SMS SPAM DETECTION USING SVM WITH VARIOUS KERNEL FUNCTIONS

Sapna Malviya
Mtech (Research Scholar)
CSE, SIRT Bhopal

Ruchi Dronawat
Assistant Professor
CSE, SIRT Bhopal

Abstract: **SMS spams are dramatically increasing year by year because of the expansion of movable users round the world. Recent reports have clearly indicated an equivalent. Mobile or SMS spam may be a physical and thriving drawback because of the actual fact that bulk pre-pay SMS packages are handily obtainable recently and SMS is taken into account as a trusty and private service. SMS spam filtering may be a relatively recent trip to deal such a haul. The amount of information traffic moving over the network is increasing exponentially and therefore the devices that are connected thereto are considerably vulnerable. Thus there's a bigger have to be compelled to secure our system from this kind of vulnerability, here network security play a really vital role during this context. In this paper , we proposed SVM Algorithm to identify Spams. SVM map input vector to a higher dimensional space where a maximal separating hyperplane is constructed and if we need a nonlinear dividing line. Rather than fitting nonlinear curves to the data, SVM handles this by using a kernel function to map the data into a different space where a hyperplane can be used to do the separation. The kernel function may transform the data into a higher dimensional space to make it possible to perform the separation, So in this we tune the algorithm based on various kernel functions such as linear, polynomial, radial and sigmoidal through which we can get the optimal accuracy.**

Keywords: data mining,review spam, classification, comparative analysis, spam detection, SVM, Kernel.

I.INTRODUCTION

Short Message Service (SMS) is that the most often and wide used communication medium. The term "SMS" is employed for each the user activity and every one sorts of short text electronic messaging in several components of the planet. it's become a medium of promotion and promotion of product, banking updates, agricultural data, flight updates and net offers. SMS is additionally utilized in direct called SMS marketing. Typically SMS promoting could be a matter of disturbance to users. These types of SMSs area unit referred to as spam SMS. Spam is one or a lot of uninvited messages, that is unwanted to the users, sent or announce as a part of a bigger assortment of messages, all having considerably identical content. The needs of SMS spam area unit promotion and promoting of assorted product, causing political problems, spreading inappropriate adult content and net offers. That's why spam SMS flooding has become a significant downside everywhere the planet. SMS spamming gained quality over alternative spamming approaches like email and twitter, thanks to the increasing quality of SMS communication. However, gap rates of SMS area unit beyond ninetieth and opened inside quarter-hour of receipt whereas gap rate in email is just 20-25% inside twenty four hours of receipt [28]. Thus, a correct SMS spam detection technique has important necessity. There are many researches on email, twitter, net and social tagging spam detection techniques. However, a awfully few researches are conducted on SMS spam detection. Spam SMS detection is more difficult than email spam detection attributable to the restricted length of SMS, use of regional content cut-off words and SMS contains less header data than an email. We have a tendency to cannot use techniques of email spam detection asis in SMS spam detection. Correct SMS spam detection technique is required to be known. This can be Associate in nursing open and relatively new analysis field. There's an enormous scope of analysis add this field.

II. LITERATURE REVIEW

According to [1], conducted a survey of ways and applications for detection and filtering uninvited advertising messages or spam in a very telecommunication network. The result showed that if the message passed the spam screening, the first mail would be delivered to its supposed destination. However, the survey focused a lot of on the techniques used

for e-mail spam detection and excluded alternative types of mobile SMS spam techniques like the unreal system[2].
Building up a classification algorithmic [3] program that channels SMS spam would provide useful equipment for portable suppliers. Since naïve mathematician has been used effectively for email spam detection, it seems to be expected that it may likewise be accustomed build SMS spam classifier. With reference to email spam, SMS spam represents further difficulties for automatic channels [4]. SMS texts area unit often restricted to one hundred sixty characters, modification the live of content which will be used to differentiate whether or not a message could be a ham or spam. Individuals have conjointly often started mistreatment shorthand notations and slang that additional makes it troublesome to differentiate between ham and spam[5]. We are going to check however well a straightforward naïve mathematician classifier manages these difficulties.[6]

III PROBLEM DEFINITION

The fact that an email box can be flooded with unsolicited emails makes it possible for the account holder to miss an important message; thereby defeating the purpose of having an email address for effective communication. These junk emails from online marketing campaigns, online fraudsters among others is one of the reasons for this paper. We try to obtain the feature sets that can best represent and distinguish the spams from ham(non-spam). We then follow both supervised and unsupervised methodology to obtain spams from the dataset. We also include sentiment analysis methodology into our spam detection. Lastly, we compare our analysis obtained from taking various types of feature sets based on text, sentiment scores, reviewer features, as well as the combined method.

IV OBJECTIVE

The main objective of this paper is to build a spam filter that can effectively categorise an incoming mail or text message as either “spam” or “ham”. We will use a dataset from the dataset repository of Center for Machine Learning and Intelligent Systems at the University of California, Irvine!.

This dataset consists of 5574 observations of 2 variables. The first variable is the content of the emails and the second variable the target variable, which is the class to be predicted. The target variable can either be a “spam” or “ham”. We will be building this classier using the text messages from the email.

- Available at UCI Machine Learning Repository
- Number of instances – 5572 ; no missing values
- Number of columns – 2 (label & message)
- Classification levels – 2 i.e. spam and not spam

V PROPOSED WORK

A machine learning techniques have been proposed for detecting and classify the reviews through various processing steps which is shown in figure 1.

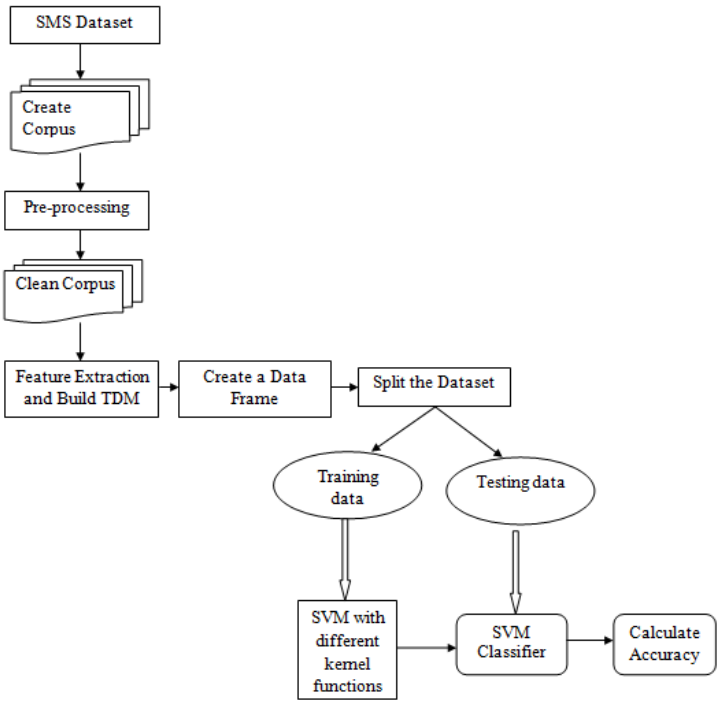

Figure-1. Proposed Flow Diagram

Our Steps or Algorithm Steps will follow:
1. Data Collection:- We will use a dataset from the dataset repository of Centre for Machine Learning and Intelligent Systems at the University of California, Irvine!.
2. Data Preprocessing: Before preprocessing we create a corpus which is nothing but a collection of documents, so we just apply the pre-processing steps on corpus and all the documents gets cleans. Data preprocessing is the most important phase in detection models as the data consists of ambiguities, errors, redundancy which needs to be cleaned beforehand so for these we use the tm(text mining) package of R in which various inbuilt pre-processing functions is present using which we can easily remove the redundancy from the data..
3. Data Transformation: Data is transformed into lowercase and change the data types according to algorithm needs which is done by using tm package and after cleaning the data we perform data extraction task through which we can gets the weighted keywords which is useful in spam classification.
4. Classification System: The words are identified based on weighted keywords for classifying process and classification system classify the content into spam or ham. For these we use the SVM algorithm and which is tuned based on various

kernel functions such as linear, sigmoidal, radial and polynomial. The kernel function may transform the data into a higher dimensional space to make it possible to perform the separation. Kernel functions are a class of algorithms for pattern analysis or recognition, whose best known element is the support vector machine (SVM).

linear: $u'v$

polynomial: $(\gamma u'v + coef0)^{degree}$

radial basis: $e^{(} - \gamma|u - v|^2)$

sigmoid: $tanh(\gamma u'v + coef0)$

V Experimental Analysis

The experimental and result analysis is done by using intel i5-2410M CPU with 2.30 GHz processor along with 4 GB of RAM and the windows operating system is running. For result analysis we use R and R studio for processing the data R could be a standard artificial language that is usually embraced by data researchers. In any case, traditional R should be dead during a solitary machine atmosphere. because the volume of accessible data income to quickly develop from an assortment of sources, versatile and execution investigation arrangements have was a basic device to upgrade business gain and financial gain. and then we load the sms dataset which consist a 5574 observation with no missing are present in the dataset. Figure 2 shows the dataset has been loaded.

```
> data_text <- read.delim("smsspamcollection", sep="\t", header=F, colClasses="chara
cter", quote="")
>
> str(data_text)
'data.frame':   5574 obs. of  2 variables:
 $ V1: chr  "ham" "ham" "spam" "ham" ...
 $ V2: chr  "Go until jurong point, crazy.. Available only in bugis n great world la
 e buffet... Cine there got amore wat..." "Ok lar... Joking wif u oni..." "Free entr
y in 2 a wkly comp to win FA Cup final tkts 21st May 2005. Text FA to 87121 to recei
ve entry question("| __truncated__ "U dun say so early hor... U c already then sa
y..." ...
>
> head(data_text)
    V1
1  ham
2  ham
3 spam
4  ham
5  ham
6 spam

                                                      V2
1                                 Go until jurong point, crazy.. Availab
le only in bugis n great world la e buffet... Cine there got amore wat...
2                                     Ok lar... Joking wif u oni...
3 Free entry in 2 a wkly comp to win FA Cup final tkts 21st May 2005. Text FA to 871
21 to receive entry question(std txt rate)T&C's apply 08452810075over18's
4                         U dun say so early hor... U c already then say...
5                Nah I don't think he goes to usf, he lives around here though
6       Freemsg Hey there darling it's been 3 week's now and no word back! I'd like
 some fun you up for it still? Tb ok! Xxx std chgs to send, £1.50 to rcv
```

Figure 2. Loading a dataset

After loading , For easy identification of the columns, we rename V1 as Class and V2 as Text. And we have to also convert the Class column from Character strings to factor. Data often come from different sources and most of the time don't come in the right format for the machine to process them. Hence, data cleaning is an important aspect of a data science project, we use the tm(text mining) package of R in which various inbuilt pre-processing functions is present using which we can easily remove the redundancy from the data. In text mining, we need to put the words in lowercase, remove stops words that do not add any meaning to the model etc. Various data cleaning steps are shown in figure 3.

```
      ham      spam
0.8659849 0.1340151
> library(tm)
Loading required package: NLP
>
> library(SnowballC)
> corpus = VCorpus(VectorSource(data_text$Text))
> as.character(corpus[[1]])
[1] "Go until jurong point, crazy.. Available only in bugis n great world la e buffet...
ere got amore wat..."
>
> corpus = tm_map(corpus, content_transformer(tolower))
> corpus = tm_map(corpus, removeNumbers)
> corpus = tm_map(corpus, removePunctuation)
> corpus = tm_map(corpus, removewords, stopwords("english"))
> corpus = tm_map(corpus, stemDocument)
> corpus = tm_map(corpus, stripWhitespace)
> as.character(corpus[[1]])
[1] "go jurong point crazi avail bugi n great world la e buffet cine got amor wat"
>
>
> #Creating the Bag of words for the model
>
> dtm = DocumentTermMatrix(corpus)
> dtm
<<DocumentTermMatrix (documents: 5574, terms: 6981)>>
Non-/sparse entries: 43801/38868293
Sparsity           : 100%
Maximal term length: 40
Weighting          : term frequency (tf)
>
> dtm = removeSparseTerms(dtm, 0.999)
>
```

Figure 3. Data pre-processing

In text mining, it is important to get a feel of words that describes if a text message will be regarded as spam or ham. What is the frequency of each of these words? Which word appears the most? In other to answer this question; we are creating a DocumentTermMatrix to keep all these words. We want to words that frequently appeared in the dataset. Due to the number of words in the dataset, we are keeping words that appeared more than 60 times. We will like to plot those words that appeared more than 60 times in our SMS spam dataset which we have collected from UCI respository. Figure 4 shows the wordcloud of the dataset.

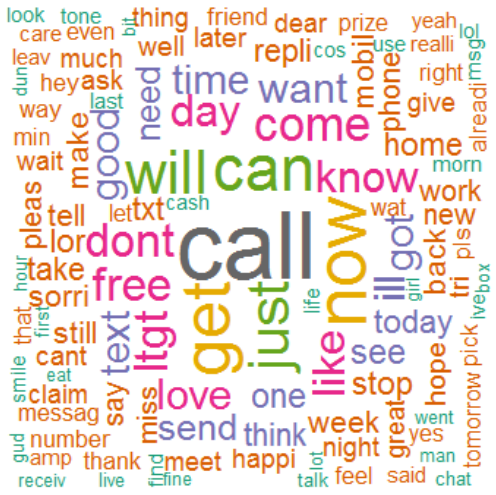

Figure 4. Presenting the word frequency as a word cloud

Usually in Machine Learning is to split the dataset into both training and test set. While the model is built on the training set; the model is evaluated on the test set which the model has not been exposed to before. We will be building a SVM Model. The Support Vector Machine is finds the hyperplane that differentiates the two classes to be predicted, ham and spam in this case; very well. SVM can perform both linear and non-linear classification problems. The figure shows the performance outcomes of the model on the testing dataset. Figure 5 shows the training step of SVM algorithms with different kernel functions.

```
## Linear Kernel
svm.linear <- svm(Class~., data=train, scale=FALSE, kernel='linear')
pred.linear <- predict(svm.linear, test[,-1])
linear <- confusionMatrix(pred.linear,test$Class)

## Linear Kernel
svm.poly <- svm(Class~., data=train, scale=FALSE, kernel='polynomial')
pred.poly <- predict(svm.poly, test[,-1])
poly <- confusionMatrix(pred.poly,test$Class)

## Radial Basis Kernel
svm.radial <- svm(Class~., data=train, scale=FALSE, kernel='radial')
pred.radial <- predict(svm.radial,test[,-1])
radial <- confusionMatrix(pred.radial,test$Class)

## Sigmoid Kernel
svm.sigmoid <- svm(Class~., data=train, scale=FALSE, kernel='sigmoid')
pred.sigmoid <- predict(svm.sigmoid,test[,-1])
sigmoid <- confusionMatrix(pred.sigmoid,test$Class)
```

Figure 5. Training Steps of SVM with different kernet function

After the SVM models get trained with different kernel functions, we can test these model on testing datasets and the prediction performance of these models are shwon in figure 6.

```
> #Accuracies
>
> Kernels <- c("Linear","Polynomial","Radial Basis","Sigmoid")
> Accuracies <- round(c(linear$overall[1],poly$overall[1],radial$overall[1],si
4)
> acc <- cbind(Kernels,Accuracies)
> kable(acc,row.names=FALSE)


|Kernels      |Accuracies |
|:------------|:----------|
|Linear       |0.9901     |
|Polynomial   |0.9874     |
|Radial Basis |0.974      |
|Sigmoid      |0.7783     |
>
```

Figure 6. Compare SVM Models With Different Kernel Functions

Performance Measure

We used accuracy, which are derived using confusion matrix.

Table-1 Confusion Matrix

	Classified as Normal	Classified as Attack
Normal	TP	FP
Attack	FN	TN

Where

TN -Instances correctly predicted as ham.

FN - Instances wrongly predicted as ham.

FP -Instances wrongly predicted as spam.

TP -Instances correctly predicted as spam.

Accuracy = Number of samples correctly classified in test data / Total number of samples in test data

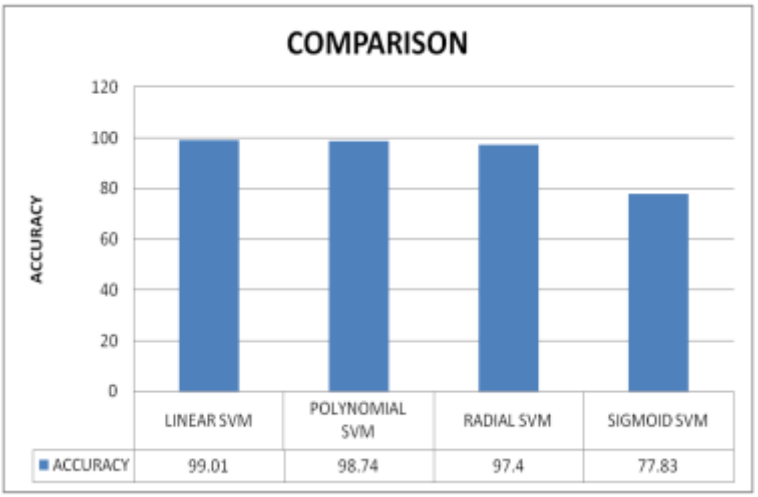

Figure 7. Accuracy Comparison

V CONCLUSION

Identifying fake reviews from a large dataset is challenging enough to become an important research problem. Business organizations, specialists and academics are battling to find the best system for opinion spam analysis. A single algorithm cannot solve all the problems' and challenges faced in today's generation with advancements in technologies, though a few are very efficient in analysis. In this we proposed a SVM algorithm are good in detecting spam reviews. In this we tune the algorithm with different kernel functions and we find that linear kernel SVM perform better as compared to others.

REFERENCES

[01] Pavas Navaney, Gaurav Dubey, Ajay Rana, "SMS Spam Filtering using Supervised Machine Learning Algorithms" in IEEE 2018.

[02Shubhi Shrivastava, Anju R, "Spam Mail Detection through Data Mining Techniques" in IEEE 2017.

[03] Shafi'i Muhammad Abdulhamid, Muhammad Shafie Abd Latiff, Haruna Chiroma, Oluwafemi Osho, Gaddafi Abdul-Salaam, Adamu I. Abubakar, and Tutut Herawan, "A Review on Mobile SMS Spam Filtering Techniques " in IEEE 2017.

[4] Jindal, Nitin, and Bing Liu. Mining comparative sentences and relations. AAAI. Vol. 22. 2006..

[5] Jindal Nitin, LIU Bing. Review spam detection[C] // Proceedings of the 16th international conference on World Wide Web. Canada. New York: ACM Press ,2007: 1189-1190.

[6] Jindal Nitin, Liu Bing. Opinion spam and analysis[C]// Proceedings of the 2008 International Conference on Web Search and Data Mining. New York: ACM Press , 2008:219-230.

[7] Xie Sihong, WANG Guan, LIN Shuyang, et al. Review spam detection via temporal pattern discovery [C] // Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery And Data Mining. New York: ACM Press , 2012:823-831.

[8] Lim Ee-Peng, Nguyen Viet-An, Jindal Nitin, et al. Detecting product review spammers using rating behaviors [C]// Proceedings of the 19th ACM international conference on Information and knowledge management. New York: ACM Press , 2010:939-948.

[9] Jindal Nitin, Liu Bing, Lim Ee-peng, et al. Finding unusual review patterns using unexpected rules[C] // Proceedings of the 19th ACM International Conference on Information and Knowledge Management. New York: ACM Press , 2010:1549-1552

[10] Wang Guan, Xie Sihong, Liu Bing,et al. Identify online store review spammers via social review graph[J]. ACM Transactions on Intelligent Systems and Technology, 2011,3(4):61.1-61.21.

[11] Nitin Indurkhya, Fred J Damerau. Handbook of natural language processing[M]. 2nd. Chapman and Hall/CRC, 2010.

[12] Ohana B, Tierney B. Sentiment classification of reviews using SentiWordNet[C]//9th. IT & T Conference. 2009: 13.

[13] Hu X, Tang J, Gao H, et al. Unsupervised sentiment analysis with emotional signals[C]//Proceedings of the 22nd international conference on World Wide Web. International World Wide Web Conferences Steering Committee, 2013: 607-618.

[14] Pak A, Paroubek P. Twitter as a Corpus for Sentiment Analysis and Opinion Mining[C]//LREC. 2010.

.