A

FIELD BASED PROJECT REPORT

on

A COMPREHENSIVE SURVEY ON COMPUTER FORENSICS, STATE OF-ART, TOOLS, TECHNIQUES, CHALLENGES AND FUTURE DIRECTIONS

BACHELOR OF TECHNOLOGY

in

INFORMATION TECHNOLOGY (IT)

Submitted by

(BATCH :30)

Students Name: B. Sharadha (227Y1A12B1)

Students Name: P. Swathi (227Y1A12C2)

Under the Guidance

of

Mr.Ch.V.V.Narasimha Raju
Assistant Professor

DEPARTMENT OF INFORMATION TECHNOLOGY (IT)

MARRI LAXMAN REDDY

INSTITUTE OF TECHNOLOGY AND MANAGEMENT

(AUTONOMOUS)

JUNE 2024

MLRS

EMPOWERING THROUGH INNOVATIONS

MARRI LAXMAN REDDY
INSTITUTE OF TECHNOLOGY AND MANAGEMENT
(AN AUTONOMOUS INSTITUTION)
(Approved by AICTE, New Delhi & Affiliated to JNTUH, Hyderabad)
Accredited by NBA and NAAC with 'A' Grade & Recognized Under Section2(f) & 12(B)of the UGC act,1956

CERTIFICATE

This is to certify that the project report titled "**A Comprehensive Survey on Computer Forensics, State of -Art, Tools, Techniques, Challenges and Future Directions"** is being submitted by **B. Sharadha (227Y1A12B1), P. Swathi (227Y1A12C2)** in **II B. Tech II Semester Information Technology (IT)** is a record bonafide work carried out by him. The results embodied in this report have not been submitted to any other University for the award of any degree.

Internal Guide **HOD**

Principal

DECLARATION

We hereby declare that the Field-based Project Report entitled, "**A Comprehensive Survey on Computer Forensics, State of -Art, Tools, Techniques, Challenges and Future Directions"** submitted for the B. Tech degree is entirely our work and all ideas and references have been duly acknowledged. It does not contain any work for the award of any other degree.

Date:

Students Name

B. Sharadha(227Y1A12B1),

P. Swathi (227Y1A12C2).

MLRS

MARRI LAXMAN REDDY

INSTITUTE OF TECHNOLOGY AND MANAGEMENT

(AN AUTONOMOUS INSTITUTION)

(Approved by AICTE, New Delhi & Affiliated to JNTUH, Hyderabad)

Accredited by NBA and NAAC with 'A' Grade & Recognized Under Section2(f) & 12(B)of the UGC act,1956

ACKNOWLEDGEMENT

We are happy to express our deep sense of gratitude to the principal of the college **Dr. R. Murali Prasad,** Professor, Marri Laxman Reddy Institute of Technology & Management, for having provided us with adequate facilities to pursue our project.

We would like to thank **Dr. M. Nagalakshmi ,** Professor and Head of Department Information Technology(IT), Marri Laxman Reddy Institute of Technology & Management, for having provided the freedom to use all the facilities available in the department, especially the laboratories and the library.

We are very grateful to our project guide **, Mr. Ch.V.V.Narasimha Raju** Asst.. Prof., Department of Information Technology (IT), Marri Laxman Reddy Institute of Technology & Management, for his extensive patience and guidance throughout our project work.

We sincerely thank my seniors and all the teaching and non-teaching staff of the Department of Information Technology for their timely suggestions, healthy criticism and motivation during the course of this work.

We would also like to thank my classmates for always being there whenever I needed help or moral support.with great respect and obedience, we thank our parents and brother who were the backbone behind our deeds.

Finally, We express our immense gratitude with pleasure to the other individuals who have either directly or indirectly contributed to our needs at right time for the development and success of this work.

CONTENTS

S NO.	TITLE	PAGE NO.
	ABSTRACT	**7**
	LIST OF FIGURES	**8**
	LIST OF TABLES	8
1	**INTRODUCTION**	**9**
	1.1 Motivation	9
	1.2 Problem	10
	1.3 Solution	10
	1.4 Scope	10
	1.5 Problem Definition	11
	1.6 Objective	11
	1.7 Limitations	11
	1.8 Organization Of Document	11
2	**LITERATURE SURVEY**	**12**
	2.1 History of Python	13
	2.2 Existing System	14
	2.3 Disadvantages of Existing System	14
	2.4Proposed System	15
	2.5 Advantages of Proposed System	15
	2.6 Summary	15
3	**ANALYSIS**	**16**

	3.1 Introduction	16
	3.2 Software Requirement Specification	16
	3.3 Data Flow Diagram	17
	3.4 Algorithms	18
	3.5 System Study	23
4	**DESIGN**	**25**
	4.1 Introduction	25
	4.2 Use case Diagram	26
	4.3 Module Design and Organization	27
	4.4 Input and output Design	28
	4.5 Summary	29
6	**TESTING AND VALIDATION**	**30**
	6.1 Introduction	30
	6.2 Design of Test Cases Scenarios & Validation	31
	6.3 result	34
7	**CONCLUSION AND FUTURE ENHANCEMENTS**	**36**
8	**REFERENCES**	**37**

ABSTRACT

Computer forensics is a critical field within cybersecurity aimed at investigating digital crimes and incidents. This survey explores the current state-of-the-art techniques, tools, challenges, and future directions in computer forensics. The field has evolved significantly to handle a wide range of cyber incidents, from data breaches to cyberattacks and fraud. Modern computer forensics utilizes advanced techniques such as disk imaging, data recovery, memory analysis, network forensics, and mobile device forensics. These techniques are essential for gathering evidence from various digital sources while maintaining the integrity and admissibility of the evidence in legal proceedings. Numerous tools and technologies have been developed to aid forensic investigators. These include software tools for disk imaging and analysis (e.g., EnCase, FTK), memory forensics tools (e.g., Volatility), network forensic tools (e.g., Wireshark), and mobile device forensics tools (e.g., Cellebrite, Oxygen Forensic Detective).

These tools help in efficiently collecting, analyzing, and presenting digital evidence. Despite advancements, computer forensics faces several challenges. These include dealing with encryption and anti-forensic techniques used by criminals to hide evidence, the rapidly evolving landscape of digital devices and operating systems, and the legal complexities of digital evidence admissibility in courts. Future directions in computer forensics include enhancing tools and techniques for handling large volumes of data , improving automation and artificial intelligence capabilities for faster analysis, addressing privacy concerns in digital investigations, and developing standardized practices and guidelines for digital evidence handling and reporting. In conclusion, computer forensics plays a crucial role in investigating and mitigating cyber incidents. Advancements in technology and methodologies continue to shape the field, aiming to better protect digital assets and ensure justice in the digital age.

LIST OF FIGURES

FIG. NO	FIG. NAME	PAGE NO.
1.1	Architecture Of The Setup	4
1.2	Flow Chart Of The System	5
1.3	Essential Use Case Diagram	15
1.4	Class Diagram	16

LIST OF TABLES

TABLE NO.	TABLE TITLE	PAGE NO.
3.1	Software Requirements	14
3.2	Hardware Requirements	14
3.3	Summary	35

INTRODUCTION

In the last years, web services usage has grown drastically due to the current digital transformation. Companies motivate the change by providing their services online, like e-banking, e-commerce or SaaS (Software as a Service) [1]. Nowadays, due to the COVID-19 pandemic, restrictions have spread out the work-from-home model, which implies extra millions of workers, students, and teachers developing their activities remotely [2], leading to a substantial additional workload for services such as email, student platforms, VPNs or company portals. Therefore, there are even more potential targets exposed to phishing attacks, where phishers try to mimic legitimate websites to steal users' credentials or payment information [3], [4]. Recent studies [5], [6] concluded that phishing is one of the most significant attacks based on social engineering during the COVID-19 pandemic, together with spam emails and websites to execute these attacks.

Identifying phishing sites through their HTTP protocol is no longer a valid rule. In the 3*rd* quarter of 2017 [7], the APWG reported that less than 25% of phishing websites were hosted under HTTPS protocol, whilst this amount has increased up to 83% in 1*st* quarter of 2021 [8]. These websites provide secure end-to-end communication, which transmits a false safe impression to the user while making an online transaction [9]. Furthermore, the Anti- Phishing Working Group (APWG) [10] has reported a significant increase in phishing attacks, i.e. from 165; 772 to 611; 877 websites, just between the first quarter of 2020 and 2021 respectively. A reason behind this increase might be that people have resorted (and still are) to online services during the COVID-19 pandemic.

1. Motivation

- **Introduction**
  - In cyberattacks, digital forensic investigators can help identify what information was accessed, stolen, copied or distributed.
  - They can identify whether attackers remain in the systems with continued access to an organization's data.
- **Digital Forensics**
  - Digital forensics is the process of extracting and analyzing data contained within digital systems to find evidence that can help resolve cyberattacks, disputes, litigation, and criminal cases.
  - Using methods of electronic discovery, trained computer forensic analysts examine computers, cell phones, hard drives, networks,

    - systems, and digital components for digital forensics investigative purposes.

2. Problem

- **Current Challenges**
    - Source Related Issues and Challenges.
    - Law Related Issues and Challenges.
- **Data Challenges**
    - Volatile Data that would be lost if the computer is turned off.
    - Persistent Data that remains unaffected when the computer is turned off.

3. Solution

- **State-of-the-Art in Computer Forensics**
    - Overview of current trends and advancements.
    - Integration of blockchain technology and its impact on digital forensics.
- **Tools and Techniques**
    - Analysis of popular forensic tools (e.g., EnCase, FTK, Autopsy).
    - Techniques for data acquisition, preservation, and analysis.
    - Mobile and cloud forensics methodologies.

4. Scope

- **Project Scope**

    - In the future, digital forensic software will improve its ability to collect data from the cloud.
    - Additionally, it will become more effective in decrypting and analyzing cloud services.
- **Exclusions**
    - ➢ Data recovery. If the data is encrypted, the investigator will not be able to decrypt the information without access to encryption keys

5. Problem Definition

- Digital forensic evidence presents a variety of risks and challenges to investigators. From volatile data that can be altered or erased to environmental concerns when devices are exposed to extreme temperatures, investigators must take great care that the integrity of data is not compromised.

6. Objective

- **Primary Objectives**
  - From a technical standpoint, the main goal of computer forensics is to identify, collect, preserve, and analyze data in a way that preserves the integrity of the evidence collected so it can be used effectively in a legal case.
- **Secondary Objectives**
  - Forensic readiness is broadly defined as maximizing the potential use of digital evidence, while minimizing the costs of a digital forensics investigation.

7. Limitations

- **Data Limitations**
  - Data Sharing: There might be limited sharing of detailed forensic data due to confidentiality agreements, proprietary concerns, or ongoing investigations.
- **Model Limitations**
  - Detailed studies and comprehensive reports on advanced forensic techniques or emerging challenges may be sparse.
- **External Factors**

  - ➢ Factors related to globalization, including international cooperation, data sovereignty issues, and differences in legal systems, pose challenges for conducting cross-border forensic investigations.

8. Organization of Document

- **Section Overview**
  - Description of how the document is organized.
  - Brief overview of each section and its purpose.
- **Flow of Information**
  - Logical flow of information from problem definition to solution and results.
- **Appendices and References**
  - Additional information and references included at the end of the document.

LITERATURE SURVEY

Python is a high-level, interpreted, interactive and object-oriented scripting language. Python is designed to be highly readable. It uses English keywords frequently where as other languages use punctuation, and it has fewer syntactical constructions than other languages.

- **Python is Interpreted:** Python is processed at runtime by the interpreter. You do not need to compile your program before executing it. This is similar to PERL and PHP.

- **Python is Interactive:** You can actually sit at a Python prompt and interact with the interpreter directly to write your programs.

- **Python is Object-Oriented:** Python supports Object-Oriented style or technique of programming that encapsulates code within objects.

- **Python is a Beginner's Language:** Python is a great language for the beginner-level programmers and supports the development of a wide range of applications from simple text processing to WWW browsers to games.

History of Python

Python was developed by Guido van Rossum in the late eighties and early nineties at the National Research Institute for Mathematics and Computer Science in the Netherlands.

Python is derived from many other languages, including ABC, Modula-3, C, C++, Algol-68, Small Talk, and Unix shell and other scripting languages.

Python is copyrighted. Like Perl, Python source code is now available under the GNU General Public License (GPL).

Python is now maintained by a core development team at the institute, although Guido van Rossum still holds a vital role in directing its progress.

Python Features

- **Easy-to-learn:** Python has few keywords, simple structure, and a clearly defined syntax. This allows the student to pick up the language quickly.

- **Easy-to-read:** Python code is more clearly defined and visible to the eyes.

- **Easy-to-maintain:** Python's source code is fairly easy-to-maintain.

- **A broad standard library:** Python's bulk of the library is very portable and cross-platform compatible on UNIX, Windows, and Macintosh.

- **Interactive Mode:** Python has support for an interactive mode which allows interactive testing and debugging of snippets of code.

- **Portable:** Python can run on a wide variety of hardware platforms and has the same interface on all platforms.

- **Extendable:** You can add low-level modules to the Python interpreter. These modules enable programmers to add to or customize their tools to be more efficient.

- **Databases:** Python provides interfaces to all major commercial databases.

- **GUI Programming:** Python supports GUI applications that can be created and ported to many system calls, libraries and windows systems, such as Windows MFC, Macintosh, and the X Window system of Unix.

- **Scalable:** Python provides a better structure and support for large programs than shell scripting.

Python has a big list of good features:

- It supports functional and structured programming methods as well as OOP.

- It provides very high-level dynamic data types and supports dynamic type checking.

- IT supports automatic garbage collection.

- It can be easily integrated with C, C++, COM, ActiveX, CORBA, and Java.

The results of this analysis demonstrate the potential of machine learning in transforming raw sales data into valuable forecasts, enabling Big Mart to make informed decisions and maintain a competitive edge in the market.

EXISTING SYSTEM

- Systematic Approach: Utilize systematic literature review methodologies to ensure comprehensive coverage of relevant studies, papers, and reports.

- Database Selection: Identify and select appropriate databases (e.g., IEEE Xplore, ACM Digital Library, Google Scholar) to search for peer-reviewed articles, conference proceedings, and technical reports.

- Search Strategy: Develop a detailed search strategy using keywords related to computer forensics, including specific tools (e.g., EnCase, FTK), techniques (e.g., disk imaging, memory forensics), challenges (e.g., data privacy, cyber threats), and future directions (e.g., AI in forensics, blockchain technologies).

DISADVANTAGES OF EXISTING SYSTEM

Knowledge Synthesis: It provides a structured synthesis of existing knowledge, consolidating dispersed information from diverse sources such as academic literature, industry reports, and practitioner insights.

Identification of Trends and Innovations: By systematically reviewing literature and case studies, the survey identifies emerging trends, innovative tools, and evolving techniques in computer forensics.

PROPOSED SYSTEM

By following this proposed system, stakeholders can effectively conduct a comprehensive survey on computer forensics, leveraging existing knowledge, identifying innovations, addressing challenges, and guiding future directions in this critical field. Implement recommendations derived from the survey findings in practice, research, policy development, and education. Monitor developments in computer forensics to update the survey periodically and keep stakeholders informed about evolving trends and technologies. Subject the survey report to peer review by experts in computer forensics to ensure methodological rigor and accuracy of interpretations. Iterate on the survey methodology based on peer feedback and incorporate suggestions for further refinement. Structure the survey report to present comprehensive insights into the state of computer forensics, tools, techniques, challenges, and future directions.

ADVANTAGES:

- **Holistic Understanding**: By systematically reviewing and synthesizing literature and expert insights, the survey provides a comprehensive understanding of the current state of computer forensics. This includes advancements in tools, techniques, methodologies, and emerging trends.

- **Evidence-Based Insights**: The survey generates evidence-based insights into effective forensic practices, validated tools, and techniques. Stakeholders can make informed decisions based on empirical data rather than anecdotal evidence or assumptions.

- **Identification of Trends and Innovations**: It identifies emerging trends and innovations in computer forensics, such as advancements in digital forensic tools, new techniques for analyzing digital evidence, and applications of artificial intelligence and machine learning.

- **Benchmarking and Best Practices**: Stakeholders can benchmark their current practices against identified best practices and standards derived from the survey. This promotes continuous improvement and adoption of effective forensic methodologies.

ANALYSIS

SOFTWARE REQUIREMENTS SPECIFICATION

The software requirements specification (SRS) for the predictive analysis of Big Mart sales using machine learning outlines the essential features and functionalities necessary to develop a robust and efficient predictive model. The system must be capable of ingesting and preprocessing vast amounts of historical sales data, including details such as product categories, store locations, seasonal trends, and promotional activities.

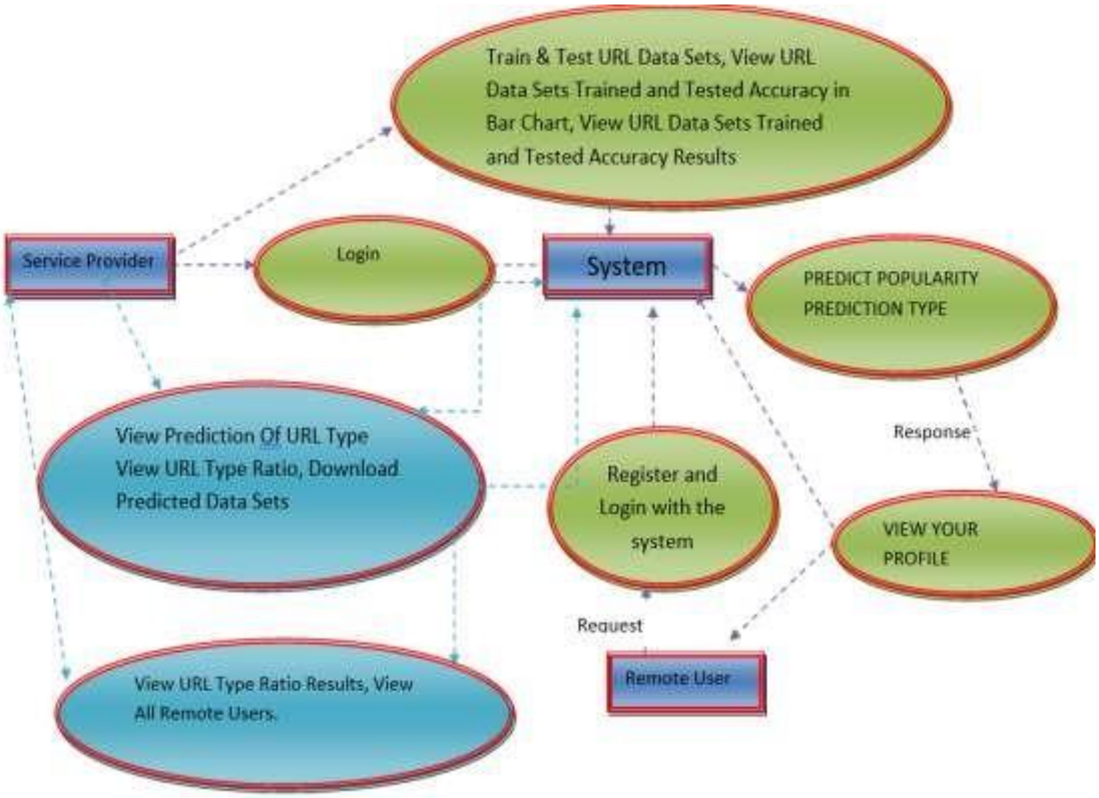
Key functional requirements include data extraction and transformation capabilities, ensuring data integrity and consistency throughout the pipeline. The system should support various machine learning algorithms, enabling the selection and comparison of models such as linear regression, decision trees, random forests, and gradient boosting machines.

The SRS also emphasizes the need for scalability and performance, as the system must handle large datasets and deliver timely forecasts. Integration with Big Mart's existing IT infrastructure is crucial, facilitating seamless data flow between different departments and systems.

Non-functional requirements include robust security measures to protect sensitive sales data, ensuring compliance with relevant data privacy regulations. The system should offer high availability and reliability, minimizing downtime and ensuring continuous operation. User accessibility and ease of use are also paramount, with the interface designed to cater to users with varying levels of technical expertise.

Finally, the SRS specifies the need for comprehensive documentation and support, including user manuals, technical documentation, and training resources, to ensure smooth implementation and ongoing maintenance. This detailed SRS provides a clear roadmap for developing an advanced predictive analysis system that leverages machine learning to enhance Big Mart's sales forecasting capabilities and operational efficiency.

DATA FLOW DIAGRAM


Train & Test URL Data Sets, View URL Data Sets Trained and Tested Accuracy in Bar Chart, View URL Data Sets Trained and Tested Accuracy Results
Service Provider
Login
System
PREDICT POPULARITY PREDICTION TYPE
Response
View Prediction Of URL Type View URL Type Ratio, Download Predicted Data Sets
Register and Login with the system
VIEW YOUR PROFILE
Request
Remote User
View URL Type Ratio Results, View All Remote Users.

ALGORITHMS

Decision tree classifiers

Decision tree classifiers are used successfully in many diverse areas. Their most important feature is the capability of capturing descriptive decision making knowledge from the supplied data. Decision tree can be generated from training sets. The procedure for such generation based on the set of objects (S), each belonging to one of the classes C1, C2, …, Ck is as follows:

Step 1. If all the objects in S belong to the same class, for example Ci, the decision tree for S consists of a leaf labeled with this class

Step 2. Otherwise, let T be some test with possible outcomes O1, O2,.. On. Each object in S has one outcome for T so the test partitions S into subsets S1, S2,… Sn where each object in Si has outcome Oi for T. T becomes the root of the decision tree and for each outcome Oi we build a subsidiary decision tree by invoking the same procedure recursively on the set Si.

Gradient boosting

Gradient is a boosting machine learning language used in regression and classification tasks, among others. It gives a prediction model in the form of an ensemble of weak prediction models, which are typically decision trees.[1][2] When a decision tree is the weak learner, the resulting algorithm is called gradient-boosted trees; it usually outperforms random forest .A gradient-boosted trees model is built in a stage-wise fashion as in other boosting methods, but it generalizes the other methods by allowing optimization of an arbitrary differentiable loss function.

K-Nearest Neighbors (KNN)

- Simple, but a very powerful classification algorithm
- Classifies based on a similarity measure
- Non-parametric
- Lazy learning
- Does not “learn” until the test example is given

- Whenever we have a new data to classify, we find its K-nearest neighbors from the training data

Example

- Training dataset consists of k-closest examples in feature space
- Feature space means, space with categorization variables (non-metric variables)
- Learning based on instances, and thus also works lazily because instance close to the input vector for test or prediction may take time to occur in the training dataset

Logistic regression Classifiers

Logistic regression analysis studies the association between a categorical dependent variable and a set of independent (explanatory) variables. The name logistic regression is used when the dependent variable has only two values, such as 0 and 1 or Yes and No. The name multinomial logistic regression is usually reserved for the case when the dependent variable has three or more unique values, such as Married, Single, Divorced, or Widowed. Although the type of data used for the dependent variable is different from that of multiple regression, the practical use of the procedure is similar.

Logistic regression competes with discriminant analysis as a method for analyzing categorical-response variables. Many statisticians feel that logistic regression is more versatile and better suited for modeling most situations than is discriminant analysis. This is because logistic regression does not assume that the independent variables are normally distributed, as discriminant analysis does.

This program computes binary logistic regression and multinomial logistic regression on both numeric and categorical independent variables. It reports on the regression equation as well as the goodness of fit, odds ratios, confidence limits, likelihood, and deviance. It performs a comprehensive residual analysis including diagnostic residual reports and plots. It can perform an independent variable subset selection search, looking for the best regression model with the fewest independent variables. It provides confidence intervals on predicted values and provides ROC curves to help determine the best cutoff point for classification. It allows you to validate your results by automatically classifying rows that are not used during the analysis.

Naïve Bayes

The naive bayes approach is a supervised learning method which is based on a simplistic hypothesis: it assumes that the presence (or absence) of a particular feature of a class is unrelated to the presence (or absence) of any other feature.

Yet, despite this, it appears robust and efficient. Its performance is comparable to other supervised learning techniques. Various reasons have been advanced in the literature. In this tutorial, we highlight an explanation based on the representation bias. The naive bayes classifier is a linear classifier, as well as linear discriminant analysis, logistic regression or linear SVM (support vector machine). The difference lies on the method of estimating the parameters of the classifier (the learning bias).

While the Naive Bayes classifier is widely used in the research world, it is not widespread among practitioners which want to obtain usable results. On the one hand, the researchers found especially it is very easy to program and implement it, its parameters are easy to estimate, learning is very fast even on very large databases, its accuracy is reasonably good in comparison to the other approaches. On the other hand, the final users do not obtain a model easy to interpret and deploy, they does not understand the interest of such a technique.

Thus, we introduce in a new presentation of the results of the learning process. The classifier is easier to understand, and its deployment is also made easier. In the first part of this tutorial, we present some theoretical aspects of the naive bayes classifier. Then, we implement the approach on a dataset with Tanagra. We compare the obtained results (the parameters of the model) to those obtained with other linear approaches such as the logistic regression, the linear discriminant analysis and the linear SVM. We note that the results are highly consistent. This largely explains the good performance of the method in comparison to others. In the second part, we use various tools on the same dataset (Weka 3.6.0, R 2.9.2, Knime 2.1.1, Orange 2.0b and RapidMiner 4.6.0). We try above all to understand the obtained results.

Random Forest

Random forests or random decision forests are an ensemble learning method for classification, regression and other tasks that operates by constructing a multitude of decision trees at training time. For classification tasks, the output of the random forest is the class selected by most trees. For regression tasks, the mean or average prediction of the individual trees is returned. Random decision forests correct for decision trees' habit of overfitting to their training set. Random forests generally outperform decision trees, but their accuracy is lower than gradient boosted trees. However, data characteristics can affect their performance.

The first algorithm for random decision forests was created in 1995 by Tin Kam Ho[1] using the random subspace method, which, in Ho's formulation, is a way to implement the "stochastic discrimination" approach to classification proposed by Eugene Kleinberg.

An extension of the algorithm was developed by Leo Breiman and Adele Cutler, who registered "Random Forests" as a trademark in 2006 (as of 2019, owned by Minitab, Inc.).The extension combines Breiman's "bagging" idea and random selection of features, introduced first by Ho[1] and later independently by Amit and Geman[13] in order to construct a collection of decision trees with controlled variance.

Random forests are frequently used as "blackbox" models in businesses, as they generate reasonable predictions across a wide range of data while requiring little configuration.

SVM

In classification tasks a discriminant machine learning technique aims at finding, based on an independent and identically distributed (iid) training dataset, a discriminant function that can correctly predict labels for newly acquired instances. Unlike generative machine learning approaches, which require computations of conditional probability distributions, a discriminant classification function takes a data point *x* and assigns it to one of the different classes that are a part of the classification task. Less powerful than generative approaches, which are mostly used when prediction involves outlier detection, discriminant approaches require fewer computational resources and less training data, especially for a multidimensional feature space and when only posterior probabilities are needed. From a geometric perspective, learning a classifier is equivalent to finding the equation for a multidimensional surface that best separates the different classes in the feature space.

SVM is a discriminant technique, and, because it solves the convex optimization problem analytically, it always returns the same optimal hyperplane parameter—in contrast to genetic algorithms (*GAs*) or perceptrons, both of which are widely used for classification in machine learning. For perceptrons, solutions are highly dependent on the initialization and termination criteria. For a specific kernel that transforms the data from the input space to the feature space, training returns uniquely defined SVM model parameters for a given training set

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

Predictive analysis for Big Mart sales using machine learning is economically feasible due to several key factors. Firstly, the use of machine learning algorithms can significantly enhance the accuracy of sales forecasts by analyzing vast amounts of historical sales data and identifying patterns that traditional methods might miss. This increased accuracy can lead to better inventory management, reducing both overstock and stockout situations, which directly translates into cost savings. Moreover, machine learning models can be continuously updated with new data, ensuring that predictions remain relevant and accurate over time. The initial investment in setting up a machine learning infrastructure, while substantial, is offset by the long-term benefits of optimized operations and improved decision-making.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

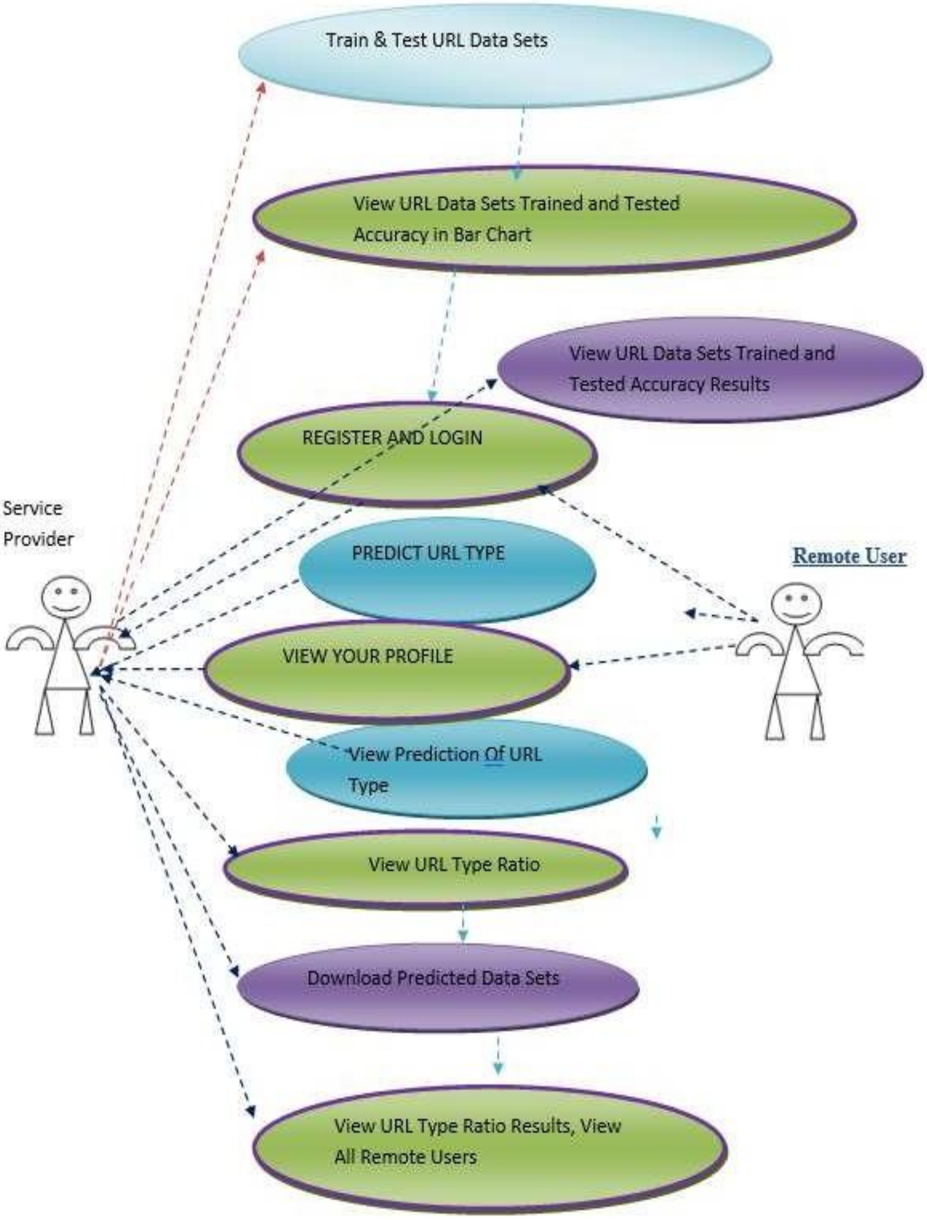
This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. This will lead to high demands on the available technical resources. This will lead to high demands being placed on the client. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system. Machine learning models, such as linear regression, decision trees, random forests, and gradient boosting, can be trained on historical sales data to predict future sales trends. The data can include various features like product type, store location, seasonal effects, promotional campaigns, and economic indicators.

SOCIAL FEASIBILITY

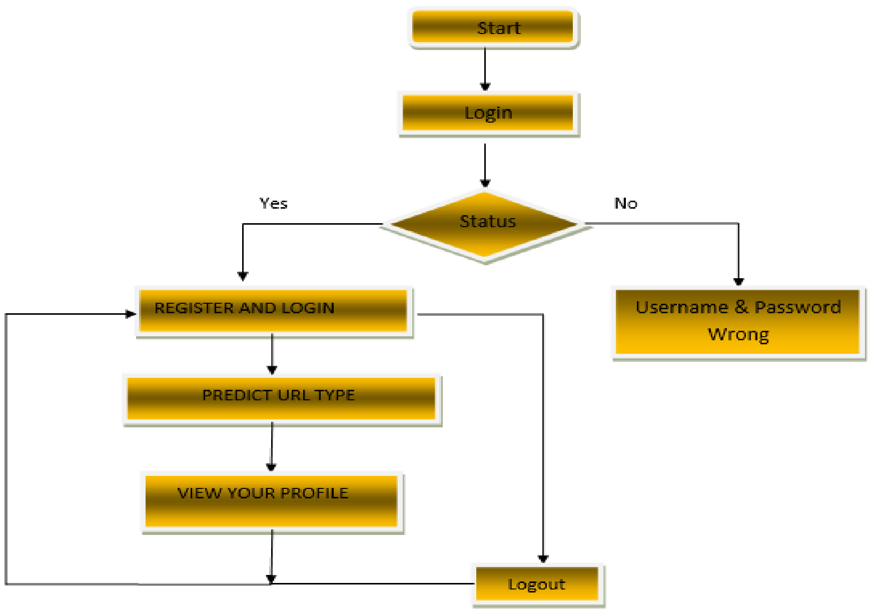
The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity. The level of acceptance by the users solely depends 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.

DESIGN

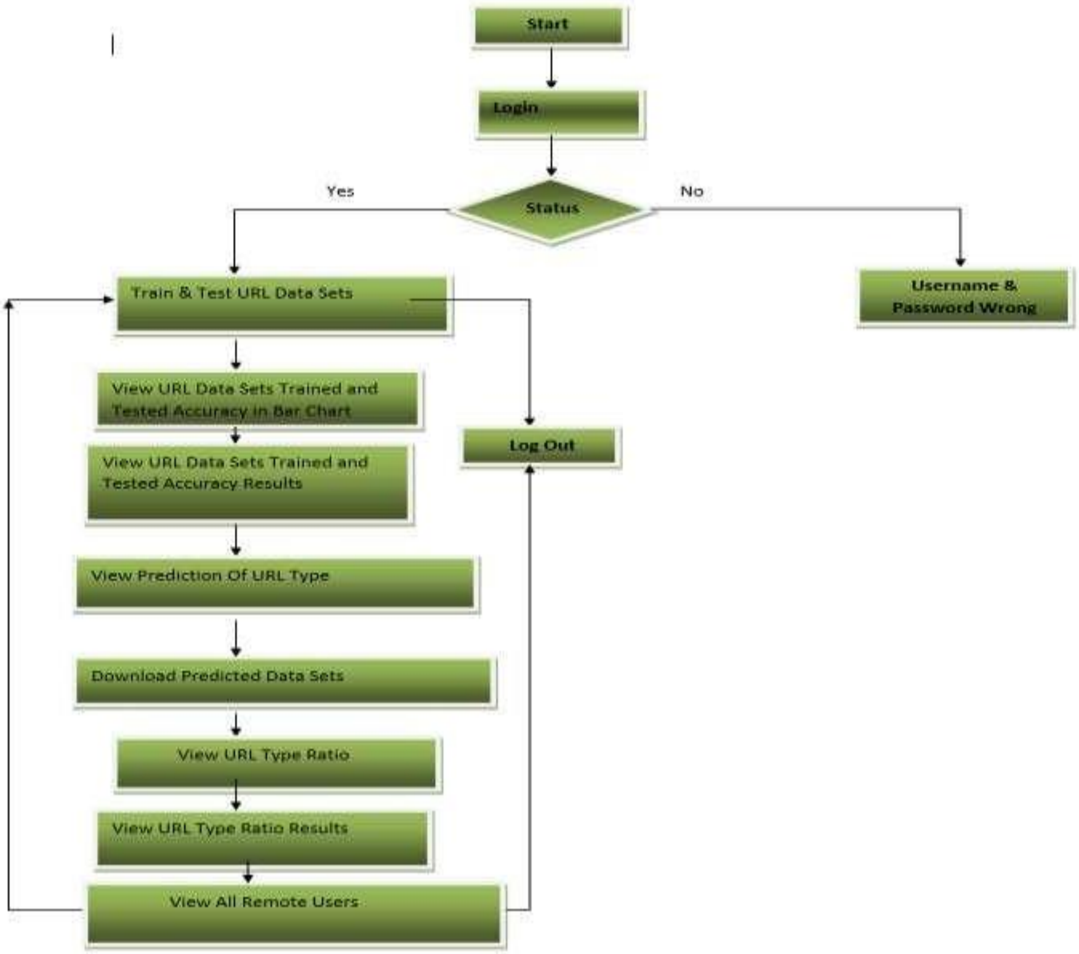
USE CASE DIAGRAM


Train & Test URL Data Sets
View URL Data Sets Trained and Tested Accuracy in Bar Chart
View URL Data Sets Trained and Tested Accuracy Results
REGISTER AND LOGIN
Service Provider
PREDICT URL TYPE
Remote User
VIEW YOUR PROFILE
View Prediction Of URL Type
View URL Type Ratio
Download Predicted Data Sets
View URL Type Ratio Results, View All Remote Users

➢ Flow Chart: Remote User



Flow Chart: Service Provider


Start
Login
Yes
Status
No
Train & Test URL Data Sets
Username & Password Wrong
View URL Data Sets Trained and Tested Accuracy in Bar Chart
Log Out
View URL Data Sets Trained and Tested Accuracy Results
View Prediction Of URL Type
Download Predicted Data Sets
View URL Type Ratio
View URL Type Ratio Results
View All Remote Users

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 Login, Train & Test URL Data Sets, View URL Data Sets Trained and Tested Accuracy in Bar Chart, View URL Data Sets Trained and Tested 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.

Remote User

In this module, there are n numbers of users are present. User should register before doing any operations. Once user registers, their details will be stored to the database. After registration successful, he has to login by using authorized user name and password. Once Login is successful user will do some operations like REGISTER AND LOGIN, PREDICT URL TYPE, VIEW YOUR PROFILE..

SYSTEM SPECIFICATIONS

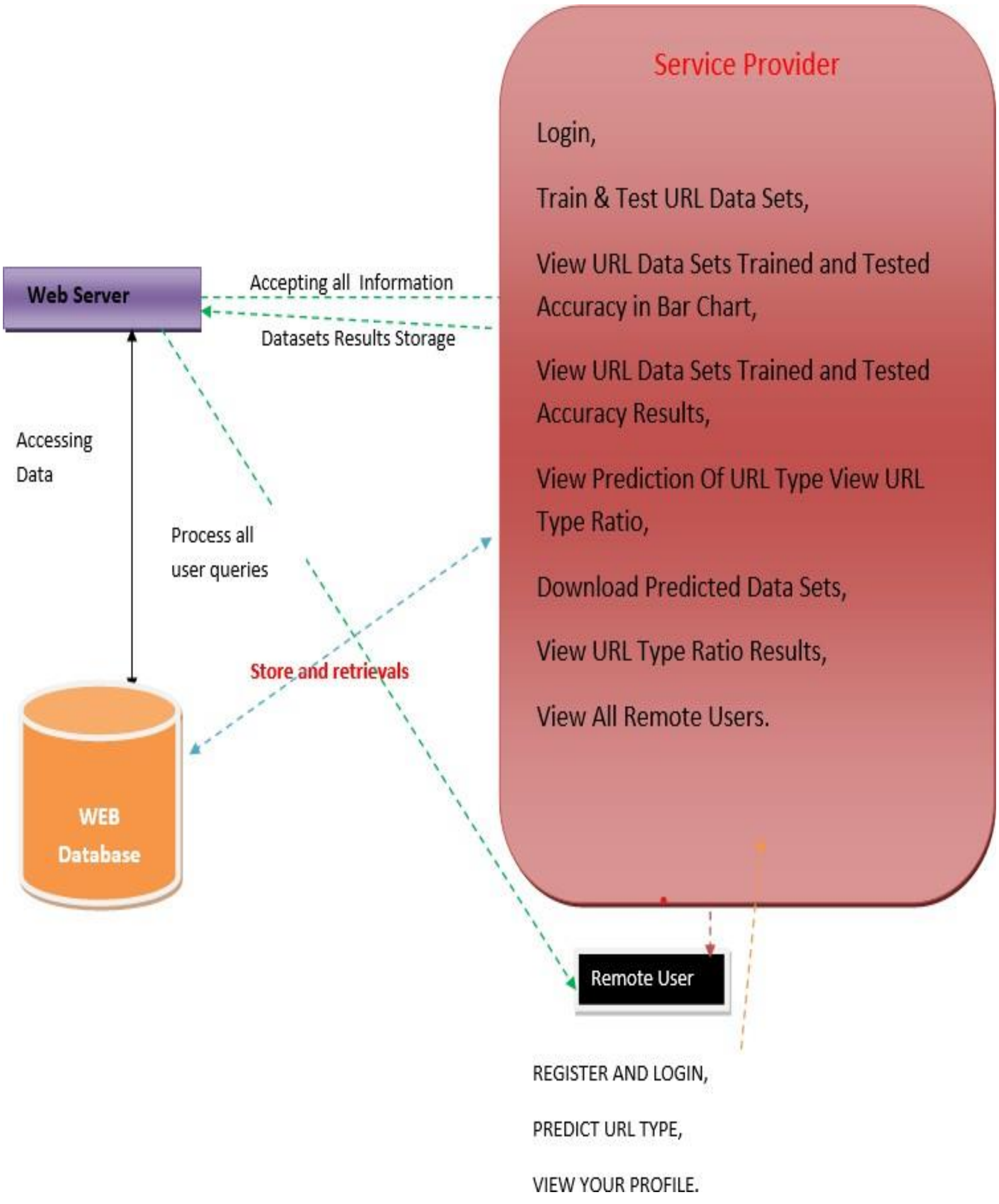
HARDWARE REQUIREMENTS:

❖ **System**	Pentium IV 2.4 GHz.
❖ **Hard Disk**	40 GB.
❖ **Floppy Drive**	1.44 Mb.
❖ **Monitor**	14’ Colour Monitor.
❖ **Mouse**	Optical Mouse.
❖ **Ram**	512 Mb.

SOFTWARE REQUIREMENTS:

❖ **Operating system**	Windows 7 Ultimate.
❖ **Coding Language**	Python.
❖ **Front-End**	Python.
❖ **Designing**	Html,css,javascript.
❖ **Data Base**	MySQL.

Architecture Diagram


Service Provider
Login,
Train & Test URL Data Sets,
View URL Data Sets Trained and Tested Accuracy in Bar Chart,
View URL Data Sets Trained and Tested Accuracy Results,
View Prediction Of URL Type View URL Type Ratio,
Download Predicted Data Sets,
View URL Type Ratio Results,
View All Remote Users.
Web Server
Accepting all Information
Datasets Results Storage
Accessing Data
Process all user queries
Store and retrievals
WEB Database
Remote User
REGISTER AND LOGIN,
PREDICT URL TYPE,
VIEW YOUR PROFILE.

TESTING AND VALIDATION

Testing and validation are crucial steps in developing a reliable predictive model. These processes ensure that the model performs well not only on the training data but also on unseen data. This helps to avoid overfitting, where a model performs well on the training data but poorly on new, unseen data.

1. **Testing**: This involves evaluating the performance of the model on a separate test dataset that was not used during the training phase. It provides an unbiased estimate of the model's performance.
2. **Validation**: Validation techniques, such as cross-validation, involve partitioning the training data into multiple subsets. The model is trained on some subsets while being validated on others. This process is repeated several times to ensure the model's robustness and reliability.

SYSTEM TEST

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.

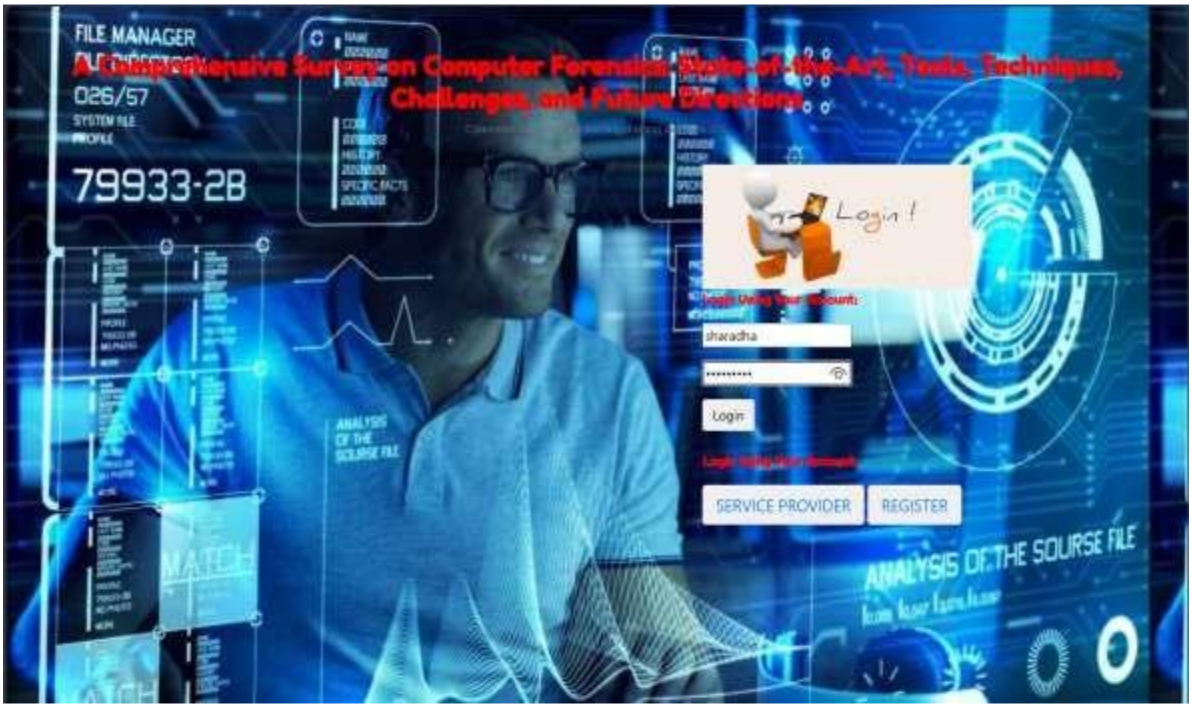
Acceptance Testing

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.

RESULT

LOGIN PAGE



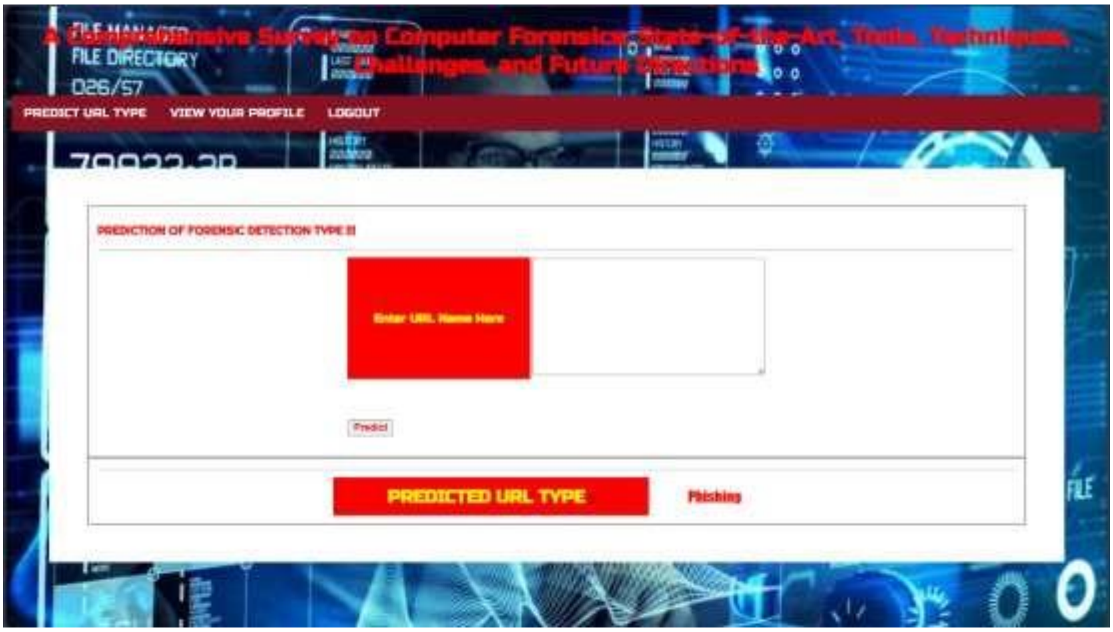
A Comprehensive Survey on Computer Forensics: State-of-the-Art, Tools, Techniques, Challenges, and Future Directions

Login Using Your Account:

sharadha

Here we can do registration.

PHISHING OR NON PHISHING



A Comprehensive Survey on Computer Forensics: State-of-the-Art, Tools, Techniques, Challenges, and Future Directions

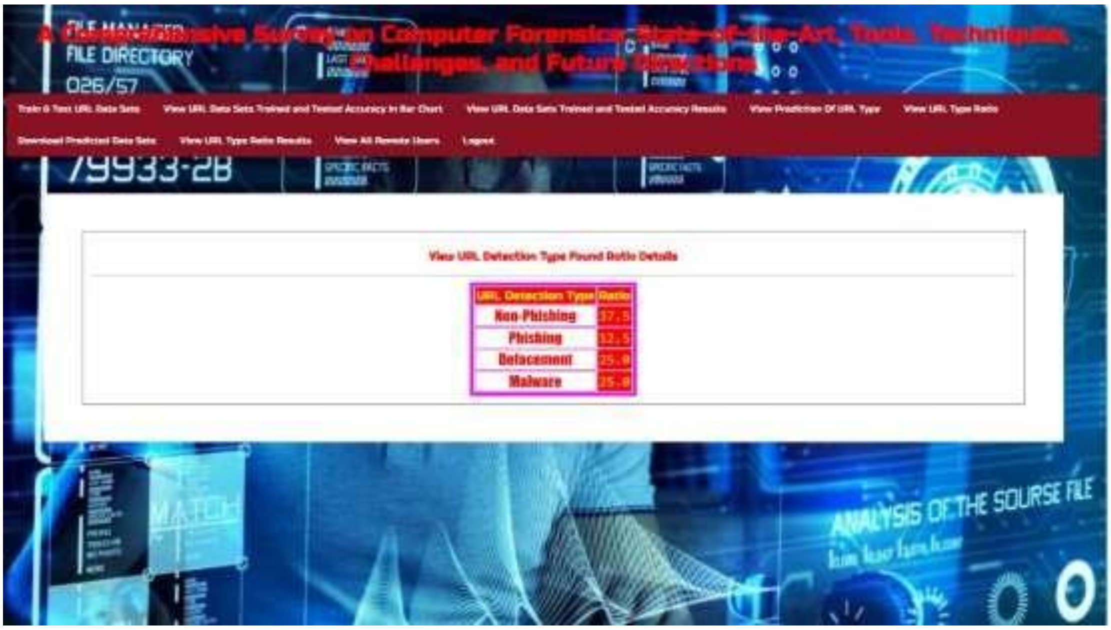
PREDICTION OF FORENSIC DETECTION TYPE !!!

Enter URL Name Here

PREDICTED URL TYPE Phishing

It states about the predicted url is harmful or not. That is phishing means it is a harmful URL whereas non phishing means it is not a harmful URL.

Predicted Data

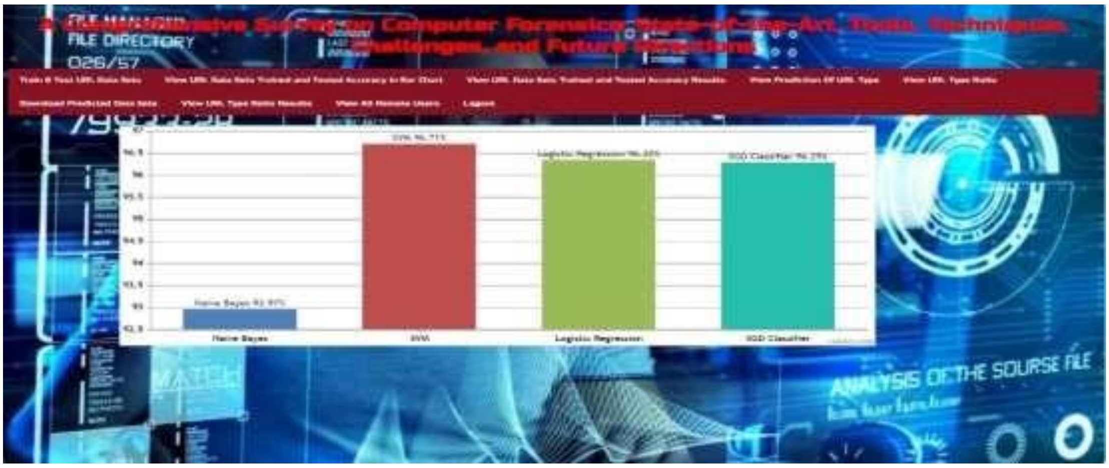

A Comprehensive Survey on Computer Forensics: State-of-the-Art, Tools, Techniques, Challenges, and Future Directions
Train & Test URL Data Sets
View URL Data Sets Trained and Tested Accuracy in Bar Chart
View URL Data Sets Trained and Tested Accuracy Results
View Prediction Of URL Type
View URL Type Ratio
Download Predicted Data Sets
View URL Type Ratio Results
View All Remote Users
Logout

View URL Detection Type Found Ratio Details

URL Detection Type	Ratio
Non-Phishing	37.5
Phishing	12.5
Defacement	25.0
Malware	25.0

The above image explains about percentage of predicted data

Bar chart



The bar chart shows the graphical representation of the predicted data.

CONCLUSION

Phishing detection mechanism aims to improve current blacklist methods, protecting users from malicious login forms. Our work provides an updated dataset PILU-90K for researchers to train and test their approaches. This dataset includes legitimate login URLs which are the most representative scenario for real-world phishing detection.

We explored several URL-based detection models using deep learning and machine learning solutions trained with phishing and legitimate home URLs. The main advantage of our approach is the low false-positive rate when classifying this type of URL. Among the different evaluated models, TFIDF combined with N-gram and LR algorithm obtained the best results with a 96:50% accuracy. In comparison with the current state-of-the-art, reviewed in Section II, our approach present three main advantages:

No dependence on external services. A limitation of the description methods that use features such as WHOIS domain age, page ranking on Google or Alexa or online blacklists, is their dependence on those services. Network slowdowns and service shortages can negatively impact analysis time, making real-time execution infeasible. Since phishing websites have a short lifespan [12], low detection times are required to warn users before accessing phishing websites.

Login website detection. Unlike other methods, which are trained with homepage URLs as representatives of the legitimate class, our model was trained with legitimate login websites. This ensures the correct classification of those websites. Therefore, our approach can be applied to the real case scenario where users have to predict whether a login form page is legitimate or phishing

REFERENCES

[1] Statista. (2020). Adoption Rate of Emerging Technologies in Organizations Worldwide as of 2020. Accessed: Sep. 12, 2021. [Online]. Available: https://www.statista.com/statistics/661164/worldwide-cio-surveyoperati% onal-priorities/

[2] R. De', N. Pandey, and A. Pal, ``Impact of digital surge during COVID-19 pandemic: A viewpoint on research and practice,'' Int. J. Inf. Manage., vol. 55, Dec. 2020, Art. no. 102171.

[3] P. Patel, D. M. Sarno, J. E. Lewis, M. Shoss, M. B. Neider, and C. J. Bohil, ``Perceptual representation of spam and phishing emails,'' Appl. Cognit. Psychol., vol. 33, no. 6, pp. 1296_1304, Nov. 2019.

[4] J. A. Chaudhry, S. A. Chaudhry, and R. G. Rittenhouse, ``Phishing attacks and defenses,'' Int. J. Secur. Appl., vol. 10, no. 1, pp. 247_256, 2016.

[5] M. Hijji and G. Alam, ``A multivocal literature review on growing social engineering based cyber-attacks/threats during the COVID-19 pandemic: Challenges and prospective solutions,'' IEEE Access, vol. 9, pp. 7152_7169, 2021.

[6] A. Alzahrani, ``Coronavirus social engineering attacks: Issues and recommendations,'' Int. J. Adv. Comput. Sci. Appl., vol. 11, no. 5, pp. 154_161, 2020.

[7] Phishing Activity Trends Report 3Q, Anti-Phishing Working Group, International, 2017. Accessed: Sep. 12, 2021.