**AI Ideation phase 5**

|  |  |
| --- | --- |
| **Date** | **31-10-2023** |
| **Team ID** | **3921** |
| **Project Name** | **Spam prediction using powered AI** |

**Table of Contents:**

|  |  |
| --- | --- |
| 1. | Problem statement |
| 2. | Design thinking process |
| 3. | Phases of development |
| 4. | Dataset Description |
| 5. | Data Preprocessing |
| 6. | Feature Extraction |
| 7. | Model Selection |
| 8. | Model Training |
| 9. | Evaluation Metrices |
| 10. | Innovative Approaches |
| 11. | conclusion |

**1.Problem Statement:**

The problem at hand is to develop a robust spam detection system using artificial intelligence. Spam detection is essential in various online communication platforms, such as email, social media, and messaging services, to protect users from unsolicited and potentially harmful messages. The primary goal of this project is to build an AI model that can accurately identify and classify messages as spam or non-spam.

**2.Design Thinking Process:**

A diagram of a machine learning

Description automatically generated

**1. Understand the Problem**

* Define the problem: Spam detection in text-based messages.
* Identify the stakeholders: End-users, administrators, and platform providers.
* Gather requirements: Accuracy, scalability, and real-time processing.

**2. Data Collection:**

* Gather a diverse dataset of labeled spam and non-spam messages.
* Ensure data privacy and ethical considerations.

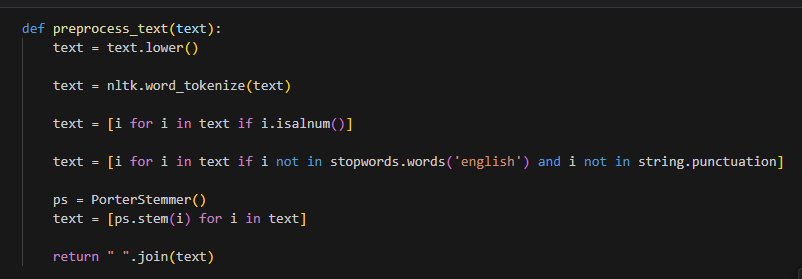


**3. Data Preprocessing:**

* Text cleaning: Remove special characters, stopwords, and perform lowercasing.
* Tokenization: Split messages into individual words.
* Feature extraction: Convert text into numerical features.

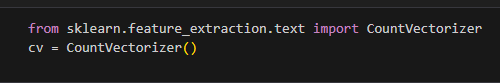
A diagram of a model

Description automatically generated



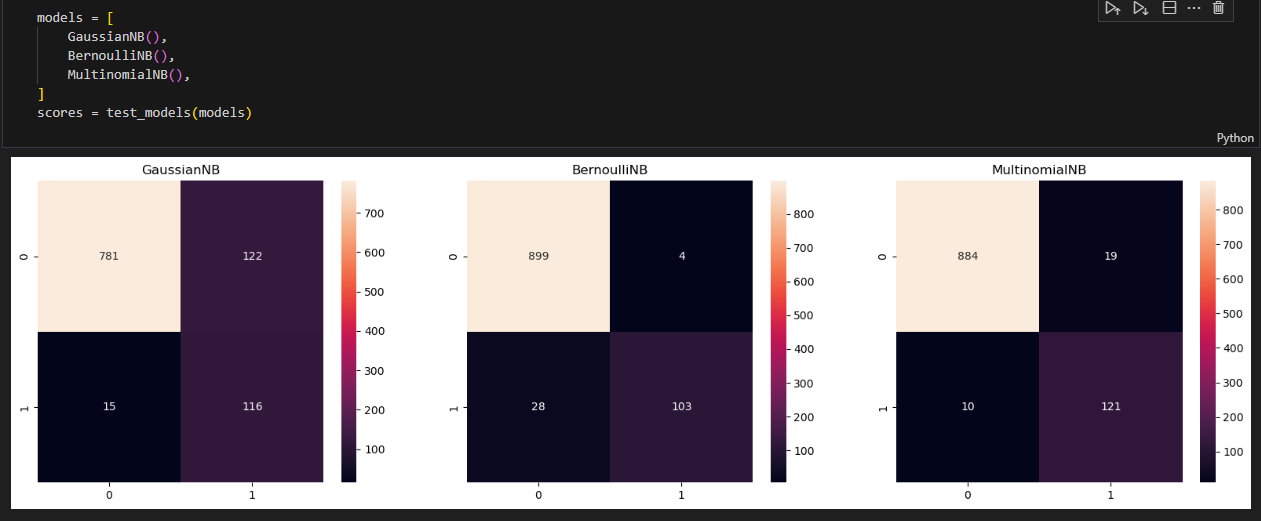
**4. Model Selection:**

* Explore various machine learning and deep learning algorithms.
* Choose a model that balances accuracy, efficiency, and scalability



**5. Model Training:**

* Split the dataset into training and validation sets.
* Train the selected model on the training data.
* Optimize hyperparameters for the best performance.



**6. Evaluation:**

* Use appropriate evaluation metrics (e.g., accuracy, precision, recall, F1-score).
* Perform cross-validation to ensure model generalization.
* Address overfitting and underfitting issues.

**7. Deployment:**

* Create an API or integrate the model into the desired platform.
* Monitor the model's performance in a real-world environment.

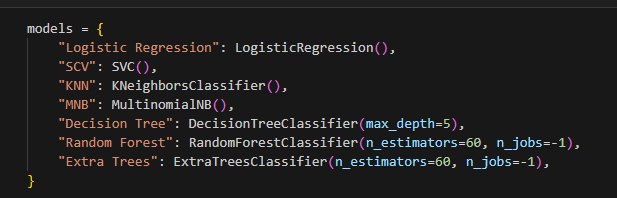
**3.Phases of Development:**

**Phase 1: Data Preparation:**

* Dataset Collection: Gather a diverse dataset of spam and non-spam messages.
* Data Cleaning: Remove special characters, stopwords, and perform text lowercasing.
* Tokenization: Split messages into individual words.
* Feature Extraction: Convert text into numerical features (e.g., TF-IDF, Word Embeddings).

**Phase 2: Model Development:**

* Model Selection: Choose a suitable algorithm (e.g., Naive Bayes, SVM, LSTM).
* Model Training: Train the selected model on the preprocessed data.
* Hyperparameter Tuning: Optimize model hyperparameters for accuracy.



**Phase 3: Evaluation and Validation:**

* Evaluation Metrics: Use metrics like accuracy, precision, recall, and F1-score.
* Cross-Validation: Ensure model generalization and robustness.
* Addressing Overfitting: Apply regularization techniques as needed.

**Phase 4: Deployment:**

* Build an API or integrate the model into the desired platform.
* Monitor Model Performance: Continuously monitor the model in a real-world environment.
* Model Updates: Periodically update the model with new data.

**4.Dataset Description:**

The dataset used for this project comprises a collection of text messages, which have been labeled as either spam or non-spam. The dataset consists of both training and testing subsets. It is balanced to ensure an equal representation of both spam and non-spam messages.

**5.Data Preprocessing:**

* Text Cleaning: Removal of special characters, punctuation, and unnecessary whitespace.
* Tokenization: Splitting text into individual words or tokens.

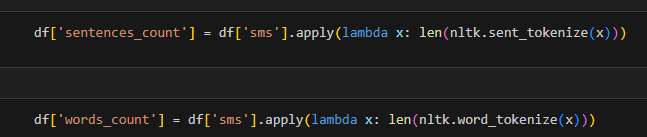
**6.Feature Extraction:**

Feature extraction is a crucial step in spam classification using AI. It involves transforming the raw data (in this case, email or text messages) into a structured format that machine learning models can understand and use for classification. Feature extraction techniques aim to capture relevant information from the text data, making it suitable for training a spam classifier. Here are some common feature extraction techniques for spam classification:

* Bag of words(BOW)
* Term Frequency-Inverse Document Frequency (TF-IDF):
* Word Embeddings
* N-grams
* Character- based Features

**A diagram of a process flow

Description automatically generated**



**7.Model Selection:**

The machine learning algorithm selected for this project is a Multinomial Naive Bayes classifier. This algorithm is known for its effectiveness in text classification tasks, especially when dealing with text data.

**8.Model Training:**

* The dataset is split into a training set and a validation set.
* The Multinomial Naive Bayes model is trained on the training set.
* Hyperparameters such as alpha (smoothing parameter) are optimized using cross-validation.

**9.Evaluation Metrics:**

The performance of the spam detection model is assessed using the following metrics:

* Accuracy: To measure overall classification correctness.
* Precision: To measure the percentage of correctly classified spam messages.
* Recall: To measure the percentage of actual spam messages correctly identified.
* F1-score: A harmonic mean of precision and recall.

**10.Innovative Approaches:**

BERT is a pre-trained transformer-based model that learns contextual representations of words by considering the entire context of a sentence or document. Key components and aspects of BERT that are used in spam classification are as follows:

* Bidirectional Context: Unlike traditional models like LSTM or CNN, BERT looks at both the left and right context of a word. This allows the model to understand the relationship between words in a sentence, capturing nuances that are crucial in distinguishing spam from legitimate text.
* Word Embeddings: BERT uses subword embeddings, which are particularly useful for capturing misspellings and variations often present in spam messages.
* Pre-trained Weights: BERT is pre-trained on a large corpus of text, giving it a vast understanding of language. Fine-tuning BERT on a specific task, like spam classification, leverages this pre-trained knowledge.
* Contextualized Representations: BERT creates contextualized word embeddings, meaning the representation of a word depends on the surrounding words. This is invaluable for understanding the true intent of the text.

**Literature and Survey Papers:**

|  |  |  |
| --- | --- | --- |
| SURVEY PAPER NAME | YEAR OF THE PAPER | ALGORITHM USED |
| Email spam detection using ML algorithms | July 2020 | Naïve bayes , random forest, neural networks |
| Prediction of spam mail using ML classification algorithm | June 2020 | SVM, decision tree, CNN, KNN, MLP, Adaboost |
| Hybrid email spam detection model using AI | February 2020 | Neural networks |
| Detection of email spam using natural language processing based random forest approach | September 2021 | NLP, NLTK, Scikit-learnare |
| Email based spam detection | June 2020 | K-Mean, Bi-gram, porter stemming |

**Conclusion:**

In conclusion, the development of a spam classification system using AI is a multi-faceted approach that combines user-centric design, innovative AI techniques, and rigorous development phases to create a robust, adaptable, and efficient solution. The implementation of AI in this context not only enhances user experience but also reinforces the security and trustworthiness of digital communication, contributing to a safer and more productive online environment. As spam patterns continue to evolve, the AI-based spam classifier will remain at the forefront of the battle against unwanted messages, improving the quality of our online interactions.