PARSHWANATH CHARITABLE TRUST'S
A.P. SHAH INSTITUTE OF TECHNOLOGY
Department of Computer Science and Engineering
Data Science

ParentShieldAI

Student ID	Group Member
23107047	Nishigandha Sawant
23107041	Nidhi Shettigar
23107004	Abhishek Sali

Project Guide
Ms. Poonam Pangarkar

Outline

- Introduction
- Literature Survey of the existing systems
- Limitations of the existing systems
- Problem statement
- System Design
- Technologies and methodologies
- Implementation
- Conclusion
- References

Sustainable Development Goals (SDG) Mapped

Our project, ParentShield.ai, contributes to the following SDGs:

SDG 4: Quality Education

Mapping: Promotes lifelong learning by enhancing the digital literacy of parents through our accessible "Digital Safety Library.“

SDG 16: Peace, Justice and Strong Institutions

Mapping: Contributes to justice by providing a tool that empowers users to actively combat digital crime and fraud, creating a more trustworthy online environment.

Introduction

The New Online Reality

Parents across India are deeply integrated into the digital world, relying on web platforms for everything from family communication to managing their finances.

A Two-Pronged Problem

This has exposed them to a constant barrage of sophisticated threats that arrive through two main channels:

- **Malicious Messages:** Phishing texts, fake job offers, and scam links sent directly via SMS and WhatsApp.
- **Fraudulent Transactions:** Deceptive UPI requests and unauthorized payment patterns designed for direct financial theft or otp requests.

The Consequence

The result is not just significant financial risk, but also the stress and uncertainty of navigating a digital world filled with potential scams.

Introduction

Objectives of ParentShieldAI

Deploy a Unified Web Platform

- To build and deploy a responsive, on-demand application using a **React.js** frontend and a **Python (Flask/FastAPI)** backend API.

Detect Multilingual Message Fraud

- To implement an intelligent system using **Pytesseract** (OCR) to extract text (including Hindi & Marathi) from screenshots, analyzed by an **SVM model** (trained with **TF-IDF & n-grams**) to identify scams, phishing, and suspicious links.

Predict Hybrid Transaction Fraud

- To deploy a hybrid ML model combining an SVM with **rule-based pattern matching** to flag fraudulent payment URLs, suspicious financial terms, and urgency language in transaction data.

Empower Users with Actionable Insights

- To enhance user safety via a "Digital Safety Library" with **interactive quizzes**, educational videos, and a **downloadable analysis report generator** that provides clear, visual summaries of detected threats.

Foster Continuous Improvement

- To integrate a **Flask-based feedback API**, allowing users to report new threats and contribute to the continuous retraining and improvement of the ML models.

Literature Survey of the existing system

Sr. No.	Title	Author(s)	Year	Outcomes	Methodology	Result
1.	[1] Phishing-Attack-Detection Model Using Natural Language Processing and Deep Learning	Sánchez-Paniagua, M., Fernández-Manzano, E.	2023	The objective was to develop a phishing detection model that considers sequential relationships in suspicious messages to improve accuracy beyond traditional bag-of-words approaches.	Deep Learning algorithms with NLP techniques were used to analyze the textual content of suspicious messages and apps while preserving word sequence and context information.	Results showed improved phishing detection accuracy by maintaining the intrinsic richness of relationships between words, proving more effective than non-sequential text analysis methods.
2.	[2] Secure UPI: Machine Learning-Driven Fraud Detection System for UPI Transactions	IEEE Conference Publication	2024	To develop an advanced fraud detection system specifically for UPI transactions using ensemble machine learning techniques to enhance precision and reduce false positives.	Used the Classifier machine learning algorithm to analyze transaction features including amount, frequency, merchant patterns, and user behavior to identify fraudulent activities.	Classifier proved highly effective for fraud detection, achieving high precision in identifying fraudulent UPI transactions while minimizing false alarms for legitimate users.

Literature Survey of the existing system

Sr. No.	Title	Author(s)	Year	Outcomes	Methodology	Result
3.	[3] Chat Analysis and Spam Detection of WhatsApp using Machine Learning	ResearchGate Publication	2023	The study aimed to develop an automated system for detecting spam and malicious content in WhatsApp group conversations, addressing the growing problem of scam messages on the platform.	Machine learning classification algorithms were applied to analyze WhatsApp chat patterns, message content, and user behavior. The system leveraged these features to distinguish between legitimate and spam messages.	The model demonstrated effective spam detection capabilities in WhatsApp conversations, providing a foundation for real-time protection against phishing and scam messages on messaging platforms.

Limitations of existing systems

1. Analysis of Key Limitations

•Passive & Not On-Demand: Educational sites offer general advice but lack any tool for a user to check a *specific, active threat*.
•"Black Box" Operations: Automated filters and bank systems often block threats without any explanation, preventing user learning.
•Fragmented Tools: Parents lack a single, trusted platform to verify different types of suspicious activity (e.g., messages and transactions).

2. Our Justification

Yes, we are confident we can address these issues. ParentShield.ai is uniquely designed to:

- Provide Simple Verdicts: We empower users with clear 'Safe' or 'Unsafe' answers through on-demand analysis.
- Unify Key Tools: We combine both message and transaction analysis in a single, easy-to-use platform.
- Offer Educational Resources: We provide a dedicated library of safety lessons, allowing users to proactively learn about the types of threats our tool detects.

Problem statement

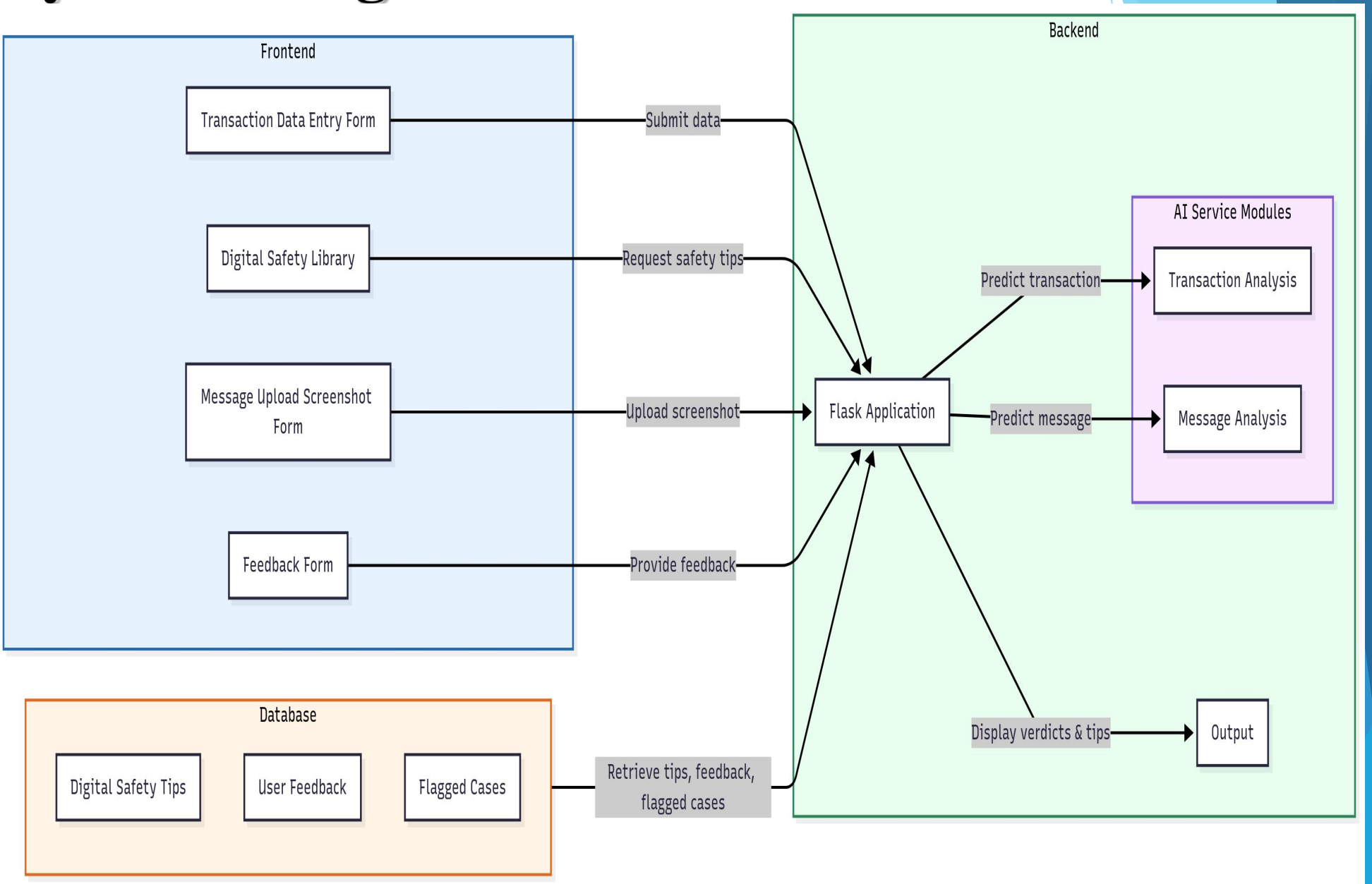
1. Problem Statement

•Target: Indian parents who are increasingly online but remain vulnerable to digital fraud.
•The Gap: A lack of a single, simple tool for on-demand verdicts on suspicious messages or transactions.
•Our Solution (ParentShield.ai): A unified web platform for straightforward threat detection, supported by a library of safety lessons.

2. Abstract Architecture & Flow

•Back-End (Python API): The server receives the request and calls the appropriate AI model for analysis.
•AI Models (ML/NLP): The model processes the data and returns a clear verdict (e.g., "Safe Message," "Fraud Flagged").
•Front-End (React UI): A user uploads a screenshot or enters transaction data.
•Result: This direct verdict is sent back and displayed to the user on the front-end.

System Design



Technologies

Frontend:

- React.js: Component-based UI library (Virtual DOM).
- TailwindCSS: Modern utility-first CSS framework for responsive design.
- React Router: Handles client-side navigation.
- Clerk: Manages user authentication and sign-in.

Backend:

- Python: High-level language for ML integration.
- FastAPI / Flask: Lightweight frameworks for building RESTful APIs.

Machine Learning & Data:

- Scikit-learn: Core library for ML models, preprocessing, and metrics.
- Support Vector Machine (SVM): Primary ensemble model for fraud classification.
- Naive Bayes: Secondary model for result verification.
- Pandas: Used for dataset creation and data manipulation.
- TF-IDF Vectorizer: Converts text messages into numerical features.

Image Processing & OCR:

- OpenCV (cv2): Used for image preprocessing (noise reduction, grayscale, blur).
- Pillow (PIL): Image loading and preparation.
- Pytesseract: OCR tool for extracting text (including Hindi/Marathi) from screenshots.
- Joblib: Model serialization for saving/loading trained models.

Methodology

Data Acquisition & Preparation:

- Collected message, transaction, and image data for dual fraud detection.
- Preprocessed text using lowercase conversion and stopword removal.
- Created separate labeled datasets for message and transaction fraud.

Feature Engineering & Selection:

- Text Features: TF-IDF with n-grams (unigrams, bigrams, trigrams).
- Transaction Features: Keywords (UPI, KYC, OTP), URLs, urgency words.
- Pattern Recognition: Suspicious links, threat language, financial terms.
- Custom risk factor algorithm combining multiple fraud indicators.

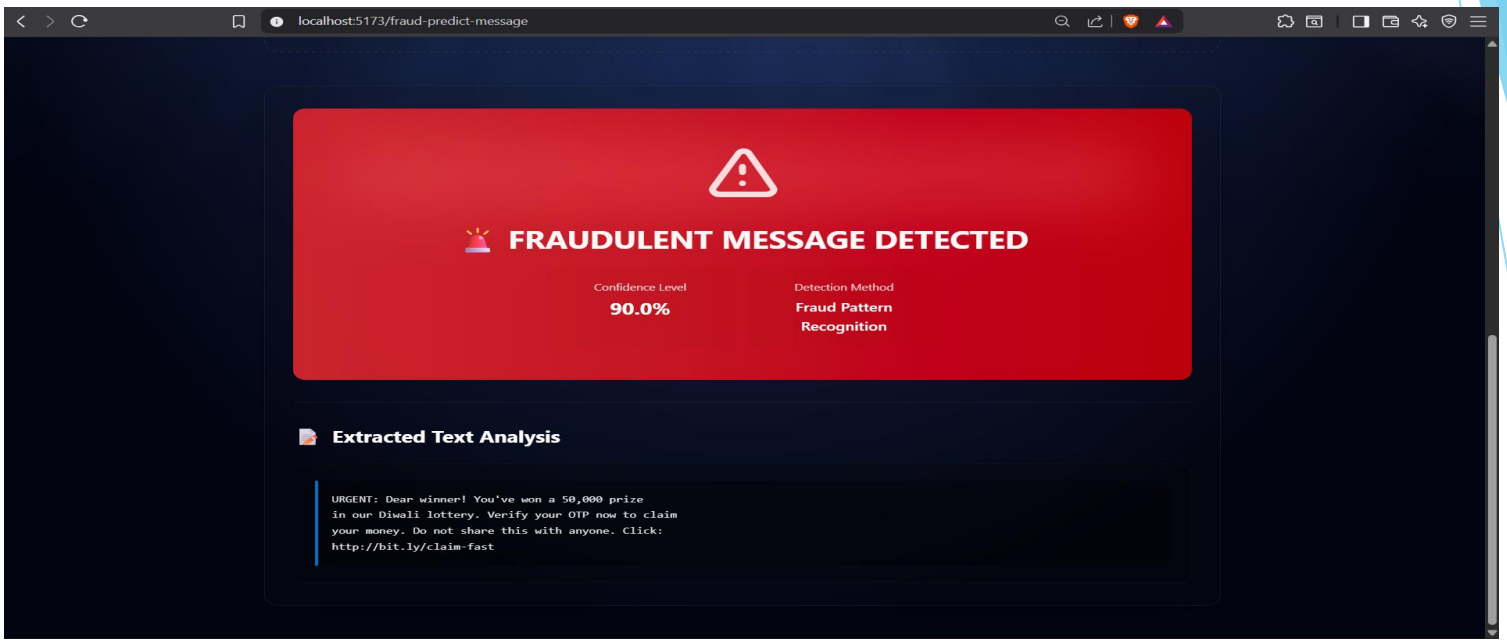
Model Training & Evaluation:

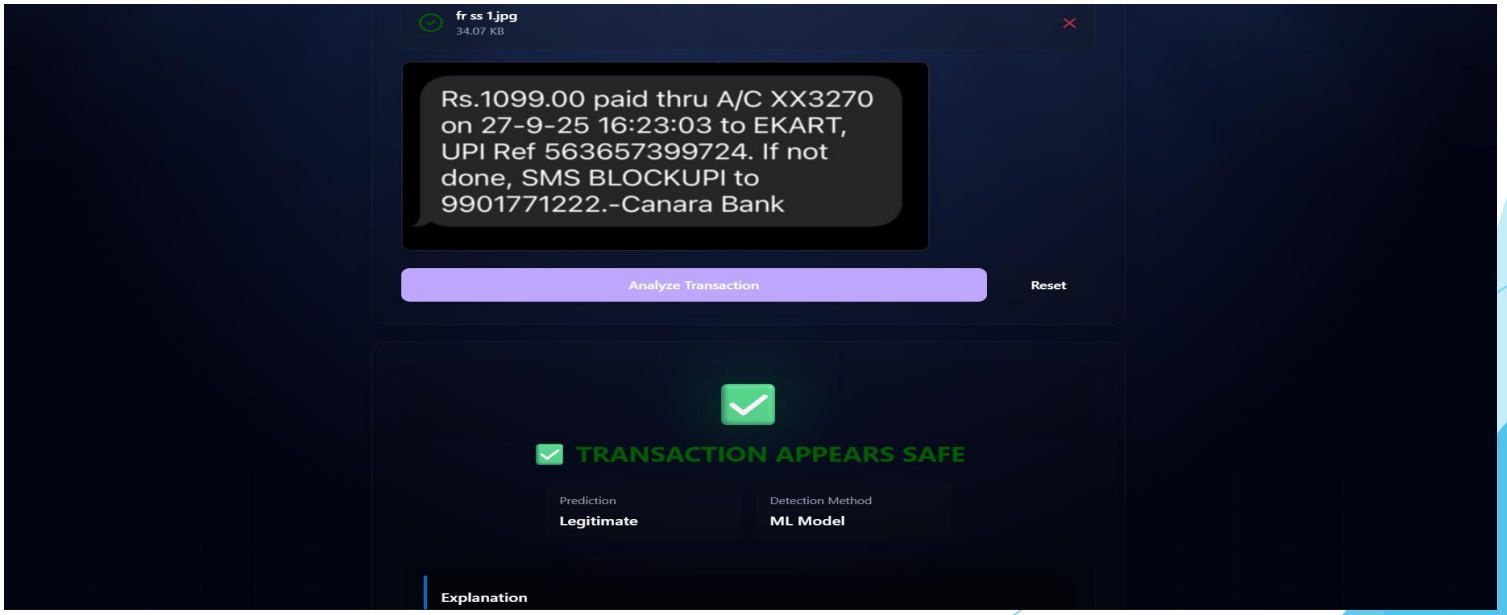
- Trained separate Support Vector Machine (SVM) models for message and transaction fraud.
- Applied 80-20 train-test split with stratified sampling for balance.
- Used class weighting to handle imbalanced fraud datasets.
- Evaluated model performance using cross-validation, accuracy, and false-positive rate.

Deployment & Integration:

- Deployed dual models as REST API using Flask / FastAPI frameworks.
- Implemented hybrid detection: Rule-based + ML + Risk scoring.
- React.js frontend enables real-time analysis of messages/transactions.
- Pytesseract OCR extracts text from screenshots for analysis.

Implementation





Conclusion

Summary & Achievements

- Deployed a Unified Platform: Launched an integrated React & FastAPI application for on-demand fraud detection.
- Achieved Multilingual Analysis: Successfully deployed SVM models to detect fraud in English, Hindi, and Marathi using OCR and TF-IDF.
- Built an Interactive Safety Hub: Added a "Digital Safety Library" complete with interactive quizzes, videos, and data visualizations.
- Generated Downloadable Reports: Implemented an analysis report generator for clear, actionable user insights.

Limitations & Future Scope

- Implement Continuous Learning: Integrate a user feedback loop for model retraining and explore advanced models (LSTMs, Transformers).
- Expand Language & Format Support: Add more regional languages (e.g., Hinglish, Bengali) and analyze new formats like voice notes or video scams.
- Develop Proactive Alerts: Evolve from a manual tool to a real-time browser extension or mobile app for instant warnings.

References

[1] Sharma, A., Gupta, R., & Chen, L. (2024). "Advancements in NLP for Real-Time Phishing and Smishing Detection in Messaging Apps." IEEE Transactions on Information Forensics and Security 19.
https://ieeexplore.ieee.org/document/9399534

[2] Patel, S. K., & Krishnan, M. (2023). "A Comparative Analysis of Machine Learning Algorithms for UPI Transaction Fraud Detection." International Journal of Computer Science and Engineering 11.
https://www.kaggle.com/code/kartikdhingra/upi-fraud-detection

[3] Smith, R. (2007). "An Overview of the Tesseract OCR Engine." Proceedings of the Ninth International Conference on Document Analysis and Recognition (ICDAR).
https://github.com/tesseract-ocr/tesseract

Thank You...!!