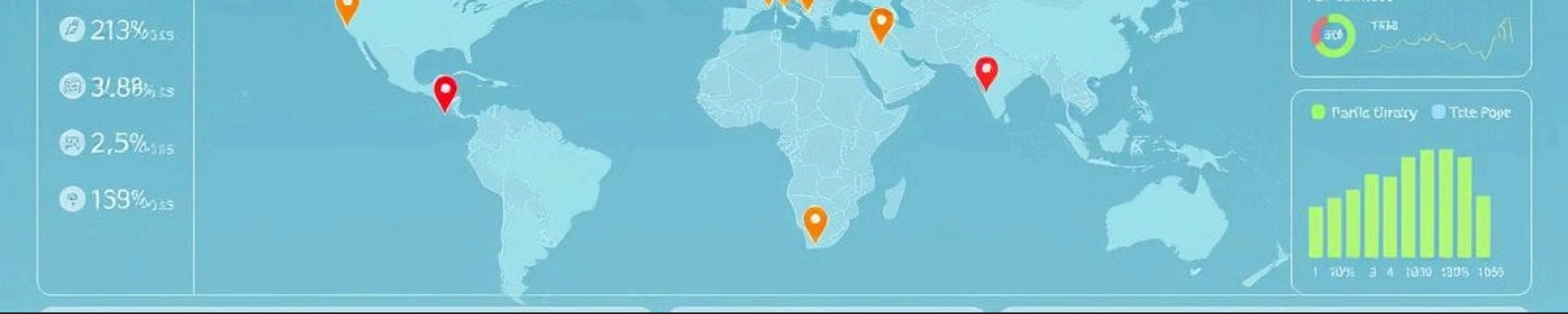
SMiShing & Scam Detector with Safe Browsing Signals

Revolutionizing cybersecurity with AI-powered detection and classification of SMiShing messages and scams.



Project Goals

1 **Detection & Classification**

Identify SMiShing messages and other scams with high accuracy.

2 **Threat Analysis**

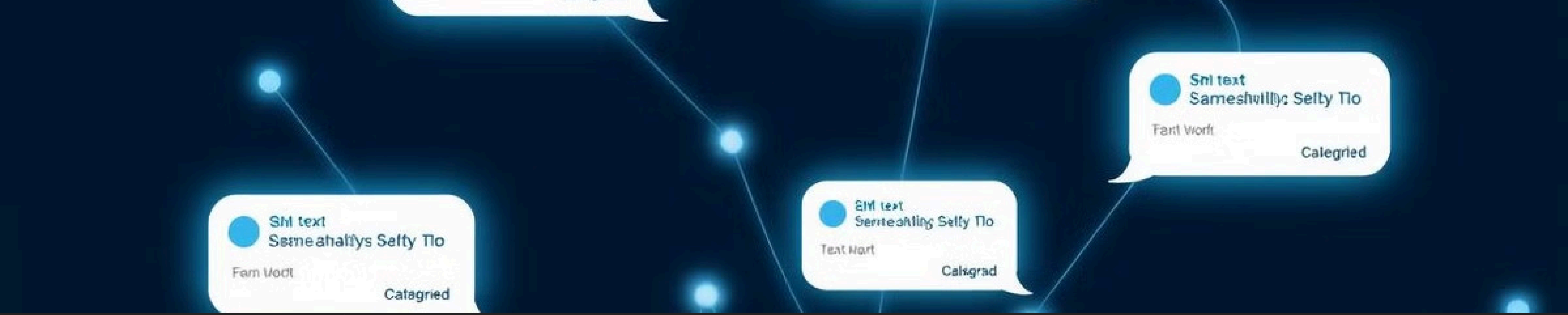
Analyze embedded URLs for phishing and malware threats.

3 **Multilingual and Image to Text Support**

Ensure effective detection in English and Spanish. Use OCR (On Character Recognition) to convert screenshots of texts (SMS) to text for analysis.

4 **Interactive Interface**

Provide an intuitive user experience via Gradio.



Data Collection & Exploration

Data Sources

UC Irvine SMS Spam collection, and manual examples.

URL Datasets

Malicious URLs via Google Safe Browsing and trusted sources.

Data Categories

SMiShing messages, other scams, and legitimate messages.

Approach to Achieve Goals

1

Problem Understanding

- People are inundated with SMS scams on a daily basis. These can result in account takeover, financial fraud, romance scams and other types of fraud.
- We began by clearly defining the problem—developing an AI tool that can detect SMiShing and scams with multilingual support, OCR (On Character Recognition) and explainable predictions.

2

Data collection

Gathered data of SMiShing, scam, and legitimate messages, as well as leveraging Google's Safebrowsing API, and OpenAI for analysis of malicious and safe links.

3

Model Design

To achieve the project's goals, we utilized a modular approach, where each functionality was designed as an independent component:

Message Classification, URL Threat Detection, OCR Integration , Explainability .

4

Testing & Evaluation

- **Model Performance Testing:**
  - **Scam tactics change frequently**, and training a fixed supervised model could become outdated.
  - **Zero-shot classification** allows the model to classify new scams without retraining.
    - Zero-shot classification is **a machine learning technique where a model can classify data into categories it has never seen before during training, relying solely on textual descriptions (our hypothesis) or other forms of auxiliary information to understand the new categories, essentially making predictions without any labeled examples for those specific classes**
    - This implementation **leverages NLI (Natural Language Inference)** by prompting the model to infer whether the given message fits one of the labels:
      - pipeline("zero-shot-classification", model="joeddav/xlm-roberta-large-xnli")
        - Unlike standard BERT-based models, **XLM-RoBERTa understands cross-lingual relationships**, making it **resilient to scam messages in different languages**.
      - **Benefits of using "joeddav/xlm-roberta-large-xnli":** joeddav/xlm-roberta-large-xnli" is considered excellent for zero-shot classification because **it is specifically fine-tuned on a multilingual Natural Language Inference (NLI) dataset called XNLI**, allowing it to effectively classify text across multiple languages without requiring any additional training data for new categories.
      - This enables our system to **detect new scam patterns dynamically** without collecting and labeling thousands of samples.
        - The **NLI-based classification** helps detect **implicit scam tactics**, even if they use:
          - **Social engineering techniques** (e.g., impersonation).
          - **Obfuscated or indirect scam wording**.
          - **Messages that don't contain explicit scam keywords**.

Effectiveness Check

1️⃣ Typical Zero-Shot Usage

- The code **calls** the zero-shot classifier:

```
pipeline("zero-shot-classification", model="joeddav/xlm-roberta-large-xnli")
```

- It **uses candidate labels**:

```
["SMiShing", "Other Scam", "Legitimate"]
```

- It applies a **hypothesis template**:

```
"This message is {}."
```

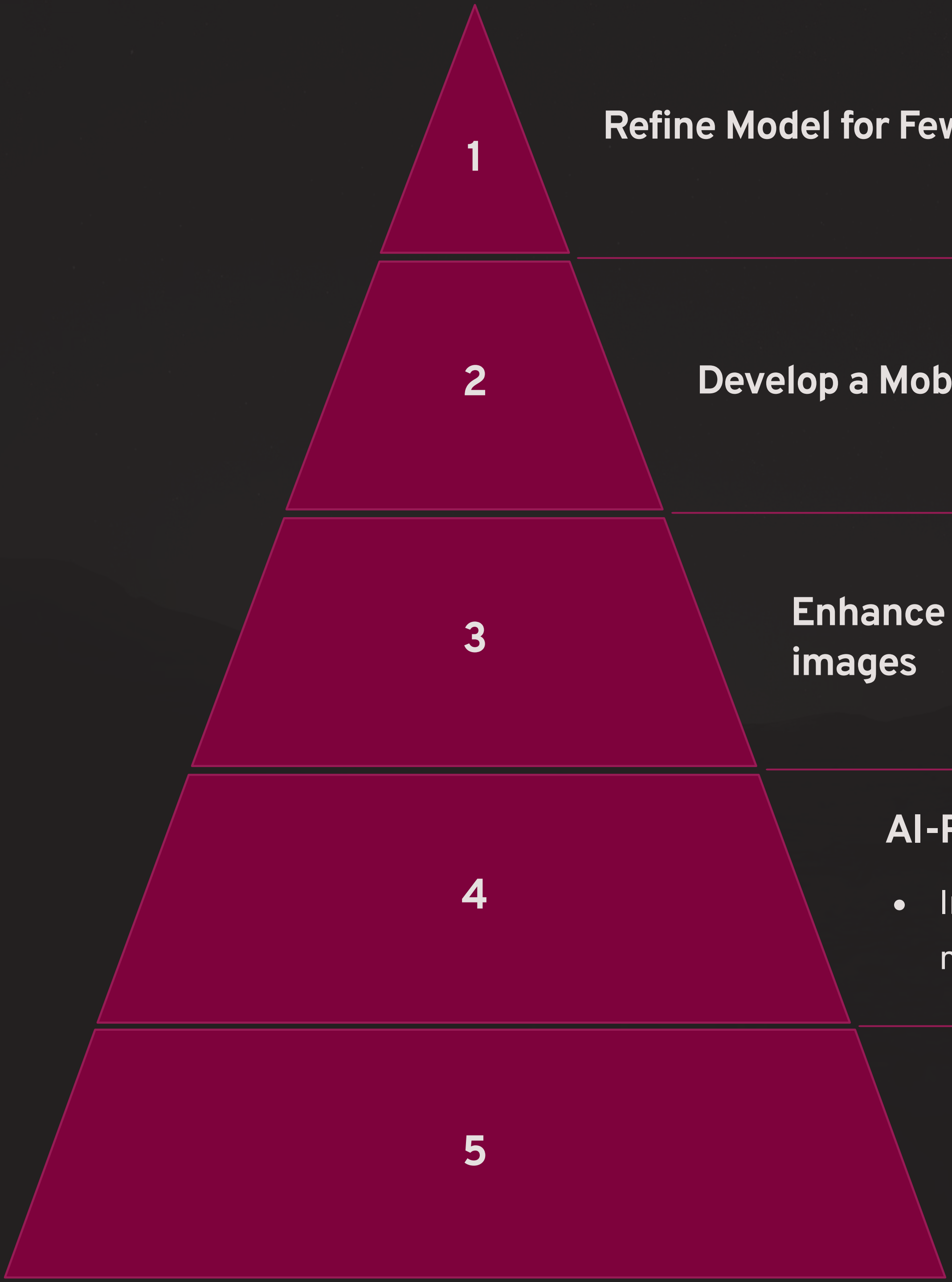
- **Component Testing:** Conducted unit testing for each functionality (e.g., message classification, URL analysis, OCR text extraction) to ensure reliability.
- **End-to-End Testing:** Tested the integrated system with real-world examples, including multilingual messages, embedded URLs, SMSishing, scam and legitimate messages.
- **Explainability Validation:** Verified the accuracy and clarity of SHAP-based explanations to ensure they matched human understanding.

5

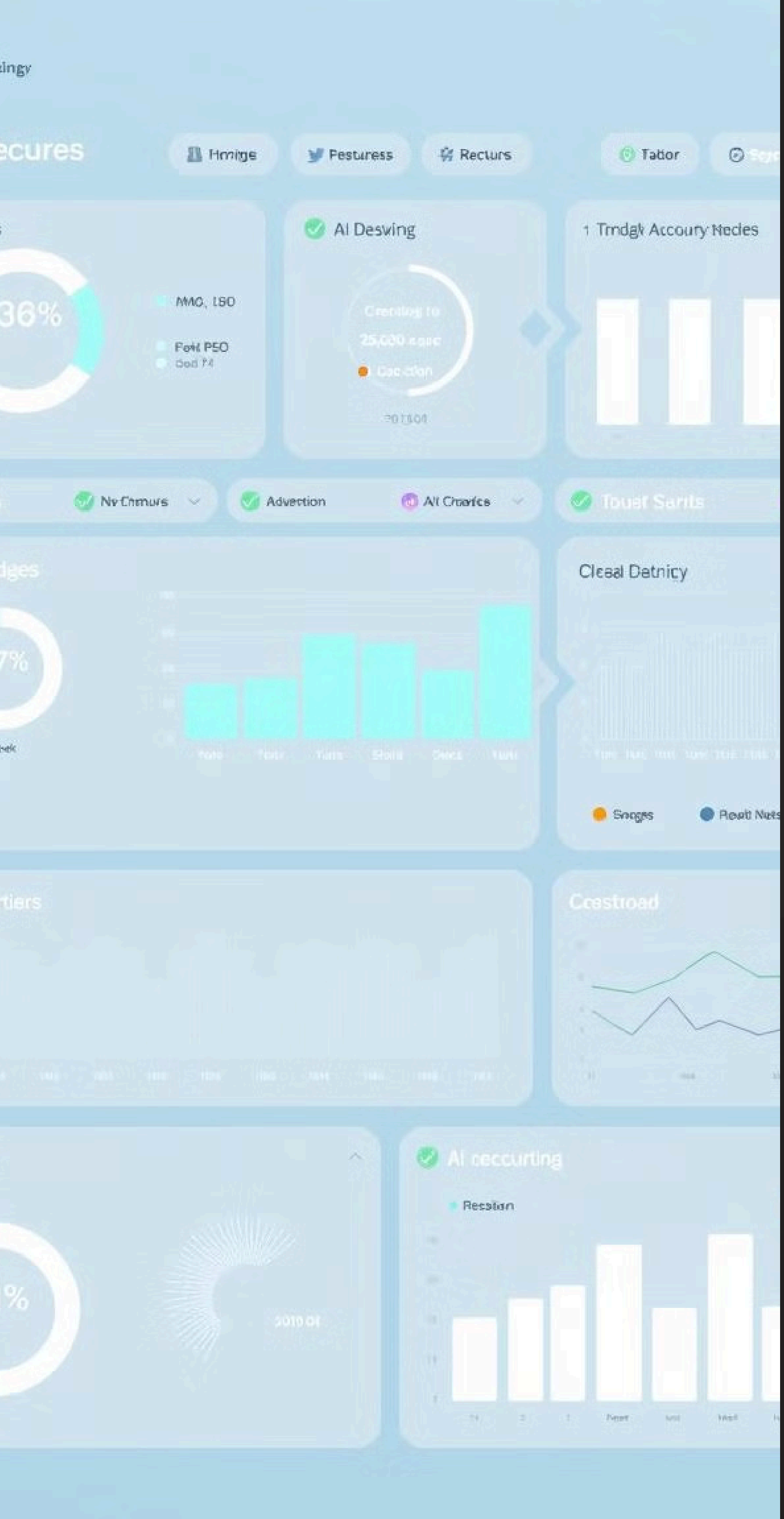
Deployment

Build Gradio interface and deploy on Hugging Face Spaces. API Keys were put into HuggingFace secrets.

Future Development Roadmap



1. **Refine Model for Fewer False Positives**
2. **Develop a Mobile App for Real-Time Detection**
3. **Enhance OCR performance for low-quality or distorted images**
4. **AI-Powered Scam Evolution Monitoring**
   - Implement **a dynamic model that updates itself** based on newly discovered scams to stay ahead of attackers.
5. **Dynamic Model Updates based on user feedback**



Results and Conclusions

Accurate Classification

100% accurate detection rates with keyword boosting.

Effective URL Analysis

High accuracy in identifying malicious URLs.

Robust Multilingual Support

Effective processing of English and Spanish messages.

Transparent Decisions

SHAP visualizations provide insights into AI reasoning.

By using **XLM-RoBERTa for zero-shot classification**, we achieved:

✅ **Multilingual support** to handle scams in different languages.

✅ **No need for labeled datasets**—it generalizes well to unseen scams.

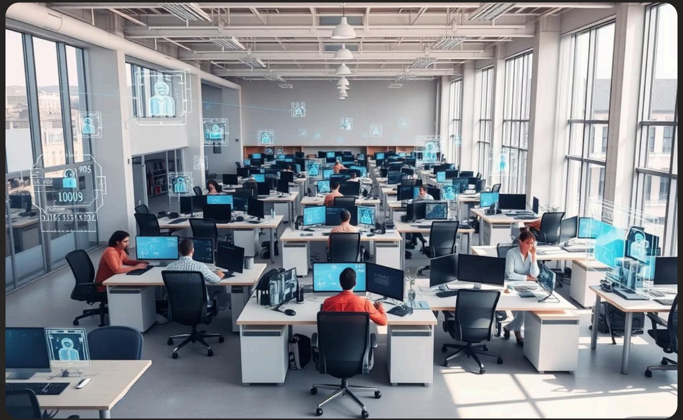
✅ **Better reasoning than keyword-based detection** for identifying hidden threats.

✅ **Fast, scalable, and cost-effective** compared to training a custom fraud model.

Practical Applications & Future Potential

Individual Protection

Verify suspicious messages before clicking potentially harmful links.



Business Security

Integrate tool to prevent phishing attacks targeting employees.

Security Research

Use transparent classification process to analyze emerging scam tactics.