Email/SMS Spam Classifier

# **Abstract**

Spam email has become a major problem on the Internet in recent years. Spam, also known to as unsolicited commercial or commercial email, is the bane of email communication. Growing spam levels are causing a number of major problems. Spam is a waste of time dealing with and a waste of storage space and transmission bandwidth. Another common definition of spam limits it to promotional emails and excludes, even if unsolicited, non-commercial solicitations like political or religious pitches. Automatic email filtering appears to be the most successful approach for preventing spam at the present, and spammers and spam filtering techniques are in fierce competition. The more sophisticated the anti-spam techniques become, the smarter the spammers' tricks become. Overall, email spam detection classifiers are a useful tool for helping to protect users from the nuisance and potential harm of spam emails. Due to the massive and unique number of available features in the dataset as well as the large number of documents, text classification, particularly email classification, presents difficulties.[1] Applicability in these datasets with existing classification techniques was limited because the large number of features make most documents undistinguishable. Only a small percentage of all features in many datasets of different documents, may be effective for classifying documents, and utilizing all features might have a negative effect on performance. The accuracy of feature selection algorithms as well as text classification algorithms relies on the training dataset's quality. All of the key constructs and their possible distribution in the category will be part of the ideal training document dataset for each category.

# **Introduction**

## **Background**

A spam detection classifier is created to rid an inbox of undesirable and hazardous emails, safeguarding the user from any possible phishing attacks or malicious software. Such spam emails can be a nuisance and occupy someone's time with the need to delete them, and can also congest an email server, making it more difficult for the user to send and receive legitimate emails. By recognizing and obstructing spam emails, a spam detection classifier can aid in increasing the overall effectiveness and safety of an email system.[2]

## **Importance**

Recognizing and eliminating email spam is a critical activity because it can cause a range of issues for people and companies. Not only are these emails irritating and occupy valuable time, but they may also contain deceiving phishing attempts that seek to dupe people into revealing private data such as login details or financial information. These hoaxes can be quite convincing and may result in victims becoming a target for identity theft or other cybercrimes.

Aside from the threat of phishing, spam emails can also contain malicious software or viruses that can jeopardize the safety of a computer or acquire confidential data. This can result in severe issues, including the pilfering of sensitive information or the obstruction of business activities.[3]

An email spam detection classifier can be an incredibly useful tool for blocking out unwanted emails and safeguarding against potential risks. This system utilizes machine learning algorithms to review the contents of emails and decide if it is likely to be spam. These classifiers are able to accurately discern spam emails, thereby protecting people and organizations from any time-related losses and potential cyber-attacks.

## **Motivation**

Unsolicited emails, sent in bulk amounts, are known as spam and can be a hazard to both people and companies. These messages may contain phishing schemes, malware, and other cyber risks. Moreover, these emails can be a nuisance to go through, and can choke up email servers. [4]A spam detection classifier is a device utilized for the recognition and removal of spam emails, so that only authentic, important messages are received by users. The purpose of creating a spam detection classifier is to secure individuals and organizations from these dangers, and to enhance the speed and efficiency of email communication.

## **Significance of the problem**

Over the last few years, email has become an increasingly popular form of official communication for many internet users. Nevertheless, the rise in email usage has also led to a proliferation of spam emails. Spam is the act of sending out bulk, unsolicited messages. In contrast, emails with meaningful content are referred to as 'Ham'. In general, an email user receives between 40-50 emails each day. Spammers make an estimated $3.5 million in revenue from spam annually, which causes financial harm to both individuals and organizations. Consequently, users are forced to spend a significant amount of work time dealing with these emails. Data shows that spam emails account for more than 50% of all email server traffic, consisting of large amounts of unwanted and unsolicited emails. This effectively reduces productivity by consuming user resources for no useful purpose. [5]

Spam emails are often utilized by spammers to promote their products and services, as well as to conduct criminal activities such as identity theft, monetary disruption, stealing confidential data, and tarnishing one's reputation. As a result, it is essential for businesses to properly manage and classify their emails to improve efficiency and avoid potential financial losses.

# **Research Methodology**

## **Dataset**

This is a collection of 5574 spam and legitimate English text messages gathered from the Kaggle, the corpus is hosted at the UCI Machine learning repository and also available in raw format publicly which are gathered from following free research sources:

* This dataset has been constructed from a variety of freely accessible or research-related sources from the internet. It includes 425 SMS spam messages taken from the Grumbletext website, a UK forum where users can talk about any spam messages they have received.
* Furthermore, 3,375 SMS ham messages have been randomly selected from the NUS SMS Corpus, a compilation of 10,000 legitimate messages from the National University of Singapore's Department of Computer Science.
* 450 SMS ham messages come from Caroline Tag's PhD Thesis.
* Lastly, we've added the SMS Spam Corpus v.0.1 Big, which contains 1,002 ham messages and 322 spam messages, and is available at the provided link.

## **Methodology**

### **Data Preprocessing**

When dealing with text, it is always beneficial to split the text into smaller components such as words and letters. These individual words and letters are subsequently transformed into tokens through the tokenization process. Tokenization is the act of breaking text into words and tokens.[6] It is possible to create a function that breaks words by space, converts text to lowercase, removes special characters and stopwords, filters out punctuation, and completes stemming. Stemming is the alteration of a word with various tenses into one common form (i.e. dance, dancing, danced = dance). Sklearn's Label Encoder was used to turn ham and spam into the values 0 and 1 respectively.

### **Tokenization/Vectorization**

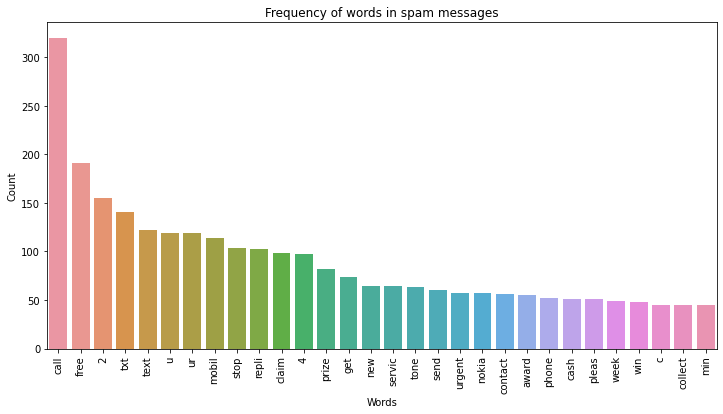
To meet the input data needs of various classifiers, various preprocessing techniques were implemented. Our approach included Term Frequency - Inverse Document Frequency (tf-idf). To calculate Term Frequency, the occurrence of a word in a document is noted, while the frequency of the word in an entire corpus of documents is known as Inverse Document Frequency. The importance of words is reflected in their tf-idf weight - words employed commonly are assigned lower weight and words used infrequently get higher weight. We initiated the process by eliminating stop words, capital letters, non-alphanumeric characters and any unneeded punctuation. After that, similar words were brought together (e.g. desks to desk)[7]. Then we used sklearn's TfidfVectorizer to convert the cleared text to tfidf features (5000 features for one entry) and create a bag of words, which is referred to as a count vector, followed by the tf-idf matrix.

Figure 1 Graph of the frequency words used in the spam messages

### **Training and Test Datasets**

The dataset was split into two parts such that 80% of the data was used in the training phase while 20% of the data was used in testing phase.

## **Models used**

Following are the classifiers used for the spam detection:

### **Naive Bayes classifier**

Naïve Bayes classifiers can be quickly and simply implemented and don't require a large amount of data to be trained as they are founded on the concept that the inclusion of any given attribute in a message is independent of the existence of any other feature.

### **Support vector machine (SVM)**

Support Vector Machines (SVMs) are an example of a supervised learning model that can be deployed for categorizing objectives. SVMs are able to identify the hyperplane in a high-dimensional region that separates the various classes most effectively.

### **Decision trees**

Decision tree classifiers are a form of model that employs a tree-like structure to generate choices. This type of classifier functions by dividing the feature space into sections, and each section is related to a particular class label.

### **Random Forest**

Random Forest is a term used to refer to a combination of decision trees. The Random Forest classifier is an ensemble learning method that brings together a collection of decision trees. To classify a new object, all the trees "vote" for the class they think it belongs to. The class which has the most votes is then chosen as the classification label.[8]

### **Bagging Classifier**

A bagging classifier is a technique employed in ensemble learning that serves to enhance the accuracy of a spam detection classifier. This is done by training a multitude of models on distinct, randomly selected subsets of the training data, then combining the results from these models through a majority vote. Bagging assists in curbing overfitting and amplifying the classifier's capacity for generalization. It is regularly utilized in conjunction with decision tree or neural network models, though it can be applied to any type of classifier.

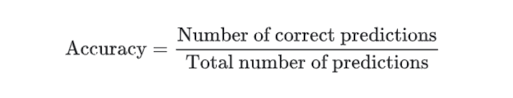
### **KNN**

This algorithm is suitable to utilize when there is noise present in the input data. It can generate both classifications and regression outputs for the designed system. One of the primary disadvantages of the algorithm is its high sensitivity to outliers in the dataset. Additionally, it has a relatively high computational cost in comparison to other machine learning algorithms, which may be why it is not as widely utilized in the reviewed studies.

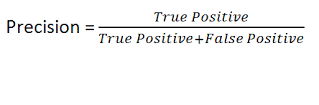
# **Result and Discussion**

For the evaluation of our system, we used two evaluation measures to check the authenticity of the models, used in the detection system, which are:

### **Accuracy**

Accuracy is defined as the ratio of the data correctly classified by the method with respect to the whole data present in the dataset. The accuracy of all the classifiers used for classifying spam messages from the dataset. The formula of accuracy is:

### **Precision**

Precision is defined as the ratio of the actual errors among all the data that were classified as errors. The precision of all the classifiers used to check how close spam messages are being detected from the dataset. The formula of precision is:

It has been noted that text message shorthand and condensed words have been replaced with their full English equivalents. This pre-processing has had the effect of slimming down the feature space, since the majority of abbreviations are made up of fairly common words. While a decrease in features has been observed, there has been an improvement in accuracy or at least a maintenance of the same level [9]. The feature space used for the classification has been diminished by 2-5%. For SMS classification, as with email, the precision rate is more important than the recall rate, since incorrectly identified negative messages are more crucial than false positive ones.

## **Results and Discussion**

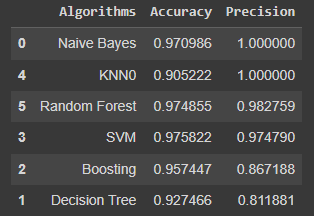
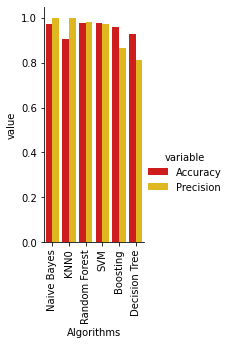
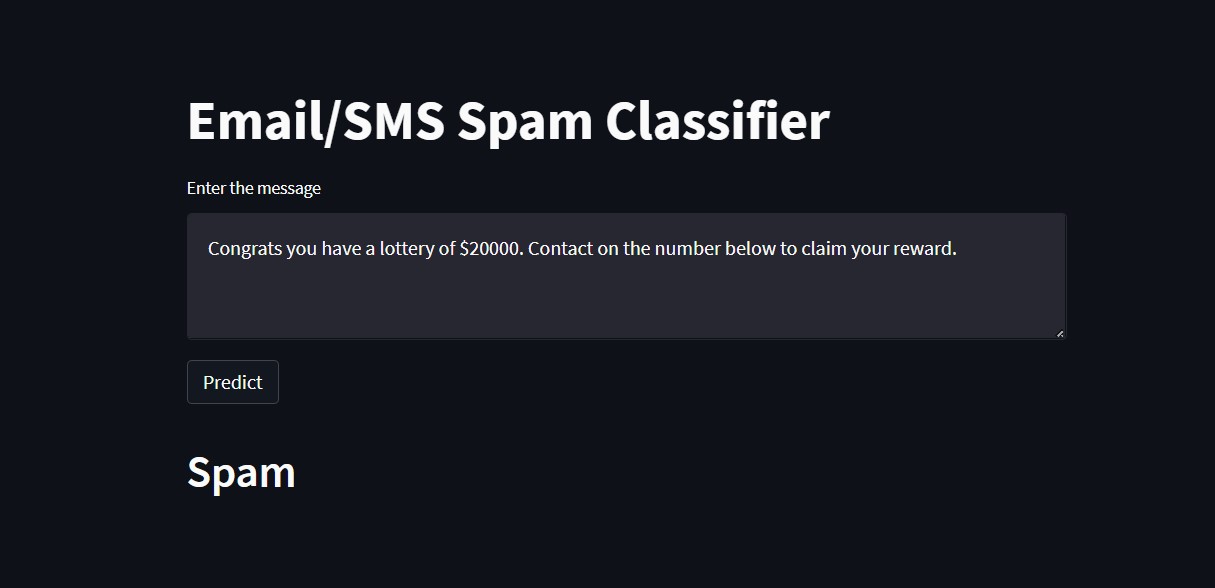
After the evaluations of all the classifiers in the system, a table was generated based on the accuracy and precision of each model, shown as:

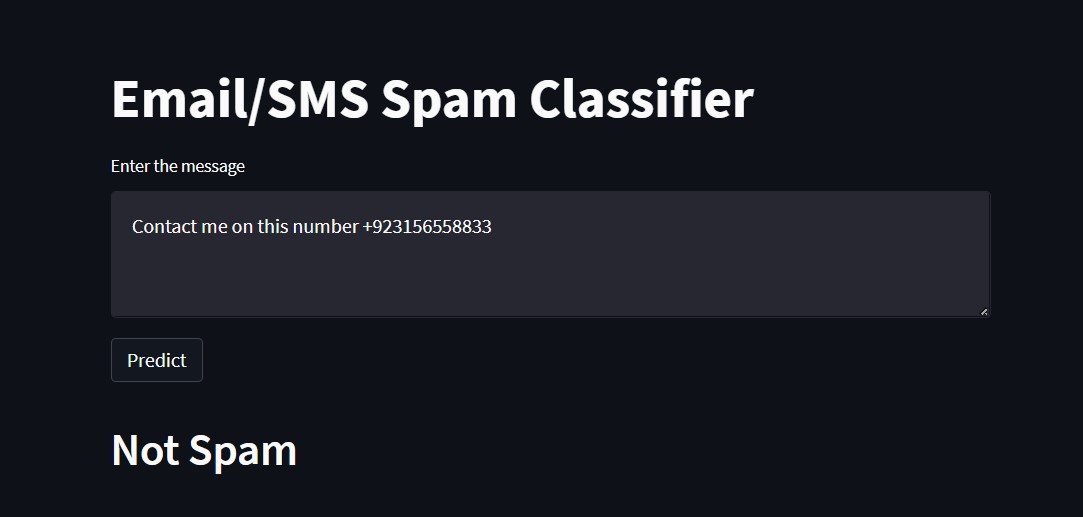
Figure 2 Table of evaluation for different classifiers

As you can see that Multinomial Naïve Bayes outperformed all the model with 100% precision and 97% accuracy. Right behind, KNN achieved 100% precision but 90% accuracy. Random Forest predicted the right output the with the precision of 98.2% and accuracy of 97.4%. Support Vector Machine achieved the precision and accuracy of 97.4% and 97.5% respectively. Meanwhile boosting and Decision Tree classifier performed fairly poor in comparison to other models and achieved 86.7% and 81.1% precision with 95.7% and 92.7% of accuracy respectively.

Here, a graph shows the visualization of each model with the difference in precision and accuracy. Keep in mind that we are preferring precision over accuracy because precision works better for the classification systems and since it is a detection-based classification system. So, the precision of each model is preferred for the main evaluation of the system. Also, the dataset was unbalanced and to overcome this situation, a right evaluation is important to use, which in this case is the precision. Since, Naïve Bayes outperformed all the classifiers[6]. So, we implemented it as the main classifier in our system.

### **Working System Results**

After implementing all the steps discussed above, including determining the probability of words in spam and ham messages, Multinomial Naïve Bayes' Classifier will be used to detect whether the exported message is spam or not. The figures below demonstrate a message that has been detected as spam or not spam.

If the exported message is moved from the inbox to the dataset, then the Naïve Bayes' Classifier will be used to detect it as spam, as can be seen below. Conversely, if the message is transferred from the inbox to the dataset, then the Multinomial Naïve Bayes' Classifier will classify it as not spam, as depicted below.

# **Conclusion and Future Works**

Research on email spam classification has been heavily pursued by the general public, in order to identify any unwanted data or potential risks. As a result, numerous researchers have sought out to attain the best classifier for detecting these spam emails. Naïve Bayes (Multinomial) and SVM have proven to be highly reliable, with a precision of 100%, compared to other machine learning algorithms. These two classifiers were successfully able to differentiate between spam and ham messages in emails and SMSs. Additionally, NLP techniques were implemented to normalize and vectorize text, since SMS messages are usually short and contain a variety of different elements, such as idioms, slang, symbols, emoticons, and abbreviations, which can compromise the performance of classifiers. Although the system is still in its early stages, it can be further developed in the future by providing an updated dataset with more data, as spammers are continuously attempting to outsmart the existing spam algorithm and gain access to user's inbox.

# **Reference**

[1] S. Gadde, A. Lakshmanarao, and S. Satyanarayana, “SMS Spam Detection using Machine Learning and Deep Learning Techniques,” in *2021 7th International Conference on Advanced Computing and Communication Systems, ICACCS 2021*, Mar. 2021, pp. 358–362. doi: 10.1109/ICACCS51430.2021.9441783.

[2] D. Fernandes, K. A. P. da Costa, T. A. Almeida, and J. P. Papa, “SMS spam filtering through optimum-path forest-based classifiers,” in *Proceedings - 2015 IEEE 14th International Conference on Machine Learning and Applications, ICMLA 2015*, Mar. 2016, pp. 133–137. doi: 10.1109/ICMLA.2015.71.

[3] T. Sultana, K. A. Sapnaz, F. Sana, and M. J. Najath, “Email based Spam Detection.” [Online]. Available: www.ijert.org

[4] *2019 Third International conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC).* IEEE.

[5] A. Alzahrani and D. B. Rawat, “Comparative Study of Machine Learning Algorithms for SMS Spam Detection,” 2019.

[6] S. I. Ao and International Association of Engineers., *International MultiConference of Engineers and Computer Scientists : IMECS 2012 : 14-16 March, 2012, the Royal Garden Hotel, Kowloon, Hong Kong*. Newswood Ltd., 2012.

[7] M. Raza, N. D. Jayasinghe, and M. M. A. Muslam, “A Comprehensive Review on Email Spam Classification using Machine Learning Algorithms,” in *International Conference on Information Networking*, Jan. 2021, vol. 2021-January, pp. 327–332. doi: 10.1109/ICOIN50884.2021.9334020.

[8] S. Aluru, Jaypee Institute of Information Technology University, University of Florida. College of Engineering, IEEE Computer Society, IEEE Computer Society. Technical Committee on Parallel Processing, and Institute of Electrical and Electronics Engineers, *2018 Eleventh International Conference on Contemporary Computing (IC3) : 2-4 August 2018, Jaypee Institute of Information Technology, Noida, India*.

[9] *2009 Canadian Conference on Electrical and Computer Engineering : “from electrons to oceans, innovation at many scales” : May 3-6, 2009, Delta St. John’s Hotel and Conference Centre, St. John’s, Newfoundland & Labrador, Canada.* IEEE, 2009.