International Journal of Innovative Computing, Information and Control
Volume **21**, Number **2**, April **2025**
ICIC International ©2025 ISSN 1349-4198
pp. **515–531**

ENHANCING SMART AGRICULTURE USING INTERNET OF THINGS AND TRANSFORMER MODEL

Ilvico Sonata and Yulyani Arifin

Computer Science Department, BINUS Graduate Program – Doctor of Computer Science
Bina Nusantara University
Jl. K. H. Syahdan No. 9, Kemanggisan, Palmerah, Jakarta 11480, Indonesia
ilvico@binus.ac.id; yarifin@binus.edu

Received May 2024; revised September 2024

Abstract. *Agricultural production levels are highly dependent on weather and soil conditions. To ensure the sustainability of agricultural production can use smart agriculture with Internet of Things (IoT) technology that currently develops rapidly along with the availability of various sensors to detect weather and soil conditions. This paper will propose a smart agriculture through monitoring agricultural soil moisture, temperature, and air humidity, as well as predicting temperature and air humidity data using IoT and the Transformer model. Data taken from these sensors will be analyzed using Transformer time series forecasting to see temperature and humidity trends for determining suitable plant types and predicting agricultural yield. The experimental results show that the Transformer model has a better level of accuracy compared to the machine learning model with MAE, MSE and RMSE of 0.027, 0.0011 and 0.034 for temperature prediction results and 0.057, 0.0053 and 0.073 for humidity prediction results.*
Keywords: Deep learning, Internet of Things, Smart agriculture, Transformer

1. **Introduction.** Agriculture is a very important aspect for a country. Agricultural development can improve the welfare of society in a country. Some examples of agricultural products include fruit, vegetables, rice, corn, etc. For this reason, sustainable agriculture needs to be developed. One way to achieve sustainable agriculture is through the application of technology in monitoring agricultural land [1]. The use of technology in monitoring agricultural land allows action to be taken to prevent crop failure and damage to agricultural land due to weather or unfavorable soil conditions. One of the developments in sustainable agriculture is switching to smart agriculture [2]. Smart agriculture changes the traditional agricultural model where data collection for decision making and action is minimal and monitoring is done manually into an agricultural system with automatic data collection and processing to obtain information that is important for automatic decision making and monitoring [3]. Smart agriculture is also suitable for supporting the sustainable of the agricultural production like the Sustainable Development Goals (SDGs) from The United Nations Educational, Scientific and Cultural Organization (UNESCO) especially Goals No 12 that is related with responsible consumption and production [4]. Using digital technologies can leverage for sustainability of agricultural production based on the United Nations Development Programme (UNDP) Report [5].

2. **Literature Review.** Mandal et al. [3] and Qazi et al. [6] in their research explains the use of IoT in the application of smart agriculture. In this smart agricultural application, IoT is used to collect soil and weather condition data using various sensors for the monitoring process and storing data in a database. Data from various sensors is used to

DOI: 10.24507/ijicic.21.02.515

regulate soil conditions such as adding water and fertilizer remotely and automatically. In addition, data from various sensors is used to observe air temperature and humidity. The use of IoT in smart agriculture can increase the efficiency of agricultural processes and increase agricultural production. Several other studies on smart agriculture using IoT have been carried out, among others, by Ayaz et al. [7], Saraswathi et al. [8], and Prathibha et al. [9].

To obtain more in-depth information regarding soil conditions and trends in changes in weather conditions based on information obtained from various sensors used in IoT, the data collected from these sensors is then analyzed to predict certain conditions based on historical data. This data is very important to use to predict agricultural yields and determine the types of crops that are suitable for planting according to the soil and weather conditions needed to produce maximum harvests [10]. Several methods for predicting these conditions currently use artificial intelligence technology such as machine learning and deep learning. The use of machine learning to process data obtained from IoT in smart agricultural applications has been carried out, among others, by Sanjana et al. [11]. In their research, they used K-Nearest Neighbors (KNN), Support Vector Machine (SVM) and logistic regression machine learning models to predict crop yields. The results of the experiments carried out show that the SVM model has the highest accuracy. Machine learning has also been used to process data from IoT through research conducted by Reddy et al. [12]. In their research, they used decision trees to predict the level of water requirements in smart agriculture. Both studies used temperature, humidity, and soil moisture sensors to measure weather and soil conditions.

The use of deep learning models in processing data obtained from various sensors used by IoT for smart agricultural applications has been discussed in depth by Zhu et al. [13]. In their discussion, they explained the concepts, applications, and opportunities for using deep learning models such as Artificial Neural Network (ANN), Back Propagation (BP), Convolutional Neural Network (CNN), Recurrent Neural Network (RNN) and Generative Adversarial Network (GAN). They concluded that the use of deep learning to process data obtained from various sensors is very useful and provides greater value in increasing agricultural productivity through plant disease detection, crop classification, weed identification, weather forecasting, and crop yield prediction. Altalak et al. [14] have also discussed the use of deep learning technology for smart agriculture. In their research, they discuss the use of various deep learning models such as CNN, RNN and Long Short-Term Memory (LSTM) as well as several combinations such as the combination of CNN and LSTM.

The combination of machine learning models and deep learning models in smart agriculture applications using IoT has been carried out by Durai and Shamili [15]. In their research, they used this artificial intelligence model for crop recommendations, weed identification, pesticide recommendations, and cost estimation. The development of the artificial intelligence model used was carried out through learning process using datasets obtained from various sensors used by IoT. The artificial intelligence models used include machine learning models such as SVM, Decision Tree Classifier, Random Forest Classifier, and XGBoost Classifier for the plant recommendation and pesticide recommendation processes. Deep learning models are used for weed identification. This deep learning model uses a pre-trained model from ResNet152V2. Meanwhile, for the cost estimation process, a regression model in machine learning, namely the XGBoost Regressor, is used.

The Transformer model was first introduced by Vaswani et al. [16] increasingly used as a sequence-to-sequence model that is very suitable for processing time series data. Research conducted by Sabililah and Adytia [17] shows that time series forecasting using a Transformer model has better accuracy results compared to other sequence to sequence models such as RNN and LSTM in predicting sea level. The use of Transformer for forecasting time series data has also been carried out by Fasvazahra et al. [18]. In their research, they used a Transformer to predict electricity utility loads. Experimental results show that the Transformer model outperforms the LSTM model. Research conducted by Li et al. [19] in predicting electricity consumption also shows that the Transformer model provides higher accuracy values than machine learning models such as Multi-Layer Perceptron (MLP), SVM and XGBoost.

From the results of the literature review above, the use of IoT combined with artificial intelligence models can be applied to creating smart agricultural models. With a better level of accuracy than RNN, CNN, LSTM models and several machine learning models such as SVM and XGBoost, Transformer models are still very little used in IoT applications for smart agriculture. The Transformer model is very suitable for estimating data taken from sensors connected to IoT because data from sensors connected to IoT is time series data such as humidity, temperature, and soil moisture data. This paper will propose the use of Transformer models to predict data obtained from IoT-connected sensors to implement smart agriculture to increase crop yields.

The next chapter of this paper will discuss the proposed method for developing smart agriculture including the types of sensors used, data processing and transmission using IoT, and the Deep Learning Transformer model. Chapter 4 will discuss each of the devices and deep learning models that will be used in more detail, Chapter 5 will discuss the experimental results, and finally the implementation plan and conclusions will be formulated from the experimental results in the implementation and conclusion chapters.

3. **Proposed Method.** We propose a smart agricultural model using IoT combined with a Transformer model to provide accurate information based on data received by IoT-connected sensors. The model we propose uses an ESP8266 Microcontroller Unit (MCU) as data acquisition whose task is to retrieve data from sensors connected to it. Sensors connected to the microcontroller will provide the data required for predictive analysis by the Transformer model. Figure 1 shows the proposed model.

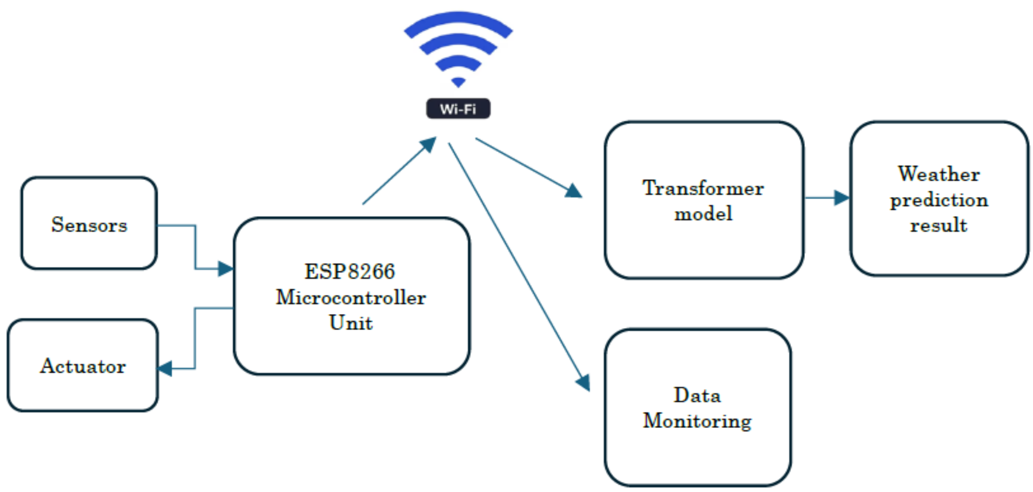

FIGURE 1. Proposed method

ESP8266 is a microcontroller that uses the nodeMCU platform which is an open source IoT platform and is very suitable for use as an IoT system controller because it can connect to Wi-Fi via 2.4 GHz Wi-Fi, using IEEE 802.11 BGN. The ESP8266 MCU will retrieve data from the sensors connected to it [20]. This data is processed by the ESP8266 as a basis for decision making in moving actuators such as motors or switches. All data is also sent via Wi-Fi to be displayed on the monitoring system and processed by the Transformer model for the prediction process.

The Transformer model will first be trained using a public dataset of temperature and humidity time series data. This Transformer model will continue to be trained using temperature and humidity data obtained directly through the IoT system. Through a continuous training process using real data, the accuracy level of the Transformer model in predicting temperature and humidity will increase.

4. **Technical Explanation.** The next sub-chapters will discuss each part of the proposed method.

4.1. **Sensor and actuator.** The sensors used in the proposed smart farming model prototype are temperature sensors, humidity sensors, and soil moisture sensors. The temperature and humidity sensor uses a DHT22 sensor which is an integrated sensor for measuring air temperature and humidity [21]. To measure soil humidity levels, the YL-69 soil moisture sensor is used [22].

DHT22 is a digital sensor that uses a voltage of 3.5V-5.5V. The resolution of this sensor is 16 bits for both temperature and humidity. The resulting measurement accuracy is $\pm 0.5°C$ for temperature and $\pm 1\%$ for humidity. The output from DHT22 is serial data with a temperature range of $-40°C$ to $80°C$ and a humidity range of 0% to 100%. Figure 2 shows the DHT22 sensor. The pins on the DHT22 consist of VCC (1), Data (2) and ground (3) pins.

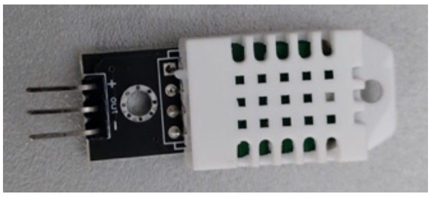

FIGURE 2. DHT22 temperature and humidity sensor

Soil moisture sensors are basically hygrometers. The sensor used to develop this smart agriculture model is the YL-69 sensor. This sensor has an operating voltage of 3.5V-5V. This sensor uses two electrodes embedded in the ground. These two electrodes will measure the resistance that occurs between the two electrodes, including due to the presence of water. In this way, this electrode will indirectly measure the water content in the soil or soil moisture. Apart from the electrodes, this sensor has an electronic part that functions to regulate the sensitivity of the measurements made. Sensitivity setting is done using a potentiometer. The output from this sensor is an analogue number 0-1023 which represents the measured soil moisture level. The YL-69 also has a digital output pin. This pin will be High and Low depending on the adjustment of the measured humidity level. This sensor has 4 pins, namely analogue output A0 (1), digital output D0 (2), ground (3) and VCC (4). Figure 3 shows the YL-69 sensor.

4.2. **Microcontroller and data acquisition.** The ESP8266 MCU is a microcontroller device that works using the nodeMCU platform. The ESP8266 MCU module is integrated with power saving Wi-Fi IEEE 802.11 b/g/n compliance modul so it can be directly connected to a Wi-Fi network with WPA/WPA2 security protocols. With this condition, the ESP8266 MCU is very suitable for use as an IoT device [23]. The ESP8266 MCU works with a voltage of 3.3V but can produce a voltage of 5V at its digital output and operates at a frequency of 80MHz (default) to 160MHz. The ESP8266

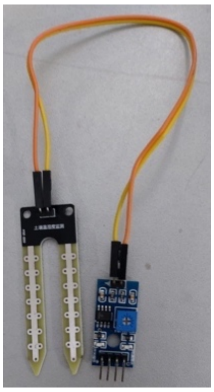

FIGURE 3. YL-69 soil moisture sensor

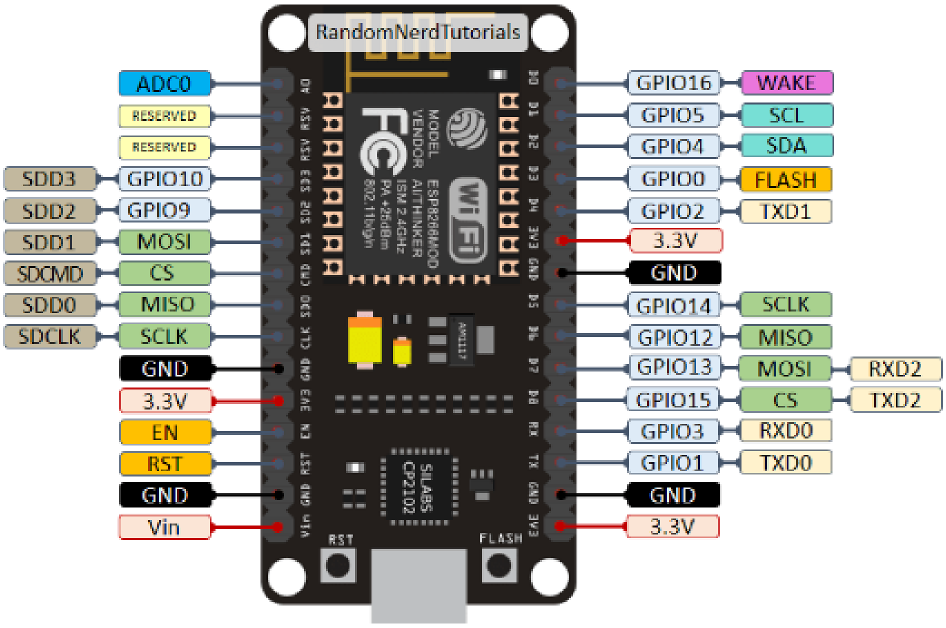

FIGURE 4. ESP8266 MCU

MCU consists of one analogue pin and 16 digital pins. For programming, the ESP8266 MCU can be programmed using the Arduino IDE including the syntax used and its libraries. Figure 4 shows the ESP8266 MCU along with its pin assignments taken from https://randomnerdtutorials.com/esp8266-pinout-reference-gpios/.

The ESP8266 MCU will retrieve data from temperature and humidity sensors, namely DHT22 and soil moisture sensors YL-69 connected to the analogue port. The ESP8266 MCU will send data via the Wi-Fi connected to it. The ESP8266 MCU will also check whether the soil moisture value is at the desired level or not. If soil moisture is below the desired level, the ESP8266 MCU will drive a servo motor connected to the digital output to open the water valve to agricultural land that needs it. If the soil moisture value matches the desired soil moisture level, the ESP8266 MCU will stop the motor so that the water flow stops as shown in Figure 5. Figure 6 shows the flowchart of the ESP8266 MCU program used.

4.3. **The monitoring system.** The monitoring system in the proposed smart farming system is used to visually display data trends obtained from the sensors used. The data that will be displayed includes data on soil moisture, air humidity and temperature, water

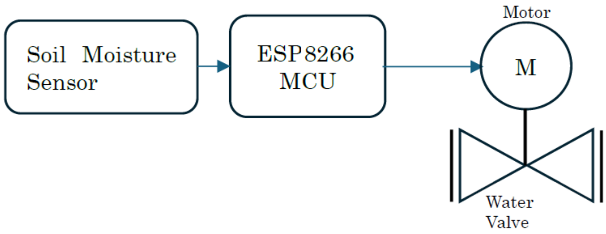

FIGURE 5. Water level monitoring system

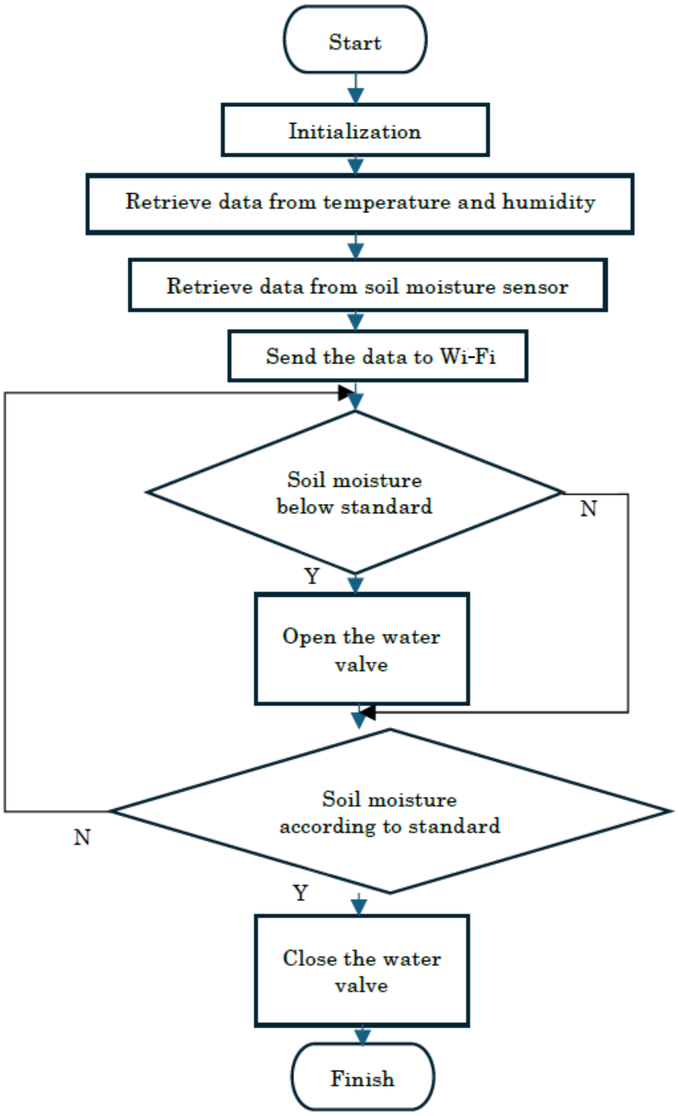

FIGURE 6. ESP8266 MCU flowchart

level and valve or sluice gate status. This display is in the form of a graph to see trends in air and ground conditions. This monitoring system uses LabVIEW software which is a virtual instrument developed by National Instrument [24].

This virtual instrument developed by LabVIEW has two interfaces for programming, namely block diagram and front panel. The block diagram consists of a function and connectivity of the instruments to be used. The front panel consists of an instrument panel and measurements that will be displayed in real time. Figure 7 shows an example

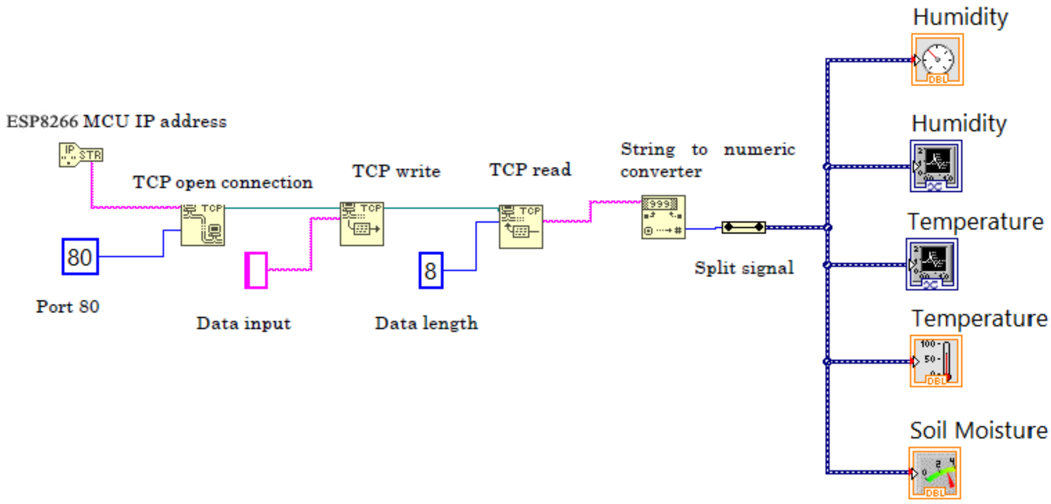

FIGURE 7. Example of a LabVIEW block diagram for smart agriculture

block diagram of the virtual instrument that will be used. The data from the sensor sent by the ESP8266 MCU via Wi-Fi is received by the LabVIEW virtual instrument using the TCP/IP port 80.

The communication process between LabVIEW and the ESP8266 MCU begins when LabVIEW opens a TCP open connection process according to the IP address and port used by the ESP8266 MCU. After the connection between LabVIEW and the ESP8266 MCU is established, the ESP8266 MCU will then send and write temperature, humidity and soil moisture data to LabVIEW via the TCP write process in LabVIEW. Then LabVIEW will read the data via TCP read with a maximum data length of 8 bytes. Next, the data in string form is converted into numeric and then broken down to be displayed on each appropriate instrument on the front panel.

The LabVIEW front panel used can be seen in Figure 8. The front panel used consists of monitoring graphs to monitor temperature and humidity history and display the current values of soil moisture, temperature, and humidity.

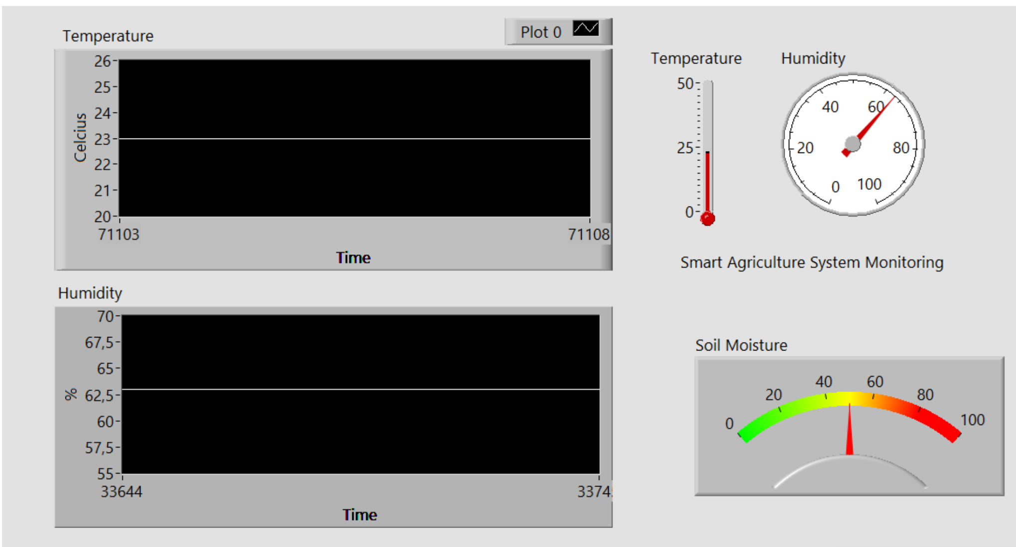

FIGURE 8. LabVIEW smart agriculture front panel

4.4. **Transformer model.** The Transformer model [16] in the proposed smart agriculture is used to predict air humidity and temperature through regression tasks. These prediction results will determine the crop yield predictions. The Transformer model consists of an encoder and decoder as shown in Figure 9. Air temperature and humidity data are received by the ESP8266 MCU and sent using the Internet to then be received and processed by the Transformer model. The Transformer model was trained using a dataset consisting of temperature and humidity data. This data is accompanied by a time label. The dataset S consisting of data on temperature T and humidity H equipped with a time

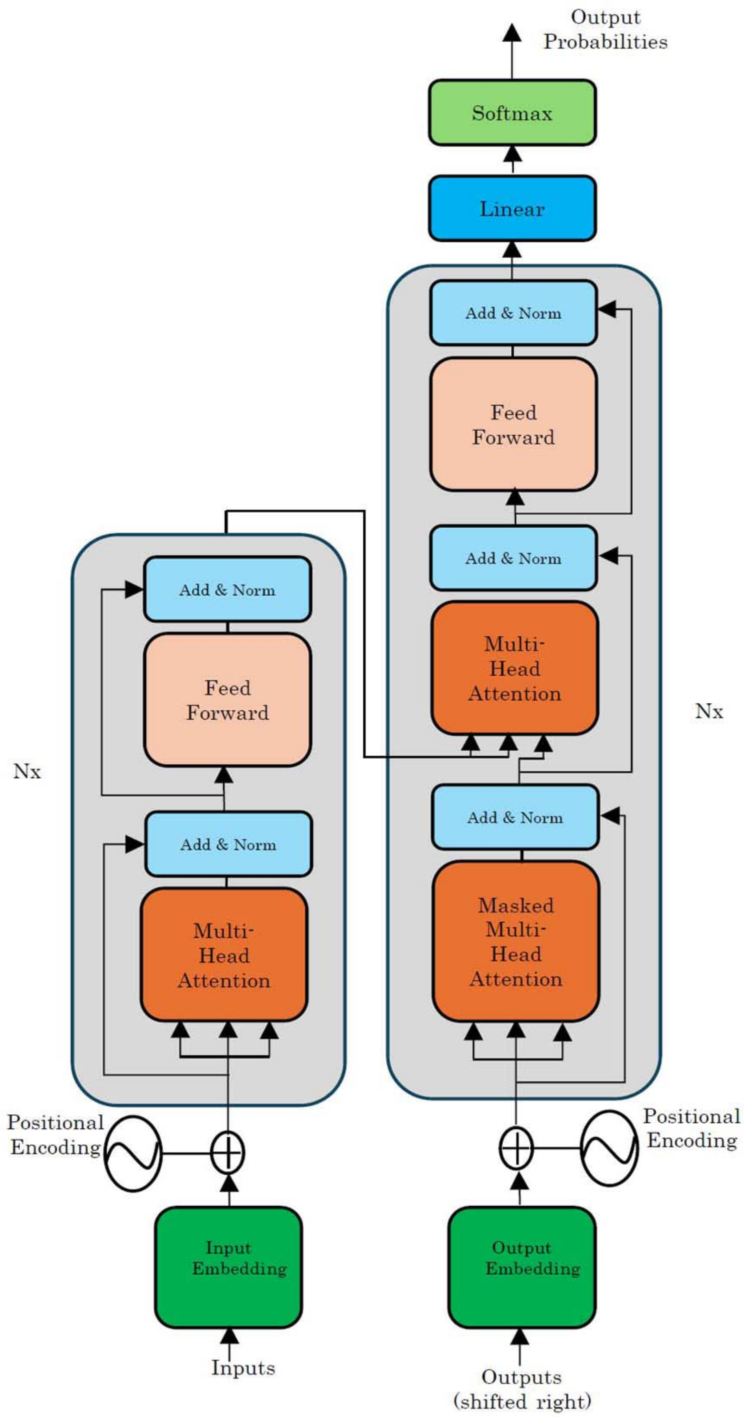

FIGURE 9. Transformer model architecture

label t can be written as $S = \{(T_1, H_1, t_1), (T_2, H_2, t_2), \ldots, (T_i, H_i, t_i)\}$. The data is then converted into a collection of vectors using the embedding process. These vectors represent the values of temperature and humidity. Each of these values can be called as a token that will be processed by the Transformer model. Because the data is time series data, each data has a close relationship with the nearest data. For this reason, the existence of Positional Encoding (PE) is very important, where positional encoding functions to provide relative positions between tokens [25]. In this Transformer model, the positional encoding used is the sine and cosine function to produce a unique vector for each token at each different position in the time series data as seen in Equations (1) and (2).

$$PE_{(pos,2i)} = \sin\left(\frac{pos}{1000^{\frac{2i}{d_{model}}}}\right) \tag{1}$$

$$PE_{(pos,2i+1)} = \cos\left(\frac{pos}{1000^{\frac{2i}{d_{model}}}}\right) \tag{2}$$

where d_{model} is the dimension of the embedding output. The next process is to give weight to each token. The weighting of each token is based on Query (Q), Key (K) and Value (V) which is calculated using a scaled dot product through an attention mechanism as shown in Equation (3). Figure 10 shows the attention mechanism process in the Transformer model.

$$\mathit{Attention}\,(Q, K, V) = \mathit{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{3}$$

where Q, K, and V are Query, Key, and Value vectors, and d_k is the dimensions of vector K. As can be seen in Figure 10, the dot product process is first carried out for vectors Q and K via matrix multiplication (MatMul). The results of matrix multiplication are then continued with a scaling process through a division process with the dimensions of vector K. The results of the scaling process are then entered into the Softmax activation function, and a dot product process is carried out with vector V to get the attention value for each token.

Increasing the performance of this attention mechanism is carried out by parallel work of several attention mechanisms through the process of linear projection Q, K, and V

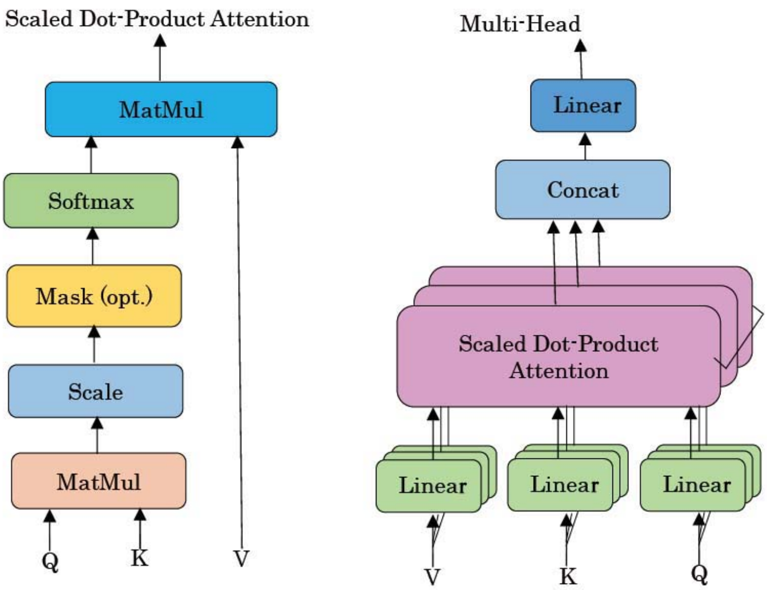

FIGURE 10. Attention mechanism

to the dimension of the vectors K (dk), K (dk), and V (dv) h times. Several attention mechanisms are concatenated to form Multi-Head Attention as shown in Figure 10 and can be calculated using Equations (4) and (5).

$$MultiHead\,(Q,K,V) = Concat\,(head_1, head_2, \ldots, head_h)\,W^O \tag{4}$$

$$head_i = Attention\left(QW_i^Q, KW_i^K, VW_i^V\right) \tag{5}$$

where $W_i^Q \in \mathbb{R}^{d_{model} \times d_k}$, $W_i^K \in \mathbb{R}^{d_{model} \times d_k}$, $W_i^V \in \mathbb{R}^{d_{model} \times d_v}$, $W^O \in \mathbb{R}^{hd_v \times d_{model}}$, W is the weight of vectors Q, K, and V obtained from the results of the Softmax function, and h refers to the number of heads used.

After going through the attention process, the next step is to carry out the normalization process on the Layer Normalization. This Layer Normalization is found in the encoder and decoder. The function of the Layer Normalization is to stabilize the training process by adjusting the gradient scale through standardizing the input value scale. In the Layer Normalization, there is also a process of adding up the Multi-Head Attention output with the original input from Multi-Head Attention which forms a residual connection. Through this process, values that are outside a certain scale can be adjusted to the desired scale so that the training process becomes more effective and efficient and produces training results with smoother gradients [26].

From the decoder side, the target label will enter the decoder which is shifted right and goes through the same process as the encoder. The final process of this Transformer is the process of predicting temperature and humidity values based on the training process using a dataset that has time labels. This prediction process uses a feed forward neural network with a Softmax activation function which can be calculated using Equation (6).

$$FeedForward\,(T,H,t) = softmax\,(0,(T,H,t)W_1 + b_1)\,W_2 + b_2 \tag{6}$$

where b_1 and b_2 are bias.

In general, the Transformer algorithm used in this model can be seen in Algorithm 1. When the program is first run, the Transformer model will load the dataset used. Next, pre-processing will be carried out on the dataset in the form of checking whether all the data is numeric and there are no empty fields. The dataset is then split into three parts, namely for the training, validation, and testing processes. The training process takes up 90% of the dataset and the validation process takes up 10% of the dataset. At this data pre-processing stage, the dataset scale is also changed using logarithmic return (log-return) to convert it into a relative value within a certain period. This change makes data distribution normally distributed and smoother. The goal of this change is to make the forecasting process more accurate.

After the split dataset process is complete, the next step is to set the hyper parameters of the Transformer model used. The hyper parameter settings that are carried out include setting the number of heads, the number of Transformer layers, and the number of neural network layers used. In addition, the batch size used for the training process, learning rate, number of epochs, activation function and loss function are also set.

After setting the hyper parameters of the Transformer model that will be used, the training and testing process is carried out for the entire dataset. In this training process, the weight metric parameters of the Transformer model are also updated. After the training process ends, the Transformer model has weight metric parameters that can generalize new data and can predict temperature and humidity according to time labels.

5. **Result and Discussion.** The proposed Transformer model was trained using 3,910 data records consisting of temperature, humidity, and time data. Dataset taken from https://www.kaggle.com/datasets/greegtitan/indonesia-climate. This dataset contains

Algorithm 1. Transformer model

load Temperature, humidity dataset
for each Temperature, humidity dataset
    delete empty records or field
    delete non-numeric data
    convert data using log-return
set train_size to 0.9
set val_size to 0.1
split dataset into training and validation according to train_size and val_size
set number of head to 3, number of layer to 2, number of dense to 256
set batch size to 250, learning rate to 0.0005, epoch to 50, activation function to Softmax, Loss function to Sparse Categorical Cross Entropy
for each record in dataset, **do** feature extraction using multi-head attention
  for epoch = 1: number of epochs
    for batch = 1: number of batches
      Generate another batch
      Train the model
      Validation the model
      Backpropagate the loss
update weight metric parameter
generate Temperature and humidity prediction
test the model

temperature, humidity, and rainfall data from 2010 to 2020 for several cities in Indonesia. In this experiment, temperature and humidity data were taken from the city of Bandung-West Java. Figure 11 shows the dataset of temperature graph and Figure 12 shows the dataset of humidity graph. The dataset was divided into 90% for training and 10% for validation. The training process used Google Colaboratory with VT100 GPU mode. In this experiment, the number of heads used was 3, 2 Transformer layers, 128 hidden units in the feed forward artificial neural network layer, 250 batch sizes and a learning rate of 0.0005. The time required for the training process using 50 epochs took 2 minutes.

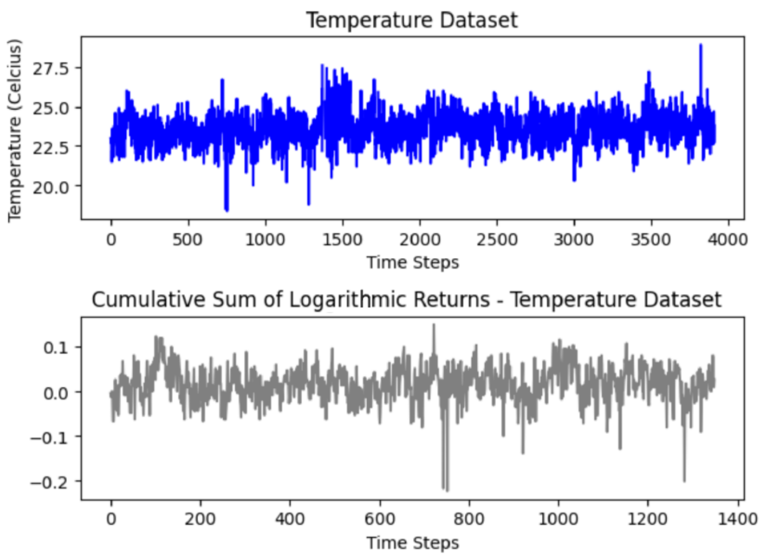

FIGURE 11. Temperature dataset

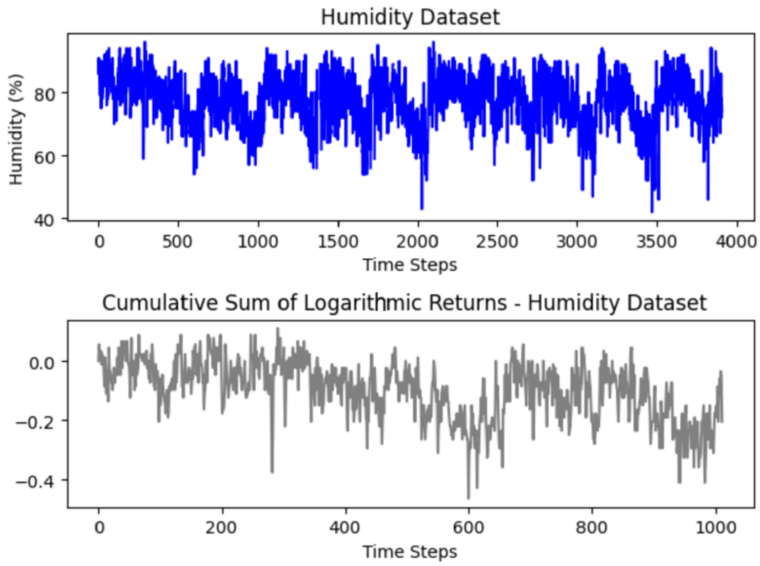

FIGURE 12. Humidity dataset

Figures 13 and 14 show the results of training the Transformer model to predict temperature and humidity. From Figure 13, the loss value resulting from the temperature prediction model validation process is 0.043, which is close to the loss value from the training process. Likewise in Figure 14, the loss value resulting from the validation process of the air humidity prediction model is 0.051, which is close to the loss value from the training process. This condition shows that the proposed Transformer model can be learned well using this dataset and there is no overfitting condition. An overfitting condition can occur if the loss value from the validation results is further away from the loss value from the training results [27]. In this way, the developed Transformer model can generalize and predict new data well [28].

Figure 15 shows the temperature prediction results using test data derived from 10% of the dataset. The prediction results in the form of a red graph look close to the black graph which is the actual temperature value. A similar thing can also be seen in Figure 16, where the red graph which is the result of the humidity prediction is close to the black graph which is the actual humidity value.

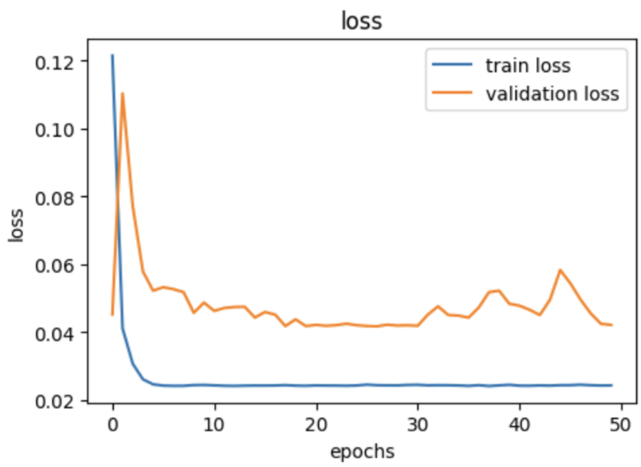

FIGURE 13. Temperature training result

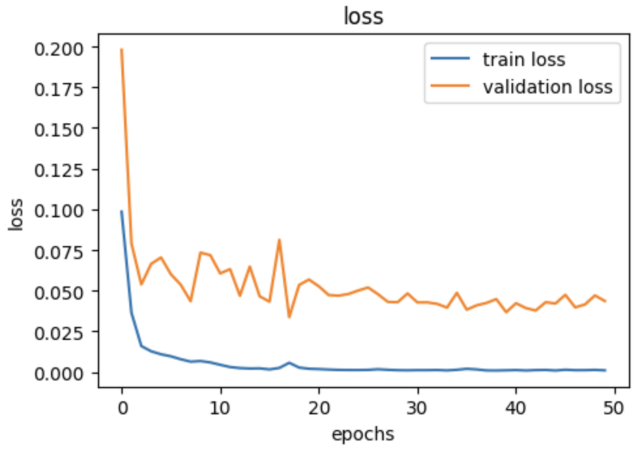

FIGURE 14. Humidity training result

Apart from using loss calculations from the training and validation process, the accuracy measurement process was also carried out using Mean Absolute Error (MAE), Mean Squared Error (MSE), and Root Mean Squared Error (RMSE) which are calculated using Equations (7)-(9).

$$MAE = \frac{\sum_{i=1}^{n} |y_i - x_i|}{n} \tag{7}$$

$$MSE = \frac{\sum_{i=1}^{n} (y_i - x_i)^2}{n} \tag{8}$$

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n} (y_i - x_i)^2}{n}} \tag{9}$$

where y is the actual temperature and humidity value and x is the predicted temperature and humidity value. The calculation of MAE, MSE and RMSE values using 10% of the dataset for testing is used to create prediction graphs in Figures 15 and 16. To see the level of accuracy of the proposed Transformer model, a comparison of accuracy was carried out

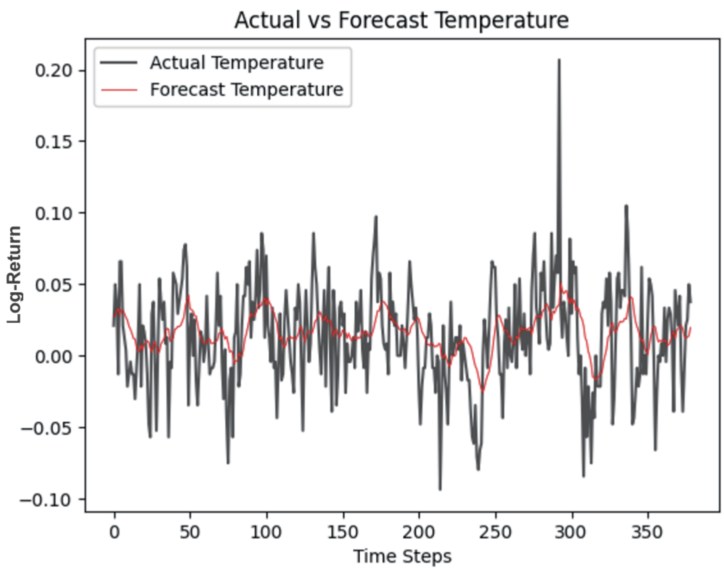

FIGURE 15. Temperature forecast

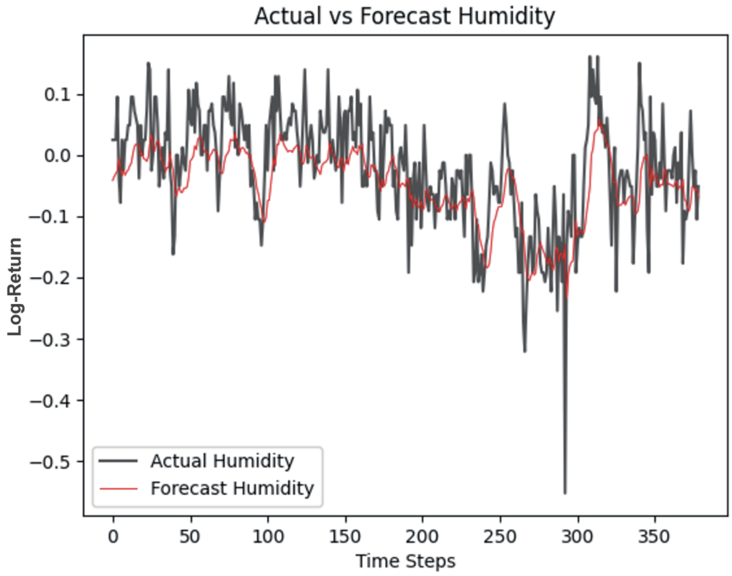

FIGURE 16. Humidity forecast

TABLE 1. Experimental result

Model		MAE	MSE	RMSE
Temperature	Transformer	**0.027**	**0.0011**	**0.034**
	SVR	0.032	0.0016	0.039
	XGBoost Regression	0.029	0.0014	0.037
Humidity	Transformer	**0.057**	**0.0053**	**0.073**
	SVR	0.063	0.0057	0.079
	XGBoost Regression	0.063	0.0055	0.077

using machine learning models, namely Support Vector Regression (SVR) and XGBoost Regression using the same dataset. The MAE, MSE and RMSE results for temperature and humidity using the Transformer, SVR and XGBoost models can be seen in Table 1.

From Table 1, the MAE, MSE and RMSE results from the Transformer model are smaller than those from the machine learning models, namely SVR and XGBoost Regression, both for temperature prediction and humidity prediction. This shows that the Transformer model has better accuracy compared to the SVR and XGBoost Regression models. The accuracy level of the Transformer model can outperform the SVR and XGBoost models because the Transformer model is very suitable for predicting time series data by paying attention to the closest data during the training process.

6. **Implementation.** Based on the results of the experiment above, this model can be implemented directly to farmers. Farmers who have land in an area can estimate the temperature and humidity for some time in the future. The results of the estimated temperature and humidity can be used to determine which plants are suitable for planting. For example, agricultural land in the Lembang – Bandung area, West Java – Indonesia, according to the forecasting results using the Transformer model for the following year will have a temperature and humidity ranging from 17°C-20°C and humidity between 80%-90%. Based on these data, farmers can determine which plants are suitable for planting. For example, strawberries are plants that are suitable for these conditions.

Daily operations of agricultural land require a long process [29]. By using this system, daily operations become simpler. The monitoring process is carried out using the

LabVIEW dashboard. This dashboard will display actual and historical temperature and humidity data. This monitoring is very important for predicting crop yields where changes in temperature and humidity greatly affect crop yields. Soil moisture control can also be done using an actuator in the form of a water valve drive motor. If the monitoring results show that soil moisture is below standard, the drive motor will open the water valve. If the soil moisture is in accordance with the needs, the drive motor will close the water valve. The use of this system can provide the right humidity value according to plant needs. With this system, it is hoped that the harvest results can be maximized.

7. **Conclusions.** Based on the experimental result, the proposed model has better accuracy and smaller prediction metric error compared with the comparison model. The monitoring of soil moisture, temperature and air humidity can be done effectively via IoT and LabVIEW virtual instrument. The irrigation process on agricultural land can be carried out more precision according to the needs and conditions of the soil. The Transformer model can be used with a good level of accuracy to predict temperature and humidity. The results of this prediction are very important for predicting agricultural yields and determining the type of plants to be planted so that they suit the temperature and humidity level requirements and can produce maximum agricultural production. The proposed model for smart agriculture using IoT and Transformer model can support the sustainability of the agricultural production.

The Transformer model training process can be carried out periodically using temperature and humidity data taken through the sensors used so that it will provide prediction results regarding temperature and humidity for the next period.

The main limitation in the implementation model on agricultural land is the lack of telecommunications infrastructure needed by IoT such as Internet and Wi-Fi networks. In addition, large agricultural land requires many sensors where the data obtained must be grouped and processed into several areas to be effective in the automation process. The wider the area of agricultural land, the more complicated the process of collecting data from sensors will be and requires special handling.

Data Availability Statement. The dataset used in this study uses a public dataset that can be accessed at the link https://www.kaggle.com/datasets/greegtitan/indonesia-climate.

Open Contributor. Ilvico Sonata conducted the design and experimentation of IoT and Transformer models. Yulyani Arifin conducted the research on SDG and the overall review of the paper.

REFERENCES

[1] M. Dhanaraju, P. Chenniappan, K. Ramalingam, S. Pazhanivelan and R. Kaliaperumal, Smart farming: Internet of Things (IoT)-based sustainable agriculture, *Agriculture (Switzerland)*, vol.12, no.10, DOI: 10.3390/agriculture12101745, 2022.

[2] Y. Abbassi and H. Benlahmer, The Internet of Things at the service of tomorrow's agriculture, *Procedia Computer Science*, pp.475-480, DOI: 10.1016/j.procs.2021.07.060, 2021.

[3] S. Mandal, A. Yadav, F. A. Panme, K. M. Devi and S. M. S. Kumar, Adaption of smart applications in agriculture to enhance production, *Smart Agricultural Technology*, vol.7, DOI: 10.1016/j.atech.2024.100431, 2024.

[4] K. Takács-György and I. Takács, Towards climate smart agriculture: How does innovation meet sustainability?, *Ecocycles*, vol.8, no.1, pp.61-72, DOI: 10.19040/ECOCYCLES.V8I1.220, 2022.

[5] D. Burra, J. Hildebrand, J. Giles, T. Nguyen, E. Hasiner, K. Schroeder, D. Treguer, A. Juergenliemk, A. Horst, A. Jarvis and W. Kropff, *Digital Agriculture Profile: Viet Nam*, Report, FAO, Rome, Italy, 2022.

[6] S. Qazi, B. A. Khawaja and Q. U. Farooq, IoT-equipped and AI-enabled next generation smart agriculture: A critical review, current challenges and future trends, *IEEE Access*, vol.10, pp.21219-21235, DOI: 10.1109/ACCESS.2022.3152544, 2022.

[7] M. Ayaz, M. Ammad-Uddin, Z. Sharif, A. Mansour and E. H. M. Aggoune, Internet-of-Things (IoT)-based smart agriculture: Toward making the fields talk, *IEEE Access*, vol.7, pp.129551-129583, DOI: 10.1109/ACCESS.2019.2932609, 2019.

[8] R. V. Saraswathi, J. Sridharani, P. S. Chowdary, K. Nikhil, M. S. Harshitha and K. M. Sai, Smart farming: The IoT based future agriculture, *2022 4th International Conference on Smart Systems and Inventive Technology (ICSSIT)*, Tirunelveli, India, pp.150-155, DOI: 10.1109/ICSSIT53264.2022.9716331, 2022.

[9] S. R. Prathibha, A. Hongal and M. P. Jyothi, IoT based monitoring system in smart agriculture, *2017 International Conference on Recent Advances in Electronics and Communication Technology (ICRAECT)*, Bangalore, India, pp.81-84, DOI: 10.1109/ICRAECT.2017.52, 2017.

[10] C. Sagana, P. Keerthika, M. Thangatamilan, R. Kamali, K. Nanthini and S. Maghathani, Identification of suitable crop based on weather condition, *2022 International Conference on Computer Communication and Informatics (ICCCI)*, Coimbatore, India, DOI: 10.1109/ICCCI54379.2022.9740905, 2022.

[11] G. Sanjana, N. M. Davasam and N. M. Krishna, Smart farming using IoT and machine learning techniques, *2020 IEEE Bangalore Humanitarian Technology Conference (B-HTC)*, Vijiyapur, India, DOI: 10.1109/B-HTC50970.2020.9297916, 2020.

[12] K. S. P. Reddy, Y. M. Roopa, K. Rajeev L.N. and N. S. Nandan, IoT based smart agriculture using machine learning, *2020 2nd International Conference on Inventive Research in Computing Applications (ICIRCA)*, Coimbatore, India, pp.130-134, DOI: 10.1109/ICIRCA48905.2020.9183373, 2020.

[13] N. Zhu et al., Deep learning for smart agriculture: Concepts, tools, applications, and opportunities, *International Journal of Agricultural and Biological Engineering*, vol.11, no.4, pp.21-28, DOI: 10.25165/j.ijabe.20181104.4475, 2018.

[14] M. Altalak, M. A. Uddin, A. Alajmi and A. Rizg, Smart agriculture applications using deep learning technologies: A survey, *Applied Sciences (Switzerland)*, vol.12, no.12, DOI: 10.3390/app12125919, 2022.

[15] S. K. S. Durai and M. D. Shamili, Smart farming using machine learning and deep learning techniques, *Decision Analytics Journal*, vol.3, 100041, DOI: 10.1016/j.dajour.2022.100041, 2022.

[16] A. Vaswani et al., Attention is all you need, *arXiv Preprint*, arXiv: 1706.03762, 2023.

[17] R. N. Sabililah and D. Adytia, Time series forecasting of sea level by using Transformer approach, with a case study in Pangandaran, Indonesia, *2023 IEEE 8th International Conference for Convergence in Technology (I2CT 2023)*, DOI: 10.1109/I2CT57861.2023.10126216, 2023.

[18] I. A. Fasvazahra, D. Adytia and A. A. Simaremare, Electricity time series forecasting by using Transformer with case study in Jakarta Banten, *2022 5th International Conference on Computer and Informatics Engineering (IC2IE 2022)*, pp.167-172, DOI: 10.1109/IC2IE56416.2022.9970104, 2022.

[19] X. Li, Y. Zhong, W. Shang, X. Zhang, B. Shan and X. Wang, Total electricity consumption forecasting based on Transformer time series models, *Procedia Computer Science*, pp.312-320, DOI: 10.1016/j.procs.2022.11.180, 2022.

[20] M. Kashyap, V. Sharma and N. Gupta, Taking MQTT and NodeMcu to IOT: Communication in Internet of Things, *Procedia Computer Science*, pp.1611-1618, DOI: 10.1016/j.procs.2018.05.126, 2018.

[21] M. J. Alam, S. A. Rafi, A. A. Badhan, M. N. Islam, S. I. Shuvo and A. M. Saleque, Low cost IoT based weather station for real-time monitoring, *2020 IEEE 2nd International Conference on Circuits and Systems (ICCS2020)*, pp.127-133, DOI: 10.1109/ICCS51219.2020.9336596, 2020.

[22] W. Y. Tan, Y. L. Then, Y. L. Lew and F. S. Tay, Newly calibrated analytical models for soil moisture content and pH value by low-cost YL-69 hygrometer sensor, *Measurement (Lond)*, vol.134, pp.166-178, DOI: 10.1016/j.measurement.2018.10.071, 2019.

[23] J. Mesquita, D. Guimarães, C. Pereira, F. Santos, L. Almeida and L. Almeida, Assessing the ESP-8266 WiFi module for the Internet of Things, *2018 IEEE 23rd International Conference on Emerging Technologies and Factory Automation (ETFA)*, Turin, Italy, pp.784-791, DOI: 10.1109/ETFA.2018.8502562, 2018.

[24] K. Kishkin, B. Ganev, D. Arnaudov and M. B. Marinov, LabVIEW based system for determining energy flows in energy storage system, *Proc. of 2021 30th International Scientific Conference Electronics*, DOI: 10.1109/ET52713.2021.9579768, 2021.

[25] Q. Zhang et al., An empirical study on the impact of positional encoding in Transformer-based monaural speech enhancement, *2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp.1001-1005, DOI: 10.1109/ICASSP48485.2024.10446337, 2024.

[26] R. Xiong et al., On layer normalization in the Transformer architecture, *arXiv Preprint*, arXiv: 2002.04745, 2020.

[27] A. B. Shanmugavel, V. Ellappan, A. Mahendran, M. Subramanian, R. Lakshmanan and M. Mazzara, A novel ensemble based reduced overfitting model with convolutional neural network for traffic sign recognition system, *Electronics (Switzerland)*, vol.12, no.4, DOI: 10.3390/electronics12040926, 2023.

[28] C. M. Gevaert and M. Belgiu, Assessing the generalization capability of deep learning networks for aerial image classification using landscape metrics, *International Journal of Applied Earth Observation and Geoinformation*, vol.114, DOI: 10.1016/j.jag.2022.103054, 2022.

[29] D. Wang, H. Shi, L. Sun and Y. Du, A collaborative allocation approach for sorting resources in rural cold chain warehouses, *International Journal of Innovative Computing, Information and Control*, vol.19, no.4, pp.1221-1238, DOI: 10.24507/ijicic.19.04.1221, 2023.

Author Biography

Ilvico Sonata S.T., M.M., MT., MBA, has experience in the Satellite Telecommunications industry for more than 27 years. He has a passion for teaching and is currently a lecturer at Doctor of Computer Science Program, BINUS University. He has scientific expertise in the fields of artificial intelligence, electrical engineering, autonomous systems and computer vision. He also teaches in Bachelor's and Master's programs. Some of the courses taught include Advanced Softcomputing and Internet of Things. He has professional certification in the fields of Satellite Telecommunications and Data Science. He is also active in writing several conference papers and Scopus indexed journals.

Yulyani Arifin S.Kom, M.M., currently serves as Deputy Head of Doctor of Computer Science, BINUS University. She completed her Doctoral Education in 2021 at Doctoral of Computer Science, BINUS University and studied Master of Management specializing in Information Systems in 2005 and Bachelor of Information Management in 1998. In 2022, she completed his Professional Engineer degree. Before focusing on academics, she had 11 years of professional experience in the ERP field. She is also active in the Indonesian Chapter of the ACM-SIGCHI organization and is a member of the international organization ISACA. Since 1998, she has been actively teaching until now. The knowledge groups taught and research topics include human-computer interaction, natural language processing, text processing, multimedia and game virtual reality, augmented reality and mixed reality, IT risk management and software engineering.