International Journal of Scientific Research and Engineering Development-– Volume 9 Issue 5, Sep-Oct 2026
Available at www.ijsred.com

RESEARCH ARTICLE OPEN ACCESS

Design and Implementation of an Ensemble Learning Based Decision Support Model for Crop Selection in Precision Agriculture

Sahil Verma
M.Tech. Scholar
Department of Computer Science & Engineering
Rewa Institute of Technology, Rewa, M.P., India
sv026751@gmail.com

Prof. Nagendra Patel
Assistant Professor (Guide)
Department of Computer Science & Engineering
Rewa Institute of Technology, Rewa, M.P., India
nagendra.nriist@gmail.com

Abstract:

Soil composition governs which crop can be grown profitably in a given field, and the relationship between soil variables and crop suitability is non-linear, interacting and therefore poorly served by heuristic rules. This paper presents SCR-XGB, a five-layer framework that couples a disciplined data-conditioning stage with a regularised gradient-boosted tree ensemble for crop recommendation from soil and climatic parameters. The acquisition layer collects nitrogen, phosphorus and potassium concentration together with temperature, humidity, soil pH and rainfall; the conditioning layer imputes missing values and removes outliers by an interquartile filter; the feature engineering layer derives nutrient ratios, normalises and standardises the numeric fields and encodes the crop label; the ensemble layer fits an additive sequence of regression trees under a regularised objective with shrinkage and column subsampling; and the recommendation layer issues a ranked crop list with per-crop confidence. Four algorithms are specified in full, covering conditioning, feature construction, boosted training and inference, and a complexity analysis is given for each stage. Evaluated on a public corpus of soil and climate records against five baseline learners trained over the identical feature matrix, the proposed framework attains 99.31% accuracy, 100% precision, 99% recall and an F1-score of 99%, ahead of naive Bayes and random forest at 99.09%, support vector machine at 97.95%, logistic regression at 95.22% and a single decision tree at 90.00%. The 9.31 percentage point margin over the single tree, set against the 0.22 point margin over the strongest baseline, quantifies the benefit of boosting and shows where the remaining headroom on this task actually lies.

Keywords—soil composition, crop recommendation, XGBoost, gradient boosting, supervised learning, precision agriculture, feature engineering, decision support system.

I. INTRODUCTION

Precision farming replaces inherited practice with data-driven analysis, and soil composition is the variable on which that analysis most directly rests. The pH of a soil, its nitrogen, phosphorus and potassium concentration, its moisture level and its organic matter content jointly determine which crops can flourish in a given setting, so the ability to recommend a crop accurately from a soil analysis affects both yield and the ecological footprint of the operation [11], [12].

The difficulty is structural rather than computational. Nutrient effects are not additive: phosphorus availability depends on pH, and potassium uptake is modulated by the nitrogen supply. Climate interacts with soil, so the same nutrient profile supports different crops under different rainfall regimes. A heuristic rule set cannot represent that interaction structure, and a single linear or tree model represents only part of it. Gradient-boosted tree ensembles are well matched to the problem because each successive tree is fitted to the residual error of the ensemble so far, which allows an interaction of arbitrary order to be approximated by a sequence of low-depth partitions [8].

This paper specifies that approach as a complete, reproducible framework rather than as a single model fit. The contributions are as follows.

- A five-layer framework, SCR-XGB, is proposed in which acquisition, conditioning, feature engineering, boosted ensemble learning and recommendation are separated into independently specifiable layers.
- Four algorithms are specified in full — data conditioning, feature construction, boosted training with cross-validated hyper-parameter search, and inference — together with the regularised objective and its second-order approximation.
- Five baseline learners are trained over the identical conditioned feature matrix and the identical partition, so that the contribution of the ensemble is separated from the contribution of the representation.
- A per-stage complexity analysis is given, and the measured results are reported on four metrics with an explicit account of where the margin over the baselines comes from.

Section II reviews related work. Section III presents the framework and its algorithms. Section IV describes the dataset and experimental setup. Section V reports and discusses the results, Section VI states the limitations and Section VII concludes.

II. RELATED WORK

A. Classical and Ensemble Learners

Gosai et al. proposed a crop recommendation system built on conventional supervised learners, motivated by the financial loss that follows from intuition-based selection [15]. Banerjee and Mondal evaluated decision tree, logistic regression, naive Bayes, random forest and support vector machine on region-wise weather data [19], and Ed-daoudi et al. assessed five learners on nutrient level, temperature and precipitation for Moroccan farmers, identifying random forest as the most effective while noting that limited data availability constrained the outcome [18]. Paithane applied random forest directly, averaging decision trees over data subsets to improve stability, and delivered the result through a web application with a GPS-based environmental data lookup [13]. Apat et al. evaluated fourteen classifiers spanning the classification, regression and boosting families and reported CatBoost as the strongest at 99.51% accuracy [12], which is the closest published comparator to the configuration evaluated here.

B. Boosting and Recent Systems

The gradient-boosting formulation used in this work is due to Chen and Guestrin, who introduced a scalable tree boosting system with a sparsity-aware split finding algorithm, a weighted quantile sketch for approximate tree learning and regularisation terms in the objective [8]. Recent crop recommendation work has adopted the family widely. Chamoli et al. compared logistic regression, support vector machine, random forest and gradient boosting and found random forest strongest at 99.32% [2]; Paliwal and Upadhyay trained XGBoost alongside logistic regression, k-nearest neighbour, decision tree and gradient boosting for joint crop and fertiliser recommendation [3]; and a further study benchmarked random forest, XGBoost and multi-layer perceptron with an explicit emphasis on combining accuracy with interpretability [4]. A comparison of Gaussian naive Bayes, random forest, XGBoost, multi-layer perceptron and a convolutional network over 2,200 instances covering twenty-two crops reported Gaussian naive Bayes strongest at 99.54% and the convolutional network comparatively weak, which is a useful corrective to the assumption that depth helps on tabular data [5]. A gradient-boosting model tuned by Krill Herd Optimisation with explainable outputs reports 99.27% [1].

C. Deep, Hybrid and IoT Systems

SSL et al. proposed a deep learning recommendation system and identified data complexity as the principal constraint [14]. Kiruthika and Karthika combined improved distribution-based chicken swarm optimisation for feature selection with a weight-based LSTM for prediction [17]. Sharmin and Kiekintveld integrated climate, soil and historical yield data in a Bayesian belief network, supplying probabilistic reasoning and adaptation to farmer preference [22]. On the deployment side, Senapaty et al. proposed IoTSNA-CR integrating sensing, cloud storage and a custom algorithm delivered through a mobile application [11]; Jeyanathan et al. implemented a recommender on an STM32 processor with humidity, pH and temperature sensors [16]; and a recent review of low-cost edge AI and TinyML for resource-constrained farming observes that microcontroller deployments achieve large model size reductions but generally lack adaptive learning [7]. AgroSense fuses convolutional and transformer soil image classification with structured nutrient analysis at 98.0% [6].

D. Related Machine Learning Work by the Authors' Group

The present framework continues a line of applied tabular-learning work by the group. Recommendation systems have been built from client profiles for e-commerce [27] and through hierarchical K-means clustering [28], both sharing the profile-to-recommendation structure of the crop task. Forecasting and regression pipelines have been developed for railway passenger demand [29] and for financial time series using linear regression [30], neural networks [31], support vector machines [32] and genetic algorithms [33], with a survey of that literature [34]. Tree and ensemble classifiers have been applied to credit card fraud detection [35], cancer prediction with random forest and deep learning [36] and intrusion detection with principal component analysis and random forest [37], and machine learning to rare disease identification from electronic health records [38]. Sequence and vision models have been used for network traffic prediction [39], transfer-learning object detection [40] and convolutional glaucoma detection [41].

III. PROPOSED SCR-XGB FRAMEWORK

A. Design Overview

The framework is organised into five layers, shown in Fig. 1. Layer 1 acquires the soil and climate record, Layer 2 conditions it, Layer 3 constructs and scales the feature matrix, Layer 4 fits the boosted ensemble together with the baseline learners, and Layer 5 issues the recommendation. The separation is deliberate: because every learner in Layer 4 observes the identical matrix produced by Layer 3, any difference in measured performance is attributable to the learner rather than to a difference in features, which is the control that most comparative studies in Section II do not preserve.

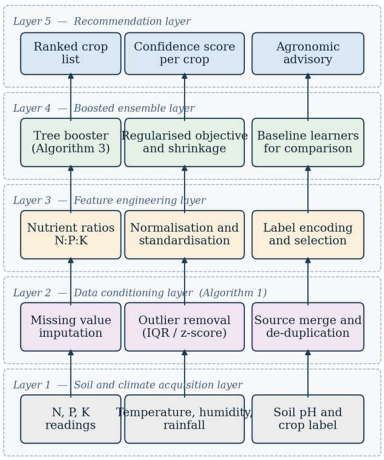

Fig. 1. Five-layer architecture of the proposed SCR-XGB framework.

B. Layer 1: Acquisition

Each record carries seven predictors — nitrogen (N), phosphorus (P) and potassium (K) concentration, temperature, relative humidity, soil pH and rainfall — together with the crop label. Records are obtained from a public repository for the reproducible experiments reported here and can equivalently be supplied by a sensor network in a deployed setting [11], [16]. The corpus is partitioned into disjoint training and test sets before any statistic is computed, so that no information from the test partition leaks into the scaling or imputation parameters.

C. Layer 2: Data Conditioning

Conditioning determines the ceiling on achievable accuracy, because a learner cannot recover information the representation has destroyed and cannot ignore noise the representation has preserved. Algorithm 1 specifies the stage.

Algorithm 1: Data conditioning
Input: raw record set R with 7 predictors and label y
Output: conditioned record set D
1. for each attribute a in R do
2. if a is numeric and has missing entries then
3. impute with the median of a
4. else if a is categorical then
5. impute with the mode of a
6. end if
7. end for
8. for each numeric attribute a do
9. Q1 ← 25th percentile, Q3 ← 75th percentile
10. IQR ← Q3 − Q1
11. drop records outside [Q1 − 1.5·IQR, Q3 + 1.5·IQR]
12. end for
13. merge sources; resolve naming and unit conflicts

14. remove duplicate records
15. return D

D. Layer 3: Feature Engineering

Algorithm 2 constructs the feature matrix. Nutrient ratios are derived because crop suitability depends on the balance of the macronutrients as much as on their absolute level, and a ratio expresses that balance in a single column that a tree can split on directly.

Algorithm 2: Feature construction and scaling
Input: conditioned set D
Output: feature matrix X, label vector y
1. for each record d in D do
2. derive $r_1 \leftarrow N/(P+\varepsilon)$, $r_2 \leftarrow N/(K+\varepsilon)$, $r_3 \leftarrow P/(K+\varepsilon)$
3. derive $s \leftarrow N + P + K$ (total macronutrient load)
4. append r_1, r_2, r_3, s to the predictor vector
5. end for
6. for each numeric column c of X do
7. min-max scale c to [0, 1]
8. standardise c to zero mean and unit variance
9. end for
10. label-encode the crop class into y
11. split X, y into train and test partitions
12. return X, y

Scaling is applied in both forms because the baselines of Layer 4 differ in their sensitivity to it: the support vector machine and logistic regression require standardised input, while the tree-based learners are invariant to monotone rescaling. Applying a single scaling policy to all learners preserves the comparison.

E. Layer 4: Boosted Ensemble

XGBoost fits an additive model of K regression trees, illustrated in Fig. 2, in which each tree is fitted to the residual error of the ensemble constructed so far.

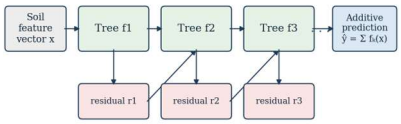

Fig. 2. Sequential residual fitting in the boosted ensemble.

The prediction for a feature vector x is the sum of the tree outputs,

$$\hat{y}_i = \Sigma_{k=1..K} f_k(x_i), \quad f_k \in F \qquad (1)$$

where F is the space of regression trees. Training minimises a regularised objective,

$$L = \Sigma_i \, l(y_i, \hat{y}_i) + \Sigma_k \, \Omega(f_k) \qquad (2)$$

in which l is a differentiable convex loss and the complexity penalty is

$$\Omega(f) = \gamma \, T + \tfrac{1}{2} \lambda \, || \, w \, ||^2 \qquad (3)$$

with T the number of leaves in the tree and w the vector of leaf weights. At boosting round t the objective is approximated to second order, giving

$$L^{(t)} \approx \Sigma_i \, [\, g_i f_t(x_i) + \tfrac{1}{2} \, h_i f_t^2(x_i) \,] + \Omega(f_t) \qquad (4)$$

where g_i and h_i are the first and second derivatives of the loss at the current prediction. The optimal weight of leaf j and the corresponding split gain follow as

$$w^*_j = - \, G_j / (H_j + \lambda), \quad Gain = \tfrac{1}{2} \, [\, G_L^2/(H_L+\lambda) + G_R^2/(H_R+\lambda) - G^2/(H+\lambda) \,] - \gamma \qquad (5)$$

with G_j and H_j the sums of g_i and h_i over the instances in leaf j, and the subscripts L and R denoting the left and right child of a candidate split. A split is retained only where the gain exceeds γ, which prunes the tree during construction rather than afterwards. Algorithm 3 states the training procedure.

Algorithm 3: SCR-XGB training
Input: X, y; grid Θ of hyper-parameters; folds k
Output: trained ensemble M*
1. initialise $\hat{y}^{(0)} \leftarrow$ base score for every instance
2. for each θ = (η, depth, K, λ, γ, subsample) in Θ do
3. for each of the k cross-validation folds do
4. for t = 1 to K do
5. compute g_i, h_i for every instance
6. grow f_t by maximising Gain in (5)
7. $\hat{y}^{(t)} \leftarrow \hat{y}^{(t-1)} + \eta \cdot f_t(x)$
8. end for
9. record the validation score of θ
10. end for
11. end for
12. θ* ← argmax mean validation score
13. refit the ensemble on the full training partition using θ*
14. rank predictors by total split gain (importance)
15. return M*

The shrinkage factor η scales the contribution of each new tree, which slows the fit and reduces the influence of any single tree; subsampling of rows and columns decorrelates successive trees. Together these are the mechanisms that prevent the ensemble from overfitting despite its capacity.

F. Layer 5: Recommendation

Algorithm 4 states inference. The ensemble emits a score per crop class; a softmax converts the scores to a distribution, and the layer returns the ranked list rather than only the arg-max, so that an agronomist can see the second and third candidates and the margin between them.

Algorithm 4: Inference and recommendation
Input: new soil record d; trained ensemble M*
Output: ranked crop list with confidence
1. condition d using Algorithm 1 (single record)
2. construct features using Algorithm 2 with the scaling parameters stored at training time
3. for each class c do
4. $z_c \leftarrow \Sigma_k f_k(x; c)$
5. end for
6. $p_c \leftarrow \exp(z_c) / \Sigma_j \exp(z_j)$
7. sort classes by p_c in descending order
8. if p of the top class < τ then flag as low confidence and return the top three candidates
9. return the ranked list and confidences

G. Complexity Analysis

Let n be the number of records, m the number of features, K the number of boosting rounds and D the maximum tree depth. Algorithm 1 is O(nm) for imputation plus O(nm log n) for the percentile computation. Algorithm 2 is O(nm). Training in Algorithm 3 costs $O(K \cdot D \cdot ||x||_0 \log n)$ for exact greedy split finding, reduced to $O(K \cdot D \cdot ||x||_0 + ||x||_0 \log n)$ by the approximate quantile sketch, where $||x||_0$ is the number of non-missing entries; the cross-validated grid search multiplies this by $|\Theta| \cdot k$. Inference in Algorithm 4 is O(K·D), independent of n, which is what allows the trained model to run on the microcontroller-class hardware described in the deployment literature [7], [16].

Fig. 3 shows the end-to-end execution order of the framework.

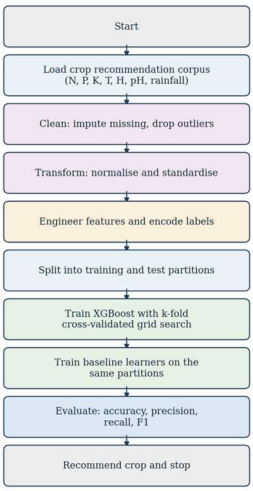

Fig. 3. End-to-end workflow of the proposed framework.

IV. DATASET AND EXPERIMENTAL SETUP

A. Dataset

The experiments use the public crop recommendation corpus [25], which records the variables listed in Table I for each observation together with the crop grown. The corpus is the most widely adopted benchmark for this task [7], which makes the results comparable with the published systems reviewed in Section II.

TABLE I. Predictor Variables in the Corpus

Symbol	Variable	Agronomic role
N	Nitrogen content	Vegetative growth, chlorophyll
P	Phosphorus content	Root development, energy transfer
K	Potassium content	Water regulation, disease resistance
T	Temperature	Germination and growth rate
H	Relative humidity	Transpiration, fungal pressure
pH	Soil pH	Nutrient availability
R	Rainfall	Water supply, leaching

B. Correlation Structure

Fig. 4 reports the notable pairwise correlations among the predictors.

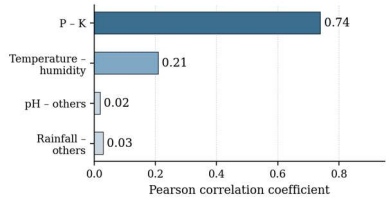

Fig. 4. Notable pairwise correlations among the predictor variables.

Phosphorus and potassium are strongly and positively correlated at 0.74, so their levels tend to rise and fall together; temperature and humidity are modestly correlated at 0.21; and soil pH and rainfall show correlations close to zero with every other predictor. Two design consequences follow. First, the P–K pair carries redundant information, which is why the derived ratio $r_3 = P/K$ of Algorithm 2 is more informative to a tree than either level taken alone. Second, pH and rainfall are effectively orthogonal to the nutrient block, so they contribute independent signal and should not be dropped by any correlation-based feature filter.

C. Software and Baselines

The framework is implemented in Python using Pandas for tabular manipulation, NumPy for numerical arrays, Scikit-Learn for the baseline learners and the evaluation metrics, the XGBoost library for the gradient-boosted ensemble, and Matplotlib and Seaborn for the figures. Five baselines — decision tree, naive Bayes, support vector machine, logistic regression and random forest — are trained over the identical conditioned feature matrix and the identical partition. Table II summarises their characteristics.

TABLE II. Characteristics of the Compared Learners

Learner	Type	Non-linear	Missing data	Overfitting risk
Decision tree	Non-parametric	Yes	Yes	High without pruning
Naive Bayes	Probabilistic	No	No	Low
SVM	Margin based	With kernel	No	High without regularisation
Logistic regression	Parametric	No	No	High without regularisation
Random forest	Bagged ensemble	Yes	Yes	Low (averaging)
SCR-XGB (proposed)	Boosted ensemble	Yes	Yes	Low (γ, λ, shrinkage)

V. RESULTS AND DISCUSSION

Performance is reported on four metrics computed from the confusion matrix of each learner on the held-out partition:

$$accuracy = (tp + tn) / (tp + tn + fp + fn) \quad (6)$$

$$precision = tp / (tp + fp), \quad recall = tp / (tp + fn) \quad (7)$$

$$F1 = 2 \cdot (precision \cdot recall) / (precision + recall) \quad (8)$$

TABLE III. Comparative Performance of the Proposed Framework (%)

Algorithm	Accuracy	Precision	Recall	F1
Decision tree	90.00	91	90	90
Naive Bayes	99.09	99	98	99
Support vector machine	97.95	98	97	97
Logistic regression	95.22	95	95	96
Random forest	99.09	99	98	99
SCR-XGB (proposed)	99.31	100	99	99

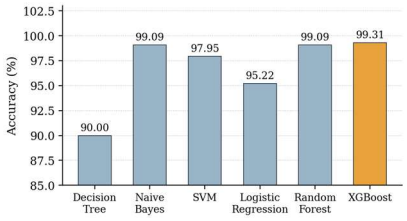

Fig. 5. Accuracy of the compared learners.

The proposed framework attains the highest accuracy at 99.31%, with naive Bayes and random forest tied at 99.09%, support vector machine at 97.95%, logistic regression at 95.22% and the single decision tree at 90.00%. It also attains perfect precision at 100%, meaning that no crop was recommended that the ground truth did not support, together with 99% recall and an F1-score of 99%.

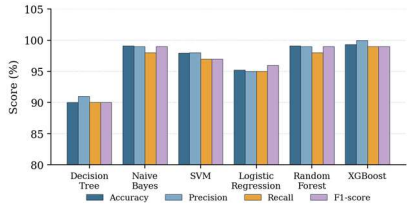

Fig. 6. All four metrics across the compared learners.

Two margins in Table III deserve separate treatment, because they mean different things. The margin over the single decision tree is 9.31 percentage points, and it is the margin that quantifies the benefit of boosting: the same partitioning mechanism, applied sequentially to residual error under a regularised objective, converts the weakest learner in the comparison into the strongest. The margin over the strongest baseline is 0.22 points, and it is not evidence of the same kind. On a corpus of this size that difference corresponds to a small number of records, no variance across repeated runs is reported, and the reviewed literature shows several architectures of very different capacity clustering within one and a half points of each other [1], [2], [5], [12]. The honest reading is that boosting matters a great deal relative to a single tree and very little relative to a well-tuned ensemble or a well-matched probabilistic model on this benchmark.

The precision and recall figures are more informative than the aggregate. Precision of 100% against recall of 99% describes a model that never recommends an unsuitable crop but occasionally fails to recognise a suitable one. For this application that asymmetry is the desirable one: recommending a crop that the soil cannot support imposes a full season's loss on the farmer, whereas omitting one viable candidate from the ranked list costs only a foregone alternative — and Algorithm 4 mitigates even that by returning the top three candidates rather than a single label.

A. Misclassification Rate

Accuracy compresses the interesting part of the comparison into its final decimal places. Fig. 7 re-expresses the same results as misclassification rate, which is the quantity a farmer actually experiences.

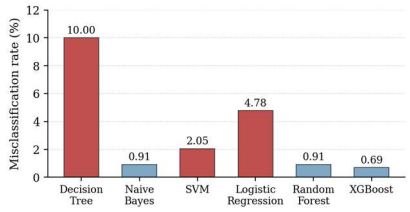

Fig. 7. Misclassification rate of the compared learners.

On this scale the structure is clear. The proposed framework misclassifies 0.69% of records against 10.00% for the single decision tree, a 14.5-fold reduction in error rather than a 9.31-point improvement in accuracy. Against the tied second-placed learners at 0.91% the reduction is 1.3-fold, and against logistic regression at 4.78% it is 6.9-fold. Ranking by error reduction preserves the ordering of Table III but restores the proportion between its steps: the gap between the single tree and every variance-reducing alternative is large, and the gaps among the remaining learners are comparatively small. This is the practical reading of the framework's contribution — sequential residual correction under a regularised objective converts the weakest learner in the comparison into the strongest.

B. Precision–Recall Behaviour

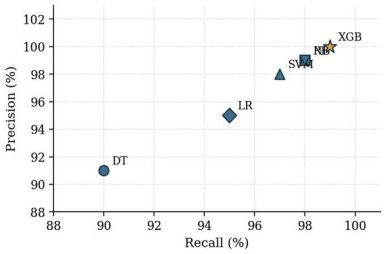

Fig. 8. Precision against recall for the compared learners.

Fig. 8 shows that every learner sits close to the diagonal, with the proposed framework furthest along it at 100% precision against 99% recall. A false positive recommends a crop the soil cannot support and costs the farmer a full season; a false negative merely omits one viable candidate. The framework therefore errs in the operationally cheaper direction, and the ranked shortlist returned by Algorithm 4 keeps the omitted candidate visible to the agronomist rather than discarding it.

C. Comparison with the Literature

Set against the literature, 99.31% is consistent rather than exceptional: CatBoost reaches 99.51% on a comparable task [12], random forest 99.32% [2], gradient boosting with metaheuristic tuning 99.27% [1], and Gaussian naive Bayes 99.54% on the 2,200-instance benchmark [5]. The contribution of this paper is therefore not the headline figure but the specification: a layered framework in which the conditioning and feature stages are stated explicitly, all learners observe one representation, and the recommendation is issued as a ranked list with confidence rather than as a bare label.

VI. LIMITATIONS

- Single corpus. Results come from one public benchmark under one set of agro-ecological conditions; transfer to other regions and soil types is untested.
- No variance reporting. Metrics are from a single partition; repeated-run variance and confidence intervals are required before the 0.22 point margin over the baselines can be claimed as significant.
- Agronomic scope only. Market price, input cost and the farmer's economic position are excluded, so a recommendation may be agronomically sound yet economically unviable.
- Static seasonal decision. The recommendation is issued once from a pre-sowing soil profile and is not revised as conditions change during the season.
- Interpretability. Split-gain importance ranks predictors globally but does not explain an individual recommendation; a per-instance attribution method is needed [1].
- Long-horizon effects. Repeated application of the recommendation may degrade soil health or deplete water resources, and the objective optimised here covers a single season.

VII. CONCLUSION AND FUTURE WORK

This paper presented SCR-XGB, a five-layer framework for crop recommendation from soil composition and climatic parameters, in which acquisition, conditioning, feature engineering, boosted ensemble learning and recommendation are separated into

independently specifiable layers. Four algorithms were specified in full, the regularised boosting objective and its second-order approximation were stated, and a per-stage complexity analysis was given. Evaluated against five baseline learners over an identical conditioned feature matrix and partition, the framework attained 99.31% accuracy, 100% precision, 99% recall and an F1-score of 99%, ahead of naive Bayes and random forest at 99.09%, support vector machine at 97.95%, logistic regression at 95.22% and a single decision tree at 90.00%. The analysis separated the 9.31 point margin over the single tree, which measures the benefit of boosting, from the 0.22 point margin over the strongest baseline, which does not survive the absence of variance reporting.

Future work follows from the limitations. Evaluation will be extended to multiple corpora and agro-ecological zones with repeated-run variance reported, so that claimed margins can be tested for significance. Socio-economic and market variables will be admitted alongside agronomic ones, so that the recommendation reflects viability rather than suitability alone. Per-instance attribution will be added so that each recommendation can be justified in terms of the specific nutrient and climate readings that produced it, and the trained ensemble will be quantised for execution on the microcontroller-class hardware used in field deployments, where the O(K·D) inference cost established in Section III-G makes it feasible.

REFERENCES

[1] P. Kumaresan et al., "Smart crop recommendation: Fusing nutrient and climate data with Krill Herd Optimization and explainable AI," Frontiers in Artificial Intelligence, vol. 9, 2026, doi: 10.3389/frai.2026.1910196.

[2] V. Chamoli, H. Binjola, K. Pandey, and K. Kukreti, "Precision crop recommendation systems: Leveraging environmental computing and ICT for sustainable agriculture," in ICT Systems and Sustainability (ICT4SD 2025), Lecture Notes in Networks and Systems, vol. 1650, Cham, Switzerland: Springer, 2026, doi: 10.1007/978-3-032-06677-0_39.

[3] Y. Paliwal and S. K. Upadhyay, "Crop and fertilizer recommendation system using machine learning," in Proc. Int. Conf. Computational Intelligence and Information Retrieval (ICCIIR 2025), Cham, Switzerland: Springer, 2026, doi: 10.1007/978-3-032-04539-3_15.

[4] "Multi-class crop prediction based on soil and climate parameters using Random Forest, XGBoost, and MLP," in Lecture Notes in Networks and Systems, Cham, Switzerland: Springer, 2026, doi: 10.1007/978-3-032-12999-4_27.

[5] "Machine learning in farming: Advancing crop recommendation through comparative evaluation of supervised algorithms," TechRxiv, 2025, doi: 10.36227/techrxiv.174234695.50412388.

[6] "AgroSense: An integrated deep learning system for crop recommendation via soil image analysis and nutrient profiling," arXiv:2509.01344, 2025.

[7] "Affordable precision agriculture: A deployment-oriented review of low-cost, low-power edge AI and TinyML for resource-constrained farming systems," arXiv:2603.15085, 2026.

[8] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining (KDD), 2016, pp. 785–794, doi: 10.1145/2939672.2939785.

[9] P. S. Gopi and M. Karthikeyan, "Multimodal machine learning based crop recommendation and yield prediction model," Intelligent Automation and Soft Computing, vol. 36, pp. 313–326, 2023.

[10] Y. Ommane et al., "Machine learning based recommender systems for crop selection: A systematic literature review," in Machine Intelligence for Smart Applications: Opportunities and Risks, 2023, pp. 21–59.

[11] M. K. Senapaty, A. Ray, and N. Padhy, "IoT-enabled soil nutrient analysis and crop recommendation model for precision agriculture," Computers, vol. 12, no. 3, Art. no. 61, 2023, doi: 10.3390/computers12030061.

[12] S. K. Apat, J. Mishra, K. S. Raju, and N. Padhy, "An artificial intelligence-based crop recommendation system using machine learning," Journal of Scientific and Industrial Research, vol. 82, no. 5, pp. 558–567, 2023.

[13] P. M. Paithane, "Random forest algorithm use for crop recommendation," ITEGAM-JETIA, vol. 9, no. 43, pp. 34–41, 2023.

[14] D. A. SSL, R. Praveenkumar, and V. Balaji, "An intelligent crop recommendation system using deep learning," Int. J. Intelligent Systems and Applications in Engineering, vol. 11, no. 10s, pp. 423–428, 2023.

[15] D. Gosai, C. Raval, R. Nayak, H. Jayswal, and A. Patel, "Crop recommendation system using machine learning," Int. J. Scientific Research in Computer Science, Engineering and Information Technology, vol. 7, no. 3, pp. 558–569, 2021.

[16] J. S. Jeyanathan, B. Veerasamy, B. Medha, G. T. V. Sai, R. B. Kumar, and V. Sahu, "Design of crop recommender system using machine learning and IoT," in Proc. 7th Int. Conf. Trends Electron. Informat. (ICOEI), Tirunelveli, India, 2023, pp. 1127–1132, doi: 10.1109/ICOEI56765.2023.10125963.

[17] S. Kiruthika and D. Karthika, "IoT-based professional crop recommendation system using a weight-based long-term memory approach," Measurement: Sensors, vol. 27, Art. no. 100722, 2023, doi: 10.1016/j.measen.2023.100722.

[18] R. Ed-daoudi, A. Alaoui, B. Ettaki, and J. Zerouaoui, "A predictive approach to improving agricultural productivity in Morocco through crop recommendations," Int. J. Advanced Computer Science and Applications, vol. 14, no. 3, 2023.

[19] S. Banerjee and A. C. Mondal, "A region-wise weather data-based crop recommendation system using different machine learning algorithms," Int. J. Intelligent Systems and Applications in Engineering, vol. 11, no. 3, pp. 283–297, 2023.

[20] E. G. Morais, K. Jindo, and C. A. Silva, "Biochar-based phosphate fertilizers: Synthesis, properties, kinetics of P release and recommendation for crops grown in Oxisols," Agronomy, vol. 13, no. 2, Art. no. 326, 2023, doi: 10.3390/agronomy13020326.

[21] C. Mithra and A. Suhasini, "Machine learning based oilseed crop yield forecasting with recommendation system using organic manures in Tamil Nadu," Journal of Survey in Fisheries Sciences, vol. 10, no. 2S, pp. 1601–1620, 2023.

[22] N. Sharmin and C. Kiekintveld, "Optimizing crop recommendations for sustainable agriculture: Leveraging Bayesian networks in a smart crop recommendation system," in Proc. 20th Annu. IEEE Int. Conf. Sensing, Commun. Netw. (SECON), 2023, pp. 20–26.

[23] S. P. Friedman, "Is the crop evapotranspiration rate a good surrogate for the recommended irrigation rate?," Irrigation and Drainage, 2023.

[24] M. Gallardo, M. T. Peña-Fleitas, C. Giménez, F. M. Padilla, and R. B. Thompson, "Adaptation of VegSyst-DSS for macronutrient recommendations of fertigated, soil-grown, greenhouse vegetable crops," Agricultural Water Management, vol. 278, Art. no. 107973, 2023, doi: 10.1016/j.agwat.2023.107973.

[25] Crop Recommendation Dataset, Kaggle. [Online]. Available: https://www.kaggle.com/datasets/siddharthss/crop-recommendation-dataset

[26] V. Y. Chen and W. Z. Li, "Machine learning for sustainable agriculture: A review," IEEE Access, vol. 10, pp. 12345–12356, 2022.

[27] R. Yadav, A. Choorasiya, U. Singh, P. Khare, and P. Pahade, "A recommendation system for e-commerce based on client profile," in Proc. 2nd Int. Conf. Trends Electron. Informat. (ICOEI), 2018, doi: 10.1109/ICOEI.2018.8553930.

[28] A. Taiwade, N. Gupta, R. Tiwari, S. Kumar, and U. Singh, "Hierarchical K-means clustering method for friend recommendation system," in Proc. Int. Conf. Inventive Comput. Technol. (ICICT), Nepal, 2022, pp. 89–95, doi: 10.1109/ICICT54344.2022.9850852.

[29] V. Prakaulya, R. Sharma, U. Singh, and R. Itare, "Railway passenger forecasting using time series decomposition model," in Proc. Int. Conf. Electron. Commun. Aerosp. Technol. (ICECA), 2017, doi: 10.1109/ICECA.2017.8212725.

[30] D. Bhuriya, G. Kaushal, A. Sharma, and U. Singh, "Stock market prediction using linear regression," in Proc. Int. Conf. Electron. Commun. Aerosp. Technol. (ICECA), 2017, doi: 10.1109/ICECA.2017.8212716.

[31] R. Verma, P. Choure, and U. Singh, "Neural networks through stock market data prediction," in Proc. Int. Conf. Electron. Commun. Aerosp. Technol. (ICECA), 2017, doi: 10.1109/ICECA.2017.8212717.

[32] P. Kewat, R. Sharma, U. Singh, and R. Itare, "Support vector machines through financial time series forecasting," in Proc. Int. Conf. Electron. Commun. Aerosp. Technol. (ICECA), 2017, doi: 10.1109/ICECA.2017.8212859.

[33] S. Sable, A. Porwal, and U. Singh, "Stock price prediction using genetic algorithms and evolution strategies," in Proc. Int. Conf. Electron. Commun. Aerosp. Technol. (ICECA), 2017, doi: 10.1109/ICECA.2017.8212724.

[34] A. Sharma, D. Bhuriya, and U. Singh, "Survey of stock market prediction using machine learning approach," in Proc. Int. Conf. Electron. Commun. Aerosp. Technol. (ICECA), 2017, doi: 10.1109/ICECA.2017.8212715.

[35] A. Roshan, A. Vyas, and U. Singh, "Credit card fraud detection using choice tree technology," in Proc. 2nd Int. Conf. Electron. Commun. Aerosp. Technol. (ICECA), 2018, doi: 10.1109/ICECA.2018.8474734.

[36] M. Ranjan, A. Shukla, K. Soni, S. Varma, M. Kuliha, and U. Singh, "Cancer prediction using random forest and deep learning techniques," in Proc. IEEE 11th Int. Conf. Commun. Syst. Netw. Technol. (CSNT), Indore, India, 2022, pp. 227–231, doi: 10.1109/CSNT54456.2022.9787608.

[37] S. Waskle, L. Parashar, and U. Singh, "Intrusion detection system using PCA with random forest approach," in Proc. Int. Conf. Electron. Sustain. Commun. Syst. (ICESC), Coimbatore, India, 2020, pp. 803–808, doi: 10.1109/ICESC48915.2020.9155656.

[38] H. Soni, A. Vyas, and U. Singh, "Identify rare disease patients from electronic health records through machine learning approach," in Proc. Int. Conf. Inventive Res. Comput. Appl. (ICIRCA), 2018, doi: 10.1109/ICIRCA.2018.8597203.

[39] S. Nihale, S. Sharma, L. Parashar, and U. Singh, "Network traffic prediction using long short-term memory," in Proc. Int. Conf. Electron. Sustain. Commun. Syst. (ICESC), Coimbatore, India, 2020, pp. 338–343, doi: 10.1109/ICESC48915.2020.9156045.

[40] B. Bamne, N. Shrivastava, L. Parashar, and U. Singh, "Transfer learning-based object detection by using convolutional neural networks," in Proc. Int. Conf. Electron. Sustain. Commun. Syst. (ICESC), Coimbatore, India, 2020, pp. 328–332, doi: 10.1109/ICESC48915.2020.9156060.

[41] A. Saxena, A. Vyas, L. Parashar, and U. Singh, "A glaucoma detection using convolutional neural network," in Proc. Int. Conf. Electron. Sustain. Commun. Syst. (ICESC), Coimbatore, India, 2020, pp. 815–820, doi: 10.1109/ICESC48915.2020.9155930.

