SAAD DAHLAB UNIVERSITY OF BLIDA 1

Faculty of Sciences

Department of Computer Science

Master's Dissertation

In Computer Science

Option : Software Engineering

THEME :

Hybrid Stacking Ensemble Method For Crop Yield Prediction

Presented by

MAZOUZ Besma

BENAMIRA Ikram

Supervised by

Dr. BOURAS Dalila

Dr. BOUMAHDI Fatima

Members of the jury

President : Dr. REZZOUG Nachida

Examinator : Dr. GUENDOUZ Amina

Academic year : 2024 - 2025

Acknowledgment

First and foremost, Alhamdulillah, all praise and thanks to Allah who granted me the strength and guidance to reach this important moment in my life. Without His blessings, none of this would have been possible.

I also want to express my deepest gratitude to myself. Yes I acknowledge the effort, persistence, and dedication I showed throughout this journey. Balancing work and study was never easy, but step by step, I kept moving forward until I reached this point. I am proud of the long hours, the sleepless nights, and the determination that brought me here.

My heartfelt thanks go to my parents, who supported me unconditionally. Their sacrifices, encouragement, and belief in me gave me the strength to continue. I hope this achievement makes them proud and marks only the beginning of many future successes.
I am grateful to my brother and sisters for their continuous care and positive influence that helped me stay focused.

To my dear friends Anfel, Yasmina, and Abdelilah, thank you for being by my side. Your encouragement, emotional support, and sincere friendship made a big difference in this journey.

Special thanks to my supervisors, Ms. Dlila and Ms. Boumehdi, for their valuable guidance, support, and trust in me to work on this thesis. Their advice and expertise shaped this work into what it is today.

Finally, I want to say that this is just the beginning a starting point. I still have a long way to go, many dreams to achieve, and I am more motivated than ever to move forward and accomplish even greater things.

Besma

First and foremost, I thank and praise **Allah** for His guidance that illuminated my path, and His provision that opened doors of hope and success. Without His grace and blessings, I would not have reached this moment, nor harvested the fruit of years of hard work.

I extend my deepest thanks to myself. To the person I have become, who never gave up, who moved forward despite exhaustion and setbacks, who stayed up late studying, cried in silence, and yet rose again and continued the journey. To my younger self who believed in me, who had faith that I could become this hardworking woman who refused to settle for anything less than excellence...I made it.

All my appreciation goes to my two amazing supervisors : **Dr Boumehdi** and **Dr Bouras**, for their unwavering support, valuable guidance, and for standing by us every step of the way. Thank you for believing in me, for your trust, and for your constant encouragement

To my beloved father, my unshakable pillar and eternal supporter thank you. Not a day passed without your care reaching me, through your voice, your prayers, or your silent presence. Your support was the greatest source of inspiration in my life.

And to my dearest mother, the source of love and peace, the only one who truly understands me, who gives me every ounce of her strength and love. Thank you for your warmth, your embrace, and your unwavering prayers.

Support is not complete without siblings, thank you to my brothers, and my sisters for your presence, encouragement, and love during my hardest times. And to your little children, my precious nieces and nephews, who never fail to bring joy and light into my life, you are pure happiness, I sincerely pray that God watches over you and protects you. and to my brothers' wives, who have been like sisters to me thank you for your kindness and support.

To my dear friends **Hanan** and **Faiza**, whom I met during university not by coincidence, but as a divine gift of true and loyal friendship. Thank you for being by my side, for your love, your words, and your support. You are rare treasures in my life.

To all my friends, thank you for the memories and the time we have enjoyed together. Every moment spent with you is a treasure I will never forget.

finally i want to thank everyone who whispered a kind prayer for me, who supported me with a sincere word or a genuine gesture.

Ikram

ملخص

مع الارتفاع المستمر في عدد سكان العالم، وصل الطلب على الغذاء إلى مستويات يصعب السيطرة عليها، مما يُثقل كاهل القطاع الزراعي لزيادة الإنتاجية. ولا تقتصر الحاجة إلى زيادة الإنتاجية الزراعية على ذلك، بل يجب أن يتم ذلك بطريقة مستدامة في ظل محدودية الموارد وظروف بيئية متغيرة ومتزايدة التقلب. ويُعد التنبؤ الدقيق بالغلة إحدى المشكلات الرئيسية في الزراعة اليوم؛ فالقرارات التي تؤثر على الأمن الغذائي، والاستخدام الأمثل للموارد، والتنبؤات الاقتصادية، وإلى حد ما، الممارسات المستدامة، تعتمد جميعها على التنبؤ الدقيق بالغلة. تُقدم هذه الأطروحة نهجا هجينا، H_StackCYP، يجمع بين التعلم الآلي التقليدي والتعلم العميق للتنبؤ بغلة المحاصيل. يتألف هذا النموذج الهجين من مجموعة مركبة من نماذج التعلم النشط التي يمكنها استخدام نماذج تنبؤية غير خطية مثل الغابات العشوائية، وXGBoost، وأشجار القرار، والشبكات العصبية التلافيفية CNN، والشبكات العصبية العميقة DNN، ونماذج المدرك متعدد الطبقات MLP، مما يُغني عن مجرد اتجاهات خطية في البيانات الزراعية.

يُعد توقع قيم الرطوبة كمعامل إدخال إضافي، أحد الجوانب المهمة في هذا العمل، مما يُوفر سياقا أوسع لمرحلة تقدير الغلة في الدراسة. كما تبحث الدراسة في الميزات الهندسية المعقدة والقابلة للتنفيذ، وضبط المعلمات الفائقة بالكامل لتحسين كل نموذج. تُظهر نتائج هذه الدراسة تفوق نموذج H_StackCYP الهجين على النماذج المطبقة بشكل فردي، بالإضافة إلى دقة جيدة، حيث أعطت R^2 القيمة 0.983 في أفضل الحالات.

يُسلط المشروع الضوء بفعالية على إمكانات استخدام أساليب التجميع للتنبؤ الزراعي، وقيمة استخدام الخصائص المناخية لتحسين التنبؤ، كما يُظهر فائدة النموذج كحل عملي وقابل للتطوير، يُمكن للمزارعين أو مُخططي الزراعة الاستفادة منه في اتخاذ القرارات لتعزيز الإنتاجية من خلال استخدام أساليب قائمة على البيانات.

الكلمات المفتاحية: التنبؤ بغلة المحاصيل، التجميع الهجين، التعلم التجميعي، التعلم الآلي، التعلم العميق، التنبؤ بالرطوبة، الزراعة.

Abstract

Food demand is already reaching unmanageable levels with the continuous rise in the world population, which puts added strain on the agricultural sector to increase productivity, not only is there a need to increase agricultural productivity, but it will have to be done in a sustainable way with limited resources and environmental conditions that are subject to changing and increasingly unpredictable conditions. Accurate yield prediction is one of the major problems in agriculture today ; decisions that impact food security, adequate resource use, economic forecasting, and to an extent sustainable practices all rely on accurate yield predictions. In this dissertation, it presents a hybrid approach, H_StackCYP, that combines traditional machine learning and deep learning to predict crop yields. The hybrid model is a stacked ensemble of active learning models that can utilize non-linear predictive models such as Random Forest, XGBoost, Decision Trees, Convolutional neural networks (CNN), Deep Neural Networks (DNN) and Multilayer perceptron (MLP) models so it is not just linear trends in agricultural data. An important aspect of this work is the anticipation of humidity values as an additional input parameter which provides more context to the yield estimation phase of the study. The study also investigates complex feasible engineered features and full hyperparameter tuning to optimize each model. The findings from this study demonstrate that the H_StackCYP hybrid model outperformed models applied individually as well as demonstrating decent accuracy, with R^2 equal 0.983 in best cases. The project effectively highlights the potential of using ensemble methods for agriculture forecasting and the value of using climate-based features to improve the prediction, also it demonstrates the usefulness of the model as a real-world and scalable solution farmers or agriculture planners may leverage in their decision-making to enhance productivity through the use of data-driven methods.

Keywords :

Crop yield prediction, Hybrid stacking, Ensemble learning, Machine learning, Deep learning, Humidity prediction, Agriculture.

Résumé

La demande alimentaire atteint déjà des niveaux ingérables avec l'augmentation continue de la population mondiale, ce qui met à rude épreuve le secteur agricole qui doit accroître sa productivité. Il est non seulement nécessaire daccroître la productivité agricole, mais cela devra se faire de manière durable, avec des ressources limitées et des conditions environnementales soumises à des conditions changeantes et de plus en plus imprévisibles. La prévision précise du rendement est l'un des problèmes majeurs de l'agriculture actuelle ; Les décisions qui ont un impact sur la sécurité alimentaire, l'utilisation adéquate des ressources, les prévisions économiques et, dans une certaine mesure, les pratiques durables reposent toutes sur des prévisions de rendement précises. Dans ce memoire, on présente une approche hybride, H_StackCYP, qui combine l'apprentissage automatique traditionnel et l'apprentissage profond pour prédire les rendements des cultures. Le modèle hybride est un ensemble empilé de modèles dapprentissage actif qui peuvent utiliser des modèles prédictifs non linéaires tels que Random Forest, XGBoost, Decision Trees, Convolutional Neural Networks (CNN), Deep Neural Networks (DNN) et Multilayer Perceptron (MLP) afin qu'il ne s'agisse pas uniquement de tendances linéaires dans les données agricoles. Un aspect important de ce travail est l'anticipation des valeurs d'humidité comme paramètre d'entrée supplémentaire qui fournit plus de contexte à la phase d'estimation du rendement de l'étude. L'étude examine également des fonctionnalités techniques complexes et réalisables ainsi qu'un réglage complet des hyperparamètres pour optimiser chaque modèle. Les résultats de cette étude démontrent que le modèle hybride H_StackCYP a surpassé les modèles appliqués individuellement et a démontré une précision décente, avec R^2 égale à 0,983 dans le meilleur des cas. Le projet met clairement en évidence le potentiel de l'utilisation de méthodes d'ensemble pour la prévision agricole et la valeur de l'utilisation de fonctionnalités basées sur le climat pour améliorer les prévisions. Il démontre également l'utilité du modèle en tant que solution concrète et évolutive que les agriculteurs ou les planificateurs agricoles peuvent exploiter dans leur prise de décision pour améliorer la productivité grâce à l'utilisation de méthodes axées sur le climat.

Mots clés :
Prédiction du rendement des cultures, Empilement hybride, Apprentissage densemble, Apprentissage automatique, Apprentissage profond, Prédiction de l'humidité, Agriculture.

Table of Content

List of Figures

List of Tables

List of Abbreviations

AI Artificial Intelligence.

ML Machine Learning.

DL Deep Learning.

RF Random Forest.

XGBoost Extreme Gradient Boosting.

DNN Deep Neural Network.

CNN Convolutional Neural Network.

MLP Multilayer Perceptron.

MAE Mean Absolute Error.

MSE Mean Squared Error.

RMSE Root Mean Squared Error.

R^2 Coefficient of Determination.

API Application Programming Interface.

H_StackCYP Hybrid Stacking for Crop Yield Prediction.

General Introduction

1 Context

As the worlds population continues to increase at an alarming rate, food demand is also increasing. Food security has become a primary area of concern for policymakers, companies, producers and in turn farmers. As the agricultural sector works to reconcile the need for food security with a rapidly increasing, and uncertain demand, they are confronting more challenges than ever. One of the challenges that have emerged from the food security crisis is that agricultural stakeholders are working without reliable information on expected crop production, or yield predictions, which is leaving uncertainties when making planting decisions. On the ground, farmers are making decisions to plant with very little useful information on the expected yield. consequently, farming has become a guessing-game of sorts, a complete shot in the dark, where farmers risk planting without knowing if there is any potential to harvest the end product, or its viability. This issue of blind planting also denotes the absence of scientific predictive tools and methodical decision-support systems to reduce doubt, assist planning, and build viability into agricultural practices. farmers are unable to estimate the productivity of their crops. So suffering economic losses and increasing their vulnerability to disruptions in production from external shocks. These separate pieces of uncertainty are the foundations of food system instability, which is compounded when taken to scale in an increasingly demanding world.

2 Dissertation organization

Our dissertation is structured as follows :

Chapter 1 : State of the Art

This chapter is divided into three sections. The first one contains the necessary information to understand agriculture and the factors that affect it. In the second section, we present the relationship between Artificial Intelligence (AI) and agriculture, with a particular focus on crop yield. Finally, the third section provides a comprehensive review of existing works related to crop yield prediction.

Chapter 2 : H_StackCYP

This chapter introduces our proposed approach, called **H_StackCYP**, for crop yield prediction. It details the steps of feature engineering, data preprocessing, and the methodology used for yield prediction.

Chapter 3 : Results and Evaluation

This chapter presents the results obtained from applying the proposed technique to the studied dataset. It analyzes the impact of predicting humidity as an additional feature and how it enhances the performance of the models. The results are then compared with previous works that used both machine learning and deep learning approaches.

Chapter 1

State of the Art

1.1 Introduction

Agriculture represents a substantially important field that continues to advance with technological improvements to address the increasing global demand for food. In particular, artificial intelligence (AI) represents an important aspect of that advancement and facilitates data-driven evaluation and decision-making in addressing and improving yield and sustainability practices. AI-enhanced yield prediction is of particular importance regarding agriculture's growing data capacity and the aim of optimizing agricultural practices. This chapter will be divided into three sections. The first section provides a brief overview of agriculture, the second discusses contemporary technologies relating to AI in agriculture, and the third provides related work for crop yield prediction.

1.2 Agriculture

Agriculture is among the oldest and most important human activities, having been a significant factor in the rise of civilizations and ultimately in the survival of the human race, particularly with respect to providing food supplies. Agriculture entails the cultivation of plants and the domestication of animals on a large scale as means of food production, raw materials for human efforts and ecosystems for healthy human living. Agricultural practices have evolved over time due to technological advancements (to mention only one influencer) resulting in increased productivity, food security and economic growth around the world. [Food and of the United Nations , FAO]

1.2.1 The importance of agricultural yield

[Pingali, 2012] demonstrate that the advancement of agricultural productivity has been a crucial factor in alleviating food deficits, diminishing poverty rates, and lowering food prices throughout large segments of the developing world. The Green Revolution resulted in a remarkable increase in production of cereal crops, multiplying output by three times with only a 30% expansion of cultivated land. The expansion of land was diverted from the conversion of forests and marginal lands to agriculture and aided environmental sustainability. The increase in productivity raised farmers' incomes, stimulated rural employment and enhanced food security. Sustaining productivity increases as well as improving the competitiveness of smallholders are paramount to address the surging demand for food as the worlds population continues to increase and sustainable agricultural development is further realized.

1.2.2 The factors affecting agricultural productivity

Multiple factors affect agricultural productivity, which can vary by season and place. Comprehending these components is crucial for enhancing production and managing difficulties such as climate change and resource constraints. By examining these factors, more effective methods for maximizing agricultural productivity while preserving ecological balance may be developed.

1.2.2.1 The climate conditions

Climate is the primary factor influencing agricultural production, and climate change has driven significant research in recent years. Research focuses on its possible physiological impacts, such as adjustments to animal and crop output.

— **Temperature :** Temperature is an important environmental variable that affects plant growth, development, and yield. Even modest temperature increases can accelerate crop phenology, shorten crop growth duration and physiological maturity, and reduce potential yield. Increasing temperatures, especially at night, increase respiratory processes and reduce gross and net photosynthesis. Increased temperature also increases evapotranspiration, and therefore crop water requirements. In addition, crop use efficiencies of nitrogen fertilizers, in terms of yields, are reduced at elevated temperatures due to increased volatilization and denitrification. Each of these temperature related physiological changes impacts water stress and nutrient stress for plants. Warmer climates also allow more insects and pathogens to survive and reproduce. [Smith and Almaraz, 2004]

— **Rainfall :** Rainfall has long been a vital factor affecting the amount of water available to crops. Adequate rainfall is essential for crops at all three stages of their growth (germination, flowering, and grain filling). Adequate and timely rainfall, if distributed evenly, will lead to healthy crop growth and optimal yields. However, climate change is expected to lead to erratic rainfall patterns. Reduced rainfall, especially when combined with rising temperatures that increase evaporation rates, can lead to water stress for plants. This stress limits photosynthesis, nutrient uptake, and overall growth, leading to reduced yields. Conversely, heavy rainfall can lead to water logging, poor root respiration, leaching of nutrients essential for crop growth, and ultimately crop failure. [Smith and Almaraz, 2004]

— **Sunlight :** Sunlight is necessary for photosynthesis, which enables plants to convert light energy into chemical energy for growth. On the other hand, stress can result from too much sunshine and extreme temperatures.[Smith and Almaraz, 2004] Sunlight influences crop yield not only via photosynthesis and interaction with ambient CO2, but elevated CO2 can also enhance the rate by which photosynthesis occurs and improve water use efficiency in 3 - carbon compound crops (e.g., wheat, rice) relative to yield, under the best possible conditions, by as much as 10 - 30% [Amthor, 2001] .

— **Relative humidity :** According to [Siebert et al., 2014] humidity plays a key role in determining how crops respond to heat stress. When relative humidity is low, plants are less able to cool themselves through transpiration, which increases canopy temperature. This elevated temperature, especially during flowering, can exceed critical thresholds and lead to a sharp decline in yield. The study found that in dry conditions, canopy temperature was up to 7 °C higher than air temperature measured at standard weather stations. This shows that ignoring humidity can lead to underestimating heat stress effects. Therefore, humidity must be considered when evaluating crop yield under climate change.

1.2.2.2 Soil Quality

Plant development depends on the quality of the soil, which includes cation exchange capacity (CEC), pH, organic matter content, nutrient availability and texture. Root growth is improved, nutrient absorption is optimized, and resistance to unfavorable circumstances is increased with a balanced soil composition.

— **Cation exchange capacity (CEC) :** The cation exchange capacity (CEC) of a soil indicates how well it can hold-on to vital nutrients. Sandy soils with low CEC need regular fertilization to maintain yields, whereas clay soils with high CEC improve agricultural output by offering consistent nutrient availability. [Sposito, 2008]

— **Soil PH :** According to [Brady et al., 2008], pH of the soil is important for the availability of nutrients ; extremes (acidic < 6 or alkaline > 7.5) prevent nutrient absorption and lower yields. A pH adjustment can increase crop yields and fertility because it influences microbial activity and the breakdown of organic materials.

— **Organic matter content :** Decomposed organic wastes, either partially or completely, make up soil organic matter (SOM), which is essential for soil fertility because it improves soil structure, retains more water, and supplies nutrients. Because of the buildup of grassland biomass, rich SOM soils such as those found in the Great Plains of North America are extremely productive. [Brady et al., 2008]

— **Nutrient availability :** [Bernhard, 2010] emphasized that plants need seventeen elements to grow, including important macronutrients like potassium, phosphorus, and nitrogen that are frequently lacking in soils and require fertilizers to produce enough yields .

— **Soil texture :** Water and nutrient retention is determined by the nature of the soil, which includes sand, silt, and clay. The balanced water retention, drainage, and aeration of loamy soils make them perfect for farming.[Sposito, 2008]

1.2.2.3 Biotic Factors : Pests and Diseases

As observed by [Khalid, 2020] Agriculture productivity is being threatened by diseases and pests that are made worse by climate change. In addition to lowering agricultural yield and quality, a rise in disease

1.2.2.4 Practices of Agricultural Management

Maintaining production requires efficient fertilization, irrigation, and crop rotation especially in regions with unreliable rainfall or poor soil. We can increase yields and optimize water consumption with modern irrigation technologies, such as drip irrigation.[Abrha, 2015]

The figure 1.1 below summarizes all the factors discussed that influence agricultural yield.

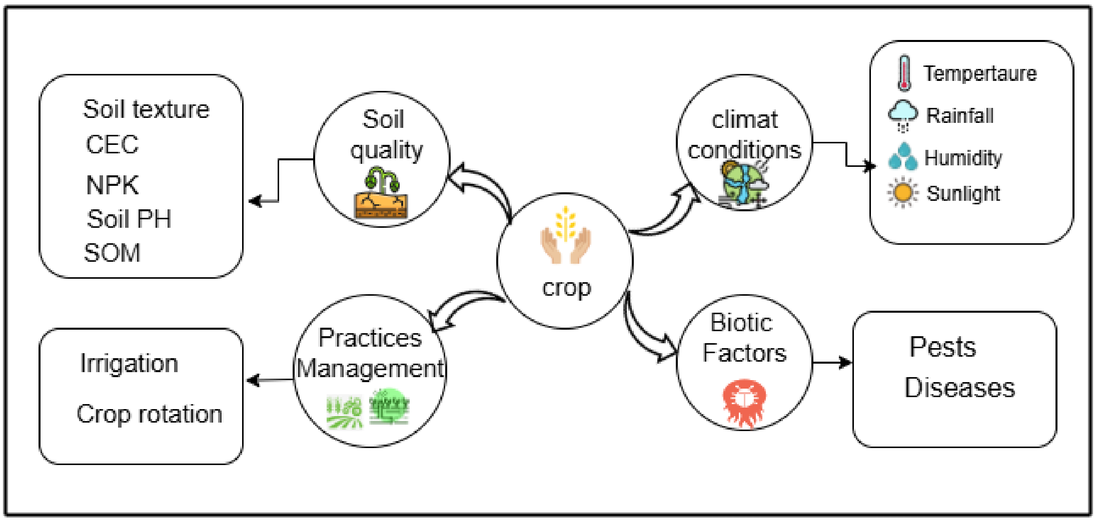

FIGURE 1.1 – Factors affecting agricultural yield

1.3 Agriculture and Artficial intelligence (AI)

In the last few years, we have witnessed the emergence of new technologies, and in particular, artificial intelligence (AI), machine learning (ML), and big data are have greatly impacted many sectors, from healthcare[Topol, 2019] and finance[Kelly et al., 2023] to transportation[Huang and Chen, 2020], education[Luckin and Holmes, 2016], manufacturing[Albukhitan, 2020], and more. As these technologies continue to advance, agriculture is emerging as a key sector for the applications of these advances.

The rapid increase in population and other environmental pressures are driving the need to produce food more effectively and sustainably. Much of modern agriculture uses several sensors and data from space, and provides timely and useful products related to crops, soil, weather, and machinery. When applied to AI and ML in these systems, farmers will potentially be able to better analyze all the data to make faster and more accurate decisions. Agriculture is moving in the direction of smart farming systems consisting of intelligent systems that can make suggestions in real-time, create efficiencies in resources, improve yield, and develop sustainable and resilient agricultural practices moving forward to address issues such as climate change. [Liakos et al., 2018]

1.3.1 AI in agricultural resource management

The implementation of artificial intelligence is addressing critical challenges in agriculture, as AI technologies can increase efficiency, improve crop yield, reduce costs, and decrease environmental footprint.[Eli-Chukwu, 2019]

1.3.2 Smart Irrigation Management

[Kumar et al., 2023] explain that smart irrigation systems incorporate built-in sensors that observe both the environmental factors coupled with the soil moisture and will make modifications to irrigation, as needed, based on the needs of each crop, thereby ensuring effective use of former resources while creating a lower impact on the environment.

1.3.3 Fertilizer Usage Management

By evaluating data from several sensors that measure soil moisture, temperature, pH, and nutrient levels, artificial intelligence (AI) enhances fertilizer delivery. Better nutritional uptake and less environmental impact result from this optimization. [Kumar et al., 2023]

1.3.4 Early Detection of Diseases and Pests Management

According to[Chin et al., 2023] a healthy crop normally depends on timely identification of pests and diseases. AI systems analyze the visual input from cameras and drones for signs of disease before they may spread, improving efficient use of inputs and reducing pesticide use more sustainably.

1.3.5 Soil management

Artificial intelligence improves soil management, reduces nitrate loss and improves soil texture predictions. Predicting soil texture, moisture content, and enzyme activity is possible by applying artificial intelligence methods, such as Artificial Neural Networks (ANN).[Zhao et al., 2009]

1.3.6 New perspectives for modern agriculture

The application of innovative technologies such as data analytics, machine learning, and precision farming is revolutionizing modern agriculture. By maximizing agricultural yields and improving sustainability, these advances are assisting farmers in making better decisions. Agricultural technology is advancing toward a more data-driven and efficient future by utilizing tools like crop yield prediction and recommendation systems to take on issues like resource management and food security.[Musanase et al., 2023]

1.3.6.1 Crop recommendation

[Balakrishnan et al., 2023] describe the Agricultural Crop Recommendation System created with machine learning is a software program that leverages artificial intelligence algorithms to enable farmers to select crops to plant in various regions based on the particular agricultural conditions. By pertaining, collecting, and processing signals of soils, climate, and other collected agricultural outputs to create personalized crop recommendations based on farmers specific situations at the local region. Each producer can simply access the recommended applications contiguous within the testing systems used either within web applications or mobile device approaches. The emergence of such systems can substantially increase crop output, diminish costs, and consequently add to the efficiency of farmer related agricultural practice. In addition, the Agricultural Crop Recommendation system using machine learning can totally redefine the agriculture space. Making prominent applications of data driven analysis for agronomy can assist growers to determines more informative decisions which enables them to generate maximum profitability while defining the capacity of agriculture with much simplified techniques and ethical operations, producing more consumable food overall.

1.3.6.2 Yield prediction

Predicting crop yields is a crucial challenge for national and regional decision-makers to make decisions quickly. Farmers may make decisions about what to plant and when to grow it with the aid of an accurate crop production prediction model. There are several methods for predicting agricultural production. Inaccurate estimates of agricultural output can result in lost money, wasteful use of resources, and challenges with strategic planning. Advanced prediction models are required to address this problem, notably ones that make use of machine learning methods. To produce accurate and timely projections, these models have to take into account a variety of factors, including pesticide usage, rainfall, and average temperature.[Xu et al., 2019]

1.4 Crop yield prediction

Yield predictions are key in addressing food security and improving agricultural planning. Data-driven models of yield prediction are increasingly used due to climatic, soil, and crop variation leading to issues with yield fprediction. This section briefly discusses the most commonly used prediction techniques, with an emphasis on the role that artificial intelligence is playing to improve effectiveness and flexibility.

1.4.1 Prediction methods

Predictive modeling in agriculture has increasingly relied on machine learning and deep learning techniques, which have demonstrated high accuracy and efficiency in yield estimation.

1.4.1.1 Machine learning

Machine learning (ML) is a branch of artificial intelligence concerned with developing algorithms to accurately model datasets such that they learn from data. In contrast to traditional computer programming, which specifies algorithms in terms of known features, machine learning (ML) specifies algorithms from aspects of data that allow for interesting combinations of features and weights that the user may not have imagined. [Choi et al., 2020]

There are four primary learning methods in ML, each tailored to different tasks, as shown in the table 1.1 below :

Learning Type	Data Type	Concepts	Tasks
Supervised Learning	Labeled	In supervised learning, the model is trained on a labeled dataset which relates input features (e.g., characteristics of housing) with a known target (e.g., housing prices). The main goal is to predict the target for new, unseen data.	Classification, Regression
Semi-supervised Learning	Labeled and Unlabeled	This approach utilizes both unlabeled and labeled data when labeling is costly or time-consuming. A small labeled portion guides the model to predict labels for the rest of the data.	Classification, Clustering
Unsupervised Learning	Unlabeled	Unsupervised learning aims to discover unknown patterns in data without predefined labels or targets. Common tasks include clustering and association rule mining.	Clustering, Association, Anomaly Detection
Reinforcement Learning	State-action-reward data	The agent learns by interacting with an environment, receiving rewards or penalties, and aims to maximize cumulative rewards over time.	Sequential Decision-Making, Control Systems

TABLE 1.1 – Machine learning types [Choi et al., 2020]

1.4.1.2 Deep learning

Deep Learning (DL) is a subfield of machine learning (ML) that represents complex relationships in data with artificial neural networks (ANN). ANNs utilize three or more layers of interconnected nodes (also known as neurons) to evaluate input features and can use hierarchical mechanisms, at scale, to process, recognize, and learn patterns from large-scale datasets. DL models typically outperform conventional ML models, especially in high-dimensional input applications (e.g., text, audio or images).While conventional machine learning has a strong dependence on hand-engineered feature extraction, DL leads to superior performance by automatically learning features from unlabeled or raw data. Consequently, DL models are particularly well suited for unstructured data, such as images or text. [Janiesch et al., 2021]

— **Time series analysis :** [Shumway et al., 2000] define time series analysis as the statistical methodology used to address problems associated with the analysis of data collected across different points in time. These problems arise due to the correlation between adjacent points in time which constrains the application of standard statistical methods which rely upon the independence of observations.

The Autoregressive Integrated Moving Average (ARIMA) model is often used for time series analysis. The ARIMA model is especially valuable for modeling and predicting future points in a series when the data displays trends or seasonality (not just random variation). This model consists of three components : moving average (MA), differencing (I), and autoregression (AR). RNNs (Recurrent Neural Networks) are also being increasingly utilized for forecasting time series data based on their capability to model complex datasets with significant long-term dependencies.[Akkem et al., 2023]

1.4.2 Crop yield prediction system

Predicting crop yields requires several steps. The first step is collecting data, in which we collect relevant data related to crop yield, weather data, soil data, and data related to crop management. This data provides the input to the prediction model. The next step is using different prediction strategies, including different algorithms in machine learning, deep learning, or statistical procedures among others to see the relative relationship between these factors and crop yield. After this, we take the training phase of the model to prepare the model to predict future yield using historical data to prepare the model for success. After training, the model takes on the evaluation phase, which evaluates the performance in a way that determines if the model can produce predictions with reliability. The end result of all these steps is to create a working system that can predict crop yield using factors related to environmental, agricultural, and management factors.

The figure 1.2 present the process of crop yield prediction system

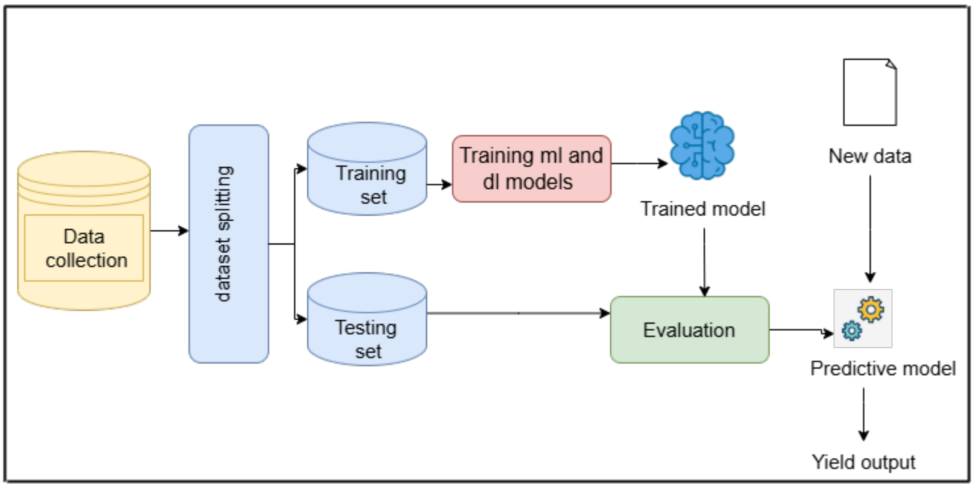

FIGURE 1.2 – Crop Yield Prediction Process

1.5 Conclusion

This chapter highlights the significance of agriculture to the human species, which has had a storied evolution through changes in practice and advances in technology. Agricultural artificial intelligence (AI) is transforming the food production system by providing new and creative mechanisms for estimating and optimizing crop economic productivity. Contextualizing the key indicators of agricultural production such as biotic content, soil health, and climate can develop better agricultural practices that improve productivity while reducing environmental costs. AI methodologies such as deep learning and machine learning are increasingly viable approaches to the modern sensibilities of agriculture and for improving resource management and decision-making. In the next chapter we will present our approach for making crop yield prediction.

Chapter 2

Proposed Approach : H_StackCYP Hybrid Stacking for Crop Yield Prediction

2.1 Introduction

Every growing season comes with a familiar uncertainty : how good or bad will the crop be this year ? This is not just important for the farmer or farm, this will have implications for food prices, supply chains, and perhaps even global food security. The ability to predict a crop yield ahead of time could save planning time, resources, and lost crop yield possibilities. However, accurate crop yield predictions are harder to accomplish than ever now that the weather patterns, soil conditions, and techniques around farming are changing as fast, if not faster, than our technology. In this chapter, we present our approach for making crop yield prediction smarter and more reliable to reflect the real world complexity of agriculture using data-driven yield estimation.

2.2 H_StackCYP : Architecture overview

Based on what we have learned in the first chapter, and the rich insights we discovered from the data, this chapter introduces the approach we followed to build our crop yield prediction model. The idea was to work with two datasets : one for crop recommendation and another for crop production. Since the production dataset was missing humidity, we first used the recommendation dataset to train different models and predict the missing humidity values. Once that step was completed, we cleaned and prepared the data. Afterward, we trained and tested several machine learning and deep learning models. Finally, we applicated the hybrid stacking methods to enhance the prediction accuracy.

2.2.1 Dataset description

During our research on crop yield prediction, we faced several challenges related to dataset availability. Most of the previous studies do not provide access to their datasets, which were never released publicly duo to privacy issues, and thus it was difficult for us to obtain a dataset with all the necessary features such as soil characteristics, weather conditions, and crop-specific factors. Most publicly available datasets had limited sizes or were missing some essential informations.

After detailed searching, we selected a comprehensive dataset from Kaggle named Crop Production Dataset. This dataset is adequate for our model work. it is of size 9.89 MB and contains 99848 instances with 12 important attributes presented in the table 2.1 below :

Variable	Description
State_Name	The name of the state where the crop is grown.
Crop_Type	Indicates the specific cropping season (Kharif, Rabi, whole year, or Summer).
Crop	The name of the crop cultivated (53 crops).
N, P, K	Levels of nitrogen, phosphorus, and potassium in the soil.
PH	Soil acidity level.
Rainfall	Measured rainfall in mm.
Temperature	Average temperature measured in degrees Celsius.
Area_in_hectares	The total land area (in hectares) under cultivation for the specific crop.
Production_in_tons	Total crop production.
Yield_ton_per_hec	Calculated yield measured in tons per hectare (target variable).

TABLE 2.1 – Crop Production Dataset features descriptions

This dataset covers agricultural data from across India and includes 53 different crops. This rich and diverse dataset, created by **Srihari Katare** and last updated two years ago, provided a solid foundation for building an accurate crop yield prediction model.

It comprehensive and contains many critical agricultural features, but it lacks one particularly important variable : humidity.

2.2.2 Step 1 : Feature engineering

We started by this first step in our approach, we focused specially on prediction humidity from the secondary dataset (Crop Recommendation Dataset) we followed this process :

— This dataset was already clean, well-structured, and required no preprocessing, which allowed us to jump straight into modeling.

— We selected the following variables as inputs : temperature, ph, rainfall, N, P, and K. Humidity was selected as the target variable.

— Then we split the data into training and testing sets, with 80% used to train the model and 20% reserved for evaluation. This ensures that the model is trained on the majority of the data while being tested on unseen data for a fair performance assessment.

— As the next step, we applied four model of machine learning :

2.2.2.1 Training Models

We trained four machine learning models because our dataset is small, which makes it unsuitable for training deep learning models. Machine learning algorithms usually perform better than deep learning models on small datasets. We selected different types of models (tree-based and kernel-based) to compare their performance and identify the most suitable one.

— **Support Vector Regressor (SVR) :**

Support Vector Regression (SVR) is a supervised learning method derived from Support Vector Machines (SVM) for modeling continuous outputs as an optimization problem based on margins [Basak et al., 2007]. The SVR algorithm introduces an ε-insensitive tube to allow a small tolerance for prediction error, focusing optimization on minimizing large deviations. This makes SVR naturally robust in noisy environments such as meteorological datasets.

SVR can model complex nonlinear mappings in high-dimensional spaces using kernel functions, making it well-suited for forecasting humidity when input features include temperature, rainfall, and soil nutrients.

This algorithm allows the model to generalize effectively, even when working with sparse and noisy meteorological data. As a result, Support Vector Regression (SVR) proves to be a valuable approach for accurately predicting relative humidity, which plays an essential role in various agricultural forecasting tasks.

Random Forest (RF)

Random Forest is used for predicting relative humidity due to its reliability and robustness. It can effectively handle complex datasets with multiple features and nonlinear relationships [Cutler et al., 2012].

Humidity prediction is influenced by various meteorological factors such as temperature, precipitation, and other environmental variables, which often interact in complex and nonlinear ways. Random Forests are particularly well-suited for this kind of prediction task because :

— They can capture and explain complex interactions between features such as temperature and precipitation, and how these affect humidity.

— They are less prone to overfitting because they build multiple decision trees and aggregate their results to provide more stable and accurate predictions.

Below is the algorithm for predicting humidity using Random Forest :

Gradient Boosting Regressor

Gradient Boosting is a robust method in machine learning used for tasks like predicting numerical values (regression) and classification [Chen and Guestrin, 2016]. For humidity prediction, Gradient Boosting is a strong candidate because :

— It gradually reduces the errors of previous models, improving overall prediction accuracy.

— It captures both linear and non-linear relationships between meteorological variables.

— It allows control over learning via parameters like learning rate and tree depth.

XGBoost Regressor

XGBoost (Extreme Gradient Boosting)is an improved version of Gradient Boosting, designed to be fast and highly efficient.[Chen and Guestrin, 2016]. It is particularly useful for predicting humidity because :

— He can effectively manage large amounts of data.

— It automatically handles missing values.

— It includes regularization to prevent overfitting.

2.2.2.2 Evaluation Metrics

After training our dataset with different models, now we move into the evaluation phase where we evaluate our models performance on new unseen data which is the test set and its ability to make predictions on data that is separate from the training set using performance metrics like Root Mean Squared Error (RMSE), coefficient of determination (R-squared or R^2), Mean Absolute Error (MAE), and Mean Squared Error (MSE).

1. **Mean Squared Error (MSE)**
   The squared difference between real and predicted values. It shows how close the line of best fit is to the set of points. If the MSE is small and closer to 0, the prediction is more accurate [Tatachar, 2021]. The formula for Mean Squared Error is given as :

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{2.1}$$

   MSE on humidity prediction quantifies the distance of the predicted humidity values from their actual, and weighs larger errors more than smaller ones. For example, if the model predicts 60% instead of the actual 80%, the squared error (400) penalizes this heavily. This helps highlight large errors in predicting given humidity levels.

2. **Mean Absolute Error (MAE)**
   It is the mean of the absolute differences between actual and predicted values. The difference between MAE and MSE is that MAE takes absolute differences, whereas MSE

takes the squares of the differences between expected and actual values [Tatachar, 2021]. The formula for MAE is :

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{2.2}$$

MAE shows us the mean size of the error in the same unit related to humidity (%). It treats all errors equally ; if the difference is small, it treats it the same as if the difference is large. It gives the average size of how many percentage points the model is away from the true humidity. For example, if MAE = 2.3, our humidity predictions are on average $\pm 2.3\%$ off.

3. **Root Mean Squared Error (RMSE)**
   RMSE is a widely used metric for regression problems. It is defined as the square root of the mean squared differences of the predicted and actual values. RMSE tells us how big prediction errors are on average. However, since it uses the squared differences, it weighs larger errors more than smaller errors. The lower the RMSE, the more credible the model, and the better the predictions are to the values in the test dataset [Tatachar, 2021]. The formula for RMSE is :

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{2.3}$$

   So, an RMSE of 6.1 means the model predictions on average are 6.1% away from the actual humidity values, which is a good performance outcome when predicting environmental variables.

4. **R-Squared (R^2)**
   The coefficient of determination is known as R^2. R-squared calculates the amount of variance in the dependent variable that can be attributed to the independent variables. It is an indication of the goodness of fit of the regression model which reflects how closely our predicted values fit the observed values [Tatachar, 2021]. The formula for R-Squared is :

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \tag{2.4}$$

   where :

   — y_i : Observed values
   — $\hat{y}_i$: Predicted values
   — n : Number of predictions
   — $\bar{y}$: Mean of observed values

2.2.3 Step 2 : Preprocessing

This section presents the components of the preprocessing phase. The preprocessing phase consists of four main stages, including handling missing values and outliers.

2.2.3.1 Data Cleaning

During this phase, we reviewed the dataset for errors and missing values. No additional cleaning steps were necessary, as the dataset was already clean. However, we took an additional step to remove the **Production** column. Since we already have the **Yield** column, which contains the yield calculated from production and area [FAOSTAT, 2024], keeping both would cause redundancy. To keep our model clean and avoid unnecessary complications, we removed the redundant column. The yield is defined as :

$$\text{Yield} = \frac{\text{Production}}{\text{Area}} \tag{2.5}$$

Furthermore, we renamed columns for semantic accuracy. The column originally labeled **CROP TYPE** was renamed to **SEASON**, because it contained values such as "Kharif" and "Rabi" which refer to planting seasons rather than crop varieties.

Handling Outliers

Outliers are extreme values that differ significantly from the rest of the data. They can affect statistical analyses and impact the performance of machine learning models. Therefore, it is important to identify and handle outliers during preprocessing to ensure reliable and stable predictive results [Tukey et al., 1977].

Before training the models, we performed outlier detection and treatment to improve data quality. We used the Interquartile Range (IQR) method to detect outliers by defining lower and upper bounds for normal values. Instead of removing outliers, we capped them at the respective bounds to reduce their impact while maintaining the dataset's structure and size.

The IQR-based outlier detection is defined by the following equations :

$$Q_1 = \text{First Quartile (25th percentile)} \tag{2.6}$$

$$Q_3 = \text{Third Quartile (75th percentile)} \tag{2.7}$$

$$IQR = Q_3 - Q_1 \tag{2.8}$$

$$\text{Lower Bound} = Q_1 - 1.5 \times IQR \tag{2.9}$$

$$\text{Upper Bound} = Q_3 + 1.5 \times IQR \tag{2.10}$$

Values are considered outliers if :

$$x < \text{Lower Bound} \implies x \text{ is an outlier}$$

$$x > \text{Upper Bound} \implies x \text{ is an outlier}$$

2.2.3.2 Encoding Categorical Data (Label Encoding)

Encoding categorical data is an essential preprocessing step in machine learning as it converts non-numerical categories into a numerical format that can be understood by models and enables the models to make predictions based on the transformed categorical features [Hancock and Khoshgoftaar, 2020], such as *crop name*, *seasons*, and *state name*. Label encoding is a common technique of categorical encoding, where each unique category is assigned a hard coded integer value.

2.2.3.3 Normalizing Data (Standard Scaler)

Data normalization is an essential preprocessing step in machine learning that transforms feature values on to a common scale without changing differences in the ranges of values. This enables algorithms to converge faster and produce more accurate results.

StandardScaler is an important and commonly used normalization method that standardizes features by removing the mean and scaling to unit variance. The transformation is defined as follows [Pedregosa et al., 2011] :

$$Z = \frac{x - \mu}{\sigma} \tag{2.11}$$

where :

— x = original value

— μ = mean

— σ = standard deviation

We applied standard normalization to the following columns : N, P, K, Humidity Calculated, Temperature, pH, Area, and Rainfall. This step was necessary because each of these variables measures different quantities and uses different units and scales. Without normalization, distance-based algorithms or gradient-based optimization methods might not perform well.

2.2.4 Step 3 : Yield Prediction

Once we have gone through the preprocessing step, which ensures the dataset is clean, formatted, and well-organized for analysis, we will begin one of the more significant steps : prediction. In this step, we will train machine learning and deep learning models on the prepared data to predict outcomes in this case, predicting crop yield. The workflow in this step typically includes :

2.2.4.1 Data Splitting

It is crucial to ensure that we can properly evaluate a machine learning model before we start the training process [Muraina, 2022]. Therefore, we will split our dataset into two parts one for training, and one for evaluating the performance of the model on unseen data. While giving us a more realistic measurement of how the model performs in real-life scenarios.

Figure 2.1 presents the portion of each set.

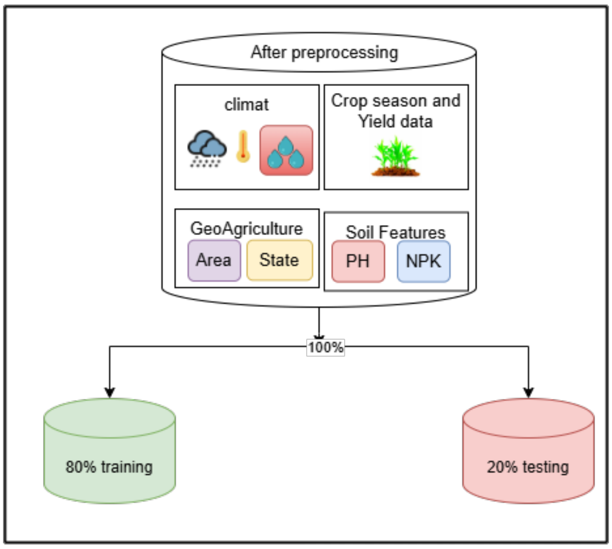

FIGURE 2.1 – Data Spliting

2.2.4.2 Training Multiple Models

We choose both machine learning models (XGBoost, Decision Tree, Random Forest, Linear Regression) and deep learning models (Multi-Layer Perceptron(MLP), Deep Neural Network(DNN), Convolutional Neural Network(CNN)) to train the model and see which model will give the best result.

1 Machine Learning Models

Random Forest (RF) : Random Forest algorithm is an ensemble learning technique. When applied to crop yield prediction, it builds multiple decision trees on random subsets of the data and random subsets of features. Each tree learns different patterns from the dataset, helping reduce overfitting and improve generalization.

In supervised regression tasks such as predicting **Yield_ton_per_hec**, each tree outputs a numerical prediction (estimated yield). The Random Forest predicts by averaging the outputs of all trees, which stabilizes predictions and reduces sensitivity to noise.

Random Forest is especially useful in agricultural prediction because it can handle :

- Complex interactions among soil, weather, and crop type data sources,
- Mixed data types (categorical and numerical),
- Some outliers and missing values.

Linear Regression (LR) : Linear Regression (LR) was used in crop yield forecasting to perform a statistical analysis of agricultural features (e.g., Nitrogen, Phosphorus, Potassium content, pH, Rainfall, Temperature, etc.) versus the crop yield per hectare (dependent variable). LR assumes that the yield can be modeled as a linear combination of the inputs.

LR is especially useful as a baseline model because it is simple and easy to interpret. It helps in understanding in what way an input feature influences the output yield.

Decision Tree (DT) : Regarding the crop yield prediction, we used the recommended Decision Tree model to split our dataset accordingly into groups based on agricultural features (N, P, K, Rainfall, Temperature, Humidity, pH, etc.) in order to predict yield per hectare.

The Decision Tree algorithm recursively splits the dataset into subsets using decision rules (e.g., if rainfall > 120 mm, if pH 6.5). This process aims to create groups that are as homogenous as possible regarding the target variable yield_ton_per_hec.

At each internal node of the tree, a decision based on a single feature is applied to the dataset. Each leaf node represents the predicted yield value. In regression trees, it is usually the average of the training samples/data that reached that leaf. In modeling crop yield, Decision Trees are appreciated for their simplicity and transparency because if features or parameters influence crop yield, Decision Trees will capture such trends.

Decision Trees are also attractive to agricultural scientists because, along with transparency, there is flexibility to model crop yield as a non-linear relationship with both feature variables and crop yield.

Gradient Boosting : In crop yield prediction, Gradient Boosting is a well-known and powerful ensemble learning technique that combines a sequence of regression trees. With Gradient Boosting, we have the capability to generate a sequence of trees where each new tree attempts to fix the errors from the past trees. With Random Forest, we averaged the predictions from the trees but with Gradient Boosting, we will take the sum of the outputs from every tree, beginning with a simple prediction (e.g., the average yield) and adding trees sequentially that predict relative residual errors (predictions differences between true yield and predicted yield). Trees are trained to minimize a loss function when trained using gradient descent. This method is particularly useful for modelling complex and non-linear relationships between agricultural features (e.g., Rainfall, Temperature, pH, N, P, K, etc.) and yield.

XGBoost (eXtreme Gradient Boosting) : XGBoost is a fast and scalable implementation of gradient boosting that excels in crop yield prediction tasks. It builds an additive model by training a series of regression trees, where each new tree attempts to correct the residual errors of the combined previous trees.

XGBoost improves upon conventional Gradient Boosting with several advanced techniques valuable for agricultural datasets :

- **Sparsity-aware learning :** Handles missing values and sparse features common in practical crop datasets.

— **Regularization :** Uses L1 and L2 penalties on tree complexity to prevent overfitting.
— **Parallel processing :** Accelerates training time, beneficial for large-scale datasets.
— **Weight quantile sketch :** Enables efficient and accurate tree splits even with continuous-valued measurements like rainfall, temperature, and soil pH.

Due to its scalability and capability to capture complex non-linear relationships between environmental variables and crop yield, XGBoost is a reliable model for crop yield prediction.

2 Deep Learning Models

Deep Neural Network (DNN) : Deep Neural Networks (DNNs) are an extension of neural networks that incorporate multiple hidden layers, allowing the model to learn abstract and high-level representations from complex data. In crop yield prediction, DNNs learn how input features such as Rainfall, N, P, K, Temperature, pH, Humidity, and Area interact to influence yield, even in ways that are not explicitly measurable.

Compared to simpler architectures like MLPs, DNNs may include :

— More hidden layers (e.g., 5 or 10).
— Dropout layers to prevent overfitting.
— Hybrid structures combining dense, convolutional, and pooling layers.

Their depth and flexibility make them suitable for modeling nonlinear, high-dimensional problems across diverse crops, soil types, and climate conditions.

Multi-Layer Perceptron (MLP) : MLPs are deep learning models that can learn complex, nonlinear relationships between input variables (e.g., Rainfall, N, P, K, pH, Temperature, Humidity) and crop yield.

An MLP consists of :

— Input layer : receives agricultural features.
— One or more hidden layers : capture complex feature interactions.
— Output layer : provides the predicted yield.

MLPs operate through a forward pass (input hidden output), followed by backpropagation to update weights and minimize error.

Convolutional Neural Network (CNN) : While CNNs are mainly used in image processing, 1D CNNs can be applied to structured tabular data like crop yield datasets. Each row in the dataset is treated as a 1D sequence, enabling the CNN to learn local feature dependencies.

The convolutional layers scan across input features (e.g., N, P, K, temperature, humidity, pH), extracting patterns and combinations. Pooling layers reduce dimensionality while retaining essential features, and dense layers finalize the prediction.

2.2.4.3 Evaluation metrics

After training our dataset with different models, we evaluated their performance using the same regression metrics that we used in humidity prediction, which are : RMSE, MAE, MSE, and R^2.

And the algorithm below summarized the process of yield prediction

2.2.4.4 Hybrid Stacking Ensemble For Crop Yield Prediction

When it comes to building a reliable predictive system you won't always have to use a single model. In reality, data from real world situations are messy, unknown and filled with hidden patterns that unfortunately can't be studied by a single mechanism alone. That's why, when multiple models are combined, each with its own way of looking at the data, the results are far more robust. Thats what ensemble learning is all about. In our case, we used stacking as a smart ensemble method where the predictions of multiple models are combined in a final model where the model learned how to make the best possible choice. For this project, it helped us make close to a more reliable and powerful crop yield prediction system.

1. **Stacking Method**
   Stacking or stacked generalization is an advanced ensemble learning framework for using multiple predictive models to jointly optimize the overall result. This ensemble approach is initially done by each baseline model on the same data set, and then collected and used as inputs by a meta-learner which is trained to optimize the output obtained.[Tang et al., 2021]
   In our study, we used a method called hybrid stacking for predicting crop yields. This technique combines both traditional machine learning models and deep learning models.

   The architecture devided into two levels :
   Training Set Preparation : The training set (80%) is split using K-Fold Cross-Validation (choice : K=5), where one fold is utilized as a validation set for every run and all the other folds are utilized to train.
2. **Level 0 (Base Learners Training with K-Fold CV)** We used a hybrid ensemble that combines three machine learning (Random forest, Xgboost, Gradient Boosting) and three deep learning models(Multi-Layer Perceptron, Deep Neural Network, Convolutional Neural Network), Each of these models is trained independently using K-Fold Cross-Validation, where they generate out-of-fold predictions on the validation folds. These predictions simulate how the model would behave on unseen data.
3. **Level 1 (Meta Learners)** The out-of-fold predictions from all six base models are concatenated to form a new feature set. As result , each instance is now represented by six features (P1 to P6), corresponding to the predictions of each base model. We applied a

Random Forest Regressor as meta learners , wich was trained on the new feature set. This model learns how to combine the outputs of the base models to produce a final prediction.
Once the meta-model is trained, it can be used to make predictions on unseen data (20%test set). The final model is evaluated using standard regression metrics (R^2, RMSE, MAE, MSE), and the result is a predictive model that leverages the strengths of multiple diverse learners. Further, using added advantage of hybrid stacking ensemble we built a more accurate and robust prediction model. This shows the benefits of models that are inspired from the "wisdom of the crowd" through collaboration.
In the next chapter we will focus on presenting and analyzing the results obtained from the models developed in this work.

2.3 Computing environments

We used two different computing environments for implementation of our approach. The first was fully featured system, with Intel i5 (11th generation) processor, 16 GB RAM and 256 GB SSD. The second system had Intel i3 (6th generation), 8 GB RAM and SSD. Furthermore, part of the work we did, namely data exploration and experimentation with models, was done on Kaggle Notebooks with integrated GPU support and pre-installed libraries. We developed the entire implementation in Python because this programming language has a rich ecosystem of libraries and tools that allow for easy data analysis, and apply machine learning and deep learning.

2.4 Results

The evaluation process is based on two experimental scenarios designed to compare the performance of different model categories and assess the contribution of humidity prediction.
Each scenario was evaluated using consistent metrics, including R^2, RMSE, MAE, and MSE. The results provide a structured comparison between different modeling strategies and demonstrate the potential advantages of combining multiple models and incorporating additional environmental variables.

2.4.1 Scenario 1 : Humidity Prediction Result

In the first scenario we focused on humidity prediction, we used the crop recommendation dataset and the prediction of humidity has shown extremely strong performance with an R^2 of 0.9305 for XGBoost, meaning that the predicted and actual values were in excellent agreement, and we have validated the results at the national meteorological office in Dar El-Beida. The figure 2.2 above shows the actual humidity values in the dataset against the predicted values. The predicted humidity followed the actual humidity trends closely throughout the different samples and demonstrated an accurate and reliable model.

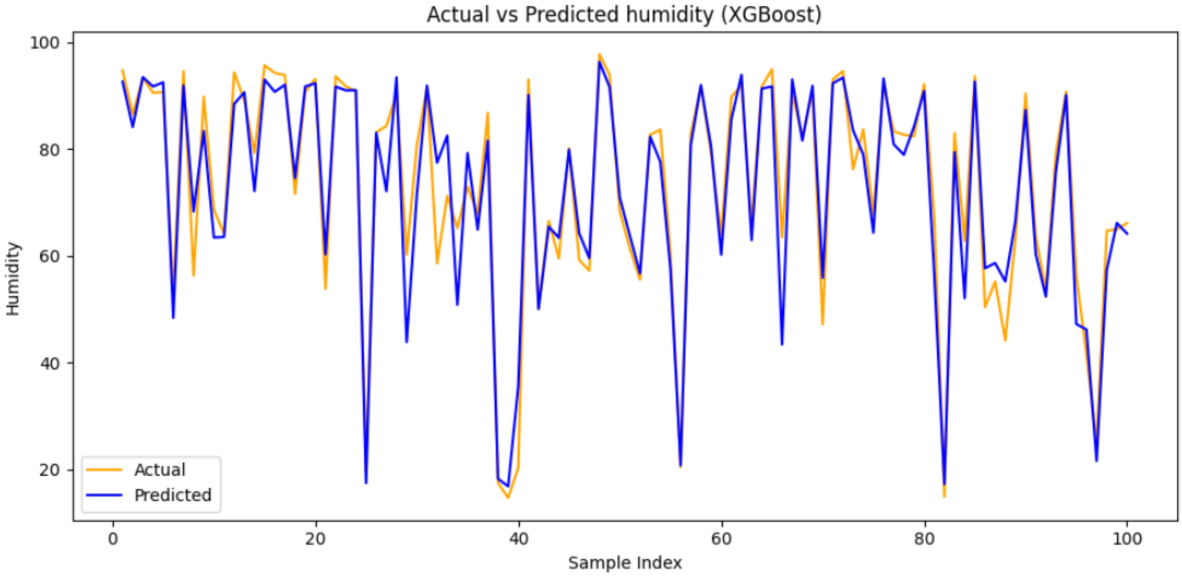

FIGURE 2.2 – Comparaison of actual and predicted humidity

2.4.1.1 Hyperparameter Tuning for Humidity Prediction

For this study, the hyperparameters for each machine learning model used to predict humidity were selected automatically with Randomized Search Cross Validation (RandomizedSearchCV). RandomizedSearchCV allows models to be evaluated quickly using a set number of random combinations from the identified distributions, making it possible to run in less time. Each model, including Random Forest, XGBoost, SVR, and Gradient Boosting, was tuned using 3-fold cross-validation with 20 random configurations for the model, in the goal of maximizing the R^2 Square and minimizing the other metrics.
The table 2.2 below summarizes the best parameters found for each algorithm :

Model	**Best Parameters**
Random Forest Regressor	n_estimators=56, max_depth=13, max_features='sqrt', min_samples_leaf=1, min_samples_split=4
XGBoost	n_estimators=114, learning_rate=0.0692, colsample_bytree=0.8821, reg_alpha=0.0078, reg_lambda=1.4234, subsample=0.8185, gamma=0.0827, max_depth=7
Gradient Boosting	n_estimators=127, learning_rate=0.0344, max_depth=7, max_features='log2', min_samples_leaf=3, min_samples_split=4
Support Vector Regressor	C=6.0865, epsilon=0.0412, gamma='scale', kernel='rbf'

TABLE 2.2 – Optimized hyperparameters for different regression models

2.4.1.2 Model Performance Evaluation

In order to evaluate the performance of different regression models (the respective performance generation of our models for humidity predictions), four basic assessment metrics were calculated : Coefficient of Determination (R^2), Root Mean Square Error (RMSE), Mean Squared Error (MSE), and Mean Absolute Error (MAE). These metrics estimate the accuracy and consistency of the model output..

The comparative results from the test set using XGBoost, Gradient Boosting, Random Forest, and Support Vector Regression (SVR), are shown in table 2.3

Model	R^2 ↑	RMSE ↓	MSE ↓	MAE ↓
XGBoost	**0.9305**	**6.1160**	**37.4049**	**4.0322**
Gradient Boosting	0.9283	6.2127	38.5975	4.1244
Random Forest	0.9255	6.3293	40.0602	4.2146
SVR	0.6732	13.2597	175.8209	9.7903

TABLE 2.3 – Performance metrics of different models

The charts below 2.3 compares the results of four regression models based on the following metrics. In the chart, the R^2 measure is expressed as percent values, which indicates the percentage of variance explained by each model. Lower RMSE, MSE, and MAE values indicate better predictive accuracy. The grouped bar chart is a great way to visually compare the models' ability to predict humidity.

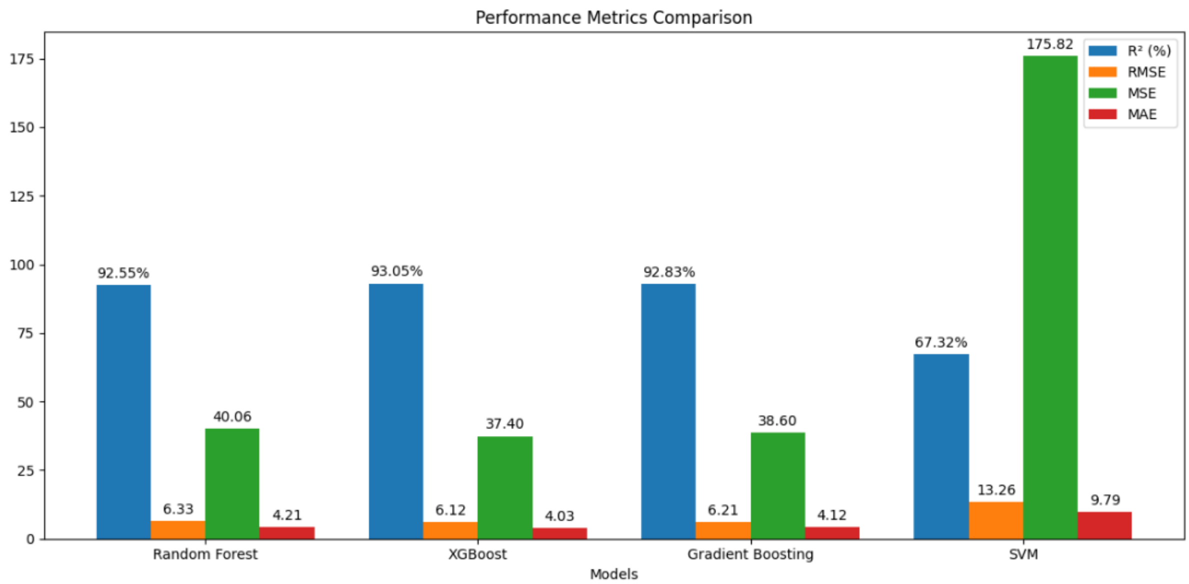

FIGURE 2.3 – Performance Metrics Comparison Between The Models

To evaluate the predictive performance of the machine learning models, the table 2.4 compares the actual with the predicted humidity values for the Random Forest, XGBoost, Gradient Boosting, and Support Vector Regressor (SVR) models. The table presents a selection of 10 representative test samples. Each value highlights how close (or far) those predictions of different models are to the actual values.

Samples	Actual values	Random Forest	XGBoost	Gradient Boosting	SVM (SVR)
1	94.73	**92.99**	92.12	92.49	92.90
2	86.52	85.05	85.02	84.57	**86.60**
3	93.35	92.48	**93.42**	92.99	63.35
4	90.56	89.47	92.26	**90.21**	50.39
5	90.70	92.25	**92.19**	**92.10**	94.82
6	52.35	**50.04**	48.42	49.33	79.41
7	94.58	91.93	91.91	91.76	**92.76**
8	56.33	**60.82**	65.30	68.59	64.18
9	89.81	80.49	**82.98**	82.37	64.24
10	68.65	62.37	**63.42**	**63.60**	58.02

TABLE 2.4 – Comparison of predicted values by different models against actual values

The models all tend to perform reasonably well on most samples, particularly Random Forest and XGBoost, whose predictions are relatively consistent and close to the actual value. However, some models, such as SVR, show larger distances from the actual predictions on some samples (in particular, samples 3, 4, and 6) indicating a lower level of robustness with respect to those samples. This tabular comparison serves as an example of the individual-level evaluation which corresponds to the summary performance metrics in the last figure.

2.4.2 Scenario 2 : Yield Prediction Results

In the second scenario of our research, we focus on yield prediction under two conditions : one without humidity and another with humidity, building upon the results obtained from the preceding humidity prediction scenario. We applied the XGBoost model to the second dataset in order to estimate the humidity values. These predicted values were then used as additional input features for yield prediction. After incorporating humidity, we applied various machine learning and deep learning models to evaluate the impact of this additional feature on prediction performance. Through this approach, we examine how including humidity as a feature in the model affects its performance.

2.4.2.1 Hyperparameter Tuning for Yield Prediction

For this step, the hyperparameters for each model used to predict the yield were automatically selected with Cross-Validation Randomized Search (RandomizedSearchCV). For machine learning models, we used 5-fold cross-validation with 30 random configurations for the model. For deep learning models, the random search was run for 10 iterations with 3-fold cross-validation.

Tables 2.5 and 2.6 summarize the best parameters found for each algorithm.

Model	Best Parameters
Random Forest Regressor	n_estimators=300, min_samples_split=10, min_samples_leaf=1, max_features='sqrt', max_depth=30, bootstrap=True
XGBoost	subsample=0.8, n_estimators=200, min_child_weight=3, max_depth=9, learning_rate=0.05, gamma=0.1, colsample_bytree=0.6
Gradient Boosting	subsample=1.0, n_estimators=200, min_samples_split=10, min_samples_leaf=2, max_depth=7, learning_rate=0.1
Decision Tree	min_samples_split=2, min_samples_leaf=2, max_features=1.0, max_depth=15
Linear Regression	solver='saga', alpha=10.0, max_iter=10000, fit_intercept=True

TABLE 2.5 – Optimized hyperparameters for different machine learning models

Model	Best Parameters
Deep Neural Network (DNN)	layers=[122, Dropout(0.0115), 57], activation='relu', output_activation='linear', optimizer=Adam(learning_rate=0.0005667), loss='mean_squared_error', batch_size=16, epochs=100, patience=15
Multi-Layer Perceptron (MLP)	layers=[147, 59], activation='tanh', output_activation='linear', optimizer=SGD(learning_rate=0.0062), loss='mean_squared_error', batch_size=64, epochs=100, patience=15
Convolutional Neural Network (CNN)	Conv1D(filters=80, kernel_size=4, activation='relu'), Dropout(0.2), Dense(224, activation='relu'), output_activation='linear', optimizer=Adam(learning_rate=0.0002968), loss='mean_squared_error', batch_size=16, epochs=100, patience=15

TABLE 2.6 – Optimized hyperparameters for different deep learning models

2.4.2.2 Model Performance Evaluation

In order to evaluate the performance of the models, four basic assessment metrics were calculated : Coefficient of Determination (R^2), Root Mean Square Error (RMSE), Mean Squared Error (MSE), and Mean Absolute Error (MAE). These metrics estimate the accuracy and consistency of the model output..

The figures 2.4, 2.5, 2.6, 2.7, 2.8, 2.9, 2.10, 2.11, 2.12 below demonstrate a comparison between actual and predicted yields for each model. In these plots, actual target values are on the x-axis, while predicted values from the models are shown on the y-axis. The red dashed line represents perfect prediction (predicted values exactly equal to actual values). Points farther from the diagonal indicate less accurate predictions, while points closer to the diagonal indicate higher accuracy.

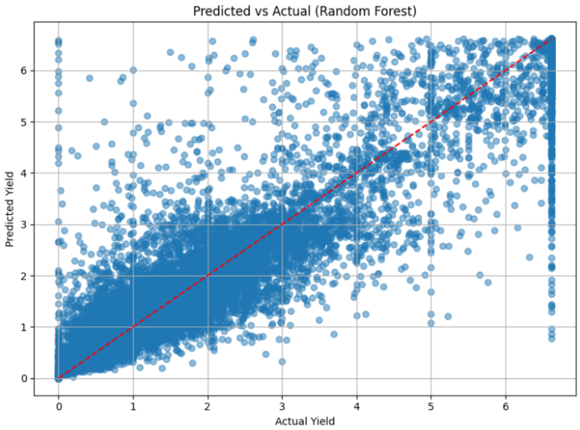

FIGURE 2.4 – Comparison between actual and predicted yield using Random Forest

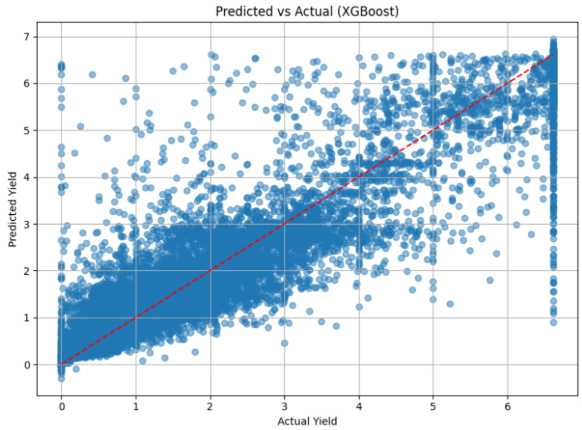

FIGURE 2.5 – Comparison between actual and predicted yield using XGBoost

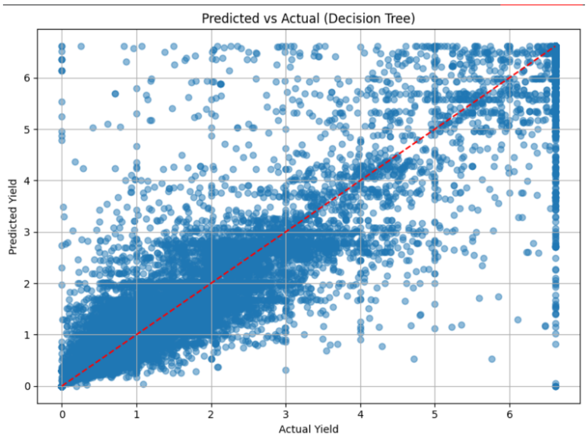

FIGURE 2.6 – Comparison between actual and predicted yield using Decision Trees

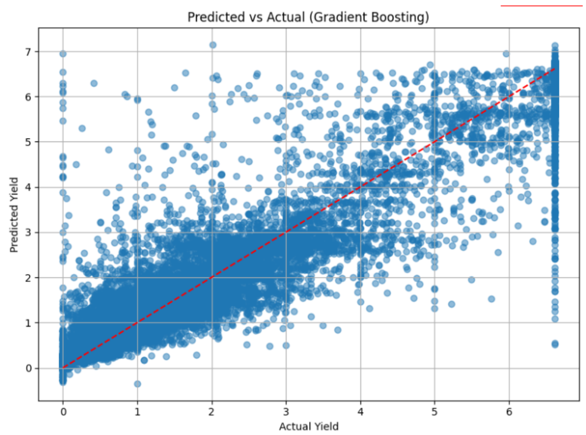

FIGURE 2.7 – Comparison between actual and predicted yield using Gradient Boosting

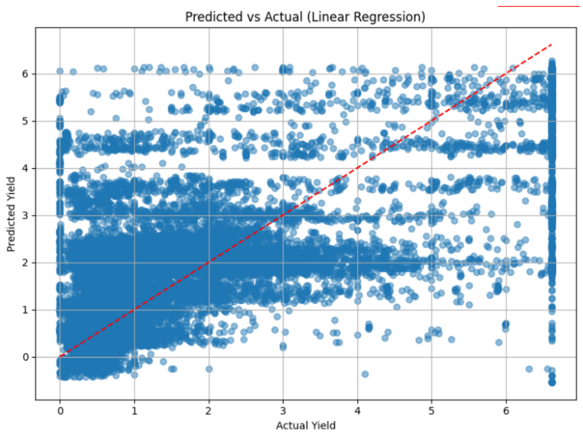

FIGURE 2.8 – Comparison between actual and predicted yield using Linear Regression

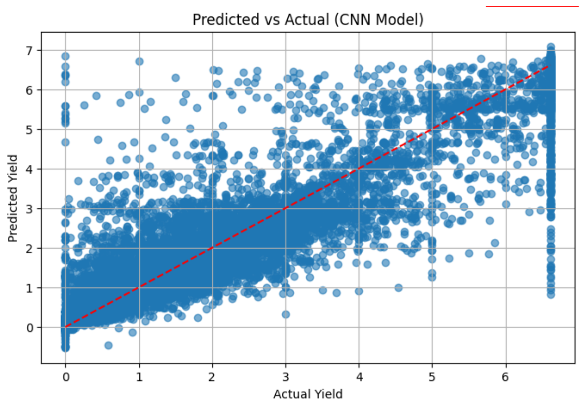

FIGURE 2.9 – Comparison between actual and predicted yield using Convolutional Neural Network

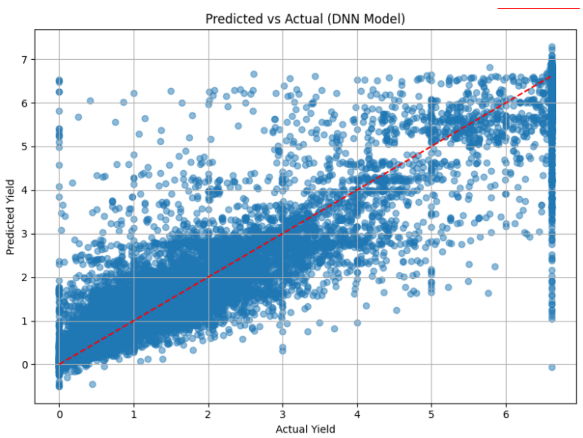

FIGURE 2.10 – Comparison between actual and predicted yield using Deep Neural Network

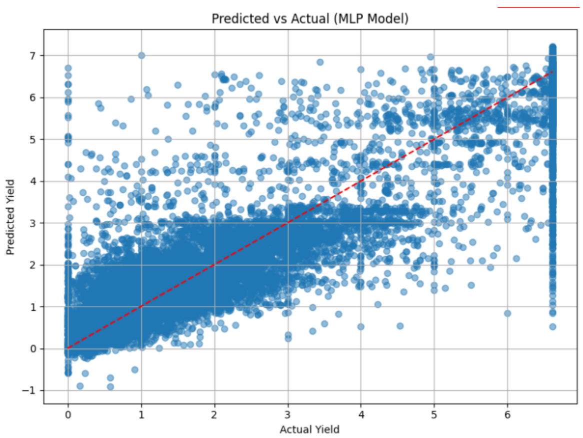

FIGURE 2.11 – Comparison between actual and predicted yield using Multilayer Perceptron

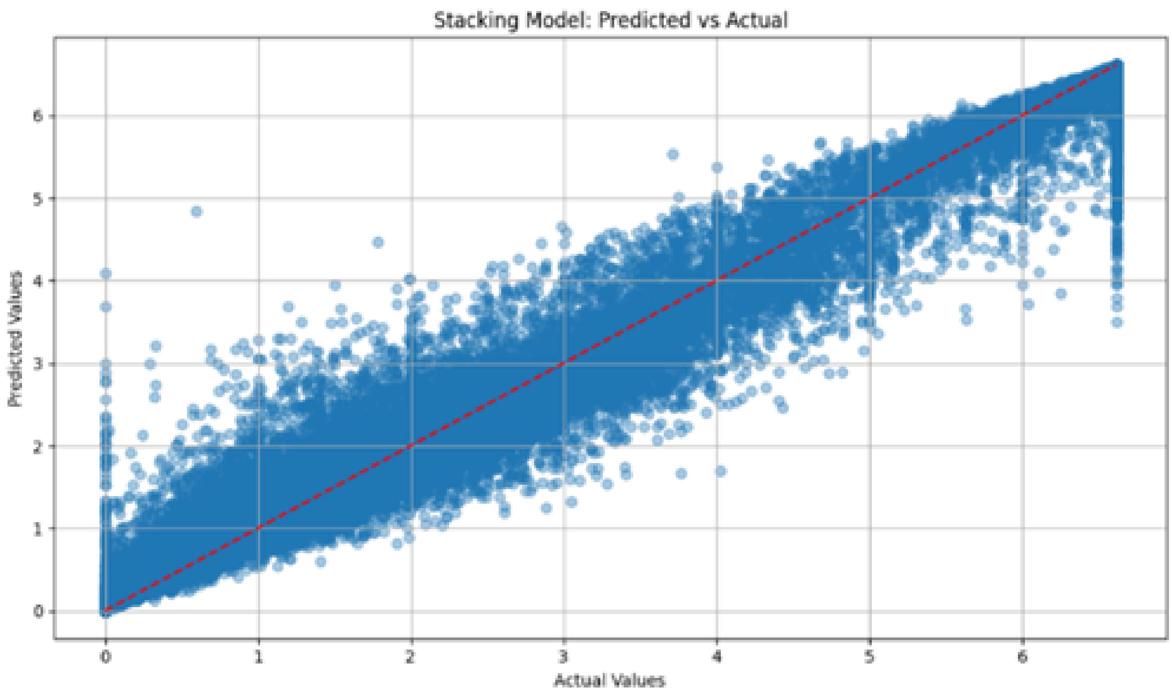

FIGURE 2.12 – Comparison between actual and predicted yield using Stacking method

And the figures 2.13, 2.14, 2.15, 2.16 below present respectively a visual comparison of the performances of different regression models according to several evaluation metrics : R^2, RMSE, MAE, MSE. Each bar illustrates the error or accuracy associated with a models, allowing us to visualize the accuracy and errors involved .

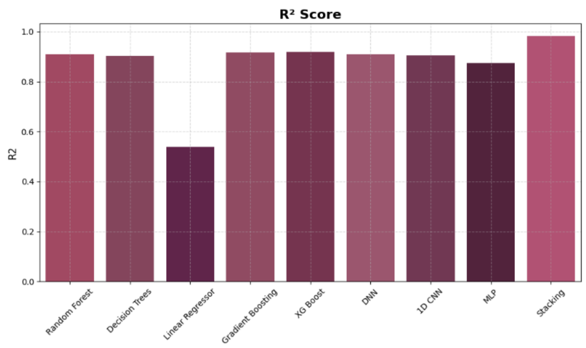

FIGURE 2.13 – R Square analyses

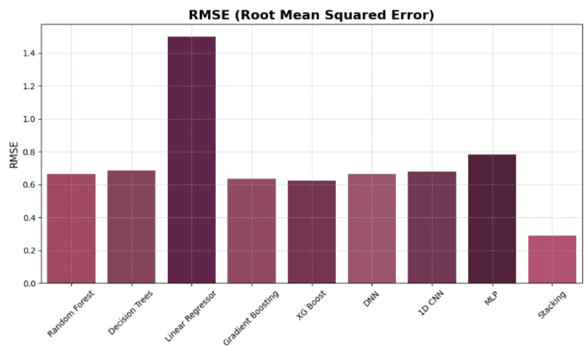

FIGURE 2.14 – Root Mean squared error analyses

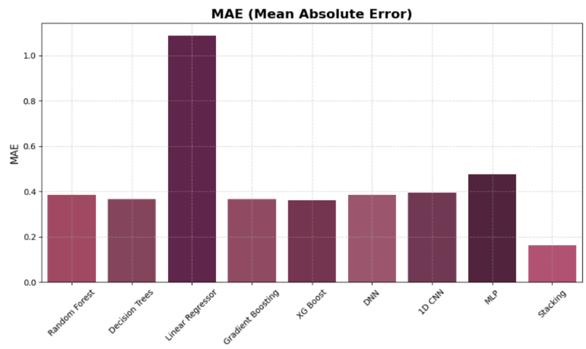

FIGURE 2.15 – Mean absolute error analyses

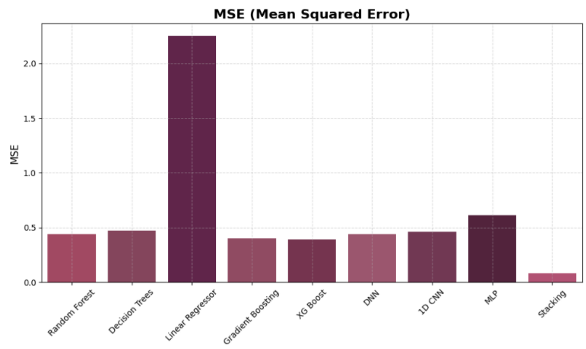

FIGURE 2.16 – Mean squared error analyses

The tables 2.7 and 2.8 below summarizes the difference between actual and predicted values for machine learning models, deep learning models and stacking ensemble method :

Sample	Actual Values	GB	LR	RF	XGBoost
0	0.104906	**0.055627**	3.236946	0.265154	0.268819
1	2.250859	**2.213442**	1.961476	**2.231512**	2.129974
2	1.000000	**1.126436**	1.867303	1.181216	**1.129299**
3	6.613910	6.591651	4.624731	**6.606687**	6.594715
4	2.245011	3.155975	**2.314691**	2.895443	3.115684
5	0.483193	0.839691	**0.418937**	0.761073	0.805020
6	1.204970	**1.278790**	1.766575	0.979098	1.332982
7	6.613910	6.530577	5.899981	**6.613609**	6.561521
8	1.022346	1.160365	2.252969	**1.144323**	1.151479
9	6.413793	5.606979	3.495609	**5.805297**	5.551867

TABLE 2.7 – Comparison of actual crop yield values with predictions from machine learning models

Sample	Actual Values	DNN	MLP	CNN	Stacking Model
0	0.104906	**0.118082**	0.306602	0.479485	0.175259
1	2.250859	2.054483	1.922152	2.411863	**2.325872**
2	1.000000	1.234164	1.387191	**1.069963**	**1.057271**
3	6.613910	6.550564	6.486544	6.506650	**6.613910**
4	2.245011	3.241740	3.195954	3.222618	**2.560243**
5	0.483193	0.784480	0.826956	0.802414	**0.557368**
6	1.204970	1.290028	**1.216801**	1.479630	0.935646
7	6.613910	6.457977	6.575888	6.394332	**6.611297**
8	1.022346	1.189617	1.070619	0.919650	**0.982265**
9	6.413793	5.595345	5.065881	5.508562	**6.189252**

TABLE 2.8 – Comparison of actual crop yield values with predictions from deep learning models and stacking method

2.5 Analysis and Comparative

2.5.1 Comparative Analysis and Discussion of Machine Learning and Deep Learning Models Results

Based on the results we have obtained, tree-based models, including Random Forest, Gradient Boosting, and XGBoost, significantly outperform the more traditional models such as Linear Regression. While XGBoost performs the best with an R^2 of 0.9196 and RMSE of 0.6267, it can be seen that the XGBoost model handles the non-linear complexity presented by agricultural data effectively.

In comparison, Linear Regression showed limited performance (R^2 = 0.5392), suggesting a simple linear relationship in a multi-factorial context has little to offer in terms of crop yield. The evaluation phase provides insight into models' strengths and weaknesses individually, but no one method offers strong generalization across crop types or regions.

Deep learning models such as DNN with (R^2 = 0.9100) and 1D CNN (R^2 = 0.9052) demonstrated strong individual performance, while the Multilayer Perceptron (MLP) performed slightly lower. These outcomes confirm the ability of neural networks to capture complex, non-linear patterns in the data.

After evaluating individual model performance, we opted for a hybrid stacking approach, utilizing K-Fold Cross Validation to verify robustness and generalization. Results of ensemble modeling are shown below.

The stacking model outperforms individual models with R^2 = 0.9830, explaining 98.3% of the variance in crop yield. The RMSE of 0.2884 suggests a very low average prediction error magnitude, and the MAE of 0.1628 indicates predictions cluster very near actual values with low dispersion. The MSE (mean squared error) of 0.0832 further emphasizes small prediction errors and avoidance of large errors. These metrics demonstrate that stacking not only improves mean predictions but also provides consistency and reliability.

2.5.2 Comparative Analysis and Discussion with Related Work and Existing Approaches

To evaluate our models performance, we compared our results to those achieved by existing machine learning and deep learning algorithms.

This table 2.9 presents a comparison between our proposed approach and other existing approaches in terms of their results.

Model	R^2 ↑	RMSE ↓	MAE ↓	MSE ↓
H_StackCYP	**0.9830**	**0.2884**	**0.1628**	**0.0832**
[Ramesh and Kumaresan, 2025]	0.9800	124.78	7.20	15570.32
[Subramaniam and Marimuthu, 2024]	0.9800	0.219	0.065	0.034
[Patrick et al., 2023]	0.9990	5.29	2.80	8.36
[Chergui, 2022]	0.96	0.04	0.0285	N/A

TABLE 2.9 – Comparison of Model Performances for Crop Yield Prediction

The study on Durum Wheat Yield by [Chergui, 2022] focuses only on a single crop type, which allows for a detailed analysis but limits the model's ability to generalize to other crops. In comparison, crop yield prediction models that incorporate multiple crop types offer broader and more versatile insights. Additionally, the study is confined to just two regions : Setif and Constantine. While these states have specific environmental factors affecting wheat yield. The study also limits its model choice to just two algorithms, Deep Neural Networks (DNN), which restricts the exploration of potentially more effective methods.

Also the research on banana yield prediction in Tanzania by [Patrick et al., 2023] showed an R^2 of 0.999, but it was simpler because it focused on one crop in one region.

Similarly, the study that achieved 98.96% accuracy by [Subramaniam and Marimuthu, 2024] using Deep Learning and Dimensionality Reduction was based on a small dataset, covering only three crops. This narrow crop focus significantly impairs the model's ability to generalize to other crops.

The study by [Ramesh and Kumaresan, 2025] employs a stacking approach using only machine learning models, without incorporating any deep learning techniques. While the method demonstrates potential, its effectiveness is limited by the small dataset used. Moreover, the model relies only on climatic features, The author acknowledges this limitation and indicates plans to include soil parameters in future work to improve the models accuracy and robustness.

In contrast, our Hybrid Stacking Ensemble Model (H_StackCYP) for Crop Yield Prediction achieved an R^2 of 0.9827, demonstrating excellent results by stacking multiple models to capture intricate data patterns. Our model combines deep learning and machine learning techniques, offering superior adaptability and performance across different crops. This success is made possible by our larger dataset of 99,848 samples, which includes 55 crops from 22 states and incorporates vital environmental data such as soil nutrients, pH, rainfall, and temperature.

2.6 Implementation Environment

Implementing of our crop yield prediction involves a combination of front end and backend technologies. This section provides an overview of the tools and technologies used in both the front-end and back-end development of the application.

The illustration 2.17 showcasing the architecture of the tools used is provided at the start of this section for better understanding.

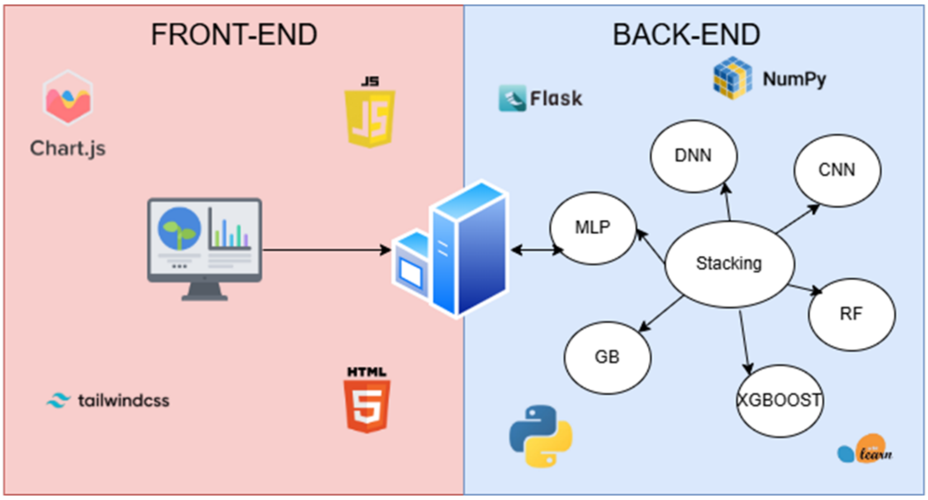

FIGURE 2.17 – Front-end and Back-end Tools Used

The tables 2.10, 2.11, 2.12 below explain the description of the front-end and back-end tools and the communication tool between them that we have used in building the application :

2.6.1 Front-End Tools

Tool	Description
HTML	HyperText Markup Language is the standard language used to create the structure of web pages.
Tailwind CSS	Tailwind CSS is a utility-first CSS framework for rapidly building custom designs. It offers low-level utility classes that can be combined to build any design directly in your HTML. Tailwind CSS is used to style the application and obtain a consistent and responsive design across its various components.
Chart.js	A powerful open-source JavaScript library for creating interactive charts and data visualizations for the web.

TABLE 2.10 – Description of Front-end Tools Used

2.6.2 Back-End Tools

Tool	Description
Python	Python is a high-level, interpreted programming language recognized for its readability and extensive ecosystem of libraries and frameworks. It is commonly used in web development, data science, and machine learning. In our application, popular libraries such as pandas, joblib, Scikit-learn, numpy, and PyTorch were utilized for tasks including data preprocessing, model training, evaluation, and visualization.
Scikit-learn	Scikit-learn is a robust machine learning library implemented in Python. It provides simple and efficient tools for data mining and data analysis. In our application, Scikit-learn is used for model training and evaluation.
PyTorch	PyTorch helped us to build and train deep learning models.
Pandas	Used to handle and manipulate our dataset.
Joblib	Used to save and load trained models for later use in deployment.
Flask	Flask is a lightweight and flexible web framework in Python. It is commonly used to implement web applications and APIs due to its simplicity and ease of integration with other tools. Our project utilized Flask to implement the backend of the system. It serves as a bridge between the frontend (user interface) and the machine learning backend model. When a user inputs data through the interface and submits it, Flask receives the data, presents it to the backend model, loads the trained model, and returns the predicted output.

TABLE 2.11 – Description of Back-end Tools Used

2.6.3 Communication Tool

Tool	Description
HTTP	The application enables communication from the frontend to the backend, using HTTP (Hypertext Transfer Protocol). Rather than use JSON based APIs, the application relies on a standard HTML tag based form submission using HTTP POST to send data between the frontend and backend. The form captures input from the user and then the backend uses Flasks request.form to read the inputs. The results are returned to the user by using Jinja2 templates to render dynamic HTML pages.

TABLE 2.12 – Communication Tool Used

2.7 H_StackCYP Web Architecture

The web application is structure around several key functional components, illustrate in figure **??** Each with a specific role. The architecture is structured as follows :

a) **User Interface :** This part is handles user interaction and input. It is done with HTML, CSS(Tailwind CSS), and JavaScript ; Farmers can input the parameters for the crop, state, season, and soil/climate characteristics. It will display online prediction results.

b) **Parameter Handler :** When a user submits input through the interface, these parameters (NPK values, rainfall, temperature, humidity, selected crop, season, area, soil ph, state) are sent to the backend in the same way that it gets HTTP requests from the Flask routing mechanism.

c) **Flask Backend Server :** The backend implemented using flask , it receives the input features from the frontend , and load the training models, performs data preprocessing, controls and executes the hybrid stacking model drafted on the application.

— **Preprocessing Module :** This module is used to cleans, scale , encode the input features to transforme into the same structure as the model training to ensure the consistency between the user data provided and the input expected by the models.

— **Prediction Engine :** This component contains the hyprid stacking method built using scikit-learn, pytorch and xgboost ,it includes the individual base models and the meta model that combines their output.

— **Result Processor :** Once the prediction yield value is generated, this module formats the result to be sent into the frontend using the response from the flask and display it.

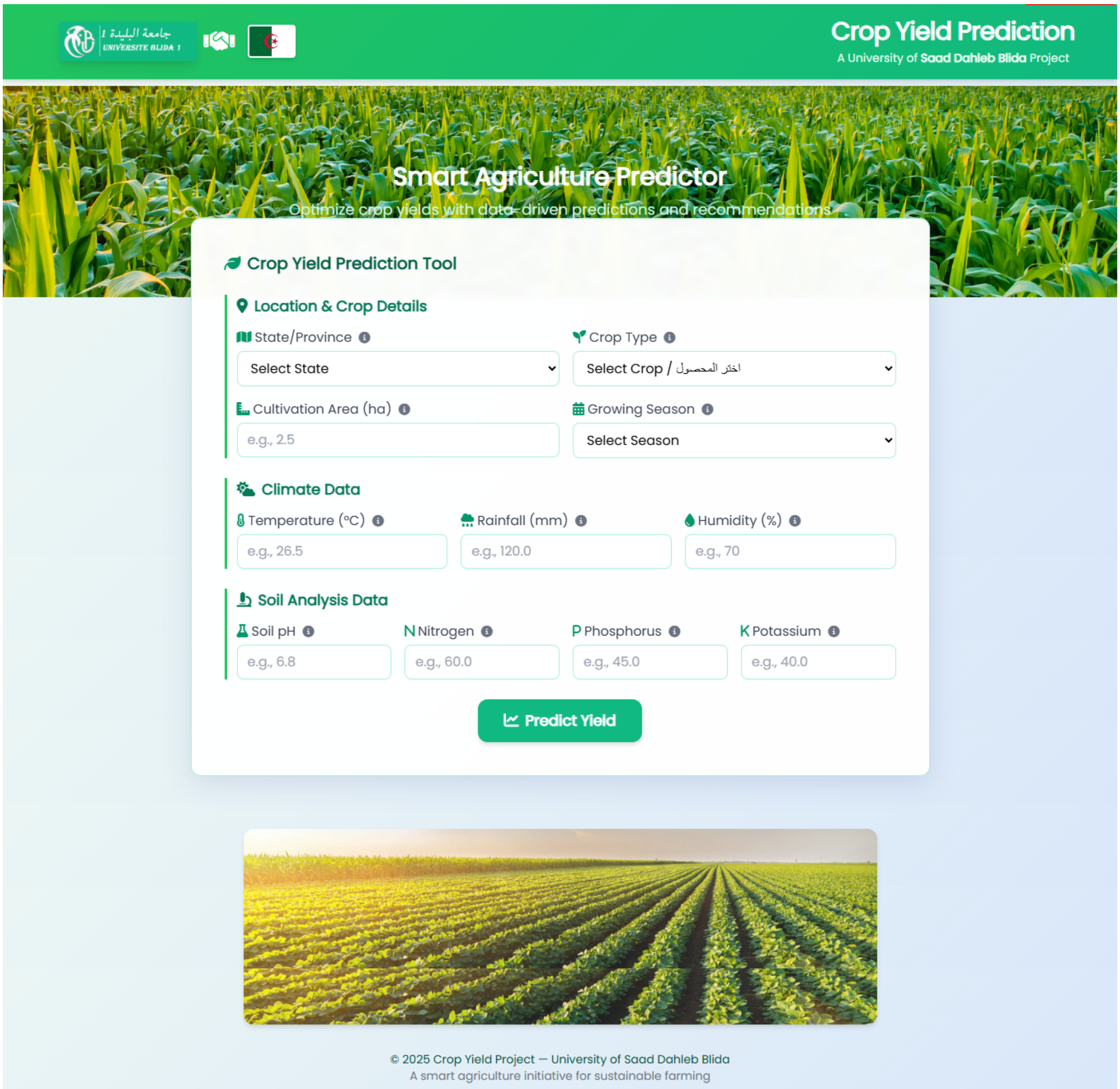

FIGURE 2.18 – Crop Yield Prediction Web Interface

2.8 Conclusion

In this chapter, we presented and assessed the performance of the proposed models for crop yield prediction while also factoring in humidity prediction as a new predictive feature. After intensive hyperparameter tuning of the machine learning and deep learning models using RandomizedSearchCV, we were able to obtain excellent results. In our comparison analysis, we showed that including humidity always improved the predictive performance for all models. We considered our Hybrid Stacking Ensemble Model (H_StackCYP) to exceed previously published techniques in the literature indicating R^2 of 0.9830 which exceeded by some comparators in recent studies in the literature. The investigation has reaffirmed the value of hybrid modeling and the incorporation of a comprehensive feature set of environmental factors for developing more accurate and reliable crop yield prediction frameworks. To demonstrate the practical utility of our work, we also developed a web-based application that allows users to input relevant agricultural parameters and instantly obtain yield predictions, making the system accessible and user-friendly for real-world deployment.

General Conclusion

In this dissertation, we have addressed the important issue of predicting crop yields in regards to global food security problems. The agricultural industry faces increasing pressures to produce more in spite of less inputs, extensive environmental variability and a growing population. Our hybrid stacking model H_StackCYP demonstrates a significant improvement on yield predictions with high accuracy using both environmental and agronomic information.

Bibliography

Abrha, B. K. (2015). *Factors affecting agricultural production in Tigray region, northern Ethiopia*. PhD thesis, University of South Africa.

Akkem, Y., Biswas, S. K., and Varanasi, A. (2023). Smart farming using artificial intelligence : A review. *Engineering Applications of Artificial Intelligence*, 120 :105899.

Albukhitan, S. (2020). Developing digital transformation strategy for manufacturing. *Procedia Computer Science*, 170 :664–671.

Amthor, J. S. (2001). Effects of atmospheric co concentration on wheat yield : Review of results from experiments using various approaches to control co concentration. *Field Crops Research*, 73(1) :1–34.

Balakrishnan, D., Kumar, A. P., Reddy, K. S. K., Kumar, R. R., Aadith, K., and Madhan, S. (2023). Agricultural crop recommendation system. In *2023 3rd International Conference on Intelligent Technologies (CONIT)*, pages 1–5. IEEE.

Basak, D., Pal, S., and Patranabis, D. C. (2007). Support vector regression. *Electrical Laboratory, Central Institute of Mining and Fuel Research*. Available from : deba65@yahoo.com ; srimanta@isical.ac.in ; dcp@iee.jusl.ac.in.

Bernhard, A. (2010). The nitrogen cycle : processes. *Players, and Human*, 3(10) :25.

Brady, N. C., Weil, R. R., and Weil, R. R. (2008). *The nature and properties of soils*, volume 13. Prentice Hall Upper Saddle River, NJ.

Chen, T. and Guestrin, C. (2016). Xgboost : A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 785–794. ACM.

Chergui, N. (2022). Durum wheat yield forecasting using machine learning. *Artificial Intelligence in Agriculture*, 6 :156–166.

Chin, R., Catal, C., and Kassahun, A. (2023). Plant disease detection using drones in precision agriculture. *Precision Agriculture*, 24(5) :1663–1682.

Choi, R. Y., Coyner, A. S., Kalpathy-Cramer, J., Chiang, M. F., and Campbell, J. P. (2020). Introduction to machine learning, neural networks, and deep learning. *Translational vision science & technology*, 9(2) :14–14.

Cutler, A., Cutler, D. R., and Stevens, J. R. (2012). Random forests. *Ensemble machine learning : Methods and applications*, pages 157–175.

Eli-Chukwu, N. C. (2019). Applications of artificial intelligence in agriculture : A review. *Engineering, Technology & Applied Science Research*, 9(4).

FAOSTAT (2024). Faostat. Accessed : 2025-04-20.

Food and of the United Nations (FAO), A. O. (2021). The state of food and agriculture 2021 : Making agrifood systems more resilient to shocks and stresses. https://www.fao.org/3/cb4476en/cb4476en.pdf. Accessed : 2025-04-08.

Hancock, J. T. and Khoshgoftaar, T. M. (2020). Survey on categorical data for neural networks. *Journal of Big Data*, 7(1) :1–41.

Huang, Y. and Chen, Y. (2020). Autonomous driving with deep learning : A survey of state-of-art technologies. *arXiv preprint arXiv :2006.06091*.

Janiesch, C., Zschech, P., and Heinrich, K. (2021). Machine learning and deep learning. *Electronic Markets*, 31(3) :685–695.

Kelly, B. T., Kuznetsov, B., Malamud, S., and Xu, T. A. (2023). Deep learning from implied volatility surfaces. *Swiss Finance Institute Research Paper*, (23-60).

Khalid, S. (2020). *Agronomy : Climate Change*. IntechOpen.

Kumar, J., Chawla, R., Katiyar, D., Chouriya, A., Nath, D., Sahoo, S., Ali, A., and veer Singh, B. (2023). Optimizing irrigation and nutrient management in agriculture through artificial intelligence implementation. *Int. J. Environ. Clim. Change*, 13(10) :4016–4022.

Liakos, K. G., Busato, P., Moshou, D., Pearson, S., and Bochtis, D. (2018). Machine learning in agriculture : A review. *Sensors*, 18(8) :2674.

Luckin, R. and Holmes, W. (2016). Intelligence unleashed : An argument for ai in education.

Muraina, I. O. (2022). Ideal dataset splitting ratios in machine learning algorithms : General concerns for data scientists and data analysts. In *7th International Mardin Artuklu Scientific Researches Conference*, pages 496–504, Mardin, Turkey. Mardin Artuklu Congress. Conference Paper.

Musanase, C., Vodacek, A., Hanyurwimfura, D., Uwitonze, A., and Kabandana, I. (2023). Data-driven analysis and machine learning-based crop and fertilizer recommendation system for revolutionizing farming practices. *Agriculture*, 13(11) :2141.

Patrick, S., Mirau, S., Mbalawata, I., and Leo, J. (2023). Time series and ensemble models to forecast banana crop yield in tanzania, considering the effects of climate change. *Resources, Environment and Sustainability*, 14 :100138.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., et al. (2011). Scikit-learn : Machine learning in python. *the Journal of machine Learning research*, 12 :2825–2830.

Pingali, P. L. (2012). Green revolution : impacts, limits, and the path ahead. *Proceedings of the national academy of sciences*, 109(31) :12302–12308.

Ramesh, V. and Kumaresan, P. (2025). Stacked ensemble model for accurate crop yield prediction using machine learning techniques. *Environmental Research Communications*, 7(3) :035006.

Shumway, R. H., Stoffer, D. S., and Stoffer, D. S. (2000). *Time series analysis and its applications*, volume 3. Springer.

Siebert, S., Webber, H., Zhao, G., and Ewert, F. (2014). Impact of heat stress on crop yieldon the importance of considering canopy temperature. *Environmental Research Letters*, 9(4) :044012.

Smith, D. L. and Almaraz, J. J. (2004). Climate change and crop production : contributions, impacts, and adaptations. *Canadian Journal of Plant Pathology*, 26(3) :253–266.

Sposito, G. (2008). *The Chemistry of Soils*. Oxford University Press.

Subramaniam, L. K. and Marimuthu, R. (2024). Crop yield prediction using effective deep learning and dimensionality reduction approaches for indian regional crops. *e-Prime-Advances in Electrical Engineering, Electronics and Energy*, 8 :100611.

Tang, Y., Gu, L., and Wang, L. (2021). Deep stacking network for intrusion detection. *Sensors*, 22(1) :25.

Tatachar, A. V. (2021). Comparative assessment of regression models based on model evaluation metrics. *International Research Journal of Engineering and Technology (IRJET)*, 8(09) :2395–0056.

Topol, E. J. (2019). High-performance medicine : the convergence of human and artificial intelligence. *Nature Medicine*, 25(1) :44–56.

Tukey, J. W. et al. (1977). *Exploratory data analysis*, volume 2. Springer.

Xu, X., Gao, P., Zhu, X., Guo, W., Ding, J., Li, C., Zhu, M., and Wu, X. (2019). Design of an integrated climatic assessment indicator (icai) for wheat production : A case study in jiangsu province, china. *Ecological indicators*, 101 :943–953.

Zhao, Z., Chow, T. L., Rees, H. W., Yang, Q., Xing, Z., and Meng, F.-R. (2009). Predict soil texture distributions using an artificial neural network model. *Computers and electronics in agriculture*, 65(1) :36–48.