Global Journal of Engineering Innovations & Interdisciplinary Research

Original Article

Global Journal of Engineering Innovations and Interdisciplinary Research

Machine learning Ensemble Methods for Detection of Phishing in Website

Ch Yamini[1], G Jamuna Rani[2], Vadlamudi Chandana Sri[3], P Nithin Reddy[3], R Durga[3] Mallikarjun

[1]Associate Professor, Department AI & DS Science, Vignan Institute of Technology and Science, Hyderabad, India
[2]Assitant Professor, Department of AI & DS Science, Vignan Institute of Technology and Science, Hyderabad, India
[2]UG Student, Department of AI&DS, Vignan Institute of Technology and Science, Hyderabad, India

Correspondence

Ch.Yamini

Associate Professor, Department of Artificial Intelligence and Data Science, Vignan Institute of Technology and Science, Hyderabad, India

- Received Date: 25 May 2025
- Accepted Date: 15 June 2025
- Publication Date: 27 June 2025

Keywords

Webpage phishing Detection, Data Set, Ensemble Machine Learning Algorithm, Deep Learning Models, Model Comparison.

Abstract

In this research article, we propose to use a learning method with combinations such as the competitive random forest algorithm and the cloud gradient boosting and algorithm to efficiently and accurately identify customers who follow phishing websites. Phishing is one of the biggest cybercrimes in today's digital world. The attackers attempt to Obtain victims' credentials, account information, and other sensitive information by impersonating existing and generally trusted individuals or organizations are visible and similar to phishing websites. On real websites. Online commerce has also grown to increase the number of phishing scams. Network Security is the most difficult task to achieve, and development. Automated systems are in place for phishing website detection. Need machine learning is one of the best solutions for this situation because it can provide the correct classification system as well as check the status of phishing strategies.

Introduction

Overview

The word phishing describes a network trapping people in a way fishing technique is proceeded, by using a deceptive URL in a website phishing is processed.

Anti-Phishing Working Group (APWG) KnowBe4, Brand Shield, Cofense.

These are few anti Phishing organizations, which use

AI-based solutions that are increasingly being used in cybersecurity to detect phishing attacks. This process involves AI systems learning to recognize and classify phishing attempts efficiently. While deep learning (DL) has grew its usage and attention due to its ability to automatically extract features, traditional machine learning (ML) methods have been shown to sometimes offer better accuracy and lower false positive rates.

In 2022, the FBI's Internet Crime Complaint Center (IC3) received 800,944 phishing reports, resulting in more than $10.3 billion in losses, for a total of $44.2 million.

In 2017, Google and Facebook lost $100 million to phishing attacks.

Ensemble Learning

Ensemble models in machine learning refer to methods of merging multiple models to design a super model with high prediction accuracy models and great applications.

The Core concept for the ensemble model is the integrated learning is that if more than one model merged together, the overall performance can be improved because errors in one model can be corrected by other models. Hybrid methods are widely used in classification, regression, and many other areas of machine learning.

Machine Learning

Our research indicates that ensemble machine learning approaches can sometimes defeat the deep learning concepts in terms of accuracy and false positive rates when tested on well-known phishing websites. Algorithms like- XG Boost, CNN, Random Forest Tree, SVM, these algorithms result the accuracy for the given data detecting high-phished websites.

This research paper centers its focus on the algorithms in machine learning that are Impactful and Successful in the detection of phishing URLs, aiming to show that they can provide credible and consistent results. The 2 types of Ml learnings i.e. supervised learning and the un-supervised learning from the history of the data and with the network and text analysis we can extract the data and distinguish the patterns hidden and a layout for the interlinked connections can be identified for every data set entity. Identifying the variable, which is the target variable, i.e. the output and training the model.

Exsisting system

TSeveral works are accomplished in the phishing detection sector, and a wide variety of

Citation: Yamini Ch, Jamuna Rani G, Vadlamudi CS, Reddy PN, Mallikarjun RD. Machine Learning Ensemble Methods for detection of phishing in website. GJEIIR. 2025;5(4):085.

research is achieved.

There are different fields of studies produced for different phishing techniques such as- Email Phishing, Spear Phishing, Whaling, Vishing, Smishing.

Sufficiency of machine learning ensemble model for webpage detection whether it is a phishing trap or not and stacked ensemble phishing detection for various sectors like NOVEL, VOTING, EMAIL, BANK.

Different ML and DL algorithms were included for detection, Example: Boosting, CNN, FNN, SVM, Decision Tree and KNN algorithms with Optimised Stacked ensemble learning based on the classifications of URL- Uniform Resource Locator, finding the fraudulent links and fake websites with the HTML- tags and elements.

Limitations of exsisting system

Research may not explore a topic in sufficient detail, leading to superficial analysis and a failure to fully address the complexities of the subject, which indicates that It has lower accuracy compared to competitive APE.Usinga list of blacklisted URLs to check the estimatedaccurate results may be too low and deviate from the original values compared to the model given values called as fitting. There are some classifications that are more sensitive than others and therefore will be more accurate if they are based on the information they have learned. low ranging values reduce accuracy and increase execution-time. There is no standard organization to publish a standard URL list. As part of ourdaily life, may come across many new URLs that are similar to the old URL and are not. It is difficult to distinguish between URLs. Many hybrid models, especially those based on traditional machine learning methods, tend to misclasssify legitimate websites as phishing attempts. Ensemble approaches often require significant computational resources due to the need to train multiple models and combine their predictionThis complexity can hinder real-time search capabilities, especially in high volume work environments such as email systems or web gateways. As phishing tactics evolve, patten combinations must be reused to maintain their effectiveness, potentially making them exploitable. The need for constant updating and retraining can limit the scalability of these systems in a dynamic environment. Hybrid models, especially those based on deep learning, can be vulnerable to attacks where attackers control devices to avoid detection. Most integration methods require an extensive and comprehensive architecture that can consume more time, and it may not capture the full impact of a phishing attack.

The expectation that the features of this book will be selected may limit the model from being modified and to accommodate new phishing tactics that do not fit previously defined patterns. This limitation requires the developing general standards that can detect phishing across multiple platforms

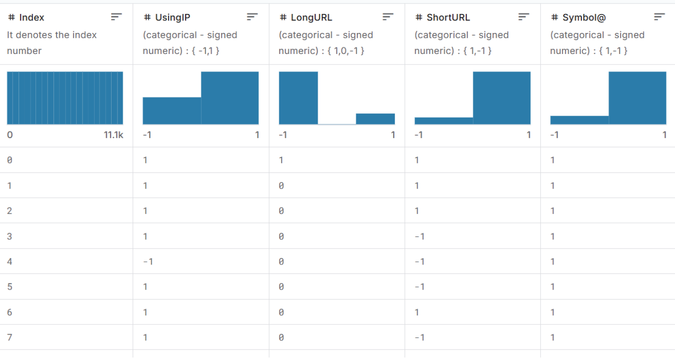
Proposed system

The proposed model is mainly modeled into 3 parts

1. Collection of Data set
2. Pre-Processing
3. Machine Learning – Ensemble Algorithms

Data collection

The collected data is from the Kaggle website, which consists of 32 columns and 1100+ rows. After the data is collected it is preprocessed where the missing values are removed, and the noisy data and the outliers are omitted from the data set. The collected data is cleaned, transformed and organized and turned the raw data into data used for building a model with training and testing and for analysis. The quality of the data is obtained, and it is prepared to give for a model. The legitimate data is collected from the trusted source Kaggle.



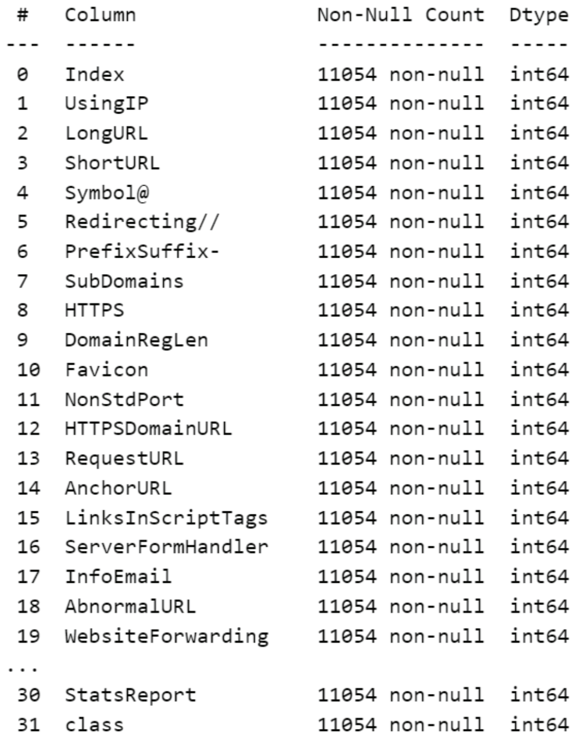
# Index	# UsingIP	# LongURL	# ShortURL	# Symbol@
It denotes the index number	(categorical - signed numeric) : { -1,1 }	(categorical - signed numeric) : { 1,0,-1 }	(categorical - signed numeric) : { 1,-1 }	(categorical - signed numeric) : { 1,-1 }
0	1	1	1	1
1	1	0	1	1
2	1	0	1	1
3	1	0	-1	1
4	-1	0	-1	1
5	1	0	-1	1
6	1	0	1	1
7	1	0	-1	1

Datasets

For instance, the United States, Belgium, and China have developed datasets of business signs. However, this paper discusses the German Traffic Sign Detection Benchmark, which has been extensively used in business sign discovery research. The GTSDB data-set is prized due to realistic business scenes captured along roadways, pastoral roads, and civic thoroughfares under varying conditions that encompass different rainfalls, lighting, and occlusions. The business signs are categorized into three important classes: prohibitory (59.5% in training, 161 in testing), obligatory (17.1% in training, 49 in testing), and peril (23.4% in training, 63 in testing). Another aspect that makes the dataset's popular is its public challenge, where scientists compare discovery models on a leaderboard showcasing state-of-the-art methods; however, processing times are not ranked. This standard has effectively reflected real-world business scenarios, making it a dependable tool for assessing business sign discovery models.

Selection of Features

Features used in phishing detection are typically extracted from the URL which is mainly of sectors such as



```
 #   Column              Non-Null Count  Dtype
---  ------              --------------  -----
 0   Index               11054 non-null  int64
 1   UsingIP             11054 non-null  int64
 2   LongURL             11054 non-null  int64
 3   ShortURL            11054 non-null  int64
 4   Symbol@             11054 non-null  int64
 5   Redirecting//       11054 non-null  int64
 6   PrefixSuffix-       11054 non-null  int64
 7   SubDomains          11054 non-null  int64
 8   HTTPS               11054 non-null  int64
 9   DomainRegLen        11054 non-null  int64
 10  Favicon             11054 non-null  int64
 11  NonStdPort          11054 non-null  int64
 12  HTTPSDomainURL      11054 non-null  int64
 13  RequestURL          11054 non-null  int64
 14  AnchorURL           11054 non-null  int64
 15  LinksInScriptTags   11054 non-null  int64
 16  ServerFormHandler   11054 non-null  int64
 17  InfoEmail           11054 non-null  int64
 18  AbnormalURL         11054 non-null  int64
 19  WebsiteForwarding   11054 non-null  int64
...
 30  StatsReport         11054 non-null  int64
 31  class               11054 non-null  int64
```

The system first provides a URL to detect phishing. Thes URLs can be filled by users who enter the generated web pages. After receiving the URL, the system extracts the relevant features from the web page. These resources are Essential for training and evaluation of machine learning models. Extract numerous type of URL from the database such as text, html and URL lines.

The Features required next after collecting the URL is the HTML And Domain features which are used to in this proposed system to detect the phishing, they are:

Domain Age
DNS Records
Web TrafficPage
Rank Google
Index Number of Links Pointing to the PageStatistics
Report Website
Redirects Status Bar
Customization Disable Right-Click
Use of Pop-UpsDomain
RecordsWeb TrafficPage

Pre-Processing Data

The data set collected is loaded in the .CSV format, and the data is fragmented into 2 parts TEST DATA and TRAIN DATA for evaluating the algorithm performance within.

The data is split with a ratio of 50:50 or 70:30 or 80:20.

The features selected face the overfitting and underfitting problems which require large data for better outcomes.

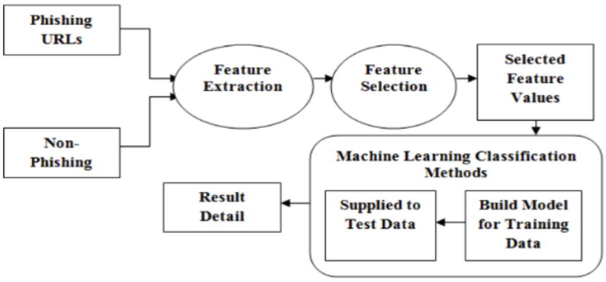
These features are processed in a machine understandable format and given to the machine learning classifier, and the features are conversant with EXPLORATORY DATA ANALYSIS tools where the data framework is used to access the data and create different frameworks of data.

The elite features which are from the data set are given to the algorithms like XGBOOST, KNN-K Nearest Neighbor, SVM-Support Vector Machine, Random Forest, Decision Tree, Navies Bayes, Multilayer Perceptron, Gradient Boosting, Logistic regression.

Arcitecture of System

As the data is collected preprocessed and the feature selections are finalized, the Next course of action is the design of the system where the architecture of the system is designed. First, make sure to upload the training materials.

The set seed function gives the appropriate values each epoch. Every time the model tree needs to learn from the dataset, it should be updated accordingly.



When the user enters a URL in the designated field, we need to verify it and let them know if it's a legitimate site or if it's a fake one.

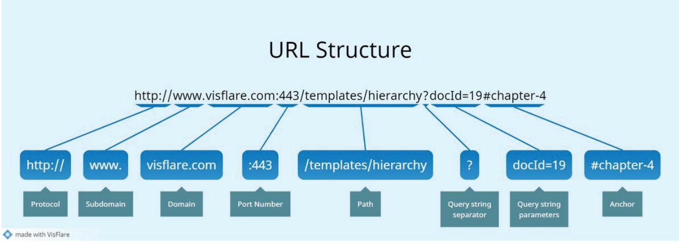
The data is visualised using data visualisation tools which shows the interrelation between one feature to other features

The data is visualised in many different visualisation techniques like Bar graphs, Pie Chats, Scatterplot etc.

Once the training is performed on the model using test data, it generates the truth and gives input to the user to validate the model's predictions against real results. The output integrated with the front-end development. The result is generated in the front-end page created. The output presented deploy the model accuracy in predicting whether the website is fake or not. The website is a genuine page, or a fraudulent page resulted in the output, The highest accuracy model is identified with the F1 Score and the best algorithm is used.

System Requirements

There are 2 data files taken Legitimate URL, Phishing URL. Where the legitimate URLs are GOOGLE, YOUTUBE, AMAZON, E MAIL, E BAY, MACFREE, SCAPO, AZUREFD where the trained model is trained with such legitimate data and predict that these URLs are legitimate and predict the testing data with high accuracy.



A URL is made up of several key components: The protocol (such as http-hypertext transfer protocol or https), the name of host (called as domain name), and the path (where the resource is located). For example, if you want to check Facebook, you will type in this URL: https://www.facebook.com/. The "https" part tells your browser how to connect to the site. Then, it sees "www.facebook.com", which is the domain name for Facebook. Finally, it requests a page on their server at the path '/', which takes you directly to their homepage.

ML PACKAGES:

Flask, Google-search, Python, NumPy, Pandas, Request, Scikit-lean, Urllib3, Whois, Gunicorn.

The Cascading Style Sheet document is required written in HTML for the index page, use-case, module setting, workspace and for templating the output.

Ensemble algorithms classification

The Machine Learning ensemble methods which is the Integration in machine learning is all about bringing together different models to boost the performance of predictions. The concept is straightforward, when you combine the predictions from several models, you can often achieve greater accuracy, strength, and detail than you would with just one model alone. It's a collaborative approach that leverages the strengths of multiple systems to enhance overall outcomes.

Gradient boosting algorithm

One of the effective and powerful algorithm is gradient boosting algorithm, this machine learning technique frequently employed in retrieval and classification tasks. What makes it unique is its sequential approach: each new model aims to correct the weaknesses of the previous one. This iterative process enhances its predictive accuracy, making it a vital tool for data professionals. This algorithm has the highest accuracy, with the accuracy rate 97.4%.

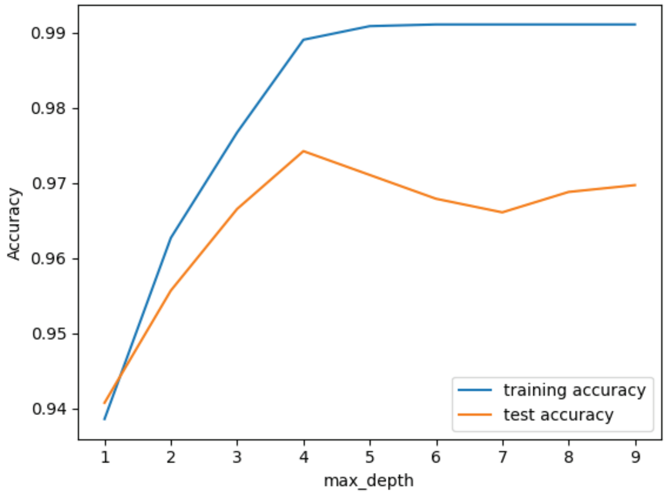
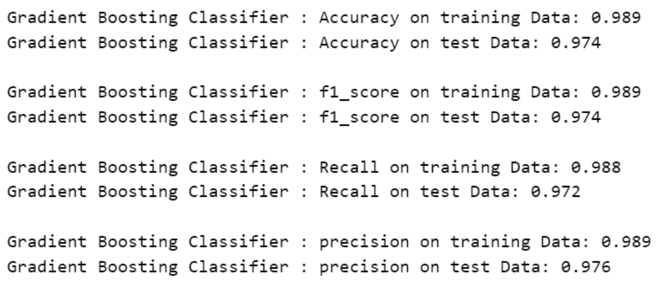


Figure 6. Accuracy of GB

The Gradient Boosting algorithm Accuracy is seen in FIGURE 6



```
Gradient Boosting Classifier : Accuracy on training Data: 0.989
Gradient Boosting Classifier : Accuracy on test Data: 0.974

Gradient Boosting Classifier : f1_score on training Data: 0.989
Gradient Boosting Classifier : f1_score on test Data: 0.974

Gradient Boosting Classifier : Recall on training Data: 0.988
Gradient Boosting Classifier : Recall on test Data: 0.972

Gradient Boosting Classifier : precision on training Data: 0.989
Gradient Boosting Classifier : precision on test Data: 0.976
```

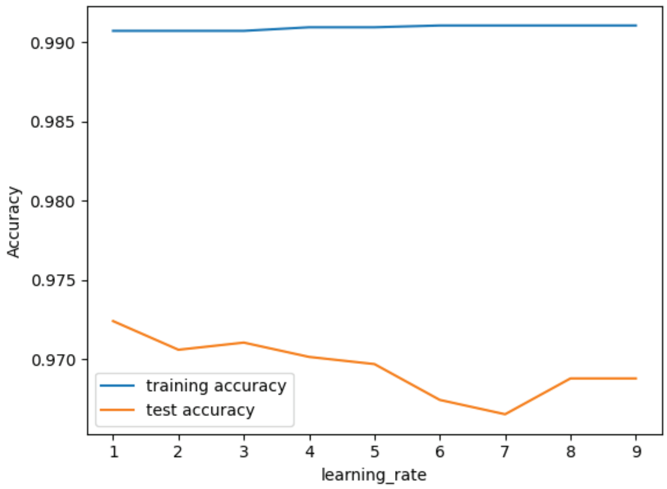
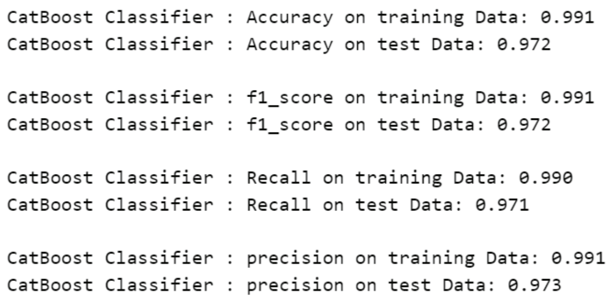


Figure 7. Accuracy of CB

The CatBoost algorithm Accuracy is seen in FIGURE 7



```
CatBoost Classifier : Accuracy on training Data: 0.991
CatBoost Classifier : Accuracy on test Data: 0.972

CatBoost Classifier : f1_score on training Data: 0.991
CatBoost Classifier : f1_score on test Data: 0.972

CatBoost Classifier : Recall on training Data: 0.990
CatBoost Classifier : Recall on test Data: 0.971

CatBoost Classifier : precision on training Data: 0.991
CatBoost Classifier : precision on test Data: 0.973
```

Catboost algorithm

The Cat Boost Classifier is an innovative machine learning algorithm developed by Yandex. One of its standout features is its seamless integration with well-known deep learning frameworks such as Google's TensorFlow and Apple's ML core. It is adaptable and capable of managing different types of data effectively. The second highest accuracy generated algorithm is cat boost with 97.2%

Model Comparision

In this proposed model there are 8 ensemble algorithms utilized, the model with highest accuracy is GRADIENTBOOSTING. Gradient boosting has some major benefits when it comes to identifying phishing websites. It has good prediction accuracy, can handle complex data structure, and helps reduce errors. Using the ensemble learning approach, it provides multiple weak classifiers to improve the overall performance and reliability of detecting phishing threats.

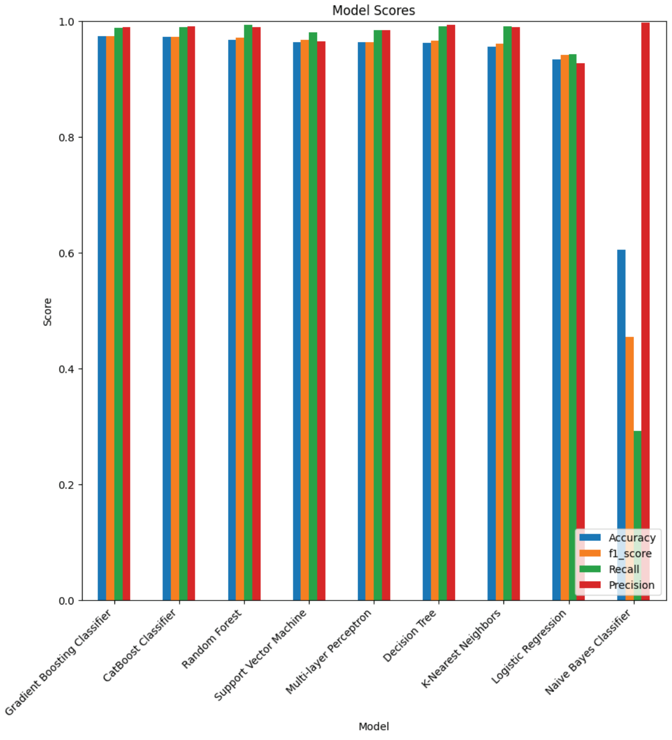


FIGURE 6. MODEL SCORE

High Prediction Accuracy:

Gradient boosting classifiers (GBCs) are really impressive when it comes to making accurate predictions. They do an excellent job of reducing errors by learning from the errors of previous models in the group.

Managing Complex Data Patterns:

One of the distinguishing features of GBC is its ability to pick out complex relationships in data. This is especially useful for spotting phishing websites, which often have subtle differences from legitimate websites that traditional methods might miss.

Reduce false positive rates:

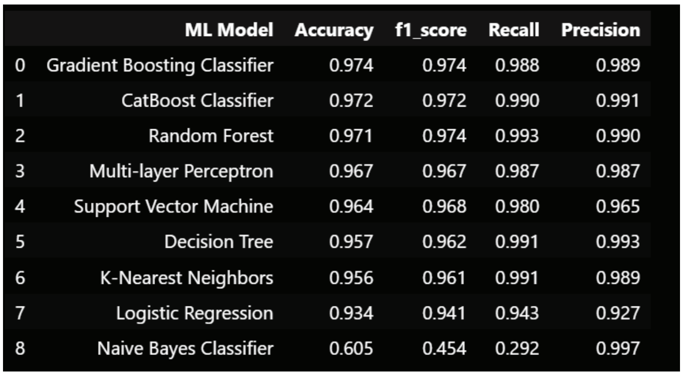
Thanks to their strengthening ability to combine several weak classifiers, gradient boosting can significantly reduce false positives. This is very important in detecting the phishing, where falsely Classifying a Legitimate platform as malicious can actually damage user trust.

Strongness against adversarial tactics:

GBCs are tough when it comes to dealing with the various tactics attackers use to hide phishing sites. Their ability to adapt and learn from complex traits makes them effective at identifying new and evolving threats.

Performance Metrics:

Studies show that GBC consistently outperforms traditional search engines in key performance metrics such as accuracy, precision, recall, and F1 scores.



	ML Model	Accuracy	f1_score	Recall	Precision
0	Gradient Boosting Classifier	0.974	0.974	0.988	0.989
1	CatBoost Classifier	0.972	0.972	0.990	0.991
2	Random Forest	0.971	0.974	0.993	0.990
3	Multi-layer Perceptron	0.967	0.967	0.987	0.987
4	Support Vector Machine	0.964	0.968	0.980	0.965
5	Decision Tree	0.957	0.962	0.991	0.993
6	K-Nearest Neighbors	0.956	0.961	0.991	0.989
7	Logistic Regression	0.934	0.941	0.943	0.927
8	Naive Bayes Classifier	0.605	0.454	0.292	0.997

Figure 8. *ACCURACY*

Confusion matrix

A confusion matrix is a simple tool for measuring how well a machine learning system is performing. It gives you a clear snapshot of how the model's predictions compare to the actual results. This is especially important for evaluating the model which is classification and its accuracy and figuring out what kinds of mistakes it might be making.

True Positive (TP): This is when we correctly identify something as positive.

True Negative (TN): This is when we accurately identify something as negative.

False Positive (FP): This occurs when we mistakenly label something as positive, even though it's not. This is referred as a Type I error.

False Negative (FN): This happens when we incorrectly say something is negative when it's positive, which is referred to as a Type II error.

Precision = TP
TP+FP
Recall = TP
TP+FN
F1 Score = 2*score obtained by precision * Score obtained by Recall / (Score of Precision + Score of Recall)

Accuracy Score = (positive true values + Negative true values)/ (positive true values + negative false values + Negative true values + Positive False values)

When we talk about performance analysis, we really dive into a detailed look at how the model is performing and go beyond just checking for accuracy.

Error analysis also plays a fundamental role here. It helps us identify specific mistakes the model is making – such as positive false values and negatives false values – which can guide us to improve and while the previous example focused on binary classification, confusion matrices can actually be modified for multi-class problems as well. This means we can compare each class to all the others, giving us a clearer picture of how the model performs in different categories.

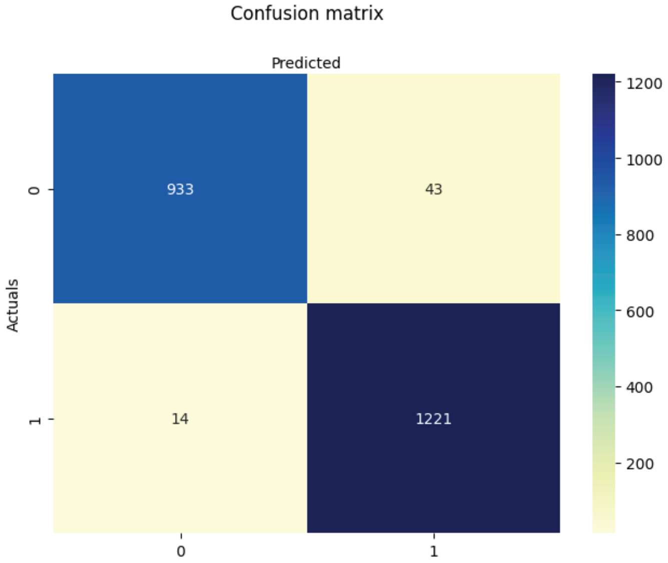


Figure 9. *Confusion matrix*

Conclusion

Here's what I got out of this project: We have successfully tested many machine learning models, explored phishing datasets, and see how they work. Creating this notebook was a great learning experience for me. We discovered a lot about what affects the model when determining whether a URL is safe. You also learned how to configure these models and see how these settings affect performance.

When we examined the phishing data set, we found that certain attributes such as "HTTPS", " URL Anchor " and "Traffic of website" play a key role in identifying whether a URL is a phishing attempt.

One of the highlights was the use of the Classifier Gradient Boosting (GB), which accurately classifies URLs with an impressive 97.4% accuracy. This is very useful to reduce the risk of encountering malicious links.

Phishing attacks are a big problem and fortunately there are ways to deal with them. Since these attacks can lead to sensitive personal data, it is important to find solutions. A good way is to use machines to learn algorithms that can classify These threats.

We already have a class that is good at predicting phishing attempts, but our research has shown that a combined approach can improve accuracy even more. When we saw that the current system was not working as we wanted, we created a new method that focuses on URLbased work.

Our conclusion provides valuable insights into how Machine learning can be used to combat phishing. Going forward, we plan to test our findings on more data with different features and examples. We also recognize that the challenge of detecting potentially difficult zero-day attacks need to be addressed. To address this issue, we aim to extract features from new phishing websites and continuously train our machine learning integration method. By monitoring emerging trends in phishing models, we hope to determine the best way to update features and train our models to ensure we can manage theexactly how to do our job every day

References

1. Badkul, Dharani., S., Gharat, K., Vidhate., & Bhosale, D. (2021). Detection of phishing websites using ensamble, machine,learning approach. In ITM Web of Conferences (Vol. 40, p. 03012). EDP Science.pp.22-25
2. Ubing Alyssa, Syukrina Jasmi Kamilia Binti, Azween Abdullah, N Jhanjhi, Supramaniam Mahadevan "Phishing Website Detection: An Improvised Accuracy through Feature Selections and Ensamble Learning" 2019.pp.252–255.
3. Wei, and Sekiya, Y., 2022. Sufficienct of ensamble machine learning methods for phishing webpage detection. IEEE Access, 10, pp.124103-124113
4. Khan Zamir, U, Iqbal T, Yousaf N, Aslam F, Anjum A, Hamdani M. Phishing web site detection using diverse machine learning algorithms. The Electronic Library. 2020 Mar 19;38(1):65-80
5. Mekhlafi Ghaleb, Z., Mohammed, Abdulkarem , B., Al-Sarem, M., Saeed, F., Hadhrami, T., Alshammari, M.T., Alreshidi, and Alshammari Sarheed, T., 2022. Phishing websites detection by using optimized stacking ensemble model. Computer Systems Science and Engineering, 41(1), pp.109-125.
6. Nisreen Innab,, Abdelgader Ahmed Osman, Mohammed Awad Mohammed Ataelfadiel, Marwan Abu-Zanona, Mohammad Elzaghmouri, Farah H. Zawaideh, and Mouiad Alawneh. "Phishing Attacks Detaction Using Ensamble Machine Learning Algorithms." Computers, Materials & Continua 80, no. 1 .pp445-456
7. Rundong Yang, Zheng Kangfeng, Wu Bin, Chunhua, and Xiujuan Wang. "Phishing website detection based on deep convolutional neural network and random forest ensemble learning." Sensors 21, no. 24 (2021): 8281.
8. Uppalapati, P., Gontla, B., Gundu, P., Hussain, S. & Narasimharo, K. (2023). A Machine Learning Approach to Identifying Phishing Websites: A Comparative Study of Classification Models and Ensemble Learning Techniques. EAI Endorsed Transactions on Scalable Information Systems, 10(5).
9. R. Bhallamudi et al., "Deep Learning Model for Resolution Enhancement of Biomedical Images for Biometrics," in Generative Artificial Intelligence for Biomedical and Smart Health Informatics, Wiley Online Library, pp. 321–341, 2025.
10. R. Bhallamudi et al., "Artificial Intelligence Probabilities Scheme for Disease Prevention Data Set Construction in Intelligent Smart Healthcare Scenario," SLAS Technology, vol. 29, pp. 2472–6303, 2024, Elsevier.
11. R. Bhallamudi, "Improved Selection Method for Evolutionary Artificial Neural Network Design," Pakistan Heart Journal, vol. 56, pp. 985–992, 2023.
12. R. Bhallamudi et al., "Time and Statistical Complexity of Proposed Evolutionary Algorithm in Artificial Neural Networks," Pakistan Heart Journal, vol. 56, pp. 1014–1019, 2023.
13. R. Krishna et al., "Smart Governance in Public Agencies Using Big Data," The International Journal of Analytical and Experimental Modal Analysis (IJAEMA), vol. 7, pp. 1082–1095, 2020.
14. N. M. Krishna, "Object Detection and Tracking Using YOLO," in 3rd International Conference on Inventive Research in Computing Applications (ICIRCA-2021), IEEE, Sept. 2021, ISBN: 978-0-7381-4627-0.
15. N. M. Krishna, "Deep Learning Convolutional Neural Network (CNN) with Gaussian Mixture Model for Predicting Pancreatic Cancer," Springer US, vol. 1380-7501, pp. 1–15, Feb. 2019.
16. N. M. Krishna, "Emotion Recognition Using Skew Gaussian Mixture Model for Brain–Computer Interaction," in SCDA-2018, Textbook Chapter, ISBN: 978-981-13-0514, pp. 297–305, Springer, 2018.
17. N. M. Krishna, "A Novel Approach for Effective Emotion Recognition Using Double Truncated Gaussian Mixture Model and EEG," I.J. Intelligent Systems and Applications, vol. 6, pp. 33–42, 2017.
18. N. M. Krishna, "Object Detection and Tracking Using YOLO," in 3rd International Conference on Inventive Research in Computing Applications (ICIRCA-2021), IEEE, Sept. 2021, ISBN: 978-0-7381-4627-0.
19. T. S. L. Prasad, K. B. Manikandan, and J. Vinoj, "Shielding NLP Systems: An In-depth Survey on Advanced AI Techniques for Adversarial Attack Detection in Cyber Security," in 2024 3rd International Conference on Automation, Computing and Renewable Systems (ICACRS), IEEE, 2024.
20. S. Sowjanya et al., "Bioacoustics Signal Authentication for E-Medical Records Using Blockchain," in 2024 International Conference on Knowledge Engineering and Communication Systems (ICKECS), vol. 1, IEEE, 2024.
21. N. V. N. Sowjanya, G. Swetha, and T. S. L. Prasad, "AI Based Improved Vehicle Detection and Classification in Patterns Using Deep Learning," in Disruptive Technologies in Computing and Communication Systems: Proceedings of the 1st International Conference on Disruptive Technologies in Computing and Communication Systems, CRC Press, 2024.
22. C. V. P. Krishna and T. S. L. Prasad, "Weapon Detection Using Deep Learning," Journal of Optoelectronics Laser, vol. 41, no. 7, pp. 557–567, 2022.
23. T. S. L. Prasad et al., "Deep Learning Based Crowd Counting Using Image and Video," 2024.