PHISHING DETECTION SYSTEM THROUGH HYBRID MACHINE LEARNING BASED ON URL

ABSTRACT

Currently, numerous types of cybercrime are organized through the internet. Hence, this study mainly focuses on phishing attacks. Although phishing was first used in 1996, it has become the most severe and dangerous cybercrime on the internet. Phishing utilizes email distortion as its underlying mechanism for tricky correspondences, followed by mock sites, to obtain the required data from people in question. Different studies have presented their work on the precaution, identification, and knowledge of phishing attacks; however, there is currently no complete and proper solution for frustrating them. Therefore, machine learning plays a vital role in defending against cybercrimes involving phishing attacks. The proposed study is based on the phishing URL-based dataset extracted from the famous dataset repository, which consists of phishing and legitimate URL attributes collected from 11000+ website datasets in vector form. After preprocessing, many machine learning algorithms have been applied and designed to prevent phishing URLs and provide protection to the user. This study uses machine learning models such as decision tree (DT), linear regression (LR), random forest (RF), naive Bayes (NB), gradient boosting classifier (GBM), K-neighbors classifier (KNN), support vector classifier (SVC), and proposed hybrid LSD model, which is a combination of logistic regression, support vector machine, and decision tree (LR+SVC+DT) with soft and hard voting, to defend against phishing attacks with high accuracy and efficiency. The canopy feature selection technique with cross fold valoidation and Grid Search Hyperparameter Optimization techniques are used with proposed LSD model. Furthermore, to evaluate the proposed approach, different evaluation parameters were adopted, such as the precision, accuracy, recall, F1-score, and specificity, to illustrate the effects and efficiency of the models. The results of the comparative analyses demonstrate that the proposed approach outperforms the other models and achieves the best results.

INTRODUCTION

The internet is a global network of computers that connect through various communication methods, such as phone lines, fiber optics, and satellites. It allows people to access information from hosts and servers using a protocol called IP-TCP. The internet is managed by various organizations, research groups, and universities, not by the government. It has become vital in many aspects of life, including education, banking, entertainment, social media, and online business.

People use the internet for fast communication (through services like email and social networks), e-commerce, and online meetings, especially during the COVID-19 pandemic. However, with the increase in online activities, there are also higher risks of cyberattacks, such as hacking, viruses, phishing, and fraud.

Phishing detection and cybersecurity are essential to protect users and their information. Over the years, many methods have been developed to detect and prevent phishing, focusing on analyzing the structure of URLs (Uniform Resource Locators). URLs are the addresses used to access websites and consist of several components, including the protocol (HTTP or HTTPS), domain name, and file path.

HTTP is used to request web resources, while HTTPS adds encryption for secure communication, protecting sensitive data like passwords and credit card information. The internet has become a tool for both convenience and potential risk, making cybersecurity and awareness critical for users.

EXISTING PROJECT:

Phishing identification systems use white lists and blacklists to categorize legitimate and phishing websites. Whitelist-based systems generate secure websites, while blacklist-based systems track IP addresses of end-user login interfaces. Blacklist systems, like Phish-Net and Google Safe Browsing API, have a high success rate of about 20%. Automatic phishing detection mechanisms, such as Delta Phish, Phish-Safe, SVM, naive Bayes, ensemble learning, Phi DMA, and typo squatting, have achieved high accuracy rates. Machine learning techniques like CANTINA, CANTINA+, and Phish- WHO have achieved 99% accuracy. A 2011 system identified phishing websites using attributes like directory, file name, domain name, special characters, and length, using Support Vector Machine (SVM) for offline classification.

DISADVANTAGES

- The complexity of data: Most of the existing machine learning models must be able to accurately interpret large and complex datasets to detect phishing urls.
- Data availability: Most machine learning models require large amounts of data to create accurate predictions. If data is unavailable in sufficient quantities, then model accuracy may suffer.
- Incorrect labeling: The existing machine learning models are only as accurate as the data trained using the input dataset. If the data has been incorrectly labeled, the model cannot make accurate predictions.

PROPOSED SYSTEM

- Phishing URL-based cyberattack detection is proposed in this study to prevent crime and protect people's privacy.
- The dataset consists of 11000+ phishing URL attributes that help classify phishing URLs based on these attributes.
- Machine learning models have been applied, such as decision tree (DT), linear regression (LR), naive Bayes (NB), random forest (RF), gradient boosting machine (GBM), support vector classifier (SVC), K-Neighbors classifier (KNN), and the proposed hybrid model (LR+SVC+DT) LSD with soft and hard voting, which can accurately classify the threats of phishing URLs.
- Cross-fold validation with a grid search parameter based on the canopy feature selection technique was used with the proposed LSD hybrid model to improve prediction results.
- The proposed methodology must be evaluated using evaluation parameters, such as accuracy, precision, recall, specificity, and F1-score.

H/W SYSTEM CONFIGURATION

Processor	Pentium –IV
RAM	4 GB (min)Hard Disk - 20 GB
Key Board	Standard Windows Keyboard
Mouse	Two or Three Button Mouse
Monitor	SVGA

SOFTWARE REQUIREMENTS:

Operating system.	Windows 7 Ultimate.
Coding Language.	Python.
Front-End	Python.
Back-End	Django-ORM
Designing	Html, css, javascript.
Data Base	MySQL (WAMP Server).

APPLIED MACHINE LEARNING ALGORITHMS

1. Decision Tree Classifier

- What it does: A decision tree is a model that makes decisions by asking a series of questions. Each question splits the data into smaller groups based on certain features, and the process repeats until each group only contains data points from a single class.
- Example: If you're deciding whether to bring an umbrella, the tree might first ask, "Is it cloudy?" If yes, it might ask, "Is the temperature below 50°F?" If yes, the tree might decide you should bring an umbrella.

2. Gradient Boosting

- What it does: This is a method where many "weak" models (usually decision trees) are combined to create a stronger model. Each new model tries to correct the mistakes of the previous one.
- Example: Imagine you are trying to predict home prices. If your first model makes some wrong predictions, a second model will focus on fixing those wrong predictions. Over time, you build a strong model by improving on previous mistakes.

3. K-Nearest Neighbors (KNN)

- What it does: KNN classifies a data point based on how similar it is to its nearest neighbors. You pick a value "K," and the algorithm looks at the K closest data points, then assigns the most common class among them.
- Example: If you want to know if a fruit is an apple or an orange, the algorithm checks which type the nearest fruits (neighbors) belong to and assigns the same type.

4. Logistic Regression

- What it does: Logistic regression is used to predict a binary outcome (like yes/no or 0/1). It tries to find the best line (or curve) that separates data points into two categories.
- Example: If you want to predict whether a student will pass or fail a test based on study hours, logistic regression finds the best way to separate the students who pass from those who fail.

5. Naïve Bayes

- What it does: This method assumes that features (like words in a document) are independent from each other. It calculates the probability of each class based on the given features and picks the one with the highest probability.
- Example: In spam email classification, the algorithm might calculate the probability that certain words (like "free" or "buy now") indicate that an email is spam.

6. Random Forest

- What it does: Random forest is an ensemble method that builds many decision trees and combines their predictions. Each tree is built using a random subset of features and data, which helps reduce overfitting.
- Example: If you're classifying emails as spam or not, a random forest will use many different decision trees to make the final decision based on the majority vote of all trees.

7. Support Vector Machine (SVM)What it does:

- SVM is used to find the best boundary (called a hyperplane) that separates different classes in the data. It tries to find the line or surface that maximizes the margin between classes.

SYSTEM STUDY

FEASIBILITY STUDY

The feasibility of the project is analyzed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major requirements for the system is essential.

Three key considerations involved in the feasibility analysis are:

- ♦ ECONOMICAL FEASIBILITY

- TECHNICAL FEASIBILITY
- SOCIAL FEASIBILITY

ECONOMICAL FEASIBILITY

This study is carried out to check the economic impact that the system will have on the organization. The amount of fund that the company can pour into the research and development of the system is limited. The expenditures must be justified. Thus the developed system as well within the budget and this was achieved because most of the technologies used are freely available. Only the customized products had to be purchased.

TECHNICAL FEASIBILITY

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. This will lead to high demands on the available technical resources. This will lead to high demands being placed on the client. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system.

SOCIAL FEASIBILITY

The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity. The level of acceptance by the users solely depends on the methods that are employed to educate the user about the system and to make him familiar with it. His level of confidence must be raised so that he is also able to make some constructive criticism, which is welcomed, as he is the final user of the system.

MODULES

Service Provider

In this module, the Service Provider has to login by using valid user name and password. After login successful he can do some operations such as Browse URL Data Sets and Train & Test, View Trained and Tested URL Data Sets Accuracy in Bar Chart, View Trained and Tested URL Data Sets Accuracy Results, View Prediction Of URL Type View URL Type Ratio, Download Predicted Data Sets, View URL Type Ratio Results, View All Remote Users.

View and Authorize Users

In this module, the admin can view the list of users who all registered. In this, the admin can view the user's details such as, user name, email, address and admin authorizes the users.

PRELIMINARY INVESTIGATION

The first and foremost strategy for development of a project starts from the thought of designing a mail enabled platform for a small firm in which it is easy and convenient of sending and receiving messages, there is a search engine ,address book and also including some entertaining games. When it is approved by the organization and our project guide the first activity, ie. preliminary investigation begins. The activity has three parts:

- **Request Clarification**
- **Feasibility Study**
- **Request Approval**

1. REQUEST CLARIFICATION

After the approval of the request to the organization and project guide, with an investigation being considered, the project request must be examined to determine precisely what the system requires.

Here our project is basically meant for users within the company whose systems can be interconnected by the Local Area Network(LAN). In today’s busy schedule man need everything should be provided in a readymade manner. So taking into consideration of the vastly use of the net in day to day life, the corresponding development of the portal came into existence.

2. FEASIBILITY ANALYSIS

An important outcome of preliminary investigation is the determination that the system request is feasible. This is possible only if it is feasible within limited resource and time. The different feasibilities that have to be analyzed a

- **Operational Feasibility**
- **Economic Feasibility**
- **Technical Feasibility**

Operational Feasibility

Operational Feasibility deals with the study of prospects of the system to be developed. This system operationally eliminates all the tensions of the Admin and helps him in effectively tracking the project progress. This kind of automation will surely reduce the time and energy, which previously consumed in manual work. Based on the study, the system is proved to be operationally feasible.

Economic Feasibility

Economic Feasibility or Cost-benefit is an assessment of the economic justification for a computer based project. As hardware was installed from the beginning & for lots of purposes thus the cost on project of hardware is low. Since the system is a network based, any number of employees connected to the LAN within that organization can use this tool from at anytime. The Virtual Private Network is to be developed using the existing resources of the organization. So the project is economically feasible.

Technical Feasibility

According to Roger S. Pressman, Technical Feasibility is the assessment of the technical resources of the organization. The organization needs IBM compatible machines with a graphical web browser connected to the Internet and Intranet. The system is developed for platform Independent environment. Java Server Pages, JavaScript, HTML, SQL server and WebLogic Server are used to develop the system. The technical feasibility has been carried out. The system is technically feasible for development and can be developed with the existing facility.

3.REQUEST APPROVAL

Not all request projects are desirable or feasible. Some organization receives so many project requests from client users that only few of them are pursued. However, those projects that are both feasible and desirable should be put into schedule. After a project request is approved, it cost, priority, completion time and personnel requirement is estimated and used to determine where to add it to any project list. Truly speaking, the approval of those above factors, development works can be launched.

DESIGN AND DEVELOPMENT SYSTEM

- **INPUT DESIGN**

Input Design plays a vital role in the life cycle of software development, it requires very careful attention of developers. The input design is to feed data to the application as accurate as possible. So inputs are supposed to be designed effectively so that the errors occurring while feeding are minimized. According to Software Engineering Concepts, the input forms or screens are designed to provide to have a validation control over the input limit, range and other related validations.

This system has input screens in almost all the modules. Error messages are developed to alert the user whenever he commits some mistakes and guides him in the right way so that invalid entries are not made. Let us see deeply about this under module design.Input design is the process of converting the user created input into a computer-based format. The goal of the input design is to make the data entry logical and free from errors. The error is in the input are controlled by the input design. The application has been developed in user-friendly manner. The forms have been designed in such a way during the processing the cursor is placed in the position where must be entered. The use Is also provided with in an option to select an appropriate input from various alternatives related to the field in certain cases.Validations are required for each data entered. Whenever a user enters an erroneous data, error message is displayed and the user can move on to the subsequent pages after completing all the entries in the current page.

- **OUTPUT DESIGN**

The Output from the computer is required to mainly create an efficient method of communication within the company primarily among the project leader and his team members, in other words, the administrator and the clients. The output of VPN is the system which allows the project leader to manage his clients in terms of creating new clients and assigning new projects to them, maintaining a record of the project validity and providing folder level access to each client on the user side depending on the projects allotted to him. After completion of a project, a new project may be assigned to the client. User authentication procedures are maintained at the initial stages itself. A new user may be created by the administrator himself or a user can himself register as a new user but the task of assigning projects and validating a new user rests with the administrator only.

The application starts running when it is executed for the first time. The server has to be started and then the internet explorer in used as the browser. The project will run on the local area network so the server machine will serve as the administrator while the other connected systems can act the clients. The developed system is highly user friendly and can be easily understood by anyone using it even for the first time.

SYSTEM TESTING

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub assemblies, assemblies and/or a finished product It is the process of exercising software with the intent of ensuring that the Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of test. Each test type addresses a specific testing requirement.

TYPES OF TESTS

Unit testing

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .it is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

Integration testing

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually

satisfaction, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

Functional test

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.

Functional testing is centered on the following items:

Valid Input : identified classes of valid input must be accepted.

Invalid Input : identified classes of invalid input must be rejected.

Functions : identified functions must be exercised.

Output : identified classes of application outputs must be exercised.

Systems/Procedures: interfacing systems or procedures must be invoked.

Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify Business process flows; data fields, predefined processes, and successive processes must be considered for testing. Before functional testing is complete, additional tests are identified and the effective value of current tests is determined.

System Test

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

White Box Testing

White Box Testing is a testing in which in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose. It is purpose. It is used to test areas that cannot be reached from a black box level.

Black Box Testing

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source document, such as specification or requirements document, such as specification or requirements document. It is a testing in which the software under test is treated, as a black box .you cannot "see" into it. The test provides inputs and responds to outputs without considering how the software works.

Unit Testing:

Unit testing is usually conducted as part of a combined code and unit test phase of the software lifecycle, although it is not uncommon for coding and unit testing to be conducted as two distinct phases.

Test strategy and approach Field testing will be performed manually and functional tests will be written in detail.

Test objectives

- All field entries must work properly.
- Pages must be activated from the identified link.
- The entry screen, messages and responses must not be delayed.

Features to be tested

- Verify that the entries are of the correct format
- No duplicate entries should be allowed
- All links should take the user to the correct page.

Integration Testing

Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects.

The task of the integration test is to check that components or software applications, e.g. components in a software system or – one step up – software applications at the company level – interact without error.

Test Results: All the test cases mentioned above passed successfully. No defects encountered.

Acceptance Testing

User Acceptance Testing is a critical phase of any project and requires significant participation by the end user. It also ensures that the system meets the functional requirements.

Test Results: All the test cases mentioned above passed successfully. No defects encountered.

TESTING METHODOLOGIES

The following are the Testing Methodologies:

- **Unit Testing.**
- **Integration Testing.**
- **User Acceptance Testing.**
- **Output Testing.**
- **Validation Testing.**

Unit Testing

Unit testing focuses verification effort on the smallest unit of Software design that is the module. Unit testing exercises specific paths in a module's control structure to ensure complete coverage and maximum error detection. This test focuses on each module individually, ensuring that it functions properly as a unit. Hence, the naming is Unit Testing.During this testing, each module is tested individually and the module interfaces are verified for the consistency with design specification. All important processing path are tested for the expected results. All error handling paths are also tested.

Integration Testing

Integration testing addresses the issues associated with the dual problems of verification and program construction. After the software has been integrated a set of high order tests are conducted. The main objective in this testing process is to take unit tested modules and builds a program structure that has been dictated by design.

The following are the types of Integration Testing:

1)Top Down Integration

This method is an incremental approach to the construction of program structure. Modules are integrated by moving downward through the control hierarchy, beginning with the main program module. The module subordinates to the main program module are incorporated into the structure in either a depth first or breadth first manner.

In this method, the software is tested from main module and individual stubs are replaced when the test proceeds downwards.

2. Bottom-up Integration

This method begins the construction and testing with the modules at the lowest level in the program structure. Since the modules are integrated from the bottom up, processing required for modules subordinate to a given level is always available and the need for stubs is eliminated. The bottom up integration strategy may be implemented with the following steps:

- The low-level modules are combined into clusters into clusters that perform a specific Software sub-function.
- A driver (i.e.) the control program for testing is written to coordinate test case input and output.
- The cluster is tested.
- Drivers are removed and clusters are combined moving upward in the program structure

The bottom up approaches tests each module individually and then each module is module is integrated with a main module and tested for functionality.

OTHER TESTING METHODOLOGIES

User Acceptance Testing

User Acceptance of a system is the key factor for the success of any system. The system under consideration is tested for user acceptance by constantly keeping in touch with the prospective system users at the time of developing and making changes wherever required. The system developed provides a friendly user interface that can easily be understood even by a person who is new to the system.

Output Testing

After performing the validation testing, the next step is output testing of the proposed system, since no system could be useful if it does not produce the required output in the specified format Asking the users about the format required by them tests the outputs generated or displayed by the system under consideration. Hence the output format is considered in 2 ways – one is on screen and another in printed format.

Validation Checking

Validation checks are performed on the following fields.

Text Field:

The text field can contain only the number of characters lesser than or equal to its size. The text fields are alphanumeric in some tables and alphabetic in other tables. Incorrect entry always flashes and error message.

Numeric Field:

The numeric field can contain only numbers from 0 to 9. An entry of any character flashes an error messages. The individual modules are checked for accuracy and what it has to perform. Each module is subjected to test run along with sample data. The individually tested modules are integrated into a single system. Testing involves executing the real data information is used in the program the existence of any program defect is inferred from the output. The testing should be planned so that all the requirements are individually tested.

A successful test is one that gives out the defects for the inappropriate data and produces and output revealing the errors in the system.

Preparation of Test Data

Taking various kinds of test data does the above testing. Preparation of test data plays a vital role in the system testing. After preparing the test data the system under study is tested using that test data. While testing the system by using test data errors are again uncovered and corrected by using above testing steps and corrections are also noted for future use.

Using Live Test Data:

Live test data are those that are actually extracted from organization files. After a system is partially constructed, programmers or analysts often ask users to key in a set of data from their normal activities. Then, the systems person uses this data as a way to partially test the system. In other instances, programmers or analysts extract a set of live data from the files and have them entered themselves.

It is difficult to obtain live data in sufficient amounts to conduct extensive testing. And, although it is realistic data that will show how the system will perform for the typical processing requirement, assuming that the live data entered are in fact typical, such data generally will not test all combinations or formats that can enter the system. This bias toward typical values then does not provide a true systems test and in fact ignores the cases most likely to cause system failure.

Using Artificial Test Data:

Artificial test data are created solely for test purposes, since they can be generated to test all combinations of formats and values. In other words, the artificial data, which can quickly be prepared by a data generating utility program in the information systems department, make possible the testing of all login and control paths through the program.

The most effective test programs use artificial test data generated by persons other than those who wrote the programs. Often, an independent team of testers formulates a testing plan, using the systems specifications.The package “Virtual Private Network” has satisfied all the requirements specified as per software requirement specification and was accepted.

REFERENCES

[1] N. Z. Harun, N. Jaffar, and P. S. J. Kassim, “Physical attributes significant in preserving the social sustainability of the traditional malay settlement,” in *Reframing the Vernacular: Politics, Semiotics, and Representation*. Springer, 2020, pp. 225–238.

[2] D. M. Divakaran and A. Oest, “Phishing detection leveraging machine learning and deep learning: A review,” 2022, *arXiv:2205.07411*.

[3] A. Akanchha, “Exploring a robust machine learning classifier for detecting phishing domains using SSL certificates,” Fac. Comput. Sci., Dalhousie Univ., Halifax, NS, Canada, Tech. Rep. 10222/78875, 2020.

[4] H. Shahriar and S. Nimmagadda, “Network intrusion detection for TCP/IP packets with machine learning techniques,” in *Machine Intelligence and Big Data Analytics for Cybersecurity Applications*. Cham, Switzerland: Springer, 2020, pp. 231–247.

[5] J. Kline, E. Oakes, and P. Barford, “A URL-based analysis of WWW structure and dynamics,” in *Proc. Netw. Traffic Meas. Anal. Conf. (TMA)*, Jun. 2019, p. 800.

[6] A. K. Murthy and Suresha, “XML URL classification based on their semantic structure orientation for web mining applications,” *Proc. Comput. Sci.*, vol. 46, pp. 143–150, Jan. 2015.

[7] A. A. Ubing, S. Kamilia, A. Abdullah, N. Jhanjhi, and M. Supramaniam, “Phishing website detection: An improved accuracy through feature selection and ensemble learning,” *Int. J. Adv. Comput. Sci. Appl.*, vol. 10, no. 1, pp. 252–257, 2019.

[8] A. Aggarwal, A. Rajadesingan, and P. Kumaraguru, “PhishAri: Automatic realtime phishing detection on Twitter,” in *Proc. eCrime Res. Summit*, Oct. 2012, pp. 1–12.

[9] S. N. Foley, D. Gollmann, and E. Snekkenes, *Computer Security—ESORICS 2017*, vol. 10492. Oslo, Norway: Springer, Sep. 2017.

[10] P. George and P. Vinod, “Composite email features for spam identification,” in *Cyber Security*. Singapore: Springer, 2018, pp. 281–289.

CONCLUSION

The Internet consumes almost the whole world in the upcoming age, but it is still growing rapidly. With the growth of the Internet, cybercrimes are also increasing daily using suspicious and malicious URLs, which have a significant impact on the quality of services provided by the Internet and industrial companies. Currently, privacy and confidentiality are essential issues on the internet. To breach the security phases and interrupt strong networks, attackers use phishing emails or URLs that are very easy and effective for intrusion into private or confidential networks. Phishing URLs simply act as legitimate URLs. A machine-learning-based phishing system is proposed in this study. A dataset consisting of 32 URL attributes and more than 11054 URLs was extracted from 11000+websites. This dataset was extracted from the Kaggle repository and used as a benchmark for research. This dataset has already been presented in the form of vectors used in machine learning models. Decision tree, linear regression, random forest, support vector machine, gradient boosting machine, K-Neighbor classifier, naive Bayes, and hybrid (LR+SVC+DT) with soft and hard voting were applied to perform the experiments and achieve the highest performance results. The canopy feature selection with cross fold validation and Grid search hyper parameter optimization techniques are used with LSD Ensemble model. The proposed approach is evaluated in this study by experimenting with a separate machine learning models, and then further evaluation of the study was carried out. The proposed approach successfully achieves its aim with effective efficiency. Future phishing detection systems should combine list-based machine learning-based systems to prevent and detect phishing URLs more efficiently.