Proceedings of the International Conference on Intelligent Computing and Control Systems (ICICCS 2019)
IEEE Xplore Part Number: CFP19K34-ART; ISBN: 978-1-5386-8113-8

Identifying Spam Patterns in SMS using Genetic Programming Approach

Dimple Sharma
Department of Computer Science & Engineering
National Institute of Technology, Raipur
Raipur, India
dimpleryp@gmail.com

Aakanksha Sharaff
Department of Computer Science & Engineering
National Institute of Technology, Raipur
Raipur, India
asharaff.cs@nitrr.ac.in

Abstract*—SMS spam, also known as mobile spam, has become a prevalent and an ever growing issue due to the availability of bulk SMS services at nominal costs. These spam messages may not only be commercial but also pose a great deal of financial threats to the users. To fight against SMS spam, a variety of solutions have been proposed including content-based filtering, semantic indexing, machine learning classifiers, etc. However, in this regard evolutionary algorithms have not been utilized. Since the nature of SMS is contemporary, the representation of text messages keep evolving with the help of slangs, symbols, misspelled words, abbreviations and acronyms. Hence, such a solution is required which can accommodate these changes, also keeping the length of SMS in consideration. The model proposed in this paper generates regular expressions as individuals of population, using Genetic Programming Approach. These regular expressions so generated are used for the classification purpose. The application of Genetic Programming in the domain of SMS spam filtering has not been explored widely. It is able to eliminate False Positive errors, thus saving legitimate messages from being misclassified. The performance tends to improve with higher number of generations. Performance and confusion matrix for different number of generations are tabulated.

Keywords*—SMS spam filtering; regular expressions; evolutionary algorithm; genetic programming algorithm; spam classification.

I. INTRODUCTION

SMS spam refers to unwanted text messages sent to mobile phones through Short Message Service, mostly for commercial purposes, such as, for promoting products and services. It also includes the attempts to scam users into providing personal information by asking the users to respond to some e-mail addresses or phone numbers that do not seem to correspond with the supposed identity of the sender. The scammers may pose as any legitimate body to gain the trust of users, like financial institutions, government agencies, delivery services, lottery organizers, and so on. Messages that do not identify the senders and messages to which users have not given consent to receiving are likely the cases of SMS spam.

The battle against SMS spam is necessary as these spam messages can also lead to cause financial losses to the mobile subscribers or users. It is easy to fall prey as SMS spam can take multiple forms, such as, a simple message, a link to a number for calling or texting purposes, a link to a website for getting more information or a link to a website for downloading a mobile application. By responding to these, even though mistakenly, the users can wind up signing expensive subscription services. There is a greater risk of malware downloads and phishing attacks as most of the users are unaware of the ways to tackle them [1].

Mobile phones have gained significant popularity since the beginning of the 21st century. With the ever increasing number of mobile users, SMS has joined the league of becoming one of the most important communication medium. Not only the availability of bulk SMS services at nominal costs but SMS being a trusted service, delivering higher response rates has gained the attraction of attackers, thereby increasing the number of spam messages rapidly. Simple filtering methods are not enough to fight SMS spam as the spammers workaround traffic analysis, identifying volume limits. Hence, content-based filtering is required. In recent times, SMS spam detection has gotten more attention of researchers who are employing various machine learning approaches for the same. The task of SMS spam filtering is relatively new. It not only inherits solutions from e-mail spam filtering but also inherits many issues in addition to having its own specific challenges.

In this paper, the proposed model uses an evolutionary algorithm to generate regular expressions from a small part of dataset and classifies the remaining text in the corpus using the so generated expressions. The dataset being used is the SMS Spam Collection Data Set provided by UCI Machine Learning Repository. The implementation of the proposed model is adapted from the work in [2] in which evolutionary computations are used for discovering spam patterns in e-mails. Its fitness function yields 0 False Positive results, ensuring that legitimate messages are not misclassified as spam. Thus, preventing loss of crucial and urgent information. This is further utilized for SMS spam detection with modifications in parameters including computing maximum population, specifying number of generations. The results achieved by the regular expressions generated by the proposed model for different number of generations are tabulated and

Authorized licensed use limited to: Wikipedia. Downloaded on June 22,2024 at 14:27:58 UTC from IEEE Xplore. Restrictions apply.

their performances are compared. Confusion matrix is also drawn for different generations. Due to absence of False Positive errors, the specificity is 100% in each of the cases. As each generation produces better set of individuals in population (i.e. regular expressions), performance improves with greater number of generations, yielding better accuracy and sensitivity. In the context of SMS spam filtering, genetic programming approach has not been utilized much.

II. Literature Review

Over the years, many content based spam filtering methods for SMS have been introduced. Gómez Hidalgo et al. made use of lexical features. Information Gain was applied for feature selection followed by using learning-based classifiers [3]. With the view of brevity in SMS, many studies have laid emphasis on expanding feature sets. One of such works is of Cormack et al. in which features were engineered additionally from the content of messages [4]. Later, Sohn et al. utilized combination of lexical and stylistic features that led to improvement in the results achieved by using content based classifiers [5].

Considerable work has been done in the area of e-mail spam detection. Efforts have been made to apply many of the algorithms used for e-mail spam filtering on SMS spam detection, however, these algorithms underperformed on SMS spam. The reasons that account for this behavior include messages having limited features, the length of messages being short, informal language used in the messages and absence of real database for SMS spam. Therefore, a real database of SMS spam was created in 2011 by UCI Machine Learning Repository which is also the corpus used in this paper. After being created, the database was made publicly available. This opened opportunities for many experiments to be performed on the dataset. Two tokenizers were used without stop words removal or stemming methods. The same year Almeida et al. proposed the SMS Spam Collection and used it for performing comparisons among the performances of different machine learning algorithms [6, 7]. The results indicated that linear SVM performs the best. Besides these, Xu et al. provided a feature-based classification technique that was utilized for performing SMS classification [8]. Ahmed et al. proposed a semi-supervised method, making use of frequent item-set and ensemble learning, in order to detect SMS spam. The learners used for ensemble in their experiment included random forest, multinomial naïve Bayes, and support vector machines (SVM) [9]. Almeida et al. evaluated a text pre-processing technique to automatically normalize and expand short message having slangs, symbols, misspelled words, abbreviations, acronyms, all of which are part of common text representation. Attributes are acquired on expanded message, yielding enhanced classification performance [10]. Silva et al. introduced a text categorization technique to filter short text messages based on minimum description length principle. Their hybrid ensemble approach utilizes NLP processing techniques of semantic indexing and text normalization to improve quality of text content [11].

The first attempt to incorporate genetic programming for generating regular expressions to perform e-mail spam filtering was done by Greenstadt and Kaminsky [12]. In their approach, after completion of breeding process which incorporated crossover and mutation, rest of the positions in the new population were filled by the best individuals taken from the previous generation. The main limitations of this approach included the creation of the first set of population using a blind method and using a fitness function which was not focused on removal of FP errors. Fitness function determines quality of individuals so as to decide whether to include them in population or not. These limitations were addressed by Conrad [13]. However, there were repetitive terms in regular expressions, leading to excessive growth of population vector. These challenges were overcome in the model proposed by Ruano-Ordás et al. along with an improved fitness function. The function gives preference to the regular expressions which are shorter in length and matches more number of spam messages.

The nature of text messages is contemporary. With time, different slangs, acronyms, abbreviations, symbols, etc. evolve for representing text in SMS. Spammers keep changing the representation of spam messages intentionally in order to avoid identification. To keep up with these, our proposed model generates regular expressions via an evolutionary algorithm. It identifies all legitimate (ham) messages correctly, resulting in zero False Positive errors. This is one of the most desirable features of the model as it eliminates the chances of important, urgent or crucial ham messages from being misclassified. In addition, the proposed model can work with small datasets as well as large datasets. Increasing the number of generations in the model enhances classification result. The utilization of genetic programming algorithm for SMS spam filtering is an unexplored subject.

III. Methodology

A. Corpus Selection

The corpus used in the model is the SMS Spam Collection Data Set provided by UCI Machine Learning Repository. It is a standard dataset used for SMS spam classification. Each SMS in the dataset is marked as “spam” (SPAM) or “ham” (not SPAM), written at the beginning of SMS.

The sources of SMS for this dataset include Grumbletext Website, NUS SMS Corpus (NSC), Caroline Tag's PhD Thesis, SMS Spam Corpus v.0.1 Big. Altogether, there are 4827 HAM messages and 747 SPAM messages in the dataset.

B. Data Pre-processing

The SMS Spam Collection Data Set is a single text file wherein each line represents a text message i.e. SMS. At the beginning of each of these text messages, a label is given as “spam” or “ham”. Firstly, the spam messages and ham messages were separated into two different text files. Then the labels were removed so as to refrain the words “spam” and “ham” from participating in regular expressions. 250 spam messages and 296 ham messages were chosen at random and saved in two separate files. These would be used as input for generating regular expressions. The remaining 497 spam messages and 4531 ham messages are classified with the help of regular expressions generated from a small part of the dataset, as specified above.

Authorized licensed use limited to: Wikipedia. Downloaded on June 22,2024 at 14:27:58 UTC from IEEE Xplore. Restrictions apply.

C. Evolutionary Algorithm

Evolutionary Algorithm is a generic population-based meta-heuristic optimization algorithm. The evolutionary algorithms are used for solving problems that cannot be easily solved in polynomial time. The premise of an evolutionary algorithm is inspired by biological evolution. It may include recombination, reproduction, selection, and mutation. Solutions, which are members of the set of possible solutions in the feasible region of a given problem, tend to play the role of individuals in a population. To determine the quality of the solutions, i.e., the individuals, fitness function is used. It allows to decide whether to include an individual in the population or not. Hence, fitness function is a cost function that determines the solutions to be retained. Evolution of the population takes place by repeatedly applying operators like mutation, etc. After repeated application of these operations, the solutions (individuals of the population) tend improve over time. In an evolutionary algorithm, fitter members will survive and proliferate, while unfit members will die, thus, not contributing to the gene pool of further generations.

D. Genetic Programming Algorithm

Genetic Algorithm is a class of Evolutionary Algorithm. It is biologically inspired computation. Genetic algorithms use mutation as well as crossover for searching the possible solution space. They can be used for discovering a functional relationship between features in data (regular regression in our case) and to group data into classes or categories (classification such as spam, ham). Being inspired by biological evolution, genetic algorithm incorporates crossover, random mutation, a fitness function and multiple generations of evolution for resolving a user-defined task. In the proposed model, genetic programming algorithm is used for generating population of individuals, where the individuals correspond to regular expressions. The generated regular expressions are then used for the purpose of classification.

E. Proposed Model

The process of SMS spam detection is done by using regular expressions. Regular expressions are sequence of characters representing search patterns. The regular expressions are generated from a small set of legitimate ham and spam messages. These expressions are then used on rest of the dataset to classify the text messages. The generation of regular expressions is performed with the help of genetic programming algorithm. It is carried out in four folds, viz., loading corpus & initialization, generating individuals, breeding population, removing surplus population. Regular expressions are stored in doubly linked list data-structure which corresponds to chromosomes in the language of Evolutionary Algorithm.

The fitness function for regular expression, used in the model is given by (Eq. 1).

$$\text{fitness}(i) = \text{matches}(i,\text{spam}) \times ((10/(\text{length}(i)) + 1) \quad (1)$$

This fitness function benefits the regular expressions which are shorter in length and matches a great amount of spam messages. It also takes care of eliminating False Positive errors which is one of the prime focuses of the model.

The first phase is responsible for initialization activities. It includes loading and splitting of the corpus into lines as well as initialization of the structures for managing the population to be generated.

In the next phase, the task of generating individuals of the population is carried out. This is done by making regular expressions with respect to each spam message, followed by computing fitness of the generated expression. Chromosomes are created for the expressions which are found fit. The chromosomes thus created are added to population.

The third phase carries out population breeding which is responsible for adding new individuals to the population. Roulette wheel selection, also called fitness proportionate selection, is used for picking potentially useful individuals for the purpose of breeding. For making children, crossover or mutation is performed. Crossover operators include OR and CAT (concatenate). Mutation operators include ARWG (Add Random Wildcard Genes i.e. .*) and DRG (Delete Random Genes). After breeding, fitness of children is computed. Fit children are added to the population and the population is updated.

The fourth phase takes care of keeping the population size under control so as to ensure optimal usage of memory. This is done by getting population size and checking it against limit of population size. If limit is exceeded then worst individuals are dropped from the population. Figure 1 shows the workflow of the proposed model as discussed above.

The process of population breeding and removal of surplus population occurs for a specified number of times. This number can be referred to as number of generations. With each generation, the population tends to improve.

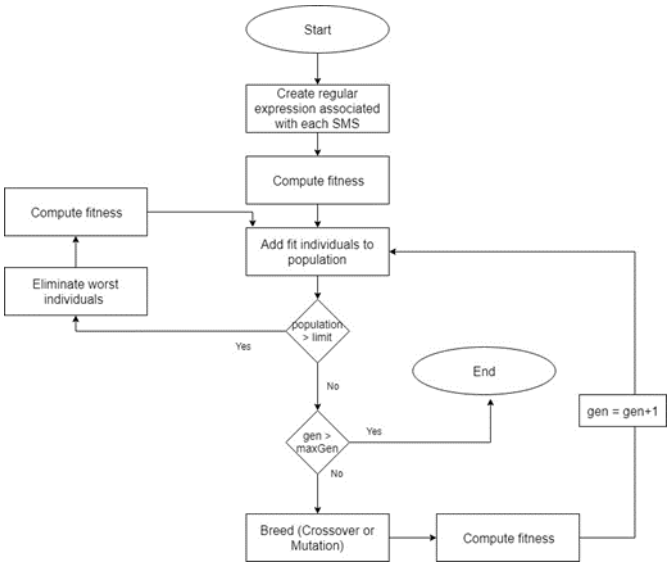

Fig. 1. Workflow diagram for SMS spam detection using the proposed model.

IV. Experimental Results

A part of dataset comprising of 250 spam messages and 296 ham messages is fed as input to the proposed model. The

Authorized licensed use limited to: Wikipedia. Downloaded on June 22,2024 at 14:27:58 UTC from IEEE Xplore. Restrictions apply.

model generates regular expressions using which the remaining data is classified as spam or ham. The computations have been performed on a computer with system configuration of intel i7 processor with 16GB of random access memory.

The proposed model is evaluated for different number of generations. From the confusion matrix of different generations shown below, it is evident that the classification results show an improving trend with consequent generations as with each generation the population tends to improve. However, as the number of generations increase, the computations increase, taking much more time for execution.

Another noteworthy observation is the complete elimination of False Positive errors. This is one of the aspects that make the proposed model desirable. FP errors are those in which a ham message is classified as spam. Needless to say how this misclassification can be hazardous and cause great loss to anyone. Hence, the proposed model is an effective technique to avoid the misclassification of legitimate text messages. Due to this very reason, the true negative rate, also referred to as specificity, is always 100%.

A dummy table for confusion matrix is created so as to have clear understanding of the terms used in evaluation of model. This table is followed by confusion matrix showing results obtained for different number of generations.

TABLE I. DUMMY CONFUSION MATRIX

Actual / Predicted	HAM (-ve)	SPAM (+ve)
HAM (-ve)	TN	FP
SPAM (+ve)	FN	TP

TABLE II. CONFUSION MATRIX FOR 30 GENERATIONS

Actual / Predicted	HAM	SPAM
HAM	4531	0
SPAM	424	73

TABLE III. CONFUSION MATRIX FOR 60 GENERATIONS

Actual / Predicted	HAM	SPAM
HAM	4531	0
SPAM	414	83

TABLE IV. CONFUSION MATRIX FOR 100 GENERATIONS

Actual / Predicted	HAM	SPAM
HAM	4531	0
SPAM	350	147

TABLE V. CONFUSION MATRIX FOR 150 GENERATIONS

Actual / Predicted	HAM	SPAM
HAM	4531	0
SPAM	321	176

The performance comparison table shown below indicates improvement in sensitivity (true positive rate) and accuracy achieved with the proposed model with greater number of generations. This is because each new generation brings about improved set of individuals in the population, which are regular expressions in our case.

The formulae used for calculating percentage of accuracy, sensitivity and specificity are specified below (Eq. 2, Eq. 3, Eq. 4).

$$\text{Accuracy} = ((TP + TN)/(P + N)) \times 100 \quad (2)$$

$$\text{Sensitivity} = (TP/(TP + FN)) \times 100 \quad (3)$$

$$\text{Specificity} = (TN/(TN + FP)) \times 100 \quad (4)$$

With reference to above equations, following are the meanings of the variables used.

P = total positive cases, i.e., total number of spam messages (P is also given by, P = FN + TP) = 4531

N = total negative cases, i.e., total number of ham messages (N is also given by, N = TN + FP) = 497

TP = number of spam message identified as spam (true positive)

TN = number of ham message identified as ham (true negative)

FP = number of ham message identified as spam (false positive)

FN = number of spam message identified as ham (false negative).

Following table records performance of the proposed model for various numbers of generations.

Authorized licensed use limited to: Wikipedia. Downloaded on June 22,2024 at 14:27:58 UTC from IEEE Xplore. Restrictions apply.

TABLE VI. PERFORMANCE COMPARISON FOR DIFFERENT GENERATIONS

Number of Generations	Accuracy	Sensitivity	Specificity
30	79.75	9.77	100
60	79.92	11.11	100
100	81.03	19.6	100
150	81.54	23.56	100

The commonality that can be observed from the above confusion matrix for different number of generations is that all 4531 HAM messages are identified correctly. This points to the fact that no legitimate message is incorrectly classified as spam and hence, False Positive error is zero. Therefore, this accounts for 100% correct classification of HAM messages, leading to 100% specificity (true negative rate). However, with earlier generations of 30 and 60, the True Positive rates are 9.77% and 11.11% respectively which are comparatively lower than the results achieved by greater number of generations. True Positive rate increases to 19.6% with 100 generations and escalates even further to 23.56% with 150 generations. As a result, this further contributes to enhancing the accuracy of the model from 79.75% for 30 generations to 81.54% for 150 generations. This is the outcome of better and more suitable regular expressions being created with each generation via genetic programming algorithm. Hence, the above results highlight improvement in the performance of the proposed model with higher number of generations to carry out spam filtering in SMS.

V. CONCLUSIONS AND FUTURE WORK

SMS spam has become a ubiquitous since the past decade. With the advent of many schemes serving SMS at minimal cost by various mobile network operators, the number of spam SMS received on a daily basis has increased multiple folds. To deal with this situation, the proposed model uses genetic programming approach to produce regular expressions (from a few SMS of dataset) that are later used in spam filtering. Genetic programming has not been utilized substantially in the sphere of SMS spam filtering. The biggest advantage offered by the proposed model is the elimination of False Positive errors. That is, the model ensures that no legitimate text messages are misclassified as spam. This makes the specificity 100% in all the cases. Higher number of generations offers promising results, showing an improving trend in accuracy and sensitivity. As only a very small set of corpus is used for generating regular expressions, the model can be applied over any sized dataset, be it large or small. Since the model gives zero False Positive errors, it can be used as pre-filtering mechanism to enhance the results of standard machine learning classifiers. Therefore, adapting this can be seen as future work.

REFERENCES

[1] Sarah Jane Delany, Mark Buckley, and Derek Greene. "SMS spam filtering: Methods and data" Expert Systems with Applications 39 (2012) 9899-9908.

[2] Ruano-Ordás, David, Florentino Fdez-Riverola, and José R. Méndez. "Using evolutionary computation for discovering spam patterns from e-mail samples." Information Processing & Management 54 (2018) 303-317.

[3] J.M. Gómez Hidalgo, G.C. Bringas, E.P. Sánz, F.C. García. "Content based SMS spam filtering." Proceedings of the ACM Symposium on Document Engineering (2006), pp. 107–114.

[4] G.V. Cormack, J.M. Gómez Hidalgo, E.P. Sánz. "Feature engineering for mobile (sms) spam filtering." Proceedings of 30th Annual International ACM SIGIR Conf. Research and Development on Information Retrieval (2007), pp. 871–872.

[5] D.N. Sohn, J.T. Lee, K.S. Han, H.C. Rim. "Content-based mobile spam classification using stylistically motivated features." Pattern Recognition Letters (2012), 33(3): 364–369.

[6] T.A. Almeida, J.M. Gómez Hidalgo, A. Yamakami. "Contributions to the study of SMS spam filtering: new collection and results." Proceedings of the Eleventh ACM DOCENG, Mountain View, California, USA (2011), pp. 259–262.

[7] Shirani-Mehr, H. "SMS spam detection using machine learning approach." CS229 Project 2013, Stanford University, USA, pp. 1–4.

[8] Q. Xu, E. Xiang, Q. Yang, J. Du, J. Zhong. "SMS spam detection using non-content features." IEEE Intell. Syst. 27 (6) (2012) 44-51.

[9] Ishtiaq Ahmed, Rahman Ali, et al. "Semi-supervised learning using frequent itemset and ensemble learning for SMS classification" Expert Systems with Applications 42 (2015) 1065-1073.

[10] Tiago A. Almeida, Tiago P. Silva, Igor Santos, José M. Gómez Hidalgo. "Text normalization and semantic indexing to enhance Instant Messaging and SMS spam filtering." Knowledge-Based Systems 108 (2016) 25-32.

[11] Renato M. Silva, Tulio C. Alberto, Tiago A. Almeida, Akebo Yamakami. "Towards filtering undesired short text messages using an online learning approach with semantic indexing". Expert Systems with Applications 83 (2017) 314-325.

[12] Greenstadt, R., & Kaminsky, M. "Evolving spam filters using genetic algorithms." Massachusetts Institute of Technology (2002).

[13] Conrad, E. "Detecting spam with genetic regular expressions." London: SANS Institute InfoSec Reading Room (2007).

Authorized licensed use limited to: Wikipedia. Downloaded on June 22,2024 at 14:27:58 UTC from IEEE Xplore. Restrictions apply.