**College code : 6208**

**Name : C.Vinod Kumar**

**IBM Reg No : au620821106099**

**Project name : Building a Smarter AI-Powerd Spam Classifier**

**Definition:**

A smarter AI-powered spam classifier is a machine learning system designed to accurately and efficiently differentiate between legitimate and unwanted messages, such as spam emails or messages. It goes beyond traditional rule-based methods and leverages advanced techniques to continually improve its performance.

This classifier typically:

\*Utilizes Machine Learning

\* Adapts Over Time

\*Considers Context

\*Multimodal Analysis

\*Regularly Updates Rules

\*User Feedback Integration

\*Scalability

**Designing**

**Data Collection :** Gather a large dataset of labelled emails or messages, distinguishing between spam and non-spam (ham).

**Data Preprocessing :** Clean and preprocessing the data, which may include removing special characters, stemming, and tokenization.

**Feature Extraction:** Extract relevant features from the text, such as word frequencies or TF-IDF scores.

**Model Selection :** Choose an appropriate machine learning or deep learning model for classification, like Naive Bayes, Support Vector Machines, or neural networks.

**Training :** Train the selected model on your labelled dataset. Consider techniques like cross-validation to optimize hyper parameters.

**Evaluation :** Evaluate the model's performance using metrics like accuracy, precision, recall, and F1-score. Tweak the model as needed.

**Ensemble Methods :** Combine multiple models or use ensemble methods to improve classification accuracy.

**Continuous Learning :** Implement mechanisms for the model to continuously learn and adapt to new spam patterns over time.

**Feature Engineering :** Experiment with different features and techniques, such as n-grams or word embeddings, to improve classification accuracy**.**

**Regularization :** Prevent over fitting by applying regularization techniques like dropout or L2 regularization.

**Threshold Tuning :** Adjust the classification threshold to balance false positives and false negatives based on your application's requirements.

**Testing and Deployment :** Test the model thoroughly and deploy it in your application or email system.

**Monitoring and Maintenance :** Continuously monitor the model's performance in real-world scenarios and update it as needed to handle new spam tactics.

**User Feedback** : Collect user feedback to further improve the classifier and reduce false positives/negatives.

**Security :** Ensure the model and its data are secure to prevent adversarial attacks and data breaches.