Future Computing and Informatics Journal

Volume 7
Issue 1 (2022) *Issue 1*

Article 1

2022

An Empirical Study Towards an Automatic Phishing Attack Detection Using Ensemble Stacking Model

Mahmoud Othman
Future University in Egypt , Egypt, msamy@fue.edu.eg

Hesham Hassan
Cairo university , Egypt, h.hassan@fci-cu.edu.eg

Recommended Citation

Othman, Mahmoud and Hassan, Hesham (2022) "An Empirical Study Towards an Automatic Phishing Attack Detection Using Ensemble Stacking Model," *Future Computing and Informatics Journal*: Vol. 7: Iss. 1, Article 1.
DOI: http://Doi.org/10.54623/fue.fcij.7.1.1
Available at: https://digitalcommons.aaru.edu.jo/fcij/vol7/iss1/1

Future Computing and Informatics Journal

Homepage: https://digitalcommons.aaru.edu.jo/fcij/

doi: http://Doi.org/10.54623/fue.fcij.7.1.1

An Empirical Study Towards an Automatic Phishing Attack Detection Using Ensemble Stacking Model

Mahmoud Othman [1, a], Hesham Hassan [2, b]
[1] Faculty of Computers and Information Technology, Future University in Egypt
[2]Faculty of Computers and Artificial Intelligence Cairo University
[a] msamy@fue.edu.eg, [b] h.hassan@fci-cu.edu.eg

ABSTRACT

Phishing attacks are one of the most attacks facing internet users, especially after the COVID-19 pandemic, as most organizations have transferred part or most of their work and communication to become online using well-known tools, like email, Zoom, WebEx, etc. Therefore, cyber phishing attacks have become progressively recent, directly and frankly reflecting the designated website, allowing the attacker to observe everything while the victim is exploring Webpages. Hence, utilizing Artificial Intelligence (AI) techniques has become a necessary approach that could be used to detect such attacks automatically. In this paper, we introduce an empirical analysis for automatic phishing detection using several machine learning classification algorithms compared with an ensemble learning model for detecting phishing sites based on the uniform resource locator (URL) using two preprocessed datasets. In this empirical study, we concluded that the ensemble model grants accuracy 97.49% for dataset 1 and 98.69% for dataset 2, which gives higher accuracy than using a single machine learning classification algorithm such as Naive Bayes (NB), Random Forest (RF), Decision Trees (DTs), K-Nearest Neighbors (KNN), and linear and Quadratic Discriminant Analysis. We also compared the proposed ensemble model with one of the most recent similar models.

Keywords: Ensemble Learning, Machine Learning, Phishing Detection, Ensemble Stacking

1.Introduction

Phishing sites are typical online entry of social attacks, as many sites continuing scams [1]. In this type of attack, attackers create website pages by copying original sites and sending them fake URLs for targeted victims through spam emails or online social networks, e.g., WhatsApp and Facebook. They are probably targeting the victim to get their personal or sensitive data. On the other hand, technology becomes more advanced, and the used techniques of cybercriminals become more advanced to prevent phishing attacks. In addition, users should know how the attackers do it and be familiar with anti-phishing techniques to protect themselves from becoming victims. Still, many users are not familiar with these attacks.

The general method to detect phishing sites is by updating blacklisted URLs and IP on the antivirus database, also known as "blacklist" method. But the attackers avoid this by using creative techniques to trick users by modifying the URL to look legitimate via blackout. Also, many other simple techniques, like quick camouflage, agents are automatically created to host a web page and create an algorithm for generating new URLs, etc. The main drawback of this method is that it cannot detect “zero-hour” phishing attacks. While the detection using Heuristic and data mining, which includes the characteristics found in phishing attacks, can detect a phishing attack in the zero-hour. But, the presence of these characteristics cannot always be guaranteed in such attacks, and its accuracy is very low as the false positive rate in discovery is very high [2].

To protect users from such deceptive attacks, we need a technology that can quickly detect new types of phishing attacks through automation. So, artificial intelligence techniques become very important to extract the necessary information to detect and block phishing automatically.

In this study, we will concentrate on the type of phishing that uses the Uniform Resource Locator (URL) aiming to:

1. Conduct an empirical study for automatic phishing detection using a set of well-known machine learning classifiers as a single classifier comparing them with an ensemble stacking model.
2. Conduct a comparison between the proposed ensemble model with the previous related works.

2. Related Works

Discovering deceptive websites becomes a significant classification problem using machine learning models. Different works proposed solution for detecting the phishing attack.

James et al. [3] propose an approach that reads the URL and analyzes it where the hostname and path are used to categorize it into a phishing URL or not. They used four classification algorithms: DT, (NB), (KNN, and Support Vector Machines (SVM); with the best accuracy 89.75%. The datasets that they used are generated by Alexa and Phishtank. Subasi et al. [4] used a set of machine learning algorithms which is an Artificial Neural Network (ANN), KNN, Random Forest (RF), and SVM. They claim that the RF that has the best accuracy of 97.26%. Also, Mao et al. [5] used various classifications algorithms, including SVM, RF, DT, and AdaBoost, to predict a phishing attack. Joshi et al. [6] also used the RF algorithm as a classifier using the dataset generated from the Mendeley site, which is provided as an input into the feature selection algorithm to identify the effective features. After that, they train the RF algorithm on specific features to predict a phishing attack. Adebowale et al. [7] used the Adaptive Neuro-Fuzzy Inference System using integrated features to detect phishing and protection attacks. Alsariera et al. [8] proposed a descriptive algorithm for phishing URLs. They utilized four models, which are BET, ABET, LBET, and ROFBET. Sahingoz et al. [9] expressed that Phishtank did not offer free dataset on the internet page, so they have made their dataset. They employed Natural Language Processing (NLP) to distinguish the phishing URL. The dataset they have created contains 73,575 URLs, 37,175 of them are phishing URLs. Abdel Hamid et al. [10] created a method called eDRI for detecting phishing attacks. They used a feature extraction algorithm using ANOVA to reduce features; their results showed a 93.5% accuracy.

Hutchinson et al. [11] proposed work on discovering phishing sites focused on the feature's selection. They utilized the UCI Machine Learning Repository dataset, containing 11,055 URLs and 30 features; they divided these into six groups. After the experiments, they choose three groups that are suitable for accurately detecting a phishing attack. Also, Tyagi et al. [12] utilize the same dataset. They utilized ML algorithms such as DT, RF, GBM, GLM, and PCA.

Al-Sarem et al. [13] proposed an optimized stacking ensemble method. They used a genetic algorithm (GA) for optimization to tune the features of several ensemble learning methods; they include random forests, AdaBoost, XGBoost, Bagging, GradientBoost, and Light. They conducted their experiments on three datasets from UCI and Mendeley, which are publicly available datasets. They also compared their results with previous works that use dataset 1 and dataset 2 while they stated that no previous works used dataset 3 as it is recently published.

Most of the previous studies state that detection accuracy is reasonable using machine learning classification algorithms. Also, most of these works mentioned the limitations of their work and common limitations of not using ensemble learning techniques, and in some studies, features have not been reduced.

In this work, we used dataset 3, which we named dataset 1, and compared our results with Al-Sarem et al. [13], and we found that the GA optimization they used has no significant improvement.

3. Dataset

Grega et al. [14] have prepared a dataset containing phishing and legitimate website instances. A set of features are used to represent each site and indicate whether it is a phishing site or not. This dataset is based on the URL resolving metrics, URL properties, and external services.

There are two different versions of this dataset, one with a total of 58,645 instances and the second version consists of 88,647 instances, with more instances with label legitimate. The purpose is to simulate the real-life situation where there are more legitimate websites [14]. The two versions are summarized in table 1

Table 1: Datasets Summary

Dataset	Legitimate	phishing	Total
dataset 1	27,998	30,647	58,645
dataset 2	58,000	30,647	88,647

3.1.Features

The datasets in total contain 111 features except for the class. The features of the datasets are divided into six groups based on [14]:

- URL properties.
- Domain properties.
- URL directory properties.
- URL file properties.
- URL parameter properties.
- URL external metrics and resolving data.
-

3.2.Methods

In this paper, we use a set of classification algorithms: NB, DT, RF, KNN, LDA, and QDA as a baseline, and we compared their results as a single machine learning algorithm with the proposed ensemble model. The scoring methods that we use are precision, recall, and F1-score to measure the performance of the classification algorithms and our ensemble model.

4. Machine Learning Algorithms

A. Naive Bayes

Feng Xin et al. [15] define it as:

"a set of supervised learning algorithms based on applying Bayes' theorem with the naïve assumption of conditional independence between every pair of features given the value of the class variable”.

In our experiments, we compared two types of Naive Bayes distributions:

- **Gaussian Naive Bayes (GNB):**

Feng Xin et al. [15] state that :
“This is an algorithm for classification. The likelihood of the features is assumed to be Gaussian”

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

the parameters σ_y and μ_y are estimated using maximum likelihood.

- **Bernoulli Naive Bayes (BNB):**

Korotcov et al. [16] state that BNB is:
“training and classification algorithms for data that is distributed according to multivariate Bernoulli distributions”.
The decision rule for BNB based on:

$$P(x_i \mid y) = P(i \mid y)x_i + (1 - P(i \mid y))(1 - x_i) \qquad \textbf{(2)}$$

In these experiments, we use cross-validation with 100 iterations, each time with 20% data randomly selected as a validation set, and the results are summarized in Table 2. By analyzing the results of the two datasets, we conclude that Bernoulli Naive Bayes is better than GNB in both datasets, and dataset 2 gives more accuracy than dataset 1. The learning curves of the models using dataset 2 are illustrated in figure 1 and figure 2. Also, the performance of each model is shown in figure 3 and figure 4

Table 2: Naive Bayes (NB) Results

Dataset	**Gaussian NB**			**Bernoulli NB**		
	Precision	Recall	F1	Precision	Recall	F1
1	77.3	72.3	**69.8**	78.7	78.7	**78.5**
2	85.9	80.0	**84.1**	86.4	87.7	**88.0**

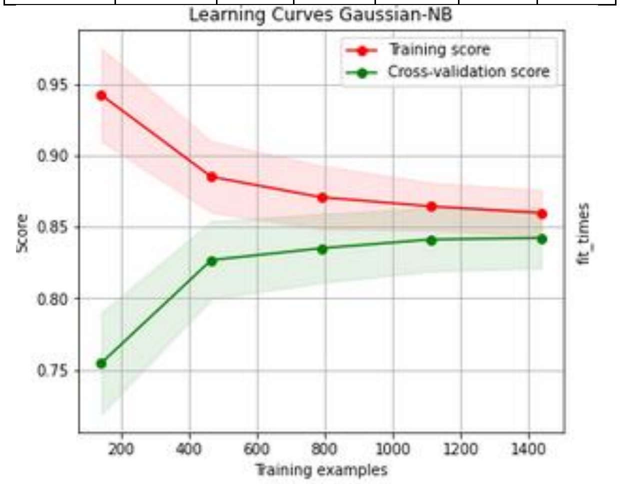

Figure 1: Learning Curves (Gaussian NB)

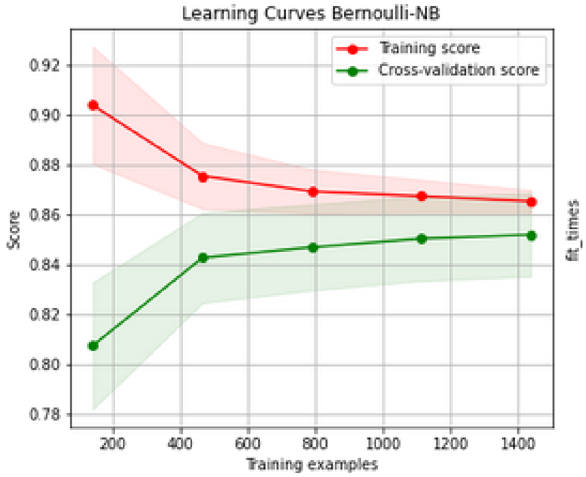

Figure 2: Learning Curves (Bernoulli NB)

DOI: http://Doi.org/10.54623/fue.fcij.7.1.1

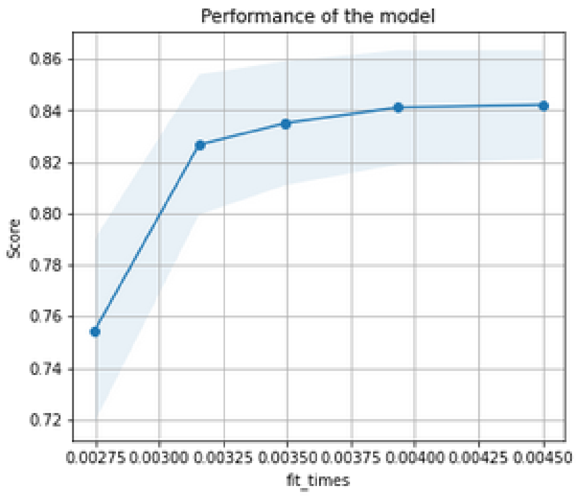

Figure 3: Performance of the Model (Gaussian NB)

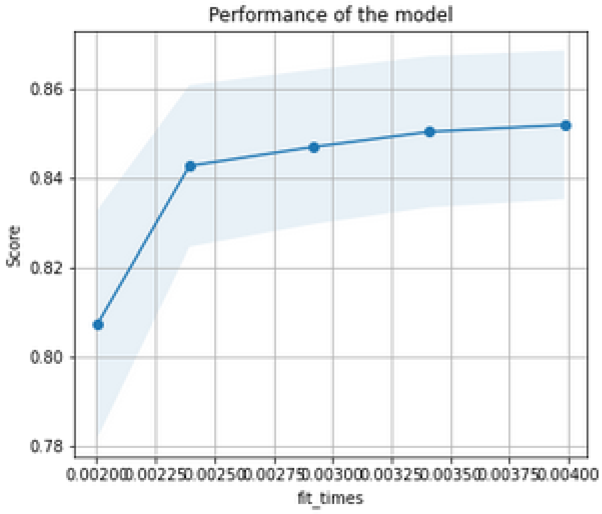

Figure 4: Performance of the Model (Bernoulli NB)

B. Decision Trees (DTs)

Almatarneh el al. state [17] that:

“DTs are a non-parametric supervised learning method used for classification; it predicts the value of a target variable by learning simple decision rules inferred from the data features.”

In this experiment, we also use cross-validation with 100 iterations, each time with 20% data randomly selected as a validation set, and the results are summarized in Table 3. After this experiment, we conclude that the decision tree gives more accuracy than Naive Bayes. The learning curves of the model using dataset 2 are illustrated in figure 5. also, the performance is shown in figure 6

Table 3: Decision Tree Results

Dataset	**Decision Tree**		
	Precision	Recall	F1-Measure
1	92.84	92.83	**92.85**
2	94.69	94.65	**95.18**

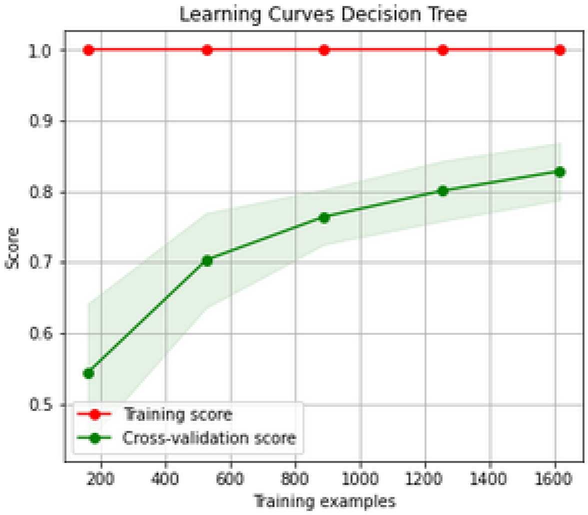

Figure 5: Learning Curves (Decision Tree)

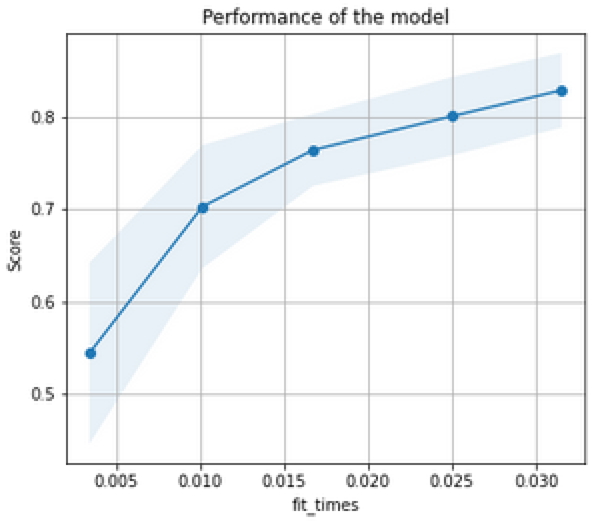

Figure 6: Performance of the Model (Decision Tree)

C. Random Forest Classifier (RF)

Meteier el al [18] state that:

“A random forest is a meta estimator that fits several decision tree classifiers on various sub-samples of the dataset and uses averaging to improve the predictive accuracy and control over-fitting.”

In this experiment, we also use cross-validation with 100 iterations, each time with 20% data randomly selected as a validation set, and the results are summarized in Table 4. After this experiment, we conclude that the random forest gives best results than the previously mentioned models. The learning curves of the model using dataset 2 are illustrated in figure 7. Also, the performance is shown in figure 8.

Table 4: Random Forest Results

Dataset	Random Forest		
	Precision	Recall	F1-Measure
1	95.52	95.48	**95.50**
2	96.59	96.7	**96.9**

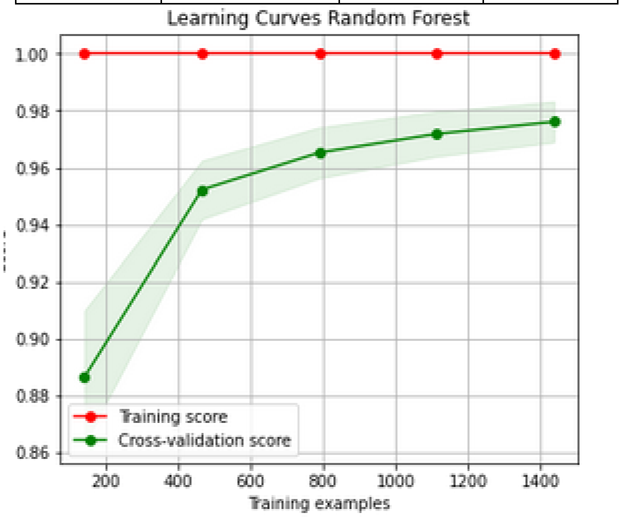

Figure 7: Learning Curves (Random Forest)

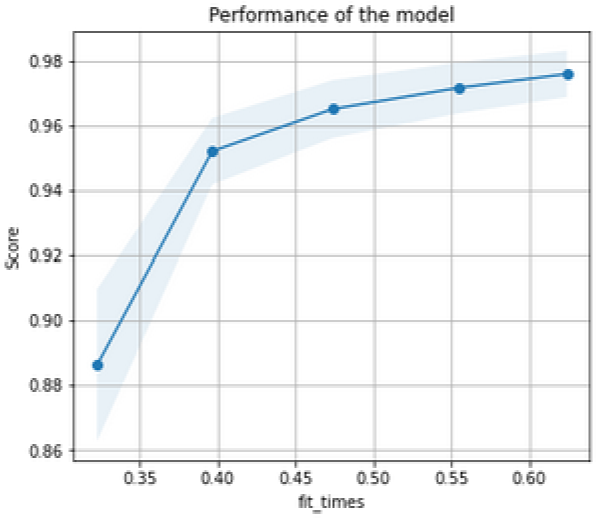

Figure 8: Performance of the Model (Random Forest)

D. K-Nearest Neighbors (KNN)

López-Hernández et al [19] states that:

“KNN is a type of instance-based learning or non-generalizing learning. It does not attempt to construct a general internal model, but simply stores instances of the training data. Classification is computed from a simple majority vote of the nearest neighbors of each point: a query point is assigned the data class which has the most representatives within the nearest neighbors of the point.”

In this experiment, we also use cross-validation with 100 iterations, each time with 20% data randomly selected as a validation set, and the results are summarized in Table 5. After this experiment, we conclude that the random forest still gives the best results. The learning curves of the model using dataset 2 are illustrated in figure 9. also, the model's performance is shown in figure 10.

Table 5: K-Nearest Neighbors Results

Dataset	**K-Nearest Neighbors**		
	Precision	Recall	F1-Measure
1	84.12	83.98	**84.07**
2	85.77	85.41	**86.98**

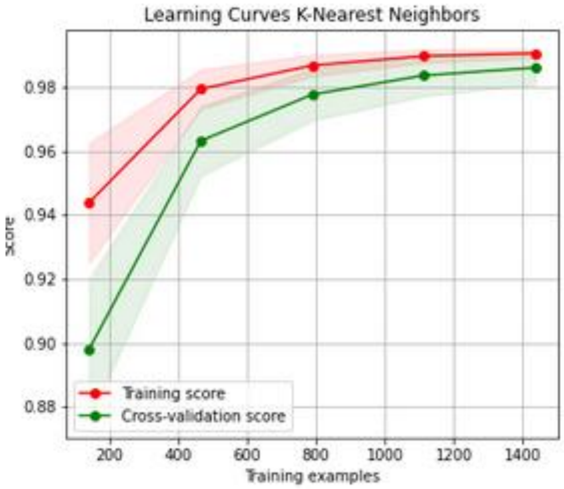

Figure 9: Learning Curves (K-Nearest Neighbors)

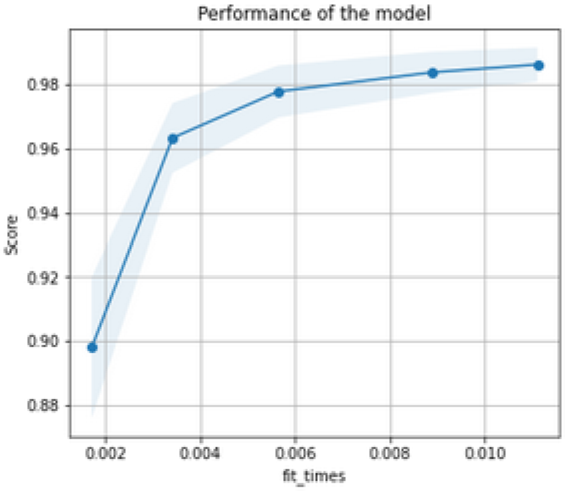

Figure 10: Performance of the Model (K-Nearest Neighbors)

E. Discriminant Analysis

Hasan et al [20] state that:

“Linear Discriminant Analysis (LDA) and Quadratic Discriminant Analysis (QDA) are two classifiers that derived from simple probabilistic models which model the class conditional distribution of the data $P(X|y = k)$ for each class k ”

Predictions can then be obtained by using Bayes’ rule, for each training sample $x \in \mathcal{R}^d$:

$$P(y = k|x) = \frac{P(x|y = k)P(y = k)}{P(x)} = \frac{P(x|y = k)P(y = k)}{\sum_l P(x|y = l) \cdot P(y = l)} \quad \textbf{(3)}$$

and we select the class k which maximizes this posterior probability. The difference is that LDA is a special case of QDA, where the Gaussians for each class are assumed to share the same covariance matrix: $\Sigma_k = \Sigma$ for all k. This reduces the log posterior. In these experiments also we use cross validation with 100 iterations, each time with 20% data randomly selected as a validation set and the results are summarized in table 6. After this experiment we conclude that the random forest still gives the best results and LDA is better than QDA in these datasets. The learning curves of the models using dataset 2 are illustrated in figure 11 and figure 12. Also, the performance of each model is illustrated in figure 13 and figure 14.

Table 6: Discriminant Analysis Results

Dataset	LDA			QDA		
	Precision	Recall	F1-Measure	Precision	Recall	F1-Measure
1	88.5	87.4	**87.6**	76.1	61.3	**53.2**
2	89.9	92.2	**91.5**	84.4	62.6	**68.5**

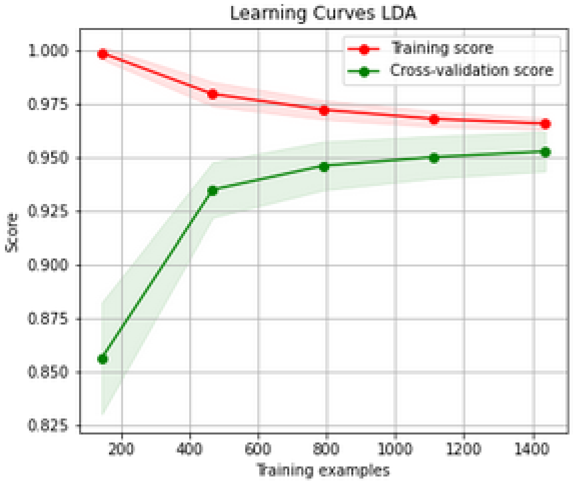

Figure 11: Learning Curves (LDA)

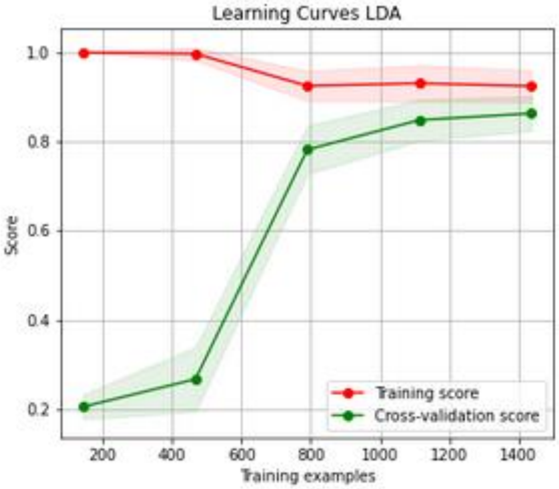

Figure 12: Learning Curves (QDA)

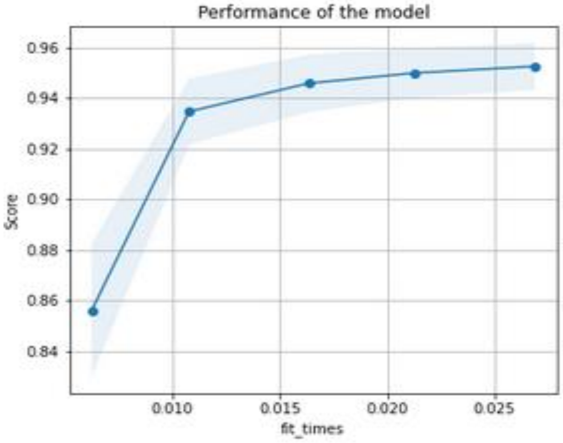

Figure 13: Performance of the Model (LDA)

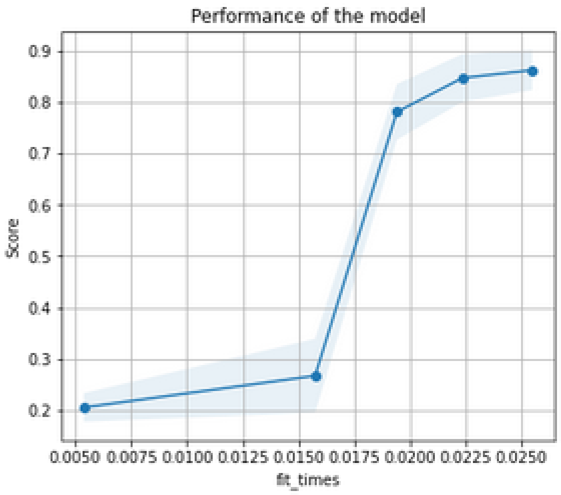

Figure 14: Performance of the Model (QDA)

5. Proposed Ensemble Model

We have compared several classification algorithms as mentioned in the previous section. However, we find that the performance of a single classifier can be enhanced. Thus, we propose a stacking algorithm that ensembles multiple classifiers to get more accuracy; Figure 15 illustrates the flow of the stacking model. After the previous experiments, we selected the RF, KNN, DT, LDA and BNB as a base classifier (level 0), while using Logistic Regression (LR) as meta-model (level 1). We feed the test results of all the basic classifiers to logistic regression to find the best ensemble of the set of classifiers.

The meta-model trained according to the prediction made by the base models to the data outside the sample. That is, data that is not included in training. The basic model is fed to the basic model for prediction, and these predictions, together with the expected outputs, give input and output pairs of the training dataset used to fit meta-model. The results are summarized in Table 7. After this experiment, we conclude that the proposed model gives the higher accuracy over the two datasets

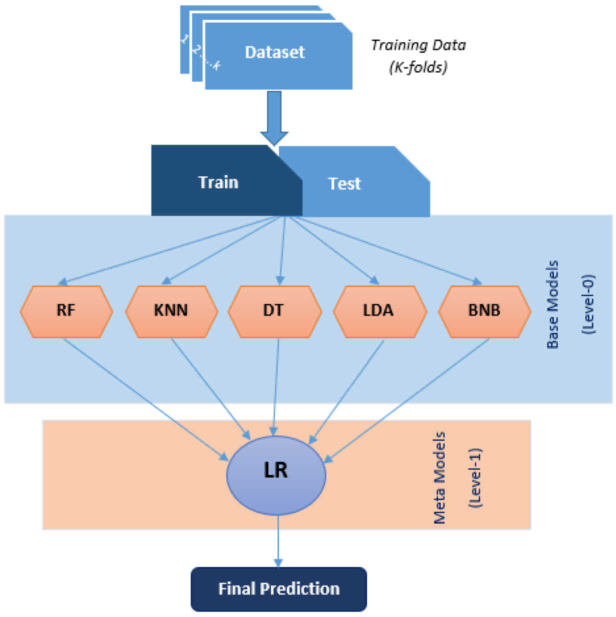

Figure 15: Proposed Model Architecture

The results are summarized in table 7. After this experiment we conclude that the proposed model gives the best accuracy over the two datasets.

Table 7: Ensemble stacking model

Dataset	**Ensemble Proposed Model**		
	Precision	Recall	F1-Measure
1	97.52	97.48	**97.49**
2	98.59	98.8	**98.69**

6. Results Summary and Discussion

In this paper, we have applied several machine learning classifiers as a single classifier used for classification, such as RF, DT, LDA, BNB, KNN, GNB, and QDA. The experimental results of these algorithms illustrate that the RF classifiers give a better accuracy as a single algorithm using the two datasets, which is 95.5% for dataset1 and 96.9% for dataset 2.

After our experiments, we conclude that the ensemble stacking model for classification to detect phishing websites based on the uniform resource locator (URL) properties gives better accuracy for the two datasets, 97.49% for dataset 1 and 98.69% for dataset 2. The results are summarized and compared in figures 16,17 for datasets 1 and 2. Moreover, by comparing our results of using dataset 1 with the previously mentioned proposed optimized ensemble model by Al-Sarem et al. [13] which has an accuracy 97.39 %, we found no significant enhancement using GA for parameters optimization. Our accuracy is 97.49%, which means that the enhancement of the ensemble model is based on the selected algorithms that are used as a base model and the number of features in the datasets.

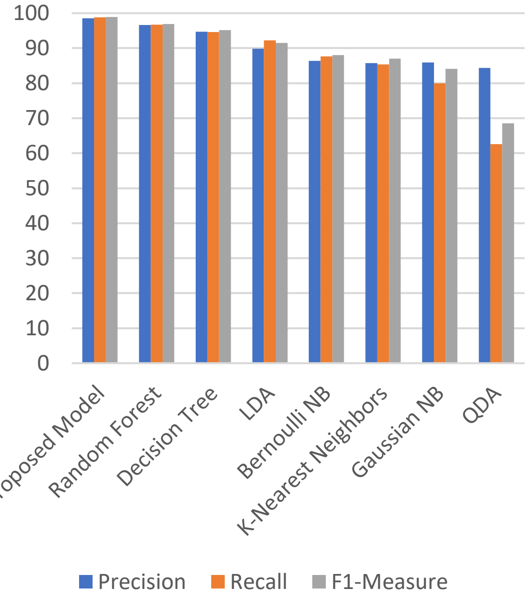

Figure 16: Result Summary (dataset 1)

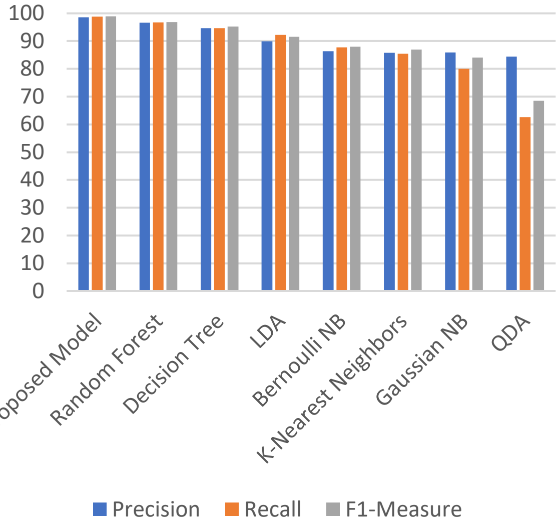

Figure 17: Result Summary (dataset 2)

7. Conclusion and Future Work

This paper aims to enhance a detection method to detect phishing websites using an ensemble learning based on the URL properties, URL resolving metrics, and external services using two preprocessed datasets compared to the well-known machine learning algorithms. Our ensemble model gives accuracy 97.49% for dataset 1 and 98.69% for dataset 2, which provides higher accuracy than using a single machine learning classification algorithm such as NB, DT, RF, KNN, LDA and QDA.

Future work can be conducted to classify if the website is legitimate or leads to a phishing attack. This is demonstrated using the dimensionality reduction of the feature and feature selection to enhance the accuracy and determine the best features to be used by the classification process. Also, the appearance of large datasets allows using Deep Learning (DL) for better classification of such attacks.

8. References

1. Hulten, G. J., Rehfuss, P. S., Rounthwaite, R., Goodman, J. T., Seshadrinathan, G., Penta, A. P., Mishra, M., Deyo, R. C., Haber, E. J. (2014) “Finding phishing sites”. US Patent 8, 839, 418
2. Basit, A., Zafar, M., Liu, X. 2021, “A comprehensive survey of AI-enabled phishing attacks detection techniques”, Tele- communication Systems,76, 39–154
3. James, J., Sandhya, L., & Thomas, C. 2013. “Detection of phishing URLs using machine learning techniques”, In 2013 International conference on control communication and computing (ICCC) (pp. 304–309). IEEE.
4. Subasi, A., Molah, E., Almkallawi, F., & Chaudhery, T. J. 2017, “Intelligent phishing website detection using random forest classifier”. In 2017 International conference on electrical and computing technologies and applications (ICECTA) (pp. 1–5). IEEE.
5. Mao, J., Bian, J., Tian, W., Zhu, S., Wei, T., Li, A., et al. 2019. “Phishing page detection via learning classifiers from page layout feature”, EURASIP Journal on Wireless Communications and Networking, 1, 43
6. Joshi, A., Pattanshetti, P., & Tanuja, R. 2019, “Phishing attack detection using feature selection techniques”, In International conference on communication and information processing (ICCIP), Nutan College of Engineering and Research., 39
7. Adebowale, M. A., Lwin, K. T., Sanchez, E., & Hossain, M. A. 2019, “Intelligent web-phishing detection and protection scheme using integrated features of images, frames and text” Expert Systems with Applications, 115, 300–313.

8.Alsariera, Y. A., Adeyemo, V. E., Balogun, A. O., & Alazzawi, A. K. 2020, “AI meta-learners and extra-trees algorithm for the detection of phishing websites”, IEEE Access, 8, 142532–142542.

9.Sahingoz, O. K., Buber, E., Demir, O., & Diri, B. 2019. “Machine learning based phishing detection from URLs”, Expert Systems with Applications, 117, 345–357.

10. Abdelhamid, N., Thabtah, F., Abdel-jaber, H. 2017, “Phishing detection: A recent intelligent machine learning comparison based on models’ content and features”, In 2017 IEEE international conference on intelligence and security informatics (ISI) (pp. 72–77). IEEE.
11. Hutchinson, S., Zhang, Z., & Liu, Q. 2018, “Detecting phishing websites with random forest”, In International conference on machine learning and intelligent communications (pp. 470–479). Springer.
12. Tyagi, I., Shad, J., Sharma, S., Gaur, S.,&Kaur, G. 2018. “A novel machine learning approach to detect phishing websites”, In 2018 5th International conference on signal processing and integrated networks (SPIN) (pp. 425–430). IEEE.
13. Al-Sarem, M.; Saeed, F.; Al-Mekhlafi, Z.G.; Mohammed, B.A.; Al-Hadhrami, T.; Alshammari, M.T.; Alreshidi, A.; Alshammari, T.S. 2021, “An Optimized Stacking Ensemble Model for Phishing Websites Detection”. Electronics, 10, 1285
14. Grega V., Iztok F., Vili P. 2020, “Datasets for phishing websites detection”, Data in Brief, Volume 3.
15. Feng Xin, Hao Xubing, Shi Ruoyao, Xia Zhiqiang, Huang Lan, Yu Qiong, Zhou Fengfeng 2020, “Detection and Comparative Analysis of Methylomic Biomarkers of Rheumatoid Arthritis”, Frontiers in Genetics , 11 , 238
16. Korotcov A, Tkachenko V, Russo DP, Ekins S. 2017, “Comparison of Deep Learning With Multiple Machine Learning Methods and Metrics Using Diverse Drug Discovery Datasets”, Molecular Pharmaceutics, 14(12):4462-4475
17. Almatarneh S., Gamallo P., Pena F.J.R., Alexeev A. 2019, “Supervised Classifiers to Identify Hate Speech on English and Spanish Tweets”, In: Jatowt A., Maeda A.,

Syn S. (eds) Digital Libraries at the Crossroads of Digital Information for the Future. ICADL 2019. Lecture Notes in Computer Science, vol 11853. Springer.

18. Meteier Q., Capallera M., Ruffieux S., Angelini L., Abou Khaled O., Mugellini E., Widmer M., Sonderegger A. 2021, "Classification of Drivers' Workload Using Physiological Signals in Conditional Automation", Frontiers in Psychology ,12 ,268

19. López H., Jesús L., González C., Lópe C., José L., Ruiz B., 2021, "Framework for the Classification of Emotions in People with Visual Disabilities Through Brain Signals: Frontiers in Neuroinformatics, 15, 12

20. Hasan K., Islam S., Rashid Khan M., Chakrabarty A., 2018, "A Machine Learning Approach on Classifying Orthopedic Patients Based on Their Biomechanical Features", 2018 Joint 7th International Conference on Informatics, Electronics & Vision (ICIEV) and 2018 2nd International Conference on Imaging, Vision & Pattern Recognition (ICIVPR).

DOI: http://Doi.org/10.54623/fue.fcij.7.1.1