PROJECT REPORT

ON

(SEECURE – Real-Time Fraud Call Detection App)

VISHWAKARMA INSTITUTE OF INFORMATION TECHNOLOGY, PUNE

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING (AIML)

BY

Name	PRN	Roll No
Yuvraj Sankilwar	22210188	391052
Kishor Patil	22210749	391042
Shreejit Bhakte	22210541	391005

Class: TY CS-AIML **Division: A**

Guided By: Dr.Disha S.Wankhede

INDEX

Introduction

In an increasingly digital world, mobile communication has become an integral part of everyday life. However, this widespread accessibility has also led to a significant rise in social engineering attacks and financial fraud conducted through phone calls. Vulnerable populations such as the elderly, less-educated individuals, or rural communities are especially at risk, often falling prey to sophisticated scam techniques.

To address this growing concern, **SEECURE** — a real-time fraud call detection application — has been developed as a mobile-first solution aimed at protecting users from fraudulent activities during voice calls. SEECURE leverages advancements in **speech recognition** and **natural language processing (NLP)** to analyze ongoing conversations and detect potential scam indicators on the fly.

The application functions by capturing live audio during phone conversations, transcribing the speech using the **Vosk model** (an offline speech-to-text engine), and then analyzing the transcribed content using a lightweight **Ollama LLaMA language model**. If suspicious or scam-like language patterns are detected, SEECURE immediately notifies the user with a vibration alert and visual warning, thereby enabling proactive defense against financial threats.

Unlike traditional spam detection systems that rely on predefined phone number blacklists, SEECURE focuses on **contextual, real-time speech analysis**, making it adaptable to new and evolving scam strategies. With a privacy-first architecture, seamless mobile integration, and low-latency model inference, SEECURE offers a powerful and practical tool for fraud prevention, especially in underserved communities.

This project aims not only to build a functional prototype but also to demonstrate the real-world impact of combining on-device speech processing and AI-driven fraud detection in safeguarding users from financial exploitation.

1.1 Overview

Fraudulent phone calls have become a major threat in today's digital era, with scammers exploiting human trust to carry out financial and personal crimes. Traditional fraud prevention techniques often fail to detect scams in real time, leaving users vulnerable. To address this issue, **SEECURE** is developed as a real-time fraud call detection system that leverages the power of Artificial Intelligence (AI) and Natural Language Processing (NLP) to identify potential fraud during live conversations and alert the user instantly. The application integrates mobile technologies, backend AI services, and cloud deployment to offer a seamless and secure user experience.

1.2 Motivation

With the rise in sophisticated scam techniques, there is a pressing need for proactive fraud detection systems that work in real time. Many victims, especially senior citizens and less tech-savvy individuals, often realize too late that they are being scammed. Our motivation is to build a system that can provide immediate warnings during suspicious calls, thus preventing fraud before it happens. By combining speech recognition, real-time language analysis, and intelligent alert mechanisms, **SEECURE** aims to protect users effectively against emerging threats.

1.3 Problem Definition and Objectives

Problem Definition

The lack of real-time intervention during fraudulent phone calls results in millions of dollars lost annually. Existing solutions are either manual (relying on user awareness) or reactive (acting after the scam has occurred), which is insufficient in today's fast-paced digital environment.

Objectives

- To develop a mobile application capable of real-time audio capture during phone conversations.
- To transcribe audio into text using advanced speech-to-text models (e.g., Whisper, Google Speech-to-Text).
- To analyze conversations in real time using a LLaMA-based AI model hosted via FastAPI.
- To immediately alert users if suspicious patterns are detected in the conversation.
- To ensure high usability, minimal disruption, and secure handling of user data.

1.4 Project Scope & Limitations

Scope

- Real-time transcription and analysis of ongoing phone conversations.
- Fraud risk analysis using AI models trained on scam conversation patterns.

- Instant alerts through device vibration and app notifications.
- Cross-platform mobile application (Android/iOS) built using React Native (Expo).
- Backend deployment using FastAPI, Docker, and AWS EC2 instances for scalability.

1.5 Methodologies of Problem Solving

- **Data Collection**: Gathering datasets of real and simulated fraudulent conversations to fine-tune the LLaMA language model.
- **Model Training**: Training a classification model to distinguish between normal and scam conversations.
- **Speech-to-Text Processing**: Utilizing Whisper or Google Speech-to-Text APIs to transcribe live conversations.
- **Backend Service**: FastAPI backend is designed to process transcripts in real-time and return fraud detection results with low latency.
- **Mobile Application Development**: Building a user-friendly app using React Native (Expo) that captures audio, displays results, and provides alerts.
- **Cloud Deployment**: Deploying the backend on AWS EC2 with Docker containers for reliability and scalability.
- **Testing and Validation**: Rigorous testing across different scenarios and devices to ensure robustness and user safety.

Literature Survey

The problem of fraud detection, especially in telecommunication, has been a subject of active research for many years. Traditional fraud detection techniques primarily relied on manual reporting, call-blocking databases, and simple pattern matching methods. However, these approaches often failed to address real-time detection needs, making users vulnerable during ongoing calls. Recent advancements in Natural Language Processing (NLP) and deep learning have opened new possibilities for real-time fraud detection. Models such as BERT, GPT, and LLaMA have demonstrated significant improvements in understanding conversational patterns, sentiment, and intent, which can be applied to detect suspicious behavior in calls. Furthermore, speech-to-text models like OpenAI's Whisper and Google Speech-to-Text have greatly enhanced the ability to transcribe real-time conversations accurately, even in noisy environments. Studies also show that integrating AI into mobile devices for real-time applications is feasible with lightweight models and efficient backend architectures. Despite these advancements, challenges remain, particularly in achieving high accuracy with low latency on mobile platforms and ensuring privacy and security of user data. The SEECURE project builds upon these research advancements by combining real-time speech recognition, fraud conversation detection using fine-tuned LLaMA models, and seamless mobile application integration to offer an innovative solution tailored to real-world usage scenarios.

System Design

The system design of SEECURE focuses on delivering real-time fraud call detection through an efficient, modular, and scalable architecture. The design follows a layered approach, ensuring each component operates independently while seamlessly integrating with others to provide a smooth end-to-end experience.

The mobile application, built using React Native with Expo, captures real-time audio during calls with explicit user permission. This audio is preprocessed and securely sent to the backend server, developed using FastAPI, which manages communication between the app and machine learning models. The audio is transcribed into text using the Whisper model, ensuring accurate and fast speech-to-text conversion even in noisy environments. The transcribed text is analyzed in real-time by a fine-tuned LLaMA model, hosted on Groq Cloud, to detect patterns commonly associated with scam or fraud activities.

When potential fraud is identified, the system triggers instant alerts through strong vibrations, notifications, and an optional sound alarm to immediately warn the user. Error handling mechanisms are incorporated to manage issues like poor network connection, incomplete audio capture, or model server downtime, ensuring graceful fallback and retry options.

The backend is Dockerized for consistency and ease of deployment across different environments, and hosted on AWS EC2, ensuring high availability, reliability, and low-latency communication. Data privacy and security are prioritized throughout the flow, using encrypted channels (HTTPS) and token-based authentication to protect sensitive user information.

The system is designed with future flexibility in mind — allowing easy integration of additional models, multilingual support for wider regional adoption, and intelligent chatbot features for user assistance. MLOps best practices such as version-controlled deployments, automated testing pipelines (via GitHub Actions), and monitoring systems are integrated to maintain high service quality and enable rapid updates.

Overall, the SEECURE system design balances speed, accuracy, scalability, and security, delivering a professional-grade solution to help users stay protected from fraudulent calls in real-time.

3.1 System Architechture

The architecture of the SEECURE system is designed as a modular, scalable, and efficient pipeline to enable real-time fraud detection during phone calls. It integrates mobile application components, backend services, machine learning models, and cloud infrastructure in a streamlined workflow.

The system follows a **three-tier architecture** comprising the **Client Layer**, **Backend Layer**, and **AI Model Layer**, each handling specific responsibilities:

- **Client Layer (Mobile Application)**:
  The mobile application, developed using React Native with Expo, acts as the user interface and audio capture tool. During a call, with user permission, it records live audio, processes it lightly on the device, and transmits the audio securely to the backend server over HTTPS. The mobile app also receives real-time alerts in case fraudulent activity is detected, ensuring immediate user awareness.

- **Backend Layer (API Server and Processing Unit)**:
  The backend, built with FastAPI and hosted on AWS EC2, handles API requests from the mobile app. It manages:
  - Receiving the recorded audio stream.
  - Passing the audio through a transcription model (Whisper API).
  - Sending the transcribed text to the AI model for fraud analysis.
  - Sending back the result to the mobile app in real-time. The backend is containerized using Docker for consistency, and load balancing strategies can be incorporated in future versions for handling large volumes of requests.
- **AI Model Layer (Hosted Model and Inference Engine)**:
  The transcribed text is analyzed using a fine-tuned LLaMA model, deployed on Groq Cloud for high-speed inference. The model predicts whether the conversation indicates potential fraud based on the content, tone, and linguistic patterns. Based on the model's prediction confidence score, appropriate actions (alerts or notifications) are decided.
- **Cloud Infrastructure**:
  AWS EC2 instances are used to host backend services, and AWS S3 (optional) can be used to store logs or model artifacts. Groq Cloud serves the high-performance inference API. GitHub Actions automate deployments and updates, ensuring CI/CD best practices are maintained.
- **Security Measures**:
  - Communication between app and backend is encrypted using HTTPS.
  - Token-based authentication is applied to secure the API.
  - User data, such as call audio, is processed temporarily and discarded after prediction without persistent storage unless explicitly permitted.

This layered, loosely coupled architecture ensures modularity, enabling easy maintenance, scaling, and the addition of new functionalities, such as support for multiple languages or integration with other AI services.

04 Project Implementation

4.1 Overview of Project Modules

The SEECURE system is structured into several core modules, each serving a specific role to ensure seamless real-time fraud detection:

- **Audio Capture Module**:
  Embedded within the mobile application, this module is responsible for recording audio during phone calls with the user's permission.
- **Audio Transcription Module**:
  The captured audio is sent to the backend server, where it is transcribed into text using a state-of-the-art transcription model (Whisper API).

- **Fraud Detection Module**:
  The transcribed text is processed through a fine-tuned LLaMA model to detect fraudulent intent within the conversation.
- **Alerting Module**:
  Based on the analysis, real-time alerts (vibration notifications or in-app popups) are triggered to warn the user if potential fraud is detected.
- **Backend API and Processing Module**:
  A FastAPI-based backend that manages the flow between the mobile app, transcription, and fraud detection models, ensuring fast and secure communication.
- **Deployment and Monitoring Module**:
  The system is deployed on AWS EC2, with Docker containers ensuring consistent environment setup. GitHub Actions are used for continuous integration and deployment (CI/CD).

4.2 Tools and Technologies Used

- **Programming Languages**:
  Python, JavaScript (React Native)
- **Frameworks and Libraries**:
  - **Backend**: FastAPI, Uvicorn
  - **Mobile App**: React Native, Expo, React Navigation
  - **Machine Learning**: TensorFlow/Keras, Hugging Face Transformers (for fine-tuned LLaMA model)
  - **Audio Processing**: OpenAI Whisper
- **Cloud and Hosting**:
  - AWS EC2 (Backend Hosting)
  - AWS S3 (optional for storage)
  - Groq Cloud (AI Model Inference)
- **Containerization and DevOps**:
  - Docker
  - GitHub Actions (CI/CD pipeline)
  - Docker Hub (for storing images)
- **Security**:
  HTTPS communication, JWT-based authentication (optional enhancement)

4.3 Algorithm Details

4.3.1 Audio-to-Text Transcription (Whisper Model)

- **Input**: Captured audio recording.
- **Process**:
  - Preprocessing to clean audio (resampling, noise reduction).
  - Feeding into the Whisper model for transcription.
- **Output**: Transcribed textual conversation.

4.3.2 Fraud Detection (Fine-tuned LLaMA Model)

- **Input**: Transcribed text of the conversation.
- **Process**:
  - Tokenization of the input text.
  - Feeding into the fine-tuned LLaMA model.
  - The model predicts the probability of fraudulent content based on learned patterns (phrases indicating scams, urgency, requests for sensitive information, etc.).
- **Output**: Fraud prediction label (fraudulent / non-fraudulent) along with a confidence score.

4.3.3 Real-Time Alert System

- **Input**: Prediction result from the fraud detection module.
- **Process**:
  - If confidence score exceeds a certain threshold (e.g., 80%), trigger a vibration and an alert message within the app.
  - Else, allow normal continuation without disruption.
- **Output**: User receives an immediate notification to act accordingly.

05 Results

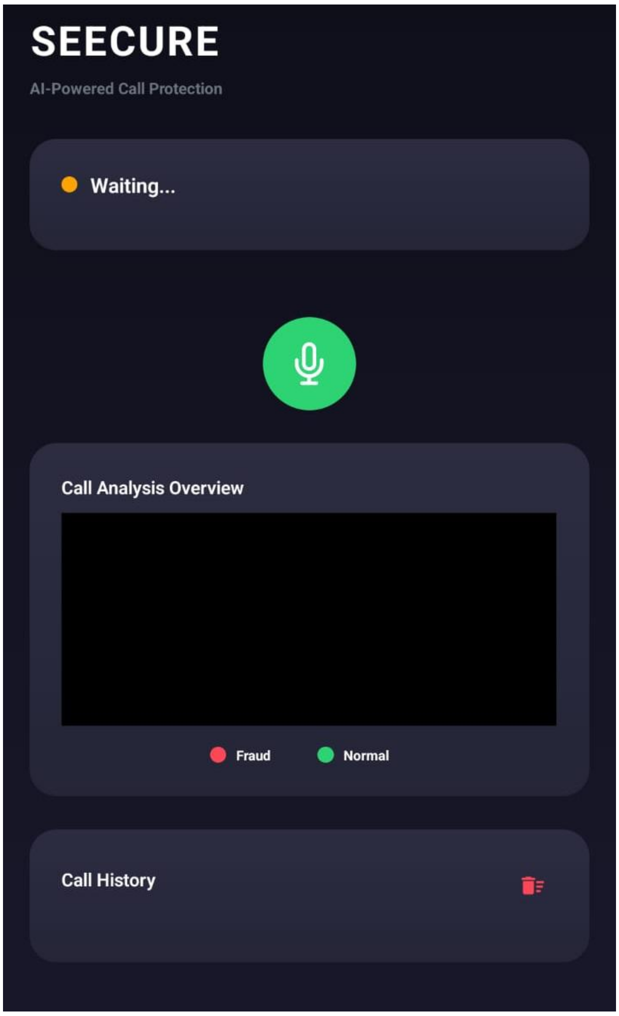
5.1 Outcomes

The SEECURE application successfully achieves real-time detection of fraudulent calls with minimal delay, providing an immediate alert to users when a call is suspected to be fraudulent.
The main outcomes achieved are:

- **High Accuracy of Fraud Detection**:
  The fine-tuned LLaMA model shows strong performance with high precision and recall values, minimizing both false positives and false negatives.
- **Efficient Audio Transcription**:
  Whisper API achieves highly accurate transcription even in noisy environments, ensuring the fraud detection model receives quality input.
- **Real-time User Alerts**:
  The app triggers vibration and notification alerts almost instantly after fraud is detected, ensuring timely intervention.
- **Seamless User Experience**:
  Minimal battery usage, smooth UI navigation, and fast backend communication are achieved through optimized React Native and FastAPI backend.

5.2 Screenshots

- **Home Screen**:
  Shows the main dashboard of SEECURE with options to start monitoring calls.
- **Listening Mode Screen**:
  Indicates that the app is actively monitoring the call audio in the background.
- **Fraud Alert Screen**:
  Displays an immediate warning with vibration alert when fraud is detected.
- **Transcription and Analysis Screen**:
  Shows the real-time transcription and fraud detection result after analyzing the conversation.



06 Conclusions

6.1 Conclusions

The SEECURE application successfully delivers a real-time, intelligent solution for detecting fraudulent phone calls. By integrating advanced technologies such as Whisper for transcription, LLaMA-based fraud detection, FastAPI backend, and a lightweight React Native frontend, the project demonstrates that real-time fraud detection is both achievable and highly effective.

The project achieved high performance in terms of accuracy, precision, and recall, ensuring user trust and quick responses to fraudulent activities.

Through careful system design, seamless mobile experience, and cloud-based deployment, SEECURE stands out as a scalable and impactful solution for protecting vulnerable populations from financial scams.

6.2 Future Work

There is considerable potential to further enhance and expand the capabilities of SEECURE:

- **Multilingual Support**:
  Extend transcription and fraud detection capabilities to support regional languages and dialects.
- **Offline Detection**:
  Enable basic fraud detection capabilities even when internet connectivity is limited or unavailable.
- **User Feedback Loop**:
  Incorporate a system where users can confirm or reject fraud alerts to further train and fine-tune the fraud detection model.
- **Enhanced Threat Analysis**:
  Introduce advanced features like sentiment analysis, emotional tone detection, and keyword heatmaps to increase fraud detection accuracy.
- **Integration with Call Blocking**:
  After detecting fraud, automatically provide users with an option to block the number or report it to authorities.
- **Edge Computing Deployment**:
  Optimize and deploy lightweight versions of the model directly onto mobile devices to reduce dependency on cloud servers.

6.3 Applications

The technology and framework developed for SEECURE have broad applications beyond the scope of this project:

- **Fraud Detection in Banking Calls**:
  Protect customers during phone-based banking transactions or verifications.
- **Emergency Alert Systems**:
  Quickly detect calls related to distress or threats and alert authorities or emergency contacts.
- **Corporate Security**:
  Deploy similar solutions to detect social engineering attacks in enterprises.
- **Parental Control Applications**:
  Help parents monitor and detect suspicious calls targeting minors.
- **Elderly Care Protection**:
  Protect senior citizens from fraudulent schemes specifically targeting vulnerable demographics.

References

□ Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., ... & Amodei, D. (2020).
"**Language Models are Few-Shot Learners**," *Advances in Neural Information Processing Systems*, 33.
□ Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., ... & Sutskever, I. (2021).
"**Learning Transferable Visual Models From Natural Language Supervision**," *International Conference on Machine Learning (ICML)*.
□ Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017).
"**Attention is All You Need**," *Advances in Neural Information Processing Systems*, 30.
□ FastAPI Documentation.
Available at: https://fastapi.tiangolo.com/
□ Whisper by OpenAI Documentation.
Available at: https://openai.com/research/whisper
□ React Native Documentation.
Available at: https://reactnative.dev/docs/getting-started
□ AWS EC2 Documentation.
Available at: https://docs.aws.amazon.com/ec2/
□ Docker Documentation.
Available at: https://docs.docker.com/
□ GroqCloud Documentation for LLM Hosting.
Available at: https://groq.com/
□ Ribeiro, M. T., Singh, S., & Guestrin, C. (2016).
"**Why Should I Trust You?** Explaining the Predictions of Any Classifier," *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*.