Project name - AI-Based Spam and Phishing Detection System

Team name - identifiers

Member name -

Name : NItesh Shedge
roll no :47
Mail- niteshshedge011@gmail.com

Name : Dishant chikhale
roll no :06
Mail- dishantchikhale1405@gmail.com

—----------

internship program: Digisuraksha Parhari Foundation
Powered by: Infinisec Technologies Pvt. Ltd. Submissio

Introduction

Background

As the digital age advances, cyber threats such as spam and phishing have grown exponentially. These malicious communications can lead to loss of sensitive data, financial fraud, and severe privacy breaches. They exploit trust, language, and user inexperience, making them difficult to detect manually. This report discusses a smart, AI-based system for detecting such threats effectively.

Project Overview

This project introduces **AI Security Detector**, a web application designed to identify spam and phishing messages using AI. Built with multilingual support (English, Hindi, Marathi), it offers real-time analysis of user input. The system integrates Google’s Gemini AI to classify messages and provide user guidance.

Objectives and Scope

Primary Objectives

* Automatically classify messages into spam, phishing, or clean.
* Offer multilingual analysis (English, Hindi, Marathi).
* Analyze linguistic and structural features of messages.

* Present results with high clarity and user understanding.

Project Scope

* Targets individual users, educational institutions, and businesses.
* Assists non-expert users in detecting potential cyber threats.
* Flexible integration into existing systems.

System Architecture Overview

System Components

1. **Frontend Interface**: Developed with HTML, CSS, JavaScript.
2. **Backend Service**: Powered by Node.js and Express.js.
3. **AI Integration**: Connected to Google’s Gemini model via SDK.

Architecture Flow

* User inputs suspicious message.
* Message is sent to the backend via REST API.
* Gemini AI processes the message.
* JSON response is interpreted and displayed with clarity.

Frontend Design

User Interface Elements

* Clean card layout for message input.
* Submit button with emoji-enhanced visuals.
* Animated spinner and progress bar during AI analysis.
* Result box with status, language, confidence, reasons, and safety tips.

Technologies Used

* **HTML5**: Structure
* **CSS3**: Styling and responsiveness
* **JavaScript**: Dynamic interaction and data handling

Backend Logic and API Design

Node.js Backend

* Uses Express.js to create the server and routing.
* Route `/api/detect` handles POST requests.
* Communicates with Gemini AI using GoogleGenerativeAI SDK.

Error Handling

* Try-catch blocks for stability
* User feedback on network/API failures

API Response Format

* JSON structure with multiple flags: is_spam, is_phishing, confidence, language, etc.

AI Integration with Gemini

Model: Gemini-1.5-Flash

* Chosen for its speed and accuracy.
* Provides structured insights including message intent and embedded risks.

Prompt Engineering

* Prompts crafted to analyze tone, keywords, and suspicious URLs.
* Detects impersonation, urgency cues, and reward tactics.
* Output format designed to be easily parsed by the frontend.

Sample Detection Workflow

Input

User submits: "You have won 1 lakh rupees! Click bit.ly/claim-prize now."

Output (JSON)

```json
{
  "is_spam": true,
  "is_phishing": true,
  "confidence": 93,
  "language": "en",
  "reasons": ["Shortened URL", "Prize scam language"],
  "suggestions": ["Do not click links", "Report as phishing"]
}
```

Display

This data is formatted and displayed to the user in real time.

Security and Usability Features

Key Features

* **Real-time detection**

* **Visual progress feedback**
* **Detailed classification breakdown**
* **Multilingual message support**
* **Structured safety suggestions**

Usability

* No account or login required
* Accessible on mobile and desktop browsers

Language and Localization

Supported Languages

* English
* Hindi
* Marathi

Language Detection

* AI detects language using NLP techniques.
* Results translated if needed for user clarity.

User Interface Screenshots (Descriptions)

* **Home Page**: Title, input area, submit button
* **Loading State**: Spinner and animated progress bar
* **Results Section**: Risk level, confidence score, detected language, detailed reasons

Testing and Evaluation

Testing Strategy

* Manual testing with diverse message samples
* Edge cases: clean messages, promotional messages, clear phishing traps

Evaluation Metrics

* Accuracy
* False positives and false negatives
* Speed of response

Limitations and Challenges

Current Limitations

* Heavily dependent on AI model accuracy
* Cannot detect attachments or media threats

* Network latency affects speed

Challenges Faced

* Prompt tuning for multilingual support
* Handling ambiguous language
* Balancing simplicity with technical detail in UI

Future Scope

Planned Enhancements

* Upgrade to Gemini-1.5-Pro
* User feedback system
* History tracking
* Admin dashboard for organizations
* Integration with email clients and messaging apps

Ethical Considerations

* Ensures user privacy: no data stored without consent
* Transparent analysis logic
* Open-source model plans for academic collaborations
* No profiling or ad targeting from user inputs

Conclusion & Team Contributions

Conclusion

The AI Security Detector serves as a powerful tool in the battle against phishing and spam. With real-time multilingual analysis, intuitive UX, and scalable architecture, it equips users with critical insights to avoid digital traps.

Team Roles

* **Frontend Engineer**: UI development and animations
* **Backend Developer**: API and AI connectivity
* **Prompt Engineer**: AI behavior optimization

Acknowledgments

* Google’s Gemini team for API resources
* Faculty/mentors for guidance
* Cybersecurity communities for open datasets and support