processes

Article

An Intelligent Early Flood Forecasting and Prediction Leveraging Machine and Deep Learning Algorithms with Advanced Alert System

Israa M. Hayder [1], Taief Alaa Al-Amiedy [2], Wad Ghaban [3], Faisal Saeed [4], Maged Nasser [5], Ghazwan Abdulnabi Al-Ali [6] and Hussain A. Younis [5,7,*]

[1] Department of Computer Systems Techniques, Qurna Technique Institute, Southern Technical University, Basrah 61016, Iraq
[2] National Advanced IPv6 (Nav6) Centre, Universiti Sains Malaysia, Gelugor 11800, Penang, Malaysia
[3] Applied College, University of Tabuk, Tabuk 47512, Saudi Arabia
[4] DAAI Research Group, Department of Computing and Data Science, School of Computing and Digital Technology, Birmingham City University, Birmingham B4 7XG, UK
[5] School of Computer Sciences, Universiti Sains Malaysia, Gelugor 11800, Penang, Malaysia
[6] Department of Computer Science (Educational Science), University of Basrah, Basrah 61004, Iraq
[7] College of Education for Women, University of Basrah, Basrah 61004, Iraq
* Correspondence: hussain.younis@uobasrah.edu.iq

Abstract: Flood disasters are a natural occurrence around the world, resulting in numerous casualties. It is vital to develop an accurate flood forecasting and prediction model in order to curb damages and limit the number of victims. Water resource allocation, management, planning, flood warning and forecasting, and flood damage mitigation all benefit from rain forecasting. Prior to recent decades' worth of research, this domain demonstrated to be promising prospects in time series prediction tasks. Therefore, the main aim of this study is to build a forecasting model based on the exponential smoothing-long-short term memory (ES-LSTM) structure and recurrent neural networks (RNNs) for predicting hourly precipitation seasons; and classify the precipitation using an artificial neural network (ANN) model and decision tree (DT) algorithm. We employ the dataset from the Australian commonwealth office of meteorology named Historical Daily Weather dataset to test the effectiveness of the proposed model. The findings showed that the ES-LSTM and RNN had achieved 3.17 and 6.42 in terms of mean absolute percentage error (MAPE), respectively. Meanwhile, the ANN and DT models obtained a prediction accuracy rate of 96.65% and 84.0%, respectively. Finally, the outcomes revealed that ES-LSTM and ANN had achieved the best results compared to other models.

Keywords: flood forecasting and prediction; multilayer perceptron (MLP); time series analysis; es-lstm; machine learning (ML); deep learning (DL); artificial neural network (ANN); decision tree (DT); recurrent neural network (RNN); exponential smoothing

Citation: Hayder, I.M.; Al-Amiedy, T.A.; Ghaban, W.; Saeed, F.; Nasser, M.; Al-Ali, G.A.; Younis, H.A. An Intelligent Early Flood Forecasting and Prediction Leveraging Machine and Deep Learning Algorithms with Advanced Alert System. *Processes* **2023**, *11*, 481. https://doi.org/10.3390/pr11020481

Academic Editor: Xiong Luo

Received: 11 January 2023
Revised: 27 January 2023
Accepted: 3 February 2023
Published: 5 February 2023

1. Introduction

Natural disasters frequently result in major and long-lasting disturbances in the entire socioeconomic system. A single catastrophic event, such as a flood, can devastate complex infrastructure systems, resulting in recurring failures and significant socioeconomic harm, so impeding progress. Ample precipitation will immediately result in unharvestable crop, floods and waterlogging disasters, rendering crops, resulting in events that lead to disasters such as landslides, collapses, waterlogging, and mudslides [1].

Precipitation causes are extremely complicated because of the bundle impact of monsoon, geography, evaporation, urbanization, and temperature adding difficulties in forecasting precipitation [2–4]. In addition, rainfall has some set properties, and the elements affecting it, such as terrain, urbanization, and temperature, will not change significantly in short term. Furthermore, precipitation exhibits a high degree of predictability. Accurate flood forecasting can help in lowering the danger of flooding and give managers insightful and feasible environmental information [5].

In further detail, sensor nodes will conduct local processing in order to minimize communication and, hence, energy expenses. Various metrics are utilized to measure the degree of prediction accuracy in forecasting the flood. One of the most popular measures is the mean absolute percentage error (MAPE) measurement that is used in this study; the lower the MAPE value, the higher the accuracy for the system's predictions.

Further, flood prediction is a complicated technique that considers multiple variables affecting a river's water level, including its location, water flow, rainfall, soil types, humidity, and catchment area. The association between these variables is complicated. The association between these variables is complicated. Time series models based on classical linear gaussian distributions are insufficient for analyzing and forecasting complicated geophysical phenomena [6,7].

Linear methods such as linear regression (LR) and time series data mining (TSDM) approaches are unable to find complex features because of the goal of identifying all-time series observations, the requirement of time series stationarity, and the necessity of residual normality and independence [8]. Over brief prediction periods of up to one day, artificial neural networks (ANNs) [9], and nonlinear prediction (NLP) provide accurate predictions.

Traditional ML models treat each sample separately, with inputs and outputs, which is considered inefficient at forecasting since they require time series information and generate temporal dependency in the data. The output of these models is frequently degraded when the input data contains missing values or environmental noise. The predictions produced by most of regularly used models contain significant inaccuracies, particularly for highly nonstationary predictands [10,11].

Additionally, precipitation is always a type of time-series data that changes dramatically over time. Therefore, the most commonly used and advanced deep learning (DL) approaches to deal with time series problems is recurrent neural networks (RNNs) [12]. RNN-based rainfall prediction is more accurate compared to some other DL approaches [13]. Rainfall [14], as well as a dynamic weather forecast [15], can be utilized to estimate radiation levels using RNN since its coding is excellent for rainfall predictions with time series or time-based data [16].

As previously stated, both process-based models and conventional ML methods achieve desperate results in forecasting streamflow. Numerous studies [17–20] have established the superiority of DL models over standard ML algorithms in solving various problems of forecasting and prediction and in various domains. To address the above-mentioned shortcomings, the following contributions are proposed in this study:

- A forecasting modal-based window size technique that adopts a hybrid model of time-series analysis using exponential smoothing (ES) and long short-term memory model (LSTM) network along with recurrent neural network (RNN).
- A prediction model based on artificial neural network (ANN) and decision tree (DT).
- A design of decision and response module to analyze the results of proposed models, formulate the final decision as well as activation of IoTs' sensor nodes.
- A prototype of forecasting and prediction model for floods in city and rural areas.

The paper is outlined according to: Section 2 covers the related literature along with the identification of used parameters and limitations. Section 3 introduces the theoretical and mathematical models for used algorithms and methods. Section 4 illustrates the methodology of the proposed work in details and provide discussion for the roles of the used models. Section 5 elaborates the experimental setup, findings, and discussion on the outcomes of the used models. Finally, Section 6 presents the conclusion and highlights the insight for future research directions.

2. Literature Review

In this section, we highlight a review on the literature studies relevant to our work. D'Addabo et al. [21] applied the Bayesian network technique to flood detection. The authors obtained the flood detection model using an existing dataset rather than performing actual measurements in their work. The authors used light detection and ranging (LiDAR) technology to gather the information. A study was conducted by Wu and Wang [22] to investigate and implement network sensors to construct a portable flood detection system. In this work, the roads were monitored by sensors to send out alerts to drivers in case the flooding is detected. As for flood detection, Khalaf et al. [23] employed sensors and ML methods in their work. This study merely used a sensor network to assess the water level and sent an SMS alert in case flooding is detected. In addition, the authors chose ML algorithms randomly to show the applicability of using architecture time series in ML algorithms.

Compared to ANNs, which have a limited number of processing units and layers due to algorithm computational limits, DL-based models such as LSTM have been used and enhanced by employing graphical processing unit (GPU) parallel processing and powerful optimization methods [24]. LSTM can also capture time series and memorize long-term associations with the addition of a forget gate, making it helpful for sophisticated long-time-lag applications [17]. Kratzert et al. [25] offered an LSTM model to predict daily streamflow, whereas Hu et al. [26] proposed another LSTM model to predict hourly streamflow using six models. In the lower portions of the Semarang region, Widiasari et al. [27] employed an LSTM model to forecast river water levels. Moreover, a flood early detection system-based IoT was proposed by Mousavi et al. [28]. In this work, the authors employed various ML and DL algorithms for real-time flood monitoring and detection.

Zhang et al. [29] utilized LSTM to anticipate water tables in agricultural zones. Xiang et al. [30] provided a model for hourly streamflow prediction based on LSTM and seq2seq structure. Damavandi et al. [18] suggested a method for forecasting streamflow utilizing a digital elevation model (DTM) as supplemental information to historical observed data as another application of LSTM layers. Dong et al. [31] introduced a dynamic sliding window approach that can mimic distinct flow times in different months of the year and used LSTM to opt for the best window. An urban flooding forecasting and alerting model was proposed by Won et al. [32]. The proposed model involves ANN, LSTM, bi-LSTM, and Stack-LSTM. The outcomes showed the superiority of bi-LSTM over other models in high-water level predictions.

Furthermore, a water level forecasting model was developed by [33]. The authors utilized a variety of ML and DL models to predict floods in the Red River of the North. They analyzed hourly level data from three U.S. Geological Survey to evaluate the water level. The results show the superiority of LSTM over other approaches. Kunverji et al. [34] devised various ML algorithm such that DT, RF, and Gradient Boost for flood prediction. The results show high flood prediction with high detection accuracy. Moreover, a deep learning model was developed by [35] for rainfall forecasting model in variable climatic condition. Chen et al. [36] introduced an urban flooding prediction model using LSTM and Numerical model. The simulation results from the numerical model is used to construct the prediction model which yielded high prediction accuracy and fast detection for daily flood with fast response time. Table 1 displays the summary of related works.

Table 1. Summary of related works.

No.	References	Aim of Study	Model(s)	Dataset(s)	Performance Metric(s)	Limitation(s)
1	D'Addabo et al. [21]	Flooding monitoring using synthetic aperture radar and interferometric SAR data	Bayesian Network (BN)	Synthetic aperture radar, interferometric SAR (InSAR)	Sensitivity, specificity, probability	• It lacks evaluation using crucial other metrics, such as Accuracy, False Positive Rate, MAPE, etc. • It uses only image dataset for analysis. • No details about the prediction/forecasting about rainfall.
2	Wu and Wang [22]	A flood monitoring and detection system using portable heterogenous sensor networks	NA	Real Time Dataset	NA	• It deploys some wireless sensors for flood monitoring and detection. • Lack of analysis in terms of the collected data and sensors' power consumption. • Lack of using ML or DL techniques with the proposed system. • Unable to forecasting of rainfall.
3	Khalaf et al. [23]	A flood detection and alert generation using ML and sensor technology	Random Forest Bagging Decision Tree Hyper Pipes	Environment Agency Offices data search	True Positive, True Negative, Recall, F-Measure, ROC	• It uses ML algorithms for flood detection and luck of using DL algorithm. • It is unable to forecast rainfall. • It is uses GSM network for communication which require additional infrastructure.
4	Balogun and Adebisi [17]	A sea Level Prediction Using ML and DL algorithms	ARIMA, SVR, LSTM	Monthly Sea Level Altimetry (SLA)	Heat Map, R accuracy, Correlation pattern	• It does provide forecasting model for future rain occurrence. • It does not provide an intelligence network for rain/flood monitoring. • No Details about the deployment strategy.
5	Kratzert et al. [25]	A rainfall prediction model using LSTM networks	LSTM	CAMELS dataset	Correlation coefficient of the observed and simulated, Flow Duration Curve, Total Volume Bias, Discharge Variance Bias	• Lack of Analysis in term of MAPE, ACC, etc. • It uses only one model for rain prediction. • There is no information available on the deployment plan. • No involvement of rain sensors.
6	Hu et al. [26]	A LSTM model for streamflow Forecasting of small river	LSTM, SVR, MLP	Collected Dataset	RMSE, MAE, coefficient of determination	• There is no information available on the deployment plan. • The developed model evaluated with old dataset. • It produces high errors in top volume forecast.
7	Widiasari et al. [27]	A flood and water elevation prediction model using context -based hydrology LSTM	LSTM	Collected Dataset	MAPE, RMSE	• It uses only one algorithm (LSTM) which could results in inaccurate prediction results. • There is no information on the availability of the dataset. • There is no information available on the deployment plan. • It limited the study sample.
8	Mousavi et al. [28]	A flood early detection system based IoT using ML and DL techniques	ANN, LSTM, GRU	historical precipitation and streamflow data	Root Mean square error (RMSE), Normalized Statistic (NSE), Regression	• Don't show to reduce human, financial, and infrastructural damage. • The study lack analysis in term of Accuracy, False Positive Rate, Sensitivity, Specificity, Precision, and Negative Predictive Value. • Do not have rain sensors to observation.

Table 1. *Cont.*

No.	References	Aim of Study	Model(s)	Dataset(s)	Performance Metric(s)	Limitation(s)
9	Zhang et al. [29]	A prediction model for water table depth in agricultural areas using LSTM	LSTM	Collected Dataset	RMSE, Coefficient of Determination	• It relies on one model (LSTM) to predict the water depth. • It uses old dataset to asset the performance of the model. • There is no information available on the deployment plan. • The mode is incapable of forecasting future rainfall.
10	Xiang et al. [30]	A rainfall prediction using LSTM-seq2seq model	Lasso regression, Ridge regression, linear regression, Gaussian processes regression, support vector regression, LSTM-seq2seq	WY2016 for Coralville Station	Nash-Sutcliffe efficiency coefficient, the correlation coefficient, normalized root-mean-square error, statistical bias	• It proposed LSTM-based seq2seq with different topographies and different watersheds. • Its study compares with Statistical Measures only, leaving Measures of Accuracy, False Positive Rate, • Sensitivity, Specificity, Precision, and Negative Predictive Value. • The model only uses hourly rainfall/runoff observation, and rainfall forecast.
11	Damavandi et al. [18]	Streamflow prediction based on LSTM	LSTM	Livneh's database	Pearson correlation, Nash–Sutcliffe model efficiency	• The LSTM was evaluated with old dataset. • No analysis of other metrics, like, RMSE, and MAPE. • No details about the deployment strategy. • The prediction process relays on one model which make it sustainable for high error rate
12	Dong et al. [31]	A dynamic sliding window-based LSTM for predicting streamflow	a dynamic window-based LSTM	Zhutuo Hydrological Station	Flow Prediction, Error Rate	• It is evaluated with old dataset. • It is unable to forecasting future rainfall. • No information is given about the deployment strategy. • The prediction model relies on sole model which make it sustainable to high false prediction.
13	Kunverji et al. [34]	A flood prediction model based on ML algorithms	Decision Tree, Gradient Boost, Random Forest	India Water Portal	Accuracy, True Positives, False Positives (FP), True Negatives (TN), False Negatives (FN),	• It is evaluated with outdated dataset. • No details about the deployment strategy. • It is unable to forecast future rainfall.
14	Won et al. [32]	An urban flooding forecasting and alerting model using DL algorithms	ANN, LSTM, bi-LSTM and Stack-LSTM	Seoul pump stations	Water level prediction, RMSE, MAE, MAPE, R2	• Limited study solely on Dorim River Basin, Seoul • It uses only image datasets for analysis. • It deploys urban flood forecasting and warning process, as a non-structural measure to mitigate urban flood. • It uses DL algorithms for flood damage and excludes of using ML algorithms. • It uses DL models that lacks high performance for univariate hydrological time series data prediction.
15	Atashi et al. [33]	A ML and DL models for water level forecasting of Red River of the North	Integrated moving average (SARIMA), Random Forest, LSTM	U.S. Geological Survey (USGS)	RMSE, Water Level Prediction	• There is no information provided on the deployment strategy. • Lack of analysis of other critical metrics, like Accuracy, and False Positive Rate.

Table 1. *Cont.*

No.	References	Aim of Study	Model(s)	Dataset(s)	Performance Metric(s)	Limitation(s)
16	Fahad et al. [35]	A rainfall forecasting model in variable climatic condition based on deep learning technique	Gated Recurrent Unit (GRU)	Climate Change Knowledge Portal (CCKP) of the World Ban	Normalized Root Mean Squared Error (NRMSE), Normalized Mean Absolute Error (NMAE), Correlation	• The prediction model depends on GRU only which might lead to false prediction of rainfall. • It is unable to forecast future rainfall. • It suffers from volatile atmospheric behavior and climatic condition. • It did not provide information about the deployment strategy.
17	Chen et al. [36]	An urban flooding prediction model-based LSTM Network and Numerical model	LSTM, Numerical Model	Self-Generated Dataset	Ponding Depth Error, Depth of Water Accumulation	• No evaluation of other critical metrics like, RMSE, and MAE. • There is no information provided on the deployment strategy. • Lack of involvement of windows side technique with LSTM. • It produces noticeable error rate.

To sum up, based on our knowledge and according to the analysis of existing literature (see, Table 1), it could be concluded that there is no study conducted to merge ML and DL techniques for forecasting and predicting floods. Consequently, this study is proposed to fulfill the loopholes by merge ML (i.e., ANN and DT) and DL (i.e., RNN and ES-LSTM) techniques to build forecasting and prediction model for early detection of flood occurrence.

3. Algorithms and Methods

We give a brief overview of the theoretical and mathematical model of the applied methods and algorithms in this section, which is followed by an illustration of the evaluation metrics applied in this work.

3.1. Time Series (Slide Window)

Time series analysis is useful in extracting information related to the research about the number of rain forecasting over time. It can also be utilized to determine the expansion of the existing modifications noticed within the same time. In addition, analysis techniques may be used to determine if time series data are continuous or seasonal [37]. A reliable method for presenting data over a long period of time at various times is the time series approach (hourly, weekly, quarterly, or yearly). More observations lead to a smoother measurement trend, which improves predictions [32].

The time series approach has the potential to identify seasonal patterns, which is a significant advantage and a crucial component of forecasting the future. In addition, the sliding window is an essential technique for time series prediction using feedforward neural networks (FNNs). Accordingly, the FNNs require fixed-size input with no memory, which is considered the most common method to feed them with time series data. The basic concept of sliding window input is shown in Figure 1.

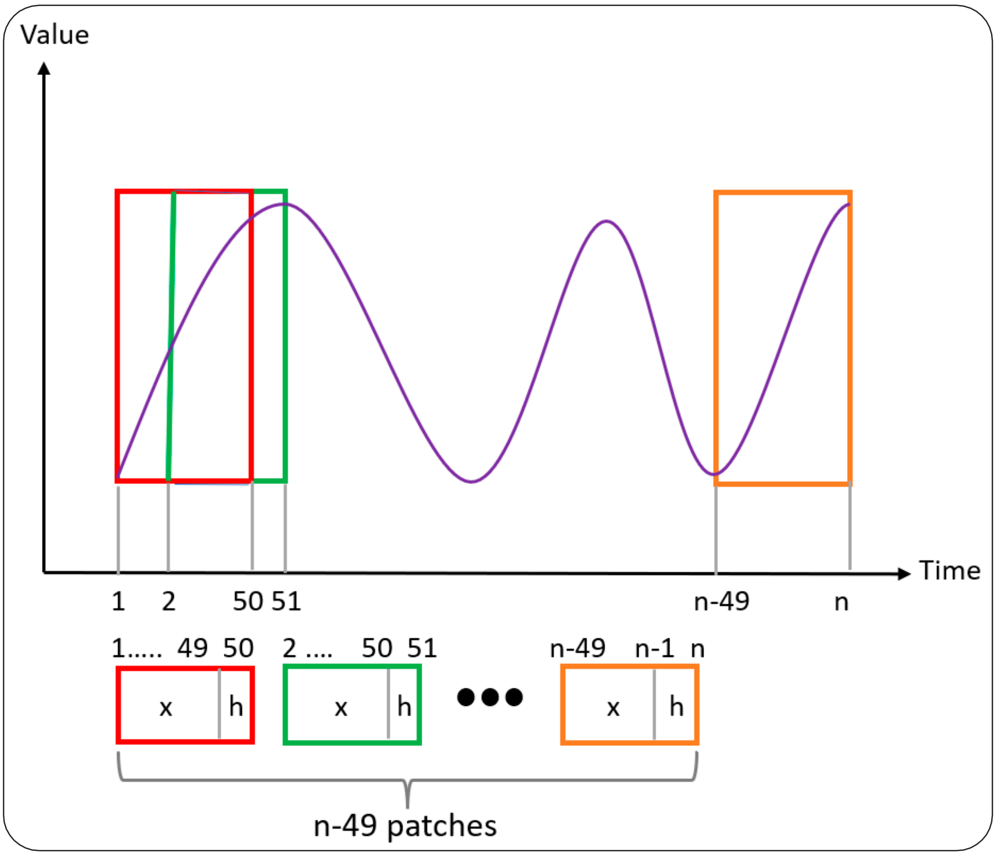

Figure 1. An illustration to the sliding window input concept.

3.2. Recurrent Neural Network (RNN)

The first RNN was developed in the 1980s [38]. The architecture of RNN consists of an input layer, one or more hidden layers, and an output layer. It has a series structure as repeating units with the idea of using these units to store important information from previous processing steps [39]. In addition, RNN can learn the sequence and solve the dependency problem in time series. Figure 2 shows the basic structure of an RNN.

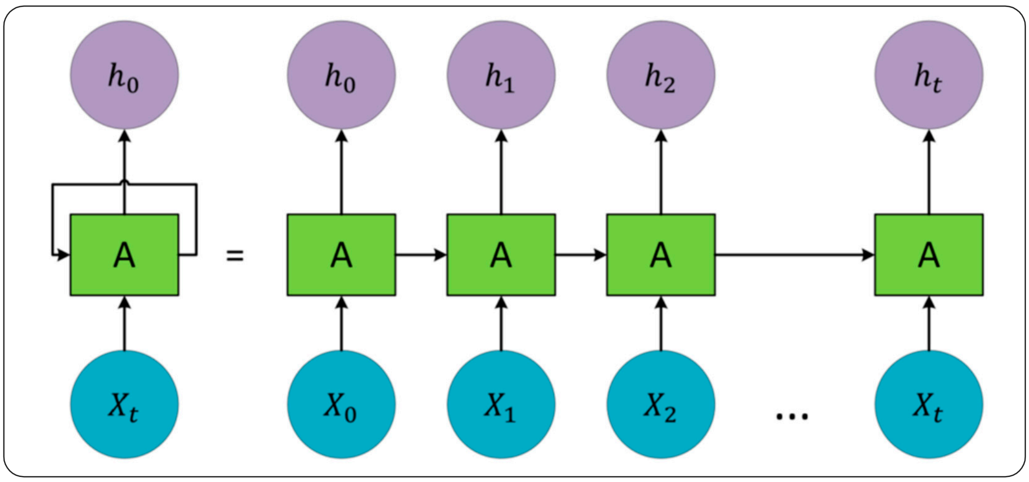

Figure 2. Sequential processing of recurrent neural network [38].

From the preceding Figure 2, it can be seen that the network is composed of an input unit, an output unit, and a recurrent hidden unit, where x^t represents the input at a given t time and h^t represents the output all at once. RNN uses the reaction propagation algorithm, which is frequently used in gradient computation and modification of the matrices that will be adjusted after adjusting the feedback process, throughout the training phase.

$$h^t = f\left(Ux^t, Wh^{(t-1)}\right) \tag{1}$$

$$o^t = \text{softmax}\left(Vh^t\right) \tag{2}$$

x^t: input vector, h^t: hidden layer vector, and o^t: output layer vector. U, W and V: parameter matrices and vector. h^t and o^t: activation functions.

Consequently, this algorithm is also typically known as back-propagation through time (BPTT) [40]. The BPTT approach works layer by layer, from the beginning of the network, with weights set based on the calculated part of the error units. Consequently, if the training data are longer, then the network will waste resources, which will slow down the update process of networks' weights. Therefore, RNN has a gradient problem and cannot consider the effect of long-term memory in practical applications. Moreover, a plug-in is required to store this memory. Thus, a special RNN known as long-short term memory model is introduced.

3.3. Exponential Smoothing-Long Short-Term Memory (ES-LSTM)

Exponential smoothing (ES) was proposed by Brown in [41], and it was later expanded by Holt and Winters. The main idea is to forecast future observations based on previous observations using a simple method of weighing past observations in a time series to predict the future. The model is often observed as suitable for data that do not have an expected upward or downward trend. The goal is to estimate the true level. This estimate of the level is then used to forecast future values. It is used in practice to forecast trends and seasonality. The Holt–Winters method would then be considered, which uses three smoothing Equations (3)–(5):

$$l_t = \alpha\left(\frac{y_t}{s_{t\,-\,m}}\right) + (1-\alpha) \times\ l_{t\,-1} b_{t\,-1} \tag{3}$$

$$l_t = \beta\left(\frac{l_t}{(l_{t\,-1})}\right) + (1-\beta) \times\ b_{t-1} \tag{4}$$

$$s_t = \gamma\left(\frac{y_t}{l_{t\,-1}\, b_{t\,-1}}\right) + (1-\gamma) \times\ s_{t\,-\,m} \tag{5}$$

$$y'_{t\,+\,h} = l_t b_t^h s_{t\,-\,m\,+\,hm} \tag{6}$$

where α, β, and γ are between 0 to 1, and the constant value of smoothing is set.

If the value is equal to 1, then the observation is current. If the value is equal to 0, then the forecasting will be same and naive. h step forecast is given by Equation (6). lt is the level of the observation at time $t.st$ is the seasonality of the observation at time t.

LSTM is a revolution in RNN which was presented by Hochreiter and Schmidhuber (1997) to process prior issues in RNN when supplementary cells are added, namely, the ability to learn long-term dependencies and remember information for a long time [38,42], the LSTM model is organized into a series structure. However, the unit of recurrence is different. Unlike standard RNN, LSTM has four interactive layers with an exceptional method of communication. The architecture of LSTM model has three gate modules: the input, output, and forget gates. The input gate is responsible for saving the information of the present, the output gate is responsible for giving information, and the forget gate chooses the information. As illustrated in Figure 3, the f_t gate denotes the exit gate and s_t gate output, and they control the cells state, and c_t stores the information $y.xt$ and h_t represents information and output for the LSTM, respectively.

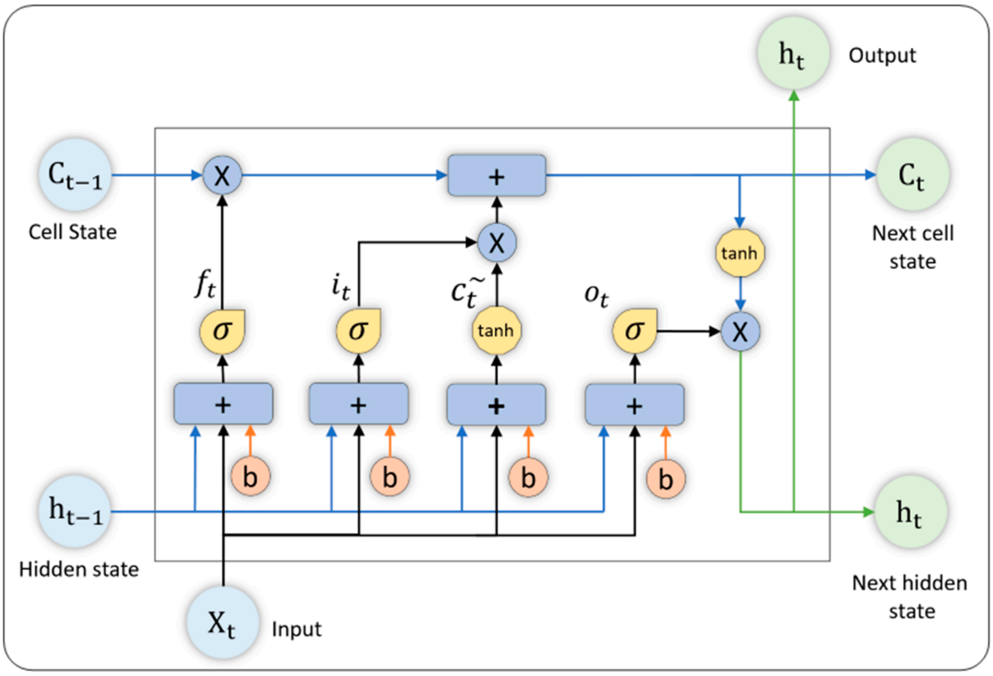

Figure 3. The structure of the LSTM model.

Initially, the LTSM cell identifies and discards undesirable information. The sigmoid function, which tracks the removal of the previous LSTM unit (h_{t-1}) at time $t-1$ and the current entry (x_t) at time t, determines this process of identification and exclusion. The purpose of this component determines the removal of the previous output. The forget gate (f_t), which has values ranging from 0 to 1, corresponding to each number in the case of the cell, $c_t \neq 1$.

$$\mathrm{f_t = \sigma\ (w_t[h_{t-1}x_t] + b_t)} \tag{7}$$

where σ is the sigmoidal function, w_t and b_t are the weights and the bias matrices, respectively.

The second step is to store and define the new input environment (x_t) to update the cell status. This process has two parts; first, whether the information should be updated or ignored (0 or 1) is determined as mandated by the Sigmoid function; second, the tanh layer ignores its weights for the values that can continue according to the level of importance (-1 to 1). The two values are multiplied to update the state of the cell. This new memory is added to the previous memory $(c_t \neq 1)$, which results in c_t.

$$\mathrm{i_t = \sigma\ (w_t[h_{t\ -1}x_t] + b_t)} \tag{8}$$

$$\mathrm{N_t = \tanh\ (w_t[h_{t\ -1}x_t] + b_n)} \tag{9}$$

$$\mathrm{C_t = C_{t\ -1}f_t + N_t i_t} \tag{10}$$

where C_{t-1} and C_t are state cells in time $t-1$ and t, while w and b are weights and bias matrices of state cells, respectively.

The third step is to determine the output (h_t) values based on the state of the output (o_t) cell but being a filtered copy. The sigmoidal layer decides that part of the cell state that reaches the exit. Then, the output gate doubles the new values generated by the tab layer of cell state (C_t), which varies between -1 and 1.

$$\mathrm{o_t = \sigma\ (w_o[h_{t\ -1}x_t] + b_o)} \tag{11}$$

$$\mathrm{h_t = o_t \tanh(C_t)} \tag{12}$$

where w_o and b_o correspond to weights and bias matrices of the output gate, respectively.

3.4. Time Series Analysis Forecasting-Based ES-LSTM

Time series refers to the number of observations over a time. Hence, it is necessary to analyze the coefficients of seasonal and level time series. It comprises of three layers, the first layer is the ES layer that acts as a filter for capturing and analyzing the time series, the second layer is the LSTM that works on capturing randomness in the time series, and finally the dense layer whose task is to collect the prediction from the first and second layers as shown in Equation (13). In addition, this model also uses wind (speed and direction and temperature) features to predict the amount of rain as shown in Figure 4.

$$\mathrm{LSTM}\ (\mathrm{X_t}) \times (\ \mathrm{s_t} \times \mathrm{l_t}) \quad (13)$$

X_t is the input state depending on time, s_t refers to the exponential smoothing and depends on time, and l_t denotes the LSTM depends on time.

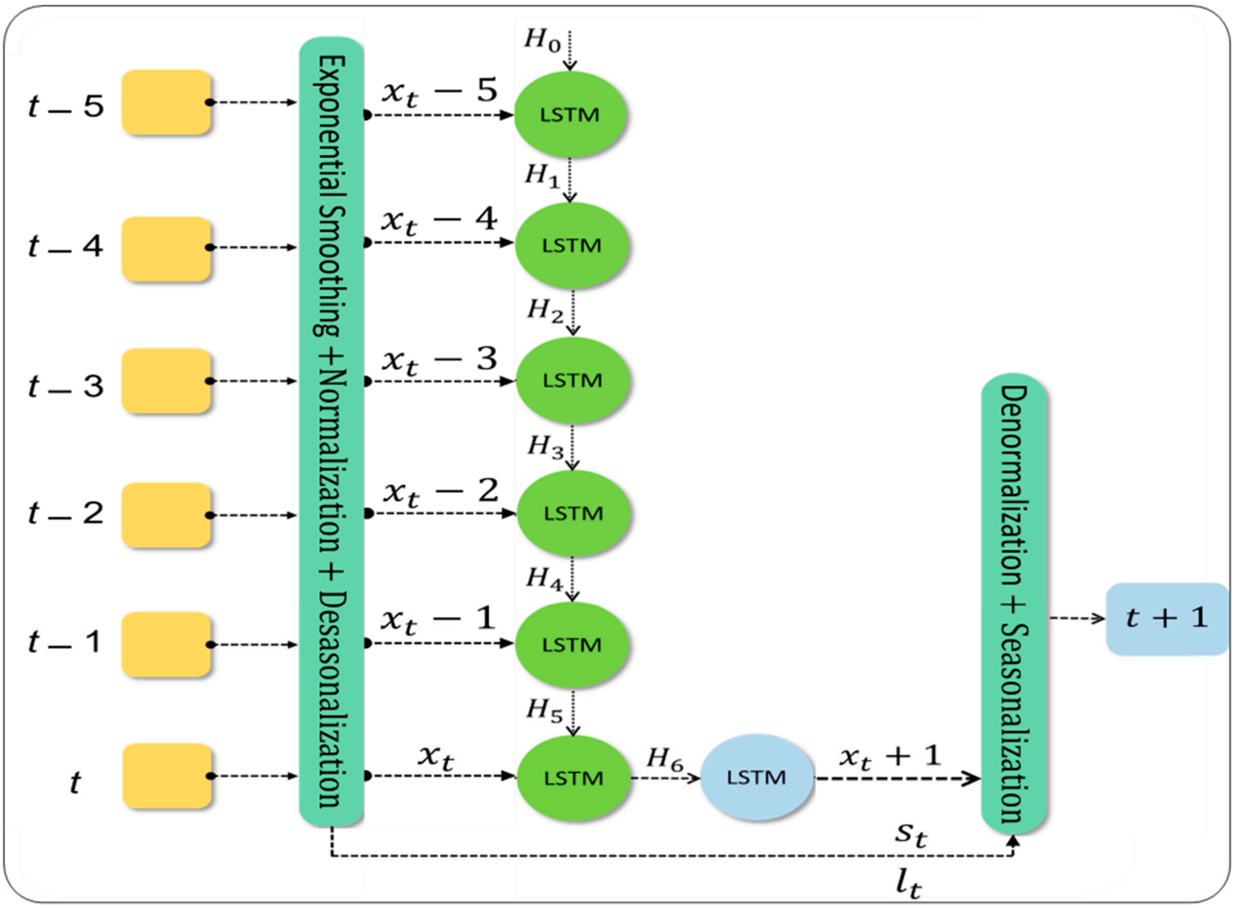

Figure 4. Exponential smoothing and LSTM architecture.

Additionally, before feeding the data into the ANN model, a sliding window technique (See Figure 5) is used to slice the time series data into a slice of window and target. The length of the window is assigned to six days, and the target is the seventh day. Further, the length of the seasonality will be one year to capture and learn the seasonality from the same day for the previous year.

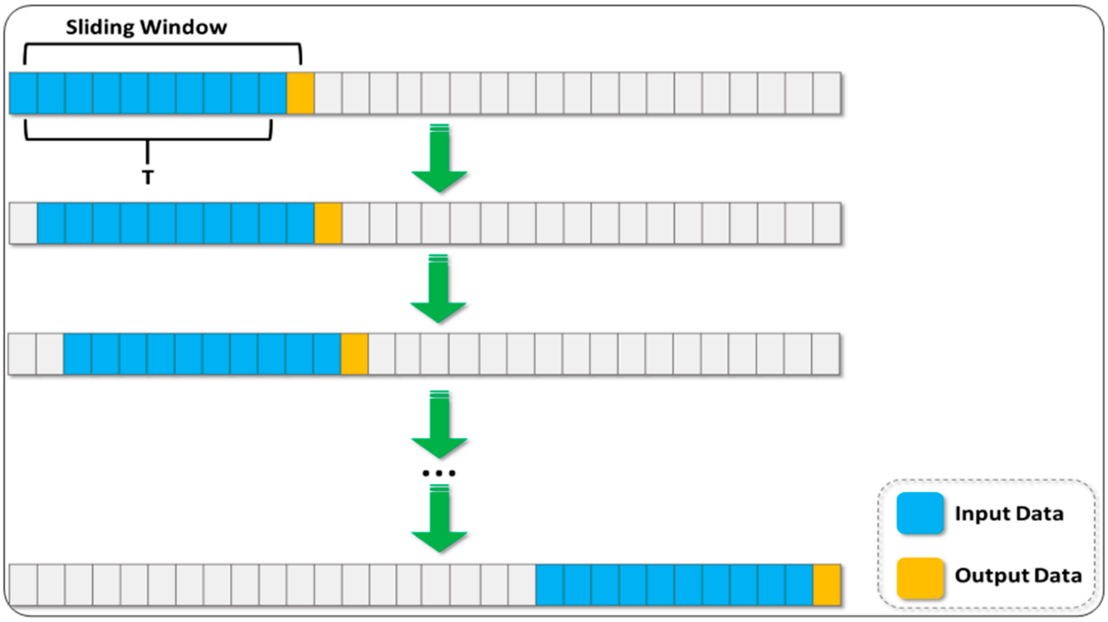

Figure 5. Steps of slide window.

The task of this model is to forecast the rain to reinforce the IoT devices capabilities in transferring from sleep mode to standby mode to be prepared for capturing the data from the sensors.

3.5. Artificial Neural Network (ANN) Model

The concept of ANN was first developed by McCulloch (1943) in [43]. It is modelled after the idea of how the human brain functions. An artificial neural network (ANN) is made up of multiple layers of neurons. ANN is one of these methods since it conducts a nonlinear mapping between inputs and outputs. Over the last few decades, ANN proves its efficacy in solving complex problem, such as function approximation and pattern recognition. Furthermore, ANN have the ability to produce meaningful solution to problems using learning and generalize from examples and input data even the input data contain some errors or are not completed. Another advantage of ANNs is their ability to function in real-time environments and store knowledge as the strength of the interconnecting weights (a numerical parameter) in ANNs is updated through a process known as learning, which employs a learning algorithm [44,45].

The application of ANNs to resolve hydrology-related problems has been documented in a number of papers. For example, French et al. [46] utilized ANNs to estimate rainfall for a watershed using artificial rainfall inputs, while Hsu et al. [47] used ANNs to model the rainfall-runoff process. Additionally, an ANN mimics this structure by distributing the computation across tiny, straightforward processing units, sometimes known as nodes or artificial neurons. An ANN has demonstrated to be a potent mathematical model using this architecture, excelling at function approximation and pattern recognition. In addition, nodes are connected to form the ANN. Nodes that share similar features are grouped together into layers. A layer can be thought of as a collection of nodes that link to other layers or the surrounding environment but do not interconnect.

A multilayer perceptron (MLP), a type of feed-forward ANN, was employed in this study to categorize the season of hourly precipitation. There are three layers of the MLP: the input layer, hidden layer, and output layer. These layers consist of neurons and perform independent computations on data before transferring it to another layer. The computation of the neuron is based on the weighted sum of the inputs. The network is fed with raw data, which is represented by the input unit. The function of each hidden unit is determined by the functions of the input units and the weights on the connections between the input units and the hidden units. The hidden units' function and the weights assigned to them in the output units' behavior determine the behavior of the hidden units.

Figure 6 shows a typical structure of an ANN which contains layers of several units and thus, it is termed as multilayer ANN. The input layer neurons receive signals from the outside world. Following that, the input layer sends the received data to the hidden layer neurons without any alteration or computations. Hence, output layer neurons provided ANN predictions [45].

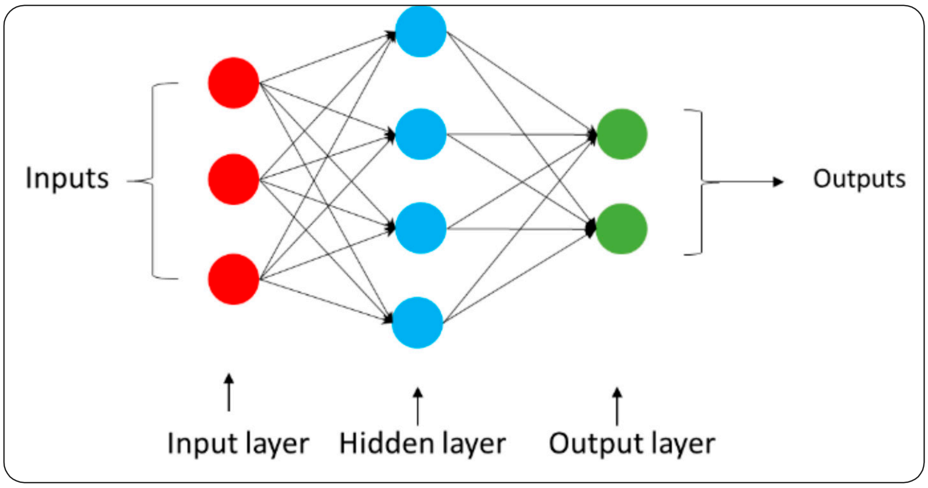

Figure 6. The typical structure of ANN model (adapted from [43]).

3.6. Decision Tree (DT)

The family of supervised ML algorithms includes the decision tree (DT) algorithm. It can be applied to regression problems as well as classification problems. The objective of this approach is to build a model that forecasts the value of a target variable. This is accomplished by using a DT, where the leaf node stands in for a class label and the internal node for an attribute [48].

3.7. Evaluation Metrics

In this work, the MAPE and accuracy metrics were utilized to evaluate the performance of the proposed models in terms of prediction errors and model accuracy. More information on those measures is provided in the subsections as follows.

3.7.1. Mean Absolute Percentage Error (MAPE)

The MAPE is used to compute prediction errors by dividing the absolute error for each period by the actual observed values for that period. The absolute percentage of error should then be averaged. This measure is helpful when determining the prediction accuracy depending on the size or magnitude of the prediction variable. MAPE denotes the level of prediction inaccuracy in comparison to the true value determined from Equation (14).

$$\mathrm{MAPE} = \frac{\sum_{t=1}^{n} \frac{|y_t - y'_t|}{y_t}}{n} \tag{14}$$

where y_t is the actual target at is the t, y'_t forecasting target resulting in t, n is the quantity predictive data.

3.7.2. Prediction Accuracy

It refers to the ratio of correctly classified rain status to the total number of samples in the dataset [43]. This metric can be computed using Equation (15):

$$\text{Accuracy (AC)} = \frac{\text{TN} + \text{TP}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \tag{15}$$

where TN (True Negative), TP (True Positive), FP (False Positive), and FN (False Negative).

4. Methodology of Proposed Work

The methodology of this work is conducted in three stages: (i) data preparation, (ii) prediction and forecasting models, and (iii) decision and response. The first stage focuses on preparing the dataset through several pre-processing steps. The second stage in contrast comprises two steps, predication, and forecasting using time series (Slide Window). As illustrated in Figure 7, the third stage is responsible for transferring out the final decision (flood prediction, email sending, and activation of the system sensors) in accordance with the findings from the previous stage.

4.1. Stage 1: Dataset Preparation

There are three steps involved in this stage: missing values, data transformation, and data normalization. The subsequent sections provide more details about those steps.

4.1.1. Step 1: Missing Values

The problem with missing values in time series is that occasionally the rain sensor is not working to indicate the occurrence of rain or not and the rain percentage, which leads to null values in the dataset records. Therefore, we address this issue by compensating zero value for the missing value. In addition, there are also missing values that need to be processed in numerical columns. Hence, we replaced all missing values with column mean value. However, for categorical columns. We applied the "get_dummies" function to convert the categorical variables into dummy/indicator variables.

4.1.2. Step 2: Data Transformation

The dataset contains a lot of object features in various formats. Such features should be in similar format (categorical, numeric, and text) to reduce processing time, computational overhead and obtain better results from the classification and prediction models.

In this work, we transform the features into numeric form, for instance the "date" feature is converted from object to date with the following formula (DD/MM/YYYY). Whereas "RainToday" and "RainTomorrow" features are represented as logical data. Consequently, we simply replace "yes" with 1 and "no" with 0.

4.1.3. Step 3: Data Normalization

In this stage, normalization is known as the transformation of the range of dataset values into a specific range of 0 and 1 values. The process requires a predefined range of values to accurately estimate the minimum and maximum of the observable values. Therefore, we used the available data for estimating those values. Estimating these predicted values can be challenging if the time series is moving up or down. Finally, this process is conducted using Equation (16):

$$z = \frac{x_i - \min(x)}{\max(x) - \min(x)} \tag{16}$$

where x_i is the input value, $\max(x)$ is the maximum value in dataset's record, and $\min(x)$ refers to the minimum value in the dataset's record.

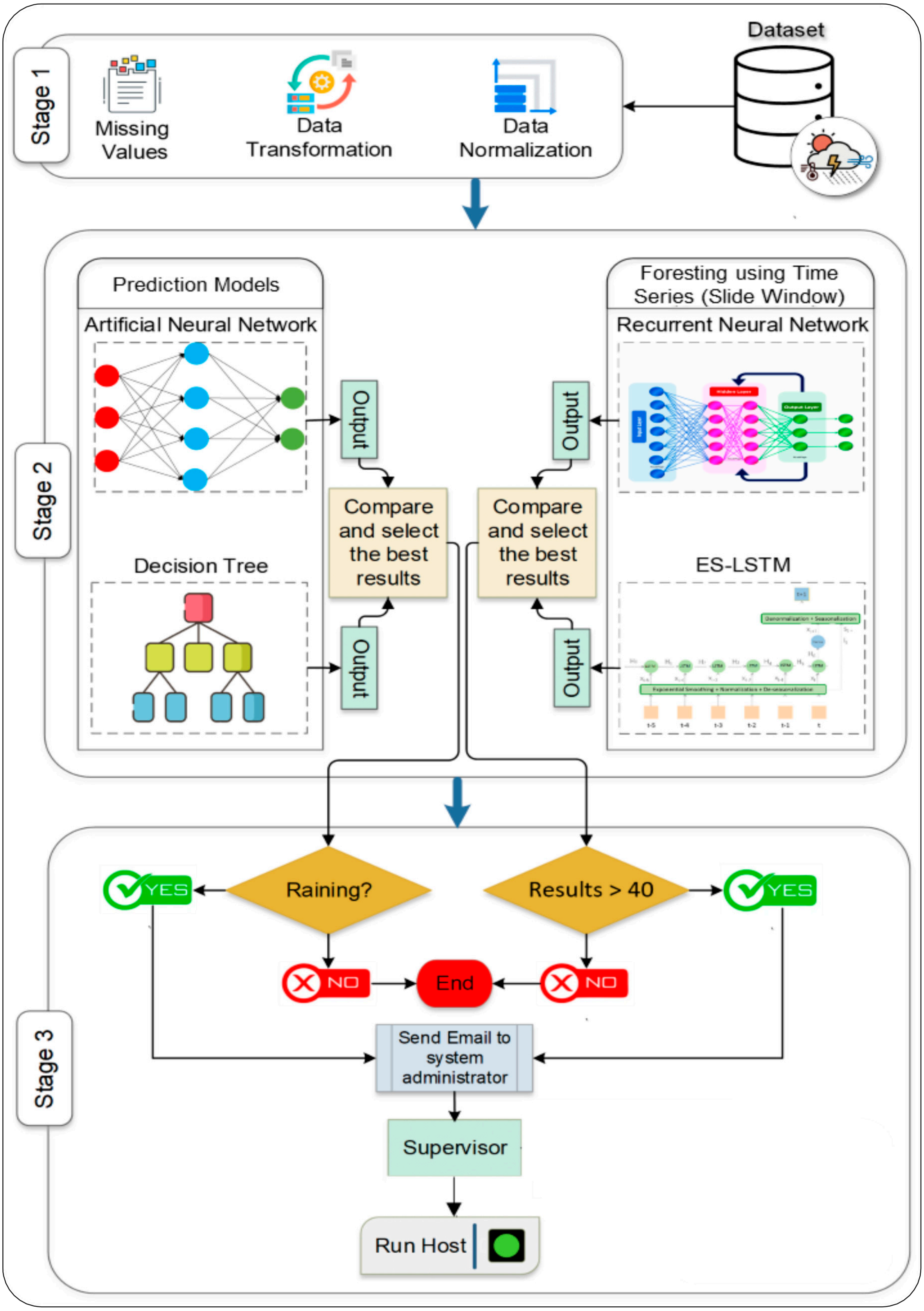

Figure 7. The methodology of the proposed approach.

4.2. Stage 2: Prediction and Forecasting Models

This stage consists of two main steps: prediction and forecasting using time series (Slide Window). The following section gives more information about the used algorithms.

4.2.1. Step 1: Prediction

This step comprises two algorithms: artificial neural network (ANN) and decision tree (DT). The following paragraphs provide more details about the role of those algorithms in the proposed methodology.

As for the role of ANN model in this stage, the Historical Daily Weather dataset is used to train ANN model for the purpose of obtaining the decision without human intervention. Wherein three layers are present: the first layer is made up of 118 neurons, and the second layer is made up of 59 neurons. In addition, we assign ReLU as an activation function: $\mathrm{f(x)} = \max(0, \mathrm{x})$. The final layer consists of a single artificial neuron, As shown in Figure 8. We utilize the sigmoid activation function (see, Equation (17)) for this layer as we have a binary prediction.

$$(\mathrm{x}) = \frac{1}{1 + e^{-\mathrm{K}\,(x\,-\,x_0)}} \tag{17}$$

where x is sigmoid's midway value, x_0 is the function's maximum value, and k is the logistic growth rate.

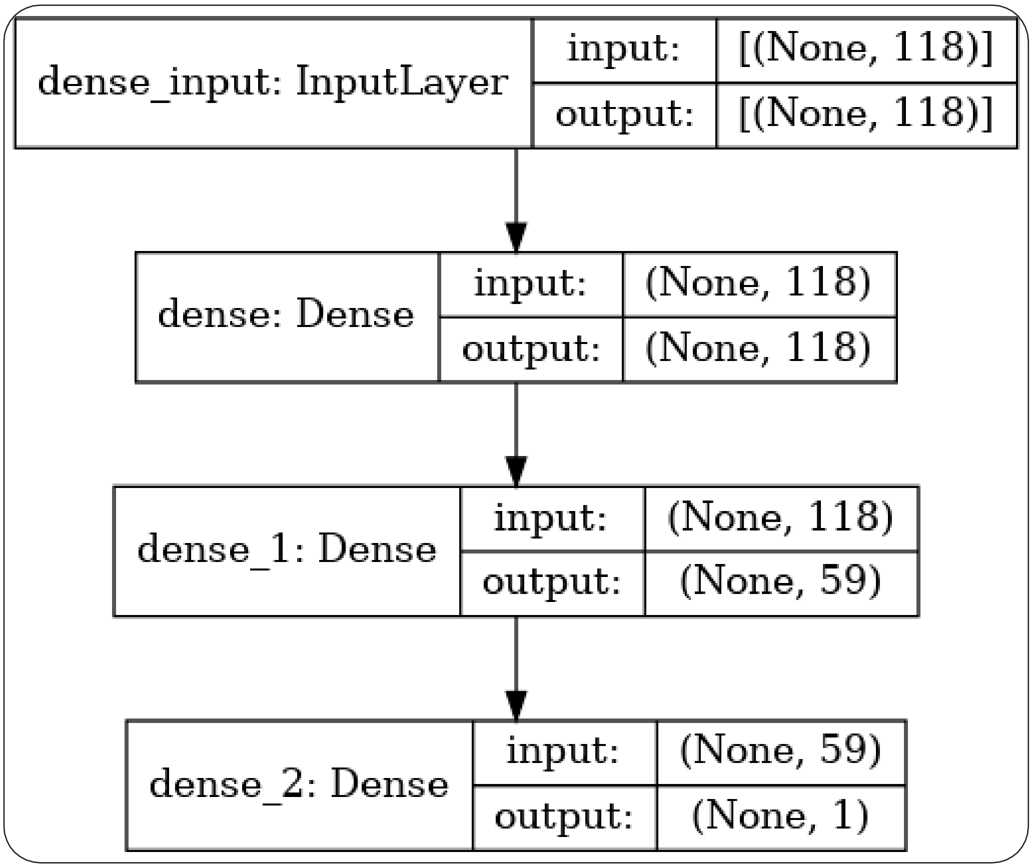

Figure 8. The ANN architecture.

It is deceptively simple, but by combining these simple decision-makers and finding ideal connection weights, we can make arbitrarily complex decisions and calculations which stretch far beyond what our biological brains allow. Additionally, we used a callback function called "Early Stopping" to end training when a monitored metric stops improving.

We compile our neural network with a binary cross-entropy as a loss function, and we use an Adam optimization [44]. The Adam optimization is a stochastic gradient descent variant that can be used besides of regular stochastic gradient descent to update network weights more efficiently.

Furthermore, the DT's classifier (j48) is used to predict the dataset records (Rain/Not Rain). After that, the outcome of the DT algorithm is compared with the results of ANN model and the best results are selected as input for the third stage.

4.2.2. Step 2: Forecasting Using Time Series (Slide Window)

This step addresses two issues. The first issue is relevant to regression, which is time series using two models: recurrent neural network (RNN) and exponential smoothing with LSTM (ES-LSTM). During the training time, the results of the models are compared against each other and, the highest results of the comparison will be used as an input to Sage 3 (Decision and Response). In addition, this step used the whole dataset to predict the occurrence of rain, unlike the first step, which only uses the rain data.

4.3. Stage 3: Decision and Response

This stage is designed to formalize the final decision and send a report to the supervisor in case the predefined conditions are satisfied. The first condition is related to rain percentage (i.e., in case the rain percentage is more than 40%), an email (notification) will be sent to the supervisor, which in turn switches on the rain detector sensors. Meanwhile, if there is a chance of rain occurrence (second condition), an email will be sent to the supervisor to turn on the rain detector sensor. As a result, these processes lead to the minimization of power consumption since the sensors only operate in the event of a possible rain occurrence. In addition, the supervisor should send an acknowledgement to the system to confirm the operation of the sensors within a specific time period. However, in case the supervisor failed to send an acknowledgement to the system. The proposed system will automatically send an alert to the sensors directly, which in turn sends a report for the presentence of rain to diminish flood incidence.

To sum up, the RNN and ES-LSTM models were involved for forecasting, whereas the DT and ANN models were utilized for prediction. In addition, the operation of the forecasting and prediction rely on the sensor' status. For example, if the rain sensors active, the prediction models start to predict the rain occurrence based on the collected data from rains' sensors. In addition, the forecasting models use the seasonal dataset for forecasting rain's occurrence. Finally, the outcome of both models (i.e.,) is considered in taking final decision.

5. Experimental Setup, Results, and Discussion

This section includes information on the dataset utilized in this study as well as the software and hardware requirements. Additionally, the experiment outcomes are examined and discussed. The software and hardware requirements used in this work are listed in Table 2.

Table 2. Hardware and software specifications.

	Specifications	Parameters
Hardware	CPU	Core i5-8th gen
	RAM	8 GB
Software	Operating System	Windows 10
	Programming Language	Python (Version 3.7)

5.1. Historical Daily Weather Dataset

In this work, we used a dataset collected from multiple locations across Australia [49]. The Australian Commonwealth Office of Meteorology created and processed a truly large model dataset to inform, analyze, and extract information. The dataset comprises a total of 11 features and 2359 records. In addition, the datasets records contain several measurements and parameters, such as mean temperature, daily rainfall total, minimum temperature, max wind speed, etc.

Furthermore, the rainfall feature is used in this work for two purposes: time-series analysis and for forecasting of Melbourne Airport Rainfall since the dataset does not contain missing values. In addition, all the features in the dataset were used for sliding window technique and ANN model, wherein the target (label) is the assigned as rain status tomorrow. It is also worth to mention that during data preprocessing steps, the data are divided into 80% learning set and 20% testing set.

The dataset being used contains some outliers. There are several processing stages that must be carried out, such as feature selection, which prepares the data for use in order to produce the best forecasts. Some characteristics in this work, such as on-going rainfall, wind speed, and direction, have an impact on floods. Since there is a time of year when rainfall is more frequent, the majority of these characteristics are seasonal. Figure 9 displays a sample of the used dataset.

Date	Station	Daily rainfall total (mm)	Highest 30 min rainfall (mm)	Highest 60 min rainfall (mm)	Highest 120 min rainfall (mm)	Mean temperature (Degrees celcius)	Maximum temperature (Degrees celcius)	Minimum temperature (Degrees celcius)	Mean wind speed (Km/h)	Max wind speed (Km/h)
2017-11-30	Admiralty	na	na	na	na	na	na	na	na	na
2017-11-29	Admiralty	na	na	na	na	na	na	na	na	na
2017-11-28	Admiralty	na	na	na	na	na	na	na	na	na
2017-11-27	Admiralty	na	na	na	na	na	na	na	na	na
2017-11-26	Admiralty	na	na	na	na	na	na	na	na	na
2017-11-25	Admiralty	na	na	na	na	na	na	na	na	na
2017-11-24	Admiralty	na	na	na	na	na	na	na	na	na
2017-11-23	Admiralty	na	na	na	na	na	na	na	na	na
2017-11-22	Admiralty	na	na	na	na	na	na	na	na	na
2017-11-21	Admiralty	na	na	na	na	na	na	na	na	na
2017-11-20	Admiralty	na	na	na	na	na	na	na	na	na
2017-11-19	Admiralty	0	0	0	0	28.2	31.7	26.1	7.2	28.4
2017-11-18	Admiralty	58	47.8	55.6	56.4	27.6	34.5	23.2	5	40.3
2017-11-17	Admiralty	0.2	0.2	0.2	0.2	27.5	31.7	24.2	6.5	44.3
2017-11-16	Admiralty	16.2	14.4	16.2	16.2	27.8	33	25	6.5	31.3

Figure 9. A sample of the used dataset.

5.2. Results of the Proposed Approach

This section illustrates the results and parameters used for the proposed approach. Tables 3 and 4 depict the utilized parameters for DT, ANN, ES-LSTM and RNN model in this work, respectively.

Table 3. The utilized configuration parameters for DT model.

Models	Criterion	Max_Depth	Metric
DT	Gini	7	Accuracy

As shown in the table above, we used the Gini index for criterion and the max_depth value is assigned to 7.

Table 4. The used configuration parameters for ANN, ES-LSTM, and RNN models.

Model	Optimizer	Loss	Epochs	Metrics
ANN	Adam	binary_crossentropy	26	Accuracy
ES-LSTM	RMSprop	mse	10	MAPE
RNN	Adam	mse	10	MAPE

Several algorithms with different contexts are used to test the precipitation detection and monitoring system. In this work, the algorithm used for prediction by time-series analysis of precipitation is ES-LSTM. Table 5 shows the results of MAPE validation and testing.

Table 5. Results of MAPE for time series (slide window) step.

Model	MAPE
ES-LSTM	3.17
RNN	6.42

The results in the Table 5 above indicate that the MAPE error is small which indicates that the model is working well. Nevertheless, there are some mis-predictions in the obtained results which is owning to the outliers. On the other hand, during the experiments, we discover that there are some flooding sensors that do not operate occasionally, so it was replaced by zero values in the used dataset. Figure 10 shows the plot between the actual and the forecasting values using ES-LSTM.

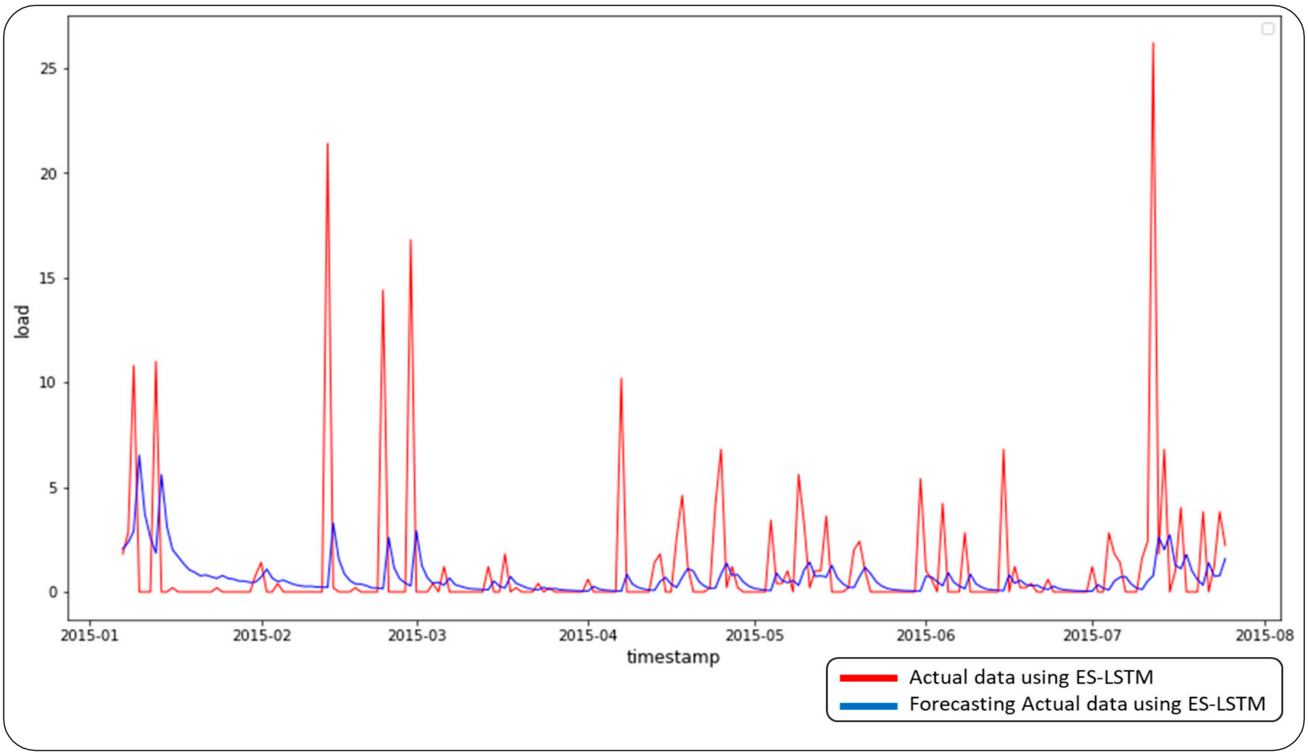

Figure 10. A comparison between actual and forecast data using ES-LSTM.

As mentioned above, the aim of this experiment is to discover the occurrence of rain through DL using ANN. Table 6 shows that the accuracy of the ANN model is higher than that of the DT classifier.

Table 6. Results of ANN and DT models.

Time Series ES-LSTM	Accuracy (%)
ANN	96.65
DT	84.0

In addition, to visualize the performance of the classification model on a specific classification problem, the confusion matrix is used which consists of rows and columns. The rows of the matrix refer to the actual class, whereas the columns represent the predicted class, or vice versa. In addition, this matrix is used for both binary and multi-class classification and relies on four parameters (i.e., FP, FN, TP, TN) that provide the total count of correct and incorrect instances predicted by the classification model. Finally, the computation of performance measures such as recall, accuracy, precision, and F-measure uses the four parameters that were obtained. Table 7 exhibits the confusion matrix performance results.

Table 7. The performance results of confusion matrix.

		Predicted	
		No Rain	**Rain**
Actual	**No Rain**	TP = 0.98098	FN = 0.06833
	Rain	FP = 0.000141	TN = 0.999595

From the table above, it can be noted that confusion matrix shows satisfactory results with a prediction of 0.0683 of (Rain) occurrence, which the actual case is (No Rain). Moreover, the model predicated 0.0001 of (No Rain). In addition, Table 8 displays the results of various evaluation metrics for the obtained outcome of the ANN model.

Table 8. The results of evaluation metrics.

Evaluation Metrics	Mathematical Model	Results
Sensitivity	$\frac{TP}{TP+FN}$	1.04931
Specificity	$\frac{TN}{TN+FP}$	0.999736
Precision	$\frac{TP}{TP+FP}$	0.981121
Negative Predictive Value	$\frac{TN}{TN+FN}$	1.067925

Table 8 presents and discusses the experimental findings of the ANN-proposed model and demonstrates how well the ANN model performed using the evaluation metrics of Sensitivity, Specificity, Precision, and Negative Predictive Value. The findings from the results shows that the result of the sensitivity evaluation metric of our ANN proposed model is 1.04931 and it significantly performed better than the others evaluation metrics which are Specificity, Precision, and Negative Predictive Value. While the result of the Negative Predictive Value evaluation metric of ANN proposed model is 1.067925 and it performed better than Specificity and Precision evaluation metrics. The results of the proposed ANN model based on Specificity and Precision evaluation metrics are 0.999736 and 0.981121 respectively.

5.3. The Proposed Prototype

MQTT is a standard publish-subscribe network protocol that allows devices to exchange messages. Despite the fact that MQTT typically works with TCP/IP, it can work with any network protocol that permits ordered, lossless, two-way communication. It is intended for connectivity to distant places that have resource constraints or have limited network capacity. The protocol is an open OASIS standard (ISO/IEC 20922) and an ISO recommendation [50]. Figure 11 presents the flow diagram for the proposed prototype.

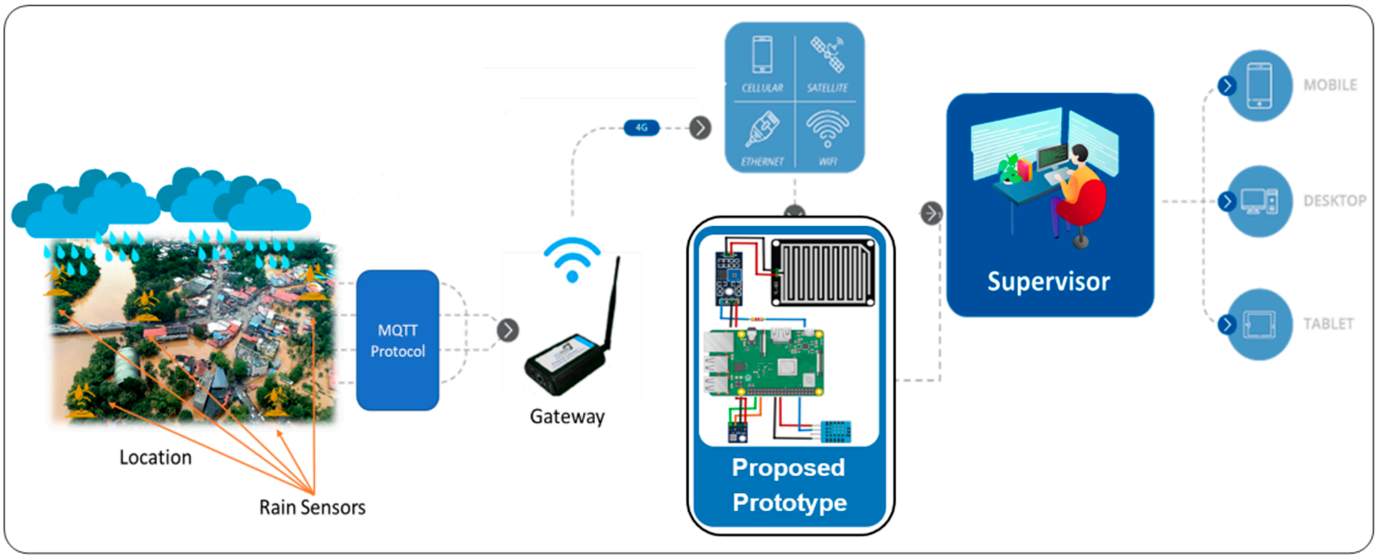

Figure 11. A flow diagram for the proposed prototype.

The proposed prototype is represented in the figure above as a group of rain installed sensors all across the city. Then, the MQTT protocol will be used to send the gathered data from the sensors to the gateway. After that, based on the information provided, the proposed prediction and forecasting model will then analyze the received information and decide based on the pre-trained model. Finally, according to the decision provided in the third module (decision and response module). Finally, if an alert is triggered, the supervisor will be notified to make the final decision and inform the rescue team about the situation.

6. Conclusions

The objective of this study is to investigate the applicability of ML and DL techniques in the event of floods in Melbourne Airport, Australia. The link between the precipitation data was the subject of the conclusions reached which is regarded as one of the most important elements influencing the incidence of floods.

In addition, this research tackles the flood prediction using two different models in order to aid disaster response efforts and curb casualties. The first model is built using ANN and DT to predict the rain based using the dataset gathered by sensors. Whereas the second model is constructed using ES-LSTM and RNN model to analyze seasonal dataset for rain forecasting. Furthermore, the simulation results reveal that the proposed model achieves promising results in term of MAPE and prediction accuracy. Meanwhile, the proposed model proves its capability to provide an early prediction of rain incidence.

In terms of future work, we intend to improve the performance of the suggested model by addressing the following issues. In order to develop a strong prediction and forecasting system, it is necessary to: (i) Enhance the performance of Section DL algorithms through turning and optimizing the weight parameters; (ii) consolidate a new group of ML and DL algorithms to build solid prediction and forecasting system; (iii) optimize the window size duration; and (iv) feed new characteristics into the prediction models.

Author Contributions: Conceptualization, G.A.A.-A. and H.A.Y.; methodology, G.A.A.-A., H.A.Y. and T.A.A.-A.; software, G.A.A.-A.; writing—original draft, G.A.A.-A., H.A.Y., I.M.H. and T.A.A.-A.; writing—review and editing, F.S., G.A.A.-A., H.A.Y., I.M.H., M.N., T.A.A.-A. and W.G.; supervision, H.A.Y. and T.A.A.-A.; project administration, F.S., M.N. and W.G.; funding acquisition, W.G., F.S. and M.N. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The dataset used in this work is openly available for researchers and can be downloaded directly from the following link: https://data.gov.sg/dataset/historical-daily-weather (accessed on 25 January 2023).

Acknowledgments: The University of Basrah, Universiti Sains Malaysia (USM), University of Tabuk, and Birmingham City University provided assistance and resources that made it possible for us to complete this research. We would like to convey our gratitude for all their help.

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

References

1. Li, Y.; Yu, G.; Zhang, J. A three-stage stochastic model for emergency relief planning considering secondary disasters. *Eng. Optim.* **2020**, *53*, 551–575. [CrossRef]
2. Seager, R.; Naik, N.; Baethgen, W.; Robertson, A.; Kushnir, Y.; Nakamura, J.; Jurburg, S. Tropical Oceanic Causes of Interannual to Multidecadal Precipitation Variability in Southeast South America over the Past Century. *J. Clim.* **2010**, *23*, 5517–5539. [CrossRef]
3. Hodnebrog, Ø.; Myhre, G.; Forster, P.M.; Sillmann, J.; Samset, B.H. Local biomass burning is a dominant cause of the observed precipitation reduction in southern Africa. *Nat. Commun.* **2016**, *7*, 11236. [CrossRef]
4. Bishop, D.A.; Williams, A.P.; Seager, R.; Fiore, A.M.; Cook, B.I.; Mankin, J.S.; Singh, D.; Smerdon, J.E.; Rao, M.P. Investigating the Causes of Increased Twentieth-Century Fall Precipitation over the Southeastern United States. *J. Clim.* **2018**, *32*, 575–590. [CrossRef] [PubMed]
5. Lara-Benítez, P.; Carranza-García, M.; Luna-Romera, J.M.; Riquelme, J.C. Temporal Convolutional Networks Applied to Energy-Related Time Series Forecasting. *Appl. Sci.* **2020**, *10*, 2322. [CrossRef]
6. Dada, E.G.; Yakubu, H.J.; Oyewola, D.O. Artificial Neural Network Models for Rainfall Prediction. *Eur. J. Electr. Eng. Comput. Sci.* **2021**, *5*, 30–35. [CrossRef]
7. Chifflard, P.; Kranl, J.; Strassen, G.Z.; Zepp, H. The significance of soil moisture in forecasting characteristics of flood events. A statistical analysis in two nested catchments. *J. Hydrol. Hydromech.* **2017**, *66*, 1–11. [CrossRef]
8. Povinelli, R.J. Identifying Temporal Patterns for Characterization and Prediction of Financial Time Series Events. In *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*; Springer: Berlin, Heidelberg, 2001; Volume 2007, pp. 46–61. ISBN 9783540417736.
9. Laio, F.; Porporato, A.; Revelli, R.; Ridolfi, L. A comparison of nonlinear flood forecasting methods. *Water Resour. Res.* **2003**, *39*, 1–4. [CrossRef]
10. Cheng, M.; Fang, F.; Kinouchi, T.; Navon, I.; Pain, C. Long lead-time daily and monthly streamflow forecasting using machine learning methods. *J. Hydrol.* **2020**, *590*, 125376. [CrossRef]
11. Feng, P.; Wang, B.; Liu, D.L.; Ji, F.; Niu, X.; Ruan, H.; Shi, L.; Yu, Q. Machine Learning-Based Integration of Large-Scale Climate Drivers Can Improve the Forecast of Seasonal Rainfall Probability in Australia. *Environ. Res. Lett.* **2020**, *15*, 084051. [CrossRef]
12. Shi, H.; Xu, M.; Li, R. Deep Learning for Household Load Forecasting—A Novel Pooling Deep RNN. *IEEE Trans. Smart Grid* **2017**, *9*, 5271–5280. [CrossRef]
13. Salman, A.G.; Kanigoro, B.; Heryadi, Y. Weather forecasting using deep learning techniques. In Proceedings of the 2015 International Conference on Advanced Computer Science and Information Systems (ICACSIS), Depok, Indonesia, 10–11 October 2015; IEEE: New York, NY, USA; pp. 281–285.
14. Liu, Z.; Sullivan, C.J. Prediction of weather induced background radiation fluctuation with recurrent neural networks. *Radiat. Phys. Chem.* **2019**, *155*, 275–280. [CrossRef]
15. Biswas, S.K.; Sinha, N.; Purkayastha, B.; Marbaniang, L. Weather prediction by recurrent neural network dynamics. *Int. J. Intell. Eng. Inform.* **2014**, *2*, 166. [CrossRef]
16. Akram, M.; El, C. Sequence to Sequence Weather Forecasting with Long Short-Term Memory Recurrent Neural Networks. *Int. J. Comput. Appl.* **2016**, *143*, 7–11. [CrossRef]
17. Balogun, A.-L.; Adebisi, N. Sea level prediction using ARIMA, SVR and LSTM neural network: Assessing the impact of ensemble Ocean-Atmospheric processes on models' accuracy. *Geomat. Nat. Hazards Risk* **2021**, *12*, 653–674. [CrossRef]
18. Damavandi, H.G.; Shah, R.; Stampoulis, D.; Wei, Y.; Boscovic, D.; Sabo, J. Accurate Prediction of Streamflow Using Long Short-Term Memory Network: A Case Study in the Brazos River Basin in Texas. *Int. J. Environ. Sci. Dev.* **2019**, *10*, 294–300. [CrossRef]
19. Smyl, S. A hybrid method of exponential smoothing and recurrent neural networks for time series forecasting. *Int. J. Forecast.* **2020**, *36*, 75–85. [CrossRef]
20. Al-Amiedy, T.A.; Anbar, M.; Belaton, B.; Kabla, A.H.H.; Hasbullah, I.H.; Alashhab, Z.R. A Systematic Literature Review on Machine and Deep Learning Approaches for Detecting Attacks in RPL-Based 6LoWPAN of Internet of Things. *Sensors* **2022**, *22*, 3400. [CrossRef]

21. D'Addabbo, A.; Refice, A.; Pasquariello, G.; Bovenga, F.; Chiaradia, M.T.; Nitti, D.O. A Bayesian network for flood detection. In Proceedings of the 2014 IEEE Geoscience and Remote Sensing Symposium, Quebec City, QC, Canada, 13–18 July 2014; IEEE: New York, NY, USA; pp. 3594–3597.
22. Wu, Y.; Wang, Y. A portable flood detection system using heterogeneous sensor networks. In Proceedings of the 2014 IEEE 33rd International Performance Computing and Communications Conference (IPCCC), Austin, TX, USA, 5–7 December 2014; IEEE: New York, NY, USA; pp. 1–2.
23. Khalaf, M.; Hussain, A.J.; Al-Jumeily, D.; Fergus, P.; Idowu, I.O. Advance flood detection and notification system based on sensor technology and machine learning algorithm. In Proceedings of the 2015 International Conference on Systems, Signals and Image Processing (IWSSIP), London, UK, 10–12 September 2015; IEEE: New York, NY, USA; pp. 105–108. [CrossRef]
24. Ta, A. *The Protective Role of Mitochondrial Akt1 Against the Development of Diabetic Cardiomyopathy*; University of California: Irvine, CA, USA, 2021; pp. 1–109.
25. Kratzert, F.; Klotz, D.; Brenner, C.; Schulz, K.; Herrnegger, M. Rainfall–runoff modelling using Long Short-Term Memory (LSTM) networks. *Hydrol. Earth Syst. Sci.* **2018**, *22*, 6005–6022. [CrossRef]
26. Hu, Y.; Yan, L.; Hang, T.; Feng, J. Stream-Flow Forecasting of Small Rivers Based on LSTM. *arXiv* **2020**, arXiv:2001.05681.
27. Widiasari, I.R.; Nugoho, L.E.; Widyawan; Efendi, R. Context-based Hydrology Time Series Data for A Flood Prediction Model Using LSTM. In Proceedings of the 2018 5th International Conference on Information Technology, Computer, and Electrical Engineering (ICITACEE), Semarang, Indonesia, 27–28 September 2018; IEEE: New York, NY, USA; pp. 385–390.
28. Mousavi, F.S.; Yousefi, S.; Abghari, H.; Ghasemzadeh, A. Design of an IoT-based Flood Early Detection System using Machine Learning. In Proceedings of the 26th International Computer Conference, Computer Society of Iran, CSICC 2021, Tehran, Iran, 3–4 March 2021; IEEE: New York, NY, USA; pp. 1–7.
29. Zhang, J.; Zhu, Y.; Zhang, X.; Ye, M.; Yang, J. Developing a Long Short-Term Memory (LSTM) based model for predicting water table depth in agricultural areas. *J. Hydrol.* **2018**, *561*, 918–929. [CrossRef]
30. Xiang, Z.; Yan, J.; Demir, I. A rainfall-runoff model with LSTM-based sequence-to-sequence learning. *Water Resour. Res.* **2020**, *56*, e2019WR025326. [CrossRef]
31. Dong, L.; Fang, D.; Wang, X.; Wei, W.; Damaševičius, R.; Scherer, R.; Woźniak, M. Prediction of Streamflow Based on Dynamic Sliding Window LSTM. *Water* **2020**, *12*, 3032. [CrossRef]
32. Won, Y.-M.; Lee, J.-H.; Moon, H.-T.; Moon, Y.-I. Development and Application of an Urban Flood Forecasting and Warning Process to Reduce Urban Flood Damage: A Case Study of Dorim River Basin, Seoul. *Water* **2022**, *14*, 187. [CrossRef]
33. Atashi, V.; Gorji, H.T.; Shahabi, S.M.; Kardan, R.; Lim, Y.H. Water Level Forecasting Using Deep Learning Time-Series Analysis: A Case Study of Red River of the North. *Water* **2022**, *14*, 1971. [CrossRef]
34. Kunverji, K.; Shah, K.; Shah, N. A Flood Prediction System Developed Using Various Machine Learning Algorithms. In Proceedings of the 4th International Conference on Advances in Science & Technology (ICAST2021), Mumbai, India, 7 May 2021; pp. 1–6. [CrossRef]
35. Fahad, S.; Su, F.; Khan, S.U.; Naeem, M.R.; Wei, K. Implementing a novel deep learning technique for rainfall forecasting via climatic variables: An approach via hierarchical clustering analysis. *Sci. Total. Environ.* **2023**, *854*, 158760. [CrossRef] [PubMed]
36. Chen, J.; Li, Y.; Zhang, C.; Tian, Y.; Guo, Z. Urban Flooding Prediction Method Based on the Combination of LSTM Neural Network and Numerical Model. *Int. J. Environ. Res. Public Health* **2023**, *20*, 1043. [CrossRef] [PubMed]
37. Wawrzyniak, Z.M.; Jankowski, S.; Szczechla, E.; Szymanski, Z.; Pytlak, R.; Michalak, P.; Borowik, G. Data-driven models in machine learning for crime prediction. In Proceedings of the 2018 26th International Conference on Systems Engineering (ICSEng), Sydney, Australia, 18–20 December 2018; IEEE: New York, NY, USA; pp. 1–8.
38. Le, X.H.; Ho, H.V.; Lee, G.; Jung, S. Application of Long Short-Term Memory (LSTM) Neural Network for Flood Forecasting. *Water* **2019**, *11*, 1387. [CrossRef]
39. Hasan, A.H.; Anbar, M.; Alamiedy, T.A. Deep learning approach for detecting router advertisement flooding-based DDoS attacks. *J. Ambient. Intell. Humaniz. Comput.* **2022**, 1–15. [CrossRef]
40. Williams, R.J.; Zipser, D. Gradient-based learning algorithms for recurrent networks and their computational complexity. *Back-Propag. Theory Archit. Appl.* **1995**, *433*, 433–486.
41. Brown, R.G.; Meyer, R.F. The Fundamental Theorem of Exponential Smoothing. *Oper. Res.* **1961**, *9*, 673–685. [CrossRef]
42. Zhao, J.; Obonyo, E. Convolutional long short-term memory model for recognizing construction workers' postures from wearable inertial measurement units. *Adv. Eng. Inform.* **2020**, *46*, 101177. [CrossRef]
43. Luk, K.C.; Ball, J.; Sharma, A. An application of artificial neural networks for rainfall forecasting. *Math. Comput. Model.* **2001**, *33*, 683–693. [CrossRef]
44. Zhang, Z. Improved Adam Optimizer for Deep Neural Networks. In Proceedings of the 2018 IEEE/ACM 26th International Symposium on Quality of Service (IWQoS), Banff, AB, Canada, 4–6 June 2018; pp. 1–2. [CrossRef]
45. French, M.N.; Krajewski, W.F.; Cuykendall, R.R. Rainfall forecasting in space and time using a neural network. *J. Hydrol.* **1992**, *137*, 1–31. [CrossRef]
46. Djavanroodi, F.; Omranpour, B.; Sedighi, M. Artificial Neural Network Modeling of ECAP Process. *Mater. Manuf. Process.* **2013**, *28*, 276–281. [CrossRef]
47. Hayder, I.M.; Al Ali, G.A.N.; Younis, H.A. Predicting reaction based on customer's transaction using machine learning approaches. *Int. J. Electr. Comput. Eng. IJECE* **2023**, *13*, 1086–1096. [CrossRef]

48. Damarla, S.K. Seshu-Damarla/Gradient-Descent-with-Adam-for-MLP-Network. Release v1.1.0. GitHub. Available online: https://github.com/seshu-damarla/Gradient-Descent-with-Adam-for-MLP-Network/releases/tag/v1.1.0 (accessed on 21 January 2023).
49. National Environment Agency Historical Daily Weather Data Files. Available online: https://data.gov.sg/dataset/historical-daily-weather?view_id=afc04c92-67f6-4e8f-97ed-71a3b4155212&resource_id=1e478275-0746-483d-9783-2f40a3535910 (accessed on 21 January 2023).
50. Geyer, C. OASIS MQTT Internet of Things Standard Now Approved by ISO/IEC JTC1. Available online: https://www.oasis-open.org/printpdf/news/pr/oasis-mqtt-internet-of-things-standard-now-approved-by-iso-iec-jtc1%0Ahttps://www.oasis-open.org/news/pr/oasis-mqtt-internet-of-things-standard-now-approved-by-iso-iec-jtc1 (accessed on 26 January 2023).