International Journal for Innovative Engineering and Management Research
PEER REVIEWED OPEN ACCESS INTERNATIONAL JOURNAL
www.ijiemr.org

CROP RECOMMENDATION SYSTEM USING MACHINE LEARNING

Mr. Uday Sharma, Mr. Arshan Khan, Mr. Noah Alvin Roopson
Undergraduate Students, BS-MS (Computer Science)
Vishwa Vishwani Institute of Systems and Management, Hyderabad, Telangana, India
Email: itsudaysharma16@gmail.com | Mobile No.: 8247614489

ABSTRACT

Agriculture supports the livelihoods of millions of people and plays an important role in India's economy. Choosing the right crop is one of the decisions farmers make each season, and that choice depends on factors such as soil nutrients, temperature, humidity, soil pH, and rainfall. Since these conditions can vary from one place to another, selecting a suitable crop is not always straightforward. This project explores how Machine Learning can help by using these measurements to suggest a crop that matches patterns found in an agricultural dataset. The proposed system takes nitrogen, phosphorus, potassium, temperature, humidity, soil pH, and rainfall as inputs. A trained classification model will analyse the values and return a predicted crop through a simple user interface. The project brings together data preparation, model training, evaluation, and application development in one accessible tool. It is intended to support learning and early-stage decision-making, rather than replace local farming knowledge or expert advice. The quality of the recommendation will depend on the dataset, the model's measured performance, and how closely the data reflects real growing conditions.

Keywords: Crop Recommendation System, Machine Learning, Precision Agriculture, Crop Prediction, Soil Nutrients

1. INTRODUCTION

Agriculture is closely linked to food supply, employment, and the daily lives of many communities in India. Yet farming conditions are not the same everywhere. Soil quality, access to water, local weather, and seasonal rainfall can differ significantly, even between nearby areas. Because of these differences, a crop that grows well in one location may not be the best choice in another. Making an unsuitable choice can also mean spending water, fertiliser, time, and labour without getting the expected result.

Farmers often draw on years of experience, observations of local conditions, and advice from other farmers or agricultural specialists. That knowledge is valuable. At the same time, digital tools can help organise several measurements together and make it easier to compare them with patterns found in existing agricultural data.

Machine Learning (ML) allows a computer to learn patterns from examples and use them to make predictions about new cases. In this project, the examples are agricultural records that contain soil and environmental measurements along with a crop label. After training on these records, the model can take a new set of values and predict which crop category most closely matches what it has learned. The proposed system uses nitrogen, phosphorus, potassium, temperature, humidity, soil

pH, and rainfall.

The aim is to build a small, easy-to-use application that connects the trained model to an interactive interface. Users will enter the required values and receive a predicted crop name. Python libraries can support data preparation and model training, while Streamlit can be used to create the interface. The result should be treated as a learning and decision-support prototype, not as a guarantee that a crop will grow successfully on a particular farm.

2. REVIEW OF LITERATURE

Machine Learning has been explored for a range of agricultural tasks, including crop classification, yield estimation, plant disease identification, and farm resource planning. In a classification problem, the system predicts a category. For this project, the category is the crop label associated with the input measurements.

Breiman (2001) introduced Random Forest, a method that combines the predictions of many decision trees. It can model relationships between several features and is often considered for structured datasets. However, its performance depends on the data and the way the model is trained and evaluated, so it should be tested rather than assumed to be the best option.

Pedregosa et al. (2011) presented scikit-learn, a widely used Python library that includes machine-learning algorithms and tools for model evaluation. Pandas, discussed by McKinney (2010), helps users load, organise, and analyse tabular data. These tools are useful for building a project workflow that moves from a dataset to a trained model and then to a prediction interface.

For crop recommendation, the model learns from the examples included in its dataset. If the data does not represent the region where the system will be used, contains errors, or covers only a narrow range of conditions, its predictions may not work as well in practice. For a final academic version, the literature review should be strengthened with verified research papers specifically about crop recommendation, including their datasets, methods, and reported results.

3. RESEARCH GAP

A simple tool that brings soil and environmental measurements together could help users understand how data-based crop prediction works. However, a prediction from a public dataset cannot automatically account for every factor that matters on a real farm.

For example, soil texture, irrigation access, crop variety, planting season, pest conditions, local weather changes, and market demand may all affect a farmer's decision. These details may not be included in a basic dataset. A model might also perform well on a test set that resembles its training data but be less reliable when used in a different region.

This project therefore focuses on developing an accessible prototype that accepts selected agricultural measurements, applies a trained classifier, and displays its prediction clearly. The goal is to demonstrate a practical workflow while being open about its limits; it is not to suggest that crop recommendation systems do not already exist.

4. OBJECTIVES OF THE STUDY

- To understand how Machine Learning can be used to support agricultural decisions.
- To identify the soil and environmental measurements included in the selected dataset.
- To inspect and prepare a labelled dataset for model development.
- To train a supervised classification model that predicts crop labels from the selected inputs.
- To create an interactive interface where users can enter measurements and view a predicted crop.
- To evaluate the model using unseen data and suitable classification metrics.
- To explain the limits of dataset-based predictions and suggest practical improvements.

5. RESEARCH METHODOLOGY

5.1 Research design

The project follows a practical, application-focused approach. The planned stages are to choose and inspect a dataset, prepare the data, train a classification model, test its performance, and connect it to a simple interface. This article describes the proposed approach; it does not report experimental results from a completed implementation.

5.2 Nature and source of data

The system is intended to use a labelled crop-recommendation dataset from a public source. Each record should contain values for the selected input features and the corresponding crop label. Before using the dataset, its source, measurement units, feature definitions, crop categories, and known limitations should be checked. Reliable agricultural publications can also help explain the meaning of the input parameters.

5.3 Data preparation

The dataset will be loaded and reviewed using Pandas. This step includes checking for missing values, duplicate records, inconsistent data types, and values that appear implausible. The input columns will then be separated from the crop label. The data will be split into training and testing sets: the model will learn from the training set, while the testing set will be kept aside to estimate how well it predicts data it has not seen. A fixed random seed and stratified splitting, where appropriate, can make the process easier to reproduce.

5.4 Model selection and training

A Random Forest classifier is proposed as the first model because it can learn non-linear relationships between numerical measurements and crop categories. The model will be trained using the selected features and crop labels. If time allows, another suitable classifier, such as a Decision Tree, can be tested using the same data split. Comparing models in a consistent way will give a fairer view of their performance.

5.5 Model evaluation

The model should be evaluated using predictions on the held-out test set. Accuracy gives an overall measure of correct predictions, while precision, recall, and F1-score provide more detail about performance for individual crop categories. A confusion matrix can show which crop labels the model tends to mix up. Actual scores should be added only after the model has been implemented and tested; no accuracy value is assumed in this article.

5.6 Application development

The interface will be created with Streamlit. It will include fields for nitrogen, phosphorus, potassium, temperature, humidity, soil pH, and rainfall. When a user submits the values, the application should check that the inputs are valid, arrange them in the same order used during training, and pass them to the saved model. The predicted crop label will then appear on the screen.

5.7 Practical considerations

The application should make it clear that its result comes from a dataset and does not guarantee crop growth, yield, or profit. It should not imply that it considers conditions that were not included in its inputs. Before using the system to guide real farming decisions, its predictions would need to be checked against regionally relevant data and reviewed with agricultural specialists.

6. CONCEPTUAL BACKGROUND

6.1 Crop Recommendation System

A Crop Recommendation System is a tool that suggests or predicts a crop category from selected agricultural inputs. In this project, the prediction is based on the soil nutrient and environmental features available in the training dataset.

6.2 Machine Learning classification

Supervised classification learns from examples for which the correct category is already known. During training, the model studies records with crop labels. During prediction, it uses the patterns it learned to assign a label to a new set of values. The output reflects the data the model was trained on; it is not a direct measurement of how a crop will perform in the future.

6.3 Nitrogen, phosphorus, and potassium

Nitrogen, phosphorus, and potassium are commonly shortened to N, P, and K. They are essential plant nutrients and support different aspects of plant growth and development. Crop needs vary, so nutrient readings should be understood in context and measured consistently. The units and ranges entered into the system must match those used in the selected dataset.

6.4 Environmental parameters

Temperature and humidity describe aspects of the growing environment, rainfall represents precipitation over a defined period, and soil pH indicates how acidic or alkaline the soil is. These factors can be relevant to crop growth, but their meaning depends on how they are measured and represented in the dataset.

6.5 Random Forest classifier

Random Forest builds several decision trees and combines their predictions to produce an overall result. This can help the model capture relationships between multiple input features. Still, predictions may be less dependable when the input values differ greatly from the examples used for training, which is why testing on unseen data matters.

6.6 Streamlit interface

Streamlit is a Python framework for creating interactive data applications. It can provide number fields, buttons, and a result area without requiring the user to work directly with the code. In this project, it will connect the input values to the trained model and display the predicted crop.

7. ANALYSIS AND DISCUSSION

7.1 Input parameters

The proposed model uses seven inputs: nitrogen, phosphorus, potassium, temperature, humidity, soil pH, and rainfall. Together, these values provide the information the classifier uses to match a record to a crop label. The interface must use the same feature names, order, and units as the model expects. Incorrect units or values outside the dataset's usual range could lead to misleading predictions.

7.2 System workflow

The process starts with loading and checking the dataset. The input columns and crop labels are separated, and the records are divided into training and testing sets. The classifier is fitted on the training data and evaluated on the test data. Once the model has been checked, it can be saved and connected to the interface. When a user enters values, the application validates them, formats them correctly, and requests a prediction.

7.3 Understanding model performance

Accuracy is useful for a quick overview, but it does not tell the whole story if some crop categories are predicted more reliably than others. Precision shows how often predictions for a particular category are correct, recall shows how many examples of that category are found, and the F1-score balances the two. A confusion matrix can reveal common mix-ups between categories. The final article should report the values produced by the actual test run.

7.4 Interpreting the recommendation

The crop shown by the system is the category the model predicts for the values entered. It should not be presented as a guaranteed best crop for a particular farm. Factors such as irrigation, soil texture, crop variety, planting time, pests, and local costs may not be represented in the dataset, yet they can influence the final decision.

7.5 Ease of use

A clear interface can make the project easier to understand and demonstrate. Each input should have a clear label, the result should be easy to find, and the application should display a message when a value is missing or invalid. A short note about the system's limitations would also help users interpret the prediction appropriately.

7.6 Limitations and future scope

The first version is limited by the dataset and the seven selected inputs. Future versions could explore regional soil information, seasonal conditions, irrigation availability, crop duration, or other factors if reliable data can be obtained. The model could also be compared with other classifiers and checked using examples reviewed by agricultural experts from the intended region.

8. MAJOR FINDINGS

The system has not yet been experimentally evaluated, so the points below are expected outcomes and design observations, not measured findings.

- The prototype is expected to demonstrate how agricultural measurements can be used to predict a crop label from a dataset.
- A single interface can bring soil nutrient and environmental inputs together for a prediction.
- Testing on unseen records is necessary to estimate how the model may perform beyond its training examples.
- Accuracy alone may be insufficient; precision, recall, F1-score, and a confusion matrix can provide a fuller picture.
- The usefulness of the output will depend on the dataset's quality, coverage, and relevance to the intended region.
- A prediction cannot guarantee successful cultivation because important real-world factors may be missing from the dataset.

9. SUGGESTIONS AND RECOMMENDATIONS

- Select a dataset with a clear source, documented features, consistent units, and understandable crop labels.
- Check the data carefully before training and record the cleaning and preparation steps.
- Test the model on unseen data and report only the scores obtained from actual testing.
- If time allows, compare more than one suitable algorithm using the same evaluation method.
- Make sure input units and feature order match the format used to train the model.
- Explain the system's limitations during the demonstration and within the interface.
- For real-world use, seek regionally relevant data and consult agricultural specialists before relying on predictions.
- Consider adding seasonal, irrigation, and regional information in future versions when dependable data is available.

10. CONCLUSION

This project explores how Machine Learning can be used to support crop selection through a simple, data-driven application. The proposed system takes nitrogen, phosphorus, potassium, temperature, humidity, soil pH, and rainfall as inputs and uses a supervised classification model to predict a crop label. A Streamlit interface will make it possible for users to enter these values and view the model's output without interacting directly with the code.

The value of the system will depend on the quality and relevance of the dataset, the model's performance on unseen data, and whether the inputs are entered correctly. Since real farming decisions also depend on local conditions and economic factors, the result should be treated as guidance from a prototype rather than a guarantee of crop suitability, yield, or profit. Further work should focus on transparent testing, regionally relevant data, and expert review before the system is considered for practical use.

REFERENCES

Breiman, L. (2001). Random forests. *Machine Learning, 45*, 5–32. https://doi.org/10.1023/A:1010933404324

McKinney, W. (2010). Data structures for statistical computing in Python. In *Proceedings of the 9th Python in Science Conference* (pp. 56–61). https://doi.org/10.25080/Majora-92bf1922-00a

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., VanderPlas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, É. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research, 12*, 2825–2830.

Streamlit. (n.d.). *Streamlit documentation*. https://docs.streamlit.io/

Food and Agriculture Organization of the United Nations. (n.d.). *FAO Soils Portal*. https://www.fao.org/soils-portal/