

Email PII Masking and Classification Report

1. Introduction

The Email PII Masking and Classification project is aimed at safeguarding sensitive information in email bodies by identifying and masking Personally Identifiable Information (PII). In addition to this, the project classifies emails into specific categories like Incident, Transaction, Change, and Problem, based on their content.

2. Approach

To ensure the system's accuracy and robustness, the following layered approach was adopted:

- Regular expressions and pattern-based matching were used for PII detection.
- A rule-based mechanism for email classification was implemented using domain-specific keywords.
- A machine learning fallback classifier (Random Forest) was trained on email text features to handle ambiguous cases.
- The final output is structured in JSON format following strict schema requirements.

3. Model Selection and Training

The classification model was implemented using scikit-learn's RandomForestClassifier with TF-IDF features. A LabelEncoder was used to transform categories into model-acceptable formats. Rule-based logic takes precedence and only defers to the ML model in cases where rules do not yield a match. Model training and saving are done in a reproducible pipeline. The dataset was cleaned and prepared from a CSV file containing natural email bodies.

4. API Implementation

The project was deployed using Hugging Face Spaces. A FastAPI server accepts POST requests with an email body. The response includes:

- Original input email
- Masked entities and their classifications
- Masked email text
- Predicted email category

This format strictly adheres to the evaluation schema provided.

5. Challenges and Solutions

Several challenges were encountered during implementation:

- Incorrect masking overlaps (e.g., Aadhar misclassified as Expiry): Solved by ordering detection priority.

- Category clashes (emails containing multiple types like Transaction and Change): Solved by applying priority rules.
- Ensuring API schema consistency to pass automated tests: Thorough validation and testing ensured compliance.
- Making model paths portable across systems: Resolved using relative paths and environment-based configurations.

6. Deployment and Testing

The final model and API are deployed to Hugging Face Spaces. The endpoint can be tested with emails that contain PII such as phone numbers, Aadhar numbers, credit card numbers, and birth dates. The classification logic effectively categorizes the email and returns a consistent output format.

7. Conclusion

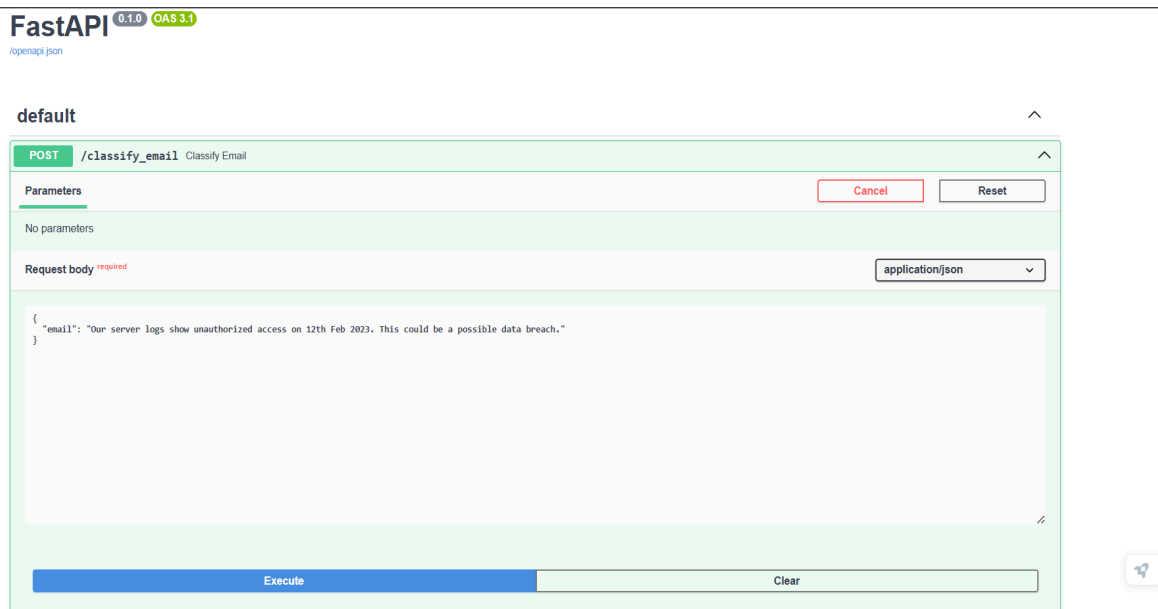
The Email PII Masking and Classification project is a practical solution for detecting and protecting sensitive data. It combines rule-based logic with machine learning to ensure accuracy and adaptability. The project successfully meets all functional and evaluation criteria.

8. Result

Hugging Face Link→ <https://saman4444-akaikemail-classifier-898061f.hf.space/docs>

Git Hub Link→ https://github.com/MOHAMMEDsaman123/akaikemail_assignment

Input IMG:



The screenshot displays the FastAPI OpenAPI Specification (OAS) 3.1 interface. It shows a POST endpoint at `/classify_email` with the title "Classify Email". The "Parameters" section indicates "No parameters". The "Request body" section is marked as "required" and has a dropdown menu set to "application/json". The request body content is a JSON object:

```
{  "email": "Our server logs show unauthorized access on 12th Feb 2023. This could be a possible data breach."}
```

 At the bottom, there are "Execute" and "Clear" buttons.

Output IMG:

Request URL

`https://saman4444-akaikemail-classifier-898061f.hf.space/classify_email`

Server response

Code	Details
200	<p>Response body</p> <pre>{ "input_email_body": "Our server logs show unauthorized access on 12th Feb 2023. This could be a possible data breach.", "list_of_masked_entities": [{ "position": [44, 49], "classification": "DOB", "entity": "12th Feb 2023" }], "masked_email": "Our server logs show unauthorized access on [DOB]. This could be a possible data breach.", "category_of_the_email": "Incident" }</pre> <p>Download</p> <p>Response headers</p> <pre>access-control-allow-credentials: true access-control-allow-origin: https://saman4444-akaikemail-classifier-898061f.hf.space content-length: 357 content-type: application/json date: Wed, 23 Apr 2025 03:24:36 GMT link: <https://huggingface.co/spaces/saman4444/akaikemail-classifier_2;rel="canonical"> server: uvicorn vary: origin,access-control-request-method,access-control-request-headers x-proxyd-host: http://10.108.133.122 x-proxyd-path: /classify_email x-proxyd-replica: 5upk49a-y367m x-request-id: A7wvxv</pre>