A Hybrid Approach for Crop Yield Prediction using Machine Learning Algorithms

H. N. Munasinghe[1*], E. G. T. Dasunika[2], W.W.L.Subhodani[3]
[1] Department of Social Statistics, University of Sri Jayewardenepura
[2,3]Department of Computer Science and Informatics, Uva Wellassa University of Sri Lanka
hnirasha14@gmail.com

Abstract

Sri Lanka's agricultural sector is the backbone of the country's economy, and it is essential to increase agricultural productivity to ensure food security. This research has demonstrated the potential of a hybrid approach that combines Machine Learning (ML) Algorithms to improve crop yield prediction and advance food security in Sri Lanka. Drawing on the literature review findings, this finding proposes a novel hybrid approach that integrates multiple ML algorithms to improve crop yield prediction. The hybrid model leverages key factors such as Humidity, Profile Soil Moisture, and yield for selected 11 districts representing the different climatic zones in Sri Lanka. Crops such as tea, paddy, rubber, and coconut significantly impact the national gross domestic product GDP in Sri Lanka. The performance of various ML algorithms, including Random Forest (RF), K-Nearest Neighbors (KNN), and Artificial Neural Networks (ANN), was evaluated separately to determine their ability to accurately predict crop yields and then hybrid models developed by combining KNN and RF, ANN and KNN, ANN and RF. As a result of these hybrid models, the highest performance was achieved by a hybrid model of KNN with RF, with an R2 value of 0.9965, Mean Squared Error (MSE) of 0.00002%, and Mean Absolute Error (MAE) of 0.06%. Root Mean Squared Error (RMSE) is 0.14%, To enable real-time predictions, a simple web application is created using Flask, a Python web framework. The trained model is then utilized within this application to make yield predictions.

Keywords: *Crop Prediction, Hybrid Approach, K-Nearest Neighbors (KNN), Random Forest (RF), Artificial Neural Networks (ANN)*

1. Introduction

The primary agricultural sector in Sri Lanka is rice production. More than 70% of the population lives in rural areas, where agriculture is the country's primary source of income. Sri Lanka's agricultural output has been consistent, except rice, whose production recently reached self-sufficiency. An increasing population, unpredictable weather, soil erosion, and a changing climate necessitate solutions to ensure timely and reliable agricultural development and output. It also urges boosting the sustainability of agricultural food production (Huang et al., 2019). A few factors influencing crop productivity are landscapes, soil quality, climate changes, genotype, the quality and accessibility of water, meteorological conditions, and harvest scheduling (Pantazi et al., 2016). Like Sri Lanka, agriculture is the backbone of the Indian economy (Ramesh & Vardhan, 2015). The primary agricultural sector in Sri Lanka is rice production. The researchers (Tanuja & Pawar, 2019) predicted crops using Artificial Neural Network (ANN) and Support Vector Machine (SVM) in India, Maharashtra state for the selection of proper crops for farming. Rainfall, minimum and maximum temperature, soil type, humidity, and soil pH value are the considered parameters to reach an accuracy of 73.48% using SVM and 86.80% accuracy with the help of an ANN. The results demonstrate that the ANN model beat the SVM regarding prediction rate and accuracy. This indicates that ANN

techniques may predict the crop type more correctly than SVM for the given dataset. The result of this endeavor aids farmers in making informed crop choices. (Sonal Agarwal & Sandhya Tarar, 2021) used a hybrid approach for crop yield prediction using Machine learning (ML) and Deep Learning (DL). They used Random Forest, Decision Tree (DT), and ANN as their algorithms to determine the best crop. They used SVM as an ML algorithm, Long-Short Term Memory (LSTM), and Recurrent Neural Networks (RNN) as DL algorithms. By utilizing DL techniques, their model is enhanced, and in addition to crop prediction, precise knowledge is gathered on the amounts of necessary soil elements and their prices. To help farmers predict a profitable crop, it analyzes the available data. Variables related to the soil and climate are considered to anticipate an acceptable yield. The accuracy of Random Forest (RF) and ANN techniques is 93%. While the LSTM, RNN, and SVM approaches are considered to be 97% accurate. As a result, DL algorithms and ML algorithms are necessary for more precisely calculating yield.

Agriculture makes a considerable contribution to national economic growth and development. It is connected with other sectors directly and indirectly. The main goal of this connection is to increase the economic development in Sri Lanka at the micro, mezzo, and macro (Wanigasundera, 2015). Agriculture is one of the most critical employment sources in the country. Therefore, Sri Lanka is an agriculture-based country. Most people choose agriculture as their livelihood. Smart agricultural activities should be introduced to the rural community to encourage them to farm and to attract the youth generation to farming because the youth generation has more power and energy to increase production in many ways (Goel et al., 2021). Through their extension efforts, extension specialists in this circumstance are crucial to putting a rural development strategy into action. Achieving sustainability over the long run is essential for eliminating poverty and improving people's living situations (Athukorala, 2017; Kazbekov & Qureshi, 2011; Wanigasundera, 2015). Accordingly, poverty still exists among most rural farmers in Sri Lanka. Extension and advisory services (EAS) have the authority and the potential to make significant steps to minimize poverty. Technology adoption is a key component of the extension system in comprehending how communities attain a sustainable living status based on a sustainable livelihood approach (Brocklesby & Fisher, 2003). Over the decade, technology transfer has been identified as one of the best approaches to poverty alleviation.

Sustainable agriculture is a process that involves producing sustainable agricultural goods in a competitive, efficient, and productive way while protecting and improving the rural community's socioeconomic and environmental situations (Braga, 2015; Osumba et al., 2021). Smart agriculture, or precision agriculture or digital farming, utilizes cutting-edge technologies, data analytics, and digital solutions to optimize agricultural practices. The primary objective of smart agriculture is to boost productivity, efficiency, and sustainability in farming while reducing resource usage and environmental harm. This approach combines diverse technologies and data streams to facilitate data-informed decision-making, automation, and continuous monitoring throughout the farming processes. Innovative agriculture technology, leveraging Internet of Things (IoT) advancements, offers numerous real-time benefits across agricultural methods and practices. These advantages encompass irrigation and plant protection, enhanced product quality, controlled fertilization processes, disease prediction, and more (Adamides et al., 2020). In the smart agriculture system, sensors are vital in measuring and monitoring various factors. For instance, soil health monitoring utilizes specific sensors to measure nutrients, phosphate levels, soil moisture, compaction, and other relevant parameters.

In recent years, agricultural sustainability has become increasingly challenging due to rising food and energy prices, climate change, water shortages, biodiversity loss, and population growth. Smallholder production systems must intensify to meet the growing consumer demand for food. Climate-smart agriculture (CSA) practices have gained attention as they promote sustainable food production while helping to adapt to and mitigate climate change. Understanding the impact of farm practices on greenhouse gas (GHG) emissions is crucial, but it's equally important to consider their implications for smallholder livelihoods. Simply calculating GHG emissions without considering productivity and food security for smallholder farmers in developing nations would be insufficient. In summary, addressing agricultural sustainability requires a comprehensive approach that incorporates climate-smart agriculture practices while considering the interplay between GHG emissions, farm productivity, and the livelihoods of smallholder farmers. Balancing environmental concerns with the needs of farmers is essential to achieve long-term sustainability in agriculture. (Linquist et al., 2012; Rosenstock et al., 2013).

ML algorithms are trained on data sets and can make predictions or decisions without being explicitly programmed to perform the task. They can improve their performance over time as they are exposed to more data. ML is used in a wide range of prediction of outcomes in fields such as agriculture finance, healthcare, transportation, and more (Issam and Martin, 2015). The agro-climatic input parameters impact this complicated phenomenon of crop production. The specifications for agricultural inputs vary from one field to another and from farmer to farmer (Veenadhari, Bharat and Singh, 2014).

ML algorithms can analyze a large amount of data and identify patterns that might not be immediately obvious to humans. By taking into account variables like temperature, rainfall, and area, the predictions provided by ML algorithms will assist farmers in choosing which crop to grow to receive the yield potential and can be especially useful in situations where it is essential to make timely decisions, such as when deciding which crops to plant or when to apply fertilizers (Nigam et al., 2019). Crop yield forecasting highlights beneficial opportunities for controlling food sustainability in a supply chain. Crop yield forecast provides data that can be the foundation for many crucial food security choices, including trading and formulating policies (Nair et al., 2011). ML has the power to help farmers to optimize their crop production and increase their yields.

The adoption of ML algorithms has significantly improved crop yield prediction by processing large volumes of data and identifying complex patterns that influence crop productivity. These algorithms offer more precise and accurate yield estimates compared to traditional methods. Popular ML algorithms in crop yield prediction include Random Forest, SVM, Neural Networks, KNN, Decision Trees, Gaussian Processes, and LSTM. These models require historical yield data, weather information, soil data, and other relevant variables as input features to predict future crop seasons. As ML techniques advance and more data become available, crop yield prediction models are expected to become even more accurate, crucial in optimizing agricultural practices and ensuring food security. Modern artificial intelligence is quickly becoming a key development technology. To choose the best crop for farming, (Fegade and Pawar, 2019) used ANN and SVM to predict crops in India's Maharashtra state. They consider rainfall, minimum and maximum temperatures, soil type, humidity, and soil pH values to achieve an accuracy of 73.48% using SVM and 86.80% accuracy using an ANN. The findings show that the ANN model outperformed the SVM regarding prediction rate and precision. This illustrates that for the given dataset, ANN approaches may predict the crop type more accurately than SVM. The project's outcome helps farmers make informed crop decisions.

To increase yield rates, (Patil, Medar & Desai, 2020) created a technique, which boosts national economies. They employed the NB approach and the KNN method; of the two, NB showed an accuracy of 91.11% and KNN demonstrated an accuracy of 75.55%.

Combining numerous ML algorithms can significantly improve the result because most ML algorithms are tuned for a specific dataset or task, and by assisting one another extend or adapt to new tasks (Abdelrahim, Merlos & Wang, 2016). The concept of hybrid ML approaches having a higher accuracy in prediction is one of the primary considerations for using them (Nosratabadi et al., 2019). A hybrid approach in ML combines multiple models or algorithms to improve the overall performance of a system. This can include combining different models, such as supervised and unsupervised learning, and using ensemble methods to combine the predictions of multiple models. The goal of a hybrid approach is often to leverage the strengths of different models to overcome the limitations of any single model.

Sri Lanka's economy is mainly based on plantations that grow tea, paddy, coconuts, and rubber. Rainfall variations. Despite being a tropical country, Sri Lanka experiences varying climates due to rainfall, elevation, and soil type differences. The majority of research articles utilized ML algorithms in various ways. Their research had already utilized ANN, an extremely precise machine-learning method (Tanuja & Pawar, 2019).

In the Sri Lankan economy, tea, paddy, rubber, and coconuts have a unique historical background regarding how they were initially introduced and how they have since flourished as a sector of the economy. These crops are doing a significant impact on the national GDP every year. Therefore, we have chosen Tea, Paddy, Rubber, and Coconut and 11 districts respectively Badulla, Colombo, Gale, Hambanthota, Kaluthara, Kandy, Kegalla, Matara, Matale, Nuwaraeliya, Ratnapura which cover the main climatic zones (Wet Zone, Dry Zone, Intermediate Zone) in Sri Lanka to continue our research. We have selected specimens such as Humidity, Profile Soil moisture, district, crop and yield to find the most suitable hybrid model for crop yield prediction. We selected three main ML algorithms, KNN, ANN, and RF, and measured the individual performance of each algorithm. Respectively we combined ANN and RF, KNN and RF, and ANN and KNN to develop the hybrid models and calculated the performance of each hybrid model.

2. Materials and Methods

Material

Table 1 : Used Materials

Hardware Requirements	Software Requirements	Libraries	Frameworks	Editors
Laptop, Desktop	Python 3.8.8	pandas 1.4.4	Flask 2.3.2	Jupyter notebook 6.4.2
Core i3/i5 processor		numpy 1.23.5	Bootstrap 5.2.3	Google Colab
Internet Connection		seaborn 0.12.1		Visual Studio Code 1.77
		sklearn 1.1.3		

		matplotlib 3.6.2		
		Tensorflow 2.10.0		
		Keras 2.10.0		
		Ipywidgets 7.6.3		
		IPython 8.2.0		

We utilized a combination of resources, including laptop and desktop computers. The setup consisted of Python 3.8.8 as the programming language and Pandas 1.4.4 for data manipulation and analysis. Jupyter Notebook 6.4.2 was the interactive development environment for coding and documenting work.

The machines were equipped with an Intel Core i3 or i5 processor, providing sufficient computational power for the tasks at hand. To handle numerical computations and array operations, we relied on numpy 1.23.5. Additionally, we leveraged Google Colab for its cloud-based computing capabilities, allowing me to run code remotely and access powerful hardware resources. The research was conducted with a reliable internet connection to ensure uninterrupted access to online resources and data.

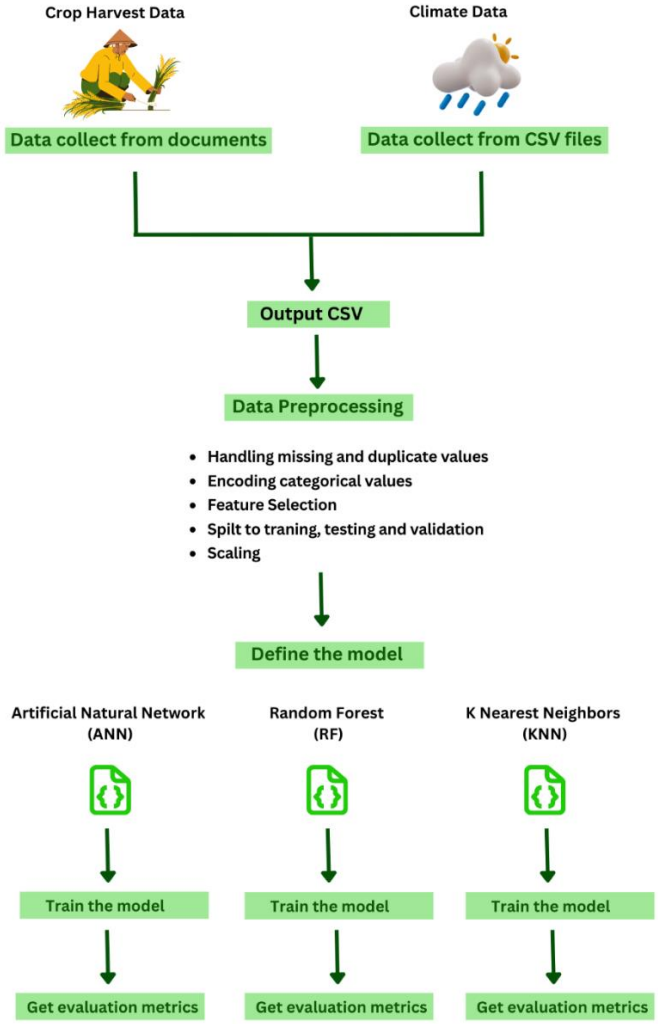
We employed seaborn 0.12.1 and matplotlib 3.6.2 to visualise data and generate plots. These libraries enabled me to create informative and visually appealing graphs to analyze and present the results of our research. We relied on the popular scikit-learn library for ML tasks, which offers various algorithms and tools for different tasks, including classification, regression, and clustering. In terms of ML, we utilized TensorFlow 2.10.0 and Keras 2.10.0, which provided a high-level API for building and training neural networks. These libraries enabled us to develop and evaluate complex ML models in ANN. We used Visual Studio Code 1.77 to create and edit code, which provided a robust and user-friendly coding environment.

Combining the power of Flask 2.3.2 and Bootstrap 5.2.3 allows easy creation of impressive web application frontends. Flask, a lightweight and versatile Python web framework, enables defining routes, managing HTTP requests, and incorporating dynamic content through its templating engine. Bootstrap 5.2.3, a widely embraced front-end framework, furnishes an array of tools and components that expedite the creation of responsive, visually appealing interfaces. Seamlessly integrating Flask and Bootstrap establishes a solid foundation for web projects. The collaboration between Flask and Bootstrap streamlines the frontend development process and equips developers with the means to create polished and interactive web applications.

Methodology

The study contains two phases. The Figure 1 represent first phase includes data collection and analysis, feature extraction by feature selection and feature engineering, model creation, and result interpretation. Then in the Figure 2 represent second phase, a Web application implementation by integrating the trained model to crop yield prediction.

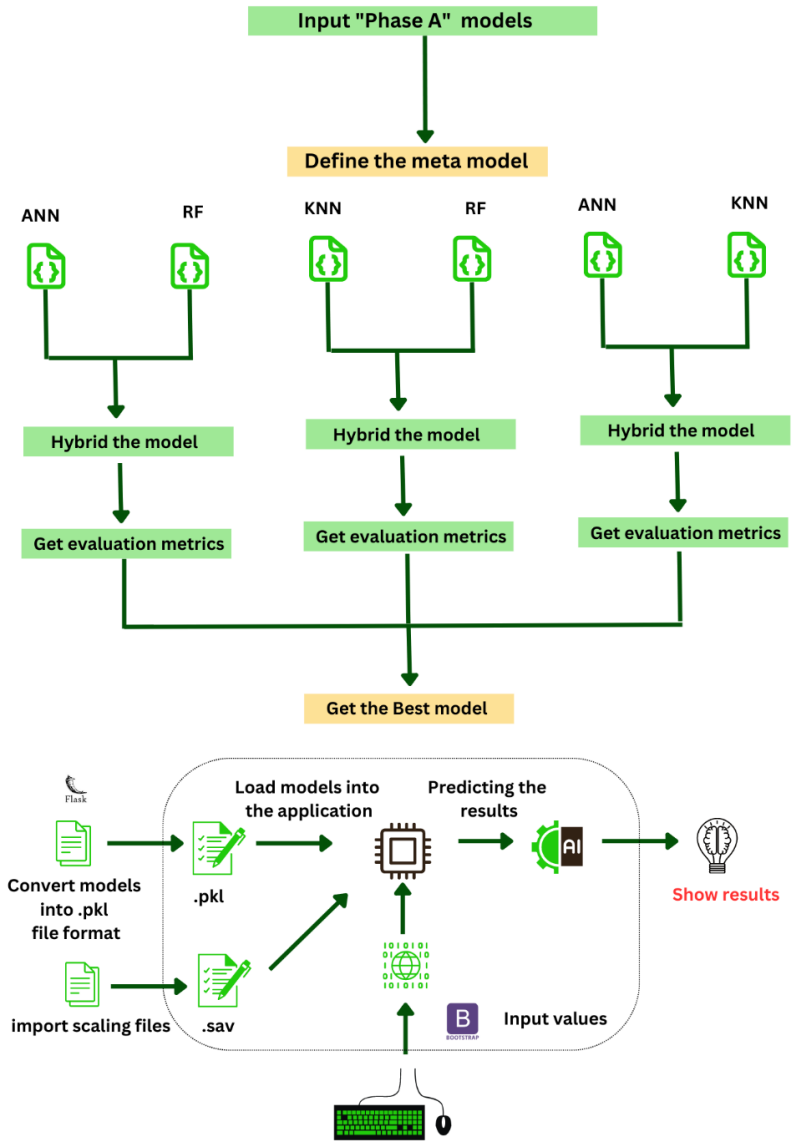
Figure 1: Phase A



Data Collection

The climate data obtained from the "NASA POWER | Data Access Viewer" (https://power.larc.nasa.gov/data-access-viewer,2012) encompasses a wide range of parameters such as Temperature, Humidity, Wind Speed, Sunshine, Rainfall, and Soil Moisture. To align our study with the specific crop harvests in Sri Lanka, we have carefully selected 11 districts that are associated with our chosen crops. For Tea harvest data, we have sourced information from the Sri Lanka Tea Board. The Coconut harvest data is obtained from the Coconut Research Institute Sri Lanka, located in Lunuwila. Additionally, the Rubber Harvest Data is gathered from the Rubber Development Department in Ratnapura. Lastly, the Paddy Harvest Data is acquired from the Department of Census and Statistics in Sri Lanka (http://www.statistics.gov.lk, 2023). By incorporating this comprehensive data collection approach, we aim to gain valuable insights into the respective crop harvests in these regions of Sri Lanka.

Figure 1: Phase B



Model Creation

ML algorithms are trained on data sets and can make predictions or decisions without being explicitly programmed to perform the task. Supervised learning is a type of ML where the algorithm learns from labeled training data to make predictions or decisions about unseen data. In supervised learning, each training example consists of input features and their corresponding labels (also known as target or output). The goal is for the algorithm to learn a mapping from input features to the correct output labels during the training process so that it can generalize well to new, unseen data.

Among the total number of data, we had taken:

- 70% for model training purposes
- 20% for testing the model
- 10% for the model validation

Our Dataset consists of 1802 data rows with 11 columns. In that column, the first 10 columns are labeled as data and the last column is labeled as a target. We use Python language to preprocess the dataset. In Python, we used Numpy, Matplotlib, pandas, Seaborn, Sklearn and Tensorflow libraries. We imported our dataset and searched if it has any null values or duplicate values in the dataset. We have 2 types of categorical data. We assigned values to each categorical data. After that get the correlation of data fields against the harvest.

Training and Testing

Figure 3: Splatted dataset into training, testing and validation

```
# Split the scaled data into training (70%), testing (20%), and validation (10%) sets
X_train, X_temp, y_train, y_temp = train_test_split(data_scaled, target_scaled, test_size=0.3, random_state=42)
X_test, X_validation, y_test, y_validation = train_test_split(X_temp, y_temp, test_size=0.33, random_state=42)
```

In Figure 3 data_scaled represents feature data (input variables), and target_scaled represents target data (output variable). This split is done using the train_test_split function from a library like scikit-learn. The test_size=0.3 parameter means that 30% of data will be allocated to the temporary set (used for further splitting), while 70% will be assigned to the training set. The random_state=42 ensures reproducibility, as the same random state will produce the same split if the code is run again. The X_temp and y_temp are the remaining data after the initial split. The test_size=0.33 parameter indicates that 33% of the temporary set will be used for testing, and the remaining 67% will be used for validation. Again, random_state=42 ensures consistent results if the code is run multiple times.

Finaly, the data is divided into three sets as follows:

- X_train and y_train: This is the training set, containing 70% of the original data. It is used to train your ML models.
- X_validation and y_validation: This is the validation set, containing 22.11% (33% of the remaining 30%) of the original data. It is used to tune hyperparameters and assess model performance during training.
- X_test and y_test: This is the test set, containing the remaining 7.89% of the original data. It is used to evaluate the final performance of your trained model on unseen data.

Scaling

Figure 4 : Scaling

```
from sklearn.preprocessing import MinMaxScaler

target=np.reshape(target, (-1,1))

scaler_data = MinMaxScaler(feature_range=(0,1))
scaler_target = MinMaxScaler()

data_scaled=scaler_data.fit_transform(data)
target_scaled=scaler_target.fit_transform(target)
```

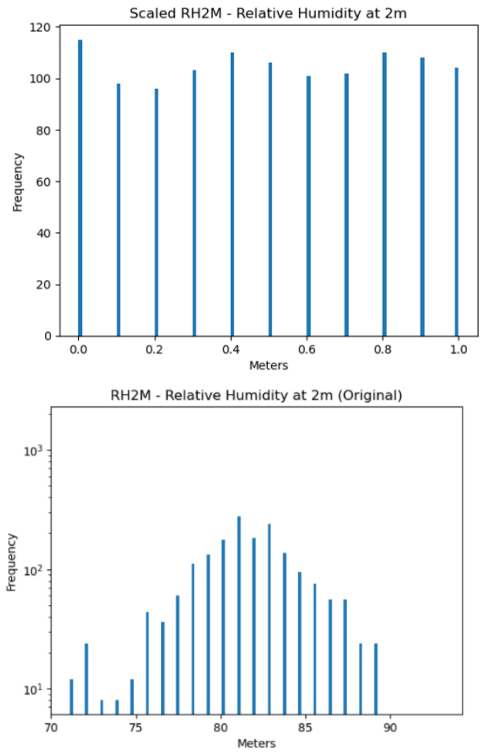
We used MinMaxScaler as a class in the sklearn.preprocessing module of the scikit-learn library. It is used for scaling numerical features to a specified range, typically between 0 and 1, using the min-max scaling technique.

Figure 5 : MinMaxScaler equation

$$x' = \frac{x - \min(x)}{\max(x) - \min(x)}$$

We imported all the CSV files and applied MinMaxScaler to them. As shown in Figure 6, before using MinMaxScaler, the values of RH2M were in the range of 70-40 meters. However, after applying MinMaxScaler, the values of RH2M are scaled to the range of 0-1 meters. Using MinMaxScaler helps ensure that the features are on a consistent scale, mitigating issues associated with varying feature magnitudes and potentially enhancing the performance and stability of our machine-learning models.

Figure 6: Before and after use MinMaxScaler to RH2M



ML algorithms are trained on data sets and can make predictions or decisions without being explicitly programmed to perform the task. Supervised learning is a type of ML where the algorithm learns from labeled training data to make predictions or decisions about unseen data. In supervised learning, each training example consists of input features and their corresponding labels (also known as target or output). The goal is for the algorithm to learn a mapping from input features to the correct output labels during the training process so that it can generalize well to new, unseen data. Regression

problems in ML involve predicting a continuous numerical value based on input features. In regression problems, the evaluation metrics commonly used to assess the performance of the model include MSE, RMSE, MAE, and R^2

In hybrid ML, "stacking" refers to a model assembling technique that combines multiple ML models to make predictions. Stacking, also known as stacked generalization, involves training several base models on the same dataset and then using a meta-model (also known as a blender or meta-learner) to learn from the predictions of these base models. The meta-model takes the base models' predictions as input and makes the final prediction based on their collective outputs.

The stacking method in hybrid ML can be used to leverage the strengths of different models and improve predictive performance. A Gradient Boosting Regression model iteratively fits new models to the residual errors of the previous models, gradually improving the prediction accuracy. It's a powerful technique for capturing complex relationships in data and is known for its ability to handle both linear and non-linear relationships effectively. By combining their predictions, the meta-model can learn to correct any potential errors or biases present in the base models, leading to more accurate and robust predictions.

Web Application Development

Developed a Web-based application to predict crop yield by integrating the trained and tested models for Humidity, Profile Soil Moisture, District, and Crop in phase B.
If users need to predict crop yield using a proposed web application, they need to follow the following simple steps.

- Enter Humidity of the District.
- Enter Profile Soil Moisture of the District
- Select a District from drop-down
- Select Crop from drop-down
- Submit the results

The application proceeds with the inputs and provides output as crop prediction. We have attached the High-fidelity prototype of the User Interfaces of the web application which we have designed using Figma and Actual User Interfaces in the appendices. Flask is a popular and lightweight web framework in Python that is used for web application development. It allows developers to build web applications quickly and efficiently by providing essential tools and libraries for handling routing, request handling, and response generation.

3. Result and Discussion

The following figures depict the six models which have been trained.

Metrics for Evaluation

Different metrics are employed to evaluate the accuracy of regression models, including: R^2 - Represents the proportion of the variance in the dependent variable (the target variable) that is predictable from the independent variables (the features) used in the model.

$$R^2 = \frac{SSR}{SST} = \frac{\sum(\hat{y}_i - \bar{y})^2}{\sum(y_i - \bar{y})^2}$$

MAE: Measures the average absolute difference between the predicted values and the actual values in the dataset.

$$MAE = \frac{1}{N}\sum_{i=1}^{N}\left|y_i - \hat{y}_i\right|$$

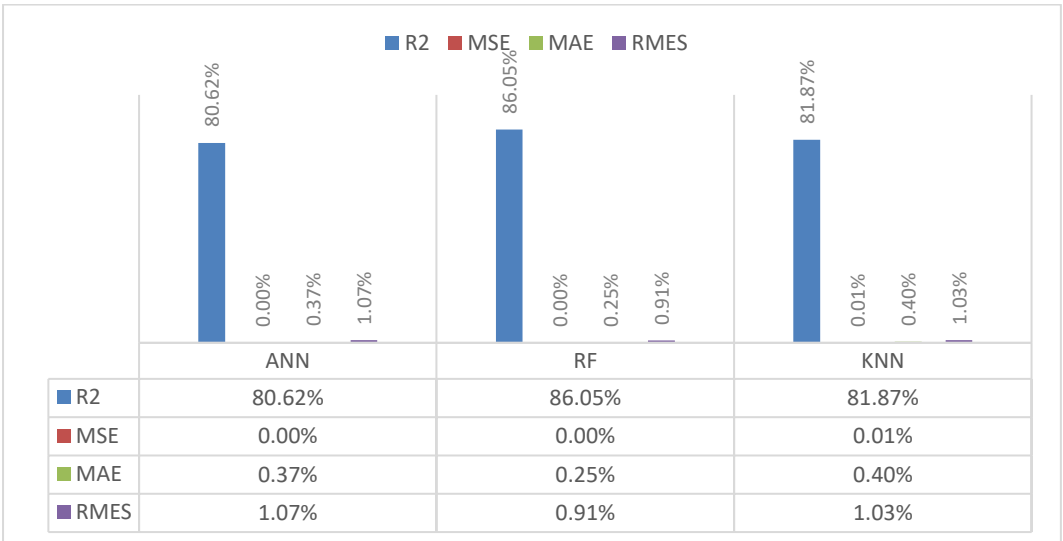
MSE: Measures the average squared difference between the predicted values and the actual values in the dataset

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \tilde{y}_i)^2$$

RMSE: Measure the performance of a predictive model

$$\text{RMSE} = \sqrt{\sum \frac{(y_{pred} - y_{ref})^2}{N}}$$

Figure 7: Summery of performance individual models



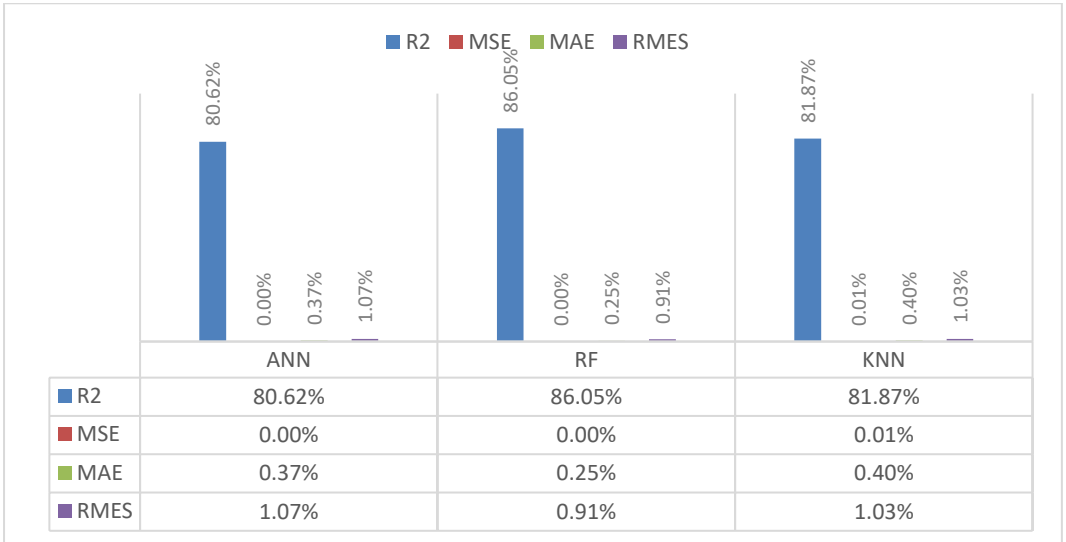
	ANN	RF	KNN
R2	80.62%	86.05%	81.87%
MSE	0.00%	0.00%	0.01%
MAE	0.37%	0.25%	0.40%
RMES	1.07%	0.91%	1.03%

According to Figure 7, Among the KNN, RF and ANN models, the RF model exhibited superior performance. Several factors contribute to this outcome. First, the ensemble nature of RF, which combines multiple decision trees, aids in reducing over fitting and enhancing generalization. Additionally, the model's provision of feature importance scores facilitates effective feature selection and engineering. Furthermore, RF's ability to handle complex nonlinear relationships within data, without necessitating explicit feature transformations, sets it apart from models like KNN. The model is also less sensitive to variations in data scaling, an advantage over KNN. Moreover, RF can manage missing values adeptly without requiring extensive imputation. The ensemble approach helps mitigate over fitting, a challenge that ANNs may face, especially with smaller datasets. Moreover, RF involves fewer hyper parameters, simplifying the tuning process compared to ANNs. This choice is particularly suitable for smaller datasets, as its parallelizable training speeds up computation. Lastly, the model's simplicity and interpretability compared to complex neural networks contribute to its suitability and success. It's important to acknowledge that model performance hinges on factors such as data characteristics, preprocessing quality, hyperparameter tuning, and chosen evaluation metrics. While KNN or ANN might excel under different circumstances, the outlined reasons shed light on why RF excelled in this specific scenario.

Hybrid Models.

The stacked ensemble aims to use the predictions of these base models as meta-features and train a Gradient Boosting Regressor model (meta_model) to make the final prediction. used the RF model and ANN model to make predictions on the test dataset (X_test). Then stacked the predictions from both models' side by side to create meta-features. This combines the predictions into a new dataset where each row contains the predictions of both models for a particular data point. Finally, we created a Gradient Boosting Regressor model, which will act as the meta-model to learn from the stacked predictions and trained the meta-model using the stacked predictions (meta_features) and the corresponding true target values (y_test).

Figure 8: Summary of performance Hybrid models



	ANN	RF	KNN
R2	80.62%	86.05%	81.87%
MSE	0.00%	0.00%	0.01%
MAE	0.37%	0.25%	0.40%
RMES	1.07%	0.91%	1.03%

The observed superiority of the hybrid model involving KNN and RF, when paired through Gradient Boosting Regressor in comparison to the hybridizations of RF with ANN and ANN with KNN, can be attributed to a confluence of factors. Notably, the amalgamation of KNN's capacity to capture local patterns and RF's proficiency in addressing intricate relationships contributes to a synergy that leverages diverse strengths. This strategic blend enables the model to harness local and global insights within the data, resulting in heightened predictive prowess. The ensemble effect stemming from the collaboration mitigates individual model weaknesses, fostering a balanced and resilient predictive framework. Moreover, the distinct feature subsets that KNN and RF may excel on, coupled with their ability to collectively mitigate over fitting, further solidify the hybrid model's performance. This amalgamation adeptly navigates the intricacies of nonlinear relationships, effectively handles outliers, and potentially optimizes hyper parameters, ultimately leading to the observed superior predictive performance. The strategic fusion of KNN and RF in the hybrid model underscores the potential of synergistic modeling techniques to excel in addressing intricate prediction tasks.

The elements encompassed in our study included meteorological conditions, soil attributes, water table depths, geographical locations of crops, and historical yield records. These elements collaborated in analyzing the four prominent crops cultivated within the Karnataka region: Jowar, Rice, Maize, and Ragi. These essential factors were utilized to train two distinct models: a MLP neural network and a RF regression model. This methodology and its outcomes were showcased in the research conducted by (Shetty et al., 2021). Their findings revealed that the MLP achieved a MAE of 0.123, a MSE of 0.034, and a RMSE of 0.185. In contrast, the RF regression model obtained and MAE of 0.124, an MSE of 0.029, and an RMSE of 0.171.

The motivation behind adopting hybrid ML approaches lies in their proven capacity to achieve heightened predictive accuracy, a concept extensively discussed by (Nosratabadi et al., 2019). In their research on Crop Yield Prediction, they employed two hybrid models: artificial neural networks-imperialist competitive algorithm (ANN-ICA) and artificial neural networks-gray wolf optimizer (ANN-GWO). The crops considered for this study were Wheat, Barley, Potato, and Sugar Beet. Among these models, the researchers identified ANN-GWO as the most effective hybrid model for crop predictions. The results showcased an R^2 value of 0.48, a MEA of 26.65, and a RMSE of 3.19

In our proposed work, we determined that the optimal hybrid model for crop prediction, particularly for Coconut, Paddy, Rubber, and Tea across 11 districts of Sri Lanka, is the combination of KNN with RF. Our study incorporated significant features such as Humidity, Soil Moisture Profile, Area, and Yield. Among these features, the hybrid KNN-RF model exhibited exceptional performance, achieving an R^2 value of 0.99, a MSE of 0.000002, a MAE of 0.0006, and a RMSE of 0.014.

4. Conclusion

A crop prediction is an innovative idea that holds the potential to greatly enhance the interest and efficiency of farmers in agriculture. In this research study, we have demonstrated that individual and hybrid ML models can be used to predict crop yield predictions in Sri Lanka. This research considered humidity and profile soil moisture as climate factors that are dependent on crop predictions. In addition to the climate factors, we also considered 11 district numbers and 4 crop types.

The performance of various ML algorithms, including RF, KNN, and ANN, was evaluated separately to determine their ability to accurately predict crop yields and then hybrid models developed by combining KNN and RF, ANN and KNN, ANN and RF. As a result of these hybrid models, the highest performance was achieved by a hybrid model of KNN with RF with an R2 value is 0.9965 (0< R2<1), MSE is 0.00002 (0<=MSE), MAE is 0.006 (0<=MAE) and RMSE is 0.014 (0<=RMSE), To enable real-time predictions, a simple web application is created using Flask, a Python web framework. The trained model is then utilized within this application to make yield predictions. Accordingly, KNN with RF will be the best-performance hybrid model for crop yield prediction.

For future work we would like to address the following suggestions for research related to the same area. The performance of each model can be increased with a larger dataset as a further improvement. We can expand our findings to provide crop management recommendations based on the predicted yield. In the future, we should consider incorporating agricultural best practices, fertilization schedules, and irrigation recommendations tailored to specific crop predictions. We need to integrate historical and real-time weather data into our crop yield prediction model, as weather conditions significantly influence crop performance, and including this data can improve prediction accuracy. Currently, we have gathered climate and crop harvest data year by year, but it would be better to gather climate and crop data monthly for finer granularity and more accurate predictions.

References

[1] Abdelrahim, M., Merlos, C., & Wang, T. (George"). (2016, February 1). *Hybrid Machine Learning Approaches: A Method to Improve Expected Output of Semi-structured Sequential Data.* IEEE Xplore. https://doi.org/10.1109/ICSC.2016.72

[2] Adamides, G., Kalatzis, N., Stylianou, A., Marianos, N., Chatzipapadopoulos, F., Giannakopoulou, M., Papadavid, G., Vassiliou, V., & Neocleous, D. (2020). Smart Farming techniques for climate change adaptation in Cyprus. *Atmosphere, 11*(6), 557. https://doi.org/10.3390/atmos11060557

[3] Agarwal, S., & Tarar, S. (2021). A hybrid approach for crop yield prediction using machine learning and deep learning algorithms. *Journal of Physics, 1714(1)*, 012012. https://doi.org/10.1088/1742-6596/1714/1/012012

[4] Agricultural Extension in South. (n.d.). Retrieved August 25, 2023, from https://www.aesanetwork.org/wp-content/uploads/2018/09/Working-Paper-5.pdf

[5] Athukorala, W. (2017). Identifying the role of agricultural extension services in improving technical efficiency in the paddy farming sector in Sri Lanka. *Sri Lanka Journal of Economic Research, 5*(1), 63–77. https://doi.org/10.4038/sljer.v5i1.58

[6] Braga, F. (2015). The Sustainable Agriculture Initiative Platform: the first 10 years. *Journal on Chain and Network Science, 15*(1), 27–38. https://doi.org/10.3920/jcns2014.x015

[7] Brocklesby, M. A., & Fisher, E. (2003). Community development in sustainable livelihoods approaches - an introduction. *Community Development Journal,*

38(3), 185–198. https://doi.org/10.1093/cdj/38.3.185

[8] Department of Census and Statistics. (2023). Department of Census and Statistics-Sri Lanka. Statistics.gov.lk. http://www.statistics.gov.lk/

[9] D Ramesh. (2015). Analysis of crop yield prediction using data mining techniques. *International Journal of Research in Engineering and Technology, 04*(01), 470–473. https://doi.org/10.15623/ijret.2015.0401071

[10] Fegade, T. K., & Pawar, B. V. (2019). Crop Prediction Using Artificial Neural Network and Support Vector Machine. *Data Management, Analytics and Innovation*, 311–324. https://doi.org/10.1007/978-981-13-9364-8_23

[11] Goel, R. K., Yadav, C. S., Vishnoi, S., & Rastogi, R. (2021b). Smart agriculture – Urgent need of the day in developing countries. *Sustainable Computing: Informatics and Systems, 30*, 100512. https://doi.org/10.1016/j.suscom.2021.100512

[12] Ho, Y. (2020). Comments on: Huang et al. (2019) Emerging trends and research foci in gastrointestinal microbiome', J. Transl. Med., 17: 67. Journal of Translational Medicine, 18(1). https://doi.org/10.1186/s12967-020-02379-9

[13] Nosratabadi, S., Szell, K., Beszedes, B., Imre, F., Ardabili, S., & Mosavi, A. (2020, October 1). *Comparative Analysis of ANN-ICA and ANN-GWO for Crop Yield Prediction*. IEEE Xplore. https://doi.org/10.1109/RIVF48685.2020.9140786

[14] Pantazi, X. E., Moshou, D., Alexandridis, T., Whetton, R., & Mouazen, A. M. (2016). Wheat yield prediction using machine learning and advanced sensing techniques. Computers and Electronics in Agriculture, 121, 57–65. https://doi.org/10.1016/j.compag.2015.11.018

[15] Stackhouse, P. (2024). POWER | DAV. Nasa.gov. https://power.larc.nasa.gov/data-access-viewer/#:~:text=The%20NASA%20POWER%20Project