Fjrs.futa.edu.ng

FUTA Journal of Research in Sciences

ISSN: 2315 – 8239 (Print); E-ISSN: 2489 - 0413

FUTA Journal of Research in Sciences, Vol. 16(1), April, 2020: 26-38

EVALUATION OF MACHINE LEARNING ALGORITHMS FOR FILTERING AND ISOLATING SPAMMED MESSAGES

[1]N.A. Azeez, [2]O.E. Adio, [3]A.W. Yekinni, and [4]C.J. Onyema
[1,2]Department of Computer Sciences, University of Lagos, Nigeria.
[3]School of Tech., Computer Science Department, Lagos State Polytechnic, Ikorodu, Lagos.
[4]Department of Computer Science, Federal University of Technology, Owerri.
nazeez@unilag.edu.ng , adioodunola@yahoo.com, dewaleyk@gmail.com, chinazo.onyema@futo.edu.ng

ABSTRACT

The use of mobile application is rising on daily basis as they offer a wide range of services and at the same time elevating the fee of those services. Short Message Service (SMS) is considered one of the most commonly deployed messaging systems. However, this deployment has led to a spike in attacks on mobile devices such as SMS Spam. In this article, Artificial Intelligence (AI) method, which discovers and filters unsolicited spam messages was adopted. The machine learning algorithms used are Logistic Regression, Decision Trees, Gaussian Naïve Bayes; Multilayer perceptron (MLP) and Support Vector Machine (SVM). After experimentation, it was observed that MLP Classifier produced the best results with 93.1% true positive rate.

Keywords*:* Spammed message; mobile application; artificial intelligence; attacks; SMS

INTRODUCTION

Short Message Service (SMS) is one of the common messaging systems where an electronic message is sent from one end to another. It is one of the key components of communication systems for mobile phone and Personal Digital Assistant (PDA) devices. SMS enables the sharing of information using standard wireless application protocols to transfer short text messages between the moving devices. Reducing the mobile firms' prices of SMS infrastructure also led to increased use of SMS. Short Message Service (SMS) is one of the simplest and most reliable forms of communication. SMS is common worldwide due to fast response rate, secure and personal services. People use SMS to communicate instead of emails because there is no need for Internet connection when sending SMS and it is simple and efficient. This growth prompted attackers culminating in a problem with SMS spam. SMS spam or mobile phone spam is any junk that is delivered as a text message to the mobile phone. Spam SMS is not only annoying but also ruin phone memory. In some countries, the receiver is also charged for receiving such SMS. It is thus duty-bound to forestall SMS spam from being received at the SMS spam accounts. For 20-30 % of all SMS traffic in some countries in Asia, several reasons inspire spammers to use this service that supports the expansion of this drawback. Factors such as increasing the range of mobile users that may be targeted, the upper response rate for service, the restricted accessibility of mobile SMS spam filtering applications, lack of legislation and rules to manage the acquisition of SMS spam, just to mention a few. Spam on SMS has caused several issues for mobile users and mobile network operators. Some forms of SMS spam are trying to charge mobile users by tricking them into business premium rate numbers to buy services or persuading users to decision-bound numbers to gather their direction to alternative functions (FTC, 2013).

Any undesirable message sent to a mobile user is usually a spam message. Spam will contribute to private information being leaked, privacy infringement or unauthorized access to knowledge from mobile devices. During this era of smartphone apps, users have confidential information on their smartphones, such as contact lists, card details, pictures and password which are making them prone to cyber-attacks via spam text messages. It helps hackers engaging in illegal practices to manipulate mobile data without end-user information, thereby jeopardizing user's privacy. Spam messages seem to increase and cause frustration to users. In addition, SMS spam can be used to push malware and key-loggers (Aditya *et al,* 2018). This problem is also affecting mobile network providers. They are liable to be losing subscribers since the load of the SMS spam will weaken the signalling efficiency of the network. SMS spammers can procure any mobile range with any code to send spam messages, making it tough to spot a spammer.

Many methods have been used to detect spam text messages. These can be classified in two ways: one is a non-content-based approach which is being used by telephone providers and the other is a content-based approach that is used by mobile phone users. Text classification is one of the content-based approaches that can be used to classify spam or ham messages. The Ham message is the one that noble users produce when advertising companies create spam messages. However, spam messages are a problem if ham messages are miscalculated as spam and spam messages are not classified as spam messages.

Using machine-learning algorithms for SMS spam classification, gives the opportunity to know which of the algorithms provides the highest level of accuracy and F1-score in the problem of SMS spam classification. There are different kinds of algorithm for SMS spam identification. Examples of these algorithms are deep learning, Support Vector Machine (SVM), Naïve Bayes(NB), Logistic Regression, decision tree(DT), Multilayer Perceptron etc. For the purpose of this study, the algorithms used are: Support Vector Machine (SVM), Gaussian Naïve Bayes (NB), Logistic Regression, Decision Tree (DT) and Multilayer Perceptron.

In this study, the selected algorithms for classifying, isolating and filtering spammed SMSs were evaluated to determine which one is best suited for use to discover and filter unsolicited spam messages.

Section 2 provides literature survey of previous works. Section 3 describes the proposed work on classification of spam. . In this chapter, the architecture of the spam classification which includes ranking and classification process are discussed. Section 4 discusses about the result of this work while section 5 presents the conclusion.

LITERATURE REVIEW

In Goswami *et al.,* (2019), a machine learning approach for SMS Spam filtering and classification was achieved with Naïve Thomas Bayes, Support Vector Machine and Random Forest. The dataset employed in this study consists of 5574 measurements of two variables. There are, however, inconsistencies in the results obtained as the performances of the algorithms are not detailed.

According to Pavas *et al.,* (2018), spam and ham messages were detected using varied supervised machine-learning algorithms like the Naïve Thomas Bayes algorithm, support vector machine algorithm and the maximum entropy algorithm. They compared their outputs in filtering the Ham and Spam messages.

Atanu and Kumar, (2018) contrasted the results for model evaluation among LR, NB, DT and GBT algorithms. In addition, the authors used ApacheTM Spark as a forum for measuring efficiency of the algorithms. Some research papers are available with the same dataset, but the context and the classifiers are entirely different. For performance analysis, precision and time-efficiency were considered. The results revealed that it took more time for GBTs to classify spam messages, while NB is better in precision and runtime (Azeez and Mbaike, 2017**)**.

Aditya *et al.*, (2018) ran comparisons between eight different classifiers. The results obtained from the classifier assessment indicate that for the two datasets, Convolutional Neural Network (CNN) classifier achieves the maximum precision of 99.19 percent, and 0.9926 and 0.9994 AR values.

Choudhary and Jain (2017) proposed a 10- feature SMS spam filtering procedure by utilizing five machine-learning algorithms, especially J48, Table of Decisions, Thomas Bayes Random Forest and Logistic Regression. Random Forest classification algorithm conveys the best outcomes with a true positive rate of 96.1%.

El-Alfy and AlHasan (2016) published a model for classifying both email and SMS text messages. They evaluated different approaches to finalize a set of features. They used two algorithms for the classification, that is: Naïve Bayes and Support Vector Machine (SVM). Eleven (11) features considered are: emotional images, uncommon characters, JavaScript code, gappy expressions, recipient address, URLs, conceivable spam terms, message metadata, work terms, point zone and spam space. They tried their hypothetical and demonstrated on five databases for mail and SMS.

Jialin *et al*., (2016) carried out a research titled “Messages Topic Model (MTM)”. They considered image words, context words, and subject terms to speak to spam messages based on the inactive semantic analysis chance presumption. By training SMS spam messages into arbitrary sporadic bunches, they utilized k-means algorithm to erase the scanty issue. Thereafter, they compiled all SMS spam messages as a single record to catchword co-occurrence designs (Jialin *et al*., 2016).

Kim *et al.*, (2015) suggested a light and the fast SMS filtering algorithm that can be implemented independently inside mobile phones. The procedure employs methods for the evacuation of unneeded information. These strategies incorporate a data sifting, the choice of highlights, data clustering and so on. With a relative volume of function values, they were able to select important features.

Shahi and Yadav (2014) suggested spam filtering of the Smartphone SMS for the Nepali text using Bayesian and SVM approaches. The study's key goal was to analyse the performance of spam filters from Naïve Bayes and SVM. The two spam filters were compared based on accuracy, precision and recall.

Shirani-Mehr, (2013) applied various machine learning algorithms to the problem of SMS spam detection. They compared their output to pick up knowledge which assisted to investigate the inconsistencies. They developed an application based on one of those algorithms that could channel high precision of SMS spams.

Table 1. Summary of reviewed articles

Author	**Year**	**Approach**	**Search Engine Independence**	**Strength**	**Weakness**
Goswami *et al.*,	2019	machine learning methodology	Yes	Effective spam filtering.	Few algorithms were considered
Pavas *et al.*,	2018	supervised machine-learning	Yes	Effective spam filtering.	URLs contain short domains and subdomains without any paths
Atanu and Kumar	2018	machine learning with ApacheTM Spark	Yes	Effective spam Detection	Computationally expensive
Aditya *et al.*,	2018	Machine Learning based	No	Very high True Positive and True Negative rates.	Few algorithms were considered.
Choudhary and Jain	2017	Machine Learning based	Yes	Low- zero False Positive Rate.	Few features were considered.
El-Alfy and AlHasan	2016	Machine learning based	Yes	Efficient spam Detection	Only two algorithms were considered.
Jialin et al.,	2016	Machine Learning based	Yes	Effective spam filtering.	The approach is narrow hence the result has limited application.
Kim *et al.*,	2015	Machine Learning based	Yes	Effective spam filtering.	Few algorithms were considered
Shahi and Yadav	2014	Machine Learning based	No	Effective spam filtering.	The work has limited application because only Nepali text was used.
Shirani-Mehr and Houshmand.	2014	Machine Learning based	No	Efficient spam Detection	Small dataset

METHODOLOGY

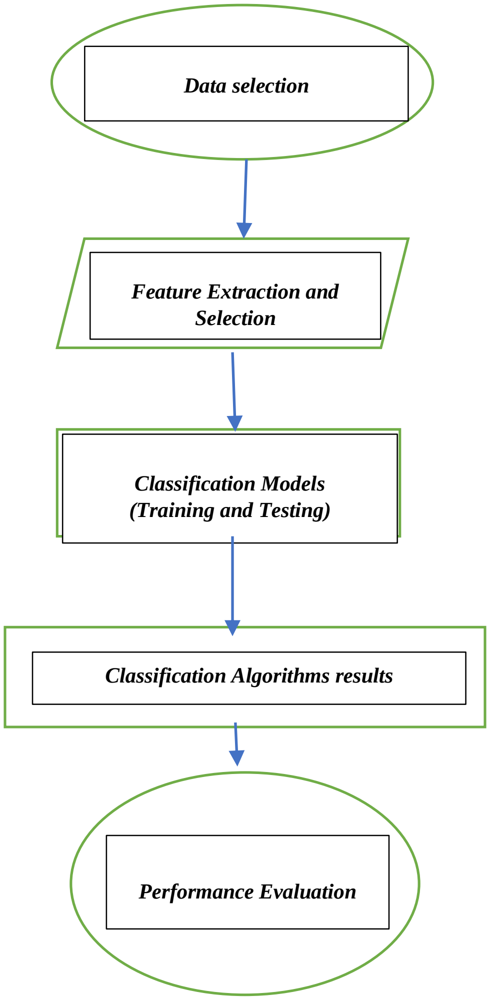
Data selection
Feature Extraction and Selection
Classification Models (Training and Testing)
Classification Algorithms results
Performance Evaluation

Figure 1. Flowchart of the adopted method

THE DATA COLLECTION

Dataset 1

A collection of 1000 SMS spam messages was physically extricated from the GrumbletextWeb http://www.grumbletext.co.uk/.

SMS Spam Corpus has 322 spam messages as well as 1,002 SMS ham messages. It is openly accessible.

Dataset 2

Grumble Text is a location to abdicate complaints of SMS Spam **(**Choudhary and Jain (2017). People, who get SMS Spam, intentionally yield the SMS on this location. http://www.grumbletext.co.uk/

A few information were collected physically from the website. 425 SMS Spam in total. A PhD Thesis titled "*A CORPUS LINGUISTICS STUDY OF SMS TEXT MESSAGING"* completed by Caroline Tag, with 11,000 content messages was examined, which contains 190,000 words and sent by 235 individuals. Nevertheless, not all 11,000 messages were composed on the thesis. 450 ham messages were spooled from the list.

For security reason, Caroline Tag and GrumbleText client have evacuated a few private information such as title, address, phone number etc.

Dataset 3

The dataset used here was obtained from:

www.dt.fee.unicamp.br/~tiago/smsspamcollection / www.esp.uem.es/jmgomez/smsspamcorpus/

This dataset has been successfully used for a similar implementation in a thesis titled "*Filtering SMS Spam in SmartPhone*" authored by Taufiq, et. al., (2010).

FEATURE EXTRACTION

Feature extraction and selection is very important regarding to SMS spam classification. The performance of spam classification will be affected by the feature extraction. Selected functionality should be associated with the type of message, so that spam message identification efficiency can be improved. SMS contains the original content (i.e. no records attachments, design, etc.) whereas within the mail, no text limitation exists as it incorporates attachments, design, etc. SMS is of two types. It can either be Ham (Non Malicious) and Spam (Malicious) (Azeez et. al., 2020). These two classifications are distinguished using various features after rigorous study of spam SMS patterns.

Features selected and extracted are summarized as follows:

- The Use of Web Links
- Special Characters
- All uppercased text
- Specific Keywords (call, now, claim, free, txt, guaranteed)
- The Use of Emojis
- The Use of Dots/full stop
- All lowercase texts
- Inclusion of numbers
- The Use of abbreviation
- The Use of symbols

Use of Web links

With web links, users are to provide their personal information, debit and credit card details and password.

The Use of special characters

For special characters, spam communications are alluded to by the use of certain symbols, for various reasons, since spammers utilize such symbols for their nefarious activities. Special symbol such as “ “ is used in various fraudulent award communications to reflect dollar currency. Similarly, “symbol is used as CONGRATULATIONS.

The Use of all uppercased text

Malicious SMS senders typically use highly qualified terms in uppercase letter as a tool to scan for the interest of the recipient. Words like WON, PRICE, FREE, CHEAP are used in this case.

The Use of Emojis

The use of emojis’ symbols seems to be a good indicator for legitimate messages because a person usually uses emotions while chatting as they are used to show expressions.

The use of all lowercase text

Checks in case the message contains lowercased words or not as all lowercased words in a message can be utilized to hunt for the user's consideration and interest.

Incorporation of numbers

Mobile number as an inclusion is to assist in recognizing spam messages since spammers ordinarily grant portable number in a message. They inquire the clients to call up a number and when client calls on the given number, assailant ask for the user’s individual points of interest.

The Use of abbreviation

The use of abbreviation can be taken as genuine messages since a client ordinarily employs shortened forms while having a chat. E.g., Can I cum now, wht abt urs?

The Use of Symbols

Malicious SMS senders ordinarily employ numerical symbols for spam sms. For illustration, symbol such as +, #, & can be utilized.

EVALUATION METRICS

To assess the viability and efficiency of this approach, the true negative rate, false negative rate, true positive rate, false positive rate, accuracy, precision, recall and f1-score/F1-measure were considered. These are the standard measurements for assessing any spam discovery system. These evaluation measurements are briefly explained as follows:

True Positive Rate (TP) - It indicates the rate of spam messages that the algorithms are accurately classified. S as Spam messages and P as spam messages precisely categorized as, then.

$$TP = \frac{P}{S}$$

True Negative Number (TN) - It indicates the percentage of ham messages which the algorithm correctly classified as ham messages. Denote the ham message as H and ham messages explicitly defined by Q as ham, then:

$$TN = \frac{Q}{H}$$

False Positive Frequency (FP)- It indicates the rate of ham messages, which the machine-learning algorithm wrongly classified as spam. H as Ham messages and ham messages which R inaccurately classified as spam, then:

$$FP = \frac{R}{H}$$

False Negative (FN) - it indicates the rate of spam messages, which the machine-learning algorithm wrongly identified as ham message. S as Spam messages and the number of SMS spam messages inaccurately identified by T as ham, then:

$$FN = \frac{T}{S}$$

Precision- it indicates the rates of spam messages which the classification algorithm actually classifies as spam. It shows just the correctness. It is expressed as:

$$Precision = \frac{TP}{TP + FP}$$

Recall- It indicates the percentage of spam messages and classifies them as spam. It displays completeness. It is expressed as:

$$Recall = \frac{TP}{TP + FN}$$

F-Measure - It is the harmonic mean of Precision and Recall. It is expressed as:

$$F - measure = \frac{2 * Precision * Recall}{Precision + Recall}$$

Accuracy – It indicates the rate of messages that are classified correctly over the total number of messages.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

Receiver Operating Characteristics – This is also known as the ROC. It is the region plotted between the True Positive Rate and the False Positive Area for different edge values.

Algorithms used for the implementation

- Logistic Regression
- Decision Trees
- Gaussian Naïve Bayes
- Multilayer perceptron (MLP)
- Support Vector Machine (SVM)

Logistic Regression

Logistic regression is used to predict the probability of an outcome that have two values. The prediction by logistic regression is premised on the use of one or many predictors which may be categorical and numerical. With logistic regression, a logistic curve is produced, which limits it between 0 and 1. The slope (b1) defines the steepness of the curve while the constant (b0) moves the curve right and left. The equation for the logistic regression can be presented in terms of an odds ratio as follows: (Azeez and Ajayi 2018).

$$\frac{p}{1-p} = \exp\left(b_0 + b_1 x\right) \quad \ldots\ldots\ldots\ldots1$$

LR can handle any number of categorical and/or numerical variables.

$$p = \frac{1}{1 + e^{-(b_0 + b_1 x_1 + b_2 x_2 + \cdots + b_p x_p)}} \quad \ldots\ldots\ldots..2$$

Decision Tree (DS)

This is a very popular supervised machine learning algorithm that is useful for regression and classification tasks (Azeez and Ajayi 2018). It adopts a set of rule for classification. With the decision tree, each of the nodes depicts an attribute, each leaf represents an outcome which is classified as a categorical value and each branch represents a rule. DSs are majorly used approach in statistical learning. They are usually constructed to work specifically for an existing set of data which can subsequently be used to predict the final outcomes on a new set of data. DSs can be formally explained with N labeled as "training records" of the form ($\boldsymbol{X}$,) where $\boldsymbol{X}$ could be regarded as a k-dimensional vector of features explaining the data under consideration, and Y is a label that this record is given. If Y takes on a single constant value for each of the regions under consideration $R1$,..,5, and Yi be the value selected for the region Ri, and if ($\boldsymbol{X}$) is considered an indicator function that equals 1 when $\boldsymbol{X} \in Ri$ then this scenario provides the benefit of getting a model that can predict Y based on $\boldsymbol{X}$:

$$\hat{Y}(\boldsymbol{X}) = \sum_{i=1}^{5} Y_i \times I_i(\boldsymbol{X}) \quad \ldots\ldots.3$$

Getting this type of model is the final goal of training a decision tree (Azeez and Imoru, 2017).

.

Gaussian Naive Bayes

Gaussian Naive Bayes (GNB) provides support for continuous valued-features. It also models each as conforming and compatible to a Gaussian (normal) distribution. Any attempt to establish a simple model is to assume that the data is well explained and properly described by a Gaussian distribution without co-variance between dimensions. GNB can be appropriate by simply determining the mean and standard deviation of the points within each label, which is only required to explain such a distribution (Azeez and Imoru 2017).

Whenever continuous data is being worked upon, it is commonly assumed that the continuous values characterized and associated with each class are distributed according to a normal (or Gaussian) distribution. The likelihood of the features is usually taken to be:

$$P(x_i \mid y) = \frac{1}{\sqrt{2\pi\sigma_y^2}} \exp\left(-\frac{(x_i - \mu_y)^2}{2\sigma_y^2}\right) \quad \ldots\ldots4$$

Multi-Layer Perceptron (MLP)

A Multi-Layer Perceptron (MLP) is a combination of an input layer, at least one hidden layer of Linear Threshold Units (LTUs) as well as an output layer of LTUs. Whenever an MLP has two or more hidden layers, it is referred to as a Deep Neural Network (DNN). The computations for MLP are the same as what is obtainable for a Perceptron. However, there are more layers of LTUs to combine until the output 'y' is obtained:

$$h^1 = step(z^1) = step(W^1 \cdot x + b^1 \quad \ldots\ldots5$$

$$y = step(z^2) = step(W^2 \cdot h^1 + b^2) \quad \ldots..6$$

'W' known as a matrix of shape (u, n), where u is considered the number of LTUs, n is regarded as the number of input values while the input vector **x** is known to be of shape (n, 1). The bias vector **b** is known to have the shape (u, 1) while the output vector **y** is characterized with shape (u, 1) (Azeez and Imoru, 2017).

Support Vector Machine (SVM)

A support vector machine (SVM) is a form of supervised machine learning algorithm that can be applied in both the regression tasks and classification. With SVM, data points are plotted as points in an n-dimensional space that is, n, being the number of characteristics features that are available. Finding the optimal hyperplane is considered the most desirable objective of SVM. Attempt to compute the SVM classifier is like minimizing the expression in 7 (Azeez and Vyver, 2018).

$$\left[\frac{1}{n}\sum_{i=1}^{n} \max\left(0, 1 - y_i(w \cdot x_i - b)\right)\right] + \lambda\|w\|^2. \quad \ldots.7$$

RESULTS

Table 2 – Results of the first Dataset.

Algorithm	Accuracy	TP rate	FP rate	Precision	ROC area	F-measure	Recall
Logistic Regression	0.9449378	0.655172	0.002101	0.9827586	0.8265357	0.7862069	0.655172
Gaussian NB	0.8898757	0.908046	0.113445	0.5939849	0.8973002	0.7181818	0.908046
MLP Classifier	0.9884547	0.931034	0.001050	0.9938650	0.9649920	0.9614243	0.931034
Decision Tree Classifier	0.9653641	0.890804	0.021008	0.8857143	0.9348981	0.8882521	0.890805
SVM	0.9493783	0.672414	0	1	0.8362069	0.8041237	0.672414

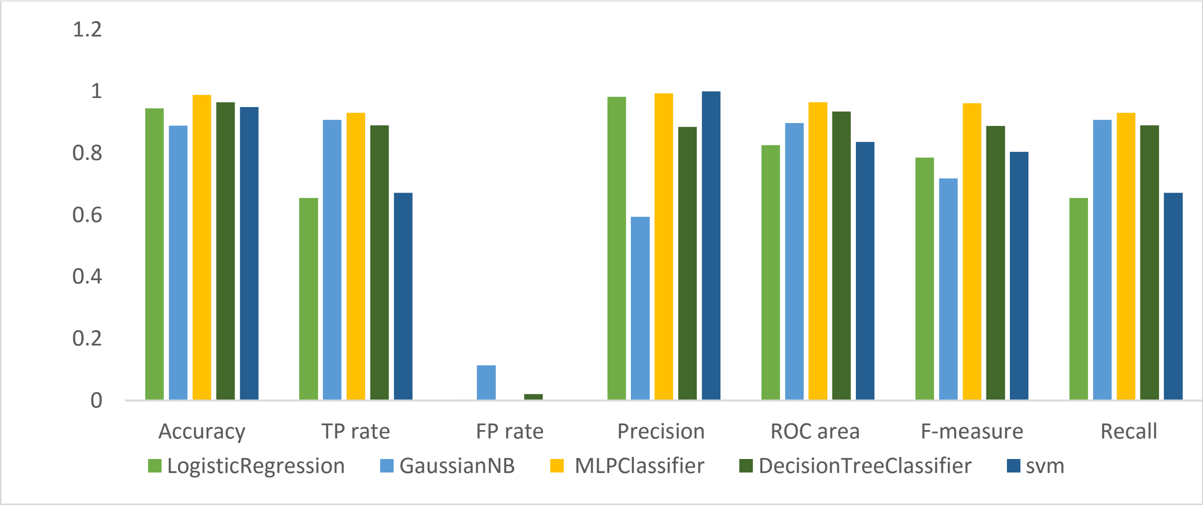

Figure 2: Graphical representation of Evaluation Metrics against Algorithms for Dataset 1

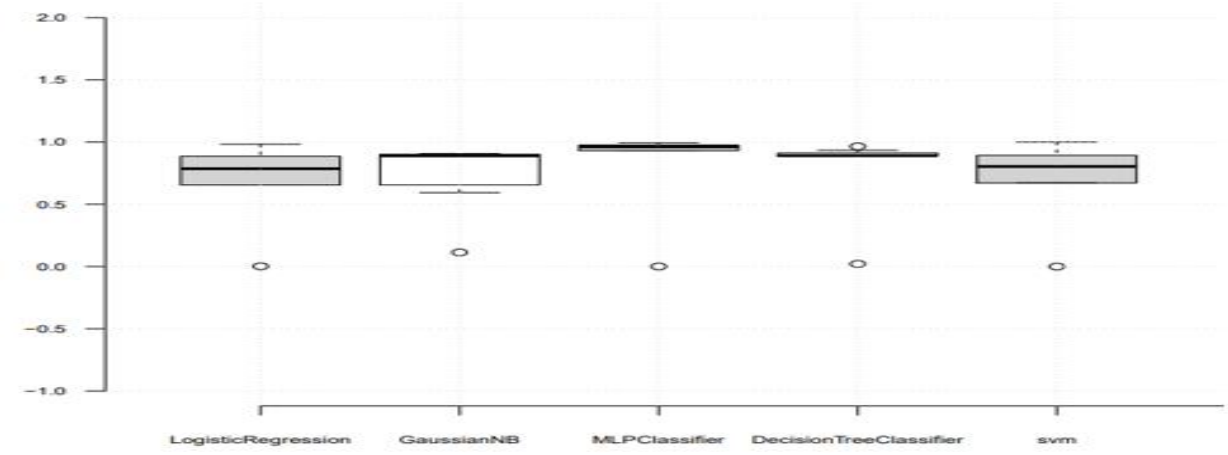

Figure 3 Analysis of evaluation metrics against algorithm for Dataset 1using Boxplot.

Table 3– Results of the second Dataset

Algorithm	Accuracy	TP rate	FP rate	Precision	ROC area	F-measure	Recall
Logistic Regression	0.9209039	0.860465	0.021978	0.9736842	0.919243	0.9135802	0.86046
Gaussian NB	0.9265536	0.930232	0.076923	0.9195402	0.926654	0.9248554	0.93023
MLP Classifier	0.9378531	0.872093	0	1	0.936046	0.9316770	0.87209
DecisionTree Classifier	0.9039548	0.860465	0.054945	0.9367088	0.902760	0.8969696	0.86046
SVM	0.8926553	0.779069	0	1	0.889535	0.8758169	0.77907

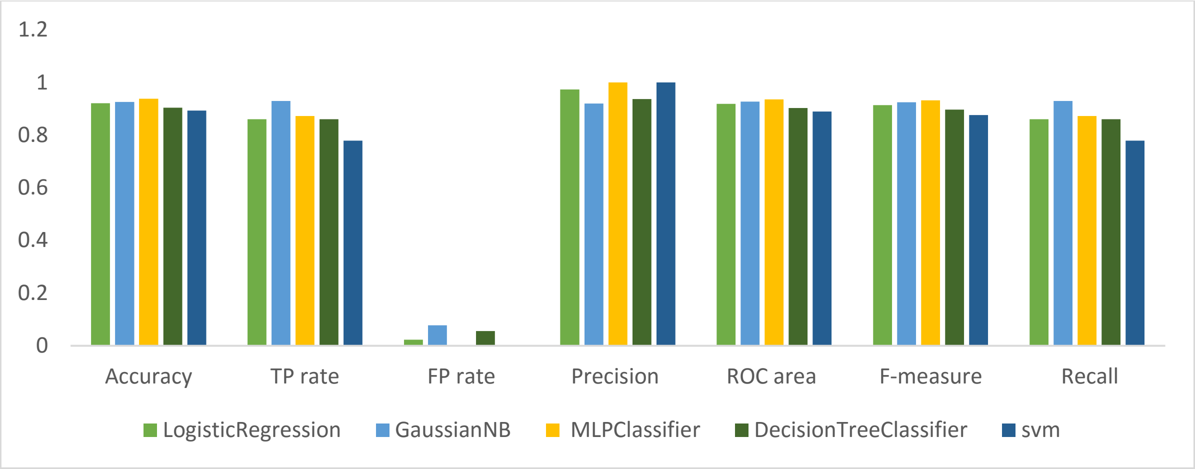

Figure 4: Graphical representation of evaluation metrics against algorithm for Dataset 2

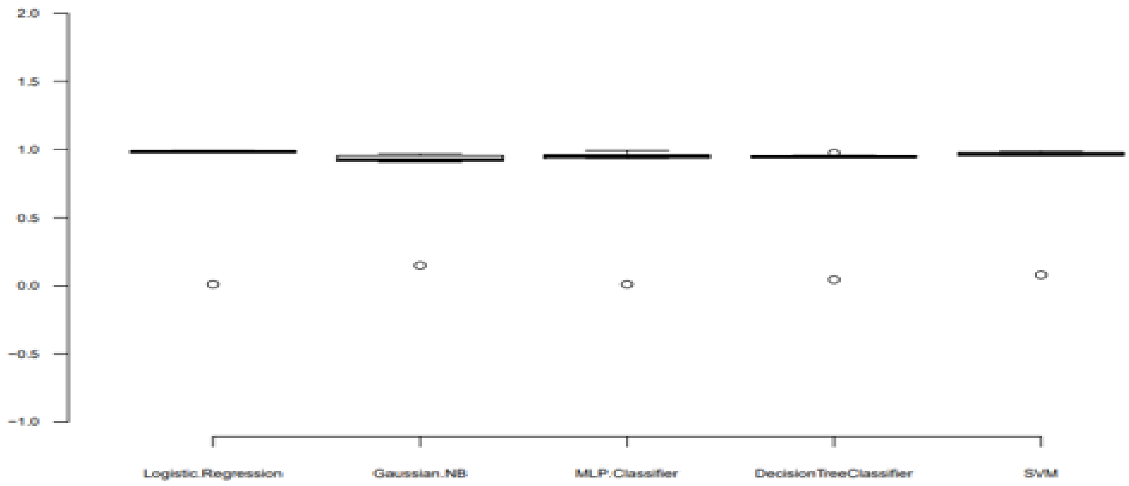

Figure 5: Analysis of evaluation metrics against algorithm for dataset 2 using Boxplot.

Table 4– Results of the third Dataset

Algorithm	**Accuracy**	**TP rate**	**FP rate**	**Precision**	**ROC area**	**F-measure**	**Recall**
Logistic Regression	0.9834711	0.9806452	0.0114943	0.993464052	0.9845755	0.987013	0.980645
Gaussian NB	0.9256198	0.9677419	0.1494253	0.920245399	0.9091583	0.9433962	0.967742
MLP Classifier	0.9545455	0.9354839	0.0114943	0.993150685	0.9619948	0.9634552	0.935484
Decision Tree Classifier	0.946281	0.9419355	0.045977	0.973333333	0.9479792	0.957377	0.941935
SVM	0.9628099	0.0804598	0.9870968	0.95625	0.9533185	0.9714286	0.987097

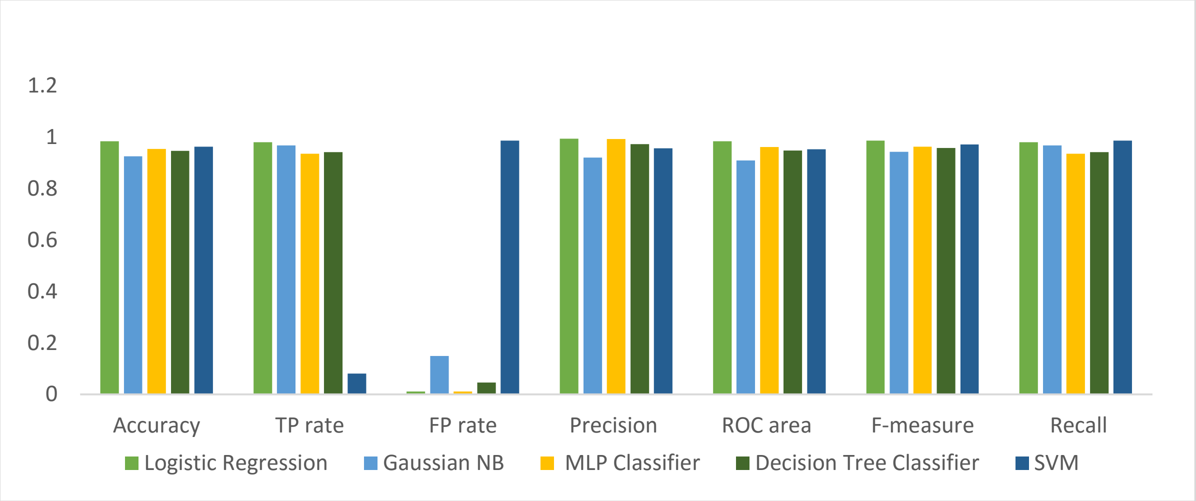

Figure 6: Graphical representation of evaluation metrics against algorithm for Dataset 3

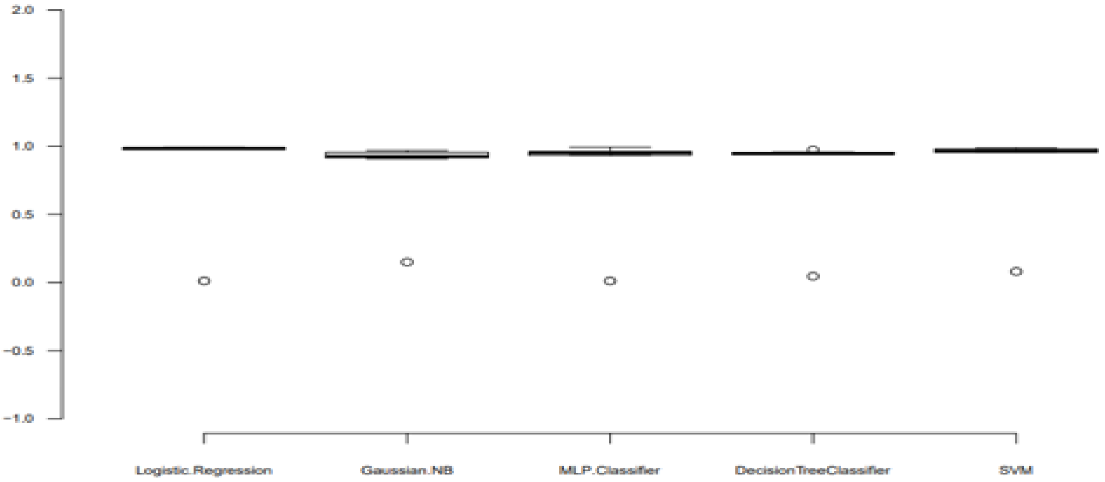

Figure 7: Analysis of evaluation metrics against algorithm for dataset 2 using Boxplot

DISCUSSION

Table 2 shows the results of the analysis of metrics for each of the algorithms for the first selected dataset (Dataset 1). The evaluation metrics considered include Accuracy, Precision, True Positive, False Positive, F-measure, Recall and the ROC area. The algorithms used are Logistic Regression, Gaussian NB MLP Classifier, Decision Tree Classifier and SVM. From the result in Table 1, MLP algorithm classified this dataset more accurately and reliably when compared to others. The performance as seen in Table 1 is that Decision Tree Classifier, SVM, Logistic Regression and Gaussian NB classified accurately in that order after MLP.

Fig. 2 shows a chart for the results of the evaluation metrics for each algorithm. The evaluation metrics are accuracy, Precision, true positive, false positive, F-measure, Recall and the ROC area for the first dataset. MLP classifier has the highest accuracy rate, True positive rate, Precision rate, and the F-measure rate with the lowest false positive rate. This shows that MLP classifier algorithm classified this dataset accurately when compared to others.

Fig. 3 shows the analysis of the five algorithms: Logistic Regression, Decision Trees, Gaussian Naïve Bayes, Multilayer perceptron (MLP) and Support Vector Machine (SVM) against the values of the metrics for the first dataset using Box plot.

Table 3 shows the results of the analysis of metrics for each of the algorithms for the second selected dataset (Dataset 2). From Table 2, MLP algorithms classified this dataset more accurately well when compared to others, followed by Decision Tree Classifier, SVM, Logistic Regression and Gaussian NB.

Fig. 4 presents a chart for the results of the evaluation metrics for each algorithm. The evaluation metrics are accuracy, Precision, true positive, false positive, F-measure, Recall and the ROC area for the second dataset. MLP classifier has the highest accuracy rate, the True positive rate, Precision rate, F-measure rate and the lowest false positive. This shows that MLP classifier classified this dataset accurately when compared to others.

Fig. 5 explains the analysis of the five algorithms along with the values of the metrics for the second dataset using Box plot**.**

Table 4 shows the results of the analysis of metrics for each of the algorithms for the third selected dataset (Dataset 3). From Table 3, MLP classifier classified this dataset more accurately when compared to others.

Fig. 6 **e**xplains a chart for the results of the evaluation metrics for each algorithm. MLP classifier has the highest accuracy rate, the True positive rate, Precision rate, F-measure rate and the lowest false positive. This shows that MLP classifier has the best accurate classification.

Fig. 7 shows the analysis of the five algorithms against the values of the metrics for the third dataset (Dataset 3) using Box plot.

From the foregoing, it has been clearly established that SMS has been recognised as one of the commonly deployed messaging frameworks in mobile technology. The deployment has, however, been characterised with numerous attacks. In this work, the process of filtering, identifying and discovering unsolicited spam messages have been achieved with Artificial Intelligence approach. Finally, it was observed that MLP classifier performed best with 93.1% true positive rate.

CONCLUSION

Nowadays, the issue of SMS spam is growing with the increased use of text messages. SMS Spam isolation is a major challenge these days. This paper proposes SMS Spam filtering technique based on 10 unique features with five machine learning algorithms namely: Logistic Regression, Gaussian NB, MLP Classifier, Decision Tree Classifier and SVM.

The dataset that used consists of 7,647. Out of all classification algorithms; MLP algorithm produced the best results with 93.1% true positive rate. The implementation of this approach will go a long way in providing a real-time solution to identify, discover and filter unsolicited spam messages in the global network.

REFERENCES

Aditya, B, **Mehul, G**, **Shubhangi, A,** and **Pulkit, M.** (2018). *A Comparative Study of Spam SMS Detection Using Machine Learning Classifiers* .2018 Eleventh International Conference on Contemporary Computing (IC3).

Atanu, G, and Ajit, K.P (2018). *Identifying spam SMS using Apache Spark MLlib, International Journal of Emerging Technologies and Innovative Research (www.jetir.org), ISSN:2349-5162, Vol.5, Issue 5, page no.893-897, MAY-2018*

Azeez, N.A and Imoru, O.R. (2017) *Using Four Learning Algorithms for Evaluating Questionable Uniform Resource Locators (URLs), Covenant Journal of Informatics and Communication Technology (CJICT), Covenant University, Nigeria, volume 5. No. 2 December 2017. pp 49-70.*

Azeez, N.A, and Mbaike, O. (2017) *SMS SPAM FILTERING FOR MODERN MOBILE DEVICES. FUTA Journal of Research in Sciences, Vol. 13 (1) 2017:177-185*

Azeez, N.A, and Ajayi, A.D. (2018) *Verification of fake and Vulnerable URLs with three Learning Algorithms, Nigerian Journal of Technological Development, University of*

Ilorin, Nigeria (UNILORIN). Vol. 16, No. 3, September 2019. pp 119-133.

Azeez, N.A and Vyver, C.V. (2018) *Access Control Model for E-Health in a Cloud-Based Environment for HIV Patients in South Africa, IST-Africa 2018 Conference 09 - 11 May 2018, Gaborone, Botswana. Pp 1-12, publisher: IEEExplore*

Azeez, N.A, Salaudeen, B.B, Misra, S, Damaševičius, R, and Maskeliūnas, R. (2020) *Identifying phishing attacks in communication networks using URL consistency features. Int. J. Electronic Security and Digital Forensics, Vol. 12, No. 2, pp 200-213*

Choudhary, N and Jain, A.K (2017) *Towards Filtering of SMS Spam Messages Using Machine Learning Based Technique, In book: Advanced Informatics for Computing Research, D. Singh et al. (Eds.): ICAICR 2017, CCIS 712, pp. 18–30, 2017. Springer Nature Singapore Pte Ltd. 2017.*

El-Alfy, E.S.M, and AlHasan, A.A. (2016) *Spam filtering framework for multimodal mobile communication based on dendritic cell algorithm.* Future Gen. Comput. Syst. 64, 98–107 (2016)

Federal Trade Commission. (2013). *Text Message Spam. Retrieved from https://www.consumer.ftc.gov/articles/0350-text-message-spam*

Goswami, V, Malviya, V, and Sharma, P. (2019). *Spam Emails/SMS Detecting Using Various Machine Learning Techniques, Journal of Applied Science and Computations. Volume VI, ,Issue VI, JUNE/2019*

Houshmand, S. (2014) "SMS Spam Detection using Machine Learning Approach." **(2014): 1-4**

Jialin, M, Zhang, Y, Liu, J, Yu, K, and Wang, X. (2016) *Intelligent SMS spam filtering using topic model. In: International Conference on Intelligent Networking and Collaborative Systems* (INCoS), pp. 380–383. IEEE **(2016)**

Kim, S.E, Jo, J.T, and Choi, S.H. (2015) ‘‘SMS spam filterinig using keyword frequency ratio,’’ Int. J. Secur. Appl., vol. 9, no. 1, pp. 329–336, 2015

Neelam, C, and Ankit, K.J. (2017)*Towards Filtering of SMS Spam Messages Using Machine Learning Based Technique in First International Conference,* ICAICR 2017, Jalandhar, India, March 17–18, 2017,

Pavas, N, and Gaurav, D.A.R. (2017) "*SMS Spam Filtering using Supervised Machine Learning Algorithms"* in IEEE 2018.Ali, W. (2017, January).

Shahi, T.B, and Yadav, A. (2014). *Mobile SMS Spam Filtering for Nepali Text Using Naïve Bayesian and Support Vector Machine* International Journal of Intelligence Science 04(01):24-28

Shirani-Mehr, H (2013) *SMS Spam Detection using Machine Learning Approach, In a Book: CS229 Project 2013, published by Stanford University, pp. 1-4.*

Taufiq, M, Abdullah, M, Choi, D, and Lee, G. (2010) *Filtering SMS spam on smart phone, Proc. of International Conference on Internet (ICONI), Korea Society for Internet and Information Systems (KSII), pp. 1-4*