SMS Spam Detection using Machine Learning

Jigar Parmar
Dharmsinh Desai University
parmarjigar4@gmail.com

Akash Patel
Dharmsinh Desai University
akash.patel2520@gmail.com

Abstract:

Due to increase in benefits of SMS the supply of spam messages has increased in the recent years. In parts of Asia up to 30% of messages were spam in 2012. As there is lack of real databases of spam messages. In this project, a database from UCI Machine Learning Repository is used for training the algorithm to identify spam.

I. Introduction:

The usage and selling of mobile phones has undergone an exponential growth in the recent years. A short-message-service(SMS) is text messaging service provided by telecommunication companies. The importing of smartphones has increased from 3% to 10.3% in 2017 [1]. The use of mobile phones is increasing exponentially, out of 131 crores people approx 118 crore people use mobile phones [2] and this number will keep on increasing. This increasing number of mobile phones has lead to use them as an advertising medium and also a basic source of communication and thus mobile devices have become the largest target for spammers.
One of the reasons why spamming has increased is due to decreasing cost of SMS. Nowadays there are many online sites or services which provide BULK SMS, which is the dissemination of large numbers of SMS messages for delivery to mobile phone terminals. The rates of BULK SMS are so low that one SMS costs about 12.0 paisa for 10,000,000 messages [3]. Even some Operators provide SMS free of charge but limited to 100 SMS per day as advised by TRAI.

Why use SMS-filtering application?
Spam detection can be done in 2 ways
1.) Filter spam message at the operator end.
2.) On the user's device.

There are many reasons why spam filtering should not be done at operator end. First of all some messages cannot be blocked by the operator according to TRAI guideline [5]. Secondly, the SMS are not sent via centralized server, instead when SMS is sent the message flows through the Short Message Service Center (SMSC), then to the tower, and the tower sends the message to the phone as a small packet of data on the control channel. As SMS is sent over a packet, it can bypass the spam filters, if deployed on the operator end. So to overcome all these hurdles a SMS app on the user side becomes necessary to filter spam messages.

II. Proposal

This project aims to lessen the hassle and enhance the usability of mobile devices by segregating the incoming messages into spam or not spam.

III. Feature Extraction

As mentioned earlier, a dataset of UCI Machine Learning Repository is used which consists of 5574 messages including spam and not spam. Extracting features will result in 7726 features. Feature extraction is done using CountVectorizer and TfidfTransformer of scikit-learn library. CountVectorizer Converts a collection of text documents to a matrix of token counts. This implementation produces a sparse representation of the counts. TfidfTransformer converts a collection of raw documents to a matrix of TF-IDF features (TF = Term frequency, IDF = Inverse Document Frequency). Apply Term Frequency Inverse Document Frequency normalization to the sparse matrix of occurrence counts by CountVectorizer.

IV. Support Vector Machine

We apply LinearSVC classifier of svm to for training on the data. The advantage of using LinearSVC over svm is that the training complexity in LinearSVC is O(m x n) where
'm' = number of training instances
'n' = number of features
while the time complexity of svm is usually between $O(m^2 \times n)$ and $O(m^3 \times n)$ this means that it gets dreadfully slow when number of training instances gets large (e.g. hundreds of thousands of instances) [8].

In LinearSVC we use 'C' hyperparameter to control the balancing. The higher the value of C leads to a narrow margin that leads to fewer margin violations and lower value of C leads to wider margin which leads to more margin violations.

Why use Support Vector machine?
In compare to all the other algorithm like RainForest, Adaboost with decision trees, k-nearest neighbour, SVM seems to have better accuracy for detecting spam [9].

V. Result

Here, we have used two svm classifiers objects one with cross validation and other without it. With cross-validation we divide the dataset in 3 parts (default fold for cross-validation), and now we compare the average accuracy of cross-validation with accuracy without cross-validation

Average accuracy with Cross-validation: 0.981521349121
Accuracy without Cross-validation: 0.987443946188

Confusion matrix:

C = [[951, 11],
[3, 150]]

C[0,0] contains True negatives i.e. True Not Spam detected as Not Spam
C[0,1] contains False Positives i.e. True Not Spam detected as Spam
C[1,0] contains False Negatives i.e. True Spam detected as Not Spam
C[1,1] Contains False Positives i.e. True Spam detected as Spam

VI. Limitation and Future Extensions

I.) Limitations: As it uses supervised machine learning so the amount of dataset applied to it is very less and thus sometimes a legit message from a known person can also be classified as spam due to content of the message

ii) Future extensions: We can use the application to gather sms from the user's device on user's permission and use the data to train our classifier very accurately and in future we can also separate Bank messages, Promotional messages, Spam messages, and various other types. And to make this more efficient we can use LSTM Neural network for text classification.

VII. References:

[1] Share of smartphone users in India in 2013 and 2017
https://www.statista.com/statistics/255903/indias-global-smartphone-market-share/

[2] Number of Mobile phones in India
https://en.wikipedia.org/wiki/List_of_countries_by_number_of_mobile_phones_in_use#cite_note-TRAI-data-5

[3] Bulk SMS Price rate
https://www.textlocal.in/bulk-sms-pricing-india

[4] https://en.wikipedia.org/wiki/Mobile_phone_spam

[5] Sms policy TRAI
https://www.medianama.com/2011/09/223-bulk-sms-policy-trai-spam/

[6] How sms is sent
https://computer.howstuffworks.com/e-mail-messaging/sms.htm

[7]http://scikit-learn.org/stable/modules/generated/sklearn.feature_extraction.text.CountVectorizer.html

[8] Hands-on with scikit-learn and Tensor Flow - Aurelien Geron

[9] Sms spam detection machine learning approach - Houshmand Shirani-Mehr (Stanford University)