International Journal of Applied Engineering Research ISSN 0973-4562 Volume 10, Number 24 (2015) pp 44483-44488

A Real-Time Application for the Detection and Mitigation of Spam SMS in Android

Anupama Sheela
Amrita Center for Cyber Security Systems and Networks Amrita University, Kollam Kerala, India
anu109.indeevaram@gmail.com

Kamalanathan Kandasamy
Amrita Center for Cyber Security Systems and Networks Amrita University, Kollam Kerala, India
kamalanathan@am.amrita.edu

Abstract
With developments in the mobile technology over recent years, SMS texting has become more popular than before. This has led to increase in the number of spam messages communicated via mobile phones. Even though email was the main source for spam in earlier days, SMS service is not far behind in contributing to spam messages these days. But nobody wants their mobile phones to get filled with spam texts. Various strategies have been developed to detect and reduce spam messages and researches are still going on regarding this topic. In this paper, a hybrid framework to detect mobile spams is proposed. Our approach is a combination of different techniques like number based filtering, URL based classification, machine learning approach, text based filtering and detection based on challenge response. Performance of the proposed system was evaluated at the end and we observed that a combination of different methods is more effective rather than using a single method. The proposed methodology was implemented in Android platform and tested using an Android device.

Keywords- Spam, Ham, Blacklist, Machine learning, Support vector machine, Challenge response, Text based filtering, Confusion matrix, Precision rate, Recall rate, Short URL

I. INTRODUCTION

Short Message Service (SMS) has become a popular means of communication and the low cost of SMS has led to a large number of spamming attacks using text messages. SMS spam refers to unsolicited or unwanted text messages sent to cellular devices through text messaging. It is also referred to as mobile spam, m-spam or text spam. An example of SMS spam is shown in Fig. 1.

There are different categories of SMS spams. Most popular kinds of SMS spam include SMS consisting of free gift offers and commercial messages. Commercial messages are those which include advertisements about goods or services for sale. This is also referred to as cell phone advertising or mobile marketing. Social engineering, another type of SMS spam, is the art of manipulating people so that they give up confidential information. This type of attack makes use of victim's trust and curiosity. For example if a malicious link or image comes from a friend, out of trust and curiosity, the victim will click that link leading to a malicious page. SMS containing urgent requests for disaster relief funds are common on the phone. Some messages may also consist of travel packages for a 'free of cost' vacation; schemes to lower your credit card interest rates etc.

Following are the major effects of SMS spams:

- Fills your inbox with unwanted messages
- Slow down the phone performance by utilizing phone's memory space
- Tricks people to disclose their personal information
- May charge the user for receiving text messages
- Malicious URLs may install malwares in the phone
- Identity theft
- Consumption of useful time
- Reduce effectiveness of legitimate advertising

Some of the popular anti-spam solutions include blacklisting, white listing, grey listing, real time black hole listing, content based filtering and challenge response. But it will be less effective when we use a single method. We propose an anti-spam technique where we combine different methods – detection based on phone numbers and URLs, machine learning approach, text based filtering and challenge response method, in order to improve the performance of the application.

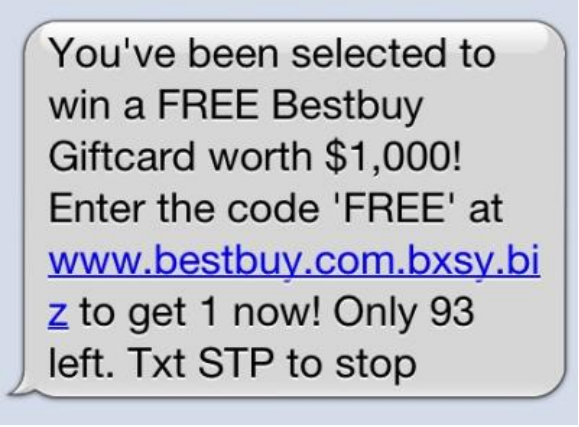

Fig. 1. **Spam SMS Example**

This paper is organized as follows: section 2 includes the work done as part of the literature survey. Section 3 describes the hybrid framework to solve the problem. Mitigation of spam SMS is discussed in section 4. Details about SMS dataset

collection is given in section 5. Section 6 includes the findings on the proposed methodology's performance and test results in Android platform. Finally, the paper concludes in Section 7 and the future scope of our work is presented in Section 8.

II. RELATED WORK

Various spam filtering technologies have been proposed by people and researches are still going on regarding this topic. A user-centric spam SMS filtering application i.e. SMSAssassin is proposed in [1]. It uses content based machine learning techniques with user generated features. Initial version of SMSAssassin was implemented in android and Symbian phones. Based on the feedback a new version was designed in this paper. Machine learning technique used is Bayesian filtering and the user generated features include blacklisted numbers, words etc. A two-level stacked classifier for short text messages is demonstrated in [2]. Two versions of the two-level stacked classifier are implemented: (i) using Bayesian classifiers at both levels, and (ii) using Bayesian classifier in first level and Support vector machine in the second. The algorithm FIMESS in [3] performs simple, yet effective checks on the message headers so as to classify an sms as spam or not. It involves comparing smsc numbers, matching timezone of smsc with that of mobile phone, blacklisting smsc numbers, non-numerical senders ID, keywords, urls etc.

Various spam filtering algorithms like Naive bayes, SVM, K-nearest neighbor, random forest, adaboost were used in [4]. The best classifier is the one utilizing SVM as the learning algorithm. Next best classifier in their work is boosted naive Bayes. Methods used in [5] are blacklisted numbers and words along with classification based on spam score (if spam score is greater than a threshold, SMS is considered as spam). Experimental results presented in this paper are based on iPhone Operating System (iOS). An anti-spam framework based on the hybrid of content-based filtering and challenge-response is proposed in [6]. Content based filtering is used to minimize the extra traffic needed for challenge response and thereby reduce the expense. It also reduces the number of SMSes subject to challenge response. Two solutions including a general filter at server level and user specific filter at mobile level were proposed in [7]. Server level filter makes use of a neural network and this filter can be mainly used by feature phone users. Mobile level filter makes use of Bayesian filtering and black lists for the purpose of classification and this can benefit smart phone users.

Content based filtering is proposed in [8] which make use of SMS specific features as well as Linguistic Inquiry and Word Count (LIWC) features. Various machine learning algorithms were applied and the best performance was given by SVM. The work proposed in [21] make use of Bayesian algorithm which calculates Bayesian score based on the spam keywords in SMSes and classification is done based on this score. They also tried SVM and observed an improvement in spam classification accuracy. Various machine learning algorithms were used for SMS spam filtering in [9]. Out of all algorithms, support vector machines showed the best performance. The work in [19] concentrates on creating a dataset without any near-duplicates. The paper evaluates many existing machine learning algorithms and concludes that SVM outperforms other evaluated methods. Online classification which involves indexing part and online filtering part is proposed in [14]. Incoming messages are represented as a set of tokens and by searching these tokens in its index pair, a spamminess score is calculated. Based on spamminess score SMS is classified as spam/ham.

According to the Bayesian classifier used in [12], while training, a hash-map of words associated with their counts is created and based on the frequency of occurrence of words in spam and genuine messages, spam probability of each word is calculated. If overall spam probability is more than 0.5 then an SMS is classified as spam. The author of [10] proposes a method by combining blacklist and Bayesian Algorithm. A probabilistic Bayes classifier using word occurrences is applied in [11]. Support vector machine is proposed in [16] to filter spam messages. A comparison of SVM and Bayes with input varying from 10-150 messages is performed in [18]. The work in [13] analyses the impact of various feature extraction and selection methodologies on SMS spam filtering. It concludes that the combination of BoW (Bag of words) and structural features contributes to better performance, rather than using BoW features alone. Different feature representation techniques are mentioned in [15] and their experiments reveal that it is better to use binary features with SVM machines. The work in [17] deals involves evaluation of many existing machine learning algorithms and they observe that SVM outperforms other evaluated methods. A literature review on different approaches available for SMS spam filtering is presented in [20]. They observed that supervised learning algorithms can be effective for SMS spam classification.

III. HYBRID FRAMEWORK

We propose a hybrid SMS spam detection framework for Android, which combines the following methods:

1. Number based filtering
2. Detection based on URLs
3. Text based filtering
4. Machine learning approach
5. Challenge response method

Features used while implementing the proposed methodology are mentioned in table I and the framework for the proposed methodology is shown in Fig. 2.

A. Number based filtering

Number based filtering is based on blacklists, unknown numbers and non-numerical numbers. Blacklist refers to a list of invalid entries. We make use of blacklist of numbers which contain those phone numbers that have been previously used to send spam. If sender's number is present in the blacklist, the SMS is identified as a spam. We also identify SMS from unknown numbers and non-numerical numbers as spam SMS.

B. Detection based on URLs

A set of blacklisted URLs were downloaded from http://urlblacklist.com [23]. This data set consists of URLs from many different categories like URLs related to advertisements, adult content, entertainment, porn, malwares, phishing etc. If the sender's number is present in the blacklist

or if any URL extracted from the SMS is present in the blacklist, it is considered as a spam.

C. Text based filtering

In this method, we classify the SMS as spam/ham based on the following spam specific features present in the SMS :

- Presence of common spam phrases (eg: click the link, you are awarded, free call, subscribe today itself)
- Length of SMS : Average length of spam SMS was calculated from a set of spam SMSes in the dataset and all SMSes whose length was greater than or equal to the average length, were considered as spam.
- Occurrence of uppercase words

If a message contains any of the above features, it is considered as spam.

D. Machine learning approach

Machine learning [24], a branch of artificial intelligence, deals with the construction and study of systems that can learn from data. It is mainly of three types:

1) Unsupervised learning

- Training based on unlabelled examples
- Artificial neural network is an example

2) Supervised learning

- Training based on labeled examples
- Support vector machine(SVM) and Bayesian network are examples

3) Semi-supervised learning

- Combines both labeled and unlabelled examples

We make use of support vector machines (SVM) [24] for spam detection. It is a type of supervised machine learning method which deals with inferring a function from labeled training data. Training data consists of a set of training examples where each example is represented as a pair of input (feature vector) and desired output (label or class). Algorithm analyses training data and produces an inferred function which can be used for mapping new examples.

Feature vector is a set of features describing the input object. There will be a feature vector representing each SMS in the dataset. A collection of these feature vectors is termed as feature matrix. Each feature vector will be of the form

< value 1 value 2..... value 'n' class >

where value 1,value 2..... value 'n' corresponds to the values of feature 1, feature 2..... feature 'n' respectively and class corresponds to spam/ham. Following SMS spam features were used while implementing SVM:

- Non-alphanumeric characters
- Spam keywords
- Numeric strings

TABLE I. FEATURES USED

Spam SMS Features
Unknown numbers
Non-numerical numbers (eg: BT-064122)
Blacklisted phone numbers
Blacklisted URLs (eg: www.wintricks.it/image/banner)
Non alphanumeric characters (eg: !, #, @)
Spam keywords (eg: Congrats, winner, awarded)
Numeric strings (eg: 9765674678, 76566)
Length of SMS
Uppercase characters
Spam phrases
Human verification mechanism (eg: Captcha)

E. Challenge response method

Challenge response is based on the response of the sender to the challenge sent by the recipient of the SMS. It is used to verify whether the sender of an SMS is human or not. It can be described as follows: when a message is received, the recipient sends back a code to the sender and waits for a response from him. If a response is received, we can ensure that the sender is a genuine user and the message is classified as ham else it will be classified as spam. Spam generation is usually automated, so we won't get a reply back from the spammers. But genuine users will most probably respond to the challenge.

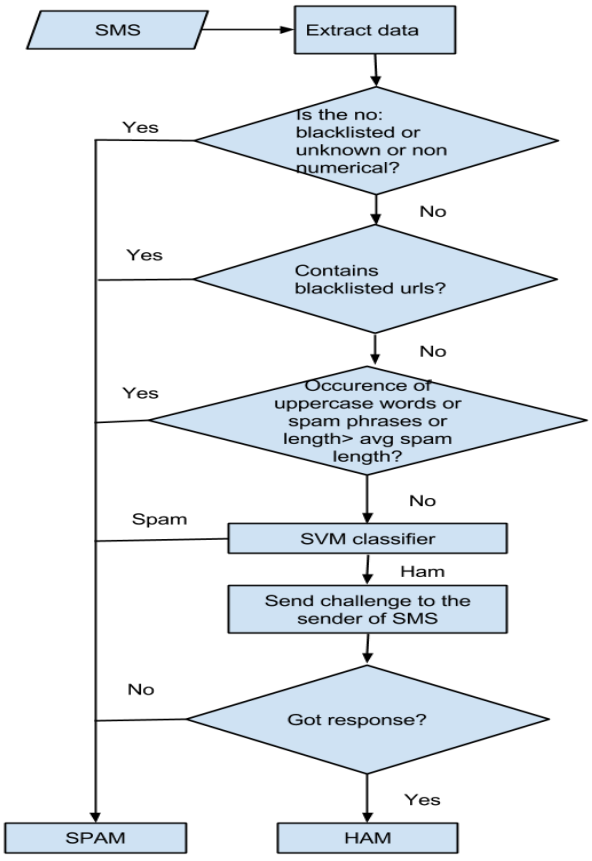

Fig. 2. **Hybrid Framework**

IV. MITIGATION OF SPAM SMS

After detecting the spam SMSes using the above methods, the sender of all those SMSes will be blacklisted ie. the number will be entered into the blacklist of numbers. This way, the blacklist of numbers get extended each time the application is run. The extended blacklist will be helpful in detecting spam in future more accurately.

V. COLLECTION OF DATASET

A good SMS dataset is always necessary for training and testing various spam filtering systems. Dataset used for our experiment is SMS Spam Collection v.1 which is available at http://www.dt.fee.unicamp.br/~tiago/smsspamcollection [22]. Fig. 3 shows a set of spam /ham examples from SMS Spam Collection v.1.

ham Thanks a lot for your wishes on my birthday. Thanks to you for making my birthday truly memorable.
ham Is that seriously how you spell his name?
spam Please call our customer service representative on 0800 169 6031 between 10am-9pm as you have WON a guaranteed £1000 cash or £5000 prize!
ham Lol your always so convincing.
ham I call you later, don't have network. If urgnt, sms me.
spam Your free ringtone is waiting to be collected. Simply text the password "MIX" to 85069 to verify. Get Usher and Britney. FML, PO Box 5249, MK17 92H. 450Ppw 16

Fig. 3. **Examples of SMSes in Spam Collection v.1**

VI. TEST RESULTS

The framework was implemented in Android platform and was tested using an Android device. For testing purpose, we populated the inbox with 100 SMSes (by sending messages from other phones). Performance of the proposed methodology was evaluated based on the confusion matrix which is shown in Table II. Confusion matrix [25] refers to a table which shows the number of true positives, true negatives, false positives and false negatives.

TABLE II. CONFUSION MATRIX

		Predicted	
		Spam	Non-Spam
Actual	Spam	a	b
	Non-Spam	c	d

a-Number of True Positives
b-Number of False Negatives
c-Number of False Positives
d-Number of True negatives

Based on the confusion matrix, we measured the precision rate, recall rate and overall accuracy [25] of the proposed methodology.

- Precision rate (also known as specificity) refers to the proportion of the predicted positive cases that were correct.
- Recall rate (also known as sensitivity) refers to the proportion of positive cases that were correctly identified.

The formulas to calculate above values are given in Table III:

TABLE III. PERFORMANCE MEASURES

Precision rate = a / (a+c)
Recall rate = a / (a+b)
Accuracy = (a+d) / (a+b+c+d)

The observed results for our experiment are shown in Table IV and Table V.

TABLE IV. PERFORMANCE MEASURES OF ANDROID APPLICATION

Precision rate	Recall rate	Overall accuracy
98%	100%	99%

The accuracy of Android application was calculated in two stages of our implementation, after implementing first three methodologies and at the end. The results are represented as a graph shown in Fig. 4.

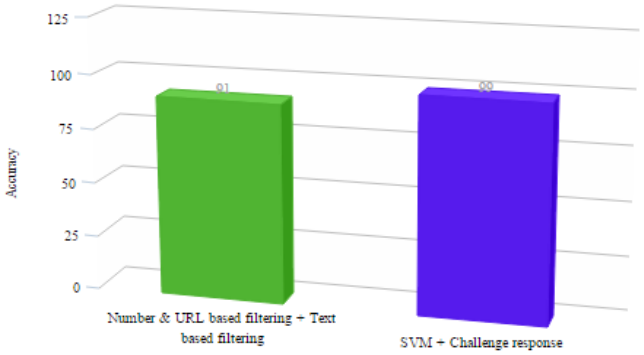

Fig. 4. **Accuracy of Android Application**

VII. CONCLUSION

With the increased popularity of mobile phones, there has been a rapid growth in the number of junk messages received by the mobile phone users. As a solution to this problem, we have proposed a hybrid framework for SMS spam detection in Android phones. The framework uses variety of spam features distributed among different spam filtering methodologies – detection based on phone numbers and URLs, machine learning approach, text based filtering and challenge response method. We have evaluated and observed a good

improvement in spam classification using the hybrid approach.

VIII. FUTURE WORK

As part of future work, the Android application can be added with more features like detecting blacklisted short URLs, deleting spam SMS and considering SMS with images.

URL shortening refers to the technique of converting a long URL to a shorter one which will be more convenient to use and read. URL shortening services are also being utilized by spammers for redirecting the users to malicious websites or for some other illegal internet activities. Since the number of characters in an SMS is limited, spammers mostly use short URLs instead of the long ones.

In most cases, the users get affected by opening the spam SMS in their inbox. So deletion of spam SMS from the inbox is another feature which can be added as part of mitigation of spam SMS. This will reduce the effect of spam since the SMS will be deleted from the inbox, once detected as spam and hence the users won't be able to open the SMS.

Currently we have considered only text messages for our experiment and we would like to consider messages containing images i.e. Multimedia messaging service (MMS) in future.

REFERENCES

[1] Kuldeep Yadav, Swetank K. Saha, Ponnurangam Kumaraguru and Rohit Kumra, "Take Control of Your SMSes : Designing an Usable Spam SMS Filtering System", *IEEE 13th International Conference on Mobile Data Management*, July 2012

[2] Akshay Narayan and Prateek Saxena, "The Curse of 140 Characters: Evaluating the Efficacy of SMS Spam Detection on Android", *Proceedings of the Third ACM workshop on Security and privacy in smartphones & mobile devices*, 2013

[3] Iosif Androulidakis, Vasileios Vlachos and Alexandros Papanikolaou, "FIMESS: FIltering Mobile External SMS Spam", *Proceedings of the 6th Balkan Conference in Informatics*, May 2013

[4] Houshmand Shirani-Mehr, "SMS Spam Detection using Machine Learning Approach", 2012

[5] Tarek M Mahmoud and Ahmed M Mahfouz, "SMS Spam Filtering Technique Based on Artificial Immune System", *Publications of International Journal for Computer Science Issues (IJCSI)*, March 2012

[6] Ji Won Yoon, Hyoungshick Kim and Jun Ho Huh, "Hybrid spam filtering for mobile communication", *Publications of Computers and Security Journal*, August 2009

[7] T. Charninda', T.T. Dayaratne', H.K.N. Amarasinghc' and J.M.R.S. Jayakody, "Content based Hybrid SMS Spam Filtering System ", 2014

[8] Amir Karami and Lina Zhou, "Improving Static SMS Spam Detection by Using New Content-based Features", *Twentieth Americas Conference on Information Systems, Savannah*, 2014

[9] José María Gómez Hidalgo, Guillermo Cajigas Bringas and Enrique Puertas Sánz, "Content Based SMS Spam Filtering", *Proceedings of the 2006 ACM symposium on Document engineering*, 2006

[10] Yin Li, Mingyu Fan and Guangwei Wang, "Research on Spam SMS Detection and Prevention on Android Platform", *Hans Journal of Data Mining*, April 2012

[11] M. Taufiq Nuruzzaman, Changmoo Lee, Mohd. Fikri Azli bin Abdullah and Deokjai Choi, "Simple SMS spam filtering on independent mobile phone", *International Journal of Security and Communication Networks*, 2012

[12] Gaurav Sethi and Vijender Bhootna, "SMS Spam Filtering Application Using Android", *International Journal of Computer Science and Information Technologies*, 2014

[13] A. K. Uysal, S. Gunal, S. Ergin and E. Sora Gunal, "The Impact of Feature Extraction and Selection on SMS Spam Filtering", *Elektronika ir Elektrotechnika (Electronics and Electrical Engineering)*, 2013

[14] Wuying Liu and Ting Wang, "Index-based Online Text Classification for SMS Spam Filtering", *Journal of Computers*, June 2010

[15] Harris Drucker, Donghui Wu and Vladimir N. Vapnik, "Support Vector Machines for Spam Categorization", *IEEE transactions on neural networks*, September 1999

Xiang, Yang, Chowdhury, Morshed and Ali Shawkat 2004, "Filtering mobile spam by support vector machine", *Third International Conference on Computer Sciences, Software Engineering, Information Technology, E-Business and Applications*, December 2004

[16] Tiago A. Almeida, José María Gómez and Akebo Yamakami, "Contributions to the Study of SMS Spam Filtering: New Collection and Results", *Proceedings of the 11th ACM symposium on Document engineering*, 2011

[17] Tej Bahadur Shahi and Abhimanu Yadav, "Mobile SMS Spam Filtering for Nepali Text Using Bayesian and Support Vector Machine", *International Journal of Intelligence Science*, 2014

[18] Tiago A. Almeida, José María Gómez Hidalgo, Tiago P. Silva, "Towards SMS Spam Filtering: Results under a New Dataset", *International journal of Information Security Science*, 2013

[19] Sarah Jane Delany, Mark Buckley and Derek Greene, "SMS Spam Filtering: Methods and Data", *International Journal of Expert Systems with Applications*, August 2012

[20] Kuldeep Yadav, Ponnurangam Kumaraguru, Atul Goyal, Ashish Gupta and Vinayak Naik, "SMSAssassin: Crowdsourcing Driven Mobile-based System for SMS Spam Filtering", *Proceedings of the 12th Workshop on Mobile Computing Systems and Applications*, March 2011

[21] http://www.dt.fee.unicamp.br/~tiago/smsspamcollection

[22] http://urlblacklist.com
[23] K. P. Soman, Shyam Diwakar, V. Ajay (2006), "Insight into Data Mining Theory and Practice", New Delhi, Prentice-Hall of India Private Limited
[24] http://www2.cs.uregina.ca/~dbd/cs831/notes/confusion_matrix/confusion_matrix.html