Building a Smarter AI-Powered Spam Classifier

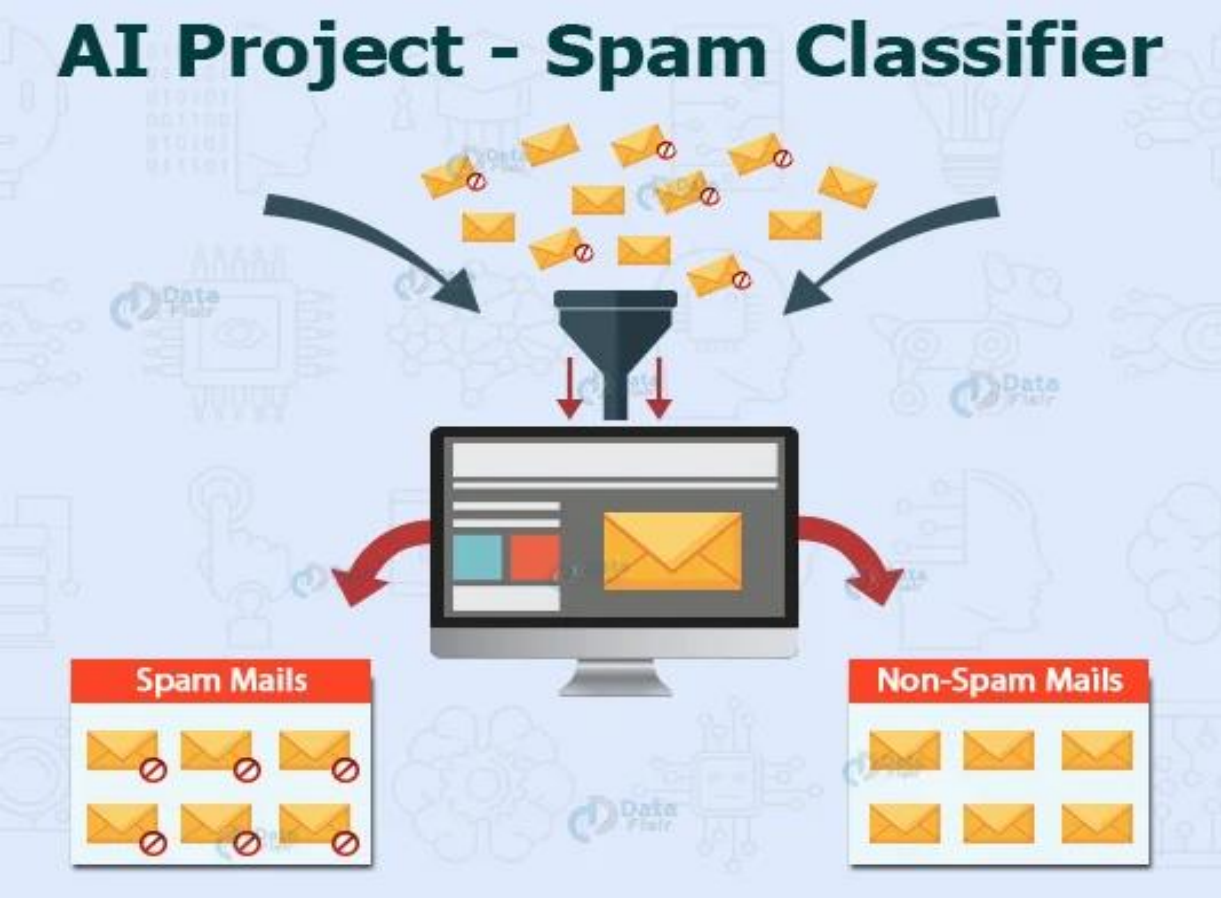
Phase-1 Document Submission

Project Overview

Project Name: Building a Smarter AI-Powered Spam Classifier

Project Phase: Problem Definition and Design Thinking

Project Description: The task at hand is to develop an AI-powered spam classifier capable of accurately distinguishing between spam and legitimate messages in emails or text messages. The primary goal is to minimize both false positives (identifying genuine messages as spam) and false negatives (missing actual spam) while maintaining a high level of overall accuracy. Achieving this balance involves using techniques like advanced natural language processing, ensemble modeling, threshold tuning, and continuous monitoring to adapt to evolving spam patterns. This spam classifier aims to enhance communication by effectively filtering out unwanted and potentially harmful content.



Project Definition

Problem Statement: The problem at hand entails the development of an AI-powered spam classifier designed to accurately differentiate between spam and non-spam messages in the context of emails or text messages

Objective: The objectives for creating an AI-powered spam classifier are straightforward. We aim to accurately distinguish spam from legitimate messages, reduce false positives and false negatives, ensure efficient and user-friendly operation, adapt to changing spam techniques, gather user feedback for improvement, and maintain user privacy within legal boundaries.

Design Thinking

1.Data Collection

Data source: We Obtained the dataset from Kaggle, specifically the Grumbletext website in Dataset

Data set: https://www.kaggle.com/datasets/uciml/sms-spam-collection-dataset

Objectives: Identify and acquire an available dataset containing Building a Smarter AI-Powered Spam Classifier

2.Data Preprocessing

Data Cleaning: The text data needs to be cleaned and preprocessed. This involves removing special characters, converting text to lowercase, and tokenizing the text into individual words.

Text Tokenization: The text data was tokenization into words for further analysis.

3.Spam Classifier Techniques

NLP Techniques: We employed Natural Language Processing Techniques for spam classifier,

Include:

- Word Embeddings
- Bog of Words
- Term Frequency-Inverse Document Frequency (TF-IDF) etc.,

4.Feature Extraction

Feature Generation: We extracted features from the text data, such as analysis the scores, keywords, and topics

Labeling: Ensure that you have labeled your data correctly, marking message as spam or non-spam

5.Visualization

Time Series Analysis: If your spam classifier operates over time, use time series analysis and visualizations to track changes in spam behavior and the effectiveness of the classifier over time.

Feature Importance Visualization: If you've engineered features or used feature selection, create visualizations to show the importance or relevance of different features in classifying spam. Bar charts or feature ranking plots can be effective for this purpose.

6.Insights Generation

Key Insights: Threshold optimization is a key insight to balance false positives and false negatives in your spam classifier.

Dashboard Insights: Extracting valuable information from a centralized dashboard that consolidates metrics, trends, and user feedback.

Conclusion: the task of building an AI-powered spam classifier is a complex yet essential endeavor aimed at enhancing communication by effectively filtering out unwanted or harmful content in emails or text messages.