LONDON METROPOLITAN UNIVERSITY

informatics college · pokhara

Module Code & Module Title

CU6051NP Artificial Intelligence

25% Individual Coursework

Submission: Final Submission

Academic Semester: Autumn Semester 2025

Credit: 15 credit semester long module

Student Name: Suraj Ghimire

London Met ID: 23049005

College ID: NP04CP4A230213

Assignment Due Date: 10/12/2025.

Assignment Submission Date: 17/12/2025

Submitted To: Suraj Shrestha

GitHub Link	*https://github.com/SurajGhimire29/AI-python.git*

I confirm that I understand my coursework needs to be submitted online via MST Classroom under the relevant module page before the deadline for my assignment to be accepted and marked. I am fully aware that late submissions will be treated as non-submission and a mark of zero will be awarded.

Table of Contents

Table of Figures

1. Introduction

1.1. Introduction to Artificial Intelligence

AI, in the fuller form, is the science of making computer systems perform tasks that would normally be manageable with human intelligence. Such tasks can be performed by enabling one to learn from data, reason, solve problems, and even reach decisions. AI combines various areas of application, including computer science, data science, mathematics, software engineering, and even cognitive science in the development and making of intelligent systems.

Most AI systems are based on the whole concept of a data-driven approach where large volumes of data will be processed with algorithms in order to learn from them and make improvements with time. Machine learning, the most important field of AI, enables the system to discover patterns in data automatically and also with much prediction or decision-making without explicitly programming for this. In this respect, AI finds applications in health, finance, transportation, and customer service with the aim of making the operations both fast and more accurate (Stryker, 2025).

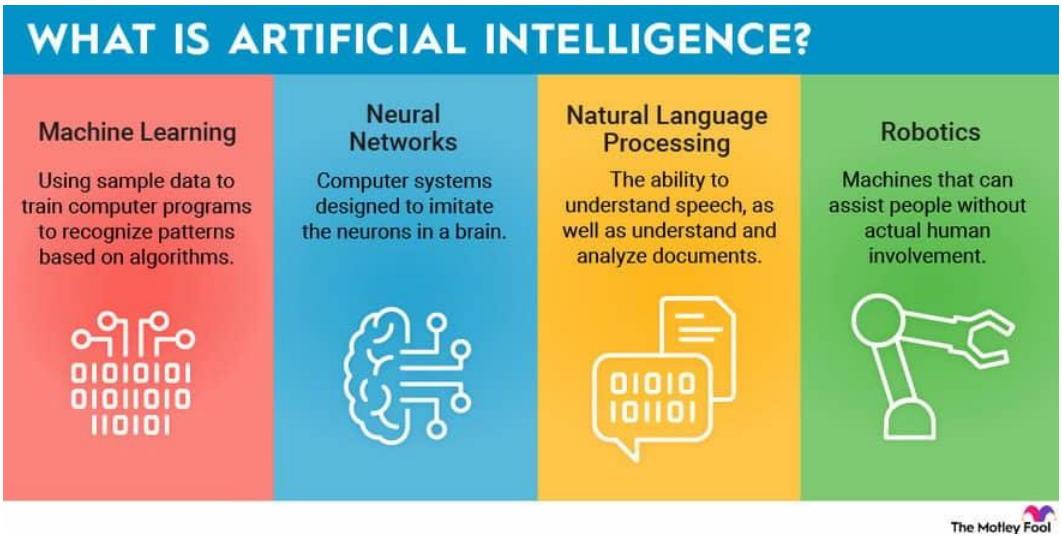

Figure 1:Artificial Intelligence

1.2. Introduction to the Chosen Topic/Problem Domain

The topic of interest for the project work of the Artificial Intelligence module is noted to be Spam Detection. This topic has been selected considering the increasing adoption of digital communication and the consequent spams.

Spam messages can also be regarded as unsolicited messages that make use of digital communication services. These services include email communication services, Short Message Service (SMS), and various social platforms. Spam messages include promotional messages. Others include phishing messages, fraudulent messages, and malicious messages. It can be observed that due to spam messages, various serious issues have arisen between individuals and various organizations. These include loss of productivity and congestion in the network. Additionally, fraud and several privacy concerns have also arisen due to spam messages. These messages include financial fraud and concerns pertaining to identity theft (Nagvekar, 2025).

Spam Filtering centers on automatically removing those unwanted mails before they reach the end recipient. As far as the concern in this aspect is relevant, it is closely related to Artificial Intelligence, specifically to Machine Learning and Natural Language Processing. Machine Learning leads to the process of learning using those labeled examples, which label mails as spam messages or not spam messages, and Machine Learning leads to proper classification of those unknown messages using that learning process developed before. Natural Language Processing is a technique of text processing, allowing it to obtain meaningful features in terms of those classifications (Riyadi, n.d.).

Relevance of Spam Detection can be explained for the following reasons:

Increase in Volume of Spam Messages: The use of the internet and mobile devices has greatly contributed to the rise in spam messages being delivered globally.

Security and Privacy Risks: The URLs within spam emails may contain phishing links that may be harmful to user data and system security.

Limitations of Conventional Methods: The rule-based filtering methods have become outdated due to the changing message patterns used by spammers in order to outsmart the filters.

Need and Requirements for Intelligent Systems: The Intelligent Systems designed to detect spams through Artificial Intelligence approaches have the capability of learning and adapting to new patterns.

Related Topics: The use of the spam filters has become prevalent in mail services, messaging applications, and social media platforms in order to improve the safety and reliability of communications.

Due to these conditions, the design of the system that identifies spams in the mail has been a major topic in the field of Artificial Intelligence. The design of the intelligent spam filter system is therefore imperative in ensuring a secure and efficient communication process in modern society.

1.3. Aims and Objective

1.3.1. Aims

The design of a machine learning-based system that should be capable of effectively detecting spam messages in digital communication platforms, including emails, SMS, and social networking sites for user security and effective communication.

1.3.2. Objective

Construct a machine learning algorithm that can classify messages into two categories-that is, either spam or not spam-based on previous datasets.

The textual inputs need to be preprocessed using natural language processing techniques and extract important features.

Evaluate and compare the performance of different machine learning algorithms, such as Naïve Bayes, Support Vector Machines or SVM, and Random Forest for spam filtering.

Design an adaptive system, which can renew itself with the knowledge of the newer spams and increase the accuracy in classification.

Use a simulator to test the spam filtering system and analyze its effectiveness when it comes to real-world applications.

1.4. Dataset

It is one of the most popular data sets in the areas of machine learning, natural language processing, and other allied disciplines, which is being used exclusively for analyzing spam. Spam detection is a very relevant problem in contemporary communication networks, since unwanted emails lead not only to poor user experience but also result in malicious content being transmitted. It is, therefore, very essential to design accurate models to identify emails as spam/signal emails.

This particular dataset was contributed to the UCI Machine Learning Repository, which is a most revered and frequently cited source of machine learning datasets. The UCI repository supplies datasets of high quality to researchers, students, and professionals

interested in machine learning. The SMS Spam Collection has become a benchmark dataset in the field of spam filtering because of its practical mixing of messages in a size that is not too small or too large and the existence of examples labeled in a manner necessary within supervised learning methods.

The total number of SMS messages in the dataset is 5,574, and each one is categorized into either the spam category or the ham category. This is because the dataset is diverse because the SMS messages come from various sources. In addition to that, the problem is inherently imbalanced because there is a huge difference between the number of ham and spam SMS messages, just like the real-world scenario.

Link: https://archive.ics.uci.edu/dataset/228/sms+spam+collection

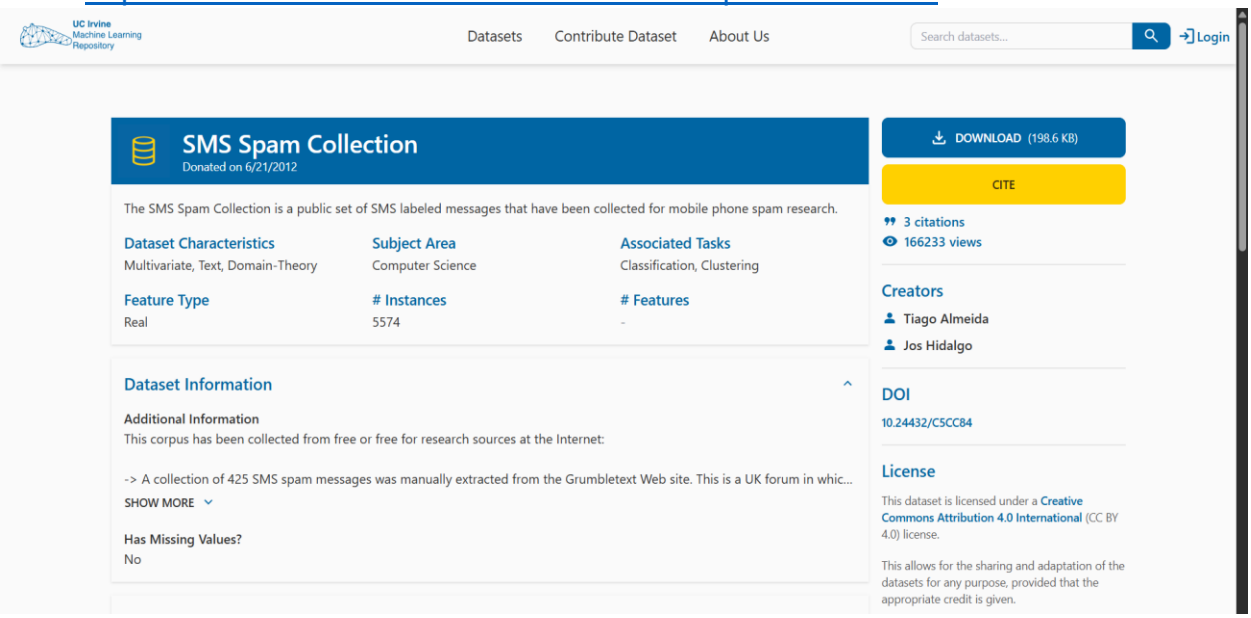

Figure 2:Dataset Website

2. Background

2.1. Research regrading the problem domain

Recent work carried out for SMS spam classification has also revealed the efficacy of machine learning approaches for effective classification purposes. A recent work carried out for SMS spam classification, as presented in a research paper published by Atma et al., in the year 2025, compared the performance of several traditional machine learning approaches such as Naïve Bayes and Support Vector Machines, along with ensemble-based approaches such as Random Forest and Gradient Boost, for effective classification of spam messages in the SMSPamCollection dataset, which is available for public utilization from the University of California Machine Learning Repository after preprocessing the individual SMS messages using several NLP approaches such as tokenization, stop word removal, and TF-IDF transcription. The results revealed the efficacy of the ensemble approaches, as their accuracy levels remained higher than those of all other approaches, with values surpassing the figure of 98% effectively, thereby establishing their efficacy for effective utilization in the presence of noisy and unbalanced data. However, this research work also highlights several limitations, which might act as obstacles for their effective utilization in the near future. This includes the fact that they have utilized an unchanging dataset, along with the absence of appropriate consideration for the presence of concept drift during their approach. This might restrict their effective utilization within real-world platforms, which continue to have dynamically fluctuating patterns for spam-SMS correspondences. However, this research work undoubtedly reinforces the efficacy of machine learning approaches for effective SMS spam classification purposes (Atma, 2025).

Link: https://jurnal.itscience.org/index.php/CNAPC/article/view/4822

Figure 3:Research article

2.2. Review and analysis of existing work

Here's a review and analysis of existing work in the SMS spam detection problem:

- Gmail Spam Filter (Google)

Gmail has highly advanced technology for handling and filtering out spam messages through highly advanced machine learning and natural language processing. The technology learns continuously from an enormous amount of messages and "Mark as spam" feedback and ever-changing patterns of spam. The technology has the capability of distinguishing between normal and spam messages, as well as malicious links found in phishing and promotional spam.

Link: https://mail.google.com/

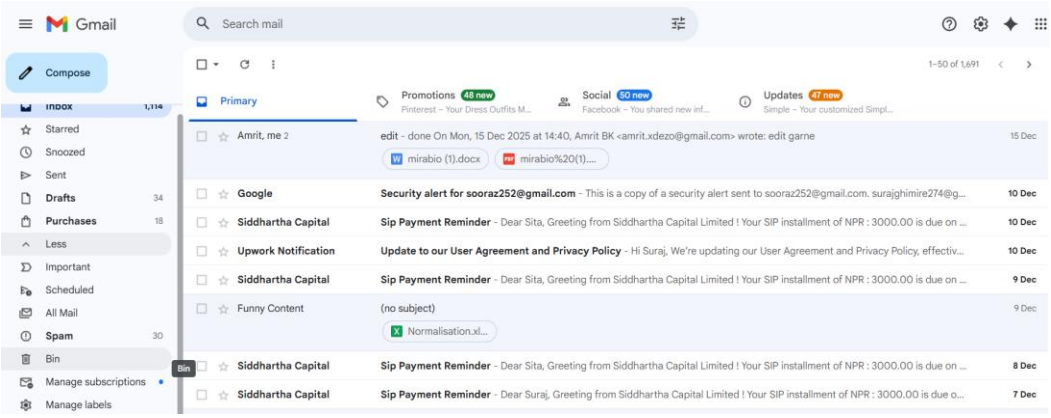

Figure 4:Gmail Spam Filter (Google)

Why it is exciting:

- Employs large-scale AI models
- Continuously improves through user interaction
- Safeguards billions of internet users globally

➢ Android Spam protection in SMS - (Google Messages)

Google Messages are also equipped with machine learning-driven SMS spam detection on Android. It automatically detects spam messages-fake lottery messages, scams, or promotional SMS-and moves them to the spam folder without user interference. More importantly, the detection often runs on-device, meaning it doesn't sacrifice user privacy.

Link: https://www.android.com/messages/

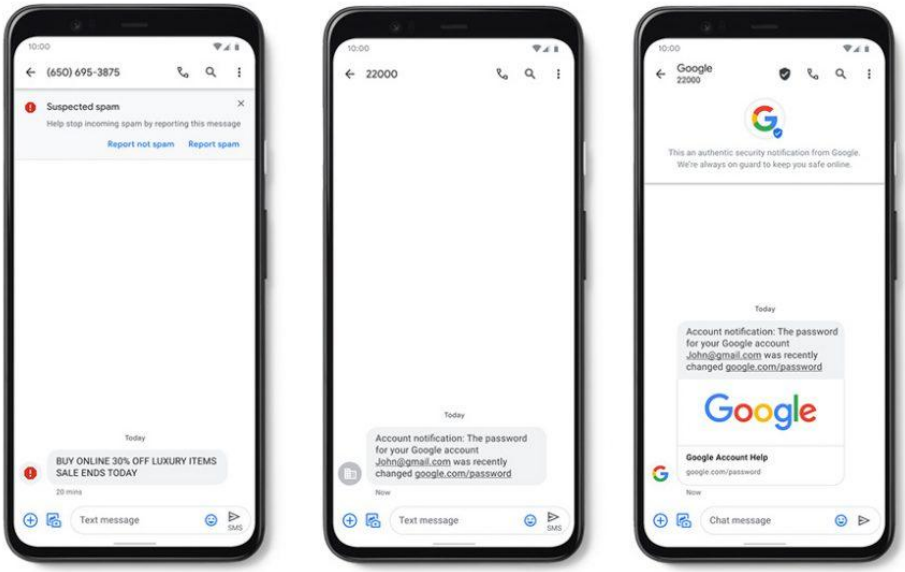

Figure 5:Android Spam protection in SMS

Why it is exciting:

- Works in real-time
- Uses on-device machine learning; Strongly privacy- and security-focused
-

➢ Truecaller Spam Detection System

Truecaller is a widely used smartphone app that uses the power of AI and community intelligence data to trace spam calls and message notifications. It has a global spam numbers database and uses machine learning algorithms that help it trace new spam patterns. It immediately alerts the user regarding spam notifications and calls.

Link: https://www.truecaller.com/truecaller-for-android

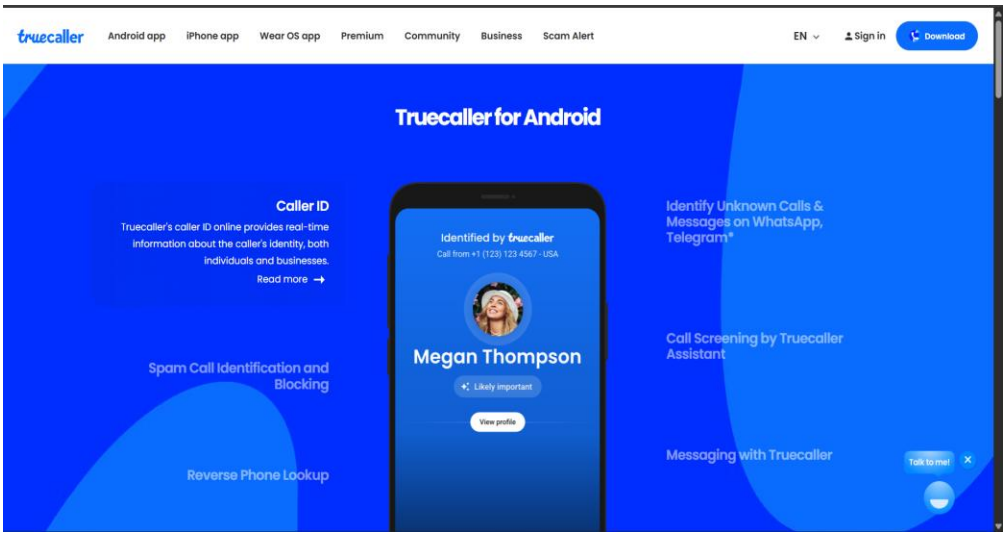

Figure 6:Truecaller Spam Detection System

Why it is exciting:

- Includes data from AI & crowdsources
- Operating in numerous nations Effective against SMS spam and scam calls

3. Solution

3.1. Explanation of expected outcomes approach to solving the problem

The purpose of the proposed Spam detection system is to classify SMS into Spam or genuine messages automatically, utilizing supervised learning classifiers. By utilizing various classification algorithms, the Spam detection system will produce more accurate results for the detection of Spam than traditional approaches because, in traditional approaches, the detection of Spam is highly dependent on rules that are not always efficient for generating precise results, since patterns are not easy to track in traditional approaches for Spam detection.

The output that the Spam detection system will generate will give not only quantitative outcomes with respect to the efficiency of the system but will also give a clear picture of

the benefit that the users shall gain by using the Spam detection system because the Spam detection system not only helps the users filter Spam messages but will also protect the users from various security threats such as phishing attacks that are basically sent through the Spam messages.

- Expected outcomes include:
  - **High Classification Accuracy**: The system should be able to classify all messages correctly, and there ought to be a small chance of either false negatives and false positives. The figure ought to exceed 95 percent for correct classifications.

  - **Efficiency of Real-Time Detection**:
    These types of messages can be classified by this system in mere seconds using optimized algorithms such as Naïve Bayes or k-NN.

  - **Robustness to Variations for Known Spam Users:**
    Through the use of machine learning algorithms that have learned from diverse sets of data, it is possible for this system to also develop an ability to handle novel instances of spam messages that have not yet been encountered in the past.

  - **Explainable Prediction Models** :
    Through these two machine learning models, it will become possible for one to see how the system to which a certain message should belong was decided upon, which would enhance the trust users build towards this system.

  - **Scalable and Flexible Solution Model**:
    The proposed system lays the groundwork for its extension to encompass additional functionalities like multimedia message filtering and/or integration with applications like Truecaller and Google Messages among others.

3.2. Explanation of the AI Algorithm Used

Artificial Intelligence Algorithm Utilized in the Proposed System

1. Naive Bayes Classifier

A probabilistic classifier, which predicts chances of a message being spam using Bayes' theorem. It assumes independence between features. Hence, it is quite effective in text classifications.

Formula (Bayes' Theorem):

$$P(C \mid X) = \frac{P(X \mid C) \cdot P(C)}{P(X)}$$

Where $C$is the class (spam or ham), and $X$is the feature vector of the message.

2. Logistic Regression

It uses Logistic Regression to predict probabilities of spam emails, which is a sigmoid function useful in binary classification.

Sigmoid Function:

$$P(y = 1 \mid x) = \frac{1}{1 + e^{-(w \cdot x + b)}}$$

Where $y = 1$denotes spam, $x$is the feature vector.

3. k-Nearest Ne

k-NN is a distance-based, collaborative filter that makes a prediction of the message class using the majority class of the top k nearest-neighbor messages in the feature space.
Euclidean Distance:

Euclidean Distance:

$$d(x_i, x_j) = \sqrt{\sum_{l=1}^{n} (x_{il} - x_{jl})^2}$$

Prediction Rule:

$$\hat{y} = \text{mode}(y_1, y_2, \ldots, y_k)$$

4. Random Forest

A Random Forest is a type of ensemble technique that involves several decision trees for better accuracy during classification and for overcoming overfitting.
Prediction Rule:

Prediction Rule:

$$\hat{y} = \text{mode}(T_1(x), T_2(x), \ldots, T_n(x))$$

Where $T_i(x)$is the prediction from the i-th tree, and $\hat{y}$is the final class.

3.3. Pseudocode

Pseudocode is an informal and high-level representation of the computer program or algorithm, where the structural patterns of programming languages are utilized but with the purpose of comprehending the code from the human side and not from the perspective of computational execution. Unlike coding, there is not much concern with the syntax or details of the code execution in pseudocode; there is much concern with the process of the algorithm rather than the details of the actual code used in the program and application. Pseudocode provides an opportunity for the programmers and analysts to conceptualize and discuss how they will resolve an issue without having to code it first and implement it into the actual application and program. Generally, pseudocode describes details from loops, if-then statements, to assigning values to some variables and calling subroutines, but all coded with simple and easy-to-understand language and coding process and syntax. For example, if there exists one for an anti-spam filter, it would depend upon steps like "Load data set," "Preprocess text," "Extract TF-IDF features," "Train classifier," and "Evaluate performance," with not much care to the details of the code syntax to implement and carry out the execution process and application.

The Pseudocode of the solution is give below:

START
IMPORT required libraries
LOAD dataset
CLEAN the dataset
PREPROCESS text data
CONVERT text to numerical features
SPLIT dataset
TRAIN models
FOR each model:
  PREDICT labels for test set
  IF prediction is correct
    Increment correct counter

```
    ELSE
        Increment incorrect counter
EVALUATE model performance
    Compare Accuracy, Precision, Recall, F1-score for all models
    Select the best-performing model
DEPLOY best model
    INPUT new SMS message
    PREPROCESS and vectorize the message
    IF model predicts spam
        Display "Spam"
    ELSE
        Display "Ham"
END
```

3.4. FlowChart

A flowchart is a graphical visualization of a process, algorithm, system, or any other set of operations, which is depicted using a variety of symbols. Flowcharts have numerous applications in a variety of fields, some of which include computer science, engineering, business, and project management. Their purpose is to communicate a process in a simple way.

This is the flow chart of Spam detection system:

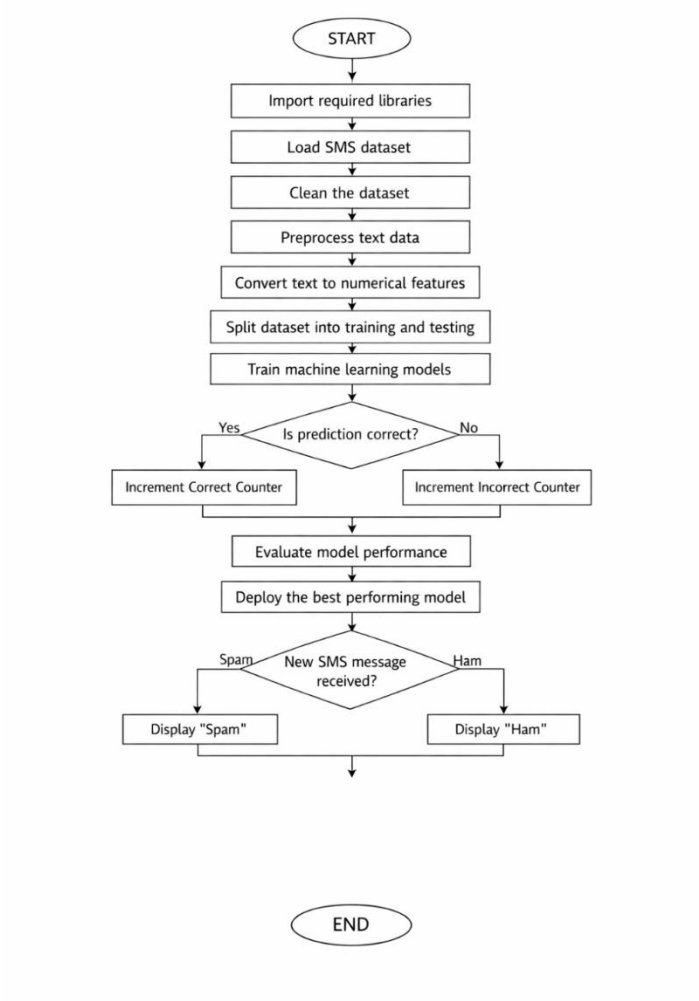

Figure 7:FlowChart

3.5. State Diagram

Another name for the State Machine Diagram is a State Diagram. This is a UML behavioral diagram that models the changes in the state of the system with regard to the occurrence of events over a period of time. This diagram models the different states in the system and how the transitions from the states are performed according to given conditions/actions. Mostly, the usage of state diagrams is related to modeling systems with behaviors relying on previous events, such as order processing systems or spam filtering systems.

Following is the state diagram of Spam Detection:

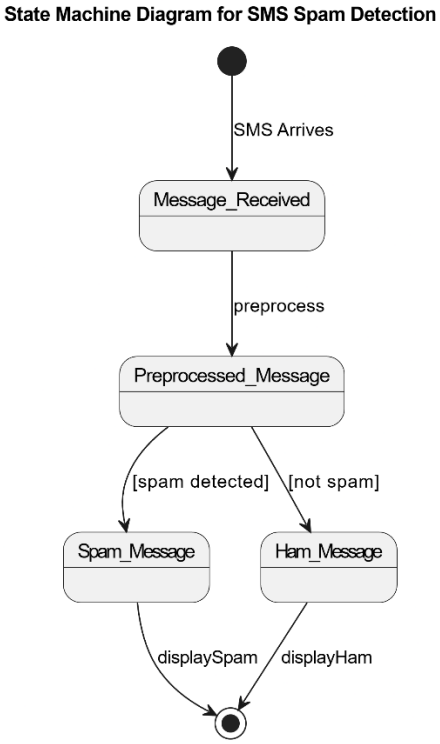

Figure 8:State diagram of Spam detection

4. Conclusion

4.1. Analysis of the work done

The topic of the project regarding spam detection involved designing an algorithm which could efficiently identify unwanted or malicious emails present in digital messaging systems such as emails and SMSs. The project began with obtaining a large number of labeled emails classified under the categories of "spam" and "ham" emails. At the preprocessing step of the project, a number of operations took place with regard to the data set, which included the ensuremnet of the absence of any inconsistencies in the data set, resolving the issue of the presence of the word "missing" in the data set, and the normalization of the data set to ensure the effectiveness of the machine learning technique used for obtaining the results of the project. Various techniques such as the use of the "BoW (Bag-of-Words)" technique took place with regard to the data set to convert the data from the text format to numeric values to be used to develop the project models. Various models of machine learning such as the use of the "Naïve Bayes" model, the use of the "Decision Tree" model, and the use of the "SVM" model took place with regard to the project. The criteria used to ensure the accuracy of the models used in the project included the use of the accuracy rate, the use of the "precision" measure, the use of the "recalls" measure, and the use of the "F1-Scores" measure with regard to the data set of the project. The data obtained illustrated the accuracy of the models used with regard to the objective of classifying the emails accurately.

- Key Analysis Points
  - Collected and labelled a big dataset related to messages
  - Applied text pre-processing techniques for cleaning and normalizing the texts
  - Applied TF-IDF and BoW Techniques for Feature Extraction
  - Validated and applied various models of supervised learning.
  - Tackled problems like noisy data and class imbalance

4.2. How the Solution Supports Real-World Scenarios

In this solution,

There are many uses of the spam detection system that is designed and employed by my project. It helps the user eliminate spam emails automatically, and thus there is greater efficiency of communication. It also protects emails from phishing and malicious URLs. Moreover, as it is automated and does not require much time and effort, it is much more reliable than other spam detection tools. It is also capable of handling a large amount of data, and thus it is useful for business mail servers and the like. It is also capable of providing insights about spam and thus can help businesses take appropriate security measures.

Real-World Benefits:

- Reduces unwanted message clutter, improving user experience
- Detects phishing and malicious content, enhancing security
- Saves time and resources through automation
- Scalable for large messaging systems
- Provides insights for proactive decision-making

4.3. Further work

Although the present spam filtering technique is efficient, there are numerous ways of upgrading the technique in the future. One of the major ways of upgrading the spam filtering technique is the implementation of the latest machine learning and deep learning techniques, including the usage of LSTM, CNN, and transformer models, which possess the capability of proper interpretation of context. The development of the service into multilingual platforms for the purpose of filtering spam will increase the applicability of the service. The incorporation of the real-time filtering technique will allow the filtering of incoming emails instantly, thus giving an immediate benefit of the service to users. The service should develop the capability of self-upgrading, where the service can upgrade itself with the help of the latest spam and threats. The service can also be implemented with the overall cybersecurity service, thus giving better and overall security and should develop the capability of customizing itself, depending on the user, for filtering spam.

- Future Enhancements:
  - Leverage deep learning models to enable effective contextual understanding
  - Implement functionality to detect spam messages in different languages
  - Facilitate real-time message filter functions for immediate protection
  - Adaptive learning for incorporation of constantly shifting Spam patterns

This is integrated with all existing cyber protection systems . Provide user-specific options for personalization .Conclusion In summary, the spam filter created not only validates the implementation of machine learning algorithms towards resolving the intricate problem of spam filtering in the modern world, but it also provides a platform for enhancement towards ensuring safe, fast, and efficient electronic communication. The level of accuracy and scalability this can achieve is invaluable in the field of digital communication, be it personal or professional.