**Enhancing Rice Production Prediction in Indonesia Using Advanced Machine Learning Models**

**ABSTRACT:**

This study delves into the application of machine learning techniques for predicting rice production in Indonesia, a country where rice is not just a staple food but also a key component of the agricultural sector. Utilizing data from 2018 to 2023, sourced from the Central Bureau of Statistics of Indonesia and the Meteorology, Climatology, and Geophysics Agency of Indonesia, this research presents a comprehensive approach to agricultural forecasting. The study begins with an Exploratory Data Analysis (EDA) to understand the variability and distribution of variables such as harvested area, production, rainfall, humidity, and temperature. Significant regional disparities in rice production are identified, highlighting the complexity of agricultural forecasting in Indonesia. Five machine learning models—Random Forest, Gradient Boosting, Decision Tree, Support Vector Machine, and XGBRegressor—are trained and tested. The XGBRegressor model stands out for its superior performance, demonstrating its high predictive accuracy and reliability. Hyperparameter tuning using the GridSearchCV technique was conducted on all five models, resulting in performance improvements across the board. This research not only underscores the effectiveness of machine learning in improving rice production predictions in Indonesia but also sets the stage for future research. It highlights the potential of advanced analytical techniques in enhancing agricultural productivity and decision-making, paving the way for further explorations into more sophisticated models and a broader range of data, ultimately contributing to the resilience and sustainability of Indonesia’s agricultural sector.

**TABLE OF CONTENTS**

|  |  |  |
| --- | --- | --- |
| **CHAPTER NO.** | **TITLE** | **PAGE NO.** |
|  | **ABSTRACT**  **LIST OF FIGURES**  **LIST OF SYMBOLS** | i  v  vii |
| 1. | **CHAPTER 1 : INTRODUCTION**   * 1. GENERAL   2. SCOPE OF THE PROJECT   3. OBJECTIVE   1.4 EXISTING SYSTEM  1.4.1 EXISTINGSYSTEM DISADVANTAGES  1.5 LITERATURE SURVEY  1.6 PROPOSED SYSTEM  1.6.1 PROPOSED SYSTEM ADVANTAGES |  |
| 2. | **CHAPTER 2 :PROJECT DESCRIPTION**  2.1 GENERAL  2.2 METHODOLOGIES  2.2.1 MODULES NAME  2.2.2 MODULES EXPLANATION  2.3 TECHNIQUE OR ALGORITHM |  |
| 3. | **CHAPTER 3 : REQUIREMENTS**  3.1 General  3.2 Hardware REQUIREMENTS  3.3 Software REQUIREMENTS |  |
| 4. | **CHAPTER 4 :SYSTEM DESIGN**  **4.1 general**  **4.2 uml diagrams**  4.2.1 USE CASE DIAGRAM  4.2.2 CLASS DIAGRAM  4.2.3 OBJECT DIAGRAM  4.2.4 STATE DIAGRAM  4.2.5 activity diagram  4.2.6 SEQUENCE DIAGRAM  4.2.7 COLLABORATION DIAGRAM  4.2.8 COMPONENT DIAGRAM  4.2.9 DATA FLOW DIAGRAM  4.2.10 DEPLOYMENT DIAGRAM  4.2.11 SYSTEM ARCHITECTURE |  |
| 5. | **CHAPTER 5 : DEVELOPMENT TOOLS**  5.1 general  5.2 History of Python  5.3 Importance of Python  5.4 Features of Python  5.5 Libraries used in python |  |
| 6. | **CHAPTER 6 :IMPLEMENTATION**  6.1 GENERAL  6.2 IMPLEMENTATION |  |

|  |  |  |
| --- | --- | --- |
| 7. | **CHAPTER 7 :SNAPSHOTS**  7.1 GENERAL  7.2 VARIOUS SNAPSHOTS |  |
| 8. | **CHAPTER 8 :SOFTWARE TESTING**  8.1 GENERAL  8.2 DEVELOPING METHODOLOGIES  8.3 TYPES OF TESTING |  |
| 9. | **CHAPTER 9 :**  **FUTURE ENHANCEMENT**  9.1 FUTURE ENHANCEMENTS |  |
| **10** | **CHAPTER 10 :**  10.1CONCLUSION  10.2 REFERENCES |  |

**LIST OF FIGURES**

|  |  |  |
| --- | --- | --- |
| **FIGURE NO** | **NAME OF THE FIGURE** | **PAGE NO.** |
|  |  |  |
| 4.1 | Use case Diagram |  |
| 4.2 | Class diagram |  |
| 4.3 | Object diagram |  |
| 4.4 | State Diagram |  |
| 4.5 | Activity Diagram |  |
| 4.6 | Sequence diagram |  |
| 4.7 | Collaboration diagram |  |
| 4.8 | Component Diagram |  |
| 4.9 | Data flow diagram |  |
| 4.10 | Deployment Diagram |  |
| 4.11 | Architecture Diagram |  |

**LIST OF SYSMBOLS**

|  |  |  |  |
| --- | --- | --- | --- |
| **S.NO** | **NOTATION**  **NAME** | **NOTATION** | **DESCRIPTION** |
| 1. | Class | *Class Name*  *-attribute*  *-attribute*  *+operation*  *+operation*  *+operation*  *+ public*  *-private*  *# protected* | Represents a collection of similar entities grouped together. |
| 2. | Association | name  Class B  Class A  Class A  Class B | Associations represents static relationships between classes. Roles represents the way the two classes see each other. |
| 3. | Actor | Class A  Class A  Class B  Class B | It aggregates several classes into a single classes. |
| 4. | Aggregation | Interaction between the system and external environment |

|  |  |  |  |
| --- | --- | --- | --- |
| 5. | Relation  (uses) | uses | Used for additional process communication. |
| 6. | Relation  (extends) | extends | Extends relationship is used when one use case is similar to another use case but does a bit more. |
| 7. | Communication |  | Communication between various use cases. |
| 8. | State | State | State of the processes. |
| 9. | Initial State |  | Initial state of the object |
| 10. | Final state |  | Final state of the object |
| 11. | Control flow |  | Represents various control flow between the states. |
| 12. | Decision box |  | Represents decision making process from a constraint |
| 13. | Use case |  | Interact ion between the system and external environment. |

|  |  |  |  |
| --- | --- | --- | --- |
| 14. | Component |  | Represents physical modules which are a collection of components. |
| 15. | Node |  | Represents physical modules which are a collection of components. |
| 16. | Data Process/State |  | A circle in DFD represents a state or process which has been triggered due to some event or action. |
| 17. | External entity |  | Represents external entities such as keyboard, sensors, etc. |
| 18. | Transition |  | Represents communication that occurs between processes. |
| 19. | Object Lifeline |  | Represents the vertical dimensions that the object communications. |
| 20. | Message | Message | Represents the message exchanged. |

**CHAPTER-1**

**INTRODUCTION**

1

1.Introduction :

Rice is not only a staple food for the vast majority of Indonesia's population but also a crucial component of the country's agricultural sector, which significantly contributes to the national economy. As one of the world's largest producers and consumers of rice, Indonesia's agricultural landscape is deeply intertwined with its socio-economic stability. Given the growing demands of a burgeoning population and the increasing challenges posed by climate change, accurately predicting rice production has become a critical priority. The ability to forecast rice yields can greatly influence policy decisions, resource allocation, and strategic planning in agriculture, ultimately enhancing food security and economic resilience.

In recent years, the rapid advancement of technology, particularly in the fields of data analytics and machine learning, has opened up new avenues for improving agricultural forecasting. Traditional methods of predicting crop yields, which often rely on basic statistical models and expert judgment, are limited in their ability to handle the complex, non-linear relationships inherent in agricultural data. These conventional models may not effectively capture the intricate interactions between multiple variables, such as climatic conditions, soil characteristics, and regional farming practices. As a result, there is a pressing need for more sophisticated predictive models that can better adapt to the dynamic nature of agriculture.

This study explores the application of advanced machine learning techniques to predict rice production in Indonesia, utilizing a dataset spanning from 2018 to 2023. The data, sourced from the Central Bureau of Statistics of Indonesia and the Meteorology, Climatology, and Geophysics Agency of Indonesia, includes a wide range of variables that influence rice yields, such as harvested area, production volume, rainfall patterns, humidity levels, and temperature fluctuations. The research begins with an Exploratory Data Analysis (EDA) to uncover the variability and distribution of these variables, providing valuable insights into the factors that affect rice production across different regions. The EDA also reveals significant disparities in rice yields among various regions of Indonesia, underscoring the complexity of forecasting agricultural outputs in such a diverse landscape.

**1.2 SCOPE OF THE PROJECT**

The scope of this project focuses on addressing the challenges of insurance fraud detection within the automobile insurance sector. It aims to enhance the accuracy and efficiency of fraud detection models by utilizing advanced machine learning techniques, specifically addressing the class imbalance problem and missing data issues. The project involves working with real-life datasets, applying the AdaBoost Classifier, and evaluating the model's performance in comparison to existing systems. Additionally, the study explores how these enhancements can lead to better prediction accuracy, reduced overfitting, and more reliable fraud detection systems.

**1.3 OBJECTIVE**

The objective of this project is to develop an advanced and efficient insurance fraud detection system tailored for the automobile insurance industry. The project focuses on addressing key challenges such as class imbalance, where fraudulent claims are underrepresented compared to legitimate claims, and missing data, which often affects the model’s accuracy. By leveraging machine learning techniques, particularly the AdaBoost Classifier, the aim is to enhance prediction accuracy and reduce overfitting, ensuring the model generalizes better on unseen data. This project also seeks to provide a framework for improving the overall effectiveness of fraud detection systems, leading to more reliable identification of fraudulent claims and aiding in better decision-making and pricing strategies for insurance companies. Through this, the project aims to contribute to reducing financial losses for insurers and improving the overall integrity of the insurance system.

**1.4 EXISTING SYSTEM:**

The existing system for predicting rice production in Indonesia primarily relies on traditional statistical methods and basic machine learning models. The forecasting models typically focus on variables such as harvested area, production, rainfall, humidity, and temperature, utilizing data from government agencies such as the Central Bureau of Statistics of Indonesia and the Meteorology, Climatology, and Geophysics Agency of Indonesia. However, the accuracy of these predictions is often limited due to the complexities of agricultural data, including regional disparities in rice production, environmental variability, and data inconsistencies. Exploratory Data Analysis (EDA) helps in understanding the distribution and relationships between these variables, but the current system lacks the advanced machine learning techniques that can capture intricate patterns in large datasets. Models like Random Forest and Gradient Boosting are used, but their predictive power is often constrained by improper tuning and insufficient data coverage, leading to suboptimal performance.

**1.4.1 EXISTINGSYSTEM DISADVANTAGES:**

* Limited Predictive Accuracy
* Suboptimal Hyperparameter Tuning
* Data Inconsistencies
* Inability to Handle Complex Patterns
* Limited Scalability

**1.5 LITERATURE SURVEY**

**Title:** Deep Learning Enables Instant and Versatile Estimation of Rice Yield Using Ground-Based RGB Images

**Authors:** Yu Tanaka, Tomoya Watanabe, Keisuke Katsura, Yasuhiro Tsujimoto, Toshiyuki Takai, Takashi Sonam Tashi Tanaka, Kensuke Kawamura, Hiroki Saito, Koki Homma and Kazuki Saito

**Year:** 2023

**Description:** Rice (*Oryza sativa* L.) is one of the most important cereals, which provides 20% of the world’s food energy. However, its productivity is poorly assessed especially in the global South. Here, we provide a first study to perform a deep-learning-based approach for instantaneously estimating rice yield using red-green-blue images. During ripening stage and at harvest, over 22,000 digital images were captured vertically downward over the rice canopy from a distance of 0.8 to 0.9 m at 4,820 harvesting plots having the yield of 0.1 to 16.1 t·ha−1 across 6 countries in Africa and Japan. A convolutional neural network applied to these data at harvest predicted 68% variation in yield with a relative root mean square error of 0.22. The developed model successfully detected genotypic difference and impact of agronomic interventions on yield in the independent dataset. The model also demonstrated robustness against the images acquired at different shooting angles up to 30° from right angle, diverse light environments, and shooting date during late ripening stage. Even when the resolution of images was reduced (from 0.2 to 3.2 cm·pixel−1 of ground sampling distance), the model could predict 57% variation in yield, implying that this approach can be scaled by the use of unmanned aerial vehicles. Our work offers low-cost, hands-on, and rapid approach for high-throughput phenotyping and can lead to impact assessment of productivity-enhancing interventions, detection of fields where these are needed to sustainably increase crop production, and yield forecast at several weeks before harvesting.

**Title:** Sample-free automated mapping of double-season rice in China using Sentinel-1 SAR imagery

**Author:** Xi Zhang, Ruoque Shen, Xiaolin Zhu, Baihong Pan

**Year:** 2023.

**Description**: Timely and accurately mapping the spatial distribution of rice is of great significance for estimating crop yield, ensuring food security and freshwater resources, and studying climate change. Double-season rice is a dominant rice planting system in China, but it is challenging to map it from remote sensing data due to its complex temporal profiles that requires high-frequency observations. Methods: We used an automated rice mapping method based on the Synthetic Aperture Radar (SAR)-based Rice Mapping Index (SPRI), that requires no samples to identify double-season rice. We used the Sentinel-1 SAR time series data to capture the growth of rice from transplanting to maturity in 2018, and calculated the SPRI of each pixel by adaptive parameters using cloud-free Sentinel-2 imagery. We extensively evaluated the methods performance at pixel and regional scales. Results and discussion: The results showed that even without any training samples, SPRI was able to provide satisfactory classification results, with the average overall accuracy of early and late rice in the main producing provinces of 84.38% and 84.43%, respectively. The estimated area of double-season rice showed a good agreement with county-level agricultural census data. Our results showed that the SPRI method can be used to automatically map the distribution of rice with high accuracy at large scales.

**Title:** The Correlation between Rainfall, Temperature, Relative Humidity, and Rice Field Productivity in Aceh Besar

**Author:** Sofia Chairani

**Year:** 2022.

**Description:** Various factors could affect rice field productivity, such as climate, management practices, and soil properties. Aceh Besar had experienced long drought, higher temperature, shifted seasons, and the decrease yield of rice productivity due to climate change. This research aimed to analyze the correlation between climate variables and rice field productivity, such as rainfall and mean, minimum, and maximum temperatures, relative humidity in Aceh Besar District. The monthly climate data and the rice field productivity data were employed for 10 (2011-2020) and 8 (2011-2018) consecutive years, respectively. The correlation between the climate variables were calculated using Pearson coefficient correlation. The results showed that rainfall and maximum temperature were positively correlated, as well as rainfall and relative humidity. In contrary, rainfall and mean temperature, rainfall and minimum temperature, rainfall and rice field productivity were negatively correlated. The latest indicating that rainfall did not impact the rice field productivity in Aceh Besar. It was quite contradictive to the reality in the field that significantly experiencing the long drought, higher temperature, shifted seasons and the decrease yield of rice field productivity. This was due to the lack of climate data employed that required longer period preferably 30 to 50 years which was not available.

**Title:**  Rice and Wheat Yield Prediction in India Using Decision Tree and Random Forest  
**Author:**  Dr. B M Sagar, Dr.N K Cauvery, Dr.Padmashree T , Dr.R. Rajkumar

**Year:** 2022

**Description**: One of the main sources of revenue and growth in Indian economy is from agriculture. It is often a gamble for the farmers to obtain a decent yield, considering the unpredictable environmental conditions. This paper deals with the prediction of the yield of rice and wheat using machine learning algorithms using the annual crop yield production and the annual rainfall in the different districts of India. In this paper, a popular prediction model is developed using algorithms such as decision tree and random forest to predict the yield of most widely grown crops in India like rice and wheat. The features used were the area of production, rainfall, season and state. The season and the state were one hot encoded features. Mean square error was used to measure the loss. The dataset was prepared by combining the crop production in the various states and the rainfall dataset in the respective states

**Title:** Paddy yield prediction based on 2D images of rice panicles using regression techniques

**Author**: Pankaj, Brajesh Kumar, P. K. Bharti, Vibhor K. Vishnoi

**Year:** 2024**.**

**Description:** Crop yield predictions are important for crop monitoring and agronomic management. The traditional methods for yield predictions are complicated and resource consuming. With the availability of affordable handheld imaging and computing devices, the image processing-based yield prediction methods are gaining popularity. In this work, RGB images of rice panicles are captured using DSLR camera with simple background and processed to determine the panicle area in terms of number of pixels. A machine learning-based model is developed to make predictions for rice yield. The model is trained to make predictions on unseen data. Various machine learning-based regression algorithms including decision tree, random forest, support vector machine, and convolution neural network are tested. The experiments are performed on a publically available dataset from China as well as on a self-acquired dataset in India. The results have shown that image processing and machine learning-based methods can make yield predictions satisfactorily as evident from the coefficient of determination (R2R2) that ranges 0.80–0.97 for different cultivars. The prediction error is determined in terms of root mean square error (RMSE) and mean absolute error (MAE). RMSE for different methods lies between 0.14 and 0.40, whereas MAE varies from 0.11 to 0.30. Among the tested algorithms, linear regression achieved the best precision with R22 = 0.97, RMSE = 0.14, and MAE = 0.11.

**1.6 PROPOSED SYSTEM**

The proposed system leverages advanced machine learning techniques to improve rice production forecasting in Indonesia, focusing on the application of XGBRegressor. Unlike traditional models, XGBRegressor is designed to handle large and complex datasets efficiently, allowing for better prediction accuracy. The system will begin with a comprehensive Exploratory Data Analysis (EDA) to understand key variables such as harvested area, production, and environmental factors. The XGBRegressor model, known for its high accuracy and flexibility, will be trained on data from 2018 to 2023, incorporating data from both the Central Bureau of Statistics and the Meteorology, Climatology, and Geophysics Agency of Indonesia. Hyperparameter tuning using GridSearchCV will be applied to optimize the model's performance, resulting in a more robust and accurate prediction system.The use of a voting ensemble leverages the strengths of each classifier—Random Forest's ability to handle noisy data, XGBoost's gradient boosting efficiency, and SVC's precision in class boundaries. By integrating these models, our approach achieves superior fault classification performance, making it ideal for predictive maintenance applications. The results demonstrate that our ensemble method outperforms standalone classifiers, providing a reliable and efficient solution for early detection of bearing faults in rotating machinery.

**1.6.1 PROPOSED SYSTEM ADVANTAGES:**

* Enhanced Fraud Detection
* Effective Missing Data Handling
* Optimized Model Performance
* Reduced Overfitting
* Increased Accuracy in Predictions

**CHAPTER 2**

**PROJECT DESCRIPTION**

**2.1 GENERAL:**

Rice is not only a staple food for the vast majority of Indonesia's population but also a crucial component of the country's agricultural sector, which significantly contributes to the national economy. As one of the world's largest producers and consumers of rice, Indonesia's agricultural landscape is deeply intertwined with its socio-economic stability. Given the growing demands of a burgeoning population and the increasing challenges posed by climate change, accurately predicting rice production has become a critical priority. The ability to forecast rice yields can greatly influence policy decisions, resource allocation, and strategic planning in agriculture, ultimately enhancing food security and economic resilience.

In recent years, the rapid advancement of technology, particularly in the fields of data analytics and machine learning, has opened up new avenues for improving agricultural forecasting. Traditional methods of predicting crop yields, which often rely on basic statistical models and expert judgment, are limited in their ability to handle the complex, non-linear relationships inherent in agricultural data. These conventional models may not effectively capture the intricate interactions between multiple variables, such as climatic conditions, soil characteristics, and regional farming practices. As a result, there is a pressing need for more sophisticated predictive models that can better adapt to the dynamic nature of agriculture.

This study explores the application of advanced machine learning techniques to predict rice production in Indonesia, utilizing a dataset spanning from 2018 to 2023. The data, sourced from the Central Bureau of Statistics of Indonesia and the Meteorology, Climatology, and Geophysics Agency of Indonesia, includes a wide range of variables that influence rice yields, such as harvested area, production volume, rainfall patterns, humidity levels, and temperature fluctuations. The research begins with an Exploratory Data Analysis (EDA) to uncover the variability and distribution of these variables, providing valuable insights into the factors that affect rice production across different regions.

**2.2 METHODOLOGIES**

**2.2.1MODULES NAME:**

**Modules Name:**

* Dataset Collection
* Analytical Review of Data
* Pre-Modeling Data Handling
* Model Utilization
* Algorithm Training
* Model Validation
* Result Estimation
  + 1. **MODULES EXPLANATION:**

1. **Dataset Collection:**

The dataset used in this study was collected from reputable sources, specifically the Central Bureau of Statistics of Indonesia and the Meteorology, Climatology, and Geophysics Agency of Indonesia. It includes comprehensive data spanning from 2018 to 2023, covering critical variables that influence rice production, such as harvested area, production volumes, rainfall, temperature, and humidity. This diverse dataset provides a robust foundation for building predictive models by capturing the complexities of Indonesia's agricultural landscape. By leveraging such detailed and extensive data, this study aims to enhance the accuracy of forecasting rice production across different regions of the country.

1. **Analytical Review of Data:**

Before model development, an **Exploratory Data Analysis (EDA)** was conducted to understand the underlying patterns, variability, and distribution of the collected data. The EDA process involved visualizing and summarizing key features, identifying significant regional disparities in rice production, and exploring the relationships between climatic factors and crop yields. This step is crucial for uncovering hidden insights, detecting outliers, and understanding the impact of each variable on rice production.

**3) Pre-Modeling Data Handling:**

The pre-modeling phase focused on **data preprocessing** to ensure the dataset was ready for training. This involved handling missing values, encoding categorical variables, normalizing or scaling numerical features, and addressing any data imbalances. Techniques such as **one-hot encoding** and **feature scaling** were applied to optimize the dataset for machine learning algorithms. The goal of this stage was to enhance the quality of the input data, thereby improving the performance of the predictive models. Proper data handling is essential for maximizing model accuracy and ensuring robust predictions.

1. **Model Utilization:**

The study experimented with five machine learning models: Random Forest, Gradient Boosting, Decision Tree, Support Vector Machine (SVM), and XGBRegressor. Each of these models was chosen for its strengths in handling various aspects of agricultural data, from tree-based methods that are effective in capturing non-linear relationships to support vector machines that can handle high-dimensional spaces. The models were initially evaluated based on their performance metrics to determine their suitability for rice production forecasting. The inclusion of multiple models allows for a comparative analysis to identify the best-performing algorithm for this specific use case.

1. **Algorithm Training:**

In the proposed system, the focus was on leveraging **XGBRegressor** for algorithm training due to its superior performance in previous tests. XGBRegressor, an implementation of the Extreme Gradient Boosting technique, was selected for its advanced capabilities in boosting, feature selection, and regularization, which are essential for handling the complex nature of agricultural data. The training process involved splitting the dataset into training and testing sets, followed by applying **GridSearchCV** for hyperparameter tuning. This method ensures that the model is optimized to achieve the best possible performance by systematically exploring a range of hyperparameters. The XGBRegressor was trained to minimize errors and maximize predictive accuracy, showcasing its potential in delivering precise forecasts for rice production.

1. **Model Validation:**

Once trained, the models underwent a **rigorous validation process** to assess their predictive accuracy and generalization capabilities. The performance of each model was evaluated using key metrics such as **Mean Squared Error (MSE)** and **R-squared (R²)** values. The validation process helps determine how well the models perform on unseen data, ensuring that the predictions are reliable and not overfitted to the training dataset. The XGBRegressor model demonstrated superior validation results, outperforming other algorithms with the lowest MSE and highest R² scores, indicating its robustness and reliability in forecasting agricultural outcomes.

1. **Result Estimation:**

After model validation, the **final results** were estimated to gauge the effectiveness of the proposed XGBRegressor model. The model's predictions were compared against actual production values to evaluate its forecasting accuracy. The results showed a significant improvement over traditional models, highlighting the advantages of using advanced machine learning techniques for agricultural forecasting. By providing more accurate predictions, the proposed system can support better decision-making in crop management, resource allocation, and strategic planning, thereby contributing to the sustainability and resilience of Indonesia’s rice production sector.

**2.3 TECHNIQUE USED OR ALGORITHM USED**

**2.3.1** **EXISTING TECHNIQUE:**

The existing algorithm for predicting rice production in Indonesia employs traditional machine learning models, such as Random Forest, Gradient Boosting, Decision Trees, and Support Vector Machines (SVM). These models are trained using historical data on harvested area, production, rainfall, temperature, and humidity, sourced from government agencies like the Central Bureau of Statistics of Indonesia and the Meteorology, Climatology, and Geophysics Agency of Indonesia. While these models provide valuable insights, their performance is often limited by issues such as insufficient data preprocessing, suboptimal hyperparameter tuning, and the challenges posed by regional disparities in production. These factors contribute to less accurate predictions, particularly in the face of complex, variable agricultural environments. The current system uses GridSearchCV for hyperparameter tuning, but the models still struggle to capture all the intricate patterns in the data, resulting in moderate forecasting accuracy.

**2.3.2 PROPOSED TECHNIQUE USED OR ALGORITHM USED:**

The proposed algorithm utilizes XGBRegressor (Extreme Gradient Boosting Regressor), an advanced machine learning technique known for its efficiency and superior performance in handling large and complex datasets. XGBRegressor excels in capturing complex relationships within the data, making it ideal for agricultural forecasting, where there are non-linear interactions between variables like climate and production. The model will be trained using the same data from 2018 to 2023, with a focus on harvested area, production, and weather-related factors. Hyperparameter tuning will be performed using GridSearchCV to fine-tune the model for optimal performance. XGBRegressor's ability to handle missing values, deal with multicollinearity, and prevent overfitting through regularization makes it a more robust and reliable choice compared to traditional algorithms.

The introduction of XGBRegressor is expected to significantly improve rice production forecasting in Indonesia. Its advanced capabilities in boosting, feature selection, and regularization allow the model to better handle the complexities of agricultural data, leading to more accurate predictions.

.

**CHAPTER 3**

**REQUIREMENTS ENGINEERING**

**3.1 GENERAL**

We can see from the results that on each database, the error rates are very low due to the discriminatory power of features and the regression capabilities of classifiers. Comparing the highest accuracies (corresponding to the lowest error rates) to those of previous works, our results are very competitive.

**3.2 HARDWARE REQUIREMENTS**

The hardware requirements may serve as the basis for a contract for the implementation of the system and should therefore be a complete and consistent specification of the whole system. They are used by software engineers as the starting point for the system design. It should what the system do and not how it should be implemented.

* PROCESSOR : DUAL CORE 2 DUOS.
* RAM : 4GB DD RAM
* HARD DISK : 500 GB

**3.3 SOFTWARE REQUIREMENTS**

The software requirements document is the specification of the system. It should include both a definition and a specification of requirements. It is a set of what the system should do rather than how it should do it. The software requirements provide a basis for creating the software requirements specification. It is useful in estimating cost, planning team activities, performing tasks and tracking the teams and tracking the team’s progress throughout the development activity.

* Operating System : Windows 10
* Platform : Spyder3
* Programming Language : Python
* Front End : Spyder3

**3.4 FUNCTIONAL REQUIREMENTS**

A functional requirement defines a function of a software-system or its component. A function is described as a set of inputs, the behavior, Firstly, the system is the first that achieves the standard notion of semantic security for data confidentiality in attribute-based deduplication systems by resorting to the hybrid cloud architecture.

**3.5 NON-FUNCTIONAL REQUIREMENTS**

**The major non-functional Requirements of the system are as follows**

**Usability**

The system is designed with completely automated process hence there is no or less user intervention.

**Reliability**

The system is more reliable because of the qualities that are inherited from the chosen platform python. The code built by using python is more reliable.

**Performance**

This system is developing in the high level languages and using the advanced back-end technologies it will give response to the end user on client system with in very less time.

**Supportability**

The system is designed to be the cross platform supportable. The system is supported on a wide range of hardware and any software platform, which is built into the system.

**Implementation**

The system is implemented in web environment using Jupyter notebook software. The server is used as the intellignce server and windows 10 professional is used as the platform. Interface the user interface is based on Jupyter notebook provides server system.

**CHAPTER 4**

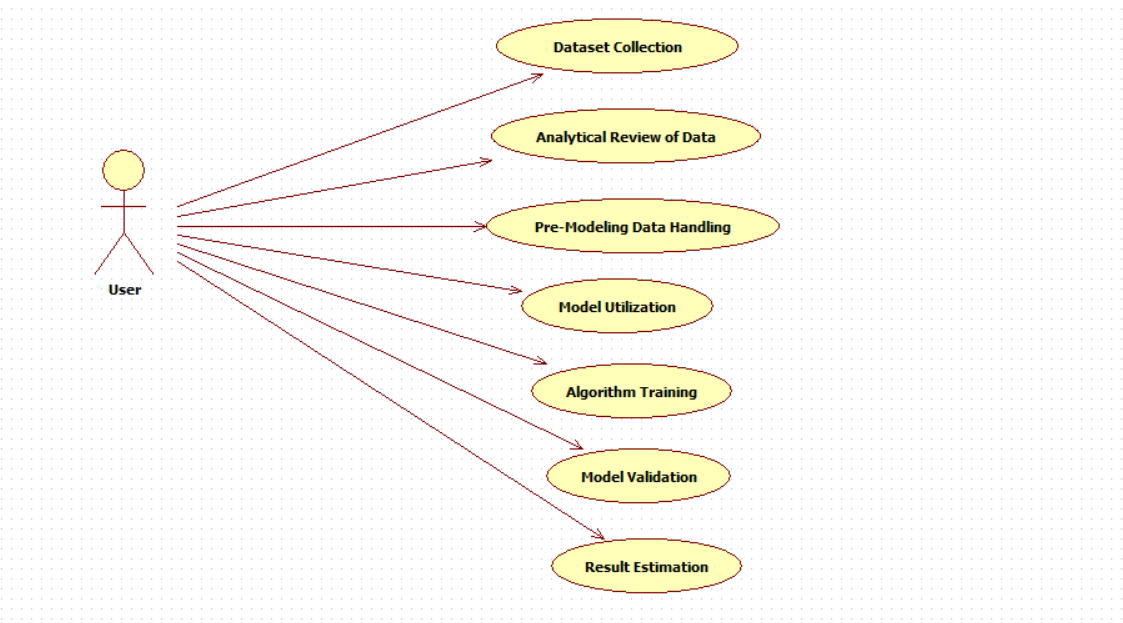
**DESIGN ENGINEERING**

**4.1 GENERAL**

Design Engineering deals with the various UML [Unified Modelling language] diagrams for the implementation of project. Design is a meaningful engineering representation of a thing that is to be built. Software design is a process through which the requirements are translated into representation of the software. Design is the place where quality is rendered in software engineering.

**4.2 UML DIAGRAMS**

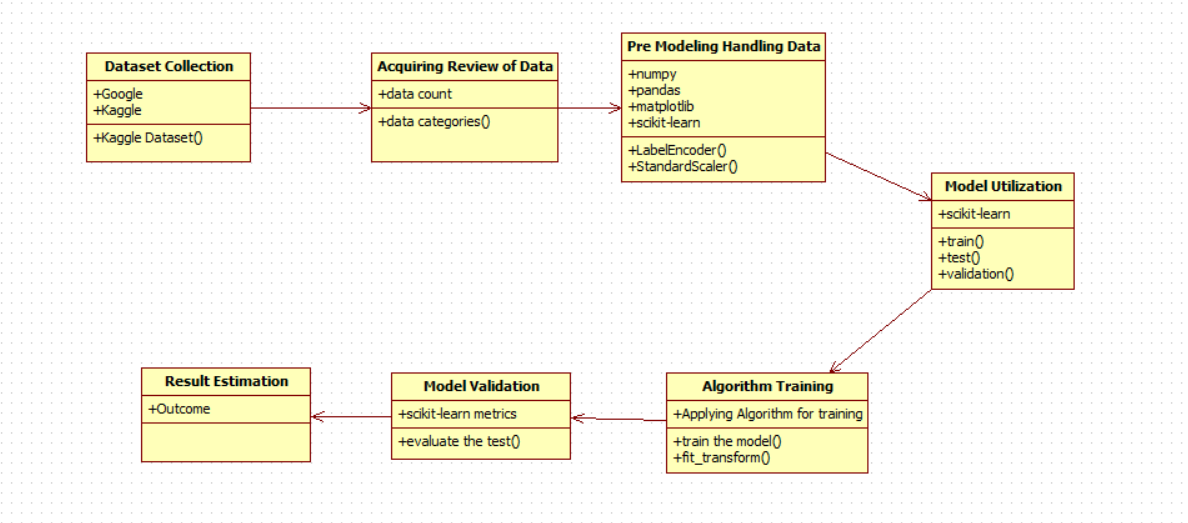
**4.2.1 USE CASE DIAGRAM**



**EXPLANATION:**

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. The above diagram consists of user as actor. Each will play a certain role to achieve the concept.

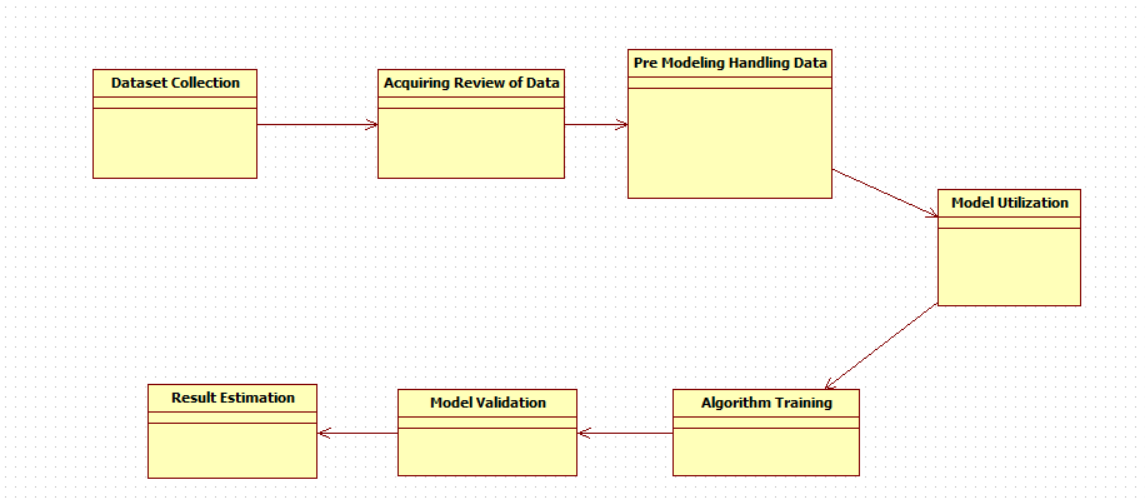
**4.2.2 CLASS DIAGRAM**

****

**EXPLANATION**

In this class diagram represents how the classes with attributes and methods are linked together to perform the verification with security. From the above diagram shown the various classes involved in our project.

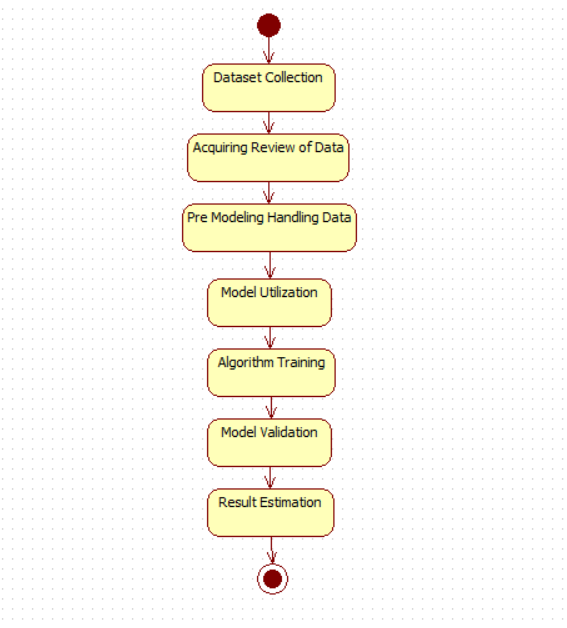
**4.2.3 OBJECT DIAGRAM**



**EXPLANATION:**

In the above digram tells about the flow of objects between the classes. It is a diagram that shows a complete or partial view of the structure of a modeled system. In this object diagram represents how the classes with attributes and methods are linked together to perform the verification with security.

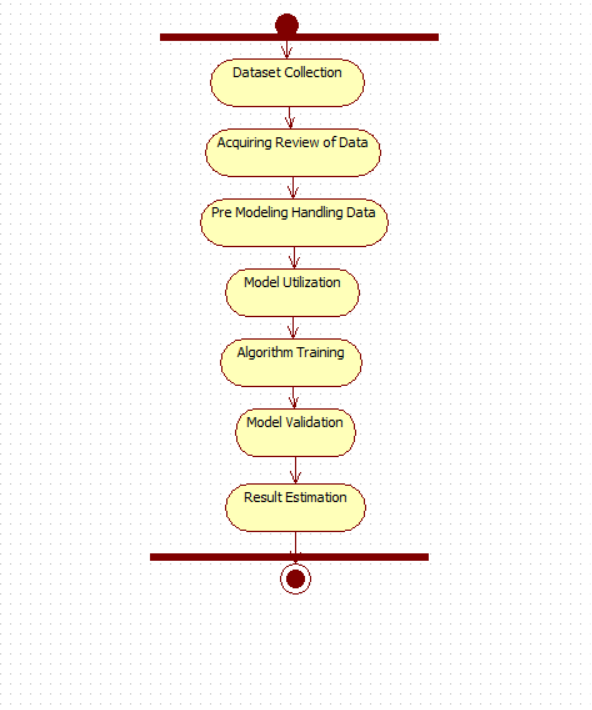
**4.2.4 STATE DIAGRAM**

****

**EXPLANATION:**

State diagram are a loosely defined diagram to show workflows of stepwise activities and actions, with support for choice, iteration and concurrency. State diagrams require that the system described is composed of a finite number of states; sometimes, this is indeed the case, while at other times this is a reasonable abstraction. Many forms of state diagrams exist, which differ slightly and have different semantics.

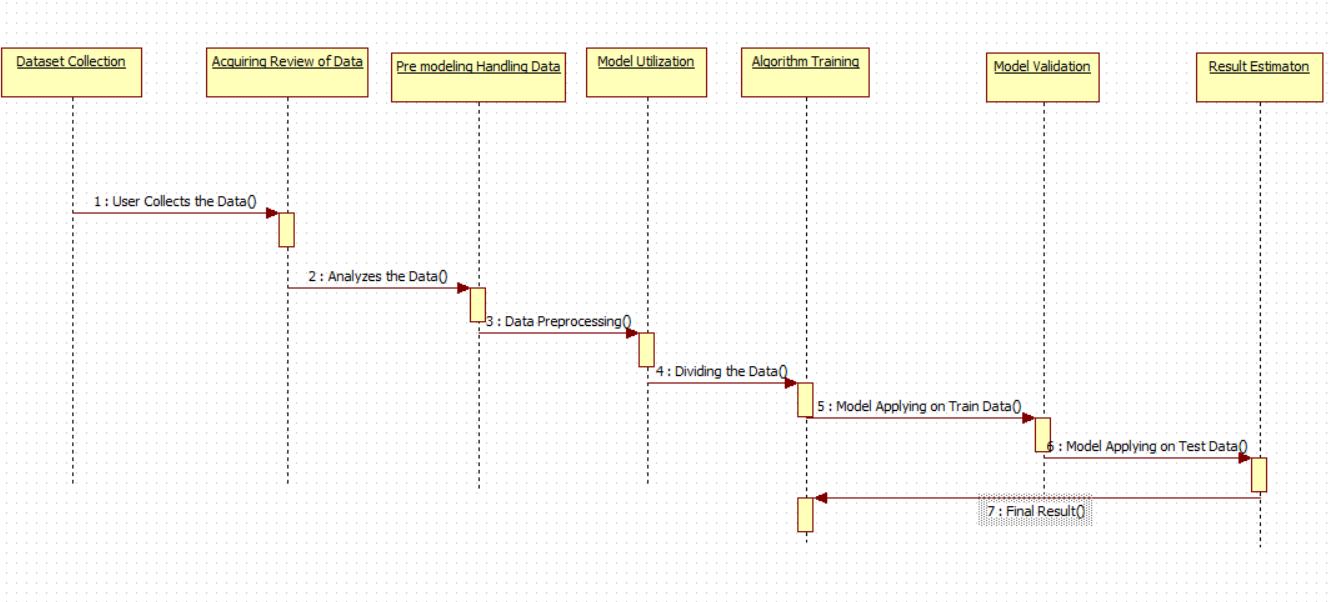
**4.2.5 ACTIVITY DIAGRAM**

****

**EXPLANATION:**

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modeling 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.

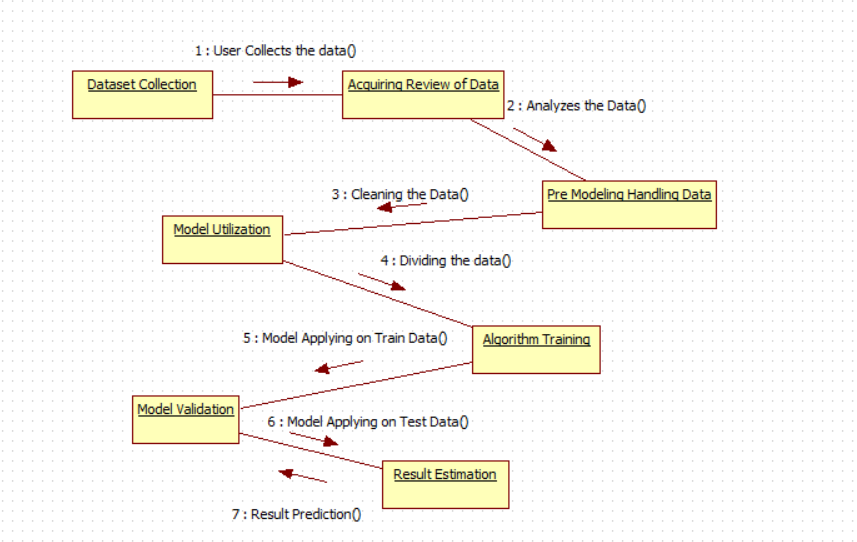
**4.2.6 SEQUENCE DIAGRAM**

****

**EXPLANATION:**

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. A sequence diagram shows object interactions arranged in time sequence. It depicts the objects and classes involved in the scenario and the sequence of messages exchanged between the objects needed to carry out the functionality of the scenario.

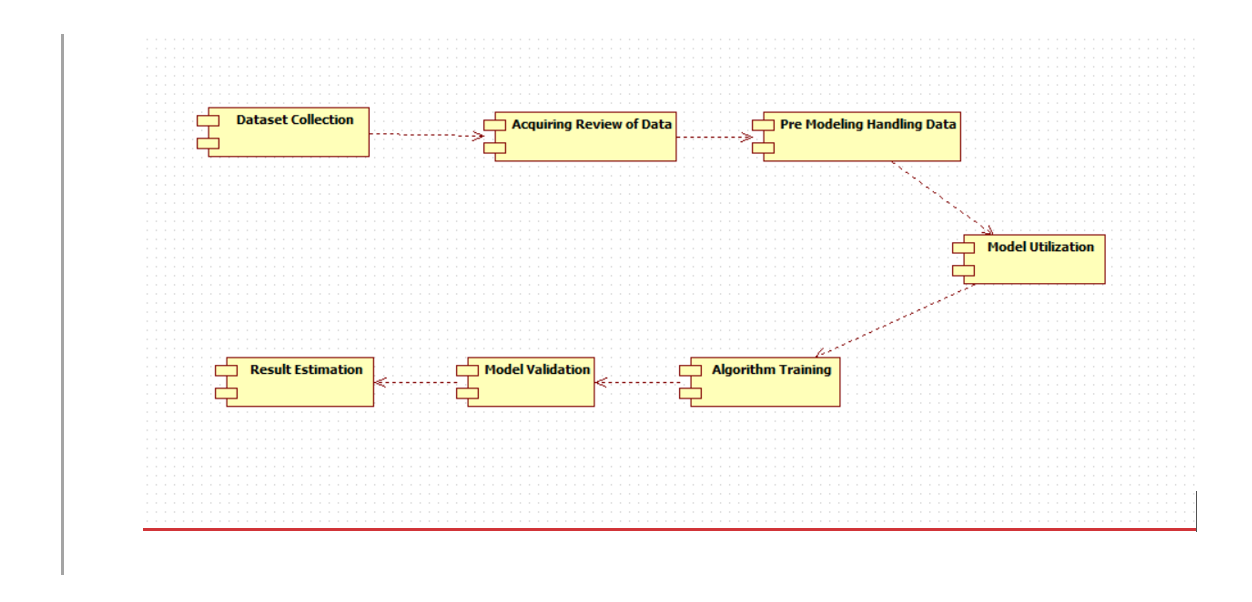
**4.2.7 COLLABORATION DIAGRAM**



**EXPLANATION:**

A collaboration diagram, also called a communication diagram or interaction diagram, is an illustration of the relationships and interactions among software objects in the Unified Modeling Language (UML). The concept is more than a decade old although it has been refined as modeling paradigms have evolved.

**4.2.8 COMPONENT DIAGRAM**



**EXPLANATION**

In the Unified Modeling Language, a component diagram depicts how components are wired together to form larger components and or software systems. They are used to illustrate the structure of arbitrarily complex systems. User gives main query and it converted into sub queries and sends through data dissemination to data aggregators. Results are to be showed to user by data aggregators. All boxes are components and arrow indicates dependencies.

**4.2.9 DATA FLOW DIAGRAM**

**Level 0**

Pre Modelling Handling Data

User

Dataset Collection

Acquiring Review of Data

**Level 1**

Result Estimation

Model Utilization

Algorithm Training

Model Validation

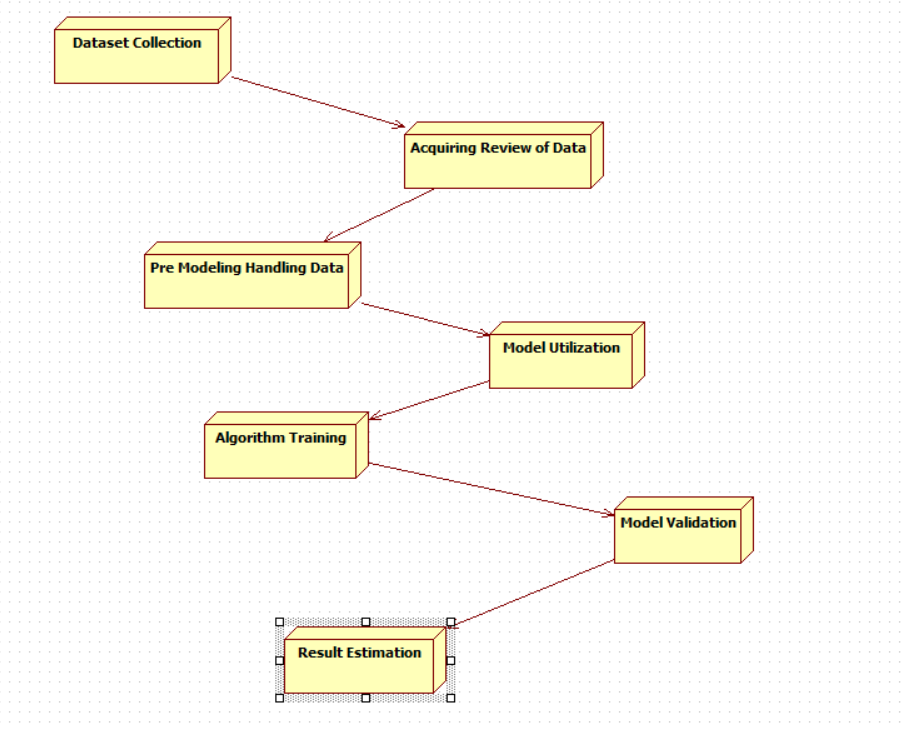
Fig 4.9: Data Flow Diagrams

**EXPLANATION:**

A data flow diagram (DFD) is a graphical representation of the "flow" of data through an information system, modeling its process aspects. Often they are a preliminary step used to create an overview of the system which can later be elaborated. DFDs can also be used for the visualization of data processing (structured design).

A DFD shows what kinds of data will be input to and output from the system, where the data will come from and go to, and where the data will be stored. It does not show information about the timing of processes, or information about whether processes will operate in sequence or in parallel.

**4.2.10 DEPLOYMENT DIAGRAM**

****

**EXPLANATION:**

Deployment Diagram is a type of diagram that specifies the physical hardware on which the software system will execute. It also determines how the software is deployed on the underlying hardware. It maps software pieces of a system to the device that are going to execute it.

**-**

**SYSTEM ARCHITECTURE:**

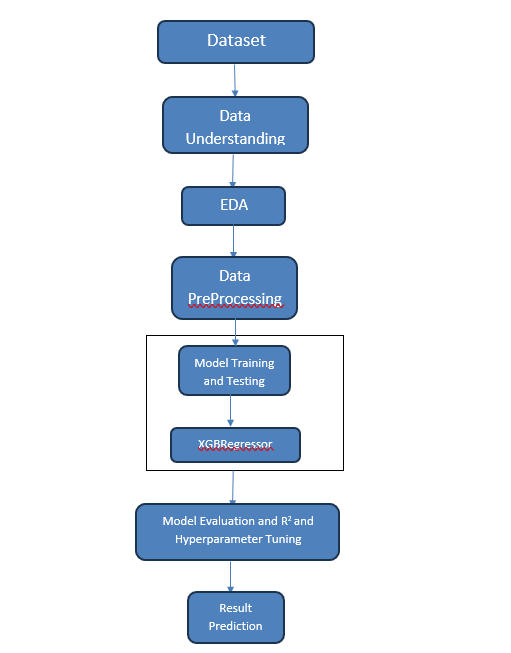


Fig 4.11: System Architecture

**CHAPTER 5**

**DEVELOPMENT TOOLS**

**5.1 Python**

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.

## 5.2 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, SmallTalk, 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.

#### 5.3 Importance of Python

* **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.

#### 5.4 Features of Python

* **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.

Apart from the above-mentioned features, Python has a big list of good features, few are listed below −

* It supports functional and structured programming methods as well as OOP.
* It can be used as a scripting language or can be compiled to byte-code for building large applications.
* 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.

**5.5 Libraries used in python**

* numpy - mainly useful for its N-dimensional array objects.
* pandas - Python data analysis library, including structures such as dataframes.
* matplotlib - 2D plotting library producing publication quality figures.
* scikit-learn - the machine learning algorithms used for data analysis and data mining tasks.



Figure : NumPy, Pandas, Matplotlib, Scikit-learn

**CHAPTER 6**

**IMPLEMENTATION**

**6.1 GENERAL**

**Coding:**

**CHAPTER 7**

**SNAPSHOTS**

**General:**

This project is implements like application using python and the Server process is maintained using the SOCKET & SERVERSOCKET and the Design part is played by Cascading Style Sheet.

**SNAPSHOTS**

**CHAPTER 8**

**SOFTWARE TESTING**

**8.1 GENERAL**

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.

**8.2 DEVELOPING METHODOLOGIES**

The test process is initiated by developing a comprehensive plan to test the general functionality and special features on a variety of platform combinations. Strict quality control procedures are used. The process verifies that the application meets the requirements specified in the system requirements document and is bug free. The following are the considerations used to develop the framework from developing the testing methodologies.

**8.3Types of Tests**

**8.3.1 Unit testing**

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program input 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.

**8.3.2 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.

**8.3.3 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.

**8.3.4 Performance Test**

The Performance test ensures that the output be produced within the time limits,and the time taken by the system for compiling, giving response to the users and request being send to the system for to retrieve the results.

**8.3.5 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.

**8.3.6 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.

**Acceptance testing for Data Synchronization:**

* The Acknowledgements will be received by the Sender Node after the Packets are received by the Destination Node
* The Route add operation is done only when there is a Route request in need
* The Status of Nodes information is done automatically in the Cache Updation process

**8.2.7 Build the test plan**

Any project can be divided into units that can be further performed for detailed processing. Then a testing strategy for each of this unit is carried out. Unit testing helps to identity the possible bugs in the individual component, so the component that has bugs can be identified and can be rectified from errors.

**CHAPTER 9**

**FUTURE ENHANCEMENT**

**9.1 FUTURE ENHANCEMENTS:**

Future enhancements for vehicle insurance fraud detection using the proposed AdaBoost Classifier could significantly elevate the accuracy and efficiency of fraud detection systems. One key area for improvement is the integration of real-time analytics to detect fraudulent claims as they occur, leveraging the adaptive capabilities of the AdaBoost algorithm to refine predictive accuracy continuously. Additionally, expanding the data sources to include telematics data from vehicles, such as driving behavior, GPS logs, and real-time accident data, could provide deeper insights into fraudulent patterns, allowing the model to differentiate between genuine and suspicious claims more effectively.

Another potential enhancement involves the incorporation of Explainable AI techniques alongside AdaBoost, such as SHAP (SHapley Additive exPlanations) and LIME (Local Interpretable Model-Agnostic Explanations). These techniques would make the fraud detection process more transparent by explaining the factors influencing each prediction, helping insurance companies to understand and trust the model's decisions. Future work could also focus on enhancing the system's scalability by optimizing the AdaBoost model to handle larger datasets, thereby improving its performance in diverse geographic regions with varying claim patterns. Lastly, implementing adaptive learning strategies that allow the system to evolve with emerging fraud techniques and trends can further fortify the robustness of insurance fraud detection, ultimately reducing financial losses for insurers and ensuring fairer premiums for policyholders.

**CHAPTER 10**

**CONCLUSIONAND REFERENCES**

**10.1** **CONCLUSION**

The conclusion of this project highlights the significant advancements achieved in vehicle insurance fraud detection by implementing the AdaBoost Classifier. The proposed model effectively addresses challenges associated with class imbalance and missing data, which are common in insurance datasets. By leveraging AdaBoost, the system enhances the detection of fraudulent claims with greater accuracy and reduced overfitting, ensuring a robust predictive model that adapts to various fraud patterns. This approach not only improves the precision of identifying fraudulent activities but also contributes to minimizing financial losses for insurance companies.

**10.2 REFERENCES**

[1] M. Shahbandeh. (2021). Rice Consumption Worldwide in 2021/2022, By Country (in 1,000 Metric Tons). Accessed: Nov. 25, 2023. [Online]. Available: https://www.statista.com/statistics/255971/top countries-based-on-rice-consumption-2012 2013/#:~:text=Asthemostpopulouscountry,consumptioninthesameperiod.

[2] U.S. Department of Agriculture. (2020). Rice Sector at a Glance 2020/21–22/23. Accessed: Nov. 25, 2023. [Online]. Available: https://www.ers.usda.gov/topics/crops/rice/rice-sector-at a-glance/#Global

[3] M.F.Ikhwali, S.Nur, D.Darmansyah,A.M.Hamdan,N.S.Ersa,N.Aida, A. Yusra, and A. Satria, ‘‘A review of climate change studies on paddy agriculture in Indonesia,’’ IOP Conf. Ser., Earth Environ. Sci., vol. 1116, no. 1, Dec. 2022, Art. no. 012052.

[4] B.SmerbeckandB.Thompson.(Nov.21,2023).HowAccurateisTheOld Farmer’s Almanac’s Weather Forecast? Almanac. [Online]. Available: https://www.almanac.com/how-accurate-old-farmers-almanacs-weather forecast

[5] S. K. Purohit, S. Panigrahi, P. K. Sethy, and S. K. Behera, ‘‘Time series forecasting of price of agricultural products using hybrid methods,’’ Appl. Artif. Intell., vol. 35, no. 15, pp. 1388–1406, Dec. 2021.

[6] P.Mishra1,‘‘Forecastingofriceproductionusingthemeteorologicalfactor in major states in India and its role in food security,’’ Int. J. Agricult. Environ. Biotechnol., vol. 14, no. 1, p. 2021, Mar. 2021.

[7] D.Sihi,B.Dari,A.P.Kuruvila,G.Jha,andK.Basu,‘‘Explainablemachine learning approach quantified the long-term (1981–2015) impact of climate and soil properties on yields of major agricultural crops across CONUS,’’ Frontiers Sustain. Food Syst., vol. 6, Apr. 2022, Art. no. 847892.

[8] M. Meroni, F. Waldner, L. Seguini, H. Kerdiles, and F. Rembold, ‘‘Yield forecasting with machine learning and small data: What gains for grains?’’ Agricult. Forest Meteorol., vols. 308–309, Oct. 2021, Art. no. 108555.

[9] P. Kamath, P. Patil, S. Shrilatha, Sushma, and S. Sowmya, ‘‘Crop yield forecasting using data mining,’’ Global Transitions Proc., vol. 2, no. 2, pp. 402–407, Nov. 2021.

[10] M. Shahhosseini, G. Hu, and S. V. Archontoulis, ‘‘Forecasting corn yield with machine learning ensembles,’’ Frontiers Plant Sci., vol. 11, pp. 1–16, Jul. 2020.

[11] A. Sharma, A. Jain, P. Gupta, and V. Chowdary, ‘‘Machine learning applications for precision agriculture: A comprehensive review,’’ IEEE Access, vol. 9, pp. 4843–4873, 2021.

[12] N. Bali and A. Singla, ‘‘Emerging trends in machine learning to predict crop yield and study its influential factors: A survey,’’ Arch. Comput. Methods Eng., vol. 29, no. 1, pp. 95–112, Jan. 2022.

[13] A. Cravero, S. Pardo, S. Sepúlveda, and L. Muñoz, ‘‘Challenges to use machine learning in agricultural big data: A systematic literature review,’’ Agronomy, vol. 12, no. 3, p. 748, Mar. 2022.

[14] R. Alfred, J. H. Obit, C. P. Chin, H. Haviluddin, and Y. Lim, ‘‘Towards paddy rice smart farming: A review on big data, machine learning, and rice production tasks,’’ IEEE Access, vol. 9, pp. 50358–50380, 2021.

[15] G. Pradeep, T. D. V. Rayen, A. Pushpalatha, and P. K. Rani, ‘‘Effective crop yield prediction using gradient boosting to improve agricultural outcomes,’’ in Proc. Int. Conf. Netw. Commun. (ICNWC), Apr. 2023, pp. 1–6.

[16] B. M. Sagar, N. K. Cauvery, T. Padmashree, and R. Rajkumar, ‘‘Rice and wheat yield prediction in India using decision tree and random forest,’’ Comput. Intell. Mach. Learn., vol. 3, no. 2, pp. 1–8, Oct. 2022.

[17] K. Choudhary, W. Shi, Y. Dong, and R. Paringer, ‘‘Random forest for rice yield mapping and prediction using Sentinel-2 data with Google Earth engine,’’ Adv. Space Res., vol. 70, no. 8, pp. 2443–2457, Oct. 2022.

[18] K. K. Paidipati, C. Chesneau, B. M. Nayana, K. R. Kumar, K. Polisetty, and C. Kurangi, ‘‘Prediction of rice cultivation in India—Support vector regression approach with various kernels for non-linear patterns,’’ AgriEngineering, vol. 3, no. 2, pp. 182–198, Apr. 2021.

[19] V. Amaratunga, L. Wickramasinghe, A. Perera, J. Jayasinghe, and U. Rathnayake, ‘‘Artificial neural network to estimate the paddy yield prediction using climatic data,’’ Math. Problems Eng., vol. 2020, pp. 1–11, Jul. 2020.

[20] S. Rathod, S. Yerram, P. Arya, G. Katti, J. Rani, A. P. Padmakumari, N. Somasekhar, C. Padmavathi, G. Ondrasek, S. Amudan, S. Malathi, N. M. Rao, K. Karthikeyan, N. Mandawi, P. Muthuraman, and R. M. Sundaram, ‘‘Climate-based modeling and prediction of rice gall midge populations using count time series and machine learning approaches,’’ Agronomy, vol. 12, no. 1, p. 22, Dec. 2021.

[21] S. Chairani, ‘‘The correlation between rainfall, temperature, relative humidity, and rice field productivity in Aceh Besar,’’ IOP Conf. Ser., Earth Environ. Sci., vol. 1071, no. 1, pp. 1–17, 2022.

[22] A. Abdullah and M. H. Nahid, ‘‘Performance analysis rice yield model based on historical weather dataset in Bangladesh,’’ in Proc. 4th Int. Conf. Sustain. Technol. Ind. 4.0 (STI), Dec. 2022, pp. 1–6.

[23] K. T. Soberano, J. S. Pisueña, S. M. R. Tee, J. C. T. Arroyo, and A. J. P. Delima, ‘‘Predictive soil-crop suitability pattern extraction using machine learning algorithms,’’ Int. J. Adv. Appl. Sci., vol. 10, no. 6, pp. 8–16, Jun. 2023.

[24] Y. Iuchi, H. Uehara, Y. Fukazawa, and Y. Kaneta, ‘‘Stabilizing the predictive performance for ear emergence in rice crops across crop ping regions,’’ in Proc. Pacific Rim Knowl. Acquisition Workshop, in Lecture Notes in Computer Science: Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics, 2021, pp. 83–97.

[25] S. Ngandee and A. Taparugssanagorn, ‘‘Improved information dissem ination services for the agricultural sector in Thailand: Development and evaluation of a machine learning based rice crop yield prediction system,’’ Inf. Develop., pp. 1–14, Nov. 2023. [Online]. Available: <https://journals.sagepub.com/doi/10.1177/02666669231208017>

[26] W. Mongkolnithithada, J. Nontapun, and S. Kaewplang, ‘‘Rice yield estimation based on machine learning approaches using MODIS 250 m data,’’ Eng. Access, vol. 9, no. 1, pp. 75–79, 2023.

[27] P. Roy, B. Kumar, P. K. Bharti, V. K. Vishnoi, K. Kumar, S. Mohan, and K. P. Singh, ‘‘Paddy yield prediction based on 2D images of rice panicles using regression techniques,’’ Vis. Comput., vol. 40, no. 6, pp. 4457–4471, Jun. 2024.

[28] X.Zhang,R.Shen,X.Zhu,B.Pan,Y.Fu,Y.Zheng,X.Chen,Q.Peng,and W.Yuan,‘‘Sample-free automated mappingofdouble-season rice in China using Sentinel-1 SAR imagery,’’ Frontiers Environ. Sci., vol. 11, pp. 1–11, Jul. 2023.

[29] L. Li, B. Wang, P. Feng, H. Wang, Q. He, Y. Wang, D. L. Liu, Y. Li, J. He, H. Feng, G.Yang,andQ.Yu,‘‘Cropyieldforecasting and associated optimum lead time analysis based on multi-source environmental data across China,’’ Agricult. Forest Meteorol., vols. 308–309, Oct. 2021, Art. no. 108558.

[30] Y. Tanaka et al., ‘‘Deep learning enables instant and versatile estimation of rice yield using ground-based RGB images,’’ Plant Phenomics, vol. 5, pp. 1–16, Jan. 2023.