ASSIGNMENT

NAME: Kara Swathi

BRANCH: CSE-AIML

COLLEGE: Dayananda Sagar University

COMPANY: Akaike

TOPIC: Email Classification for Support Team

Introduction to the Problem Statement

In current electronic communications, emails are often carrying Personally Identifiable Information (PII) in the form of names, email addresses, phone numbers, and financial information. Unencrypted PII is a security threat, which can result in data breaches, identity theft, and compliance issues.

To solve this problem, the PII Masking and Email Classification API is created to:

•Detect and mask PII for greater privacy protection.

•Classify emails into pre-defined categories based on content.

• Ensuring privacy law compliance while ensuring usability across different applications.

This API employs machine learning (ML) and natural language processing (NLP) to process emails automatically while keeping data confidential.

Methodology for PII Masking and Classification

1.PII Masking

The methodology for PII detection and masking involves:

• Regular Expressions (Regex): For identifying common patterns for names, emails, and phone numbers.

• Masking Mechanism: Replaced detected entities with generalized placeholders.

e.g., "John Doe" to [full_name] and "johndoe@example.com" to [email].

2.Email Classification

The classification part of the API does the following:

• Toketnization, cleaning and numerical representation of emails using TF-IDF vectorization.

• Model Prediction: Trained Naïve Bayes model predicts emails to be classified as incident, problem, request etc.

• The API also makes sure that its output strictly conforms to the expected JSON structure by sending both masked email text and classification label.

Model Training Details:

1.PII Detection Model

Define PII Patterns:

- Regex patterns (PII_FIELDS) assist in the detection of particular types of sensitive information, like:

Full Names: Matches first & last names in capitalized form (John Doe to [full_name])

Emails : Identifies correct email addresses (johndoe@example.com to [email])

Phone Numbers: Identifies standard 10-digit numbers (9876543210 to [phone_number])

Validate the Input:

- Safeguards against empty or malformed text prior to processing.

Apply Regex-Based Matching:

- Loops over defined patterns using re.finditer() to detect sensitive entities

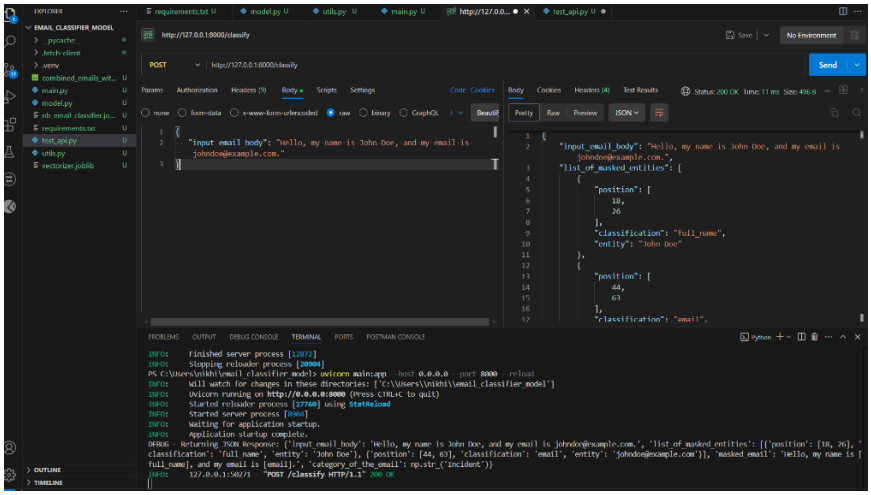
- Obtain start and end positions for masking

Replace Identified Entities with Placeholders:

- Each match is replaced with a masked identifier (e.g., [email], [full_name]).

- A structured list of masked entities is formed for tracking.

Return Masked Text & Entities List



Email Classification Model

- Algorithm: Multinomial Naïve Bayes (well suited to text classification).

•Feature Extraction: TF-IDF vectorization to map text to structured data.

•Training Dataset: Labeled emails that span types such as incident, request, problem,change.

•Evaluation Metrics: Accuracy score on test data using Scikit-Learn metrics.

•The trained model is saved using Joblib for deployment in FastAPI, ensuring efficient response time during classification.

Challenges Faced and Solutions Implemented

1. Handling Complex PII Patterns

•Issue: Simple regex rules failed to detect all PII accurately.

•Solution: Integrated NER models for improved entity recognition beyond predefined patterns.

2. Deployment Issues on Hugging Face Spaces

• Issue: Routes of FastAPI (/classify) were returning 404 Not Found.

• Solution: Resolved route registration, explicitly defined POST requests, and reset the API.

3. Model Performance Optimization

• Issue: The original classification model was low in accuracy on noisy email data.

• Solution: Implemented enhanced preprocessing methods, balanced training data, and TF-IDF parameter optimization.

4. Ensuring API Adherence to Strict Format

• Issue: Submission guidelines demanded a structured JSON output format.

• Solution: Pydantic-based redesigned response models to adhere rigidly to required structure.

Conclusion

The PII Masking and Email Classification API effectively automates text analysis in emails with guaranteed privacy and enhanced classification efficiency. Advanced NLP algorithms and machine learning models ensure accurate results while ensuring compliance with security standards.

Future enhancements involve deeper multi-language support for more general email processing, application of deep learning models such as transformers for enhanced PII detection, increased training data for higher accuracy in classification.

This API illustrates the potential of machine learning for privacy protection, ensuring more secure and automated handling of emails