

**Project Report**

**Project Title: Spam Detection**

**Course Title: Machine Learning**

**Course Code: CSE475**

**Section: 02**

**Semester: SUMMER’21**

**Group ID: EWUSM2021CSE47511**

**SUBMITTED BY:**

* **Sumaita Tanjim Hridy**

**Id: 2018-1-60-251**

* **Nawfat Tahsin Azad**

**Id: 2018-1-60-247**

**1.Introduction:** In this era email and messages are most commonly used communication system. Due to the increasing number of people using email, there has been an increase in the number of spam emails. Aside from being annoying, spam emails can also pose a security threat to computer systems. This spamming is not only annoying people but also spam and phishing activities has made it a target of criminals. There are lots of case where people’s personal information are getting leaked by spam emails. Now in our project we will discuss how machine learning can help in detection of spam emails so that we can get rid of these type of problems.

**1.1Objectives:** There are some objectives for this project. They are:

1. To find spam emails by some clustering techniques.
2. Compare different types of machine learning systems to see the accuracy.
3. Using machine learning techniques to design an application that can filter out spams with high accuracy.
4. Compare real data with the machine learning algorithm data.

**1.2Motivation:** When spam starts to hits your inbox, you're a target. Attackers will try to trick users into clicking on links that they shouldn't, or they will expose their sensitive data if they do. So once personal or official data exposed to the attacker people become victim to their illegal activities as the criminals most of the time blackmail them to do what they want. In these type of situation if there could be a spam filter or spam detector people can prevent these attacks. As the spam detector can be used as a protector of their internal data. Because spam filtering has become more prevalent. It is required by organizations to prevent the exploitation of their users' data.

**1.3Existing Work:** There are many other existing work on spam detection:

* S. M. Lee, D. S. Kim, J. H. Kim and J. S. Park, "Spam Detection Using Feature Selection and Parameters Optimization," *2010 International Conference on Complex, Intelligent and Software Intensive Systems*, 2010, pp. 883-888, doi: 10.1109/CISIS.2010.116.
* Dima Suleiman, Ghazi Al-Naymat,SMS Spam Detection using H2O Framework,Procedia Computer Science,Volume 113,2017,

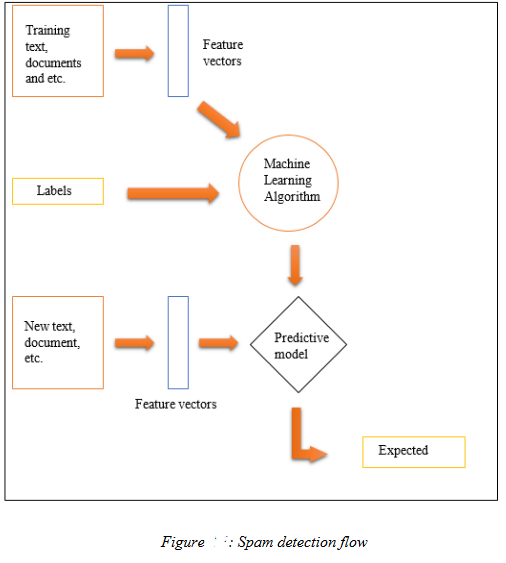
https://doi.org/10.1016/j.procs.2017.08.335.

(<https://www.sciencedirect.com/science/article/pii/S1877050917317453>)

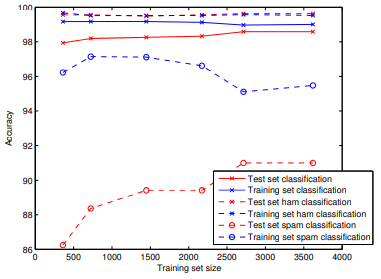
* S. Sharmin and Z. Zaman, "Spam Detection in Social Media Employing Machine Learning Tool for Text Mining," *2017 13th International Conference on Signal-Image Technology & Internet-Based Systems (SITIS)*, 2017, pp. 137-142. doi: 10.1109/SITIS.201732

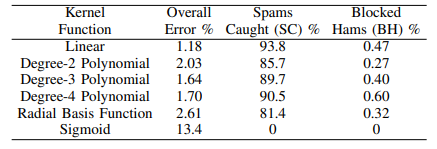
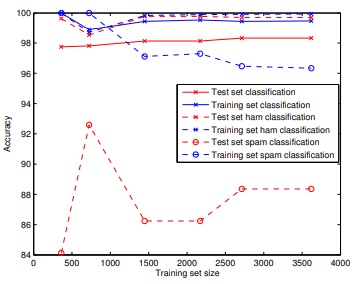
**1.4Necessity:** Spamming is not anymore garbage, it is threat to all the internet users. Because in these spam mails through one click personal and official information can be exposed. All system can be hacked. So spam detection has become the necessity now a days. Because people are now using mail and short messages through internet to communicate and without spam filtering their data in internet is not safe at all.

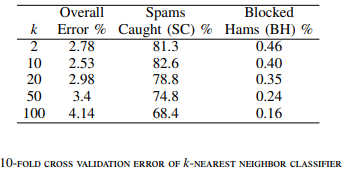
**2.Methodology:** Spam detection can be implemented anywhere, and multiple stages can occur at the same time. In the figure we can see the spam detection process:



In our project we have used 5types of machine learning algorithm for spam detection. We have compared those different types of machine learning techniques to see their accuracy rate, so that we can have a conclusion that which method is more compatible.

* **Multinominal Naïve Bayes Method:** A Naive Bayes classifier is a statistical technique used to filter e-mails. They use features such as bag-of-words to identify spam. It is very easy to use and high accuracy rate and also it is very fast. That’s why it is most commonly used in spam detection. By applying multinominal naïve bayes model to the dataset , we can see overall error is 1.12%, SC(spam caught) 94.5% and BH(blocked hams) 0.51%.
* **Support Vector machines :** We have used SVM in our project to show the accuracy of spam detection. Here we applied SVM with different kernels, they are : linear, Degree-2 Polynomial, Degree-3 Polynomial, Degree-4 Polynomial, Radial basis function, sigmoid. Overall error, SC and BH are:

there is the final graph after applying SVM in our datasets:

* **K-nearest Neighbor:** This algorithm is a simple instance learning algorithm. That’s why we used this to detect spam. There are some overall errors, SC, BH for different k values in our dataset:
* **Random Forest:** Random forest is a classification method combined with decision trees built from a bootstrap sample. It is an averaging ensemble method. In this method, 2 numbers estimators are simulated. Compared to naïve bayes method there is no improvement in the performance though complexity of model increase.
* **Adaboost:**  This method is a boosting ensembled method. Among all the methods, this method is very weak. But this classifier is definitely better than random guessing. Adaboost method usually used with another method to get better ensembled method. So we used it with decision trees using scikit-learn library. Though its complexity is high than naïve bayes, performance wise naïve bayes is more compatible than adaboost.

**3.Implementation:**

At first we have collected a dataset of both spam and ham SMS. Then Split them into train and test dataset. Test size is 30% and train size is 70%. Then find F1 score and Accuracy for those 5 ML.

**3.1. Data Collection:**

Link: [**https://www.kaggle.com/uciml/sms-spam-collection-dataset/version/1**](https://www.kaggle.com/uciml/sms-spam-collection-dataset/version/1)

Data Information:

Here we have total 5572 data where 4825 samples are ham and 747 are spam. After split training set has 3900 samples and testing set has 1672 samples. There are 2 columns “v1” and “v2”. “v1” for Class(spam/ham) , “v2” for Text.

**3.2. Data processing:**

* To filtering useless data we use stop words here. Stop words are ‘the’, ‘a’, ‘an’, ‘in’ etc. words that a search engine has programmed to ignore. NLTK has its own list of words for 16 different languages from where we use the list of English.
* “CountVectorizer” is used to convert a collection of text documents to a vector of term/ token counts. It also enables the pre-processing of text data prior to generating the vector representation. This functionality makes it a highly flexible feature representation module for text.
* “Fit-transform” is used for training data so that we can scale the training data and also learn the scaling parameter of that data. It calculate the means od columns and replacing the missing values also.

**3.3. Model Development:**

We used python language for develop our code and use Anaconda , Jupiter Notebook as our project environment.

**3.4. Result:**

|  |  |  |
| --- | --- | --- |
| **Model** | **F1 Score** | **Accuracy in %** |
| Multinomial NB | 0. 9237472766884532 | 97.9066985645933 |
| Linear SVC | 0.9220183486238532 | 97.96650717703349 |
| K Neighbors Classifier | 0.4768211920529802 | 90.55023923444976 |
| Random Forest | 0.883495145631068 | 97.1291866028708 |
| Ada Boost | 0.884160756501182 | 97.06937799043062 |

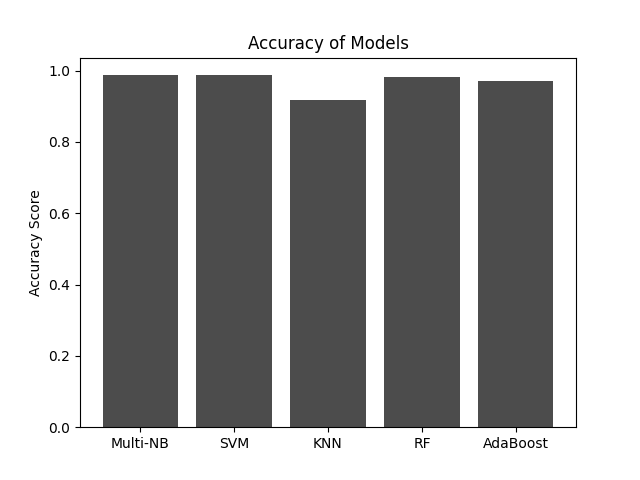
****

Fig: Accuracy of different Classifier Model

**4.Conclution:**

The results of multiple classification models applied to the SMS Spam dataset are shown in table IV. From simulation results, multinomial naive Bayes with Laplace smoothing and SVM with linear kernel are among the best classifiers for SMS spam detection. The best classifier in the original paper citing this dataset is the one utilizing SVM as the learning algorithm, which yields overall accuracy of 97.97% . Next best classifier in their work is boosted naive Bayes with overall accuracy of 97.91%.

**4.1. Challenges:**

* As we have used huge datasets so it consumes lots of time.

**4.2. Limitation:**

* here are some time complexity for big datasets.
* We have used 5000+ data, but if its accedes than the implementation cost and complexity cost will be higher.

**4.3. Future Directions:**

Adding meaningful features such as –

1. the length of messages in number of characters,
2. adding certain thresholds for the length,
3. analyzing the learning curves and misclassified data

have been the factors that contributed to this improvement in results.