Received March 15, 2021, decisioned March 28, 2021, revised April 2, 2021, accepted April, 7, 2021, date of publication April 10, 2021

Digital Object Identifier 10.46470/03d8ffbd.993cf635

Comparative Analysis of Feature Selection Techniques for Malicious Website Detection in SMOTE Balanced Data

NAMAN BHOJ¹, ASHUTOSH TRIPATHI², GRANTH SINGH BISHT³, ADARSH RAJ DWIVEDI⁴,
BISHWAJEET PANDEY⁵ and NITIN CHHIMWAL⁶

^{1,2,3,4}Plastic Associated Waste Information and Treatment for Research and Analysis (PAWITRAA), Haldwani, Uttarakhand, India.

^{1,5,6}Department of Computer Science and Engineering, Birla Institute of Applied Sciences (BIAS), Bhimtal, Uttarakhand, India.

e-mail: namanbhoj99@gmail.com¹, at88685@gmail.com², granthbisht@gmail.com³, rajadarsh082@gmail.com⁴, bishwajeet.pandey@gssi.it⁵
nitinchhimwal1978@gmail.com⁶

Corresponding author: Naman Bhoj (e-mail: namanbhoj99@gmail.com).

ABSTRACT The advancement in network technology has led to an exponential rise in the number of internet users across the globe. The increase in internet usage has resulted in an increase in both the number of malicious websites and cybercrimes reported over the years. Therefore, it has become critical to devise an intelligent solution that can detect malicious websites and be used in real-time systems. In our paper, we perform a comparative analysis of various feature selection techniques to build a time-efficient and accurate predictive model. To build our predictive model, a set of features are selected by feature selection methods. The selected features consist of at least 70% of the categorical features in all feature selection techniques examined in this paper. Keeping the end goal of real-time deployment of models in context the cost of processing or storing these features is far cheaper when compared to text or image-based features. We started out with a class imbalance in our data which was later dealt with using Synthetic Minority Oversampling Technique. Our proposed model also bested the existing work in the literature when compared over various evaluation metrics. The result indicated that Embedded feature selection was the best technique considering the accuracy of the model. The Filter-based technique may also be used in the context of developing a low latency system at the cost of the accuracy of the model.

INDEX TERMS Malicious Domain, Synthetic Minority Oversampling Technique (SMOTE), Balanced Dataset Classification, Feature Selection, Machine Learning, Artificial Intelligence, Internet Security.

I. INTRODUCTION

With the recent advancement in wireless network technology and cellular networks, there has been an exponential rise in internet usage. Statistics suggest, 62% of the world's population is using the internet on a daily basis with an average internet speed of 24.8 Mbps [1,2]. The number is only growing rapidly with almost 875,000 new users using the internet daily [3]. This has led to the transition of everything to a data-driven market which has consequently made accessibility to various e-commerce, edtech, or fintech resources possible on the internet. The primary interconnection medium between the customers and business entities providing these services or products is

a website. A website can simply be understood as a collection of files hosted on the server which are accessible to the end-user with custom functionality that the host wants to provide. Each website possesses a unique identification which is called a domain name that is used by the end-user to access the website. Currently, there are almost 1.82 billion websites on the internet [4].

The increase in internet usage has made accessibility of anything anywhere a reality but unfortunately, it also poses certain problems. As the amount of data generated and consumed is rapidly increasing as a result of these advancements. It has become critical to ensure the safety and privacy of the end-user over the internet from

cybercriminals. As cybercriminals use various techniques to deceive the end-user to extract their private information such as bank details, credit card details, passwords which compromise the safety of the internet users and leads to financial loss or identity theft. The majority of the successful scams executed over the internet are performed using websites as an interaction medium with the user. Statistics report there have been more than 60,000 phishing scams [5]. From the financial year of 2017 to 2020, the number of cases of fraudulent usage of debit/credit cards was 140,471 in India itself [6]. Data released by the Federal Trade Commission reports that the number of identity theft in 2020 was a whopping 1,387,615 [7]. The primary reason for these attacks being successful is the lack of technological literacy among the majority of the population using the internet on a daily basis [8]. During the third quarter of 2020, 19.2% and 13.4% of all attacks constituted the financial institution and payment sector [9].

Pertaining to this humongous volume of internet scams that leverage websites as a medium. In our research, we aim to build a robust malicious website detection model to ensure the safety of end-users over the internet. We employ various machine learning techniques to build our model. On top of that, we examine various feature selection techniques to identify strong indicative features of malicious websites to make the model fast and efficient for real-time deployment. The rest of the paper is structured as follows: Section 2 describes the existing research conducted in the domain, Section 3 delves into the methodology used in this paper followed by Section 4 which analyses and discusses results. Finally, the research conducted in this paper is summarized in Section 5.

II. LITERATURE REVIEW

In this section, we go through the past research conducted in the domain of detection of malicious websites. We discuss the formal technique, algorithms and evaluation measures achieved by various researchers.

In their research, A. Altaher [10] proposed a combination approach where they combined two algorithms namely: K-NN and SVM respectively for detecting phishing websites. In the initial stage, a hybrid model of KNN and SVM is employed. During the second stage, they employ SVM to classify the input data into three classes namely: Legitimate, Suspicious, Phishing. Using this approach in hybrid models of both algorithms achieved the highest accuracy value of 90.04%.

K. Al Messabi et al. [11] worked on a model for the detection of websites having malicious content. They employed 8 unique features which consisted of domain name-based features, character indicator features, top-level domain-based features and tokens to achieve better accuracy. J48 Decision Tree classifier was employed using 10 cross folds to validate the results. Dataset was partitioned into 10 folds, 9 were used for training and 1 for testing. Using this approach, they achieved the highest F1-Score value of 0.775.

R. Kumar et al. [12] proposed a multilayered model, in which they use 4 algorithms as a filter to determine the malicious URLs. In the 1st layer, they used a black and white list filter, in which a normal URL is assigned to the white list and a malicious URL is assigned with the blacklist. In the second and third layer, Naive Bayesian classifier and CART Decision Tree classifier are used respectively. The 2nd and 3rd layer are used to train the model and set the threshold. Final filter consisted of SVM for final classification on the combined filtered URLs in the above layers. This 4th layered model has achieved an accuracy of 79.55%.

D. Liu et al. [13] proposed a different method for malicious website detection. They used CNN (Convolution Neural Networks) for image recognition. In this method, they used webpage Screenshots of various websites to train the model. Consequently, the use of image data makes this method time inefficient. This method achieved an F1- score of 0.9503.

In their paper F.C. Dalgic et al. [14] proposes visual comparison for phishing website detection. They approached phishing detection as an image classification problem. Initially, MPEG7 was employed to find differences between screenshots, and then image classification is performed using RF and SVM. They achieved an F1-score value of 0.905.

D.N. Goswami [31] employs two approaches: Where they assign weights to all features first and then use the output to classify the URLs as malicious or not. Using this approach, they achieve an accuracy score of 74.4%.

Whereas, in their paper Jiajing Wu et. al. [32] employs network embedding to detect phishing nodes. Feature embedding is done using trans2vec networking and finally classification is performed using SVM which gives a F1-Score of 90.8%.

Considering the research conducted in the past. In our research we aim to improve upon the existing research done in the literature. Some of the key points investigated and improved in our research are:

- Investigating the time taken by various Feature Selection techniques. Computation time is of essence when the system is brought into real-time deployment to build low latency systems.
- Dataset class balancing is performed to reduce the bias while training and testing the data.
- Experiments are performed to find the Feature Selection technique which can best extract the most suitable feature for the problem.

The following experimentation has helped us to improve the existing research in terms of evaluation metrics which is discussed comprehensively in the next section.

III. RESEARCH METHODOLOGY

In this section we aim to provide insights to the methodology used in this paper.

Figure 1 shows the schematic workflow of conducted research which is primarily distributed in 8 main stages. The acronyms used in the paper are described in the Table 1 for further references.

FIGURE 1. Research Methodology Workflow

TABLE 1. ACRONYMS

Acronym	Description
KNN	K-Nearest Neighbour
NB	Naive Bayes
DT	Decision Tree
SMOTE	Synthetic Minority Oversampling Technique
FFS-DT	Filter Feature Selection - Decision Tree
WFS-KNN	Wrapper Feature Selection- K-Nearest Neighbour
EFS-DT	Embedded Feature Selection-Decision Tree

A. DATASET

The research conducted in this paper used the malicious and benign website dataset for building and testing the results of the proposed model [15]. The dataset initially consisted of 21 columns out of which 1 column was the dependent variable (malicious or benign) and the other 20 columns were a set of independent variables used to predict the dependent variable. The independent variable consisted of 7 categorical features and 13 numeric features. The detailed analysis of data is described in the next subsection.

B. DATA ANALYSIS

In this subsection, we discuss the data analysis task performed for the data used in our research. We analyzed our dataset for null values and found that the feature column “server” consisted of 1 null value, the feature column “content-length” consisted of 812 null values, and the feature column “DNS-query times” consisted of 1 null value.

Out of the 3 features that consisted of null value “content-length” and “DNS-query times” were numeric features and the “server” feature was a categorical feature. To deal with missing values in our numeric feature we filled the empty or NaN values with the median value of the distribution for that particular column. Whereas for the categorical feature missing value, we filled it with the category occurring the most in that feature column. We then dropped three features from our dataset namely: “url”, “whois_regdate”, “whois_updated_date” as in most of the cases user is rather redirected to a malicious website from some external source, and hence the “url” feature is not something the user knows and the other two dropped features were generally data stored by domain registrar which the user doesn’t have any information about. After this, we performed feature extraction in 4 of our categorical feature columns and applied One Hot Encoding.

This expanded our feature space from 17 features to 78 features. This huge feature space posed a problem of curse of dimensionality which could negatively affect our model and pose a problem for it to generalize well on the data [16]. Therefore, it was really necessary to perform feature selection to eliminate irrelevant features, reduce noise, improve data quality and increase predictive accuracy which consequently improves the performance of the model [17].

C. FEATURE SELECTION

In this subsection, we discuss the 3 feature selection techniques used to build our model namely: Filter, Wrapper, and Embedded feature selection. We also analyze the feature selection techniques based upon the number of features selected and time taken for selecting those features.

1) FILTER BASED FEATURE SELECTION

This method analyses the intrinsic properties of features measured and is fast and computationally inexpensive. We used the Chi-Square test [18]. It computes a score between each independent variable and dependent variable and then it selects the desired number of independent variables based upon the Chi-Square scores of each independent variable. For implementing the algorithm, we used SelectKBest Model and chi2 API available in Scikit-Learn [19,20]. This resulted in the selection of 30 independent variables (features) for our dataset.

2) WRAPPER BASED FEATURE SELECTION

This feature selection technique requires a specific algorithm to search the space of all possible feature subsets and assess their quality. The machine learning algorithm is

fitted to the given dataset. It follows a greedy approach for evaluating all possible combinations of features against the evaluation metric. In our research, we use Sequential Feature Selector which employs Logistic Regression for feature selection [21,22]. As a result, it selects the 30 best features for our model.

3) EMBEDDED FEATURE SELECTION

The embedded technique of feature selection leverages both the advantages of the wrapper and filter-based methods. It does so by including the interaction of features and also at the same time maintaining computational cost. This method is more iterative as it extracts those features which contribute the most to the training in the particular iteration. We employed Lasso Regression for embedded feature selection using SelectFromModel API of Scikit-Learn [23,24]. This resulted in the selection of 27 best features from the whole set of features.

FIGURE 2. Time taken by feature selection methods

FIGURE 3. No. of selected features

From Figure 2 we can see that Filter based feature selection took the least amount of time to select potentially best subsets of features from our feature space. Whereas the Wrapper and Embedded methods were more time-consuming. Also, from Figure 3 it is evident that the Embedded feature selection technique selected the minimum number of features from feature space. Hence, if we look at feature selection from a time consumption perspective Filter based feature selection is the best algorithm as it consumes minimum time and will be greatly helpful for systems that use the model in a real-time deployment scenario. Whereas Embedded feature selection selects minimum number of features and hence can potentially create a computationally inexpensive system.

Meanwhile, it is also essential to note that in terms of reaching conclusive evidence that which feature selection technique was able to find the best set of features for our problem, we will need to look into the combination of feature selection technique and machine learning algorithm and then evaluate their performance. This is further discussed in the Results and Analysis section of the paper.

D. DEALING WITH CLASS IMBALANCE

For classification problems, it is really essential to have a balanced class dataset. The imbalance of classes in classification causes challenges in predictive modeling as most of the machine learning algorithms used for classification were designed based on the assumption of an equal number of instances of each class. Also, to generate correct inferences from a classification model it is essential to have a balance classification dataset.

Initial analysis of our dataset showed that the number of instances of malicious websites was 7.24 times lesser than that of benign websites which can be seen in Figure 4. This severe class imbalance would cause our model to be biased as our model will be trained on a severely greater number of instances of benign websites as the number of instances of malicious websites were lesser than that of benign websites.

FIGURE 4. % instances of classes in the imbalanced dataset.

To deal with this problem we used Synthetic Minority Oversampling Technique (SMOTE). Using it we generated more examples of the minority class to balance the classes by matching the number of instances of minority and majority class. Therefore, oversampling the minority class to balance the dataset unless both the classes have an equal number of instances [25]. After using the Synthetic Minority Oversampling Technique to balance our classes resulted in an increase of data size from the initial size as the minority classes were oversampled. It leads to both the classes of our data being balanced with 1556 instances of both classes which can be seen in Figure 5.

balanced Dataset

FIGURE 5. % instances of classes in the balanced dataset.

After balancing our dataset we scaled our data to a similar scale using Standard Scaler [26]. Balancing and scaling our dataset improved the predictive accuracy for generating unbiased inference from our model. We distributed our data into an 80-20 train-test split. The train data was used to train the model while the test data validated the model's performance.

E. ALGORITHMS

1) K-NEAREST NEIGHBOUR

K-Nearest Neighbors [27] is a supervised classification and regression technique. In our paper we use it for labeled classification. It works by classifying an example datapoint based upon other nearest data points where it uses euclidean distance for measuring the nearest datapoint from the example. Example data is assigned to a class in which the majority of nearest data points are assigned. K is an important parameter in the algorithm as it decides how many neighbors to consider to label the example data point. Generally, in order to reduce the noise and achieve better accuracy, a higher value of K is preferred because a low value of K may alter the outliers in the model.

It is simple to implement KNN and it is a lazy learning algorithm because it doesn't learn in training, instead at the

time of classification an action is performed on the dataset which consequently leads to higher computational cost. Also, in each case euclidean distance calculation for every datapoint in the dataset is not an optimal method.

2) NAIVE BAYES

Naive Bayes [28] is a classification algorithm, which basically works on the foundation of Bayes theorem. It makes an assumption that every feature in the feature set is independent of other features. It is largely used for problems with huge datasets. It can be used for both binary and multiclass classification. The first step is conversion of a dataset set into a frequency table followed by generation of likelihood probability. Finally, posterior probability is calculated using Bayes Formula shown in equation 1.

Assuming A is class and B is a feature.

$$P(A/B) = P(B/A) * P(A) / P(B) \quad (1)$$

$P(A/B)$ represents the probability of a class over a given feature B, $P(B/A)$ is the probability of feature for class A, $P(A)$ and $P(B)$ are the probability of the class and feature respectively. Result of prediction using this algorithm is given to the class with maximum value of posterior probability. NB algorithm is simple, speedy and easy to implement for various classification problems.

3) DECISION TREE

Decision Trees [29] are highly used algorithms. It is used in varying fields such as statistics, various sorting techniques and in machine learning. DT can be used for both classification and regression but in our problem, we use it for classification. In a DT each node has a nested condition which represents a tree-like structure of this technique. Each leaf node contains the class labels in which test data is about to be classified and every non-leaf node contains the feature which is able to classify the dataset into subnodes on the basis of given attributes to that node. Further subnodes divide the data points with respect to the attributes given to that subnode. In order to trace the path from root node to leaf node DT uses recursive splitting of nodes.

Gini impurity and entropy are calculated to calculate the cost of splitting which helps in optimal splitting of a node for a given feature. Value of entropy is in the range [0,1] and determines the boundary of classification. Gini impurity represents the probability of a misclassification, and its lower value leads to a good split of nodes. The node having least cost is splitted into two child nodes. A small variation can be seen in output pertaining to a small change in input due to the nature of DT being low biased and high variance.

We have used various feature selection techniques and machine learning techniques to build our model. In the next section we aim to analyze the results of the conducted research.

IV. RESULT AND ANALYSIS

In this section, we aim to analyze and discuss the result achieved by combining 3 different feature selection techniques with 3 different machine learning models on test data. This section is divided into 2 subsections. In the first subsection, we aim to analyze the result of each feature selection algorithm in combination with the machine learning algorithms followed by a comparative analysis of these techniques. In the second subsection, we draw a comparison between the past research and our proposed model.

A. COMPARATIVE ANALYSIS OF FEATURE SELECTION TECHNIQUES

TABLE 2. PERFORMANCE EVALUATION USING FILTER FEATURE SELECTION

Algorithm	Accuracy (%)	F1-Score
KNN	93.64	0.9363
NB	77.68	0.7669
DT	94.46	0.9446

Table 2 shows the performance of various machine learning algorithms when Filter based technique was used for feature selection. The result indicates that for Filter based feature selection technique, DT classifier performed the best with an accuracy of 94.46% whereas an f1-score of 0.9446. Also, from Figure 3 and Figure 2 above, we can see that the Filter based feature selection technique selected 30 features in a time duration of 0.0153 seconds.

TABLE 3. PERFORMANCE EVALUATION USING WRAPPER FEATURE SELECTION

Algorithm	Accuracy (%)	F1-Score
KNN	91.55	0.9155
NB	71.42	0.6913
DT	90.90	0.9089

Table 3 shows the performance of various machine learning algorithms using Wrapper based feature selection technique. It is evident from the results that for Wrapper based feature selection technique, K-NN classifier performed the best with an accuracy of 91.55% and an f1-score of 0.9155. Though this technique also selected 30 features. The feature selection time was 303.95 seconds which can be seen from Figure 3 and Figure 2 above.

Table 4 shows the performance of various machine learning algorithms using the Embedded feature selection technique. Results indicate that the Embedded feature selection technique, DT classifier performed the best with an accuracy of 97.91% and an f1-score of 0.9790. The technique also selected 30 features with a feature selection time of 0.1235 seconds which can be seen from Figure 3 and Figure 2. It can also be seen from Table 2, Table 3, and Table 4 that the DT classifier outperformed the other models for the same sets of features in two out of three cases.

TABLE 4. PERFORMANCE EVALUATION USING EMBEDDED FEATURE SELECTION

Algorithm	Accuracy (%)	F1-Score
KNN	95.98%	0.9597
NB	77.84	0.7650
DT	97.91	0.9790

TABLE 5. AGGREGATED RESULTS FOR COMPARATIVE ANALYSIS

Technique	FFS -DT	WFS-KNN	EFS-DT
Best Accuracy (%)	94.46	91.55	97.91
Best F1-Score	0.9446	0.9155	0.9790
# Features	30	30	27
Time Taken (sec)	0.0153	303.95	0.1235

Aggregating all the results in Table 5 for analysis it is evident from Table 5 that the Embedded feature selection technique in combination with the DT outperformed other techniques in terms of accuracy and f1-score. Using Embedded feature selection for creating a set of features for our DT classifier we were able to achieve an accuracy of 97.91% and an f1-score of 0.9790 in a balanced dataset. The Embedded feature selection technique was also able to achieve higher accuracy with lesser number of features when compared to other feature selection techniques and consequently is expected to be computationally less expensive. Therefore, it is evident that it was better able to model the underlying features which were useful for our model for predictive modeling.

Keeping real-time deployment of the system in context we can see from Table 5 that the Filter feature selection technique had minimal time for feature selection and was approximately 8 times lesser than Embedded feature selection. This could be an important criterion for real-time systems with a humongous amount of data as the latency of data transmission to users is a major concern but this would come at the cost of the accuracy of the system. In such scenarios where latency is a major concern and accuracy of prediction is not the strongest requirement for the task, Filter feature selection would be a better choice.

B. COMPARATIVE ANALYSIS OF PROPOSED WORK WITH EXISTING RESEARCH

In this section we perform comparative analysis of existing research with our proposed work. Table 6 aggregates the results for comparison.

TABLE 6. COMPARISON WITH EXISTING WORK

Existing Research	Evaluation Metric	Proposed Work
A. Altaher [10]	Accuracy = 90.04%	Accuracy = 97.91%
K. Al Messabi [11]	F1-Score = 0.775	F1-Score = 0.9790
R. Kumar [12]	Accuracy = 79.55%	Accuracy = 97.91%
D. Liu [13]	F1-Score = 0.9503	F1-Score = 0.9790
F.C. Dalgic [14]	F1-Score = 0.905	F1-Score = 0.9790
D.N. Goswami [31]	Accuracy = 74.4%	Accuracy = 97.91%
Jiajing Wu [32]	F1-Score = 0.908	F1-Score = 0.9790

It is evident from Table 6 that our proposed model outperforms the existing work in the literature in terms of evaluation metric. It is important to note that the research conducted in our work is done after balancing the imbalance dataset using SMOTE and hence making accuracy and f1-score reliable measures for evaluating the performance of the proposed model [30].

Therefore, the performance of our model in terms of accuracy and f1-measure has bested the recent work in the literature. Also, our work compares and selects efficient feature selection techniques which are essential for low latency and building accurate predictive systems for real-time deployment.

V. CONCLUSION

Our research draws a comparison between various feature selection techniques for malicious website detection. We propose Embedded feature selection - DT classifier which

uses 27 features. 70% of features were categorical features and the model achieved an accuracy of 97.91%. Huge number of categorical features particularly makes the set of features simple in nature unlike image or text features but at the same time makes the modeling of the relationship between independent and dependent features a complex problem. The results also indicate that Embedded feature selection is an efficient feature selection technique when we want a smaller feature set to achieve higher accuracy. Whereas, Filter feature selection has the least amount of feature selection time when compared to other techniques. This can be really important when our goal is to reduce the latency but it would negatively affect the accuracy of the system.

Considering real-time systems with huge amounts of data inflow every second it is important to keep in mind that both accuracy and latency of the system are essential as they represent the trust of the user on the prediction made by the system and user experience while using the system respectively. Pertaining to this considering the advancements in network technology and high processing power of computational systems we can confidently say that in the future accuracy of the model as a concern would outweigh the latency concerns, we have with our existing model.

Therefore, the performance of future models can be improved by introducing domain registration-based features which would allow domain registrars to mark up any new registration as suspicious. This would help search engines to keep an eye on such new registrations marked suspicious by domain registrars and hence even before the act of cybercrime is executed necessary action could be taken. To improve the accuracy of the model in future we can also experiment with various Boosting, Bagging, and Deep Learning techniques.

REFERENCES

- [1] "Internet Growth Statistics 1995 to 2019 - the Global Village Online," internetworldstats.com, Accessed on: March 3 2021, [Online], Available: <https://www.internetworldstats.com/>
- [2] A. Willige, "These places have the fastest (and slowest) internet speeds," World Economic Forum, October 5 2020, Accessed on: March 3 2021, [Online], Available: <https://www.weforum.org/agenda/2020/10/fastest-slowest-internet-speeds-countries-world/>
- [3] "Global Digital Overview," Data Reportal, 2021, Accessed on: March 4 2021, [Online], Available: <https://datareportal.com/global-digital-overview>
- [4] "Internet Live Stats," Accessed on: March 4 2021, [Online], Available: <https://www.internetlivestats.com/watch/websites/>
- [5] D. Meharchandani, "Staggering Phishing Statistics in 2020," Security Boulevard [Online], Available: <https://securityboulevard.com/2020/12/staggering-phishing-statistics-in-2020/>

- [6] S. Barik, "FY20 Saw Over 50,000 Cases Of Fraud Usage Of Debit, Credit Cards," Medianama, March 18 2020, Accessed on: March 3 2021, [Online], Available: <https://www.medianama.com/2020/03/223-fraud-cases-credit-debit-cards/#:~:text=In%20FY20%2C%20more%20than%2050%2C000,Dhotre%20revealed%20in%20Parliament%20today>
- [7] K. Skiba, "Pandemic proves to be Fertile Ground For Identity Thieves", AARP, Feb 5 2021, Accessed on: March 4 2021, [Online], Available: <https://www.aarp.org/money/scams-fraud/info-2021/fic-fraud-report-identity-theft-pandemic.html#:~:text=The%20figures%20released%20Thursday%20by,2019%3B%20and%2044%2C344%20in%202018>
- [8] D. Finkelhor, K. Walsh, L. Jones, K. Mitchell, A. Collier, "Youth Internet Safety Education: Aligning Programs With the Evidence Base," *Trauma, violence, and abuse*, pp 1-15, Apr 2020, Accessed on: March 3 2021, DOI: 10.1177/1524838020916257, [Online].
- [9] "Phishing activity trends report," Anti Phishing Working Group (APWG), 2020, [Online]. Available: <https://apwg.org/trendsreports/>
- [10] A. Altaher, "Phishing websites classification using hybrid SVM and KNN approach", *International Journal of Advanced Computer Science and Applications (IJACSA)*, vol.8, pp. 90-95, Jan. 2017. Accessed on: March, 3, 2021, DOI: 10.14569/IJACSA.2017.080611, [Online].
- [11] K. A. Messabi, M. Aldwairi, A. A. Yousif, A. Thoban, F. Belqasmi (2018), "Malware Detection using DNS Records and Domain Name Features," presented at 2nd International Conference on Future Networks and Distributed Systems [Online], Available: <https://dl.acm.org/doi/proceedings/10.1145/3231053>
- [12] R. Kumar, X. Zhang, H. R., A. Tariq, R. U. Khan (2017), "Malicious URL detection using multi-layer filtering model", presented at *International computer conference on wavelet active media technology and information processing (ICCWAMTIP)* [Online], Available: <https://ieeexplore.ieee.org/document/8301457>
- [13] D. Liu, J. Lee, W. Wang and Y. Wang (2019) "Malicious Websites Detection via CNN based Screenshot Recognition*", presented at *International Conference on Intelligent Computing and its Emerging Applications (ICEA)* [Online], Available: <https://ieeexplore.ieee.org/document/8858300/footnotes#footnotes>
- [14] F. C. Dalgic, A. S. Bozkir and M. Aydos (2018) "Phish-IRIS: A New Approach for Vision Based Brand Prediction of Phishing Web Pages via Compact Visual Descriptors," *2nd International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT)* [Online], Available: <https://ieeexplore.ieee.org/document/8567299>
- [15] C. Urcuqui, "Malicious and Benign Websites Dataset", Accessed on: March 3 2021, [Online], Available: <https://www.kaggle.com/xwolf12/malicious-and-benign-websites>
- [16] K. P. Murphy, "Machine Learning A Probabilistic Perspective," London, England: The MIT Press Cambridge, 2012.
- [17] S. Khalid, T. Khalil and S. Nasreen (2014) "A survey of feature selection and feature extraction techniques in machine learning," presented at *Science and Information Conference*, [Online], Available: <https://ieeexplore.ieee.org/document/6918213>
- [18] Plackett, Robin L. "Karl Pearson and the chi-squared test", *International Statistical Review/Revue Internationale de Statistique*, vol. 51, pp. 59-72, 1983. Accessed on: March, 7, 2021, DOI: 10.2307/1402731, [Online].
- [19] C. Urcuqui, "Malicious and Benign Websites Dataset", Accessed on: March 3 2021, [Online], Available: <https://www.kaggle.com/xwolf12/malicious-and-benign-websites>
- [20] Sklearn Feature Selection, *Chi2*, [Online], Available: https://scikit-learn.org/stable/modules/generated/sklearn.feature_selection.chi2.html#sklearn.feature_selection.chi2
- [21] Sklearn Feature Selection, *Sequential Feature selector*, [Online], Available: https://scikit-learn.org/stable/modules/generated/sklearn.feature_selection.SequentialFeatureSelector.html
- [22] Sklearn Feature Selection, *Sequential Feature selector*, [Online], Available: https://scikit-learn.org/stable/modules/generated/sklearn.feature_selection.SequentialFeatureSelector.html
- [23] J. Berkson, "Application of the Logistic Function to Bio-Assay," *Journal of the American Statistical Association*, Vol.39, pp. 357-365, Accessed on : March, 3, 2021, DOI: 10.1080/01621459.1944.10500699, [Online]
- [24] R. Tibshirani, "Regression Shrinkage and Selection Via the Lasso," *Journal of the Royal statistical society*, Vol.58, pp. 267-288, 1996, Accessed on : March, 3, 2021, DOI: 10.1111/j.2517-6161.1996.tb02080.x, [Online].
- [25] Sklearn Feature Selection, *Select From Model*, [Online], Available: https://scikit-learn.org/stable/modules/generated/sklearn.feature_selection.SelectFromModel.html
- [26] N.V. Chawla, K.W. Bowyer, L.O. Hall, W.P. Kegelmeyer, "SMOTE: Synthetic Minority Over-Sampling technique," *Journal of Artificial intelligence*, Vol.16, Jan 2002, Accessed on : March, 3, 2021, DOI: 10.1613/jair.953, [Online].
- [27] Sklearn Preprocessing, *Standard Scaler*, [Online], Available: <https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.StandardScaler.html>
- [28] T. Cover and P. Hart, "Nearest neighbor pattern classification," *IEEE Transactions on Information Theory*, vol. 13, no. 1, pp. 21-27, January 1967, Accessed on: March 4 2021, DOI: 10.1109/TIT.1967.1053964, [Online].

- [29] T. Bayes, “ An essay towards solving a problem in the doctrine of chances” *Philosophical transactions of the Royal Society of London*, Vol. 59, pp. 370-418, Jan 1763, Accessed on: March 4 2021, DOI: 10.1098/rstl.1763.0053, [Online].
- [30] Quinlan, J. Ross. “Simplifying decision trees”, *International Journal of man-machine studies*, vol. 27, pp. 221-234, September 1987. Accessed on: March, 7, 2021, DOI:10.1016/S0020-7373(87)80053-6, [Online].
- [31] Branco, Paula, Luis Torgo, and Rita Ribeiro “A survey of predictive modelling under imbalanced domains”, *ACM Computing Survey (CSUR)*, vol. 49, pp. 1-50, August 2016, . Accessed on: March, 5, 2021, DOI:10.1016/S0020-7373(87)80053-6, [Online].
- [32] D. N. Goswami, M. Shukla and A. Chaturvedi, "Phishing Detection Using Significant Feature Selection," 2020 IEEE 9th International Conference on Communication Systems and Network Technologies (CSNT), Gwalior, India, 2020, pp. 302-306, Accessed on March,22,2021. DOI: 10.1109/CSNT48778.2020.9115782.
- [33] Wu et al., "Who Are the Phishers? Phishing Scam Detection on Ethereum via Network Embedding", *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, pp. 1-11, 2020. Accessed on March 22, 2021 Available: 10.1109/tsmc.2020.3016821.

NAMAN BHOJ is currently a Senior Year undergraduate student at Birla Institute of Applied Sciences, Bhimtal. He is majoring in Computer Science and Engineering and his broad research interest is in problems related to Natural Language Processing, Computer Vision and Machine Learning. In the past he has worked on Machine Learning, Natural Language Processing, Machine Learning based enhanced Cyber-Security and Time-Series analysis related projects. 3 of his papers have been accepted for publication in IEEE CSNT 2021 and will be available latest by May 2021. Mr. Naman also co-founded Plastic Associated Waste Information and Treatment for Research and Analysis (PAWITRAA) in 2019 as a step to promote research based education in schools and higher institutions. This research project is a part of collaboration between Plastic Associated Waste Information and Treatment for Research and Analysis and Department of Computer Science and Engineering, Birla Institute of Applied Sciences.

ASHUTOSH TRIPATHI is pursuing Bachelor's degree in Computer Science and Engineering course. Currently he is in 3rd year of his undergraduate studies. Along with this he has been a volunteer and research intern at Plastic Associated Waste Information and Treatment for Research and Analysis (PAWITRAA) since 2019. He has volunteered for various educational seminars organised by the organisation. He was also part of the team of PAWITRAA which conducted free technical seminar classes for school students. The event witnessed participation of 82 school going students. One of his research papers as a research intern in PAWITRAA has also been accepted in IEEE conference and is currently in press for publication.

GRANTH SINGH BISHT has been a member of Plastic Associated Waste Information and Treatment for Research and Analysis (PAWITRAA) since its inception. He has been a part of various Computer Science and Environment related educational seminars and projects organised by the organisation.

ADARSH RAJ DWIVEDI is pursuing B.tech in Electronics and Communication Engineering. He is in the 3rd year of his studies. He is also a core volunteering member of Plastic Associated Waste Information and Treatment for Research and Analysis (PAWITRAA) since 2020. He has been a part of conducting various educational seminars in the organisation and this is his first research project in the organisation. His major interests include Microprocessors and the Internet of Things.

Dr Bishwajeet Pandey has completed his PhD in CSE from Gran Sasso Science Institute, L'Aquila, Italy under guidance of Prof Paolo Prinetto, Politecnico Di Torino (World Ranking 13 in Electrical Engineering). He has worked as an Asst. Professor in Department of Research at Chitkara University, Junior Research Fellow (JRF) at South Asian University and Lecturers in Indira Gandhi National Open University. He has completed Master of Technology (IIT Gwalior) in CSE with Specialization in VLSI, Master of Computer Application, R&D Project in CDAC-Noida. He has authored and co authored 137 papers with 1600+ Citation He has experience of teaching Innovation and Startup, Computer Network, Digital Logic, Logic Synthesis, System Verilog. His area of research interest is Green Computing, High Performance Computing, Cyber Physical System, Artificial Intelligence, Machine Learning, and Cyber Security. He is on the board of directors of many startup of his Students e.g. Gyancity Research Consultancy Pvt Ltd.

NITIN CHHIMWAL received his Masters of Information technology degree in Information Systems from Queensland University of technology, Brisbane Australia. Currently, he is pursuing his Ph.D. degree in Computer Science Engineering. He has published 2 research papers in reputed national & international journals.

His research interest include Databases, Datawarehouse & Mining, Big Data. He has over 17 years of teaching experience as an Assistant professor in Birla Institute of Applied Sciences, Bhimtal in Department of Computer Application and Computer Science Engineering.