Machine Learning Research
2016; 1(1): 1-14
http://www.sciencepublishinggroup.com/j/mlr
doi: 10.11648/j.mlr.20160101.11

Review Article

SMS Spam Filtering Using Machine Learning Techniques: A Survey

Hedieh Sajedi[1, *], Golazin Zarghami Parast[1], Fatemeh Akbari[2]

[1]Dept. of Mathematics, Statistics and Computer Science, College of Science, University of Tehran, Tehran, Iran

[2]Dept. of Electrical, Computer and Information Technology, Islamic Azad University, Tehran, Iran

Email address:

hhsajedi@ut.ac.ir (H. Sajedi)

[*]Corresponding author

To cite this article:

Hedieh Sajedi, Golazin Zarghami Parast, Fatemeh Akbari. SMS Spam Filtering Using Machine Learning Techniques: A Survey. *Machine Learning Research*. Vol. 1, No. 1, 2016, pp. 1-14. doi: 10.11648/j.mlr.20160101.11

Received: September 28, 2016; **Accepted**: November 5, 2016; **Published**: December 5, 2016

Abstract: *Objective*: To report a review of various machine learning and hybrid algorithms for detecting SMS spam messages and comparing them according to accuracy criterion. *Data sources:* Original articles written in English found in Sciencedirect.com, Google-scholar.com, Search.com, IEEE explorer, and the ACM library. *Study selection:* Those articles dealing with machine learning and hybrid approaches for SMS spam filtering. *Data extraction:* Many articles extracted by searching a predefined string and the outcome was reviewed by one author and checked by the second. The primary paper was reviewed and edited by the third author. *Results*: A total of 44 articles were selected which were concerned machine learning and hybrid methods for detecting SMS spam messages. 28 methods and algorithms were extracted from these papers and studied and finally 15 algorithms among them have been compared in one table according to their accuracy, strengths, and weaknesses in detecting spam messages of the Tiago dataset of spam message. Actually, among the proposed methods DCA algorithm, the large cellular network method and graph-based KNN are three most accurate in filtering SMS spams of Tiago data set. Moreover, Hybrid methods are discussed in this paper.

Keywords: Spam Filtering, Machine Learning Algorithms, SMS Spam

1. Introduction

By the development of mobile communication technology and the expansion of mobile phones, Short Message Service system or SMS has become one of the most important communication modes according to its simple operation and low price. According to the report provided by Portio Research [1], the worldwide mobile messaging market was worth USD 179.2 billion in 2010, 200 billion in 2011, and maybe will reach USD 300 billion in 2014.

Among all types of short messages, we are going to focus on spam messages. Spam messages have several disadvantages including waste of traffic, storage space and computational power, which lead to financial problems. According to Cloudmark stats[2], the number of mobile phone spams varies widely from region to region. For instance, in North America less than 1% of SMS messages were spam in 2010, while in parts of Asia up to 30% of messages were spam messages. In China and during 2008, the number of daily sent messages was 1.9 billion, and China's mobile phone users received an average of 10.35 spam messages per week[3].

There are many methods, which have been applied for detecting SMS spams. We can divide them into two groups: Content-based approaches and non-Content-based approaches. Social network analysis [1, 2] is a typical non- Content-based approach. This approach is often used by telecom operators instead of mobile phone users. On the other hand, approaches such as automatic text classification techniques, Support Vector Machines (SVMs) [3], K-Nearest Neighbor algorithm [4, 5], logistic regression algorithm [6] and Winnow algorithm [7] are content-based. Among these methods, SVMs are

1 http://www.portioresearch.com/MMF11-15.html
2 http://www.cloudmark.com/en/article/
3 http://www.miit.gov.cn/

considered to be the most suitable one [8]. In recent researches, some evolutionary algorithms such as artificial immune system [9], have also been applied to the problem and researchers have drawn comparison among the performance of these algorithms [10]. Hybrid approaches have also been proposed, which combine Content-based filtering with challenge-response, a technique that sends a reply to the message sender and requires the sender to give a reply. CAPTCHA algorithm [11] is one of those hybrid approaches, which sends an image to the message sender and requires a reply to confirm whether the sender is a robot or not.

In this paper, we provide a structured overview of the existing learning-based approaches for spam filtering. At first, we describe the spam phenomenon and then we introduce some datasets, which have been provided by scientists to be used in detecting spam messages. Afterward, we overview a wide variety of filtering techniques, and pay more attention to evaluation and comparison of different approaches.

The paper is organized as follows: Section 2 is dedicated to researcher method, Section 3 is an overview of SMS spam filtering that includes survey performance measures, describing SMS spam phenomenon, feature extraction and some datasets that are introduced in this regard. The methods of machine learning, which have been used for spam filtering, are overviewed in Section 4. In Section 5 we compare the discussed methods. Finally, Section 6 we make the conclusion.

2. Research Method

In order to study, evaluate and compare various machine learning algorithms for spam filtering we had to systematically find proper publishers in this field to ensure that all proper papers have been studied and algorithms have been chosen preciously. Therefore, searching keywords in this field to discover appropriate papers was very important. We have used a review protocol to find beneficial papers, which is described in Section 2.2. Moreover, we need a standard criterion for comparing in Section 2.4.

2.1. Study Selection

The first phase was to extend the terms, which should be used in searching procedure: the following sub-steps were used during this phase:

(1). Using terms extracted from the sentence ‘‘SMS spam detection techniques”.

(2). Applying synonyms for the terms obtained from the previous step.

(3). Identifying and looking for the keywords in related articles.

2.2. Information Sources

The search was applied to various resources: Sciencedirect.com, Google-scholar.com, Search.com, IEEE explorer, and the ACM library. Next, we evaluated the references of the resulting papers to obtain additional relevant papers. Finally, we reviewed all collected papers.

2.3. Data Collection Process

Articles chosen in searching procedure analyzed by one author (by reading the abstract and some parts of the articles) to check whether they are related to our research subject or not. Then the selected articles were analyzed by the second author so that best articles have been chosen. Ultimately, after the accomplishment of the paper, the third author have revised the linguistic problems, completed the content, and reconstructed the structure according to standard frameworks to ensure that it is written systematically and according to paper publication standards.

2.4. Comparison Criterion

As mentioned above we need a standard criterion to compare the strength of several machine learning algorithms which we have studied. There are numerous criterions, which are used in most of the spam message identification methods including Recall, Precision, and Accuracy to measure their performance. In this paper, we use the Accuracy as most of the investigators have used it to rate their algorithms and hence it helps us to compare algorithms fairly.

3. Overview of SMS Spam Filtering

In the last decade, by the development of mobile communication technology, the number of spam messages that cause problems for users by advertising has been increasing intensively. Hence, researchers have developed various spam detection techniques during last few years to preserve the accuracy of results. Between 2004 and 2015, the researchers worked in this field and they published many papers. As shown in Figure 1, an obvious publication peak appears around the year 2012, but in the years of 2013, 2014 and 2015 the number of published papers for SMS spam filtering are decreasing [12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23]. Figure 1 presents the distribution of research attention to SMS spam detection by publication year, which published from the year of 2004 to 2015 about SMS spam.

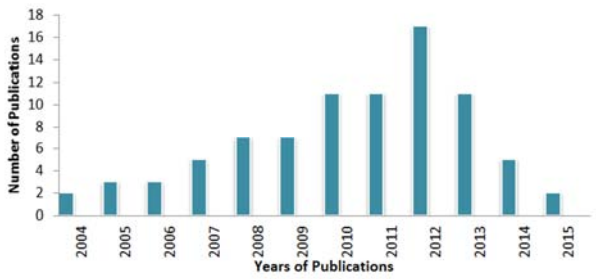

Figure 1. *Number of papers about SMS spam that published in Conferences' proceedings and Journals from the year of 2004 to 2015.*

3.1. What is SMS Spam

SMS Spam in the context is very similar to email spams, typically, unsolicited bulk messaging with some business

interest[4]. SMS spam is used for commercial advertising and spreading phishing links. Commercial spammers use malware to send SMS spam because sending SMS spam is illegal in most countries. Sending spam from a compromised machine reduces the risk to the spammer because it obscures the provenance of the spam.

SMS can have limited number of characters, which includes alphabets, numbers, and a few symbols. A look through the messages shows a clear pattern. Almost all of the spam messages ask the users to call a number, reply by SMS or visit some URL. This pattern is observable by the results obtained by a simple SQL query on the spam corpus.

3.2. Performance Measurement Criterion

In this section, we review important performance indices to measure the strength of spam filtering algorithms. There are various performance criterions such as Recall, Precision, Accuracy, and Area Under the ROC Curve (AUC) used by authors [25]. ROC curves and their relatives are very useful for exploring the tradeoffs among different classifiers over a range of costs. Roughly speaking, the larger area under the curve shows the better performance. To determine other three criterions, first we should define some terms:

True positive (TP): The rate of legitimate SMS messages that have been classified correctly.

False positive (FP): The rate of spam SMS messages that have been classified correctly.

True negative (TN): The rate of legitimate SMS messages that have been incorrectly classified as spam messages.

False negative (FN): The rate of Spam SMS messages that have been incorrectly as legitimate messages.

False-positive error, which diverts a legitimate SMS as spam is generally considered more serious than a False-negative.

Now the so called performance measurement criterions could be defined as:

$$Spam\ Recall\ = T_P/\ (T_P + F_N)$$

$$Spam\ Precision\ =\ T_P/\ (T_P + F_P)$$

$$Spam\ Accuracy\ =\ T_P + T_N/\ (T_P + T_N + F_N + F_P)$$

Actually, Recall determines the proportion of legitimate messages, which have been correctly categorized, precision determines the proportion of all correctly categorized messages, which are legitimate, and Accuracy determines the proportion of all messages, which have been categorized correctly.

In this paper, we use Accuracy for comparing the performance of the methods.

3.3. Feature Extraction

In the classification, feature selection is necessary in order to remove the noisy features and select the best characteristics of messages to categorize them. Furthermore, it also simplifies the calculation, avoids over-fitting and increases the accuracy [14]. The feature selection method should not be very complicated to avoid the significant delay in messaging services but the feature should be highly correlated to message type to increase the spam detection accuracy [42]. The success of machine learning techniques depends mainly on the selection of a suitable feature set for the problem in question. Scientists in the field of feature engineering have been investigating to identify the best features of messages, which could be used in message representation and classification. In this section, we describe various features of messages proposed in different papers. Some of these features could also act as classification rules and populated by users, which enables personalized filtering.

In Table 1, we introduce 17 classes of feature sets, which have been used for SMS spam filtering. For example, class 1 considers two different tokenizers for feature extraction and class 4 considers the whole string of SMS as a feature vector.

Table 1. *The way of Feature Extraction in different researches.*

Number of Feature class	Feature sets	Reference Number
1	Considering two different tokenizers: 1) tok1: tokens start with a printable character, followed by any number of alphanumeric characters, excluding dots, commas and colons. 2) tok2: any sequence of characters separated by blanks, tabs, returns, dots, commas, colons and dashes are considered as tokens.	[26]
2	Tag Cloud : Considering the tag cloud generated from ham and spam messages	[27]
3	Octet Values: Extract the octet values in the hexadecimal format from the payload of SMS.	[39]
4	String of SMS	[24], [40]
5	Static Features: The number of messages.	[28]
6	Temporal Features: Number of messages during a day, on each day of week, size of messages during a day, on each day of week, Time-of-day.	[28], [15]
7	Network Features: Number of recipients, clustering coefficient.	[28], [15]
8	Message Metadata: message length, which is the overall byte length of SMS, number of tokens and average token length.	[42], [21], [20]
9	OCR (optical character recognition) and image of objects	[37]
10	Spam Words: Tokenization SMS string by removing non-letter words and considering the number of spam words such as {“buy”, “free”, “Viagra”, “SMS”, and “book”}.	[41]
11	First treat SMS with word segmentation, after finishing word segmentation of SMS text, characteristic representation of text should be carried out.	[33]
12	Considering different tabs, sender Blacklisting and sender Whitelisting, report as ham or spam, user preferences, customized notifications, automatic filtering and crowdsourcing.	[38]
13	X2-Statistic (CHI) and Mutual Information: the remained of SMS string after using MI	[14]

4 http://paulgraham.com/spam.html

Number of Feature class	Feature sets	Reference Number
	and CHI feature selection methods subsequently.	
14	X2-Statistic(CHI): the remained of SMS string after using CHI feature selection method.	[18]
15	Phone Number, URLs and Time of SMS.	[21], [20]
16	Special characters: characters like "$$$" used by spammers instead of other words to avoid spam detection.	[42]
17	Emotion Symbols: good indicators of legitimate messages	[42]

3.4. Data set Selection

An important phase in evaluating the performance of SMS classification methods is choosing a data set of spam messages to test the method's performance on it. The lack of real, public, and available databases compromises the development of different methods. Unlike email spam, which has a large variety of datasets, the mobile spam filtering has very few corpora. Various resources have been used by authors to construct a comprehensive dataset. Some of these resources are:

1. NUS (National University of Singapore) SMS corpus
2. Jon Stevenson corpus
3. Grumbletext Website
4. Caroline Tag's PhD Theses

In this section, we introduce datasets used by authors to experiment the algorithms. Many authors have used 2 datasets to analyze their methods; however, as the SMS Spam Collection V.1 designed by Tiago is the biggest one, we write the accuracy of each algorithm according to its power on classifying the messages of this dataset.

3.4.1. SMS Spam Corpus V.0.1 Big

This corpus is a collection of 1,002 legitimate messages and 322 spam SMSs in English language. The legitimate SMS messages were randomly selected from the National University of Singapore (NUS) SMS corpus (10,000 legitimate SMSs) and the Jon Stevenson corpus (202 legitimate SMSs). The spam messages were collected manually. From the Grumbletext Website, which is a public forum where users claims SMS spam messages. The average word length is 4.44 characters and the average number of words per message is 15.725. This dataset is available at (http://www.esp.uem.es/jmgomez/smsspamcorpus/) and has been used in [42].

3.4.2. SMS Spam Collection V.1

To make a more comprehensive dataset, Tiago et al. have made a real, public, and non-encoded SMS spam collection, which is the biggest one ever. This SMS corpus has been collected from various sources in the internet. Firstly, a collection of 425 SMS spam messages was manually extracted from the Grumbletext website**.** This website is a UK forum, which cellphone users propose their public ideas about SMS spam messages but they do not include the actual messages. Therefore, it is very difficult to identify the text of SMS spams through the claims [26]. Secondly a subset of 3,375 SMS, have randomly chosen from ham messages of the NUS+ SMS Corpus. In addition, legitimate samples have been added to corpus by inserting 450 SMS messages collected from Caroline Tag's PhD research. Finally, 1002 ham messages and 322 spam messages have been incorporated to Tiago et al. collection from SMS Spam v.0.1 Big dataset. Therefore, the ultimate SMS Spam Collection is composed of a total of 5,574 short messages (4,827 legitimate messages and 747 mobile spam messages) and is publicly available at: http://www.dt.fee.unicamp.br/~tiago/SMSspamcollection/ [26]. To the best of our knowledge, it is the largest available SMS spam corpus that currently exists and most of the authors have used it to calculate the accuracy of their methods. We also write the accuracy of each method in confronting to this data set to make a fair comparison among them.

3.4.3. India Dataset

The New Delhi**,** India dataset is another SMS spam dataset. This collection is composed of 2195 legitimate messages and 2123 spam messages, a total of 4318 short messages [27].

Table 2. *Shows the descriptive statistics of the data sets and compare the data sets from different aspects.*

Reference Number	Creators	Country	Number of Hams	Number of Spams	Number of Unknown-type	Number of instances	Year of creation
[42]	Tiago A. Almida, José María Gómez Hidalgo	Spain	1002	322	-	1324	2011
[26]	Tiago A.	United Kingdom	4827	747	-	5574	2012
[27]	Kuldeep Y., Ponnurangam K., Atul G., Ashish G. and Vinayak N.	India	2195	2123	-	4318	2011
[28]	Qian X., Evan Wei X. and Qiang Y.	China	3589661	215	1310592	4900468	2010
[29]	Alper U., Serkan G., Semih E. and Efnan G.	Turkish	430	420	-	850	2012

3.4.4. Chinese Dataset

The Chinese dataset is a realistic data from a Telco[5] in China, which is also one of the largest telecommunications operators in the world. In this dataset, they have 4,900,468 SMS senders. The SMS dataset collected in seven days (25/03/2010 -31/03/2010) from a province in China. In all, they have 3,589,661 legitimate senders and 1,310,592 unknown-type senders. Domain experts manually identified 215 spammers that serve as positive examples of spam messages [28].

3.4.5. Turkish Dataset

The other available dataset for spam filtering is Turkish

5 Telecommunications Company

dataset. The collection consists of 420 spams and 430 legitimate messages that are collected from the volunteers. The collection, namely TurkishSMS, is publicly available at: http://ceng.anadolu.edu.tr/par/ [29].

4. Learning-Based Methods for Spam Filtering

Filtering is a popular solution to the problem of spam. It can be defined as automatic classification of messages into spam and legitimate SMS. Existing filtering algorithms are quite effective, often showing accuracy of above 90%. In general, a spam filter is an application, which implements a function like $f(m,\theta)$ in Equation (1):

$$f(m,\theta) = \begin{cases} C_{spam}, & if\ the\ message\ m\ is\ considered\ spam \\ C_{leg}, & if\ the\ message\ m\ is\ considered\ legitimate\ SMS \end{cases} \quad (1)$$

where m is a message to be classified, θ is a vector of parameters, c_{spam}, and c_{leg} are labels assigned to the messages. Most of the filtering methods are based on machine learning classification techniques. In a learning-based technique, the vector of parameters θ is the result of training the classifier on a pre-collected dataset. θ and M are as follows:

$$\theta = \vartheta(M) \quad (2)$$

$$M = \{(m_1, y_1), (m_2, y_2), \ldots (m_n, y_n)\}, \quad \in \{C_{spam}, C_{leg}\}, \quad (3)$$

where $m_1, m_2, \ldots m_n$ are previously collected messages, y_1, $y_2, \ldots y_n$ are the corresponding labels, and ϑ is the training function.

In recent years, a variety of methods and techniques has been proposed by researchers in the area of detection of SMS spam. In the following, we will review the machine learning techniques that have been employed for SMS spam detection so far. The methods exhibiting a high level of accuracy in results are categorized in the following. In the first subsection, some well-known classification methods used for spam filtering are introduced and in the second part, the special methods for spam filtering are investigated.

4.1. Classification Methods

In the following, we investigate some well-known classification methods employed in spam filtering applications.

- Naive Bayes Algorithm (NB)

The Naive Bayes algorithm creates a probabilistic model for classification of SMS messages. Even though all features contribute towards the overall probability of classification, Naive Bayes algorithm assumes that the features are statistically independent of each other. Although this assumption may not hold true for all cases, Naive Bayes algorithm has shown promising results in comparison with other well-known classification algorithms. An advantage of Naive Bayes is that it only requires a small amount of training data to estimate the parameters necessary for classification. Because independent variables are assumed, only the variances of the variables for each class need to be determined and not the entire covariance matrix [30].

The basic decision rule can be defined as follows:

$$f(\bar{x}) = \underset{y=\{C_{spam}, C_{leg}\}}{argmax} \left(\hat{P}(y) \prod_{j:\, x^j=1} \hat{P}\,(x^j = 1 |\, y)\right) \quad (4)$$

where x^j is the jth component of the vector $\bar{x}$, $\hat{P}(y)$, and $\hat{P}(x^j = 1 |\, y)$ are probabilities estimated using the training data.

- Support Vector Machine (SVM)

Support vector machines are supervised learning models, which analyze data and recognize patterns, used for classification and regression analysis. Given a set of training examples, each marked as belonging to one of two categories, an SVM training algorithm builds a model that assigns new examples into one category or the other, making it a non-probabilistic binary linear classifier. A SVM model is a representation of the examples as points in space, mapped so that the examples of the separate categories are divided by a clear gap that is as wide as possible. New examples are then mapped into that same space and predicted to belong to a category based on which side of the gap they fall on. In addition to performing linear classification, SVMs can efficiently perform a non-linear classification. An SVM method is based on structural risk minimization [31]. It avoids the use of many training documents, employing only those near the classification border, to construct an irregular border separating positive and negative examples. By employing a suitable kernel functions, it can learn polynomial classifiers, radial basis functions, and three-layered sigmoid neural nets, thus acquiring universal learning ability.

- K-Nearest Neighbors (KNN)

The KNN technique [32] works by choosing first random data points as initial *seed* clusters. Then, it enters a learning phase when training data points are iteratively assigned to a cluster whose center is located at the nearest distance (*e.g.* Euclidean distance). Cluster centers are repeatedly adjusted to the mean of their currently acquired data points. The classification algorithm tries to find the K-Nearest Neighbor of a test data point and uses a majority vote to determine its class label. The performance of KNN classifier is primarily determined by (i) an appropriate choice of *K,* and (ii) the distance metric applied.

The Idea of using KNN for solving the spam detection problem is so easy. Actually, if we are going to classify the message d, we consider the classified messages in its neighbors and d would be in the same class as most of its neighbors. This could be summarized in Equation (5) and (6):

$$y(d, C_j) = \sum_{d_i\ is\ in\ KNN} Sim(d, d_i) \times y(d_i, C_j) - b_j\,; \quad (5)$$

$$y(d, c) = \begin{cases} 0, \text{if } d \text{ is in class C} \\ 1, \text{if } d \text{ is not in class C} \end{cases} \quad (6)$$

where C_j is the class j (in this field we have two classes: spam and legitimate) and the amount of $y(d, C_j)$ is achieved from the right hand of first formula and shows that d is in class C_j

or not. Moreover, b_j is the predetermined threshold of C_j. The similarity of two entities in the KNN algorithm ($Sim(d, d_i)$) is generally calculated according to the Euclidean distance and in the field of spam detection the Hamming distance could be useful too [14].

- Voted Perceptron

This algorithms proposed by Freund and Schapire, which replaces all missing values and transforms nominal attributes into binary ones [34]. The voted perceptron method is based on the perceptron algorithm of Rosenblatt and Frank. The algorithm takes advantage of data that are linearly separable with large margins. The implementation of this algorithm is simple, has no computational complicity time as compared to Vapnik's SVM and also is comparable to SVM in terms of accuracy. The algorithm can also be used in very high dimensional spaces using kernel functions [24].

- Ada Boost

AdaBoost, short for "Adaptive Boosting", is a machine learning meta-algorithm, which proposed by Yoav Freund and Robert Schapire. Boosting is based on creating a highly accurate prediction rule by combining many relatively weak and inaccurate rules and works by repeatedly running a given weak learning algorithm on various distributions over the training data

It can be used in conjunction with many other types of learning algorithms to improve their performance. The output of weak learners is combined into a weighted sum that represents the final output of the Boosted classifier. The individual learners can be weak, but the performance of them will be increase by combining them and the final model can be proven to converge to a strong learner [35].

- Lazy KStar

Cleary et al. proposed a type of lazy algorithm, which called KStar (K *) [36]. An instance-based classifier that is the class of a test instance is based upon the class of those training instances similar to it, as determined by some similarity function. It differs from other instance-based learners and it uses an entropy-based distance function. The main advantage of a lazy learning method is that the target function will be approximated locally, such as in the KNN algorithm. Because the target function is approximated locally for each query to the system, lazy learning systems can simultaneously solve multiple problems and deal successfully with changes in the problem domain. But the disadvantages of lazy learning is the large space requirement to store the entire training dataset. Particularly noisy training data increases the case base unnecessarily, because no abstraction is made during the training phase.

4.2. Special Methods for Spam Filtering

Some special and hybrid methods proposed for spam filtering are described in the following.

- Dendritic Cell Algorithm (DCA) for mobile spam detection

This method [42] proposes a new approach for SMS spam detection by benefiting the combination of two feature sets to enhance the performance and fusing the result of two content-based algorithms (SVM and Naïve Bayes) using the Dendritic Cell algorithm. Dendritic Cell Algorithm is a classification approach in machine learning developed based on the behavior and function of Dendritic Cells (DCs) in the biological immune system. This algorithm receives signals and antigens as input and the combination of signals and antigen temporal correlation and diversity of DC population is responsible for the detection capability of the DCA. This method operates in the following phases:

1. Preprocesses phase

a. The SMS string is converted to lower case.

b. The SMS string is tokenized.

c. All the tokens are reduced to their root word. Actually, all the prefixes and suffixes should be eliminated.

d. Two feature sets should be elected. In this approach, Metadata and spam words of the message are used as feature sets because experiments have shown that the combination of these two sets enhances the performance highly.

e. Each SMS is shown by a feature vector such as $X = (x_1, x_2, \dots, x_m)$ where m is the numbers of its features and x_i is the weight of feature i in that SMS.

2. Classification phase

In this phase, the input signals of the DCA algorithm are generated. DCA has three types of input signals: PAMP signals (a measure of confidence that the antigen represents a spam), DANGER signals (a measure which indicates a potential abnormality) and SAFE signals (a measure that increases in value in conjunction with legitimate messages). We give the feature vector of a message to NB and SMV algorithms separately and they generate a decision with a confidence level. There are three different conditions:

1. Both of the algorithms determine that X is a spam message. A PALM signal is generated equal to the maximum confidence level of NB and SVM.
2. Both algorithms determine that X is a legitimate message. A SAFE signal is generated with the maximum confidence level of NB and SVM.
3. NB and SVM make opposite decisions. In this situation A DANGER signal is generated with the average confidence level of NB and SVM.

The derived signals and associated antigens are passed to the DCA algorithm as input. The final decision that a message is spam or legitimate is made by the DCA classification algorithm [42].

- Blacklist/Whitelist technology

Blacklist and Whitelist are phone number lists of spam SMS senders and trustful senders respectively. "Blacklist" method is exclusion and SMS users in Blacklist are not allowed to send any SMS. "Whitelist" method is inclusion, which is typically used for confirming legal SMS. SMS users in Whitelist are not limited to send SMS and SMS sent out are approved as normal SMS [33].

Although Blacklist/Whitelist filtering technology is simple and efficient, has little consumption of system resource, and is easy to be implemented, but name lists should be manually maintained. Also, this technique is not so accurate and sometimes Blacklist technology may reject normal SMS sent

from the number because spam SMS senders may discard used numbers after sending bulk spam SMS, Blacklist technology can only play complementary role in filtering scheme of spam SMS.

- Keywords filtering technology

Keywords filtering technology makes identification and processing through keywords matching on spam SMS content adopting some simple, or complex word list related to spam SMS. For instance, those with the subject including "free", "hot selling", "impassion", and other words often existing in spam SMS. However, there is obvious relation between keywords filtering capacity and keywords, and an abundant filtering keywords list must be created. When processing spam SMS adopting such technology, the system will consume large quantity of system resource. Besides, spam SMS senders will often misspell some words filter via misspelling some words, or adopt variant words, and homophonic words to escape from words filter, therefore, words filter should be upgraded frequently and add keywords alteration [33].

- Discriminative Multinomial Naive Bayes Text

This is a class for building and using a Discriminative Multinomial Naive Bayes (DMNB) classifier, which proposed by Su et al. DMNB Text is a simple Bayesian classifier with discriminative parameter learning for text categorization [24]. The motivation of DMNB is to hold the frequency information while considering the discriminative nature of classification and thus is an integration of generative and discriminative learning. In addition, DMNB algorithm performs competitively with other state-of-the-art text classification algorithms, but often achieves better testing accuracy with small training data.

- Bayes Net (Simple Estimator + K2)

It is a probabilistic graphical model, which represents a set of random variables and their conditional dependencies via a Directed Acyclic Graph (DAG). This algorithm uses simple estimator and K2. This is the base class for a Bayes Network classifier, and provides data structures and facilities common to Bayes Network learning algorithms. Simple estimator is used for estimating the conditional probability tables of a Bayes Network once the structure has been learned. K2 is a Bayes Network learning algorithm that uses a hill climbing algorithm [24].

- CAPTCHA

CAPTCHA[6] systems are used to identify and distinguish between human users and computer programs automatically. The focus of CAPTCHA is based on questions, which the human users can answer easily but the computer programs cannot answer. In this technique, a bank of images and their names is prepared and each image is converted to its specific format as a two-color image. Because the SMS picture messages have small sizes (only 256 byte), we can save a large number of pictures on the mobile phone. When a SMS is received, a CAPTCHA test is prepared and sent for the SMS sender as a SMS message. For generating the CAPTCHA test, image of an object in SMS picture message format is selected from the bank of images, and inserted in a SMS. Also name of that object and name of three other objects is written in that SMS. This SMS is sent to the SMS sender, the SMS sender should recognize the object, choose the name of that object and then sent that objects' number back in reply as a SMS text message. Since the current programs have difficulties in recognizing the object pictures, if the SMS sender can pass the CAPTCHA test, it will be identified as legitimate SMS [37].

- SMS Assassin

In this method, Kuldeep et al. design and implement a user-centric mobile-based application, which can filter spam SMSes. Since, Content-based filtering techniques have limited capabilities due to short size of SMSes, SMSAssassin uses Content-based filtering with user-generated features to automatically filter spam SMSes. It uses different viewing space for different type of SMSes to make the management of SMSes easier for the user. In addition, it prepares an interface for customized notifications using, which user can personalize to receive notifications on the reception of useful content only. Whenever a new SMS comes, applications extract all the words and compute a score with the help of training file. Then this score is compared with a threshold parameter δ[7] to decide on a SMS is spam or ham.

Here, there are some of the features of the application, which some of them acts as classification rules, and populated by users and enables personalized filtering.

1) Different Tabs: Their application has three different tabs; first tab is for ham (Inbox), second one is for spam SMSes (SpamBox), and the third one is for user preferred SMSes. Whenever a new SMS comes, the application automatically decides on corresponding tab based on its filtering mechanism.

2) Sender Blacklisting and Sender Whitelisting: Sender Blacklisting feature is used to block a specific sender.

3) Report as Ham or Spam: If a SMS is spam and wrongly put into Inbox by the application, then the user can report it as spam to reveal the application about wrong decision.

4) User-Preferences: In this feature, user can bind any keyword (for instance, "Pizza") or phone number(s) to user-preferred tab. All the SMSes, which are from the user-preferred sender or contains the user-preferred word will come to user preferred tab.

5) Customized Notifications: Application provides a customized notification mechanism for each tab, which can be set by user according to her own requirements.

6) Automatic Filtering and Crowdsourcing: If a SMS does not get classified by sender Blacklisting/Whitelisting or any of user generated rules, then it is passed through a trained classifier based on Bayesian filtering to make a decision whether it is ham or spam.

The application uses crowdsourcing to keep itself updated. SMSAssassin prepares ways to predict different SMS message based on their content and sender to different tabs, so that they automatically gets organized and user has to do less effort than

6 Completely Automatic Public Turing Test to Tell Computer and Human Apart

7 Value of δ is determined empirically from the training dataset

the previous case [38].

- Using Evolutionary Learning Classifiers

This approach filters SMS spam at the access layer of a mobile phone. It analyzes a SMS in hexadecimal notation and extracts two features from this format: (1) Octet bigrams (2) Frequency distribution of bytes. This UCS [8] based spam filtering system has met three requirements: (1) It provides 93% detection rate with 0% false alarm rate, (2) It obtains this performance by training only on 500 messages, (3) It takes less than 21 KB of memory to store the features and approximately 1 second to analyze a message at the access layer. This method uses a combination of octet bigrams, and frequency distribution of bytes to reduce the dimensionality of the features' space, which makes it appropriate for resource-constrained mobile phones. This filtering system does not depend on semantics of language, because it works with the hexadecimal notation. Moreover, they have also used the dataset, which introduced by Tiago [39].

-Crowdsourcing Driven Mobile-based System

SMS Assassin uses crowdsourcing to keep itself updated. In the training phase, it computes the occurrence of a word in spam as well as legitimate SMS to learn the probability of finding that word into spam/ham. After training, Bayes theorem is applied to calculate the probability of the message being a spam with different words in that message. Then, the technique computes the combined probability with basic assumption, which all of these are independent events. Finally, the combined probability value is computed, then compared with a threshold ρ[9], if the threshold is greater than ρ, then the message is likely to be a spam otherwise ham [27]. The SMS database that used contains total of 4,318 SMSes from New Delhi, India.

Table 3. *Classification accuracy comparison of machine learning approaches by using the same training and testing set [27].*

	Ham Accuracy	Spam Accuracy
Bayesian Learning	97%	72.5%
SVM	93%	86%

-Hybrid spam filtering for mobile communication

Ji Won et al. proposed a spam filtering based on combination of Content-based filtering approach with the challenge-response protocols, and they demonstrate that combined approach can be more effective and efficient in detecting spam messages. In By using the Content-based filtering approach, obvious spams are filtered first to reduce the number of messages subject to challenge-response, the challenge-response protocol then classifies uncertain messages with high accuracy. By combining the content filtering algorithm with the challenge-response protocol, they show that, high accuracy and low message traffic can be achieved simultaneously. Challenge-response protocols are based on CAPTCHA.

In this method, SMS messages are first classified into three different regions by using the Content-based filtering method; ham, uncertain, and spam. Actually by choosing two thresholds in traditional content-based method, if a specific ratio (odd ratio) is between these two thresholds the message is considered as uncertain. By using the Content-based filtering approach, obvious spams and hams are filtered first (messages with the ratio more than the larger threshold or less than the smaller threshold). The challenge-response method is then used to further classify the uncertain messages into ham, and spam regions. The majority of spam messages is generated by machines, Therefore a human verification mechanism such as challenge-response is used to detect whether an uncertain message falls into the ham or spam region. They would suggest that the message center should be given the full responsibility of running their framework for the following reasons:

-To reduce the traffic usage by filtering spam messages at the earliest possible stage; that is, before forwarding them to the recipient

- By using the challenge-response protocol, the message center will be able to collect a large number of sample data in real time; these can be used to develop highly effective classifiers and continuously improve the performance of filtering algorithms.

-It would be difficult to install and maintain homogeneous anti-spam software on all mobile devices; instead they rely on one solution deployed in the message center [40].

- Independent and Personal SMS Spam Filtering

Taufiq et al. proposed an independent filtering system that does not need a computer system support. The training, filtering, and updating processes were done on mobile phone. Their proposed approach filters SMS spam on an independent mobile phone, while obtaining reasonable accuracy, minimum storage consumption, and acceptable processing time. The method includes the following steps:

1. Feature extraction: the feature set for this method results from non-letter tokenization.
2. Vector creation: to create a vector from the primitive SMS message and send it to classification algorithm as input, the number of word occurrences is used because of its simplicity (as the method computes on mobile phones).
3. Filtering process: for filtering phase, the Naive Bayes algorithm is used to filter unknown incoming messages.
4. Updating filtering system: after receiving a new SMS first and second steps should be repeated. If a word already exists in the word occurrences table, we will just update the word occurrences table. If the word does not exist, we will add the word to the word occurrences table.

-Using Non-Content Features

Content-based spam filtering methods need the contents of SMS messages, which are expensive or impractical to obtain and cannot ensure the privacy of users. Thus, Qian et al. intend to find content-less methods for spam detection. They only concentrate on finding SMS spam on the server side. The main dataset they consider is a realistic data from a Telco in China.

8 Supervised Classifier System

9 The designer of the filter can decide on the threshold value.

For features selection Qian et al. extracted these features:

Static Features: The number of messages, Message size

Temporal Features: Number of messages during a day, on each day of week, size of messages during a day, on each day of week, time-of-day

Network Features: Number of recipients, Clustering Coefficient.

In this method for classification, they consider KNN and SVM algorithms and the result demonstrate that SVM works better [28].

-Graph-based KNN Algorithm for Spam SMS Detection

Huang et al. proposed a method, which is a combination of graph-based text representation technique and KNN algorithm. For extracting features, they used mutual information (MI) and X2-Statistic (CHI). The MI is used to compute the mutual dependence of two random variables and the $\chi 2$-statistic measures the lack of independence between token and class. Firstly, they separate the message collection into many small message groups that each group is represented by a graph and is also as an entity in KNN algorithm. Secondly, they tokenize the content of messages in a group by white spaces and punctuations. After tokenizing the content of messages into space-delimited words, they remove several words that have only one character and calculate the weight of each word by using the feature extraction. Then, they select the words, which have high weight and use them for constructing the graphs in the next section. Finally, they use all of sample graphs and testing graph in KNN algorithm to decide the label of the new messages. The experimentation is carried out on SMS message collections and the results ensures the efficiency of the proposed method, with high accuracy which is 98.9% and small processing time enough for detecting spam messages directly on mobile phones in real time [14]

-Using Latent Dirichlet Allocation with Social Network Analysis

In this method Abiodun et al., introduced Latent Dirichlet Allocation (LDA) with Social Network Analysis (SNA) to extract and evaluate latent features arising from mobile Short Messaging Services communication. This would help to automatically filter SMS spam before their delivery. Experiments were successfully performed by collecting time-stamped short messages via mobile phones across a number of different categories on the Internet, using an English language-based platform, which is available on streaming APIs [17].

-Understanding SMS Spam in a Large Cellular Network

Jiang et al. have done a comprehensive analysis of SMS spam activities in a large cellular network by combining user reported spam messages and spam network records. They studied in-depth various aspects of SMS spamming activities, including spammer's device type, tenure, voice and data usage, spamming patterns and so on. They understand that most spammers selected victims randomly and spam numbers sending similar text messages exhibit strong similarities and correlations from various perspectives. Based on these facts, they proposed several spam detection methods, which showed results in terms of detection accuracy and response time. For instance, they devise a new algorithm for detecting related spam numbers. In this algorithm firstly, all the SMS senders in the network are monitored and senders who send message to more than β numbers in time intervals of length T, would be assumed as spam senders. Secondly, if the presumed number is confirmed to be spam sender (by the report of a user or other methods), other numbers with the same spamming location as this number will be added to spam numbers watch list and they become spam candidates. The algorithm obtained a high accuracy of 99.4 % in real network data. Moreover, 72 % of these spam numbers are detected at least 10 hours before user reports [15].

-FIMESS: FIltering Mobile External SMS Spam

Androulidakis et al. proposed an algorithm, which performs simple, because checks on the message headers to classify an SMS as being spam or not. Their method is able to use the important information in the SMS headers and identify SMS spam messages. The proposed approach was tested on the Android platform and is based on the technical manoeuvres. The proposed algorithm flags inconsistencies in suspicious messages and, if multiple criteria are satisfied, identifies them as spam. Initially, the application monitors incoming SMS messages and records in a lightweight database the SMS of every sender. It then applies the following rules to distinguish between legitimate and unsolicited communication:

1-Whenever a new SMS arrives, the originator number is checked to determine whether it has sent an SMS before. If the number already exists in the database, the system compares the SMS of the received message to that in the database record containing the same originator number.

2-Usually, the few digits of an SMS match the respective digits of the sender's number. This leads to the provider's identification via the numbering plan, unless number portability is in effect.

3-The sender's identification is checked to determine if it is a purely numerical one, or it contains other characters. A message with a non-numerical sender ID has a great chance of being spam.

4-The time zone of the SMS and that of the mobile phone can also be used as possible indicators of a spam message.

5-Keywords Blacklist. Blacklisted words such as \viagra", \free" etc. immediately characterize a message as being questionable.

6- The application checks whether HTTP links exist in the SMS and informs the user accordingly, since such a link could suspicious to spam.

7- In addition, the SMS protocol ID (TP-PID) is checked. These can be used for determining whether a mobile phone is switched on or not, without making the user aware of this action [16].

- Improving Static SMS Spam Detection by Using New Content-based Features

Karami et al. proposed features that can improve the performance of SMS spam detection. The quality of a classifier depends on the quality of features, for this purpose,

they explored two broad categories of features: SMS-specific content-based features and Linguistic Inquiry and Word Count (LIWC)-based features. They tested 40 classification algorithms with different test settings and chose chi-square as the feature selection method on Tiago dataset. The classification accuracy ranges are between 92% and 98% across various algorithms. Among the algorithms that are tested, boosting of Random Forest and SVM algorithms showed the best performance. Their experiment results show that incorporating semantic categories improve the performance of SMS spam detection [18].

- Semi-supervised learning using frequent item set and ensemble learning for SMS classification

Ishtiaq et al. have proposed a new semi-supervised learning method, which use frequent item set and ensemble learning (FIEL). In this method, to find the frequent item set, Apriori algorithm has been used while Multinomial Naive Bayes, Random Forest, and LibSVM are used as base learners for ensemble learning. Initially, positive features are calculated using certain minimum support in ham SMS and later on, these features are gradually expanded with the help of unlabeled SMS. Moreover, the negative features are also generated by positive features and unlabeled SMS. Therefore, these unlabeled instances (SMS) are labelled with the argument that maximizes the mode of the features. Finally, these newly labeled dataset are fed to train the base classifiers such as Multinomial Naive Bayes, Random Forest, and LibSVM. This method is evaluated on UCI SMS spam collection dataset; SMS spam collection Corpus v.0.1 Small and Big dataset. The proposed approach produces good result especially when the positive instances ratio is low and throughout different positive dataset ratio, it is highly stable. However, the minimum support for finding frequent item set varies largely that depends on the dataset volume [19].

- Semi-Synthetic Data for Enhanced SMS Spam Detection

In this method, Eshmavi et al. want to study the effect of using Synthetic Minority Oversampling Technique on the detection of SMS spam. Their study shows an improvement in performance of the classifiers trained on semi-synthetic datasets compared to the performance of the same classifiers trained on the original dataset. To report the results of classifiers' predictions, they use two testing: 1) 10-Fold Cross Validation on the first 80% of the data, which are selected from Tiago dataset. 2) 80%-20% holdout testing approach. In addition, to generate the feature vector for each message, they extract the following features:

• Term Frequencies: they used the words that are present for at least 25 times in the spam messages.

• HasNumber: because most of the time spam messages will include a number to call, to send text to, or a code to reply with.

• HasLink: this feature shows the presence of a link.

• CapitalRatio: messages have capitalized words to catch the user's attention. It is calculated by dividing the number of capitalized words by the number of tokens name message.

• TokNum: the number of tokens in the spam messages that usually exceeds the number of tokens in the ham messages.

The results show an improvement in all the classifiers performance. This means that, by solving the class imbalance problem of the spam corpus using synthetic minority oversampling technique for generating synthetic samples, the classifiers can do a better job predicting new spam SMS messages [20].

- A Multi-agent System for Smartphone Intrusion Detection Framework

Ghorbani et al. proposed a multi-agent system, which developed by using JADE platform for observing Android Smartphone features and monitoring SMS services. They suggested this technology to solve the limitation of Android smartphone resources and make their framework more efficient. Their framework applies hybrid detection approaches in order to counteract botnet attacks, by investigating damaging SMS botnet activities through the examination of Smartphone behavior. These approaches use multi-agent technology to recognize malicious SMS and prevent users from opening these kinds of messages, by applying behavioral analysis to find the correlation between suspicious SMS messages and the profiles reported by the agents. To identify SMS botnets in Android mobile devices, they have defined a multi-agent system, which has the ability to monitor and observe Android device activities, and then capture and report suspicious behavior to a central server. Their framework has a model, which applies signature-based detection to Smartphone SMS messages, and behavior detection on collected data at the central server. Profiling behavioral analysis is conducted in the central server in order to spot unknown SMS botnet using a multi-detection system. This is done in order to collect data from mobile devices by the agents and send the collected data to the central server. This leads to a response from the Decision-and-Action Module, which finally sends an action to be performed on the Smartphone [21].

- SMS Spam Filtering Based on "Cloud Security"

In this method, Wu et al. applied "filter cloud" strategies of filter spam messages based on "Cloud Security" in order to achieve the purpose of filtering spam messages by addressing its root causes. "Filter cloud" applying mature Black and White list strategy, Bayesian filtering strategy and flow analysis strategies in it. The idea is that filter will get abnormal suspicious number of spam messages through monitoring and analysis of large number of Black and white list, rate of flow and message of phone client and achieve the newest information. Spam message filter analysis system get the latest information about abnormal suspicious number of spam messages through monitoring and analysis of large number of Black and White list, flow and text message content (SMSC). The SMSC would update Blacklist and refuse to send short message while the number in blacklist request service directly. The spam messages filter system based on cloud security mainly consists of three parts: filter cloud making up of large number cellular phone client, filter analysis system, and short message service center [22].

5. Discussion

In this section, the main characteristics of the top 5 accurate

algorithms are discussed widely. Furthermore, to compare different machine learning algorithms, the characteristics of main methods discussed in previous section are presented in summery in Table 4. According to Table 4, Naïve Bayes, Hybrid method, Graph-based KNN method, Cellular Network method and DCA algorithm achieved the highest accuracy and they are the most successful methods. In this section, we are going to discuss these top 5 methods in details and beyond just the algorithm to be able to choose the best method by considering all of their aspects. The independent method should also be discussed as it has accuracy of 98.29 in some conditions

The DCA algorithm is the most accurate algorithm discussed in this paper. Actually, the proposer of this algorithm has evaluated its performance by using two datasets. The first dataset is SMS Spam Corpus V.0.1 Big and the second one is SMS Spam Collection V.1. The experiments on both data sets have shown that using Spam words and Metadata as feature sets for Naïve Bayes and SVM separately, highly increases the AUC in comparison with other feature sets. Indeed, the combination of these two feature sets leads to the highest AUC in both NB and SVM in comparison with other feature sets. After the election of feature sets for combination, the author analyzed the effect of fusing NB and SVM using DCA, on the accuracy of the classification. Results have shown that on both of the datasets, the proposed algorithm is more accurate than NB or SVM independently. Actually, the accuracy of proposed algorithm is 99.77 in classifying the messages of SMS Spam Corpus V.0.1 Big, and 99.95 in classifying the messages of SMS Spam Collection V.1. This algorithm is fast although using SVM might reduce the speed. Moreover, it is simple does not need complicated calculations and processing requirements [42].

The Cellular Network method relies less on user report of spam messages or even do not require user participation. This accounts as a significant improvement in the spam detecting methods as it reduces the delay of users report and low spam report rate in user-driven approaches. Actually, this delay leads to the loss of many spam numbers, which is somehow solved by Cellular Network method. This method is highly accurate and only 0.06% of the spam sender candidates are not verified during the experiments. By using these algorithms to detect spam numbers, the number of spam messages could be reduced by 50%. In this method, the network features, temporal features and static features have been used. Actually, the number of messages sent from specific geolocations and the number of recipients have been considered. The algorithm yields a high accuracy of 99.4% on real network data. Moreover, 72% of these spam numbers are detected at least 10 hours before user reports [15].

The Graph-based KNN method is in the third place. This method is very fast so that using it we could detect spam messages on mobile phones in real time. Actually, this method processes 875 messages in only 22 seconds, which is equal to 0.025 seconds for each message. In this method, two text feature selection methods have been used: mutual information (MI) and X2-Statistic (CHI). Furthermore, two different data sets have been used to evaluate the algorithm. The First data set was NUS SMS corpus and the second one was downloaded from [Uysal and Yildiz] and contains 875 SMS messages. The accuracy of this method is 98.9.

The hybrid method is in the third place by the accuracy rate of 98.63. As it was explained in section 4.16, this method is the mixture of content-based filtering and challenge response. This method achieves high accuracy in comparison with conventional content-based methods and regardless of the content-based algorithm being used (although the characteristics of the content-based method being used affects the traffic usage and might increase the traffic usage compared to the content-based filtering method, we should choose the content-based method according to the preference of accuracy and traffic usage). Hybrid method could control high amount of spams and also the traffic usage and in this manner low traffic and high accuracy could be gained simultaneously. Ji Won et al. has evaluated the performance of this hybrid method by designing a synthetic data set (as opposed to real data sets) with specific parameters and then randomly initializing these parameters to determine the performance in various environments. Actually the performance have been analyzed by first using different proportion of ham and spam messages and in another condition by fixing the proportion of ham and spam messages and just varying other parameters. The message center network could benefit the hybrid framework by using it to filter spams before forwarding them to recipients and reduce the traffic usage. Moreover, using challenge-based protocol allows the message center to collect spam samples in real time and use them to develop effective classifiers.

The independent method has been developed to filter SMS spams on mobile phones. This method is independent as it does not need the help of a companion computer. Moreover, it is private as it ensures the user's privacy and does not store SMS anywhere, secure as the spammer does not have access to filtering system, personal as users could create their personal filtering system, simple as it uses small training data set and could be trained with only 20 training SMS, and updatable as it is continuously updated to filter new messages. The proposed approach filters SMS spam on an independent mobile phone, while obtaining reasonable accuracy, minimum storage consumption, and acceptable processing time. In this method and in experiments 885 SMS have been used for filtering and updating. If only misclassified SMS are used to update the filtering system, they obtained 90.17% accuracy and a processing time of 0.04 seconds per incoming SMS. If all incoming SMS are used to update, they obtained 95.32% accuracy. Naive Bayes Text Classification algorithms with low computational complexity for both training and filtering have been used as there is no computer to perform complicated computations. All experiments performed on a Google Android HTC Nexus One with the following specifications:

(1) QUALCOMM QSD8250™, 1-GHz Processor.
(2) Android™ 2.1 (Éclair) Operating System.
(3) ROM Memory 512MB and RAM Memory 512MB.

(4) Micro SD™ memory card (SD 2.0 compatible).

Table 4. *Evaluation and comparison of different approaches (ND means not defined and "-" means these methods don't have data set)*

Ref No.	Algorithm	Strengths	Weaknesses	No. of features	Accuracy (%)	Data set
[24]	Naive Bayes	Simple algorithm Efficient to train and use Independence assumption minimizes computational complexity Easy to update with new data Wide applicability	1. It has a wrong feature independence assumptions 2. Bayesian poisoning	7	98.2	[19]
[26]	SVM	There is no need to calculate all features in the training dataset to achieve desired accuracy	Entails a long training time	1	97.64	[19]
[42]	DCA	1. Very accurate 2. Simple 3. low CPU processing requirements	Using SVM increases process time	2	99.95	1,2
[24]	KNN	Easy to implement and modify	Performance is degraded with increase of noise in training data	7	94.3	[19]
[33]	Blacklist/Whitelist	1.Simple and efficient 2.Easy to be implemented 3.Has little consumption of system resource	1.User-centric 2.Not so accurate	9	ND	-
[24]	Voted Perception	1.Simple for linear classification 2.Easy to be implemented 3.More efficient in terms of computation time as compared to SVM	It's more suitable for linear classifier	7	97.0	[19]
[24]	Ada Boost	1.Help to improve performance 2.Less susceptible to the over fitting problems than other learning algorithms	1.It's not so accurate 2.Sensitive to noisy data	7	96.8	[19]
[24]	Lazy Kstar	1Target function will be approximately locally 2.Updatable classifier	1.It's not accurate 2.Large space requirement to store entire dataset 3.Slow to evaluate	7	95.1	[19]
[28]	Non-Content features	1.Not as expensive as content base features 2.Sacrifice the privacy of user	It's not as accurate as content base algorithms	5	ND	[19]
[40]	Hybrid Method	1.So accurate 2.Simple to understand for human	1. Large space requirement to store dataset 2.consumption of cellular network bandwidth	4	98.63	Synthetic dataset
[39]	Using evolutionary classifier	1.Filter SMS at access layer of mobile 2.Analyzes a SMS in hexadecimal notation	It's not accurate	3	93	[19]
[41]	Independent Method	1.Independent, private and personal 2.Updatable	1.Increase hardware cost 2.does not eliminate network bandwidth consumption	8	90.17	[19]
[14]	Graph-based KNN	1.So accurate 2.Efficient 3.Small processing time	1.Sensitive to noisy data 2.It is slow if there are large number of training example	1	98.9	-
[15]	Large Cellular Network	1.So accurate 2.Efficient 3.reduces user participation in spam report 4.faster than user-driven approaches	-	3	99.4	-
[24]	Bayes Net	Facilities common to Bayes Network learning algorithms	It's just suitable for network learning algorithms	7	97.2	[19]
[43]	GentleBoost	Fast, because the number of extracted feature is small	-		98%	[19]

Moreover, two datasets was combined to form the desired dataset to evaluate this method. Actually, 425 SMS spam messages and 450 SMS ham messages from Caroline Tag's PhD thesis were chosen. Data set was divided into training and filtering sets with various proportions in three experiments and the results showed different accuracy rates. In first experiment, the accuracy of 98.29% was achieved but as the average time for classification was high and it was not practical, another proportion of training and filtering data used (20 training and 855 filtering and updating) which led to the accuracy of 90.17%. Therefore, the accuracy of the practical version of this method is 90.17% and it classifies an incoming SMS in 0.04 second. The problem of this method is that it detects spam messages on end-user so it does not eliminate the consumption of network's traffic by spam messages.

The Naïve Bayes algorithm simplifies the learning phase by assuming that features are independent. This independency assumption minimizes computational complexity and even though it is a poor presumption the algorithm still have accurate results. Moreover, this algorithm is optimal in two completely opposite situations: when features are considered

completely independent (this case is obvious as the algorithm has been designed in this way) and when features are functionally dependent (which is surprising) [30].

For choosing the best method, there are some other important factors such as cost, simplicity, applicability, and storage space. According to these factors, among these methods, DCA technique is one of the best approaches towards spam filtration to optimize performance in the SMS context. Actually, DCA algorithm has high accuracy and is very simple even though using SVM in this method might increase the processing time. The number of feature sets used for this algorithm is not large which simplifies the calculations. Implementing DCA is easy and it detects Spam messages before sending them to users and avoids traffic waste. However, the fact that the success percentage of the deterministic SQL query ranks among the top 5 intelligent methods indicates the possibility that the SMS spam could have highly balanced data with a clear pattern, because of which it should be possible to make the search much simpler and faster. However, the AI methods have a challenge in the fact that these methods are highly process intensive and also require more memory in order to store the learning data. The SQL query gives us a proper result that reveals the possibility of optimizing the Bayesian methods towards more efficiency, and simplicity or applying tokenization methods adapted to the SMS paradigm along with possible keys to identify a call-back reference.

However, the results also indicate that the best sixth algorithms achieved almost similar performance and all of them accomplished an accuracy rate more than 97% that can be considered as a very good baseline in such context. Note that, although most of them have obtained the accuracy rate greater than 90%, they have correctly filtered about only 50% of spams or even less. Therefore, based on the achieved results, we can certainly conclude that the linear SVM and Naive Bayes offer good performance for further comparison and so as DCA amplifies these two methods it is the best approach among the proposed methods.

6. Conclusion

The task of automatic filtering SMS spam still is a challenge nowadays. There are three main problems hindering the development of algorithms in this specific field of research: the lack of public and real datasets, the low number of features that can be extracted per message, and the fact that the text is rife with idioms and abbreviations. To fill some of those gaps, in this paper we described some more popular datasets and some practical and effective methods. Finally, we presented the accuracy results of different text classifiers on different datasets for spam filtering. We also assessed the strengths and weaknesses of each technique when considering its application to SMS filtering. From the survey presented, we observe that significant work has been done in the field of statistical text classification. We compared the performance achieved by several established machine learning methods, and the results indicate that SVM and Naive Bayes offer good performance and the DCA algorithm is the best one as it develops the performance of these two methods.

References

[1] Huang W. L., Liu Y. and Zhong Z. Q., "Complex network based SMS filtering algorithm", pp. 990–996, 2009.

[2] Wang C., Zhang Y. and Chen X., "A behavior-based SMS anti-spam system", IBM Journal of Research and Development, pp. 1-16, 2010.

[3] Xiang Y., Chowdhury M. and Ali S., "Filtering mobile spam by support vector machine", In Proceedings of the Third International Conference on Computer Sciences, Software Engineering, Information Technology, pp. 1–4, 2004.

[4] Healy M., Delany S. and Zamolotskikh A., "An assessment of case-based reasoning for short text message classification", In Proceedings of 16th Irish Conference on Artificial Intelligence and Cognitive Science, pp. 257–266, 2005.

[5] Duan L. Z., Li A. and Huang L. J., "A New Spam Short Message Classification", In Proceeding of the 1st International Workshop on Education Technology and Computer Science, pp. 168–171, 2009.

[6] Zheng X., Liu C. and Zou Y., "Chinese Short Messages service spam filtering based on logistic regression", Journal of Heilongjiang Institute of Technology, pp. 36–39, 2010.

[7] Cai J., Tang Y. Z. and Hu R. L., "Spam filter for short messages using Winnow", In Proceeding of the 7th International Conference on Advanced Language Processing and Web Information Technology, pp. 454–459, 2008.

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

[9] Zhang J., Li X. M. and Xu W., "Filtering algorithm of spam short messages based on artificial immune system", In Proceeding of International Conference on Electrical and Control Engineering, pp. 195–198, 2011.

[10] Junaid M. B. and Farooq M., "Using evolutionary learning classifiers to do mobile spam (SMS) filtering", In Proceedings of the 13th annual conference on Genetic and evolutionary computation, pp. 1795-1802, 2011.

[11] He P. Z., Sun Y. and Zheng W., "Filtering short message spam of group sending using CAPTCHA", In Proceeding of the 1st International Workshop on Knowledge Discovery and Data Mining, pp. 558–561, 2008.

[12] Liang Ch., Zheng Y., Weidong Zh. and Kantola R., "Implementation of an SMS Spam Control System Based on Trust Management", In Proceedings of IEEE International Conference on Green Computing and Communications and IEEE Internet of Things and IEEE Cyber, Physical and Social Computing, pp. 887-894, 2013.

[13] Liumei Zh., Jianfeng M. and Yichuan W., "Content Based Spam Text Classification: An Empirical Comparison between English and Chinese", In Proceeding of the 5th International Conference on Intelligent Networking and Collaborative Systems, pp. 69-76, 2013.

[14] Tran H., Ho-Seok K. and Sung-Ryul K., "Graph-based KNN Algorithm for Spam SMS Detection", Journal of Universal Computer Science, 2013.

[15] Nan J., Yu J., Ann S. and Zhi-Li Zh, "Understanding SMS Spam in a Large Cellular Network: Characteristics, Strategies and Defenses", In Proceeding of 16th International Symposium, pp. 328-347, 2013.

[16] Iosif A., Vasileios V. and Alexandros P., "FIMESS: FIltering Mobile External SMS Spam", In Proceeding of the 6th Balkan Conference in Informatics, pp. 221-227, 2013.

[17] Abiodun M., Oludayo O. and Sunday O., "Filtering of Mobile Short Messaging Service Communication Using Latent Dirichlet Allocation with Social Network Analysis", Transactions on Engineering Technologies, pp. 671-686, 2014.

[18] Amir K. and Lina Zh., "Improving Static SMS Spam Detection by Using New Content-based Features", Twentieth Americas Conference on Information Systems, 2014.

[19] Ahmed I, Ali R., Guan D., Lee Y., Lee S. and Chung T., "Semi-supervised learning using frequent itemset and ensemble learning for SMS classification", Expert Systems with Applications, vol.42, No. 3, pp. 1065-1073, 2015.

[20] Ala' E. and Suku N., "Semi-Synthetic Data for Enhanced SMS Spam Detection", In Proceedings of the 6th International Conference on Management of Emergent Digital EcoSystems, pp. 206-212. 2014.

[21] Abdullah J. and Ali Gh., "A Multi-agent System for Smartphone Intrusion Detection Framework", In Proceeding. of the 18th Asia Pacific Symposium on Intelligent and Evolutionary Systems, pp. 101-113 2015.

[22] Hongli W. and YH. Jiang, "SMS Spam Filtering Based on "Cloud Security", Applied Mechanics and Materials, pp. 2015-2019, 2015.

[23] Atefeh H., Mohammad ali T., Naomie S., Zahra H., "Detection of review spam: A survey", Expert Systems with Applications, vol. 42, No. 7, pp. 3634–3642, 2015.

[24] Kuruvilla M. and Biju I., "Intelligent Spam Classification for Mobile Text Message", In Proceeding of Computer Science and Network Technology, pp. 101-105, 2011.

[25] Ian H. Witten and Eibe F., "Data Mining, Practical Machine Learning Tools and Techniques", Morgan Kaufmann Publishers, Third edition, 2005.

[26] Tiago A., José G. and Akebo Y., "Contributions to the Study of SMS Spam Filtering: New Collection and Results", In Proceedings of the ACM Symposium on Document Engineering, 2011.

[27] Kuldeep Y., Ponnurangam K., Atul G., Ashish G. and Vinayak N., "SMS Assassin: Crowdsourcing Driven Mobile-based System for SMS Spam Filtering", In Proceedings of the 12th Workshop on Mobile Computing Systems and Applications, pp. 1-6, 2011.

[28] Qian X., Evan Wei X. and Qiang Y., "SMS Spam Detection Using Non-Content Features", In Proceeding of Intelligent Systems IEEE, pp. 44-51, 2012.

[29] Alper K., Serkan G., Semih E. and Efnan G., "The Impact of Feature Extraction and Selection on SMS Spam Filtering", ELEKTRONIKA IR ELEKTROTECHNIKA, vol. 19, No. 5, 2013.

[30] Rish I., "An empirical study of the Naive Bayes classifier", In Proceeding of IJCA Workshop on Empirical Methods in AI, 2001.

[31] Richard D., Peter E. and David S., "Pattern Classification", Wiley-Intersceince Publisher, 2nd Edition, 2000.

[32] Stephen M., "Machine learning An Algorithm Perspective", Chapman & Hall/CRC Publisher, 2009.

[33] Hong Z. and Wei W., "Application of Bayesian Method to Spam SMS Filtering", In Proceedings of Information Engineering and Computer Science, pp. 1-3, 2009.

[34] Freund Y. and Schapire R. E., "Large margin classification using the perceptron algorithm", In Proceedings of the 11th annual conference on Computational learning theory, pp. 209-217, 1999.

[35] Freund Y., Schapire E. and Hill M., "Experiments with a new boosting algorithm", In Proceedings of the 13th International Conference on Machine Learning, San Francisco, 1996.

[36] Cleary G. E. and Trigg L. E., "K*: An Instance-based Learner Using an Entropic Distance Measure", In Proceedings of the 12th International Conference on Machine Learning, pp. 108-114, 2008.

[37] Hassan Sh. and Mohammad Sh., "An Anti-SMS-Spam Using CAPTCHA", In Proceedings of International Colloquium on Computing, Communication, Control, and Management, 2008.

[38] Kuldeep Y., Swetank K., Ponnurangam K. and Rohit K., "Take Control of Your SMSes: Designing a Usable Spam SMS Filtering System", In Proceedings of the 13th International Conference on Mobile Data Management , 2012.

[39] Bilal J. and Muddassar F., "Using Evolutionary Learning Classifiers To Do Mobile Spam (SMS) Filtering", In Proceeding of genetic and evolutionary computation conference, 2011.

[40] Ji Won Y., Hyoungshick K. and Jun H., "Hybrid spam filtering for mobile communication", In Proceeding of Computers and Security, 2009.

[41] Taufiq N., Changmoo L. and Deokjai Ch., "Independent and Personal SMS Spam Filtering", In Proceeding of Computer and Information Technology, 2011.

[42] Ali A. Al-Hasan$_a$, El-Sayed M. El-Alfy., "Dendritic Cell Algorithm for Mobile Phone Spam Filtering", 6th International Conference on Ambient Systems, Networks and Technologies, ANT, 2015.

[43] Julie Greensmith1 and Uwe Aickelin, "The Deterministic Dendritic Cell Algorithm," Artificial Immune Systems, pp. 291-302, 2008.

[44] Akbari, F., Sajedi, H., "SMS Spam Detection using Selected Text Features and Boosting Classifiers." In Proceedings of 7th Conference on Information and Knowledge Technology (IKT), April 5-6, 2015.