Journal of Theoretical and Applied Information Technology
15th April 2023. Vol.101. No 7

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

ENHANCED PREDICTION OF CROP YIELD USING DYNAMIC ANT COLONY OPTIMIZATION-BASED RECURRENT NEURAL NETWORK (DAC0-RNN)

SHANMUGA PRIYA S[1],DR.M.SENGALIAPPAN[2]

[1]Research Scholar, Kovai Kalaimagal College of Arts & Science, Bharathiar University, Coimbatore

[2]Associate Professor & Head, Department of MCA, Nehru College of Management, Coimbatore

E-mail: [1]shanpriya.s@gmail.com, [2]cmsengs@gmail.com

ABSTRACT

Machine learning (ML) based crop yield prediction is a method of using classification algorithms to analyze and predict the yield of crops based on historical data and current environmental conditions. Crop yield prediction models require a large amount of data for training and validation, such as weather data, soil data, and historical yield data. ML-based crop yield prediction aims to improve crop yields by providing farmers with accurate and timely information about their crops. This paper presents a new deep learning strategy called Dynamic Ant Colony Optimization-based Recurrent Neural Network (DACO-RNN) to address the issues of low classification accuracy often encountered with traditional ML algorithms during training and testing phases. The DACO-RNN is inspired by the foraging behavior of ants in nature and utilizes an optimization technique to improve the performance of the RNN, which otherwise would degrade during the back-propagation phase. The effectiveness of the DACO-RNN was evaluated using two crop yield datasets and standard ML metrics, and the results showed that it had better performance than existing classifiers.

Keywords: *Crop Yield, Prediction, Classification, Ant Colony, Neural Network, Optimization*

1 INTRODUCTION

As a significant provider of sustenance, agriculture ranks high on the list of socially essential industries. A growing number of people in the world's developing countries has left many of them hungry. To counteract the consequences of an increasing population, changing environment, soil degradation, reliable and timely crop growth, natural weather unpredictability, and output are essential[1]. It's also essential to help ensure the long-term viability of the agricultural food supply system. As a result of these demands, it is clear that assessing land, safeguarding crops, and predicting agricultural yields are of paramount significance to the world's food supply. To improve national food security, policymakers need reliable forecasts of future crop yields to make informed decisions about whether to export or import[2].

But it's hard to estimate harvest success because of all the variables. Landscape, soil, insect infestation, genotype, water quality and availability, weather, harvest timing, and a host of other variables have a role in crop output. The methods and processes that affect crop output vary in time and are inherently non-linear[3]. Complexity also arises because these techniques consider various external and non-arbitrable elements. Historically, farmers relied on their knowledge and reliable historical data to forecast crop yields and make major cultivation choices in light of those forecasts. However, the development of new inventions such as crop model simulation and machine learning, as well as the capability to analyze massive amounts of data with high-performance computers, have seemed to help estimate yields more accurately in recent years. Many recent studies show that machine learning algorithms have more promise than conventional statistical methods[4].

An Artificial Intelligence (AI) branch called "machine learning" allows computers to acquire new skills without being explicitly programmed. Such methods triumph over non-linear or linear agricultural systems by guaranteeing a remarkable predictive ability[5]. A self-learning agriculture system discovers the strategies. Instructing a train to carry out a specified mission is one example of such a method. After the model has finished training, it

will automatically assume that the data should be sent to the testing phase[6].

Bio-inspired optimization algorithms, such as particle swarm optimization and genetic algorithms, are commonly used in machine learning for crop yield prediction because they can efficiently search for optimal solutions in high-dimensional and complex parameter spaces. These algorithms are also robust to noise and can handle non-linear and non-differentiable objectives. Additionally, bio-inspired optimization can incorporate domain knowledge and constraints into the optimization process, leading to more accurate and reliable predictions in different fields, including networking [7], [8], [17], [18], [9]–[16], [19]. Some of the advantages of Bio-Inspired Optimization Algorithms (BIOA) are listed below:

- Global optimization capabilities: BIOA, such as genetic algorithms and particle swarm optimization, can search the entire solution space, making them well-suited for complex and non-linear problems.
- Handling high-dimensional problems: BIOA can handle problems with a large number of variables, making them useful for high-dimensional problems like crop yield prediction.
- Handling noisy and uncertain data: BIOA, such as ant colony optimization, can handle noisy and uncertain data, making them useful for problems where data quality is an issue.
- Handling constraints: BIOA have built-in mechanisms for handling constraints, such as bounds on variables, making them useful for problems where constraints are essential.
- Handling multiple objectives: BIOA can handle multiple objectives, such as maximizing yield while minimizing costs, making them useful for problems with multiple objectives.
- Handling dynamic and non-stationary environments: BIOA, such as artificial immune systems, can adapt to dynamic and non-stationary environments, making them useful for problems where the optimal solution changes over time.
- Handling non-differentiable functions: BIOA can optimize non-differentiable functions, making them useful for problems where gradient-based optimization is not possible.

1.1 Problem Statement

Several issues can arise when using machine learning to predict crop yields, including:

- Lack of quality data: Crop yield predictions rely heavily on data, such as weather patterns, soil conditions, and previous yield data. The predictions may be unreliable if this data is not accurate or complete.
- Climate change and weather variability: Crop yields are also highly dependent on weather and climate conditions which are difficult to predict and can vary significantly year by year. This can cause issues when trying to predict crop yields using machine learning.
- Limited generalizability: Some machine learning models work well in specific regions or climates but may not apply to other areas. This can be a problem when making predictions for a different location or time of year.
- Temporal and spatial variability: Climate, weather, and the environment are highly variable both temporally and spatially. Therefore, ML models need to be designed in a way that they can capture this variability to make accurate predictions.
- Handling missing data: It's common to have missing data in crop yield prediction due to many factors like sensor malfunction and data loss. Missing data can cause issues when training the model, as well as when making predictions.
- High computational cost: Some advanced ML models, like deep learning models, require a lot of computational resources to train and make predictions. This can be a problem for farmers and researchers who may not have access to these resources.
- Pest and disease: Pest and disease can affect crop yields significantly, and it's hard to predict when and how much it will affect the yields.
- Lack of feature engineering: Crop yield prediction is a complex task that depends on many factors, such as weather, soil, and farming practices. If the model is not designed to take these factors into account, it may not be able to make accurate predictions.
- Complexity of crop growth: Crop growth is influenced by various factors, including weather, soil conditions, and pests. Modeling the complex interactions between these factors can be challenging.
- Model complexity: Artificial Intelligence based learning models can be very complex and challenging to interpret, making it difficult to understand why the model makes confident predictions.
- Data quality: The accuracy of a model's predictions can be greatly influenced by the quality of data used during its training. If the data contains inconsistencies or errors, the model may not be able to make accurate predictions. It is crucial to ensure

that the data used to train a model is accurate, relevant, and consistent to achieve the best results.

- Lack of robust and diverse data: The data available to train these models is often limited, inconsistent, and not diverse enough, which can impact the accuracy of the predictions.

1.2 Motivation

Following is the list of things that motivated this research work:

- Improving food security and reducing global hunger
- Increasing crop productivity and efficiency
- Optimizing crop management and resource usage
- Reducing crop loss due to weather and climate change
- Improving economic profitability for farmers and agricultural industries.
- Reducing the environmental impact of agriculture
- Developing new crop varieties and hybrids
- Enhancing precision agriculture.
- Improving the predictability of the food market.
- Supporting sustainable development goals.

1.3 Research Objective

The significant intention of this research work is to develop a new method for predicting crop yields that combine the optimization capabilities of ant colony optimization (ACO) algorithms with the predictive capabilities of recurrent neural networks (RNNs), namely "Dynamic Ant Colony Optimization-based Recurrent Neural Network (DACO-RNN)". The goal would likely be to create a more accurate and efficient method for predicting crop yields, which could be useful for farmers, agricultural researchers, and policymakers. Additionally, the dynamic aspect of the research could be to develop an adaptable model which can handle changing environmental conditions and varying crop types.

1.4 Organization of the Paper

Section 1 summarized the Introduction with a clear and concise statement of the problem being addressed in the research paper, an explanation of why the problem is important and relevant, and a statement of the specific objectives of the research. Section 2 critically reviews existing literature on the topic, including relevant research studies, theories, and models. Section 3 provides a detailed description of the proposed classifier, including its architecture, algorithms, and any novel contributions. Section 4 illustrates the description of the dataset used for the research, including its size and features. Section 5 describes the metrics used to evaluate the performance of the classifiers. Section 6 discusses the results obtained from the classifiers, including any comparison with other existing methods. Section 7 summarizes the main findings and contributions of the research, as well as any limitations and future work.

2 LITERATURE SURVEY

A literature review on crop yield prediction would involve conducting an extensive examination of existing research on the topic. This would include identifying and analyzing previous studies, articles, and publications that have investigated crop yield prediction. The review would look at different techniques that have been employed for this purpose, such as statistical models, machine learning algorithms, and remote sensing techniques. This examination would provide a comprehensive understanding of the state of the field and highlight areas that require further research. The review would also likely examine the factors that affect crop yield, including weather conditions, soil properties, and crop management practices. Additionally, the review may look at the accuracy and reliability of different prediction methods and their potential applications in agriculture and food production. Overall, a literature review on crop yield prediction would aim to provide a comprehensive overview of the current state of research in this field.

A model utilizing Random Forest (RF) techniques[20] is presented to forecast China's wheat, maize, and rice crop output. The model utilizes seven years of recorded data from 2013 to 2019 and three primary data sources, including climate data, vegetation indices, and soil characteristics. The study found that the RF model accurately predicted yields for all three crops, with correlation coefficients surpassing 0.75 and normalized root mean square errors falling below 18.0%. Additionally, the researchers discovered that certain factors, such as solar radiation and vegetation indices, played a crucial role in predicting winter wheat yields. In contrast, vegetation indices and drought were significant in predicting spring maize yields, soil moisture was important for summer maize, late rice, and mid rice, and precipitation was crucial for early rice. "Hybrid Machine Learning-based Method" [21] combines a physical approach with machine learning to estimate Nitrogen content in crops. It uses models to generate a training database which is then used with advanced machine learning techniques, such as the Gaussian process and heteroscedastic GP regression, to provide confidence intervals for the

estimates. Additionally, it uses a GP-based algorithm to analyze band-specific information for estimating aboveground Nitrogen. The method has been tested and validated using multiple field campaigns and destructive Nitrogen measurements. A model called "Bayesian Neural Network Approach (BNNA)" [22] has been proposed for predicting corn yields at the county level using various publicly available data sources. These include satellite imagery, climate observations, soil maps, and historical corn yield records. The model was tested and found to be accurate in predicting yields, even in years with extreme weather events. The BNNA model provides a strong framework for predicting crop yields during the growing season and emphasizes the importance of understanding the impact of environmental stress on crop yields.

"Adaptive Adversarial Domain Adaptation Approach (AADAA)" [23]is proposed for predicting corn yield. This method is designed to adaptively learn domain-invariant features and accurately predict yield at the same time. The study collects and combines data on vegetation indices and weather observations from various sources to the county level and then uses these features to train the AADAA model. That research was conducted in the US corn belt, with counties divided into two different ecological regions, and evaluated over four years from 2016 to 2019.A method called "Multimodal Data Fusion and Deep Learning"[24]is suggested for estimating soybean yield using thermal images, multispectral, and RGB collected from Unmanned Aerial Vehicles at a test site in the USA. The method utilizes various features extracted from the images, such as texture, thermal, structure, spectral, and canopy. It uses (i) Deep-Neural-Network, (ii) Partial-Least-Squares-Regression, (iii) Random-Forest-Regression, and (iv) Support-Vector-Regression to make predictions. "Gated-Recurrent Convolutional Neural Network (G-RecConNN)" [25] has been proposed for identifying crop diseases, specifically in plantain tree cultivation. The model combines RNN and CNN and takes sequences of plant images as input. The model is tested using real-time data from Tamil Nadu, India. The proposed model has several benefits, such as less need for data pre-processing, ease of evaluating performance in real-time, and improved results with less actual data. A technique called "Feature Selection based Classification" [26] is proposed for predicting rice yields. It involves selecting and combining various data characteristics from the threshing and separating equipment, using statistical and signal analysis methods to simplify decision-making. Then, techniques such as Principal Component and Kernel Principal Component analyzes are utilized to (i) remove correlation and nonlinearity among the chosen features, (ii) decrease the dimensionality of the feature set, and (iii) improve the accuracy of the prediction model. Finally, models such as Support Vector Machine Classifier, Extreme Learning Machine Classifier, and Random Forest Classifier are used to classify the reduced feature set using metrics such as accuracy rate and recognition time for evaluating and comparing the models.

"Prediction of wheat yield in Australia"[27]has been suggested, which employs machine learning and examines the influence of large-scale climate factors associated with ENSO. This model, called the Random Forest-based Model (RFM), starts by using RFM to forecast spring rainfall in four major wheat-producing states based on historical climate data. Then, it combines this forecasted rainfall with other climate factors to predict variations in yield. The research found that the RFM resulted in 43-59% prediction of the changes in spring rainfall, and by incorporating forecasted spring rainfall with ENSO indices, the model resulted in 33-66% predictions, which is a greater percentage than the 22-50% yielded by ENSO-related indices alone. "Salp Swarm Algorithm - Extreme Learning Machine (SSA-ELM)"[28] is suggested for identifying different types of dry bean plants. SSA-ELM aims to build a dataset with 14 different categories of dry beans and evaluate the effectiveness of combining the Extreme Learning Machine model fused with the GoogLeNet transfer learning approach on this dataset. The parameters of the ELM classifier are also optimized with the help of the Salp Swarm Method (SSA), a swarm intelligence algorithm. The findings demonstrate that the suggested hybrid model provides superior classification accuracy and performance compared to more conventional machine learning methods. The "Synergistic Integration" [29] method is proposed for estimating crop yields by combining multi-sensor remote sensing data from two satellite sensors, MODIS and SMAP. As a new measure, the time difference between the Enhanced Vegetation Index and the Vegetation Optical Depth is used, and the full-time series data is combined using machine learning methods. Total county-level production and individual yield estimates for corn, soybeans, and wheat are estimated using regularized linear and kernel ridge regression in the SI research.

"Random Forest Algorithm (RFA)"[30]is a method for ensemble learning that utilizes

multiple decision trees to improve the accuracy of predictions. This algorithm is supervised and can be applied to both classification and regression problems. The RFA creates multiple decision trees during the training phase and then takes the average of their predictions during the testing phase. This averaging process helps to reduce the variance and bias of the individual decision trees, resulting in a more robust and accurate model. One of the key advantages of RFA is its ability to handle high-dimensional data and a large number of input variables without overfitting. It also can identify important features in the dataset. However, it can be computationally expensive and may not be the best choice for real-time predictions. Overall, RFA is a powerful algorithm widely used in industry and research, but performance degrades when applied to predict crop yields."Support Vector Machines (SVM)"[31]is a form of supervised machine learning algorithm that can be employed for both classification and regression tasks. The goal of an SVM is to identify the optimal boundary, also known as a "hyperplane," that separates the data points into distinct categories. These algorithms are particularly useful in situations where the data has numerous features or is not linearly separable. To accomplish this, the algorithm maps the data into a higher-dimensional space where a linear boundary can be found, and then projects it back into the original space. One of the major benefits of SVMs is their ability to handle non-linearly separable data through the use of the kernel trick, which maps the data into a higher-dimensional space where it becomes linearly separable. Some common kernel types include linear, polynomial, and radial basis function (RBF). Additionally, SVMs have a regularization parameter which allows for control of the balance between maximizing the margin and minimizing misclassification errors. These algorithms are frequently utilized in various applications such as text and image classification, and bioinformatics. However, they can be less efficient when working with large datasets and they are also sensitive to the choice of kernel and regularization parameter. Overall, SVM classifier is a powerful algorithm, which is particularly useful in cases where the data is not linearly separable and when the number of features is large. As SVM is utilized for forecasting crop yield, its efficiency diminishes.

3 DYNAMIC ANT COLONY OPTIMIZATION - BASED RECURRENT NEURAL NETWORK (DACO-RNN)

Ant Colony Optimization (ACO) is a metaheuristic algorithm inspired by ant colonies' behavior. ACO is a method used to address and solve optimization challenges, such as determining the most efficient route for a traveling salesperson or performing the better classification. On the other hand, RNN, or Recurrent Neural Network, is a type of artificial neural network that is specifically designed to handle sequential data, and is commonly utilized for tasks such as language processing, recognizing spoken words, and predicting future events based on past patterns. Combining ACO with RNN would involve using the principles of ACO to optimize the parameters of the RNN, such as the weights and biases, to improve its performance on a specific task. This may be done by using the ACO algorithm to search for the optimal set of parameters and then using those parameters to train the RNN. However, this is a research area, and little information is available on this topic.

3.1 Recurrent Neural Network

Inputs to the proposed model come from crop data over time and sensed data. Typically, crop data were gathered at regular periods. To represent crop data as a time series, this research work uses the vector p, wherein $p = (p_1, p_2, \ldots . p_f)$, and p_f is the crop information (overall amount of crop grown etc) at the fth point time, $f\omega T^*$. In most cases, the duration of a time series is unknown, making it difficult to include all of the data in a model. It must be neatly separated into smaller sub-series $p_f = (p_{f-Z}, p_{f-Z+1}, \ldots . p_{f-1})$, wherein Z is the series data delays, often a tiny integer number, $Z\omega T^*$. At the f time point, these brand-new corrected characteristics of the sequence will be uncovered by breaking it down into sub-series.

Numerous contextual factors—climate, location, date/time, events, and so on—often accompany time series. Sequence characteristics are necessary for understanding it. These factors are similarly specified as I, wherein $I = (i_1, i_2, \ldots . i_f)^F$, and shown as a matrix I as seen below:

$$I = \begin{pmatrix} i_1 \\ i_2 \\ \vdots \\ i_f \end{pmatrix} = \begin{pmatrix} i_1^1 & i_1^2 & \ldots & i_1^C \\ i_2^1 & i_2^2 & \ldots & i_2^C \\ \vdots & \vdots & \ddots & \vdots \\ i_f^1 & i_f^2 & \ldots & i_f^C \end{pmatrix} \quad (1)$$

in which $i_f^c$is the cth other than at the fth time point $s\omega(1, \ldots\ldots, C)$, C is the number of types of contextual factors, and ωT^*. Date/time, location, and other contextual variables are only some of the s-th kind of parameters represented by the column vector i^s in the matrix.For simplicity, this research work will refer to the row vector representing the state of nature at time $f$as i_f. Currently, this research work has a slice of the time series for data transmission and the vectors of contextual factors at time f. All of these pieces of information are needed to run the proposed model properly, and they may be mashed together to form a single input vector (referred to as P_f):

$$P_f = [p_f, i_f] = [(p_{f-Z}, p_{f-Z+1}, \ldots\ldots p_{f-1}), (I_f^1, I_f^2, \ldots\ldots I_f^c)] \quad (2)$$

In this case, this research work has data on a crop series spanning from$(f - Z)$ to $(f - 1)$ and data on a collection of all possible inner factors at time f. Same as above in terms of definition and scope of allowed variation. Due to the varying dimensions of input data, i.e., the hour. The objective variable $\hat{q}_f$ at period f determines the forecasting aim or model output. To achieve the desired transformation from $P_f$to $\hat{q}_f$, a map function must be found.

3.1.1 Knowledge acquisition by sequence

David Rumelhart's research from 1986 formed the basis for recurrent neural networks (RNNs). One type of artificial neural network, RNN, is characterized by a directed graph connecting its nodes that is organized chronologically. As a result, it is endowed with temporal dynamic behavior. When processing sequences of inputs, RNNs have an advantage over feed-forward neural networks because of their ability to employ internal states (i.e., memory). This means they may be used for temporally dynamic tasks.

Among the many variants of RNNs that have been refined and improved, long short-term memory (LSTM) networks continue as best. Hochreiter and Schmidhuber found it for the first time to fix the gradient back-propagation phase's disappearing and exploding issue, which occurs when gradient signals multiply by a significant factor. The Gates concept, which is repeated, is commonly used to enhance LSTM. This strategy substitutes the exponential gate activation for the Constant Error Carousel (CEC) weight, which allows memory to be cleared automatically when the data flow becomes stale. Since there might be gaps of undetermined duration between crucial occurrences in a time series, LSTM networks excel at categorizing, processing, and generating predictions from time series data.

The memory cell is a novel structure that serves as the foundation of LSTM. It has a nerve cell with a connection that goes back to itself and three gates connected to it. The self-recurrent connection ensures that the memory cell's state is preserved from one-time step to the next without any external influence. The three gates control the entry and exit of data to and from the memory cell, allowing it to store values for an indefinite time. LSTM may learn to recall or forget information over time using these principles.

All of the gates are programmed to start generating some degrees based on the current input p_f, the last production response $l_{f-1}$and the cell state U_{f-1} from the previous step to determine whether to accept the input data, ignore the previously retained information, oroutput the newly generated memory condition. This research work describes the changes to the state of the memory module with the following gates and assumes that the input at time $f$is p_f, the output response is l_f, and the output response at time f-1 isl_{f-1}.

$$s_f = \in \left(N_{ps} p_f + N_{ls} l_{f-1} + v_s\right) \quad (3)$$

$$g_f = \in \left(N_{pg} p_f + N_{lg} l_{f-1} + v_g\right) \quad (4)$$

$$j_f = j\left(N_{pu} p_f + N_{lu} l_{f-1} + v_u\right) \quad (5)$$

$$k_f = \in \left(N_{pk} p_f + N_{lu} l_{f-1} + v_k\right) \quad (6)$$

$$U_f = g_f \circ U_{f-1} + s_f \circ j_f \quad (7)$$

$$l_f = k_f \circ l\left(U_f\right) \quad (8)$$

When two matrices of the same size are multiplied, the result is called the Hadamard product, also known as the element-wise product, denoted by the notation °. During training, the network's matrix of weights N and bias vectors v will be dynamically learned and modified. In most cases, the activation functions of a block's input j and output l are both hyperbolic tangents, whereas the activation function of a non-linear gate $\in$ is the logistic sigmoid. The time steps are indicated by the subscript f, and the constants of U_0 and l_0 are both 0. Once data are introduced to the model, the

LSTM component will recursively process it. The memory will produce a value of l at each time step, with the final result, l_f, representing the expected outcome of these iterations.

Multiple LSTMs are typically used in multi-layer recurrent neural networks. Each time series'$e_f = (l_{f-Z}, l_{f-z+1}, \ldots\ldots, l_f)$the state can serve as inputs to the subsequent LSTM layer. Therefore, every LSTM memory in the preceding layer will send the sequence $e_f$correspondingly to the LSTM memory cells in the next layer. The final hidden state l_f from the last LSTM layer is the intended result of the stacking LSTMs. Finally,M and A are the total numbers of layers in a stacked LSTM, whereas $l_f \omega B^{M\times A}$, is the dimension of the input features M and A. This research work feedsthe crop series data into the LSTMs solely. Therefore$M = 1$. It's important to note that the initial input dimensionality will change depending on whether or not volume, occupancy, or other factors are considered.

3.1.2 Context based feature extraction

Feature extraction is a set of machine-learning methods that lets a system figure out what representations are needed for feature detection or categorization. This eliminates the need for human involvement in feature engineering and paves the way for an automated system to discover and employ the features in question. To replicate its input in its output, autoencoders are a form of deep neural network. It uses the same number of hidden layers to replicate the input code described by numerous internal layers. The two primary components are an encoding stage, where the input is transformed into the encoding, and a decoding stage, when the output code is transformed back into the original input, represented by the transitions τ and $\emptyset$, respectively.

$$\tau: I \to \Theta$$

$$\emptyset: \Theta \to I \qquad (9)$$

$$\tau, \emptyset: arg \min_{\tau,\emptyset} \|I - (\tau \circ \emptyset)\|^2$$

The coding process has three distinct phases. In the first step, the encoder, the input $I\omega B^{1\times C}$ is transformed into the output $\Theta\omega B^{1\times T}$:

$$\Theta = \tau(NI + v) \qquad (10)$$

The code representation isa common name,i.e., Θ. It's a layered representation that is learned and gradually abstracted and composited. Weight matrix N, bias vector v, and activation function τ are utilized fully. Secondly, Θ becomes I' at the decoder step of encoding:

$$I' = \tau(N'\Theta + v') \qquad (11)$$

where decoder's τ, N' and v' notation is identical but more complex than the encoder's τ, N and v notation. Even though the$I'$is specified as the rebuilding of I using the same form, it has always been allocated as I only.

Successful completion of this level is optimum. At this point, procedures are used to train and update the network to improve its parameters continuously. These settings reduce reconstruction mistakes like the most popular squared errors and are determined by the Autoencoder's architecture.

$$Z(I, I') = \frac{1}{2}\|I - I'\|^2 = \frac{1}{2}\left\|I - \tau\left(T'\left(\tau(NI + v)\right) + v'\right)\right\|^2 (12)$$

The training set for features is I, and the abstract coding is Θ. The factors Θ may be thought of as a differently-shaped compression of the data I.If the function set,Θ has fewer dimensions,then the input vector is set as I. The feature code may be considered a complete representation of the input space for higher-dimensional feature spaces.

There will be more neurons in the hidden layer than in the output or input layers. While the "sparse" network's brief and simplistic depiction acts as a key characteristic dispersed over a denser neuron space, the core characteristics and fascinating structure remain identified by applying additional restrictions.

Sparse constraints in the autoencoders improve the sparsity, and it has many useful qualities. It can simultaneously expand the space between features and draw attention to their essential expression. A high dimension-expanding effect is used to locate finer-grained characteristics; however, it also introduces noisy data. In reality, a high dimension-expanding effect can be continually trained and upgraded to lessen the effect of noise. Contextual factor vectors are considered input, and the network's internal state is updated using a linear cum balanced activation function. The result isa

high-level expanded representation of the factors. The final expanded code for contextual features is generated by the final layer of the coding neural network and written as $\Theta = \rho^a = (\rho_f^1, \rho_f^2, \ldots\ldots, \rho_f^A)$, where $\rho_f^a$is the output of the ath neuron within the last stack at the moment f as well as A is the final scale ofthe neuron in the coding layer's code or the quantity of those neurons, $A\omega T^*$.

3.1.3 Integration capability

The contextual feature coding yielded an extended factors code $\Theta = \rho^a = (\rho^1, \rho^2, \ldots\ldots, \rho^A)$, where ρ^A is the production of the a-th neuron at the time f, and A is the extended coding for the dimensionality of factors. A corresponds to the total number of neurons in the final encoder layer, $A\omega T^*$. Additionally, the sequence features are learned via a layered LSTM neural network, yielding an abstraction time-series data state set $l = (l^1, l^2, \ldots. l^A)$, where A is the final dimension of sequence features and is also the number of neurons in the last Hidden layers, $A\omega T^*$. The fact that these two vectors are of the same dimension allows for a tensor injection operation. The extended factors code Θ can be used to assign a value to each feature in a time series, state l. The softmax function multiplies the intermediary LSTM state transmission l by each element in Θ, shrinking it to the range [0,1]. When time series characteristics of crop data are combined with extended features of contextual variables, the result will be the greater abstract features $D = (d^1, d^2, \ldots. d^a)$ as shown in Eq.(13).

$$D = softmax(\Theta)°l = \frac{h^\Theta}{\sum h^\Theta}°l \qquad (13)$$

Hadamard product is represented as °,and internal factors are encoded as Θ, and sequence characteristics are represented as l. This pair of vectors is the same as the dimension. The next step in predicting future crop speeds $\hat{q}$ is to add (i)a fully connected layer and (ii) a predictor to the top of the module. A mathematical definition of the predictor is expressed in Eq.(14).

$$\hat{q} = \in (N_d D + v_d) \qquad (14)$$

wherein∈ is a non-linear activation function, N_d is the weight of the matrix, and v_d is the bias vector.

The proposed model is trained using the back-propagation approach, eventhough it comprises several distinct types of neural networks.

3.1.4 Model training through back-propagation

Every part of the proposed model may be trained separately using standard training techniques because it is derived from a neural network. Minimizing transfer error and training each component may be done sequentially using the gradient descent stochastic and back-propagation techniques. The RNN and the autoencoder neural network each took the input vectorsp_f and $i_f$with the corresponding output vector l and Θ. After combining $l$and Θ, this research work gets D, and it was sent to a third neural network, i.e., the predictor. It helps us to figure out what the objective should be. This process of forward propagation can be represented mathematically as Eq.(15).

$$\hat{q} = \in (N_d(softmax(\Theta)°l) + v_d) \qquad (15)$$

Partially deriving bias and weight parameters are straightforward after the errors at the output for each component. This research work first defines the value function W as calculating its partial derivatives, represented as Eq.(16).

$$W(\hat{q}, q) = \|\hat{q}, q\|^2 = \frac{1}{2}\|\in (N_d D + v_a) - q\|^2 = \frac{1}{2}\|\in (N_d D(softmax(\Theta)°l) + v_a) - q\|^2 \qquad (16)$$

Machine learning relies heavily on the cost function, which quantifies how much a given answer deviates from the optimal option. The parameters may be quickly determined by locating the cost function's extrema. The Predictor network requires two adjustments to its settings. First, using chain rules, one may determine both the N_d and$v_d$gradients.

$$\frac{\varepsilon w}{\varepsilon \hat{q}} = \frac{\varepsilon \frac{1}{2}\|\hat{q} - q\|^2}{\varepsilon \hat{q}} = \hat{q} - q$$

$$\frac{\varepsilon W}{\varepsilon N_d} = \frac{\varepsilon W}{\varepsilon \hat{q}} \cdot \frac{\varepsilon \hat{q}}{\varepsilon N_d} = \frac{\varepsilon W}{\varepsilon \hat{q}} . \in' (N_d D + v_d). D \qquad (17)$$

$$\frac{\varepsilon W}{\varepsilon v_d} = \frac{\varepsilon W}{\varepsilon \hat{q}} \cdot \frac{\varepsilon \hat{q}}{\varepsilon v_d} = \frac{\varepsilon W}{\varepsilon \hat{q}} . \in' (N_d D + v_d)$$

The sigmoid function ∈ derivative is $\in'$, and $\in' (p) = \in (p)\big(1 - \in (p)\big)$. In this case, the parameters for the update can be set using Eq.(18).

$$N(f+1) = N(f) + \alpha \frac{\varepsilon W}{\varepsilon N} + \xi(f) \qquad (18)$$

Here,$\xi(f)$is a random term and α is the learning rate. Parameter v can also be kept using this approach. Second, the link rules were used once more to determine the slopes of $D, l,$ and Θ, yielding the results expressed in Eq.(19).

$$\frac{\varepsilon W}{\varepsilon D} = \frac{\varepsilon W}{\varepsilon \hat{q}} \cdot \frac{\varepsilon \hat{q}}{\varepsilon D} = \frac{\varepsilon W}{\varepsilon \hat{q}} . \in^{'} (N_d D + v_d). N_d$$

$$\frac{\varepsilon W}{\varepsilon l} = \frac{\varepsilon W}{\varepsilon \hat{q}} \cdot \frac{\varepsilon \hat{q}}{\varepsilon D} \cdot \frac{\varepsilon D}{\varepsilon l} = \frac{\varepsilon W}{\varepsilon D} \circ softmax(\Theta) \qquad (19)$$

$$\frac{\varepsilon W}{\varepsilon \Theta} = \frac{\varepsilon W}{\varepsilon \hat{q}} \cdot \frac{\varepsilon \hat{q}}{\varepsilon D} \cdot \frac{\varepsilon D}{\varepsilon \Theta} = \frac{\varepsilon W}{\varepsilon D} \circ l \circ \frac{\varepsilon softmax(\Theta)}{\varepsilon \Theta}$$

The following demonstrates how to determine a softmax function's partial derivative:

$$\frac{\varepsilon softmax(\Theta)}{\varepsilon \Theta} = \frac{\varepsilon}{\varepsilon \rho^s} \frac{exp(\rho^a)}{\sum_w exp(\rho^w)}$$
$$= \begin{cases} \frac{exp(\rho^a)\left[\left(\sum_w exp(\rho^w)\right) - exp(\rho^s)\right]}{\left(\sum_w exp(\rho^w)\right)^2} = softmax(\rho^a).\left(1 - softmax(\rho^s)\right) if\ s = a \\ \frac{0.\sum_w exp(\rho^w) - exp(\rho^a).exp(\rho^s)}{\left(\sum_w exp(\rho^w)\right)^2} = -softmax(\rho^a).softmax(\rho^s)\ if\ s \neq a \end{cases} \qquad (20)$$

After calculating the gradients of l and Θ, Back-Propagation techniques may be used to train and update RNNs and Autoencoder neural networks. Autoencoders, similar to Predictors, are trained via a Back-propagation technique. Due to the cyclical nature of time series, this research work will use a customized form of Live Data Recurrent Learning.

3.2 Dynamic Ant Colony Optimization

The classification accuracy attained by the RNN has attained lower results than expected results. So, this research work attempts to enhance its performance by utilizing the bio-inspired optimization strategy, Ant Colony Optimization (ACO). This research enhances the ACO in pheromone concentrations to achieve better classification accuracy than RNN.

ACO (Ant Colony Algorithm) was developed by Dorigo (1992). A metaheuristic algorithm mimics how ants find the quickest route to food. This is because ants release a chemical that causes the surrounding air to become extremely stale when seeking food. Ants are sensitive to the amount of the volatilization material used to choose their next foraging step. Ants always make their way to food along the specified path with the highest pheromone concentration. In other words, since there is a greater pheromone concentration along the path, more ants will opt to go along it. Ants can locate the quickest route based on the positive feedback system

3.2.1 Probability of a Transition

The ant infestation system is distinct from ACO. It chooses the most likely path between nodes and ants, known as the roulette selection path, at each transition point with a predetermined probability. x_0, with the constant parameter x_0 taking on values between 0 and 1. Eq.(21) is used to determine the next food source.

$$E = \begin{cases} argmax(\beta_{sw} . \alpha_{sw}) x \leq x_0 \\ M_{sw} x > x_0 \end{cases} \qquad (21)$$

If β_{sw} is the pheromone, α_{sw} is the heuristic function, $\alpha_{sw} = \frac{1}{y_{sw}}$ is the predicted gradation of ant transition, and y_{sw} increments the logical gap between the food source s and the food source w. On the range [0,1], x is a random number. Ants will move on to the node with coordinates x and x_0 that they have mutually determined. The ant transition probability is m_{sw}, and the greatest probability path is represented by arg max $(\beta_{sw}, \alpha_{sw})$.Here this research work utilizes the steps used to get at m_{sw}.

$$m_{sw} = \begin{cases} \frac{\beta_{sw}^{\delta} . \alpha_{sw}^{\gamma}}{\sum \beta_{sw}^{\delta} . \alpha_{sw}^{\gamma}} E > allow_a \\ 0 \qquad\qquad E < allow_a \end{cases} \qquad (22)$$

wherein δ represents the pheromone relevance factor that highlights the significance of pheromone accumulation. In the ant colony transition process, the value of the heuristic function or the distance between these nodes is represented by the factor γ. Furthermore, E has been specified in Eq. (21), where $allow_a$ indicates the route taken by the ath ant as it travels from food source s to food source w.

3.2.2 Pheromone Concentration

Once an individual ant finishes its trail, it deposits pheromones along it. However, when pheromones diffuse across nodes, their concentration drops. Since this pheromone concentration between nodes is updated after all ants have visited their routes, where ants can communicate with one another, this research details the revised formula in Eq.(23) and Eq.(24).

$$\beta_{sw} = (1 - \varphi)\beta_{sw} + \varphi \Delta \beta_{sw} \qquad (23)$$

$$\Delta \beta_{sw}^{a}(f) = \frac{X}{route_{Length}} \qquad (24)$$

wherein φ is the pheromone evaporation, and it is the coefficient that specifies the concentration of pheromone volatilization following each cycle, $\Delta\beta_{sw}$ is the remaining quantity of pheromones, f is the present iteration count, $\Delta\beta_{sw}^{a}(f)$ is the remaining pheromone amount for the ath ant between food source s and food source w, X is the total amount of path length, denoted as $route_{Lengt}$, is used to calculate the residual pheromone concentration between food source s and food source w.

3.2.3 Quantifying Pheromones

The individual ant's route distance is treated as the best fitness. It is used to find the best combinations of the pheromone significance factor (δ) and the pheromone evaporation coefficient (φ). The quantity of pheromone X is determined in each iteration by using the optimum global value, which is found using Eq.(25).

$$X = \frac{d*\gamma^{1.8}}{\varphi} \quad (25)$$

where in δ is the pheromone significance factor, φ is the coefficient of volatilization of pheromones, and γ the significance factor of heuristic functions.

4 ABOUT THE DATASET

A dataset in machine learning also refers to a structured collection of data used to train an AI model typically represented in a table format, with rows and columns, where each column is a feature, and each row represents an individual sample.

4.1 Soybean Crop Dataset

The first dataset used for this work contains information regarding the Soybean crop cultivated along with the nine states of the US Corn belt. The dataset holds 25345 records and three different features, namely: (a) Soil Feature, (b) Weather Feature, and (c) Crop Management Feature.

4.1.1 Soil feature

The data present in this feature contains 11 variables measured at six different depths in 250 square meter resolution, which are: (i) 0-5cm, (ii) 5-15cm, (iii) 15-30cm, (iv) 30-60cm, (v) 60-100cm, and (vi) 100-200cm.

Table 1: Six Different Depths

Interval	Top Depth(cm)	Bottom Depth(cm)
I	0	5
II	5	15
III	15	30
IV	30	60
V	60	100
VI	100	200

Table 2:Soil Features

Features	Units	Description
Bulk Density (bdod)	Centigram Per Cubic Centimetre	Density in a volume of the fine-grained earth
Cation Exchange Capacity (cec)	Millimole Per Kilogram	Soil pH7 CEC: Cation Exchange Capacity (i.e., the potential of Hydrogen value-7)
Coarse Fragments (cfvo)	Cubic Centimetre Per Cubic Decimetre	The proportion of coarse particles (> 2 mm) by volume.
Clay	Gram Per Kilogram	Percentage of clay particles in the fine earth fraction
Total Nitrogen (Nitrogen)	Centigram Per Kilogram	N_2O (i.e., the Entire Gas)
Organic Carbon Density (ocd)	Kilogram Per Cubic Meter	The molecular weight of carbon in organic compounds
Organic Carbon Stock (ocs)	Kilogram Per Square Meter	Carbon Remaining in Organic Matter
pH in H_20	Percentage	Hydrogen's Untapped Potential in Water
Sand	Gram Per Kilogram	Sand's representation in the fine earth's particle distribution
Silt	Gram Per Kilogram	Microscopic particles of silt relative to those of fine earth
Soil Organic Carbon (soc)	Decigram Per Kilogram	The fine-grained organic carbon content of the soil

4.1.2 Weather feature

The data present in this feature contains five components which are measured throughout the year.

Table 3:Weather Features

Features	Units	Description
Precipitation	Millimetre	Week typical precipitation
Solar Radiation	Watt Per Square Meter	Surfaces are exposed to solar energy
Maximum Temperature	Degree Centigrade	Weekly mean highest temperature
Minimum Temperature	Degree Centigrade	Weekly mean low temperature
Vapor Pressure	Pascal	Speed of a liquid's evaporation

4.1.3 Crop Management feature

Each year beginning in April, this feature provides statistics on the cumulative proportion of planted fields throughout each state over 52 weeks.

Table 4:Crop Management Features

Features	Units	Description
Weekly Cumulative Percentage of Planted-fields (CWPP)	Percentage	The average weekly proportion of fields that have been planted

4.2 Rice Crop Dataset

The research utilized a second dataset that includes data on rice crop cultivation across 20 districts in Tami Nadu between 1990 and 2015, with 520 records and two features: soil and weather. Through replicating records in the dataset multiple times, this research work was able to extensively analyze classifier performance and reach a dataset size of 10400 records.

4.2.1 Soil feature

The soil data includes six different characteristics recorded over a year from 1990 to 2015.

Table 5:Soil Feature

Features	Units	Description
Irrigated area of rice	Ha	Rice Irrigated Area
Consumption of Nitrogen	Tons	NitrogenFertilizer
Consumption of Potassium	Tons	PhosphateFertilizer
Consumption of Potash	Tons	PotashFertilizer
Presence of OrthidSoil	Percentage	Orthid Soil Level
Presence of Sandy Soil	Percentage	Sandy Alfisol Soil Level

4.2.2 Weather feature

The five elements of the weather are tracked over 12 months, spanning from 1990 to 2015.

Table 6:Weather Feature

Features	Units	Description
JAN_MR - DEC_MR	mm	Rainfall in Months
JAN_MT - DEC_MT	cg	Temperature at Minimum Level
JAN_XT - DEC_XT	cg	Temperature at Maximum Level
JAN_PT - DEC_PT	mm	Precipitation
JAN_ET - DEC_ET	mm	Evapotranspiration Potential

5 PERFORMANCE METRICS

The performance of the newly proposed classifier is measured using the metric outlined below compared to the existing classifier.

- **Classification Accuracy (CA):**A ratio of the number of correct predictions made by a classifier to the total number of predictions made. It can be expressed as a percentage, where a higher percentage indicates better performance.
- **F-Measure (FM):** It is also known as the F1 score. It is a measure of a test's accuracy. It considers the test's precision and recalls to compute the score. The F1 score is the harmonic mean of precision and recall, where an F1 Score reaches its best value at 1 (perfect precision and recall) and worst at 0.
- **Matthews correlation coefficient (MCC):** It measures the quality of binary (two-class) classifications, which ranges from -1 to 1. The value 1 indicates a perfect prediction, 0 indicates a random prediction, and -1 indicates total disagreement between prediction and observation.
- **Fowlkes-Mallows Index (FMI):** It is a comparison metric to evaluate the similarity between different clustering algorithms or the same algorithm with different parameter settings.

6 RESULTS AND DISCUSSION

6.1 CA and FM Analysis

The performance of classifiers using soybean and rice crop datasets is examined in terms of CA and FM through figures 1 and 2. It is clear from the data presented in these figures that the proposed classifier performs better due to its optimization strategy. In contrast, the RFA and SVM classifiers perform worse due to a lack of optimization. The

result values shown in figures 1 and 2 can also be found in table 7.

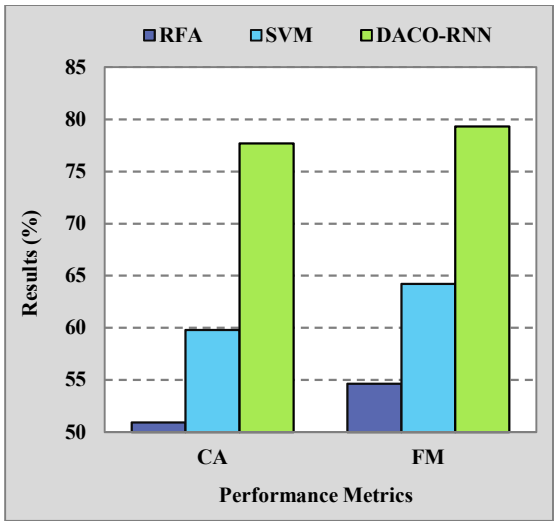

Figure 1: Analysis of CA and FM on Soybean Crop Dataset

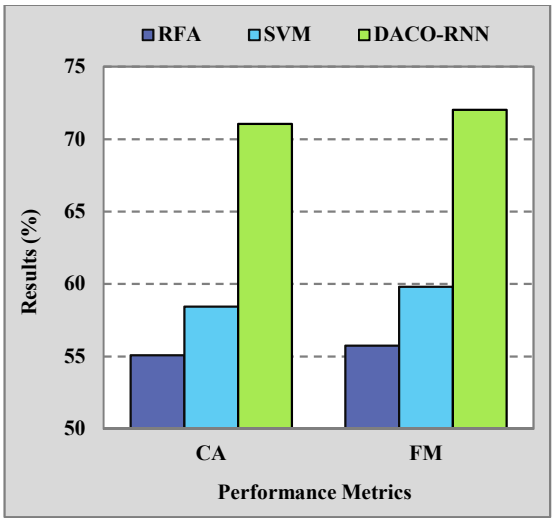

Figure 2: Analysis of CA and FM on Rice Crop Dataset

Table 7:Result Values of CA and FM

Dataset →	Soybean Crop Dataset		Rice Crop Dataset	
Metrics→ / Classifiers↓	CA (%)	FM (%)	CA (%)	FM (%)
RFA	50.925	54.632	55.077	55.749
SVM	59.795	64.210	58.442	59.818
DACO-RNN	77.688	79.323	71.048	72.014

6.2 FMI and MCC Analysis

The performance of classifiers for soybean and rice crop datasets is evaluated in terms of FMI and MCC in Figure 3 and Figure 4. It can be seen from the results that the proposed classifier is reliable and can be employed for further analysis. However, the results obtained from RFA and SVM classifiers suggest that additional improvements are necessary. The values of the results presented in Figure 3 and Figure 4 are also provided in Table 8.

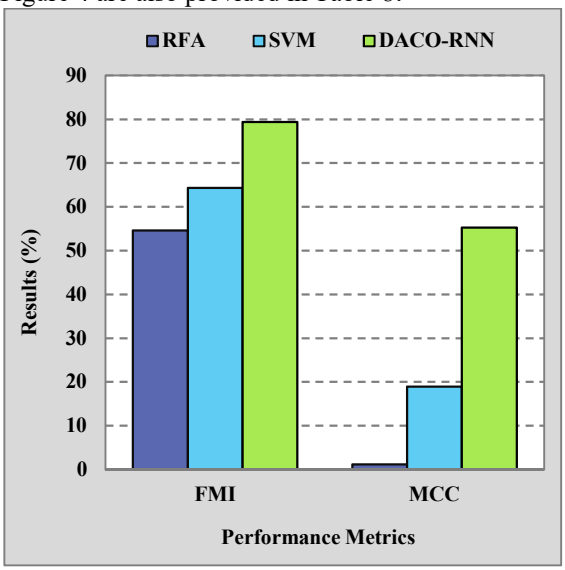

Figure 3: Analysis of FMI and MCC on Soybean Crop Dataset

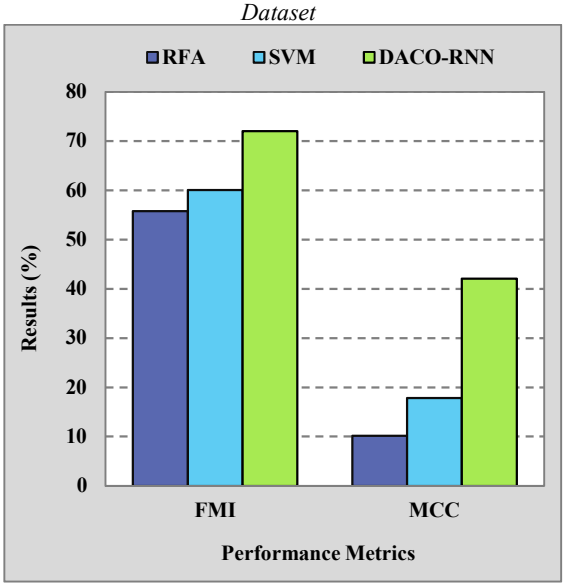

Figure 4: Analysis of FMI and MCC on Rice Crop Dataset

Table 8: Result Values of CA and FM

Dataset →	Soybean Crop Dataset		Rice Crop Dataset	
Metrics→ / Classifiers↓	FMI (%)	MCC (%)	FMI (%)	MCC (%)
RFA	54.633	1.194	55.750	10.135
SVM	64.336	18.943	60.060	17.805
DACO-RNN	79.361	55.269	72.020	42.051

7 CONCLUSION

Machine learning models are highly complex and difficult to interpret for crop yield prediction. This can make it challenging to understand the reasons behind a model's predictions and to identify potential errors or biases in the model. Crop yields are a complex task that requires a good understanding of the underlying physical processes and accurate data on current and future climate conditions.This study suggests using a combination of Dynamic Ant Colony Optimization and Recurrent Neural Network (DACO-RNN) to anticipate crop production. The modified ACO approach improves the performance of RNN, leading to higher classification accuracy. The effectiveness of DACO-RNN is demonstrated by testing it on two commonly used datasets for predicting crop yields. Results show that it outperforms other methods in classification accuracy, making it a valuable tool for farmers to forecast crop yields.

REFERENCES:

[1] S. Khanal, J. Fulton, A. Klopfenstein, N. Douridas, and S. Shearer, "Integration of high resolution remotely sensed data and machine learning techniques for spatial prediction of soil properties and corn yield," *Comput. Electron. Agric.*, vol. 153, pp. 213–225, 2018, doi: 10.1016/j.compag.2018.07.016.

[2] P. S. Maya Gopal and R. Bhargavi, "A novel approach for efficient crop yield prediction," *Comput. Electron. Agric.*, vol. 165, p. 104968, 2019, doi: 10.1016/j.compag.2019.104968.

[3] Y. Cai *et al.*, "Integrating satellite and climate data to predict wheat yield in Australia using machine learning approaches," *Agric. For. Meteorol.*, vol. 274, pp. 144–159, 2019, doi: 10.1016/j.agrformet.2019.03.010.

[4] A. Avneri *et al.*, "UAS-based imaging for prediction of chickpea crop biophysical parameters and yield," *Comput. Electron. Agric.*, vol. 205, p. 107581, 2023, doi: 10.1016/j.compag.2022.107581.

[5] H. Guan *et al.*, "An improved approach to estimating crop lodging percentage with Sentinel-2 imagery using machine learning," *Int. J. Appl. Earth Obs. Geoinf.*, vol. 113, p. 102992, 2022, doi: 10.1016/j.jag.2022.102992.

[6] J. Cao *et al.*, "Integrating Multi-Source Data for Rice Yield Prediction across China using Machine Learning and Deep Learning Approaches," *Agric. For. Meteorol.*, vol. 297, p. 108275, 2021, doi: 10.1016/j.agrformet.2020.108275.

[7] R. Jaganathan and R. Vadivel, "Intelligent Fish Swarm Inspired Protocol (IFSIP) for Dynamic Ideal Routing in Cognitive Radio Ad-Hoc Networks," *Int. J. Comput. Digit. Syst.*, vol. 10, no. 1, pp. 1063–1074, 2021, doi: 10.12785/ijcds/100196.

[8] J. Ramkumar and R. Vadivel, "Improved Wolf prey inspired protocol for routing in cognitive radio Ad Hoc networks," *Int. J. Comput. Networks Appl.*, vol. 7, no. 5, pp. 126–136, 2020, doi: 10.22247/ijcna/2020/202977.

[9] J. Ramkumar and R. Vadivel, "Whale optimization routing protocol for minimizing energy consumption in cognitive radio wireless sensor network," *Int. J. Comput. Networks Appl.*, vol. 8, no. 4, pp. 455–464, 2021, doi: 10.22247/ijcna/2021/209711.

[10] J. Ramkumar and R. Vadivel, "Multi-Adaptive Routing Protocol for Internet of Things based Ad-hoc Networks," *Wirel. Pers. Commun.*, vol. 120, no. 2, pp. 887–909, Apr. 2021, doi: 10.1007/s11277-021-08495-z.

[11] R. Jaganathan and V. Ramasamy, "Performance modeling of bio-inspired routing protocols in Cognitive Radio Ad Hoc Network to reduce end-to-end delay," *Int. J. Intell. Eng. Syst.*, vol. 12, no. 1, pp. 221–231, 2019, doi: 10.22266/IJIES2019.0228.22.

[12] P. Menakadevi and J. Ramkumar, "Robust Optimization Based Extreme Learning Machine for Sentiment Analysis in Big Data," *2022 Int. Conf. Adv. Comput. Technol. Appl. ICACTA 2022*, pp. 1–5, Mar. 2022, doi: 10.1109/ICACTA54488.2022.9753203.

[13] J. Ramkumar, C. Kumuthini, B. Narasimhan, and S. Boopalan, "Energy Consumption Minimization in Cognitive Radio Mobile Ad-Hoc Networks using Enriched Ad-hoc On-

demand Distance Vector Protocol," *2022 Int. Conf. Adv. Comput. Technol. Appl. ICACTA 2022*, pp. 1–6, Mar. 2022, doi: 10.1109/ICACTA54488.2022.9752899.

[14] R. J, "Meticulous Elephant Herding Optimization based Protocol for Detecting Intrusions in Cognitive Radio Ad Hoc Networks," *Int. J. Emerg. Trends Eng. Res.*, vol. 8, no. 8, pp. 4548–4554, 2020, doi: 10.30534/ijeter/2020/82882020.

[15] J. Ramkumar, "Bee inspired secured protocol for routing in cognitive radio ad hoc networks," *Indian J. Sci. Technol.*, vol. 13, no. 30, pp. 2159–2169, 2020, doi: 10.17485/ijst/v13i30.1152.

[16] J. Ramkumar and R. Vadivel, "Improved frog leap inspired protocol (IFLIP) – for routing in cognitive radio ad hoc networks (CRAHN)," *World J. Eng.*, vol. 15, no. 2, pp. 306–311, 2018, doi: 10.1108/WJE-08-2017-0260.

[17] J. Ramkumar, R. Vadivel, and B. Narasimhan, "Constrained Cuckoo Search Optimization Based Protocol for Routing in Cloud Network," *Int. J. Comput. Networks Appl.*, vol. 8, no. 6, pp. 795–803, 2021, doi: 10.22247/ijcna/2021/210727.

[18] J. Ramkumar and R. Vadivel, "CSIP—cuckoo search inspired protocol for routing in cognitive radio ad hoc networks," in *Advances in Intelligent Systems and Computing*, 2017, vol. 556, pp. 145–153. doi: 10.1007/978-981-10-3874-7_14.

[19] R. Jaganathan, V. Ramasamy, L. Mani, and N. Balakrishnan, "Diligence Eagle Optimization Protocol for Secure Routing (DEOPSR) in Cloud-Based Wireless Sensor Network," 2022, doi: 10.21203/rs.3.rs-1759040/v1.

[20] L. Li *et al.*, "Crop yield forecasting and associated optimum lead time analysis based on multi-source environmental data across China," *Agric. For. Meteorol.*, vol. 308–309, p. 108558, 2021, doi: 10.1016/j.agrformet.2021.108558.

[21] K. Berger *et al.*, "Retrieval of aboveground crop nitrogen content with a hybrid machine learning method," *Int. J. Appl. Earth Obs. Geoinf.*, vol. 92, p. 102174, 2020, doi: 10.1016/j.jag.2020.102174.

[22] Y. Ma, Z. Zhang, Y. Kang, and M. Özdoğan, "Corn yield prediction and uncertainty analysis based on remotely sensed variables using a Bayesian neural network approach," *Remote Sens. Environ.*, vol. 259, p. 112408, 2021, doi: 10.1016/j.rse.2021.112408.

[23] Y. Ma, Z. Zhang, H. L. Yang, and Z. Yang, "An adaptive adversarial domain adaptation approach for corn yield prediction," *Comput. Electron. Agric.*, vol. 187, p. 106314, 2021, doi: 10.1016/j.compag.2021.106314.

[24] M. Maimaitijiang, V. Sagan, P. Sidike, S. Hartling, F. Esposito, and F. B. Fritschi, "Soybean yield prediction from UAV using multimodal data fusion and deep learning," *Remote Sens. Environ.*, vol. 237, p. 111599, 2020, doi: 10.1016/j.rse.2019.111599.

[25] M. Nandhini, K. U. Kala, M. Thangadarshini, and S. Madhusudhana Verma, "Deep Learning model of sequential image classifier for crop disease detection in plantain tree cultivation," *Comput. Electron. Agric.*, vol. 197, p. 106915, 2022, doi: 10.1016/j.compag.2022.106915.

[26] Z. Ma *et al.*, "Recognition methods of threshing load conditions based on machine learning algorithms," *Comput. Electron. Agric.*, vol. 200, p. 107250, 2022, doi: 10.1016/j.compag.2022.107250.

[27] B. Wang, P. Feng, C. Waters, J. Cleverly, D. L. Liu, and Q. Yu, "Quantifying the impacts of pre-occurred ENSO signals on wheat yield variation using machine learning in Australia," *Agric. For. Meteorol.*, vol. 291, p. 108043, 2020, doi: 10.1016/j.agrformet.2020.108043.

[28] M. Dogan, Y. S. Taspinar, I. Cinar, R. Kursun, I. A. Ozkan, and M. Koklu, "Dry bean cultivars classification using deep cnn features and salp swarm algorithm based extreme learning machine," *Comput. Electron. Agric.*, vol. 204, p. 107575, 2023, doi: 10.1016/j.compag.2022.107575.

[29] A. Mateo-Sanchis, M. Piles, J. Muñoz-Marí, J. E. Adsuara, A. Pérez-Suay, and G. Camps-Valls, "Synergistic integration of optical and microwave satellite data for crop yield estimation," *Remote Sens. Environ.*, vol. 234, p. 111460, 2019, doi: 10.1016/j.rse.2019.111460.

[30] L. Breiman, "Random forests," *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, Oct. 2001, doi: 10.1023/A:1010933404324/METRICS.

[31] B. E. Boser, I. M. Guyon, and V. N. Vapnik, "A Training Algorithm for Optimal Margin Classifiers",COLT '92: Proceedings of the fifth annual workshop on Computational learning theory, 1992, pp. 144–152.