Enhancing Spam Detection on Mobile Phone Short Message Service (SMS) Performance using FP-Growth and Naive Bayes Classifier

Dea Delvia Arifin[1,], Shaufiah[2] and Moch. Arif Bijaksana[3]
School of Computing
Telkom University
Bandung, Indonesia
[1]delvia.dea@gmail.com, [2]shaufiah@telkomuniversity.ac.id , [3]arifbijaksana@telkomuniversity.ac.id

Abstract— **SMS (Short Message Service) is still the primary choice as a communication medium even though nowadays mobile phone is growing with a variety of communication media messenger applications. However, nowadays along with the SMS tariff reduction leads to the increase of SMS spam, as used by some people as an alternative to advertise and fraud. Therefore, it becomes an important issue as it can bug and harm the users and one of its solution is with automatic SMS spam filtering. One of most challenging in SMS spam filtering is its accuracy. In this research we proposed to enhanced SMS spam filtering performance by combining two of data mining task association and classification. FP-growth in association is utilized for mining frequent pattern on SMS and Naive Bayes Classifier is used to classify whether SMS is spam or ham. Training data was using SMS spam collection from previous research. The result of using collaboration of Naive Bayes and FP-Growth performs the highest average accuracy of 98, 506% and 0,025% better than without using FP-Growth for dataset SMS Spam Collection v.1, and improves the precision score; thus, the classification result is more accurate.**

Keywords— SMS spam; Naïve Bayes; FP-Growth; text classification

I. Introduction

SMS is a text-based communication media that allows mobile phone users to share a short text (usually it is well beyond 160 characters in 7-bit) [1]. Along with the widespread use and popularity as the most important communications media, there are plenty of those who use it for commercial purposes such as advertising media and even fraud. The reduced SMS rate is one of the causes of increasing SMS spam as well, as in China, the SMS tariff is well under than $ 0.001 [1]. Moreover, based on the Korea Information Security (KISA), this amount exceeds the email spam. For instance, mobile users in the US gains 1.1 billion SMS spams, and Chinese users receives 8,29 billion SMS spams weekly [2].

There is a solution that could be performed to solve the above problems. It is by filtering SMS based on the text classification. There are some popular text classifications techniques, including decision trees, Naive Bayes, rule induction, neural network, nearest neighbors, and Support Vector Machine. Nevertheless, this SMS classification is different from the classification on a regular document text or e-mail due to the very short text (160 7-bit characters maximum), plenty of abbreviated texts, and tend to be informal text in SMS [1]. If SMS is very short, it causes another question "is the feature is well enough to distinguish between SMS spam and non-spam?”. In addition, today, types of SMS are going exceedingly various; hence, another technique is necessary in order to add features being able distinguish between SMS spam and non-spam. However, every variation of the existing SMS has still a similar pattern, in particular for SMS spam. That case can be the base to use the technique involving the appearance of the words emerging simultaneously as an additional feature to distinguish between SMS spam and non-spam.

In this experiment, a collaboration of two methods is carried out: Naive Bayes classifier and FP-Growth Algorithm frequent itemset. Naive Bayes is regarded as one of the learning algorithm which is very effective and important for machine learning in information retrieval. Besides, based on the referenced paper [3], it states that user-specified minimum support implementation can enhance the accuracy compared to only the Naive Bayes implementation. Since the minimum support obtaines the frequent itemset as an additional feature, therefore every frequent word is considered not only mutually independent, but also single, independent and mutually exclusive [3]. Furthermore, it is able to raise the score of opportunities and leads to more accurate system in classification. In the referenced paper, Apriori Algorithm is performed in gaining the frequent itemset; on the other hand, this study carries out the FP-Growth Algorithm having better capabilities than Apriori Algorithm [4].

II. Design of System

The constructed system generally consists of two phases, namely the training and testing process and general design of the system shown on figure 1.

A. Process

It is the data training process aiming to form the classification model. Besides, the testing process is the process to test the results of the classification based on the model that has been obtained.

The first process is preprocessing data. The Preprocessing of the training data and testing is separately performed to simplify the further process. Preprocessing is carried out on the early stage before the process of training (using training data) and before the process of testing (using the testing data). The phases being taken in the preprocessing is including:

1. Case Folding dan characters erase

   All text is converted to lowercase aiming to homogenize the data and delete the characters other than letters, numbers and punctuation remove.
2. Tokenization

   Prior to the next processes are taken (from process 3 to process 6), token process is carried out first to break the string to be a token or an individual word; hence, it can facilitate in the process of token search.
3. Handle Slang Words

   In the dataset, there are plenty of informal words referred to in slang words. To handle those words, a dictionary is created containing the slang words are equipped with a real sense of the words. The Slang dictionary word list is taken from the [5].
4. Stopword Removal

   In order to delete the words that belong to the stopwords, the word-dictionary matching technique is performed consisting of stopwords lists taken from the site [6].
5. Stemming

   In the dataset, there are plenty of words that have prefixes; the process of stemming is therefore necessary to carry back theose words into a root form. It is intended to alleviate the variations of words that should have the equal meaning yet have different affixes shapes. This process performs a Snowball Tartarus library that implements the Porter Stemmer algorithm, taken from [7].
6. Handle Number

This process merely handles the numeric characters in the form of a phone number. It is carried out since based on the observation, there are plenty of telephone numbers turning up on the SMS dataset, in particular belonging to a class of spam; then, the phone numbers may be a unique feature for SMS text classification. Therefore, the provision is created for a numeric character in a token with a length of ≥ 7 (the length of a minimum standard telephone number). Moreover, a character token consisting of those numbers is converted to "phonenumber" string to homogenize all the data of the phone numbers from the numeric characters set into a same word. If the numeric characters meet in a token yet are unable to meet meet the length requirement, they will be removed.

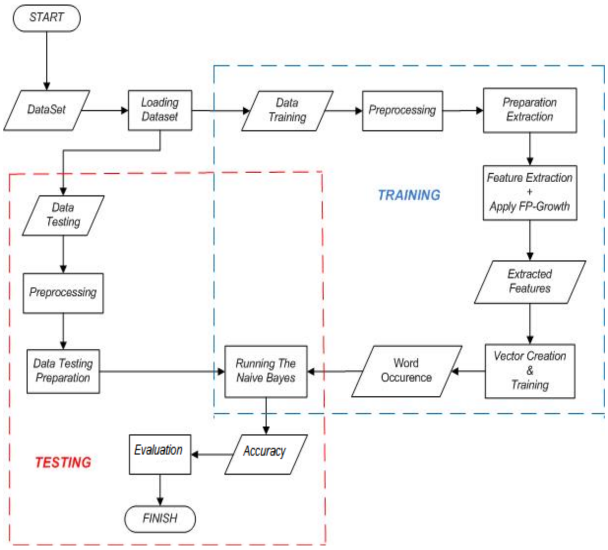


Fig. 1. The general description of the system

B. Feature Extraction

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

.

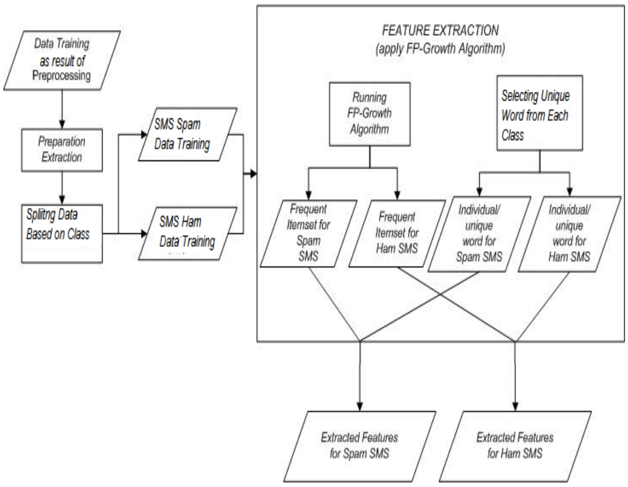

Data Training as result of Preprocessing
Preparation Extraction
Spliting Data Based on Class
SMS Spam Data Training
SMS Ham Data Training
FEATURE EXTRACTION (apply FP-Growth Algorithm)
Running FP-Growth Algorithm
Selecting Unique Word from Each Class
Frequent Itemset for Spam SMS
Frequent Itemset for Ham SMS
Individual/ unique word for Spam SMS
Individual/ unique word for Ham SMS
Extracted Features for Spam SMS
Extracted Features for Ham SMS

Fig. 2. The process of feature extraction

Here are the Explanations of Fig. 2. is as follow:

1. Prior to be processed, the data of SMS contents, in the format of the word, is converted into a numeric format by means of the process of extraction preparation. Afterwards, the data is separated into each class; hence, two input files are gained for further processing.
2. FP-Growth running is carried out with minimum support specified on the testing,
3. The results of the FP-Growth process is in the form of frequent itemset becoming the new features for each class in the classification process,

4. The New features are combined with individual features of

C. *Vector Creation and Training*

In this process, the calculation for each of the word that has been extracted in each class is performed. To simplify the calculations, the vector table is created and converted to a form of word occurrence table.

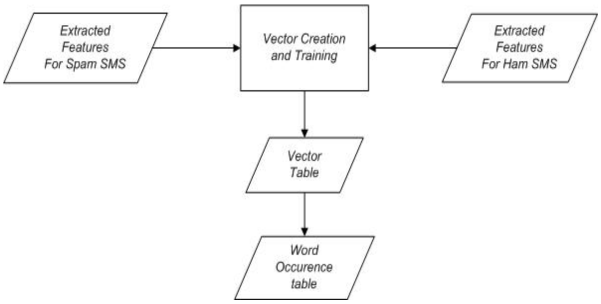

Extracted Features For Spam SMS
Vector Creation and Training
Extracted Features For Ham SMS
Vector Table
Word Occurence table

Fig. 3. The process of vector creation and training

Vector table is a table that shows the appearance of features or words in each SMS sentence. Meanwhile, the word occurrence table is a table that contains the number of occurrences of all words contained in each class.

D. *Running The Naive Bayes System*

In this stage, the process of classification with Naive Bayes algorithm is started. Words that had been counted on word occurrence table, the calculating of the total and the prior probability of each class (spam and ham) are performed. Afterwards, data testing that has been carried out the process of preparation is entered in order to do the classification. In the classification stage, the calculation of Laplace estimator or Laplace smoothing is applied to avoid the 0 probability score. Fig. 4. is an overview of the classification with Naive Bayes.

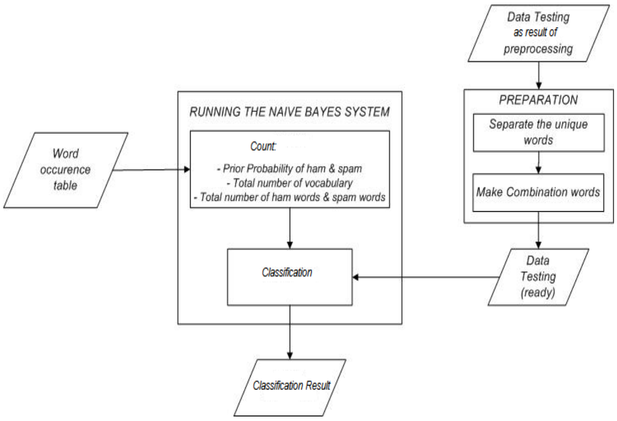

Data Testing as result of preprocessing
RUNNING THE NAIVE BAYES SYSTEM
PREPARATION
Separate the unique words
Make Combination words
Word occurence table
Count:
- Prior Probability of ham & spam
- Total number of vocabulary
- Total number of ham words & spam words
Classification
Data Testing (ready)
Classification Result

Fig. 4. running naive bayes

E. *Evaluation (Testing)*

Through the evaluation process, it can be known whether the model that has been obtained is feasible to be implemented or not by calculating the score of accuracy. It means that the model is the results obtained after the training

each class.

process in the form of scores such as prior probability, the number of acquired vocabularies and the number of words gained in each class. If the result of the accuracy attains a high score, the model is feasible to be utilized in the classification process of a new SMS.

The way how to effectively measure the performance of a text classification to a term is by measuring the recall (r) and precision (p). The Precision is the degree of accuracy of the information requested by the user with the answers given by the system. In the meantime, the recall is the success rate of the system in rediscovering information.

$$p = \frac{tp}{tp+fp} \qquad r = \frac{tp}{tp+fn} \tag{1}$$

$$F\ measure = \frac{2 x\, p\, x\, r}{p+r} \tag{2}$$

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

Once the definition of any precision and recall is obtained, the accuracy may be calculated. Accuracy is defined as the degree of closeness between the predicted score and the actual score, formulated as follows:

$$\text{Accuracy} = \frac{tp+tn}{tp+tn+fp+fn} \tag{3}$$

After having obtained the good accuracy based on the training and testing process, the next model (based on the calculation of the word occurrence) can be used in the process of a new SMS predictive classification. In this process, the new SMS data still passes preprocessing and preparation stage as applied to the data testing. Afterwards, it is classified to determine between the SMS spam or ham.

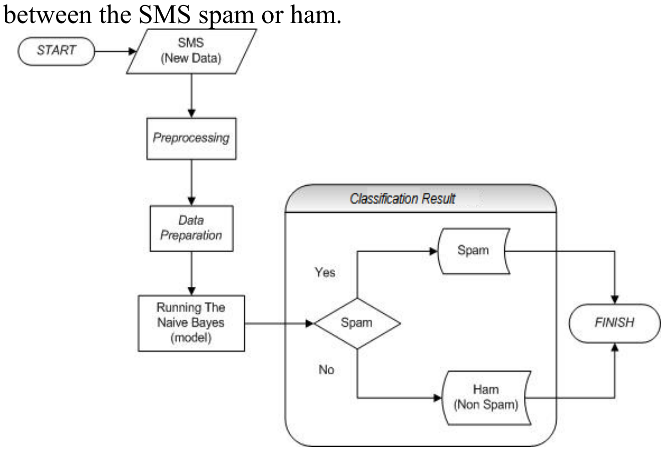


Fig. 5. new SMS prediction process

III. TESTING

A. *Dataset*

To identify the performance of SMS filtering system built in this research, the dataset derived from Corpus SMS is used as follow SMS Spam Collection v.1 [8] with SMS amounted to 5574 SMS consisting of 4,827 ham SMS and 747 spam SMS, as well as from SMS Spam Corpus v .0.1 Big [9] with SMS amounted to 1324 SMS consisting of 1002 ham SMS and 322 spam SMS.

B. The Testing Process

In this test, 6 scenarios of testing process is carried out as follow:

1. Naive Bayes Testing without FP-growth using SMS Spam Corpus v.0.1 Big dataset
2. Naive Bayes Testing with FP-growth using SMS Spam Corpus v.0.1 Big dataset
3. Naive Bayes Testing without FP-growth using SMS Spam Collection dataset v.1
4. Naive Testing Bayes with FP-growth using SMS Spam Collection dataset v.1
5. Naive Bayes Testing without FP-growth using both datasets
6. Naive Bayes Testing with FP-growth using both datasets
7. Naive Bayes Testing with FP-growth based on the characteristics of the dataset

C. The Result of the Testing

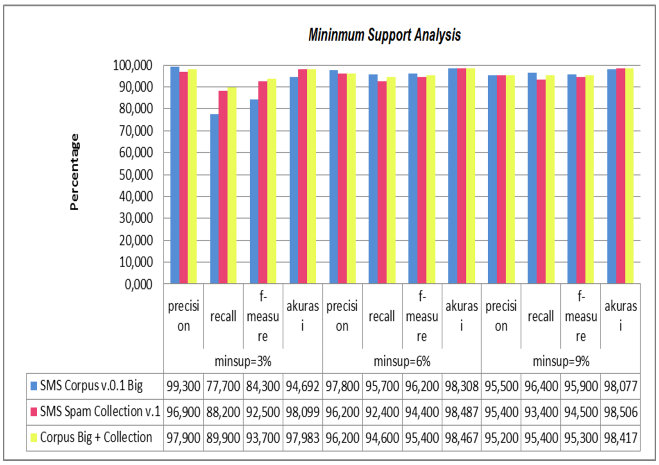
Based on the 1st until the 6th testing process, the analysis of the application of the optimal minimum support and comparative analysis of the use of Naive Bayes merely with the use of the Naive Bayes and FP-Growth collaboration are performed.

In the 7th testing process the analysis of the data characteristics suitable to be implemented by using the method of FP-Growth is then performed.

1) The Analysis of minimum support on FP-Growth

The application of minimum support score in each dataset has different results. From the obtained test results, the optimal minimum support results in each dataset are analysed. Fig. 6. is presented to simplify the analysis process.

The Fig. 6 seen in the SMS Corpus v.0.1 Big produces the highest precision current score of 3% minimum support. It is carried out due to a great number of new features produced particularly for this class of spam; therefore, the level of system accuracy in providing answers to the use of the requested information goes up. Nevertheless, the result is inversely contradictive with the results of his recall that produces the lowest score. As seen on the figure 6 that represents the 3% minimum support for SMS data Corpus v.0.1 Big, the feature reaching thousands spam SMS is produced. Based on the formula of word opportunity, the number of occurrences of the word in a class is inversely proportional to the results of the word opportunity on that class; thus, the greater the number of occurrences of words in a class are, the smaller chance the word opportunity enters the class. Besides, the composition of a smaller amount of spam data is approximately 1/3 of the ham data; therefore its prior probability score is clearly smaller, and the smaller posterior probability score will be generated to be classified in the class of spam. It causes a lot of spam SMS which is incorrectly classified. In the meantime, in SMS Corpus Big v.01, the most optimal minimum support is obtained at the time by 6% to 98.308% accuracy results.



	minsup=3%				minsup=6%				minsup=9%			
	precision	recall	f-measure	akurasi	precision	recall	f-measure	akurasi	precision	recall	f-measure	akurasi
■ SMS Corpus v.0.1 Big	99,300	77,700	84,300	94,692	97,800	95,700	96,200	98,308	95,500	96,400	95,900	98,077
■ SMS Spam Collection v.1	96,900	88,200	92,500	98,099	96,200	92,400	94,400	98,487	95,400	93,400	94,500	98,506
■ Corpus Big + Collection	97,900	89,900	93,700	97,983	96,200	94,600	95,400	98,467	95,200	95,400	95,300	98,417

Fig. 6. The minimum support analysis

The use of SMS Spam Collection dataset always produces the higher accuracy score than the other dataset uses. It shows that the three parameters of minimum support are reasonable to be used in this dataset. In this Collection Spam dataset, the highest accuracy is obtained when the 9% minimum support is equal to 98.506%.

Meanwhile, the use of both combined datasets generates nearly the same accuracy with the use of SMS Spam Collection dataset, it may be due to any number of used datasets closely has the same quantity; hence, when it is applied to the three parameters of minimum support, it has significant similarities. In this dataset, the highest accuracy is obtained when the 6% minimum support is equal to 98.467%.

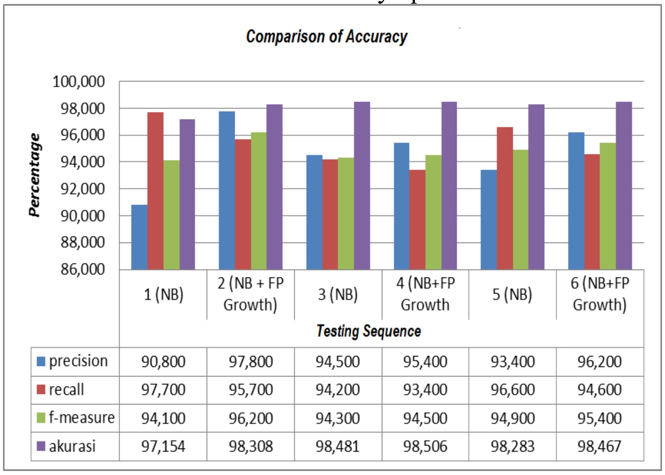
The conclusion of the testing with the use of three datasets is that the quantity and quality of the dataset is influential in producing a good score. The quantity of the dataset use is inversely proportional to the quantity of minimum support parameters score. In the meantime, for the small quantity data, such as the SMS Corpus Big v.0.1, it is not recommended to use the very small minimum support score since it will produce a great amount of the worst new feature. On the other hand, for very huge data such as a combination of the two datasets, it is necessary to need the smaller parameter score in order to gain the optimal new features. Since if the minimum support is very huge, the new features will be undelivered at all and it is failure to raise up the score of accuracy.

2) The Comparative analysis of both methods

Accuracy rate of the test results in both methods (by FP-Growth and without FP-Growth) will be compared by using the average evaluation score.

Based on test results shown in fig. 7, it is proved that by applying the method of FP-Growth collaborated with Naive Bayes always produce an f-measure score and 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 providing answers to the information requested by the user. Even though the recall score is inversely smaller, it is proper due to a documents composition ratio is larger than that of ham. 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. Furthermore, it can protect the SMS filtered into spam SMS if it is known as an important SMS.

Therefore, based on the testing, each dataset use produces the superior accuracy by implementing the FP-Growth, SMS Corpus v.0.1 Big raises up the accuracy up to 1.154%, Spam Collection SMS increases the accuracy up to 0.025%, and the use of both datasets elevates the accuracy up to 0.184%. Meanwhile, the highest accuracy is obtained on SMS Spam Collection v.1 dataset with accuracy up to 98.506%.



	1 (NB)	2 (NB + FP Growth)	3 (NB)	4 (NB+FP Growth	5 (NB)	6 (NB+FP Growth)
🟦 precision	90,800	97,800	94,500	95,400	93,400	96,200
🟥 recall	97,700	95,700	94,200	93,400	96,600	94,600
🟩 f-measure	94,100	96,200	94,300	94,500	94,900	95,400
🟪 akurasi	97,154	98,308	98,481	98,506	98,283	98,467

Fig. 7. accuracy comparison

IV. Conclusion

Based on the analysis of the tests performed in this research, it can be concluded that:

1. Both methods used in this research, the performances of both methods is equally well for SMS classification with average of the accuracy above 90%. The use of collaboration methods, Naive Bayes and FP-Growth, is superior to the average accuracy for each dataset. On the spam Corpus v.0.1 Big SMS, it excels 1.154%, on Spam Collection SMS, it excells 0.025% and the combined dataset excells 0.184% ,
2. The Accuracy best average is obtained when the SMS Spam Collection v.1 dataset with the 9%minimum support is used and the implementation of the FP-Growth has an accuracy up to 98.506%.
3. 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.
4. The use of datasets with varied training data is agreeable to be applied by using the FP-Growth.
5. The provision of minimum support parameter score is inversely proportional to the dataset quantity. For the heavier data, the smaller minsup is used in order to obtain more optimal new features (if minsup is too big, it is unable to produce new features). In the meantime, for the lower dataset, the bigger minsup is used (if minsup is too small, it is unable to gain the lower and effective new features)
6. 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.

References

[1] Shirani-Mehr, Houshmand. "SMS spam detection using machine learning approach." (2013): 1-4.

[2] 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.

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

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

[5] Noslang.com, "Slang Dictionary - Text Slang & Internet Slang Words", 2015. [Online]. Available: http://www.noslang.com/dictionary/. [Accessed: 23- Apr- 2015].

[6] Ranks.nl, "Stopwords". [Online]. Available: http://www.ranks.nl/stopwords. [Accessed: 23- Apr- 2015].

[7] Snowball.tartarus.org, "Snowball - Download". [Online]. Available: http://snowball.tartarus.org/download.php. [Accessed: 26- Apr- 2015].

[8] Dt.fee.unicamp.br, "YouTube Spam Collection". [Online]. Available: http://www.dt.fee.unicamp.br/~tiago/SMSspamcollection/. [Accessed: 12- Mar- 2015].

[9] Hidalgo, "SMS Spam Corpus v.0.1", Esp.uem.es. [Online]. Available: http://www.esp.uem.es/jmgomez/SMSspamcorpus/. [Accessed: 12- Mar- 2015].