MIT | Academy of Engineering

(An Autonomous Institute Affiliated to Savitribai Phule Pune University)

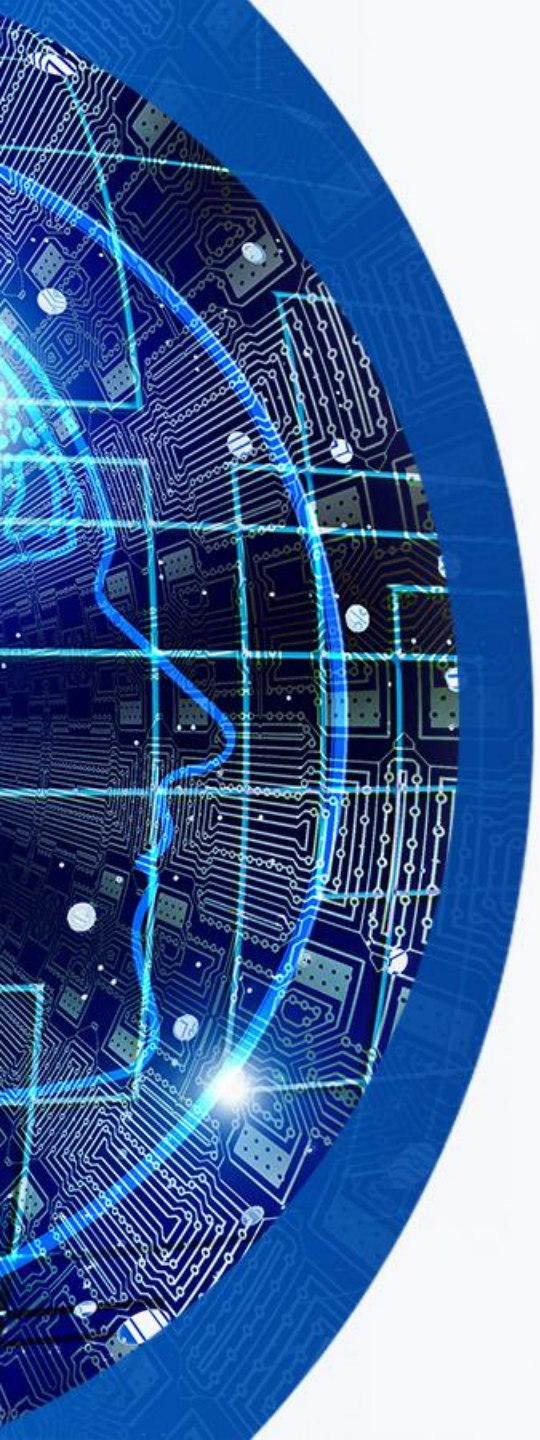
AI Practical Examination

Name: Aditya C. Karpe

PRN No: 0220200243

Roll No: 166

Exam seat No: T-214161



Problem Statement:

Build a model by using collected data and apply Spam Filtering.

Solution:

Apply Naïve Bayes on Spam Dataset and classify whether SMS is Spam or not.

Naïve Bayes Algorithm

- Naïve Bayes Classifier are a popular statistical technique of e-mail filtering. They typically use bag of words features to identify spam e-mail, an approach commonly used in text classification.
- Naive Bayes classifiers work by correlating the use of tokens (typically words, or sometimes other things), with spam and non-spam e-mails ,SMS and then using Bayes's to calculate a probability that an email is or is not spam.

The formula used by the software to determine that, is derived from Bayes' theorem

$$\Pr(S|W) = \frac{\Pr(W|S) \cdot \Pr(S)}{\Pr(W|S) \cdot \Pr(S) + \Pr(W|H) \cdot \Pr(H)}$$

where:

- $\Pr(S|W)$ is the probability that a message is a spam, knowing that the word "replica" is in it;
- $\Pr(S)$ is the overall probability that any given message is spam;
- $\Pr(W|S)$ is the probability that the word "replica" appears in spam messages;
- $\Pr(H)$ is the overall probability that any given message is not spam (is "ham");
- $\Pr(W|H)$ is the probability that the word "replica" appears in ham messages.

Workflow of Machine Learning Based Case Study

- Exploring the Dataset
- Training and Test Set
- Data Cleaning
- Letter Case & Punction
- Creating The Vocabulary
- The Final Training Set
- Calculating Constants First
- Calculating Parameters
- Classifying A New Message
- Measuring the Spam Filter's Accuracy

Exploring the Dataset

The Spam SMS Dataset

For SMS Spam Dataset, there are total 5000 rows(training samples) and 2 columns in dataset. Each column details are as below

Column Name Details

1. Label1 :-Label of messages which are spam or not
2. Label2 :- Mail Body

Statistical Summary

	Label	SMS
count	5572	5572
unique	2	5169
top	ham	Sorry, I'll call later
freq	4825	30

Data Preparation

1. **Exploring the Data**

Let's start by opening the SpamFiltering with the read_csv() function from the pandas package. We're going to use:

sep='\t' because the data points are tab separated.

header=None because the dataset doesn't have a header row.

names=['Label', 'SMS'] to name the columns.

2. **Training and Test Set**

We're now going to split our dataset into a training set and a test set. We'll use 80% of the data for training and the remaining 20% for testing.

3. **Data Cleaning**

When a new message comes in, our multinomial Naive Bayes algorithm will make the classification based on the results it gets to these two equations below, where "w1" is the first word, and w1,w2, ..., wn is the entire message.

4. **Letter Case and Punctuation**

Let's begin the data cleaning process by removing the punctuation and making all the words lowercase.

5. **Creating the Vocabulary**

We transform each message in theSMS column into a list by splitting the string at the space character — we're using the Series.str.split() method.

6. **The Final Training Set**

Testing the Vocabulary for Transformation. initializing a dictionary named word_counts_per_sms

7. **Calculating Constants First**

Using the Formula:

$$P(w_i|Spam) = \frac{N_{w_i|Spam} + \alpha}{N_{Spam} + \alpha \cdot N_{Vocabulary}}$$

$$P(w_i|Ham) = \frac{N_{w_i|Ham} + \alpha}{N_{Ham} + \alpha \cdot N_{Vocabulary}}$$

8. **Calculating Parameters**

The parameters are calculated using these two equations:

$$P(w_i|Spam) = \frac{N_{w_i|Spam} + \alpha}{N_{Spam} + \alpha \cdot N_{Vocabulary}}$$

$$P(w_i|Ham) = \frac{N_{w_i|Ham} + \alpha}{N_{Ham} + \alpha \cdot N_{Vocabulary}}$$

9. **Classifying A New Message**

Takes in as input a new message (w1, w2, ..., wn).
Calculates P(Spam|w1, w2, ..., wn) and P(Ham|w1, w2, ..., wn).
Compares the values of P(Spam|w1, w2, ..., wn) and P(Ham|w1, w2, ..., wn), and:
If P(Ham|w1, w2, ..., wn) > P(Spam|w1, w2, ..., wn), then the message is classified as ham.
If P(Ham|w1, w2, ..., wn) < P(Spam|w1, w2, ..., wn), then the message is classified as spam.
If P(Ham|w1, w2, ..., wn) = P(Spam|w1, w2, ..., wn), then the algorithm may request human help.

10. **Measuring the Spam Filter's Accuracy**

Its Compare The Two Results. After Comparing It uses the formula & state the Accuracy

$$\text{Accuracy} = \frac{\text{number of correctly classified messages}}{\text{total number of classified messages}}$$

Thank You