Hybrid SMS Spam Filtering System Using Machine Learning Techniques

Hind Baaqeel[1]
College of Computer Science and Information Technology,
Imam Abdulrahman Bin Faisal University,
P.O. Box 1982, Dammam 31441, Saudi Arabia
[1] 2190500152@iau.edu.sa

Dr.Rachid Zagrouba[2]
College of Computer Science and Information Technology,
Imam Abdulrahman Bin Faisal University,
P.O. Box 1982, Dammam 31441, Saudi Arabia
[2] rmzagrouba@iau.edu.sa

Abstract— **Due to the massive proliferation of Short Message Service (SMS), Spammers got the interest to dig their way into it in the hope to reach more targets. Spam SMS can trick mobile users into giving away their confidential information which can result in severe consequences. The seriousness of this problem has raised the need to develop an accurate Spam filtration solution. Machine learning algorithms have emerged as a great tool to classify data into labels. This description fits our case perfectly as it classifies SMS into two labels: spam or ham. This paper will tackle the SMS spam filtration solutions by introducing a hybrid system using two types of machine learning techniques: supervised & unsupervised machine learning algorithms. The new hybrid system is designed to achieve better spam filtration accuracy and F-measures**

Keywords—SMS, Spam Filtering, Supervised Machine Learning, Unsupervised Machine Learning, Security

I. Introduction

Communication through mobiles has witnessed substantial growth over the past few years. Statistics predict that mobile users accessed over messaging will increase to 2.48 billion in 2021 [1]. As more people are relying on the mobiles for communication, Short Message Service (SMS) has become a vital point in the commercial industry. SMS is text communicating tool that usually has a limited number of characters (160 seven-bit characters or less). SMS has gained its popularity from being considered as an affordable & instant way of sending text messages between users. However, this proliferation has caught the attention of cybercriminal, who became eager to take advantage of this technique for their profits. Spam SMS usually contain a malicious link(s), phone numbers, binary data, etc. The cybercriminal takes into account the fact that users trust SMS to receive important notifications such as financial notifications, Jobs application responses and latest offers from the network service provider. For this reason, mobile users can give away their personal information mistakenly thinking they have been asked by their bank or by their service provider. The results of such actions can be catastrophic.

There was an interesting observation which noted that the total count of spam messages has exceeded the total of spam emails in recent years [2]. The other thing to notice is that the SMS cost is decreasing year by year which in return has played a rolling hand in increasing its popularity. For example, in China, the SMS cost rate falls under $ 0.001 [3]. On the other hand, this decrement has helped spams spread more. The fact is that there still little real-life implementation of SMS filtering by Network Service Provider is making this subject an interesting one to explore.

To solve such problems, some organization has put effort into raising people awareness of not giving away sensitive information over SMS or calling suspicious numbers listed in an SMS. Another effort has been proposed by a Saudi company called Saudi Telecom Company (STC) to prevent SMS spam by encouraging users to have a roll in expanding the SMS spam database. They have provided a unified public number to report any suspicious SMS spam sender#. However, these methods have shown a little effect on the average spam S
MS received daily.

Nowadays, machine learning applications have emerged as an excellent tool for classifying information into different labels. That description satisfies this research objective perfectly as it intends to classify SMS into two labels: Spam or Ham. This paper will add to the body of knowledge by introducing a hybrid system using multiple machine learning techniques to efficiently filter the spam messages based on their contents and similar features.

This paper is organized as follows: The first section will include an introduction. The second section will cover the background of the research topic. The third section will include the problem statement & literature review. Forth

978-1-7281-8855-3/20/$31.00 ©2020 IEEE

section will discuss the methodology in which the research was conducted. The fifth section will discuss the system results and findings. Lastly, the sixth section will include a conclusion and suggestion for future work.

II. BACKGROUND

SMS Spam messages are any type of unwanted or harmful messages, such as advertisements, frauds, business services, etc. They annoy end users, consume the resource of mobile devices, including memory spaces, and lead to overloading SMS channels.

To develop an efficient spam filtering solution, it has to support important features [4] which listed as follows:

1- Real-time filtering support: SMS spam filtering should make decisions on real-time in order for it to be useful for mobile device users.
2- Self-learning: SMS spam filtering should include machine learning algorithms to adapt to a new kind of spams as spammers are changing their ways constantly.
3- Accuracy: the system should be designed with the highest possible accuracy.
4- No-Deletion: the system should not discard any suspected spam as it might be misclassified and important to the users.
5- Black & White Lists: the filtration system should keep track of suspected sender patterns and add them to the blacklist of spammers.
6- Privacy: SMS consider as a type of private communication and should not be vulnerable to any type of misuse by third-parties.
7- Personalization: spam msg classification defer from one person to the other; therefore, the system should adapt to user preferences.

SMS filtration services fit well with supervised machine learning algorithms as a classification tool. Supervised ML works by being fed by trained data labels, and then it gains the ability to predicts labels on new data. Understanding the common characteristics of A spam SMS is a crucial part of training a spam filtration system. A spam SMS often contains some special keywords, such as “free” or “winner,” it might include extensive use of punctuation marks and capital letters, such as “BUY!!” or “MONEY”, or it includes phones numbers and personal information requests. Such keywords become the basic training knowledge of a machine learning-based spam classification model. The steps of building such a model are shown in Figure 1.

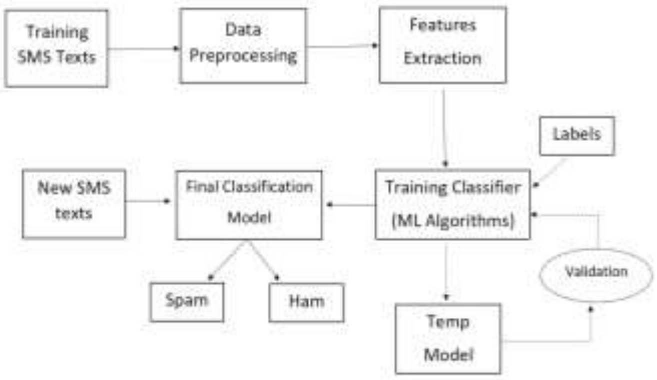


Figure 1: SMS spam filtration model based on supervised machine learning

Many SMS Spam messages detection techniques are available these days to block spam messages and filter them. The most famous machine learning algorithm used in spam filtering is the following:

- Naive Bayes (NB): it is considered one of the simplest classification methods. It got its name from the famous Bayes theorem, and it works by learning from observing each feature individually. In NB, “every frequent word is considered not only mutually independent but also single, independent and mutually exclusive” [5].

- Random Forests: A random forest is just a composition of grouped trees. The way random forest algorithm works is that each branch of the tree can produce predictions that are somehow different from other trees. That difference can give us a better generalization of the results by averaging them.

- Logistic regression: This type of algorithm best suits binary classification. The goal of logistic regression is to find the best fitting that describe the relationship between dichotomous features. In another algorithm, our goal is to select parameters that minimize the sum of squared errors like in Naïve Bayes. However, logistic regression chooses parameters that maximize the likelihood of observing the sample values [2].

- Support Vector Machine (SVM): the principle view of SVM is to find an “Optimal” hyperplane that best classifies the learning data. The optimal solution produced by SVM is characterized by having the maximum margins from the input data.

The most straightforward formulation of SVM is a linear one.

- AdaBoost

AdaBoost is short for Adaptive Boosting. AdaBoosting was the first successful boosting algorithm developed for binary classification. Each instance in the training dataset is weighted. A weak classifier is prepared on the training data using the weighted samples. Only binary classification problems are supported. So each decision stump makes one decision on one input variable. And outputs a +1.0 or -1.0 value for the first- or second-class value. The misclassification rate is calculated for the trained model. Traditionally, this is calculated as:

$$Error = \frac{Correct - N}{N}$$

Where *Error* is the misclassification rate. On the other hand, *Correct* is the number of training instance predicted by the model. Furthermore, N is the total number of training instances [9].

On the other hand, there is another type of machine learning algorithm that does not require any prior training called unsupervised machine learning. Unsupervised ML techniques work toward analyzing raw (unlabeled) datasets, thus helping in generating analytic insights from that row data as humans does efficiently [6]. This type of algorithm depends on approximation because in contrary to supervised ML, the actual desired outcome is not defined. For example, suggesting new products for customers.

Unsupervised ML algorithms have evolved and caught the attention of researcher all around the world because of its strong ability to learn on completely strange environments. Some of its applications are

- Data clustering: grouping data points based on how much are they similar to each other. The process should be mutually exclusive meaning a data point should not belong to two groups at the same time. However, it is prone to overfit the data point on a certain group by falsely measuring the similarity[].

- Dimensionality reduction: this type of application reduces the number of features among the data sets because real-life data usually contain a massive number of dimensions (features). However, affecting features are usually contained within a small number. For this reason, those features are the one that became the target for extraction, and the rest is ignored.

- Feature learning: this application preserves the typical behaviour patterns to form specific rules on what should go next on the pattern. This type works best in shopping experience to follow customers preferences.

- Anomaly detection: this type of application focus on defining the outliers among data points. This type of application suits the best real-life problems that include fault and virus detection.

Most recently, it has advanced to include development in some artificial intelligence fields such as "deep learning" [7].

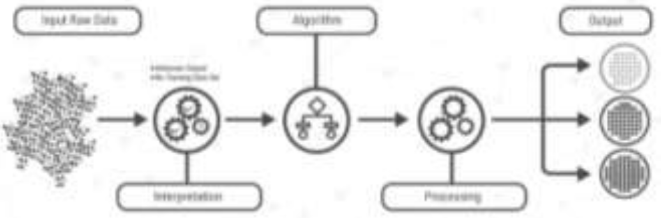


Figure 2: Unsupervised Machine Learning Model [8]

III. Literature Review

Based on the vast number of SMS users, the short message centre (SMC) receives a massive amount of messages stream at a very high speed. Spam messages only make things worse, as they usually sent rapidly and automatically, which will eventually increase the messages stream. Thus, the current inbox design is somehow random until users themselves classify the SMS's manually and personalize then according to their preceptive of wanted vs unwanted messages. Also, in an inbox that full of accumulated messages (ham messages mixed with spams), finding a particular message can be a tedious task. Spam filtering has always been a challenge that interest researchers from all around the world due to its massive scalability. Recently, machine learning algorithms have been introduced in many research papers as a solution to this problem. Researchers like Sethi et al. [2] have contributed to comparing the performance of different ML algorithms. They have tested Naïve Base, Random Forest and Logistic Regression and concluded that Naïve Bayes performed the best among the others. Gupta et al. [9] have compared 8 different classifiers. Their comparison concluded that Convolutional Neural Network Classifier scores the highest accuracy of 98.25% - 99.19%. Healy et al. [10] have compared three different classifiers, which are K-nearest-neighbor, Support Vector Machine and Naïve Based. Their results showed that Naïve based & SVM has significantly outperformed KNN classifier. Memoona et al. [11] have compared focused on a different type, which is unsupervised machine learning algorithms. Their research paper has conducted a comparative survey between the unsupervised algorithms used for Automation, Classification and Maintenance. In comparison, they have considered against 16 parameters such as performance, accuracy, robustness, complexity, reliability, etc.

In any research area, there is always a place for improvement. Many researchers have suggested new ways to integrate ML techniques into SMS spam filter in the hope to achieve better performance. Boujnouni [12] have proposed an SMS spam filter based on three components: N-gram to extract features from text messages, enhanced version of SVM and information gain ration to specify the most relevant features. His experiment result has shown that choosing an optimal value for suggested filter parameters can make it reach an accuracy of 95.13% in the training set & 89.32% in the testing set. J. Zhang et al. [13] have suggested another intelligent solution called message topic model (MTM) that based on famous probability topic model (PTL). This model was initially created to overcome the sparsity problem in SMS spam classification. It has successfully proven to be a suitable model for text messaging classification. M. Rafiq & S. Abulaish [14] has proposed a novel graph-based filter architecture to detect SMS spams on mobile phones. Their GLM scheme has produced during the classification process a remarkable 98% detection rate and a false alarm rate of 0.08%. On the other side, Q. Xu et al. [15] have shown concern that most ML filters depend on content-based algorithms which violate user's privacy in certain ways. They proposed a non-content filtering scheme that process on the service side by using graph data mining to recognize suspicious behaviour from ordinary senders. In E. Ezpeleta et al. [16] paper, they have suggested adding the personality influence measure in the spam filtration process, which proved to improve accuracy up to 98.94%, and the false-positive rates.

Following up from the previously mentioned contributions, this paper will contribute to the field of knowledge by introducing a new hybrid system that includes two types of machine learning techniques: supervised & unsupervised. As spammers continuously changing their style of writing spam SMS, the benchmark in the proposed system is that it will achieve higher accuracy by distinguishing both trained spam features and the unknown ones.

IV. PROPOSED SOLUTION

In this section, the general design of the workflow of the hybrid system development process will be explained. The main tools used in our system is machine learning tools for both analyzing and classification purposes. The programming language used to develop this system is Python as huge machine learning libraries support it. The IDE used to develop the solution is called Jupyter Notebook by Anaconda.

Every supervised ML system needs a useful data set to be trained on in order for it to be sufficient enough to predict to new results. The dataset used in training and testing the current hybrid system was collected from UCI [17]. It contains 5574 instances of SMS messages in English. Each data only includes two feature, which is the label (spam or ham) and the content of the text message. Secondly, the data was converted from text form to CSV (Comma Separated Value) form for easier feature extraction. Then preprocessing is done by cleaning the data by removing punctuation marks and removing stop words as they appear very often and cannot help in concluding the final classification. Furthermore, the general data is analyzed by implementing various feature extractions techniques.

Another popular feature abstraction technique is called Term Frequency-Inverse Document Frequency. It is a statistic feature extraction method that used to evaluate the importance of the word to a document in a corpus.it is a is often used to create a vector space model in information retrieval. It evaluates the importance of a word within the contest. The importance increases proportionally with the number of times that a word reappears in a document, compared to the inverse proportion of the same word in the whole collection of documents. [18]

Then, the data is separated into two sets, 67% training set and 33% test set to be used later for the performance analysis. After that, a different classifier is applied and analyzed according to their precision and accuracy in classifying the data. Finally, the Confusion Matrix is obtained from the data set. The Confusion Matrix can bring us a powerful insight into the performance of the classifier by stating the results of F-measures. The most used measures to test the performance of an algorithm are precision, recall and accuracy, which can be calculated according to the following equations:

$$Precision = \frac{TruePositive}{TruePositive + FalsePositive}$$

$$Recall = \frac{TruePositive}{TruePositive + FalseNegative}$$

$$Accuracy = \frac{TruePositive + TrueNegative}{Total}$$

The meaning of F-measures results can differ from one application to another. In our case, false positive is more critical because if one important ham message was classified as spam, the users might miss it. For that reason, precision gained higher priority over the recall. After analyzing the results of supervised classifiers, the one with the best performance will be used for the final hybrid system.

After that, the unsupervised machine learning algorithm will be developed and analyzed. Finally, the hybrid system will take place in the development process. Its implementation depends on the concept Head – to – Head, meaning that the result of the two models will support each other. If both models agree on the same label; the prediction percentage will increase.

This can be summed up as follows:

- Preparing the dataset
- Preprocessing of data.
- Building the classifier
- Convert text to a vector class.
- Building a classified model.
- Train model with the labeled training dataset.
- Test given model with the trained dataset.
- Analyze the result using different machine learning tool.
- Choosing the best-supervised machine learning algorithm according to their performance.
- Building the unsupervised machine learning model
- Combine the previous chosen supervised model with the unsupervised model.
- Analyze the final results using machine learning tools

V. Results & Findings

The process of building the system started by tokenizing the message contents into sperate words (tokens) so that each word can be treated separately. All tokens have been converted to lower case letters so it can be compared correctly with other strings. Then, the data has been preprocessed and cleaned by removing stop words, punctuation marks and words with less than three letters. Also, any null values or rows in the dataset was removed. After this step, the initial examination has been made to analyze common features in an SMS spam message. First, we analyzed the most common words used in spam and ham messages according to how often they appear as listed in the following figures:

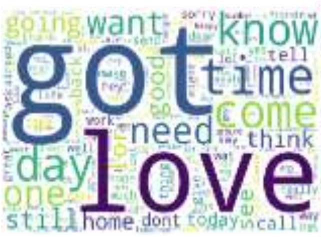


Figure 3: Most Common words in SMS Spam Message

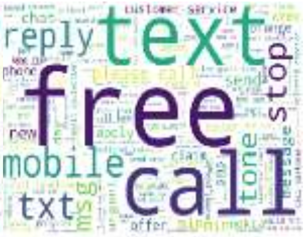


Figure 4: Most Common Word in SMS Ham Message

As we notice from the previous figures, spam usually contains words such as free and call. On the other hand, ham messages contain simple life words such as love and time. The second observation was about the length of spam messages vs ham messages. Looking from the below graphs, spam message tends to have higher messages length (approximately above 130 characters) than ham messages (who mostly falls around 100 characters).

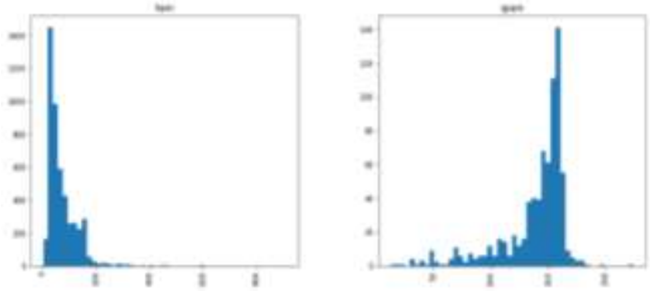
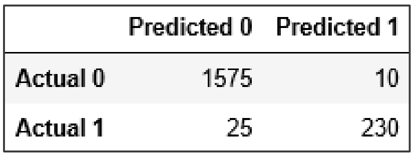


Figure 5: Length of SMS Spam and ham messages

a) Supervised Machine Learning Models

At the first implementation phase, Six different supervised classifiers were trained and tested using the SMS dataset to classify SMS messages into spam or ham. The first one is the Naïve Base classifier. We created different model instances by changing alpha values. Searching from the resulted models for one model that has the least False Positive Rate has yielded a final model has achieved with 98% precision and 97.4% accuracy.

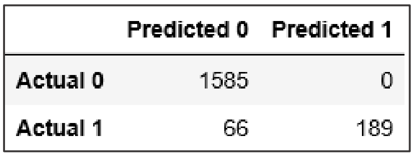
Table1: Confusion Matrix of the Final Naive Base Model



	Predicted 0	Predicted 1
Actual 0	1575	10
Actual 1	25	230

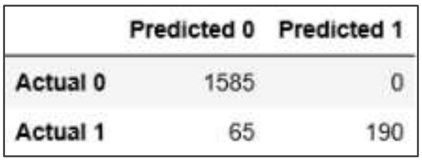
The second classifier is called Support Vector Machine. We trained different SVM models by changing the regularization parameter C, and finally, we obtained the best model with 0% False positive rate. The final model misclassified 66 spam messages as ham. The final model achieved 100% precision with 96% accuracy rate.

Table 2: Confusion Matrix of the final SVM Model



	Predicted 0	Predicted 1
Actual 0	1585	0
Actual 1	66	189

The third classifier is logistic regression. This classifier defines the relationship between the dependent variable (1: spam or 0: ham) and other variables which are the message contents features. This classifier resulted in 100% precision rate and 96% accuracy.

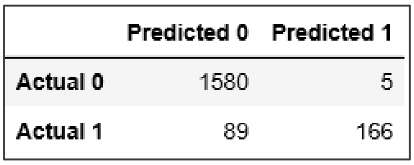


	Predicted 0	Predicted 1
Actual 0	1585	0
Actual 1	65	190

Table 3: Confusion Matrix of Logistic Regression Model

The fourth classifier is the most famous ML model, which called the K-Nearest Neighbor (KNN). It is KNN is a non-parametric, lazy learning algorithm. This algorithm strength lays on the fact that takes little or no consideration on the distribution of data. It focuses only on the feature similarity by a majority of votes among its neighbours. When it tested for performance on our dataset, it has achieved 94% accuracy.

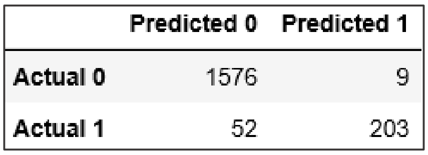
Table 4: Confusion Matrix of KNN



	Predicted 0	Predicted 1
Actual 0	1580	5
Actual 1	89	166

The fifth classifier is decision trees. In this model, features become part of a series of tryouts by considering different split points. Each path is then tested using a cost function, and the best path with the best cost (or lowest) will be chosen for the final model. This model has yielded 96%.

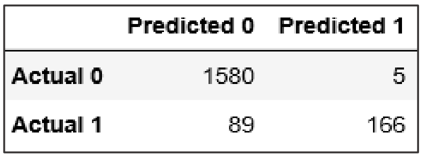
Table 5: Confusion Matrix of Decision Tree



	Predicted 0	Predicted 1
Actual 0	1576	9
Actual 1	52	203

The final classifier is one of the advanced ensemble methods called AdaBoost classifier. The ensemble model is another way of saying the combined ML model. It combines different low-performance ML models in order to achieve a better-combined performance. In the process, the vote of each classifier is considered, and then the final model is made by the majority of votes. This model has achieved a 94% & 95% precision rate.

Table 6: Confusion Matrix of AdaBoost



	Predicted 0	Predicted 1
Actual 0	1580	5
Actual 1	89	166

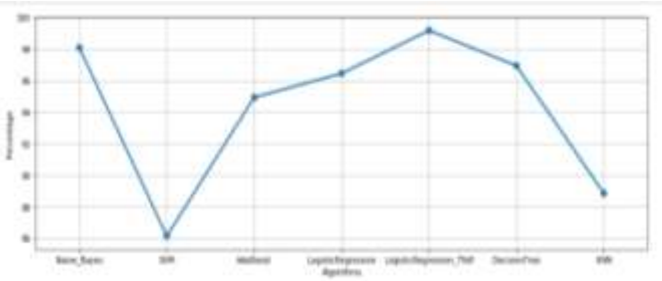


Figure 6: Comparison of supervised Machine Learning Algorithms

By observing the results of six supervised models: Naïve base, SVM, Logistic Regression, KNN, Decision Trees and AdaBoost. It is clear that SVM outperformed them and achieved the best accuracy along with the best precision in classifying the test set into spam and ham. For this reason, SVM has been chosen to be the supervised classifier in the final hybrid system. However, we will try to combine the other supervised models and see if their performance is raised higher within the hybrid system.

b) Unsupervised Machine Learning Model

At the next step, one unsupervised model was implemented and tested on the SMS data set, which k-means clustering. It does not require any prior training, so it only dealt with the message contents. It divides and assigns message tokens iteratively into groups or clusters based on the features that are provided. By comparing the results of this model and the accuracy file that only contains true data labels, this model has achieved 96.03% accuracy.

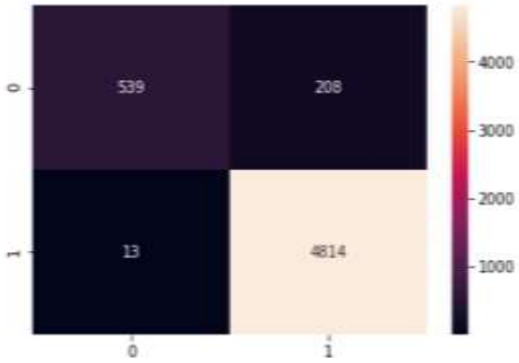


Figure 7: Heat map of K-means Clustering

c) Hybrid System

At this stage, the combination process has taken place by following simple rules: unlabeled dataset will be fed to the k-means model, and the resulted labels will be saved on a new data set. The resulted data set then will be divided training and testing data set and then fed to one of the supervised machine learning models. This strength of this combination concept is robust is that the hybrid system will be able to be used in any data without the need of acquiring prior training.

Only the messages from our dataset were extracted and preprocessed for word tokenization & cleaning. Then the resulted messages were fed to k-means model to predict a label for each message using clustering. In this case, it was only two clusters (0: Ham or 1: Spam). Then the resulted from the labelled data set was used with three different supervised classification model: NB, SVM and Logistic Regression. The accuracy & precision of the combined hybrid model was as follows:

Table 7: Comparison Between final Hybrid models

Model Name	Accuracy	Precision
K-means_NB	96.7%	99.8%
K-means_SVM	**98.8%**	**99.2%**
K-means_LR_Tfidf	98.65%	99.33%

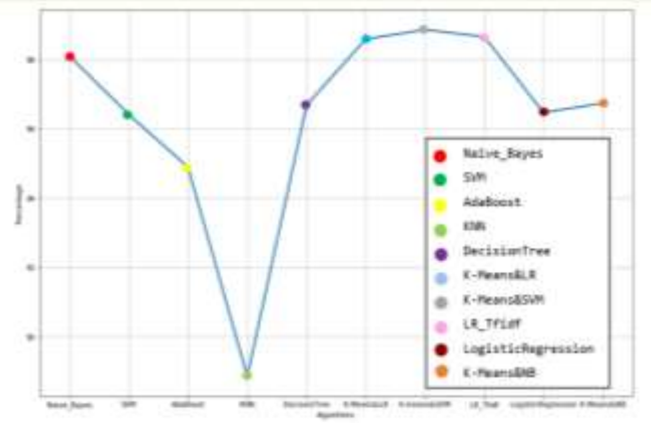


Figure 8: Comparison Between Final Hybrid models

As clear from the comparison, the hybrid system of k-means with support Vector machine has outperformed all others by 98.8% accuracy. So, this model is best suited to be used as a model for SMS spam detection system. However, if the system service provider is interested mostly in test precision regardless of its accuracy, it is better to use SVM individually as it achieved 100% precision rate. Another observation that the hybrid system has performed better than each of its model on its own except in k-means with NB. Naïve base had performed better individually when it was trained on true data rather than the resulted clustering data.

VI. CONCLUSION & FUTURE WORK

Since communication though mobile phones are increasingly becoming a necessity, subscribers became more demanding on having a safe mobile network. In this paper, different supervised ML classifiers have been tested based on performance and then tested by being a part of the final hybrid model. By comparing six different supervised models, SVM showed to have the highest precision with 0 false-positive rates and KNN has the least performance among other classifiers. Then, after implementing a different combination of hybrid models, merging K-means with SVM has stood out with the highest accuracy of 98.8%. So, even though overall, supervised algorithms have performed more accurately than the unsupervised algorithm(k-mean), at the end hybrid system has achieved much better than a single classifier. So, we can conclude that using a hybrid system best suits the spam filtering system since it takes advantage of both schemes of ML models and guarantees to deliver better classification results.

For future work, more ML models can be tested and tried out to be part of the hybrid system. Furthermore, the results of can be brought up a little bit by adding more preprocessing steps such as including more Wight to $ sign in spam classification. Lastly, a real-time application can be developed to interpret the suggested hybrid model and tested on real-time performance.

References

[1] "statista," 2017. [Online]. Available: https://www.statista.com/statistics/483255/number-of-mobile-messaging-users-worldwide/. [Accessed February 2019].

[2] P. Sethi, V. Bhandari and B. Kohli, "SMS spam detection and comparison of various machine learning algorithms," in *2017 International Conference on Computing and Communication Technologies for Smart Nation (IC3TSN)*, 2017.

[3] D. D. Arifin, Shaufiah and M. A. Bijaksana, "Enhancing Spam Detection on Mobile Phone Short Message Service (SMS) Performance using FP-Growth and Naive Bayes Classifier," in *The 2016 IEEE Asia Pacific Conference on Wireless and Mobile (APWiMob)*, 2016.

[4] K. Yadav, P. Kumaraguru, A. Goyal, A. Gupta and V. Naik, "SMSAssassin: crowdsourcing driven mobile-based system for SMS spam filtering," in *In Proceedings of the 12th Workshop on Mobile Computing Systems and Applications (pp. 1-6). ACM.*, 2011.

[5] Han, Jiawei, M. Kamber and J. Pei, Data mining: concepts and techniques, 3rd Edition. Morgan, 2013.

[6] M. Usama, J. Qadir, A. Raza, H. Arif, K.-L. Yau, Y. Elkhatib, A. Hussain and A. Al-Fuqaha, "Unsupervised Machine Learning for Networking: Techniques, Applications and Research Challenges," 2017.

[7] Y. LeCun, Y. Bengio and G. Hinton, "“Deep learning,” „" *Nature,* vol. 521, no. 7553, p. 436–444, 2015.

[8] R. v. Loon, "Machine learning explained: Understanding supervised, unsupervised, and reinforcement learning," www.bigdata-madesimple.com, 2018. [Online]. Available: https://bigdata-madesimple.com/machine-learning-explained-understanding-supervised-unsupervised-and-reinforcement-learning/.

[9] M. Gupta, A. Bakliwal, S. Agarwal and P. Mehndiratta, "A Comparative Study of Spam SMS Detection using Machine Learning Classifiers," in *Proceedings of 2018 Eleventh International Conference on Contemporary Computing (IC3), 2-4 August, 2018, Noida, India*, 2018.

[10] M. Healy, A. Zamolotskikh and S. J. Delany, "An Assessment of Case Base Reasoning for Short Text Message Classification," in *Proceedings of the 15th. Irish Conference on Artificial Intelligence and Cognitive Sciences (AICS'04), Castlebar*, 2004.

[11] M. Khanum, T. Mahboob, W. Imtiaz, H. A. Ghafoor and R. Sehar, "A Survey on Unsupervised Machine Learning Algorithms for Automation, Classification and Maintenance," *International Journal of Computer Applications,* vol. 119, no. 13, p. 34, 2015.

[12] M. E. Boujnouni, "SMS Spam Filtering Using N - gram method, Information Gain Metric and an Improved Version of SVDD Classifier," *JOURNAL OF ENGINEERING SCIENCE AND TECHNOLOGY REVIEW,* Vols. 10. 131-137. 10.25103/jestr.101.18. , 2017.

[13] Y. Zhang, J. Ma, J. Liu, K. Yu and X. Wang, "Intelligent SMS Spam Filtering Using Topic Model," in *2016 International Conference on Intelligent Networking and Collaborative Systems*, 2016.

[14] M. Z. Rafique and S. Muhammad Abulaish, "Graph-Based Learning Model for Detection of SMS Spam on Smart Phones," in *2012 8th International Wireless Communications and Mobile Computing Conference (IWCMC)*, 2012.

[15] Q. Xu, E. W. Xiang, Q. Yang, J. Du and J. Zhong, "SMS Spam Detection Using non-content Features," *IEEE Intelligent Systems,* vol. 27, no. 6, 2012.

[16] E. Ezpeleta, U. Zurutuza and J. M. Gómez, "Short Messages Spam Filtering Using Personality Recognition," in *the 4th Spanish Conference*, 2016.

[17] T. A. Almeida and J. M. G. Hidalgo, "Machine Learning Rospoitory, SMS Spam Collection Data Set," 2012. [Online]. Available: https://archive.ics.uci.edu/ml/datasets/sms+spam+collection#:. [Accessed Jan 2019].

[18] U. Erra, S. Senatore, F. Minnella and G. Caggianese, "Approximate TF–IDF based on topic extraction from massive message stream using the GPU," *Information Sciences,* p. 292:143–161, 2015.