Smart Farming: Crop Recommendation using Machine Learning with Challenges and Future Ideas

Devendra Dahiphale, Pratik Shinde, Koninika Patil, and Vijay Dahiphale

***Abstract*—Crop analysis and prediction is a rapidly growing field which is vital in optimizing agricultural practices. Crop recommendation is pivotal in agriculture, empowering farmers to make informed decisions about the most suitable crops for their land and climate conditions. Traditionally, this process heavily relied on expert knowledge, which proved time-consuming and labor-intensive. Moreover, considering the projected global population of 9.7 billion by 2050, the need to produce more food sustainably becomes imperative. Machine learning techniques can play a crucial role in effectively automating crop recommendations and detecting pests and diseases to enable farmers to optimize their yield from the land while simultaneously maintaining soil fertility and replenishing essential nutrients. This paper analyses the performance of crop recommendation across seven distinct machine-learning algorithms. The proposed system leverages various features, including soil composition and climate data, to accurately predict the most suitable crops for specific locations. This system has the potential to revolutionize crop recommendation, benefiting farmers of all scales by enhancing crop yields, sustainability, and overall profitability. Through extensive evaluation of a comprehensive historical data set, we have achieved near-perfect accuracy by training and testing models the machine learning algorithms with various configurations. We demonstrate accuracy consistently over 95% across all models, with the highest achieved accuracy reaching 99.5%.**

***Impact Statement*—This work presents machine-learning models for crop recommendation that achieves near-perfect accuracy. The system leverages various features, including soil composition and climate data, to predict the most suitable crops accurately. This system has the potential to revolutionize crop recommendation, benefiting farmers of all scales by enhancing crop yields, sustainability, and overall profitability. Some of the potential impacts include increased crop yield, improved sustainability, increased profitability, improved decision-making, and avoiding dependency on experts for crop recommendation. We believe that this system has the potential to revolutionize crop recommendations and help to ensure a sustainable food supply for the future. The world population is nearing 8 billion, and we all depend on agriculture for food, so ensuring that our agricultural systems are sustainable and resilient is essential. An end-to-end system can be built using our models, and additionally, farmers' surveys can be taken for impacts in terms of numbers which is one of the future scopes for this manuscript.**

***Index Terms*—Machine Learning, Prediction, Data Analysis, Recommendation, Big Data, Agriculture, Crop, Food, Environmental Factors, Agricultural Productivity.**

Devendra Dahiphale was with the University of Maryland Baltimore County, Baltimore, MD 21250, USA. (e-mail: devendr1@umbc.edu)

Pratik Shinde was with the University of Maryland Baltimore County, Baltimore, MD 21250, USA. (e-mail: pratiks1@umbc.edu)

Koninika Patil was with the University of Maryland Baltimore County, Baltimore, MD 21250, USA. (e-mail: koni1@umbc.edu)

Vijay Dahiphale was with Pune Institute of Computer Technology, Pune, India. (e-mail: vijaydahiphale96@umbc.edu)

This paragraph will include the Associate Editor who handled your paper.

I. Introduction

MACHINE learning [1] [2] is a field of study that gives computers the ability to learn without being explicitly programmed, a definition by Arthur Samuel (1959). Machine learning algorithms [3] are trained on large amounts of data to make predictions or decisions.

Agriculture, being a major sector worldwide, requires farmers to cultivate profitable and sustainable crops. Not choosing the right crop can have a significant impact on crop yield, leading to decreased productivity and potential financial losses for farmers. When farmers fail to consider crucial factors such as climate suitability, soil conditions, and market demand, the chosen crops may struggle to thrive and achieve their full yield potential. Unsuitable crops may suffer from inadequate adaptation to the local climate, resulting in poor growth, increased vulnerability to pests and diseases, and reduced overall yield. Moreover, crops that do not align with market demand may face difficulties in finding buyers or fetching favorable prices, further exacerbating the economic impact on farmers. By leveraging machine learning-based crop recommendation systems, farmers can mitigate these challenges and make informed decisions to maximize crop yield and ensure long-term agricultural viability.

Machine learning and agricultural data converge to revolutionize how farmers understand and optimize their practices. With the increasing availability of data from sources such as weather stations, satellites, sensors, and farm equipment, machine learning algorithms can analyze vast amounts of information and extract valuable insights. These algorithms can uncover complex patterns, correlations, and predictive models that were previously hidden within the data. By combining machine learning techniques with agricultural data, farmers gain the ability to make data-driven decisions, ranging from crop selection and irrigation management to pest control and yield prediction. This integration empowers farmers to achieve higher efficiency, resource optimization, and sustainable practices, ultimately leading to improved productivity and profitability in the agricultural sector.

Crop recommendation systems can be used to analyze a variety of data, such as weather data, soil data, and market data. This data can be used to train machine learning models to predict which crops will likely be successful in a given location. Crop recommendation systems can also inform farmers about the best practices for growing specific crops.

The development of crop recommendation systems using machine learning has the potential to improve the productivity and sustainability of agriculture. By helping farmers to choose

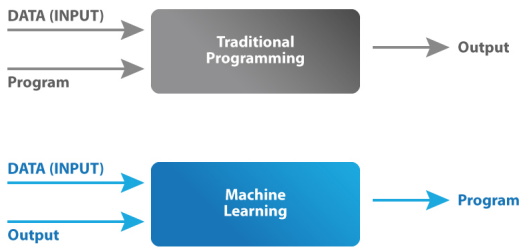

Fig. 1: Traditional Programming vs. Machine Learning

suitable crops to grow, crop recommendation systems can help to increase crop yields and reduce the use of resources. Additionally, crop recommendation systems can help improve agriculture's resilience to climate change [4]. Furthermore, machine learning can address several other challenges [5] in agriculture, for example, predicting crop yield, identifying pests and diseases, optimizing crop production, improving water efficiency, reducing the use of pesticides and fertilizers, soil management, etc.

Crops are a significant source of food and fiber for the world's population. World Resource Institute is trying to solve the problem of how to feed ten billion people sustainably by 2050. Therefore, increasing high-quality crop yield is very important. The choice of crops to plant can significantly impact crop yields and profitability. Climate change and other environmental factors make it tough to predict which crop will succeed, given the location.

In this paper, we use machine learning to recommend crops to farmers. First, collect the dataset and preprocess it. Then, we train and test models using features such as soil content and type, soil pH value, temperature, humidity, and rainfall. We also attempted feature engineering concepts to verify if the model performs better using a combination of different features and use it as a new feature in the same dataset. Agriculture has general challenges, and in the context of machine learning, therefore, we highlight these challenges thoroughly. Eventually, we present some exciting ideas for the readers to venture into.

II. Background Survey

A. Machine Learning

Machine learning gives computers the ability to learn without being explicitly programmed. In other words, machine learning is turning things or data into numbers and finding patterns in those numbers. The identified patterns help in predicting output for new data points. The fundamental difference between traditional programming and machine learning is shown in Figure 1. Traditional programming and machine learning are two different approaches to solving problems. Traditional programming involves writing code that defines the steps that the software should take to solve the problem. On the other hand, machine learning involves training a model on data so that the model can learn to solve the problem on its own. Machine learning algorithms are primarily categorized into three types based on how machines learn.

1) Supervised Learning: Models are trained on labeled data in supervised machine [2] [6] learning. This means that the data has been tagged with the correct output. The model then learns to predict the outcome for new data that has not been labeled. Several supervised machine learning algorithms exist, such as decision trees, Logistic regression, support vector machines, and neural networks.

2) Unsupervised Learning: Unsupervised learning [2] [6] is a type of machine learning where the model is trained on a set of unlabeled data. This means that the data does not have any labels associated with it. The model then learns to find patterns in the data and to group similar data points. Some examples of unsupervised learning are k-means clustering, hierarchical clustering, apriori algorithm, principal component analysis, etc.

3) Reinforcement Learning: Reinforcement learning (RL) [2] [6] is a type of machine learning that allows an agent to learn how to behave in an environment by trial and error. The agent receives rewards for actions that lead to desired outcomes and punishments for actions that lead to undesired results. Over time, the agent learns to take actions that maximize its rewards. The algorithms such as q-learning, policy gradients, and actor-critic fall into the category of reinforcement learning.

B. Machine Learning Algorithms Used

Although many machine learning algorithms are commonly used, we are only highlighting the following algorithms in this survey because they are the ones that we used in our study.

1) Logistic Regression: Logistic regression [3] [7] [8] is a statistical method that predicts the probability of an event occurring. It is a type of regression analysis that is used to model the relationship between one or more independent variables and a categorical dependent variable.

2) Decision Tree: A decision tree [3] [9] is a supervised learning algorithm that uses a tree-like structure to represent the relationship between the input and output data. A decision tree is made up of nodes and branches. The nodes represent decisions, and the branches represent the possible outcomes of those decisions.

3) Random Forest: A random forest [3] [10] is an ensemble learning algorithm comprising a collection of decision trees. Random forests are created by training many decision trees on different subsets of the training data. Each decision tree is trained using a random subset of the features. To make prediction, each decision tree in the random forest makes a prediction. The final prediction is made by taking the majority vote on the predictions from the individual decision trees. Random forests are often used for classification and regression tasks.

4) K-Nearest Neighbours: The k-nearest neighbors (KNN) [3] [11] algorithm is a supervised learning algorithm. KNN works by finding the k most similar neighbors in training set to a new input instance and then predicting the label of the new input instance based on the labels of the k nearest neighbors. This algorithm can be used for both classification and regression problems.

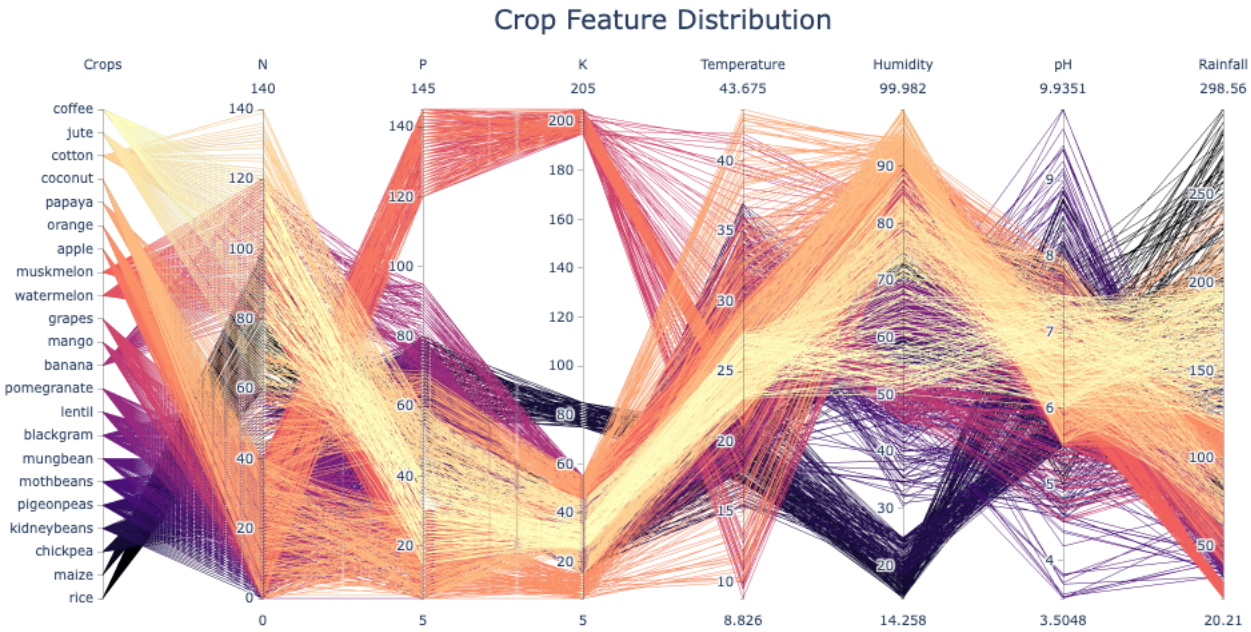

Fig. 2: Crop Features Distribution

5) Naive Bayes: The naive Bayes algorithm [3] [12] is a supervised learning algorithm that uses Bayes' theorem. It is a simple and versatile algorithm that can be used for various tasks, such as spam filtering, text classification, and medical diagnosis. The naive Bayes algorithm works by assuming that the presence of a particular feature in a class is unrelated to the presence of any other feature. This assumption is not always true. Therefore, it is called "naive". However, it is a good approximation in many cases, making the algorithm very simple to train and interpret. To classify an object, the Naive Bayes algorithm first calculates the probability of each class. It then calculates the probability of each feature given to each class. The class with the highest probability is the class the object is assigned to.

6) SVM: A support vector machine (SVM) [3] [13] [14] is a supervised learning algorithm. SVMs are based on finding a hyperplane that separates the data into two classes. The hyperplane is chosen to maximize the distance between the hyperplane and the closest data points on either side. This algorithm can be used for both classification and regression tasks.

7) Neural Network: The human brain inspires a neural network [3] [15] [16]. It is a network of interconnected (also called edges or connections) neurons called nodes. Neural networks are made up of multiple layers of nodes. The first layer of neurons is called the input layer, whereas the last layer is called the output layer. The layers in between are referred to as hidden layers. Each neuron in a neural network has a number of inputs and a single output. The inputs to a neuron are the outputs of the neurons in the previous layer. The result of a neuron is calculated using a function called an activation function [17]. The activation function is a non-linear function that transforms the input to the neuron into an output. The most common activation function is the sigmoid [2], [17] function. However, we can provide a custom activation function based on our requirements.

C. Machine Learning vs. Big Data Processing Frameworks

Machine learning and big data processing frameworks like MapReduce [18], [19] are powerful tools that can be used to analyze large datasets. However, they have different strengths and weaknesses.

In some cases, machine learning, and big data processing frameworks can be used together. For example, machine learning models can be created using machine learning and then used by big data processing frameworks to generate final results.

Machine learning models are trained on large datasets and can then be used to make predictions or decisions without human intervention. Big data processing frameworks, on the other hand, are designed to process large datasets quickly and efficiently. They are often used to process data for tasks such as data mining, data warehousing, large graph processing, and analytics [20], [21].

We need a tool that is accurate, fast, scalable, and easy to use. Therefore, we decided to use machine learning. Machine learning can be used directly to make predictions, while big data processing frameworks require additional tools to make predictions on top of the data processing framework.

D. Existing Research in Crop Recommendation

In the last few years, there have been slight increases [22] in research in the field of crop recommendation. For

example, Priyadharshini A et al. (2021) present "Intelligent Crop Recommendation System" [23], Zeel Doshi et al. (2018) present a system called AgroConsultant [24], SM Pande, et al. (2021), in their paper [25] proposes a viable and user-friendly yield prediction system for farmers, RK Rajak et al. (2017), the paper [26] proposes a model with a majority voting technique using a support vector machine (SVM) and ANN as learners to recommend a crop, Reddy et al. in their paper [27] present a survey of the existing techniques for crop recommendation, Ghadge et al. in their paper [28] present a theory on the crop recommendation, Kulkarni et al. in their research paper [29] showcase the work on improving crop productivity through a crop recommendation system using ensembling technique; Pudumalar et al., in their paper [30] (most cited on IEEE Xplore) present a similar approach using machine learning on data collected from a district in Tamil Nadu, India; however, the paper does not talk about models' accuracy or have not described data used. Another famous review on "Machine Learning in Agriculture" [31] by Konstantinos G. Liakos et al. covers most of the applications of machine learning in agriculture, however, it does not mention anything on crop recommendation. There is some other agricultural-related literature that is indirectly related to crop recommendation, for example; Ayaz Muhammad, et al. in their work [32] mainly talks about the Internet of Things and sensors for collecting agricultural data. A few other kinds of crop-related literature are about predicting crop yield. Our survey suggests little research on crop recommendation; much of the above-referred literature is from the last four to five years. We believe this could be because of inherent challenges in the field of the agriculture sector (which are presented in the section VI.), in addition to the difficulties related explicitly to machine learning in agriculture.

III. Our Contribution

Although the existing literature, primarily covered in section II-D, provides a good foundation for the research topic on crop recommendation models, they have some limitations. For example, many authors do not comprehensively overview their research process. This includes not mentioning their dataset sources, the accuracy of their models, or how their models were trained and tested. Additionally, much of the research lacks implementation details and does not specify the features used. Finally, many manuscripts only present surveys or theoretical work on crop recommendation topics.

Our paper addresses these limitations by developing comprehensive crop recommendation models. We describe each step of our process in detail, including our data collection, feature engineering, model training, and evaluation. We show that our system has the highest accuracy of any crop recommendation model in the literature. We achieved this by conducting feature engineering, which transforms the data to make it more useful for machine learning algorithms. We also analyzed data using seven different machine learning algorithms and with different configurations to achieve the highest accuracy, keeping these models' performance in mind.

All in all, first, we preprocess the data. Further, we apply several machine learning algorithms for recommending a crop. We train models using the following algorithms and find and compare the accuracy of each of the models for the recommendation system. Moreover, we try various configurations for each model to achieve better performance and accuracy:

1) Logistic Regression.
2) Decision Tree.
3) Random Forest.
4) K-Nearest Neighbours.
5) Naive Bayes.
6) SVM.
7) Neural Network.

Second, we address the limitations of the papers highlighted in the section II-D. We present a comprehensive crop recommendation system with all the details.

Third, we highlight the challenges in the agriculture sector, both in general and in the context of applying machine learning techniques to agricultural data.

Finally, we present multiple good ideas as future work for our work. The list of future work items is stated in the section VII at the end of the manuscript.

We believe that our work significantly contributes to the field of crop recommendation. Our comprehensive approach to the problem, high accuracy, and attempts at feature engineering are all novel contributions. We believe that our work will be helpful to farmers, agricultural researchers, and other stakeholders in the farming sector.

IV. Data Description, Methodology, and Experimentation

A. Data Description

We are preprocessing an existing dataset from Kaggle [33] for our model. Table I shows the main features of the data, and the first few rows of the data are shown in Table II. Figure 4 and 5 show the pictorial representation of the features and their count. Figure 3 shows the pairplot for the features used. A pairplot is a type of statistical graph that shows the relationships between multiple features in the form of a matrix in a dataset, with each row and column representing a different variable. The plots in the matrix's diagonal show each variable's distribution, while the plots in the off-diagonal show the relationships between pairs of variables. This dataset was created by augmenting datasets of rainfall, climate, and fertilizer data available for India.

We used a total of 22 unique labels for the data used are listed in Table III.

These labels are extracted from a database of around 100k records; because there is one good crop for a given setting, the records were reduced to approximately 2.2k.

B. Methodology

This section presents an overview of the methodology pictorially in Figure 6 that we have used to train various models. First, we iterated all the following steps with all the selected machine learning algorithms listed in Section III.

Index	*FeatureName*	*FeatureDescription*
1	Nitrogen (N)	Nitrogen is largely responsible for the growth of leaves on the plant.
2	Phosphorus (P)	Phosphorus is largely responsible for root growth and flower and fruit development.
3	Potassium (K)	Potassium is a nutrient that helps the overall functions of the plant perform correctly.
4	Temperature	temperature in degree Celsius
5	humidity	relative humidity in %
6	pH	pH value of the soil
7	rainfall	rainfall in mm

TABLE I: Main Features

Index	*N*	*P*	*K*	*Temperature*	*Humidity*	*Ph*	*Rainfall*	*Label*
0	90	42	43	20.879744	82.002744	6.502985	202.935536	rice
1	85	58	41	21.770462	80.319644	7.038096	226.655537	rice
2	60	55	44	23.004459	82.320763	7.840207	263.964248	rice
3	74	35	40	26.491096	80.158363	6.980401	242.864034	rice
4	78	42	42	20.130175	81.604873	7.628473	262.717340	rice

TABLE II: First 5 Rows of Data

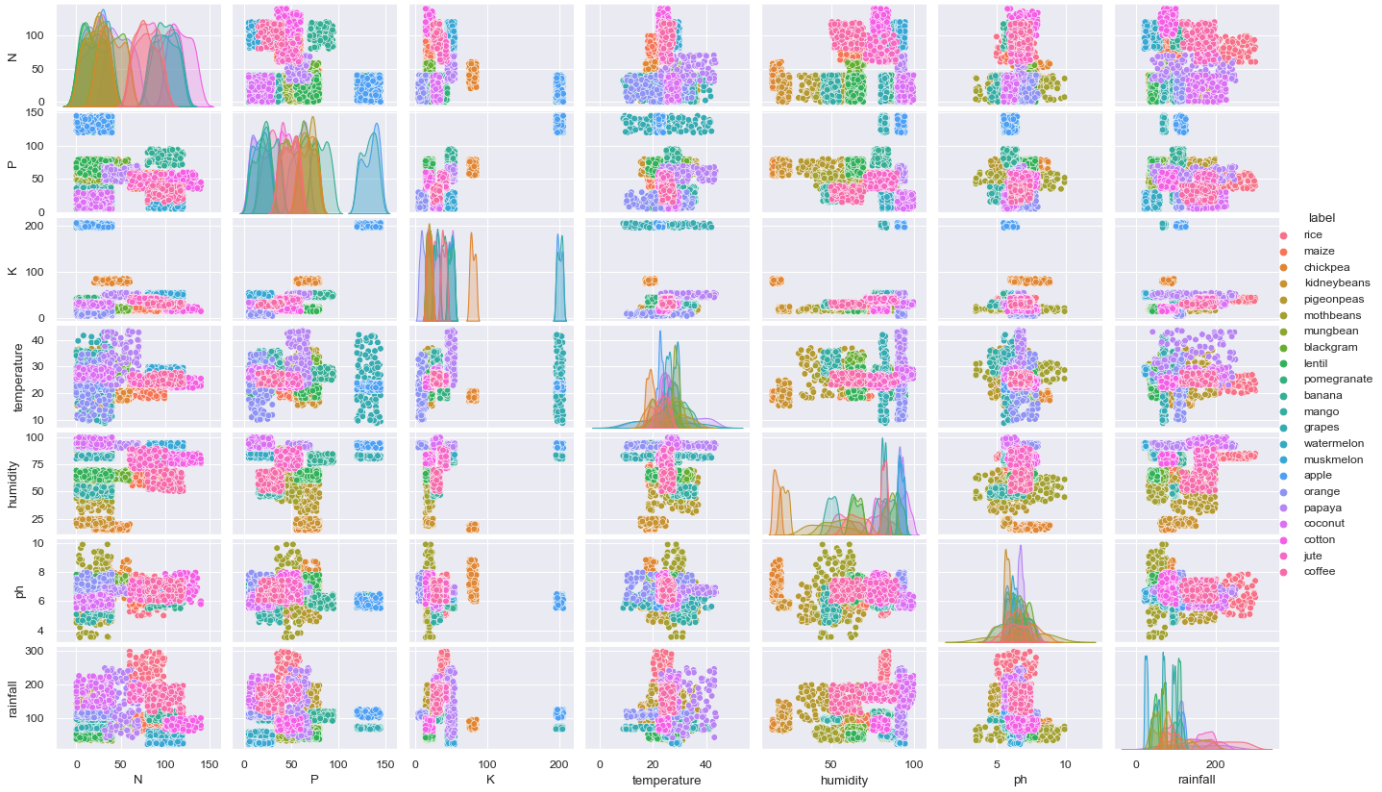

Fig. 3: Pair Plotting of All Data

1) Input Data: Because the quality and quantity of the data significantly impact a model's accuracy, we ensured the data was clean and well-labeled. As shown in Figure 6, the input to the system is a combination of soil and environmental characteristics. Table II shows a sample of raw data we used to train and test our models.

2) Preprocessing: We cleaned the data, removed outliers, and transformed the data into a format that your machine-learning algorithm could understand. We primarily removed all null and duplicate records, segregated features from the label column, creating new features from existing features (also called feature engineering), and described & plotted all the data to ensure no outliers.

3) Choose a machine learning algorithm: In each iteration, we chose one of the seven algorithms we had decided to use. For every selected algorithm, we iterated steps from preprocessing to testing or validating the model to tune the model.

4) Model Configurations: To achieve higher test and cross-validation accuracy, we used various configurations such as activation function, epoch, decision tree depth, and a number of nearest neighbors. Figure V shows some other important

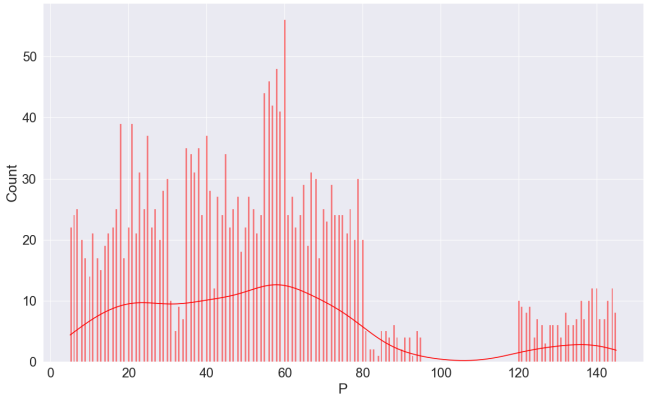

Fig. 4: Feature Graph for Temperature

Index	*LabelName*
1	rice
2	maize
3	chickpea
4	kidneybeans
5	pigeonpeas
6	mothbeans
7	mungbean
8	blackgram
9	lentil
10	pomegranate
11	banana
12	mango
13	grapes
14	watermelon
15	muskmelon
16	apple
17	orange
18	papaya
19	coconut
20	cotton
21	jute
22	coffee

TABLE III: All Unique Labels

configurations. Note that we have to keep in mind the performance of the model also; for example, increasing the decision tree's depth could cause to degrade the performance of the mode; Similarly, increasing the value of the number of time data should be fed to the neural network model will also highly impact the performance of the model.

5) Trainning Models: This is where the machine learning algorithm learns from the data prepared in the "Preprocessing" step.

6) Testing Accuracy of the Model: We evaluate the accuracy of the created model against the test data. In addition, we measured the cross-validation accuracy of the model. If the accuracy is unsatisfactory, we iterate the process by returning to the "Model Configuration" step. In some instances, we experimented with the feature engineering approach. Suppose the model's accuracy and performance are good at this step. In that case, we return to choosing a new algorithm step to repeat the same procedure with another advanced machine learning algorithm.

C. Experimentation

1) Model using Multi-Class Neural Network: A multi-class neural network can be used to classify data into multiple classes. This is in contrast to a single-class neural network, which can only classify data into one category. We created a four-layered neural network using the TensorFlow [34] framework; the listing 1 shows one of the examples. The first layer (input layer) contains thirty neurons, the second twenty neurons, the third includes ten neurons, and the fourth layer (output layer) includes twenty-two neurons. The second and third layers are called hidden layers. In addition, we experimented with different combinations of "relu", "softmax", and "sigmoid" activation functions [35], [36] to tune the model for better accuracy and performance. Finally, we experimented with the network with multiple epoch [37] values until we found the optimal ones. Increasing epoch value decreases the performance of the neural network.

Furthermore, we used $CategoricalCrossentropy$ as a loss function [15], [38], called categorical crossentropy. It is used for training multi-class classification models. It measures the distance between the predicted probabilities and the actual labels. The lower the categorical crossentropy, the better the model is performing.

Finally, we use an optimizer called Adams [39] optimizer. Adam is a popular optimization algorithm for training deep learning models. It is an extension of the AdaGrad [40] and RMSProp [41] algorithms, and it is effective for a wide range of problems.

Listing 1: Neural Network

```
import tensorflow as tf

model = tf.keras.models.Sequential([
    tf.keras.layers.Dense(30, activation = 'relu',
        input_shape = (7,)),
    tf.keras.layers.Dense(20, activation = 'relu'),
    tf.keras.layers.Dense(10, activation = 'relu'),
    tf.keras.layers.Dense(labels_count, activation =
        'softmax')
])

model.compile(
    loss = tf.keras.losses.CategoricalCrossentropy(),
    optimizer = tf.keras.optimizers.Adam(),
    metrics = ['accuracy']
)

model.fit(x_train, y_train, epochs = 60,
    validation_data = (x_test, y_test), batch_size =
    32)
```

2) Rest of the Models: All models except those using neural networks were created, trained, tested, and validated similarly. We used the classifiers listed in Section IV to build models with different algorithms. Some notable differences are presented in this section.

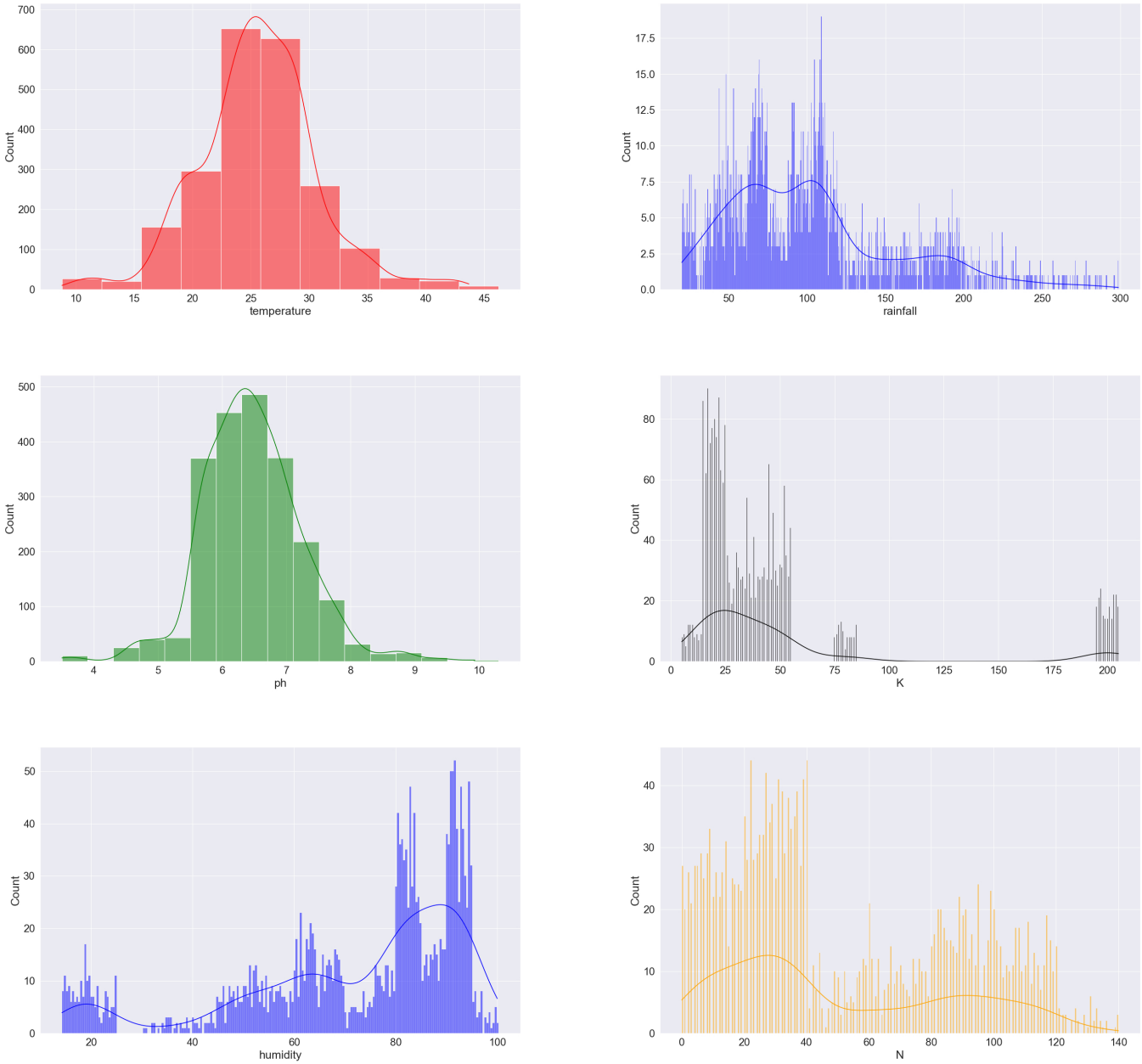

Fig. 5: Feature Graphs

Index	*ModelName*
1	$LogisticRegression(...)$
2	$DecisionTreeClassifier(...)$
3	$RandomForestClassifier(...)$
4	$KNeighborsClassifier(...)$
5	$GaussianNB(...)$
6	$svm.SVC(...)$

TABLE IV: Primary Classifiers Used

For the decision tree algorithm, we used *Gini* and *entropy* [42], two impurity measures used in decision trees. We also used max depth for the decision tree as another parameter.

For the k-nearest neighbors algorithm, we tried a configuration called $n_neighbors$, which determines the total number of nearest outputs that should be considered.

For SVM, we used *kernel* configuration. Kernel machines are a class of algorithms for pattern analysis, and their best-known member is the support-vector machine (SVM).

Furthermore, We evaluated and cross-validated models with the functionality from listing 2.

Listing 2: Model Evaluation

```
kfold = KFold(number_splits=5, shuffle=True,
    random_state=42)

def evaluate(my_model):
```

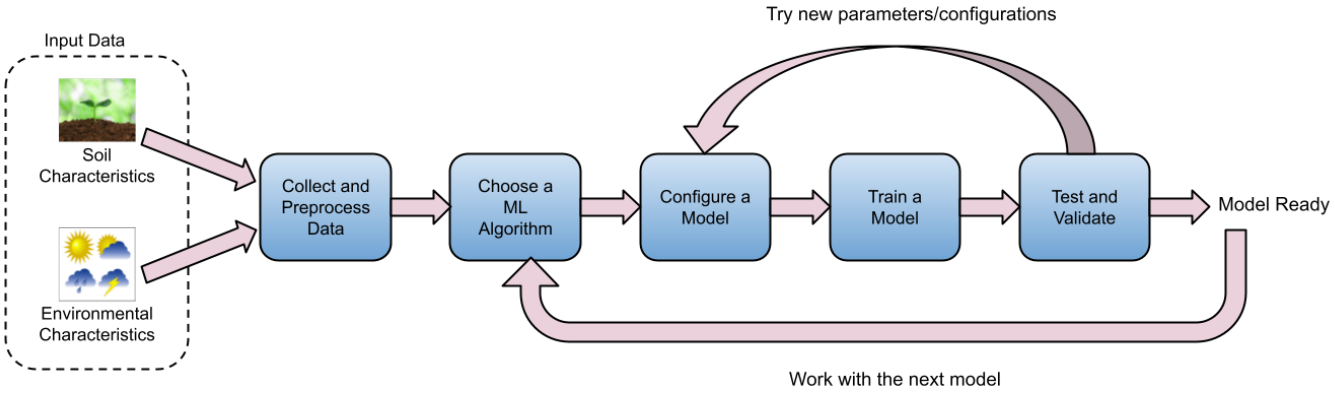

Fig. 6: Methodology

```
    predictions = model.predict(x_test_data)
    accuracy = accuracy_score(predictions,
        y_test_data)
    return round(accuracy*100, 3)

def perform_cross_val(my_model):
    scores = cross_val_score(model, features,
        labels, cv=kfold)
    mean_score = round(scores.mean()*100, 3)
```

V. Results and Evaluation

Table V shows the results of our experiments. We tried different splits for train and test data and eventually settled down for 70% training data and 30% testing data. For all the models, we achieved an accuracy of at least 95% when the proper configurations were used. We experimented with various configurations for each model. The configurations in Table V are optimal in terms of performance and accuracy.

For example, increasing the depth value of the decision tree increases the accuracy but also increases the training and prediction time. We achieved an accuracy of 99.5% with a random forest algorithm with 100 estimators.

For the neural network model, we observed that the number of epochs plays a significant role in the accuracy and performance. The experiments clearly show that the performance is inversely proportional to the number of epochs, and the accuracy is directly proportional to the number of epochs. For example, we achieved an accuracy of 97.73% with 100 epochs.

In machine learning, precision and recall [43] are two metrics used to evaluate the performance of a model. Precision measures the fraction of positive predictions that are positive, while recall measures the fraction of positive instances that are correctly identified. The last column in Table V shows precision and recall for each model. The formulas to calculate precision and recall are: $Precision = TP/(TP + FP)$ $Recall = TP/(TP + FN)$ where, TP = True Positives, FP = False Positives, and FN = False Negatives.

Based on our experimentation, the model using the Naive Bayes algorithm showed the highest accuracy. However, we believe that general neural networks would perform much better when the dataset size is much larger; also, it would be data agnostic.

We believe this work will help other developers or researchers understand the impact of different configurations on the accuracy and performance of machine learning models.

1) Average Conditions for Each Crop: Table VI shows the average weather and soil characteristics values for each crop. Given the specific conditions in their region, this table can help farmers and other agricultural stakeholders make informed decisions about which crops to grow.

VI. Challenges in Agriculture

Agriculture faces many challenges today, both in general and in the context of machine learning. Some of the most concerning challenges are listed and explained below.

1) Climate change: Climate change [4] [44] is already significantly impacting agriculture, and the effects are expected to worsen. Climate change is causing more extreme weather events, such as droughts and floods, which can damage crops and livestock. Climate change is also causing changes in temperature and precipitation patterns, making it difficult for farmers to grow the necessary crops.
2) Increasing population: The world's population is expected to reach 9.7 billion by 2050. This means that we will need to produce more food to feed everyone. The increasing population puts a strain on agricultural resources, such as land, water, and fertilizer.
3) Water scarcity: Water scarcity [45] [46] is a significant challenge in many parts of the world. Agriculture is a primary water user, and growing crops will become more challenging as water resources become scarce.
4) Soil degradation: Soil degradation [47] [48] is a significant problem in many parts of the world. Various factors, such as overgrazing, deforestation, and poor agricultural practices, can cause soil degradation. Soil degradation makes it difficult to grow crops and can lead to erosion.

Index	*ModelName*	*Accuracy%*	*ValidationAccuracy%*	*Configurations*	*Precision/Recall*
1	Logistic Regression	94.545	95.955	-	0.95/0.95
2	Decision Tree	99.091	98.682	with Gini and Max Depth=12	0.99/0.98
3	Decision Tree	99.091	98.409	with Entropy and Max Depth=10	0.99/0.99
4	Random Forest	99.545	99.5	with n_estimators=100	0.99/0.99
5	KNearest Neighbors	98.636	98.045	n_neighbors=5	0.98/0.98
6	Naive Bayes	99.545	99.5	-	0.99/0.99
7	SVM	97.727	97.682	kernel=rbf	0.98/0.98
8	SVM	99.242	98.682	kernel=linear	0.99/0.99
9	Neural Network (relu and softmax)	-	95.00	epoch=60	0.99/0.99
10	Neural Network (sigmoid)	-	97.73	epoch=1000	0.99/0.99

TABLE V: Model Accuracy

Index	*CropName*	*Nitrogen*	*Phosphorous*	*Potassium*	*Temperature*	*Humidity*	*pH*	*Rainfall*
1	rice	79.89	47.58	39.87	23.69	82.27	6.43	236.18
2	maize	77.76	48.44	19.79	22.39	65.09	6.25	84.77
3	chickpea	40.09	67.79	79.92	18.87	16.86	7.34	80.06
4	kidneybeans	20.75	67.54	20.05	20.12	21.61	5.75	105.92
5	pigeonpeas	20.73	67.73	20.29	27.74	48.06	5.79	149.46
6	mothbeans	21.44	48.01	20.23	28.19	53.16	6.83	51.20
7	mungbean	20.99	47.28	19.87	28.53	85.50	6.72	48.40
8	blackgram	40.02	67.47	19.24	29.97	65.12	7.13	67.88
9	lentil	18.77	68.36	19.41	24.51	64.80	6.93	45.68
10	pomegranate	18.87	18.75	40.21	21.84	90.13	6.43	107.53
11	banana	100.23	82.01	50.05	27.38	80.36	5.98	104.63
12	mango	20.07	27.18	29.92	31.21	50.16	5.77	94.70
13	grapes	23.18	132.53	200.11	23.85	81.88	6.03	69.61
14	watermelon	99.42	17.00	50.22	25.59	85.16	6.50	50.79
15	muskmelon	100.32	17.72	50.08	28.66	92.34	6.36	24.69
16	apple	20.80	134.22	199.89	22.63	92.33	5.93	112.65
17	orange	19.58	16.55	10.01	22.77	92.17	7.02	110.47
18	papaya	49.88	59.05	50.04	33.72	92.40	6.74	142.63
19	coconut	21.98	16.93	30.59	27.41	94.84	5.98	175.69
20	cotton	117.77	46.24	19.56	23.99	79.84	6.91	80.40
21	jute	78.40	46.86	39.99	24.96	79.64	6.73	174.79
22	coffee	101.20	28.74	29.94	25.54	58.87	6.79	158.07

TABLE VI: Features' Mean Values for Each Crop

5) Pests and diseases: Pests and diseases [49] can damage crops and livestock, leading to significant losses for farmers. Pests and diseases are becoming more resistant to pesticides, making it more difficult to control them.
6) Labor shortages: There is a labor shortage in many parts of the world, including the agricultural sector. This is due to several factors, such as an aging population, migration, and low wages. Labor shortages make it difficult for farmers to harvest crops and care for livestock.
7) Economic challenges: Farmers face several economic challenges, such as low crop prices, high input costs, and competition from imported food. These challenges can make it difficult for farmers to make a living.
8) Data availability: Machine learning algorithms require large amounts of data to train. This data can be difficult and expensive, especially in developing countries.
9) Data quality: The data used to train machine learning algorithms must be high quality. This means that the data must be accurate, complete, and consistent.
10) Model interpretability: Understanding how machine learning models make decisions is essential. This is important for farmers who need to be able to trust the decisions made by the models.
11) Awareness: Many farmers lack resources and government subsidiaries to tackle their problems. Developing countries like India are getting better at this challenge. However, some countries still lack interest access and thus related resources to educate farmers.
12) Losses, inefficiencies, and waste in the global food system: First, because global agricultural dry biomass consumed as food is 6% (energy 9.0% and protein 7.6%); and second, 44% of harvested crops dry matter lost before human consumption. This is more detailed in the reference [50].
13) Crop damage: Crop damages by wild animals [51] is a serious problem affecting farmers worldwide. Wild animals can damage crops in various ways, such as eating, trampling, polluting, and transmitting diseases. As

a result, crop damage by wild animals can significantly impact farmers' livelihoods. In some cases, it can even lead to financial ruin.

VII. Future Work/Ideas

There are multiple ways this work can be extended. Some of the examples are listed below. The readers are encouraged to extend this work by picking any of the following ideas.

1) Survey farmers to determine how much money they save using the technique. This would help to determine the economic impact of these machine learning models.
2) Mobile application, an end-to-end system for end-users (farmers or agribusiness owners), can be implemented on top of this work as a mobile application. That is, developing a business model using the models discussed in the paper. This would help ensure these techniques are available to farmers and other users.
3) Collect data from different regions; this would help us to determine if the techniques are accurate in different settings.
4) Use a larger dataset; the more data we have, the better the models can learn the relationships between different factors and crop yields.
5) Evaluate the economic and environmental impact, which would help determine the potential benefits of the technique for farmers and the environment.
6) By installing sensors on the farm, you can collect data that can be used to improve crop recommendation systems, get predictions in real time, and reduce the risk of crop losses. It will help farmers reduce labor costs, improve decision-making (because of the real-time data), and increase sustainability.

VIII. Conclusion

In conclusion, this research paper has presented crop recommendation models to predict the best crops to grow using multiple advanced machine learning algorithms and a deep neural network. The technique is scalable and easily adapted to new data and regions or countries.

The results of this study have several positive implications for the agricultural industry. First, the technique can be used by farmers to make more informed decisions about what crops to grow. Second, the method can be used by governments to develop policies that support the agricultural sector. Third, the method can be used by businesses to create new products and services that support the agricultural industry; Fourth, it will help keep the agricultural goods prices stable.

Next, we thoroughly presented agricultural challenges and some interesting future ideas to venture into.

Overall, this research has made a significant contribution to the field of agriculture. The technique is scalable, accurate, and easy to use, making it a valuable tool for farmers, governments, and businesses.

References

[1] A. L. Samuel, "Some studies in machine learning using the game of checkers," *IBM Journal of Research and Development*, vol. 3, no. 3, pp. 210–229, 1959.

[2] H. Wang, Z. Lei, X. Zhang, B. Zhou, and J. Peng, "Machine learning basics," *Deep learning*, pp. 98–164, 2016.

[3] G. Bonaccorso, *Machine learning algorithms*. Packt Publishing Ltd, 2017.

[4] A. Calzadilla, T. Zhu, K. Rehdanz, R. S. Tol, and C. Ringler, "Climate change and agriculture: Impacts and adaptation options in south africa," *Water Resources and Economics*, vol. 5, pp. 24–48, 2014.

[5] T. Partap, "Hill agriculture: challenges and opportunities," *Indian Journal of Agricultural Economics*, vol. 66, no. 902-2016-67891, 2011.

[6] T. O. Ayodele, "Types of machine learning algorithms," *New advances in machine learning*, vol. 3, pp. 19–48, 2010.

[7] D. W. Hosmer Jr, S. Lemeshow, and R. X. Sturdivant, *Applied logistic regression*. John Wiley & Sons, 2013, vol. 398.

[8] R. E. Wright, "Logistic regression." 1995.

[9] B. Charbuty and A. Abdulazeez, "Classification based on decision tree algorithm for machine learning," *Journal of Applied Science and Technology Trends*, vol. 2, no. 01, pp. 20–28, 2021.

[10] M. R. Segal, "Machine learning benchmarks and random forest regression," 2004.

[11] K. Taunk, S. De, S. Verma, and A. Swetapadma, "A brief review of nearest neighbor algorithm for learning and classification," in *2019 International Conference on Intelligent Computing and Control Systems (ICCS)*. IEEE, 2019, pp. 1255–1260.

[12] S. Chen, G. I. Webb, L. Liu, and X. Ma, "A novel selective naïve bayes algorithm," *Knowledge-Based Systems*, vol. 192, p. 105361, 2020.

[13] M. A. Hearst, S. T. Dumais, E. Osuna, J. Platt, and B. Scholkopf, "Support vector machines," *IEEE Intelligent Systems and their applications*, vol. 13, no. 4, pp. 18–28, 1998.

[14] I. Steinwart and A. Christmann, *Support vector machines*. Springer Science & Business Media, 2008.

[15] K. Gurney, *An introduction to neural networks*. CRC press, 1997.

[16] S. Albawi, T. A. Mohammed, and S. Al-Zawi, "Understanding of a convolutional neural network," in *2017 International Conference on Engineering and Technology (ICET)*, 2017, pp. 1–6.

[17] S. Sharma, S. Sharma, and A. Athaiya, "Activation functions in neural networks," *Towards Data Sci*, vol. 6, no. 12, pp. 310–316, 2017.

[18] J. Dean and S. Ghemawat, "Mapreduce: Simplified data processing on large clusters," in *OSDI'04: Sixth Symposium on Operating System Design and Implementation*, San Francisco, CA, 2004, pp. 137–150.

[19] D. Dahiphale, R. Karve, A. V. Vasilakos, H. Liu, Z. Yu, A. Chhajer, J. Wang, and C. Wang, "An advanced mapreduce: Cloud mapreduce, enhancements and applications," *IEEE Transactions on Network and Service Management*, vol. 11, no. 1, pp. 101–115, 2014.

[20] R. Kiveris, S. Lattanzi, V. Mirrokni, V. Rastogi, and S. Vassilvitskii, "Connected components in mapreduce and beyond," in *SOCC 2014*, 2014. [Online]. Available: http://delivery.acm.org/10.1145/2680000/2670997/18-kiveris.pdf?ip=104.132.34.82&id=2670997&acc=OA&key=4D4702B0C3E38B35%2E4D4702B0C3E38B35%2E4D4702B0C3E38B35%2E5945DC2EABF3343C&CFID=703717904&CFTOKEN=84396624&__acm__=1481663305_40287628fb1ed23150a1625b6d3dcf1f

[21] D. Dahiphale, "Mapreduce for graphs processing: New big data algorithm for 2-edge connected components and future ideas," *IEEE Access*, vol. 11, pp. 54 986–55 001, 2023.

[22] A. Oikonomidis, C. Catal, and A. Kassahun, "Deep learning for crop yield prediction: a systematic literature review," *New Zealand Journal of Crop and Horticultural Science*, vol. 51, no. 1, pp. 1–26, 2023.

[23] P. A, S. Chakraborty, A. Kumar, and O. R. Pooniwala, "Intelligent crop recommendation system using machine learning," in *2021 5th International Conference on Computing Methodologies and Communication (ICCMC)*, 2021, pp. 843–848.

[24] Z. Doshi, S. Nadkarni, R. Agrawal, and N. Shah, "Agroconsultant: Intelligent crop recommendation system using machine learning algorithms," in *2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA)*, 2018, pp. 1–6.

[25] S. M. PANDE, P. K. RAMESH, A. ANMOL, B. R. AISHWARYA, K. ROHILLA, and K. SHAURYA, "Crop recommender system using machine learning approach," in *2021 5th International Conference on Computing Methodologies and Communication (ICCMC)*, 2021, pp. 1066–1071.

[26] R. K. Rajak, A. Pawar, M. Pendke, P. Shinde, S. Rathod, and A. Devare, "Crop recommendation system to maximize crop yield using machine learning technique," *International Research Journal of Engineering and Technology*, vol. 4, no. 12, pp. 950–953, 2017.

[27] D. Reddy and M. R. Kumar, "Crop yield prediction using machine learning algorithm," in *2021 5th International Conference on Intelligent Computing and Control Systems (ICICCS)*, 2021, pp. 1466–1470.

[28] R. Ghadge, J. Kulkarni, P. More, S. Nene, and R. Priya, "Prediction of crop yield using machine learning," *Int. Res. J. Eng. Technol.(IRJET)*, vol. 5, 2018.

[29] N. H. Kulkarni, G. N. Srinivasan, B. M. Sagar, and N. K. Cauvery, "Improving crop productivity through a crop recommendation system using ensembling technique," in *2018 3rd International Conference on Computational Systems and Information Technology for Sustainable Solutions (CSITSS)*, 2018, pp. 114–119.

[30] S. Pudumalar, E. Ramanujam, R. H. Rajashree, C. Kavya, T. Kiruthika, and J. Nisha, "Crop recommendation system for precision agriculture," in *2016 Eighth International Conference on Advanced Computing (ICoAC)*, 2017, pp. 32–36.

[31] K. G. Liakos, P. Busato, D. Moshou, S. Pearson, and D. Bochtis, "Machine learning in agriculture: A review," *Sensors*, vol. 18, no. 8, p. 2674, 2018. [Online]. Available: https://www.mdpi.com/1424-8220/18/8/2674

[32] M. Ayaz, M. Ammad-Uddin, Z. Sharif, A. Mansour, and E.-H. M. Aggoune, "Internet-of-things (iot)-based smart agriculture: Toward making the fields talk," *IEEE Access*, vol. 7, pp. 129 551–129 583, 2019.

[33] V. Iglovikov, S. Mushinskiy, and V. Osin, "Satellite imagery feature detection using deep convolutional neural network: A kaggle competition," *arXiv preprint arXiv:1706.06169*, 2017.

[34] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin *et al.*, "Tensorflow: Large-scale machine learning on heterogeneous distributed systems," *arXiv preprint arXiv:1603.04467*, 2016.

[35] S. Sharma, S. Sharma, and A. Athaiya, "Activation functions in neural networks," *Towards Data Sci*, vol. 6, no. 12, pp. 310–316, 2017.

[36] B. Ding, H. Qian, and J. Zhou, "Activation functions and their characteristics in deep neural networks," in *2018 Chinese Control And Decision Conference (CCDC)*, 2018, pp. 1836–1841.

[37] J. Brownlee, "What is the difference between a batch and an epoch in a neural network," *Machine Learning Mastery*, vol. 20, 2018.

[38] J. T. Barron, "A general and adaptive robust loss function," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 4331–4339.

[39] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[40] A. Lydia and S. Francis, "Adagrad—an optimizer for stochastic gradient descent," *Int. J. Inf. Comput. Sci*, vol. 6, no. 5, pp. 566–568, 2019.

[41] A. M. Taqi, A. Awad, F. Al-Azzo, and M. Milanova, "The impact of multi-optimizers and data augmentation on tensorflow convolutional neural network performance," in *2018 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR)*, 2018, pp. 140–145.

[42] P. Gulati, A. Sharma, and M. Gupta, "Theoretical study of decision tree algorithms to identify pivotal factors for performance improvement: A review," *Int. J. Comput. Appl*, vol. 141, no. 14, pp. 19–25, 2016.

[43] J. Davis and M. Goadrich, "The relationship between precision-recall and roc curves," in *Proceedings of the 23rd international conference on Machine learning*, 2006, pp. 233–240.

[44] T. B. Pathak, M. L. Maskey, J. A. Dahlberg, F. Kearns, K. M. Bali, and D. Zaccaria, "Climate change trends and impacts on california agriculture: a detailed review," *Agronomy*, vol. 8, no. 3, p. 25, 2018.

[45] N. Mancosu, R. L. Snyder, G. Kyriakakis, and D. Spano, "Water scarcity and future challenges for food production," *Water*, vol. 7, no. 3, pp. 975–992, 2015.

[46] E. Vallino, L. Ridolfi, and F. Laio, "Measuring economic water scarcity in agriculture: a cross-country empirical investigation," *Environmental Science & Policy*, vol. 114, pp. 73–85, 2020.

[47] A. Alam, "Soil degradation: a challenge to sustainable agriculture," *International Journal of Scientific Research in Agricultural Sciences*, vol. 1, no. 4, pp. 50–55, 2014.

[48] R. Lal, "Restoring soil quality to mitigate soil degradation," *Sustainability*, vol. 7, no. 5, pp. 5875–5895, 2015.

[49] M. Donatelli, R. D. Magarey, S. Bregaglio, L. Willocquet, J. P. Whish, and S. Savary, "Modelling the impacts of pests and diseases on agricultural systems," *Agricultural systems*, vol. 155, pp. 213–224, 2017.

[50] P. Alexander, C. Brown, A. Arneth, J. Finnigan, D. Moran, and M. D. Rounsevell, "Losses, inefficiencies and waste in the global food system," *Agricultural Systems*, vol. 153, pp. 190–200, 2017. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0308521X16302384

[51] B. Awasthi and N. B. Singh, "Status of human-wildlife conflict and assessment of crop damage by wild animals in gaurishankar conservation area, nepal," *Journal of Institute of Science and Technology*, vol. 20, no. 1, pp. 107–111, 2015.

Devendra D. Dahiphale (Member, IEEE) has received a Diploma in Teacher Education (DTEd.) from the University of Pune, India, the B.A. degree in English from Yashvantrao Chavan Maharashtra Open University (YCMOU), Nashik, India, the B.E. degree in computer science and engineering from the Pune Institute of Computer Technology, India, and the M.Sc. degree in computer science from the University of Maryland, Baltimore County. He was a Software Engineer with Google, USA; Cisco Systems, USA; Amazon.com, USA; and Imagination Technologies, India. He works as a Software Engineer at Google and is also a member of Dreamz Group, a group of software professionals passionate about innovation and technology development. He has more than ten years of professional experience as a Software Engineer.

Pratik Shinde (Member, IEEE) Pratik holds a Bachelor of Engineering Degree in Computer Science and Engineering from Pune University, India and Master's degree in Computer Science from University of Maryland Baltimore County, USA. He currently works as a Software Engineer with Amazon.com. Pratik has more than ten years of experience as a Software Developer.

Koninika Patil holds a Bachelor of Engineering Degree in Computer Science and Engineering from Pune University, India, and a Master's degree in Computer Science from University of Maryland Baltimore County, USA. She currently works as a Software Engineer with Amazon.com. Koninika has more than seven years of experience as a Software Developer, and she has primarily focused on Machine Learning and Computer Vision in her thesis work.

Vijay Dahiphale received a Bachelor of Engineering degree in Electronics and Telecommunication from the Pune Institute of Computer Technology (PICT), India; currently, Vijay is working as a Software Engineer at Mobiquity Pvt. Ltd has more than five years of working experience in the I.T. Industry. He has developed a k; Ciscoerest in IoT security, machine learning, embedded system design, and especially lightweight cryptography, where he has done much work and research. He has published six research papers in IEEE and Springer conferences and journals. Moreover, he has also received a sponsorship from MSM.digital Ltd to present his work at the international conference in London, UK. His various projects also include work on ARM-7, Atmega, and Raspberry Pi-3 platforms. He has also designed a system to recognize the characters on the vehicle number plate using the KNN machine learning algorithm. In addition, he has participated and won accolades in multiple competitions - INC-17 Tantra, an international-level embedded c coding competition, being one of them. Furthermore, he has guided two groups of undergraduate students for their academic mini-project. His present research is focused on designing new lightweight cryptographic algorithms and compact implementation of these algorithms for resource-constrained environments.