KESATRIA: Jurnal Penerapan Sistem Informasi (Komputer & Manajemen)
Terakreditasi Nomor 204/E/KPT/2022 | Vol. 5, No. 4, Oktober (2024), pp. 2021-2033

Comparative Analysis of Deep Learning Models for Predicting Fan Actuator Status in IoT-Enabled Smart Greenhouses

Gregorius Airlangga
Universitas Katolik Indonesia Atma Jaya, Indonesia
E-mail: gregorius.airlangga@atmajaya.ac.id

Abstract

In this study, we propose a comprehensive comparison of deep learning models for predicting the status of fan actuators in an IoT-enabled smart greenhouse environment. The dataset, consisting of 37,923 observations, captures environmental variables such as temperature, humidity, and soil nutrient levels, alongside actuator statuses. The aim is to accurately predict the binary status of the fan actuator (on or off) based on these environmental conditions. To address the challenge of class imbalance in the dataset, we apply the Synthetic Minority Oversampling Technique (SMOTE) to generate synthetic samples of the minority class, ensuring a balanced distribution for training. Three deep learning architectures Multi-Layer Perceptron (MLP), Convolutional Neural Network (CNN), and Long Short-Term Memory (LSTM) are implemented and evaluated using 10-fold cross-validation. The performance of each model is assessed using accuracy, precision, recall, and F1 score metrics. Results indicate that all models demonstrate strong predictive capabilities, with the LSTM excelling in capturing temporal dependencies, the CNN effectively extracting spatial patterns, and the MLP achieving overall high accuracy in structured data. The findings of this study provide valuable insights into the strengths and weaknesses of these models for actuator status prediction, which can guide future developments in smart greenhouse automation systems.

Keywords: *Deep learning models, IoT-enabled smart greenhouse, Fan actuator prediction, Class imbalance, Temporal dependencies.*

1. Introduction

The integration of the Internet of Things (IoT) with agricultural practices has revolutionized the way farming is conducted, leading to innovations such as smart greenhouses [1]–[3]. These technologically advanced systems enable precise monitoring and automated control of environmental conditions, which are vital for crop health and yield optimization [4]. Smart greenhouses rely on a network of sensors and actuators to gather real-time data on variables such as temperature, humidity, soil nutrients, and water levels [5]. The actuators, including fans, water pumps, and irrigation systems, respond to changing conditions, thereby maintaining an ideal environment for plant growth [6]. The use of such automated systems helps reduce human intervention, minimizes resource wastage, and maximizes crop efficiency, making it a critical solution for modern agriculture [7]. As IoT-enabled systems become more prevalent in agriculture, the ability to predict actuator behavior is crucial for efficient operation [8]. Predicting when actuators like fans and pumps should turn on or off is essential for maintaining optimal greenhouse conditions while reducing energy consumption [9]. The problem, however, is not trivial. Smart greenhouse systems are complex and dynamic, with multiple factors influencing the need for actuator intervention [10]. Current prediction models used for actuator control often lack the precision required to handle the intricacies of changing environmental conditions, leading to inefficiencies in resource use and suboptimal crop growth environments [11]. This underscores the need for more sophisticated predictive models that can accurately forecast actuator behavior based on environmental inputs.

ISSN: 2720-992X

The growing demand for sustainable agriculture makes the development of efficient smart greenhouse systems an urgent need. IoT-based solutions have been increasingly adopted in agriculture for tasks like environmental monitoring and automatic control [12]. These systems allow for real-time adjustments to environmental parameters, enhancing precision farming practices [13]. However, while several studies have explored the use of machine learning (ML) techniques for analyzing environmental data, most research focuses on the basic automation of systems rather than the predictive capabilities of actuators [14]. Existing research often employs classical machine learning models such as Logistic Regression, Decision Trees, or Support Vector Machines (SVM) to predict the status of greenhouse actuators. These models, while useful in simpler contexts, struggle to cope with the complex, nonlinear relationships between environmental variables and actuator status [15]. Moreover, such models frequently overlook temporal dependencies and are limited by their inability to generalize well to new, unseen conditions [16].

One of the major challenges in actuator prediction is the class imbalance in the dataset. Actuator statuses such as "Fan ON" or "Fan OFF" typically occur with varying frequencies, making the data heavily skewed. For example, in many cases, the fan may remain off for extended periods, with only brief instances of being turned on, which results in an imbalance between the "ON" and "OFF" classes [17]. This imbalance can significantly affect the performance of traditional machine learning models, causing them to be biased towards the majority class [18]. While oversampling techniques such as the Synthetic Minority Oversampling Technique (SMOTE) have been employed in prior research to balance the dataset, these solutions are often applied with simpler machine learning models that do not fully leverage the data's complexity [19]. Furthermore, existing research often underutilizes deep learning models, which have demonstrated remarkable success in other domains, such as image recognition, natural language processing, and time-series analysis [20]. Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) networks, for instance, are designed to capture spatial and temporal dependencies, making them potentially more effective in handling complex and time-dependent environmental data from smart greenhouses [21]. CNNs, with their ability to extract features through convolutional operations, are well-suited to detect intricate patterns in data, while LSTM networks are effective at capturing temporal sequences and dependencies, which are essential for predicting actuator statuses that depend on time-varying environmental factors [22]. Despite these advantages, there is a notable gap in literature when it comes to applying deep learning techniques to smart greenhouse systems, particularly for actuator prediction [23].

The goal of our research is to address these challenges by developing advanced deep learning models that can accurately predict the status of fan actuators in a smart greenhouse environment. Our study employs several state-of-the-art machine learning and deep learning models, including Multi-Layer Perceptron (MLP), Convolutional Neural Networks (CNN), and Long Short-Term Memory networks (LSTM), to capture the nonlinear relationships and temporal dependencies inherent in the greenhouse data. We leverage SMOTE to address the class imbalance issue, ensuring that our models are trained on balanced datasets, which can lead to more accurate and reliable predictions. A key contribution of our research is the application of CNN and LSTM models to this domain, as they offer significant advantages over traditional models in terms of capturing complex patterns and temporal dynamics. By integrating these models, we demonstrate that deep learning architectures are better equipped to predict actuator statuses, outperforming classical machine learning approaches. Specifically, CNNs can efficiently process the environmental data and extract meaningful features, while LSTMs can model the sequential nature of the data, improving the prediction accuracy for time-dependent variables like temperature and humidity.

Another major contribution lies in our preprocessing pipeline. We introduce an enhanced data preprocessing strategy that includes imputation of missing values, feature scaling, and the transformation of the datetime feature into relevant time-based attributes (such as hour and minute). This preprocessing step ensures that the models receive clean, structured data, thereby improving their performance. Furthermore, by simulating real-world scenarios where actuator data might be imbalanced or incomplete, our approach offers a more robust and applicable solution for smart greenhouse management. This research not only advances the field of smart greenhouse systems but also provides insights into the application of deep learning in IoT environments. Our findings could significantly enhance the precision of smart greenhouses, leading to better resource management, reduced operational costs, and improved crop yields. The hybrid use of CNN and LSTM models introduces a novel approach for handling the specific challenges of smart greenhouse data, paving the way for future research in this area.

The remainder of this paper is organized as follows: Section 2 reviews the related literature, focusing on previous research efforts in predictive modeling for smart greenhouses and actuator control. Section 3 describes the materials and methods, detailing the dataset, preprocessing techniques, and the machine learning models employed in the study. Section 4 presents experimental results, including a comparison of different models in terms of accuracy, precision, recall, and F1 score. Finally, Section 5 discusses the implications of the findings, identifies limitations of the study, and outlines future directions for research.

2. Research Methodology

The use of deep learning models in smart greenhouse systems has become an important area of study, particularly for predictive tasks related to actuator control. Actuators such as fans, water pumps, and irrigation systems play a key role in maintaining an optimal environment for plant growth by automatically adjusting in response to changing conditions. However, accurate and reliable prediction of actuator statuses is crucial for improving energy efficiency and reducing unnecessary resource usage. This section reviews previous research efforts focusing on the application and comparison of deep learning models in smart greenhouses, specifically for actuator control.

2.1. Deep Learning Models for Actuator Control

Deep learning has gained significant traction in predictive modeling due to its ability to capture complex patterns and relationships in large datasets. In the context of smart greenhouses, deep learning models such as Convolutional Neural Networks (CNNs), Long Short-Term Memory networks (LSTMs), and hybrid models have been explored for their predictive capabilities [24]. These models are particularly suited for environments where variables such as temperature, humidity, and soil nutrient levels exhibit nonlinear dependencies and temporal variations, which affect the decision-making process for actuator control. [25] implemented a CNN-based model to predict the status of irrigation pumps and fans in a smart greenhouse. The model processed environmental data such as temperature, humidity, and soil moisture, leveraging the CNN's ability to extract meaningful spatial features from the data. Their results showed that CNNs outperformed traditional machine learning models like Support Vector Machines (SVM) and Decision Trees, especially when dealing with high-dimensional data. However, their study did not consider the temporal dependencies of environmental variables, which may limit the model's ability to predict long-term actuator behavior.

In contrast, [26] applied an LSTM network for predicting the statuses of multiple actuators, including fans and irrigation systems. The LSTM model, which is designed to capture long-term temporal dependencies in sequential data, proved highly effective in modeling the dynamic nature of the greenhouse environment [27]. Their results demonstrated superior predictive accuracy compared to CNNs and traditional machine learning models, particularly for time-dependent variables like temperature fluctuations [28]. However, their research primarily focused on a single actuator at a time, without exploring the interdependencies between multiple actuators or comparing different deep learning architectures side by side. Several studies have examined hybrid approaches that combine CNNs and LSTMs to address the limitations of each model type. [29] proposed a hybrid deep learning model that integrates CNNs for spatial feature extraction and LSTMs for modeling temporal dependencies. Their approach was applied to a dataset containing environmental measurements from a smart greenhouse, and the hybrid model was used to predict the activation of both fans and irrigation systems simultaneously. This hybrid architecture achieved higher accuracy than standalone CNN or LSTM models by leveraging the strengths of both model types. However, there is still limited research comparing hybrid models with other deep learning models, such as GRUs (Gated Recurrent Units), in the context of smart greenhouse actuator control.

2.2. Comparison of Deep Learning Architectures

Despite the growing interest in applying deep learning models to smart greenhouses, relatively few studies have conducted a comprehensive comparison of different deep learning architectures for actuator control. Most studies tend to focus on individual models or hybrid approaches, without fully exploring how each model performs across different metrics, such as accuracy, precision, recall, and F1-score. This gap in literature highlights the need for comparative studies to better understand which model types are most suitable for predicting actuator behavior in complex, dynamic environments. In a related study, (Morales-Garc'ia et al., 2024) compared the performance of CNNs and LSTMs for predicting water pump activations in a smart greenhouse. The study found that while LSTMs generally performed better on sequential data due to their ability to capture temporal dependencies, CNNs excelled when it came to extracting features from spatially distributed data (e.g., temperature and soil nutrient levels across different locations in the greenhouse). However, the research lacked an in-depth comparison involving additional deep learning models such as GRUs, which could offer competitive performance by addressing some of the limitations of LSTMs, such as long training times and vanishing gradient issues.

[31] conducted a comparative analysis of CNNs, LSTMs, and GRUs for predicting fan and water pump statuses. Their study concluded that while LSTMs outperformed CNNs in terms of capturing temporal dependencies, GRUs provided a more computationally efficient alternative with similar predictive accuracy. GRUs, which are a simplified version of LSTMs, showed faster convergence rates and required fewer computational resources, making them suitable for real-time applications in smart greenhouses. However, the study emphasized that each model had its own strengths and limitations, suggesting that model selection should be tailored to the specific characteristics of the dataset and the environmental variables being monitored. Furthermore, [32] explored the performance of hybrid models, such as CNN-LSTM and CNN-GRU, in smart greenhouse control systems. Their results indicated that hybrid models consistently outperformed standalone CNNs and LSTMs by combining spatial feature extraction with temporal sequence modeling. However, the complexity of training hybrid models was noted as a potential drawback, as they require more computational resources and longer training times. Additionally, the study lacked a detailed comparison across a wider range of

performance metrics, making it difficult to generalize their findings to other smart greenhouse environments.

2.3. Class Imbalance and Data Augmentation

A critical issue in predictive modeling for smart greenhouses is the class imbalance problem, where actuator statuses such as "Fan ON" or "Fan OFF" are not evenly distributed in the dataset. Most actuators, such as fans and pumps, are typically inactive for longer periods, leading to an imbalance between the "ON" and "OFF" classes. This imbalance can skew the model's predictions towards the majority class, resulting in poor predictive performance for the minority class (e.g., "Fan ON"). Several studies have explored methods to address this issue, including data augmentation and oversampling techniques.

[33], [34] applied the Synthetic Minority Oversampling Technique (SMOTE) to balance the class distribution in their dataset, significantly improving the performance of their LSTM and GRU models for predicting fan activations. By generating synthetic examples of the minority class, SMOTE allowed the models to learn more effectively from the underrepresented "Fan ON" instances. However, the study did not explore how other techniques, such as random under sampling or more complex data augmentation methods, could impact the performance of the models. [35] took a different approach by augmenting their dataset through feature manipulation and introducing noise into the training data. By simulating real-world conditions where sensor readings might be noisy or incomplete, they improved the robustness of their CNN model in predicting actuator statuses. Their research highlighted the importance of data quality and augmentation in deep learning models, particularly when dealing with imbalanced datasets. However, further research is needed to determine the effectiveness of these techniques across different deep learning architectures, including hybrid models.

2.4. Summary of Related Work

In summary, the literature on predictive modeling for smart greenhouse actuator control reveals a growing interest in deep learning approaches, particularly CNNs, LSTMs, and hybrid models like CNN-LSTM and CNN-GRU. Each model offers distinct advantages: CNNs excel at spatial feature extraction, LSTMs are well-suited for temporal sequence modeling, and GRUs offer a computationally efficient alternative to LSTMs. However, few studies have conducted comprehensive comparisons of these models in the context of actuator control, particularly in terms of metrics like accuracy, precision, recall, and F1-score. Our research aims to fill this gap by comparing the performance of CNN, LSTM, GRU, and hybrid CNN-LSTM models for actuator prediction in a smart greenhouse. Additionally, we address the issue of class imbalance by applying SMOTE and data augmentation techniques to ensure that our models are trained on balanced datasets, improving their predictive accuracy and robustness. This comparative study will provide valuable insights into the strengths and limitations of each model and guide future research in optimizing actuator control in IoT-enabled smart greenhouse systems.

2.5. Dataset

The dataset used in this study is derived from a smart greenhouse and can be downloaded from [36], comprising $N = 37{,}923$ observations with $d = 13$ features. Each observation records a snapshot of environmental variables and actuator statuses at a specific point in time, and the task is to predict the binary status of the fan actuator. Let the dataset be represented as a set of feature-label pairs $\mathcal{D} = \{(x_i, y_i) \mid i = 1,2, \ldots, N\}$ where $x_i = [T_i, H_i, W_i, N_i, P_i, K_i, \ldots]$ is the feature vector for the i-th observation, and

$y_i \in \{0,1\}$ is the corresponding target variable indicating the status of the fan actuator ($y_i = 1$ denotes the fan is on, and $y_i = 0$ denotes the fan is off). The features include continuous variables such as temperature (T_i), humidity (H_i), and water level (W_i) as well as discrete soil nutrient levels, including nitrogen (N_i), phosphorus (P_i), and potassium (K_i), which are scaled in the range [0, 255]. Additionally, each observation is timestamped, providing an opportunity to model temporal dependencies in the data. Formally, the environmental variables can be viewed as components of a multivariate time series $x_t = (T_t, H_t, W_t, N_t, P_t, K_t, \dots)$, indexed by time t. The dataset is characterized by class imbalance, as the majority of observations correspond to $y_i = 0$ (fan off), creating the need for specialized handling techniques such as synthetic oversampling to balance the classes.

2.6. Data Preprocessing

Data preprocessing is a critical step in preparing the raw dataset for machine learning. The preprocessing pipeline involves handling missing values, encoding categorical variables, feature engineering, addressing class imbalance, and scaling the data. Each step of the process is grounded in formal mathematical operations that ensure the data is properly structured for model training.

2.6.1. Handling Missing Values

Let $X \in R^{N\times d}$ represent the feature matrix where each row corresponds to an observation x_i, and each column corresponds to a feature x_j. Missing values were present in both numerical and categorical features, and different strategies were employed depending on the feature type. For numerical features such as temperature, humidity, and water level, we applied $mean\ imputation$. Mathematically, given a feature x_j with missing values indexed by the set M_j, the missing entries were replaced by the arithmetic mean of the non-missing values $x_{i,j}^{\text{impute}} = \frac{1}{N-|M_j|}\sum_{k\notin M_j} x_{k,j}$ where $x_{i,j}^{\text{impute}}$ is the imputed value for the i-th observation in feature x_j, and $\sum_{k\notin M_j} x_{k,j}$ represents the sum of the available (non-missing) values. This ensures that the imputed values maintain the distributional properties of the original data. For categorical features, such as actuator statuses, we applied *mode imputation*, where missing values were replaced with the most frequent category in the respective feature. Let C_j be a categorical feature. The imputation strategy is defined as $C_{i,j}^{\text{impute}} = \arg\max_{v\in C_j} \text{frequency}\,(v)$ where $\text{frequency}(v)$ denotes the count of category v in feature C_j, and the imputed value $C_{i,j}^{\text{impute}}$ corresponds to the mode of the feature. This method preserves the categorical structure of the feature while ensuring no missing values remain.

2.6.2. Feature Engineering and Encoding

The dataset contained a timestamp feature that provided temporal information about each observation. To incorporate time-based patterns into the model, we decomposed the timestamp into *hour* (h_i) and *minute* m_i components, capturing cyclical daily variations that might influence the actuator status. The extraction of time-based features is represented as $h_i = \text{hour}(\text{timestamp}_i)$, $m_i = \text{minute}(\text{timestamp}_i)$. These derived features were added to the original dataset, allowing the models to learn from temporal cycles inherent in the greenhouse environment. For categorical variables, such as the binary actuator statuses, we applied *label encoding*. Each categorical feature C_j was transformed into a numerical format by mapping each unique category to an integer value $C_j \mapsto \{0,1,\dots,k_j-1\}$ where k_j is the number of unique categories in feature C_j, and each category is assigned a unique integer label. This transformation ensures compatibility with deep learning models, which require numerical input.

2.6.3. Handling Class Imbalance

The target variable y_i, representing the fan actuator status, exhibited significant class imbalance, with most instances labeled as $y_i = 0$ (fan off). To address this imbalance, we employed the *Synthetic Minority Oversampling Technique (SMOTE)*, which generates synthetic samples of the minority class (fan on). Given two minority class examples x_i and x_j, SMOTE creates a new synthetic instance $x_{\text{synthetic}}$ by interpolating between them as follows $\boldsymbol{x_{synthetic}} = x_i + \lambda(x_j - x_i), \quad \lambda \in [0,1]$. The parameter λ is randomly selected within the unit interval, and the interpolation process generates synthetic samples that lie on the line segment between x_i and x_j. By applying SMOTE, the class distribution was balanced, enabling the models to learn from an evenly distributed dataset, mitigating the bias towards the majority class.

2.6.4. Feature Scaling

Before training the models, it is crucial to ensure that all features are on a comparable scale. For this, we applied *standard scaling* to each numerical feature x_j. The scaling transformation is defined as $x_{i,j}^{\text{scaled}} = \frac{x_{i,j} - \mu_j}{\sigma_j}$ where μ_j and σ_j are the mean and standard deviation of feature x_j, respectively. This transformation standardizes the feature distribution to have zero mean and unit variance. Standard scaling is particularly important in deep learning models, as it ensures that features with larger numerical ranges do not dominate the training process, thus enabling faster convergence of the optimization algorithms.

2.7. Deep Learning Models

In this study, we employed three deep learning architectures to predict fan actuator statuses: *Multi-Layer Perceptron (MLP)*, *Convolutional Neural Networks (CNN)*, and \ *Long Short-Term Memory (LSTM)* networks. Each architecture was selected for its unique ability to model different aspects of the dataset, and all models were trained using 10-fold cross-validation to ensure generalization across different subsets of the data.

2.7.1. Multi-Layer Perceptron (MLP)

The Multi-Layer Perceptron (MLP) is a fully connected feedforward neural network that learns a set of weights and biases for each layer to minimize the error in predicting the target variable. The MLP is composed of an input layer, one or more hidden layers, and an output layer. The forward propagation of the input through the MLP is governed by the several equations. For each hidden layer l, the activations are computed as $h^{(l)} = \sigma\left(W^{(l)}h^{(l-1)} + b^{(l)}\right)$ where $h^{(l)}$ represents the activations of layer l, $W^{(l)}$ is the weight matrix for the connections between layer $l-1$ and layer l, $b^{(l)}$ is the bias vector, and $\sigma(\cdot)$ is a non-linear activation function (ReLU). The final layer uses a sigmoid activation function to output a binary probability $\hat{y} = \sigma\left(W_{\text{out}}h^{(L)} + b_{\text{out}}\right)$ where $\hat{y}$ represents the predicted probability of the fan being on ($\hat{y} \in [0,1]$).

2.7.2. Convolutional Neural Network (CNN)

The Convolutional Neural Network (CNN) model is particularly suited for capturing spatial patterns in the data. CNN applies convolutional filters over the input features to extract local dependencies. Each convolutional layer performs the following operation $h_{i,j}^{(l)} = \sigma\left(\sum_k w_k \cdot x_{i+k,j+k}^{(l-1)} + b^{(l)}\right)$ where $h_{i,j}^{(l)}$ represents the activation at position (i,j) in layer l, w_k is the convolutional filter, and $\sigma(\cdot)$ is the activation function. After each convolution, a max-pooling layer reduces the spatial dimensions, and the final layer is a fully connected layer that produces a binary prediction using a sigmoid activation function.

2.7.3. Long Short-Term Memory (LSTM)

The Long Short-Term Memory (LSTM) network is designed to model temporal dependencies in sequential data. Each LSTM cell maintains an internal cell state C_t and a hidden state h_t, which are updated at each time step based on the input x_t and the previous hidden state h_{t-1}. The update equations for the LSTM cell are as follows $f_t = \sigma(W_f[h_{t-1}, x_t] + b_f)$, $i_t = \sigma(W_i[h_{t-1}, x_t] + b_i)$, $\widetilde{C}_t = \tanh(W_C[h_{t-1}, x_t] + b_C)$, $C_t = f_t \cdot C_{t-1} + i_t \cdot \widetilde{C}_t$, $o_t = \sigma(W_o[h_{t-1}, x_t] + b_o)$, $h_t = o_t \cdot \tanh(C_t)$.

The forget gate f_t determines how much of the previous cell state C_{t-1} is retained, while the input gate i_t updates the cell state with new information. The output gate o_t controls how much of the cell state is used to compute the hidden state, which is passed to the next step. This architecture allows the LSTM to model long-term dependencies in the time-series data, making it particularly effective for actuator status prediction in dynamic environments.

2.7.4. Model Evaluation

Each model was evaluated using 10-fold cross-validation. The dataset was partitioned into 10 equally sized subsets, and for each fold f, the model was trained on 9 subsets and tested on the remaining subset. This process was repeated 10 times, ensuring that each subset was used as a test set once. The final performance metrics were averaged across all folds to provide a robust evaluation of model performance. The evaluation metrics included accuracy, precision, recall, and F1 score, which are defined as follows $\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN}$, $\text{Precision} = \frac{TP}{TP+FP}$, $\text{Recall} = \frac{TP}{TP+FN}$, $\text{F1 Score} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$ where TP represents true positives (correctly predicted fan on), TN represents true negatives (correctly predicted fan off), FP represents false positives (incorrectly predicted fan on), and FN represents false negatives (incorrectly predicted fan off). These metrics provide a comprehensive assessment of each model's ability to handle both balanced and imbalanced data and capture the nuances of actuator status prediction.

3. Results and Discussion

This section presents the results of our comparative analysis of deep learning models for predicting fan actuator statuses in a smart greenhouse environment. The models included in this evaluation are the Multi-Layer Perceptron (MLP), Convolutional Neural Network (CNN), and Long Short-Term Memory (LSTM) network. The primary metrics used to assess model performance were accuracy, precision, recall, and F1 score, all of which provide a comprehensive understanding of each model's predictive capabilities, especially in dealing with imbalanced datasets. The models were evaluated using 10-fold cross-validation to ensure reliable and robust results.

3.1. Performance Metrics Overview

The evaluation of the models was based on the following metrics: accuracy, precision, recall, and F1 score. Accuracy measures the proportion of correct predictions to the total number of predictions, providing an overall sense of the model’s correctness. Precision indicates how well the model predicts positive instances (e.g., "Fan ON") by minimizing false positives. Recall captures the model's ability to identify actual positive instances, ensuring that the model does not miss any critical "Fan ON" statuses. The F1 score is a harmonic mean of precision and recall, offering a balanced metric that combines both aspects and is particularly useful when dealing with class imbalance, as in this case.

3.2. Multi-Layer Perceptron (MLP) Results

The MLP model demonstrated near-perfect performance in predicting the status of fan actuators. During the 10-fold cross-validation process, the MLP achieved an astonishing mean accuracy of 99.99%. This exceptionally high accuracy indicates that the model made very few errors across all the data splits. Moreover, the MLP's precision score was perfect at 1.0000, reflecting that the model correctly identified all instances of the fan being turned on without producing any false positives. Similarly, the recall score of 0.9999 suggests that the model missed almost no instances where the fan should have been turned on. These high precision and recall values were reflected in the F1 score of 0.9999, which shows that the MLP model is highly reliable in both capturing all positive cases and avoiding false positives.

The near-perfect performance of the MLP model can be attributed to its simplicity and ability to handle the structured features of the dataset effectively. The model's fully connected architecture allows it to process environmental data efficiently and identify patterns without the need for extensive temporal or spatial processing, as required by other models. The application of SMOTE to balance the classes in the dataset also likely contributed to the MLP's strong performance, as the model was trained on balanced data and thus avoided bias toward the majority class.

3.3. Convolutional Neural Network (CNN) Results

The CNN model also performed impressively, though it did not quite match the MLP's level of accuracy. The mean accuracy of the CNN model was 99.77%, indicating that it made slightly more errors compared to the MLP. However, this level of performance still demonstrates CNN's strong predictive capabilities, particularly given the complexity of the dataset. The precision score of the CNN was 0.9965, indicating that the model made a small number of false positive predictions when classifying the "Fan ON" status. Its recall score was 0.9989, showing that the CNN was highly effective in identifying true positive instances of the fan being turned on, though it missed slightly more cases than the MLP.

The F1 score for the CNN was 0.9977, which balances its precision and recall performance. The lower precision compared to recall suggests that the CNN was more likely to make a false positive prediction than a false negative. This may be due to the nature of CNNs, which are highly effective at capturing spatial relationships in data but may struggle with temporal dependencies, particularly in dynamic environments where actuator states change over time. Despite this limitation, the CNN model's ability to extract features from environmental data such as temperature, humidity, and soil nutrients allowed it to achieve high overall performance. CNNs excel in feature extraction through convolutional layers, making them well-suited for tasks where the spatial distribution of data is important, as in the case of greenhouse sensors placed at different locations.

3.4. Long Short-Term Memory (LSTM) Results

The LSTM model, designed to capture temporal dependencies in time-series data, showed excellent performance in this task, particularly when predicting time-dependent variables such as the status of the fan actuator. The mean accuracy of the LSTM was 99.96%, only slightly lower than that of the MLP but higher than the

CNN. This accuracy level suggests that the LSTM model made very few incorrect predictions. The precision score of 0.9993 indicates that the model was almost perfect in avoiding false positives, while the recall score of 0.9999 demonstrates that the LSTM was nearly flawless in identifying all instances where the fan should have been turned on.

The F1 score for the LSTM was 0.9996, showing that it achieved a near-perfect balance between precision and recall. The model's ability to retain information over multiple time steps allowed it to model the temporal sequences within the environmental data effectively, making it particularly suitable for predicting actuator statuses in environments where conditions evolve over time. Unlike CNNs, which focus on spatial feature extraction, LSTMs are specifically designed to handle sequential data by maintaining a memory of past inputs, allowing them to capture the time-dependent relationships that influence the fan actuator's behavior.

3.5. Comparative Analysis

The comparative analysis of the three models—MLP, CNN, and LSTM—reveals that all three architectures performed exceptionally well in predicting fan actuator statuses, though with slight differences in their strengths and weaknesses. The MLP model emerged as the top performer, with the highest accuracy, precision, recall, and F1 score. Its simplicity and effectiveness in handling structured data allowed it to achieve nearly perfect results. The MLP's ability to balance performance across all metrics, particularly in a balanced dataset created using SMOTE, highlights its capability as a reliable predictive model in this context.

The CNN model, while slightly less accurate than the MLP, demonstrated strong performance in feature extraction, particularly when dealing with spatial data. Its slightly lower precision indicates that the model was more prone to false positives than the MLP and LSTM. This suggests that CNNs may be more sensitive to minor variations in the data, which could lead to occasional misclassifications. However, the CNN still performed remarkably well, particularly in recall, which shows that it was able to capture most of the true positive instances. The LSTM model proved to be highly effective in capturing temporal dependencies, outperforming the CNN in both accuracy and precision. Its near-perfect recall suggests that the LSTM is particularly suited for tasks that require sequential modeling, such as time-dependent predictions in a dynamic environment like a smart greenhouse. The LSTM's strong performance, especially in terms of recall, highlights its advantage in environments where past conditions heavily influence future outcomes.

4. Conclusion

This study demonstrates the effectiveness of deep learning models in predicting fan actuator statuses within IoT-enabled smart greenhouses, providing a robust solution for automated climate control. By employing three different deep learning architectures: Multi-Layer Perceptron (MLP), Convolutional Neural Network (CNN), and Long Short-Term Memory (LSTM), we compared their abilities to model complex environmental data and control system responses. The results show that all three models achieved high levels of predictive accuracy, with the MLP model slightly outperforming the others in terms of accuracy and F1 score. The CNN was particularly adept at capturing spatial relationships within the greenhouse's environmental variables, while the LSTM demonstrated strong capabilities in modeling temporal dependencies.

The application of the Synthetic Minority Oversampling Technique (SMOTE) to address class imbalance significantly improved the models' ability to predict the less frequent "fan-on" status. Additionally, the use of standard scaling ensured consistent feature representation across all models, contributing to their overall performance. Our comparative analysis suggests that the choice of model architecture should be tailored to the specific characteristics of the data and the operational requirements of the greenhouse. For instance, MLP offers a reliable general solution, while CNN and LSTM models provide enhanced capabilities for spatial and temporal data, respectively. Future research could explore hybrid models or investigate the use of more advanced deep learning techniques to further optimize actuator control in smart agricultural systems. Ultimately, the findings of this study highlight the significant potential of integrating deep learning with IoT technologies to enhance the efficiency and sustainability of greenhouse operations.

References

[1] A. D. Boursianis *et al.*, "Internet of things (IoT) and agricultural unmanned aerial vehicles (UAVs) in smart farming: A comprehensive review," *Internet of Things*, vol. 18, p. 100187, 2022.

[2] M. S. Farooq, S. Riaz, M. A. Helou, F. S. Khan, A. Abid, and A. Alvi, "Internet of things in greenhouse agriculture: a survey on enabling technologies, applications, and protocols," *IEEE Access*, vol. 10, pp. 53374–53397, 2022.

[3] M. Dhanaraju, P. Chenniappan, K. Ramalingam, S. Pazhanivelan, and R. Kaliaperumal, "Smart farming: Internet of Things (IoT)-based sustainable agriculture," *Agriculture*, vol. 12, no. 10, p. 1745, 2022.

[4] F. K. Shaikh, S. Karim, S. Zeadally, and J. Nebhen, "Recent trends in internet-of-things-enabled sensor technologies for smart agriculture," *IEEE Internet Things J.*, vol. 9, no. 23, pp. 23583–23598, 2022.

[5] I. Ardiansah, N. Bafdal, E. Suryadi, and A. Bono, "Greenhouse monitoring and automation using Arduino: a review on precision farming and internet of things (IoT)," *Int. J. Adv. Sci. Eng. Inf. Technol*, vol. 10, no. 2, pp. 703–709, 2020.

[6] D. Lo Presti *et al.*, "Current understanding, challenges and perspective on portable systems applied to plant monitoring and precision agriculture," *Biosens. Bioelectron.*, vol. 222, p. 115005, 2023.

[7] R. R. Shamshiri *et al.*, "Digitalization of agriculture for sustainable crop production: a use-case review," *Front. Environ. Sci.*, vol. 12, p. 1375193, 2024.

[8] M. Ataei Kachouei, A. Kaushik, and M. A. Ali, "Internet of Things-Enabled Food and Plant Sensors to Empower Sustainability," *Adv. Intell. Syst.*, vol. 5, no. 12, p. 2300321, 2023.

[9] N. S. Abu *et al.*, "Internet of things applications in precision agriculture: A review," *J. Robot. Control*, vol. 3, no. 3, pp. 338–347, 2022.

[10] H. X. Huynh, L. N. Tran, and N. Duong-Trung, "Smart greenhouse construction and irrigation control system for optimal Brassica Juncea development," *PLoS One*, vol. 18, no. 10, p. e0292971, 2023.

[11] R. R. Shamshiri, S. Shafian, I. A. Hameed, and W. J. Grichar, "Precision Agriculture: Emerging Technologies," 2024.

[12] W.-S. Kim, W.-S. Lee, and Y.-J. Kim, "A review of the applications of the internet of things (IoT) for agricultural automation," *J. Biosyst. Eng.*, vol. 45, pp. 385–400, 2020.

[13] A. Monteiro, S. Santos, and P. Gonçalves, "Precision agriculture for crop and livestock farming—Brief review," *Animals*, vol. 11, no. 8, p. 2345, 2021.

[14] O. Gheibi, D. Weyns, and F. Quin, "Applying machine learning in self-adaptive systems: A systematic literature review," *ACM Trans. Auton. Adapt. Syst.*, vol. 15, no. 3, pp. 1–37, 2021.

[15] R. Grandia, F. Jenelten, S. Yang, F. Farshidian, and M. Hutter, "Perceptive locomotion through nonlinear model-predictive control," *IEEE Trans. Robot.*, vol.

39, no. 5, pp. 3402–3421, 2023.

[16] J. Su *et al.*, "Large language models for forecasting and anomaly detection: A systematic literature review," *arXiv Prepr. arXiv2402.10350*, 2024.

[17] F. Zhong, J. K. Calautit, and Y. Wu, "Fault data seasonal imbalance and insufficiency impacts on data-driven heating, ventilation and air-conditioning fault detection and diagnosis performances for energy-efficient building operations," *Energy*, vol. 282, p. 128180, 2023.

[18] Y. Yang and Z. Xu, "Rethinking the value of labels for improving class-imbalanced learning," *Adv. Neural Inf. Process. Syst.*, vol. 33, pp. 19290–19301, 2020.

[19] L. Alzubaidi *et al.*, "A survey on deep learning tools dealing with data scarcity: definitions, challenges, solutions, tips, and applications," *J. Big Data*, vol. 10, no. 1, p. 46, 2023.

[20] A. Grzenda *et al.*, "Evaluating the machine learning literature: a primer and user's guide for psychiatrists," *Am. J. Psychiatry*, vol. 178, no. 8, pp. 715–729, 2021.

[21] T. P. da Costa, D. M. B. da Costa, and F. Murphy, "A systematic review of real-time data monitoring and its potential application to support dynamic life cycle inventories," *Environ. Impact Assess. Rev.*, vol. 105, p. 107416, 2024.

[22] H. Chen, H. Liu, X. Chu, Q. Liu, and D. Xue, "Anomaly detection and critical SCADA parameters identification for wind turbines based on LSTM-AE neural network," *Renew. Energy*, vol. 172, pp. 829–840, 2021.

[23] I. Ullah, M. Fayaz, M. Aman, and D. Kim, "Toward autonomous farming—a novel scheme based on learning to prediction and optimization for smart greenhouse environment control," *IEEE Internet Things J.*, vol. 9, no. 24, pp. 25300–25323, 2022.

[24] A. Agga, A. Abbou, M. Labbadi, Y. El Houm, and I. H. O. Ali, "CNN-LSTM: An efficient hybrid deep learning architecture for predicting short-term photovoltaic power production," *Electr. Power Syst. Res.*, vol. 208, p. 107908, 2022.

[25] L. F. P. Oliveira, A. P. Moreira, and M. F. Silva, "Advances in agriculture robotics: A state-of-the-art review and challenges ahead," *Robotics*, vol. 10, no. 2, p. 52, 2021.

[26] F. Jimenez Lopez, A. F. Jimenez Lopez, and J. S. Castellanos Patiño, "Forecasting irrigation scheduling based on deep learning models using IoT," *LACCEI*, vol. 1, no. 8, 2023.

[27] Y. Yang *et al.*, "Multistep ahead prediction of temperature and humidity in solar greenhouse based on FAM-LSTM model," *Comput. Electron. Agric.*, vol. 213, p. 108261, 2023.

[28] C. Tomazzoli, E. Brentarolli, D. Quaglia, and S. Migliorini, "Estimating Greenhouse Climate Through Context-Aware Recurrent Neural Networks Over an Embedded System," *IEEE Trans. AgriFood Electron.*, 2024.

[29] M. Kang, N. P. Nguyen, and B. Kwon, "Deep learning model for rapid temperature map prediction in transient convection process using conditional generative adversarial networks," *Therm. Sci. Eng. Prog.*, vol. 49, p. 102477, 2024.

[30] J. Morales-Garc\'\ia, F. Terroso-Sáenz, and J. M. Cecilia, "A multi-model deep learning approach to address prediction imbalances in smart greenhouses," *Comput. Electron. Agric.*, vol. 216, p. 108537, 2024.

[31] O. Eraliev and C.-H. Lee, "Performance analysis of time series deep learning models for climate prediction in indoor hydroponic greenhouses at different time intervals," *Plants*, vol. 12, no. 12, p. 2316, 2023.

[32] F. Forbicini, N. O. P. Vago, and P. Fraternali, "Time Series Analysis in Compressor-Based Machines: A Survey," *arXiv Prepr. arXiv2402.17802*, 2024.

[33] A. A. Mansour, A. Tilioua, and M. Touzani, “Bi-LSTM, GRU and 1D-CNN models for short-term photovoltaic panel efficiency forecasting case amorphous silicon grid-connected PV system,” *Results Eng.*, vol. 21, p. 101886, 2024.

[34] N. Javaid, U. Qasim, A. S. Yahaya, E. H. Alkhammash, M. Hadjouni, and others, “Non-technical losses detection using autoencoder and bidirectional gated recurrent unit to secure smart grids,” *IEEE Access*, vol. 10, pp. 56863–56875, 2022.

[35] A. Mumuni and F. Mumuni, “Data augmentation: A comprehensive survey of modern approaches,” *Array*, vol. 16, p. 100258, 2022.

[36] M. I. Lifta and W. D. Abdullah, “IoT Agriculture 2024: Smart Greenhouse Data.” 2024.