Evaluation of SMS Spam Message Classification Model

Victor Chua Min Chun
School of Computing and Creative Media
University of Wollongong Malaysia KDU University College
Shah Alam, Malaysia
0129219@kdu-online.com

Abstract – The use of SMS text messaging has seen an exponential growth in the past decade with the improved accessibility of smartphones. The reduction in messaging costs has encouraged the use of SMS as a form of communication. However, this has also enabled groups of people with malicious intent to send fraudulent and spam messages for their own benefit. This is potentially harmful to users as they are at risk of being deceived by scammers. In order to solve the problem, researchers have been focused on developing spam filtering algorithms to deter users from being a victim of fraudulent scams. In this study, we have used several classification models to compare their performance in discerning legitimate and fake messages. The SMS messaging dataset is obtained from Kaggle, and is compiled from several reviewed datasets. Term Frequency-Inverse Document Frequency (TF-IDF) and Bag of Words (BoW) are utilized for feature extraction of the words. The dataset has then undergo pre-processing techniques to clean the text prior for model training. The models are then compared for evaluation in order to choose the most optimal machine learning algorithm.

Keywords – Spam Detection, SMS, Classification, Confusion Matrix, Machine Learning

I. INTRODUCTION

With the exponential increase in the usage of internet and wireless communications in the modern era, there have been cases of fraudulent scams targeting people in order to obtain their personal information. One of the main methods that are employed to perform these illegal acts are sending spam messages. Spam messages are unsolicited and sent in large volumes to a large number of people in hopes to gain the attention of unsuspecting users. In order to prevent spams and scams from occurring, there has been an increase in educating the public about how to identify and prevent such issues. Modern computers and phones also come with the feature of blocking the number or email that is sending spam messages.

However, this still does not prevent the messages from being sent to users, and this still requires the user to correctly identify spam messages to be blocked. Users might mistake spam messages for genuine messages, which can then lead to them falling for scams. In order to prevent spam from reaching users, machine learning algorithms can be utilized to analyze messages, and identify whether they are genuine or spam.

This study will look into the text classification of Short Messaging Service (SMS) text messages, where it will determine if the messages sent are spam or non-spam messages. It is the process of classifying a group of text under a defined category [1]. Text classification is intended to structure through a variety of text, ranging from documents, social media, surveys, emails, and more. This significantly cuts down the time required to analyze text data, automate otherwise lengthy business processes, and gain insights to make data-driven decisions. I will be performing data pre-processing and training several models that are commonly used for text classification. The performance of the models used will then be compared to determine which model is the most optimal in being able to determine the legitimacy of the messages.

II. RELATED WORKS

With the continued developments in mobile network technology and infrastructure in the past decade, there has been a widespread adoption of mobile phones to quickly communicate with each other. This has also created the need for a solution in addressing SMS spam, which has significantly increased. This problem is a global matter, and is not limited to specific countries. In Asia, SMS spam messages takes up to 20-30% of all SMS traffic in India and China [2]. A survey conducted in the US has reported that at least 44% of mobile device owners have received SMS spam [3].

Some of the current technologies that have been devised are anti-phishing and spoofing measures that can identify SMS messages that have been altered to forge its location information in order to avoid charges. UK's Mobile Ecosystem Forum (MEF) has developed a SMS Protection Registry, which deters spoofing and phishing messages by blocking unverified sender IDs [4]. Automated text classification are also deployed, where it utilizes supervised machine learning (ML) algorithms to differentiate between spam and legitimate messages [5]. However, the limited text length of an SMS message would mean that users would shorten their words with abbreviations in order to convey meaning, which has to be considered when using this approach.

Content based SMS spam filtering is also one of the approaches ued. Cong-Jie Chen et al. conducted a study with filtering SMS messages by using a Naive Bayes algorithm, which has a better performance compared to a keyword frequency filtering system [6]. Healy et al. have also used Naive Bayes along with Support Vector Machines and have discovered that different features and classifiers are required depending on the length of the text messages [7]. Hidalgo et al. compared several classification algorithms on two SMS spam datasets and found out that the methods used on email spam filtering can also be used on SMS spam messages, with SVM being the optimum model [8]. A hybrid approach, where content-based filtering and challenge-response, which sends a reply to the sender that requires them to perform actions to confirm the delivery of their message, has also been proposed to deter spam messages from being sent [9].

In this project, several models of classification will be deployed to determine the optimal model. The models used are listed below:

- Multinomial Naive Bayes
- K- nearest neighbour
- Support Vector Classifier
- Stochastic Gradient Descent
- Gradient Boosting Classifier

III. DESCRIPTION OF DATA SOURCE

The dataset that will be used to train these models will be an SMS dataset that includes 5572 messages as well as spam/ham labels and is obtained from Kaggle [10]. The dataset's information is collected from the following sources:

- Manual extraction of 425 SMS spam messages from the Grumbletext Website, a UK forum where cell phone users report spam messages. The identification of spam messages is done via meticulous scanning of a large number of web pages.
- A subset of 3375 ham messages was randomly sampled from the National University of Singapore (NUS)'s SMS Corpus, which has stored 10 thousand legitimate messages by their Computer Science Department.
- 450 SMS ham messages collected from Caroline Tag's Ph.D. Thesis
- 1002 SMS ham messages and 322 spam messages from the SMS Spam corpus v.0.1 Big.

IV. PRE-PROCESSING

Data pre-processing techniques are done in order to filter out irrelevant information that would not benefit the model training. As the dataset collected is a collection of SMS messages, punctuations, and common stopwords are removed. In order to execute the pre-processing, the NLTK library is imported for use.

The Bag of Words (BoW) model is implemented through the use of CountVectorizer(). This model extracts the features from text to be used in the classifier models [11], while CountVectorizer() allows us to tokenize the messages and build a vocabulary of words from it and count the occurrences of the tokens.

Term Frequency Inverse Document Frequency (TFIDF) is used to calculate the word frequencies. It highlights words that are relevant to the dataset [12]. Next, the data is split to be used for TFIDF matrix model classification. This is used to test the accuracy of classification through the use of the Multinomial NB model.

V. DESCRIPTION OF MODEL TRAINING

CLASSIFICATION MODELS

After applying the different set of features for the same classification model, we then proceed to split the dataset into train and testing data, in a 70:30 ratio. Next, pipelines are used to simultaneously apply CountVectorizer and TfidfTransformer onto both test and train sets. The pipelines are also used for fitting, predicting and evaluating the dataset.

The following classification models are used:

1. Naive Bayes (Multinomial NB)
2. Support Vector Machine (SVM)
3. K-Nearest Neighbors (KNN)
4. Stochastic Gradient Descent (SGD)
5. Gradient Boosting Classifier (GBC)

For this study on spam detection in SMS messages, there will be 4 possible outcomes for a binary classification task, which are: True Positive (TP), False Positive (FP), True Negative (TN), and False Negative (FN). TP represents the True Positive, where the model correctly predicts the positive class. TN represents the True Negative, where the model correctly predicts the negative class.

A False Positive (FP) would mean that the model incorrectly predicts the positive class, while a False Negative incorrectly predicts the negative class [13].

True Negatives (TN): ham messages labelled as ham

False Positives (FP): ham messages labelled as spam

False negatives (FN): spam messages labelled as ham

True positives (TP): spam messages labelled as spam

The models are then put into comparison through the use of a confusion matrix plot and evaluated.

A confusion matrix is a table that is regularly utilized to depict the performance of a classification model on a set of test data for which true values are known [13]. Visualization of predictive analytics like accuracy, precision, recall, and specificity often utilizes confusion matrices. Through the table, we are able to extract and understand the overview of the classification results.

On the table, the diagonal elements would speak for the number of points for which the predicted label matches the true label, while the off-diagonal elements are the ones that were mislabelled by the classifier. When the results of the diagonal values of the confusion matrix is high, it is more viable as this demonstrates that it has numerous correct predictions. The rows of a confusion matrix are parallel with the true (actual) classes whereas the columns are parallel with the predicted classes. For our spam detection classification, we used a seaborn heatmap for a nice plot of the confusion matrix.

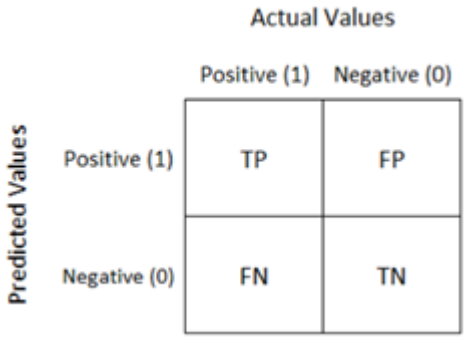

Fig 1. Values of Binary Classification

VI. RESULT DISCUSSION

The confusion matrix for the five models are generated:

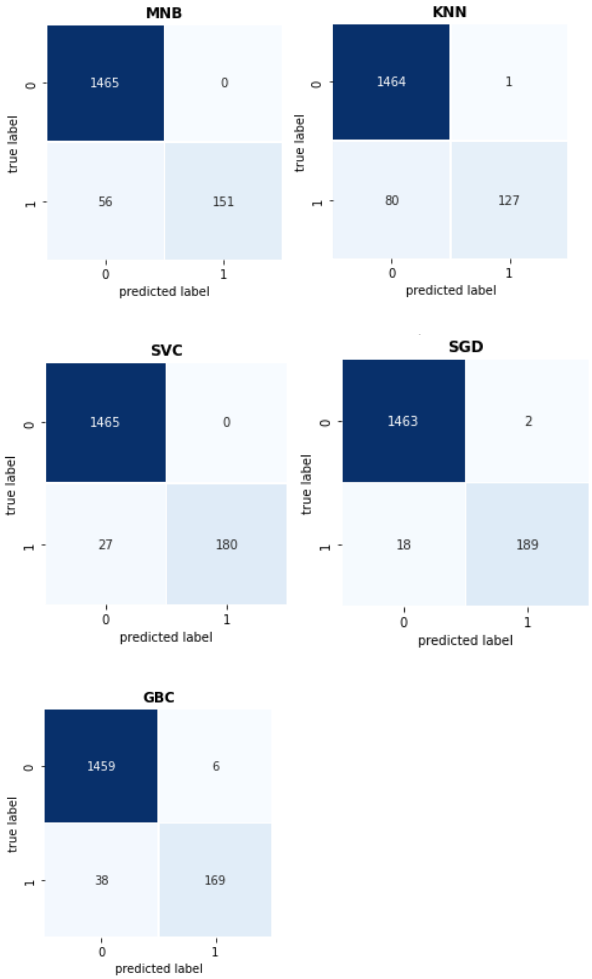

Fig. 2. Confusion Matrix of the five models used.

The performance of the classification models used are generated and compared. The following metrics for each model are collected:

- Accuracy
- F1 Score
- Recall Score
- Precision Score

ACCURACY SCORE

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

Fig. 3. Accuracy Formula

The following is the accuracy of each classifier used:
MNB 0.9665071770334929

KNN 0.9515550239234449

SVC 0.9838516746411483

SGD 0.9880382775119617

GBC 0.9736842105263158

Based on the scores above, it clearly shows that SGD Classifier has the highest accuracy compared to the rest. However, we cannot possibly choose the best classifier just by its accuracy score when comparing it with the confusion matrices as it may not be the suitable parameter to decide the suitable classifier. This is proven as SGD Classifier may look as though it is a model with a high accuracy score, it has incorrectly classified 6 ham messages as spam which is not viable for a spam detector as important messages could get lost. Although Multinomial Naive Bayes (MNB) has a lower accuracy score compared to SGD, it would be a better option as it is able to classify all ham messages correctly which in turn prevents important messages from being flagged as spam. Similarly, SVC is also a classifier which is able to classify ham messages correctly but when compared to MNB, it classifies more spam messages accurately.

PRECISION SCORE:

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

Fig. 4. Precision Formula

In simpler terms, Precision Score which is also known as positive predictive value is when we want to discover how many instances are actually positive out of the many instances that are retrieved. The precision is instinctively the capacity of the classifier to not label a sample that is negative, a positive sample. The Precision of a classifier goes up when the FP goes down. Precision = 1 for FP = 0.

MNB 1.0

KNN 0.9921875

SVC 1.0

SGD 0.9895287958115183

GBC 0.9657142857142857

Typically, the precision of the classifier is calculated for the negative class (label = 0, ham messages). It is also often used as a default when calling precision scores without the need of further parameters.

Based on the scores, the MNB and SVC classifiers have achieved a perfect score of 1, indicating that there are no false positives classified under the model. KNN comes off the next best model, coming at a 0.99 precision score. All of the applied models have a precision score of more than 0.95, suggesting that these models are reliable in discerning ham and spam messages.

RECALL SCORE:

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

Fig. 5. Recall Formula

Recall Score is used to measure whether our model is correctly identifying all True Positives samples. These are the recall scores of the classifiers which are defined regarding the positive class (label = 1, spam messages).

MNB 0.7294685990338164

KNN 0.6135265700483091

SVC 0.8695652173913043

SGD 0.9130434782608695

GBC 0.8164251207729468

Based on the scores the different models have a large variance in the score. The SGD model has shown the highest recall score, while KNN has the lowest recall score. This suggests that not all models are proficient at determining true positives in the SMS message dataset.

F1 SCORE:

$$F\text{-}measure = \frac{2TP}{2TP + FP + FN}$$

Fig. 6. F-1 Formula

The F-score, which is also known as the F1-score is the measure of a model's accuracy on a dataset. In addition, it is also used to evaluate the binary classification systems, which classify samples into 'positive' or 'negative'. The F1-score can be interpreted as a derivative of the precision and recall. These are the F-1 scores of the five classifiers used:

MNB 0.8435754189944134

KNN 0.7582089552238805

SVC 0.9302325581395349

SGD 0.949748743718593

GBC 0.8848167539267014

A model has performed well when the value is at 1, while the model is a total failure when the value is 0. SGD has the highest value of 0.94, while the lowest remains to be KNN, with the score sitting at 0.75.

VII. CONCLUSION

This study has shown that the application of spam detection in SMS messages is highly replicable and effective. The comparison between the different classification models has shown that different models can lead to varying results, and each model can be used for unique use cases. Careful consideration and selection of the classification model must be made when designing a product. In our study, we have decided that the SGD model is the optimal model, in regards to the overall high scores that are measured.

Another important factor to consider when evaluating the said models is the hardware that is used to run the model. Computers with better processing power can significantly reduce the computation time of the models. It is noticeable that when the models are unable to execute parallel processing the computation time drastically increases. This can adversely affect the performance of the product as well as the users that will be interacting with it.

On the other hand, the distribution of data in the dataset should also be taken into account based on the results. The dataset has a skewed distribution of data, where there are significantly more ham messages in comparison to spam messages. This can affect the training of the machine learning model as the model might not have enough information to distinguish between the different messages. This is represented by the false negatives and positives in the confusion matrix. Having a more balanced dataset would assist in the accuracy of the identification of spam messages.

VIII. REFERENCE

[1] M. Ikonomakis, S.Kotsiantis, V.Tampakas, "Text Classification Using Machine Learning Techniques." WSEAS Transactions on Computers, Issue 8, Vol 4. Aug 2005.

[2] GSMA (2011b). "SMS spam and mobile messaging attacks - Introduction. Trends and examples." GSMA spam reporting service. Jan 2011.

[3] Sarah Jane Delany, Mark Buckley, Derek Greene. "SMS Spam Filtering: Methods and Data." Expert Systems with Applications 39 (2012), Elsevier Ltd. 2012.

[4] E&T, "Smishing and spoofing targeted for eradication by SMS Protection Registry." Institute of Engineering and Technology. Sep 2021. (Online) Available: https://eandt.theiet.org/content/articles/2021/09/smishing-and-spoofing-targeted-for-eradication-by-sms-protection-registry/

[5] Sebastiani, F. "Machine learning in automated text categorization." ACM Computing Surveys. 34, 1–47. 2002.

[6] Liu, Cheng-Lin; Zhang, Changshui; Wang, Liang. "Study of Spam Short Message Filtering Based on Features Selection of Key Words." Communications in Computer and Information Science, Pattern Recognition Volume 321. 2012.

[7] Matt Healy, Sarah Jane Delany, Anton Zamolotskikh, "An Assessment of Case Base Reasoning for Short Text Message Classification." Proceedings of the 15th Irish Conference on Artificial Intelligence and Cognitive Sciences (AICS'04), 2004.

[8] Gómez Hidalgo, J. M., Bringas, G. C., Sánz, E. P., & Garcı´a, F. C. "Content based SMS spam filtering." In D. Bulterman, & D.F. Brailsford (Eds.), Proceedings of the 2006 ACM symposium on document engineering DocEng '06 (pp. 107–114). New York, NY, USA: ACM . 2006.

[9] UCI Machine Learning, "SMS Spam Collection Dataset." Kaggle.com, 2016 (Online). Available: https://www.kaggle.com/uciml/sms-spam-collection-dataset

[10] Jason Brownlee, "A Gentle Introduction to the Bag-of-Words Model." Deep Learning for Natural Language Processing, Machine Learning Mastery, 2019 (Online). Available: https://machinelearningmastery.com/gentle-introduction-bag-words-model/

[11] Jure Leskovec, Anand Rajaraman, Jeff Ullman, "Mining of Massive Datasets." Stanford University, 2014 (Online). Available: http://www.mmds.org/

[12] Yerushalmy J., "Statistical Problems in Assessing Methods of Medical Diagnosis, with Special Reference to X-Ray Techniques." Public Health Reports (1896-1970), Vol. 62, No. 40, Tuberculosis Control Issue No. 20 (Oct. 3, 1947), pp. 1432-1449, 1947.

[13] Stephen V. Stehman, “Selecting and Interpreting Measures of Thematic Classification Accuracy.” Remote Sensing of Environment, 62(10, 77-89. 1997.