Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 113 (2017) 154–161

Procedia Computer Science

www.elsevier.com/locate/procedia

The 8th International Conference on Emerging Ubiquitous Systems and Pervasive Networks (EUSPN 2017)

SMS Spam Detection using H2O Framework

Dima Suleiman[a,b,*], Ghazi Al-Naymat[a]

[a] COMPUTER SCIENCE DEPARTMENT KING HUSSEIN FACULTY OF COMPUTING SCIENCES PRINCESS SUMAYA UNIVERSITY FOR TECHNOLOGY AMMAN, JORDAN

[b] BUSINESS INFORMATION TECHNOLOGU DEPARTMENT THE UNIVERSITY OF JORDAN AMMAN, JORDAN

Abstract

SMS spams are one of the concerns and many people do not like to receive them since they are annoying. Many SMS spam detection methods already exist and different classifiers were used, such classifiers depended on Support Vector machine, Naïve Bays and many other machine learning algorithms. In this paper, new classifier is proposed which depends mainly on using H2O as platform to make comparisons between different machine learning algorithms. Moreover, Machine learning algorithms that are used for comparisons are random forest, deep learning and naïve bays. In addition to using deep learning and random forest as classifiers, they are also used to determine the most important features that can be used as input to random forest, deep learning and naïve bays classifiers. Experimental results show that the most significant features that can affect the detection of SMS spam are the number of digits and existing of URL in SMS text. The dataset that is used in experiment is the one proposed by UCI Machine Learning Repositories. Therefore, experiments show that the faster algorithm that achieves high performance is naïve bays with runtime 0.6 seconds, however after comparing it with deep learning and random forest it has the lowest precision, recall, f-measure and accuracy. On the other hand, random forest is the best in term of accuracy with 50 trees and 20 maximum depths, where precision, recall, f-measure and accuracy are 96%, 86%, 91% and 0.977% respectively; nevertheless the runtime is high 30.28 seconds.

Peer-review under responsibility of the Conference Program Chairs.

Keywords: SMS spam, Random Forest, Naïve Bays, Deep Learning, H2O.

*Corresponding author. Tel.: +962795016922.
E-mail address: dimah_1999@yahoo.com

1877-0509
Peer-review under responsibility of the Conference Program Chairs.
10.1016/j.procs.2017.08.335

1. Introduction

SMS (Short Message Service) is one of popular ways of communication between millions of people, where transmission of messages must occur according to communication standard protocols. Therefore, there is a need for text classification algorithms that can be used in classifying the messages either to ham or spam messages. While ham messages are the one that is created by legitimate users, spam messages are not desirable. Thus spam messages must be detected and removed once they arrived to mobile station, example of spam messages are the ones created by promotional companies. In addition to the fact that SMS spam are annoying, they are also consuming time, resources, money and network bandwidth, nevertheless the availability of spam filtering software for detecting SMS spam are limited. On the other hand, there is another problem that may be generated when ham messages are removed and blocked when they are misclassified as spam[1].

Email and SMS spam are both annoying to users and may cause performance degradation of the service[2], SMS spam usually reach group of people and broadcasted through network of mobile, however World Wide Web transfers email spam. However, different solutions of SMS spam detection were inherited from email spam filtering and classification methods[3]. Moreover, there are various difficulties that face researchers of SMS spam detection such as the limitation of publicly availability dataset. In addition, SMS spams have large variety and size of existing dataset is still small, also there are huge efforts to generate the dataset. From other side, another issue that is related to SMS spam detection is that not all methods of email spam filtering performing good with SMS spam[1].

There are various techniques used for SMS spam detection, such as using support vector machine (SVM), k-nearest neighbour (KNN), naïve Bayes (NB), artificial neural network, decision tree and random forest, in addition to hybrid methods[4]. However different comparisons and experiments were made between different techniques using different datasets; the results showed that classifiers that use SVM and NB provided highest accuracy[5], however techniques such as decision tree, logistic regression and Bayesian classification still suffer from time consuming[5] problem. Most of existing researches used Weka for their experiments[6,7].

In this paper, the authors propose new SMS spam detection method that will be based on using H2O as platform, however there are no other research papers that used H2O as platform, thus most of the researches used Weka. Comparisons will be made between different machine learning algorithms: NB, KNN and deep learning. Moreover, the matrices that will be used for evaluating the model are the accuracy, precision, recall, f-measure in addition to measuring the time efficiency. In addition, the dataset that will be used for experiments is the same dataset that is available in UCI Machine Learning Repositories. Consequently, the dataset will be explored and Python will be used to make preprocessing.

The rest of this paper will be organized as following: Section 2 will cover the literature and related work. Section 3 will describe the background which will include information about H2O frames work and the classification methods that will be used. Section 4 will explain the proposed framework which contains the selected dataset, feature selection and extraction, and evaluation metrics. Experiential results will be discussed in section 5. Finally conclusion and future work will be presented in section 6.

2. Related Work

Filtering system was used to address the problems of mobile SMS spam; however performance of spam detection is very important and must reach adequate level by making certain adaptation on filtering techniques. In their research[8] they tried to increase the performance of SMS spam detection by applying the same filter used in email spam filters which achieved the highest performance. Two datasets were used for testing, one for English and another for Spanish. English dataset consists of 1002 hams and 82 spams, while Spanish contains 1157 legitimate and 199 spams. Most of machine learning algorithms that were applied in vector representation of messages used certain features such as words, lowercase words, bigram and trigram of characters and words bigrams.

Email filtering algorithms became underperformed when used with SMS spam, the reasons of this refers to different reasons: messages with limited features, there was no real database for SMS spam, the informal language of the messages and the short length of it. Therefore, a real database of SMS spams was created in 2011 UCI Machine Learning repository, this database was created and made publicly available[1]. Consequently, many experiments were made in this dataset and comparisons between different machine learning algorithms take place

in[1,9]. Moreover, in[1] two tokenizers were used, the first one used to split the words in patterns that contains dots, commas, or colons at the middle such as email, and the second tokenizer splitted the sequence of characters separated by commas, blanks, dots and others. However, no pre-processing was used such as stop word removal or stemming since many researchers found that such pre-processing may affect the accuracy. On the other hand, comparisons between many classifiers were made and all results showed that linear SVM is the best. Also, in[9] the same as in[1] no stemming was used, while special characters were removed and dataset was splitted into tokens. After that, the initial analysis of data was made using NB algorithm, and results showed that, three tokens were very important, they are: the number of numeric strings, number of dollar signs ($) and the length of the message, in addition, special characters such as "://" and "@" are crucial to classify emails. Accordingly, the analysis showed that the message length considered as a good criterion to detect spams. Many classifiers were used such as SVM, KNN, also two ensemble methods, random forests and Adaboost were used. The results showed that the overall error was reduced by more than half of error in[1].

A new dataset was collected and made public; this dataset was presented in [10]. However, this data set consists of 747 spam messages and 4827 ham and also different machine learning algorithms were applied on it. The same as in[1] there was no stop words removal or stemming methods and two tokenizers were applied. The comparisons between different classifiers showed that SVM is the best.

Feature extraction and selection is very important and critical for SMS spam detection, it affects the accuracy and performance of classifiers. Therefore, the impact of such feature selection was analyzed after applying it in two languages: English and Turkish in[11]. Also, combination between the feature originated from bag-of-words (BoW) model and structural features (SF) were used. On the other hand, collection of Turkish SMS was created and made public, this dataset consists of 430 ham and 420 spam messages. Therefore, and for the purpose of making comparisons, English data set of which consists of 425 spam and 450 ham messages were used. In order to be able to calculate the frequency of terms, Term Frequency - Inverse Document Frequency (TF-IDF) were applied, and vector space model was used to represent the document as collection of words and their frequencies. Some of significant features that were used to detect spams are: the length of the message, upper or lower case characters, number of significant terms, numeric and alphanumeric characters such as ("!", "$"), in addition to URL link. SVM and KNN were selected to make classification, and SVM was the best according to results. Frequency of diagrams, monograms, in addition to message size was the main features used in[12] for detecting spams. The dataset collected by volunteers which contains 6600 messages. Comparisons between three classifiers were made, they are: Artificial neural network, Decision tree and NB algorithms and results showed that NB was the best.

In their work Karami and Zhou proposed new features in order to increase the performance of SMS spam detection which mainly based on content features[13]. While most of previous researches focused on tokens without concerning the deep level semantic, dealing with semantic is not easy since spamming tactics evolve constantly. On the other hand, in their work[13] they used the category semantic of the word which results in efficiency improvements. However, many features were used such as: Semantic of words, capital and spam words, SMS segments, unique words, URL, SMS frequency, using word "Call" and the rate of URL. Also, other Linguistic Inquiry features were used such as linguistic processes, psychological processes and Spoken features and many others. Accordingly, precision, F-measure, accuracy, recall, ROC curve and other metrics were utilized to make evaluation after applying SVM, boosting and random forest as classifiers, the accuracy ranges from 92% to 98%.

Moreover, many text classifications can be used as classifiers for SMS spam such as Neural Network, tree structure and Increment Component Analysis (ICA) algorithms[14]. However, these algorithms were applied after extracting features such as: number of characters, ratio of number digit characters, whitespaces and alpha characters, frequency of each letter and special characters, total number of words, ratio of short words, ratio of number of characters in words, average of sentence length according to number of words and number of characters, in addition to average word length and others.

The first time GentleBoost algorithms were used in classifying SMS spam was in[15]. The reason for using boosting algorithm is the existing of unbalanced data. However, with unbalanced data and binary classification, booting is the best choice, especially GentleBoost which combined the features of AdaBoostM1 and LogitBoost. The proposed method[15] minimized the consumption of storage and at the same time keeping the accuracy high. The storage was reduced since unused features will be removed and optimized. The stop words and symbols such as "the", "☺" will be removed, then the tokenization process will be applied. In order to avoid removing word

attributes that may be important for classification process, the probability of each word for being ham or spam will be calculated, then these probabilities will be compared. Accordingly if the probability of word attribute to be ham is greater than the probability of it to be spam the word will be removed, otherwise it must remain. The experiments were made in Tiago's dataset which consists of 5572 messages, and the accuracy was 98%.

Apriori and NB classifiers were used for SMS spam detection[16], each word in a dataset is considered as independent word. However, in pre-processing phase, the discard words that are not important will be removed. Many surveys and reviewed were made in SMS spam detection researches[2, 3, 6, 7]. Two different tools were used Weka and RapidMiner to classify and cluster SMS spam, all researches experiments in this survey[6] used UCI dataset, the results of comparisons showed that the two tools provided the same results. However, in our experiments we will use H2O platform where no other researchers used it.

3. Background

3. 1 H2O Framework

Many of machines Learning (ML) libraries are available in H2O platform where H2O platform is an open source framework. In addition to having many ML libraries there are engines for parallel processing, math libraries, analytics and deep learning which characterize by being fast and scalable algorithm. Also H2O have tools to enable data processing and models evaluation[17,18] due to size limitation of the paper for more details refer to[17].

3.2 Classification models

Unseen data can be classified using machine learning algorithms; this can be achieved by learning the representation and the pattern of unseen data. In this research, three main machine learning algorithms will be used: Deep Learning, Random Forest and Naïve Bays.

3. 2. 1 Deep Learning

Deep learning algorithm make effective decisions by analyzing complex problems, it is much related to Artificial intelligent and tries to imitate what human brain can do. However, taking input and then making non-linear transformation will produce output from output layer. As a result, the output of each layer will be the input for the second layer in hierarchical manner; this structure can help in learning since the level of abstraction of certain layer will determine the level of abstraction of the next layer. Moreover, the level of deepness will be determined according to number of layers, where the more deep meaning the more layers will result in highly nonlinear function.

3.2.2 Random Forests

Random Forest (RF) depends on random selection of variables and data to develop large number of decision trees. Random Forest uses decision trees to create bootstrap by selecting random features, it is considered as special case from bagging[19]. However, the main idea of RF is that Shallow trees which are called stumps will be pruned and tuned and the output after tuning and pruning will be aggregated, then RF will rely on such aggregation. The aggregation will lead to accurate prediction by eliminating the error from trees.

3.2.3 Naïve Bayes

One of classifiers that based in Bays theorem and considered as the simplest probabilistic one is Naïve Bayes. Naïve Bays based on independent assumption that deals with each word independently, it is very simple model. In addition, if two words are independent of each other the performance will be reasonable; in this case, new data can be classified using Naïve Bayes. Classification can be made by using training data to count the term frequency[20]. Moreover, Naïve Bays is fast in term of training and classifying data and it can deal with discrete data and assume

that the features are independent.

4. Proposed framework

In this research, framework for SMS spam detection is proposed. The framework consists of a set of processes: the first one is the selection of dataset, then the features will be selected and extracted from the dataset. In the next process, the classification methods will be determined; this framework will use three classifiers: random forest, deep learning and naïve bays moreover all the experiments will be made in H2O platform. After that, the evaluation metrics will be specified and finally after applying the model in selected dataset many experiments will be made in order to make comparison between different classifiers. These processes can be seen in Fig 1.

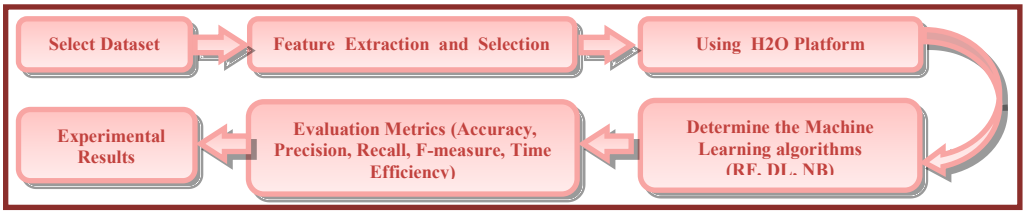

Fig 1: Steps of Proposed Method

4.1 Data Selection

In this research, authors will use UCI Machine Learning repository dataset[21] which was gathered in 2012. The dataset consists of 5574 text messages classified as ham and spam messages, the number of spam messages is 747 while the number of ham messages is 4,827 messages. 425 spam messages were collected manually from the Grumbletext website, since users of mobile can announce publicly for the existence of SMS spam, and 322 spam messages were collected from Corpus v.0.1 Big. However 3375 of ham messages were randomly selected from NUS SMS Corpus (NSC) and 450 ham messages collected from PhD thesis for Caroline Tag, the rest of ham messages which are around 1002 were gathered from Corpus v.0.1 Big. The dataset is saved in a text file where each line represents one message; the line consists of label of a message and text string. Table 1 displays examples of messages in dataset.

Table 1: Example of ham and spam messages in UCI dataset

Ham	Go until jurong point, crazy.. Available only in bugis n great world la e buffet... Cine there got amore wat...
Ham	Ok lar... Joking wif u oni...
spam	Free entry in 2 a wkly comp to win FA Cup final tkts 21st May 2005. Text FA to 87121 to receive entry question(std txt rate)T&C's apply 08452810075over18's

4.2 Feature extraction and selection

Feature extraction is very important since it affects the performance of SMS spam detection classifiers. Therefore, features that will be used in classification must add values, thus the features that will not add any value will not be considered in order to keep space and time. For the purpose of feature extraction, Python code was written, read the data from the source file and save the output in CSV format.

Using all the features in classification is time consuming process and may affect the accuracy of SMS spam detection reversely, thus in order to determine the most significant features, we made experiments using H2O platform. However, deep learning and random forest machine learning algorithm were used to select the most important features. After applying Deep learning, the features selected were: the existence of URL, word “call” and digits and they are sorted according to their importance, on the other hand the total number of words with length less than three was the least important feature. However, after applying random forest algorithm, the order of significant features differs. Therefore, the most important features were the total number of digits and the ratio of digits, while the least important one was the existence of URL. In this research we selected the common and significant features

from both algorithms; and a list of all features that were used in this research are summarized in Table 2.

4.3 Evaluation metrics

SMS spam detection classifiers proposed in this research were evaluated using five metrics which are: Accuracy, Precision, Recall, F-measure and Run time. However, for classifying binary class the evaluation can primary reference the confusion matrix. In order to compute the evaluation metrics, identifiers must be defined such as true positive, true negative, false positive and false negative. True positive (TP) is the messages that are correctly classified as spam, on the other hand true negative (TN) is the messages that are correctly classified as ham, however false positive (FP) is the SMS message that are classified as spam while they are ham and finally false negative (FN) is the misclassified messages that are spam and classified as ham. According to the definition of the four variables TP, TN, FP, FN the metrics will be defined as follows[22,23]: Accuracy is the percentage of the messages that are classified correctly over the total number of messages. Furthermore, precision can be defined as how many of classified messages are spam message, moreover recall refers to how many of the spam messages are correctly classified as spam. Another measure that combines precision and recall into one measure is f-measure. F-measure and accuracy values must be high in order to get better classification. The Final metric is the runtime that is the time needed to build a model. For better results runtime must be low, however balance between the five metrics must be made.

Table 2: List of Features

Feature	Feature description	Feature criteria
#1	Message Length (ML)	The number of characters in the message
#2	Number of Words (NW)	The number of words in message, usually spam messages contains large number of words
#3	Ratio of Number of Words with length less than three (RNW3)	Number of words less than 3 (NW3) over to the total number of words (NW)
#4	Ratio of Number of Capital (RCW)	Number of Capital Words (CW) over total number of words (NW)
#5	Ratio of Alphanumeric Characters (RAC)	Number of Alphanumeric Characters over Message Length (ML)
#6	Ratio of Special Characters (RSC) such as “,*, _ ,+,=,%,$,@, , \,/,”	Number of Special Characters over Message Length (ML)
#7	Ratio of Punctuation Characters (RPC) such as “‘; ? ! : ( ) – “ « » < > [ ] { }”	Number of Punctuation Characters over Message Length (ML)
#8	Total number of Digit Characters (DC)	Normalized by maximum number of digit characters.
#9	The existence of word “Call” and digits together	The value of this field is either true or false
#10	The existence of URL in the message	The value of this field is either true or false

5. Experiments and results

In our research, comparisons between three machine algorithms were made: RF, DL and NB. In all experiments the same dataset were used which is the one that was proposed by UCI Machine Learning Repositories, however several tuning and changes of parameters for each algorithm were made and the average of five runs for each change was taken. In addition, two evaluation models were used, 3-fold and 10-fold cross validation. In 3-fold cross validation the data are divided into equally three parts, where 2 parts will be used for training and the third one will be used for testing. On the other hand, in 10-fold cross validation instead of using two parts for training, 9 parts will be used for training and the 10^{th} for testing. However, for the purpose of reliability 3-fold cross validation will be repeated three times while 10-fold cross validation will be repeated 10 times. Furthermore, the final results were calculated by computing the average of runs. Accordingly, experiments showed that 3-fold cross validation is faster than 10-fold cross validation, while the later is better in term of precision, recall and f-measure. After tuning the parameters of deep learning, the best results was achieved when the rectifier activation function is used, the number of neurons in each hidden layer is 20 neurons and epoch is 10. On the other side in RF the best results occurred when the number of trees used is 5 and maximum depth is 20, and finally the default NB was used. All the results of using of using 10-fold cross validation is show in Fig 2 and Fig 3.

H2O platform was used to compare several classifiers; since H2O provides a collection of machine learning algorithms such as DL, RF and NB. Moreover, authors used H2O in order to determine the best features that will be used to improve the SMS spam detection process. In addition, the significant features from the dataset were extracted by using Python for preprocessing. As a result of making experiments, the following conclusions were reached: In DL, the increase in number of neurons in each hidden layer and number of epochs will result in increase in precision, recall, f-measure and accuracy, which provides desirable results. Nevertheless, the runtime will increase which may make the classifier not desirable, thus the tuning of parameters must be made in order to make balance between runtime, precision, recall, f-measure and accuracy. Moreover, the best results of SMS spam detection was achieved when the number of neurons is 20 and epoch value is 10. On the other hand in RF the best results were achieved when the number of tree is 5 and maximum depth is 20, also as in DL, several experiments were made using RF, the results showed that as number of trees and maximum depth increase the precision, recall, f-measure, accuracy and runtime will increase. Overall results of accuracy, f-measure, precision and recall can be seen in Fig 2 and results of runtime can be seen in Fig 3, where the results showed that RF is the best algorithm in term of precision, recall, f-measure and accuracy with 95%, 85% 0.89% 0.97% values respectively, however RF is not the best in term of time. On the other hand, NB is the best classifier in term of runtime with 0.6 seconds. Nevertheless, NB is the worst according to precision, recall, f-measure and accuracy.

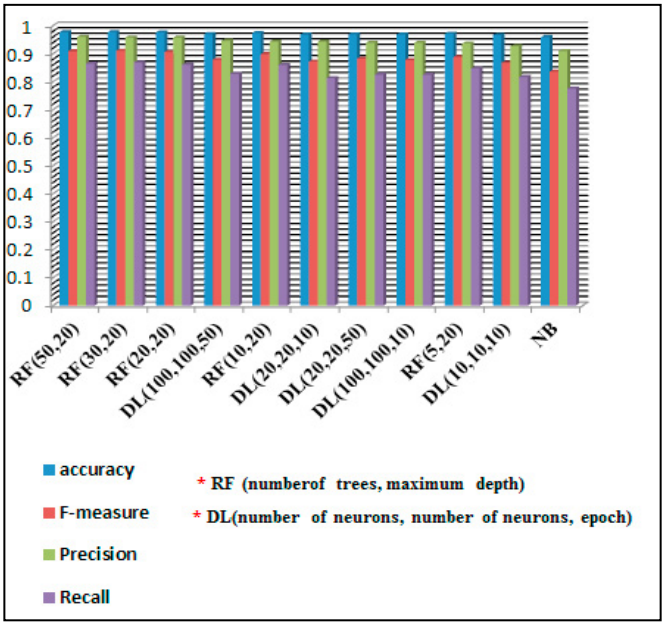

Fig 2: Comparisons between RF, DL, NB using 10-fold cross validation according to accuracy, f-measure, precision and recall

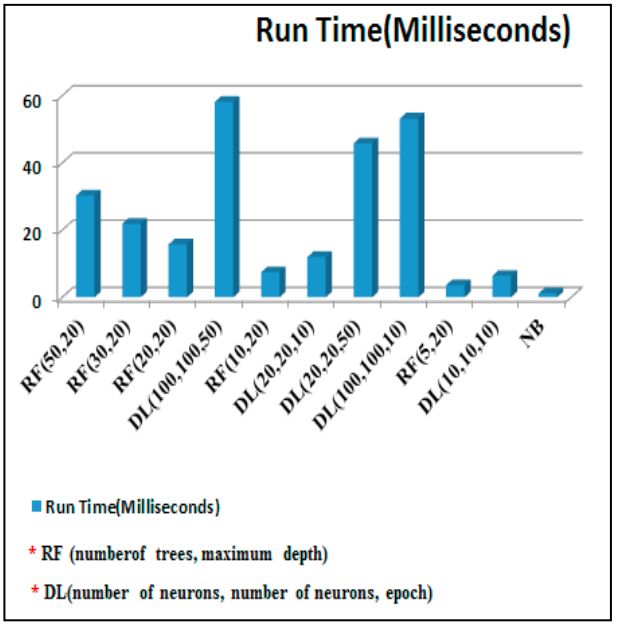

Fig 3: Comparisons between RF, DL, NB using 10-fold cross validation according to runtime

In addition, the UCI Machine Learning Repositories dataset which was used in this research was also used by other researchers[2,3], in order to detect SMS spam. However the platforms that were used for experiments are different, while in this research we used H2O platform, in[2] Weka was used. Furthermore, another difference related to the evaluation metrics, the evaluation metric that was used in[2] is the accuracy and in[3] is the error rate, nevertheless in this research accuracy, f-measure, precision, recall and runtime metrics were utilized. Moreover, the machine learning algorithms that are compared in different researches were different; this is the first time where DL algorithm was used in SMS spam detection.

6. Conclusion

SMS is one of the most significant communication methods between people, where SMS messages are two types: ham and spam, spam message are the undesirable messages that must be removed or blocked before the user

receiving them. Therefore, in this research, three machine learning algorithms were used in order to improve SMS spam detection process DL, RF, NB. Moreover, experiments were made using H2O platform and the data that was used is UCI Machine Learning Repositories dataset, while the same dataset was used in other researches, the evaluation metrics and experimental environment are different. In addition to using DL and RF in classification, they were also utilized to determine the significant features of SMS spam detection. Accordingly, the significant features are the number of digits and existing of URL in message text. Furthermore, the metrics that were used for making evaluation are accuracy, f-measure, precision, recall and runtime, thus in order to make balance between the different metrics tuning of parameters was made. For each change in parameters, the average of five runs was taken, also 3-fold and 10-fold cross validation evaluation models were used. As the number of folds increases the precision, recall, f-measure and accuracy will increase which is considered as an excellent improvement however, the runtime will increase which is undesirable. The experiments showed that the naïve bays classifier is the best in term of runtime, however it is the worst with respect to precision, recall, f-measure and accuracy. On the other hand the best algorithm in term of precision, recall, f-measure and accuracy is the random forest which achieved significant improvement with values equal to 96%, 86%, 91% and 0.977% respectively.

References

1. Tiago A. Almeida , José María G. Hidalgo , Akebo Yamakami, Contributions to the study of SMS spam filtering: new collection and results, Proceedings of the 11th ACM symposium on Document engineering, September 19-22, 2011.
2. G. Cormack. Email Spam Filtering: A Systematic Review. Foundations and Trends in Information Retrieval, 1(4):335–455, 2008.
3. H. Ji and H. Zhang, "Analysis on the content features and their correlation of web pages for spam detection," Communications, China, 2015
4. Sajedi H., Parast G., Akbari F., SMS Spam Filtering Using Machine Learning Techniques:A Survey, Machine Learning Research 2016; 1(1): 1-14 http://www.sciencepublishinggroup.com/j/mlr doi: 10.11648/j.mlr.20160101.11
5. Chaudhari N., Jayvala, Vinitashah, Survey on Spam SMS filtering using Data mining Techniques, International Journal of Advanced Research in Computer and Communication Engineering ISO 3297:2007 Certified Vol. 5, Issue 11, November 2016
6. Zainal, K., Sulaiman, N. F., Jali, M. Z.: An Analysis of Various Algorithms For Text Spam Classification and Clustering Using RapidMiner and Weka. International Journal of Computer Science and Information Security (IJCSIS), 13(3), pp. 66–74. 2015
7. Kawade D.,Oza K., SMS Spam Classification using WEKA, International Journal of Electronics Communication and Computer Technology
8. Gordon V. Cormack , José María Gómez Hidalgo , Enrique Puertas Sánz, Feature engineering for mobile (SMS) spam filtering, Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval, July 23-27, 2007, Amsterdam, The Netherlands [doi>10.1145/1277741.1277951]
9. Shirani-Mehr, H. (2013). SMS spam detection using machine learning approach. CS229 Project 2013, Stanford University, USA, pp. 1–4
10. T. Almeida, J. M. G. Hidalgo, and T. P. Silva. Towards sms spam filtering: Results under a new dataset. International Journal of Information Security Science, 2(1), 2013
11. Uysal AK, Gunal S, Ergin S, . The impact of feature extraction and selection on SMS spam filtering. Electronics and Electrical Engineering 2013; 19(5): 67–72
12. Mujtaba, G., Yasin, M.,SMS spam detection using simple message content features. J. Basic Appl. Sci. Res. 4, 275–279 (2014)
13. Karami and L. Zhou, "Improving static SMS spam detection by using new content-based features," 2014.
14. Anchal, Sharma A., SMS Spam Detection Using Neural Network Classifier, International Journal of Advanced Research in Computer Science and Software Engineering, Volume 4, Issue 6, June 2014
15. Akbari, F., Sajedi, H.: SMS spam detection using selected text features and boosting classifiers. In: Information and Knowledge Technology. IEEE (2015)
16. SABLE S, SMS CLASSIFICATION BASED ON NAIVE BAYES CLASSIFIER AND SEMI-SUPERVISED LEARNING, INTERNATIONAL JOURNAL OF INNOVATIONS IN ENGINEERING RESEARCH AND TECHNOLOGY , VOLUME 3, ISSUE 7, JULY-2016 (IJECCT) VOLUME 5 ISSUE ICICC (MAY 2015)
17. Arora, A., Candel, A., Lanford, J., LeDel, E., & Parmar, V. (2015, August). Deep Learning with H2O. H2O.ai, Inc. Retrieved from https://h2o-release.s3.amazonaws.com/h2o/master/3190/docs-website/h2o-docs/booklets/DeepLearning_Vignette.pdf
18. Landset, S., Khoshgoftaar, T. M., Richter, A. N., & Hasanin, T. (2015). A survey of open source tools for machine learning with big data in the Hadoop ecosystem. Journal of Big Data, 2(1), 24. https://doi.org/10.1186/s40537-015-0032-1
19. Breiman, L.: Random forests. Mach. Learn. 45, 5–32 (2001)
20. Z. Cataltepe and E. Aygun. “An improvement of centroid-based classification algorithm for text classification”, in Proc. IEEE 23rd International Conference on Data Engineering Workshop, 2007, pp. 952956.
21. SMS Spam Collection Data Set from UCI Machine Learning Repository, http://archive.ics.uci.edu/ml/datasets/SMS+Spam+Collection
22. Chinchor N and Sundheim B. MUC-5 evaluation metrics. In: Proceedings of the 5th conference on message understanding, Baltimore, MD, 25 August 1993, pp. 69–78. Stroudsburg, PA: Association for Computational Linguistics.
23. Wang D, Navathe SB, Liu L et al. Click traffic analysis of short URL spam on Twitter. In: Proceedings of the 9th international conference on collaborative computing: networking, applications and worksharing (collaboratecom), Austin, Texas, USA, 20 October 2011