**Building A Smarter AI\_Powered Spam Classifier**

**Phase:4**

**A Comparative Analysis:**

The increase in the number of unwanted emails known as spam has produced a need for the development of more reliable and effective anti-spam filters. Recently, machine learning algorithms have been effectively utilised to detect and filter spam emails. In the field of natural language processing, there are numerous algorithms available for this kind of classification. Typically, spam emails contain a few recognisable terms that are pretty obvious indicators that the email is spam.

In this article, we will go through the processing of the data, exploring the data and applying the algorithms to find and compare the efficiency of several machine learning techniques such as KNN, Random Forest, Naive Bayes, SVM, and Logistic Regression, and more approaches. A collection of data consisting of around 60,000 emails including both authentic and spam emails was utilised in this investigation. These strategies are also thoroughly compared in terms of accuracy, precision, recall, etc.

In general, all email communications are labelled as “Ham” or “Spam.” In a mailbox, Ham communications are intended or safe acceptable messages, whereas Spam messages are trash, unwanted mass, or commercial messages. This filtering or categorization of email communications into Ham and Spam aids in separating them and automating the deletion of spam messages. Typically, there are several variables or components that contribute to the detection of spam emails.

**Importing The Relevant Libraries:**

import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

import seaborn as sns

import nltk

from wordcloud import WordCloud

from os import walk

from string import punctuation

from random import shuffle

from collections import Counter

import multiprocessing

import email

import sklearn as sk

from nltk.corpus import stopwords

nltk.download('stopwords')

nltk.download('wordnet')

nltk.download('punkt')

nltk.download('omw-1.4')

nltk.download('averaged\_perceptron\_tagger')

%matplotlib inline

**Dataset Selection:**

The Enron Spam dataset was used for this analysis. The dataset includes 3675 spam and 1437 non-spam (“ham”) e-mail messages (5112 records total). The original datasets, on the other hand, are documented in such a way that each and every email is in its own txt-file, which is spread across numerous folders. This can make understanding the data more difficult, especially for novices. Because data collection is such an amazing resource, let’s organize the data into a single CSV file.

**Text Processing:**

Building a machine learning model requires the preprocessing of data, and the quality of the preprocessing determines the model’s performance.

Preprocessing text is the initial stage in NLP’s model-building process.

**Following are the various text preparation steps:**

Removing Punctuations

Tokenization

Removing Stopwords

Stemming

Lemmatization

**Feature Selection:**

i. Spamicity

iii. Random Forest

**Algorithm Application**

For classification and the development of systems for auto spam detection, the following Machine Learning techniques can be used:

i. Naive Bayes

ii. Support Vector Machine

iii. Random Forest

iv. K Nearest Neighbors

v. Logistic Regression