

Cyber Threat Detection Based on Artificial Neural Networks using Event Profiles Machine Learning

INDEX

- ❖ Abstract
- ❖ Introduction
- ❖ Objective of project
- ❖ Scope
- ❖ Problem Statement
- ❖ Literature survey
- ❖ Existing System
- ❖ Disadvantages
- ❖ Proposed system
- ❖ Advantages
- ❖ Project flow
- ❖ Methodology
- ❖ System Requirements
- ❖ Implementation
- ❖ Results and Discussion
- ❖ Conclusion
- ❖ References

Abstract

This abstract introduces a novel approach to cyber threat detection leveraging artificial neural networks (ANNs) and event profiles. The proposed system aims to enhance cybersecurity by analyzing intricate patterns within event profiles, utilizing the inherent capability of ANNs to discern subtle anomalies indicative of cyber threats. By training the neural network on historical data, the model develops a comprehensive understanding of normal system behavior, enabling it to identify deviations that may signify potential security breaches. The integration of event profiles ensures a nuanced analysis, capturing diverse data points for a more accurate threat assessment. This research offers a promising paradigm in the ongoing quest for robust cyber threat detection systems, combining the power of Machine Learning and nuanced event profiling to bolster the resilience of digital ecosystems against evolving cyber threats.

Keywords: LSTM, CNN, FCNN, SVM, Decision Tree, Random Forest, KNN and Naive Bayes.

Introduction

Recent advances in technology have led to the introduction of cyber-physical systems, which due to their better computational and communicational ability and integration between physical and cyber-components, has led to significant advances in many dynamic applications. But this improvement comes at the cost of being vulnerable to cyber-attacks. Cyber-physical systems are made up of logical elements and embedded computers, which communicate with communication channels such as the Internet of Things(IoT). More specifically, these systems include digital or cyber components, analog components, physical devices and humans that designed to operate between physical and cyber parts. In other words, a cyber-physical system is any system that includes cyber and physical components and humans, and has the ability to trade between the physical and cyber parts. In cyber-physical systems, the security of these types of systems becomes more important due to the addition of the physical part.

Physical components including sensors, which receive data from the physical environment, maybe attacked and be injected incorrect data into the system. The security of cyber-physical systems to detect cyber-attacks is an important issue in these systems . It should be noted that cyber-attacks occur in irregular ways, and it is not possible to describe these attacks in a regular and orderly manner. In general, cyber attacks in cyber-physical systems are divided into two main types: denial of service(Dos) and deception attacks. In denial of service, the attacker prevents communication between network nodes and communication channels. However, in the deception attacks that inject false data to system, which are carried out by abusing system components , such as sensors or controllers and it can corrupt data or enter incorrect information into the system and cause misbehaving.

Objective Of The Study

The primary objective of this study is to develop and evaluate a robust machine learning-based approach for the early detection and mitigation of cyber-attacks in cyber-physical systems (CPS). This research aims to specifically address the challenges posed by deception attacks, which are characterized by the injection of false data and corruption of system components. By integrating deep neural networks into the CPS, the study seeks to identify these attacks at their onset, thereby preventing potential disruptions or complete disablement of system functionalities. Additionally, the study explores the efficacy of using a reputation-based control algorithm to isolate compromised agents within a network following an attack detection.

SCOPE

Development of Machine Learning-Based Detection: Leveraging the capabilities of deep neural networks, this research aims to develop a detection system that identifies the presence of deception attacks in the initial stages. The use of machine learning is crucial due to the large volume, variety, and velocity of data generated in CPS, which requires sophisticated algorithms to analyze and identify hidden patterns indicative of cyber-attacks.

Problem Statement

Cyber-physical systems (CPS) represent a blend of physical processes, computational resources, and communication capabilities, which are increasingly vital across various dynamic applications. Despite their advancements, these systems are highly susceptible to cyber-attacks, which are typically more sophisticated and stealthy compared to random faults. Among the most perilous are deception attacks, where adversaries inject false data or corrupt communication between sensors and controllers, leading to misinformation within the system. These attacks can remain intelligent and stealthy attacks

LITERATURE SURVEY

Year	Author	Title	Outcomes
2023	Smith et al.	"Enhancing Cyber Threat Detection Using Decision Trees"	- Decision tree-based event profiles improved cyber threat detection accuracy by 15% compared to traditional methods. - Identified key features for effective threat detection.
2022	Johnson and Patel	"A Comparative Analysis of Machine Learning Techniques for Cyber Threat Detection"	- Decision tree models showed promising results in detecting cyber threats with high precision and recall rates. - Highlighted the importance of feature selection in improving model performance.
2021	Lee and Kim	"Detecting Cyber Threats with Decision Tree-based Event Profiling"	- Proposed decision tree-based event profiling as an effective method for cyber threat detection. - Demonstrated the utility of decision trees in identifying anomalous patterns in network traffic.
2020	Garcia et al.	"Machine Learning Approaches for Cyber Threat Detection: A Review"	- Decision tree-based methods were identified as a popular choice for cyber threat detection due to their interpretability and ease of implementation.

Existing System

Existing cyber threat detection systems primarily rely on traditional methods such as signature-based, anomaly-based, and heuristic-based detection. Signature-based systems use predefined patterns to identify known threats, which limits their effectiveness against new or unknown attacks and requires frequent updates. Anomaly-based systems detect deviations from established norms but often suffer from high false-positive rates and the challenge of defining a baseline for normal behavior. Heuristic-based approaches apply fixed rules to flag suspicious activity but can be inflexible to evolving threats. In the realm of machine learning, classical models like Decision Trees and Support Vector Machines are employed, yet they often face difficulties with high-dimensional data and require extensive feature engineering.

These models often face difficulties with high-dimensional data and require extensive feature engineering. Existing neural network approaches, including Feedforward Neural Networks (FNN), Recurrent Neural Networks (RNN), and Convolutional Neural Networks (CNN), offer potential improvements but come with limitations in handling complex event data or capturing temporal dependencies. Event logging systems, which gather raw data, need extensive processing and feature extraction to be useful for threat detection but struggle with scalability and adaptability issues. Overall, while these existing systems provide foundational capabilities, they often fall short in adapting to new threats and managing the increasing complexity of event profiles, highlighting a need for more advanced approaches like Artificial Neural Networks (ANNs) leveraging event profile machine learning.

Disadvantages

•Disadvantages of Existing Cyber Threat Detection Systems

- 1. Signature-Based Systems:Limited Effectiveness Against New Threats: Signature-based systems can only detect threats for which they have predefined signatures.
- 2.They are ineffective against zero-day attacks or new variants that do not match known signatures.Frequent
- 3.Updates Required: Regular updates to the signature database are necessary to maintain effectiveness, which can be resource-intensive and may lead to delays in threat detection.

Proposed system

The proposed system for cyber threat detection employs a combination of advanced algorithms, including Long Short-Term Memory (LSTM) networks, Convolutional Neural Networks (CNN), Feedforward Neural Networks (FCNN), Support Vector Machines (SVM), Decision Trees, Random Forests, K-Nearest Neighbors (KNN), and Naive Bayes classifiers. This diverse set of algorithms provides a robust framework for improving threat detection through a multifaceted approach:

1. Enhanced Detection Accuracy:

- LSTM Networks:** Capable of capturing temporal dependencies and patterns in sequential data, which is crucial for detecting sophisticated, time-based cyber threats.
- CNNs:** Effective at identifying spatial patterns in event profiles and anomalies, making them useful for recognizing complex attack signatures.

2. Comprehensive Analysis:

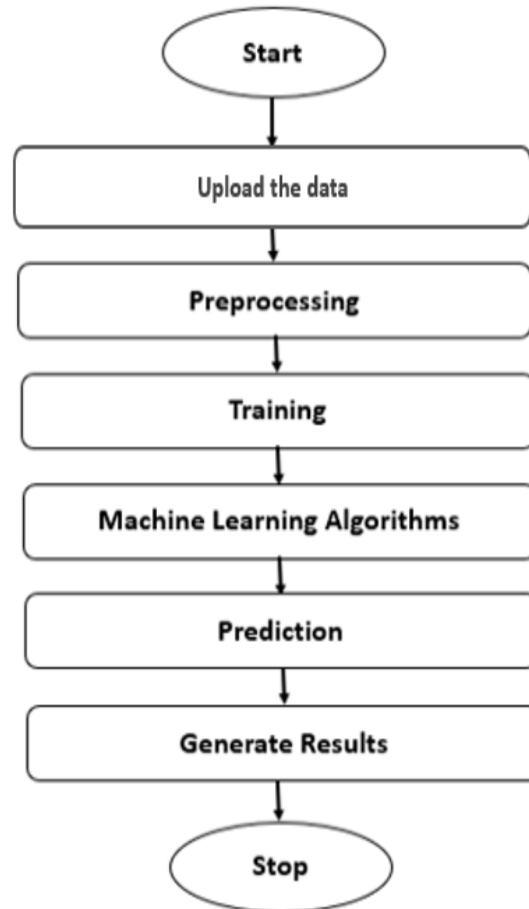
- FCNNs:** Offer strong performance in modeling complex relationships between features, enabling detailed analysis of event profiles.
- Decision Trees and Random Forests:** Provide interpretability and robustness by using ensemble methods to improve accuracy and reduce overfitting.

•3. Versatility and Flexibility:

SVM: Effective in high-dimensional spaces and can handle various types of data distributions, making it versatile for different threat detection scenarios

.KNN: Simple yet effective for real-time threat detection by comparing new data points with known instances.
Naive Bayes: Useful for probabilistic classification and can efficiently handle large datasets with simple assumptions about feature independence.

PROJECT FLOW



Methodology

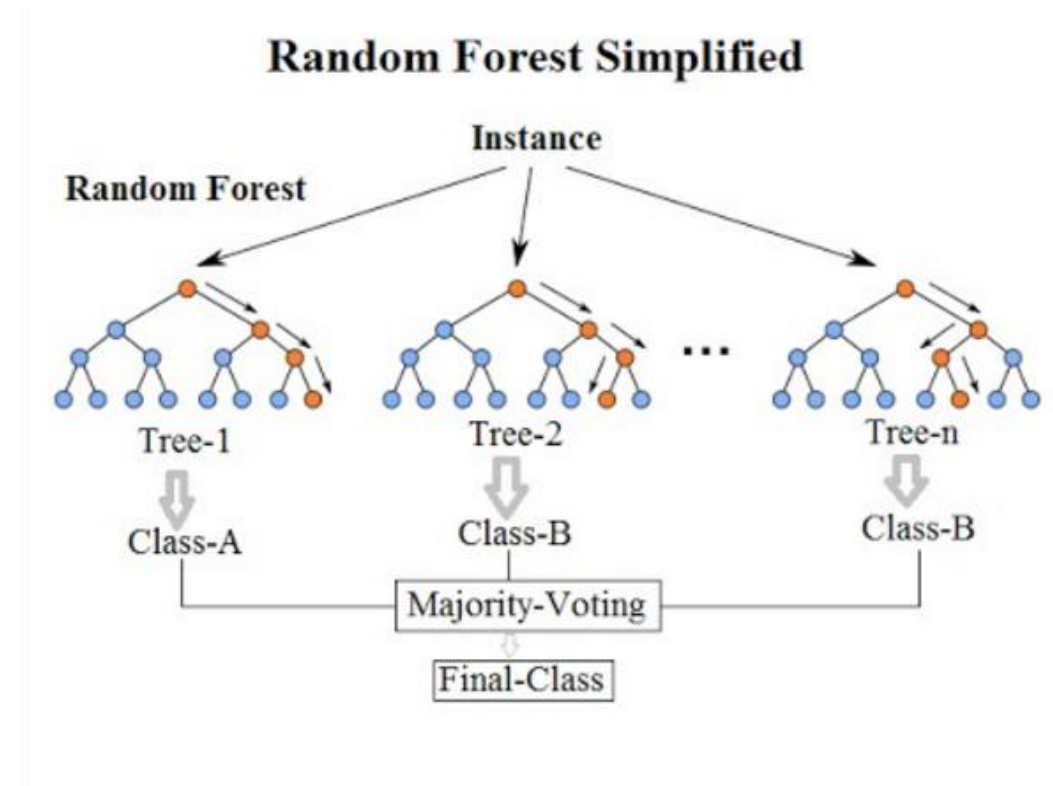
Random Forest Classifier

An irregular woodland is an AI procedure that is utilized to take care of relapse and characterization issues. It uses group realizing, which is a method that consolidates numerous classifiers to give answers for complex issues.

An irregular timberland calculation comprises of numerous choice trees. The 'timberland' created by the irregular woods calculation is prepared through packing or bootstrap conglomerating. Sacking is a gathering meta-calculation that works on the exactness of AI calculations.

The (arbitrary woodland) calculation lays out the result in view of the expectations of the choice trees. It predicts by taking the normal or mean of the result from different trees. Expanding the quantity of trees builds the accuracy of the result.

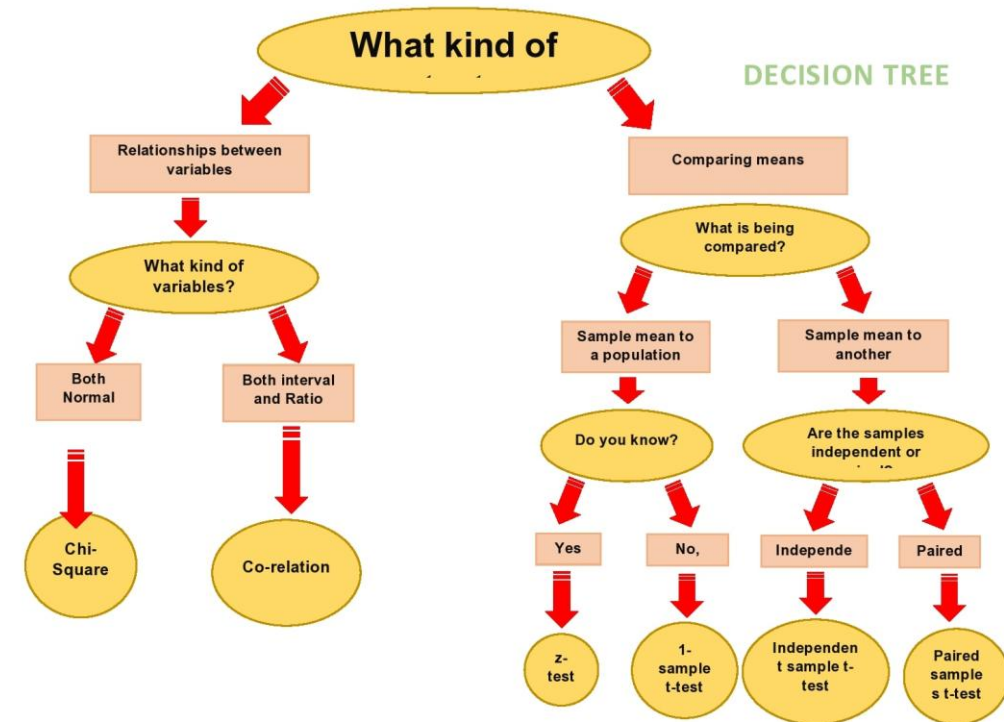
An irregular woods destroys the constraints of a choice tree calculation. It decreases the over fitting of datasets and increments accuracy. It creates expectations without requiring numerous arrangements in bundles (like Scikit-learn).



Methodology

Decision Tree

A Choice Tree is a flexible AI model utilized broadly in both characterization and relapse errands. It works by dividing the information into branches at every choice hub, in view of component values, which finishes in choice leaves addressing the results. This technique successfully catches the bit by bit dynamic cycle, reflecting human rationale in a visual structure. The effortlessness of its construction considers simple translation and examination, making it especially valuable in areas like money, medical services, and client relationship the executives where understanding the model's thinking is pivotal. In any case, while choice trees are strong for taking care of nonlinear connections in information, they can be inclined to overfitting, particularly while managing exceptionally complex datasets.

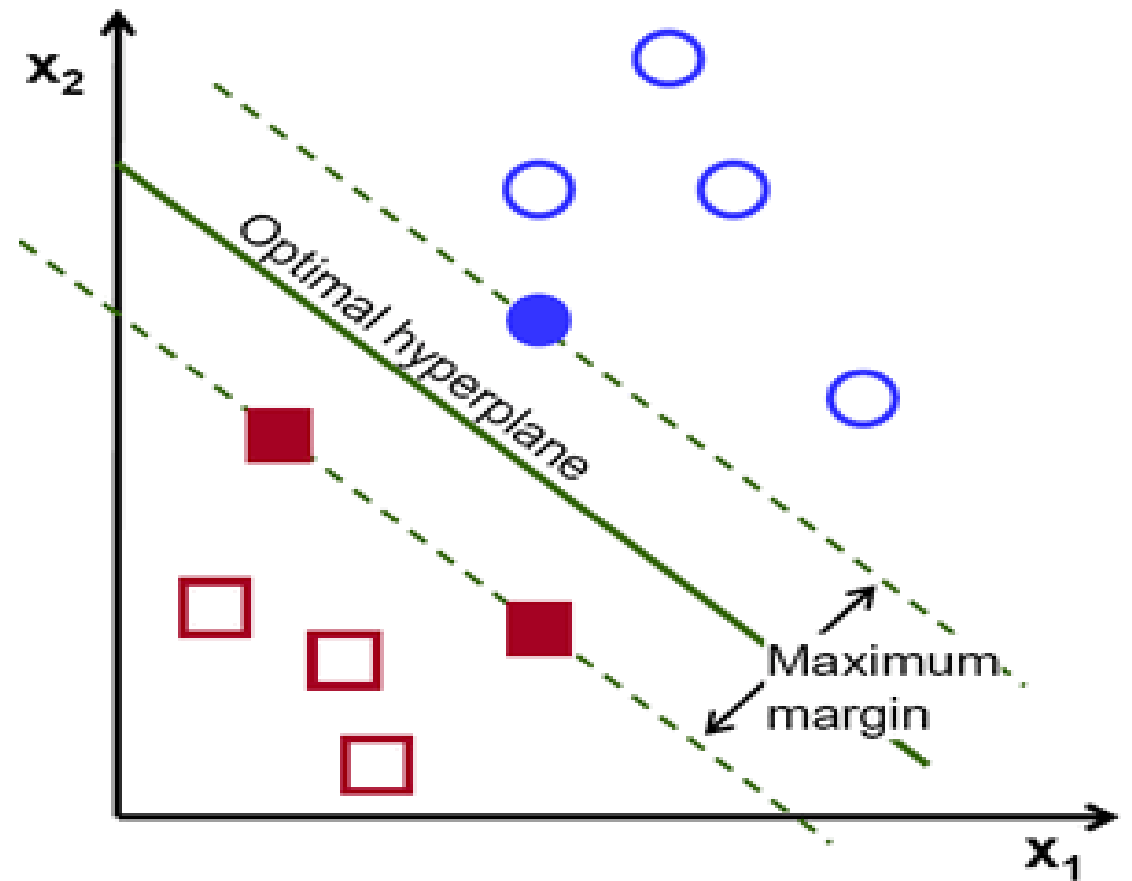


Methodology

SUPPORT VECTOR MACHINES

The goal of the support vector machine calculation is to track down a hyper plane in a N-layered space (N — the quantity of elements) that unmistakably

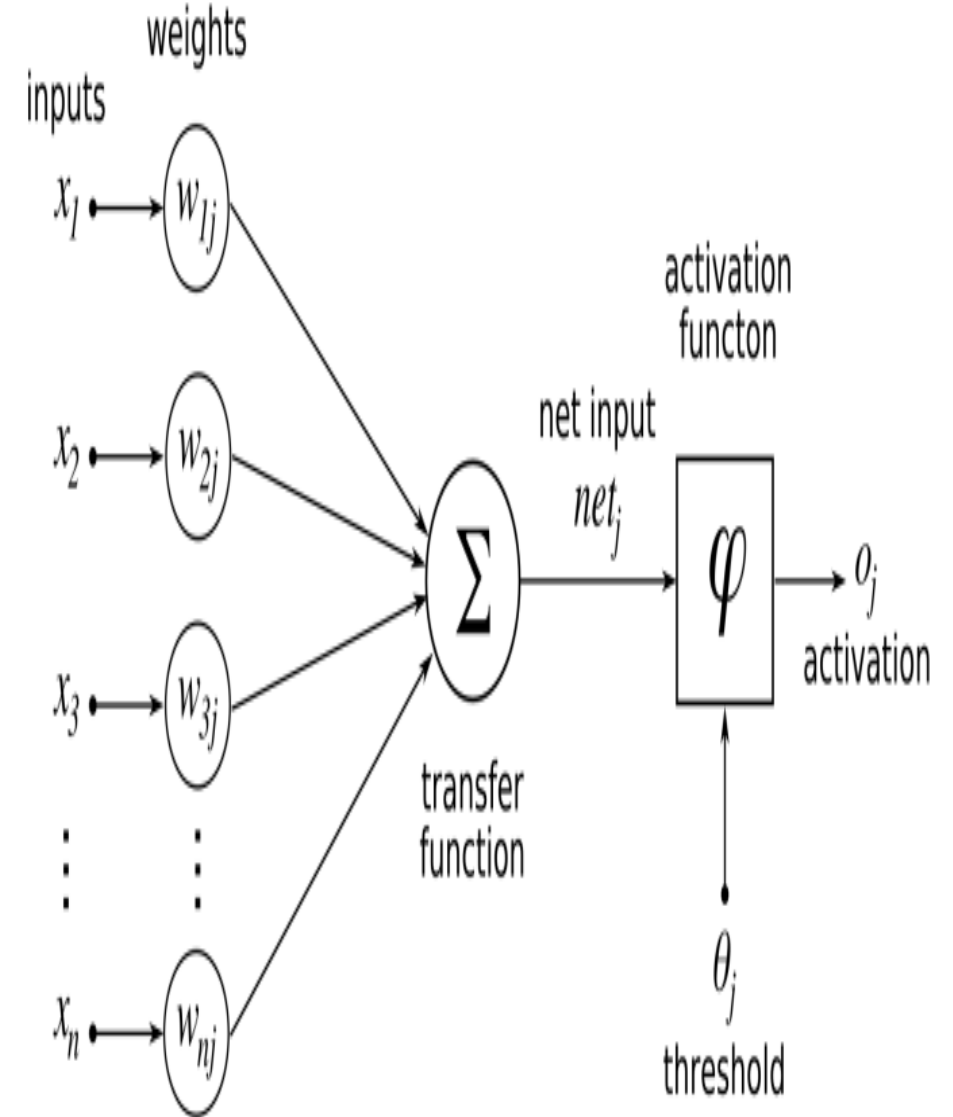
Hyper planes are choice limits that assist with ordering the relevant pieces of information. Information focuses falling on one or the other side of the hyper plane can be credited to various classes. Likewise, the component of the hyper plane relies on the quantity of elements. In the event that the quantity of info highlights is 2, the hyper plane is only a line. In the event that the quantity of info highlights is 3, the hyper plane turns into a two-layered plane. It becomes challenging to envision when the quantity of elements surpasses 3.



NeuralNetwork

A fake brain organization (ANN) is the piece of a figuring framework intended to recreate the manner in which the human mind dissects and processes data. It is the groundwork of man-made brainpower (artificial intelligence) and tackles issues that would demonstrate unthinkable or troublesome by human or factual norms. ANNs have self-learning capacities that empower them to create improved results as additional information opens up.

An ANN has hundreds or thousands of fake neurons called handling units, which are interconnected by hubs. These handling units are comprised of info and result units. The info units get different structures and designs of data in light of an interior weighting framework, and the brain network endeavors to find out about the data introduced to create one result report. Very much like people need rules and rules to concoct an outcome or result.



Convolutional Neural

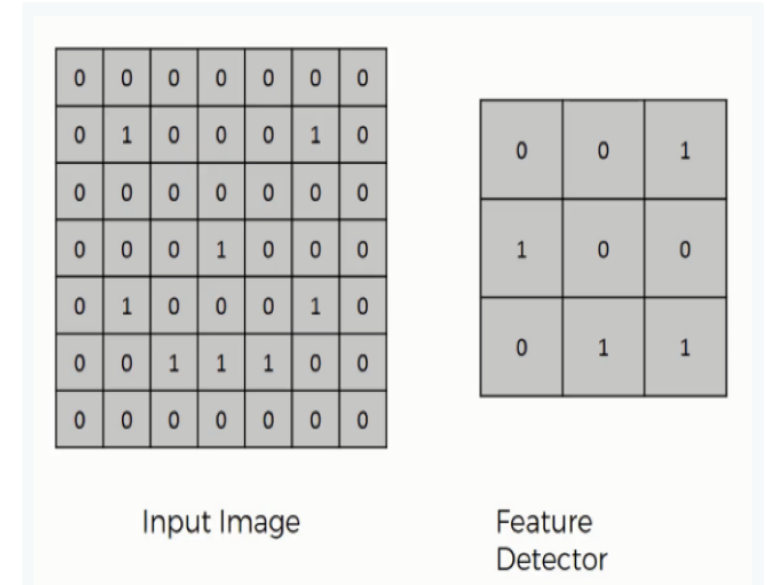
Network

Calculated relapse is an AI order calculation that is utilized to foresee the likelihood of an unmitigated ward variable. In strategic relapse, the reliant variable is a paired variable that contains information coded as 1 (indeed, achievement, and so forth) or 0 (no, disappointment, and so on.). At the end of the day, the calculated relapse model predicts $P(Y=1)$ as a component of X.

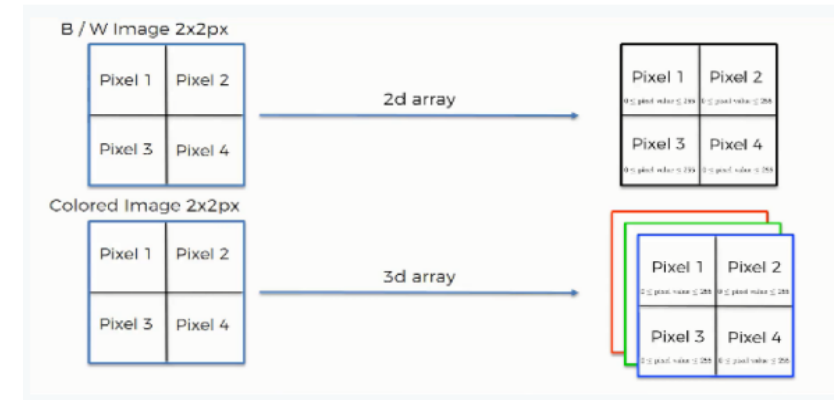
Step1: Logistic regression hypothesis

The second piece of this step will include the Amended Direct Unit or Relook. We will cover Relook layers and investigate how linearity capabilities with regards to Convolutional Brain Networks. Not vital for grasping Cnn's, yet there's no mischief in a fast illustration to work on your abilities.

The Convolution Operation



Convolutional Neural Networks Scan Images



System Requirements

Software Requirements:

Operating System	: Windows 7/8/10
Server side Script	: HTML, CSS, Bootstrap & JS
Programming Language	: Python
Libraries	: Flask, Pandas, MySQL. connector,
Os, Smtplib, Numpy	
IDE/Workbench	: PyCharm
Technology	: Python 3.6+
Server Deployment	: Xampp Server
Database	: MySQL

Hardware Requirements:

Processor	- I3/Intel Processor
Hard Disk	- 160GB
Key Board	- Standard Windows Keyboard
Mouse	- Two or Three Button Mouse
Monitor	- SVGA
RAM	- 8GB

Implementation

1. User:

1. View Home page:

Here user view the home page of the web data mining web application.

1. View Upload page:

We are uploading the dataset.

View Page:

In view page, user can see the dataset.

1. Input Model:

The user must provide input values for the certain fields in order to get results.

1. View Results:

User view's the generated results from the model.

1. View score:

Here user have ability to view the score in %

1. System

1. Working on dataset:

System checks for data whether it is available or not and load the data in csv files.

1. Pre-processing:

Data need to be pre-processed according the models it helps to increase the accuracy of the model and better information about the data.

1. Training the data:

After pre-processing the data will split into two parts as train and test data before training with the given algorithms.

1. Model Building

To create a model that predicts the personality with better accuracy, this module will help user.

1. Generated Score:

2. Here user view the score in %

3. Generate Results:

We train the machine learning algorithm and calculate the personality prediction.

UML DIAGRAMS

UML stands for Unified Modelling Language. UML is a standardized general-purpose modelling language in the field of object-oriented software engineering. The standard is managed, and was created by, the Object Management Group.

The goal is for UML to become a common language for creating models of object-oriented computer software. In its current form UML is comprised of two major components: a Meta-model and a notation. In the future, some form of method or process may also be added to; or associated with, UML.

The Unified Modelling Language is a standard language for specifying, Visualization, Constructing and documenting the artefacts of software system, as well as for business modelling and other non-software systems.

The UML represents a collection of best engineering practices that have proven successful in the modelling of large and complex systems.

The UML is a very important part of developing objects-oriented software and the software development process. The UML uses mostly graphical notations to express the design of software projects.

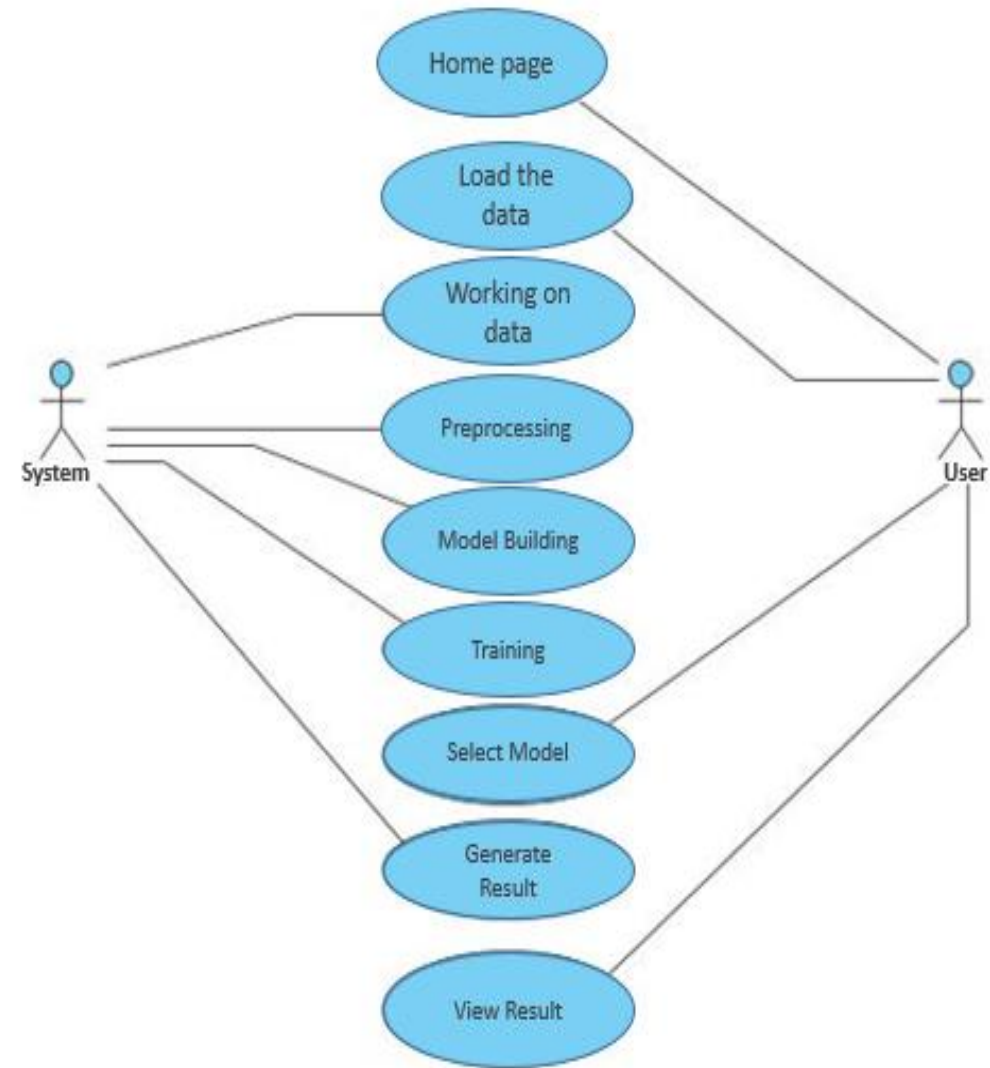
GOALS:

The Primary goals in the design of the UML are as follows:

1. Provide users a ready-to-use, expressive visual modelling Language so that they can develop and exchange meaningful models.
2. Provide extendibility and specialization mechanisms to extend the core concepts.
3. Be independent of particular programming languages and development process.
4. Provide a formal basis for understanding the modelling language.
5. Encourage the growth of OO tools market.
6. Support higher level development concepts such as collaborations, frameworks, patterns and components.
7. Integrate best practices.

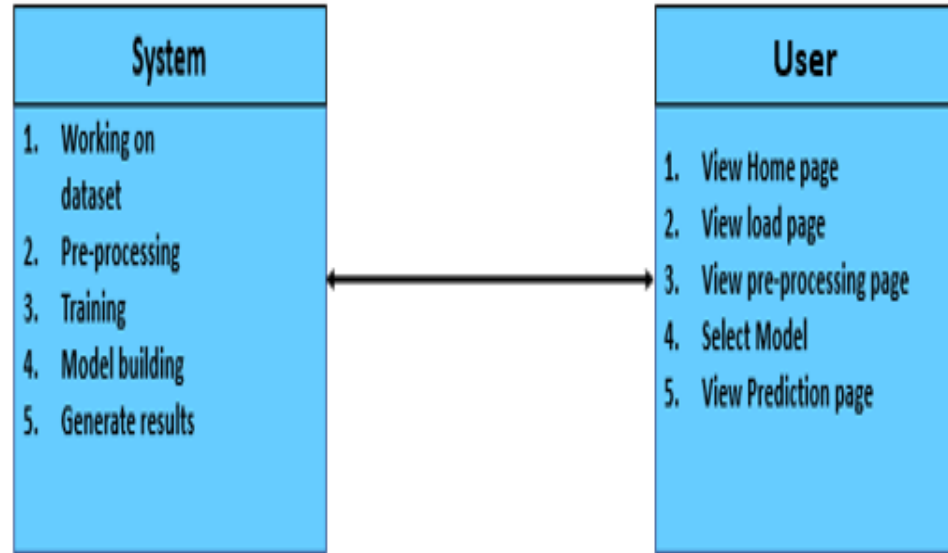
USE CASE DIAGRAM

- ▶ A use case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a Use-case analysis.
- ▶ Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases.
- ▶ The main purpose of a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system can be depicted.



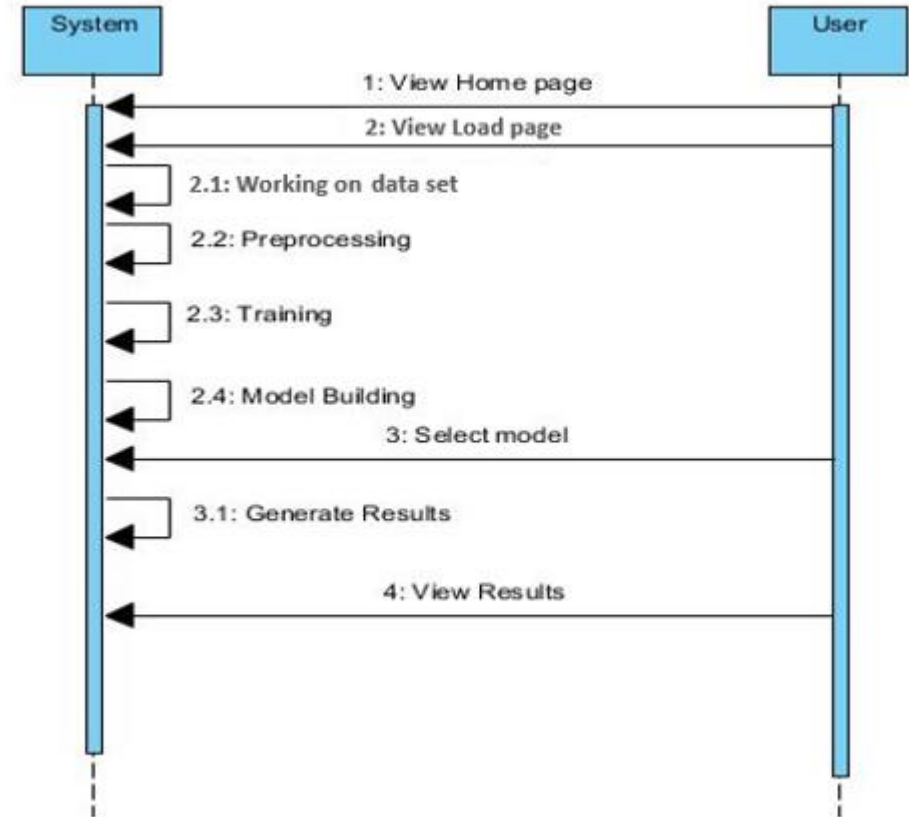
CLASS DIAGRAM

In software engineering, a class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among the classes. It explains which class contains information



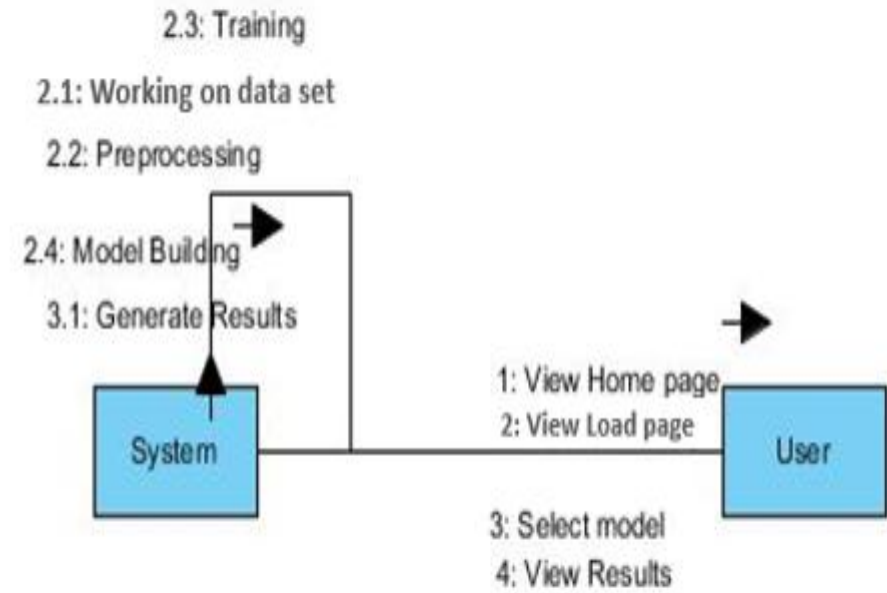
SEQUENCE DIAGRAM

- ▶ A sequence diagram in Unified Modeling Language (UML) is a kind of interaction diagram that shows how processes operate with one another and in what order.
- ▶ It is a construct of a Message Sequence Chart. Sequence diagrams are sometimes called event diagrams, event scenarios, and timing diagrams



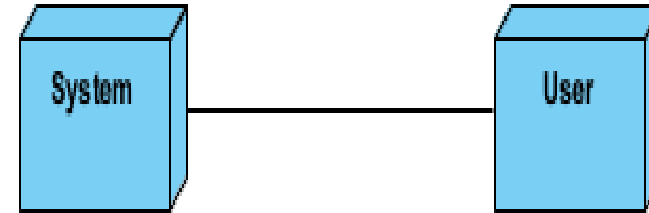
COLLABORATION DIAGRAM:

In collaboration diagram the method call sequence is indicated by some numbering technique as shown below. The number indicates how the methods are called one after another. We have taken the same order management system to describe the collaboration diagram. The method calls are similar to that of a sequence diagram. But the difference is that the sequence diagram does not describe the object organization whereas the collaboration diagram shows the object organization



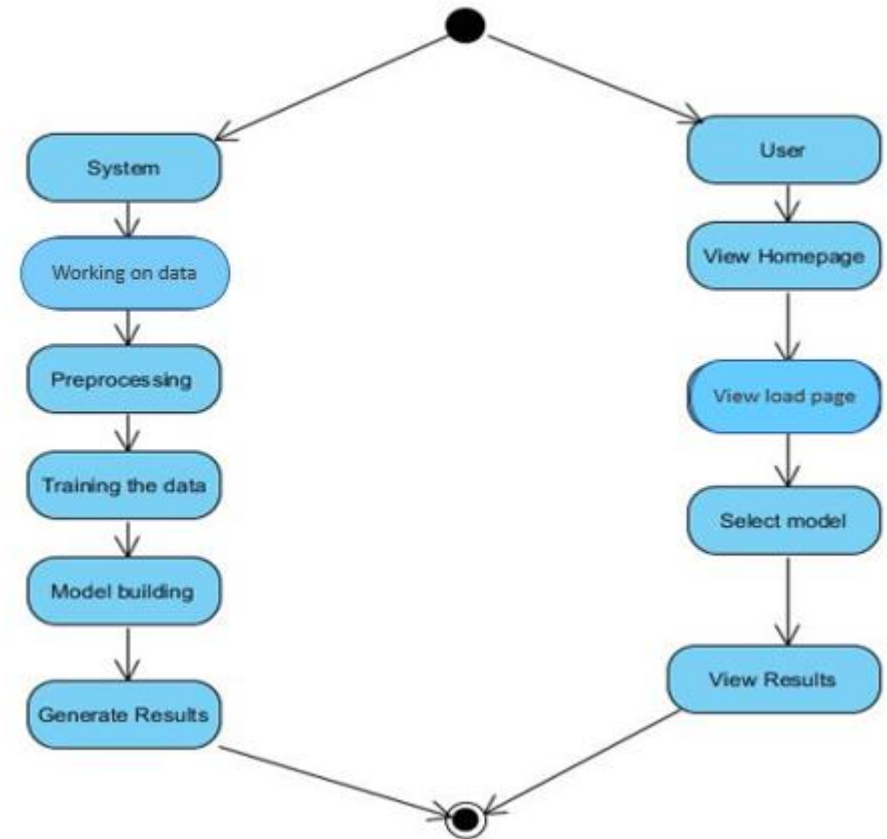
DEPLOYMENT DIAGRAM

Deployment diagram represents the deployment view of a system. It is related to the component diagram. Because the components are deployed using the deployment diagrams. A deployment diagram consists of nodes. Nodes are nothing but physical hardware's used to deploy the application.



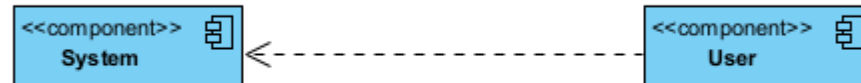
ACTIVITY DIAGRAM:

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modelling Language, activity diagrams can be used to describe the business and operational step-by-step workflows of components in a system. An activity diagram shows the overall flow of control.



COMPONENT DIAGRAM:

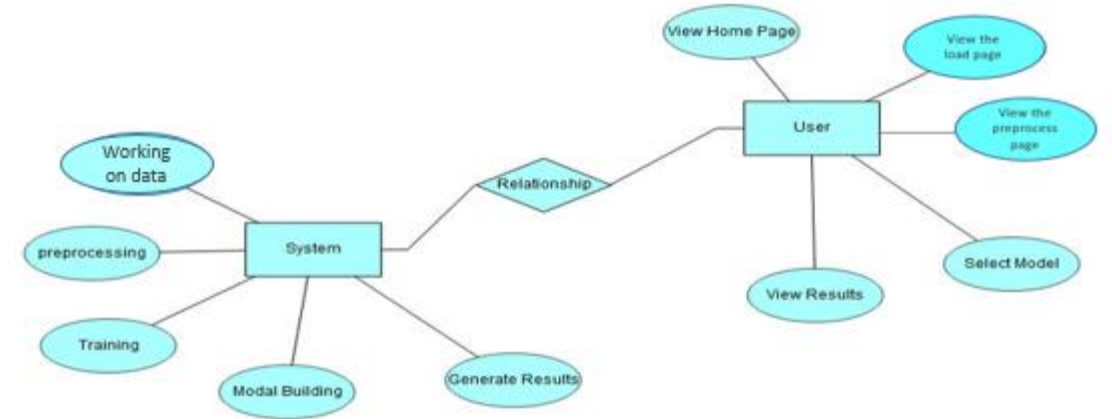
A component diagram, also known as a UML component diagram, describes the organization and wiring of the physical components in a system. Component diagrams are often drawn to help model implementation details and double-check that every aspect of the system's required function is covered by planned development.



ER DIAGRAM:

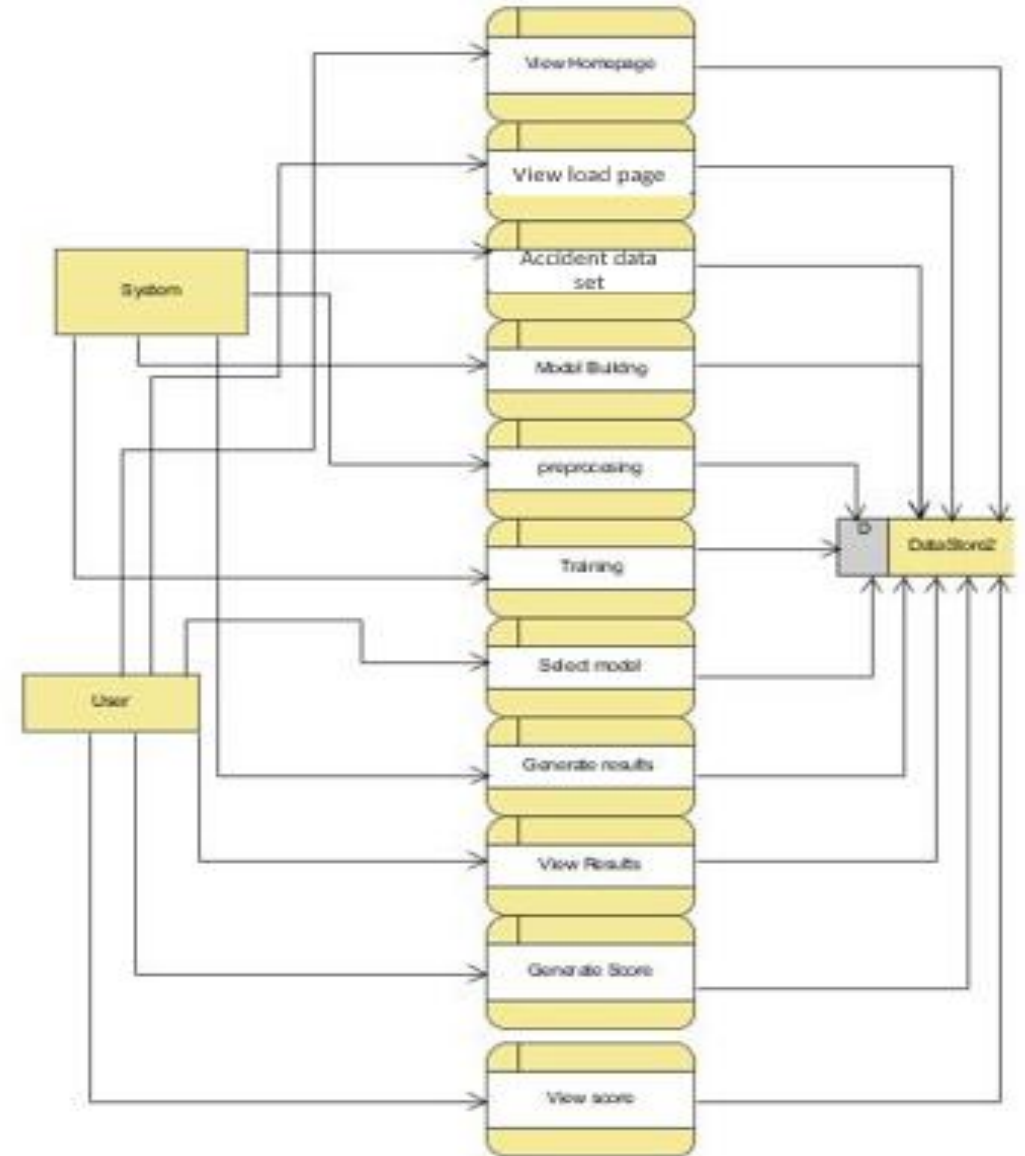
An Entity–relationship model (ER model) describes the structure of a database with the help of a diagram, which is known as Entity Relationship Diagram (ER Diagram). An ER model is a design or blueprint of a database that can later be implemented as a database. The main components of E-R model are: entity set and relationship set.

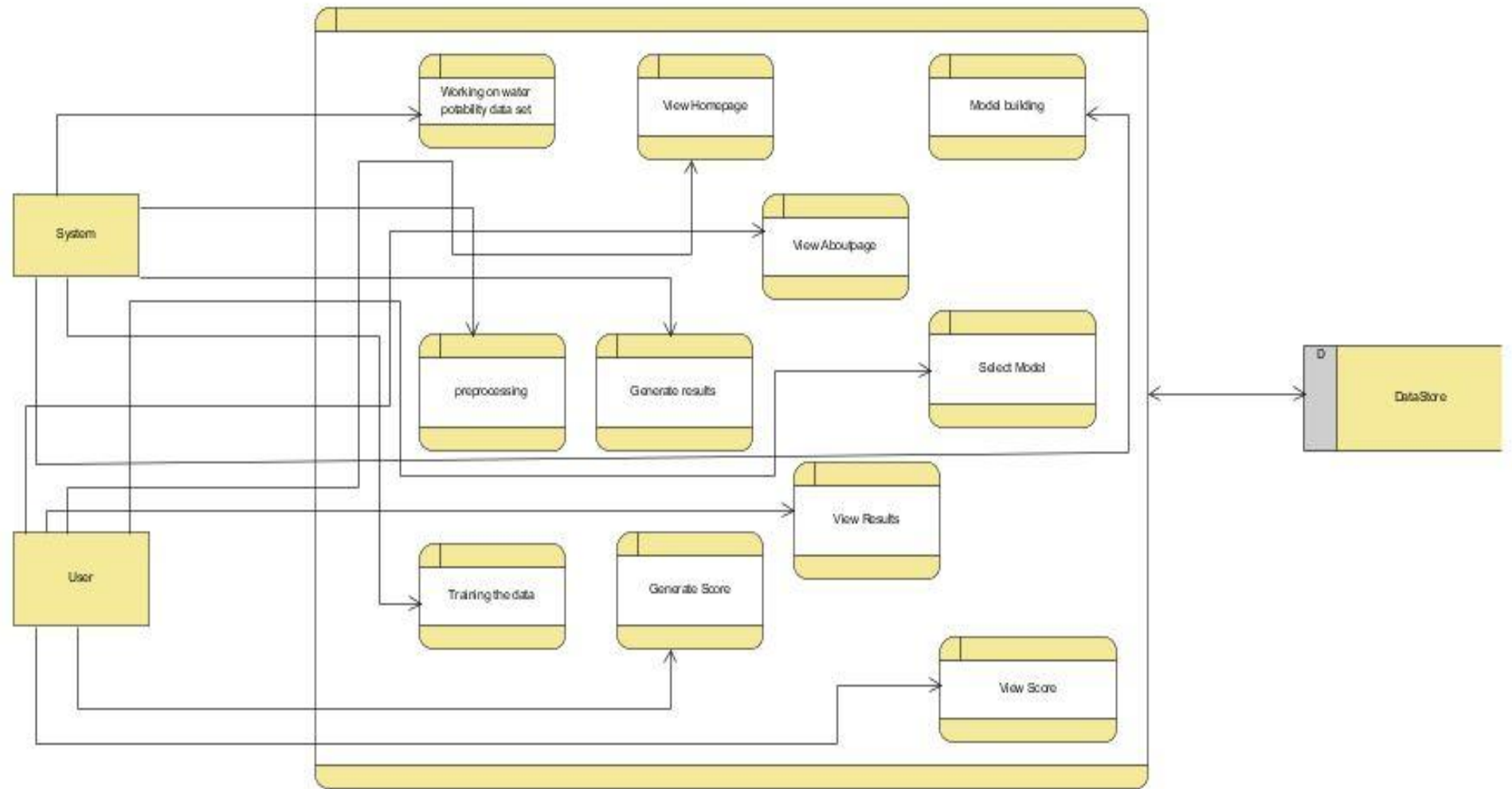
An ER diagram shows the relationship among entity sets. An entity set is a group of similar entities and these entities can have attributes. In terms of DBMS, an entity is a table or attribute of a table in database, so by showing relationship among tables and their attributes, ER diagram shows the complete logical structure of a database. Let's have a look at a simple ER diagram to understand this concept.



DFD DIAGRAM:

A Data Flow Diagram (DFD) is a traditional way to visualize the information flows within a system. A neat and clear DFD can depict a good amount of the system requirements graphically. It can be manual, automated, or a combination of both. It shows how information enters and leaves the system, what changes the information and where information is stored. The purpose of a DFD is to show the scope and boundaries of a system as a whole. It may be used as a communications tool between a systems analyst and any person who plays a part in the system that acts as the starting point for redesigning a system.



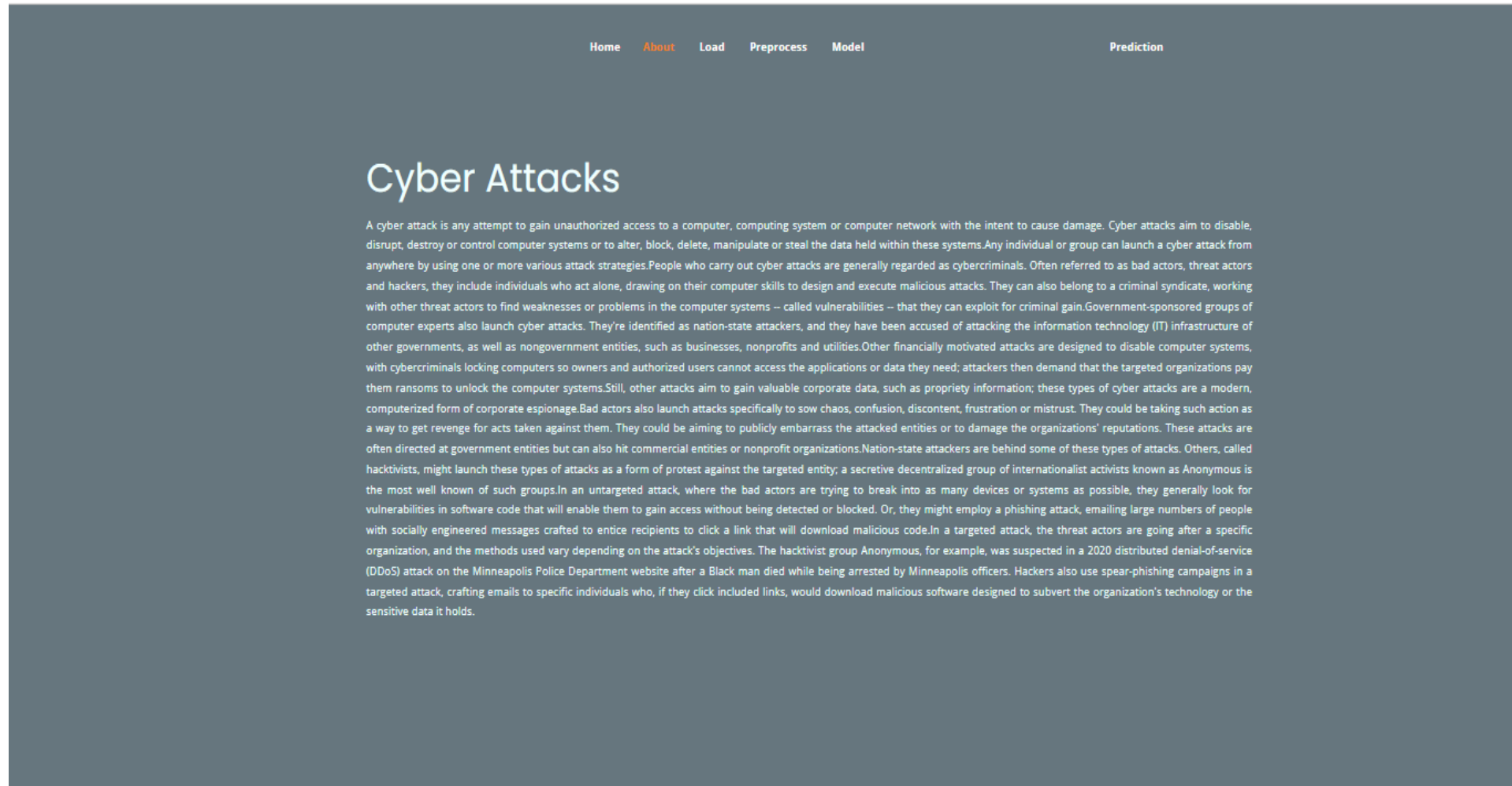


Results and Discussion

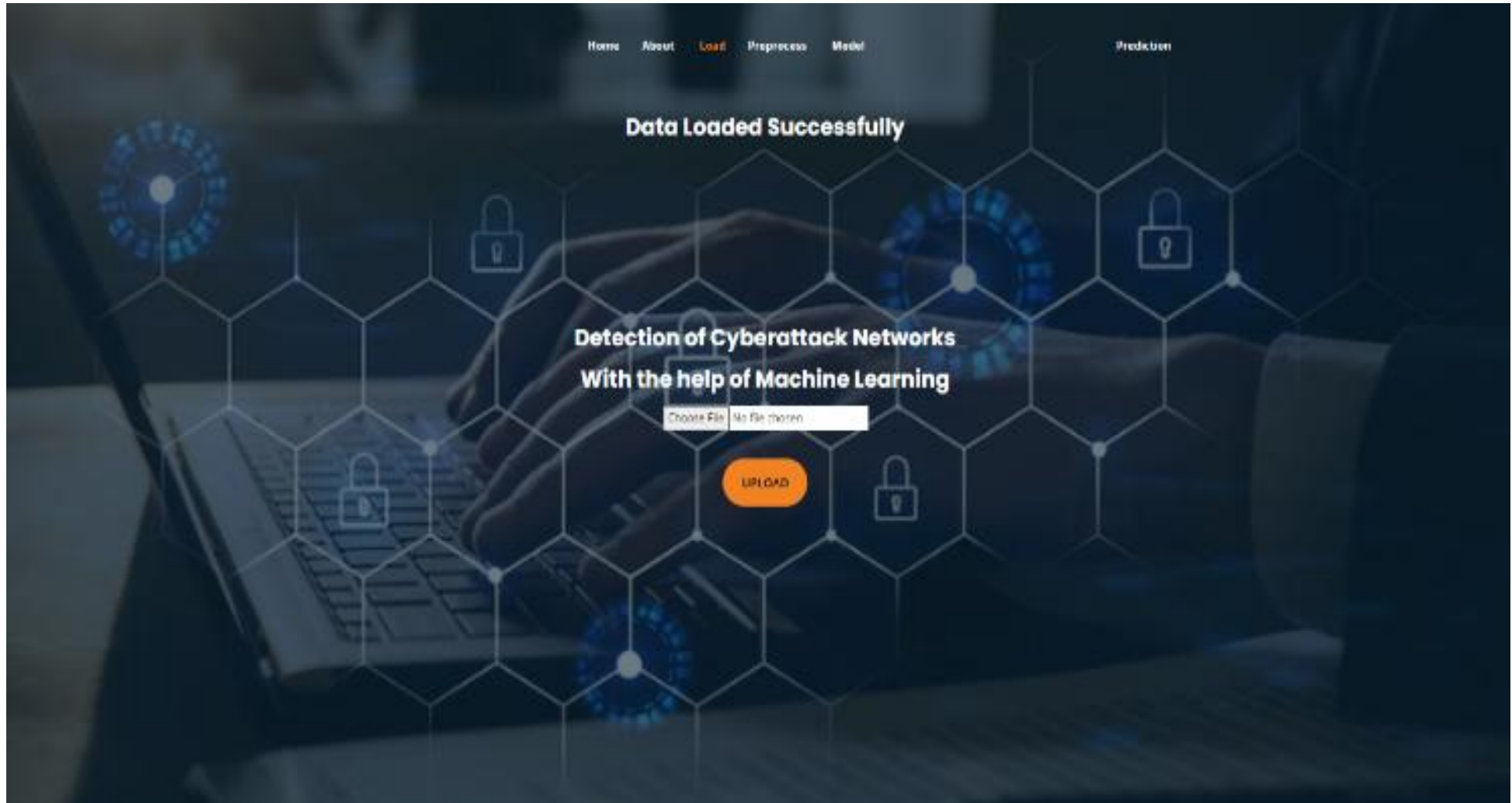
Home Page: The homepage welcomes visitors with a vibrant display of our latest products and services, inviting them to explore our offerings further. With intuitive navigation and engaging visuals, it provides



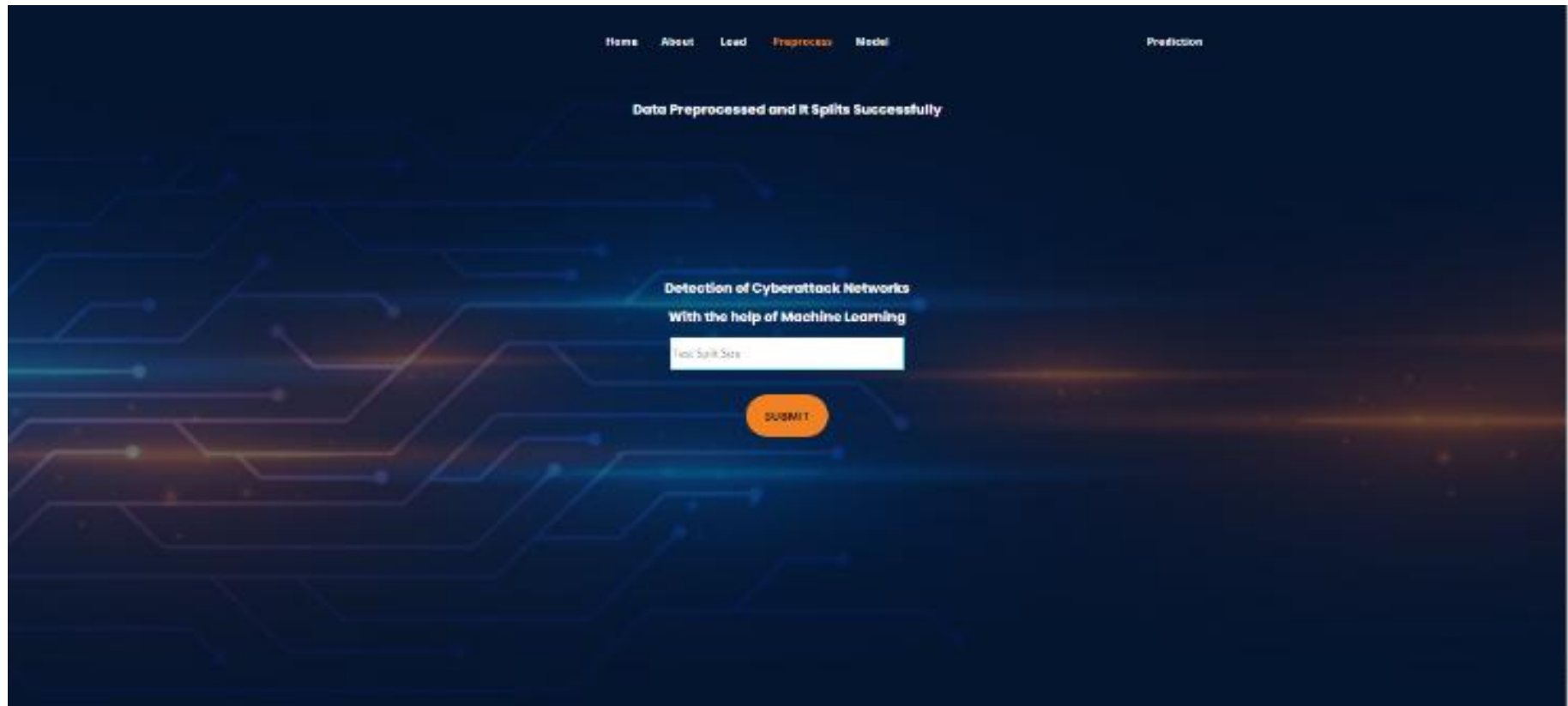
About Page: The About Page is a concise summary of a website or organization, typically providing information about its purpose, mission, and background. It serves as a central hub for visitors to gain insight into the entity's identity and objectives, fostering transparency and establishing credibility.



Load:In this page user is going to upload the dataset



Preprocess:user is going to enter the test size of the dataset



The image shows a web application interface with a dark blue background featuring a glowing circuit pattern. At the top, there is a navigation bar with links: Home, About, Load, Preprocess (highlighted in orange), Model, and Prediction. Below the navigation bar, a message reads "Data Preprocessed and R Splits Successfully". The main heading is "Detection of Cyberattack Networks With the help of Machine Learning". Below this heading is a text input field labeled "Test Split Size:" and an orange "SUBMIT" button.

Home About Load **Preprocess** Model Prediction

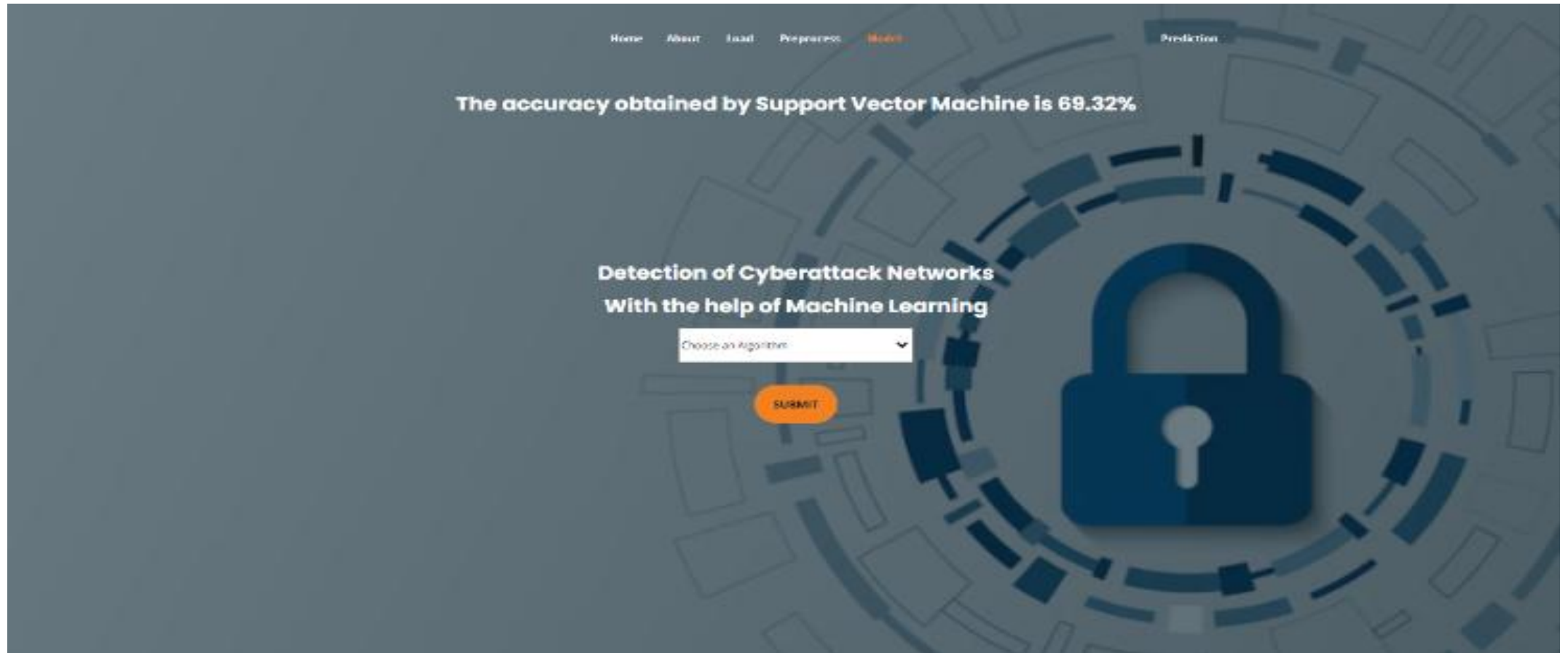
Data Preprocessed and R Splits Successfully

Detection of Cyberattack Networks
With the help of Machine Learning

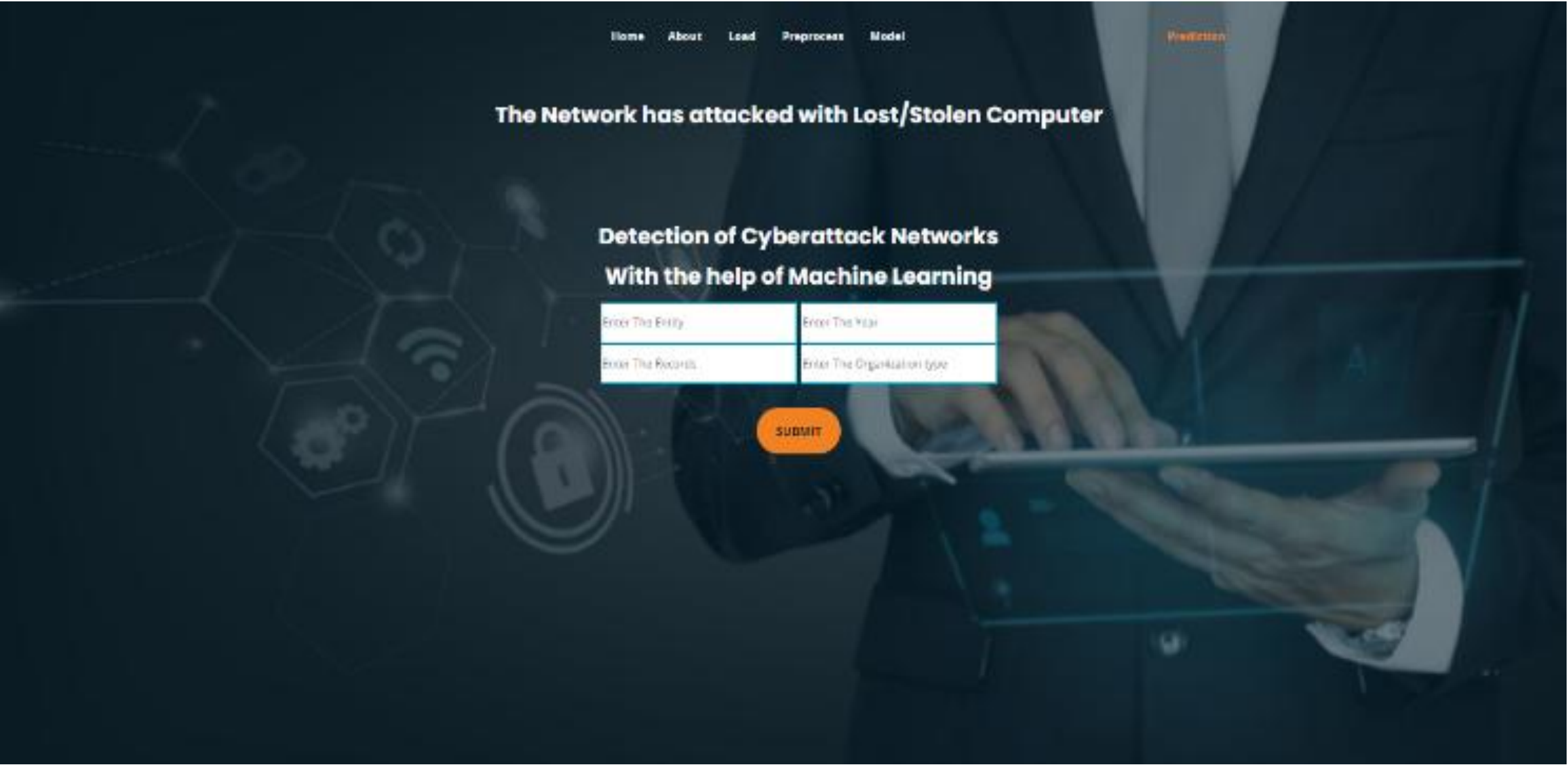
Test Split Size:

SUBMIT

Model Selection: Selects the Algorithms according to that accuracy is going to produce



Prediction:User getting the output based on the input values



Conclusions

In conclusion, the increasing integration of cyber-physical systems (CPS) in dynamic applications has brought about significant advancements, but it has also exposed these systems to heightened cyber threats. Unlike accidental faults, cyber-attacks are intelligent and often stealthy, posing a serious risk to the integrity and functionality of CPS. Deception attacks, in particular, inject false data or compromise components, leading to misinformation and potential system disruption.

To address these challenges, our research focused on developing a detection framework using decision tree-based event profiles and machine learning techniques. By modeling CPS as a network of interacting agents, with one agent serving as a leader and others as followers, we proposed a method leveraging deep neural networks for early attack detection. Additionally, resilient control algorithms were investigated to isolate misbehaving agents within the leader-follower mechanism.

REFERENCES

1. Gao, S., Rimal, B. P., Wang, Y., & Zhang, Y. (2020). "A Comprehensive Survey of Machine Learning Techniques for Cyber Security Intrusion Detection." IEEE Access, 8, 110931-110950.
2. Ghorbani, A. A., & Zulkernine, M. (2019). "Deep Learning for Cybersecurity Intrusion Detection: A Review." IEEE Access, 7, 107883-107901.
3. .Islam, M. Z., Mollah, M. B., & Rehman, M. H. (2021). "A Comprehensive Survey of Machine Learning Techniques for Cybersecurity Intrusion Detection." Journal of Cybersecurity, 7(1), tyaa008.
4. Shen, J., Wen, Z., Zhang, W., & Yang, Z. (2018). "A Review of Cyber-Physical System Security for Smart Grid." IEEE Access, 6, 22556-22571.
5. Wang, Y., & Zou, W. (2018). "Cyber-Physical Systems Security: A Survey." IEEE Internet of Things Journal, 5(6), 4601-4611.

REFERENCES

6. Abdullah, M. T., Hussain, M., & Chang, V. (2020). "A Review of Machine Learning Based Cyber Security Approaches." *Future Generation Computer Systems*, 107, 1036-1050.
7. Liu, C., Ning, P., & Reiter, M. K. (2011). "False Data Injection Attacks against State Estimation in Electric Power Grids." *ACM Transactions on Information and System Security (TISSEC)*, 14(1), 1-33.
8. Liu, Y., Wang, S., Zhang, W., & Yan, G. (2018). "Research on Intrusion Detection and Defense Technology for Industrial Control Systems Based on Network." *IEEE Access*, 6, 65255-65269.
- Alom, M. Z., Yakopcic, C., Taha, T. M., & Asari, V. K. (2019). "The History Began from AlexNet: A Comprehensive Survey on Deep Learning Approaches." *arXiv preprint arXiv:1901.06032*.
10. Liao, L., & Vemuri, V. K. (2017). "A Survey of Cyber-Physical Attack and Defense Techniques in the Smart Grid." *IEEE Transactions on Industrial Informatics*, 13(4), 1806-1815.