SHORT MESSAGE SERVICE SPAM DETECTION USING MACHINE LEARNING TECHNIQUES

[1]Mrs.M.Mounika, [2]Mrs.R.Pallavi Reddy
[1]PG Student , mounumedari@gmail.com, [2]Assistant Professor, reddygaripallavi@gmail.com
G Narayanamma Institute of Technology & Science for Women (Autonomous)
Affiliated to JNTUH

Abstract—Short Message Service (SMS) is one of the popular communication services in which a message is sent electronically. The reduction in the cost of SMS services by telecom companies has led to the increased use of SMS. This rise attracted attackers which have resulted in SMS spam problem. A spam message is generally any unwanted message that is sent to user's mobile phone. Spam messages include advertisements, free services, promotions, awards, etc. People are using SMS messages to communicate rather than emails because while sending SMS message there is no need of internet connection and it is simple and efficient. The existing system, depends on SVMs and Apriori Algorithms for designing classifiers that filters spam SMS. However, these applications have some limitations. The number of support vectors (SV) is directly proportional to the size of the training dataset which forces SVMs to use unnecessary basis functions. SMS spam filtering based on machine learning techniques mainly concentrates to enhance SMS spam filtering. The SMS spam filtering is a combination of data mining and machine learning techniques, which are used to perform tasks like association and classification. FP growth is used for association and Naive Bayes classifier is used for classification. By using Naive Bayes and FP growth algorithms, high accuracy can be achieved.

Index Terms— SMS spam; Text Classification; Naïve Bayes; FP-Growth

1. INTRODUCTION

SMS is a text-based media which allows users of mobile phones to share a short text (usually well beyond 160 7-bit characters)[1]. In addition to the widespread use and popularity as the most important media of communication, there are many who use it for commercial purposes such as advertising media and even fraud.

The solution to this problem is by filtering text messages based on the type of content. Some common text classification techniques are Decision trees, Naïve Bayes, Rule-based inference, Neural Network, Nearest Neighbors, and Vector Machine support. This classification of SMS is different from the classification on a standard document text or e-mail because of the very short text (maximum 160 7-bit characters), lots of abbreviated texts and appears to be informal text in SMS[1]. If SMS is really short, another question arises "is the functionality good enough to differentiate between SMS spam and non-spam?". Moreover, today there are incredibly different types of SMS; therefore, a different technique is required to add functionality that can differentiate between SMS spam and non-spam. However, there is still a similar pattern in any variation of the current SMS, particularly for SMS Spam. This case may be the basis of usage, the strategy involving the presence of simultaneously evolving words as an additional function for distinguishing between SMS spam and non-spam.

The collaboration of two methods is carried out in this paper: Naïve Bayes classifier and frequent item set of the FP-Growth Algorithm.

Naïve Bayes is considered one of the learning algorithms which is highly effective and important in information retrieval machine learning. In addition, based on [3], it notes that the implementation of user-specified minimum

support will boost accuracy compared to the implementation of Naïve Bayes only. Since the minimum support gets the frequent word and is regarded not only as mutually independent, but also as single, independent and mutually exclusive[3]. It is also able to increase the score of opportunities and contributes to a more reliable classification scheme. Apriori Algorithm is performed in the referenced paper in gaining the frequent item set; on the other hand, this study conducted the FP-Growth Algorithm having better capabilities than the Apriori Algorithm[4].

2. SYSTEM DESIGN

The designed system generally consists of two phases, namely the cycle of training and testing and also the general design of the system shown in figure 1.

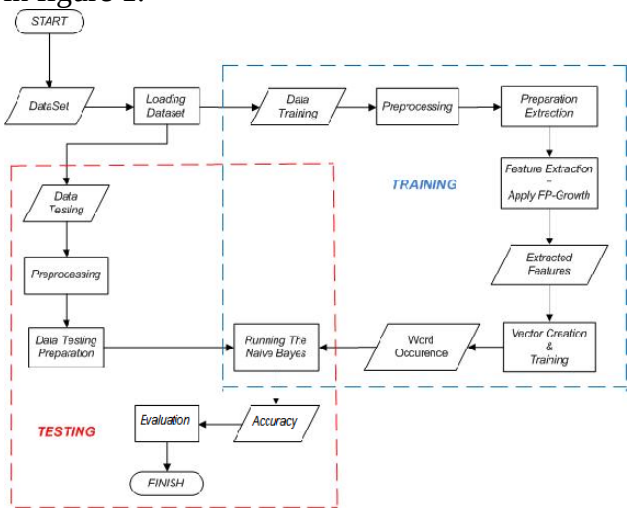

Fig1. The General Description of the System

A. Process

The process of data training is aimed at shaping the classification model. In addition, the research is the method of evaluating classification outcomes based on the model that was received.The first is pre-processing of data. To simplify the further process, the pre-processing of training and testing is done separately. Pre-processing is performed early before the training process and before the testing. The pre-processing phases shall include:

1. **1. Case Folding and Character erase**
2. The text is converted into lowercase, with the aim of homogenizing the data and deleting characters other than letters, numbers and punctuations.

2. Tokenization

Tokenization depicts passages that are divided into sentences, or sentences into individual words. Sentence Boundary Disambiguation (SBD) may be used to render a description of single sentences. That depends on NLTK's pre-prepared, clear language calculations such as the Punkt Models. By a similar method, sentences may be composed of individual words and accentuation. This most often split cross-sectionally over blank areas.Before the next processes are taken, the token process is first performed to break the string into a token or a single word; it can therefore facilitate the token search process.

3. Handle Slang Words

There are plenty of informal terms listed in slang words in the dataset. A dictionary containing the slang words is created to handle those words and is equipped with a true meaning of the words. The list of Slang terms is taken from the [5].

4. Stop word Removal

Stop words "are the most widely known words in a language such as" the," "an," "on," "is," "all. "These words do not convey significant meaning and are usually removed from writing. It is conceivable to remove stop words using Natural Language Toolbox (NLTK), a suite of libraries and projects for the preparation of emblematic and factual language characteristics.

5. Stemming

There are plenty of words in the dataset that have prefixes; thus, the stemming process is required to bring those words back into a root form. This is intended to eliminate the variations of words which seem to have the same meaning but have different types of affixes.

6. Handle Number

This process only handles the numerical character as a phone number. It is performed since there are plenty of telephone numbers appearing on the SMS dataset based on the observation, in particular belonging to a spam class; then, the telephone number may be unique feature for the classification of text messages. Then the forecast is generated in a token with a duration of > 7 for a numeric character. In addition, a character token consists of those numbers that are translated to the string "phone number" in order to homogenize all phone number data from the

numeric characters set into the same word. If the numeric charactermeet in a token,yet are unable to meet the length requirement, they will be removed.

7. Feature Extraction

In the training, the feature extraction is performed with the involvement of FP-Growth algorithm to gain the frequent item set shown in figure 2.

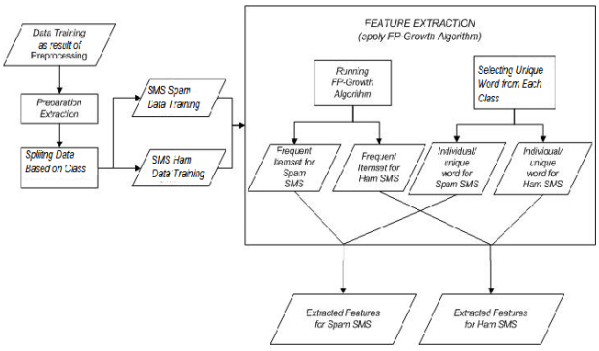

Fig2. The Process of Feature Extraction

1. Before processing, SMS content data, in word format, is converted to a numeric format via the extraction preparation process. The data is then split into each class; thus two input files are obtained for further processing.
2. The FP-Growth algorithm is performed with the minimum support listed on the report.
3. The outcome of FP-Growth process is to become the new features for each class in the classification process in the form of frequent item collection.

8. Vector Creation and Training

During this step, calculation is performed for each of the terms in each class that have been extended. The vector table is created to simplify the calculations and translated into a type of word occurrence table.

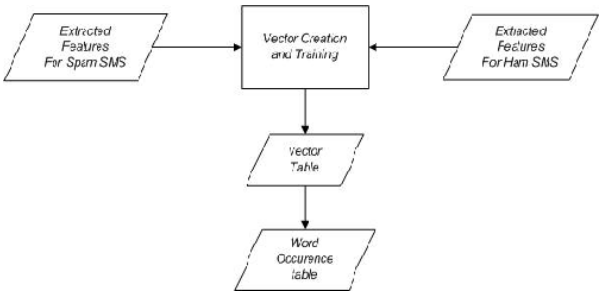

Fig3. The process of Vector Creation and Training

9. Running the Naïve Bayes System

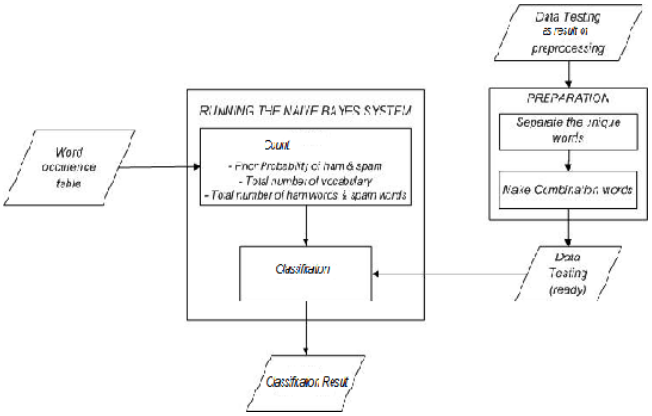

Fig4. Running the Naïve Bayes

At this point, the classification process is considered with the Naïve Bayes algorithm. From the training part, words are performed which had been counted on the word occurrence table and from the test part calculating the total and the prior probability of each class (spam and ham). Subsequently, data testing performed during the preparation process is entered to do the classification. The estimation of Laplace estimator or Laplace smoothing is applied in the classification stage in order to prevent the likelihood score of 0.

10. Evaluation(Testing)

Through the evaluation process, it can be determined whether or not the model obtained by measuring the accuracy score is feasible for implementation.The evaluation process is accomplished by taking into account precision, recall and Fmeasure. If the accuracy result reaches a high score, then the model is feasible for use in the new SMS classification process.

To measure the performance of a text classification to a term, the recall (r) and precision (p) is calculated. The Precisionis the degree of accuracy of the information requested by the user with the answers given by the system in rediscovering the information.

$P= (tp) / (tp+fp)$ and $R= (tp) / (tp+fn)$ (1)

$Fmeasure= (2xPxR)/ (P+R)$(2)

A true positive is an outcome where the model correctly predicts the positive class. Similarly, a truenegative is an outcome that correctly predicts the negative class. A falsepositive is an outcome where the model incorrectly predicts the positive class.

True positives(tp) can be interpreted as a spam message, while false positive (fp) is a ham message considered as a spam message, and false negative (fn) is a spam message that is considered as a ham message.

Once the precision and recall is obtained, the accuracy can be calculated. Accuracy is defined as the degree of closeness between the predicted score and the actual score.

$$Accuracy=(tp+tn)/(tp+tn+fn+fp) \quad (3)$$

The next model (based on calculating the word occurrence) is the process of a new predictive classification of SMS after accuracy is based on the training and testing process. The latest SMS data in this process passes the stage of preprocessing and preparation as applied to the data check.

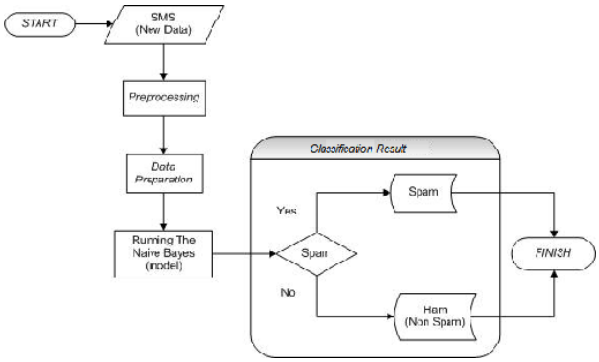

Fig5. New SMS Prediction Process

3. METHODOLOGY

1.Initially the dataset is loaded and divides the data into training and testing.
2. In training part, preprocessing techniques are used to remove the unwanted data.
3. The features are extracted by using FP-Growth algorithm.
4. Vector creation is applied on the training data.
5. The word occurrence table is created.
6. In testing part, preprocessing and data testing preparation is done.
7. Finally Naïve Bayes algorithm is applied on training and testing data to achieve accuracy.

FP-Growth Algorithm:

FP-growth algorithm finds frequent item sets or pairs, sets of things that commonly occur together, by storing the dataset in a special structure called an FP-tree.

FP-Growth algorithm proposed by Han, is an efficient and scalable method for mining the complete set of frequent patterns by pattern fragment growth, using an extended prefix-tree structure for storing compressed and crucial information about frequent patterns named frequent-pattern tree (FP-tree).

The FP-growth algorithm scans the dataset only twice. The basic approach is to find frequent item sets using the FP-growth algorithm is as follows:
1 Build the FP-tree.
2 Find frequent item sets fromthe FP-tree.
The FP stands for “Frequent Pattern.” An FP-tree looks like other trees in computer science, but it has links connecting similar items. The linked items can be thought of as a linked list. The FP-tree is used to store the frequency of occurrence for sets of items. Sets are stored as paths in the tree set.

Algorithm 1: FP-tree construction

Input: A transaction database DB and a minimum support threshold.
Output: FP-tree, the frequent-pattern tree of DB.
Method: The FP-tree is constructed as follows.

- Scan the transaction database DB once. Collect F, the set of frequent items, and the support of each frequent item. Sort F in support-descending order as FList, the list of frequent items.
- Create the root of an FP-tree T, and label it as “null”. For each transaction Trans in DB do the following:
- Scan the transaction database DB once. Collect F, the set of frequent items, and the support of each frequent item. Sort F in support-descending order as FList, the list of frequent items.
- Create the root of an FP-tree, T, and label it as “null”. For each transaction Trans in DB do the following:
- Select the frequent items in Trans and sort them according to the order of FList. Let the sorted frequent-item list in Trans be [p | P], where p is the first element and P is the remaining list. Call insert tree ([p | P], T).
- The function insert tree ([p | P], T) is performed as follows.

IfT has a child N such that N.item-name = p.item-

Name, then increment N value by 1; else create a node N, with its count initialized to 1, its parent link linked to T, and its node-link linked to the nodes with the same item-name via the node-link structure. If P is nonempty, call insert tree (P, N) recursively.

Algorithm 2: FP-Growth

Procedure FP-Growth(Tree, a) {

(1)if Tree contains a single prefix path then { // Mining
single prefix-path FP-tree}
(2) let P be the single prefix-path part of Tree;
(3) let Q be the multipath part with the top branching node
replaced by a null root;
(4) for each combination (denoted as ß) of the nodes in the
path P do
(5) Generate pattern ß ∪ a with support = minimum support
of nodes in ß;
(6) let freq pattern set(P) be the set of patterns so generated;
}
(7) else let Q be Tree;
(8) for each item ai in Q do { // Mining multipath FP-tree
(9) Generate pattern ß = ai ∪ a with support = ai .support;
(10) Construct ß's conditional pattern-base and then ß's
conditional FP-tree Tree ß;
(11) if Tree ß ≠ Ø then
(12) call FP-growth(Tree ß , ß);
(13)let freq pattern set(Q) be the set of patterns so
generated;}
(14) return(freq pattern set(P) ∪ freq pattern set(Q) ∪ (freq
pattern set(P) × freq pattern set(Q)))}

Example for FP-Growth

Given Message : "Hello hail Hai"

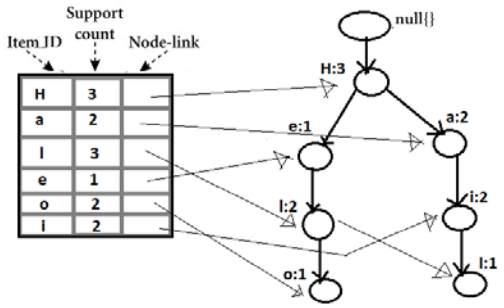

Fig 6. Example for FP-Growth

Naive Bayes Algorithm:

Naïve Bayes approach is used to filter incoming SMS of unknown origin. It is one of the simplest probabilistic classifiers, with a strong naive assumption of independence based on the Bayes theorem. This presumption treats every word as entity, mutually exclusive and independent. Suppose all attributes are X1,., Xn is conditionally independent. This assumption dramatically simplifies and reduces the complexity and representation of P(X|Y)as well as the problem of estimating it fr om the training data.

$$P(spam \mid words) = \frac{P(words \mid spam)P(spam)}{P(words)}$$

- Start
- Collect SMS from different incoming messages.
- Assume all the attributes X1... Xn are conditionally and mutually independent given Y.
- Considering the case where X = (X1, X2). P(X|Y) =P(X1,X2|Y)=P(X1|X2,Y)P(X2|Y)=P(X 1|Y)P(X2|Y)
- This can be represented as P (X1|Y) = =P (Xi|Y)
- Calculate the probability that Y can take k_{th} possible value
- Classify the unknown incoming SMS.

4. IMPLEMENTATION

4.1. Dataset

The dataset derived from Corpus of SMSconsists of 5574 SMS. In 5574 messages 4,827 ham messages and 747 spam SMS.

4.2. Pre-Processing of data

Pre-Processing includes Case Folding Dan Character Erase, Tokenization, Handle Slang words, Stop Words Removal and Stemming.

4.2.1. Case Folding dan Character Erase

All the text is converted into lowercase aiming to homogenize the data and delete characters other than letters, numbers and removepunctuation.

4.2.2. Tokenization

Tokenization depicts parting passages into sentences, or sentences into individual words.

4.2.3. Handle Slang Words

In the dataset, there are plenty of informal words referred to in slang words. To handle these words, a dictionary is created containing the slang word list.

4.2.4. Stop Word Removal

"Stop words" are the most widely recognized words in a language like "the", "an", "on", "is", "all". These words don't convey significant importance and are typically expelled from writings. It is conceivable to expel stop words utilizing Natural Language toolbox (NLTK), a suite of libraries and projects for emblematic and factual characteristic language preparing.

4.2.5. Stemming

In the dataset there are plenty of words that have prefixes; the process of stemming is therefore necessary to carry back those words into a root form. It is intended to alleviate the variations of the words that should have the equal meaning yet have different affixes.

5. TEST RESULTS

SMS Spam detection uses the dataset which is amounted to 5574 SMS.It consists of 4827 ham SMS, and 747 spam SMS.

In the testing procedure, Naïve Bayes with FP-Growth SMS spam collection dataset always produce an f-measure score with higher accuracy. It means that the system is more appropriate in performing the classification. Moreover FP-Growth is able to significantly elevate the score of precision. Thus the system is more precise in answers to the information requested by the user. Even though the recall score is inversely smaller. However it has another advantage, since if there is a text that has unknown features previously in advance by training, the class will tend to be classified into a ham. Otherwise it can filter Spam SMS.

Naive Bayes and FP-Growth algorithms are used in SMS spam detection, the performances of both methods is equally well for SMS classification with average accuracy above 90%. The use of collaboration methods, Naive Bayes and FP-Growth, is superior to the average accuracy for each dataset. FP-Growth with Naïve Bayes has accuracy up to 98.506%.

6. CONCLUSION and FUTURE WORK

The implementation of minimum support facilitates the problems dealing with limited features due to the limited number of characters in SMS; it therefore produces the new features to differentiate between spam SMS and ham SMS. The use of datasets with varied training data is agreeable to be applied by using the FP-Growth. By implementing the FP-Growth for feature extraction, it can elevate the score of precision. Thus, the system becomes more precise in providing the information requested by the users in response to the SMS classification.

In future, SMS Spam filtering problem can be solved using other machine learning and data mining techniques, which may process in less time with more accuracy. For more accurate results, a good data set should also be taken into consideration. In future, it can be recommended to use a general methodology which relies on a variety of non-linguistic characteristics such as SMSC originator, Reply Path, HTTP links, Mobile Station International ISDN Number (MSISDN), and Protocol Identifiers such as TP-PID of the mobile text messages in order to decide if a message is spam or non-spam.

REFERENCES

[1].Dt.fee.unicamp.br, "You Tube Spam Collection".
[Online].Available:http://www.dt.fee.unicamp.br/~tiago/SMSspamcollecti on/. [Accessed: 12-Mar- 2015].

[2] Han, Jiawei, Jian Pei, and Yiwen Yin. “Mining frequent patterns without candidate generation.” ACM sigmod Record. Vol.29. No.2. ACM, 2000.

[3] Han, Jiawei, Micheline Kamber, and Jian Pei. Data mining: concepts and techniques 3rd Edition. Morgan Kaufmann Publishers, 2013.

[4].Hidalgo, "SMS Spam Corpus v.0.1", Esp.uem.es.
[Online].Available:http://www.esp.uem.es/jmgo

mez/SMSspamcorpus/. [Accessed: 12- Mar- 2015]
[5].http://abcnews.go.com/blogs/technology/2012/08/69-of-mobile-phone-users-get-text-spam/.
[6]Noslong.com, “Slang Dictionary- Text Slang and Internet Slang Words available: http://www.noslang.com/dictionary/. [Accessed: 23- Apr- 2015].
[7] Qian, Wang, Han Xue, and Wang Xiaoyu. "Studying of classifying junk messages based on the data mining." Management and Service Science, 2009. MASS'09. International Conference on. IEEE, 2009.
[8] Ranks.nl, "Stopwords". [Online]. Available: http://www.ranks.nl/stopwords. [Accessed: 23- Apr- 2015].
[9] Shirani-Mehr, Houshmand. "SMS spam detection using machine learning approach." (2013): 1-4.
[10].Snowball.tartarus.org, "Snowball - Download". [Online]. Available: http://snowball.tartarus.org/download.php. [Accessed: 26- Apr- 2015].
[11]. Spam - Definition and More from the Free Merriam-Webster Dictionary. Merriam-webster.com. 2012-08-31. Retrieved 2013-07-05.
[12].Text4ever. White Paper: UK spam study, Oct. 2009. http://www.txt4ever.com/study/spamstudy.pdf