A Neural meta model for Predicting winter wheat crop yield

Yogesh Bansal[1*], M-Tahar Kechadi[1,2], and David Lillis[1]

School of Computer Science, University College Dublin, Ireland.
yogesh.bansal@ucdconnect.ie, david.lillis@ucd.ie
Insight Centre for Data Analytics,
tahar.kechadi@ucd.ie

Abstract. This study presents the development and evaluation of machine learning models to pre- dict winter wheat crop yield using heterogeneous soil and weather data sets. A concept of an error stabilisation stopping mechanism is introduced in an LSTM model specifically designed for heterogeneous datasets. The comparative analysis of this model against an LSTM model highlighted its superior predictive performance. Furthermore, weighted regression models were developed to capture environmental factors using agroclimatic indices. Finally, a neural meta model was built by combining the predictions of several individual models. The experimental results indicated that a neural meta model with an MAE of 0.82 and RMSE of 0.983 tons/hectare demonstrated a notable performance, highlighting the importance of incorporating weighted regression models based on agroclimatic indices. This study shows the potential for improved yield prediction through the proposed model and the subsequent development of a meta model.

Keywords: Neural Meta Model · Soil and weather dataset· Agro-climatic indices· Feature engineering · LSTM · Weighted Regression ML models

1 Introduction

Crop production plays a key role in ensuring food security and supporting economies around the world. An extensive number of studies have been conducted in this field using a variety of machine learning (ML) models, covering deep learning and ensemble models on diverse datasets to improve crop yield predictions. However, a recurring challenge in this field is the effective management of heterogeneous data sets, which can lead to less optimal results. Motivated by this challenge, our aim is to improve a previously built long short-term memory (LSTM) model specifically designed to handle such datasets. Previously, we established an LSTM [3] model designed specifically for heterogeneous data sets. In this paper, we advance our model by introducing error stabilisation stopping within LSTM with the aim of improving the accuracy of yield predictions.

* Corresponding author: yogesh.bansal@ucdconnect.ie

The presence of agroclimatic indices is important to capture the environmental con- ditions on a zone level. To do this, we build weighted regression ML models (decision tree, random forest, and gradient boosting). Lastly, we build several neural meta models that combines the strengths of individual models. The aim of this study is to provide an improved machine learning model for the prediction of winter wheat crop yield.

The paper is structured as follows. The literature is presented in Section 2, followed by a description of the data in Section 3. Preprocessing information is provided in Section 4, followed by "experiment setup" and "results" in Sections 6 and 7, respectively. Section 9 concludes the study.

2 Literature

The incorporation of machine learning (ML), including deep learning techniques, into agricultural crop yield predictions has garnered considerable attention in recent years. These data-driven methodologies play an increasingly vital role in shaping the direction of research in this field. The increasing interest is reflected in the volume of studies and surveys that have rigorously explored, evaluated, and compared various predictive models.

Elavarasan et al. [9] began by surveying various machine learning models and comparing their performance using error measures. Expanding on this, Klompenburg et al. [24] provided a systematic review of the applications of machine learning in the prediction of crop yields. Building on these initial works, Oikonomidis et al. [14] focused on deep learning methods, demonstrating their potential to improve prediction accuracy. Bali & Singla. [2] explored emerging trends, underscoring the influence of various factors on crop yield. More recently, Toomula & Pelluri [23] and Muruganantham et al. [13] offered updated views on deep learning-based models, the latter identifying LSTM and CNN as the most effective techniques, thus signifying continuous innovation in this rapidly evolving field.

Hybrid models, which combine various ML models, have demonstrated superior performance in crop yield predictions compared to individual ML models. The study by Sun et al. [22] provides a notable example, where the authors proposed deep convolutional neural networks (CNN) for the prediction of soybean yield at the end of the season and at the county level. Their findings indicated that the proposed CNN-LSTM model outperformed either the CNN or LSTM model alone, suggesting that such hybrid model architectures have considerable potential to improve yield predictions for various other crops, including corn, wheat, and potatoes. This sentiment echoes the research by Wang et al. [25], who developed an LSTM-CNN model to estimate winter wheat yield at the county level in China. The LSTM-CNN model was based on weather and remote sensing data, and the results indicated that it significantly improved the model's yield prediction capabilities.

Agroclimatic indices also play a crucial role in crop yield modelling. As Mathieu and Aires. [12] highlighted, these indices, including factors such as temperature and precipitation, can significantly improve crop yield modelling compared to using direct weather variables alone. This understanding of the agroclimatic conditions of a region can enable farmers to better manage their crops, improve yields, and minimise the risks associated with climate variability. Chergui et al. [6] provide a comprehensive review of machine learning models applied to crop management, shedding light on various ML techniques and their effectiveness in different contexts. Wang et al. research [25] further emphasises the potential of ML, where they successfully used neural models to estimate winter wheat production based on soil and weather data, and the neural models outperform other models.

Sharma et al. [19] used an LSTM model to predict agricultural yields through satellite data in India, outperforming other ML techniques. A similar superiority of LSTM was observed in studies by Iniyan et al. [10] and Bali and Singla. [1]. Furthermore, combining LSTM with random forest showed an improved forecast accuracy, as shown by Shen et al. [20]. Bansal et al. [3, 4] highlighted the importance of weather data and introduced a deep learning model to refine the predictions of the winter wheat crop yield.

Shahhosseini et al.'s research [17, 18] centred on ensemble machine learning models, with CNN-DNN ensembles presenting accurate results. Dang et al. [7] compared various models, finding Deep Neural Networks to be the most effective. Pravallika et al. [16] and Srivastava et al. emphasised the significant impact of environmental variables on yield predictions [21]. Zhang et al. advocate for combining deep learning and ensemble approaches [26, 27], and the integration of various data sources, such as satellite imagery, has been explored in numerous studies [8, 15]. Hybrid deep learning frameworks like that of Khaki et al. have shown promising outcomes [11].

3 Data Description

The data sources used in this study consist of soil and weather data sets that cover a large number of farms for several years. A farm is divided into several fields, and a field is further divided into zones. A zone is defined as a subregion within a field that has similar characteristics in terms of crop management [24]. Let Z be a set of zones: $Z = \{z_1, z_2, \dots, z_n\}$. For each zone z_i by year, the following features are grouped into two categories: soil data and weather data. The soil data contains information on the soil tests conducted in the agricultural zones. These soil tests are infrequent (typically every 3 to 4 years) because they are expensive and also because the values do not vary substantially over relatively short periods of time. Since we are using a time frame of one year per instance, we assume that the soil properties do not change meaningfully in that time, and so treat them as constants. If a zone does not have a soil test in a given year, the most recent test done in the same zone in the year before is used to map the soil. However, weather data are a collection of a series of time points. An instance I in the data set can be represented as follows:

$$I = [Year, ZoneID, S, W, T, Y] \tag{1}$$

where $S = \{s_1, s_2, \cdots, s_x\}$ is the set of x soil variables. $W = \{w_1, w_2, \cdots, w_y\}$ is the set of y weather variables, T is time in weeks, and Y is yield.

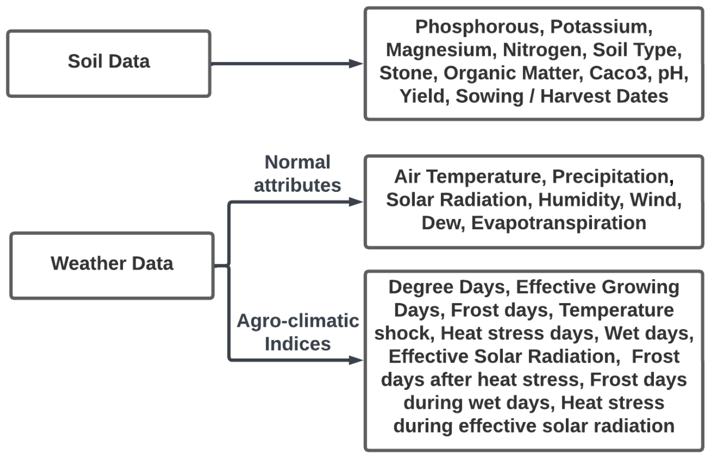

Fig. 1: Data Description

Figure 1 shows the attributes of the soil and weather data used in this study from 2013 to 2018. It comprises soil nutrients (P, K, Mg), physical properties (soil type, stone content), chemical properties (organic matter, $CaCO_3$, pH), and the yield for a zone along with the sowing and harvest dates. The numerous soil type classifications in the data set are shallow, medium, deep clay and deep fertile; stone content is stoneless, low, moderate, high and gravel; organic matter is low, moderate and very high; and $CaCO_3$ is slightly calc, calc, extremely calc and potentially acidic. Furthermore, weather data include air temperature, precipitation, solar radiation, humidity, wind, dew, and evapotranspiration. More details of the data are presented in this study [4] by the same authors.

4 Data Preprocessing

The data pre-processing step is crucial in our study for crop yield prediction, as it involves cleaning and transforming raw data into a format that can be used by machine learning algorithms. It is necessary to remove inconsistencies, missing values, and errors in the data, which could lead to incorrect predictions. Also, it involves transforming the data into a format that is suitable for analysis, such as normalisation or encoding categorical variables.

In this study, after preprocessing soil and weather data sets, those weather features that are calculated by week are integrated with soil features to form a diverse and heterogeneous data set. The features are then represented weekly starting from the 1^{st} week up to the t^{th} week followed by filtering them to focus on the growth period, specifically from the 17^{th} week to the harvest period. This time frame captures the period that has the most significant effect on the yield of the winter wheat crop.

Figure 2 illustrates the preprocessing steps of the soil and weather data. Weather features are calculated both weekly and on the basis of the winter wheat growth stages. The winter wheat growth stages are as follows. Foundation, Construction, Production. All weather features are calculated according to the sowing / harvest dates of a zone/year. These weather features determined weekly are integrated with the soil data after undergoing a principal component analysis. Meanwhile, the weather features derived from winter wheat growth stages pertain to agro-climatic indices that are infrequently observed, leading to an imbalanced data set if calculated by week. When calculating weather features by growth stage, the data can be better balanced, ensuring that the model can learn better.

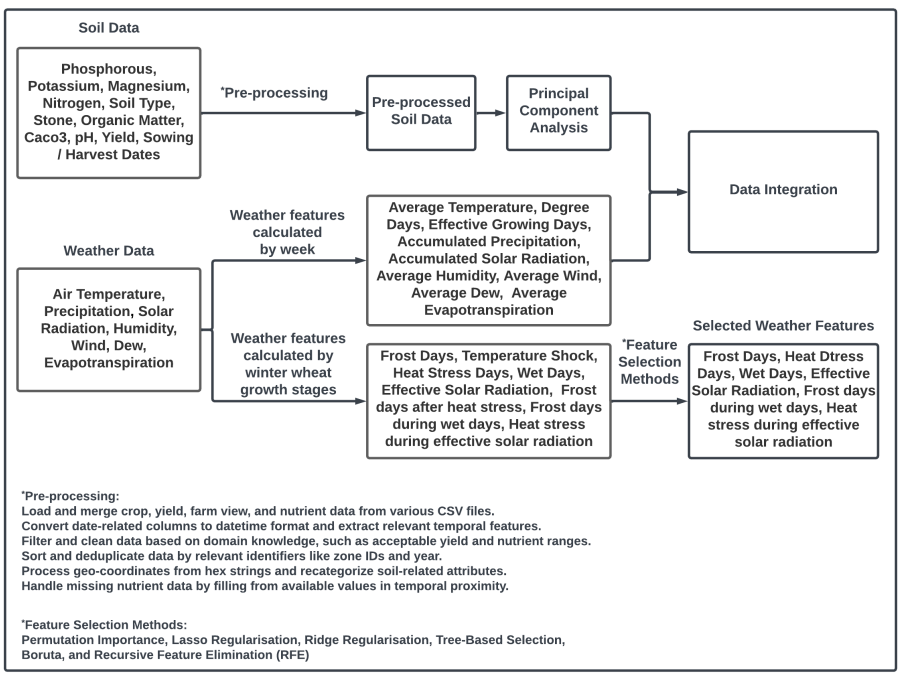

Fig. 2: Data Pre-processing

The formulas of all preproceseed weather features are listed below.

- $T_{\text{avg}} = \text{mean}((T), \text{week})$ where T_{avg} is the average temperature and T is temperature.

- $DD = \text{sum}(\max(0, (T_{\max} + T_{\min})/2), \text{week})$ where DD is degree days.
- $EGD = \text{count}((T_{\text{avg}} > 5°C), \text{week})$ where EGD is effective growing days.
- $AP = \text{sum}((P), \text{week})$ where AP is accumulated precipitation.
- $SR_{\text{sum}} = \text{sum}((SR), \text{week})$ where SR_{sum} is the total solar radiation.
- $H_{\text{avg}} = \text{mean}((H), \text{week})$ where H_{avg} is average humidity.
- $W_{\text{avg}} = \text{mean}((\text{Wind}), \text{week})$ where W_{avg} is average wind speed.
- $D_{\text{avg}} = \text{mean}((\text{Dew}), \text{week})$ where D_{avg} is average dew point.
- $EVT_{\text{avg}} = \text{mean}((EVT), \text{week})$ where EVT_{avg} is average evapotranspiration.
- $FD = \text{count}((T_{\min}) \leq 0°C, \text{GrowthStage})$ where FD is the total number of frost days.
- $TS = \text{count}(((T_{\text{avg}} > 5°C)\&(T_{\min} > 0°C) \geq 10)\&(T_{\min} < 0°C), \text{GrowthStage})$ where TS is temperature shock.
- $HSD = \text{count}((T_{\max}) > 25°C, \text{GrowthStage})$ where HSD is the total number of heat stress days.
- $WD = \text{count}((P > 10\text{mm}), \text{GrowthStage})$ where WD is the total number of wet days.
- $ESR = \text{count}((SR > 5°C), \text{GrowthStage})$ where ESR is effective solar radiation.
- $FDAHS = \text{count}(((T_{\min}) \leq 0°C \text{ after } (T_{\max}) > 25°C), \text{GrowthStage})$ where $FDAHS$ is the number of frost days after heat stress.
- $FDWD = \text{count}((T_{\min} < 0°C)\&(P > 10\text{mm}), \text{GrowthStage})$ where $FDWD$ is the number of frost days during wet days.
- $HSESR = \text{count}((T_{\max} > 25°C)\&(T_{\text{avg}} > 5°C), \text{GrowthStage})$ where $HSESR$ is the total number of heat stress days during effective solar radiation.

In this study, we also implemented multiple feature selection techniques [5] to identify the most significant predictors in our data set. These methods included permutation importance, lasso regularisation, ridge regularisation, tree-based selection, boruta, and recursive feature elimination (RFE). These diverse techniques were chosen to ensure a comprehensive evaluation of feature importance and to minimise the risk of missing relevant predictors. Our analysis revealed that different feature selection methods yielded varying results. However, two methods, Boruta and recursive feature elimination (RFE), consistently identified the same subset of features (i.e., Frost days, heat stress days, wet days, effective solar radiation, frost days during wet days, heat stress during effective solar radiation) as the most important. In particular, these findings aligned well with our existing domain knowledge, further supporting the importance of these features.

5 Proposed Approach

Figure 3 outlines a methodology for a neural meta model aimed at predicting the yield of the winter wheat crop. In this study, the data set encompasses the years 2013 to 2018. The training set is derived from the years 2013 to 2017, while the 2018 data serve as the test set. A fivefold cross-validation method is employed on the training data to identify the optimal hyperparameters for each model utilised in this research. Following cross-valida- tion, the final model is trained using the best hyperparameters and subsequently evaluated against the test set.

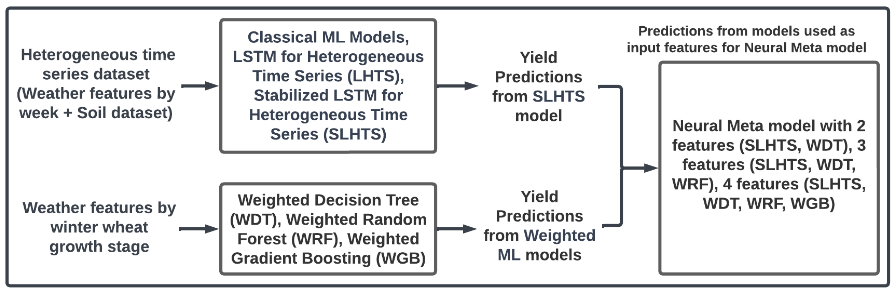

Fig. 3: Proposed Approach

Initially, we employed a range of classical ML models such as decision tree (DT), support vector (SVR), random forest (RF), extra trees (ET), lightGBM (LGBM), gradi- ent boosting (GB), as well as LSTM and its stabilised version designed for heterogeneous time series datasets. These data sets are primarily made up of weekly calculated weather data features and soil data characteristics. Additionally, weather features determined by the growth stage are used to forecast winter wheat crop yield using weighted regression models such as the weighted decision tree (WDT), the weighted random forest (WRF), and weighted gradient boosting (WGB). The yield predictions sourced from the stabilised LSTM for heterogeneous time series (SLHTS) and the weighted ML models are subsequently fed as input features into a neural meta model. This model is configured in varia- tions to accommodate two input features (SLHTS and WDT), three input features (SLHTS, WDT, WRF), and four input features (SLHTS, WDT, WRF, WGB). The following subsections provide a comprehensive explanation of each methodology used in our proposed approach.

5.1 LSTM for Heterogeneous Time Series (LHTS)

Figure 4 shows the description of time steps and forward/backward propagation of the LSTM model designed for handling heterogeneous datasets. Each time step represents a week. Since only the winter wheat growth period until harvest weeks are considered in the analysis, the 1^{st} time step starts from the 17^{th} week, the 2^{nd} time step is the 18^{th} week, and the t^{th} time step corresponds to the $16+t^{\text{th}}$ week where t starts from 1. Initially, the cell state and hidden state are initialised to 0 which gets updated at each time step based on the input and the previous hidden state.

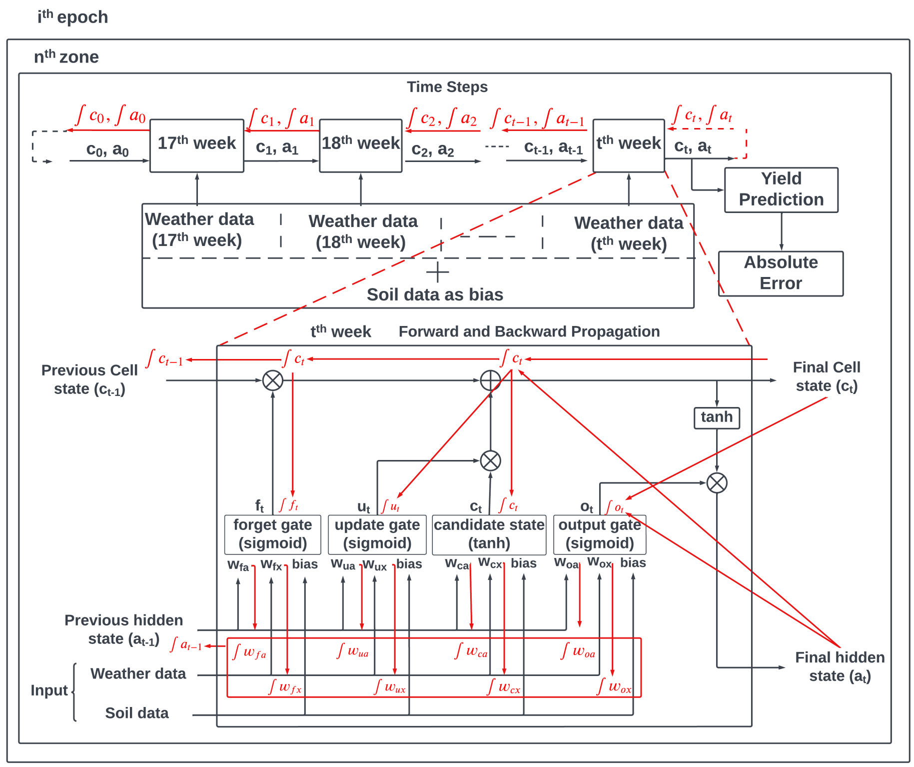

Fig. 4: Architecture of LSTM for Heterogeneous Time Series (LHTS)

In the forward time step, weather data for each week of a zone along with its bias is propagated from 17^{th} week to $16 + t^{\text{th}}$ week of that zone. In the backward time step, gradients of weight parameters related to only sequence data, i.e., weather data, are generated. Since nonsequence data, that is, the soil remains constant throughout the year for a zone, it is considered as a bias and its gradients are not computed.

The following are the notations used in figure 4:

f_t, u_t, and o_t represents forget, update and output gate, respectively. $\int f_t$, $\int u_t$, and $\int o_t$ are the respective derivatives. w_{fa}, w_{ua}, and w_{oa} are the weight parameters related to the hidden state of forget, update and output gate, respectively. $\int w_{fa}$, $\int w_{ua}$, $\int w_{oa}$ are the respective derivatives. w_{fx}, w_{ux}, and w_{ox} are the weight parameters related to the weather data from forget, update and output gate, respectively. $\int w_{fx}$, $\int w_{ux}$, $\int w_{ox}$ are the respective derivatives. a_{t-1}, a_t represents the previous and current hidden state. Similarly, c_{t-1}, c_t represents the previous and current cell state. $\int a_{t-1}$, $\int a_t$, $\int c_{t-1}$, $\int c_t$ are the respective derivatives.

In forward propagation, weight parameters of forget gate, update gate, candidate state, output gate, and input comprising of weather and soil data are propagated. Here, a candidate state updates the cell state on the basis of the input and the previous hidden state. After each time step, the cell state and hidden state are updated. Calculate the gradients of the sequence weather data only by back-propagation over time at time step t using the chain rule. No gradients of soil data are computed during back propagation; instead, it is inputted as a bias. The training and testing steps of the LHTS model are as follows.

- Data loading, parameter initialisation, and training loop: After reading the datasets and initialising all the model parameters, the model is trained for a specified number of epochs and zones in the training set. For each epoch and zone, the model performs a forward propagation, calculates the absolute error, and updates the model parameters using the Adam optimiser in the backward propagation.
- Cross-validation scheme: A fivefold cross-validation is used. During each fold, we leave out one year from the training set to serve as the validation set, and the remaining years form the training set.
- Model training and selection of best hyperparameters: Train the model using different sets of hyperparameters and validate its performance using cross-validation. After the training process is completed, the model will have the best hyperparameters.
- Final model training and model evaluation: Train the model with the best hyperparameters obtained from cross-validation. The trained model is then evaluated on the test set by predicting the yield and calculating the mean absolute error.

5.2 Error Stabilised LSTM for Heterogeneous Time Series (SLHTS)

In this subsection, we dive deeper into the specifics of the SLHTS model, detailing its design, operation, and the distinct characteristics that make it an effective choice for this research. In time-series deep learning models, managing the learning process is of high importance to achieve accurate and reliable predictions. Section 5.1 describes a system architecture for a time series deep learning model suited for heterogeneous data sets and used a conventional backpropagation approach. Although effective, it did not fully take advantage of dynamic control of the learning process based on error stabilisation. Due to this, a novel learning strategy that incorporates the idea of error stabilisation into the training process is presented. The architecture is described in Figure 5 and applied to heterogeneous data sets.

This learning methodology involves monitoring previous n errors during training where n is set to 10 in this study. This parameter can be adjusted according to the characteristics and size of the data used. The error stabilisation condition is invoked when the absolute difference between the current error and the previous error falls within or below the minimum of the previous n errors. Here, current error refers to the absolute difference between the actual yield of the

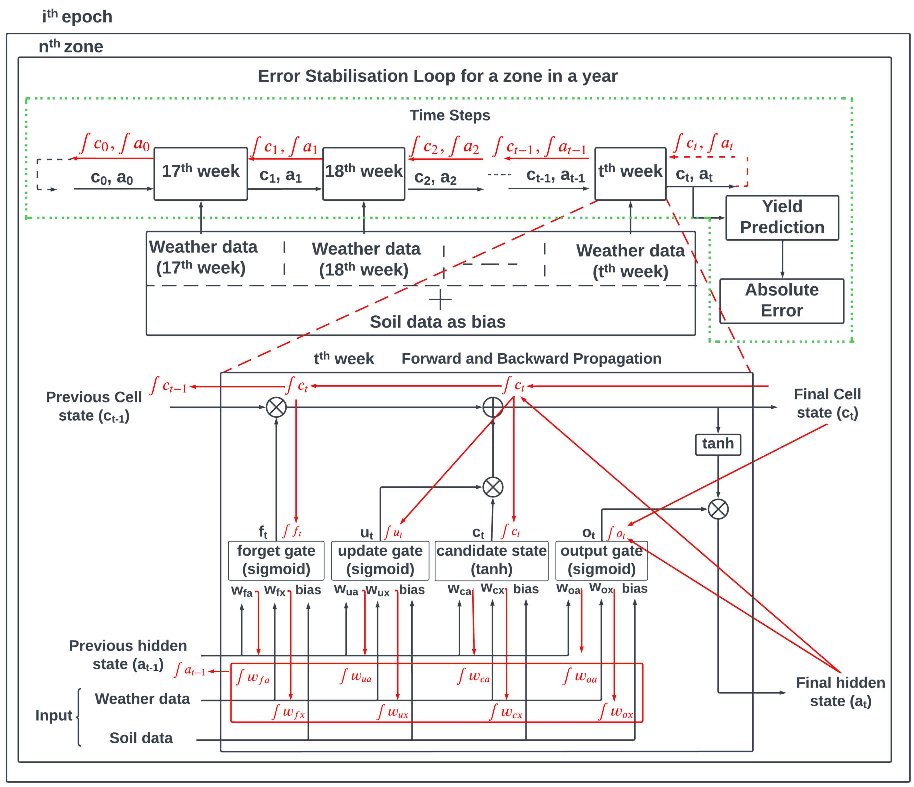

Fig. 5: Architecture of Stabilised LSTM for Heterogeneous Time Series (SLHTS)

zone in a year and the yield predicted by the model for the current iteration of the training phase. The previous error refers to the similar absolute difference, but for the previous iteration in the training phase. This condition states that if the change in error (i.e. the absolute difference between the current and previous errors) does not surpass the smallest change recorded in the recent past (i.e., the minimum of the previous n errors), that means the model learning has stagnated, thereby needing a pause in the training for the current zone in a year. The training and testing steps of the SLHTS model are as follows.

- Data loading, parameter initialisation, and training loop: After reading the datasets and initialising all the model parameters, the model is trained for a specified number of epochs and zones in the training set. For each epoch and zone, the model performs a forward propagation, calculates the absolute error, and updates the model parameters using the Adam optimiser in the backward propagation.
- Error computation and parameter update: In this methodology, the model performs iterative updates to its parameters, simultaneously calculating a

new error with each iteration. This iterative process of parameter update and error computation continues until the difference between the new and old error diminishes to a value smaller than the smallest error observed in the last 10 iterations. Throughout this procedure, the error and predictions for each zone in a year are stored for subsequent analysis and evaluation.

- Cross-validation scheme: A fivefold cross-validation is used. During each fold, we leave out one year from the training set to serve as the validation set, and the remaining years form the training set.
- Model training and selection of best hyperparameters: Train the model using different sets of hyperparameters and validate its performance using cross-validation. After the training process is completed, the model will have the best hyperparameters.
- Final Model Training and Model Evaluation: Train the model with the best hyperparameters obtained from cross-validation. The trained model is then evaluated on the test set by predicting the yield and calculating the mean absolute error.

This approach allows the model to learn until the error stabilises for each zone in a year, providing more fine-tuned control over the learning process.

5.3 Architecture for Weighted Regression Models

This subsection focusses on the model architecture specifically designed for the weather features (i.e., agro-climatic indices) calculated by the winter wheat growth stage. Figure 6 illustrates the architecture of the weighted regression models used in our study. The training and testing steps of this methodology are as follows:

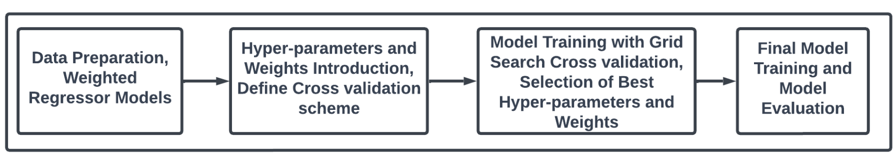

Fig. 6: Model Architecture for Agro-climatic Indices dataset

- Data loading, and weighted regression models: After reading the datasets, three classical ML regression models, i.e., Decision Tree, Random Forest, and Gradient Boosting are customised to be weighted regression models.
- Hyperparameters and weights introduction: Customising these regression models involves the application of weights along with model-specific hyper-parameters to the critical points (lower and higher extreme yield values). Both weights and these critical points are adjustable parameters.

- Cross-validation scheme: The same fivefold cross-validation is used by leaving out one year from the training set to serve as the validation set and the remaining years to form the training set.
- Model training and selection of best hyperparameters/weights: Train the model using different sets of hyperparameters and weights. The performance of these models is validated using cross-validation which fine-tunes not only the model hyperparameters but also the weights assigned to the critical data points. After the training process is completed, the regression models will have the best hyperparameters, the best weights, and the critical yield thresholds at which these weights are applied.
- Final model training and model evaluation: Train the models using the best weights and hyperparameters obtained from cross-validation. The trained model is then evaluated on the test set by predicting the yield and calculating the mean absolute error.

5.4 Neural Meta Model Architecture

This subsection explores the architecture of the neural meta model as stated in figure 7. The training and testing steps of a neural meta model are as follows:

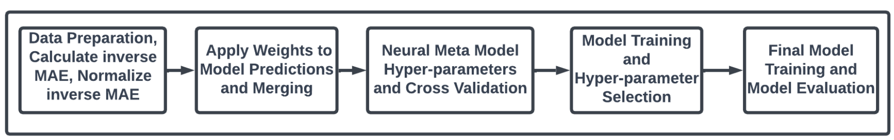

Fig. 7: Neural Meta model Architecture

- Data Loading, calculation and normalisation of inverse MAE: Obtain train, test predictions, and train MAE from previously trained models (SLHTS and weighted regression models). Then, calculate the inverse MAE and normalise to get weights for each model based on their performance. A model with a lower MAE will have a higher inverse MAE.
- Weighted predictions and data preparation for a neural meta model: These weights are then multiplied by the model predictions to get weighted predictions. These serve as the input features and are merged to form the data set for a neural meta model.
- Cross-validation scheme: The same fivefold cross-validation is used by leaving out one year from the training set to serve as the validation set, and the remaining years to form the training set.
- Model training and selection of best hyperparameters: Train the neural meta model using different sets of hyperparameters, validate its performance using cross-validation. After the training process is completed, the model will have the best hyperparameters.

- Final model training and model evaluation: Train the model with the best hyperparameters obtained from cross-validation. The trained model is then evaluated on the test set by predicting the yield and calculating the mean absolute error.

6 Experiment Setup

- Computational setup: All experiments were carried out on a machine equipped with Intel Core i7 processor, CPU @ 3.20GHz 6 cores, and 32 GB RAM. The models were implemented using pandas, numpy, sklearn python libraries and trained on CONTOUR soil dataset and Iteris ClearAg weather dataset described in Section 3.
- Data partitioning: The data set was divided according to years, with training data covering 2013-2017 and test data for 2018. This temporal split ensures that the model is evaluated on unseen future data.
- Evaluation methodology: Five-fold cross-validation was used in the training set (2013-2017). The performances of the models were evaluated using MAE, RMSE, and R^2 metrics. The primary metrics for this study were MAE and RMSE, with R^2 being secondary. The test set (2018) was used to evaluate the performance of the trained models.
- SLHTS model: A comprehensive search was conducted to find the best hyperparameters in a validation set using the Adam optimiser. Hyperparameter values included hidden units ranging from 10 to 40, learning rates ranging from 0.001, 0.002, to 0.01, and epochs spanning 10 to 100.
- Classical and weighted regression ML models: Hyperparameters such as maximum depth (2, 3, 4), minimum sample split (2, 3, 4), and minimum sample leaf (2, 3, 4, 5), learning rate (0.001, 0.01, 0.1) and estimators (100, 200, 300) were considered.
- Neural meta model: We explored a range of learning rates from 0.001 to 0.01 in increments of 0.001, epochs including 100, 200, 300, 400, and 500, hidden units of 64 and 128, and batch sizes of 16 and 32.
- Performance evaluation: Consultations with agronomists highlighted that MAE is the most intuitive metric for them due to its expression in metric tons per hectare. Additionally, one-tailed paired t-tests were utilised to assess the statistical significance of performance differences between the various ML models in crop yield predictions.

7 Experimental Results

In this section, we dive into the experimental results derived from various ML models deployed to predict winter wheat crop yields. The performance of these models is evaluated based on regression metrics, namely MAE, RMSE, and R^2. We start with traditional ML approaches, transitioning to an LSTM model designed for heterogeneous datasets that combine soil and weekly calculated weather features. This is succeeded by an LSTM model enhanced for error stabilisation. Subsequently, we present the performance of weighted regression models that use weather features calculated by the winter wheat growth stages based on different harvesting periods in different zones. Concluding the section, results for neural meta model are presented.

7.1 Classical ML models

Table 1 show the performance of traditional ML models on heterogeneous datasets that combine soil and weather features calculated weekly to predict the yield of winter wheat crops. All ML models have very low R^2 values, indicating that they explain a small amount of variance in the dependent variable. Although GB has the best MAE and RMSE, its R^2 is not the highest among models. Given our focus on prediction accuracy, GB is the best performing ML model. In addition, significant differences are observed between SVR and DT ($p \approx 9.61 \times 10^{-37}$), and between RF and SVR ($p \approx 3.53 \times 10^{-8}$). However, comparisons of RF with ET ($p \approx 0.104$), LGBM ($p \approx 0.103$), and GB ($p \approx 0.226$) showed no significant difference in their performance.

Table 1: Classical ML Models

ML Models	**Best Hyper-parameters**	**MAE**	**RMSE**	**R2**
Decision Tree	$md = 2$, $msl = 3$, $mss = 2$	2.57	2.97	0.015
Support Vector	$kernel = rbf$, $C = 1.0$, $epsilon = 0.1$	1.6	1.91	**0.03**
Random Forest	$md = 2$, $msl = 3$, $mss = 3$, $ne = 200$	1.45	1.77	0.002
Extra Trees	$md = 2$, $msl = 3$, $mss = 3$, $ne = 200$	1.48	1.79	0.008
Light GBM	$lr = 0.01$, $ne = 100$	1.51	1.83	0.008
Gradient Boosting	$md = 3$, $msl = 3$, $mss = 2$, $lr = 0.01$, $ne = 100$	**1.4**	**1.69**	0.01

Note: $md = max\_depth$, $msl = min\_samples\_leaf$, $mss = min\_samples\_split$, $lr = learning\_rate$, $ne = n\_estimators$

7.2 Error Stabilised LSTM

Table 9 and Figure 8 show the performance of Error Stabilised Heterogeneous Time Series (SLHTS), LHTS, GB on heterogeneous datasets that combine soil and weekly calculated weather features.

The error stabilised LSTM model (SLHTS) has the best MAE, RMSE, and R^2 among the models and therefore the best performing model. Additionally, significant differences are observed between SLHTS and LHTS ($p \approx 8.91 \times 10^{-40}$), between SLHTS and GB ($p \approx 1.095 \times 10^{-11}$), and between LHTS and GB ($p \approx 0.0024$).

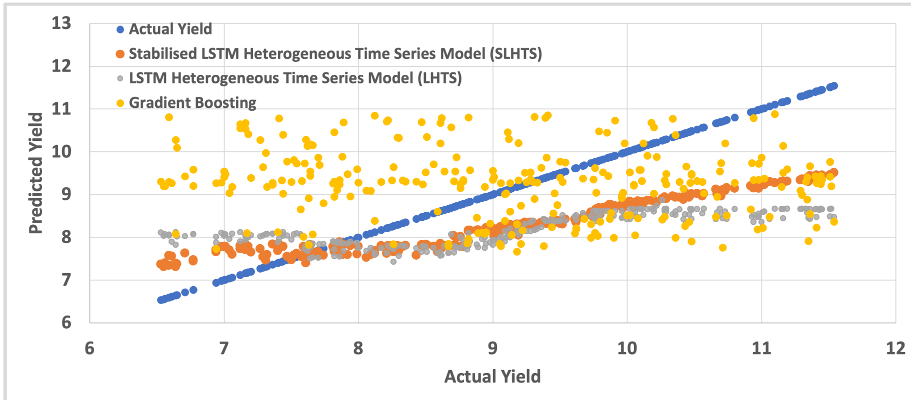

Fig. 8: SLHTS vs LHTS vs GB

Fig. 9: Comparison of SLHTS, LHTS and GB Model. Best Hyper-parameters (h=Hidden units, lr=Learning rate, e=Epochs, md = max_depth, msl = min_samples_leaf, mss = min_samples_split, ne = n_estimators).

Models	**Best Hyper-parameters**	**MAE**	**RMSE**	**R2**
SLHTS	lr=0.005, e=10, h=10	**0.98**	**1.125**	**0.928**
LHTS	lr=0.005, e=20, h=10	1.25	1.452	0.591
GB	$md = 3$, $msl = 3$, $mss = 2$, $lr = 0.01$, $ne = 100$	1.4	1.69	0.01

7.3 Comparison of WDT, WRF, and WGB

Table 2 show the performance of weighted regression models Weighted Decision Tree (WDT), Weighted Random Forest (WRF), and Weighted Gradient Boosting (WGB) using weather features calculated by the winter wheat growth stage. While WDT has the best MAE and RMSE, its R^2 is not the highest among the models. It provides the smallest average error and penalizes larger errors the least. Being MAE and RMSE as the primary evaluation metrics in this study, WDT is the best performing model. In addition, significant differences are observed between WDT and WRF ($p \approx 1.98 \times 10^{-5}$), between WDT and WGB ($p \approx 2.14 \times 10^{-14}$), and between WRF and WGB ($p \approx 3.65 \times 10^{-14}$). The output of these weighted regression models fed as input to the neural meta model whose results are presented in the next subsection.

Table 2: Comparison of Weighted Regression Models

ML Models	Best Hyper-parameters	MAE	RMSE	R2
WDT	$md = 2$, $msl = 3$, $mss = 2$, $lyt = 7$, $wl = 1.0$, $hyt = 11$, $wh = 1.2$	**1.199**	**1.142**	0.0058
WRF	$md = 2$, $msl = 3$, $mss = 3$, $ne = 200$, $lyt = 7$, $wl = 2.6$, $hyt = 11$, $wh = 1.6$	1.23	1.45	**0.06**
WGB	$md = 3$, $msl = 3$, $mss = 2$, $lr = 0.01$, $ne = 100$, $lyt = 8$, $wl = 1.8$, $hyt = 11$, $wh = 2.2$	1.678	1.95	0.039

Note: $WDT = Weighted\_Decision\_Tree$, $WRF = Weighted\_Random\_Forest$, $WGB = Weighted\_Gradient\_Boosting$, $md = max\_depth$, $msl = min\_samples\_leaf$, $mss = min\_samples\_split$, $lr = learning\_rate$, $ne = n\_estimators$, $lyt = low\_yield\_threshold$, $hyt = high\_yield\_threshold$, $wl = weight\_low$, $wh = weight\_high$

7.4 Neural meta model (NMM) vs error stabilised LSTM for heterogeneous time series (SLHTS)

In Table 11 and Figure 10, we present the performance of various neural meta models alongside error stabilised LSTM model, SLHTS. The neural meta models, abbreviated as NMM, are differentiated by their input features: - NMM_2 with two inputs: predictions from WDT and SLHTS. - NMM_3 with three inputs: predictions from WDT, WRF, and SLHTS. - NMM_4 with four inputs: predictions from WDT, WRF, WGB, and SLHTS. NMM_2 has the best MAE and RMSE, suggesting that it tends to have the smallest average prediction error and also penalizes larger errors the least compared to the other models. NMM_3 has the highest R^2, implying that it can explain the highest variability in the target variable. Given NMM_2 has the best MAE and RMSE, it is considered as the best performing model. Furthermore, significant differences are observed between NMM_2 and NMM_3 ($p \approx 0.003$), and between NMM_2 and NMM_4 ($p \approx 0.0001$), and between NMM_2 and SLHTS ($p \approx 2.41 \times 10^{-39}$).

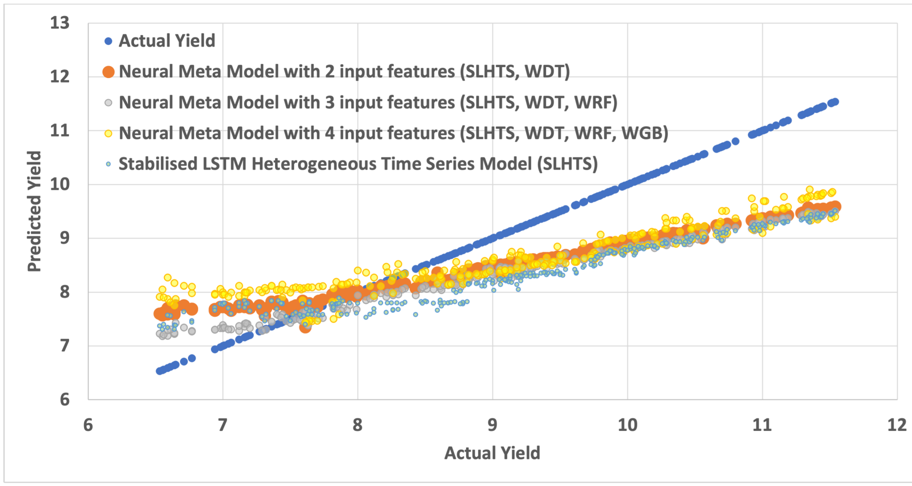

Fig. 10: Neural Meta Model vs SLHTS

Fig. 11: Comparison of neural meta model and SLHTS. Best Hyper-parameters (lr=Learning rate, e=Epochs, h=Hidden units, b=batch size).

Models	**Best Hyper-parameters**	**MAE**	**RMSE**	**R2**
NMM_2		**0.824**	**0.983**	0.98
NMM_3	$lr = 0.01$, $e = 200$, $h = 32$, $b = 16$	0.854	1.038	**0.988**
NMM_4		0.869	1.0	0.863
SLHTS	$h = 10$, $lr = 0.005$, $e = 10$, $b = 1$	0.98	1.126	0.929

Note: NMM_2, NMM_3, and NMM_4 represent neural meta model with 2, 3, and 4 input features respectively. These features are predictions from weighted regression models and the SLHTS model

8 Ablation Study

The importance of this study is shown through an ablation analysis, which provides an understanding of the performance contributions of different modeling techniques for predicting winter wheat crop yield. As evidenced in Table 3, the evaluation emphasises MAE and RMSE as the primary metrics, since they directly reflect the prediction accuracy, which is central to this study. Following are the observations:

- Traditional ML model Gradient Boosting serves as a baseline with an MAE of 1.4 and RMSE of 1.69.
- Moving from LSTM model for heterogeneous time series to error stabilised LSTM results in a noticeable performance improvement, with a drop in MAE from 1.25 to 0.98 and in RMSE from 1.452 to 1.125.
- Among neural meta models (NMM), which integrate predictions from other models as input features, the model with two input features performs best with the lowest MAE of 0.824 and RMSE of 0.983.

In conclusion, a neural meta model with 2 input features came out to be the best performing model in our analysis, emphasizing the value of meta learning in crop yield prediction.

Table 3: Comparison of GB, LHTS, SLHTS and NMM Models

Models	**MAE**	**RMSE**
Gradient Boosting	1.4	1.69
LSTM for heterogeneous time series	1.25	1.452
Error stabilised LSTM for heterogeneous time series	0.98	1.125
Neural meta model with 2 input features	**0.824**	**0.983**
Neural meta model with 3 input features	0.854	1.038
Neural meta model with 4 input features	0.869	1.0

9 Conclusion

In this study, we worked on using machine learning to predict crop yield, focusing on winter wheat. We developed and tested many ML models, from classical models to advanced neural architectures, trying to find the best way to predict yield. Our experimental tests showed the effectiveness of meta model learning, particularly the neural meta model with 2 input features, which came out as the top performing model. The neural meta model performs well in combining individual model strengths, but it has a static nature, i.e., it does not adjust based on the evolving patterns in the data. A key limitation is that it assigns a fixed weight to the predictions of the individual models based on the performance of each model.

To overcome the identified limitation, our future work will refine the neural meta model to be more adaptable. One strategy involves dynamically adjusting the weights based on the absolute errors of each instance rather than using the overall performance of the individual models. This dynamic adjustment will enable the model to continually adapt to the evolving patterns in the data. In addition, we plan to test the neural meta model with larger datasets to check its robustness and ability to scale.

Acknowledgement

This research is funded under the SFI Strategic Partnerships Program (16/SPP/3296) and is co-funded by Origin Enterprises Plc.

References

1. Nishu Bali and Anshu Singla. Deep learning based wheat crop yield prediction model in punjab region of north india. Applied Artificial Intelligence, 35(15):1304–1328, 2021.

2. Nishu Bali and Anshu Singla. Emerging trends in machine learning to predict crop yield and study its influential factors: a survey. Archives of computational methods in engineering, 29(1):95–112, 2022.
3. Yogesh Bansal, David Lillis, and M.-Tahar Kechadi. A deep learning model for heterogeneous dataset analysis - application to winter wheat crop yield prediction. In Jemal Abawajy, Joao Tavares, Latika Kharb, Deepak Chahal, and Ali Bou Nassif, editors, Information, Communication and Computing Technology, pages 182–194, Cham, 2023. Springer Nature Switzerland.
4. Yogesh Bansal, David Lillis, and Tahar Kechadi. Winter wheat crop yield prediction on multiple heterogeneous datasets using machine learning. In 2022 International Conference on Computational Science and Computational Intelligence (CSCI), pages 206–212, 2022.
5. Girish Chandrashekar and Ferat Sahin. A survey on feature selection methods. Computers & Electrical Engineering, 40(1):16–28, 2014.
6. Nabila Chergui and Mohand Tahar Kechadi. Data analytics for crop management: a big data view. Journal of Big Data, 9(1):1–37, 2022.
7. Chaoya Dang, Ying Liu, Hui Yue, JiaXin Qian, and Rong Zhu. Autumn crop yield prediction using data-driven approaches:-support vector machines, random forest, and deep neural network methods. Canadian Journal of Remote Sensing, 47(2):162–181, 2021.
8. M Sarith Divakar, M Sudheep Elayidom, and R Rajesh. Forecasting crop yield with deep learning based ensemble model. Materials Today: Proceedings, 58:256–259, 2022.
9. Dhivya Elavarasan, Durai Raj Vincent, Vishal Sharma, Albert Y Zomaya, and Kathiravan Srinivasan. Forecasting yield by integrating agrarian factors and machine learning models: A survey. Computers and electronics in agriculture, 155:257–282, 2018.
10. S Iniyan, V Akhil Varma, and Ch Teja Naidu. Crop yield prediction using machine learning techniques. Advances in Engineering Software, 175:103326, 2023.
11. Saeed Khaki, Lizhi Wang, and Sotirios V Archontoulis. A cnn-rnn framework for crop yield prediction. Frontiers in Plant Science, 10:1750, 2020.
12. Jordane A Mathieu and Filipe Aires. Assessment of the agro-climatic indices to improve crop yield forecasting. Agricultural and forest meteorology, 253:15–30, 2018.
13. Priyanga Muruganantham, Santoso Wibowo, Srimannarayana Grandhi, Nahidul Hoque Samrat, and Nahina Islam. A systematic literature review on crop yield prediction with deep learning and remote sensing. Remote Sensing, 14(9):1990, 2022.
14. Alexandros Oikonomidis, Cagatay Catal, and Ayalew Kassahun. Deep learning for crop yield prediction: a systematic literature review. New Zealand Journal of Crop and Horticultural Science, pages 1–26, 2022.
15. Alexandros Oikonomidis, Cagatay Catal, and Ayalew Kassahun. Hybrid deep learning-based models for crop yield prediction. Applied artificial intelligence, 36(1):2031822, 2022.
16. K Pravallika, G Karuna, K Anuradha, and V Srilakshmi. Deep neural network model for proficient crop yield prediction. In E3S Web of Conferences, volume 309, page 01031. EDP Sciences, 2021.
17. Mohsen Shahhosseini, Guiping Hu, and Sotirios V Archontoulis. Forecasting corn yield with machine learning ensembles. Frontiers in Plant Science, 11:1120, 2020.
18. Mohsen Shahhosseini, Guiping Hu, Saeed Khaki, and Sotirios V Archontoulis. Corn yield prediction with ensemble cnn-dnn. Frontiers in plant science, 12:709008, 2021.

19. Sagarika Sharma, Sujit Rai, and Narayanan C Krishnan. Wheat crop yield prediction using deep lstm model. arXiv preprint arXiv:2011.01498, 2020.
20. Yulin Shen, Benoît Mercatoris, Zhen Cao, Paul Kwan, Leifeng Guo, Hongxun Yao, and Qian Cheng. Improving wheat yield prediction accuracy using lstm-rf framework based on uav thermal infrared and multispectral imagery. Agriculture, 12(6):892, 2022.
21. Amit Kumar Srivastava, Nima Safaei, Saeed Khaki, Gina Lopez, Wenzhi Zeng, Frank Ewert, Thomas Gaiser, and Jaber Rahimi. Winter wheat yield prediction using convolutional neural networks from environmental and phenological data. Scientific Reports, 12(1):3215, 2022.
22. Jie Sun, Liping Di, Ziheng Sun, Yonglin Shen, and Zulong Lai. County-level soybean yield prediction using deep cnn-lstm model. Sensors, 19(20):4363, 2019.
23. Srilatha Toomula and Sudha Pelluri. An extensive survey of deep learning-based crop yield prediction models for precision agriculture. In Proceedings of the International Conference on Cognitive and Intelligent Computing: ICCIC 2021, Volume 1, pages 1–12. Springer, 2022.
24. Thomas Van Klompenburg, Ayalew Kassahun, and Cagatay Catal. Crop yield prediction using machine learning: A systematic literature review. Computers and Electronics in Agriculture, 177:105709, 2020.
25. Xinlei Wang, Jianxi Huang, Quanlong Feng, and Dongqin Yin. Winter wheat yield prediction at county level and uncertainty analysis in main wheat-producing regions of china with deep learning approaches. Remote Sensing, 12(11):1744, 2020.
26. Yuefan Zhang, Lunche Wang, Xinxin Chen, Yuting Liu, Shaoqiang Wang, and Lizhe Wang. Prediction of winter wheat yield at county level in china using ensemble learning. Progress in Physical Geography: Earth and Environment, 46(5):676–696, 2022.
27. Yuzhen Zhang, Jingjing Liu, and Wenjuan Shen. A review of ensemble learning algorithms used in remote sensing applications. Applied Sciences, 12(17):8654, 2022.