Building a Smarter AI-Powered Spam Classifier

Name : MAHALAKSHMI .S

Reg No : 513521106019

Dept : ECE

Year : III

NM id: au513521106019

E-Mail: **s.ishwarya2010@gmail.com**

PROBLEM STATEMENT

- The objective of this project is to develop an Alpowered spam classifier capable of accurately distinguishing between spam and non-spam messages in email and text messages. The primary challenge lies in reducing the occurrences of false positives, where legitimate messages are incorrectly classified as spam, and false negatives, where actual spam messages are missed, while maintaining a high level of overall accuracy. The spam classifier will enhance the efficiency and security of communication channels by ensuring that unwanted and potentially harmful content is reliably filtered out.

PROJECT OVERVIEW

The primary objective of this project will be to develop a Artificial Intelligence model that will accurately classify spam and non-spam messages based on a set of relevant features

- **Dataset Source:** We will acquire our dataset from Kaggle, specifically the “SMS Spam Collection Dataset” dataset.
- **Dataset Link:** **https://www.kaggle.com/datasets/uciml/smsspam- collection-dataset**

TEAM MEMBERS

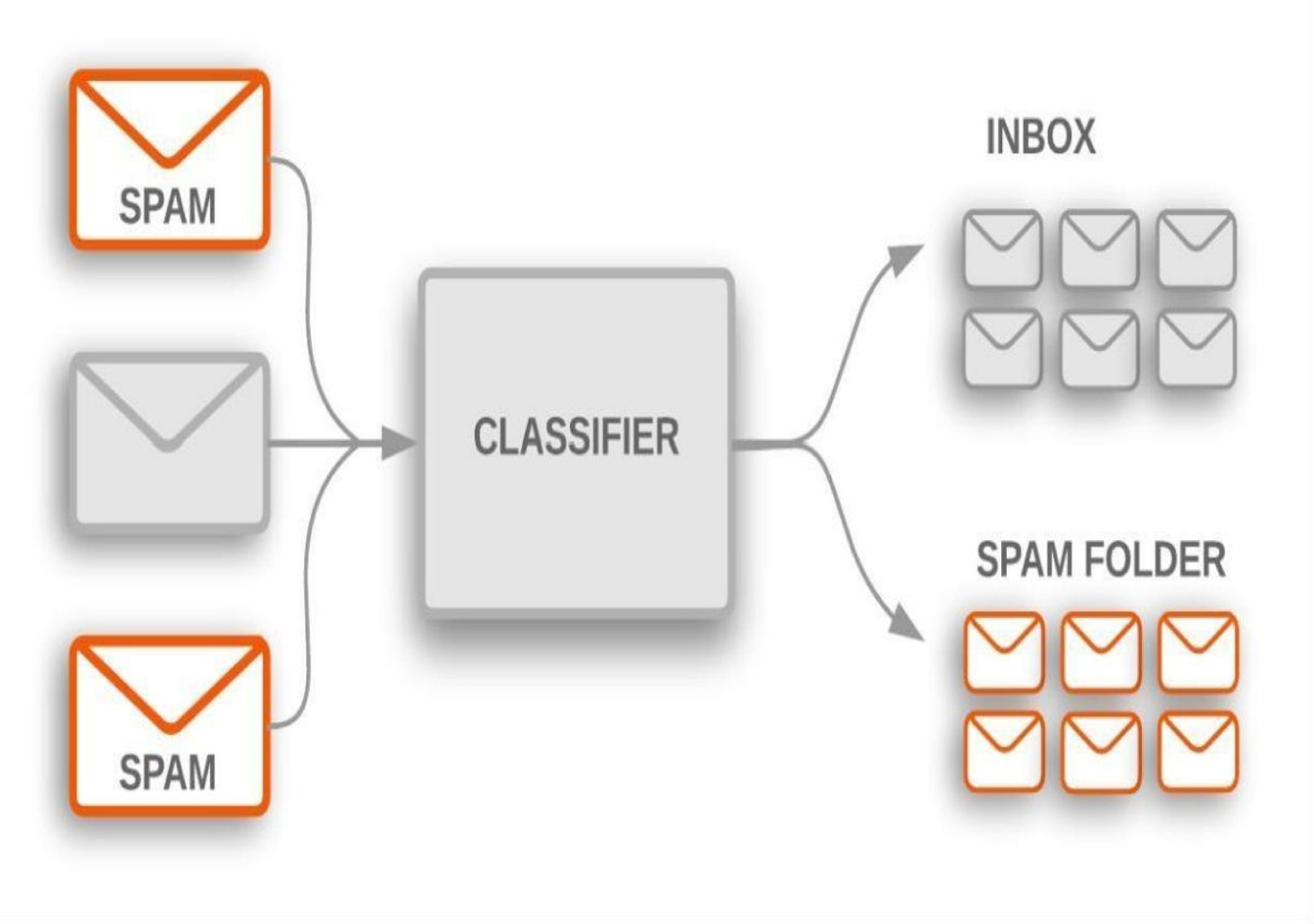
- SAFREEN.S
- MAHALAKSHMI.S
- LOKESWARI.R
- KEERTHANA.R
- LOGESHWARI.S

DECISION THINKING

- Our approach to solving this problem will be structured into several phases, each with specific objectives and tasks.
- These phases will include
- data collection,
- data preprocessing,
- feature extraction,
- model selection,
- evaluation,
- iterative improvement.
- This structured approach will ensure that we systematically address all aspects of the problem.

PROBLEM DEFINITION

The problem is to build an AI-powered spam classifier that can accurately distinguish between spam and nonspam messages in emails or text messages. The goal is to reduce the number of false positives (classifying legitimate messages as spam) and false negatives (missing actual spam messages) while achieving a high level of accuracy.



DATA SOURCE

- **Data Source: Kaggle Dataset - SMS Spam Collection Dataset**
- **Description:** The data source you've chosen is the "SMS Spam Collection Dataset" available on Kaggle. This dataset is a widely used and publicly available collection of SMS (text message) data, which is
labeled as either "spam" or "ham" (non-spam). It's commonly used for building and evaluating spam classification models.
- **Key Characteristics of the Dataset:**

1. **Message Text:** The dataset contains a collection of SMS messages. Each message is represented as a text string.
2. **Labels:** Each SMS message is labeled as either "spam" or "ham" (non-spam). This labeling is essential for supervised machine learning, as it provides the ground truth for training and evaluating your spam
classifier.
3. **Data Size:** The dataset typically contains hundreds or thousands of SMS messages, which is usually sufficient for building an initial spam classifier.
4. **Message Text:** The dataset contains a collection of SMS messages. Each message is represented as a text string.

5. **Labels:** Each SMS message is labeled as either "spam" or "ham" (non-spam). This labeling is essential for supervised machine learning, as it provides the ground truth for training and evaluating your spam classifier.

6. **Data Size:** The dataset typically contains hundreds or thousands of SMS messages, which is usually sufficient for building an initial spam classifier.

DATA PREPROCESSING

- Data preprocessing involves cleaning text data by removing special characters, symbols, and punctuation while converting text to lowercase for consistency.
- Additionally, tokenization breaks down text messages into individual words (tokens) for analysis. These steps standardize the text data, reduce noise, and prepare it for feature extraction and model training. Care should be taken to balance noise reduction and information preservation, and we may decide whether to remove or keep common words (stop words) based on your project's requirements.

- After preprocessing, the data is ready for further analysis and machine learning model development.
- Tools/Modules:
  - Python with NLP libraries
  - TF-IDF
  - ML models

FEATURE EXTRACTION

• We will carefully select the most relevant features that will contribute to classifying spam messages accurately. This will be achieved through exploratory data analysis and feature importance analysis. • We will utilize techniques like TF-IDF (Term Frequency-Inverse Document Frequency) or word embeddings to convert text into numerical features.

MODEL SELECTION

- We will select a suitable classification algorithm (eg: Naive Bayes, Support Vector Machines, and more advanced techniques like deep learning using neural networks.) for the spam classification task.
- Naïve Bayes calculates the probability of some event that has happened and ensures for the accuracy.
- SVM is one of the most popular Supervised Learning algorithms, which is used for Classification as well as

Regression problems.

- deep learning, that uses interconnected nodes or neurons in a layered structure that resembles the human brain.

EVALUATION

- We will assess the model's performance using the following metrics:

- **Accuracy:** Measures the proportion of correctly classified instances overall.
- **Precision:** Evaluates the accuracy of positive predictions and the rate of false alarms.
- **Recall:** Measures the model's ability to correctly identify all relevant instances.
- **F1-score:** Strikes a balance between precision and recall, considering both false alarms and missed instances.

ITERATION IMPROVEMENT

- In the iterative improvement phase, we will systematically fine-tune the model's hyperparameters, explore different model architectures, implement cross-validation, apply regularization techniques, consider ensemble methods, revisit feature engineering, monitor performance metrics, assess learning curves, and, if applicable, conduct A/B testing to optimize the spam classifier's accuracy and effectiveness.

POSSIBLE FUTURE WORK

- Future possible works for your AI-powered spam classifier include exploring advanced NLP techniques, extending to handle multimedia content, implementing real-time monitoring, using transfer learning, adapting to user feedback, enabling multiclass classification, handling non-textual data, customizable filters, cross-platform integration, multilingual support, explain ability and fairness, privacypreserving detection, user education, continuous evaluation, and benchmarking for ongoing improvements.