Efficient Financial Fraud Detection on Mobile Devices Using Lightweight Large Language Models

Lakpriya Senevirathna
Informatics Institute of Technology (IIT)
57, Ramakrishna Road, Colombo 06,
Sri Lanka
lakpriya1024@gmail.com

Deshan Sumanathilaka
School of Computing
Swansea University, Swansea
United Kingdom
deshankoshala@gmail.com

Abstract

The growth of mobile financial transactions presents new challenges for fraud detection, where traditional and ML methods often miss emerging patterns. While Large Language Models (LLMs) offer advanced language understanding, they are typically too resource-intensive for mobile deployment and raise privacy concerns due to cloud reliance. This paper proposes a lightweight, privacy-preserving approach by fine-tuning and quantizing compact LLMs for on-device fraud detection from textual data. Models were optimized using Open Neural Network Exchange (ONNX) conversion and quantization to ensure efficiency. The fine-tuned quantized *Llama-160M-Chat-v1 (bnb4)* achieved 99.47% accuracy with a 168MB footprint, while fine-tuned quantized *Qwen1.5-0.5B-Chat (bnb4)* reached 99.50% accuracy at 797MB. These results demonstrate that optimized LLMs can deliver accurate, real-time fraud detection on mobile devices without compromising user privacy.

1 Introduction

Financial fraud has become a massive issue with the rapid development of financial tasks on mobile devices. This threatens consumers and institutions. (Al-Hashedi and Magalingam, 2021). Fraudsters use growing connectivity to target mobile users, especially those who do not have technological literacy (Silva et al., 2021). Organizations such as banks, mobile money operators, and government institutions often use popular communication methods such as Short Message Service (SMS), calls, and emails. Fraudsters use the same techniques to trick users with fake messages and calls by taking advantage of people's trust in financial institutions such as banks and government organizations (Razaq et al., 2021).

Traditional fraud detection and ML methods are often implemented on cloud servers. They face challenges and limitations when adapting to mobile environments (Botchey et al., 2020). Rule-based systems have predefined rules, which limit their ability to adapt to new fraud patterns. ML models rely on structured data for fraudulent patterns and require significant feature engineering (Tax et al., 2021). Thus, Traditional ML methods often need frequent retraining. The complex, non-linear patterns indicative of advanced fraud schemes are often beyond the capability of traditional methods like rule-based systems and statistical models to analyze effectively (Cao et al., 2024).

LLMs have shown significant advancement in Natural Language Understanding (NLU), pattern recognition, and anomaly detection, which are closely related to financial fraud detection from textual data (Sun et al., 2020). LLM can generalize from language and pattern understanding, enabling zero-shot or few-shot detection (Papasavva et al., 2024; Sumanathilaka et al., 2024). Using LLM is computationally intensive, which often makes it unsuitable for mobile devices with limited resources. LLMs like FinBERT (Huang et al., 2023a) and BloombergGPT (Wu et al., 2023) have shown impressive results in financial text analysis, which makes them essential to identify fraud signs in unstructured data such as transactional records and messages. Deploying LLMs on mobile devices requires careful optimization using techniques such as quantization (Silva et al., 2021).

Current financial fraud detection solutions are based on cloud services, which introduce privacy concerns because cloud-based processing of users' private data and sensitive financial information increases the risk of data breaches and privacy violations (Bajracharya et al., 2023). Developing and deploying fine-tuned, quantized versions of LLMs that process financial and user data locally is a promising approach to address these issues. Fur-

Proceedings of Recent Advances in Natural Language Processing,pages 1090–1098
Varna, Sep 8–10, 2025
https://doi.org/10.26615/978-954-452-098-4-126

thermore, this research will evaluate the various fine-tuned LLMs that are used for mobile deployment.

To bridge this gap, the current research proposes an efficient, privacy-preserving framework for on-device financial fraud detection from textual data using lightweight LLMs. Our goal is to bring advanced NLP capabilities to mobile platforms without compromising on performance or user data security. The proposed solution involves fine-tuning pre-trained LLMs on domain-specific financial fraud datasets, optimizing them through quantization techniques such as INT8 compression, and converting them into ONNX formats for mobile compatibility. This approach ensures that the models can perform real-time inference directly on Android and iOS devices, minimizing the data transfer to external servers and reducing latency.

The novelty of this research lies in the development of a set of mobile-optimized LLMs for textual fraud detection and their integration into a mobile application. Unlike previous works that focus on cloud-based deployment or general NLP tasks, this study adapts lightweight LLMs to detect fraud signals in unstructured financial text such as SMS, emails, and in-app messages. Techniques like Low-Rank Adaptation (LoRA) and QLoRA were used to fine-tune models with minimal computational overhead (Dettmers et al., 2023). Moreover, the models were rigorously evaluated using various metrics, including accuracy, precision, recall, F1-score, and inference latency.

From a privacy standpoint, on-device processing ensures compliance with modern data protection standards by eliminating the need to transmit sensitive financial data to cloud servers. This is especially important in financial contexts where breach risks and regulatory penalties are significant. Furthermore, mobile hardware advancements, such as Neural Processing Units (NPUs) in modern smartphones offer new opportunities for accelerating on-device inference, thereby improving both speed and energy efficiency (Yin et al., 2024).

Our main contributions include:

- A novel framework for fine-tuning lightweight LLMs on domain-specific financial fraud datasets using parameter-efficient techniques.
- A mobile application that enables on-device deployment and inference of quantized LLMs for real-time financial fraud detection from textual data on Android[1] and iOS[2] platforms.
- Comprehensive empirical benchmarking of multiple LLM architectures and quantization variants, analyzing trade-offs between accuracy, model size, and inference latency.
- Open-source release of the codebase[3] , fine-tuned models[4], and mobile deployment pipeline[5] to support reproducibility and community adoption.

The rest of the paper is organized as follows. Section 2 reviews related work on fraud detection and LLM deployment strategies. Section 3 describes the datasets we used. Section 4 details the methodology, including data preprocessing, model selection, fine-tuning, evaluation and deployment. Section 5 discusses the results and insights from benchmarking. Section 6 summarizes our main experimental findings and conclusions with future directions followed by limitations section of this study which covers dataset coverage, deployment constraints, and resource-related considerations.

2 Related Work

Rule-based systems use predefined rules and conditions set by experts to identify suspicious patterns that are associated with known fraud. These systems are the earliest solution to fraud detection and are used in some sectors even today. These systems are simple to implement, easy to understand and can identify known frauds, but adaptability to new fraud schemes is limited and may generate false positive results (Kumar and Goswami, 2024).

Financial fraud detection involves analyzing both structured and unstructured data to identify and prevent fraudulent activities. Unstructured data, like SMS, emails, and chats, can contain valuable insights into fraudulent activities and structured data, like transaction details and account information, can provide qualitative information (Bajracharya et al., 2023). Machine learning approaches provide data-driven methods that are capable of identifying complex patterns.

LLMs can analyze massive amounts of textual data, including various types of transactions,

[1] Android App

[2] iOS App

[3] Python notebooks for fine-tuning LLMs

[4] Fine-tuned models

[5] FraudShild app source code

emails, messages, social media activities, and even legal documents, to identify subtle patterns that might be fraudulent activities (Cao et al., 2024). LLMs have become increasingly influential in supporting human cognition and interaction, highlighting significant advancements in natural language generation through deep learning models and transformers. The rise of large-scale data sets and enhanced computing power has led to the development of foundation models, which achieve state-of-the-art performance across various tasks (Bommasani et al., 2021). LLMs can identify subtle deviations in transaction history by analyzing unstructured data sources such as customer communications and social media. This capability allows them to potentially detect fraud in real-time. In the context of banking, LLMs are particularly effective for fraud detection from textual data due to their ability to analyze vast amounts of data and pinpoint anomalies that may indicate fraudulent activity (Anwaar, 2024). LLMs can be used to detect unusual patterns in transactional descriptions that might identify suspicious language in customer emails (Li, 2023; Cao et al., 2024). The LLMs like FinBERT and BloombergGPT have shown effective in financial analysis tasks but they do not address their usage in financial fraud detection from textual data (Huang et al., 2023b; Wu et al., 2023). Furthermore, existing studies like MobileBERT mainly target NLP tasks but it does not address any financial fraud context (Sun et al., 2020).

Mobile devices have extreme computational and memory limitations compared to server-based fraud detection systems. This constraint restricts the complexity and size of the LLM that can be deployed efficiently (Murthy et al., 2024). For example, to deploy GPT-3 175B parameter model required at least five 80GB A100 GPUs and 350GB of memory, even when using the FP16 format (Naveed et al., 2023). Most smartphones only have a CPU which can lead to rapidly draining the battery when running LLM on mobile devices, which can affect user experience (Yin et al., 2024). As the number of parameters and calculations in the LLM increases, response latency may delayed, affecting the performance of real-time systems. This increased memory demand can lead to bottlenecks, especially in resource-constrained environments like mobile devices (Touvron et al., 2023).

Given the strong performance of LLMs across various NLP tasks, we hypothesize that they can be effectively applied to financial fraud detection from textual data. To the best of our knowledge, this is the first study to explore the use of lightweight LLMs for on-device financial fraud detection on mobile platforms.

3 Data

We evaluated our approach using a publicly available dataset containing financial fraud-related textual data. Specifically, we utilized the Fraud Email Dataset curated by (Verma, 2018), which includes a mixture of legitimate and fraudulent messages collected from real-world sources. From this dataset, we randomly selected 4,000 labeled entries to train and evaluate our models.

The dataset consists of unstructured financial communication data such as emails and messages, annotated with binary labels: fraud or legit. These texts include phishing attempts, fake banking notices, and scam transaction prompts, making them representative of common fraud vectors targeting mobile users. Given the known issue of class imbalance in fraud detection datasets, we ensured a balanced distribution of fraudulent and non-fraudulent samples during training to improve model generalizability. To validate the quality of labels, inter-annotator agreement was calculated using Cohen's Kappa, resulting in scores above 0.87 across annotators, indicating strong consistency.

Training LLMs typically requires access to massive-scale datasets, often referred to as pre-training corpora or foundation datasets. However, the objective of this study is different. Instead of training LLMs from scratch, we focus on fine-tuning existing lightweight LLMs using a relatively small, domain-specific dataset. The goal is to evaluate how well these compact models can adapt to financial fraud detection tasks when deployed in resource-constrained environments such as mobile devices. This dataset served as the foundation for fine-tuning and evaluating multiple lightweight LLMs, providing a practical basis for assessing their real-world performance and relevance in detecting financial fraud on-device.

4 Methodology

We divide our method into five distinct stages, each designed to enable the development, optimization, and evaluation of lightweight LLMs for financial fraud detection on mobile devices. This pipeline focuses on adapting existing pre-trained models to a resource-constrained and privacy-sensitive environment, ensuring that the models remain efficient and accurate after deployment. The five stages are outlined below in subsections, along with Mobile Integration.

4.1 Data Collection and Preprocessing

We cleaned the dataset[6] by removing HTML tags, URLs, special characters, and duplicates. Text normalization techniques were applied, including lowercasing and whitespace trimming. To ensure fair model training and evaluation, the dataset was balanced across both classes and split into training, validation, and testing sets.

4.2 Model Selection

Several lightweight LLMs were chosen based on their size, compatibility with mobile deployment, and support for fine-tuning. The selected models included *Llama-160M, Qwen1.5-0.5B, SmolLM2-135M, Minueza-32M, and TinyLlama-1.1B*. These models were prioritized for their relatively low memory requirements and their suitability for adaptation to binary classification tasks (Popov et al., 2025).

4.3 Fine-Tuning

Each model was fine-tuned using a binary classification objective to detect fraud. We used parameter-efficient fine-tuning techniques such as LoRA and QLoRA to minimize memory and compute usage. Fine-tuning was performed on preprocessed data using Hugging Face's Transformers and PEFT libraries, with hyperparameters selected based on empirical performance on the validation set.

4.4 Quantization and ONNX Conversion

After fine-tuning, each model was optimized through quantization to reduce memory usage and improve inference efficiency. These optimizations are essential for enabling real-time processing on mobile devices with limited computational resources. Instead of relying solely on INT8 quantization, we experimented with a variety of quantization formats to compare performance and resource trade-offs. The following quantized variants were generated:

- 8-bit integer quantization (INT8)
- Unsigned 8-bit integer (UINT8)
- 4-bit quantization (Q4)
- 4-bit quantization with 16-bit floating-point weights (Q4F16)
- Block-wise 4-bit quantization using bits-and-bytes (BNB4)
- 16-bit floating-point precision (FP16)
- Default full precision export (FULL)

Each quantized model was then converted into the ONNX format using the Hugging Face Optimum library. This step ensures cross-platform compatibility and efficient deployment on mobile devices through ONNX Runtime. The full model preparation pipeline is illustrated in Figure 1, outlining the steps from pulling the base LLM to pushing mobile-optimized variants.

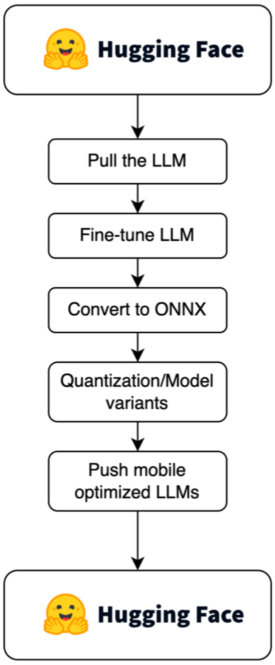

Figure 1: End-to-end model training and optimization workflow

4.5 Evaluation

Model evaluation was performed in a Python notebook environment using a held-out test subset. Each model was assessed on standard classification metrics, including accuracy, precision, recall, F1-score, and inference latency. To examine model convergence and training efficiency, we fine-tuned each model using two different training durations: 5 epochs and 10 epochs. This comparative setup allowed us to observe the effect of training length on overall performance and stability. Additionally,

[6] Dataset

performance was evaluated across various quantization levels (e.g., INT8, Q4, FP16, BNB4) to analyze the trade-offs between compression and classification accuracy, particularly in the context of mobile deployment.

4.6 Mobile Integration

Following evaluation, the ONNX models were integrated into a React Native-based mobile application using ONNX Runtime, enabling cross-platform support for both Android and iOS. The application is designed to load models locally and operate offline, ensuring that sensitive user data remains on-device. Users can input suspicious messages and receive real-time fraud classification powered by lightweight LLMs. As illustrated in Figure 2, the app dynamically downloads the mobile-optimized LLM from the Hugging Face Model Hub, then loads and executes the model using ONNX Runtime and Transformers.js. This modular approach allows users to choose from different quantized variants (e.g., INT8, FP16, BNB4) based on device capability.

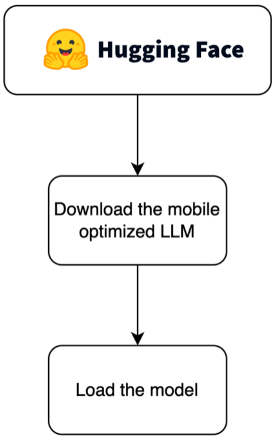

Figure 2: On-device model loading workflow

5 Results and Discussion

To evaluate the effectiveness of lightweight LLMs in detecting financial fraud on mobile devices, we conducted a series of experiments using multiple fine-tuned and quantized models. This section presents the evaluation results based on standard classification metrics, including accuracy, precision, recall, and F1-score. We also discuss the trade-offs introduced by different quantization levels in terms of performance, model size, and runtime efficiency. To showcase the real-world applicability of our approach, we developed a mobile application named *FraudShield*. The app allows users to input suspicious messages and receive instant feedback based on on-device fraud classification. Users can select from multiple lightweight LLM variants deployed using ONNX Runtime, ensuring fast, private, and offline inference. The app interface provides clear fraud alerts and safety notifications, enhancing accessibility for non-technical users.

5.1 Classification Performance

To assess the adaptability of lightweight LLMs to the financial fraud detection task, we fine-tuned each model using two different training durations: 5 epochs and 10 epochs. This allowed us to compare the impact of training time and convergence on performance metrics such as accuracy. The results, summarized in Table 1, show that most models benefited from extended training. For instance, *Llama-160M-Chat-v1 (bnb4 variant)* achieved an accuracy of 98.25% at 5 epochs and improved to 99.47% at 10 epochs. Similarly, *Qwen1.5-0.5B-Chat (bnb4 variant)* increased from 95.50% to 99.50% across the same epoch range.

Smaller models such as *Minueza-32M* and *SmolLM2-135M* exhibited slightly lower accuracies but still maintained strong performance. *Minueza-32M* achieved 95.26% accuracy at 5 epochs and 96.03% at 10 epochs. *SmolLM2-135M* improved from 92.27% to 95.50%, indicating that even ultra-light models can generalize well given modest training. Models with higher parameter counts, such as *TinyLlama-1.1B* and *Llama-3.2-1B*, performed consistently well.*TinyLlama-1.1B* achieved 98.75% accuracy under both epoch settings, suggesting early convergence. *Llama-3.2-1B* reached 98.50% with 10 epochs and peaked at 99.00% at 5 epochs, showing stable performance regardless of training duration. These findings confirm that even with limited data and compute resources, fine-tuning lightweight LLMs yields high accuracy in financial fraud detection. The comparative training times also indicate the feasibility of iterating model development efficiently, with most smaller models completing training in under two hours.

This pattern is further supported by the performance heatmap shown in Figure 3. Fine-tuned variants of all evaluated LLMs consistently outperform their base versions across key evalua-

LLM	Epoch	Learning Rate	Batch Size	Best Variant	Accuracy	Training Time(min)
Minueza-32M-Chat	10	2e-5	4	model_fp16	0.960317	83
	5	2e-5	4	model_q4	0.952618	43
SmolLM2-135M-Instruct	10	2e-5	4	model_bnb4	0.955026	224
	5	2e-5	4	model_bnb4	0.922693	118
Llama-160M-Chat-v1	10	2e-5	4	model_bnb4	0.994709	89
	5	2e-5	4	model_bnb4	0.982544	47
Qwen1.5-0.5B-Chat	10	2e-5	4	model_bnb4	0.995012	191
	5	2e-5	4	model_bnb4	0.955026	224
Llama-3.2-1B	10	2e-5	4	model_fp16	0.985037	174
	5	2e-5	4	model_fp16	0.990025	90
TinyLlama-1.1B-Chat-v1.0	10	2e-5	4	model_bnb4	0.987531	197
	5	2e-5	4	model_bnb4	0.987531	90

Table 1: Results for various LLMs under different training epochs.

tion metrics, including precision, recall, and F1-score. Notably, quantized formats such as *bnb4* and *fp16* maintained strong accuracy while reducing model size, demonstrating their suitability for deployment on resource-constrained mobile environments. These results highlight the potential of optimized lightweight LLMs to deliver both accuracy and efficiency in real-world fraud detection use cases.

5.2 Impact of Quantization

We evaluated the same fine-tuned models under multiple quantization formats to examine the effect of compression on predictive performance. INT8 and bnb4 quantization formats showed minimal loss in accuracy while significantly reducing model size. For instance, the bnb4 variant of *Llama-160M-Chat-v1* reduced the model size to approximately 168MB, making it suitable for mobile deployment without compromising prediction quality.

More aggressive quantization, such as q4 and q4f16, led to slight drops in recall and F1-score, especially in smaller models. However, these variants still maintained accuracy above 90%, indicating that low-bit quantization is viable when storage and inference speed are prioritized over marginal gains in precision.

5.3 Resource Usage

We conducted inference testing on real devices, including a Samsung S21 Plus (Android) and an iPhone 14 Pro (iOS), using the ONNX Runtime integrated into the mobile application. The results showed that models under 200MB loaded in under 1 second and delivered predictions in less than 500 milliseconds, making them responsive enough for real-time fraud detection. The bnb4 and int8 models exhibited the most balanced performance in terms of size, accuracy, and latency. Resource usage measurements revealed that memory consumption consistently stayed below 1GB and CPU usage remained modest, with brief spikes only during model initialization. These findings demonstrate stable memory and CPU behavior across platforms, confirming the feasibility of deploying optimized lightweight LLMs on mid- to high-end mobile devices.

5.4 Comparative Analysis

The results, presented in Figure 4, show that fine-tuned lightweight LLMs significantly outperformed traditional machine learning models in terms of classification accuracy. Among all models evaluated, *Qwen1.5-0.5B-Chat (bnb4)* and *Llama-160M-Chat-v1 (bnb4)* achieved the highest accuracy, reaching 99.50% and 99.47% respectively with 10 epochs of training. In contrast, the best-performing classical model, Random Forest, achieved an accuracy of 98.67%. Other models such as SVM, Logistic Regression, and Decision Tree ranged between 95% and 98%.

The figure compares models trained for 5 and 10 epochs. Although the LLMs generally improved with longer training, some models like *TinyLlama-1.1B-Chat-v1.0* and *Llama-3.2-1B-Instruct* reached strong performance even with shorter training durations. In addition to higher accuracy, LLMs required no manual feature engineering and were capable of processing raw, unstructured input like messages and emails. This adaptability, combined with their robustness across quantization formats,

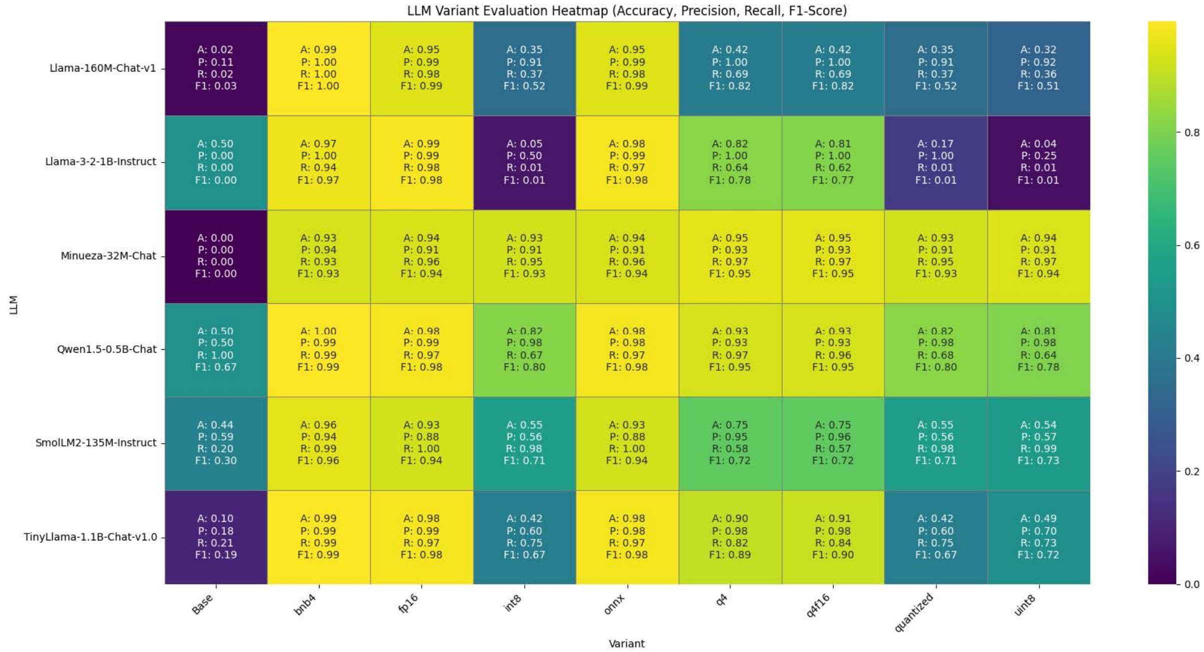

Figure 3: LLM variant evaluation heatmap showing performance (Accuracy, Precision, Recall, F1-Score) after 10 epochs of fine-tuning.

supports the viability of lightweight LLMs for privacy-preserving mobile fraud detection.

5.5 Error Analysis

During evaluation, most fine-tuned lightweight LLMs achieved high accuracy in detecting financial fraud on unstructured text. However, error patterns indicated that smaller models, such as Minueza-32M and SmolLM2-135M, occasionally misclassified messages containing ambiguous language or less frequent fraud cues. False negatives were primarily observed in edge cases where the language used was overly polite or mimicked legitimate communication. Quantization also contributed marginally to performance degradation in ultra-low-bit formats (e.g., Q4), particularly affecting recall. These errors suggest a need for expanding training data diversity and incorporating real-world multilingual fraud samples to improve generalizability and robustness across nuanced and evolving fraud tactics.

6 Conclusion

Through fine-tuning multiple LLMs on a domain-specific dataset and applying quantization strategies such as INT8, BNB4, and Q4, we achieved high classification performance while maintaining small model sizes suitable for on-device execution. Notably, models such as *Llama-160M-Chat-v1* and *Qwen1.5-0.5B-Chat* reached over 99% accuracy even in quantized formats, outperforming traditional machine learning models trained on the same data. Furthermore, inference latency and resource usage measurements confirmed that these models can operate efficiently on modern smartphones.

In addition to quantitative metrics, this study validated its real-world applicability through mobile integration, user testing, and benchmarking against conventional methods. The results confirm that lightweight LLMs are a viable and privacy-preserving solution for secure, real-time fraud detection on mobile platforms.

Future work includes multilingual support, streaming input, and on-device continual learning. Additionally, further research into ultra-efficient quantization techniques and hardware-specific acceleration (e.g., using NPUs) may further enhance performance on low-end devices.

Limitation

This study was based on a publicly available English-language dataset, which may not fully capture the complexity and diversity of real-world financial fraud, particularly across different languages, regions, and evolving scam tactics. As a result, the models may have limited generalizability outside the scope of the dataset used for training. Additionally, due to budget limitations, access to

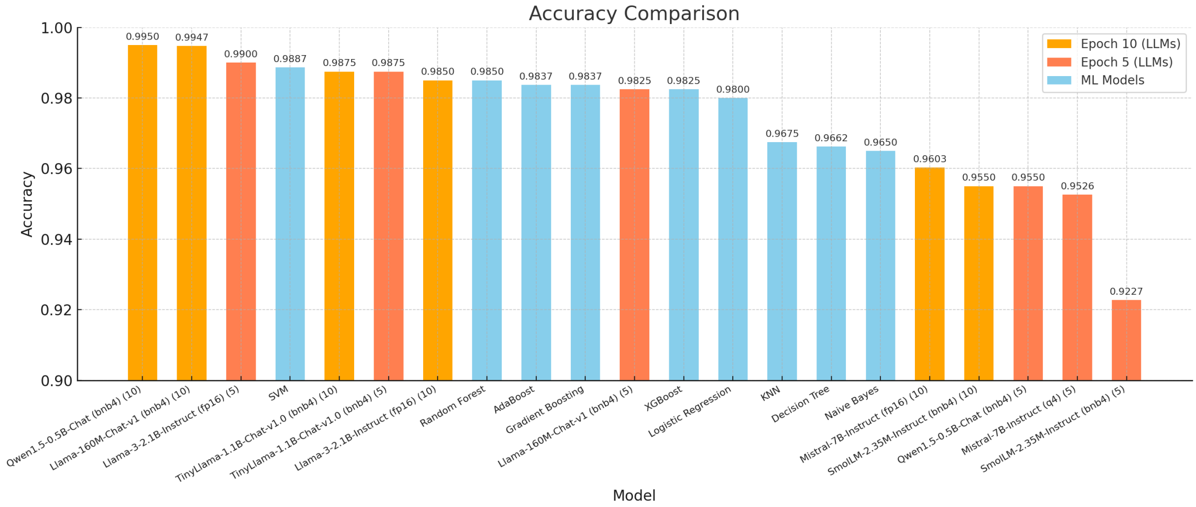

Figure 4: Accuracy comparison of LLMs (across 5 and 10 epochs) and traditional ML models. LLMs consistently outperform classical models, particularly with longer training.

large-scale proprietary or multilingual datasets was not possible, which could have enhanced model robustness. Testing was conducted on high-end devices, and the results may not reflect performance on lower-end smartphones with limited memory or processing power. Time constraints also restricted broader device testing and experimentation with more advanced training configurations or larger models. Moreover, the current system relies on static, pre-trained models without real-time learning capabilities, which may affect its long-term adaptability to new fraud patterns.

References

Khaled Gubran Al-Hashedi and Pritheega Magalingam. 2021. Financial fraud detection applying data mining techniques: A comprehensive review from 2009 to 2019. *Computer Science Review*, 40:100402.

Salman Anwaar. 2024. Harnessing large language models in banking: Banking innovation with operational and security risks. *World Journal of Advanced Engineering Technology and Sciences*, 13(1).

Aakriti Bajracharya, Barron Harvey, and Danda B. Rawat. 2023. Recent advances in cybersecurity and fraud detection in financial services: A survey. In *2023 IEEE 13th Annual Computing and Communication Workshop and Conference (CCWC)*, pages 0368–0374.

Rishi Bommasani, Drew A Hudson, Ehsan Adeli, Russ Altman, Simran Arora, Sydney von Arx, Michael S Bernstein, Jeannette Bohg, Antoine Bosselut, Emma Brunskill, et al. 2021. On the opportunities and risks of foundation models. *arXiv preprint arXiv:2108.07258*.

Francis Effirim Botchey, Zhen Qin, and Kwesi Hughes-Lartey. 2020. Mobile money fraud prediction—a cross-case analysis on the efficiency of support vector machines, gradient boosted decision trees, and naïve bayes algorithms. *Information*, 11(8).

Xinwei Cao, Shuai Li, Vasilios Katsikis, Ameer Tamoor Khan, Hailing He, Zhengping Liu, Lieping Zhang, and Chen Peng. 2024. Empowering financial futures: Large language models in the modern financial landscape. *EAI Endorsed Transactions on AI and Robotics*, 3.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2023. Qlora: Efficient finetuning of quantized llms. *Advances in neural information processing systems*, 36:10088–10115.

Allen H. Huang, Hui Wang, and Yi Yang. 2023a. Finbert: A large language model for extracting information from financial text. *Contemporary Accounting Research*, 40(2):806–841.

Allen H. Huang, Hui Wang, and Yi Yang. 2023b. Finbert: A large language model for extracting information from financial text. *Contemporary Accounting Research*, 40(2):806–841.

Jitendra Kumar and Pankaj Kumar Goswami Goswami. 2024. Credit card fraud detection using machine learning. 6(2):19237.

Qiuru Li. 2023. Textual data mining for financial fraud detection: A deep learning approach. *arXiv preprint arXiv:2308.03800*.

Rithesh Murthy, Liangwei Yang, Juntao Tan, Tulika Manoj Awalgaonkar, Yilun Zhou, Shelby Heinecke, Sachin Desai, Jason Wu, Ran Xu, Sarah Tan, et al. 2024. Mobileaibench: Benchmarking llms and lmms for on-device use cases. *arXiv preprint arXiv:2406.10290*.

Humza Naveed, Asad Ullah Khan, Shi Qiu, Muhammad Saqib, Saeed Anwar, Muhammad Usman, Naveed Akhtar, Nick Barnes, and Ajmal Mian. 2023. A comprehensive overview of large language models. *arXiv preprint arXiv:2307.06435*.

Antonis Papasavva, Shane Johnson, Ed Lowther, Samantha Lundrigan, Enrico Mariconti, Anna Markovska, and Nilufer Tuptuk. 2024. Application of ai-based models for online fraud detection and analysis. *arXiv preprint arXiv:2409.19022*.

Ruslan O Popov, Nadiia V Karpenko, and Volodymyr V Gerasimov. 2025. Overview of small language models in practice. In *CEUR Workshop Proceedings*, pages 164–182.

Lubna Razaq, Tallal Ahmad, Samia Ibtasam, Umer Ramzan, and Shrirang Mare. 2021. ”we even borrowed money from our neighbor”: Understanding mobile-based frauds through victims’ experiences. *Proc. ACM Hum.-Comput. Interact.*, 5(CSCW1).

Julio Cezar Soares Silva, David Macêdo, Cleber Zanchettin, Adriano L.I. Oliveira, and Adiel Teixeira de Almeida Filho. 2021. Multi-class mobile money service financial fraud detection by integrating supervised learning with adversarial autoencoders. In *2021 International Joint Conference on Neural Networks (IJCNN)*, pages 1–7.

Deshan Sumanathilaka, Nicholas Micallef, and Julian Hough. 2024. Assessing gpt’s potential for word sense disambiguation: A quantitative evaluation on prompt engineering techniques. In *2024 IEEE 15th Control and System Graduate Research Colloquium (ICSGRC)*, pages 204–209. IEEE.

Zhiqing Sun, Hongkun Yu, Xiaodan Song, Renjie Liu, Yiming Yang, and Denny Zhou. 2020. MobileBERT: a compact task-agnostic BERT for resource-limited devices. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2158–2170, Online. Association for Computational Linguistics.

Niek Tax, Kees Jan de Vries, Mathijs de Jong, Nikoleta Dosoula, Bram van den Akker, Jon Smith, Olivier Thuong, and Lucas Bernardi. 2021. Machine learning for fraud detection in e-commerce: A research agenda. In *Deployable Machine Learning for Security Defense*, pages 30–54, Cham. Springer International Publishing.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Abhishek Verma. 2018. Fraud email dataset.

Shijie Wu, Ozan Irsoy, Steven Lu, Vadim Dabravolski, Mark Dredze, Sebastian Gehrmann, Prabhanjan Kambadur, David Rosenberg, and Gideon Mann. 2023. Bloomberggpt: A large language model for finance. *arXiv preprint arXiv:2303.17564*.

Wangsong Yin, Mengwei Xu, Yuanchun Li, and Xuanzhe Liu. 2024. Llm as a system service on mobile devices. *arXiv preprint arXiv:2403.11805*.