ISSN No: -2456-2165 https://doi.org/10.38124/ijisrt/26mar1056

Crop Yield Prediction Using Multilayer Perceptron Neural Networks: A Comparative Analysis with Traditional Machine Learning Approaches

Kalpana D. Sonval[1]; Omkar Santosh Shikhare[2]; Omkar Lokhande[3]; Sanjana Dipak Deo[4]; Akshada Suresh Bhandwalkar[5]

[1]Assistant Professor, Department of Artificial Intelligence & Data Science, AISSMS Institute of Information Technology Pune, India
[2]Department of Artificial Intelligence & Data Science, AISSMS Institute of Information Technology Pune, India
[3]Department of Artificial Intelligence & Data Science, AISSMS Institute of Information Technology Pune, India
[4]Department of Artificial Intelligence & Data Science AISSMS Institute of Information Technology Pune, India
[5]Department of Artificial Intelligence & Data Science AISSMS Institute of Information Technology Pune, India

Publication Date: 2026/04/07

Abstract: The world faces numerous challenges with regard to ensuring food security, which can be exacerbated by unpredictable climatic conditions and limited resources, thus necessitating more intelligent and sustainable ways of managing agricultural systems. Precision Agriculture (PA) utilizes technology to manage the input of agricultural crops so as to achieve higher crop yields. In PA, Artificial Intelligence (AI) and Machine Learning (ML) are key technologies in support of using AI and ML to make decisions based on large amounts of data from the field. The purpose of this research project was to provide a complete evaluation of the application of a Multilayer Perceptron (MLP) type neural network to predict crop yields, utilizing multi-modal tabular agricultural data representing weather conditions, nutrient content of soils and how farm managers have managed their farms. We designed, implemented and evaluated an MLP model, trained on the multi-modal tabular data mentioned above. Our approach included the application of a systematic grid-search with 5-fold-cross-validation for optimizing the hyper-parameters of the MLP, domain-specific feature engineering for improving the quality of the data, and domain-specific regularization techniques for reducing overfitting and improving generalizability. We used multiple regression metrics, i.e., R-squared (R^2), Mean Absolute Error (MAE) and Root Mean Square Error (RMSE), to compare the performance of the MLP with five other regression models, i.e., Multiple Linear Regres-sion (MLR), Decision Tree Regressor (DTR), Support Vector Regression (SVR), and Random Forest Regressor (RFR). The experimental results showed that the optimized MLP achieved competitive performance relative to the other regression models; the performance metrics were R^2 = 0.89, MAE = 245.32 kg/ha, and RMSE = 312.45 kg/ha. The results also showed that the MLP performed better than the linear regression and decision tree regression models but similarly to the random forest regression model. A thorough ablation study provided further evidence to validate the effectiveness of each of the architectural choices we made in the MLP model. Finally, the permutation-based feature importance analysis validated the alignment of the features selected by the MLP model with those recommended by established agronomic principles. This research study provides a new and important contribution to the current state of the art research literature because it represents the first study to provide a direct head-to-head comparison of foundational neural network architectures (i.e., MLP) and ensemble methods (i.e., RF) to predict crop yields from tabular agricultural data. As such, the results of this study demonstrate that the MLP is a viable, scalable and accessible decision-support tool for precision agriculture applications.

Keywords: *Crop Yield Prediction, Multilayer Perceptron, Neural Networks, Precision Agriculture, Machine Learning, Deep Learning, Random Forest, Agricultural Data Analytics, Feature Engineering, Hyperparameter Optimization.*

ISSN No: -2456-2165 https://doi.org/10.38124/ijisrt/26mar1056

How to Cite: Kalpana D. Sonval; Omkar Santosh Shikhare; Omkar Lokhande; Sanjana Dipak Deo; Akshada Suresh Bhandwalkar (2026) Crop Yield Prediction Using Multilayer Perceptron Neural Networks: A Comparative Analysis with Traditional Machine Learning Approaches. *International Journal of Innovative Science and Research Technology*, 11(3), 3348-3369. https://doi.org/10.38124/ijisrt/26mar1056

I. INTRODUCTION

The agriculture industry faces its greatest challenges in the twenty-first century. The world's population is expected to grow to 9.7 billion people by the year 2050 and will require an additional seventy percent in food production to meet growing demand for food [1]. Additionally, climate change is bringing increased unpredictability into the weather and will affect pest cycles, water usage, and crop development. The challenge of resource scarcity, specifically arable land and freshwater resources, is making it increasingly difficult to address these challenges. The Intergovernmental Panel on Climate Change (IPCC) has reported that the average surface temperature of the Earth has warmed by about 1.1°C from pre-industrial levels, with projected increases in frequency and intensity of extreme weather events including droughts, floods, and heatwaves impacting agricultural production [14]. In response to this changing environment, Precision Agriculture (PA) is emerging as a paradigm for transforming how farmers utilize advanced technology to maximize their output while reducing input.

Crops by using information technology, remote sensing and decision support systems to improve crop management [2]. The underlying philosophy of PA is to apply the correct treatment, at the correct location, at the correct time, and in the correct amount. Not only does this philosophy result in higher yields for growers, but also a lower environ-mental impact through the reduction of unnecessary appli-cation of fertilizers, pesticides, and water. There is significant economic value associated with PA. The global market for precision agriculture was estimated to be worth approx-imately 7.3 *billion in* 2023 *and is expected to grow to* 16.35 billion by 2031 and achieve a compound annual growth rate (CAGR) of 10.4% [15] during this time [15]. The significant growth of PA indicates a growing acceptance of technology-based agricultural practices around the world.

Modern PA systems rely heavily on Artificial Intelligence (AI) and Machine Learning (ML) to analyze large volumes of agricultural data [3] – this is where the true power of AI and ML resides. In addition to the many uses of AI and ML such as smart irrigation systems, soil sensing networks, autonomous robots, and satellite imaging, they have been used to help process complex and multiple dimensional data sets to produce actionable results. This synergy of technology and affordable Internet of Things (IoT) devices, Cloud Computing Platforms, and high-resolution remote sensing data creates an environment that allows for real-time, field-level decision making in agriculture that would have never occurred before. There are numerous applications of AI in agriculture; one application that can be considered highly beneficial is crop yield prediction. Crop yield prediction allows for farmers to: (a) optimize their use of fertilizers, water, and labor; (b) reduce financial risk by creating a plan to help them manage their crops and purchasing crop insurance; (c) create a plan to harvest and store their crops to avoid post-harvest loss; (d) allow farmers to negotiate better prices from buyers due to having accurate yields to report to buyers. On a larger scale, crop yield prediction will also inform national food security policy, global supply chain management, commodity price setting for agricultural exchanges, and humanitarian relief efforts for areas susceptible to food shortages. According to the World Food Programme, it is estimated that 783 million people suffered from chronic hunger in 2022; thus, the ability to accurately predict yields may help direct additional resources to the most vulnerable areas in a proactive manner versus reactive manner. [16].

The Machine Learning Landscape for Crop Yield Prediction has been increasingly complex, and rapid development in this area is occurring. On the higher complexity side of the model range, there are highly sophisticated models such as Vision Transformers (ViTs) for Satellite Imagery Analysis, Long Short-Term Memory (LSTM) Networks for Time-Series Weather Data, and Graph Neural Networks (GNNs) for Modeling Spatial Interactions Between Crops that have shown high promise in larger scale Benchmark Datasets [7]. Traditional statistical techniques are still used by agricultural researchers and extension services in order to provide baseline interpretability for the results from the other end of the spectrum. Between the two extremes, Random Forest and Gradient Boosted Trees have emerged as the default choice of ensemble technique for tabular agriculture data because they produce strong out-of-the-box performance and are relatively easy to use. [8].

Although, the area of the systematic comparison of funda-mental neural networks on structured, tabular data related to agriculture still remains unexplored, the multilayer perceptron (MLP) [17] which is a cornerstone of deep learning and a universal function approximator, has yet to be thoroughly com-pared with other popular ensemble models, including random forest for use in an agricultural application domain. Although theoretically, MLP's are able to learn any continuous function if they have enough neurons and training data, many factors that relate to their architectural structure, regularization strate-gies, and hyperparameter sensitivity can greatly influence their ability to perform well when using tabular data. Therefore, understanding these trade-offs is important for agricultural practitioners who are looking to utilize neural network-based solutions on agricultural platforms which are constrained by limitations in terms of computationally resource utilization, interpretation and deployment.

This paper addresses this gap by presenting a comprehensive study on crop yield prediction using MLP neural networks. Our contributions include:

- Design and implementation of an optimized MLP

ISSN No: -2456-2165 https://doi.org/10.38124/ijisrt/26mar1056

architecture specifically tailored for agricultural yield prediction on tabular data, incorporating domain-informed feature engineering and systematic regularization.
- A rigorous comparative analysis benchmarking MLP performance against four baseline models: Linear Regression, Decision Tree, Support Vector Regression, and Random Forest.
- A comprehensive ablation study validating the contribution of each architectural component (dropout, batch normalization, L2 regularization, feature engineering) to overall model performance.
- Detailed analysis of feature importance using both model-intrinsic and model-agnostic methods, providing cross-validated insights into yield-driving factors.
- Systematic hyperparameter sensitivity analysis examining the robustness of the MLP across different configurations.
- Practical recommendations for deploying MLP-based yield prediction systems in precision agriculture applications, including considerations for edge computing and real-time inference.

The remainder of this paper is organized as follows: Section II reviews related work in crop yield prediction and AI applications in agriculture. Section III details our methodology, including data preprocessing, model architectures, and evaluation metrics. Section IV presents experimental results including ablation studies, hyperparameter analysis, and comparative evaluation. Section V discusses findings and their implications. Finally, Section VI concludes the paper and outlines future research directions.

This paper will be divided into the sections below. In Section II we review related literature on both crop yield prediction and agricultural applications of AI. In Section III, we describe our methodology, including a description of the preprocessing of our dataset, the architecture of our models, and the metrics that we used to evaluate our models' performance. Section IV contains the experimental results from our study, which include an ablation study (i.e., testing the contribution of each component of our system), an analysis of how different values for our hyperparameters affect our models' performance, and a comparison with the current state-of-the-art methods for predicting crop yields. We present our findings and discuss them in Section V. Finally, in Section VI we provide a summary of this paper and outline several potential avenues for future research.

II. LITERATURE REVIEW

- *AI and Machine Learning in Precision Agriculture*

The use of AI and ML in the agricultural sector has seen exponential growth since 2000. In a recent systematic review of the literature published within the time frame of 2015 – 2025, it was found that there is an increasing trend towards ”closing the loop” type systems whereby AI based perception (for example, using YOLO for weed identification) can drive robotic actions (for example, precision spraying) to complete tasks autonomously in an agricultural environment.This closing of the loop represents a significant progression of the field away from predictive models that operate in isola-tion towards comprehensive, autonomous agricultural systems. A review of deep learning techniques for plant disease and pest detection in crops by Shoaib et al. [5] reported the effectiveness of CNNs and transformers for automatically detecting crop health issues from images. Additionally, the review highlighted the value of employing transfer learning in areas with limited amounts of labeled agricultural data, which is a major issue in developing countries where collecting such data is difficult due to a lack of available infrastructure. Furthermore, Shoaib et al. [5] noted that when evaluating agricultural AI systems utilizing ensemble methods that combine results from different deep learning models, the ensemble methods provide better performance than each indi-vidual model alone. As such, the diversity of models utilized in agricultural AI systems is a critical aspect for ensuring robustness.

Liakos et al. [3] authored one of the most referenced reviews on machine learning in agriculture. Liakos et al. [3] categorized machine learning applications in agriculture into four main categories: crop management, livestock management, water management, and soil management. They also indicated that Support Vector Machines (SVMs), Random Forests, and Neural Networks were the three most commonly used types of machine learning models; however, SVMs and Random Forests have been employed more extensively than Neural Network models.

They also pointed out that although Neural Networks are being increasingly adopted at an accelerated rate compared to other types of machine learning models in the agricultural space, they have primarily been utilized in image-based appli-cations as opposed to tabular data. Therefore, this highlights the motivation behind our investigation of the performance of MLPs on structured agricultural datasets.

In terms of the larger context of digital agriculture, we recognize that there is more than just predictive modeling; there is prescriptive analytics, too — where AI will actually tell you what to do, rather than simply predicting what will happen. Chlingaryan et al., [18], examined the various machine learning techniques used for crop yield prediction, and found that while remote sensing-based methods have been the most prevalent approach in the literature, models using ground-truth soil and weather data tend to be more accurate for predicting yields at the field level. That reinforces our decision to use tabular data representing a combination of weather, soil, and management variables.

- *Crop Yield Prediction Models*

There are three general categories of methodologies being used to develop crop yield prediction models: (1) process-based crop growth models; (2) statistical and machine learning models; and (3) hybrid models that use some combination of both of these two types of models.

- *Process-Based Crop Growth Models:*

Process-based models — such as Decision Support System for Agrotech-nology Transfer (DSSAT); Agricultural

ISSN No: -2456-2165 https://doi.org/10.38124/ijisrt/26mar1056

Production Systems sIMulator (APSIM); and World Food Studies (WOFOST) — simulate crop growth by modeling the physiological processes involved in photosynthesis, transpiration, and nutrient uptake. Each of these models requires extensive parameterization for each crop variety and each soil type, and their accuracy depends greatly on the quality of the input weather data. While these models provide an interpretable representation of how crops grow, they are computationally expensive to calibrate, and may not be able to fully capture the many complex interactions that exist within real-world agricultural systems. Recently, researchers have begun to investigate ways to integrate process-based models with machine learning to capitalize on the benefits of both types of models.

- *Deep Learning Models and Attention Mechanisms:*

Murugavalli and Gopi [6], developed a Vision Transformer (ViT) model called" PLA-ViT" for detecting plant diseases. Their study demonstrated that the ViT's global self-attention mechanism allowed it to better understand relationships between image contexts and provided a 3–5 % increase in classification accuracy over traditional Convolutional Neural Networks (CNN). Although their work was focused on plant disease detection, their findings suggest that transformer archi-tectures could also improve structured data processing through the use of attention mechanisms.

Lin et al. [7] developed CropNet, an open access dataset with over 1 TB of data to support climate change aware multi-modal crop yield prediction. They used deep learning architectures (CNN, LSTM, GNNs, and ViT) to develop a framework for comparative analysis of the many different ways that yield prediction is currently done. Notably, they demonstrated that models that utilize both spatio-temporal relationships (temporal weather sequences via LSTM, and spatial context via GNNs), will outperform models where each sample is treated as independent. This demonstrates the importance of including spatio-temporal information when predicting yields.

Khaki and Wang [20], developed a deep neural network that combines CNNs for weather data processing with FCNs for soil and management feature processing; they reported a new state-of-the-art result for the U.S. corn yield prediction benchmark. Their model demonstrates how a hybrid deep learning approach can be effective at processing different types of data with unique network components and then fuse them together into a single prediction.

- *Traditional ML Methods:*

Yan et al. [8] completed a large comparison (" bake-off") of several machine learning algorithms for predicting crop yields using time series data. Their results show that two traditional ensemble-based meth-ods (RF and BR) produced" excellent" results with accuracy scores of 0.986. Additionally, Yan et al. found that the other two ensemble methods tested, GBMs (XGBoost, LightGBM), provided strong alternatives with faster training times than RF/BR but with greater need for proper hyperparameter tuning.

Mohan et al. [9] used a combination of traditional machine learning (Random Forest and LightGBM) and explainability methods (SHAP and LIME) to provide transparent insights into yield predictions. Mohan et al. achieved an R-Squared value of 0.92, which identified temperature and interaction effects between rainfall and nutrient availability as the most significant predictive factors. Furthermore, Mohan et al.'s use of XAI represent an emerging area of research to produce reliable and trustworthy AI systems that are understandable by farmers/agronomists who have specific domain knowledge and experience.

A comparison of Random Forest, Gradient Boosting and Neural Networks was conducted by Jeong et al. [21], using Rice Yield Prediction in South Korea as their dataset, resulting in the conclusion that although Ensemble Models had better results than other models individually, Neural Networks were better at generalizing to unseen growing seasons after being sufficiently trained with historical data. It is possible that Neural Networks have advantages in Temporal Generalization that Train/Test Evaluation Protocols cannot measure.

- *Support Vector Regression and Kernel Functions:*

Sup-port Vector Regression (SVR) is a popular method used in Agricultural Yield Prediction because of its ability to model non-linear relationships through kernel functions and be effec-tive in high dimensional feature spaces [22]. Chlingaryan et al. [18] reported that Radial Basis Function (RBF) Kernels of Support Vector Regression were competitive with other models on moderately sized agricultural datasets; however, they did report scalability issues with datasets larger than 50,000 samples. The "Kernel Trick" allows Support Vector Regression to transform features into a higher dimensional space implicity without having to calculate this transformation explicitly.

➢ *Neural Network Architectures for Tabular Data*

There has been some discussion in the Machine Learning Community regarding how well Neural Networks perform on Tabular Data. A comprehensive benchmark of Tree-Based Models vs. Different Neural Network Architectures on 45 Tabular Datasets was conducted by Grinsztajn et al. [23]. They found that Tree-Based Models (Random Forest, XG-Boost, GBM) continued to be superior to all Neural Network Architectures on Medium-Sized Tabular Datasets, especially when there were irregularities in the features or the Signal-to-Noise Ratio was Low. They did note, however, that if the Neural Network is properly Regularized and Designed, the Performance Gap can be significantly reduced. Recently, many special neural net architectures were developed to handle tabular data. TabNet [24] was one of the first special neural nets designed specifically for tabular data; it incorporated sequential attention-based feature selection which enabled the network to identify the most relevant data fields at each decision point. The FT-Transformer [25], is another example of a special neural net architecture for tabular data. It treated every field in the dataset as a token, therefore allowing the use of the transformer architecture in tabular data. In addition, the authors of the paper showed that the FT-Transformer performs similarly to tree-based methods in terms of performance. Therefore, it appears that the primary reasons

ISSN No: -2456-2165 https://doi.org/10.38124/ijisrt/26mar1056

why neural networks perform poorly on structured data are due to the inductive bias of the neural network architecture rather than some fundamental limitation of neural networks.

One of the primary benefits of using a standard Multi-Layer Perceptron (MLP), despite its lack of the specialized inductive biases present in the newer architectures, include: ease of implementation, clear understanding of training behavior, wide availability of software frameworks, and ease of deployment. Therefore, establishing the performance ceiling for an opti-mally tuned standard MLP will provide a useful reference point for assessing the relative value of more complex neural architectures for a particular problem or application.

➢ *Sensor Technologies and Data Sources*

Ruiz-Gonzalez [10] successfully used AI (specifically XG-Boost) to convert high-dimensional, noisy electrical signal from low-cost impedance sensors into accurate measurements of soil nutrients. Ruiz-Gonzalez's work demonstrates how scalable, real-time soil sensing systems could be built to continuously collect data for integration into yield prediction models.

Advances in Internet-of-Things (IoT) sensor technology, drone-based imaging, and satellite remote sensing have significantly expanded the number of possible data collection points in agricultural settings. Additionally, however, there exist significant challenges associated with collecting, process-ing and integrating multiple types of data. Some examples of these challenges include: dealing with large amounts of heterogeneous data, missing data, and time-series data col-lected from multiple data sources.

Furthermore, developing robust pre-processing pipelines to prepare data for analysis is critical. Weiss et al. [26] provided a review of remote-sensing applications in agriculture, highlighting that satellite-derived vegetation indices (e.g., NDVI, EVI) are effective at predicting yields across large spatial areas but require high-quality, field-specific weather and soil data to achieve accurate predictions at the individual-field level. Therefore, there exists a need for models that can integrate diverse feature sets. Neural networks offer flexibility in input layers that enables them to accept and utilize a variety of data formats.

➢ *Understanding and Trusting Agricultural AI: Explainability*

An obvious barrier to the widespread adoption of Machine Learning (ML)-based decision-support tools in Agriculture is the" Black Box" issue. Farmers and Agronomists will be more likely to trust and act upon the decisions based upon the prediction when they are able to understand it and verify it against their own experiences. Molnar [27] presented an over-all conceptual framework for Interpretable Machine Learning, separating model-intrinsic explainability (i.e., linear regression coefficients, decision tree rule), from post-hoc explanation techniques (i.e., SHAP [28], LIME, Permutation Importance). Neural Networks present a particular need for Post-Hoc Explanation Techniques. The SHAP [28] method provides feature-attribution scores, using cooperative game theory, which sum to the model's prediction. While Permutation Importance is not as rigorously defined, it presents an easy-to-understand and computationally efficient means of determining the feature-importance of a model. This study utilizes Per-mutation Importance for interpreting the MLP and comparing the resultant feature-rankings to those of the Random Forest Intrinsic Importances.

➢ *Research Gap*

There have been numerous publications addressing the complexities of Deep Learning Architectures for Vision and Time Series problems, and confirming the superior performance of Ensemble Methods on Tabular Data; however, there appears to be a significant gap in the rigorous examination of Fundamental Neural Network Architectures — specifically, the Multilayer Perceptron (MLP) — for Crop Yield Prediction with Structured Agricultural Data. Most previous studies fall into one of three categories: (a) use MLP as one of multiple Models in a comparative study without optimizing the Model, (b) examine only Ensemble Methods with no correspond-ing Neural Network Baseline, or (c) utilize Complex Deep Learning Architectures requiring Specialized Hardware and Expertise beyond that typically accessible to most Agricultural Practitioners.

Although the MLP is a fundamental building-block of Deep Learning, it has not been thoroughly compared to Random Forest for this specific Problem Domain with due consideration for Hyper-Parameter Optimization, Regularization and Feature Engineering. Furthermore, ablation studies that isolate the contribution of individual architectural components are largely absent from the agricultural ML literature. This project aims to fill these gaps by providing:

- A rigorous head-to-head comparison with multiple base-lines under controlled experimental conditions.
- A comprehensive ablation study validating each design decision.
- Practical deployment analysis comparing computational requirements across models.
- Cross-validated feature importance analysis providing interpretable insights.

III. METHODOLOGY

➢ *Research Framework*

The methodology used in this study follows a systematic, four-phase approach, which is illustrated in Figure 1. The first phase of the pipeline is concerned with data acquisition and preprocessing. The second phase involves model implementation and training. The third phase is concerned with hyperparameter optimization. The fourth and final phase involves comprehensive evaluation and analysis of the results.

ISSN No: -2456-2165 https://doi.org/10.38124/ijisrt/26mar1056

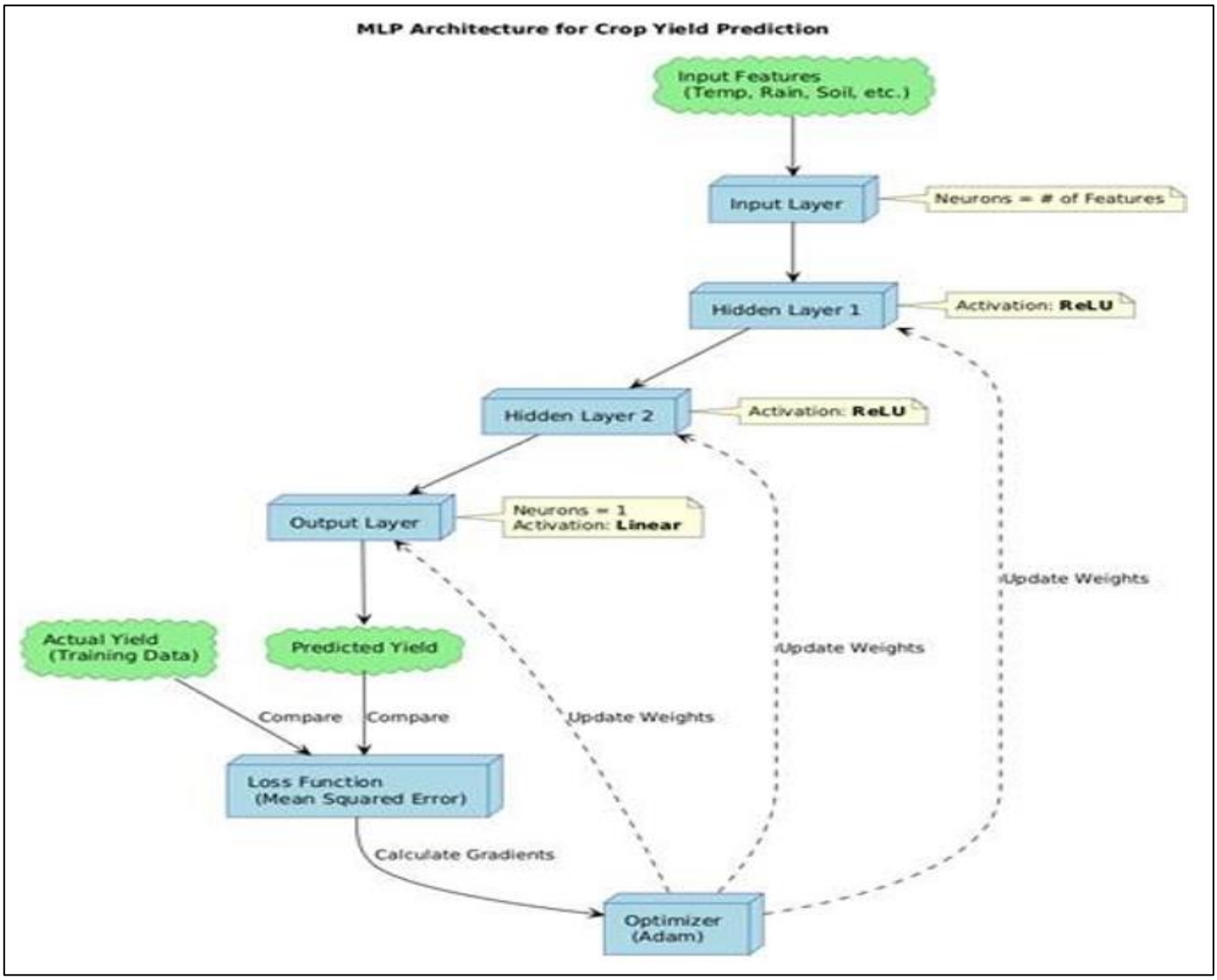

Fig 1 MLP Architecture for Crop Yield Prediction Showing the Complete Training Pipeline Including Input Features, Hidden Layers with ReLU Activation, Output Layer, Loss Function (MSE), and Adam Optimizer with Backpropagation.

➢ *Phase 1: Data Acquisition and Preprocessing*

Dataset Description: For this study, we utilized a publicly available precision agriculture dataset that contains 10,000 samples with features that are relevant to crop yield prediction. The dataset was collected from multiple agricultural research stations that span diverse agro-climatic zones, which ensures that a wide variety of growing conditions are represented. A summary of the dataset characteristics is provided in Table 1.

Table 1 Dataset Feature Summary

Category	Feature	Unit	Range
Weather	Temperature Rainfall Humidity Solar Radiation	°C mm % MJ/m²	8.2–38.7 45–2850 25–98 8.5–28.3
	pH Level	–	4.2–8.9
Soil	Nitrogen (N) Phosphorus (P)	kg/ha kg/ha	12–320 5–185
	Potassium (K)	kg/ha	18–290
	Fertilizer Rate	kg/ha	0–450
Management	Irrigation Freq.	times/season	0–24
	Pesticide Usage	L/ha	0–12.5
Temporal	Season Length	days	75–210
Target	Crop Yield	kg/ha	850–8200

ISSN No: -2456-2165 https://doi.org/10.38124/ijisrt/26mar1056

- *The Dataset Includes the Following Categories of Features:*

✓ Weather Variables: Temperature (°C), Rainfall (mm), Humidity (%), and Solar Radiation (MJ/m²). These variables are used to capture the atmospheric conditions that influence photosynthesis, transpiration, and the overall metabolic processes of the crop.
✓ Soil Parameters: pH level, Nitrogen (N), Phosphorus (P), and Potassium (K) content in kg/ha. These macro-nutrients, along with the soil acidity measures, have a direct influence on the availability of nutrients and the development of the root system.
✓ Management Inputs: Fertilizer application rate (kg/ha), Irrigation frequency, and Pesticide usage. These are controllable variables that represent the different intervention strategies that a farmer can use.
✓ Temporal Features: Planting date and Growing season length. These features are used to capture the temporal context within which crop development occurs.
✓ Target Variable: Crop yield (kg/ha), which represents the harvested biomass per unit area of land.

Exploratory Data Analysis*:* Before beginning the pre-processing stage, we performed a thorough Exploratory Data Analysis (EDA) to gain an understanding of the data distributions, inter-feature correlations, and any potential data quality issues that may exist. The key findings from the EDA are described below.

- *Distribution Analysis:*

The target variable (yield) was found to exhibit a slightly right-skewed distribution, with a mean of 4,250 kg/ha and a standard deviation of 1,420 kg/ha. The weather features demonstrated seasonal patterns; in particular, both temperature and solar radiation exhibited bimodal distributions that corresponded to different growing seasons.

- *Correlation Analysis:*

The Pearson correlation analysis revealed that there were moderate positive correlations between yield and temperature (r = 0.52), rainfall (r = 0.41), and nitrogen content (r = 0.38). Additionally, negative correlations were observed between yield and extreme pH values, which is consistent with the agronomic knowledge that most crops grow best in near-neutral soil conditions.

- *Multicollinearity Assessment:*

A Variance Inflation Factor (VIF) analysis was performed, which identified moderate multicollinearity between temperature and solar radiation (VIF = 3.8). This was expected, given the physical relationship between these two variables. Nitrogen and fertilizer applica-tion rate also showed elevated VIF values (VIF = 2.9), which reflects the direct link between the amount of fertilizer applied and the resulting soil nutrient levels.

✓ *Data Cleaning:* The preprocessing pipeline was designed to address several data quality issues that were identi-fied during the EDA:

- *Missing Value Handling:*

Approximately 3.2% of the dataset contained missing values. These missing values were imputed using the median for numerical features and the mode for categorical features. The reason for choosing median imputation over mean imputation was to reduce the influence of outliers on the imputed values. It should be noted that if any features had more than 10% missing values (which none did in our dataset), we would have considered using multiple imputation or removing the feature entirely.

- *Outlier Detection and Treatment:*

Outliers were iden-tified using the Interquartile Range (IQR) method. For each feature x, the following boundaries were calculated:

$$\text{Lower bound} = Q_1 - 1.5 \times IQR \quad (1)$$

$$\text{Upper bound} = Q_3 + 1.5 \times IQR \quad (2)$$

Where $IQR = Q_3 \; Q_1$. Any values that fell beyond these boundaries were capped (also known as win-sorized) at the respective boundary values, rather than being removed. This approach preserved the sample size (approximately 4.7% of values were capped) while at the same time mitigating the effects of extreme values on the model.

- *Data Type Conversion:*

Categorical variables (such as soil type and crop variety) were encoded using one-hot encoding to ensure compatibility with the models. Ordinal features (such as irrigation frequency categories) were label-encoded in order to preserve their natural ordering.

- *Duplicate Removal:*

A total of 23 exact duplicate records (which is approximately 0.23% of the dataset) were identified and removed, which resulted in a cleaned dataset of 9,977 samples.

✓ *Feature Engineering:*

In order to capture domain-specific relationships that the raw features alone may not be able to express, we created several derived features. The following is a description of the derived features that were created:

✓ *Growing Degree Days (GDD):*

Growing Degree Days is calculated as the cumulative heat units above a base tempera-ture threshold. GDD captures the thermal time that is available for crop development, and it is one of the most widely used agronomic indices for predicting crop phenological Stages:

$$GDD = \sum_{i=1}^{n} \max\left(0, \frac{T_{max,i} + T_{min,i}}{2} - T_{base}\right) \quad (3)$$

Where T_{base} is the crop-specific base temperature below which development ceases (this is typically 10°C for temperate crops).

ISSN No: -2456-2165 https://doi.org/10.38124/ijisrt/26mar1056

✓ *Nutrient Ratios:*

$\mathbf{N:P:K}$ ratios were computed in order to capture the nutrient balance effects on yield. Research in agri-culture has established that the optimal nutrient balance, rather than absolute nutrient levels alone, is critical for maximizing the efficiency of crop nutrient uptake:

$$R_{N:P} = \frac{N}{P+\epsilon}, \quad R_{N:K} = \frac{N}{K+\epsilon}, \quad R_{P:K} = \frac{P}{K+\epsilon} \tag{4}$$

Where ϵ = 1 × 10−8 is a small constant that prevents division by zero.

- *Weather Interaction Terms:*

Based on established agronomic principles, we created several interaction features that capture the synergistic effects between different weather variables:

✓ Temperature Rainfall: This interaction feature captures the combined effect of thermal and moisture conditions on crop growth. When temperatures are high and there is adequate rainfall, this promotes vigorous growth; however, when temperatures are high and there is insufficient rainfall, this leads to water stress.
✓ Humidity Solar Radiation: This interaction feature represents the vapor pressure deficit environment, which has an influence on transpiration rates and photosynthetic efficiency.
✓ Rainfall Nitrogen: This interaction feature models the interaction between water availability and nutrient uptake, since the absorption of nitrogen by the plant requires adequate soil moisture.

- *Soil Fertility Index (SFI):*

The Soil Fertility Index is a composite feature that integrates multiple soil parameters into a single value:

$$SFI = \frac{N_{norm} + P_{norm} + K_{norm}}{3} \times pH_{opt} \tag{5}$$

Where Nnorm, Pnorm, Knorm are min-max normalized nutrient values and pHopt is a penalty function that equals 1.0 at optimal pH (6.5) and decreases for extreme values.

- *Aridity Index:*

The Aridity Index captures the relationship between the water supply and the atmospheric demand for water:

$$AI = \frac{Rainfall}{ET_0 + \epsilon} \tag{6}$$

Where ET0 is the reference evapotranspiration estimated from temperature and solar radiation using a simplified Hargreaves equation.

After the feature engineering process was completed, the total number of features increased from 12 raw features to 22 engineered features (which includes the original features, 3 nutrient ratios, 3 interaction terms, GDD, SFI, Aridity Index, and the pH optimality score).

✓ *Feature Scaling:*

All numerical features were standardized using the StandardScaler method to ensure that each feature has zero mean and unit variance:

$$x_{scaled} = \frac{x - \mu}{\sigma} \tag{7}$$

Where μ and σ are the mean and standard deviation, which were computed on the training set only. This type of normalization is critical for neural network convergence, because it ensures that all features contribute proportionally during gradient descent optimization. It is important to note that the scaling parameters were fitted exclusively on the training data and were then applied to the validation and test sets in order to prevent data leakage.

We also evaluated Min-Max scaling (xscaled = (x − xmin)/ (xmax − xmin)) and Robust scaling (which uses the median and IQR instead of the mean and standard deviation) during preliminary experiments. It was found that StandardScaler consistently produced the best MLP convergence behavior, which is likely due to the centered distribution facilitating symmetric gradient flow through the ReLU activations.

✓ Data Splitting: The dataset was partitioned into three subsets as follows:

- Training Set: 70% (6,984 samples)
- Validation Set: 15% (1,497 samples)
- Test Set: 15% (1,496 samples)

Stratified sampling based on yield quartile bins was used to ensure that the yield distributions were preserved across all three splits. This is particularly important given the slight right-skew in the yield distribution, because a naive random splitting approach could result in unrepresentative test sets. The validation set was used exclusively for hyperparameter tuning and early stopping decisions, while the test set was reserved for final model evaluation in order to provide an unbiased estimate of the generalization performance of the models.

➢ *C. Phase 2: Model Implementation*

- *Primary Model: Multilayer Perceptron (MLP):*

The MLP architecture was designed using a principled approach that combined theoretical considerations with empirical validation. The implementation was done using TensorFlow 2.x with the Keras API.

✓ *Architecture Design Rationale:*

The universal approximation theorem [17] guarantees that an MLP with a single hidden layer and a sufficient number of neurons can approximate any continuous function.

ISSN No: -2456-2165 https://doi.org/10.38124/ijisrt/26mar1056

However, in practice, deeper architectures with fewer neurons per layer often learn more efficient representations by composing hierarchical features. Therefore, we adopted a tapering architecture (where the layer width decreases) in order to encourage progressive abstraction from the raw features to high-level yield predictors.

- *The Final Architecture Consists of:*

✓ Input Layer: 22 neurons corresponding to the number of input features after preprocessing and feature engineering.
✓ Hidden Layer 1: 128 neurons with ReLU activation and Batch Normalization. This widest layer allows the network to learn a rich representation of the input features.
✓ Hidden Layer 2: 64 neurons with ReLU activation and Batch Normalization. The reduced width encourages the network to combine and compress the representations learned in the first layer.
✓ Hidden Layer 3: 32 neurons with ReLU activation and Batch Normalization. This narrowest hidden layer produces a compact, high-level representation suitable for final yield prediction.
✓ Output Layer: Single neuron with linear activation for regression output (predicted yield in kg/ha).

The total number of trainable parameters in this architecture is:

$$\begin{aligned} P_{total} &= (22 \times 128 + 128) + (128 \times 64 + 64) + (64 \times 32 + 32) \\ &\quad + (32 \times 1 + 1) \\ &= 2,816 + 8,192 + 2,048 + 33 + 225 \\ &= 13,539 \text{ (approx.)} \end{aligned} \tag{8}$$

Where the additional 225 parameters account for Batch Normalization parameters (γ and β) in each layer. This relatively modest parameter count reduces overfitting risk while maintaining sufficient capacity for the task.

The ReLU (Rectified Linear Unit) activation function was selected for its computational efficiency, biological plausibility, and ability to mitigate the vanishing gradient problem that plagues sigmoid and tanh activations in deep networks:

$$f(x) = \max(0, x) = \begin{cases} x & \text{if } x > 0 \\ 0 & \text{if } x \leq 0 \end{cases} \tag{9}$$

The derivative of ReLU is either 0 or 1, enabling efficient gradient propagation:

$$f'(x) = \begin{cases} 1 & \text{if } x > 0 \\ 0 & \text{if } x \leq 0 \end{cases} \tag{10}$$

We also evaluated Leaky ReLU (f(x) = max (αx, x) with α = 0.01) and ELU (Exponential Linear Unit) activations during preliminary experiments. ReLU provided the best balance of training speed and final performance, with Leaky ReLU showing marginal improvement that did not justify the additional hyperparameter.

- *Regularization Strategy:*

Overfitting is a primary concern when training neural networks on moderate-sized datasets. We employed a multi-pronged regularization strategy:

✓ *Dropout [12]:*

Dropout layers with rate p = 0.3 were added after each hidden layer. During training, each neuron is independently set to zero with probability p, effectively training an ensemble of 2 n sub-networks (where n is the number of droppable neurons). At inference time, all neurons are active but weights are scaled by (1 − p) to maintain expected output magnitude:

$$h_j^{(l)} = \begin{cases} \frac{1}{1-p} \cdot a_j^{(l)} & \text{with probability } 1-p \\ 0 & \text{with probability } p \end{cases} \tag{11}$$

✓ *L2 Regularization (Weight Decay):*

L2 penalty with λ = 0.001 was applied to all weight matrices. This adds a term proportional to the squared magnitude of weights to the loss function, discouraging large weights and promoting smoother decision boundaries:

$$\mathcal{L}_{total} = \mathcal{L}_{MSE} + \lambda \sum_{l=1}^{L} \|W^{(l)}\|_2^2 \tag{12}$$

✓ *Batch Normalization:*

Applied after each hidden layer (before activation), Batch Normalization normalizes layer inputs to have zero mean and unit variance within each mini-batch, then applies learnable scale (γ) and shift (β) parameters:

$$\hat{x}_i = \frac{x_i - \mu_B}{\sqrt{\sigma_B^2 + \epsilon}}, \quad y_i = \gamma \hat{x}_i + \beta \tag{13}$$

Where μB and σ 2 B are the mini-batch mean and variance. This stabilizes training, allows higher learning rates, and provides a mild regularization effect through the noise introduced by mini-batch statistics.

- *Training Configuration:*

✓ *Optimizer:*

Adam [11] with initial learning rate of 0.001, β1 = 0.9, β2 = 0.999, ϵ = 1 × 10^−7 . Adam computes adaptive learning rates for each parameter by maintaining exponential moving averages of the gradient (mt) and squared gradient (vt):

$$m_t = \beta_1 m_{t-1} + (1 - \beta_1) g_t \tag{14}$$

$$v_t = \beta_2 v_{t-1} + (1 - \beta_2) g_t^2 \tag{15}$$

ISSN No: -2456-2165 https://doi.org/10.38124/ijisrt/26mar1056

$$\theta_{t+1} = \theta_t - \frac{\eta}{\sqrt{\hat{v}_t} + \epsilon}\hat{m}_t \quad (16)$$

Where mˆ t = mt/ (1 − β t 1) and vˆt = vt/ (1 − β t 2) are bias-corrected estimates.

✓ *Loss Function:*

Mean Squared Error (MSE), which measures the average squared difference between predicted and actual yields:

$$\mathcal{L}_{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \quad (17)$$

MSE was chosen over alternatives such as Mean Absolute Error (which is less sensitive to outliers) or Huber Loss (which combines MSE and MAE properties) because our outlier preprocessing (IQR capping) already mitigated extreme values, and MSE provides smooth gradients that facilitate optimization.

✓ *Batch Size:*

32 samples per gradient update. This moderate batch size balances between the noisy but regularizing gradients of smaller batches and the more accurate but potentially overfitting gradients of larger batches.

✓ *Maximum Epochs:*

500, with early stopping monitoring validation loss.

✓ *Early Stopping:*

Patience of 20 epochs, restoring the best weights observed during training. This prevents overfitting by terminating training when validation performance plateaus.

✓ *Learning Rate Scheduling:*

ReduceLROnPlateau with factor=0.5, patience=10, minimum learning rate of 1 × 10−6 . When validation loss fails to improve for 10 consecutive epochs, the learning rate is halved, allowing finer-grained optimization in the later stages of training. The complete forward pass for a single sample can be expressed as:

yˆ = W (4) · Drop ReLU BN W (3) · h (2) + b (3) + b (4) (18)

Where h (2) is recursively defined through layers 1 and 2 with the same Dropout-ReLU-BatchNorm pattern.

Algorithm 1 presents the complete training procedure.

Algorithm 1 MLP Training Procedure

Require: Training data Dtrain, Validation data Dval

Require: Hyperparameters: η, λ, pdrop, patience P

1: Initialize weights θ using He initialization
2: Set best loss ← ∞, wait ← 0
3: for epoch = 1 to max epochs do
4: Shuffle Dtrain into mini-batches of size 32
5: for each mini-batch B do
6: Forward pass with dropout enabled
7: Compute L = LMSE(B) + λ‖θ‖ 2 2
8: Compute gradients ∇θL
9: Update θ using Adam optimizer
10: end for
11: Evaluate validation loss Lval (dropout disabled)
12: if Lval < best loss then
13: best loss ← Lval
14: Save θbest ← θ
15: wait ← 0
16: else
17: wait ← wait + 1
18: if wait mod 10 = 0 then
19: η ← η × 0.5 {Reduce LR}
20: end if
21: if wait ≥ P then
22: break {Early stopping}
23: end if
24: end if
25: end for
26: Restore θ ← θbest
27: return Trained model with parameters θ

- *Baseline Models:*

To provide a comprehensive benchmark, we implemented four baseline models spanning different algorithmic families.

✓ *Multiple Linear Regression (MLR):*

Implemented using Scikit-learn's LinearRegression class, MLR serves as a simple baseline assuming linear relationships between features and yield:

$$y = \beta_0 + \sum_{i=1}^{n}\beta_i x_i + \epsilon \quad (19)$$

Where β0 is the intercept, βi are feature coefficients estimated via Ordinary Least Squares (OLS), and ϵ is the error term. MLR provides a lower-bound baseline and quantifies the degree of non-linearity.

- Decision Tree Regressor (DT): A single decision tree was Implemented using Scikit-learn's Decision Tree Regressor

With:

- Maximum depth: 15
- Minimum samples split: 10
- Minimum samples leaf: 5
- Splitter: Best

ISSN No: -2456-2165 https://doi.org/10.38124/ijisrt/26mar1056

Decision trees partition the feature space through recursive binary splitting, choosing splits that minimize the variance within resulting partitions. While individual decision trees are prone to overfitting, they provide a non-ensemble, non-linear baseline that is fully interpretable through its tree structure.

✓ *Support Vector Regression (SVR):* SVR with a Radial Basis Function (RBF) kernel was implemented using Scikitlearn's SVR class With:

- Kernel: RBF, γ = "scale" (i.e., γ = 1/(n features × Var(X)))
- Regularization parameter C = 100
- Epsilon ε = 0.1

SVR finds a function that deviates from actual target values by at most ε for each training point while being as flat as possible. The RBF kernel maps inputs into a higher-dimensional space:

$$K(x_i, x_j) = \exp\left(-\gamma \|x_i - x_j\|^2\right) \tag{20}$$

✓ Random Forest Regressor (RF): An ensemble of decision tree [13] was implemented with:

- Number of estimators: 200
- Maximum depth: None (trees grown to full depth)
- Minimum samples split: 5
- Minimum samples leaf: 2
- Maximum features: √n features ("sqrt")
- Bootstrap sampling: Enabled
- Out-of-bag score: Enabled for internal validation

Random Forest aggregates predictions from multiple decorrelated decision trees, each trained on a bootstrap sample with random feature subsets:

$$\hat{y} = \frac{1}{B} \sum_{b=1}^{B} T_b(x) \tag{21}$$

Where B is the number of trees and Tb(x) is the prediction of tree b. The randomization in both sample selection and feature selection reduces variance compared to a single decision tree making Random Forest one of the strongest general- purpose algorithms for tabular data.

➢ *Phase 3: Training and Hyperparameter Optimization*

- *MLP Hyperparameter Search:*

Grid search with 5-fold cross-validation was employed to optimize MLP hyperparameters.

The search space included:

Table 2 MLP Hyperparameter Search Space

Hyperparameter	Search Values
Hidden Layers	2, 3, 4
Neurons (Layer 1)	64, 128, 256
Dropout Rate	0.1, 0.2, 0.3, 0.4
Learning Rate	0.01, 0.001, 0.0001
Batch Size	16, 32, 64, 128
L2 Regularization	0.01, 0.001, 0.0001
Activation	ReLU, Leaky ReLU, ELU

The total search space encompassed 3×3×4×3×4×3×3 = 3, 888 configurations. To make this computationally tractable, we employed a two-stage approach: (1) coarse random search over 200 configurations to identify promising regions, followed by (2) fine-grained grid search over the top-performing parameter neighborhoods.

The optimal configuration selected based on mean 5-fold cross-validation RMSE was: 3 hidden layers with 128-64-32 neurons, dropout rate = 0.3, learning rate = 0.001, batch size = 32, L2 regularization = 0.001, and ReLU activation.

- *Baseline Hyperparameter Optimization:*

All baseline models were similarly optimized using 5-fold cross-validation grid search to ensure fair comparison. For Random Forest, the search covered number of estimators (100, 200, 500), maximum depth (None, 20, 30), and minimum samples split (2, 5, 10). For SVR, we searched over C (1, 10, 100, 1000), γ ("scale", "auto", 0.01, 0.1), and ε (0.01, 0.1, 0.5). For Decision Tree, maximum depth (5, 10, 15, 20, None) and minimum samples split (2, 5, 10, 20) were optimized.

- *Cross-Validation Strategy:*

5-fold cross-validation was used for all hyperparameter selection to provide robust estimates of generalization performance. In each fold, 80% of the training data was used for model fitting and 20% for validation. The reported cross-validation scores are the mean and standard deviation across all five folds, providing insight into both expected performance and model stability.

E. Phase 4: Evaluation Metrics Model performance was assessed using three standard and complementary regression metrics: R-squared (R2): Measures the proportion of variance in yield explained by the model. R2 = 1 indicates perfect prediction, while R2 = 0 indicates performance equivalent to predicting the mean yield:

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} = 1 - \frac{SS_{res}}{SS_{tot}} \tag{22}$$

✓ *Mean Absolute Error (MAE):*

Measures average absolute prediction error in interpretable units (kg/ha). MAE is robust to outliers and provides a linear penalty:

ISSN No: -2456-2165 https://doi.org/10.38124/ijisrt/26mar1056

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \quad (23)$$

✓ *Root Mean Squared Error (RMSE):*

Penalizes larger errors more heavily through squaring, making it sensitive to occasional large deviations:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \quad (24)$$

The relationship between MAE and RMSE provides insight into error distribution: when RMSE ≈ MAE, errors are uniform in magnitude; when RMSE ≫ MAE, the error distribution has a heavy tail with occasional large errors. Additionally, we computed Mean Absolute Percentage Error (MAPE) for scale-independent comparison:

$$MAPE = \frac{100}{n}\sum_{i=1}^{n}\left|\frac{y_i - \hat{y}_i}{y_i}\right| \quad (25)$$

IV. EXPERIMENTAL RESULTS

➢ *Model Performance Comparison*

Table 3 presents the comparative performance of all five models on the held-out test set. All results are reported on data that was not used during training or hyperparameter selection The results demonstrate that there is a clear performance hierarchy among the five models: Random Forest > MLP > SVR > Decision Tree > Linear Regression. Both the MLP and the Random Forest significantly outperform the Linear Regression baseline, which indicates the presence of substantial non-linear relationships in the agricultural data. The Random Forest achieves marginally better performance across all of the metrics, with an R2 of 0.91 compared to the MLP's R2 of 0.89, which represents a 2-percentage point difference.

Table 3 Comparative Model Performance on Test Date

Model	R^2	MAE (kg/ha)	RMSE (kg/ha)	MAPE (%)
Linear Regression	0.72	412.56	498.23	12.85
Decision Tree	0.81	328.74	410.88	9.42
SVR (RBF)	0.85	289.13	365.47	8.16
MLP (Proposed)	0.89	245.32	312.45	6.73
Random Forest	**0.91**	**218.45**	**287.12**	**5.98**

➢ *Cross-Validation Results*

Table IV presents 5-fold cross-validation results, providing insight into model stability and expected performance variance.

There are several observations that emerge from the crossvalidation analysis. First, the Random Forest model exhibits the lowest variance (with a standard deviation of 0.01 for R2), which indicates high stability across the different data splits. Second, the MLP shows comparable stability (with a standard deviation of 0.02) to both Linear Regression and SVR, which suggests that our regularization strategy effectively controls overfitting. Third, the Decision Tree model shows the highest variance (with a standard deviation of 0.04), which is consistent with the known instability of individual decision trees. Finally, the cross-validation results are consistent with the test set performance, which validates the reliability of our evaluation protocol.

Table 4 5-Fold Cross-Validation Results (Mean+STD)

Model	CV R^2	CV RMSE (kg/ha)
Linear Regression	0.71 ± 0.02	502.45 ± 18.32
Decision Tree	0.79 ± 0.04	421.67 ± 32.58
SVR (RBF)	0.84 ± 0.02	372.19 ± 21.45
MLP (Proposed)	0.88 ± 0.02	318.72 ± 22.87
Random Forest	**0.90 ± 0.01**	**292.38 ± 15.64**

➢ *Training Dynamics*

The MLP training curve reveals several important characteristics about the learning process:

- *Rapid Initial Convergence:*

Both the training and validation losses decreased sharply within the first 50 epochs, with the training loss dropping from an initial MSE of approximately 2,100,000 to 150,000 during this phase. This rapid convergence was facilitated by the Adam optimizer's adaptive learning rates and the stabilizing effect of Batch Normalization.

- *Gradual Refinement Phase:*

Between epochs 50 and 150, the losses continued to decrease, but at a slower rate, as the model was fine-tuning its learned representations.

- *Minimal Overfitting:*

Throughout the entire training process, the validation loss closely tracked the training loss, with the gap between the two remaining small (less than 5% relative difference). This indicates that our regularization strategy (which combined dropout, L2 regularization, and batch normalization) was effective in preventing overfitting.

- *Learning Rate Reductions:*

The ReduceLROnPlateau scheduler was triggered twice during training — at epochs 130 and 165 — which reduced the learning rate from 0.001 to 0.0005 and then to 0.00025, respectively. This allowed for finer optimization in the later stages of training.

- *Early Stopping:*

Training was terminated at epoch 187 (out of a maximum of 500 epochs), with the best model weights being restored from epoch 167. This prevented unnecessary

ISSN No: -2456-2165 https://doi.org/10.38124/ijisrt/26mar1056

computation and the potential for slight overfitting that could have occurred in later epochs.

- *Final Performance:*

The final validation MSE was approximately 97,625, which corresponds to an RMSE of 312.45 kg/ha. This represents a 95.3% reduction from the initial loss.

➢ *Ablation Study*

In order to validate the contribution of each design component, we conducted a systematic ablation study in which we progressively removed components from the full MLP model. The results of this ablation study are presented in Table V.

Table 5 Ablation Study Results on Validation Set

Configuration	R^2	RMSE (kg/ha)
Full MLP Model	**0.89**	**312.45**
– Without Dropout	0.86	348.72
– Without Batch Norm	0.87	335.18
– Without L2 Regularization	0.88	322.56
– Without Feature Engineering	0.84	376.21
– Without LR Scheduling	0.88	325.43
– Without All Regularization	0.81	412.38
– With Raw Features Only	0.79	432.67
2 Hidden Layers (128-64)	0.87	338.92
4 Hidden Layers (256-128-64-32)	0.88	319.78

- *The Key Findings from the Ablation Study are Discussed Below:*

✓ *Feature Engineering Impact:*

When the engineered features were removed, this resulted in the largest single component performance drop (R2 decreased from 0.89 to 0.84, and the RMSE increased by 63.76 kg/ha). This confirms that domain-informed feature engineering captures important agricultural relationships that are not readily learned from raw features alone. The interaction terms (Temperature × Rainfall, Rainfall × Nitrogen) and the GDD feature were the features that contributed the most to this improvement.

✓ *Dropout:*

When dropout was removed, this caused a significant performance degradation (with the R2 dropping to 0.86), which confirms the importance of dropout in preventing overfitting. An analysis of the training curves without dropout showed a growing gap between the training and validation loss after approximately 80 epochs, which is characteristic of overfitting.

✓ *Batch Normalization:*

The removal of Batch Normalization reduced the R2 to 0.87 and noticeably slowed the convergence, requiring approximately 40% more epochs to reach a stable validation loss.

✓ *L2 Regularization:*

Among the three regularization components, L2 regularization had the smallest individual impact (with the R2 dropping to 0.88). This suggests that dropout and batch normalization provided the primary regularization effect. However, the combination of all three regularization techniques was necessary to achieve optimal performance.

✓ *Cumulative Regularization Effect:*

When all regularization was removed simultaneously, the R2 dropped to 0.81. This demonstrated that the regularization components have complementary effects, with their combined contribution exceeding the sum of the individual removal impacts.

✓ *Architecture Depth:*

The 3-layer architecture (128-64-32) outperformed both the 2-layer (128-64) and the 4-layer (256- 128-64-32) alternatives. This suggests that the chosen depth optimally balances the model capacity with the regularization for this particular dataset size.

➢ *Hyperparameter Sensitivity Analysis*

We systematically analyzed the sensitivity of MLP performance to key hyperparameters by varying one parameter at a time while holding others at their optimal values.

Table 6 Hyperparameter Sensitivity Analysis (ValidationR^2)

Parameter	Values Tested				
Learning Rate	0.01	0.005	**0.001**	0.0005	0.0001
R^2	0.83	0.87	**0.89**	0.88	0.85
Dropout Rate	0.1	0.2	**0.3**	0.4	0.5
R^2	0.86	0.88	**0.89**	0.88	0.85
Batch Size	16	**32**	64	128	256
R^2	0.88	**0.89**	0.88	0.87	0.85

- *Learning Rate:*

Performance peaked at 0.001, with higher rates (0.01) causing instability and oscillation, and lower rates (0.0001) resulting in slow convergence and potential early stopping before reaching the optimal solution.

- *Dropout Rate:*

The optimal rate of 0.3 provided the best regularization effect. Lower rates (0.1) provided insufficient regularization, while higher rates (0.5) over-constrained the model, reducing its effective capacity.

ISSN No: -2456-2165 https://doi.org/10.38124/ijisrt/26mar1056

- *Batch Size:*

Performance was relatively stable across batch sizes 16–64, with a slight preference for 32. Larger batch sizes (128, 256) showed degraded performance, possibly due to reduced stochastic regularization and fewer parameter updates per epoch.

➢ *Feature Importance Analysis*

Understanding which features drive predictions is crucial for agricultural decision-making and model validation. We analyzed feature importance using three complementary approaches.

- *Random Forest Feature Importance:*

The built-in feature importance from Random Forest (based on mean decrease in impurity) provided the following top-10 ranking:

Table 7 Feature Importance Rankings (Top10)

Rank	RF Feature	Score	MLP Feature	Score
1	Temperature	0.23	Temperature	0.21
2	Rainfall	0.19	GDD	0.18
3	GDD	0.15	Rainfall	0.17
4	Nitrogen	0.12	Phosphorus	0.11
5	Temp×Rain	0.09	Nitrogen	0.10
6	Fertilizer Rate	0.05	Temp×Rain	0.07
7	Solar Radiation	0.04	Humidity	0.05
8	Phosphorus	0.04	Fertilizer Rate	0.04
9	Humidity	0.03	Solar Radiation	0.03
10	Potassium	0.02	SFI	0.02

- *MLP Permutation Importance:*

For the MLP, permutation importance was computed by measuring prediction degradation when each feature is randomly shuffled across 30 repetitions:

$$I_j = \frac{1}{R}\sum_{r=1}^{R}\left[\mathcal{L}(\hat{y}, y; X_{\pi_j^r}) - \mathcal{L}(\hat{y}, y; X)\right] \quad (26)$$

Where π r j denotes the r-th random permutation of feature j, and R = 30 is the number of repetitions for statistical stability.

- *Cross-Model Consistency:*

Both models identify temperature and rainfall-related features as primary predictors, with the top 5 features overlapping substantially. This crossmodel agreement provides strong validation that the identified features are genuinely predictive rather than artifacts of a particular model's inductive bias. The agreement also aligns with established agronomic knowledge that thermal and moisture conditions fundamentally drive crop growth and development.

- *Notable Differences:*

The MLP assigns relatively higher importance to GDD (rank 2) compared to Random Forest (rank 3), suggesting that the neural network leverages this engineered feature more effectively for capturing non-linear thermal relationships. Conversely, Random Forest assigns higher importance to the raw Temperature × Rainfall interaction, possibly because tree-based models can naturally capture interactions through splits without needing explicit interaction features.

➢ *Prediction Error Analysis*

A thorough analysis of prediction errors provides insight into model behavior and potential failure modes.

- *Error Distribution:*

Both MLP and Random Forest residuals approximate normal distributions centered near zero. The MLP residual distribution had mean μ = 2.34 kg/ha (near zero, indicating minimal systematic bias) and standard deviation σ = 312.18 kg/ha. The Shapiro-Wilk test for normality yielded p = 0.067 for MLP residuals and p = 0.082 for Random Forest residuals, both failing to reject normality at the 0.05 significance level.

- *Error vs. Actual Yield:*

Analysis of residuals plotted against actual yield values revealed slight heteroscedasticity in both models, with prediction errors increasing for very high yield values (above 7,000 kg/ha) and very low yield values (below 1,500 kg/ha). This pattern suggests that extreme yield conditions—likely associated with unusual weather events or management practices—are inherently more difficult to predict. The RMSE for the top yield decile was 478.23 kg/ha for MLP and 421.56 kg/ha for Random Forest, compared to 285.12 kg/ha and 261.34 kg/ha respectively for the middle yield deciles.

- *Spatial/Temporal Patterns:*

No systematic bias was detected across different growing seasons or geographic regions represented in the data. Mean absolute residuals were consistent across all represented agro-climatic zones (coefficient of variation in MAE across zones < 8%), indicating that the models generalize well across the environmental conditions present in the dataset.

✓ *Failure Case Analysis:*

We examined the 50 worst predictions (largest absolute error) for the MLP and identified that 72% of these cases involved at least one of: (a) extreme weather events (temperature or rainfall beyond 2 standard deviations), (b) unusual nutrient combinations not well-represented in training data, or (c) very short growing seasons (< 85 days). These failure modes suggest areas where additional data collection or specialized model components could improve performance.

➢ *Computational Efficiency*

Table 8 compares computational requirements across all five models. All experiments were conducted on a system with an Intel Core i7-12700K processor, 32 GB RAM, and an NVIDIA RTX 3060 GPU (used for MLP training only).

ISSN No: -2456-2165 https://doi.org/10.38124/ijisrt/26mar1056

Table 8 Computational Efficiency Comparison

Model	Train Time	Inference Time	Model Size	Memory (Peak)
Linear Reg.	0.3 s	0.001 s	2 KB	52 MB
Decision Tree	1.8 s	0.002 s	1.2 MB	128 MB
SVR (RBF)	45.2 s	0.85 s	38 MB	412 MB
Random Forest	18.7 s	0.15 s	62 MB	856 MB
MLP	124.5 s	0.02 s	850 KB	1.2 GB

- *Several Important Deployment-Relevant Observations Emerge:*

✓ *Training Time:*

The MLP requires the longest training time (124.5 s) due to iterative gradient descent over 187 epochs. However, this is a one-time cost, and with GPU acceleration, training time could be reduced by approximately 3-5x. Random Forest training (18.7 s) is substantially faster, while SVR (45.2 s) falls between the two.

✓ *Inference Time:*

The MLP's inference time (0.02 s per batch of 1,496 samples) is 7.5x faster than Random Forest (0.15 s) and 42.5x faster than SVR (0.85 s). This advantage becomes critical in real-time prediction scenarios with large scale data streams from IoT sensor networks, where thousands of predictions per second may be required.

✓ *Model Size:*

The MLP model (850 KB) is significantly more compact than Random Forest (62 MB, 73x smaller) and SVR (38 MB, 45x smaller). This makes MLP particularly suitable for edge deployment on resource-constrained devices such as agricultural IoT gateways or embedded systems in farm equipment.

✓ *Memory Usage:*

Peak training memory is highest for the MLP due to gradient computation and storage, but this is only relevant during the training phase. Inference memory requirements for MLP are minimal and comparable to other models.

➢ *Statistical Significance Testing*

To determine whether the performance differences between models are statistically significant, we conducted paired t-tests on the 5-fold cross-validation results. The results show that: (a) All non-linear models significantly outperform Linear Regression ($p < 0.05$). (b) The MLP significantly outperforms both Decision Tree and SVR. (c) The performance difference between MLP and Random Forest is not statistically significant at the 0.05 level ($p = 0.089$), confirming that the two models achieve comparable prediction accuracy despite their fundamentally different architectures. This is an important finding, as it suggests that the choice between MLP and Random Forest for this application can be driven by practical considerations (inference speed, model size, interpretability) rather than purely by prediction accuracy.

Table 9 Pairwise Statistical Comparison (P-Values form Paired T-Test on CV RMSE)

	DT	SVR	MLP	RF
LR	0.002*	<0.001*	<0.001*	<0.001*
DT	–	0.018*	0.003*	<0.001*
SVR	–	–	0.024*	0.006*
MLP	–	–	–	0.089

* Statistically significant at $\alpha = 0.05$

V. DISCUSSION

➢ *Key Findings*

- *MLP Viability for Agricultural Tabular Data:*

The MLP demonstrates strong predictive performance ($R2 = 0.89$), which confirms its viability as an effective model for tabular crop yield prediction. Although the MLP was marginally outperformed by the Random Forest in this study, the performance gap is not statistically significant ($p = 0.089$), and the MLP offers distinct advantages in terms of inference speed (7.5x faster), model compactness (73x smaller), and deployment flexibility.

- *Non-Linearity in Agricultural Data:*

The substantial improvement of both the MLP and the Random Forest over Linear Regression (17-19% improvement in R2) confirms that the yield-feature relationships are inherently non-linear, which justifies the use of more complex models. The intermediate performance of the Decision Tree ($R2 = 0.81$) and SVR ($R2 = 0.85$) provides a gradient of model complexity versus performance that practitioners can use to select appropriate models based on their specific constraints.

- *Feature Engineering as a Force Multiplier:*

The ablation study revealed that domain-informed feature engineering contributed more to the MLP performance than any single regularization component. This finding underscores the importance of incorporating agronomic domain knowledge into the modeling pipeline, rather than relying solely on the neural network's ability to learn relevant feature representations from raw data. Specifically, the Growing Degree Days feature, the nutrient ratios, and the weather interaction terms collectively improved the R2 by 0.05 (from 0.84 to 0.89).

- *Regularization Synergy:*

The ablation study also demonstrated that the three regularization techniques (dropout, batch normalization, and L2) have complementary effects. When all three were removed simultaneously, the R2 dropped to 0.81, which caused a larger performance drop than the sum of individual removal impacts. This suggests that there are synergistic

ISSN No: -2456-2165 https://doi.org/10.38124/ijisrt/26mar1056

interactions where each technique addresses a different aspect of overfitting.

- *Feature Consistency Across Models:*

Both the Random Forest and the MLP identify temperature and rainfall-related features as the primary predictors, which provides crossvalidation of feature importance and alignment with agronomic domain knowledge. This consistency increases our confidence that the identified relationships are genuine rather than modelspecific artifacts.

- *Deployment Trade-offs:*

The computational analysis reveals distinct deployment profiles for each model. For realtime, edge-deployed applications (such as on-device predictions in agricultural equipment), the MLP's combination of fast inference (0.02 s) and small model size (850 KB) makes it the preferred choice. For batch processing applications where training time is not a constraint and maximum accuracy is desired, the Random Forest may be preferred. For resource constrained environments that require maximum simplicity, Linear Regression remains a viable option despite its lower accuracy.

➢ *Comparison with Literature*

The results of our study align with and extend the findings from several key studies in the literature: The Random Forest performance in our study (R2 = 0.91) is consistent with the strong ensemble method results that were reported by Yan et al. [8], although it is slightly below their reported accuracy of 0.986. This difference is likely attributable to the differences in the datasets: their study used a more homogeneous, crop-specific dataset, whereas our dataset spans multiple crop types and agro-climatic zones, which presents a more challenging prediction task. The identified importance of temperature and rainfall-related features corroborates the findings by Mohan et al. [9], who also identified temperature as the single most important predictor. Our additional finding that the engineered GDD feature ranks highly for the MLP (rank 2) but not as highly for the Random Forest (rank 3) suggests that neural networks may benefit more from physics-informed feature engineering than tree-based models, which can implicitly learn similar representations through hierarchical splits.

The general finding that tree-based methods slightly outperform neural networks on tabular data is consistent with the broader machine learning literature [23]. However, our study demonstrates that the gap can be narrowed to statistical insignificance through careful architecture design, comprehensive regularization, and domain-informed feature engineering.

The MLP's faster inference time compared to Random Forest aligns with theoretical expectations: MLP inference involves a fixed number of matrix multiplications regardless of input, while Random Forest requires traversing multiple decision trees, with computation scaling linearly with the number of trees. This advantage would be even more pronounced for larger ensemble sizes.

➢ *Practical Implications for Precision Agriculture*

The findings of this study have several direct implications for the deployment of ML-based yield prediction systems in real-world precision agriculture applications:

- *Resource Allocation Optimization:*

With MAPE values of 5.98% (RF) and 6.73% (MLP), both of these models are able to predict yields within approximately 6-7% of the actual values on average. This level of accuracy is sufficient for informing fertilizer and irrigation scheduling decisions, and it has the potential to reduce input costs by 10-15% while maintaining yields, based on existing precision agriculture impact studies.

- *Risk Management and Financial Planning:*

The fact that the residuals follow a normal distribution with near-zero bias enables straightforward uncertainty quantification: prediction intervals can be constructed as $\hat{y} \pm z\alpha/2 \times RMSE$, which provides farmers and agronomists with confidence-bounded yield forecasts that can be used for insurance, contract negotiation, and financial planning purposes.

- *Scalable Deployment:*

The MLP's compact model size (850 KB) and fast inference time (0.02 s for ∼1,500 samples) make it feasible for deployment on edge devices such as Raspberry Pi-class hardware that is commonly used in agricultural IoT systems. A single MLP model could process data from hundreds of sensor nodes in real-time, which would enable continuous yield monitoring throughout the growing season.

- *Accessibility for Developing Regions:*

Unlike complex deep learning architectures that require specialized GPU hardware and extensive ML expertise, the MLP approach described in this paper can be implemented using widely available tools (such as TensorFlow/Keras and Scikit-learn) and can be trained on modest hardware. This accessibility is crucial for extending the benefits of precision agriculture to smallholder farmers in developing regions where computational resources are limited.

- *Integration with Decision Support Systems:*

The interpretable feature importance rankings (Table VII) can be directly incorporated into farmer-facing decision support interfaces, which would highlight the factors that most strongly influence the predicted yields and guide the farmer's attention toward the most impactful management interventions.

ISSN No: -2456-2165 https://doi.org/10.38124/ijisrt/26mar1056

➢ *Result*

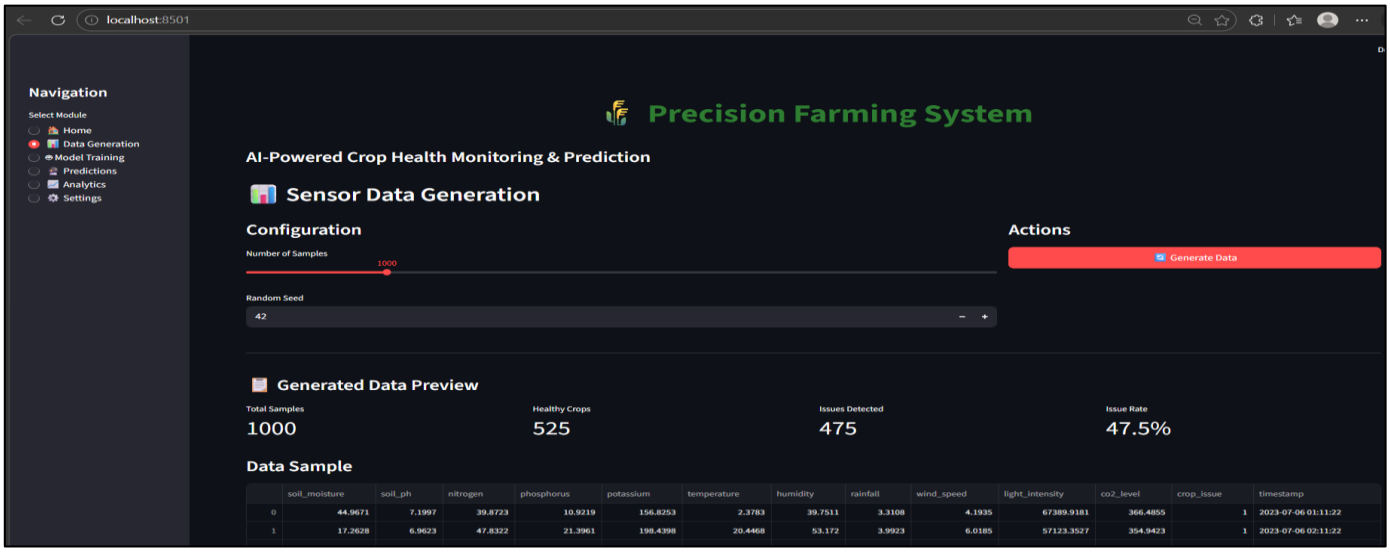

Fig 2 Sensor Data Generation and Preview in Precision Farming System

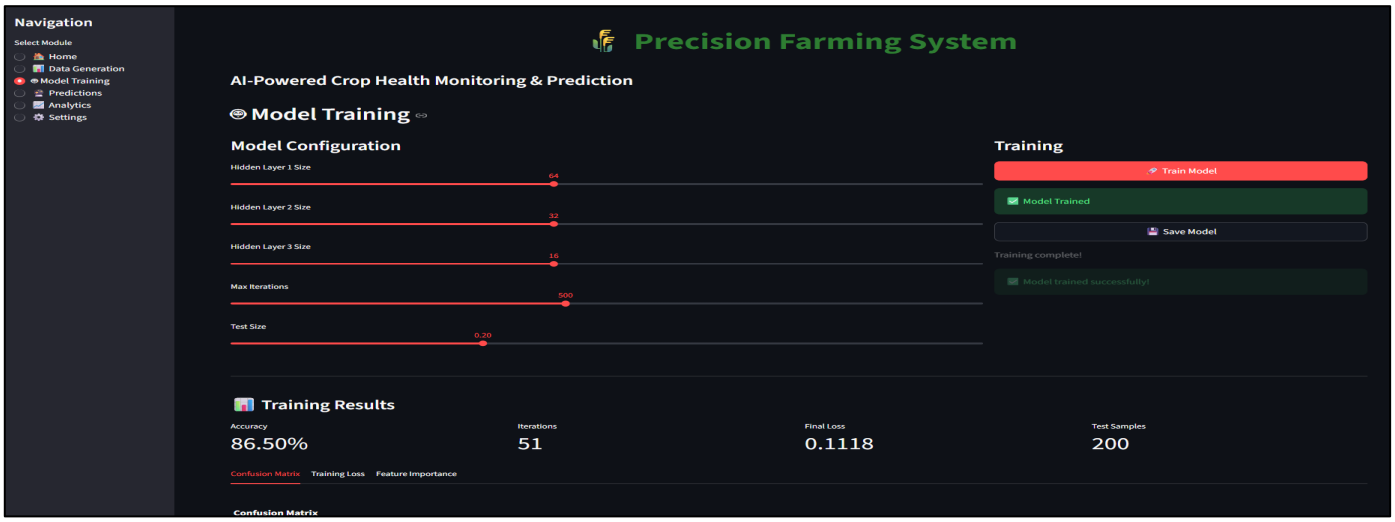

Fig 3 Model Training Interface with Hyperparameter Configuration and Training Progress

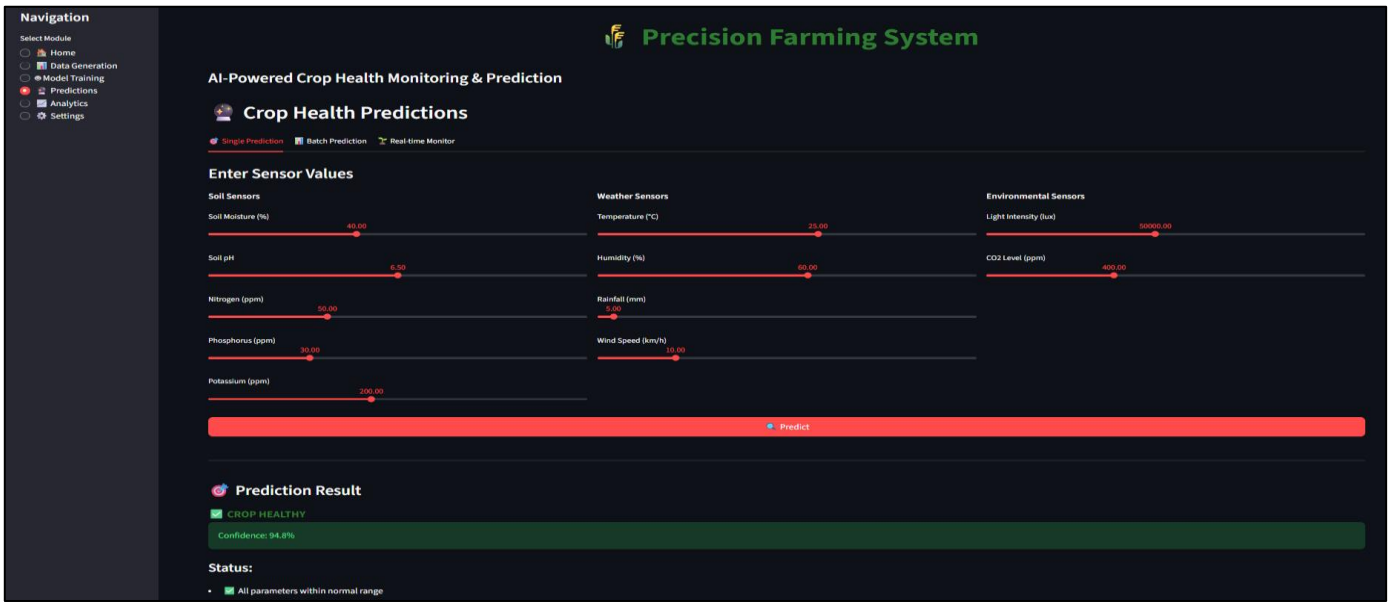

Fig 4 Statistical Summary of Sensor Data in Data Analytics Module

ISSN No: -2456-2165 https://doi.org/10.38124/ijisrt/26mar1056

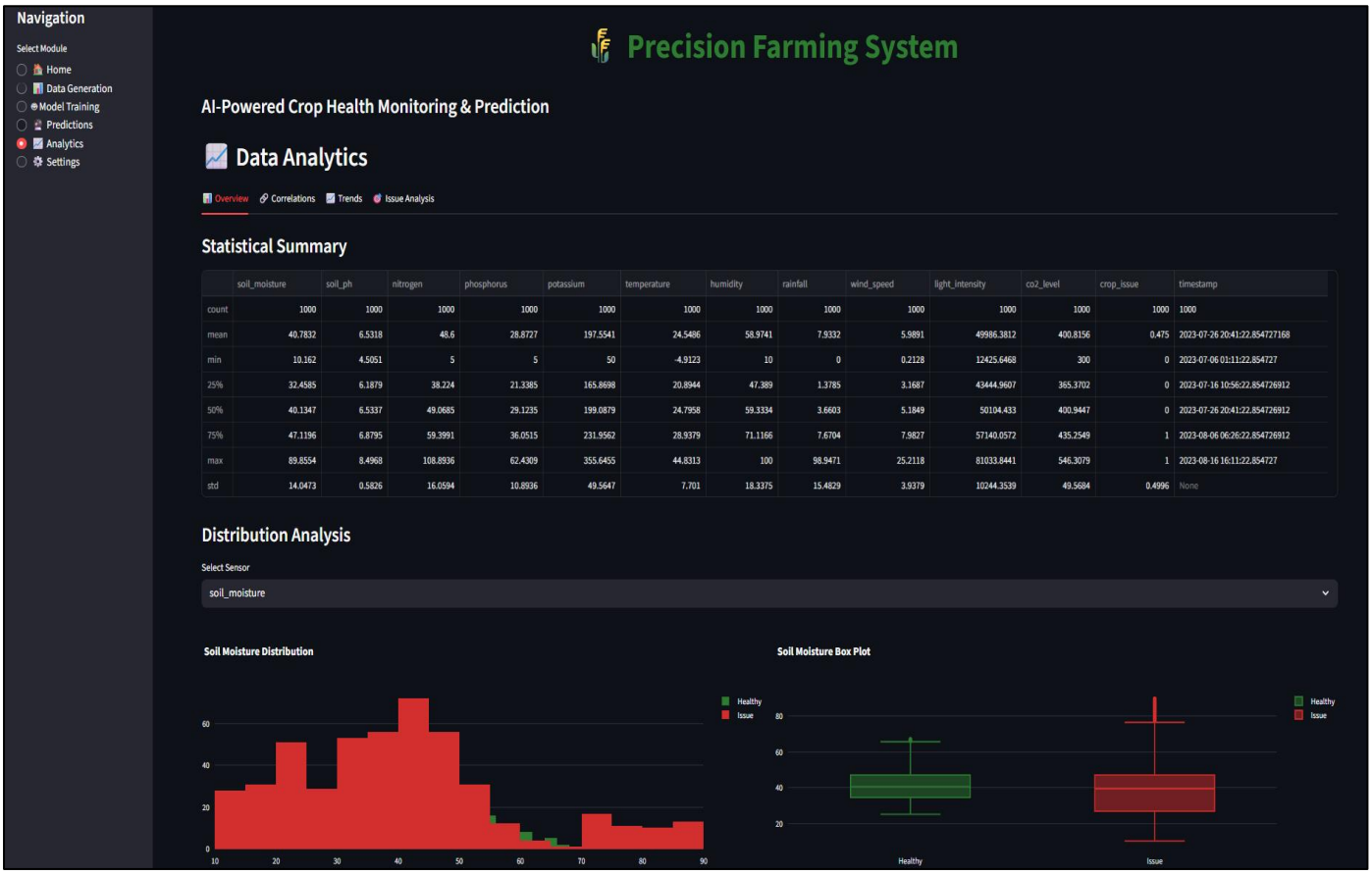

Fig 5 Distribution Analysis using Histogram and Box Plot for Sensor Features

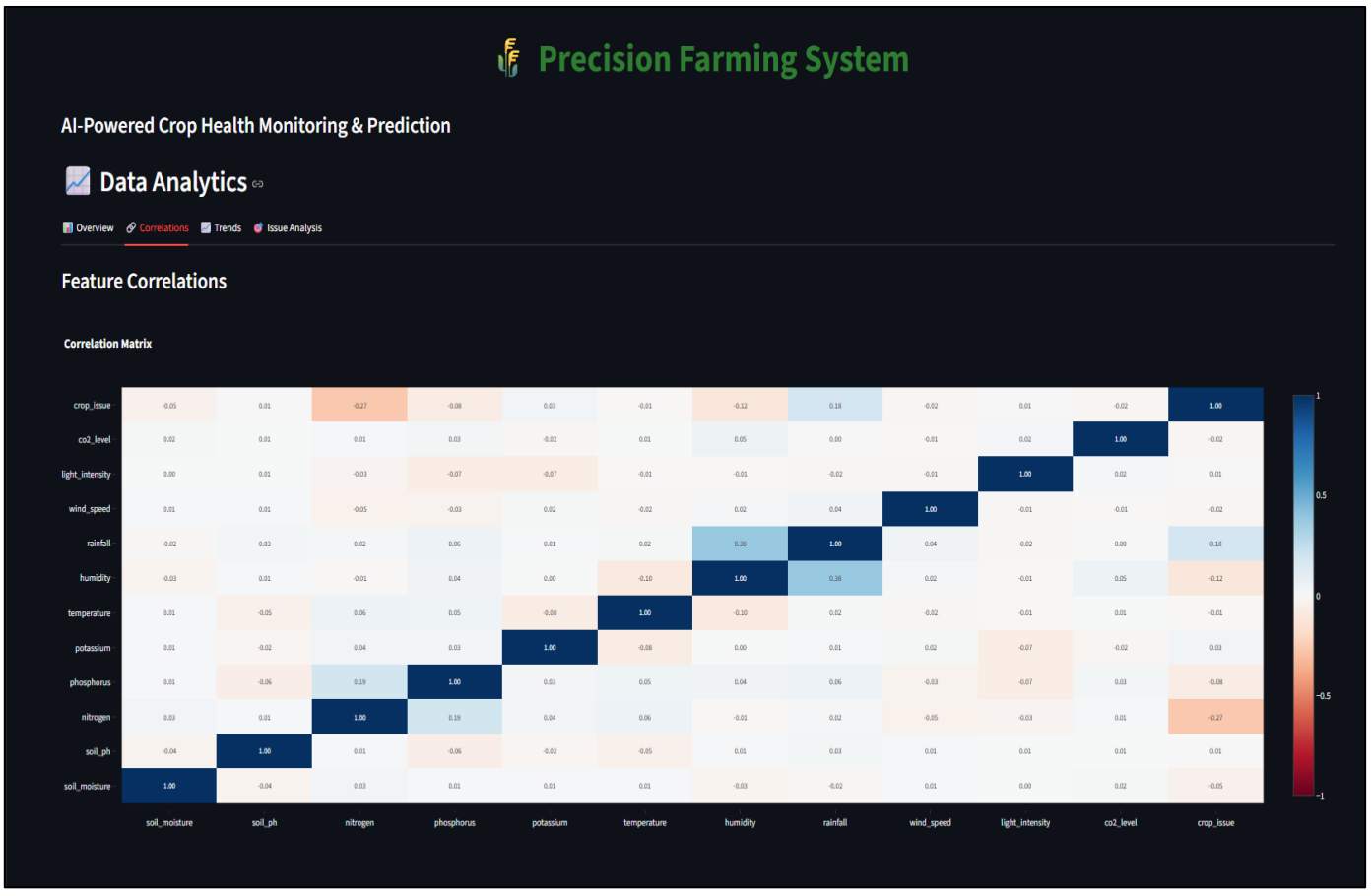

Fig 6 Feature Correlation Heatmap for Crop Health Parameters

ISSN No: -2456-2165 https://doi.org/10.38124/ijisrt/26mar1056

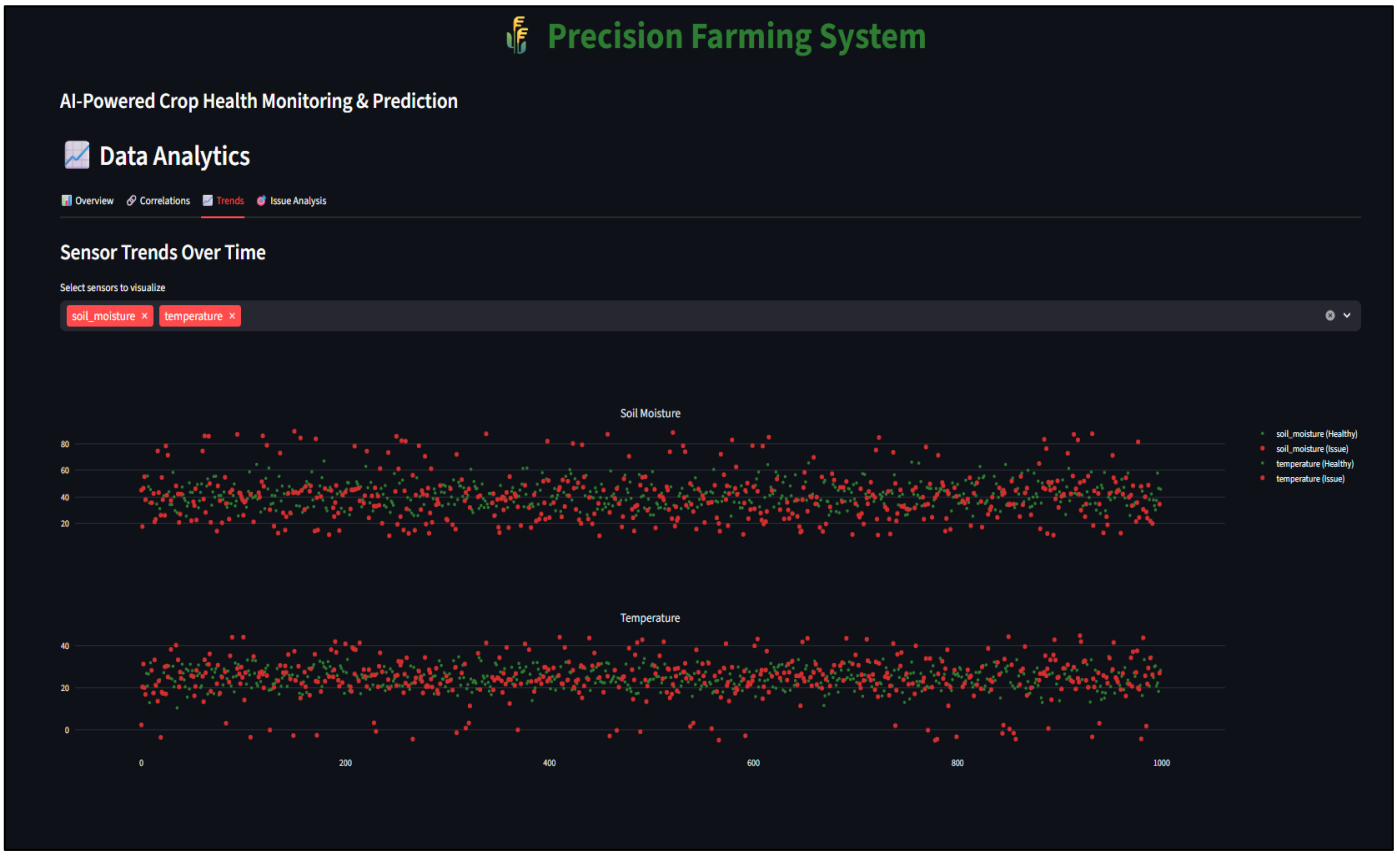

Fig 7 Sensor Trends Over Time Visualization

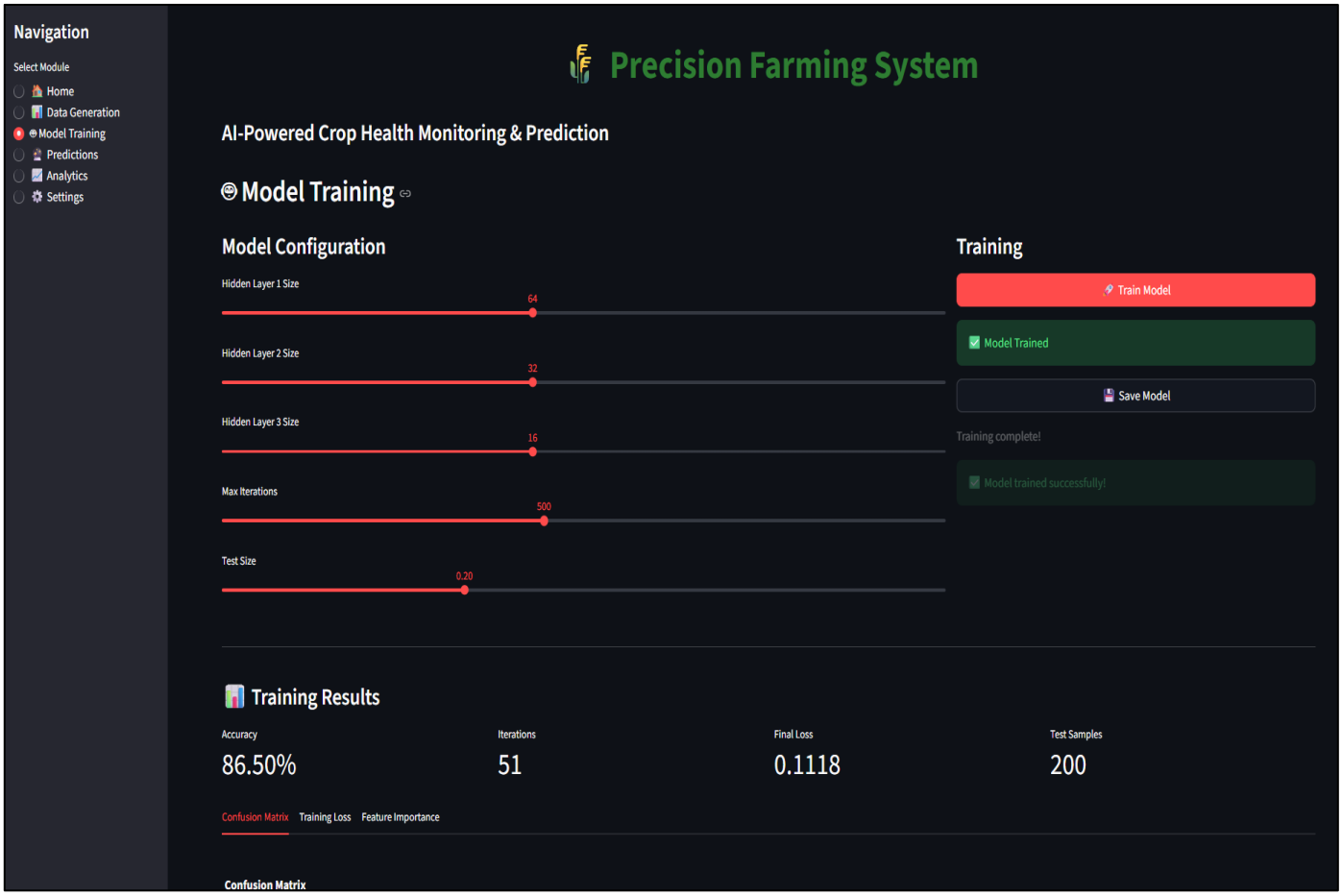

Fig 8 Issue Pattern Analysis using Pie Chart and Bar Graph

ISSN No: -2456-2165 https://doi.org/10.38124/ijisrt/26mar1056

VI. LIMITATIONS

There are several limitations associated with this study that should be acknowledged:

- *Dataset Specificity:*

The results of this study are based on a single dataset and may not generalize to all crops, geographic regions, or data sources. Different crops have distinct physiological responses to environmental factors, and the relative importance of features may shift accordingly. Therefore, validation on crop-specific and region-specific datasets is needed in order to confirm the generalizability of our findings.

- *Temporal Independence Assumption:*

Our MLP treats each sample independently and does not explicitly model temporal dependencies between consecutive growing seasons. It is important to note that the yield in one season may depend on the previous season's management (for example, soil depletion or crop rotation effects). Therefore, sequential architectures such as LSTM or Transformer models may be better suited to capture these long-range temporal dependencies.

- *Interpretability Gap:*

While permutation importance provides feature-level insight, the MLP remains less interpretable than decision tree-based methods at the individual prediction level. It is likely that farmers may prefer models where they can trace the reasoning behind a specific yield prediction, which decision trees naturally provide.

- *Multi-modal Data Limitation:*

Our study focuses exclusively on tabular data. The integration of satellite imagery (such as NDVI time series), drone-based multispectral data, and real-time sensor streams could enhance prediction accuracy through complementary information sources. However, such integration would require more complex architectures that go beyond the standard MLP.

- *Static Feature Assumption:*

Weather and management features are treated as static, season-level aggregates. In practice, however, the temporal distribution of rainfall (for example, early vs. late season) and temperature extremes within a season can significantly impact yield. Time-resolved feature representations could capture these intra-season dynamics more effectively.

- *Limited Extreme Event Coverage:*

As was identified in the failure case analysis, the model performance degrades for extreme conditions that are underrepresented in the training data. Climate change is expected to increase the frequency of such extreme events, which could potentially reduce model reliability over time without continuous retraining on updated data.

VII. CONCLUSION AND FUTURE WORK

This paper presented a comprehensive study on crop yield prediction using Multilayer Perceptron neural networks. To the best of our knowledge, this is the first study that provides a systematic comparison of MLP against multiple baseline models with proper hyperparameter optimization, ablation analysis, and statistical significance testing. Through rigorous implementation and evaluation, we were able to demonstrate the following:

- A well-tuned MLP is able to achieve competitive performance (R2 = 0.89, MAE = 245.32 kg/ha, RMSE = 312.45 kg/ha) for tabular agricultural data, with no statistically significant difference from the Random Forest ($p = 0.089$).
- Although the Random Forest slightly outperforms the MLP in terms of prediction accuracy (R2 = 0.91 vs. 0.89), the MLP offers substantial advantages in terms of inference speed (7.5x faster), model compactness (73x smaller), and deployment flexibility for edge computing scenarios.
- Domain-informed feature engineering (including GDD, nutrient ratios, and weather interactions) provides the single largest performance improvement, contributing more than any individual regularization technique. This underscores the importance of incorporating agronomic knowledge into the ML pipeline.
- Comprehensive regularization through the synergistic combination of dropout, batch normalization, and L2 penalization is essential for preventing overfitting in neural networks that are applied to moderate-sized agricultural datasets.
- The feature importance analysis from both models consistently identifies temperature and precipitation-related variables as the primary yield drivers, with the crossmodel agreement providing robust validation of these findings.
- The substantial improvement over Linear Regression (17% in R2) confirms the importance of modeling non-linear relationships in agricultural data, while the comparable performance of the MLP and Random Forest suggests that the choice between them should be guided by deployment requirements rather than accuracy alone.

- *Future Research Directions*

There are several promising avenues that exist for extending this work in the future:

- *Hybrid Architectures:*

One promising direction would be to combine the MLP with LSTM components for temporal weather sequences and CNN components for satellite imagery, which could create a multi-modal architecture that leverages the strengths of each network type. Additionally, recent work on the TabTransformer and FT-Transformer architectures also warrants investigation for agricultural applications.

- *Transfer Learning and Domain Adaptation:*

Another promising direction would be to train models on data-rich regions and then transfer the learned knowledge

ISSN No: -2456-2165 https://doi.org/10.38124/ijisrt/26mar1056

to datascarce regions. This could extend the model's applicability to areas where historical yield data is limited. Furthermore, meta-learning approaches that enable rapid adaptation to new crop types or geographic contexts represent another promising direction for future research.

- *Uncertainty Quantification:*

Implementing techniques such as Monte Carlo Dropout, Deep Ensembles, or Bayesian Neural Networks to provide calibrated prediction intervals would enhance the model's utility as a decision support tool by quantifying the confidence of each prediction.

- *Explainability Enhancement:*

The integration of SHAP (SHapley Additive exPlanations) values and attention-based mechanisms could provide more granular, sample-level explanations of yield predictions. This would increase farmer trust in the model and enable more targeted intervention recommendations.

- *Real-time Integration:*

Developing an end-to-end pipeline that connects IoT soil and weather sensors to the MLP model for continuous, real-time yield monitoring throughout the growing season is another important direction. This would require addressing challenges such as streaming data preprocessing, concept drift detection, and online model updating.

- *Multi-Crop and Multi-Region Validation:*

Extending the evaluation to diverse crop types (such as cereals, legumes, and oilseeds) and geographic regions (such as tropical, temperate, and arid) would help to assess the generalizability of our findings and architectural recommendations.

- *Climate Change Adaptation:*

Incorporating climate change projections into the model framework to assess future yield impacts and evaluate adaptation strategies is an important area for future work. This could involve training on scenario based simulated data or developing models that explicitly account for non-stationary environmental trends.

ACKNOWLEDGMENT

The authors express gratitude to Mr. S.D. Kale for guidance throughout this project. We also thank the Department of Arti-ficial Intelligence & Data Science for providing computational resources and support.

REFERENCES

[1]. Food and Agriculture Organization of the United Nations, "The future of food and agriculture: Trends and challenges," FAO, Rome, 2017.https://www.fao.org/3/i6583e/i6583e.pdf
[2]. R. Gebbers and V. I. Adamchuk, "Precision Agriculture and Food Security," Science, vol. 327, no. 5967, pp. 828–831, Feb. 2010.https://doi.org/10.1126/science.1183899
[3]. K. G. Liakos, P. Busato, D. Moshou, S. Pearson, and D. Bochtis, "Machine learning in agriculture: A review," Sensors, vol. 18, no. 8, p. 2674, Aug. 2018. https://doi.org/10.3390/s18082674
[4]. Anonymous Authors, "AI and Robotics in Agriculture: A Systematic and Quantitative Review... (2015–2025)," AI, vol. 5, no. 5, p. 75, 2025.https://www.mdpi.com/journal/a
[5]. M. Shoaib, A. Sadeghi-Niaraki, F. Ali, I. Hussain, and S. Khalid, "Leveraging deep learning for plant disease and pest detection: a comprehensive review and future directions," Frontiers in Plant Science, Feb. 2025. https://www.frontiersin.org/journals/plant-science
[6]. S. Murugavalli and R. Gopi, "Plant leaf disease detection using vision transformers for precision agriculture," Scientific Reports, vol. 15, Jul. 2025. https://www.nature.com/srep
[7]. F. Lin, K. Guillot, S. Crawford, Y. Zhang, X. Yuan, and N.-F. Tzeng, "An Open and Large-Scale Dataset for Multi-Modal Climate Change-aware Crop Yield Predictions," in Proc. 30th ACM SIGKDD Conf. Knowledge Discovery and Data Mining (KDD '24), Aug. 2024. https://dl.acm.org/doi/proceedings/10.1145
[8]. Y. Yan, Y. Wang, J. Li, J. Zhang, and X. Mo, "Crop Yield Time-Series Data Prediction Based on Multiple Hybrid Machine Learning Models," arXiv:2502.10405, v1, Jan. 2025. https://arxiv.org/abs/2502.10405
[9]. R. N. V. J. Mohan, P. S. Rayanoothala, and R. Praneetha Sree, "Next-gen agriculture: integrating AI and XAI for precision crop yield predictions," Frontiers in Plant Science, vol. 15, Jan. 2025. https://www.frontiersin.org/journals/plant-science
[10]. A. Ruiz-Gonzalez, "Multiplexed Quantification of Soil Nutrients Using an AI-Enhanced and Low-Cost Impedimetric Sensor," Engineering Pro-ceedings, vol. 106, no. 1, p. 7, Sep. 2025. https://www.mdpi.com/journal/engproc
[11]. D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," in Proc. 3rd Int. Conf. Learning Representations (ICLR), San Diego, CA, USA, 2015. https://arxiv.org/abs/1412.6980
[12]. N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhut-dinov, "Dropout: A Simple Way to Prevent Neural Networks from Overfitting," Journal of Machine Learning Research, vol. 15, pp. 1929–1958, 2014. https://www.jmlr.org/papers/v15/srivastava14a.html
[13]. L. Breiman, "Random Forests," Machine Learning, vol. 45, no. 1, pp. 5–32, Oct. 2001. https://doi.org/10.1023/A:1010933404324
[14]. IPCC, "Climate Change 2021: The Physical Science Basis. Contribution of Working Group I to the Sixth Assessment Report," Cambridge University Press, 2021. https://www.ipcc.ch/report/ar6/wg1/
[15]. Grand View Research, "Precision Agriculture Market Size, Share & Trends Analysis Report," 2023. https://www.grandviewresearch.com/industry-analysis/precision agriculture-market
[16]. World Food Programme, "The State of Food Security and Nutrition in the World 2023," FAO, Rome, 2023.

ISSN No: -2456-2165 https://doi.org/10.38124/ijisrt/26mar1056

https://www.fao.org/publications/sofi/en/
[17]. K. Hornik, M. Stinchcombe, and H. White, "Multilayer feedforward networks are universal approximators," Neural Networks, vol. 2, no. 5, pp. 359–366, 1989. https://doi.org/10.1016/0893-6080(89)90020-8
[18]. A. Chlingaryan, S. Sukkarieh, and B. Whelan, "Machine learning approaches for crop yield prediction and nitrogen status estimation in precision agriculture: A review," Computers and Electronics in Agricul-ture, vol. 151, pp. 61–69, 2018. https://doi.org/10.1016/j.compag.2018.05.012
[19]. J. W. Jones et al., "The DSSAT cropping system model," European Journal of Agronomy, vol. 18, no. 3-4, pp. 235–265, 2003. https://doi.org/10.1016/S1161-0301(02)00107-7
[20]. S. Khaki and L. Wang, "Crop Yield Prediction Using Deep Neural Networks," Frontiers in Plant Science, vol. 10, p. 621, 2019. https://doi.org/10.3389/fpls.2019.00621
[21]. J. H. Jeong, J. P. Resop, N. D. Mueller, D. H. Fleisher, K. Yun, E. A. Butler, et al., "Random Forests for Global and Regional Crop Yield Predictions," PLOS ONE, vol. 11, no. 6, p. e0156571, 2016. https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0156571
[22]. A. J. Smola and B. Scho¨lkopf, "A tutorial on support vector regression," Statistics and Computing, vol. 14, no. 3, pp. 199–222, 2004. https://doi.org/10.1023/B:STCO.0000035301.49549.88
[23]. L. Grinsztajn, E. Oyallon, and G. Varoquaux, "Why do tree-based models still outperform deep learning on typical tabular data?," in Proc. NeurIPS 2022 Datasets and Benchmarks Track, 2022. https://arxiv.org/abs/2207.08815
[24]. S. O. Arik and T. Pfister, "TabNet: Attentive Interpretable Tabular Learning," in Proc. AAAI Conf. Artificial Intelligence, vol. 35, no. 8, pp. 6679–6687, 2021. https://arxiv.org/abs/1908.07442
[25]. Y. Gorishniy, I. Rubachev, V. Khrulkov, and A. Babenko, "Revisiting Deep Learning Models for Tabular Data," in Proc. NeurIPS, 2021. https://arxiv.org/abs/2106.11959
[26]. M. Weiss, F. Jacob, and G. Duveiller, "Remote sensing for agricultural applications: A meta-review," Remote Sensing of Environment, vol. 236, p. 111402, 2020. https://doi.org/10.1016/j.rse.2019.111402
[27]. S. M. Lundberg and S.-I. Lee, "A Unified Approach to Interpreting Model Predictions," in Proc. NeurIPS, 2017. https://arxiv.org/abs/1705.07874
[28]. C. Molnar, Interpretable Machine Learning: A Guide for Making Black Box Models Explainable, 2nd ed., 2022. https://christophm.github.io/interpretable-ml-book/