Second, refining the voting mechanisms by integrating adaptive weighting strategies based on model confidence or scam category could further optimize detection accuracy. Third, real-time deployment and scalability are critical for practical applications. Future efforts should focus on optimizing inference speed, integrating real-time data streams, and evaluating model performance in dynamic environments. Additionally, incorporating user feedback loops to refine the detection system based on evolving scam tactics could further strengthen resilience against adversarial modifications.

Overall, this work underscores the potential of combining fine-tuned large language models with ensemble voting frameworks to improve adversarial scam detection. Addressing these limitations in future research can further enhance the adaptability and effectiveness of AI-driven scam detection systems in real-world applications.

References

1. BeenVerified: Scamwarners website (nd), https://www.scamwarners.com, accessed: 2025-02-21
2. Calderon, M.H.H., Palad, E.B.B., Tangkeko, M.S.: Filipino online scam data classification using decision tree algorithms. In: 2020 International Conference on Data Science and Its Applications (ICoDSA). pp. 1–6. IEEE (2020)
3. Chang, C.W.: Training dataset (2025), https://github.com/wilsonchang17/adversarialscam_dataset, accessed: 2025-02-26
4. Chang, C.W., Sarkar, S., Mitra, S., Zhang, Q., Salemi, H., Purohit, H., Zhang, F., Hong, M., Cho, J.H., Lu, C.T.: Exposing llm vulnerabilities: Adversarial scam detection and performance pp. 3568–3571 (2024)
5. Chang, Y.C., Aïmeur, E.: "is this site legit?": Llms for scam website detection. In: International Conference on Web Information Systems Engineering. pp. 230–245. Springer (2024)
6. Christophe, C., Kanithi, P.K., Munjal, P., Raha, T., Hayat, N., Rajan, R., Al-Mahrooqi, A., Gupta, A., Salman, M.U., Gosal, G., et al.: Med42–evaluating fine-tuning strategies for medical llms: full-parameter vs. parameter-efficient approaches. arXiv preprint arXiv:2404.14779 (2024)
7. DevilDyno: Email spam or not classification dataset (2023), https://www.kaggle.com/datasets/devildyno/email-spam-or-not-classification, accessed: 2025-02-21
8. Dileep, M., Navaneeth, A., Abhishek, M.: A novel approach for credit card fraud detection using decision tree and random forest algorithms. In: ICICV 2021. pp. 1025–1028. IEEE (2021)
9. Gao, J., Lanchantin, J., Soffa, M.L., Qi, Y.: Black-box generation of adversarial text sequences to evade deep learning classifiers. In: 2018 IEEE Security and Privacy Workshops (SPW). pp. 50–56. IEEE (2018)
10. Gautam, A.K., Bansal, A.: Email-based cyberstalking detection on textual data using multi-model soft voting technique of machine learning approach. Journal of Computer Information Systems **63**(6), 1362–1381 (2023)
11. Hu, E.J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., Chen, W.: Lora: Low-rank adaptation of large language models (2021), https://arxiv.org/abs/2106.09685
12. Jiang, L.: Detecting scams using large language models. arXiv preprint arXiv:2402.03147 (2024)
13. Kaggle: Fraudulent job posting dataset (2017), https://www.kaggle.com/datasets/uciml/sms-spam-collection-dataset, accessed: 2025-03-08
14. Kaggle: Fraudulent job posting dataset (2023), https://www.kaggle.com/datasets/subhajournal/fraudulent-job-posting, accessed: 2025-02-21
15. Kaggle: Spam-ham classifier dataset (2024), https://www.kaggle.com/datasets/ilyazored/spam-ham-classifier-dataset?resource=download, accessed: 2025-03-08
16. Kannagi, A., Mohammed, J.G., Murugan, S.S.G., Varsha, M.: Intelligent mechanical systems and its applications on online fraud detection analysis using pattern recognition k-nearest neighbor algorithm for cloud security applications. Materials Today: Proceedings **81**, 745–749 (2023)
17. Maharana, K., Mondal, S., Nemade, B.: A review: Data pre-processing and data augmentation techniques. Global Transitions Proceedings **3**(1), 91–99 (2022)

18. Miestamo, M.: Linguistic diversity and complexity. Lingue e linguaggio **16**(2), 227–254 (2017)
19. Ozler, H.: Spam or not spam dataset (2023), `https://www.kaggle.com/datasets/ozlerhakan/spam-or-not-spam-datasetand`, accessed: 2024-10-09
20. Polikar, R.: Ensemble based systems in decision making. IEEE Circuits and systems magazine **6**(3), 21–45 (2006)
21. Rebuffi, S.A., Gowal, S., Calian, D.A., Stimberg, F., Wiles, O., Mann, T.A.: Data augmentation can improve robustness. Advances in neural information processing systems **34**, 29935–29948 (2021)
22. Salman, M., Ikram, M., Kaafar, M.A.: An empirical analysis of sms scam detection systems. arXiv preprint arXiv:2210.10451 (2022)
23. Shen, Z., Wang, K., Zhang, Y., Ngai, G., Fu, E.Y.: Combating phone scams with llm-based detection: Where do we stand? arXiv preprint arXiv:2409.11643 (2024)