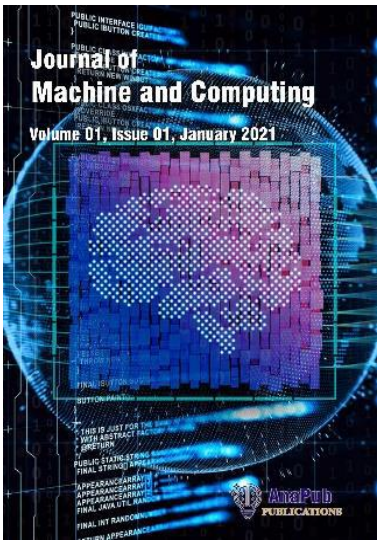
Journal Pre-proof



Automatic Crop Recommendation System Using LightGBM and Decision Tree Machine Learning Models

Ravi Kumar Banoth and Ramana Murthy B V

DOI: 10.53759/7669/jmc202505026

Reference: JMC202505026

Journal: Journal of Machine and Computing.

Received 28 May 2024

Revised form 16 August 2024

Accepted 22 November 2024

Please cite this article as: Ravi Kumar Banoth and Ramana Murthy B V, “Automatic Crop Recommendation System Using LightGBM and Decision Tree Machine Learning Models”, Journal of Machine and Computing. (2025). Doi: https:// doi.org/10.53759/7669/jmc202505026

AnaPub
PUBLICATIONS

Automatic Crop Recommendation System Using LightGBM and Decision Tree Machine Learning Models

Ravi Kumar Banoth[1], Dr. B.V. Ramana Murthy[2]
[1] Research Scholar, Department of Computer Science and Engineering, Osmania University College of Engineering, Osmania University, Hyderabad, Telangana, India.
[2] Professor, Department of Computer Science and Engineering, Stanley College of Engineering & Technology for Women, Abids, Hyderabad, Telangana, India.
brkouce@gmail.com[1], drbvrm@gmail.com[2]

Orchid Id: 0009-0005-3950-0373

Abstract

An Automatic Crop Recommendation System is a system that makes use of data analysis and algorithms to recommend crops that are suitable and proper with regard to soil quality, climate, and local factors. Such a system eases the decision-making process for farmers. The necessity for efficient agricultural techniques is growing rapidly, and it is impossible without the application of modern technology that would promote the quality of the ideal crop selection list and production. This paper introduces a new concept of the Automatic Crop Recommendation System, integrating the LightGBM and the Decision Tree algorithms. The research uses the strengths of LightGBM, a type of gradient boosting framework, and Decision Tree, a conventional machine learning model, to form a powerful mixed ensemble approach. These approaches are combined to exploit their complementary strengths, leading to a more accurate and dependable agricultural advisory system. The effectiveness of the proposed algorithm's approach is verified through several experimental results; it has the following accuracies, recalls, and F-1 scores. The process has proven very successful; an accuracy of 98.64% makes it possible to recommend appropriate and accurate crops.

Keywords: Crop Recommendation, LightGBM, Gradient Boosting, Decision tree, Ensemble model.

1. Introduction

The agricultural industry is a critical element in ensuring the prolongation of human life through the availability of necessary resources such as food, fiber, and raw materials. Nevertheless, one of the cornerstones of production – cultivation of crops – is being impeded by a variety of complex issues, ranging from unexpected meteorological patterns and soil compositional differences to the rapid alteration of the market's interests. Given the fast-paced nature of agriculture, farmers face numerous hurdles as soon as they need to determine a crop most suitable for cultivation on their land. Crops selecting is further complicated due to the unique combination of soil qualities and circumstances in which the meteorological window and market interests lie. Furthermore, due to the lack of readily available and accurate information, farmers have an increased quantity of barriers in terms of making an informed decision. It is here where technical advancements are needed in the form of a reliable automatic system that would provide accurate recommendations on crop choices tailored to specific agricultural needs [2]. Given the nature of these difficulties, the Automatic Crop Recommendation System using Machine Learning approach seems to be an adequate way of addressing crop choice issues.

Machine Learning is a technology which can be applied to combat the issues related to complexities in agriculture. It is a kind of algorithm which can be used for processing the large datasets containing the information of the various fields in agriculture sector such as concerned soil of variety, last year's information of weather, landscape territory wise classification, and finally feedback of every crop, etc. [3].As a result of dealing with an abundant number of details, machine learning models can find relationships, associations, and patterns which can be used to make well-informed recommendations for a particular crop. Further, due to their adaptive and scalable nature, machine learning algorithms improve accuracy and affordability over time since machine learning discovery and refinement are standard procedures.

Integration of Machine Learning methods. This change in agricultural decision-making is applied in an Automatic Crop Recommendation System [4]. Due to predictive analytics and pattern recognition, it enables farmers to identify which crop to plant. Therefore, it already have the outcome of maximum potential for yield and reduction in the use of resources consumed. It is also important that the abovementioned systems have other theoretical potential effects that are not limited to specific outcomes as the level of a farm specific out [5].

Machine Learning methods being integrated into an Automatic Crop Recommendation System. Not only does this entail predictive analytics and pattern recognition, but it also enables farmers to make educated decisions about what they should be planting on the field. As a result, it has the potential to "optimize the payoff function and minimize the resource consumed". Moreover, it is noteworthy that the systems described earlier have other possible implications that are beyond the scope of particular outcomes on a farm-by-farm level.

In reality, the widespread adoption of such systems could significantly contribute to the development of sustainable agricultural processes and reduce the risk of food insecurity on a global level. The purpose of the present work is to analyze the process of developing an Automatic Crop Recommendation System using Machine Learning, including planning, developing, and evaluating the system. The goal is to prove its efficaciousness and transformative power in the agricultural environment by providing farmers with actionable insights and helping them develop their capacity for efficient crop selection in a rapidly changing environment. The organization of the paper as follows: The Literature survey is discussed in section-II and proposed model is explained in section-III. The results and analysis is described in section-IV.

2. Literature

Sita Rani et al [6] presented a novel crop selection framework that use machine learning techniques to incorporate meteorological conditions and soil factors. The weather evaluation incorporates the use of Long Short-Term Memory Recurrent Neural Networks (LSTM RNN), even with the crop selection procedure utilizes the Random Forest Classifier. The results indicate that this particular model exhibits superior performance in terms of weather forecast accuracy compared to Artificial Neural Networks (ANN). Murali Krishna Senapaty et al [7] discovered a fresh way via the integration of algorithms. An algorithm has been devised by using a multi-class support vector machine integrated with a directed acyclic graph, and then enhanced by the use of the fruit fly optimisation technique. This algorithm is referred to as MSVM-DAG-FFO.

Nizom Farmonov et al [8] utilized imagery obtained from the Deutsches Zentrum für Luft- und Raumfahrt Earth Sensing Imaging Spectrometer (DESIS) to categorize the prevailing crop types (hybrid corn, soybean, sunflower, and winter wheat) in Mezőhegyes, a region situated in southeastern Hungary. Several methods, such as the Wavelet-attention convolutional neural network (WA-CNN), random forest, and support vector machine (SVM), were used to autonomously define and map the aforementioned crops inside the agricultural regions.

Ankit R. Sawant et al [9] propose an approach to assist agricultural practitioners in making well-informed decisions on what types of crops to produce based on a variety of parameters that relate to their contextual and environmental conditions. For example, by developing predictive models to identify essential factors that impact crop growth, such as soil nutrients, pH, humidity, and the amount of rainfall. Different machine learning models can be considered, including Decision Tree (DT), Support Vector Machine (SVM), Logistic Regression (LR), and Gaussian Naïve Bayes (GNB).

Tawseef Ayoub Shaikh et al [10] illustrated some of the consequences that could occur due to the development of Information and Communication Technology (ICT) in traditional agriculture. This research indicates the limitation that might occur when integrating new technologies into agriculture. Many other aspects, including the use of Robotics, IoT devices, and machine learning, are being considered according to this article inside the agriculture process. Also, a broad examination of machine learning, artificial intelligence, and sensors' responsibilities and capabilities in the agricultural industry is conducted. This study examines the potential use of unmanned aerial vehicles (UAVs), often known as drones, in the context of agricultural monitoring and the enhancement of crop yield management. Moreover, it provides insights into worldwide and state-of-the-art Internet of Things (IoT)-based agricultural systems and platforms, when appropriate. A thorough examination of current scholarly works within each specific field is undertaken. The study finishes by providing an overview of current and prospective developments in the field of artificial intelligence (AI). It also emphasizes the research problems that now exist and are anticipated to arise in the future in relation to the use of AI in agriculture. These insights are derived from a thorough and complete review conducted in the study.

Emna Ben Abdallah et al [11] presented a prominent contribution to this area by introducing a generalized technique based on machine learning that finds the optimal amount of water favoring the growth of the plant. The proposed methodology is the one that uses feature selection methods along with a stacking ensemble method. First of all, the importance of the features is evaluated using the Random Forest, Recursive Feature Elimination (RFE), and SelectKBest methods. Then, a stacking ensemble model is formulated, including a combination of regressors like regression as a tree, CART, Gradient Boost Regression, Random Forest , and XGBoost, with the optimal set of features given by the feature selection method. Our models have been trained and tested with a wider dataset with planted crops like tomato, grape, lemon, and varied data like meteorological, soil data, and irrigation data along with crop-related factors. Altogether, this work offers strong support for the use of the Random Forest model to evaluate feature importance. The resultant features selected are, along with the relative importance, the addition of the two depletion and deficiency components, and the evapotranspiration parameter.

Kalaiselvi Bhakthavatchalam et al [12] detailed a supervised learning strategy to produce the perfect model making use of the dedicated machine learning algorithms in the WEKA software. The following machine learning algorithms were applied in the classification problem: the multilayer perceptron as well as rules-based classifier JRip and decision table classifier. In summary, the target of this case study is the creation of a model that forecasts high-yield crops in the area of precision farming effectively. These solutions employ future technology, such as the Internet of Things, and existing agricultural indicators to increase accuracy and usefulness in farming activities.

Akanksha Gupta et al [13] presented a novel two-tier machine learning model for predicting the crop yield that uses the IoT technology. The processes of the present study can be divided into three distinctive stages: pre-processing, feature selection, and classification. Initially, the dataset must be preprocessed, following which the Correlation based Feature Selection (CBFS), and VIF must be used to select the features. The IoT based smart agricultural system should utilize the novel two-tier machine learning model, consisting of the Adaptive K-Nearest Centroid Neighbour Classifier and the Extreme Learning Machine algorithm. The soil quality should be assessed using the aKNCN; in other words, the soil samples should be put into multiple categories depending on the input soil parameters. The use of the ELM should allow for forecasting crop production. The optimized procedure involves establishing the weights using the modified Butterfly Optimization Algorithm to be able to change the weights of the performance of Extreme Learning Machines Python was the tool chosen for the implementation of the present system. The soil dataset was used as the basis for the performance evaluation of the prediction model. Several factors were studied to understand the performance of the model: accuracy, root mean square error (RMSE), R-squared (R2), mean squared error (MSE), median absolute error (MedAE), mean absolute error (MAE), mean squared logarithmic error (MSLE), mean absolute percentage error (MAPE), and explained variance score (EVS).

Manik Rakhra et al [14] employ framework has three separate machine learning models or closest neighbors, logistic regression and decision tree. The most prevalent model among the models under consideration appeared to be the K-nearest neighbors, followed by logistic regression and a decision tree. In other to determine the most advantageous decision the competitive analysis was conducted on the models offer of algorithms. There was an evident difference in the precision tree model which is superiorly efficient compared to the model used in the above framework. The decision tree model has multiple inputs which include the crop type, the period of the harvest, and the equipment requirements. Therefore, the model is likely to have remarkable social and economic impact on farmers' survival.

Priyadharshini A et al [15] Introduced an innovative methodology to assist farmers in determining the optimal crop. A system that considers many parameters, such as sowing season, soil conditions, and geographical location is used. Furthermore, the adoption of modern agricultural technology has enabled the application of precision agriculture which is becoming more popular in developing countries. It is based on the careful control of crops in certain locations.

SHILPA MANGESH PANDE et al [16] presented a system that is convenient and feasible, developed specifically to predict agricultural turnover in a way that can be tailored to the needs of farmers. The writing system gives farmers access through a mobile application that can

determine the user's location using GPS technology. Farmers communicated certain pieces of information, such as the regional location of cultivation and the specifics of the soil they used, and then machine learning method analyzed this information. These computational algorithms help determine which crop has a better chance of generating revenue or surfacespecthe amount of return with each year planting requested by the farmer. The system uses multiple Machine Learning techniques, including Support Vector Machine, Artificial Neural Network, Random Forest, Multivariate Linear Regression, and K-Nearest Neighbour to predict crop yield.

Shuting Yang et al [17] presented a novel approach for selecting training samples is totally automated. This approach is initially created through image processing by following a sliding window concept. They then complete the Geo-3D convolutional neural network and Geo-Conv1D to classify crops using the time-series Sentinel-2 image. Particularly, this methodology embeds spatial agricultural data in the deep learning networks system. Finally, they use an active learning technique to realize the classification advantages of Geo-3D CNN and Geo-Conv1D. The results of the experiment in Northeast China indicate that theproposed sampling approach continuously generates a large number of samples and labels them properly. These results suffice to be applicable across different networks under deep-learning design since they are tested conclusively.

3. Proposed Model

This section details the proposed model. With the combination of the potent LightGBM and traditional Decision Tree algorithms, a novel Crop Recommendation System was developed. Inquiry the correlation of the LightGBM's robust gradient boosting architecture and use the well-known Decision Tree model to invent a powerful ensemble model.

3.1 LightGBM

LightGBM is a powerful gradient boosting framework, created especially for the distributed and efficient training of ensembles of decision trees. LightGBM, which was created by Microsoft, may be used in many technical contexts, not exclusive to Microsoft's own. It is specifically excellent for dealing with large datasets and feature spaces that are sparse or high dimension. The system's architecture and approach were made to make training much faster and memory use more efficient, all while achieving equivalent predictive performance to what current approaches achieve.

A different approach to learning underlies the design of LightGBM. LightGBM uses a leaf-wise design wherein very little regularization is used. Each level does not result in producing a tree. Rather than creating each level of the tree, the system creates the leaf node that gives the greatest reduction to the loss. This leaf-wise design allows the researchers to create a little more vertical tree. What separates LightGBM from XGBoost's structure is the mode of linkage of tree structures in the training data. A histogram-based computation of gradients is one of the most crucial extensions in the training step.

LightGBM discretizes the function values into pails and constructs histograms instead of observing all points for the division point. As a result, less idle work is done for each section, which speeds up convergence and reduces memory consumption in each step during training.

LightGBM's training is based on numerous processes. Firstly, the model is initiated by one leaf and with every upcoming step, more leaf is attached with the tree structure. The most

appropriate split points for each leaf are determined using a gradient-based optimization technique which on the first step uses histogram data. The process is recursively carried out until a certain number of trees is produced or whenever the approach converges.

LightGBM can also do parallel and distributed training, implying that it is well-suited for scaling up with many datasets. It is because of data parallelism and the usage of feature parallelism enables the distribution work over many computer resources hence reduces the number of model training. LightGBM also supports most hyperparameter tuning, enabling the consumer to modify different aspects of the model according to the use and preference.

The LightGBM processing steps may be presented as:

Step 1. Data Loading, LightGBM data importation involves importing the training data, presumably as a dataset or a data frame.

Step 2. Data Preprocessing, Data pre-processing is the preparation of the dataset, such as value replacement, handling categorical characteristics, and other activities included in data cleaning.

Step 3. Data Splitting, the dataset used should be subjected to splitting into the training and validation sets to monitor the model performance during training.

Step 4. Feature Engineering, these are skills intended to create new features or redesign existing ones to boost the ability of the machine learning model to identify patterns. In other words, additional functionalities may be generated or current functionalities altered.

Step 5. Parameter Configuration, these include hyperparameter defaults, such as learning rates, tree depth, and a range of other parameters that govern the model behavior..

Step 6. Training the Model, LightGBM classifier is trained on the training set with boosting to minimize the variance of the model's prediction.

Step 7. Evaluation of the Model, this is an action involving testing the model on the validation set to obtain a performance measure and prevent overfitting.

Step 8. Hyperparameter Tuning, parameters are fine-tuned appropriately and carefully to increase model performance rates, preferably through grid search or random search techniques.

Step 9. Training the Final Model, when satisfied with the performance of the model, the model is then, trained using the entire data.

Step 10. Prediction, LightGBM classifier prowess is excellent in predicting new data that it have not seen, and we, thus, use it to generate prediction.

3.2 Decision Tree

Decision Tree Classifier is one of the most flexible machine learning techniques applied to categorization and prediction problems. The structure of the basic tree is similar. The tree has core nodes that signify the basis of any decision relying on some features. Afterward, it has some branches that explain the attainable output of that decision and, ultimately, the leaf nodes that signify the end result or the probable label. Several basic structural fragments and a procedure for constructing a Decision Tree are included in the establishment of a decision tree. Figure 1 depicts Decision Tree architecture.

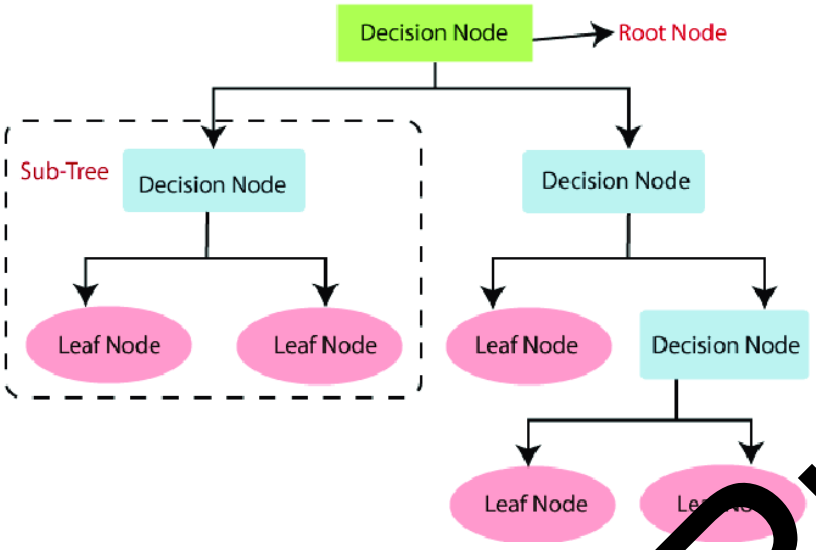


Figure 1: Decision Tree Architecture

Starting with the root node, which does not have a parent and is the tree's top, the construction presents the node that shows the best possible characteristic for splitting the dataset. For this purpose, subsets are selected from the input data source according to the level of information gain, Gini impurity, or entropy. The method's goal is to maximize the separateness of the subgroups created in the process. The following nodes after the root are internal, which also may be called decision nodes. They represent decision points based on how the characteristic selected is different from other candidates. Each internal node further divides into subgroups depending on all possible values for the selected characteristic. The branches describe the outcome of the selection, which may lead to the new internal node or a leaf. The terminus nodes of the latter ones are located at the branches after the internal nodes and represent the result of the choosing or predicted label. Usually, they are determined by the majority class within the layer.

Thus, the tree is built iteratively and self-referencing. At each node, the algorithm selects the optimal feature for splitting, separates the dataset into subsets accordingly, and continues doing so until a certain stop condition is met. It may be a specified depth or a minimum number of samples in a node. Meanwhile, the leaf nodes are created through tree construction and labeled based on most occurrences in this leaf.

The method of decision tree categorization is constituted by moving through the tree structure and commencing at the root so as to advance through the leaf, depending on the input's characteristics. At an internal stage, the algorithm first assesses the attribute condition's credibility and selects the child node in an attuned direction. The navigation proceeds down until a leaf node is discerned, where the category label of this category is ascribed to the input data as the expected class.

Pruning, on the other hand, is an alternative method used to reduce the complexity and overfit of the tree. It eliminates the root or branch conditions with inconsiderable effect on the model's correctness. The consequence is a more generalized tree, which is effective at making predictions on unknown data. The quintessence of making predictions and associations from this instance is that the decision tree is well-known for its ease-of-interpretation, inferencing, and use for multiple data instances.

The process of decision tree processing includes the following steps:

Step 1. Initialization, start by designating the whole dataset as the root node.
Step 2. Feature Selection, pick for the most optimal feature to divide the dataset by using a criterion.
Step 3. Splitting, divide the dataset into subgroups according to the selected characteristic.
Step 4. Recursive Process, iterate steps 2 and 3 for every subset (child node) until a stopping condition is satisfied (e.g., a predetermined depth or a minimum number of samples in a node).
Step 5. Decision Making, determine the class label for each terminal node (leaf) by selecting the class that is most prevalent among the samples in that node.
Step 6. Tree Pruning (Optional), enhance the tree's performance by eliminating nodes that have little impact on accuracy, hence avoiding overfitting.
Step 7. Final Tree, the generated tree is the decision tree classifier used for making predictions.

3.3 The Proposed Combination of LightGBM and Decision Tree

The presented study synergistically leverages the capabilities of a Decision Tree Classifier and a LightGBM Classifier to improve predictive modeling. Firstly, a Decision Tree Classifier is created and trained using the provided training data. The decision tree acts as the first step for the later LightGBM model.

The LightGBM model is then setup with predetermined hyperparameters, including the boosting type, number of leaves, maximum depth, and a random seed. When training the LightGBM model, the starting model parameter is assigned as the decision tree classifier that was trained before. The initialization phase enables the LightGBM model to use the acquired patterns from the decision tree.

The merged LightGBM model is used to predict using a test set after model training. The expected classifications are then evaluated based on the classifications in the test dataset. The extent of correctness of the merged model is provided by the accuracy score.

The proposed technique in this study is a hybrid technique that subjective a decision tree's in-depthness to provide guidance was a motivator for using the LightGBM model. The LightGBM model is used due to its flexibility and efficiency, while the decision tree is used due to its decent explainability. Thus, the performance of the model in predictions is considerably high.

Step 1: Create an object of the Decision Tree Classifier class to instantiate a decision tree.

Step 2: Fit it with training data to train the decision tree classifier.

Step 3: Convert the Decision Tree easily usable by LightGBM.

- Instantiate a LightGBM Classifier with predetermined hyperparameters (boosting type, number of leaves, maximum depth, and random seed).
- Train the LightGBM Classifier using the training data.
- Utilize the Decision Tree Classifier, which has been trained, as the starting model for the LightGBM Classifier.

Step 4: Generate predictions on the test set. Utilize the trained LightGBM model to generate predictions on the test set.

Step 5: Evaluate Accuracy

- Assess the accuracy by comparing the predicted labels with the real labels of the test set.
- Determine the precision of the amalgamated model.
- Display the precision on the console.

3.4 Hyperparameters

Hyperparameters are essential in training machine learning models since they act as external configuration settings that direct the learning process. Contrary to the internal parameters of a model that are acquired during training, hyperparameters are predetermined by the data scientist or machine learning engineer prior to the commencement of training. They function as the control mechanisms that directly impact the behavior and efficacy of a machine learning system. The learning rate is a crucial hyperparameter that dictates the magnitude of the increments made throughout the optimization procedure. An elevated learning rate might result in fast convergence, but it runs the danger of surpassing the ideal solution, while a reduced learning rate may converge gradually or get trapped in local minima. Attaining the ideal model performance requires the precise calibration of several factors.

- **Boosting**

Boosting is an ensemble learning approach that combines the predictions of a number of weak learners into a strong and flexible model. Boosting also uses a learning rate, which is a critical hyperparameter. The learning rate dictates the constituent weak learners' relative influence in the strong model. The learning rate also determines the impact of every iteration in the progressive development procedure for the weak learners. A modest learning rate necessitates a large number of weak learners to achieve an adequate match. Nevertheless, the model may improve its diversity and predictive accuracy. A high learning rate promotes fast convergence to a fit weak learner but may contribute to substantial overfitting. Therefore, identifying the best learning rate is important to ensure there is an optimal tradeoff between model complexity and generalization.

In a boosting approach, the count of estimators, commonly referred to as the count of weak learners, is an important hyperparameter. In other words, it is the overall count of models that are combined in the ensemble. For most boosting models, the more estimators that are added to the model, the better the model's performance becomes as it approaches a particular maximum. Nevertheless, adding more estimators beyond this point leads to diminishing advantages and increasing overfitting risk. Thus, the count of estimators is one of the hyperparameters that require tuning in the boosting method as a learning ensemble. By tuning this hyperparameter, the boosting approach builds a better ensemble model that generalizes excellently to new data with no underfitting or overfitting issues.

- **Number of leaves**

The other essential hyperparameter for tree-based models is the "Number of leaves." The cycle of tree structure is recursive when tree-based models, including a decision tree , as well as gradient boosting implementations, such as XGBoost or LightGBM, split a dataset into subgroups due to the traits of its features. A terminal is occasionally called a leaf in a tree, and the hyperparameter "Number of leaves" specifies the maximum quantity of terminal nodes or leaves that the tree may have.

By modifying the "Number of leaves" hyperparameter, the model can self-regulate the tree's complexity, and subsequently evaluate the overall model's complexity. A model with many leaves will be complex and overfit, while one with a smaller amount of leaves will be simple and underfit. Therefore, it is paramount to achieve an optimal trade-off between the model's complexity and its ability to predict with respect to this hyperparameter. The multiple settings of the "Number of leaves" hyperparameter should be explored using the grid search, random search, or more sophisticated optimization algorithms, and the optimal setting should be selected. The criteria of the accuracy of the model's predictions on the validation dataset should be used to optimize this hyperparameter. In this process, it is crucial to weight underfitting and overfitting to ensure that the model indeed predicts novel, unknown data with as much accuracy as possible.

- **Maximum depth**

The term "maximum depth" is concerned with a hyperparameter vital in controlling the relative complexity of the decision tree models. Decision tree refers to a commonly used supervised learning technique applicable in both classification and regression tasks. The tree is made up of decisive points known as nodes and possibilities branches. The depth of a decision tree refers to the farthest number of edges that can exist between the root node and any child node . The hyperparameter sets a limit on how long the routes inside the decision tree can be, thus restricting the complexity and the number of nodes in it. Determining the right depth is a critical step in the adjustment to the ideal decision tree model for getting optimal output. A low value of the maximum depth results in high simplification of data patterns leading to underfitting and generating low accuracy predictions from the model.

On the other hand, if the maximum depth is very high, the model may also learn the irrelevant idiosyncrasies and differentiations from the training data, which may thereby cause the overfitting and eventually production of less general patterns that could be applied to the upcoming previously unexplored data. Therefore, finding the maximum depth with the best-fitted value is an essential notion in the hyperparameter tuning. In this technique, it is necessary to compromise between the model's complexity and its capability to determine the complex and complicated patterns in the dataset that could be generalized into precise predictions.

- **Random state**

The phrase "random state", which is an exceptional example of a hyperparameter in machine learning, can be singled out in the field of the latter. Hyperparameters are required, in particular, where the model is random or stochastic. Here the term "hyperparameter" stands for such a decision, which is obtained before the data and training. Random state is an external parameter in the model that determines the entrainment of the randomness during the training or testing.

Similarly, it is important to set a random state "whenever random tasks are performed" during an algorithm. That is, such tasks as the initialization of weights, the shuffling of training data,

and the division of data into training and testing sets must have a defined random state. More specifically, if the random state is set to a particular value, the generation of random integers will be performed in the same way. That is, the same sequence will be generated each time a user executes the algorithm numerous times under the same random state. That is important, particularly to facilitate replication of the study by other scholars or validation of the results by industry practitioners. Hence, random state configuration is an important aspect when working as a team on a project or sharing code, enhancing the transparency and reproducibility of machine learning experiments.

The combination of a Gradient Boosting Machine, LightGBM and a Decision Tree for crop selection approach can be regarded as an innovation in the scope of agricultural predictive modeling. This approach was developed as an attempt to harness the advantages of both models, resulting in a more robust and accurate crop selection system. It is unique mainly because a Decision Tree Classifier is intentionally introduced in the first step of the LightGBM model implementation. As a rule, a decision tree s characterized by a high degree of interpretability; therefore, it is particularly intriguing to investigate the patterns and decisions in the agricultural data application. In other words, the decision tree allows the model to grasp the explicit rules and linages present in the data, providing a more profound insight into the factors that affect the crop selection procedure.

However, LightGBM excels in handling extensive and intricate datasets, and it improves the model's performance via gradient-based optimization. Furthermore, the decision tree's interpretability is enhanced since it can effectively handle non-linear interactions and capture intricate patterns. The integration of these two models combines the interpretability of decision trees with the predictive ability of LightGBM. This solution integrates both components to effectively handle the intricacies and fluctuations of the crop recommendation system.

Finally, its learning and adaptation capabilities in diverse and dynamic agricultural environments make it suitable for crop recommendation. The LightGBM improves the model as it learns and continues to train the data, thereby enhancing and correcting the errors it makes during the training process. Specifically, this improves the accuracy and reliability of the model, an aspect vital in agriculture due to various seasonal and climatic variations. Therefore, the model is distinctive due to its ability to blend the accuracy existing in the Reflectivity Tree and the LightGBM's forecasting capabilities to form a modified model, hybrid. This makes the approach in crop recommendation unique, since, in addition to ensuring high performance, the model is also explainable. Therefore, decision-makers get an opportunity to see the model's performance and then make a decision based on the data pattern they understand properly.

4. Experimental Results

This section explains the results obtained from the simulations made using the recommended approach. The database that was used in this study was downloaded from Kaggle. A dataset that provides the user with the capability to create a forecasting model, which shows the types of crops that can be recommended to grow on a particular farm using various factors. This dataset was constructed by combining datasets of rainfall, climate, and fertilizer data that were already accessible for India. Figure 2, 3 and 4 shows the analysis of Nitrogen (N), Phosphorus (P) and Potassium (K) for crop recommendation respectively.

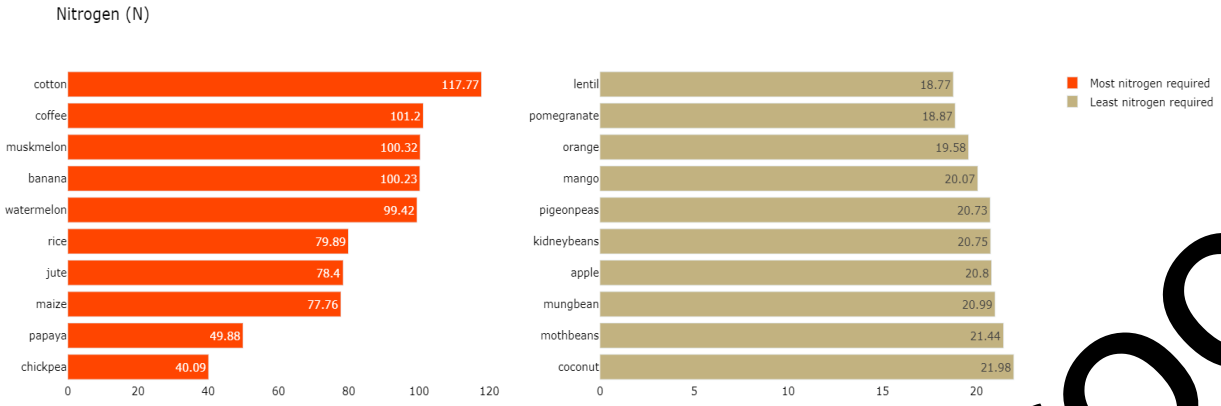


Figure 2: Nitrogen (N) Analysis for crop recommendation

The second picture illustrates the nitrogen needs for different crops in a graphic format. The image displays the crops that need the greatest quantities of nitrogen, with cotton ranking first at 117.77 and watermelon closely behind at 99.42. In contrast, the graph also emphasizes the crops that have the lowest nitrogen needs. Lentils exhibit the least nitrogen need, with a value of 18.77, whilst oranges necessitate a slightly higher amount of 19.58. Within the spectrum of these two extremes, the diagram showcases several types of crops and their corresponding levels of nitrogen needs. As an example, rice has a nitrogen demand of 79.89, which is considered to be in the mid-range. On the other hand, chickpeas have a more modest nitrogen requirement of 40.09. The graphic depiction successfully conveys the diverse nitrogen needs across various crops, offering significant insights into their nutritional prerequisites.

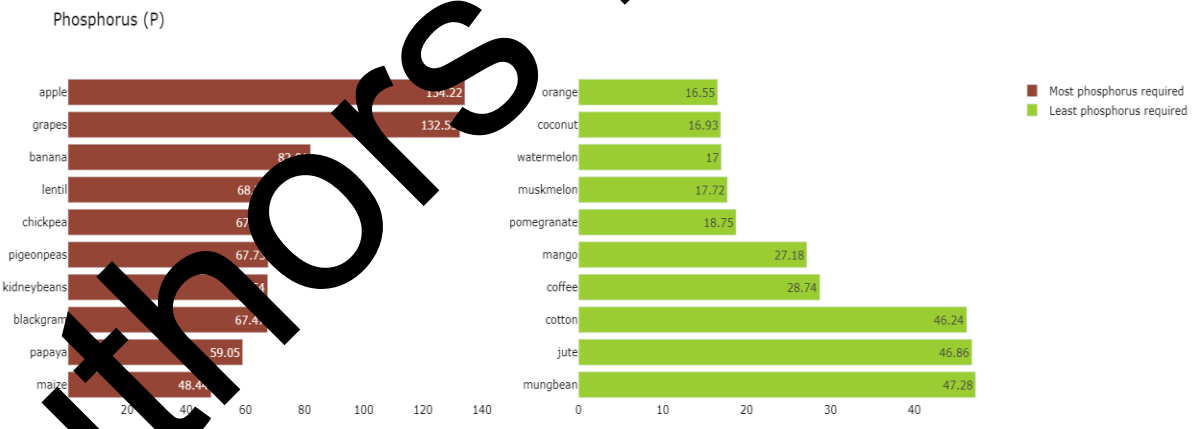


Figure 3: Phosphorus (P) Analysis for crop recommendation

Figure 3 depicts the phosphorus requirements of different crops, visually presenting the particular quantities of phosphorus (P) needed for each crop. Regarding the need for phosphorus, the data indicates the crops that need the highest amount of phosphorus, ranging from 134.22 for apples to 48.44 for maize. In contrast, the section focused on low phosphorus demand showcases crops that require very little phosphorus, with quantities ranging from 28.74 for coffee to 16.55 for oranges.

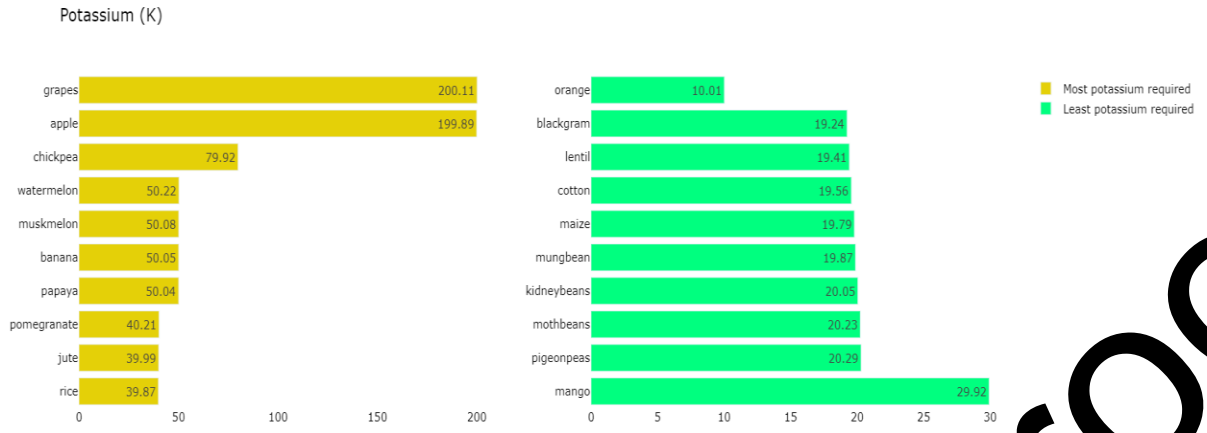


Figure 4: Potassium (K) Analysis for crop recommen[illegible]

Figure 4 illustrates the Potassium needs of several crops, graphically displaying the specific amounts of Potassium (K) necessary for each crop. The data reveals the specific crops that need the greatest quantity of Potassium, with grapes requiring the highest amount at 200.11 and rice requiring the lowest amount at 39.87. Conversely, the section that emphasizes low Potassium consumption highlights crops that necessitate less Potassium, ranging from 29.92 for mangoes to 10.01 for oranges.

Figure 5 shows N, P and K comparison for crops.

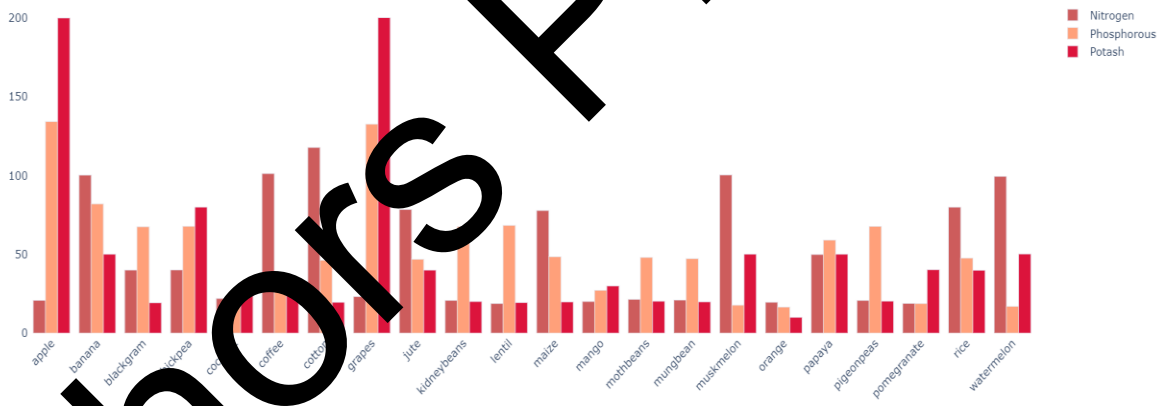


Figure 5: Nitrogen (N), Phosphorus (P) and Potassium (K) Comparison for crops

The provided diagram Figure 5 depicts the crop yields categorized according to their nitrogen (N), phosphorus (P), and potassium (K) needs.

Some crops have high demands for nitrogen (N) and produce large yields. Cotton, muskmelon, and coffee belong to this group. Meanwhile, grapes, bananas, and oranges have a modest need for nitrogen and produce matching yields. Lentil, kidney bean, and mungbean have modest nitrogen needs and yields at the lower end.

While considering phosphorus (P), the data shows that apple, banana, and maize need larger amounts of phosphorus, which leads to higher crop yields. Blackgram, lentil, and orange have intermediate phosphorus (P) needs and yields, while jute, watermelon, and mothbean exhibit the lowest P requirements and yields.

When studying the potassium (K) needs of crops, it is evident that some plants such as bananas, apples, and oranges have notable requirements and produce large yields. Maize, cotton, and grapes belong to the group of crops with modest potassium needs, which results in commensurate yields. In contrast, lentil, mungbean, and kidney bean have the lowest potassium needs and produce the lowest yields.

The graphic indicates a positive relationship between crop yields and the amount of nitrogen, phosphorous, and potassium needed for their growth.

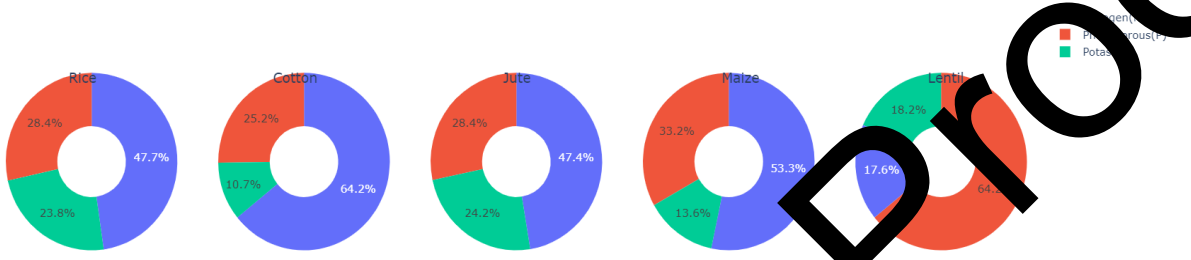


Figure 6: N, P, K Ratio for Rice, Cotton, Jute, Maize, Lentil

Figure 6 illustrates the specific crops (Rice, Cotton, Jute, Maize, and Lentil) and their corresponding requirements for nitrogen, phosphorous, and potassium in order to facilitate their development. Cotton necessitates 64.2% nitrogen, while Lentil necessitates 17.6% nitrogen. Lentil necessitates a phosphorous content of 64.2%, whereas cotton necessitates a phosphorous content of 25.2%. Jute necessitates 24.2% potassium, whereas cotton necessitates 10.7% potassium.

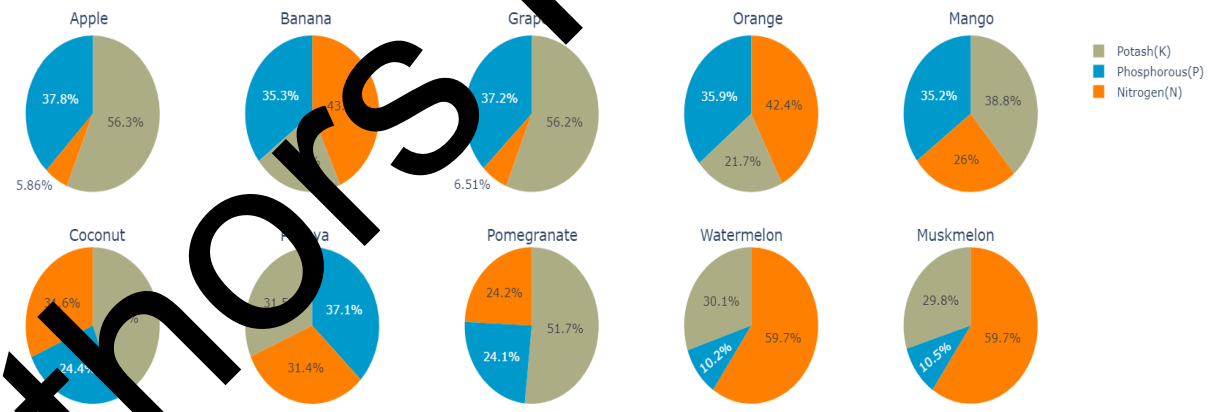


Figure 7: N, P, K Ratio for Fruits

Figure 7 shows N, P and K ratio for Fruits (Apple, Banana, Grapes, Orange, Mango, Coconut, Papaya, Pomegranate, Watermelon and Muskmelon). Apple requires 56.3% of potassium, Grapple requires 37.8% of phosphorous and Watermelon along with Muskmelon requires 59.7% of nitrogen.

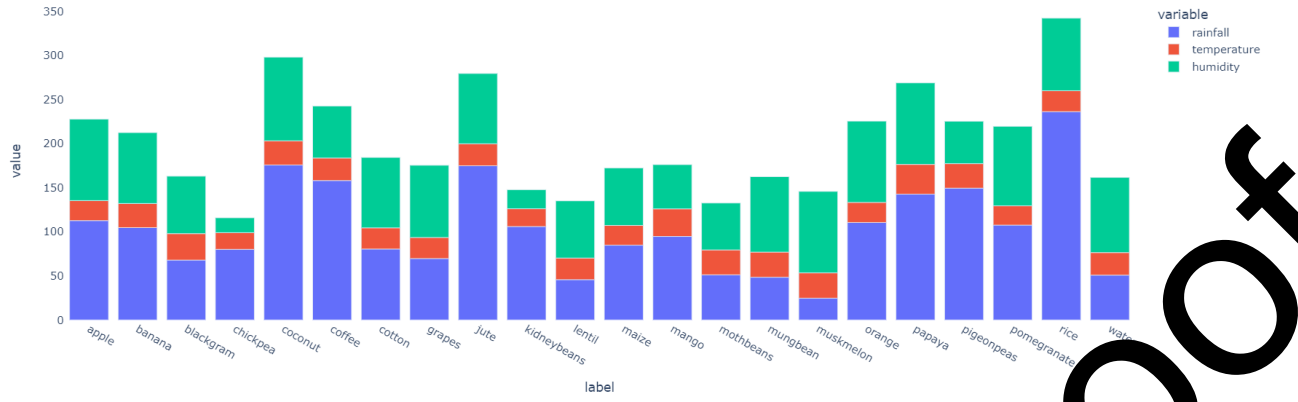


Figure 8: Comparison between Rainfall, Temperature and Humidity

Figure 8 shows Comparison between Rainfall, Temperature and Humidity for crops. The interaction between rainfall, temperature, and humidity is vital in influencing the outcome of crop production. Sufficient precipitation is crucial for providing water to crops, therefore guaranteeing optimal growth and development. The rate of photosynthesis, germination, and total plant metabolism is affected by temperature. Each crop has distinct temperature needs for maximum performance. The degrees of humidity affect the amount of water lost via transpiration and may have an effect on the occurrence of diseases.

Following the completion of the data preparation phase, the attention shifts towards the analysis of crucial agricultural components, namely Nitrogen, Phosphorous, and Potassium, customized for individual crops. Following that, a comprehensive examination of Rainfall, Temperature, and Humidity is carried out, together with crop suggestions. The suggested approach applies the knowledge received from this extensive investigation to providing significant outcomes, eventually enhancing informed decision-making in agriculture.

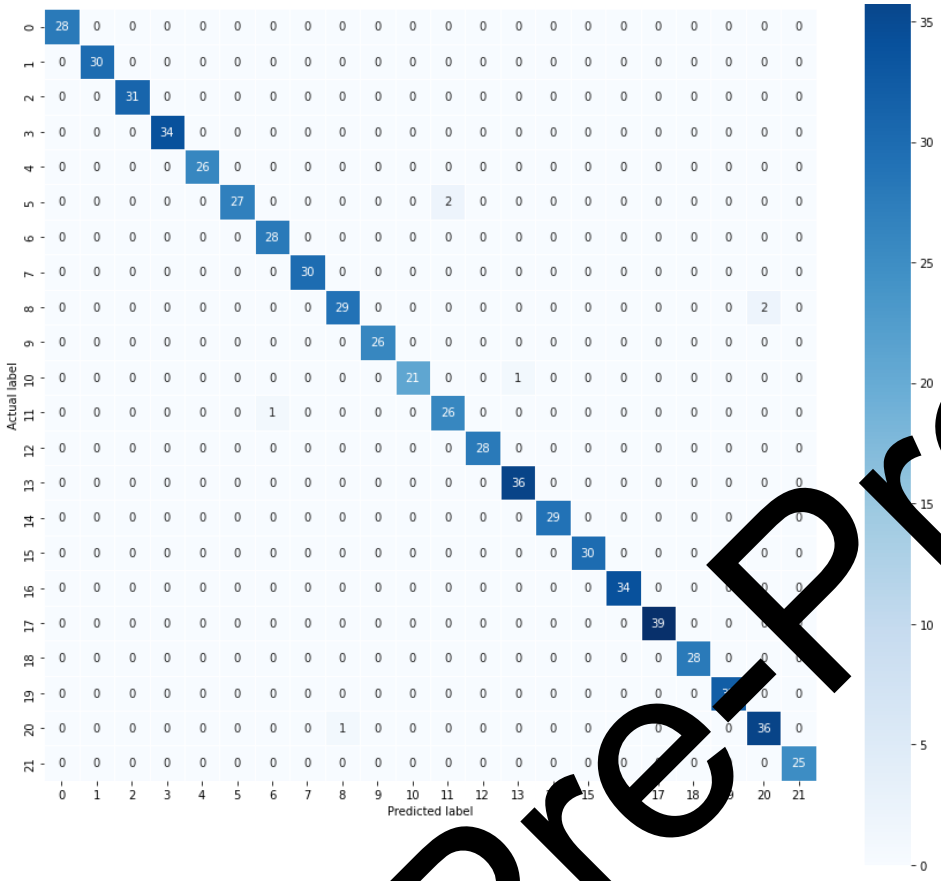


Figure 9: Confusion Matrix

Figure 9 shows the confusion matrix. Confusion matrix is a device in machine learning and data analysis that lets us evaluate how well we are doing in a classification issue. It is particularly useful in instances when a binary or a multi-class classification issue is explored. It is a methodical manner of presenting how the model's output's predicted and actual classification is arranged, shown in a tabular form. The confusion matrix contains four important components: True Positives, which is a situation where positive events are correctly predicted. True Negatives, which implies that negative results have been correctly predicted. False Positive when the model predicts incorrectly that the instance is positive. False Negative, when the model predicts correctly that the instance is negative. Using this information, other metrics, such as accuracy, precision, recall, F1 score, among others, can be calculated. These analyses will provide a detailed performance summary.

There are many other situations where inputs of the confusion matrix can be useful when evaluating a classification model. What is key to assess is how the model would be affected by imbalanced datasets, which is relevant to the learning-stage. As soon as a confusion matrix is set up, one can see what areas are performing poorly almost instantly, be that the fact of more false negatives or positives produced by the model. Metrics that can be computed based on a confusion matrix are, therefore, more informative as to how the model is actually performing, since the use of accuracy can be deceptive primarily when dealing with imbalanced datasets.

Table 1: Classification Report

	Precision	**Recall**	**F1-Score**

Apple	1.00	1.00	1.00
Banana	1.00	1.00	1.00
Blackgram	1.00	1.00	1.00
Chickpea	1.00	1.00	1.00
Coconut	1.00	1.00	1.00
Coffee	1.00	0.93	0.96
Cotton	0.97	1.00	0.98
Grapes	1.00	1.00	1.00
Jute	0.97	0.94	0.95
Kidneybeans	1.00	1.00	1.00
Lentil	1.00	0.95	0.98
Maize	0.93	0.96	0.95
Mango	1.00	1.00	1.00
Mothbeans	0.97	1.00	0.99
Mungbean	1.00	1.00	1.00
muskmelon	1.00	1.00	1.00
Orange	1.00	1.00	1.00
Papaya	1.00	1.00	1.00
Pigeonpeas	1.00	1.00	1.00
pomegranate	1.00	1.00	1.00
Rice	0.95	0.97	0.96
Watermelon	1.00	1.00	1.00

Table 4 shows a classification report; it is an extremely valid and mandatory method for determining the success of a classification model on more than two classes or groups. Multiple measures are used to confirm such a classification; Precision, Recall, and F1-Score, reporting on each class about that measure. Precision is a measure expressing the ratio of correctly predicted positive observations to the whole of the predicted positives. In the same fashion, Recall expresses the same ratio but with precision's denominator replaced by the real class' ratio. F1-Score essentially serves as the harmonic mean of the two previous measures; all of

these measures indicate how well the model predicts whether or not an example fits each of the classes.

As per the table shown in Table 1, shows the assessing the model's performance with correctly classifying various crops. Each data row specifies a crop; these include Apple, Banana, Blackgram, etc. The table columns are the exact Precision, Recall, and F1-Score columbns for every crop. Whenever a value of 1.00 appears, it implies that the model directly received an accuracy and Recall or F1-Score of 100 percent. This means that the model was truly exceptional in being able to distinguish examples of that class. Nonetheless, several crops have slightly lower figures, which factor in areas that need corrections. Coffee, Jute, Lentil, Maize, and Rice contain high discrepancies between precision, recall, and F1-Score, and perhaps these crops are going to be difficult to predict with maximum accuracy. Overall, the table provides a representation of the classification model's correct accuracy within each class and perhaps providing valid recommendations of problematic crops to predict.

Table 2: Comparative Analysis

Methods	**Accura**
KNN [18]	93%
Logistic Regression [19]	94%
LightGBM [20]	97%
LightGBM + Decision Tre **(Proposed)**	**98.64%**

A comparative examination of several approaches is provided in Table 2 to determine the three approaches' accuracy in a given assignment. The mentioned approaches are KNN or K-Nearest Neighbors, Logistic Regression, and LightGBM. The approaches' performance is determined in terms of accuracy, which is the percentage of true predictions from each model. Here, KNN has an accuracy of 93%, a high enough accuracy to detect patterns in the given data and make predictions but it could be higher when compared to the other models. Logistic Regression performs better as its accuracy is 94%, meaning that it can make better predictions of the correlation between the input data and the target variable.

The new approach in the model is shown at the end of the table. Figure 2 shows that the new approach is Light GBM added to a Decision Tree. The new approach has an accuracy of 98.64% which is the best adequate when compared to the existing approaches. This result means that the fusion of LightGBM to a Decision Tree will result in a model with a more accurate prediction of the target variable, as one can take advantage of the pros of both methods. The combined model's accuracy is higher than all the other models, reaching 98.64%.

5. Conclusion

The Automatic Crop Recommendation System will undoubtedly make significant contributions to maximizing agricultural productivity by recommending the most suitable

crops based on soil and environmental data. The Automatic Crop Recommendation System will increase efficiency and productivity of resource harvested. The technology will offer more value than the current methods and ensure farmers have convenience and human-harvested. In addition, it will contribute to improving sustainability and the efficiency of resource use in general. The Automatic Crop Recommendation System using the integration of LightGBM and Decision Tree has demonstrated excellent validity and performance in predicting precise crop recommendations. Results from the accuracy, recall, and f1-score indicate high performance and robustness in predicting the right kind of crops to plant. The test shows the system's ability to use machine learning. Machine learning has an accuracy of 98.64%, which is quite impressive compared to the three tests. Crop recommendation is an activity that can be accurately accomplished using machine learning. The combination of LightGBM and Decision Tree will not only improve the prediction of accuracy but also create a cost-effective, scalable method for crop recommendation.

References

1. Suruliandi, A., G. Mariammal, and S. P. Raja. "Crop prediction based on soil and environmental characteristics using feature selection techniques." *Mathematical and Computer Modelling of Dynamical Systems* 27, no. 1 (2021): 117-140.
2. Fraga, Helder, Joaquim G. Pinto, Francesco Viola, and João A. Santos. "Climate change projections for olive yields in the Mediterranean Basin." *International Journal of Climatology* 40, no. 2 (2020): 769-781.
3. Balaska, Vasiliki, Zoe Adamidou, Yiasis Vryzas, and Antonios Gasteratos. "Sustainable crop protection via robotics and artificial intelligence solutions." *Machines* 11, no. 8 (2023): 774.
4. Issad, Hassina Ait, Rachida Aoudjit, and Joel JPC Rodrigues. "A comprehensive review of Data Mining techniques in smart agriculture." *Engineering in Agriculture, Environment and Food* 12, no. 4 (2019): 511-525.
5. Weselek, Axel, Andrea Ehmerle, Jens Hartung, Sabine Zikeli, Iris Lewandowski, and Petra Högy. "Agrivoltaic system impacts on microclimate and yield of different crops within an organic crop rotation in a temperate climate." *Agronomy for Sustainable Development* 41, no. 5 (2021): 59.
6. Rani, Sita, Amit Kumar Mishra, Aman Kataria, Saurav Mallik, and Hong Qin. "Machine learning based optimal crop selection system in smart agriculture." *Scientific Reports* 13, no. 1 (2023): 15997.
7. Venapally, Murali Krishna, Abhishek Ray, and Neelamadhab Padhy. "IoT-Enabled Soil Nutrient Analysis and Crop Recommendation Model for Precision Agriculture." *Computers* 12, no. 3 (2023): 61.
8. Rakhimov, Nizom, Khilola Amankulova, József Szatmári, Alireza Sharifi, Dariush Abbasi-Moghadam, Seyed Mahdi Mirhoseini Nejad, and László Mucsi. "Crop type classification by DESIS hyperspectral imagery and machine learning algorithms." *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 16 (2023): 1576-1588.
9. Sawant, Ankit R., Yash Sivramkrishnan, Hemish Ganatra, and Ameya A. Kadam. "Smart Crop–Precision Agriculture using Machine Learning." *International Journal of Research in Engineering, Science and Management* 6, no. 10 (2023): 13-17.

10. Shaikh, Tawseef Ayoub, Tabasum Rasool, and Faisal Rasheed Lone. "Towards leveraging the role of machine learning and artificial intelligence in precision agriculture and smart farming." *Computers and Electronics in Agriculture* 198 (2022): 107119.
11. Abdallah, Emna Ben, Rima Grati, and Khouloud Boukadi. "A machine learning-based approach for smart agriculture via stacking-based ensemble learning and feature selection methods." In *2022 18th International Conference on Intelligent Environments (IE)*, pp. 1-8. IEEE, 2022.
12. Bakthavatchalam, Kalaiselvi, Balaguru Karthik, Vijayan Thiruvengadam, Sriram Muthal, Deepa Jose, Ketan Kotecha, and Vijayakumar Varadarajan. "IoT framework for measurement and precision agriculture: predicting the crop using machine learning algorithms." *Technologies* 10, no. 1 (2022): 13.
13. Gupta, Akanksha, and Priyank Nahar. "Classification and yield prediction in smart agriculture system using IoT." *Journal of Ambient Intelligence and Humanized Computing* 14, no. 8 (2023): 10235-10244.
14. Rakhra, Manik, Sumaya Sanober, Noorulhasan Naveed Quadri, Neha Verma, Samrat Ray, and Evans Asenso. "Implementing machine learning for smart farming to forecast farmers' interest in hiring equipment." *Journal of Food Quality* 2022 (2022).
15. Priyadharshini, A., Swapneel Chakraborty, Aayush Kumar, and Omen Rajendra Pooniwala. "Intelligent crop recommendation system using machine learning." In *2021 5th international conference on computing methodologies and communication (ICCMC)*, pp. 843-848. IEEE, 2021.
16. Pande, Shilpa Mangesh, Prem Kumar Ramesh, ANMOL ANMOL, B. R. Aishwarya, KARUNA ROHILLA, and KUMAR SHAURYA. "Crop recommender system using machine learning approach." In *2021 5th international conference on computing methodologies and communication (ICCMC)*, pp. 1066-1071. IEEE, 2021.
17. Yang, Shuting, Lingjia Gu, Xiaofeng Li, Fang Gao, and Tao Jiang. "Fully automated classification method for crops based on spatiotemporal deep-learning fusion technology." *IEEE Transactions on Geoscience and Remote Sensing* 60 (2021): 1-16.
18. Chakraborty, Soham, and Sushruta Mishra. "A Smart Farming-Based Recommendation System Using Collaborative Machine Learning and Image Processing." In *Cognitive Informatics and Soft Computing: Proceeding of CISC 2021*, pp. 703-716. Singapore: Springer Nature Singapore, 2022.
19. Banerjee, Saikat, and Abhoy Chand Mondal. "A Region-Wise Weather Data-Based Crop Recommendation System Using Different Machine Learning Algorithms." *International Journal of Intelligent Systems and Applications in Engineering* 11, no. 3 (2023): 283-297.
20. Choudhry, Rashmi, Vinay Rishiwal, Preeti Yadav, Kaustubh Ranjan Singh, and Mano Yadav. "Automatic Smart Irrigation Method for Agriculture Data." In *Towards the Integration of IoT, Cloud and Big Data: Services, Applications and Standards*, pp. 57-73. Singapore: Springer Nature Singapore, 2023.