SMS Spam Detection Using Machine Learning

SMS Spam Detection Using Machine Learning

Suparna DasGupta[1], **Soumyabrata Saha**[2*], **Suman Kumar Das**[3]

[1,2]Department of Information Technology, JIS College of Engineering, Kalyani, 741235, West Bengal, India
[3]Zenlabs, Zensar Technologies, Pune, 411014, Maharashtra, India
[2*]som.brata@gmail.com

Abstract. In the modern world where digitization is everywhere, SMS has become one of the most vital forms of communications, unlike other chatting-based messaging systems like Facebook, WhatsApp etc, SMS does not require active internet connection at all. As we all know that Hackers / Spammer tries to intrude in Mobile Computing Device, and SMS support for mobile devices had become vulnerable, as attacker tries to intrude to the system by sending unwanted link, with which on clicking those link the attacker can gain remote access over the mobile computing device. So, to identify those messages Authors have developed a system which will identify such malicious messages and will identify whether or not the message is SPAM or HAM (malicious or not malicious). Authors have created a dictionary using the TF-IDF Vectorizer algorithm, which will include all the features of words a SPAM SMS possess, based on content of message and referring to this dictionary the system will be classifying the SMS as spam or ham.

Keywords: SMS, SPAM and HAM, Machine Learning, TF-IDF Vectorizer, Text Classification

1. Introduction

SMS is one of the most effective forms of communication. It is based on cellular communication systems, just the cell phone needs to be in the network coverage area in order to send or receive the message. Almost everyone is using this service for communication. Various organizations deal with SMS for communicating with their clients / customers, banks and other government organizations also use SMS for communication. Also, many business organizations use this service for advertising purposes. Thus, SMS is playing a vital role, as active internet connection is not required at all in this framework. So due to large usage of SMS, it has become one of the most favorite places for hackers and spammers. It is quiet easy for a hacker to compromise any one's cell phone just by passing or transmitting Malicious link to end user, the mobile device will automatically be compromised if end user click on the link or message being transmitted by hacker / spammer, and we can know the rest how a hacker can exploit the system if he gains control of the system. So it has become very much important to restrict the content which the end user is receiving. So there must be a system which could tell the end user whether the received message is SPAM or not, Non SPAM message is known as HAM. So by identifying the above mentioned problems and issues, authors have developed a system which can identify whether a Message is SPAM or HAM based on the content of the message using Machine Learning technique. In this section authors have given a brief overview of Machine Learning; various types of Machine Learning and the techniques authors have used for developing the Machine Learning Model. Machine Learning: Machine learning is a fascinating domain as it incorporates substantial parts of different fields namely statistics, artificial intelligence theory, data analytics and numerical methods. Machine learning can be defined as semi-automated extraction of

knowledge from data sets or data. Let's break down the definition into three component parts. i) Firstly, machine learning always starts with data, with an objective to extract knowledge insight from the used data or data set. ii) Secondly, machine learning involves a certain amount of automation rather than trying to gather insights from the data manually. iii) Lastly, machine learning is not fully automated i.e. it requires human interventions to make many smart decisions for the process to be successful. Simply we can put, machine learning is an application that can improve its prediction results with successive iterations or it improves with experience. The process of an application improving with experience is, naturally enough, called Training. It can take significant iterations to gradually improve results. During the process of training, data is given to a machine-learning algorithm, which then refines its internal representation, numerical parameters, as it encounters any deviations or Training errors. The purpose of this stage is to minimize cost function, error function or maximizing likelihood by adjusting the algorithm's internal weights. When the algorithm accuracy improves, we call this learning. Once the results are accurate enough also known as scoring, the machine-learning application can be deployed to solve the problem that it was supposed to.

Machine learning is broadly categorized into two categories**:** a) Supervised Learning[3], b) Unsupervised Learning[4]. Main Categories of Machine Learning: Supervised Learning: Supervised learning also known as predictive modelling, is the process of making predictions using data. Examples of Supervised learning are Classification[5] and Regression[6]. A supervised learning Training data set is pre labelled for classification problems or function values are known in case of regression. After training is done and the model has a minimum cost function for the training data set, later switch for scoring where we can predict values for new data.

Classification: It identifies group membership. That means that if we have multiple events characterized by input parameters, which can be labelled differently, and we want our system to predict which label should be used.

Regression: Regression is a combination of multi-dimensional power supply and function interpolation. The regression problem is used to find the approximation of the function with a minimum error deviation or a cost function. In other words, the regression technique simply tries to predict numeric dependence, a function value, for example, of a data set. Figure 1. Diagrammatically shows how supervised learning is to solve problems

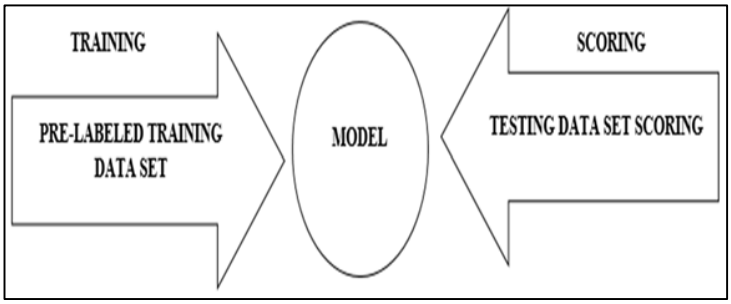

Fig. 1. Supervised Learning

Example of supervised learning, if a system has a data set which is a series of email messages, supervised learning task is to predict whether each email message is spam or non-spam(ham). This is supervised learning because there is a specific outcome namely spam or ham.

Unsupervised Learning: Unsupervised Learning is the process of extracting structure from data or how to best represent data. Examples of Unsupervised Learning are Clustering[7] (is partitioning a data set into meaningful similar sub classes called cluster) and Association[8] (method for discovering relations between existing attributes within a data set or data base). In an unsupervised learning situation, where the algorithm detects data features automatically, this depends on the purpose of the algorithm as well as the assumptions

made on what the properties and observed values are. Figure 2. Describes how unsupervised learning is used to solve problems.

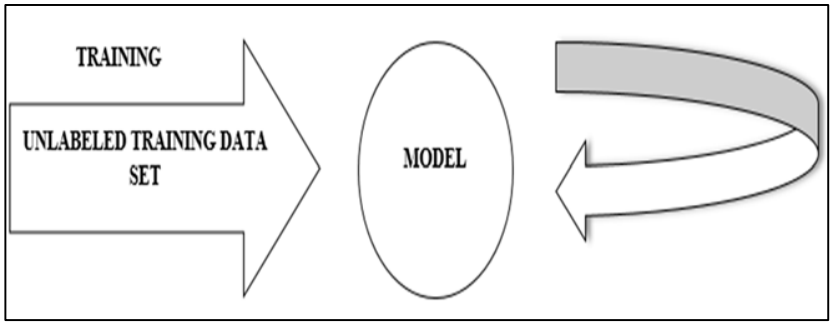

Fig. 2. Unsupervised Learning

For example, if any data set was the characteristics and purchasing behavior of shoppers at grocery stores, the unsupervised learning task might be to segment the shoppers into groups or "clusters" that exhibit similar behaviors. Such learning methods might find that college students, parents with young children, and older adults have characteristic shopping behaviors that are similar within each group but dissimilar from the other. This is an unsupervised learning task because there is no right or wrong about how many clusters can be found in the data, which people belong in which cluster, or even how to describe each cluster.

Now after having a clear understanding of Machine Learning, authors have used the same in generating the rules, which will help in governing or identifying based on inputs whether or not the message is SPAM or HAM. For processing the document content authors have used TF-IDF[9] vectorization for generating the Word Cloud. After that authors have briefly described TF-IDF vectorization works.

TF-IDF stands for Turn Frequency Inverse Document Frequency, used in machine learning and text mining as a weighting factor for identifying words features. The weight increases as the word frequency in a document increases, i.e. weight increases, the more times a term occurs in the document, but that offset by the number of times the word appears, in the entire data set or this offset helps remove the importance from really common words like 'the' or 'a' appear quite often in all across the document. It is used very often in relevance ranking and scoring and to move stop words from ML Model, where these stop words don't give any relevant information about a particular document type or class. Figure 3 represents the TF-IDF mathematical formula.

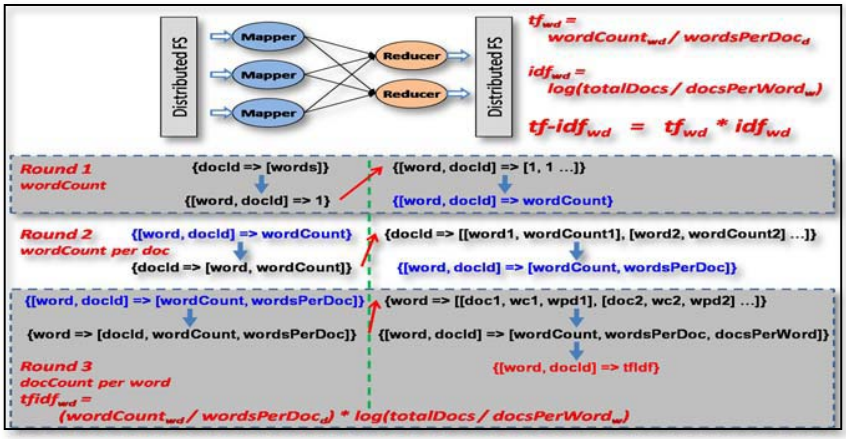

FIGURE 3. Equation for calculating TF-IDF

2. Related Work

After the evolution of Machine Learning algorithm and its usage in document classification, a lot has been research done on identifying the features of text. In this section authors have described the work done by researchers in field of identifying Texts features by limiting their study solely on the field of SPAM identification. M. Nivaashini et al[10] has used various Deep Neural Network (DNN) techniques in identifying the SPAM and HAM after collecting the dataset from UCI Machine Learning Repository. Authors have compared all the used algorithms based on their accuracy, False-Positives, False Positives and high chances for identifying SPAM with low False Positive rates, in order to identify the best algorithm. Dr. Dipak R. Kawade, Dr. Kavita S. Oza[11] have identified SMS SPAM using spam filtering techniques, by using open source python software, they have achieved 98% accuracy. For studying and preprocessing they have used WEKA too. P. Navaney[12] et. Al has used various supervised based machine learning algorithms like Naive Bayes, Support Vector Machine Algorithm and Maximum Entropy Algorithm, and they have done an accuracy comparison, and it was found that SVM was having more accuracy. Bichitrananda[13] et. al have used various ML algorithm like SVM (Support Vector Machine), Decision Tree, KNN (K-Nearest Neighbor), Neural Network (including Back-Propagation, Perceptron, Stochastic Gradient) for automatic classifying text documents on Datasets obtained from 20Newsgroup, IMDB, BBC News & BBC Sports, also they have compared the performance of all the Algorithms using metrics such as Kappa Statistics, Error Rate, Precision Call, Accuracy, F-Measure. Bichitrananda[14] et. al have built an automated document classifier for biomedical data sets (like TREC 2006 genetic Track, Farm-Das, Bio Creative Corpus III) using ML algorithms. All the Algorithms used for the task were evaluated and compared on the basis of ML Classification metrics like accuracy, precision, recall & f1-measure. Leila Arras[15] et. al have demonstrated a method to extract the abstract out from the document using Machine Learning Algorithms like Convolutional Neural Network & SVM Classifier. Francis M Kale[16] et. al have proposed a framework for performing text mining & text clustering used the K Means algorithm and its application in various areas. This paper gives guidance to researchers for test clustering being the state of the art of text mining. Ting S.L[17] et. al performed text mining on vast and large datasets using various classification-based Machine Learning algorithms like decision tree, neural network, SVM (support vector machines) and also compared each of the classifiers on the basis of computational efficiency and accuracy. Naïve Bayes was found to be the best & efficient classifier amongst all other classifiers.

3. Methodology

3.1. Workflow:

Data Collection: In this phase authors have collected a dataset based on which they have performed the experimentation from Kaggle Repository[14].

Data Cleaning: In this phase the authors have cleansed all the data which were taken into consideration. Authors have removed all the white-spaces, lowered the alphabet so that words like Equal and equal become the same, remove the remaining punctuation, like! is not that much important, tokenize each message, to represent the message as a list of words and done stemming, converting all the words to their root word, like floor, floored to floor.

Generating Testing and Training Data Sets: Authors have created the testing and training data on the converted cleansed datasets.

Generating Word Cloud Vector: Authors have used the TF-IDF vectorization for creating the word-vector. On the basis, the spam feature will be classified.

Prediction: Authors have given input messages to check whether the message is SPAM or HAM.

Figure 4. shows the workflow or architectural layout of how authors have classified the SPAM.

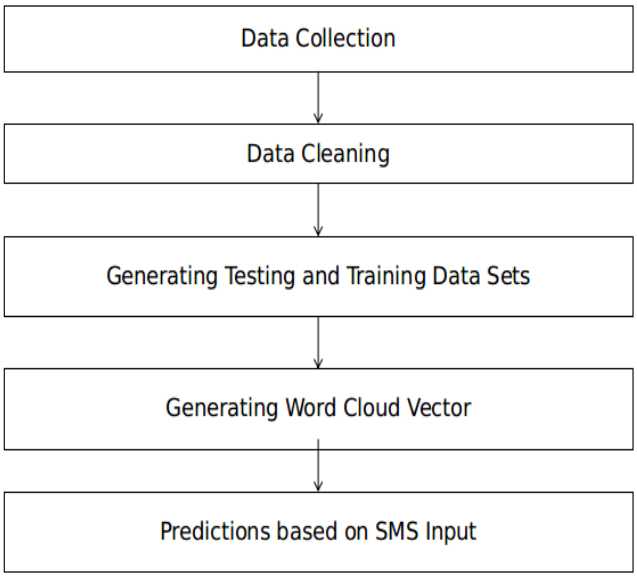

Figure 4. Proposed Process flow of the Entire Work

4. Experimentation

As a part of experimentation, authors after creating the vector set, passed 2 inputs to test whether or not the model (including the word vectors) is able to check whether the message is SPAM or HAM.

Input I: given to the developed system: “Hello, how are you?”
Input II: given to the developed system: “Congratulations!!! You have won 5000$”

Output of the above inputs are discussed in the Section IV of this literature.

5. Results and Discussion

Technology Stack Used in this research: Python based Flask Platform. Python Module Dependencies: beautifulsoup4==4.6.0, numpy==1.13.1, scikit-learn==0.19.0, scipy==0.19.1, sklearn==0.0, pandas, flask.

In this section authors have discussed the output generated by the system on the basis of Inputs given in the system, discussed in Section III of this literature. Figure 5, represents the Output 1 where the system was given Input I as: “Hello, how are you?”

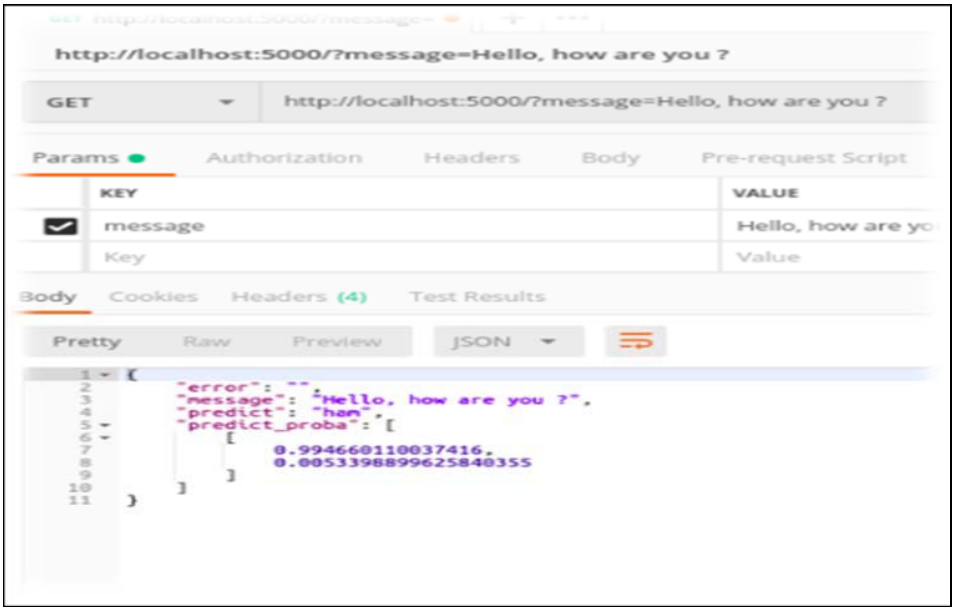

Fig. 5. Prediction 1 based on Input I

Similarly Figure 6 represents Prediction 2 based on Input II discussed in Section III.

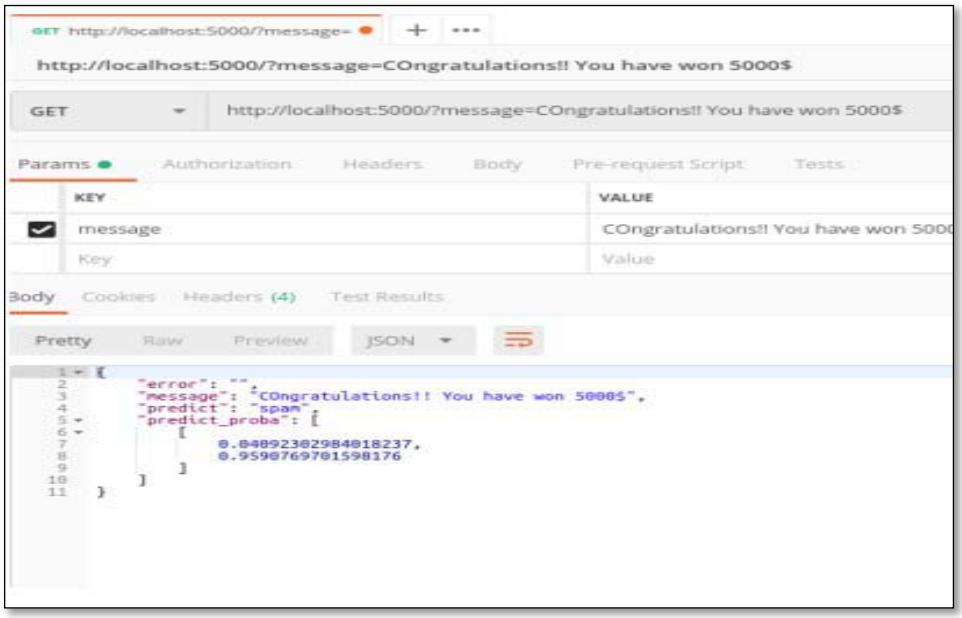

Fig. 6. Prediction 1 based on Input II

Figure 7. Shows evidence for both the inputs passed / POST at Server http://localhost:5000 for Processing.

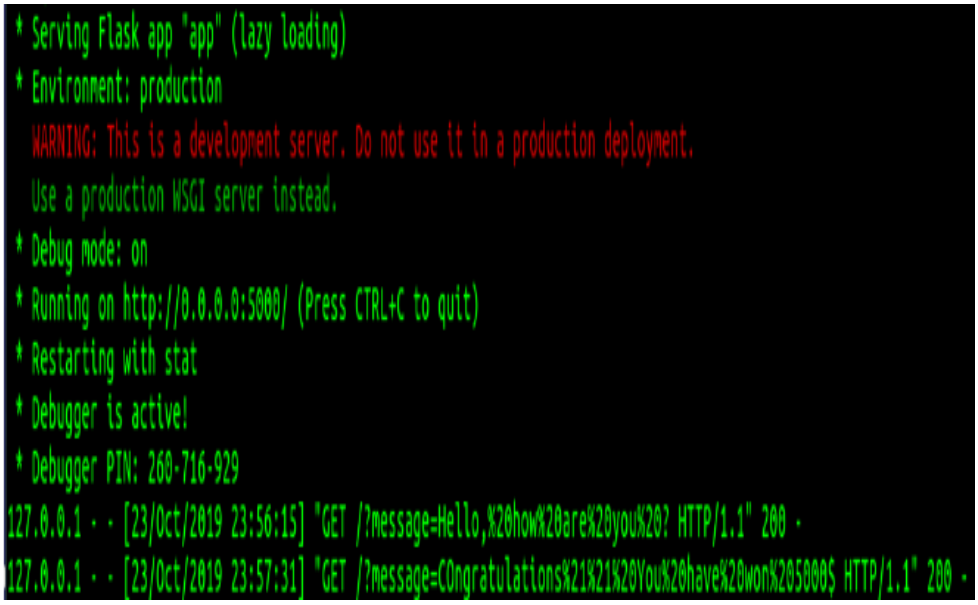

Fig. 7. Input I and II received at Python Backend Server

It is clear from Figure 6 that the system identifies the message "Hello, how are you?" as HAM which is true, with accuracy of 99.46%, and from Figure 7 the system has identified the message "Congratulation!! You have won 5000$" as SPAM, which is also true, with accuracy of 95.90%.

6. Conclusion

From the above discussion and experimentation authors have concluded that Machine Learning algorithms can play a vital role in identifying SPAM SMS. The accuracy obtained in this work is more than 95% in both the cases.

References

1. Njoku, Mary Gloria. (2015). The use of short message service in post-secondary education.
2. Mehryar Mohri, Afshin Rostamizadeh, Ameet Talwalkar, "Foundations of Machine Learning", The MIT Press ISBN 9780262018258, 2012.
3. Sumit Das, Aritra Dey, Akash Pal, Nabamita Roy, "Application of Artificial Intelligence in Machine Learning: Review And Prospect", International Journal of Computer Applications", Volume 115, Number 9. 2015.
4. S.B. Kotsiantis. "Supervised Machine Learning: A Review of Classification Techniques" Informatica 31 (2007) 249-268
5. Nanhay Singh, Ram Shringar Raw, Chauhan R.K. ," Data Mining With Regression Technique", Journal of Information Systems and Communication ISSN: 0976-8742 & E-ISSN: 0976-8750, Volume 3, Issue 1, 2012, pp.-199-202
6. Amandeep Kaur Mann & Navneet Kaur, "Review Papers on Clustering Techniques", Global Journal of Computer Science and Technology Software & Data Engineering, Volume 13, Issue 5, Version 1.0, 2013.
7. Ashima Sethi, Prerna Mahajan, "Association Rule Mining: A Review", The International Journal of Computer Science and Application, Volume 1, No. 9, November 2013.
8. Swati Gupta, "A Regression Modeling Technique on Data Mining", International Journal of Computer Applications, Volume 116, No. 9, April 2015.

9. Qaiser, Shahzad & Ali, Ramsha. (2018). Text Mining: Use of TF-IDF to Examine the Relevance of Words to Documents. International Journal of Computer Applications. 181. 10.5120/ijca2018917395.
10. M. Nivaashini, R.S.Soundariya, A.Kodieswari, P.Thangaraj, SMS Spam Detection using Deep Neural Network, International Journal of Pure and Applied Mathematics, Volume 119 No. 18 2018, 2425-2436 ISSN: 1314-3395 (on-line version), url: http://www.acadpubl.eu/hub/, Special Issue
11. Dipak, R & Kawade, Dipak & Oza, Kavita. (2018). CONTENT-BASED SMS SPAM FILTERING USING MACHINE LEARNING TECHNIQUE. 12.
12. P. Navaney, G. Dubey and A. Rana, "SMS Spam Filtering Using Supervised Machine Learning Algorithms," 2018 8th International Conference on Cloud Computing, Data Science & Engineering (Confluence), Noida, 2018, pp. 43-48. doi: 10.1109/CONFLUENCE.2018.8442564.
13. Behera, Bichitrananda & Kumaravelan, G.. (2019). Towards the Deployment of Machine Learning Solutions for Document Classification. International Journal of Computer Sciences and Engineering. 7. 193-201. 10.26438/ijcse/v7i3.193201.
14. Bichitrananda Behera, G.Kumaravelan. (2020). Performance evaluation of Machine learning algorithms in Biomedical Document Classification. International Journal of Advanced Science and Technology,29(05),5704-5716.
15. Retrieved from http://sersc.org/journals/index.php/IJAST/article/view/15054
16. Arras L, Horn F, Montavon G, Müller K-R, Samek W (2017) "What is relevant in a text document?": An interpretable machine learning approach. PLoS ONE 12(8): e0181142. https://doi.org/10.1371/journal.pone.0181142
17. Francis M. Kwale, “An Efficient Text Clustering Framework”, International Journal of Computer Applications (0975 – 8887) Volume 79, No.8, October 2013.
18. Ting, S.L. & Ip, W.H. & Tsang, Albert. (2011). Is Naïve Bayes a Good Classifier for Document Classification?. International Journal of Software Engineering and its Applications
19. Open_Access:https://www.kaggle.com/uciml/sms-spam-collection-dataset