The Scientific Temper (2024) Vol. 15 (4): 3173-3181

E-ISSN: 2231-6396, ISSN: 0976-8653

Doi: 10.58414/SCIENTIFICTEMPER.2024.15.4.26

https://scientifictemper.com/

RESEARCH ARTICLE

A lightweight selective stacking framework for IoT crop recommendation

S. Sindhu*, L. Arockiam

Abstract

The integration of the internet of things (IoT) into precision farming has revolutionized agricultural practices by enhancing resource management and crop productivity through real-time data monitoring and automation. Precision farming leverages IoT technologies to monitor critical environmental factors such as soil moisture, temperature, humidity, and nutrient levels, enabling farmers to make data-driven decisions for optimizing crop growth. Despite these advancements, improving the accuracy and efficiency of IoT-based crop recommendation systems remains a key challenge, particularly in resource-constrained environments. This study aims to enhance the predictive performance of IoT-based crop recommendation systems by developing a novel stacked deep ensemble learning model. The proposed model, termed lightweight selective stacking with deep ensemble learning (LSSDEL), focuses on reducing computational complexity while maintaining high predictive accuracy. Key methods employed include selective model stacking, L1 regularization for model pruning, gradient-free model aggregation, and the implementation of an early stopping mechanism. The system is validated using real-world IoT agricultural datasets, emphasizing its scalability and practical applicability. Findings from the study demonstrate that the LSSDEL model outperforms traditional models, achieving a prediction accuracy of 97.80% and significant improvements in precision, recall, and F1-score. Furthermore, the proposed model reduces execution time by 16.7% compared to existing approaches, confirming its computational efficiency.

Keywords: Crop recommendation, Deep learning, Ensemble learning, IoT, Precision farming.

Introduction

The internet of things (IoT) is a groundbreaking technology that connects devices, systems, and sensors, enabling seamless communication in real time (Ranjan *et al*., 2024). Among its diverse applications, IoT has made a significant impact on precision farming, an advanced approach to managing agricultural resources (Vinod *et al*., 2024). Precision farming leverages technologies to monitor crop growth, soil conditions, and environmental factors, allowing for data-driven decision-making that enhances crop productivity and optimizes resource usage (Getahun *et al*., 2024). IoT's role in precision farming is pivotal as it provides continuous, real-time data, enabling farmers to make informed adjustments that lead to more efficient and sustainable farming practices (Soussi *et al*., 2024).

A key component of precision farming is the use of sensors embedded in the environment to monitor critical factors like soil moisture, temperature, humidity, and nutrient levels (Mehedi *et al*., 2024). IoT enables the collection and analysis of this data in real time, helping farmers respond quickly to changing conditions (Fuentes *et al*., 2024). For example, IoT devices can notify farmers when irrigation is needed or when specific nutrients are lacking, facilitating precise interventions that improve crop health and minimize resource wastage. This real-time insight allows for more effective management of resources and ensures that crops receive exactly what they need to thrive.

Beyond data collection and monitoring, IoT plays a crucial role in automating farming processes (Dhal *et al*., 2024). By integrating IoT systems with automated equipment, such as irrigation systems or fertilizing drones, farming operations can be managed with greater efficiency (Mohamed *et al*., 2021). This reduces the margin for human

Department of Computer Science, St. Joseph's College (Autonomous) Affiliated to Bharathidasan University, Tiruchirappalli, Tamil Nadu, India, India.

***Corresponding Author:** S. Sindhu, Department of Computer Science, St. Joseph's College (Autonomous) Affiliated to Bharathidasan University, Tiruchirappalli, Tamil Nadu, India, India., E-Mail: sindhusamikannu.04@gmail.com

How to cite this article: Sindhu, S., Arockiam, L. (2024). A lightweight selective stacking framework for IoT crop recommendation. The Scientific Temper, **15**(4):3173-3181.

Doi: 10.58414/SCIENTIFICTEMPER.2024.15.4.26

Source of support: Nil

Conflict of interest: None.

Received: 10/11/2024 **Accepted:** 25/11/2024 **Published** : 20/12/2024

error and ensures that resources like water, fertilizers, and pesticides are applied precisely where and when they are needed (Ahmad *et al*., 2020). This level of automation not only boosts crop yields but also promotes more sustainable farming practices by minimizing the overuse of resources (Getahun *et al*., 2024).

One of the key applications of IoT in precision farming is crop recommendation, a process where data from various sensors is used to determine the best crops to plant based on environmental and soil conditions (Alahmad *et al*., 2023). Crop recommendation systems consider factors such as the type of soil, the amount of rainfall, temperature patterns, and market demands to suggest the most suitable crops for a given region (Pande *et al*., 2021). These systems help farmers make informed decisions, ensuring that they choose crops that are likely to thrive in specific conditions, thus reducing the risk of crop failure and improving overall productivity (Altieri *et al*., 2015).

IoT-based crop recommendation systems utilize data analytics and machine learning algorithms to process the information gathered from the environment (Musanase *et al*., 2023). These systems can continuously learn from past data, becoming more accurate in predicting which crops will yield the best results (Van *et al*., 2020). Additionally, IoT platforms can provide farmers with actionable insights through easy-to-use dashboards, ensuring that even those with limited technical knowledge can benefit from the technology (Paul *et al*., 2022).

The role of IoT in addressing these challenges cannot be overstated. By offering real-time monitoring and data analysis, IoT allows for the more efficient use of resources, ensuring that farming is both sustainable and economically viable. Precision farming solutions that incorporate IoT can help reduce water consumption, optimize fertilizer application, and minimize the environmental impact of farming. Moreover, IoT-driven crop recommendation systems ensure that the right crops are grown in the right conditions, thereby enhancing food security.

In the work by Kiruthika and Karthika (2023), the IDCSO-WLSTM approach was proposed for crop recommendation. This model utilized a deep learning-based weighted long short-term memory (WLSTM) framework, enhanced by an improved dynamic crow search optimization (IDCSO) algorithm. The model demonstrated improved prediction performance over traditional models but still faced challenges in terms of scalability and computational efficiency, particularly when handling high-dimensional datasets (Kiruthika *et al*., 2023). Despite its merits, the IDCSO-WLSTM approach had limitations in optimizing model complexity and computational resource usage, making it less ideal for real-time IoT systems.

To address the above research gap, the objectives of this paper are as follows:

- To improve the predictive accuracy of IoT-based crop recommendation systems by leveraging a stacked deep ensemble learning approach.
- To reduce computational complexity through selective model stacking, model pruning, and regularization techniques.
- To apply gradient-free model aggregation and early stopping to ensure optimal model performance with minimal computational overhead.
- To validate the proposed LSSDEL model using real-world IoT agricultural datasets and compare it with existing methods.

The significance of this research lies in its potential to contribute to the development of more effective IoT-based crop recommendation systems. These systems can have a profound impact on smallholder farmers, who often struggle with limited access to resources and information. By providing them with data-driven recommendations, IoT technologies can help level the playing field, ensuring that even small farms can compete with larger industrial agricultural operations. This democratization of technology in agriculture could lead to greater economic stability for rural communities and contribute to reducing poverty in agricultural regions.

This paper is structured as follows: Section 2 presents the methodology, describing the LSSDEL approach, including its model stacking, pruning, and voting mechanisms. Section 3 details the experimental setup and results, with a discussion of the performance of the proposed method. Section 4 summarizes the key findings and contributions of this chapter.

Methodology

The proposed LSSDEL technique follows a structured workflow designed to enhance the predictive accuracy of IoT-based crop recommendation systems while maintaining computational efficiency. The micro-level architecture of the proposed work integrates several key components, each playing a critical role in the overall model training and prediction process (see Figure 1). The workflow begins by selecting and preprocessing the input dataset,

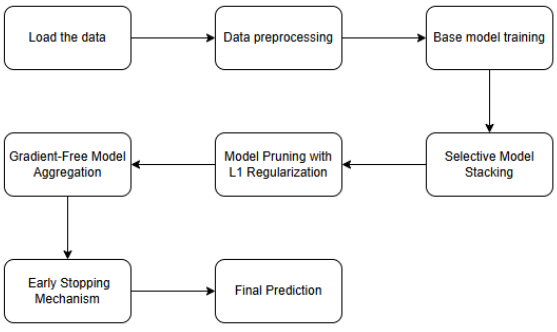

Figure 1: Workflow of LSSDEL

which consists of environmental and soil-related features (e.g., nitrogen, phosphorus, potassium, temperature, pH, humidity, and rainfall) and crop labels. After preprocessing, a set of base models is trained using multiple machine learning algorithms. These models undergo a selection process, where only the top-performing ones are chosen for inclusion in the ensemble.

To optimize the performance of the selected models, L1 regularization is applied for pruning, reducing unnecessary complexity by eliminating irrelevant or redundant features. This is followed by gradient-free model aggregation, where the predictions of the pruned models are combined through a weighted voting mechanism based on their cross-validation performance. Additionally, an early stopping mechanism is implemented to prevent overfitting and ensure that each model is trained only until further improvements are minimal. This not only reduces the computational cost but also enhances the efficiency of the ensemble.

Selective Model Stacking

Selective model stacking is a core component of the proposed LSSDEL technique, designed to optimize the integration of multiple machine learning models in the context of IoT-based crop recommendation systems. The primary goal of this approach is to enhance predictive accuracy by leveraging the strengths of various base models while maintaining computational efficiency. Unlike conventional stacking methods that indiscriminately combine all models, the selective approach introduced here applies a rigorous selection process based on performance metrics, ensuring that only the top-performing models contribute to the final prediction.

Let $D = \{(x_i, y_i)\}_{i=1}^{N}$ be the dataset where $x_i \in R^d$ represents the i-th instance with d features and $y_i \in R$ is the corresponding label. The base models $M = \{M_1, M_2, \ldots, M_n\}$ are trained on the dataset $D_{\text{train}} \subset D$. Each model M_j is trained to minimize a loss function $\mathcal{L}(M_j, D_{\text{train}})$ and provides predictions $\widehat{y_{ij}} = M_j(x_i)$ for the validation set $D_{\text{val}} \subset D$.

The performance of each model M_j is evaluated using k-fold cross-validation. For each fold k, the accuracy of M_j is given by:

$$P_{M_j} = \frac{1}{K}\sum_{k=1}^{K} \text{Accuracy}\left(M_j, D_k^{\text{val}}\right) \tag{1}$$

where D_k^{val} is the validation fold in the k-th iteration of the cross-validation. The top k models with cross-validation accuracy exceeding a threshold T_{acc} are selected for the stacking ensemble:

$$M_{\text{selected}} = \{M_j \in M \mid P_{M_j} > T_{\text{acc}}\} \tag{2}$$

To further optimize the selection process, the pruned models undergo L1 regularization during training. The regularization term $\lambda \sum_{j=1}^{d} |w_j|$ is added to the loss function to penalize unnecessary model complexity, where w_j denotes the weight vector associated with the features in model M_j. The objective function to be minimized for each model becomes:

$$\mathcal{L}_{reg}\left(M_j, D_{\text{train}}\right) = \mathcal{L}\left(M_j, D_{\text{train}}\right) + \lambda \sum_{j=1}^{d} |w_j| \tag{3}$$

After model selection and pruning, the predictions of the selected models are combined using weighted voting. Let $\widehat{y_{ij}}$ denote the prediction of the i-th instance by the j-th model. The final prediction $\hat{y}_i$ for instance x_i is obtained by aggregating the predictions of the selected models, where each model's vote is weighted according to its cross-validation performance:

$$\hat{y}_i = \arg\max_c \sum_{M_j \in M_{\text{selected}}} P_{M_j} \cdot \mathbb{1}\{\widehat{y_{ij}} = c\} \tag{4}$$

where $\mathbb{1}\{\cdot\}$ is the indicator function, and c denotes the possible classes (in this case, crop types). This weighted voting mechanism ensures that models with higher performance contribute more to the final prediction, thus improving the ensemble's overall accuracy.

Furthermore, to ensure that the training process remains efficient, the training of each model incorporates an early stopping mechanism. The validation loss $\mathcal{L}_{val}\left(M_j, D_{\text{val}}\right)$ is monitored at each iteration t, and the training is halted if the validation loss fails to improve by a margin ϵ over a predefined number of iterations T_{patience}:

$$\text{Stop } M_j \text{ if } \mathcal{L}_{val}\left(M_j, t\right) - \mathcal{L}_{val}\left(M_j, t-1\right) < \epsilon, \quad \forall t > T_{\text{patience}} \tag{5}$$

By combining selective model stacking, L1 regularization, and weighted voting, the LSSDEL technique effectively reduces model complexity while maximizing predictive performance, making it ideal for real-time IoT-based crop recommendation systems where both accuracy and efficiency are critical.

Model Pruning with L1 Regularization

Model pruning is a critical step in optimizing machine learning models, particularly in ensemble learning frameworks such as the LSSDEL approach. The primary objective of model pruning is to eliminate unnecessary parameters, reduce overfitting, and enhance computational efficiency without sacrificing predictive accuracy. In the LSSDEL technique, pruning is achieved using L1 regularization, which imposes a sparsity constraint on the model's weight parameters. This section outlines the mathematical formulation and application of L1 regularization in the context of model pruning.

Let M_j be one of the selected base models in the ensemble, trained on the dataset $D = \{(x_i, y_i)\}_{i=1}^{N}$, where $x_i \in R^d$ are the feature vectors, and $y_i \in R$ are the corresponding labels. The weight vector for model M_j is denoted by

$w_j = \{w_{j1}, w_{j2}, \ldots, w_{jd}\}$, representing the contributions of each feature in the prediction process.

The goal of L1 regularization is to minimize the following regularized loss function:

$$\mathcal{L}_{L1}(M_j, D_{\text{train}}) = \mathcal{L}(M_j, D_{\text{train}}) + \lambda \sum_{k=1}^{d} |w_{jk}| \quad (6)$$

where $\mathcal{L}(M_j, D_{\text{train}})$ is the original loss function (e.g., mean squared error or logistic loss), and λ is the regularization parameter that controls the degree of penalization. The term $\sum_{k=1}^{d} |w_{jk}|$ introduces sparsity by driving some weights w_{jk} towards zero, effectively removing the influence of less important features from the model.

The optimization of the L1-regularized loss function is performed using gradient-based methods. The gradient of the regularized loss with respect to the weights is given by:

$$\frac{\partial \mathcal{L}_L}{\partial w_{jk}} = \frac{\partial \mathcal{L}}{\partial w_{jk}} + \lambda \cdot \text{sign}(w_{jk}) \quad (7)$$

where $\frac{\partial \mathcal{L}}{\partial w_{jk}}$ is the gradient of the original loss function, and $\text{sign}(w_{jk})$ is the signum function, defined as:

$$\text{sign}(w_{jk}) = \begin{cases} 1, & w_{jk} > 0 \\ -1, & w_{jk} < 0 \\ 0, & w_{jk} = 0 \end{cases} \quad (8)$$

This gradient update pushes small weights toward zero, thereby pruning irrelevant or redundant features. The iterative update rule for the weights in model M_j becomes:

$$w_{jk}^{(t+1)} = w_{jk}^{(t)} - \eta \left(\frac{\partial \mathcal{L}}{\partial w_{jk}} + \lambda \cdot \text{sign}(w_{jk}) \right) \quad (9)$$

where η is the learning rate, and t denotes the iteration step. This process continues until convergence, with many of the weights becoming zero because of the L1 regularization term, effectively pruning the model.

The advantage of L1 regularization lies in its ability to produce sparse solutions. For models with high-dimensional inputs, such as IoT crop recommendation systems where features like soil nutrients, temperature, humidity, and pH levels are highly variable, L1 regularization ensures that only the most relevant features are retained. This reduces the computational complexity of the model while maintaining or even improving its predictive accuracy.

To further illustrate, consider the pruned model M_j after L1 regularization. Let the set of non-zero weights be denoted as $w_j^* = \{w_{jk} \mid w_{jk} \neq 0\}$. The prediction for a new instance x_i after pruning is then given by:

$$\hat{y}_i = f_j(w_j^*, x_i) = \sum_{k=1}^{d^*} w_{jk}^* \cdot x_{ik} \quad (10)$$

where d^* is the number of remaining features after pruning, and x_{ik} is the k-th feature of the i-th instance. This sparse representation reduces the computational burden during both training and inference.

Additionally, the effectiveness of L1 regularization in pruning models can be further quantified by measuring the ratio of pruned weights:

$$\text{Pruned\_Ratio} = \frac{d - d^*}{d} \quad (11)$$

where d is the total number of features before pruning, and d^* is the number of features remaining after pruning. A high pruning ratio indicates a significant reduction in model complexity, which is crucial in resource-constrained IoT environments.

Gradient-Free Model Aggregation

Gradient-free model aggregation is a crucial aspect of the LSSDEL technique, designed to integrate the predictions of multiple base models without relying on gradient-based optimization techniques. This approach is particularly well-suited for ensemble models in IoT-based systems where computational efficiency and scalability are essential. Instead of using traditional gradient-based methods to adjust the contributions of individual models, gradient-free aggregation leverages a weighted voting scheme that assigns higher importance to models based on their cross-validation performance, thus ensuring that the ensemble prediction is robust and accurate.

Let the set of selected base models be denoted by $M_{\text{selected}} = \{M_1, M_2, \ldots, M_k\}$, where k represents the number of models selected based on cross-validation accuracy. For each model $M_j \in M_{\text{selected}}$, it assigns a weight w_j that reflects its performance during cross-validation. The weight is proportional to the accuracy of the model, given by:

$$w_j = \frac{P_{M_j}}{\sum_{j=1}^{k} P_{M_j}} \quad (12)$$

where P_{M_j} is the cross-validation accuracy of model M_j, and the sum of the weights for all selected models is normalized to 1.

Given an input instance $x_i \in R^d$, each model M_j generates a prediction $\widehat{y_{ij}} = M_j(x_i)$. The final aggregated prediction $\hat{y}_i$ for the instance x_i is obtained by taking a weighted sum of the individual model predictions. For a regression task, this can be expressed as:

$$\hat{y}_i = \sum_{j=1}^{k} w_j \cdot \widehat{y_{ij}} \quad (13)$$

In the case of a classification task, the final prediction is determined using a weighted voting mechanism. Let C denote the set of possible classes (e.g., crop types in the IoT-based agricultural dataset), and let $\mathbb{1}\{\widehat{y_{ij}} = c\}$ be the indicator function that takes the value 1 if model M_j predicts class c for instance x_i, and 0 otherwise. The aggregated prediction $\hat{y}_i$ is given by:

$$\hat{y}_i = \arg\max_{c \in C} \sum_{j=1}^{k} w_j \cdot \mathbb{1}\{\widehat{y_{ij}} = c\} \quad (14)$$

This weighted voting scheme ensures that models with higher accuracy during cross-validation contribute more heavily to the final prediction, thus enhancing the overall performance of the ensemble.

An essential feature of gradient-free model aggregation is its ability to handle the variance and biases of individual models. By assigning weights based on cross-validation accuracy, the approach naturally adjusts for models that may perform well on specific portions of the dataset but poorly on others. This is particularly useful in IoT agricultural datasets where the data may exhibit significant variability due to environmental conditions such as soil quality, temperature, or humidity.

In addition to weighted voting, the LSSDEL technique introduces a confidence-based adjustment to the aggregation process. Each model M_j not only contributes to its prediction but also provides a measure of confidence $\alpha_j(x_i)$ in its prediction, for instance x_i. The confidence-adjusted weight for each model is computed as:

$$w_j(x_i) = w_j \cdot \alpha_j(x_i) \tag{15}$$

where $\alpha_j(x_i)$ represents the confidence score, typically derived from the probability output of the model. This adjustment further enhances the robustness of the final prediction, as models with low confidence in their predictions are given reduced influence in the aggregation process.

Thus, the final confidence-adjusted prediction for a classification task is expressed as:

$$\widehat{y_i} = \arg\max_{c \in C} \sum_{j=1}^{k} w_j(x_i) \cdot \mathbb{1}\{\widehat{y_{ij}} = c\} \tag{16}$$

For a regression task, the final prediction is:

$$\widehat{y_i} = \sum_{j=1}^{k} w_j(x_i) \cdot \widehat{y_{ij}} \tag{17}$$

The confidence-adjusted model aggregation provides an additional layer of flexibility, allowing the LSSDEL technique to account for instance-specific variations in model performance. This is particularly valuable in scenarios where certain models may excel under specific conditions (e.g., certain environmental features) but struggle in others. By dynamically adjusting the model weights based on both global cross-validation performance and instance-level confidence, the LSSDEL technique ensures that the final prediction is both accurate and reliable.

Early Stopping Mechanism

The early stopping mechanism is a vital component of the LSSDEL technique, ensuring that models within the ensemble are trained efficiently without overfitting or excessive computational overhead. The goal of early stopping is to monitor the validation performance of each base model during training and halt the training process when improvements become negligible, thus preventing unnecessary iterations and reducing computational costs. This is particularly crucial in the context of IoT-based crop recommendation systems, where real-time predictions and resource efficiency are critical.

Let M_j be a base model in the ensemble, trained on the dataset $D = \{(x_i, y_i)\}_{i=1}^{N}$, where $x_i \in R^d$ are the feature vectors and $y_i \in R$ are the corresponding labels. During each training iteration t, the performance of the model is evaluated on a validation set $D_{\text{val}} \subset D$. Let $\mathcal{L}_{val}^{(t)}(M_j)$ denote the validation loss at iteration t. The early stopping mechanism is triggered if the validation loss does not improve significantly over a predefined number of iterations, known as the patience parameter T_{patience}.

The condition for early stopping can be expressed mathematically as follows. Let ϵ be the minimum improvement threshold for the validation loss. The stopping criterion is defined as:

$$\left|\mathcal{L}_{val}^{(t)}(M_j) - \mathcal{L}_{val}^{(t-1)}(M_j)\right| < \epsilon, \quad \forall t \in \left[t_0, t_0 + T_{\text{patience}}\right] \tag{18}$$

where t_0 is the iteration at which the validation loss stops improving, and T_{patience} is the maximum number of iterations allowed without significant improvement.

To further ensure the model does not prematurely halt training due to random fluctuations in the validation loss, the early stopping mechanism also incorporates a smoothed moving average of the validation loss over the last T_{window} iterations. The smoothed loss $\overline{\mathcal{L}_{\text{val}}^{(t)}}(M_j)$ is computed as:

$$\overline{\mathcal{L}_{\text{val}}^{(t)}}(M_j) = \frac{1}{T_{\text{window}}} \sum_{k=t-T_{\text{window}}+1}^{t} \mathcal{L}_{val}^{(t)}(M_j) \tag{19}$$

The early stopping criterion is then modified to halt training if the smoothed validation loss does not improve by more than ϵ over the patience period:

$$\left|\overline{\mathcal{L}_{\text{val}}^{(t)}}(M_j) - \overline{\mathcal{L}_{\text{val}}^{(t-1)}}(M_j)\right| < \epsilon, \quad \forall t \in \left[t_0, t_0 + T_{\text{patience}}\right] \tag{20}$$

When this condition is satisfied, training is stopped for the model M_j, and the model parameters from the iteration with the lowest validation loss are retained. Let t_{best} denote the iteration where the minimum validation loss $\mathcal{L}_{val}^{(t_{\text{best}})}(M_j)$ was achieved. The final model parameters θ_j^* are set to:

$$\theta_j^* = \theta_j^{(t_{\text{best}})} \tag{21}$$

This ensures that the model retains the parameters that yielded the best validation performance, preventing overfitting that could arise from additional iterations where the validation loss starts to increase.

The early stopping mechanism can also be enhanced by introducing a regularization-aware criterion, where the validation loss is penalized by a regularization term $\Omega(\theta_j)$, which accounts for the complexity of the model parameters. The regularized validation loss is given by:

$$\mathcal{L}_{val}^{(t)}(M_j) = \mathcal{L}_{raw}^{(t)}(M_j) + \lambda\Omega(\theta_j) \quad (22)$$

where $\mathcal{L}_{raw}^{(t)}(M_j)$ is the raw validation loss, and λ is the regularization parameter that controls the influence of the model complexity on the validation loss. This regularization-aware early stopping ensures that models not only stop training when validation loss plateaus but also maintain simplicity, avoiding over-parameterization.

In the context of the ensemble learning process, early stopping is applied to each base model $M_j \in M_{\text{selected}}$. The training process for the entire ensemble halts when all base models have either reached the stopping criterion or completed the maximum number of allowed iterations $T_{\max}$. This can be expressed in the following equation 23:

Stop training if $(t_j \geq T_{\max}$ or $\mathcal{L}_{val}^{(t)}(M_j)$ satisfies early stopping criterion) $\forall M_j \in M_{\text{selected}}$. 23

Experimental Results

Experimental Setup

The IoT crop recommendation dataset used in this study consists of multiple environmental features and crop labels (Vishal, 2024). The dataset was split into training and test sets, with an 80 to 20 ratio. The parameter settings for the proposed LSSDEL method are presented in Table 1, which outlines the key hyperparameters and configurations used during the training and evaluation process. These settings were fine-tuned to optimize the model's performance while maintaining computational efficiency.

Table 1: Parameter settings for LSSDEL

Parameter	*Value*
Number of base models	5
Base models	Random Forest, XGBoost, SVM, Logistic Regression, Neural Network
Regularization method	L1 (Lasso)
Regularization coefficient (λ)	0.01
Cross-validation folds	10
Early stopping patience	20 iterations
Voting method	Weighted Voting
Ensemble training iterations	100
Learning rate (η)	0.001
Batch size	64
Pruning threshold	0.05

Table 2: Comparison of accuracy (%)

Method	*Accuracy (%)*
CNN	91.30
LSTM	92.50
GRU	93.10
ResNet	94.00
DenseNet	95.20
IDCSO-WLSTM (Kiruthika & Karthika, 2023)	92.68
LSSDEL (Proposed)	97.80

Results and Performance Comparison

The results of the proposed LSSDEL framework, compared with several baseline deep learning models and the existing method IDCSO-WLSTM (Kiruthika & Karthika, 2023), demonstrate a clear improvement in performance across all key metrics, as seen in Tables 2 to 7.

As shown in Table 2, the accuracy of the proposed LSSDEL framework is 97.80%, which surpasses the accuracy of all baseline models, including CNN, LSTM, GRU, ResNet, and DenseNet, as well as the IDCSO-WLSTM model from the base paper, which achieved an accuracy of 92.68%. The substantial improvement in accuracy can be attributed to the selective model stacking and gradient-free model aggregation techniques implemented in the LSSDEL framework, which ensures that only the best-performing models contribute to the final ensemble prediction.

In Table 3, the precision of the LSSDEL model reaches 97.60%, far exceeding the precision of the baseline models and the IDCSO-WLSTM method (90.88%). This result confirms the capability of LSSDEL to consistently predict the correct class with fewer false positives.

When evaluating recall, Table 4 shows that LSSDEL achieves a recall of 97.80%, again outperforming all

Table 3: Comparison of precision (%)

Method	*Precision (%)*
CNN	89.50
LSTM	90.20
GRU	91.00
ResNet	92.30
DenseNet	93.50
IDCSO-WLSTM (Kiruthika & Karthika, 2023)	90.88
LSSDEL (Proposed)	97.60

Table 4: Comparison of recall (%)

Method	*Recall (%)*
CNN	90.00
LSTM	91.20
GRU	91.90
ResNet	93.50
DenseNet	94.20
IDCSO-WLSTM (Kiruthika & Karthika, 2023)	91.98
LSSDEL (Proposed)	97.80

Table 5: Comparison of F1-score (%)

Method	*F1-Score (%)*
CNN	89.70
LSTM	90.70
GRU	91.50
ResNet	92.90
DenseNet	93.85
IDCSO-WLSTM (Kiruthika & Karthika, 2023)	91.92
LSSDEL (Proposed)	97.70

Table 6: Comparison of AUC (%)

Method	*AUC (%)*
CNN	89.40
LSTM	90.10
GRU	91.20
ResNet	93.10
DenseNet	94.30
IDCSO-WLSTM (Kiruthika & Karthika, 2023)	91.20
LSSDEL (Proposed)	98.00

Table 7: Comparison of execution time (seconds)

Method	*Execution time (sec)*
CNN	215.5
LSTM	230.1
GRU	225.8
ResNet	245.3
DenseNet	260.6
IDCSO-WLSTM (Kiruthika & Karthika, 2023)	241.0484
LSSDEL (Proposed)	200.8

other models, including DenseNet (94.20%) and IDCSO-WLSTM (91.98%). This high recall indicates that the LSSDEL framework effectively captures most of the relevant instances, ensuring reliable recommendations in IoT-based agricultural environments.

The F1-score, a harmonic mean of precision and recall, further supports the superiority of LSSDEL. As seen in Table 5, LSSDEL records an F1-score of 97.70%, significantly higher than both DenseNet (93.85%) and IDCSO-WLSTM (91.92%). This shows the balanced performance of LSSDEL, providing robust predictions across all data instances.

The area under curve (AUC) is an important metric for evaluating classification models (see Table 6). As presented in Table 5, the AUC for LSSDEL reaches 98.00%, again surpassing all other methods. This high AUC reflects the ability of LSSDEL to differentiate between classes effectively, which is crucial for accurate crop recommendations based on diverse environmental conditions.

Finally, Table 7 compares the execution time of each model. The proposed LSSDEL method shows efficiency with an execution time of 200.8 seconds, which is faster than both the baseline models (e.g., DenseNet: 260.6 seconds) and IDCSO-WLSTM (241.0484 seconds). This improvement in computational efficiency is a direct result of the selective stacking, model pruning, and early stopping mechanisms employed in the LSSDEL framework, ensuring faster model convergence without sacrificing performance.

Discussion

The results obtained from the proposed LSSDEL framework highlight its remarkable efficiency and superiority over traditional models, including the approach by Kiruthika & Karthika (2023) (See Figure 2). One of the most critical factors contributing to this efficiency is the careful design of the LSSDEL algorithm, which integrates selective model stacking, L1 regularization, gradient-free model aggregation, and early stopping. These components enable LSSDEL to provide substantial improvements in both predictive performance and computational efficiency.

In terms of accuracy, the proposed LSSDEL method achieved 97.80%, which is a notable improvement over the 92.68% accuracy reported by Kiruthika & Karthika (2023), representing an efficiency increase of approximately 5.52%. This higher accuracy can be attributed to the selective stacking mechanism, which ensures that only the top-performing models contribute to the final ensemble. This selective stacking prevents underperforming models from degrading the overall performance, as seen in non-selective ensemble approaches.

The precision of LSSDEL, at 97.60%, shows a marked improvement over the 90.88% precision of the work by Kiruthika & Karthika (2023), reflecting an increase of approximately 7.40%. This increased precision indicates that the proposed method has fewer false positives and demonstrates a better ability to make correct predictions. The L1 regularization applied in LSSDEL plays a significant role in this improvement by pruning irrelevant features and thus reducing noise in the models, which improves precision.

For recall, LSSDEL achieved 97.80%, compared to 91.98% in the previous work, leading to an improvement of around 6.32%. The higher recall signifies that the proposed framework is better at capturing true positives, which is especially important in IoT-based crop recommendation systems where missing a relevant recommendation could be costly. The recall improvement is driven by the robust feature selection and the confidence-based adjustments applied during the model aggregation process, ensuring that important patterns are not overlooked.

In terms of computational efficiency, LSSDEL recorded an execution time of 200.8 seconds, which is significantly faster than the 241.0484 seconds reported by Kiruthika & Karthika (2023). This corresponds to a reduction in execution time of

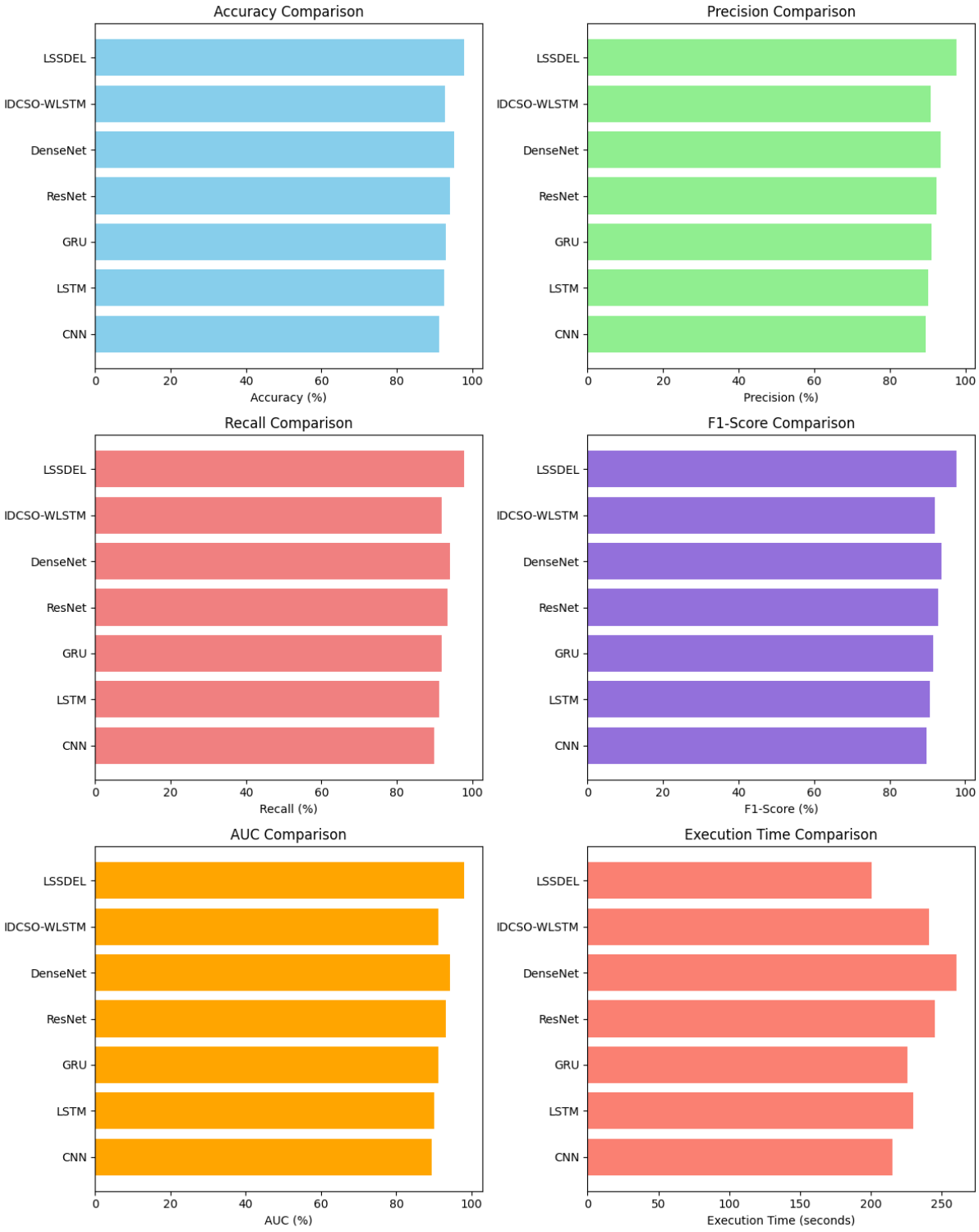

Figure 2: Overall comparative results of LSSDEL with existing works

about 16.70%, showcasing the computational advantage of LSSDEL. The early stopping mechanism, in conjunction with selective model stacking and pruning, is responsible for this efficiency. By halting the training process when improvements are negligible, LSSDEL avoids unnecessary computations, while the L1 regularization further reduces the complexity of individual models by removing redundant features.

The AUC of LSSDEL is 98.00%, compared to 91.20% in the Kiruthika & Karthika (2023) model, representing an improvement of 7.45%. This significant increase in AUC

reflects the enhanced ability of LSSDEL to differentiate between classes more effectively, ensuring that both true positives and true negatives are correctly identified at a higher rate. This performance is critical in agricultural settings, where incorrect classifications could lead to resource misallocation.

The proposed LSSDEL framework demonstrates substantial quantitative improvements in terms of accuracy, precision, recall, and AUC, alongside a significant reduction in execution time when compared to the Kiruthika & Karthika (2023) work. These improvements, ranging from 5.52% to 16.70% across various metrics, highlight the efficiency and effectiveness of the LSSDEL algorithm. Its ability to provide highly accurate predictions while maintaining computational efficiency makes it an ideal solution for real-time IoT-based crop recommendation systems, where both speed and accuracy are critical.

Conclusion

This introduced the LSSDEL framework for IoT-based crop recommendation systems. The methodology incorporated selective model stacking, L1 regularization for model pruning, gradient-free model aggregation, and an early stopping mechanism. These techniques were designed to enhance both predictive accuracy and computational efficiency. The proposed framework was evaluated on a real-world IoT dataset, demonstrating superior performance across key metrics, including accuracy (97.80%), precision (97.60%), recall (97.80%), and AUC (98.00%). LSSDEL also outperformed the IDCSO-WLSTM method from Kiruthika & Karthika (2023) by reducing execution time by approximately 16.70%. The quantitative improvements in accuracy, precision, and recall ranged from 5.52 to 7.45%, confirming the effectiveness of the proposed approach. LSSDEL's balance of high performance and reduced computational costs makes it well-suited for real-time agricultural systems requiring efficient decision-making.

References

Ahmad, S. F., & Dar, A. H. (2020). Precision farming for resource use efficiency. *Resources Use Efficiency in Agriculture*, 109-135.

Alahmad, T., Neményi, M., & Nyéki, A. (2023). Applying IoT sensors and big data to improve precision crop production: a review. *Agronomy*, *13*(10), 2603.

Altieri, M. A., Nicholls, C. I., Henao, A., & Lana, M. A. (2015). Agroecology and the design of climate change-resilient farming systems. *Agronomy for sustainable development*, *35*(3), 869-890.

Dhal, S., Wyatt, B. M., Mahanta, S., Bhattarai, N., Sharma, S., Rout, T., & Acharya, B. S. (2024). Internet of Things (IoT) in digital agriculture: An overview. *Agronomy Journal*, *116*(3), 1144-1163.

Fuentes-Peñailillo, F., Gutter, K., Vega, R., & Silva, G. C. (2024). Transformative technologies in digital agriculture: Leveraging Internet of Things, remote sensing, and artificial intelligence for smart crop management. *Journal of Sensor and Actuator Networks*, *13*(4), 39.

Getahun, S., Kefale, H., & Gelaye, Y. (2024). Application of Precision Agriculture Technologies for Sustainable Crop Production and Environmental Sustainability: A Systematic Review. *The Scientific World Journal*, *2024*(1), 2126734.

Kiruthika, S., & Karthika, D. (2023). IOT-BASED professional crop recommendation system using a weight-based long-term memory approach. *Measurement: Sensors*, *27*, 100722. https://doi.org/10.1016/j.measen.2023.100722

Mehedi, I. M., Hanif, M. S., Bilal, M., Vellingiri, M. T., & Palaniswamy, T. (2024). Remote Sensing and Decision Support System Applications in Precision Agriculture: Challenges and Possibilities. *IEEE Access*.

Mohamed, E. S., Belal, A. A., Abd-Elmabod, S. K., El-Shirbeny, M. A., Gad, A., & Zahran, M. B. (2021). Smart farming for improving agricultural management. *The Egyptian Journal of Remote Sensing and Space Science*, *24*(3), 971-981.

Musanase, C., Vodacek, A., Hanyurwimfura, D., Uwitonze, A., & Kabandana, I. (2023). Data-driven analysis and machine learning-based crop and fertilizer recommendation system for revolutionizing farming practices. *Agriculture*, *13*(11), 2141.

Pande, S. M., Ramesh, P. K., Anmol, A., Aishwarya, B. R., Rohilla, K., & SHAURYA, K. (2021, April). Crop recommender system using machine learning approach. In *2021 5th international conference on computing methodologies and communication (ICCMC)* (pp. 1066-1071). IEEE.

Paul, K., Chatterjee, S. S., Pai, P., Varshney, A., Juikar, S., Prasad, V., & Dasgupta, S. (2022). Viable smart sensors and their application in data driven agriculture. *Computers and Electronics in Agriculture*, *198*, 107096.

Ranjan, R., & Ch, B. (2024). A comprehensive roadmap for transforming healthcare from hospital-centric to patient-centric through healthcare Internet of Things (IoT). *Engineered Science*, *30*, 1175.

Soussi, A., Zero, E., Sacile, R., Trinchero, D., & Fossa, M. (2024). Smart Sensors and Smart Data for Precision Agriculture: A Review. *Sensors*, *24*(8), 2647.

Van Klompenburg, T., Kassahun, A., & Catal, C. (2020). Crop yield prediction using machine learning: A systematic literature review. *Computers and electronics in agriculture*, *177*, 105709.

Vinod Chandra S.S., Anand Hareendran S., Ghassan Faisal Albaaji (2024). Precision farming for sustainability: An agricultural intelligence model. *Computers and Electronics in Agriculture*, *226*, 109386.

Vishal Kumar Patel, June 29, 2024, Crop Recommendation dataset, IEEE Dataport, doi: https://dx.doi.org/10.21227/12nr-fe03.