AI Engineer Round 1 - Coding Assessment

Problem Statement:
Given a dataset of customer support emails, build a solution to classify messages into categories and detect high-priority requests to improve response efficiency.

Approach & Methodology:
1. Combined email subject and body into a single feature.
2. Flagged priority using keywords such as *urgent*, *immediate*, *error*.
3. Used TF-IDF vectorization and a Naive Bayes classifier for categories: Account Issue, Billing Issue, Subscription, Technical/Integration, General Query.
4. Exported predictions and key metrics for evaluation.

Execution Steps:
1. Install dependencies: *pip install pandas scikit-learn reportlab*
2. Place *Sample_Support_Emails_Dataset.csv* and *classify_emails.py* in this folder.
3. Run: *python classify_emails.py* to produce *classified_emails_output.csv*.
4. Run this script: *python generate_pdf.py* to create the documentation PDF.

Results Summary:
Account Issue: 14
Billing Issue: 3
General Query: 2
Subscription: 1

Sample Classified Emails:

sender	subject	priority	category
eve@startup.io	Help required with account verification	1	Account Issue
diana@client.co	General query about subscription	1	Account Issue
eve@startup.io	Immediate support needed for billing error	1	Billing Issue
alice@example.com	Urgent request: system access blocked	1	Account Issue
eve@startup.io	Question: integration with API	0	Account Issue

Submission Notes:
- Public GitHub repository with all code, dataset, README, and this PDF.
- Demo video showing execution and outputs.
- Documentation structured for quick evaluation.