INTRODUCTION

1.1 BACKGROUND

Today, spam has become a big internet issues. Recent 2017, the statistic shown spam accounted for 55% of all e-mail messages, same as during the previous year. This chances has been extensively exploited by irresponsible organizations and resulting to clutter the mail boxes of millions of people all around the world. Evolving from a minor to major concern, given the high offensive content of messages, spam is a waste of time. It also consumed a lot of storage space and communication bandwidth. End user is at risk of deleting legitimate mail by mistake. Moreover, spam also impacted the economical which led some countries to adoptlegislation.

Text classification is used to determine the path of incoming mail/message either into inbox or straight to spam folder. It is the process of assigning categories to text according to its content. It is used to organized, structures and categorize text. It can bed one either manually or automatically. Machine learning automatically classifies the text in a much faster way than manual technique. Machine learning uses pre-labelled text to learn the different associations between pieces of text and it output. It used feature extraction to transform each text to numerical representation in form of vector which represents the frequency of word in predefined dictionary.

Text classification is important in the context of structuring the unstructured and messy nature of text such as documents and spam messages in a cost-effective way. A Machine learning platform has capabilities to improve the accuracy of predictions. With regard to Big Data, a Machine Learning platform has abilities to speed up analysing of gigantic data. It is important especially to a company to analyse text data, help inform business decisions and even automate business processes.

For example, text classification is used in classifying short texts such as tweets or headlines. It can be used in larger documents such as media articles. It also can be applied to social media monitoring, brand monitoring and etc.

In this project, a machine learning technique is used to detect the spam message of a mail. Machine learning is where computers can learn to do something without the need to explicitly program them for the task. It uses data and produce a program to perform a task such as classification. Compared to knowledge engineering, machine learning techniques require messages that have been successfully pre-classified. The pre classified messages make the training dataset which will be used to fit the learning algorithm to the model in machine learning studio. A specific algorithm is used to learn the classification rules from these messages. Those algorithms are used for classification of objects of different classes. The algorithms are provided with input and output data and have a self-learning program to solve the given task. Searching for the best algorithm and model can be time consuming. The two-class classifier is best used to classify the type of message either spam or ham.

A tight competition between filtering method and spammers is going on per day, as spammers began to use tricky methods to overcome the spam filters like using random sender addresses or append random characters at the beginning or end of mails subject line. There is a lack of machine learning focuses on the model development that can predict the activity. Spam is a waste of time to the user since they hav e to sort theunwanted junk mail and it consumed storage space and communication bandwidth. Rules in other existing must be constantly updated and maintained make it more burden to some user and it is hard to manually compare the accuracy of classified data.

1.2 PROBLEM STATEMENT

A tight competition between filtering method and spammers is going on per day, as spammers began to use tricky methods to overcome the spam filters like using random sender addresses or append random characters at the beginning or end of mails subject line. There is a lack of machine learning focuses on the model development that can predict the activity. Spam is a waste of time to the user since they h ave to sort theunwanted junk mail and it consumed storage space and communication bandwidth. Rules in other existing must be constantly updated and maintained make it more burden to some user and it is hard to manually compare the accuracy of classified data.

1.3 OBJECTIVES

There are four objectives that need to be achieved in this project:

i. To study on how to use machine learning techniques for spam detection.

ii. To modify machine learning algorithm in computer system settings.

iii. To leverage modified machine learning algorithm in knowledge analysis software.

iv. To test the machine learning algorithm real data from machine learning data repository.

1.4 PROJECT SCOPE AND LIMITATION OF WORK

This project needs a coordinated scope of work. These scopes will help to focus on this project. The scopes are:

i. Modified existing machine learning algorithm.
ii. Make use and classify of a data set including data preparation, classification and visualization.
iii. Score of data to determine the accuracy of spam detection.

The limitation of this project are:

i. This project can only detect and calculate the accuracy of spam messages only
ii. It focus on filtering, analysing and classifying the messages.
iii. Do not block the messages.

SYSTEM REQUIREMENTS

System requirements are the configuration that a system must have in order for a hardware or software application to run smoothly and efficiently. Failure to meet these requirements can result in installation problems or performance problems. The former may prevent a device or application from getting installed, whereas the latter may cause a product to malfunction or perform below expectation or even to hang or crash.

We can specify the system requirements in terms of hardware and software system requirements. Hardware system requirements often specify the operating system version, processor type, memory size, available disk space and additional peripherals, if any, needed. Software system requirements, in addition to the aforementioned requirements, may also specify additional software dependencies (e.g., libraries, driver version, framework version).

2.1 HARDWARE REQUIREMENTS

Processor	12th Gen Intel(R) Core(TM) i5-1235U 1.30 GHz
Graphics Processing Unit (GPU)	NVIDIA GEFFORCE
Random Access Memory (RAM)	8GB
Hard Disk	500MB

2.2 SOFTWARE DEPENDENCIES

Flask	1.1.1
Numpy	**1.9.2**
Matplotlib	1.4.3
pandas	0.19
scipy	0.15.1
Jinja2	2.10.1

2.3 SOFTWARE REQUIREMENTS

Google Chrome	Used to run web based system
Microsoft Word	Creating and editing report
Github	Get Dataset
Snipping Tool	Captures and Screenshot images
Visual Studio	Implementation and deployment
Winzip	Extract the data

LITERATURE REVIEW

3.1 INTRODUCTION

This chapter discusses about the literature review for machine learning classifier that being used in previous researches and projects. It is not about information gathering but it summarizes the prior research that related to this project. It involves the process of searching, reading, analysing, summarising and evaluating the reading materials based on the project.

Literature reviews on machine learning topic have shown that most spam filtering and detection techniques need to be trained and updated from time to time. Rules also need to be set for spam filtering to start working. So eventually it become burdensome to the user

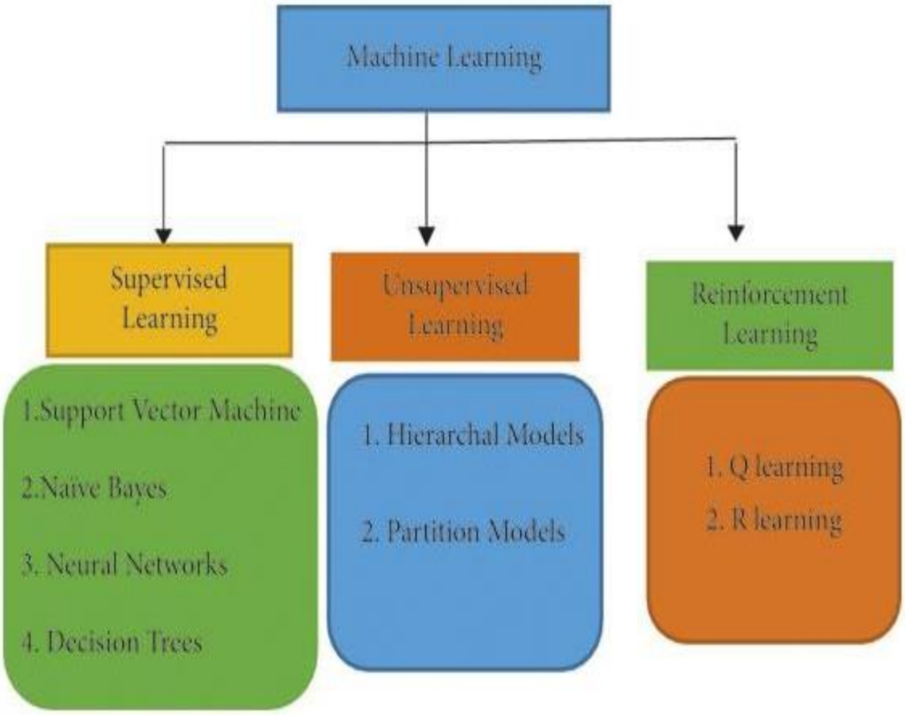
3.2 MACHINE LEARNING

In this project, existing machine learning algorithm is used and modified to fit the need of project. The reasons are because machine learning algorithm is adept at reviewing larges volume of data. It is typically improves over time because of the ever increasing data that are processed. It gives the algorithm more experience and be used to make better predictions.

Machine learning allows for instantaneous adaption without human intervention. It identifies new threats and trends and implements the appropriate measures. It is also save time as it is it automated nature.

Machine Learning-Based Spam Filtering Methods

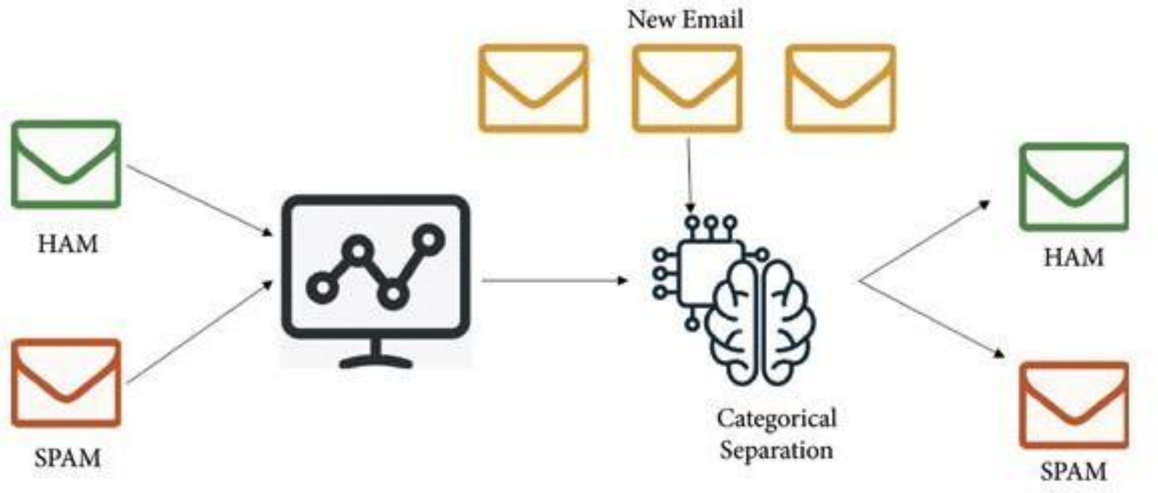
Machine learning facilitates the processing of vast quantities of data. Though it typically provides faster and more accurate results to detect unwanted content, it can also require extra time and resources to train its models for a high level of performance. Integrating machine learning with AI and cognitive computing [37] can make handling massive amounts of data even more powerful. Figure demonstrates various kinds of machine learning.



Supervised Machine Learning

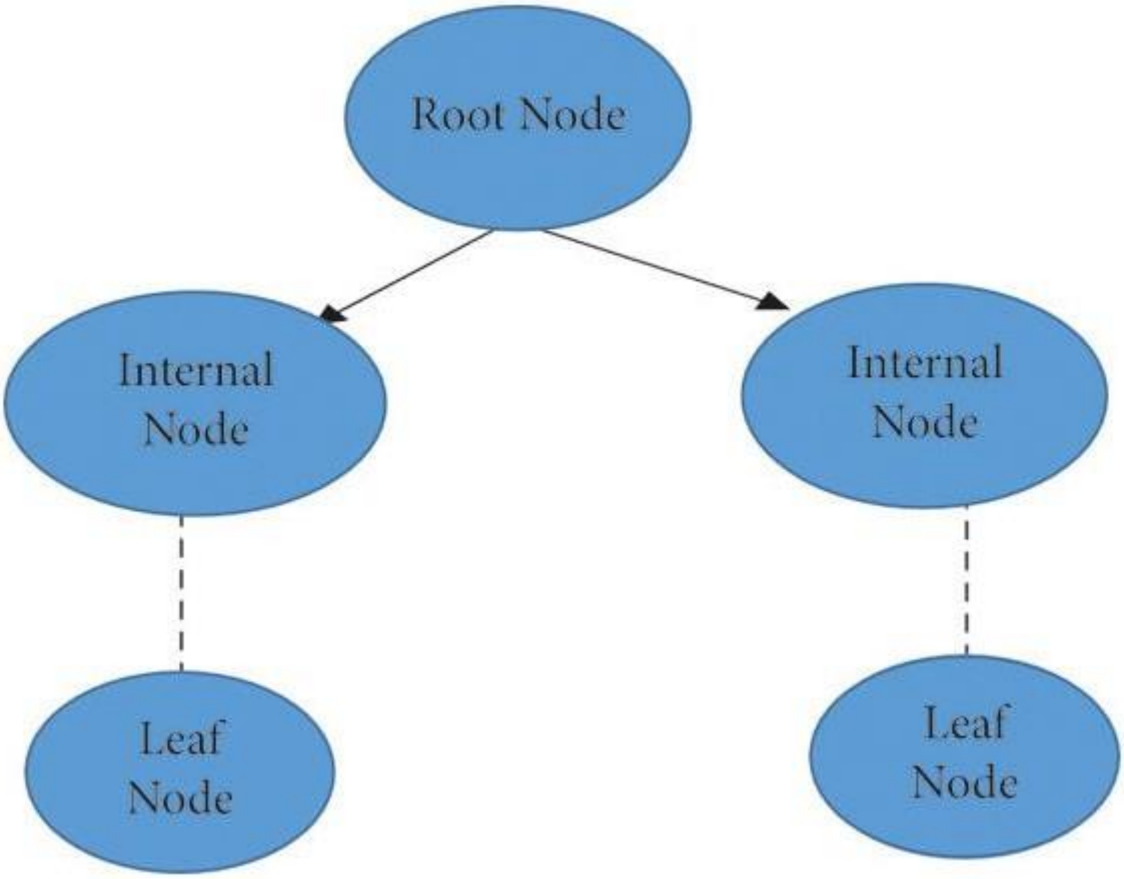
Supervised machine learning algorithms [18] are machine learning models that need labeled data. Initially, labeled training data is provided to these models for training, and after training models predict future events. In other words, these models begin with the analysis of an existing training dataset, and they generate a method to make predictions of success values. Upon proper training, the system can provide [38] the prediction on any new data related to the user's data at the training time. Furthermore, the learning algorithm accurately compares the output to the expected output and identifies errors to modify the model.

Supervised learning uses labeled data for training, and then it can predict the new data. This type of learning can be used in solving various problems, i.e., advertisement popularity, spam classification, face recognition, and object classification. The process of supervised learning is illustrated in Figure 7.



Decision Tree Classifier

Decision tree classifier is a machine learning algorithm [39], which has been widely used since the last decade for classification. This algorithm applies a simple method of solving any problem of classification. A decision tree classifier is a collection of well-defined questions about test record attributes. Each time we get an answer, a follow up question is raised until a decision is not made on the record [40]. Tree-based decision algorithms define models that are constructed iteratively or recurrently based on the data provided. The decision tree-based algorithms goal is used to predict a target variable's value on a given set of input values. This algorithm uses a tree structure to solve classification and regression problems [41]. Figure 8 shows the basic structure of the decision tree.



Figure

Structure of decision tree.

Some of the decision tree algorithms are the following:(i)Random forest(ii)Classification and regression tree (CART)(iii)C4.5 and C5.0(iv)Chi-square.

The following section deliberates some proposed email spam detection and prevention techniques by using decision tree algorithms.

DeBarr and Wechsler [42] discuss a spam filtering technique using random forest algorithms to classify spam emails and active learning to refine the classification [43]. They used the data of email messages from RFC 822 (Internet) [44] and divided each email into two sections. Then, they find term frequency and inverse document frequency of all features of each email (TF/IDF). For the training dataset, they select a set of emails with clustering to label the data. After considering the cluster prototype mails for training, they experiment with supervised machine learning algorithms: random forest, Naïve Bayes, support vector machine, and KNN [45]. The research results show that the algorithm "random forest" classifies data more efficiently with an accuracy of 95.2%.

Takhmiri and Haroonabadi [46] present a different technique to detect spams using a fuzzy decision tree and the Naïve Bayes algorithm. They use the baking voting algorithm to extract patterns of spam behaviour. They do this because obvious characteristics do not exist in the real world. The cross-linking degree for explaining or describing characters is rational and neutral. Decision trees use fuzzy Mamdani rules for the classification of spam and ham email. Then, Naïve Bayes classifier [47] is used by them on the dataset. Finally, the baking method is used by dividing votes into smaller sections. This solution gives them an optimized weight that can be implemented on obtained percentages that achieve a higher accuracy level. The dataset used in this study contains 1000 emails, from which 350 (35%) were spam and 650 (65%) were ham.

Verma and Sofat [48] used supervised machine learning algorithm ID3 [49] to render the decision trees of the problem and the hidden Markov model [50] to measure the probabilities of events that could occur as a combination to classify the emails as junk mail or ham. The proposed model initially marks all emails as spam or legitimate by measuring each e-mail's total likelihood with the aid of subsequently classified email terms. After that, it makes the decision trees of emails one by one. The Enron dataset [51] is used in this study that contains 5172 emails. From all 5172 emails, 2086 were spam, while 2086 were legitimate emails. Their model can categorize the emails as spam and ham by using the feature set obtained by the Enron dataset. They got an 11% error by using the sklearn library's fitness function in the proposed model. Their model got 89% of accuracy results on the given dataset.

Li et al. [52] proposed an email-classification technique for IoT systems based on supervised machine learning. They use a multiview technique that focuses on the collection of richer information for classification. A double view dataset is created with internal and external feature sets. The proposed approach can be used in both labeled and unlabeled data and was evaluated on two datasets with a real network environment. The results of this study indicate that the multiview model can achieve more accuracy than simple email classification. In the end, the multiview model is compared with various existing models.

A spam filtering approach based on different decision tree algorithms is presented by Subasi et al. [40] to compare the accuracy and find the best one for their dataset. They implement classification and regression tree (CART), C4.5, REP tree, LAD tree, NBT, random forest, and rotation forest algorithm on the dataset to classify emails. Their results show that the proposed modified random forest model got the highest accuracy than other decision tree methods for publicly available datasets. **Support Vector Machine (SVM)**

The support vector machine (SVM) is an essential and valuable machine learning model [53]. SVM is a formally defined discriminative supervised learning classifier that takes labeled examples for training and gives a hyperplane as output, classifying new data [54]. A set of objects belonging to various class memberships are separated by decision planes. Figure 9 shows the classification concept of linear support vector machines. In the figure, some circles and stars are called objects. These objects can belong to any of two classes, i.e., the class of stars or dots. The isolated lines determine the choice of objects between green and brown objects. On the lower side of the plane, the objects are brown stars, and on the upper side of the plane all objects are green dots showing that two unique objects are classified into two different classes. If a new object black circle is given to the model, it will classify that circle into one of the classes according to the training examples provided in the training phase.

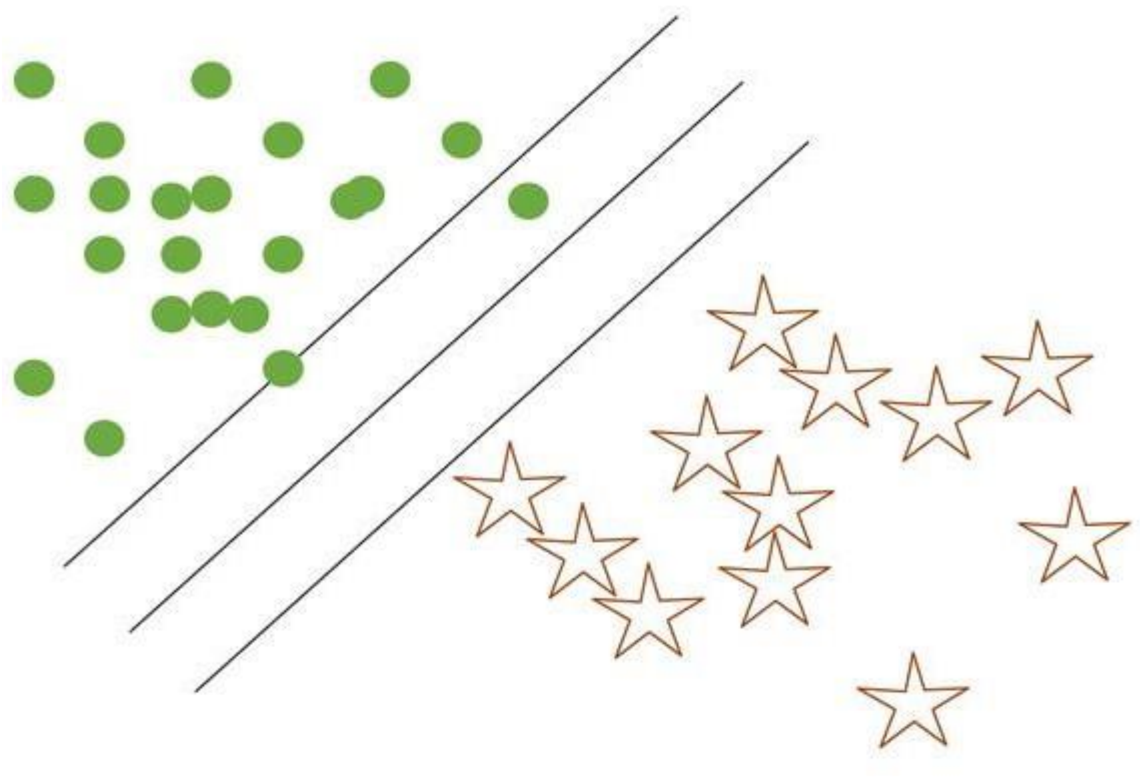


Figure _

Support vector machine classification.

Banday and Jan [55] present research in which they define the procedure of statistical spam filters. They design those filters using Naïve Bayes, KNN, support vector machines (SVM), and regression trees [56]. They use all these supervised machine learning algorithms and evaluate the results based on precision, recall, and accuracy. Using these machine learning techniques, they found that classification and regression trees (CART) [57] and Naïve Bayes classifiers are the most effective algorithms for the dataset. This approach estimates that, during spam filtering, calculations of false positive are costlier than a false negative.

Zheng et al. [12, 58] present a procedure for detecting spammers and spam messages in any social network. Today, everyone uses social media, and many social media users spend a considerable amount of time communicating with their loved ones. The spammers take advantage of various social media networks and users' posts to send malicious content, advertisements, information, etc., into the social media user's profiles. So, this paper discusses how to detect those posts or malicious content on social media platforms. Their study uses the Sina Weibo social network [59] and machine learning algorithm support vector machine (SVM) for the detection of spammers. The dataset that was used in this study was 16 million messages that were collected from several users. They used 18 features as a feature vector set. The clients of the networks were divided into two categories, legitimate users and spammers. 80% of data was used for the model's training, while 20% was used for testing. For better accuracy, they used 1 : 2 between spammers and nonspammers of the training dataset. With this ratio, the proposed model gives an accuracy level of 99.5% for classifying spammers and nonspammers [60].

A novel fitness framework based on IoT-enabled blockchain technology and machine learning techniques is presented by Jamil et al. [10]. Their proposed model is composed of two modules. The first one is a blockchain-based network used for the security of sensing devices and an intelligent contract-enabled relationship and an inference engine that uncovers hidden insights and usable information from IoT and user device data. The improved smart contract gives users a useful application that allows real-time monitoring, more control, and quick access to several devices distributed across various domains. The inference engine module attempts to uncover underlying patterns and usable information from IoT environment data, assisting in effective decision-making and providing convenient services. Their proposed model can be used to improve system throughput and resource usage, according to their findings. The proposed system in this article may be used in various fields, including healthcare and smart businesses.

Olatunji [61] developed a spam filtering tool using support vector machine and extreme learning machine algorithms. He used the standard dataset for the development of the spam detection model. SVM got an accuracy of 94.06% in his work, and the extreme learning machine (ELM) model got a 93.04% accuracy level, suggesting just 1.1% performance improvement that SVM achieved over ELM. He indicated that SVM's improvement over ELM accuracy is marginal. It implies that, in situations where detection time is critical, as in real-time systems, the ELM spam detector should be given preference over SVM spam detection. Although SVM got a higher accuracy level in his research, it takes more time for training than the ELM system. Tretyakov [62] also discussed various machine learning techniques for email spam filtering. This paper compared the precision results between false positives and precision results after eliminating false positives. They show the result after eliminating false positives, which were more accurate and reliable than before.

Naïve Bayes Classifier (NB)

The Naïve Bayes classifier [47] is based on the Bayes theorem. It assumes that the predictors are independent, which means that knowing the value of one attribute impacts any other attribute's value. Naïve Bayes classifiers are easy to build because they do not require any iterative process and they perform very efficiently on large datasets with a handsome level of accuracy. Despite its simplicity, Naïve Bayes is known to have often outperformed other classification methods in various problems.

Rusland et al. [63] present research on email spam filtering and perform the analysis using a machine learning algorithm Naïve Bayes. They used two datasets evaluated on the value of accuracy, F-measure, precision, and recall. As we know, Naïve Bayes uses probability for classification, and the probability is counting the frequency and combination of values in a dataset. This research uses three steps for the filtration of emails, i.e., preprocessing, feature selection, and, at last, it implements the features by using the Naïve Bayes classifier. The preprocessing step removes all conjunction words, articles, and stop words from the email body. Then, they used the WEKA tool [64] and made two datasets called spam data and spam base dataset. The average accuracy was 89.59% using two datasets, while the spam data got 91.13% accuracy. The spam base dataset got an accuracy of 82.54%. The average precision results for spam data were 83%, while, for spam base, the precision result was 88%. They claimed that the Naïve Bayes classifier performs better on spam base data as compared with spam data.

Arif et al. [11] presented an article on machine learning-based spam detection techniques for IoT devices. They used five ML models and analyzed their results using various performance metrics. A large number of input features were used for the training of proposed models. Each model calculates a spam score based on the input attributes. This score represents the trustworthiness of an IoT device based on a variety of factors. The suggested approach is validated using the REFIT smart home dataset. They claim that their proposed system can detect spam better than currently used spam detection systems. Their work can be utilized in smart homes and other places where intelligent devices are used.

Kumar et al. [14] discussed email spam detection using various ML algorithms. Their article explores ML methods and how to implement them on datasets. The optimal algorithm for email spam detection with the highest precision and accuracy is identified from various ML algorithms. They concluded that the Multinomial Naïve Bayes algorithm produces the best results, but it has limitations due to class-conditional independence, which causes the machine to misclassify some inputs. Ensemble models come after Multinomial Naïve Bayes with the best and reliable results in this study. The proposed system in this study can only detect spam from the body of emails.

Singh and Batra [65] proposed a semisupervised machine learning technique for spam detection in social IoT platforms. They used an ensemble-based framework that is consists of four classifiers. The architecture is based on the use of probabilistic data structures (PDS) such as Quotient Filter (QF) to query the database of URLs, spam users, databases of spam keywords, and Locality Sensitive Hashing (LSH) for similarity search. The proposed model minimizes, so it decides by an adaptive weighted voting approach based on each classifier's output. The hybrid sampling technique minimizes the computational efforts, which sample the data according to each classifier. This study indicates that the proposed model can be used for spam detection on large datasets. The proposed model's efficiency was evaluated by comparing PDS with standard data models and the typical evaluation metrics, including accuracy, recall, and F-score.

Artificial Neural Networks

An artificial neural network (ANN) is a computational model based on the functional aspects of biological neural networks, also known as the neural network (NN) [66]. Many sets of neurons are joined in a neural network, and information is interpreted using a computational approach connection. In most situations, an ANN is an adaptive system, which changes its structure depending on external or internal information flowing through the network during the learning phase. Current neural networks are nonlinear approaches to statistical data processing. These are commonly used when there are complex relationships between inputs and outputs or unusual performance patterns [6]. Figure 10 shows the basic structure of the neural network.

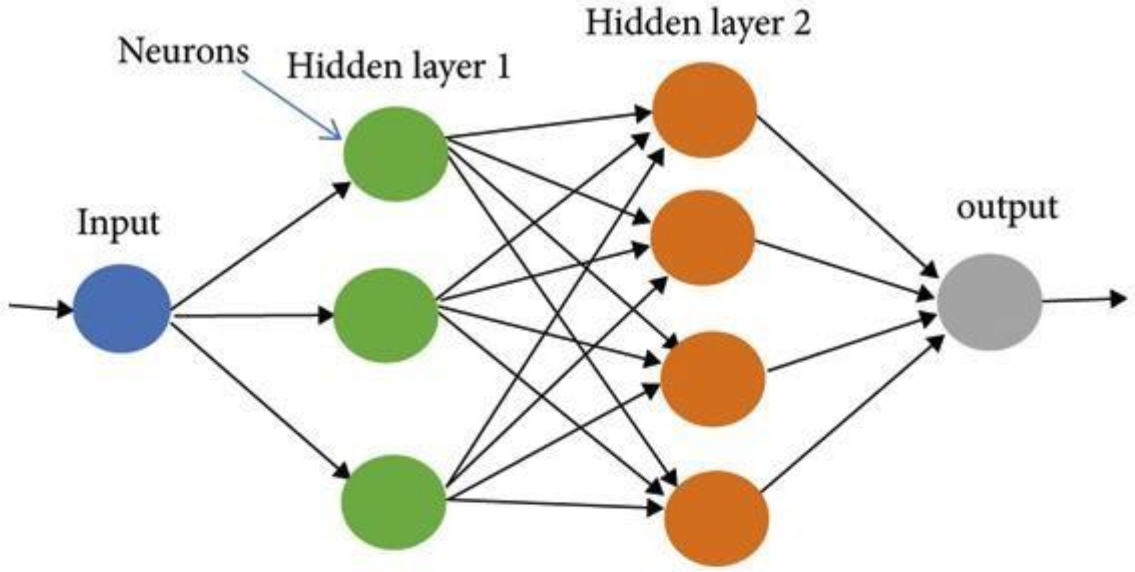


Figure _

Basic structure of neural network.

The following section elaborates some proposed email spam detection and prevention techniques by using neural networks.

Xu et al. [67] present a method for the detection of spam in online social networks. Their work focuses on the combination of spam messages in one social network to another social network. By using Twitter, they gathered 1937 spam and 10943 ham tweets for processing. They also used 1338 spam posts and 9285 ham posts. In TSD, 75.6% of tweets contained URL links for spam tweets, while 24.4% contained different words. Out of 10942 ham tweets, 62.9% contained URL links and words, while 37.1% had only words. For the spam posts of FSD, 32.8% of posts consist of different web links, and the remaining 67.2% of spam posts contain only words [68]. Of 9285 ham posts, 95.1% have web links, and the other 4.9% consist of words. They used the top twenty feature words from Facebook spam data and Twitter spam data. They divide the TSD and FSD into two sets, i.e., training dataset and testing dataset. These datasets were used to train various machine learning classifiers like Naïve Bayes, random forest, logistic regression random tree, and Bayes Net. After analyzing the accuracy of different classifiers, they combine the spam dataset of Facebook into the training dataset of Twitter and the spam dataset of Twitter into the training dataset of Facebook. Then, they used the combined dataset for the training and testing of classifiers. In the end, they compare the results of classifiers on the above-mentioned social networks after measuring the precision, accuracy, recall, and F-1 measure. They found that the accuracy of combined datasets was higher than that of other datasets [68, 69].

Guo et al. [70] proposed a spammer detection technique using a collaborative neural network in IoT applications. They present a novel spam detection mechanism called Cospam for IoT applications. At first, the user and contents of speech at different timestamps are viewed as feature sequences. In the second step, a collaborative neural network model is used. The collaborative model consists of three models: (1) Bi-AE model, (2) GCN model, and (3) LSTM model. These models are used for the identification of the nature of the user. In the end, a series of experiments were conducted for the evaluation of the proposed technique. The proposed model was able to obtain 5% more accuracy than existing spammer detection approaches. Cospam consumes more time than existing techniques because of a large number of parameters.

Makkar and Kumar [71] proposed a deep learning model for web spam detection in an IoT environment. Their system enhances the cognitive ability of search engines for the detection of web spam. This model removes spam pages with the help of a web page rank score calculated by a search engine. Their framework uses the extensive features of deep learning. The first time in which the LSTM model was used to detect spam is used for many problems like weather forecasting. In this study, the proposed model is compared with ten different machine learning models. The WEBSPAM-UK 2007 standard dataset is used in this study. The preprocessing of the dataset is done by a novel technique called “Split by Oversampling and Train by Underfitting.” The accuracy of the proposed model was 95.25%. After the optimization of the system, the proposed model got an accuracy of 96.96%.

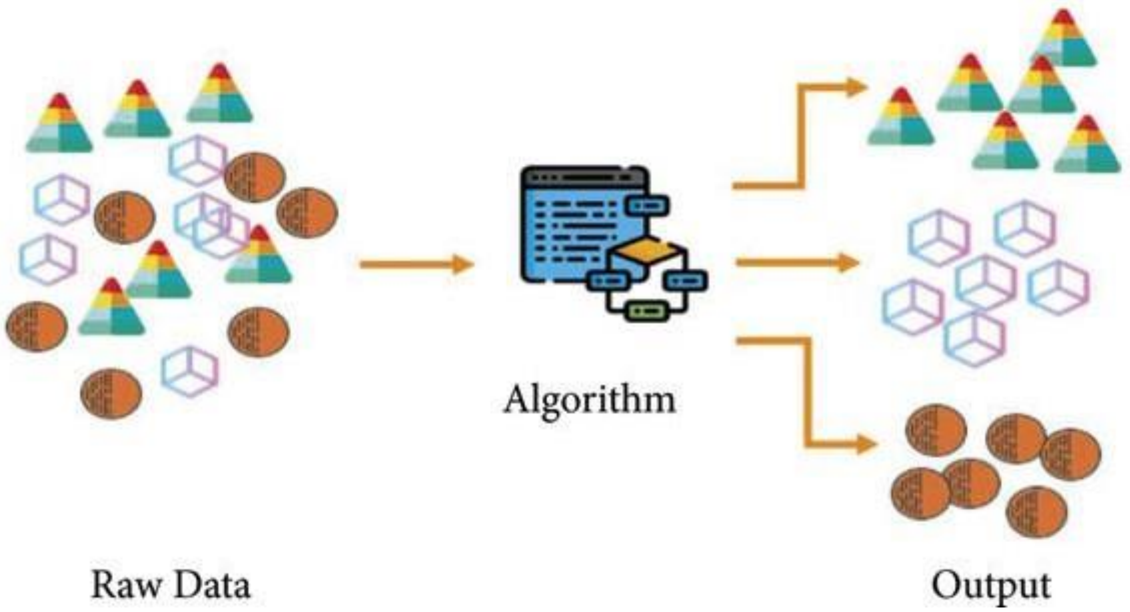
Zavvar et al. [72] present a paper on spam detection by considering combined particle swarm optimization and neural networks to select features. They also used SVM for classifying and separating spam. They compared the proposed approach with other approaches such as a self-organizing map and k-means data grouping based on the region under curve parameters. This article uses the UCI base dataset to evaluate spam classification and provide a PSO-ANN and ANFIS algorithm-based approach for spam detection. Seventy percent of data was used for training, and 30 percent was used for testing the models. RMSE, NRMSE, and STD principles were analyzed and got 0.08733, 0.0185, and 0.08742 results in the testing phase. The results show that the proposed method has good accuracy and performance for detecting spam emails. Table 2 summarizes supervised machine learning techniques presented for spam detection.

Discussions and Learned Lessons

Supervised machine learning techniques, i.e., decision trees, random forests, support vector machines, and artificial neural networks, can be used for email spam detection or filtering. Support vector machines classify different objects by using the idea of the hyperplane. Objects are classified into two classes. If a new object is given to the model, it will be classified into one of both classes. Zavvar et al. [12], Garavand et al. [72], and Idris et al. present different techniques for spam detection using the support vector machine (SVM) model. They got a good accuracy level on different spam datasets. Olatunji et al. [73] used the support vector machine and extreme learning machine algorithms on the standard dataset and got 94.06% accuracy using the support vector machine. In their system, extreme learning machines perform better than SVM but take more time, so a time-consuming ELM performs better than SVM. Zheng et al. got the highest accuracy level using Weibo social network dataset. They use two types of features, i.e., content base and user behavior base, to classify spammers and nonspammers. Naïve Bayes classification is another supervised machine learning technique, which predicts some events based on its naïve theorem. Naïve Bayes classifiers are quite simple, and they do not use an iterative process; they perform very efficiently on large datasets with a handsome level of accuracy. Hijawi et al. [41] use the Naïve Bayes network for the detection of spam. They did not get outstanding results using the spam assassin dataset as their accuracy level was only 89%. Another technique which is widely used in the last decade is decision tree. These decision algorithms define models that are constructed iteratively or recurrently based on the data provided. The decision tree-based algorithms goal is to predict a target variable's value on given set of input variables. Subasi et al. [40] used different decision tree-based algorithms for spam detection on the UCI machine learning platform dataset. They used 10-fold cross-validation for the evaluation of decision tree classifiers. They use open-source Weka tools for the development of the model. DeBarr and Wechsler [42] used a tree-based random forest algorithm for email spam detection and active learning for refining the classification. They used the data of email messages from RFC 822 (Internet) and got the highest accuracy level of 95.2% by using the dataset's custom collection of emails. In all supervised machine learning techniques, Zheng et al. [12] got the highest accuracy level among all researchers using the support vector machine (SVM) technique for email spam detection.

Unsupervised Machine Learning

Unsupervised machine learning algorithms are used when we do not have labeled data [74]. Unsupervised learning explores how programs can explain a hidden structure by inferring a feature from unlabeled data [75]. The machine does not evaluate the appropriate output but examines the data and can draw inferences from datasets to explain hidden constructs from unlabeled data. Unsupervised learning works on unlabeled data and makes clusters of the data based on the features of that data. This type of learning can be used for various problems like Recommender Systems, identifying Buying Habits, Grouping User Logs, dimensionality reduction, etc. The process of unsupervised learning is illustrated in Figure 11.



Figure

Process of unsupervised learning.

Clustering is the main application of unsupervised learning that has two main types. Different clustering techniques are discussed as follows.

Hierarchical Clustering

Hierarchical clustering identifies clusters with a hierarchy achieved either by iteratively combining smaller clusters into a more significant cluster or by splitting a more massive cluster into smaller clusters. This cluster hierarchy, generated through a clustering algorithm, is called a dendrogram [76]. A dendrogram is one way of representing the hierarchical clusters. The user can understand different clusters based on the level at which the dendrogram is defined. It uses a similarity scale representing the distance between the clusters grouped from the massive cluster. A dendrogram is a visual representation of hierarchical clustering that is illustrated in Figure 12.

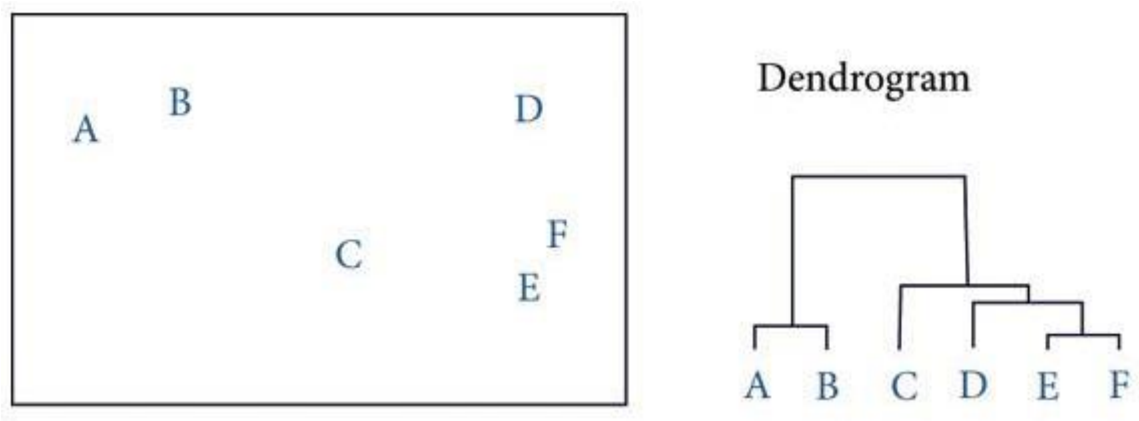


Figure _

Structure of dendrogram.

Partitional Clustering

A partitional clustering divides a single set of data objects into nonoverlapping subsets (clusters) so that each data object is in only one subset [77]. Partitional clustering algorithms make different partitions of data and then evaluate the required results based on some criteria. Figure 13 illustrates the basic structure of partitional clustering algorithms. In Figure 13, partitions (A, B, and C) are created based on some characteristics. Partitional clustering breaks down a dataset into a collection of clusters of disjoints. The partitioning technique forms different partitions of data by using the formula K (N/K); each partition represents a cluster based on a set of N points in the data, that is, by fulfilling the following conditions:(1)Each class contains one point or more(2)Each point comes as part of exactly one group

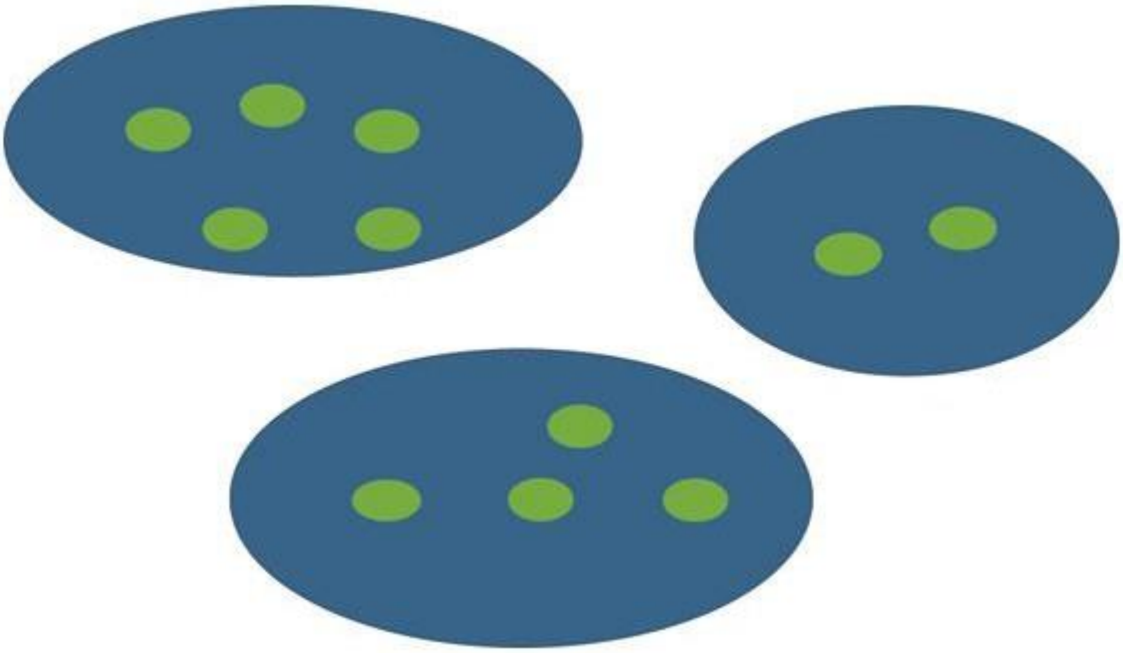


Figure _

Partitioned clustering structure.

Let us discuss some work on filtering email spam using unsupervised machine learning techniques.

Sharma and Rastogi [78] propose a strategy using unsupervised techniques. They performed various experiments on email spam datasets. After data gathering, they use the k-means clustering model for the clustering of emails. They use various distance measures for this purpose. The study's findings show that the proposed model performs well and cluster spam and ham emails are efficient.

Tan et al. [79] developed a reliable model for spam detection. First, they present a Sybil defense-based automated spam detection scheme called SD2, which considerably outperforms current techniques by considering the social network relationship. They further developed an unsupervised spam detection system called UNIK to address increased spam attacks effectively. Instead of directly detecting spammers, UNIK operates by intentionally eliminating nonspammers from the network. They used the social graph as well as the user-link graph for the detection of the spammer. UNIK's fundamental basis is that spammers actively change their patterns to avoid detection, while nonspammers are not expected to do so. Therefore, we have a reasonably nonvolatile pattern. When tested on a broad network platform, UNIK has a similar performance as SD2 and substantially beats SD2 as spam attack rates go up. They evaluate several known spam activities in the social network platform by the identification of UNIK. Their proposed system, UNIK, can be used for email spam classification. The result shows that various spammer clusters exhibit different characteristics, suggesting the instability of spamming and UNIK's ability of automatically extracting junk mail signatures.

Ahmed [80] used an improved digest algorithm with DBSCAN clustering to classify spam emails. They create a different digest (parts) of emails before clustering. Their proposed model has two key steps. When the system receives emails, it first enters the digest generation phase, where an improved digest algorithm processes it, and the output is the set of digests of each email. These digests are then given to the clustering algorithm, i.e., DBSCAN, in the next phase. In the clustering phase, similar emails are classified in the clustering process in a cluster of spam mails based on similarities among their digests, where mails that do not look like any other digest are considered noise and not clustered. Such emails that are not clustered are standard (ham) emails.

Using unsupervised artificial neural networks (ANNs), Cabrera-León et al. [81] propose a hybrid antispam filter. Their method contains two main steps. The first step is preprocessing of content, and the second one is actual processing. Each step is based on various models of computation. These models are "programmed and neural (using Kohonen SOM) [55]. This proposed system used the Enron dataset for ham or legitimate emails, while for spam emails they used two distinct sources. The first phase preprocessing was done based on thirteen (13) thematic features found in spam and ham emails. The terms frequency (TF) and inverse term frequency (IDF) were used in their system for the sake of feature extraction. Their results were the same as those of other researchers for the same dataset since they use distinct machine learning techniques and attributes. They evaluated their system with various datasets, defined by interdependent origins, ages, users, and forms like image spam samples. Their system got an accuracy level between 75% and 96%. They show that model performance degradation can vary by variations, in datasets, especially in dates. This phenomenon is known as "topic drift." Generally, it affects all classifiers, but it more affects those classifiers that use offline learning. The same case is with adversarial machine learning problems like spam filtering. Their method is robust to phrase obfuscation, which is commonly used in spam content. It was also independent of the need to use lemmatization or stemming.

Sasaki and Shinnou [82] introduce a new approach for spam detection using the vector-space model of content clustering. Their system automatically calculates disjoint clusters using a spherical k-means technique for all spam and nonspam emails. It collects centroid vectors of clusters for the extraction of vector definition. Each centroid is labeled with spam and nonspam to measure several spam emails in the clusters. The system measures the cosine similarity between the current mail vector and the centroid vector as a new email arrives. Eventually, the new mail is assigned the label of the most appropriate cluster. They obtain several kinds of spam and nonspam email topics by using the proposed approach and effectively identifying the spam emails. They introduce the spam detection framework in this paper and demonstrate the research outcomes utilizing the series of Ling-spam datasets. They got 98.06% accuracy with their model.

Narisawa et al. [83] suggest an unsupervised approach for detecting spam documents from several documents relying on string equivalence. They provide three metrics to quantify a string's alienation, which means how distinct they are inside the documents from other substrings. In their proposed model, a document labeled as spam includes a substring with a significant alien degree in an equivalence class. The proposed approach was unsupervised, independently of language, and scalable. Japanese web forum data were used for computational experiments to show the proposed approach's performance on real data. Table 3 presents comparison of unsupervised learning techniques used for spam filtering.

Discussion and Learned Lessons

Several unsupervised machine learning models are being used for email spam detection and filtering. Hierarchical clustering and partitioning clustering are commonly used clustering techniques. Ahmed [80] used DBSCAN clustering and an improved digest algorithm to classify emails. He used the spam assassin dataset for the development of his model. This approach significantly enhances filtering accuracy by 30 percent against the newly proposed algorithms and increases spam detection tolerance against increased spammer's obfuscation effort while maintaining successful email detection at a comparable level of older filtering methods.

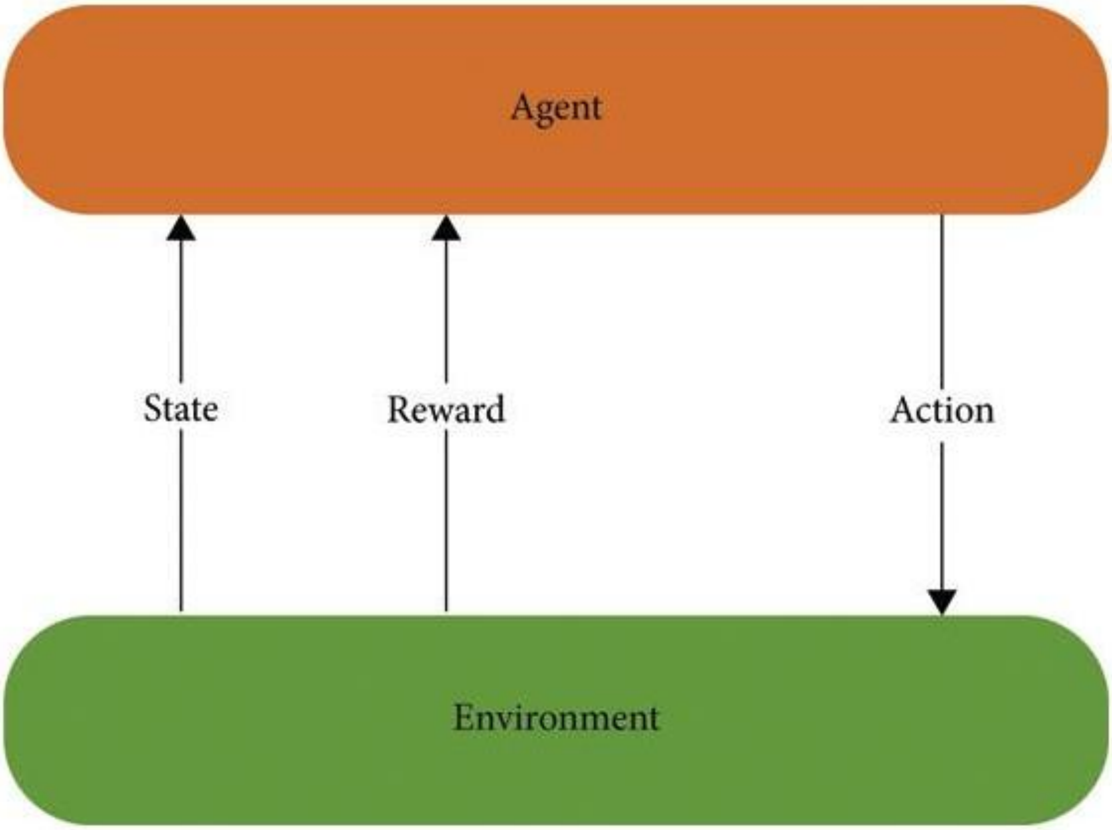
Sharma and Rastogi [78] used a machine learning algorithm (k-mean clustering) with local concentration-based content extraction for spam detection and got a handsome accuracy level. Cabrera-León et al. [81] used an artificial neural network that contains two necessary steps. In the first step, they do preprocessing and then in the second step they process cleaned data for computing the results. These steps are based on distinct models of computation. Its accuracy was 95%. Narisawa et al. [83] introduced an unsupervised approach to identify a spam document from a collection of documents based on string equivalence. This solution was a language-independent and scalable method for spam detection. It was tested on the Japanese web forum. Among all the researchers, Sharma Rastogi [78] and Ahmed et al. got the highest accuracy level using DBSCAN and

K-mean algorithm, respectively, for the email spam detection. Ahmed [80] used spam assassin dataset for the implementation of his model.

5.3. Reinforcement Machine Learning

Reinforcement learning is another type of machine learning which works on reward taken from its environment. It takes suitable actions to make or get the maximum reward in a given situation [84]. Many machines and software employ it to find the optimal path to take in a specific situation.

The main difference between supervised and reinforcement learning is that supervised learning needs training data with correct labels. Simultaneously, there is no correct label in reinforcement learning, but the agent decides what to do to perform the given task. The agent is bound to learn from its experience if there is no training dataset [85]. Figure 14 illustrates the simple reinforcement learning process in which an agent passes an action to the environment. The environment sends back the reward of action and state to the agent. Let us discuss some research work done on email spam detection using reinforcement learning.



Figure

Basic structure of reinforcement learning.

Chiu et al. [86] propose an alliance-based approach to classify, identify, and exchange relevant information on spam email contents. Their spam filter consisted of a rough set theory, a machine learning classifier (XCS), and a genetic algorithm. They used several metrics to evaluate the model results. From their paper, two main conclusions can be drawn, and they are given as follows: The spam filter is based on a combination of rough set theory, genetic algorithm, and machine classifier XCS. Many metrics are used to assess spam mails filtering results by an alliance-based approach and provide a reasonable output indicator. They may draw two key conclusions which are the following:(a)The rules that have been shared from many other email servers do help the spam filter to block more spam emails than before(b)A blend of several techniques increases precision and decreases false positives for the spam detection task

3.3 SPAM DETECTION

In theory, spam detection can be implemented at any location and multiple stages of process can occur at the same time. Figure 1 shows the spam detection process.

Spam Messages

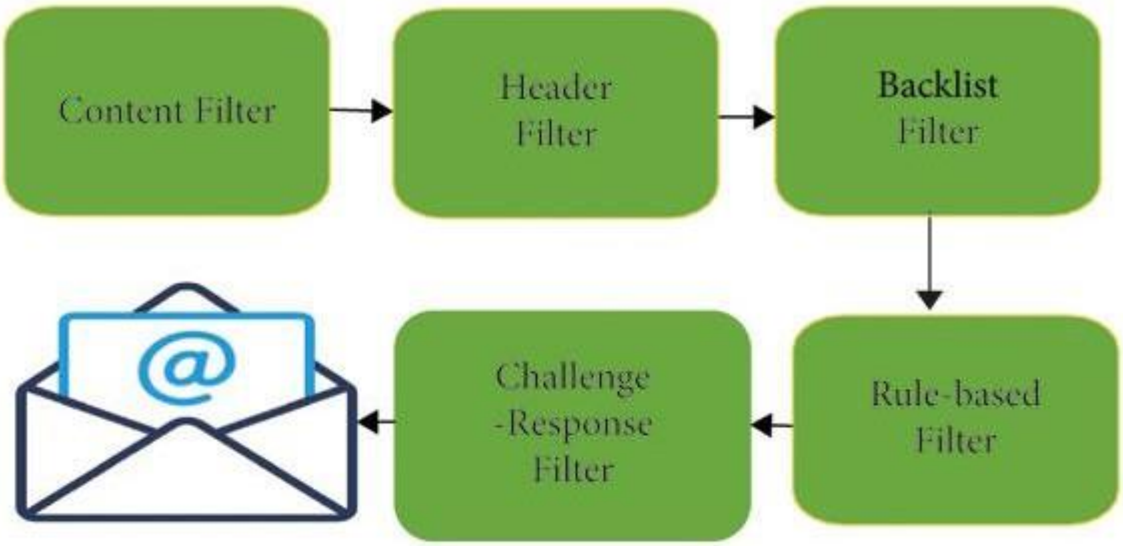
The email spam definition is ambiguous since everybody has their views on it. At present, email spam is getting the attention of everyone. Email spam ordinarily includes particular spontaneous messages sent in mass by individuals you do not know. The term spam is obtained from the Monty Python sketch [23], in which the Hormel canned meat item has numerous tedious emphases. While the term spam was purportedly first utilized in 1978 to allude to unwanted email, it increased rapidly in the mid-1990s, as we get to turn out to be progressively typical outside scholastic and research circles [24]. A notable model is the development expense trick in which a client receives an email with an offer that should bring about a prize. In the era of technology, the dodger/spammer shows a story where the unfortunate casualty needs forthright financial help so that the fraudster can gain a lot bigger total of cash, which they would then share. The fraudster will either earn a profit or avoid communication when the unfortunate victim completes the installment.

Spam Filtering Methods in Email and IoT Platforms

The number of spam emails is rapidly increasing in marketing, chain communications, stock market tips, politics, and education [24]. Currently, various companies develop different techniques and algorithms for efficient spam detection and filtering. We address some filtering strategies in this section to understand the filtering process.

The Standard Spam Filtering Method

Standard spam filtering is a filtering system that implements a set of rules and works with that set of protocols as a classifier. Figure 1 illustrates a standard method for filtering spam. In the first step, content filters are implemented and use artificial intelligence techniques to figure out the spam [25]. The email header filter, which extracts the header information from the email, is implemented in the second step. After that, backlist filters are applied to the emails to clinch the emails coming from the backlist file to avoid spam emails. After this stage, rule-based filters are implemented, recognizing the sender using the subject line and user-defined parameters. Eventually, allowance and task filters are used by implementing a method that allows the account holder to send the mail [26].



Figure

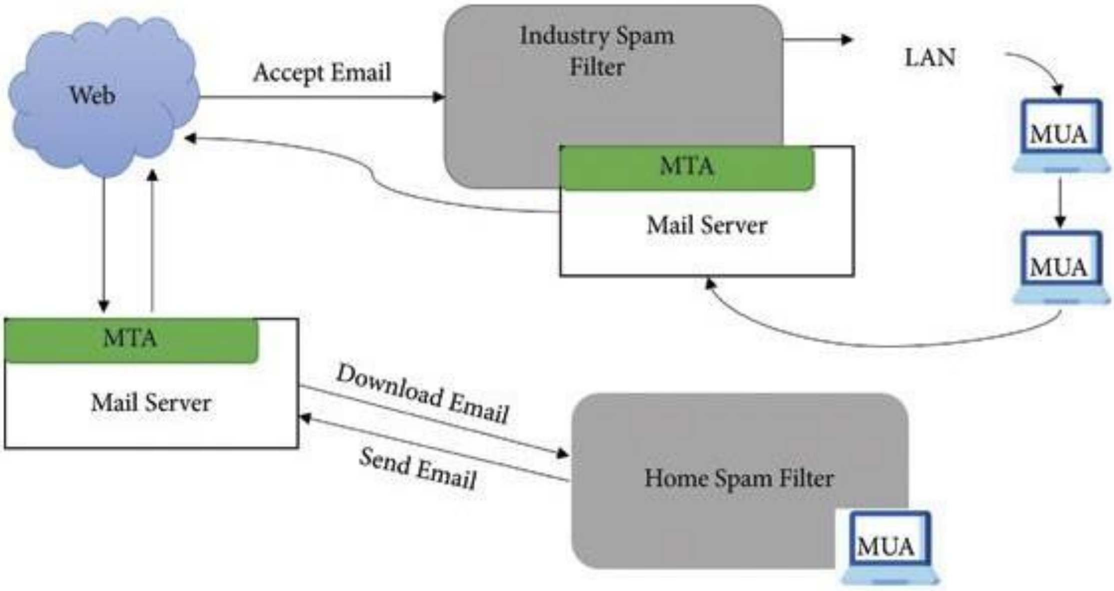
Standard spam filtering.

The Client Side Spam Filtering

A client is a person who can use the Internet or email network to send or receive an email [27]. Spam detection at the client point offers different rules and mechanisms to ensure secure communications transmission between people and organizations. For transmission of data, a client should deploy multiple existing frameworks on his/her system. Such systems connect with client mail agents and filter the client's mailbox by compositing, accepting, and managing the incoming emails [28, 29].

Enterprise Level Spam Filtering

Email spam detection at the enterprise level is a technique in which various filtering frameworks are installed on the server, dealing with the mail transfer agent and classifying the collected emails into one spam or ham [30]. This system client uses the system consistently and effectively on a network with an enterprise filtering technique to filter the emails. Existing methods of spam detection use the rule of ranking the email. A ranking function is specified in this principle, and a score is generated against every post. The junk mail or ham message is given specific scores or ranks [31]. Since spammers use different approaches, all tasks are regularly modified by implementing a list-based technique to block the messages automatically. Figure 2 is reproduced from Bhuiyan et al. [20]. Figure 2 shows the architecture of the client and enterprise level spam filtering process.

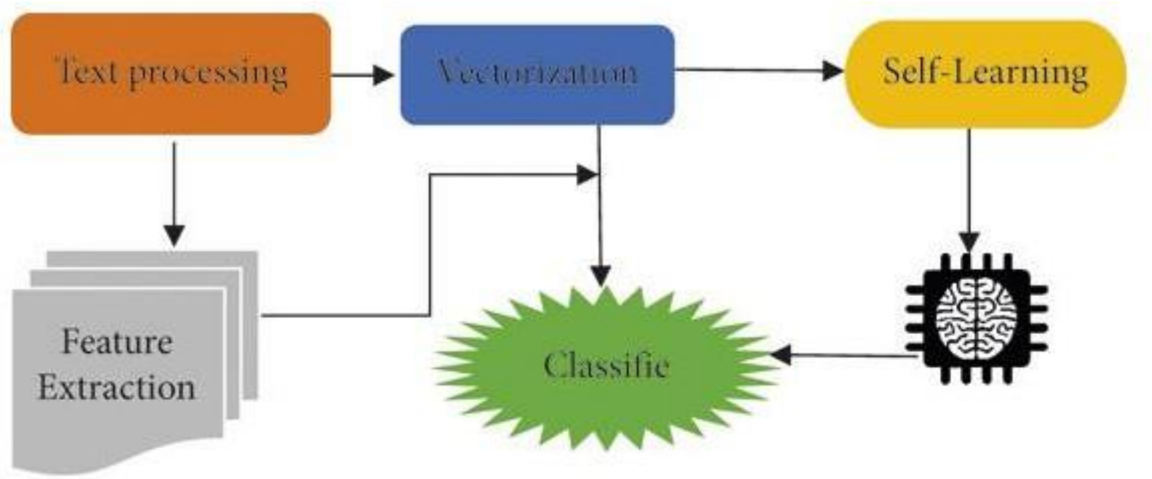


Figure

Client based and enterprise level spam filtering [20].

Case-Based Spam Filtering

One of the well-known and conventional machine learning methods for spam detection is the case-based or sample-based spam filtering system [32]. A typical case base filtering structure is illustrated in Figure 3. There are many phases to this type of filtering with the aid of the collection method; it collects data (mails) during the first step. After that, the major transition continues with the preprocessing steps through the client graphical user interface, outlining abstraction, and choice of email data classification, testing the entire process using vector expression and classifying the data into two classes: spam and legitimate email.



Figure

Case-based spam filtering.

Finally, the machine learning technique is extended to training sets and test sets to determine whether this is an email. The final decision is made through two steps: self-observation and classifier's result, deciding whether the email is spam or legitimate

Challenges of Spam Detection

Some critical challenges faced by spam filters are discussed as follows:

(i) The growing amount of data on the Internet with various new features is a big challenge for spam detection systems.

(ii) Features' evaluation from several dimensions such as temporal, writing styles, semantic, and statistical ones is also challenging for spam filters.

(iii) Most of the models are trained on balanced datasets, while self-learning models are not possible.

(iv) Many spam detection models face adversarial machine learning attacks that will decrease their effectiveness. Adversaries can throw a variety of attacks during the training and testing of ML models. Adversaries can harm training data to cause a classifier to classify the data incorrectly (poisoning attack), create unfavorable samples during testing to evade detection (evasion attack), and obtain sensitive training data via a learning model (privacy attack)

(v) Deep fake is another big challenge that is being faced by spam detection systems. To generate, modify, and style pictures and videos, neural network models such as GPT-2,3 and image generation models like BigGAN, StyleGAN, and CycleGAN are adopted. Deep fakes can be used to disseminate false information.

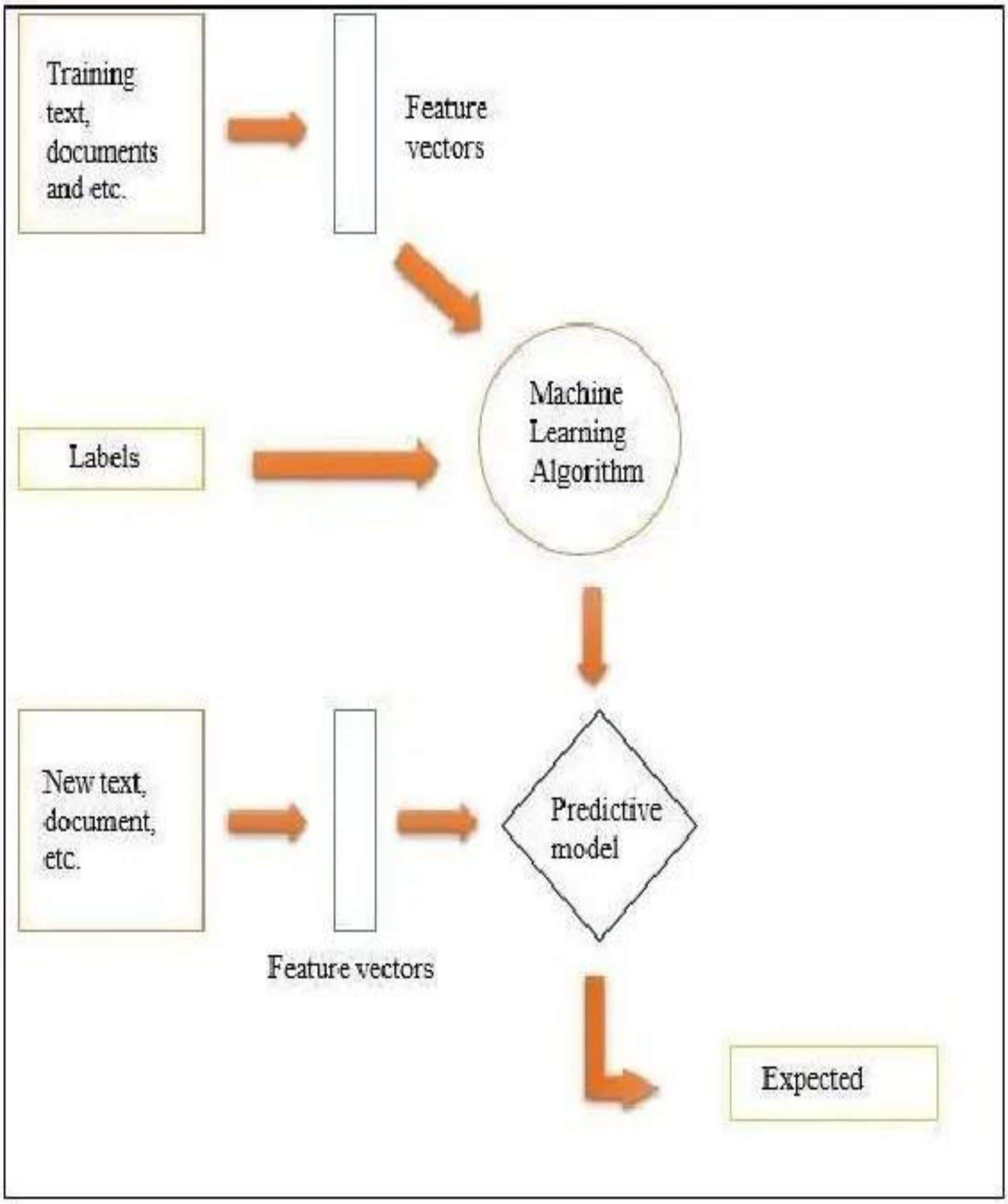


Fig: Spam Detection Flow

METHODOLOGY

4.1 INTRODUCTION

This chapter will explain the specific details on the methodology being used in order to develop this project. Methodology is an important role as a guide for this project to make sure it is in the right path and working as well as plan. There is different type of methodology used in order to do spam detection and filtering. So, it is important to choose the right and suitable methodology thus it is necessary to understand the application functionality itself.

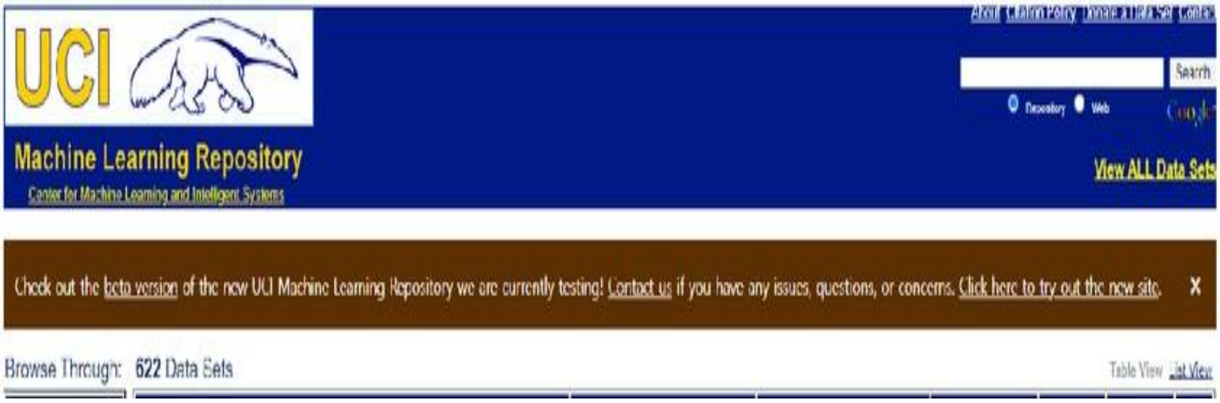
4.2 FRAME WORK

4.2.1 DATA SOURCE

Collecting data is utterly difficult due to numerous constraints for instances the volume of data and the throughput required for proper and timely ingestion.

The dataset that I’ve used in this project is the real existing data that ca n be downloaded from machine learning data repository site.

There is one website that I’ve visit to get the dataset to be used in this project.


UCI
Machine Learning Repository
Center for Machine Learning and Intelligent Systems
About Citation Policy Donate a Data Set Contact
Search
Repository Web
Google
View ALL Data Sets
X
Table View List View

Check out the beta version of the new UCI Machine Learning Repository we are currently testing! Contact us if you have any issues, questions, or concerns. Click here to try out the new site.

Browse Through: 622 Data Sets

Fig: **Data Set**

4.2.2 DATA SETS

A list of data set provided by each website. These dataset might contain more than 1000 labelled messages for training and testing. The data first need to be reformatted into .CSV by splitting them into training.csv and testing.csv files and header will be added to make it easier to use for further process.

"spamHad your mobile 11 months or more? U R entitled to Update to the latest colour mobiles with camera for Free! Call The Mobile Update Co FREE on 08002986030"

"ham I'm gonna be home soon and i don't want to talk about this stuff anymore tonight" k? I've cried enough today.

"spamURGENT! You have won a 1 week FREE membership in our Â£100" 000 Prize Jackpot! Txt the word: CLAIM to No: 81010 T&C www.dbuk.net LCCLTD POBOX 4403LDNW1A7RW18

"ham I've been searching for the right words to thank you for this breather. I promise i wont take your help for granted and will fulfil my promise. You have been wonderful and a blessing at all times."

"ham I HAVE A DATE ON SUNDAY WITH WILL!!"

"ham Oh k...i'm watching here:)"

"ham Eh u remember how 2 spell his name... Yes i did. He v naughty make until i v wet."

"ham Is that seriously how you spell his name?"

"ham Iâ€˜m going to try for 2 months ha ha only joking" "ham Aight yo" dats straight dogg

"ham You please give us connection today itself before &lt;DECIMAL&gt; or refund the bill"

"ham Both :) i shoot big loads so get ready!"

"ham What's up bruv" hope you had a great break. Do have a rewarding semester.

"ham Home so we can always chat"

"ham K:)k:)good:)study well."

"ham Yup... How Ã¼ noe leh..."

"ham Sounds great! Are you home now?"

"ham Finally the match heading towards draw as your prediction."

"ham Tired. I haven't slept well the past few nights."

"ham Easy ah?sen got selected means its good.."

"ham Would really appreciate if you call me. Just need someone to talk to."

"ham Hey company elama po mudyadhu."

"ham Dear good morning now only i am up" "ham Get down in gandhipuram and walk to cross cut road. Right side &lt;#&gt; street road and turn at first right."

"ham Dear we are going to our rubber place"

"ham Sorry battery died" yeah I'm here

"ham Yes:)here tv is always available in work place.."

"ham I have printed it oh. So &lt;#&gt; come upstairs"

"ham Or ill be a little closer like at the bus stop on the same street"

"ham Where are you?when wil you reach here?"

"ham Tomarrow final hearing on my laptop case so i cant."

"ham PLEASSSSSSSEEEEEE TEL ME V AVENT DONE SPORTSx"

"ham Okay. No no"   just shining on. That was meant to be signing but that sounds better.

"ham U don't remember that old commercial?"

"ham Too late. I said i have the website. I didn't i have or dont have the slippers"

"ham I asked you to call him now ok"

"ham Kallis wont bat in 2nd innings."

"ham Usf I guess" might as well take 1 car "ham No objection. My bf not coming."

"ham Thanx..."

"ham Tell rob to mack his gf in the theater"

"ham Awesome" I'll see you in a bit

"ham Just sent it. So what type of food do you like?"

"ham All done? All handed in? Celebrations in full swing yet?"

"ham You got called a tool?"

"ham Ok. I asked for money how far"

"ham Okie..."

"ham Yeah I think my usual guy's still passed out from last night"
"ham K" I might come by tonight then if my class lets out early

"ham Ok.."

"ham Hey you told your name to gautham ah?"

"ham Haf u found him? I feel so stupid da v cam was working."

"ham Oops. 4 got that bit."

"ham Are you this much buzy"

"ham I accidentally deleted the message. Resend please."

"ham So Ã¼ pay first lar... Then when is da stock comin..."

"ham Aft i finish my lunch then i go str down lor. Ard 3 smth lor. U finish ur lunch already?"

"ham Fffffffff. Alright no way I can meet up with you sooner?"

"ham Just forced myself to eat a slice. I'm really not hungry tho. This sucks. Mark is getting worried. He knows I'm sick when I turn down pizza. Lol"

"ham Lol your always so convincing."

"ham Did you catch the bus ? Are you frying an egg ? Did you make a tea? Are you eating your mom's left over dinner ? Do you feel my Love ?"

"ham I'm back &amp; we're packing the car now" I'll let you know if there's room

"ham Ahhh. Work. I vaguely remember that! What does it feel like? Lol"

"ham Yeah he got in at 2 and was v apologetic. n had fallen out and she was actin like spoilt child and he got caught up in that. Till 2! But we won't go there! Not doing too badly cheers. You? "

"ham K tell me anything about you."

"ham For fear of fainting with the of all that housework you just did? Quick have a cuppa"

"spamThanks for your subscription to Ringtone UK your mobile will be charged Â£5/month Please confirm by replying YES or NO. If you reply NO you will not be charged"

"ham Yup... Ok i go home look at the timings then i msg Ã¼ again... Xuhui going to learn on 2nd may too but her lesson is at 8am"

"ham Oops" I'll let you know when my roommate's done

"ham I see the letter B on my car"

"ham Anything lor... U decide..."

"ham Hello! How's you and how did saturday go? I was just texting to see if you'd decided to do anything tomo. Not that i'm trying to invite myself or anything!"

"ham Pls go ahead with watts. I just wanted to be sure. Do have a great weekend. Abiola"

"ham Did I forget to tell you ? I want you " I need you I crave you ... But most of all ... I love you my sweet Arabian steed ... Mmmmmm ... Yummy

"spam07732584351 - Rodger Burns - MSG = We tried to call you re your reply to our sms for a free nokia mobile + free camcorder. Please call now 08000930705 for delivery tomorrow"

"ham WHO ARE YOU SEEING?"

"ham Great! I hope you like your man well endowed. I am &lt;#&gt; inches..."

"ham No calls..messages..missed calls"

"ham Didn't you get hep b immunisation in nigeria."

ham Home so we can always chat
ham K:)k:)good:)study well.
"ham Watching tv lor..."
"ham Thank you baby! I cant wait to taste the real thing..."

"ham You should change your fb to jaykwon thuglyfe falconerf"

"ham If we win its really no 1 side for long time."

"ham Dear reached railway. What happen to you"

"ham Sorry" in meeting I'll call later
"ham What class of &lt;#&gt; reunion?"
"ham Are you free now?can i call now?"
"ham Got meh... When?"
"ham Nope... Think i will go for it on monday... Sorry i replied so late"

"ham Some of them told accenture is not confirm. Is it true."

"ham Kate jackson rec center before 7ish" right?

"ham Dear i have reache room"

"ham When can Ã¼ come out?"

"ham Check with nuerologist."

"ham Lolnice. I went from a fish to ..water.?"

"ham En chikku nange bakra msg kalstiya..then had tea/coffee?"

"ham Carlos'll be here in a minute if you still need to buy"

"ham This pay is &lt;DECIMAL&gt; lakhs:)"

"ham Have a good evening! Ttyl"

"ham Did u receive my msg?"

4.4.3 PROCESS MODEL

Process model is a series of steps, concise description and decisions involved in order to complete the project implementation. In order to finish the project within the time given, the flows of project need to be followed. The framework below shows how the overall flow of this project in order to separate between a spam and ham message.

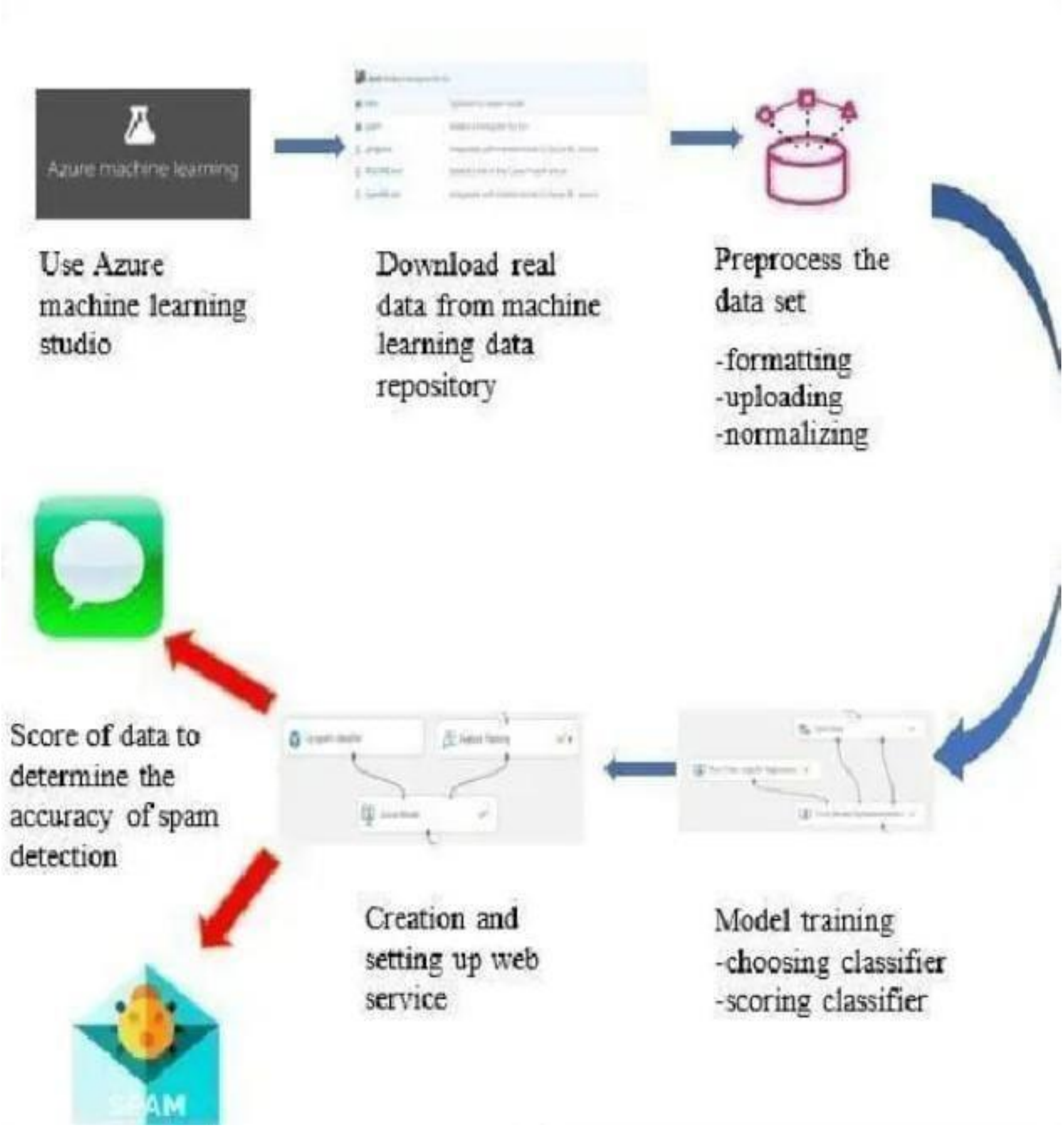

Azure machine learning
Use Azure machine learning studio
Download real data from machine learning data repository
Preprocess the data set
-formatting
-uploading
-normalizing
Model training
-choosing classifier
-scoring classifier
Creation and setting up web service
Score of data to determine the accuracy of spam detection
SPAM

Fig: Process framework

4.4.4 DATA MODEL

As for data model, it refers to the documenting a complex system and data flow between different data elements and design as an easily understood diagra musing text and symbol. The data flow below shows how the data flow of these project in order to detect the spam messages and classify them into two separate type which is spam and ham message.

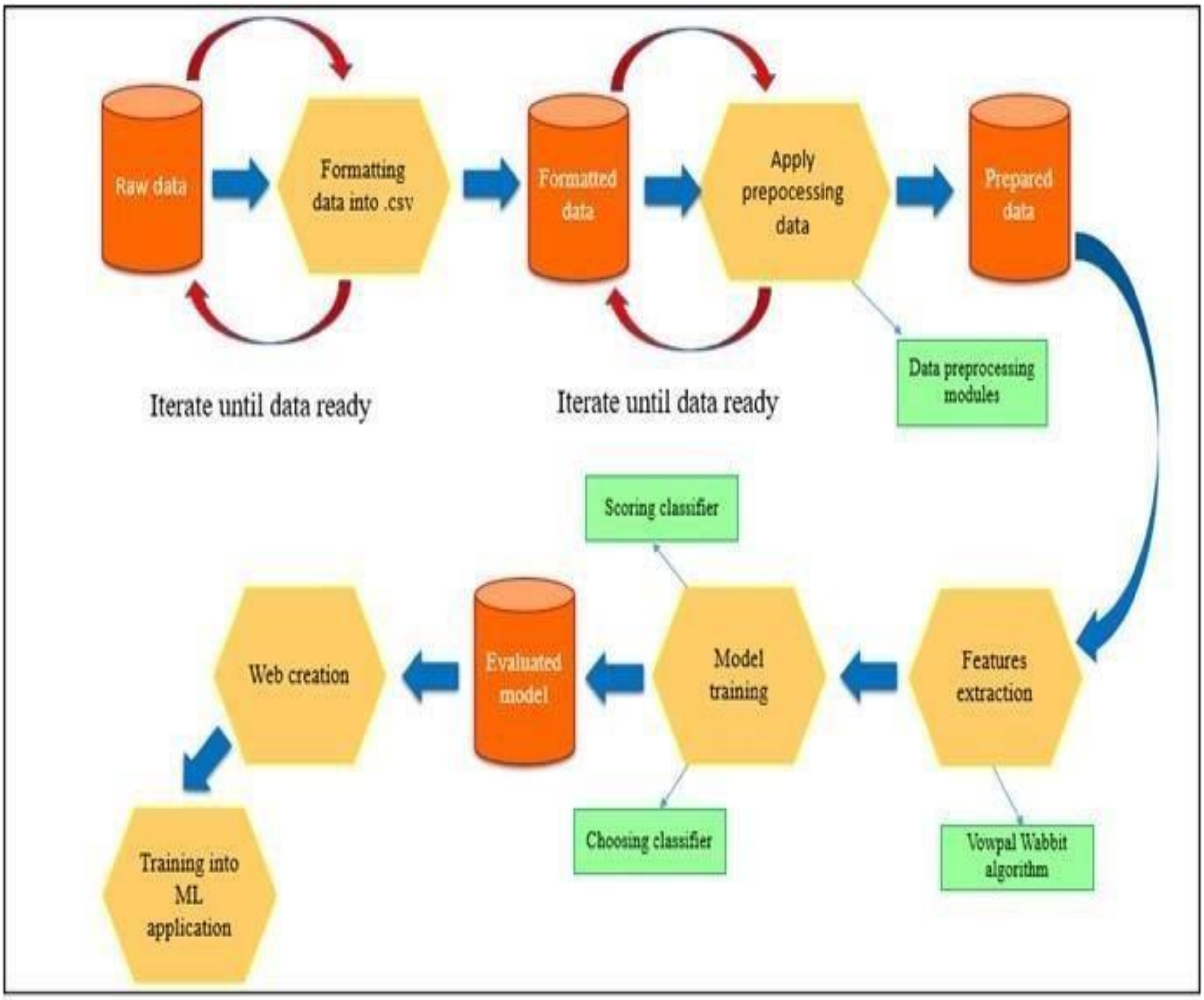


Fig: Data Model Flow

IMPLEMENTATION AND RESULT

5.1 INTRODUCTION

In this chapter, it will discuss about the project implementation. Implementation is the stage of the project development. Implementation is necessary to verify that the project development of the trained model meet the requirement.

So, this chapter will generally discuss the implementation, deployment of the entire project after being developed.

5.2 IMPLEMENTATION

All the implementation process of the spam detection by using machine learning based binary classifier project will be presented.

5.3 SPAM CLASSIFIER

```
import pandas as pd
import numpy as np
import pickle
import re
import nltk
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline

from sklearn.metrics import accuracy_score,fbeta_score,classification_report
from wordcloud import WordCloud
from nltk.tokenize import word_tokenize

from nltk.corpus import stopwords
nltk.download('stopwords')
stop=stopwords.words("english")

from nltk.stem.porter import PorterStemmer
from nltk.stem import SnowballStemmer
ss = SnowballStemmer("english")
ps = PorterStemmer()
```

```
msg_df = pd.read_csv('spam.csv', sep='\t', names=["label", "message"])
msg_df.shape
```

In[9]:

```
msg_df = pd.read_csv('spam.csv', sep='\t', names=["label", "message"])
msg_df.shape
msg_df.head(5)
```

Out[9]:

```
        label       message

0       ham         Go until jurong point, crazy.. Available only ...
1       ham         Ok lar... Joking wif u oni...
2       spam        Free entry in 2 a wkly comp to win FA Cup fina..
3       ham         U dun say so early hor... U c already then say...
4       ham         Nah I don't think he goes to usf, he lives aro...
```

In[10]:

```
msg_df.describe()
```

Out[10]:

```
          Label      message
Count     5572       5572
Unique    2          5169
Top       ham        Sorry, I'll call later
Freq      4825       30
```

In[11]:

```
msg_df.groupby('label').describe().T
```

Out[11]:

Label	ham	spam
Count	4825	747
Unique	4516	653
Top	Sorry, I'll call later	Please call our customer service..
Freq	30	4

In[12]:

```
msg_df["label"].value_counts()
```

Out[12]:

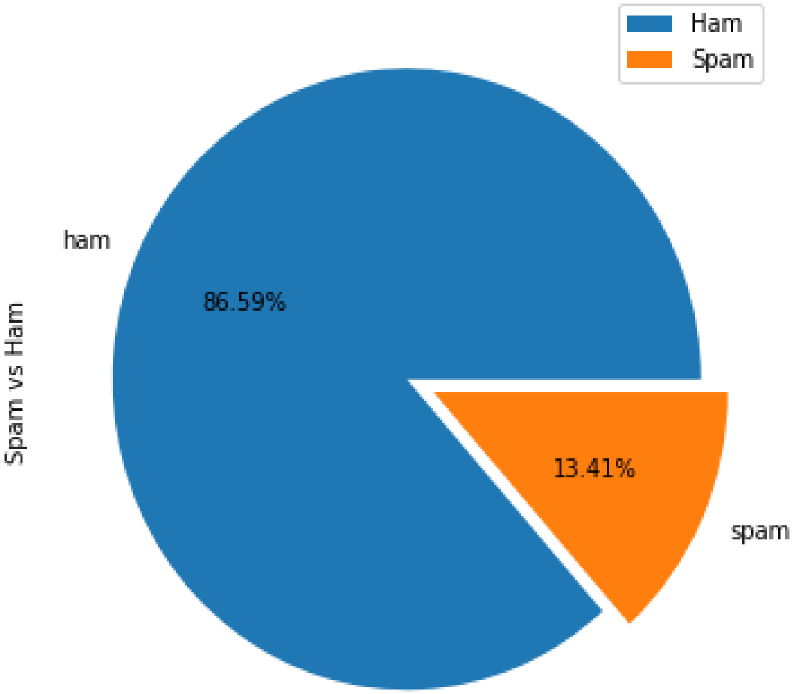
```
ham:     4825
spam:    747
Name:   label, dtype: int64
```

In[13]:

```
msg_df["label"].value_counts().plot(kind = 'pie', explode = [0,     0.1],
figsize = (6, 6), autopct = '%1.2f%%')
plt.ylabel("Spam vs Ham")
plt.legend(["Ham", "Spam"])
plt.show()
```

Out[13]:

```
plt.ylabel("Spam vs Ham")
plt.legend(["Ham", "Spam"])
plt.show()
```



Html Code:

Home.html

```
<!DOCTYPE html>
<html>
<head>
	<title>Home</title>
	<link rel="stylesheet" type="text/css"  href="style(sp).css">
</head>
<body>

	<header>
		<div class="container">
		<div id="brandname">
			Spam Detector App using NLP
```

```
        </div>
        <h2>Spam Detector For Email Messages</h2>

    </div>
    </header>

    <div class="ml-container">

        <form action="/predict" method="POST">
        <p>Enter Your Message/Mail Here which you want to predict Spam or Not spam (Ham)</p>
        <textarea name="message" rows="6" cols="50" placeholder="Enter Your Message..." ></textarea>
        <br/>

        <input type="Submit" class="btn-info" value="Predict">

    </form>

    </div>
</body>

</html>
```

Result.html

```
<!DOCTYPE html>
<html>
<head>
```

```
        <title></title>
   <link rel="stylesheet" type="text/css"  href="style(sp).css">
</head>
<body>

        <header>
                <div class="container">
                <div id="brandname">
                        Spam Detector App using NLP
                </div>
                <h2>Spam Detector For Email Messages</h2>

        </div>
        </header>

        <p style="color:white;font-size:20;text-align: center;"><b>Results for Message</b></p>
        <div class="results">

        {% if prediction == 1%}
        <h1 style="color:red ;text-align: center;">Spam &#128721; </h1>
        {% elif prediction == 0%}
        <h1 style="color:green;text-align: center;" >Not a Spam  &#128140; </h1>
        {% endif %}
        </div>

</body>
```

</html>

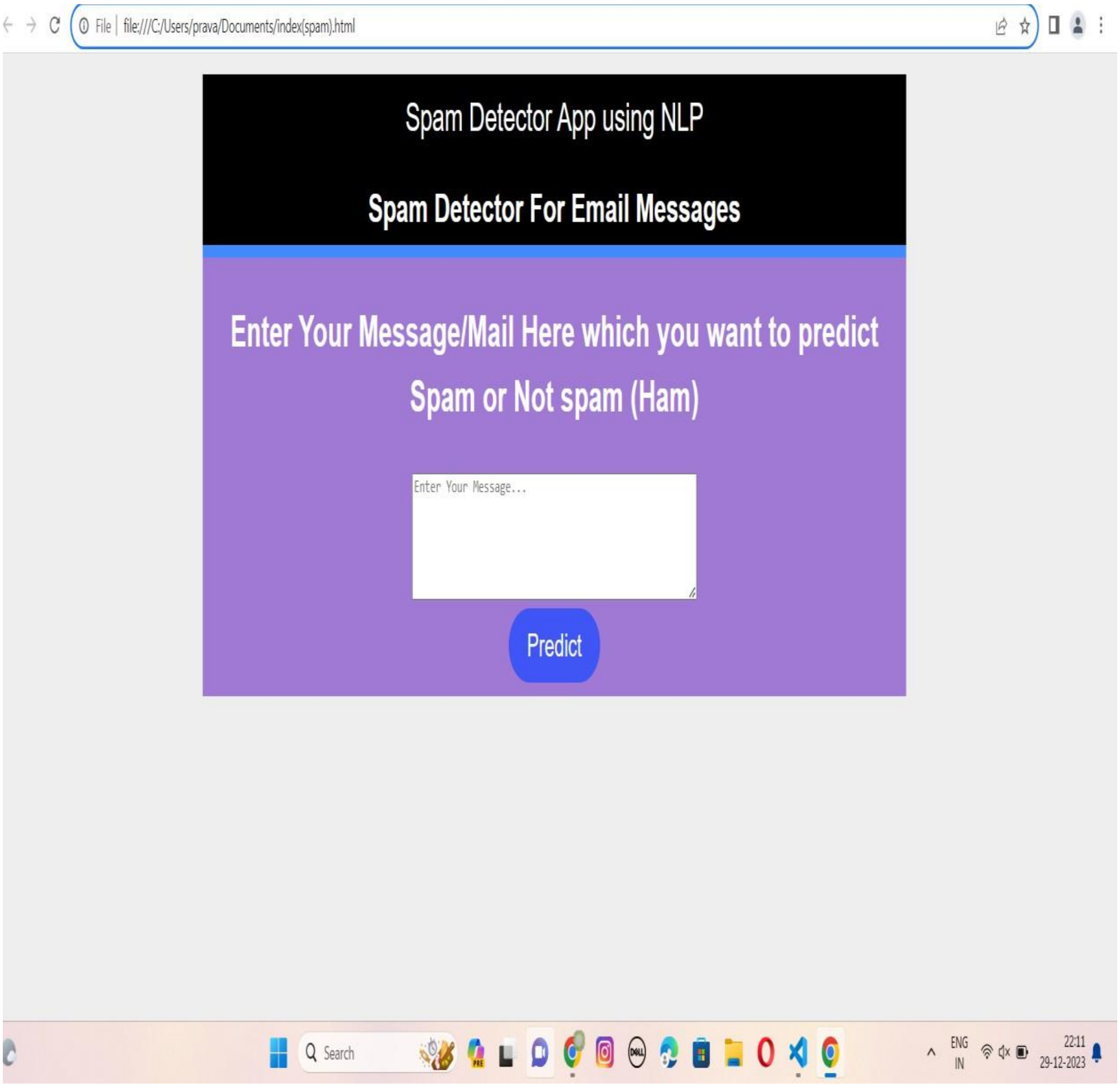
Css Code:

Style.css

```
html { font-family: sans-serif; background: #eee; padding: 1rem; }

body{ max-width: 960px; font:15px/1.5 Arial, Helvetica,sans-serif;  padding: 0px;  margin: 0 auto; background-color: #a079d4; }


.container{ width:100%; margin: auto; overflow: hidden;}


header{ background:#000000; border-bottom:#448AFF 10px solid;height:120px; width:100%;    padding-top:10px;}


.main-header{ text-align:center;background-color: blue;    height:100px;width:100%;margin:0px;    }

form { text-align:center;  }

#brandname{ font-size:30px;   text-align:center;   color: #fff;}

p{font-size:33px;  text-align:center; color: white; font-weight:bolder}

header h2{   text-align:center;   font-size:30px;     color:#fff;}

.btn-info {padding: 15px 25px; font-size: 24px; cursor: pointer; text-align: center; outline: none; color: #fff; background-color: #4055f5; border: none; border-radius: 40px;margin-bottom: 10px;}

.btn-info:hover {background: #3045e6; }

p.copyright { position: absolute;font-weight: bold;  color: #fff; width: 100%;line-height: 40px; bottom:0;text-align: center;  font-size: 0.7em;}

footer.footer {position: relative; height:60px; width: 100%;background-color: #333333;}

.resultss{     border-radius: 15px 50px;  background: #345fe4;  padding: 20px; width: 200px; height: 150px; }
```

SCREENS AND REPORTS

- Enter the string which you want to predict whether it is a spam or ham (Not Spam)

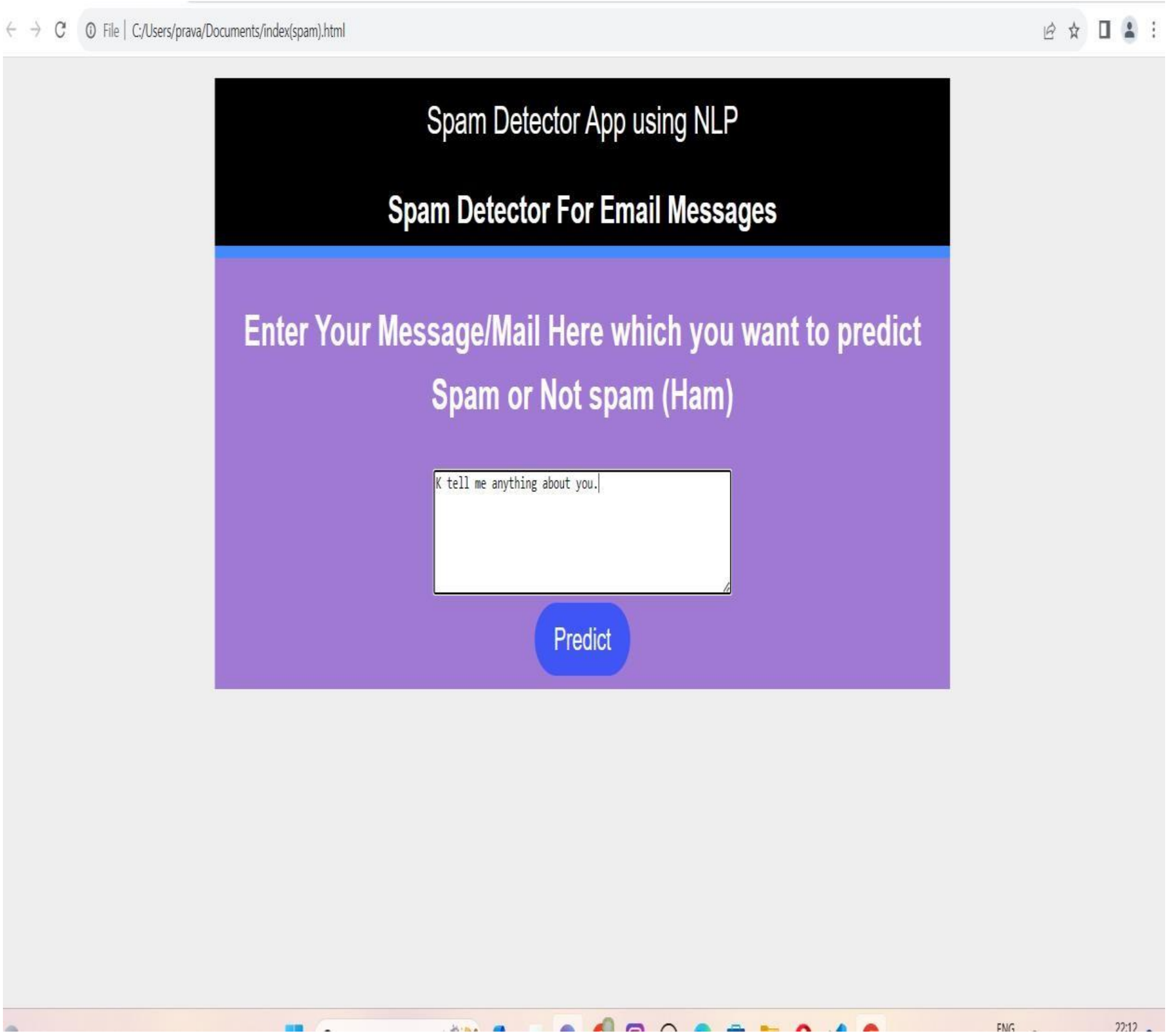


Spam Detector App using NLP

Spam Detector For Email Messages

Enter Your Message/Mail Here which you want to predict Spam or Not spam (Ham)

- now at prediction template enter the string which you want to predict whether it is a spam or ham and click on predict as shown below:



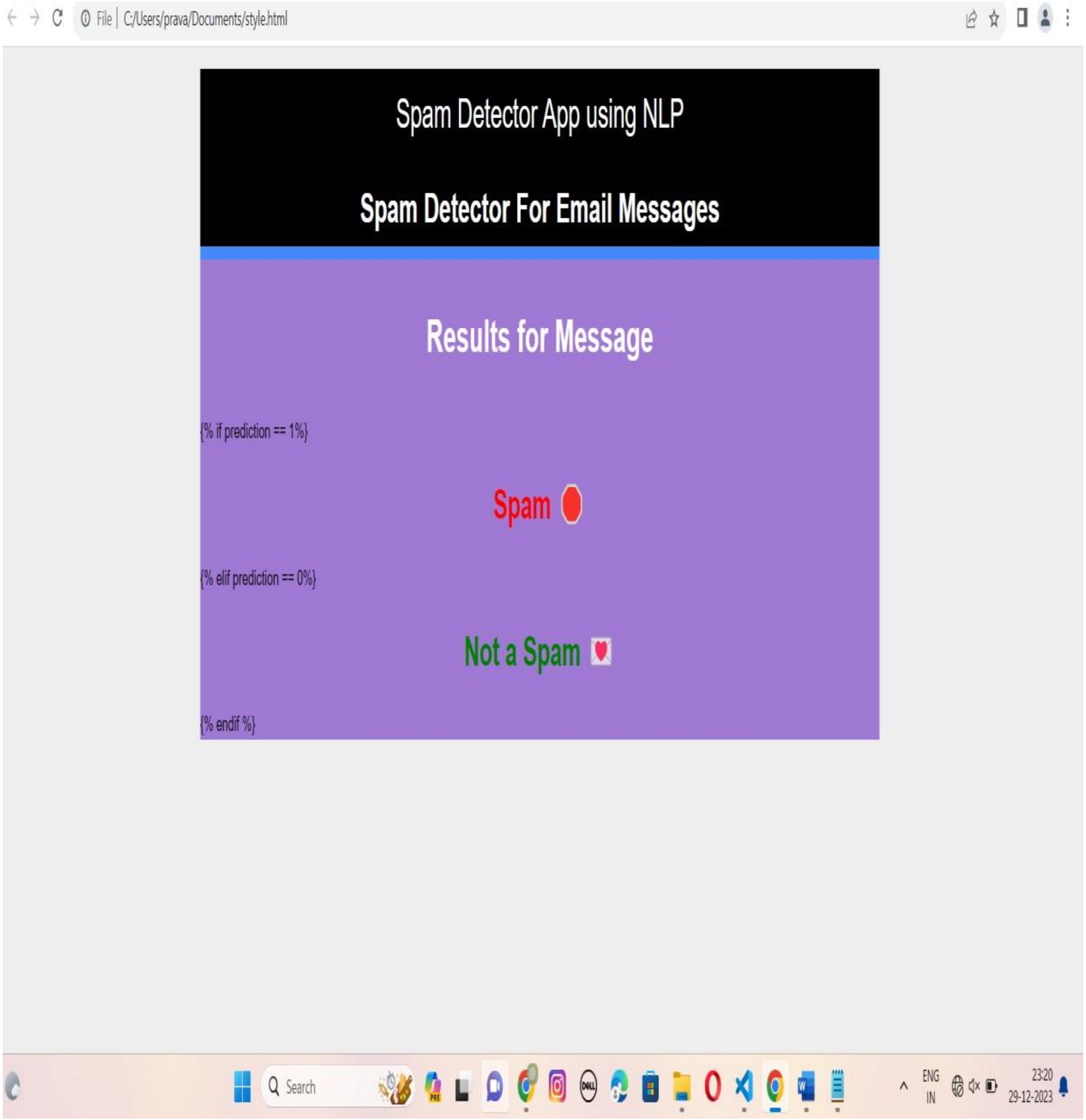
Spam Detector App using NLP

Spam Detector For Email Messages

Enter Your Message/Mail Here which you want to predict Spam or Not spam (Ham)

K tell me anything about you.

- After the message is predicted, it will shown if the message is SPAM or NOT SPAM (ham).



Spam Detector App using NLP

Spam Detector For Email Messages

Results for Message

{% if prediction == 1%}

Spam 🛑

{% elif prediction == 0%}

Not a Spam 💌

{% endif %}

TESTING

In this section, it will discuss about the testing that has been made in order to detect spam, classify data and list down all the important elements such as correct and wrong words.

TYPES OF TESTING

a) Test with word model

   By using word model, classification type will be determined by labelling the messages into two types which is spam and ham. The threshold of the messages will be calculated and then it will be compared with the cutover to see either It falls below or above the cutover value. If it falls below then it will be classified as spam, and if it falls above then it is ham.

b) Test with Azure model

   By using Azure model, we will test the detection of classification type to see whether it works as needs.

   Testing a spam email detector using machine learning involves several key steps:

   1. Data Collection:
      - Gather a diverse dataset of emails, including both spam and non-spam (ham) samples.
      - Ensure the dataset is representative of the types of emails the model will encounter.

   2. Data Preprocessing:
      - Clean and preprocess the data, including tasks such as removing duplicates, handling missing values, and standardizing text (lowercasing, stemming, etc.).

   3. Feature Engineering:
      - Extract relevant features from the email data, such as word frequencies, presence of specific keywords, or characteristics of the sender's address.

   4. Split Data:
      - Divide the dataset into training, validation, and test sets to assess the model's performance on unseen data.

   5. Model Selection:

- Choose a suitable machine learning algorithm for the task, such as Naive Bayes, Support Vector Machines, or deep learning models like neural networks.

6. Training:
   - Train the selected model on the training dataset, adjusting hyperparameters as needed.

7. Validation:
   - Evaluate the model on the validation set to fine-tune parameters and prevent overfitting.

8. Testing:
   - Assess the model's performance on the test set, using metrics like precision, recall, and F1 score.

9. Performance Metrics:
   - Measure the model's effectiveness using metrics relevant to the task, considering factors like false positives and false negatives.

10. Tuning:
    - If needed, refine the model based on the test results, adjusting features, algorithms, or hyperparameters.

11. Cross-Validation:
    - Perform cross-validation to ensure the model's robustness by training and testing on different subsets of the data.

12. Evaluation on Real Data:
    - Test the spam email detector on real-world data to assess its practical performance.

13. Monitoring and Updating:
    - Continuously monitor the model's performance and update it as needed to adapt to changing spam patterns.

14. Documentation:
    - Document the entire testing process, including data sources, preprocessing steps, model selection, and performance metrics.

By following these steps, you can systematically test and refine your spam email detector to improve its accuracy and reliability.

We Tested the data sets and found out which email is Spam and which email is Ham.

CONCLUSION

In the last two decades, spam detection and filtration gained the attention of a sizeable research community. The reason for a lot of research in this area is its costly and massive effect in many situations like consumer behaviour and fake reviews. The survey covers various machine learning techniques and models that the various researchers have proposed to detect and filter spam in emails and IoT platforms.

The study categorized them as supervised, unsupervised, reinforcement learning, etc. The study compares these approaches and provides a summary of learned lessons from each category. This study concludes that most of the proposed email and IoT spam detection methods are based on supervised machine learning techniques.

A labelled dataset for the supervised model training is a crucial and time-consuming task. Supervised learning algorithms SVM and Naïve Bayes outperform other models in spam detection. The study provides comprehensive insights of these algorithms and some future research directions for email spam detection and filtering.

REFERENCES

1) https://techvidvan.com/tutorials/spam-detection-using-svm/
2) https://www.hindawi.com/journals/scn/2022/1862888/
3) https://www.researchgate.net/publication/342113653_Email_based_Spam_Detection
4) https://ieeexplore.ieee.org/abstract/document/9183098
5) https://github.com/Govind155/Spam-Classification-using-NLP/tree/main

6) Le, H. V., Nguyen, M. T., & Nguyen, T. T. (2018). Email spam detection based on ensemble learning of extreme learning machine. International Journal of Machine Learning and Cybernetics, 9(4), 591-602.