electronics

MDPI

Article

A Temporal–Geospatial Deep Learning Framework for Crop Yield Prediction

Lei Wang, Zhengkui Chen, Weichun Liu and Hai Huang *

School of Computer Science and Technology, Zhejiang Sci-Tech University, Hangzhou 310018, China; 202230603120@mails.zstu.edu.cn (L.W.); chenzk@zstu.edu.cn (Z.C.); 202130504121@mails.zstu.edu.cn (W.L.)
* Correspondence: haihuang1005@gmail.com

Abstract: With the rapid development of information technology, the demand for digital agriculture is increasing. As an important agricultural production topic, crop yield has always attracted much attention. Currently, artificial intelligence, particularly machine learning, has become the leading approach for crop yield prediction. As a result, developing a machine learning method that accurately predicts crop yield has become one of the central challenges in digital agriculture. Unlike traditional regression prediction problems, crop yield prediction has a significant time correlation. For example, weather data for each county show strong temporal correlations. Moreover, geographic information from different regions also impacts crop yield to a certain extent. For example, if a county's neighboring counties have a good harvest, then this county is likely to have high yields as well. This paper introduces a novel hybrid deep learning framework that combines convolutional neural network (CNN), graph attention network (GAT) and long short-term memory (LSTM) modules to enhance prediction accuracy. Specifically, CNN is employed to extract the features from the input data for each county in each year. GAT is introduced to model the geographical relationships between neighboring counties, allowing the model to capture spatial dependencies more effectively. LSTM is used to extract the temporal information within many years. The proposed hybrid deep learning framework CNN-GAT-LSTM captures both the temporal and spatial relationships, thereby improving the accuracy of yield prediction. We conduct experiments on a nationwide dataset that includes data from 1115 soybean-producing counties in 13 states in the United States covering the years from 1980 to 2018. We evaluate the performance of our proposed CNN-GAT-LSTM model based on three metrics, namely root of the mean squared error (RMSE), R-squared (R^2) and correlation coefficient (Corr). The experimental results demonstrate that the proposed model achieves significant performance improvements over the existing state-of-the-art model, with RMSE reduced by 5%, R^2 improved by 6% and Corr enhanced by 4%.

Keywords: crop yield prediction; temporal–geospatial; deep learning; machine learning

Citation: Wang, L.; Chen, Z.; Liu, W.; Huang, H. A Temporal–Geospatial Deep Learning Framework for Crop Yield Prediction. *Electronics* **2024**, *13*, 4273. https://doi.org/10.3390/electronics13214273

Academic Editors: Aryya Gangopadhyay and George A. Tsihrintzis

Received: 9 September 2024
Revised: 20 October 2024
Accepted: 30 October 2024
Published: 31 October 2024

1. Introduction

Since traditional agriculture is vulnerable to extreme weather and natural disasters, and crop yields are difficult to predict in advance, major natural disasters may lead to crop yield reductions and cause significant losses. However, if various data, such as weather data and soil data, can be used to predict future crop yields and provide early warnings for potential disasters, this will greatly promote the development of modern agriculture and help avoid major losses. Based on this demand, the concept of digital agriculture has been proposed, which aims to use computers to predict future crop yields.

A crop growth model [1–3] simulates plant growth and development based on underlying physiological mechanisms, and can be used to predict crop yield. However, a crop growth model has some limitations [4,5]. For example, the descriptions of specific steps in these methods are overly simplified or ambiguous, which affects the accuracy of crop prediction. Moreover, the crop growth model demands highly specialized domain knowledge

and significant investment in calibrating its parameters. In contrast, machine learning [6] is data-driven and excels at managing complex nonlinear relationships, outperforming the crop growth model in this regard.

Various machine learning methods [7], such as decision trees [8], artificial neural networks [9] and support vector machines [10], have long been used for crop yield prediction. Although machine learning has made significant progress in agricultural yield prediction, it also has some shortcomings. For example, machine learning methods require manual feature extraction to improve the model's predictive ability, which limits the scalability and generalizability of the model. In addition, existing machine learning methods are usually based on small self-collected regional datasets and may not perform well on large datasets like the entire country.

In recent years, due to the rapid development of deep learning, neural network-based deep learning [11–13], an extension of machine learning, has become increasingly popular in the field of crop yield prediction. In comparison with the traditional shallow machine learning models, the main advantage of deep learning is its powerful feature extraction capability without the need for manual feature extraction by experts. Therefore, deep learning models tend to show better adaptability, generalization and feature learning capabilities in agricultural crop prediction applications.

Existing deep learning-based crop yield prediction methods typically focus on modeling temporal relationships within a dataset. However, they overlook the critical role of geographical relationships in crop yield prediction, treating each county as an independent and identically distributed (i.i.d.) sample in their models. In fact, the geographical relationship between counties also plays an important role in crop prediction. For example, yields in neighboring counties are usually not independent, i.e., if a county's neighboring counties have a good harvest, then this county is likely to have high yields as well.

Recent work [14] proposed a deep learning framework for crop yield prediction, utilizing a graph neural network (GNN) model [15] to capture geographical relationships among counties. However, GNN assumes a fixed structure where all neighboring nodes are treated equally, leading to a potential loss of important spatial information and an inability to account for the varying influence of different neighboring counties on yield predictions. This lack of differentiation may result in the suboptimal modeling of geographical relationships, ultimately compromising prediction accuracy. Therefore, it is essential to develop more effective models that better leverage both temporal and spatial information for accurate crop yield prediction.

1.1. Contribution

This paper proposes a temporal–geospatial deep learning framework, CNN-GAT-LSTM, for crop prediction. First, a convolutional neural network (CNN) module is employed to extract features from the input data for each county in each year. Second, to address the varying influence of different counties on a particular county's yield, we introduce a graph attention network (GAT) [16,17] module to capture the importance of different counties. The attention-based approach enables GAT to assign varying weights to different neighbors, resulting in a more flexible and potentially more expressive model. Finally, a long short-term memory (LSTM) module is used to effectively capture and utilize temporal information over multiple years. By integrating these components, the proposed deep learning framework CNN-GAT-LSTM effectively captures geospatial information while modeling temporal sequence relationships, leading to improved accuracy in yield predictions.

We conduct experiments on a nationwide dataset that includes data from 1115 soybean-producing counties in 13 states in the United States covering the years from 1980 to 2018. The performance of the proposed CNN-GAT-LSTM model is evaluated based on three metrics, namely root of the mean squared error (RMSE), R-squared (R^2) and correlation coefficient (Corr). The experimental results demonstrate that our model significantly outperforms the existing state of the art, achieving a 5% reduction in RMSE, a 6% improvement in R^2 and a 4% increase in Corr.

1.2. Organization

The rest of this paper is organized as follows. Section 2 reviews the existing machine learning methods for crop yield prediction. Section 3 outlines the components of our proposed crop yield prediction model. Section 4 describes the architecture of our model and Section 5 provides the implementation details of our model. Section 6 presents the visualization results and Section 7 conducts ablation experiments. Section 8 provides limitations and discussion. Finally, Section 9 summarizes this paper.

2. Related Work

Existing machine learning approaches for crop prediction can generally be categorized into traditional machine learning-based methods and deep learning-based techniques.

2.1. Traditional Machine Learning-Based

Several works have used traditional machine learning-based methods, e.g., random forest, K-nearest neighbor (KNN), artificial neural network and gradient boosting, to predict crop yields.

Random forest. Curtis J. Ransom [8] evaluated various machine learning models using soil and weather variables from a regional database in the U.S. Midwest, demonstrating that the random forest method outperformed stepwise regression and ridge regression models. M. Kuradusenge et al. [18] proposed machine learning models, such as random forests and support vector regressors, to predict crop yields (i.e., Irish potatoes and corn) in the Musanze region of Rwanda based on weather and yield history data. Nikhil [19] proposed machine learning models to predict the yield of various crops such as rice, sorghum, cotton, sugarcane and rabi, with the Extra Trees Regressor and achieved the highest performance among the models examined. These models were trained using weather, soil and crop data from specific states of India.

KNN. Suresh et al. [20] proposed a prediction method for the major crops of Tamil Nadu, India, using K-means and a modified KNN. Karn [21] investigated a crop recommendation system utilizing the KNN algorithm with a dataset consisting of 22 columns and 2200 rows of data from Kaggle. Kumar et al. [22] also proposed the KNN algorithm for crop recommendation, using a dataset gathered from Kaggle for different crops.

Neural network. Sarr et al. [9] used several machine learning methods, including neural network, support vector machine and random forest, to predict crop yields (peanut, maize, millet and sorghum) in 24 departments of Senegal. The dataset does not include soil conditions for crop yield prediction. Das [23] proposed a hybrid approach to predict crop yields by combining a feature selection method, MARS, with either artificial neural network or support vector regression (SVR). They experimented on a small lentil dataset. Sadenova et al. [24] proposed several machine learning models for predicting crop yield in eastern Kazakhstan, among which neural network methods outperformed other methods. This dataset is based on remote sensing data obtained from free sources.

Gradient boosting. Shahhosseini et al. [25] evaluated the potential of several machine learning algorithms for maize yield and nitrate loss prediction on a simulated dataset obtained from a factorial simulation experiment. They concluded that XGBoost is the most accurate machine learning model. P. Mishra et al. [26] used gradient boosting regression to predict the crop yields for districts in France. Pradeep et al. [27] proposed gradient boosting regression to predict the crop yield for districts in India. Yasaswy et al. [28] used the gradient boosting for crop prediction, showing better performance than random forest on a dataset from Kaggle.

Others. Niketa Gandhi et al. [29] employed Naive Bayes and Bayesian networks to predict the yield of Indian rice based on eight parameters, including precipitation, minimum temperature and area. The results showed that Bayesian networks outperformed Naive Bayes in prediction. Abdel-salam et al. [10] integrated a feature selection approach with an optimized SVR model to efficiently enhance prediction accuracy. The dataset for this

study was collected from official Indian government websites belonging to several agricultural ministries.

2.2. Deep Learning-Based

In recent years, due to the rapid development of deep learning, neural network-based models, e.g., CNN and LSTM, have become increasingly popular in the field of crop yield prediction. The traditional shallow machine learning models mentioned above typically require manual feature extraction and are generally only suited for small datasets. In contrast, deep learning models have inherent feature extraction capabilities and perform better on larger datasets.

CNN. Some works have proposed deep learning frameworks, such as CNN, to extract spatial relationship features for crop yield prediction. Khaki et al. [11] designed a deep neural network model to predict corn yield across 2247 locations in the United States. Kim et al. [30] developed a deep neural network model for crop yield prediction using optimized input variables from satellite products and meteorological datasets between 2006 and 2015 in the midwestern United States. Kumar et al. [31] proposed a hybrid deep capsule autoencoder with softmax regression (Hybrid DCAS) model, using a dataset collected from from the Agra district in the Uttar Pradesh (UP) region of India. Subramaniam et al. [32] proposed weight-tuned deep convolutional neural network (WTDCNN) and dimensionality reduction (DR) approaches for the crop yield prediction of Indian regional crops.

LSTM. Some works have proposed deep learning frameworks, such as LSTM, to extract temporal relationship features for crop yield prediction. Shen et al. [12] proposed a framework combining a long short-term memory neural network and random forest (LSTM-RF) to predict wheat yield in Xinxiang, Henan Province, China. The remote sensing datasets were acquired using multispectral and thermal sensors. Bhimavarapu et al. [33] proposed a new and improved optimization function (IOF) to train the LSTM model, aiming to reduce the training and testing loss in crop prediction. The dataset was collected from districts in India. Wang et al. [34] proposed a LSTM model for wheat yield prediction on a remote sensing dataset obtained from Henan Province, China. Di et al. [35] proposed a Bayesian optimization-based long short-term memory model (BO-LSTM) to construct a multi-source data fusion-driven crop growth feature extraction algorithm for winter wheat yield prediction, using satellite meteorological data from Hengshui, Hebei Province, China. Lin et al. [36] proposed a LSTM-based deep learning framework named DeepCropNet (DCN), which can hierarchically capture features to effectively predict corn yields at the county level in the United States. Haider et al. [37] proposed a LSTM neural network model for wheat prediction in Pakistan on a dataset obtained from the Federal Bureau of Statistics, Pakistan.

CNN-LSTM. Some other works have proposed hybrid deep learning frameworks, such as CNN-LSTM, to extract spatiotemporal relationship features for crop yield prediction. Khaki et al. [13] proposed a hybrid CNN-LSTM model, which accurately predicted crop yields across the entire Corn Belt in the United States. The proposed method significantly outperformed other popular methods such as LASSO, random forest and CNN. You et al. [38] used another CNN-LSTM model to capture spatiotemporal structures in data, and introduced a Gaussian process component to further improve accuracy. They also introduced a novel dimensionality reduction technique that enables the model to capture effective features in sparse training data. Saini et al. [39] proposed a deep learning approach called CNN-BI-LSTM-CYP for sugarcane yield prediction. The dataset was obtained from India's major sugarcane-producing states. Boppudi et al. [40] proposed an improved feature ranking fusion process for feature selection and a hybrid LSTM-DBN model for crop prediction in India.

GNN-RNN. The above deep learning models either consider spatial or temporal relationships, or a combination of both. However, they usually treat each county as an independent

and identically distributed (i.i.d.) sample in their models and do not fully explore the important role of geographical relationships in crop yield prediction. Fan et al. [14] introduced a GNN-RNN framework for crop prediction, which used GNN to extract geospatial features from input data and LSTM to extract temporal relationships. Experimental results showed that the performance of this model was significantly better than the existing deep learning models.

3. Background Knowledge

3.1. CNN

Convolutional neural network [41,42] is a type of feedforward neural network designed to extract features from data with grid-like structures. A CNN model, as depicted in Figure 1, typically consists of stacked convolutional layers, pooling layers and fully connected layers. In a CNN, the convolutional layers and pooling layers are responsible for mapping the original data to a hidden feature space, while the fully connected layers map the extracted features to the label space of the samples.

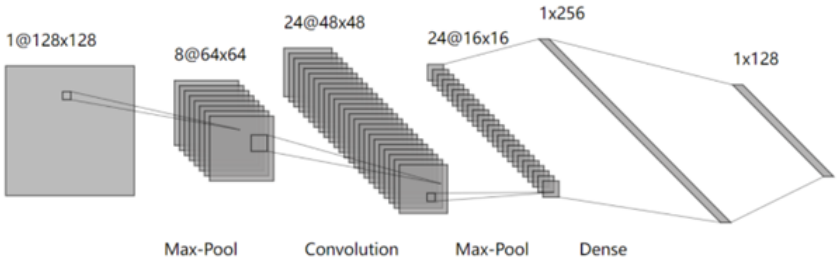

Figure 1. Convolutional neural network structure.

In the convolutional layer, information is extracted from input features using predefined filters (convolutional kernels). These filters scan the entire feature map from left to right and top to bottom with a predetermined stride. Typically, each convolutional layer contains multiple filters to extract features from the input.

Pooling layers downsample the input data to reduce the output data volume while retaining essential information, further simplifying the model. Common pooling layers include max pooling and average pooling.

A fully connected layer is usually found at the end of the neural network and is responsible for producing the final output prediction by connecting every neuron between two consecutive layers through weighted connections.

3.2. Graph Attention Networks

Graph neural networks [15,43,44] provide a general and effective framework for learning from graph-structured data. In a GNN, each node updates its state layer by layer by interacting with its neighbors, and the primary distinction among various GNN models lies in the specific techniques they use for node aggregation.

A graph attention network [16,17], as shown in Figure 2, is a specific type of graph neural network architecture that incorporates attention mechanisms to capture important relationships between nodes in a graph. Specifically, in GAT, every node updates its representation by attending to its neighbors, using its own representation as the query. This attention-based approach allows GAT to assign different weights to different neighbors, providing a more flexible and potentially more expressive model. GAT has become one of the most popular GNN architectures and is considered one of the most advanced neural architectures for graph learning.

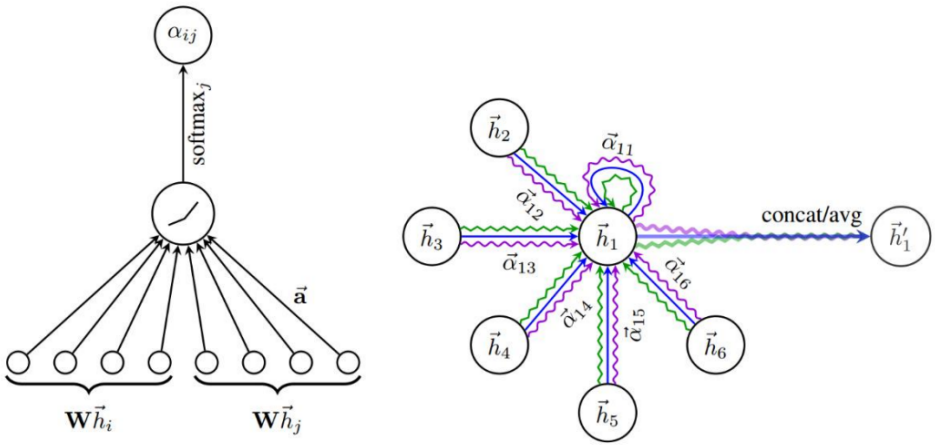

Figure 2. Graph attention model.

3.3. LSTM

LSTM [45] is a unique type of Recurrent Neural Network (RNN). Unlike traditional RNNs, LSTM addresses the issues of vanishing and exploding gradients by incorporating a cell state and three gate functions. These gate functions and the cell state effectively enhance the memory capacity of the recurrent unit. The structure of the LSTM module is illustrated in Figure 3.

The three gate functions are the forget gate, input gate and output gate. The forget gate primarily uses the sigmoid function to determine which historical information to discard and which to retain. The sigmoid function maps variables to values between 0 and 1, where a value closer to 1 indicates higher retention of information, while a value of 0 means discarding all information.

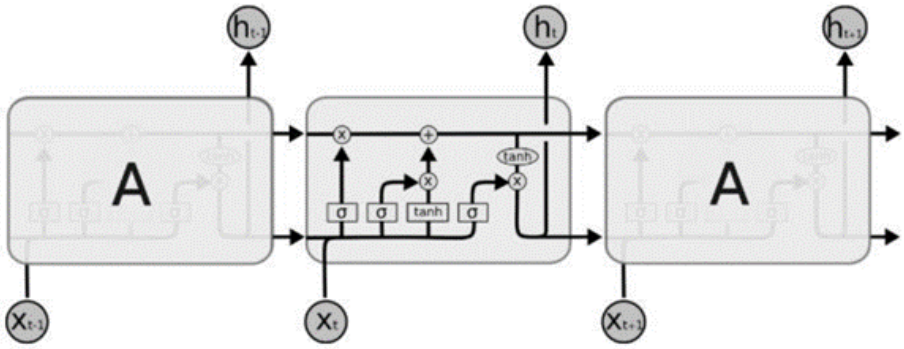

Figure 3. LSTM structure.

4. A CNN-GAT-LSTM Framework for Crop Yield Prediction

This section introduces a deep learning framework, CNN-GAT-LSTM, as shown in Figure 4 for predicting crop yields. Since crop yield prediction depends not only on temporal information but also on spatial information, e.g., climate data and soil data, this proposed model aims to capture both temporal and spatial information. The overall CNN-GAT-LSTM model is divided into three modules: the CNN module, the GAT module and the LSTM output module.

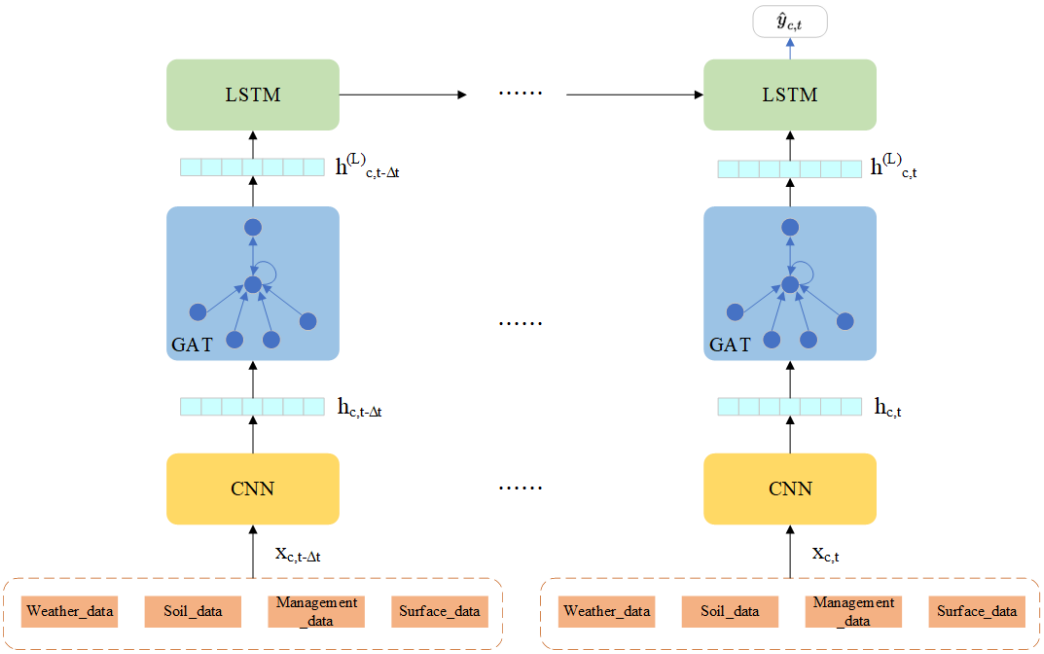

Figure 4. CNN-GAT-LSTM model.

4.1. CNN Module

The dataset used in this paper consists of weather data, soil data, management data and surface data. The structure of our CNN module is similar to that of [13]. Due to the unique structures of different data types, corresponding CNN structures are designed for each data type to effectively extract features. For weather data, which varies over time, we use a 1D-CNN module containing series of 1D convolution, ReLU and average pooling layers. This sequence is repeated four times followed by a fully connected layer. For soil data collected at various depths and independent of time, we employ another 1D-CNN module consisting of series of convolutional, RELU layers. This sequence is repeated three times followed by a fully connected layer. Management and surface data have relatively small dimensions, so no additional processing is required. Finally, we concatenate the extracted weather and soil features with the management and surface data to form the output of the CNN module, as shown in Figure 5.

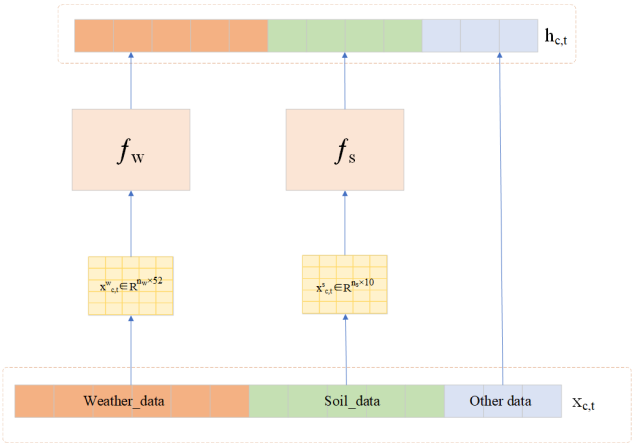

Figure 5. CNN module.

Weather data are represented as $x^w_{c,t} \in R^{n_w \times 52}$, where c represents county, t represents time and n_w represents the number of weekly weather features, and 52 represents the number of weeks in a year. Soil data is represented as $x^s_c \in R^{n_s \times 10}$, where n_s represents the number of soil features at each depth, and 10 represents the number of depths. Management

and surface data are represented as $x_c^m \in R^{n_m}$ and $x_c^e \in R^{n_e}$, respectively. All input features are represented as $x_{c,t} = \{x_{c,t}^w, x_c^s, x_c^m, x_c^e\}$. Weather and soil data are processed as follows.

$$h_{c,t}^w = f_w(x_{c,t}^w) \tag{1}$$

$$h_c^s = f_s(x_c^s) \tag{2}$$

where f_w and f_s are two separate 1D-CNN modules, used to process weather data and soil data, respectively. Since x_c^m and x_c^e have a small data size, they are directly concatenated with $h_{c,t}^w$, h_c^s and then output.

$$h_{c,t} = (h_{c,t}^w, h_c^s, x_c^m, x_c^e) \tag{3}$$

4.2. GAT Module

The geographical locations of different counties have a significant impact on yield prediction. For example, if one county has a good yield, it often means that its neighboring counties will also have good yields because weather and soil data between different counties tend to be similar and do not change abruptly. Therefore, effectively utilizing the geographical locations between different counties can greatly improve the performance of yield prediction models.

We introduce the graph attention network, which can assign weights according to the importance of neighbor nodes, to enhance the expressiveness of the model. However, we also observed that the attention distribution of the graph attention network is not influenced by the query node, which hinders the network from fitting the training data well. On the other hand, the latest GATv2 [17] model transforms static attention into dynamic attention, allowing different queries and keys to receive different attention, effectively enhancing the expressiveness and robustness of GAT. Therefore, we use GATv2 to extract geographical information from the data.

Formally, a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ consists of nodes $\mathcal{V} = 1, \ldots, N$ and edges $\mathcal{E} \in \mathcal{V} \times \mathcal{V}$. In our crop yield prediction task, each node is a county. Every node $c \in \mathcal{V}$ has an initial representation $h_{c,t} \in \mathbb{R}^d$ where t represents time, and each layer l updates the representation of node c by aggregating the representations of neighboring nodes $\mathcal{N}_c = \{c' \in \mathcal{V} | (c', c) \in \mathcal{E}\}$. A scoring function $e(\mathbb{R}^d \times \mathbb{R}^d) \to \mathbb{R}^d$ computes an attention score for each edge (c', c), which represents the importance of the features of the neighbor c' to the node c:

$$e(h_{c,t}, h_{c',t})^{(l)} = a^{(l)^\mathsf{T}} \text{LeakyReLU}([W^{(l)}[h_{c,t} || h_{c',t}]]) \tag{4}$$

where $h_{c,t}^{(0)} = h_{c,t}$, $l \in \{1, \ldots, L-1\}$ represents the layer, $a^{(l)} \in \mathbb{R}^{2d'}, W^{(l)} \in \mathbb{R}^{d \times d}$ are learned and LeakyReLU is an activation function. Denote $e(h_{c,t}, h_{c',t})^{(l)}$ by $e_{cc',t}^{(l)}$ for ease of notation. Then, a softmax function is used to normalize these attention scores across all neighbors $c' \in \mathcal{N}_c$.

$$\alpha_{cc',t}^{(l)} = \frac{\exp(e_{cc',t}^{(l)})}{\sum_{k \in \mathcal{N}_c} \exp(e_{ck,t}^{(l)})} \tag{5}$$

Finally, the weighted average of the transformed features of the neighboring nodes (followed by a nonlinearity σ) is used as the new representation of node c.

$$h_{c,t}^{(l+1)} = \sigma\Big(\sum_{c'=\mathcal{N}_c} a_{cc',t}^{(l)} W^{(l)} h_{c',t}\Big) \tag{6}$$

4.3. LSTM Module

Crop yield depends not only on weather and soil data from the whole local neighborhood in a year, but also on data from previous years. To obtain historical knowledge,

the output data from the GAT are embedded into LSTM to capture temporal information, as shown in Equation (7).

$$\hat{y}_{c,t} = r(h^{(L)}_{c,t-\Delta t}, \ldots, h^{(L)}_{c,t-1}, h^{(L)}_{c,t}) \tag{7}$$

where $h^{(L)}_{c,t'}$ represents the output of the last layer of GAT for the t'th year; Δt represents the length of the time series.

Finally, the log-cosh function is used as the loss function, which is less sensitive to outliers than mean square loss. Its representation is as follows.

$$L(\hat{y}_{c,t}, y_{c,t}) = \log(\cosh(\hat{y}_{c,t} - y_{c,t})) \tag{8}$$

where $y_{c,t}$ represents the actual yield value for county c in year t.

5. Experiment and Analysis

5.1. Experimental Setting

We conduct our experiments in a Linux environment, using PyTorch (version 1.10.2) to build the model and Conda (version 24.5.0) to manage the project environment. We use an NVIDIA GTX 4090 GPU (Santa Clara, CA, USA) with 24 GB of memory for the training process.

5.2. Dataset

Some early works have used satellite images or remote sensing data such as the normalized difference vegetation index (NDVI) to predict yield. However, these methods have difficulty in directly modeling the relationship between crop yield and the environmental factors that actually affect yield. This work does not use remote sensing datasets, but uses national environmental-related datasets for yield prediction. The dataset used in this work comes from [13], a publicly available dataset that includes weather data, soil data, management data and surface data from 1115 soybean-producing counties in 13 states in the United States, covering the years from 1980 to 2018. Each county corresponds to a sample, and its annual actual yield is measured in bushels per acre (bu/ac). Data from 1980 to 2016 (inclusive) serve as the training set, data from 2017 are used for validation, and data from 2018 are designated for testing.

The weather data include daily precipitation, solar radiation, snow water equivalent, maximum temperature, minimum temperature and water vapor pressure, with a spatial resolution of one square kilometer. The soil data are from the United States Soil Survey Geographic Database [46], including parameters such as wet bulk density, dry bulk density, clay percentage, plant available water capacity upper limit, plant available water capacity lower limit, hydraulic conductivity, organic matter percentage, pH, sand percentage and volumetric water content measured at depths of 0–5, 5–10, 10–15, 15–30, 30–45, 45–60, 60–80, 80–100, 100–120 and 120–150 centimeters. The management data are from the United States Department of Agriculture [47], primarily including the planting area percentages for each state starting from April.

5.3. Evaluation Metrics

In this experiment, three evaluation metrics are used to assess our model in the test year, i.e., RMSE, R^2 and Corr. RMSE represents the root of the mean squared error between the actual values and the predicted values, as shown in the following equation. The unit of RMSE is also bu/ac.

$$\text{RMSE} = \sqrt{\frac{\sum_c (y_c - \hat{y}_c)^2}{N}} \tag{9}$$

where y_c represents the actual yield value of county c, $\hat{y}_c$ represents the predicted value for county c and N is the total number of counties in the dataset.

R-squared (R^2) measures the strength of the relationship between the model and the data, and its formula is shown in the following equation.

$$R^2 = 1 - \frac{\sum_c (y_c - \hat{y_c})^2}{\sum_c (y_c - \bar{y})^2} \tag{10}$$

where $\bar{y}$ is the average yield. The numerator is the sum of squares of residuals (difference between the actual yield and the predicted yield), and the denominator is the total sum of squares of the difference between the actual yield and the average yield.

Pearson's correlation coefficient (Corr) measures the strength of the linear relationship between the actual values and the predicted values, and its formula is shown in the following equation.

$$\text{Corr} = \frac{\sum_c (y_c - \bar{y})(\hat{y_c} - \bar{\hat{y}})}{\sqrt{\sum_c (y_c - \bar{y})^2} \sqrt{\sum_c (\hat{y_c} - \bar{\hat{y}})^2}} \tag{11}$$

where $\bar{y}$ and $\bar{\hat{y}}$ represent the mean values of $y_c s$ and $\hat{y}_c s$, respectively.

5.4. Dataset Preprocessing

5.4.1. Data Normalization

This dataset includes weather, soil and other attributes, which vary significantly in numerical scales. The difference in scale can lead to the loss of important features during the training. For example, features with larger magnitudes can dominate the learning algorithm, making it less sensitive to other features. Additionally, the large differences in the value ranges can cause the objective function to flatten out, leading to inefficient gradient updates and slow convergence during training.

To address these issues, data standardization ensures that each feature contributes equally to the model, thereby eliminating the impact of different feature scales on model training and improving the convergence speed. We use Z-score normalization to rescale the data to have zero mean and unit variance.

$$x_{\text{standardized}} = \frac{x - x_{\text{mean}}}{x_{\text{std}}} \tag{12}$$

where $x_{\text{standardized}}$ represents the standardized value, x is the original value, x_{mean} is the mean and x_{std} is the standard deviation.

5.4.2. Adjacency Matrix

Since a graph attention neural network is used, it is necessary to construct an adjacency matrix based on the adjacency relationships between counties within states to incorporate geographic information during the training process.

Taking Illinois as an example, the geographic map of the state is presented in Figure 6. Some counties in the soybean-producing regions of Illinois are numbered, as illustrated in Figure 7. Based on their geographic information, an adjacency matrix has been constructed, shown in Figure 8. In this matrix, Adamas county, numbered 0, is adjacent to Brown county, which is numbered 4. Consequently, the entry in the 0th row and 4th column of the adjacency matrix in Figure 8 is 1, indicating that Adamas and Brown are neighboring counties. The adjacency matrix for other states is processed in a similar way to Illinois. Finally, this adjacency matrix is input into the model to capture the geographic information of different counties.

Figure 6. The geographic map of Illinois.

	State	County	ID_loc
1	illinois	adams	0
2	illinois	alexander	1
3	illinois	bond	2
4	illinois	boone	3
5	illinois	brown	4
6	illinois	bureau	5
7	illinois	calhoun	6
8	illinois	carroll	7
9	illinois	cass	8

Figure 7. Numbering of counties in Illinois.

	0	1	2	3	4	5	6	7	8
0	1	0	0	0	1	0	0	0	0
1	0	1	0	0	0	0	0	0	0
2	0	0	1	0	0	0	0	0	0
3	0	0	0	1	0	0	0	0	0
4	1	0	0	0	1	0	0	0	1
5	0	0	0	0	0	1	0	0	0
6	0	0	0	0	0	0	1	0	0
7	0	0	0	0	0	0	0	1	0
8	0	0	0	0	1	0	0	0	1

Figure 8. Adjacency matrix for Illinois.

5.5. Experimental Results and Comparisons

In this section, we compare the proposed CNN-GAT-LSTM model with other models such as LASSO, Ridge, Gradient-boosting, GRU, CNN-RNN and GNN-RNN using the evaluation metrics RMSE, R^2 and Corr. Although there are many machine learning methods applied to crop yield prediction, most rely on relatively small datasets. In contrast, this paper utilizes a nationwide dataset, making the experimental results not directly comparable.

In this experiment, the proposed model is compared with all other baseline models on the same dataset. We select the epoch and hyperparameters that yield the lowest RMSE in the validation year as the optimal epoch and hyperparameters. Finally, we choose the Adam optimizer with cosine decay, a learning rate of 5×10^{-5}, a weight decay of 1×10^{-5} and a batch size of 64 after experimenting with various hyperparameter configurations.

Figure 9 displays the training and validation results for each epoch, with Figure 9a illustrating the RMSE results, Figure 9b presenting R^2 results and Figure 9c showing the Corr results, respectively. We evaluate the performance of our model in the validation year at each epoch to identify the optimal epoch.

Tables 1 and 2 summarize the performance of all models in the validation year and the test year, respectively. As shown in the tables, our CNN-GAT-LSTM model outperforms all the baseline models above in terms of the evaluation metrics RMSE, R^2 and Corr. For example, compared with the latest GNN-RNN [14] model, which employs GNN to extract geographical information between different counties, our CNN-GAT-LSTM model introduces an attention mechanism, i.e., GAT, to better capture spatial information between neighboring counties, resulting in significant improvements across all evaluation metrics. Specifically, it reduces RMSE by 5%, increases R^2 by 6% and enhances Corr by 4% in the test year.

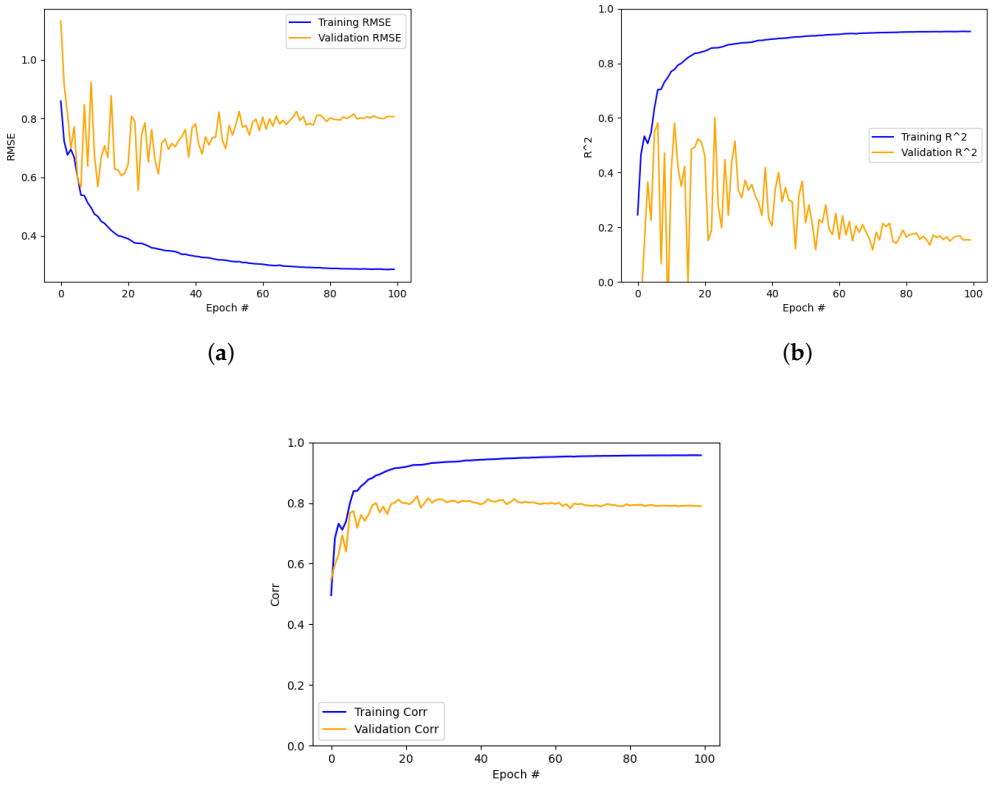

Figure 9. Training and validation results for each epoch. (**a**) Training and validation RMSE (bu/ac) results for each epoch. (**b**) Training and validation R^2 results for each epoch. (**c**) Training and validation Corr results for each epoch.

Table 1. Comparison of experimental results in the validation year.

Model	RMSE (bu/ac)	R^2	Corr
LASSO	0.7487	0.2650	0.6635
Ridge	0.6600	0.4288	0.6756
Gradient-Boosting	0.5979	0.5312	0.7533
GRU	0.6154	0.5034	0.7217
GNN	0.7731	0.2236	0.7358
CNN-RNN [13]	0.5686	0.5935	0.7866
GNN-RNN [14]	0.5828	0.5588	0.7886
CNN-GAT-LSTM (ours)	0.5544	0.6008	0.8226

Table 2. Comparison of experimental results in the test year.

Model	RMSE (bu/ac)	R^2	Corr
LASSO	0.7762	0.3235	0.7090
Ridge	0.8359	0.2155	0.6559
Gradient-Boosting	0.6591	0.5122	0.7817
GRU	0.6913	0.4634	0.7958
GNN	0.5654	0.6329	0.8164
CNN-RNN [13]	0.5882	0.6168	0.7954
GNN-RNN [14]	0.5161	0.6942	0.8374
CNN-GAT-LSTM (ours)	0.4623	0.7546	0.8775

6. Visualization

The experimental results of the three evaluation metrics, RMSE, R^2 and Corr, in Table 2 are calculated based on the predicted values and the actual values for all counties in the test year, and there is no intuitive explanation provided for the yield prediction results of individual counties. We understand that yield prediction for individual counties is helpful to users in making decisions. Figure 10 displays yield maps where different colors represent the yield values for each county in the test year. Figure 10a illustrates the actual yield for each county, Figure 10b presents the predicted yield for each county and Figure 10c shows the difference between the actual and predicted yields. As shown in Figure 10c, while the CNN-GAT-LSTM model effectively captures large-scale trends in crop yields, it occasionally produces overly smooth predictions within specific regions. For instance, it may overestimate yields in the central region while underestimating them in the western region. We estimate that this phenomenon is likely driven by the behavior of the GAT module. For example, the key feature of the GAT model is that it learns these spatial dependencies through its graph structure. If a particular county is overestimated (or underestimated) due to a certain reason, its neighboring nodes may also learn from this erroneous prediction, leading to a continuous bias in the yield predictions. Enhancing the model's ability to capture fine-scale variations without oversmoothing them is an important area for future development. We believe that this visualization can intuitively demonstrate the prediction effect of the proposed yield prediction model in different counties, thereby helping users make corresponding decisions.

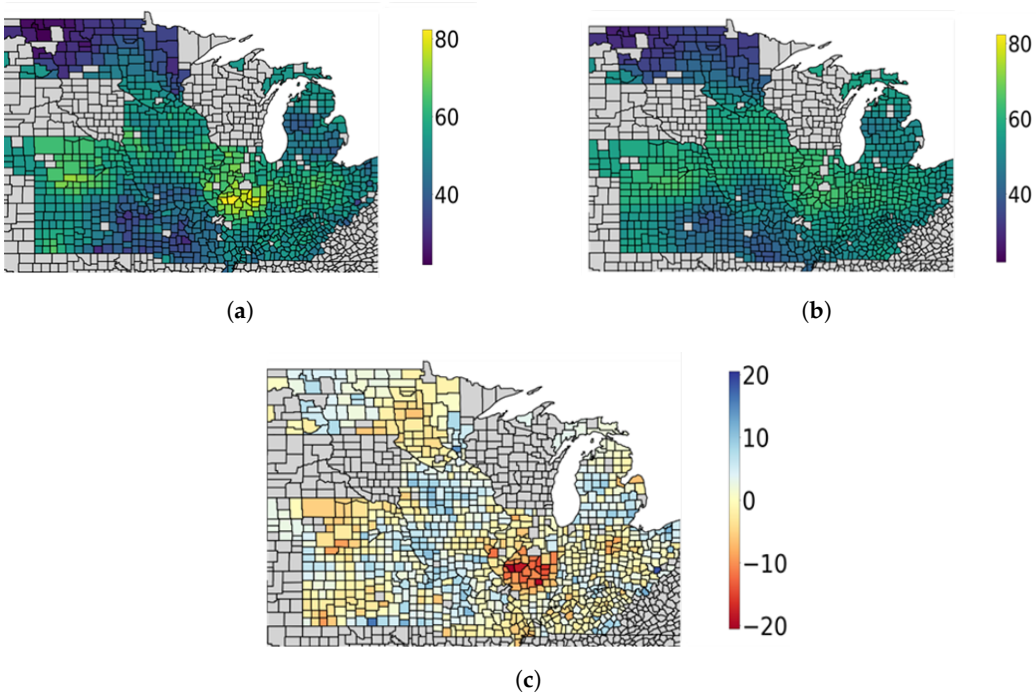

Figure 10. Comparison of actual yield, predicted yield and their difference in the test year. (**a**) Actual yield (bu/ac) in the test year. (**b**) Predicted yield (bu/ac) in the test year. (**c**) Difference between actual and predicted yields.

7. Ablation Experiments

Since crop yield prediction is not only related to time but also to geospatial relationship, the proposed CNN-GAT-LSTM model incorporates both the temporal and spatial relationships, thereby improving the accuracy of yield prediction.

To verify the importance of each module in the proposed CNN-GAT-LSTM model, this section conducts ablation experiments on various parts of the model, as shown in Table 3, to explore their contributions to the overall model. CNN refers to a convolutional neural network, which excludes the GAT and LSTM modules from the CNN-GAT-LSTM model. LSTM refers to the model where the CNN and GAT modules are removed from the CNN-GAT-LSTM model. CNN-GAT, GAT-LSTM and CNN-LSTM are obtained by removing a specific module from the CNN-GAT-LSTM model. In the ablation experiments, the hyperparameters of the models are consistent with those of the CNN-GAT-LSTM model.

Table 3. Results of ablation experiments.

Model	RMSE (bu/ac)	R^2	Corr
CNN	0.6584	0.5661	0.7988
LSTM	0.5430	0.7026	0.8459
CNN-LSTM	0.5301	0.6775	0.8439
CNN-GAT	0.6661	0.4905	0.7800
GAT-LSTM	0.5169	0.6932	0.8379
CNN-GAT-LSTM	0.4623	0.7546	0.8775

From Table 3, we can observe that different modules contribute differently to the overall model performance. The performance of the LSTM model is significantly higher than that of the CNN model. However, compared with the LSTM model, while RMSE of the CNN-LSTM model improves, R^2 and Corr decrease slightly. The CNN-GAT model shows a significant decrease in all evaluation metrics. Our analysis suggests that this may be due to the fact that GAT and CNN only capture spatial information within the dataset and cannot capture temporal information between different years without the use of LSTM. Therefore,

when LSTM is omitted, the performance of the CNN-GAT model is inferior to that of the CNN-LSTM model. Moreover, by removing the CNN module and incorporating the LSTM module, the performance of the GAT-LSTM model significantly surpasses that of all the aforementioned models. Finally, by combining all modules, the evaluation metrics of the CNN-GAT-LSTM model are significantly better than those of the models that omit certain parts. From these experiments, we can conclude that in the proposed CNN-GAT-LSTM model all modules are necessary and effective.

8. Limitations and Discussion

There is still room for improvement in this work. First, existing machine learning-based agricultural yield prediction work usually designs different machine learning methods and conducts experimental evaluations on self-collected datasets. The differences between these datasets are usually in location (United States, China, India), crop type and time range. Due to the different datasets used, these works are usually not directly comparable. Therefore, it is an important research topic to establish a large public agricultural yield dataset covering a wider geographical range, richer crop types and a longer time span to provide a benchmark for fair comparison of model performance. In other machine learning application fields such as computer vision and natural language processing, it is common practice to evaluate the model performance on public datasets. Therefore, the construction of a benchmark dataset for agricultural yield prediction is one of the future research directions.

Second, the current model, which integrates CNN, GAT and LSTM, has been specifically designed and trained using a national dataset that incorporates both geospatial and temporal data unique to the United States. While the framework demonstrates strong predictive performance within this dataset, its direct transferability to other regions or countries is limited due to differences in geographical, environmental and agricultural factors. However, the architecture itself is flexible and could be adapted to other contexts by retraining the model with data from the target region. This would involve reconstructing the geographic graph structure to reflect the spatial relationships specific to the new region. Further investigation into cross-regional transferability could be a valuable direction for future research.

Finally, current approaches to crop yield prediction involve creating custom machine learning models from scratch, which can be time-consuming and costly. Pre-trained models, e.g., BERT [48], RoBERTa [49] and BART [50], are deep learning models that have been trained on a large dataset in a self-supervised learning approach. These models can be further fine-tuned with domain-specific data for various downstream tasks. Pre-trained models provide a general foundation that can be fine-tuned and applied to various downstream machine learning tasks, forming a unified machine learning framework. Pre-trained models generally exhibit higher performance than custom models and have been widely used in fields such as natural language processing and computer vision. We believe that pre-trained models also hold great potential for crop yield prediction. In future work, we plan to apply pre-trained models to crop yield prediction to further enhance the model performance.

9. Conclusions

This paper proposes a deep learning framework, CNN-GAT-LSTM, for crop yield prediction that effectively handles both temporal and geospatial information in the data, thereby improving prediction accuracy. The proposed hybrid model integrates three components: CNN, GAT and LSTM. The CNN module extracts features from the input data for each county in each year, the GAT module captures the varying geographical relationships among counties and the LSTM module models the temporal information over multiple years. The experimental results show that the proposed CNN-GAT-LSTM model significantly outperforms existing state-of-the-art models on a nationwide dataset.

Author Contributions: Conceptualization, H.H. and Z.C.; methodology, H.H. and Z.C.; software, L.W. and W.L.; validation, H.H. and Z.C.; writing—original draft preparation, L.W. and W.L.; writing—review and editing, H.H.; supervision, H.H. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported in part by the "Pioneer" and "Leading Goose" R&D Program of Zhejiang under Grant No. 2023C01119.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data and code presented in this work are available at [https://github.com/lei330326/CGL], accessed on 29 October 2024.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Jones, J.W.; Antle, J.M.; Basso, B.; Boote, K.J.; Conant, R.T.; Foster, I.; Godfray, H.C.J.; Herrero, M.; Howitt, R.E.; Janssen, S.; et al. Toward a new generation of agricultural system data, models, and knowledge products: State of agricultural systems science. *Agric. Syst.* **2017**, *155*, 269–288. [CrossRef] [PubMed]
2. Gavasso-Rita, Y.L.; Papalexiou, S.M.; Li, Y.; Elshorbagy, A.; Li, Z.; Schuster-Wallace, C. Crop models and their use in assessing crop production and food security: A review. *Food Energy Secur.* **2024**, *13*, e503. [CrossRef]
3. Fritz, S.; See, L.; Bayas, J.C.L.; Waldner, F.; Jacques, D.; Becker-Reshef, I.; Whitcraft, A.; Baruth, B.; Bonifacio, R.; Crutchfield, J.; et al. A comparison of global agricultural monitoring systems and current gaps. *Agric. Syst.* **2019**, *168*, 258–272. [CrossRef]
4. Shahhosseini, M.; Hu, G.; Huber, I.; Archontoulis, S.V. Coupling machine learning and crop modeling improves crop yield prediction in the US Corn Belt. *Sci. Rep.* **2021**, *11*, 1606. [CrossRef] [PubMed]
5. Zhang, N.; Zhou, X.; Kang, M.; Hu, B.G.; Heuvelink, E.; Marcelis, L.F.M. Machine learning versus crop growth models: An ally, not a rival. *AoB Plants* **2022**, *15*, plac061. [CrossRef]
6. Murphy, K.P. *Machine Learning—A Probabilistic Perspective*; Adaptive Computation and Machine Learning Series; MIT Press: Cambridge, MA, USA, 2012; pp. I–XXIX, 1–1067, ISBN 0262018020.
7. Shingade, S.D.; Mudhalwadkar, R.P. Analysis of crop prediction models using data analytics and ML techniques: A review. *Multimed. Tools Appl.* **2024**, *83*, 37813–37838. [CrossRef]
8. Ransom, C.J.; Kitchen, N.R.; Camberato, J.J.; Carter, P.R.; Ferguson, R.B.; Fernández, F.G.; Franzen, D.W.; Laboski, C.A.M.; Myers, D.B.; Nafziger, E.D.; et al. Statistical and machine learning methods evaluated for incorporating soil and weather into corn nitrogen recommendations. *Comput. Electron. Agric.* **2019**, *164*, 104872. [CrossRef]
9. Sarr, A.B.; Sultan, B. Predicting crop yields in Senegal using machine learning methods. *Int. J. Clim.* **2023**, *43*, 1817–1838. [CrossRef]
10. Abdel-salam, M.; Kumar, N.; Mahajan, S. A proposed framework for crop yield prediction using hybrid feature selection approach and optimized machine learning. *Neural Comput. Appl.* **2024**, *36*, 20723–20750. [CrossRef]
11. Khaki, S.; Wang, L. Crop yield prediction using deep neural networks. *Front. Plant Sci.* **2019**, *10*, 621. [CrossRef]
12. Shen, Y.; Mercatoris, B.; Cao, Z.; Kwan, P.; Guo, L.; Yao, H.; Cheng, Q. Improving Wheat Yield Prediction Accuracy Using LSTM-RF Framework Based on UAV Thermal Infrared and Multispectral Imagery. *Agriculture* **2022**, *12*, 892. [CrossRef]
13. Khaki, S.; Wang, L.; Archontoulis, S.V. A cnn-rnn framework for crop yield prediction. *Front. Plant Sci.* **2020**, *10*, 1750. [CrossRef] [PubMed]
14. Fan, J.; Bai, J.; Li, Z.; Ortiz-Bobea, A.; Gomes, C.P. A GNN-RNN approach for harnessing geospatial and temporal information: Application to crop yield prediction. In Proceedings of the AAAI Conference on Artificial Intelligence, Virtually, 22 February–1 March 2022; Volume 36, pp. 11873–11881.
15. Hamilton, W.; Ying, Z.; Leskovec, J. Inductive representation learning on large graphs. *Adv. Neural Inf. Process. Syst.* **2017**, *30*, 1024–1034.
16. Velickovic, P.; Cucurull, G.; Casanova, A.; Romero, A.; Lio, P.; Bengio, Y. Graph attention networks. In Proceedings of the International Conference on Learning Representations, Vancouver, BC, Canada, 30 April–3 May 2018.
17. Brody, S.; Alon, U.; Yahav, E. How Attentive are Graph Attention Networks? In Proceedings of the 2022 International Conference on Learning Representations, Online, 25–29 April 2022.
18. Kuradusenge, M.; Hitimana, E.; Hanyurwimfura, D.; Rukundo, P.; Mtonga, K.; Mukasine, A.; Uwitonze, C.; Ngabonziza, J.; Uwamahoro, A. Crop Yield Prediction Using Machine Learning Models: Case of Irish Potato and Maize. *Agriculture* **2023**, *13*, 225. [CrossRef]
19. Nikhil, U.V.; Pandiyan, A.M.; Raja, S.P.; Stamenkovic, Z. Machine Learning-Based Crop Yield Prediction in South India: Performance Analysis of Various Models. *Computers* **2024**, *13*, 137. [CrossRef]
20. Suresh, A.; Kumar, P.G.; Ramalatha, M. Prediction of major crop yields of Tamilnadu using K-means and Modified KNN. In Proceedings of the 2018 3rd International Conference on Communication and Electronics Systems (ICCES), Coimbatore, India, 15–16 October 2018; pp. 88–93.

21. Karn, R.K.; Suresh, A. Prediction of Crops Based on a Machine Learning Algorithm. In Proceedings of the 2023 International Conference on Computer Communication and Informatics (ICCCI), Coimbatore, India, 23–25 January 2023; pp. 1–8.
22. Kumar, R.; Gupta, M.; Singh, U. Precision Agriculture Crop Recommendation System Using KNN Algorithm. In Proceedings of the 2023 International Conference on IoT, Communication and Automation Technology (ICICAT), Gorakhpur, India, 23–24 June 2023; pp. 1–6.
23. Das, P.; Jha, G.K.; Lama, A.; Parsad, R. Crop Yield Prediction Using Hybrid Machine Learning Approach: A Case Study of Lentil (Lens culinaris Medik.). *Agriculture* **2023**, *13*, 596. [CrossRef]
24. Sadenova, M.; Beisekenov, N.; Varbanov, P.S.; Pan, T. Application of Machine Learning and Neural Networks to Predict the Yield of Cereals, Legumes, Oilseeds and Forage Crops in Kazakhstan. *Agriculture* **2023**, *13*, 1195. [CrossRef]
25. Shahhosseini, M.; Martinez-Feria, R.A.; Hu, G.; Archontoulis, S.V. Maize Yield and Nitrate Loss Prediction with Machine Learning Algorithms. *Environ. Res. Lett.* **2019**, *14*, 124026. [CrossRef]
26. Mishra, P.; Khan, R.; Baranidharan, D.B. Crop Yield Prediction using Gradient Boosting Regression. *Int. J. Innov. Technol. Explor. Eng.* **2020**, *9*, 2293–2297. [CrossRef]
27. Pradeep, G.; Rayen, T.D.V.; Pushpalatha, A.; Rani, P.K. Effective Crop Yield Prediction Using Gradient Boosting to Improve Agricultural Outcomes. In Proceedings of the 2023 International Conference on Networking and Communications (ICNWC), Chennai, India, 5–6 April 2023; pp. 1–6.
28. Yasaswy, M.K.; Manimegalai, T.; Somasundaram, J. Crop Yield Prediction in Agriculture Using Gradient Boosting Algorithm Compared with Random Forest. In Proceedings of the 2022 International Conference on Cyber Resilience (ICCR), Dubai, United Arab Emirates, 6–7 October 2022; pp. 1–4.
29. Gandhi, N.; Armstrong, L.J.; Petkar, O. Predicting Rice crop yield using Bayesian networks. In Proceedings of the 2016 International Conference on Advances in Computing, Communications and Informatics (ICACCI), Jaipur, India, 21–24 September 2016; pp. 795–799.
30. Kim, N.; Ha, K.-J.; Park, N.-W.; Cho, J.; Hong, S.; Lee,Y.-W. A comparison between major artificial intelligence models for crop yield prediction: Case study of the midwestern united states, 2006–2015. *ISPRS Int. J. Geo-Inf.* **2019**, *8*, 240. [CrossRef]
31. Kumar, R.; Pandey, S. An accurate prediction of crop yield using hybrid deep capsule auto encoder with softmax regression. *Multimed. Tools Appl.* **2023**, *82*, 15371–15393. [CrossRef]
32. Subramaniam, L.K.; Marimuthu, R. Crop yield prediction using effective deep learning and dimensionality reduction approaches for Indian regional crops. *e-Prime-Adv. Electr. Eng. Electron. Energy* **2024**, *8*, 100611. [CrossRef]
33. Bhimavarapu, U.; Battineni, G.; Chintalapudi, N. Improved Optimization Algorithm in LSTM to Predict Crop Yield. *Computers* **2023**, *12*, 10. [CrossRef]
34. Wang, J.; Si, H.; Gao, Z.; Shi, L. Winter Wheat Yield Prediction Using an LSTM Model from MODIS LAI Products. *Agriculture* **2022**, *12*, 1707. [CrossRef]
35. Di, Y.; Gao, M.; Feng, F.; Li, Q.; Zhang, H. A New Framework for Winter Wheat Yield Prediction Integrating Deep Learning and Bayesian Optimization. *Agronomy* **2022**, *12*, 3194. [CrossRef]
36. Lin, T.; Zhong, R.; Wang, Y.; Xu, J.; Jiang, H.; Xu, J.; Ying, Y.; Rodriguez, L.; Ting, K.C.; Li, H. DeepCropNet: A deep spatial-temporal learning framework for county-level corn yield estimation. *Environ. Res. Lett.* **2020**, *15*, 034016. [CrossRef]
37. Haider, S.A.; Naqvi, S.R.; Akram, T.; Umar, G.A.; Shahzad, A.; Sial, M.R.; Khaliq, S.; Kamran, M. LSTM Neural Network Based Forecasting Model for Wheat Production in Pakistan. *Agronomy* **2019**, *9*, 72. [CrossRef]
38. You, J.; Li, X.; Low, M.; Lobell, D.; Ermon, S. Deep gaussian process for crop yield prediction based on remote sensing data. In Proceedings of the AAAI Conference on Artificial Intelligence, San Francisco, CA, USA, 4–9 February 2017; Volume 31.
39. Saini, P.; Nagpal, B.; Garg, P.; Kumar, S. CNN-BI-LSTM-CYP: A deep learning approach for sugarcane yield prediction. *Sustain. Energy Technol. Assess.* **2023**, *57*, 103263. [CrossRef]
40. Boppudi, S.; Jayachandran, S. Improved feature ranking fusion process with Hybrid model for crop yield prediction. *Biomed. Signal Process. Control* **2024**, *93*, 106121. [CrossRef]
41. LeCun, Y.; Boser, B.; Denker, J.S.; Henderson, D.; Howard, R.E.; Hubbard, W.; Jackel, L.D. Backpropagation applied to handwritten zip code recognition. *Neural Comput.* **1989**, *1*, 541–551. [CrossRef]
42. Li, Z.; Liu, F.; Yang, W.; Peng, S.; Zhou, J. A survey of convolutional neural networks: Analysis, applications, and prospects. *IEEE Trans. Neural Netw. Learn. Syst.* **2021**, *33*, 6999–7019. [CrossRef] [PubMed]
43. Gori, M.; Monfardini, G.; Scarselli, F. A new model for learning in graph domains. In Proceedings of the 2005 IEEE International Joint Conference on Neural Networks, Montreal, QC, Canada, 31 July–4 August 2005; IEEE: Piscataway, NJ, USA, 2005; Volume 2, pp. 729–734. [CrossRef]
44. Scarselli, F.; Gori, M.; Tsoi, A.C.; Hagenbuchner, M.; Monfardini, G. The graph neural network model. *IEEE Trans. Neural Netw.* **2008**, *20*, 61–80. [CrossRef] [PubMed]
45. Hochreiter, S.; Schmidhuber, J. Long Short-Term Memory. *Neural Comput.* **1997**, *9*, 1735–1780. [CrossRef] [PubMed]
46. gSSURGO [EB/OL]. (2019-12-6) [2023-12-10]. Soil Survey Staff. Gridded Soil Survey Geographic (gSSURGO). Available online: https://www.nrcs.usda.gov/resources/data-and-reports/gridded-soil-survey-geographic-gssurgo-database (accessed on 29 October 2024).
47. USDA-NASS. [EB/OL]. (2019-12-6) [2023-12-10]. USDA—National Agricultural Statistics Service. Available online: https://www.nass.usda.gov/ (accessed on 29 October 2024).

48. Devlin, J.; Chang, M.; Lee, K.; Toutanova, K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In Proceedings of the NAACL-HLT, Minneapolis, MN, USA, 2–7 June 2019; pp. 4171–4186.
49. Liu, Y.; Ott, M.; Goyal, N.; Du, J.; Joshi, M.; Chen, D.; Levy, O.; Lewis, M.; Zettlemoyer, L.; Stoyanov, V. Roberta: A robustly optimized bert pretraining approach. *arXiv* **2019**, arXiv:1907.11692.
50. Lewis, M.; Liu, Y.; Goyal, N.; Ghazvininejad, M.; Mohamed, A.; Levy, O.; Stoyanov, V.; Zettlemoyer, L. BART: Denoising sequence-tosequence pre-training for natural language generation, translation, and comprehension. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Online, 5–10 July 2020; Jurafsky, D., Chai, J., Schluter, N., Tetreault, J.R., Eds.; Association for Computational Linguistics: Stroudsburg, PA, USA, 2020; pp. 7871–7880.