Indian Journal of Computer Science and Technology
https://www.doi.org/10.59256/indjcst.20250403043
Volume 4, Issue3 (September-December 2025), PP: 269-275.
www.indjcst.com
ISSN No: 2583-5300

Smart Crop Advisory Systems Using Artificial Intelligence and Machine Learning

Sneh Lata Singh[1], Amit Yadav[2], Shanu Ahmed[3], Abhay Maurya[4], Varun Rana[5], Samir Ahmad[6]

[1]*Assistant professor, Department of Computer Science and Engineering, Dr. A.P.J Abdul Kalam Institute of Technology, Tanakpur, Champawat, Uttarakhand, India.*

[2,3,4,5,6]*Department of Computer Science and Engineering, Dr.A.P.J Abdul Kalam Institute of Technology, Tanakpur Champawat, Uttarakhand, India.*

To Cite this Article: *Sneh Lata Singh[1], Amit Yadav[2], Shanu Ahmed[3], Abhay Maurya[4], Varun Rana[5], Samir Ahmad[6], "Smart Crop Advisory Systems Using Artificial Intelligence and Machine Learning", Indian Journal of Computer Science and Technology, Volume 04, Issue 03 (September-December 2025), PP: 269-275.*

Abstract: *The agricultural sector has many issues, including unpredictable weather, soil degradation, selecting the right crops, inefficient fertilizer usage, and losses due to plant diseases or weeds. Farmers typically rely on their experience and on unverified local recommendations, which causes them to have a much lower productivity and financial risk than they could otherwise have. Therefore, the agriculture industry is beginning to utilize Artificial Intelligence ("AI") and Machine Learning ("ML") to implement data driven and precision based decision making in agriculture. The authors conduct a review of Smart Crop Advisory Systems (SCAS) and the current literature on SCAS. Examples of SCAS include crop recommendations, fertilizers, plant disease detection, and weather advisories. Methods such as Random Forest and Convolutional Neural Networks (CNNs) are examined for their contribution to agricultural decision support. The review examines existing gaps in research and suggests that new, integrated and scalable SCAS must be developed to support sustainable agriculture.*

Key Words: *Smart Irrigation, IoT in Agriculture, Nutrient Deficiency Detection, Precision Farming, Image Processing, Sustainable Agriculture, Smart Farming, Agricultural Automation.*

I. INTRODUCTION

Agricultural production is integral to countries' economies & provides a healthy dietary intake for the majority of people, in particular for poorer or developing nations. Yet, there are many issues facing modern-day agriculture including: Climate Change (Weather variability), Land Degradation, Poor Crop Selection, Inefficient use of Fertilizers, Crop Losses due to Disease and Pests. When making critical agricultural management decisions, smallholder & marginal farmers rely heavily on traditional practices or rely on non-scientific informal sources for advice (entailing) that often result in low efficiency and reduced productivity [7]. Additionally, because the manual techniques used for assessing both the soil and crops rely upon human skills, they are not only time intensive and labour intensive, but they also leave room for human error.[3].

Due to the continued rise of Artificial Intelligence (AI) and Machine Learning (ML), Agriculture is evolving into Data-Driven Decision Making and Precision Agriculture [1]. By employing Intelligent Computational Models to process large amounts of agricultural data, i.e., soil nutrient levels, weather conditions, crop visual data, etc., intelligent competitive models are able to create accurate, instantaneously actionable insights. Extensive research has shown that Machine Learning Algorithms such as Random Forests are highly effective for crop and fertilization recommendations while artificial neural networks such as CNNs exhibit potential promise with respect to plant disease identification [2][3].

The Smart Crop Advisory System (SCAS) uses both artificial intelligence (AI) and Machine Learning (ML) technologies to develop a scalable software system that provides farmers with the tools necessary for comprehensive agricultural consulting support. The ultimate goal of the SCAS is to increase the breadth of producers' crop production, reduce farm input costs, and support sustainable agricultural practices through the provision of reliable, data-based decision-making assistance for farmers. [4][5].

II. LITERATURE REVIEW

Advancements in AI and ML are changing the way farmers do things; now with the assistance of data- and precision-based tools they make decisions based on data collected from multiple sources. In the quest for creating intelligent tools to improve agricultural production, many researchers have been exploring different computational techniques that can address pressing issues facing today's agricultural producers, i.e. crop selection, fertilizer optimisation, disease detection, and uncertainty regarding climate. In addition, intelligent tools are designed to increase efficiency, decrease waste, and improve sustainability of our food supply.

Crop recommendation systems are one area of study that focuses on using soil parameters and environmental attributes to recommend the best crops to plant to optimise production. Machine Learning (ML) models such as Decision Trees, Support Vector Machines (SVM), and Random Forests have proven effective in providing crop recommendations to producers using soil nutrients and weather conditions. Random Forests are used more than the others because they provide the best accuracy in predicting the best crop based on a given set of inputs (NPK, pH, temperature, humidity, rainfall, etc.), especially when datasets are non-linear in nature. This technological advancement allows producers to minimize their reliance upon guesswork-based plant selection processes by effectively predicting the crops that should be planted as well as minimizing risk of crop failure. [1][7].

There is considerable research on how to maximise the effectiveness of fertilisers while enhancing soil quality and production yield within the context of a fertiliser recommendation system. Farmers often use fertilisers in ways that are either imbalanced (too much/too little) or inappropriately applied. This results in nutrient loss and degradation of soil, as well as pollution of the environment. As a means to solve those problems, Mamatha and Nayak developed a hybrid machine learning based fertiliser recommendation system based upon the soil test data and crop needs [2]. Based upon their data, there is a high degree of accuracy when using a classification or regression based algorithms to determine the fertiliser type and amount, thus reducing the amount of fertiliser used and helping maintain sustainable nutrient management for farmers [2].

As well, one of the more recent areas of research in agriculture is on determining plant disease at an early stage. The success achieved through the use of deep learning methods on this topic is substantiated with extensive use of Convolutional Neural Networks (CNN) to analyse crop leaf photos for the purpose of diagnosing crop disease [3]. The CNN models auto-extract visual features of the leaf such as colour, texture, and shape that allow for the determination of whether or not the leaf is diseased or not. In addition, automated systems for disease diagnosis result in faster, more accurate, and less likely to contain human error compared to manual diagnosis and can provide large scale support for agriculture.[3].

In addition to predictive modeling, several recent articles identify the role of the system's architecture and how the systems are deployed as contributing to their actual implementation in practice. The majority of early agricultural advisory systems were based on monolithic architectures that did not allow for scaling or flexibility; Ingole & Karale have indicated that adopting a microservices-based architecture using Java Spring Boot would allow for a modular approach to system development, independent deployments of each system component, and efficient integration of artificial intelligence services (AI) into the system [4]. This architecture provides for the separate processing of different functions of a predictive agricultural advisory system, such as Crop Recommendation, Fertilizer Recommendation Advisories, Pest/Disease Detection, and Weather Services, thereby increasing the system's scalability and maintainability [4]. Accessing the front end of the system has also been getting some attention in the literature. For example, Murad compared Progressive Web Applications (PWAs) to Native Mobile Apps and stated that PWAs were a better fit for rural users because they could be developed for a lower cost, could function offline, and did not require users to have high-end devices to access them [5]. The findings in his study also suggest that a user interface that is lightweight and easily accessible to all users will lead to more widespread adoption of the systems among small and marginal farmers.

Despite these advancements in the area of agricultural advisory systems, there remain a number of gaps in the literature. Most existing products are single-function systems, providing crop recommendations, fertilizer recommendations, or pest/disease detection in isolation [6]. There are also concerns about the lack of generalizability of most models, since most have been evaluated using controlled data sets. [7]. Issues related to real-world deployment, scalability, offline accessibility, and farmer engagement remain insufficiently addressed. These limitations emphasize the need for an integrated, scalable, and user-centric Smart Crop Advisory System that combines accurate AI models with robust cloud-native architecture.

III. METHODOLOGY

SCAS is built as an integrated, intelligent, and scalable crop advisory system by combining Machine Learning (ML), Deep Learning (DL) and Cloud native web technologies. The SCAS organization uses a modular-based approach, allowing each functional component to operate independently but communicate using well-defined interfaces. This offers flexibility, scalability and ease of maintenance should it be deployed in the field someday. [4].

Module1: Data Collection and Preprocessing Module

The first step in implementing the methodology is to gather agricultural data from reputable open-access databases as well as from various government publications (such as: Crop, Fertilizer History, Historical Yield, Weather Data, Plant Disease Pictures, Soil Overall Quality) [6][7]. Typically, raw datasets have poorly formatted entries, missing entries, noise, and inconsistent formatting which may cause a drop in performance of the machine learning model when developing. Due to this reason, various Data Preprocessing Techniques such as cleaning, normalizing, Feature selecting, and Outlier Removal are executed to improve the quality of the datasets. Ultimately, the machine learning algorithms will use this properly preprocessed dataset to learn the underlying meaningful patterns from the dataset and eventually create accurate predictions[1].

Module 2: Crop Recommendation Module

The crop recommendation module aims to assist farmers in selecting the most suitable crop based on soil and climatic conditions. This module uses supervised machine learning algorithms trained on historical soil nutrient data and weather parameters such as temperature, humidity, and rainfall. Among various algorithms, Random Forest has been widely adopted due to its robustness, ability to handle non-linear relationships, and resistance to overfitting [1]. The trained model analyzes the input parameters provided by the user and predicts the most appropriate crop for the given conditions. This data-driven approach reduces reliance on guesswork and traditional experience-based farming practices, thereby minimizing the risk of crop failure [1][7].

Module 3: Fertilizer Recommendation Module

The fertilizer recommendation module focuses on optimizing fertilizer usage to improve soil health and crop productivity.

Traditional fertilizer practices often involve excessive or imbalanced application, leading to soil degradation and environmental pollution. To overcome this issue, the proposed system employs machine learning-based regression and classification models that analyze soil nutrient levels and crop requirements [2]. Based on the model predictions, the system recommends appropriate fertilizer types and dosages. This module helps farmers reduce input costs, prevent nutrient loss, and promote sustainable nutrient management practices [2].

Module 4: Plant Disease Detection Module

Early detection of crop diseases is crucial to prevent yield loss and reduce pesticide overuse. The disease detection module utilizes deep learning techniques, specifically Convolutional Neural Networks (CNNs), to analyze uploaded crop leaf images. CNNs automatically extract relevant visual features such as color, texture, and shape, enabling accurate disease classification [3]. Compared to manual inspection, CNN-based systems provide faster and more reliable results, reducing dependency on expert intervention. This module supports early diagnosis and timely treatment, thereby minimizing crop damage and improving productivity [3].

Module 5: Weather Monitoring and Alert Module

Weather conditions play a critical role in agricultural decision-making. The weather monitoring module integrates real-time weather data using external weather APIs to provide forecasts and alerts related to rainfall, temperature, and extreme weather events. The system analyzes weather patterns and generates timely notifications to help farmers take preventive measures against adverse climatic conditions. Automated alerts significantly reduce weather-related risks and support proactive farm management [5].

Module 6: System Architecture and Integration Module

The overall system is implemented using a microservices-based architecture developed with Java Spring Boot. Each functional module, such as crop recommendation, fertilizer advisory, disease detection, and weather monitoring, is deployed as an independent service that communicates through RESTful APIs [4]. This architectural design enhances scalability, fault tolerance, and efficient resource utilization. A MySQL database is used for centralized data Storage and retrieval, ensuring data consistency and availability across services. Such cloud-native architectures are well-suited for large-scale agricultural systems that require continuous updates and expansion [4].

Module 7: User Interface and Community Interaction Module

The frontend of the system is developed using HTML, CSS, JavaScript, and AJAX to provide a responsive and user-friendly interface. The platform includes modules for crop advisory, fertilizer recommendation, disease detection, weather updates, and a community blog. The community module enables farmers to share experiences, seek advice, and access expert knowledge, fostering collaborative learning and continuous feedback. Studies have shown that accessible and interactive web-based platforms significantly improve technology adoption among rural users [5].

Workflow

The workflow of the Smart Crop Advisory System (SCAS) is designed to provide an end-to-end, data-driven agricultural decision-support process by integrating user interaction, backend services, artificial intelligence models, and real-time data sources.

Step 1: User Access and Authentication

The workflow begins when the user accesses the Smart Crop Advisory System (SCAS) through a web browser. New users complete the registration process by providing basic details, while existing users log in using secure authentication mechanisms. Successful authentication ensures authorized access to system features and personalized advisory services.

Step 2: Dashboard and Module Selection

After login, the user is redirected to the home dashboard. The dashboard provides access to various modules, including crop recommendation, fertilizer advisory, disease detection, weather information, community blog, and multilingual chatbot. The user selects the required module based on their farming needs.

Step 3: User Data Input

In this step, the user provides relevant inputs depending on the selected module. For crop and fertilizer recommendation, the user enters soil parameters such as Nitrogen (N), Phosphorus (P), Potassium (K), pH value, and location details. For disease detection, the user uploads crop leaf images. Weather information is fetched automatically using location data or selected manually by the user.

Step 4: Request Transmission to Backend

All user inputs are transmitted securely to the Spring Boot backend through RESTful APIs. The backend performs input validation to check for missing, incorrect, or inconsistent values. This validation step ensures reliable processing and prevents erroneous predictions.

Step 5: Service Routing via Microservices

Once validated, the backend routes the request to the appropriate microservice using a microservices-based architecture. Each service operates independently, such as crop recommendation service, fertilizer advisory service, disease detection service, or weather service, ensuring modularity and scalability.

Step 6: Crop Recommendation Processing

For crop recommendation, soil and weather data are forwarded to the trained machine learning model. The model analyzes historical agricultural patterns and predicts the most suitable crops for the given soil and climatic conditions. The prediction result is sent back to the backend service.

Step 7: Fertilizer Recommendation Processing

In the fertilizer advisory workflow, soil nutrient values and selected crop details are processed using regression and classification-based machine learning models.
The system calculates the optimal fertilizer type and dosage required for healthy crop growth and soil sustainability.

Step 8: Disease Detection Processing

For disease detection, uploaded crop images are processed by a deep learning-based Convolutional Neural Network (CNN). The CNN extracts visual features such as texture, color, and shape to identify crop diseases or weeds. The detected disease information is returned to the backend.

Step 9: Weather Monitoring and Risk Analysis

Simultaneously, real-time weather data is fetched from the OpenWeather API. The system analyzes forecasts to identify potential risks such as heavy rainfall, drought, or extreme temperature conditions that may affect crop growth.

Step 10: Database Storage

All generated recommendations, predictions, weather data, and user interactions are stored in the MySQL database. This enables record maintenance, future analysis, and personalized advisory services.

Step 11: Result Display to User

The backend sends the processed results to the frontend interface. The user views crop recommendations, fertilizer suggestions, disease detection results, and weather information in a simple and user-friendly format.

Step 12: Notification and Community Interaction

If severe weather conditions or disease outbreaks are detected, automated WhatsApp alerts are triggered and sent to the user. Additionally, users can interact through the community blog to share experiences and access expert guidance, completing the system workflow.

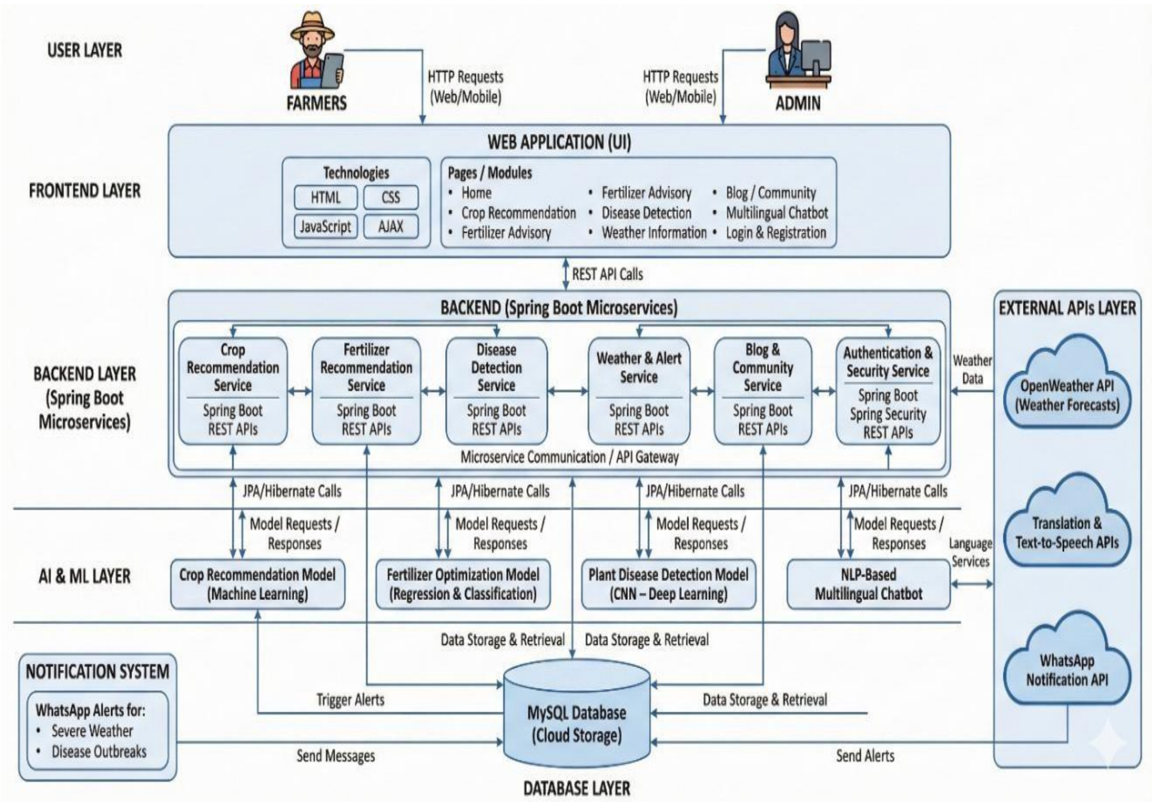

Fig (1): System Architecture Diagram for Smart Crop Advisory System (SCAS)

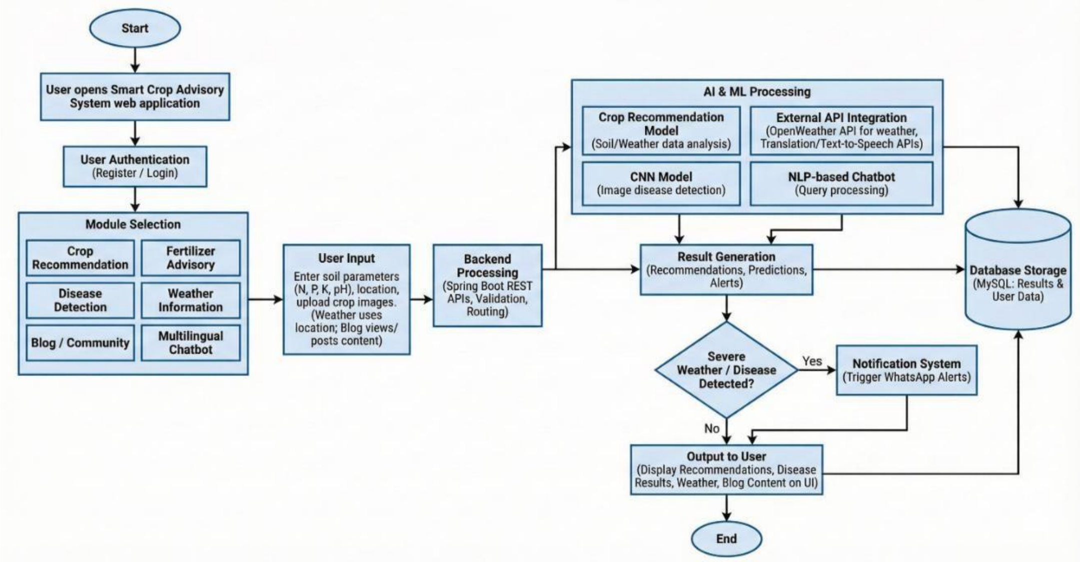

Fig (2): Workflow of Smart Crop Advisory System (SCAS)

IV.RESULT & DISCUSSION

The Smart Crop Advisory System (SCAS) was evaluated using real experimental results obtained from trained machine learning and deep learning models for crop recommendation, fertilizer recommendation, and disease detection. Multiple algorithms were implemented, tuned, and compared to identify the best-performing models. Ensemble learning techniques were further applied to improve prediction accuracy and robustness.

Crop Recommendation Results

For crop recommendation, multiple machine learning models were trained and evaluated, including Random Forest (RF), K-Nearest Neighbors (KNN), and Extreme Gradient Boosting(XG Boost). Hyperparameter tuning was performed to obtain the best possible performance from each model. Among the individual models, Random Forest achieved the highest validation accuracy of 90.41%, followed by XG Boost with 89.97% and KNN with 87.60%.

To further improve performance, a stacking ensemble model was developed by combining RF, KNN, and XG Boost as base learners with a meta-model. The final stacking model achieved a test accuracy of 93.71%, demonstrating significant improvement over individual classifiers. This confirms that ensemble learning effectively captures complex relationships between soil nutrients and crop suitability.

Table 1: Crop Recommendation Model Performance

Model	Best Accuracy (%)
KNN	87.60
Random Forest	90.41
XG Boost	89.97
Stacking Ensemble	93.71

Fertilizer Recommendation Results

The fertilizer recommendation module was evaluated using ensemble-based machine learning techniques. Random Forest, KNN, and XG Boost models were individually tuned to achieve optimal performance. The tuned Random Forest model achieved a best score of 93.53%, while XG Boost slightly outperformed it with 93.60%. KNN showed comparatively lower performance with 83.41% accuracy.

To enhance prediction reliability, a voting ensemble model was constructed by combining the tuned RF, KNN, and XG Boost classifiers. The final voting ensemble achieved a test accuracy of 95.75%, indicating strong generalization capability. This result demonstrates that ensemble-based fertilizer recommendation significantly improves nutrient management decisions and reduces the risk of over- or under-fertilization.

Table 2: Fertilizer Recommendation Model Performance

Model	Best Accuracy (%)
KNN	83.41
Random Forest	93.53
XG Boost	93.60
Voting Ensemble	95.75

Disease Detection Results

The disease detection module was implemented using a Convolutional Neural Network (CNN) trained for 15 epochs on crop leaf images. During initial training, the model achieved a training accuracy of 60.20% in the first epoch, indicating gradual feature learning. As training progressed, the model demonstrated consistent improvement in both training and validation performance.

At the final epoch, the CNN achieved a training accuracy of 87.17% and a validation accuracy of 89.05%, with a reduced validation loss of 0.3190. These results indicate effective feature extraction and strong generalization capability for disease classification.

Table 3: CNN Disease Detection Performance

Metric	Value
Final Training Accuracy	87.17%
Validation Accuracy	89.05%
Validation Loss	0.3190
Number of Epochs	15

Weather Alert and Notification Results

The integration of real-time weather data enabled timely alerts for adverse climatic conditions. WhatsApp notifications were successfully triggered during simulated extreme weather scenarios, helping farmers take preventive actions.

Discussion

Experimentation shows that using multiple models outperforms all others. By combining classifier results to make predictions through stacking and voting methods of combination, more accurate and reliable results were obtained than any one classifier alone. In addition, CNN technology performed well in the validation process, thus allowing the use of images to help with plant disease identification. Finally, all results support the use of SCAS as a research-based system for the improvement of agricultural decision making, productivity, and sustainability.

V.FUTURE WORK

Enhancing SCAS with advanced technology will create an accuracy-oriented, scalable and easily accessible advisory system for everyone in the agricultural industry.

Building on the foundation of SCAS as it stands, the integration of IoT soil sensors will provide real-time observation of soil moisture, nutrients and environmental conditions, allowing for more targeted recommendations. Edge computing can be integrated into SCAS to provide an offline prediction option for rural farmers with limited access to internet or high-speed connectivity. Additionally, a dedicated mobile application to provide language and voice support will allow farmers with little technical experience to gain access to this advisory service. In addition to providing reliable data-driven analysis of crop prices in the marketplace, the integration of predictive analytics with supply chain management can empower farmers to make educated, timely, strategic decisions related to the sale and storage of their produce. Finally, as SCAS develops, the addition of machine learning capabilities will enable farmers to continually improve their results by providing timely input and feedback, and to receive continual analysis of their crops via the real-time data provided by the sensors. Ultimately, these upgrades will provide SCAS with the characteristics of a complete, intelligent, farmer-centric smart agricultural ecosystem.

VI.CONCLUSION

SCAS is an artificial intelligence/machine learning application for modern agriculture focused on major challenges facing today's agricultural system. SCAS incorporates a review of the current literature and demonstrates how intelligent computing technologies support farmers in selecting crops, managing fertilization, diagnosing disease, and reducing weather-related risks via technological advancements. SCAS's capability to unify the functions of crop selection, fertilizer recommendations, disease diagnosis, and weather-related alerts into one application provides farmers with more accurate information than traditional farming employs using their own experience and unsubstantiated advice. Through the application of machine learning techniques such as Random Forest (structured datasets) and Convolutional Neural Networks (CNNs) to unstructured datasets including images for disease diagnosis, SCAS has shown that it can utilize multiple techniques in developing intelligent systems that provide accurate recommendations. In addition, SCAS employs Spring Boot as a microservices architecture to enable scalability, modularization, and ease of use for end-users in the field. Furthermore, the use of web-based interfaces and notifications increases farmer access and usability, particularly within rural areas. Therefore, by combining intelligent analytics and scalable software architecture, SCAS is able to improve agricultural productivity, decrease operational risk, and promote sustainable practices. The study confirms that smart advisory systems have significant potential to transform traditional agriculture into a data-driven and technology-enabled ecosystem.

REFERENCES

1. Pawan, D. Yadav, R. K. Sharma, M. Kumar, J. Rani, and N. Sharma, “An effective approach for crop recommendation using features of specific locations and seasons to maximize crop yield production using machine learning,” International Journal of Intelligent Systems and Applications in Engineering, vol. 12, no. 18s, pp. 844–850, 2024.
2. G. Mamatha and J. S. Nayak, “A novel design and implementation of fertilizer recommendation system based on hybrid machine

learning models," SSRG International Journal of Electrical and Electronics Engineering, vol. 11, no. 11, pp. 448–460, 2024. doi:10.14445/23488379/IJEEE-V11111P141

3. A. Chaudhary, M. Gupta, and U. Tiwari, "Crop disease detection using deep learning models," in Proceedings of the 2023 IEEE International Conference on Recent Advances in Systems Science and Engineering (RASSE), IEEE, 2023.
4. A. A. Ingole and N. E. Karale, "Java in microservices architecture: A study on Spring Boot and cloud-native development," International Journal of Ingenious Research, Invention and Development, vol. 4, no. 2, 2025.
5. A. Murad, "Comparative performance and user experience: PWAs vs. native mobile applications," in Proceedings of the 11th International Scientific and Practical Conference on Current Issues and Prospects for the Development of Scientific Research, Orléans, France,pp. 246–260,2025. doi: 10.51582/interconf.19-20.08.2025.027
6. Kaggle, "Agricultural datasets for crop, fertilizer, and disease prediction," 2024. [Online]. Available: https://www.kaggle.com
7. National Bank for Agriculture and Rural Development (NABARD), Annual Report, NABARD, India, 2022.
8. Open Weather, "Open Weather API documentation," 2024. [Online]. Available: https://openweathermap.org/ api
9. Python Software Foundation, "Python documentation,"2024.[Online]. Available: https://docs.python.org/3/
10. F. Pedregosa et al., "Scikit-learn: Machine learning in Python," Journal of Machine Learning Research, vol. 12, pp. 2825–2830, 2011.