SenseFlow: Phishing & Social Engineering Detector Using Semantic Intent Analysis

Problem Statement

- 1. The Failure of Conventional Defense Current email security relies on **static signature matching** and keyword filtering. These systems fail against modern threats because they only look for known "bad words" or malicious links, missing the context entirely.

- 2. The Evolving Threat: "Payload-Free" Attacks Attackers have shifted to Business Email Compromise (BEC)—sophisticated, "clean" emails that contain no malware. Instead, they leverage **psycholinguistic vectors** (like manufactured urgency or authority bias) to trick the user directly.

- 3. The Critical Gap Since these attacks lack technical signatures, they bypass firewalls. There is an imperative need for a system capable of **Semantic Intent Analysis** to detect these invisible, context-driven threats.

Technologies/Tools to be Used

1. AI Libraries & Frameworks:
   - Hugging Face Transformers,
   - LIME (Library)
   - PyTorch

2. Programming & Data Tools
   - Python 3.9+
   - Pandas & NumPy
   - Regex

3. Development Environment
   - VS Code
   - GitHub

4. Interface
   - Streamlit

Expected Outcome

1. Software Application (Prototype)
   - A fully functional Web-Based Security Dashboard (built with Streamlit) where users can paste emails for real-time scanning.
   - It will output a Risk Score and a Visual Heatmap showing exactly which words triggered the alert.

2. Intelligent AI Model
   - A fine-tuned BERT Transformer Model capable of detecting Psychological Triggers (Urgency, Authority) that traditional antivirus software misses.

3. Research Findings
   - A Comparative Analysis Report proving that SenseFlow successfully detects "Clean" attacks (no malicious links) which failed to be caught by standard Keyword/Regex Filters.

4. Visualization Tool
   - An Explainable AI (LIME) Module that highlights suspicious words in RED, providing transparency on why the AI flagged the email.

Existing Technologies/Methods

1. Signature-Based Detection (e.g., Traditional Antivirus)
   - Approach: Scans incoming emails against a database of known malware file hashes and blacklisted URLs.
   - Limitation: It is effective against known viruses but fails completely against "Zero-Day" attacks or text-only emails that contain no malicious attachments.

2. Keyword & Heuristic Filters (e.g., SpamAssassin)
   - Approach: Assigns a "Spam Score" based on fixed rules (e.g., +5 points if the subject contains "Winner" or "Free").
   - Limitation: These are Context-Blind. Attackers easily bypass them by changing their vocabulary (e.g., replacing "Wire money" with "Process payment") or using professional business language.

3. Sender Authentication Protocols (SPF, DKIM, DMARC)
   - Approach: Verifies the sender's IP address and domain ownership to prevent spoofing.
   - Limitation: It authenticates the machine, not the intent. It cannot detect Business Email Compromise (BEC), where a legitimate, authenticated account has been hacked and is used to send threats.

Innovation & Difference

1. Shift from Syntax to Semantics
   - Existing Solution: Traditional filters analyze Syntax (keywords like "Lottery" or "Click Here").
   - Our Innovation: SenseFlow uses Transformers (BERT) to analyze Semantics (Meaning). It understands the relationship between words—for example, it detects that a request for "gift cards" coming from a "CEO" context is suspicious, even if the words themselves are safe.

2. Explainable AI (XAI) Integration
   - Existing Solution: Most AI security tools are "Black Boxes"—they block an email without explaining why.
   - Our Innovation: We integrate LIME (Local Interpretable Model-agnostic Explanations) to generate a transparency report.
   - This visualizes exactly why the email was flagged (e.g., highlighting "Urgent wire transfer" in RED), turning the system into an educational tool for the user.

3. Psycholinguistic Feature Detection
   - Existing Solution: Standard tools look for Technical Anomalies (bad IPs, malware hashes).
   - Our Innovation: SenseFlow is specifically fine-tuned to detect Human Manipulation Patterns (Psycholinguistic vectors). It scans for specific triggers like Manufactured Urgency, Authority Bias, and Scarcity, which are the hallmarks of modern Social Engineering.

Expected progress for Review - 1

Phase I: Progress & Logic Review (from Feb 16, 2026)

Goal:

- To demonstrate that the SenseFlow architectural foundation is functional, specifically proving that the Data Preprocessing Pipeline is operational and establishing a Baseline Failure Benchmark to justify the shift to Semantic Analysis.

Status Report:

- Planned: Literature survey and initial dataset selection.
- Actual: Implementation of the Data Preprocessing Pipeline (cleaning text, removing HTML, and masking PII).

Proof of Work:

- GitHub Repository: Initialized with core project structure and a functional "Benchmark Failure Script".
- Data Artifacts: A standardized processed_data.csv containing sanitized training samples.

Trial & Error Log:

- Initial Test: Evaluated Naive Bayes and Keyword Matching on "clean" phishing samples.
- The Pivot: Switched to RoBERTa-based Transformers after discovering that traditional models failed to capture psychological intent.

Domain Expectations (Machine Learning):

- Data Engineering: Completion of automated tokenization and normalization.
- Metrics: Established baseline accuracy and loss targets for model fine-tuning.

Expected progress for Review - 2

Phase II: Prototype & Integration Review (from March 23, 2026)

Goal:

- To present a fully integrated, 90% functional prototype where the Fine-Tuned BERT Model successfully communicates with the Streamlit Web Dashboard to detect and explain phishing threats in real-timeFunctional Prototype: Live demo of the project.

Functional Prototype:

- Live Demo: A working Streamlit Web Application where users can input email text for real-time analysis.

Full Integration:

- System Architecture: The Fine-tuned BERT Model is fully connected to the LIME Explainer and the Streamlit frontend.

Performance Metrics:

- Final Accuracy: Achieving a target of >92% on previously unseen "Spear Phishing" data.
- UX Data: Optimized inference latency to under 1.5 seconds per scan.

Domain Expectations (Machine Learning):

- Optimization: The final model is fully optimized and tested on diverse real-world edge cases to ensure reliability.

Team Members

- Markandeyan Gokul (AIML)
  Project Lead (Model Architecture).

- Narlapati Venkata Mohana Krishna (Cyber Security)
  Security Analyst (Threat Research).

- Kanamarlapudi Sai Charithanjali (AIML)
  ML Engineer (Data & Optimization).