Forecasting Agricultural Time Series Sensor Data Using Long-Short Term Memory Autoencoders

Towards Regenerative Farming

Derick Wagambula
Faculty of Information Technology
Beijing University of Technology, BJUT
Beijing, China
Email: wagambuladerick@gmail.com

***Abstract*—Conventional Agriculture is evolutionizing towards Regenerative Farming, which involves a range of techniques supported by innovative technologies to address climate change. Among them is the IoT (Internet of Things) technology in agriculture, which has seen continuous streams of data in real-time. From the use of drones to deployment of Wireless Sensors in the field, data is collected and transmitted via a communication channel to an Internet of Things platform. In this paper, we analyze the use of digital tools in regenerative farming, specifically soil sensors, and demonstrate this with the use of Long-Short Term Memory (LSTM) autoencoders to forecast future sensor readings based on historical data which can help a farmer make better farming decisions. LSTM networks are a type of Recurrent Neural Networks (RNNs) and have the ability to capture long-term dependencies, handle complex patterns in sequential data, and learn from past errors. This is evident through their use in predicting household power consumption, network traffic speed prediction, and predicting the crop yields. The proposed model is applied to the Cook Agronomy Farm (CAF) dataset, which contains field-scale sensor dataset for soil moisture and soil temperature at various levels. Using the Root Mean Square Error (RMSE) to evaluate the performance, the proposed model takes in multiple features as input and forecasts multiple steps and multiple parallel features. Traditional models such as Autoregressive Integrated Moving Average (ARIMA) have been used to forecast multivariate time series data. However, the proposed LSTM autoencoders perform with high accuracy and robustness in forecasting agricultural sensor data.**

Keywords-time series forecasting; LSTM autoencoders; precision farming; wireless sensors.

I. INTRODUCTION

Traditionally, farmers have been relying on the natural resources like rain [21] and sunshine for plant growth, as well as farmers instincts based on routine practices with emphasis on manual labor and simple tools like hoes. This generally leads to low yields and losses due to uncertainty caused by climate change. Over time, farming has evolved with farmers adopting modern farming practices, organizations and governments investing in advanced technology, and mechanization. Wireless sensors are deployed in the garden to measure soil moisture, temperature, Nitrogen, Phosphorus, and Potassium (NPK) and soil nutrients. The data collected can be analyzed to help improve farming practices.

Current prediction methods for agriculture sensor data focus on the real-time data to make recommendations. For example, in 2020, an IoT-based software system was proposed [1] for monitoring soil nutrients such as Nitrogen, Phosphorus, Potassium, soil pH, and temperature in real-time and can make recommendations regarding the quantity of water and fertilizers. Reashma and Pillai [3] discussed the use of machine learning techniques like Random Forest, Support Vector Machine (SVM) in three soil factors which are soil properties, soil moisture, and selection of crops. This is quite important when incorporated with domain knowledge to determine the course of action. This approach, however, seems hectic and would require much attention to the predictions. Time series data is a sequence of data collected over time intervals, allowing for tracking changes of a certain magnitude over time [5]. Sensor data collected over a period of time exhibits patterns such as trends, seasonal fluctuations, irregular cycles and occasional shifts in level or variability. Analyzing such series data helps us to extrapolate the dynamic patterns in the data to forecast future observations, estimate the effect of known exogenous interventions, and to detect unsuspected interventions. This has helped address real-world problems, like health monitoring, Web-Visitor traffic, and Network-wide traffic speed prediction [7].

Time series data can be univariate, i.e., data containing only one feature variable, or it can be multivariate i.e., data with multiple feature variables. Traditionally, time series forecasting includes methods such as K-Nearest Neighbor (KNN) [8] and Autoregressive Integrated Moving Average (ARIMA) [9], which can handle time-dependent data and achieve high forecasting accuracy on multiple frequencies (e.g., hourly, daily, weekly, monthly). However, the recent advancement of deep learning, neural network architectures, and compute capacity has seen breakthroughs in robustness and performance for a variety of problems including sequence-to-sequence-learning tasks [10][11] surpassing traditional forecasting models with data generated from retail, stock markets, traffics, to mention but a few, and are yet to gain momentum in the field of agriculture. Thus, in this paper, we aim to demonstrate the significance of deep learning in the shift towards regenerative learning particularly, building a model for multistep time series forecasting of agriculture sensor data. We use publicly available sensor data collected from different fields over a certain period of time and analyze it using the LSTM autoencoder.

The major contributions of this paper are:

- A detailed explanation on the significance of deep learning to achieve regenerative farming.
- An approach for multistep output forecasting using LSTM autoencoders.
- A demonstration of the proposed work using the publicly available sensor data [4] for validation.

The rest of this document is structured as follows: Section 2 explains regenerative farming in detail and briefly surveys the literature on LSTM and time series forecasting using LSTM autoencoders. Section 3 briefly formulates the challenge that we address in this study. Section 4 formally defines the proposed approach and provides the details of the implemented model. Section 5 describes the experimental evaluations and provides an interpretation of the results. Finally, Section 6 concludes the paper and discusses the next steps of this work.

II. REGENERATIVE FARMING

Regenerative farming is an evolution of conventional agriculture, where farmers rotate different types of crops over time reducing the use of water and other inputs and preventing land degradation and limiting pest infestations. It protects and improves soil biodiversity, climate resilience, and water resources while making farming more productive and profitable. From Africa to Asia, all the way to Europe and America, we have witnessed the impacts of climate change where some areas have had devastating impacts and others are yet to. This makes it hard for conventional farming to be profitable with high productivity. Hence, the need for more sustainable practices aimed at restoring soils and biodiversity, as seen in Figure 1. These practices, though they vary from place to place, include:

- Minimizing soil disturbances by adopting no-till or reduced till techniques.
- Planting cover crops between cash crops to prevent soil erosion and increase carbon inputs.
- Integrating livestock when possible.
- Diversifying crops in time and space by adopting intercropping.
- Precision application of biological and inputs.

Data-driven is a key part of regenerative agriculture, which involves the use of digital tools like wireless sensors connected to other IoT systems which collect the data, process and analyze it to provide a farmer with clear insight with what is happening on the ground. This, in turn, leads to the use of the optimal amount and the right type of product needed for a productive crop. Regenerative agriculture mitigates climate change through carbon dioxide removal, that is, it draws carbon from the atmosphere and sequesters it [12]. Deep learning has been widely adopted for time series data analysis [13] as models learn better with huge amounts of data, with the ability to extract both temporal and spatial features effectively. It offers significant potential to enhance regenerative agriculture practices through precise soil data analysis and forecasting. By analyzing soil sensor data, deep learning models can provide real-time updates on soil moisture, temperature, nutrient levels, and other critical parameters. We can forecast crop yields based on soil conditions [22] and determine the optimal timing and amount of fertilizer based on soil nutrient needs [6], reducing waste and environmental impact.

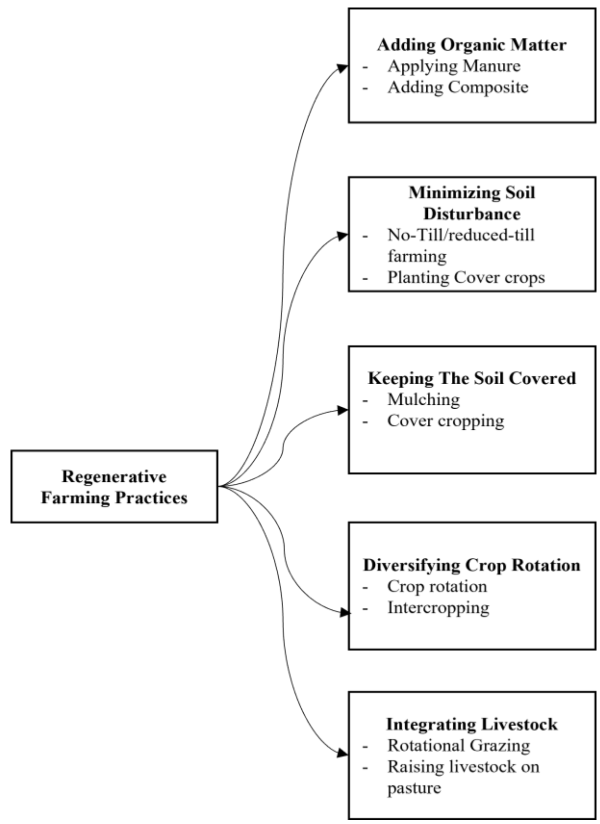

Figure 1. Core Principles of Regenerative Farming.

A. *Time Series Forecasting*

Time Series Forecasting plays an important role in weather prediction, stock market forecasting, etc., it is the use of a computer model to predict future values based on previously observed values, i.e., fitting a model to historical, time-stamped data in order to predict the future values. Traditional approaches include moving average, exponential smoothing, and ARIMA but recently, due to massive data generated by IoT devices, deep learning models like Recurrent Neural Networks (RNNs), Transformers, XGBoost, etc., have proven effective in extracting features from the data for forecasting. One of the most advanced models for forecasting time series is the Long-Short Term Memory (LSTM) Neural Network.

B. *Precision Farming*

Precision Farming is the use of technology to make farming more accurate, controlled, and optimized. It involves observing, measuring, and responding to inter- and intra-field variability. Precision Farming can help implement and maintain Regenerative Agriculture practices like precisely applying chemicals and monitoring soil health by leveraging technology to optimize resource use and maximize yield, while minimizing environmental impact.

1) *Long Short Term Memory Networks*

The Long Short-Term Memory (LSTM), illustrated in Figure 2, is a type of Recurrent Neural Network (RNN) designed to overcome the exploding and vanishing gradient descent with the ability to effectively capture temporal dependencies and to make accurate predictions. Through the standard recurrent layer, self-loops, and the internal unique gate structure, the LSTM network effectively improves the exploding and gradient vanishing problem existing in the traditional RNN. It has the form of a chain of repeated modules of neural networks, where each module includes three control gates, i.e., the forget gate, the input gate, and the output gate. As seen in Figure 2, each gate is composed of a sigmoid neural net layer and a pointwise multiplication operation. The sigmoid layers output numbers in the interval [0, 1], representing a portion of input information that should be let through. As the use of a RNN for time series data, the LSTM reads a sequence of input vectors $x = \{x_1, x_2, \ldots, x_t, \ldots\}$, where $x_t \in R_m$ represents an m-dimensional vector of readings form variables at time-instance t.

Given the new information x t in state t, the LSTM module works as follows. Firstly, it decides what old information should be forgotten by outputting a number within [0, 1], say f_t with

$$f_t = \sigma_1(\mathbf{W}_f .[h_{t-1}, x_t] + \mathbf{b}_f), \quad (1)$$

where h_{t-1} is the output in state $t-1$, W_f and b_f is the weight matrices and the bias of the forget gate. Then, x_t is processed before storing in cell state. The value i_t is determined in the input gate along with a vector of candidate values $\tilde{C}_t$ generated by a tanh layer at the same time to updated in the new cell state C_t, in which

$$i_t = \sigma_2 (\mathbf{W}_i .[h_{t-1}, x_t] + \mathbf{b}_i), \quad (2)$$

$$\tilde{C}_t = tanh(\mathbf{W}_c [h_{t-1}, x_t] + \mathbf{b}_c) \quad (3)$$

and

$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t, \quad (4)$$

where $(\mathbf{W}_i, \mathbf{b}_i)$ and $(\mathbf{W}_c, \mathbf{b}_c)$ are the weight matrices and the biases of input gate and memory cell state, respectively.

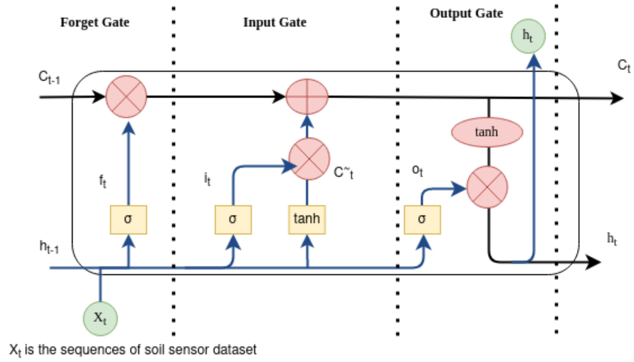

Figure 2. LSTM Network.

Finally, the output gate, which is defined by

$$o_t = \sigma_3 (\mathbf{W}_o .[h_{t-1}, x_t] + \mathbf{b}_o), \quad (5)$$

$$h_t = o_t * \tanh(C_t). \quad (6)$$

where $\mathbf{W}_o$ and $\mathbf{b}_o$ are the weight matrix and the bias of output gate, determines a part of the cell state being outputted. Figure 2 presents an illustration of the structure and the operational principle of a typical vanilla LTSM module. In this figure, the cell state runs straight down the entire chain, maintaining the sequential information in an inner state and allowing the LSTM to persist the knowledge accrued from subsequent time steps. Note that there are no weights and biases that can modify the cell state (Long Term memory). This allows it to flow through a series of unrolled units without causing the gradient to **explode** or **vanish**. Short-Term memories are directly connected to weights that can modify them. The first stage in the Long Short-term Memory unit determines what percentage of the Long-term memory is remembered. It is usually called the **Forget Gate**. The other part of the LSTM is usually called the **Input Gate**. The final stage of the LSTM updates the Short-term memory. The new long-term memory is used as input to the Tanh activation function. The previous three cases to determine the percentage of long-term memory to remember we use a sigmoid activation function. Because the new short-term memory is the output from this entire LSTM unit, this stage is called the **output gate**.

Besides forecasting, LSTMs have been used to solve other sequence learning problems like language modeling and translation, audio and video data analysis, handwriting recognition and generation among others.

2) *Autoencoders*

An autoencoder is a special type of feed forward neural network trained to efficiently compress (encode) input data down to its lower dimensional representation containing essential features or latent variables only (bottleneck), then reconstruct (decode) the original input from this compressed representation. Most autoencoders are used to solve AI related tasks like feature extraction [15], data compression

 ISBN: 978-1-68558-323-1

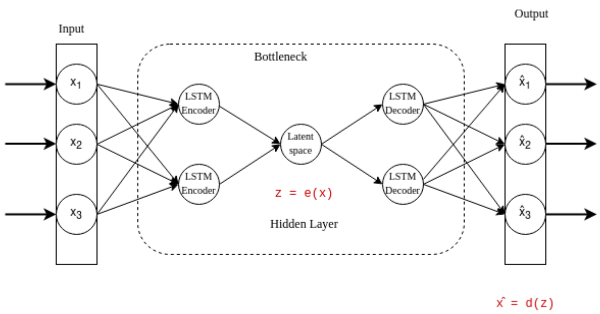

Figure 3. Illustration of LSTM Autoencoders.

[16][17], image denoising [18], anomaly detection [19], and facial recognition [20]. We use LSTM autoencoders, as illustrated in Figure 3, for multistep output forecasting of time series data.

LSTM autoencoders (Figure 3) utilize the capabilities of both the LSTM neural network and autoencoder which builds the LSTM network on the encoder and decoder schemes of Autoencoder. To forecast, we provide each one-dimensional time series to the model as a separate input sequence. The network then creates an internal representation of each input sequence that will together be interpreted by the decoder.

III. Dataset Description

The Cook Agronomy Farm (CAF) sensors folder [4] consists of a field-scale sensor network dataset for monitoring and modeling the special and temporal variation of soil moisture in dryland agricultural field. It includes hourly and daily measurements of Volumetric Water content (VW) sensor and soil Temperature (T) sensor readings, collected at 42 monitoring locations, and 5 depths (30, 60, 90, 120, and 150 cm) across Cook Agronomy Farm, collected from 2007 to 2016. As described below:

- VW_30cm: Volumetric Water readings at 30 cm depth (m^3/m^3)
- VW_60cm: Volumetric Water readings at 60 cm depth (m^3/m^3)
- VW_90cm: Volumetric Water readings at 90 cm depth (m^3/m^3)
- VW_1200cm: Volumetric Water readings at 120 cm depth (m^3/m^3)
- VW_150cm: Volumetric Water readings at 150 cm depth (m^3/m^3)
- T_30cm: temperature readings at 30 cm depth (C)
- T_60cm: temperature readings at 60 cm depth (C)
- T_90cm: temperature readings at 90 cm depth (C)
- T_120cm: temperature readings at 120 cm depth (C)
- T_150cm: temperature readings at 150 cm depth (C)

Note that not all these features will be used. For demonstration purposes, only a few features will be selected. Figure 4 is a plot for the sensor data between 2009 to 2012 and helps us to see the trends and seasonality.

A. Problem Statement

The CAF sensors data above is multivariate time series data describing the soil moisture and temperature sensor readings at different ground levels. Before planting any crop, it is important to have an idea of the crop requirements aforehand. We will use the data to address the question:

“We know the optimal soil water content and soil temperature for a certain crop at various stages of growth so, given the recent soil sensor readings, what is the expected soil sensor readings for the week ahead?”

This calls for the building of a predictive model to forecast the soil sensor readings over the next seven days. Technically, this is a multi-step time series problem, given the multiple forecast steps. Since we are dealing with multiple input variables, and predicting multiple steps ahead, this is called multi-step multivariate time series forecasting.

Note that, before we extract any useful insights from the data, we must clean the raw data by performing data wrangling and reshaping it into formats acceptable by the model for training. Good enough, the CAF data set is already separated into daily and hourly so in this paper, we are working with the daily sensor readings, not the hourly. We see from Table 1 that the data contains a lot of missing values. Table 2 shows how the dataset looks like after removing the missing values, converting the data type to numeric, and setting the date column as index.

The dataset has been split into training and test dataset. Furthermore, the train and test dataset has been organized into sequences of 7 days. The training dataset has 203 sequences, while the test dataset has 46 sequences. Remember it is a multivariate time series data, so we are dealing with 4 features.

Deep learning makes it easy for the farmer to analyze the soil and other parameters for better course of action such as knowing when to apply fertilizers, irrigating or performing drainage. In this paper, we are using LSTM Autoencoders on historical soil sensor readings to predict the possible future readings. The data in Table 2. Is not ready to be ingested into the LSTM model yet. We first normalize it using either the standard scaler or the minmax scaler to improve the model performance before splitting it into train and test dataset.

TABLE 1. RAW DATA CONTAINING MISSING VALUES.

Date	VW_30cm	VW_60cm	VW_90cm	T_30cm
04/20/2007	nan	nan	nan	nan
04/21/2007	nan	nan	nan	nan
04/22/2007	nan	nan	nan	nan
04/23/2007	nan	nan	nan	nan
04/24/2007	nan	nan	nan	nan

TABLE 2. SAMPLE DATA AFTER CLEANING.

Date	VW_30cm	VW_60cm	VW_90cm	T_30cm
05/21/2009	0.244	0.273	0.303	14.49
05/22/2009	0.243	0.276	0.308	13.61
05/23/2009	0.244	0.277	0.311	14.42
05/24/2009	0.244	0.279	0.313	15.15
05/25/2009	0.244	0.28	0.315	15.35

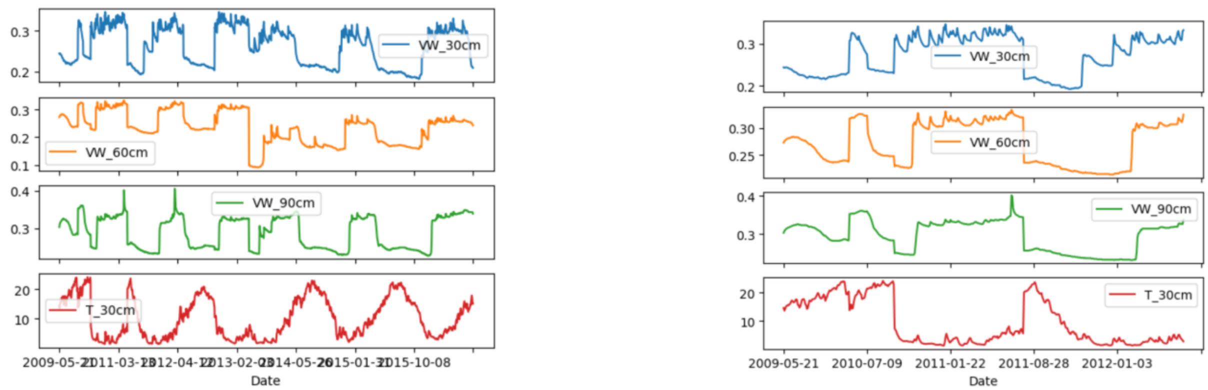

Figure 4. Soil Sensor Readings from 2009 to 2012.

B. *Evaluation Metrics*

Our model is going to forecast seven values, each representing the reading for a day in the week ahead. We will evaluate each forecasted timestep separately, doing so helps us to:

- Comment on the skill at a specific lead time (for example, +3 days versus +6 days) thereby helping us select an accurate forecast horizon.
- Contrast models based on their skills at different lead times

We will use the Root Mean Square Error (RMSE) as our performance metric. Evaluating the performance for each lead time from day 1 to day 7.

IV. MODEL ARCHITECTURE

We built the Encoder Decoder LSTM model to forecast Multiple Parallel Input and Multi-step multivariate time series sensor data using Tensorflow. Figure 5 shows the summary of the model architecture.

The Encoder-decoder architecture is good for sequence-to-sequence learning and as seen above, each is configured with 200 LSTM units. The first layer of the LSTM is the encoder, and the second one is the decoder. The latent vector is a 1-D array which is converted to the original number of features in the decoder level. The encoder is responsible for reading and interpreting the input, it compresses the input into the small representation of the original input (latent vector), which is then given to the decoder part as input for interpretation and forecasting. A RepeatVector layer is used to repeat the context vector obtained from the encoder. It is repeated for the number of future time steps (7 in our case) and fed to the decoder. The output received from the decoder in terms of each mixed. A fully connected Dense layer is applied to each time step via TimeDistributed wrapper, which separates the output for each time step.

The RepeatVector increases the dimension of the output shape by 1. TimeDistributed is kind of a wrapper and expects another layer as an argument. It applies this layer to every temporal slice of input and therefore allows to build models that have one-to-many, many-to-many architectures and expects inputs of at least 3 dimensions.

Model: "sequential"

Layer (type)	Output Shape	Param #
lstm (LSTM)	(None, 200)	164,000
repeat_vector (RepeatVector)	(None, 7, 200)	0
lstm_1 (LSTM)	(None, 7, 200)	320,800
time_distributed (TimeDistributed)	(None, 7, 100)	20,100
time_distributed_1 (TimeDistributed)	(None, 7, 1)	101

Figure 5. Summary of the LSTM Model

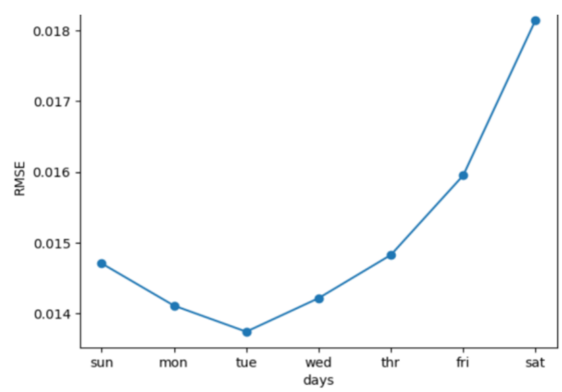

Figure 6. Plot of the RMSE for the 7 days.

A. *Model Performance*

We ran several experiments tuning the batch size, number of epochs, number of LSTM units and the time steps and obtained different results. However, when we set the batch size to 4 and ran 100 epochs, looking back 14 days to predict the next 7 days of the soil sensor readings (since we used the Root Mean Square Error as the evaluation metric), the model performed well with the overall RMSE of 0.015 (Figure 6).

 ISBN: 978-1-68558-323-1

V. Conclusion and Future Work

The agricultural sector is undergoing a significant transformation, driven by the urgent need for sustainable and regenerative practices. Among the latest innovations making waves are the technology-driven solutions which include use of advanced sensors and data analytics to assess soil quality, organic matter, and nutrient levels, guiding tailored interventions. Adopting precision farming by leveraging drones, satellite imagery, and Artificial Intelligence can help to optimize the use of resources, monitor crop health, increase yields and attaining regenerative farming in the process. The challenge, however, is that many farmlands are located in rural areas with poor network connectivity but with time, infrastructures are being put in place to improve connectivity. The LSTM autoencoders are state of the art networks and have been used in predicting indoor air quality, power load forecasting, among others. We just demonstrated its use in forecasting agriculture sensor data which is crucial in regenerative farming as it saves farming costs through effective use of resources. The experiments carried out to test the proposed model show that the performed well with high accuracy. This means farmers can confidently make better decisions depending on the forecast.

As future work, we will develop a Farm Management Information System (FMIS) using Fiware Technology and embed the proposed model to forecast. The FMIS will automate the farm activities thereby saving the farmer time and money.

References

[1] R. Madhumathi, T. Arumuganathan and R. Shruthi, "Soil NPK and Moisture analysis using Wireless Sensor Networks," 2020 11th International Conference on Computing, Communication and Networking Technologies (ICCCNT), Kharagpur, India, 2020, pp. 1-6, doi: 10.1109/ICCCNT49239.2020.9225547.

[2] M. S. Suchithra and M. L. Pai, "Improving the prediction accuracy of soil nutrient classification by optimizing extreme learning machine parameters, Information Processing in Agriculture," vol. 7, issue 1, 2020, pp. 72-82, ISSN 2214-3173.

[3] S. R. K. J. Reashma and A. S. Pillai, "Edaphic factors and crop growth using machine learning — A review," 2017 International Conference on Intelligent Sustainable Systems (ICISS), Palladam, India, 2017, pp. 270-274, doi: 10.1109/ISS1.2017.8389413.

[4] C. Gasch and D. Brown, "Data from: A field-scale sensor network data set for monitoring and modeling the spatial and temporal variation of soil moisture in a dryland agricultural field." Ag Data Commons. Dataset, 2017, https://doi.org/10.15482/USDA.ADC/1349683

[5] G. C. Tiao, "Time Series: ARIMA Methods," Editor(s): N. J. Smelser, P. B. Baltes, "International Encyclopedia of the Social & Behavioral Sciences, Pergamon," 2001, pp. 15704-15709, ISBN 9780080430768, https://doi.org/10.1016/B0-08-043076-7/00520-9.

[6] J. Gao, W. Zeng, Z. Ren, C. Ao, G Lei, T. Gaiser, and A. K. A Srivastava, "Fertilization Decision Model for Maize, Rice, and Soybean Based on Machine Learning and Swarm Intelligent Search Algorithms." *Agronomy* 2023, *13*, 1400. https://doi.org/10.3390/agronomy13051400

[7] B. Sirisha, S. Naveena, G. Palanki, and P. Snehaa, "Multivariate Time Series Sensor Feature Forecasting Using Deep Bidirectional LSTM," Procedia Computer Science, vol. 218, 2023, pp. 1374-1383, ISSN 1877-0509, https://doi.org/10.1016/j.procs.2023.01.116.

[8] Z. Cui, R. Ke, Z. Pu, and Y. Wang, "Stacked bidirectional and unidirectional LSTM recurrent neural network for forecasting network-wide traffic state with missing values." *Transportation Research Part C: Emerging Technologies* 118, 102674, 2020.

[9] S. Tajmouati, E. W. Bouazza, A. Bedoui, and A. Abarda, "Applying k-nearest neighbors to time series forecasting : two new approaches." https://doi.org/10.48550/arXiv.2103.14200, 2021.

[10] S. Mehrmolaei and M. R. Keyvanpour, "Time series forecasting using improved ARIMA," 2016 Artificial Intelligence and Robotics (IRANOPEN), Qazvin, Iran, 2016, pp. 92-97, doi: 10.1109/RIOS.2016.7529496.

[11] A. Graves, "Supervised Sequence Labelling with Recurrent Neural Networks." Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-24797-2, 2012.

[12] A. Graves, A. R. Mohamed, and G. Hinton, "Speech Recognition with Deep Recurrent Neural Networks" arXiv:arXiv:1303.5778, 2013.

[13] Regenerative agriculture, https://en.wikipedia.org/w/index.php?title=Regenerative_agriculture&oldid=1221967942 (last visited May 30, 2024)

[14] H. I. Fawaz, G. Forestier, J. Weber, L. Idoumghar, and P. K. Muller, "Deep learning for time series classification: a review. Data Mining and Knowledge Discovery." 33, 2019, 10.1007/s10618-019-00619-1.

[15] S. Hochreiter, J. Schmidhuber. (1997). "Long Short-term Memory." Neural computation. 9. pp. 1735-80. 10.1162/neco.1997.9.8.1735.

[16] B. G. Cambuí, R. G. Mantovani and R. Cerri, "Exploring Autoencoders for Feature Extraction in Multi-Target Classification," *2021 International Joint Conference on Neural Networks (IJCNN)*, Shenzhen, China, 2021, pp. 1-8, doi: 10.1109/IJCNN52387.2021.9533976.

[17] S. Sriram et al., "DeepComp: A Hybrid Framework for Data Compression Using Attention Coupled Autoencoder." *Arab J Sci Eng*. 47, pp. 10395–10410, 2022. https://doi.org/10.1007/s13369-022-06587-x

[18] R. K. Senapati, R. Badri, A. Kota, N. Merugu and S. Sadhul, "Compression and Denoising of Medical Images Using Autoencoders," *2022 International Conference on Recent Trends in Microelectronics, Automation, Computing and Communications Systems (ICMACC)*, Hyderabad, India, 2022, pp. 466-470, doi: 10.1109/ICMACC54824.2022.10093634.

[19] S. Jadhav and P. Kulkarni, "Image Denoising using Deep Auto-Encoder Network for Production Monitoring in Real-Time," *2021 International Conference on Advances in Electrical, Computing, Communication and Sustainable Technologies (ICAECT)*, Bhilai, India, 2021, pp. 1-7, doi: 10.1109/ICAECT49130.2021.9392554.

[20] T. Tziolas, K. Papageorgiou, T. Theodosiou, E. Papageorgiou, T. Mastos, and A. Papadopoulos, "Autoencoders for Anomaly Detection in an Industrial Multivariate Time Series Dataset." *Eng. Proc.,* 2022, *18*, 23. https://doi.org/10.3390/engproc2022018023

[21] R. Hammouche, A. Attia, S. Akhrouf, and Z. Akhtar, "Gabor filter bank with deep autoencoder based face recognition system", Expert Systems with Applications, vol. 197, 2022, 116743, ISSN 0957-4174, https://doi.org/10.1016/j.eswa.2022.116743.

[22] D. Molden et al., Water Availability and Its Use in Agriculture, Elsevier, 2011, pp. 707-732, ISBN

9780444531995, https://doi.org/10.1016/B978-0-444-53199-5.00108-1.

[23] S. M. G, S. Paudel, R. Nakarmi, P. Giri and S. B. Karki, "Prediction of Crop Yield Based-on Soil Moisture using Machine Learning Algorithms," *2nd International Conference on Technological Advancements in Computational Sciences (ICTACS)*, Tashkent, Uzbekistan, 2022, pp. 491-495, doi: 10.1109/ICTACS56270.2022.9988186.

[24] Y. Hasegawa and M. Tanaka, "Efficient Scheduling Focusing on the Duality of MPL Representatives," Proc. IEEE Symp. Computational Intelligence in Scheduling (SCIS 07), IEEE Press, Dec. 2007, pp. 57-64, doi:10.1109/SCIS.2007.357670.