STATISTICS, OPTIMIZATION AND INFORMATION COMPUTING
Stat., Optim. Inf. Comput., Vol. **16**, October 2026, pp 3272–3293.
Published online in International Academic Press (www.IAPress.org)

Hybrid Modeling Combining Neutrosophic Logic and Bayesian Deep Learning for Modeling Precipitation in Northern Iraq

Aymen M. Hameed, Muzahem M. Al-Hashimi*

Department of Statistics and Informatics, University of Mosul, Mosul 41002, Iraq

Abstract This paper proposes a hybrid model combining neutrosophic logic and Bayesian deep learning to enhance the accuracy of precipitation forecasting in northern Iraq. Precipitation does not depend on a single climatic factor but results from the simultaneous interaction of several physical factors. The model uses specific humidity (QV2M), temperature at a height of 2 meters (T2M), soil moisture (GWETTOP), wind speed at 2 meters (WS2M), and their interactions and lagged variables. To reduce skewness in the target distribution, the Yeo–Johnson transformation was applied. Neutrosophic logic was used to transform the climatic variables into three components: truth (T), indeterminacy (I), and falsity (F). A Bayesian Neural Network (BNN) was then trained using the proposed Bayesian neural network architecture. A deep ensemble strategy was employed, using three models to generate the final estimates. The proposed model achieved high forecasting accuracy. The results indicate that the model explained approximately 89.30% of the variance in actual precipitation data. NSE and R^2 yielded very similar values, both ≈ 0.90. The values ($RMSE = 8.4081,\ MAE = 4.9232$) indicate relatively low forecasting errors, given observed monthly precipitation occasionally exceeding 100 mm, suggesting high forecasting accuracy. Therefore, combining neutrosophic logic with Bayesian neural networks offers a promising way to understand climate variability better and support sustainable water resources management.

Keywords Precipitation Forecasting, Neutrosophic Logic, Bayesian Neural Networks, Probabilistic Forecasting, Climate Variables, Northern Iraq

AMS 2010 subject classifications 62M45, 62F15, 62M20, 62P12

DOI: 10.19139/soic-2310-5070-4158

1. Introduction

Traditional machine learning models have shown strong predictive performance. They may remain challenging to generalize, particularly for complex regression tasks [42], suffer from overfitting, and rely on point estimates, which are limited in capturing uncertainties due to factors such as variations in weather or precipitation [47]. Probabilistic forecasting provides a more comprehensive representation of uncertainty [31, 59, 5]. Machine learning techniques have been widely used to capture complex patterns in data, for example, in time-series analysis [44, 34] and financial applications [21, 10]. Data-driven approaches and machine learning techniques such as LSTM, CNN, and GRU can enhance forecasting, decision-making, and reliability when uncertainty quantification is incorporated into predictive models [29, 39]. Hybrid architectures have also been developed to enhance generalization and increase predictive power [20, 3, 56]. Many studies have shown that Bayesian models and Bayesian neural networks can provide accurate predictions and uncertainty estimates, model complex relationships, support reliable predictions, and inform risk-based decisions [17, 29, 38, 60]. These studies also show that Bayesian models can capture uncertainty in the weight distribution and in predictions, thereby providing greater reliability than deterministic models. Precipitation forecasting is of critical importance to many sectors [53]. Accurate forecasts can reduce

*Correspondence to: Muzahem M. Al-Hashimi, Department of Statistics and Informatics, University of Mosul, Mosul 41002, Iraq.
Email: muzahim_alhashime@uomosul.edu.iq

ISSN 2310-5070 (online) ISSN 2311-004X (print)

the risk of floods and droughts [43, 50] and improve irrigation in agriculture [53]. Predicting precipitation is a challenging problem due to measurement errors, large-scale datasets, atmospheric instability, and extreme-weather variability [7]. Precipitation is also affected by primary variables such as temperature, mean humidity, dew point, wind speed, atmospheric pressure, and radiation [32, 26]. Weather data are often irregular and unpredictable, which makes it difficult to capture using traditional methods [13]. Previous studies have shown that the use of Bayesian methods in neural network training results in better weather forecasting performance [22, 55, 49], providing adaptive parameter estimation, which leads to greater flexibility and reliability in the results [45], and helping improve prediction while overcoming the limitations of other models [23, 24, 52]. This paper contributes by combining Neutrosophic Logic and Bayesian Deep Learning to enhance the accuracy and reliability of precipitation forecasting in northern Iraq, thereby addressing the gap between advanced statistical models and their ability to effectively capture the complex and dynamic nature of precipitation. This aids in effective environmental planning and natural disaster management strategies.

2. Theoretical Background

2.1. The Bayesian paradigm

In classical statistics, probability is defined as the long-run relative frequency of an event occurring among an infinite number of trials of an experiment [27, 8]. In the Bayesian framework, probability represents a degree of belief about whether an event will occur. This allows the model to provide a probabilistic estimate of uncertainty even with very limited or rare data. Bayesian theory relies on two ideas: First, Prior Belief, which is the belief or prior knowledge we hold before seeing the data. This is denoted by $p(\theta)$, where θ represents the model's coefficients (weights), and $p(\theta)$ is our prior belief about these coefficients [40]. For new data that provides additional information, the probability of the observed data is denoted by $p(D \mid \theta)$ and is called the likelihood. Combining prior knowledge with new data (likelihood), we obtain $p(\theta \mid D)$. This is called the posterior distribution, which represents the updated belief about the model parameters θ after viewing and analyzing the data D. Bayes' equation can be expressed as [30]:

$$p(\theta \mid D) = \frac{p(D \mid \theta)p(\theta)}{p(D)} = \frac{\text{Prior} \times \text{Likelihood}}{\text{Evidence}}$$

Bayesian neural networks rely on the same statistical concepts, but they apply them to the weights of deep neural networks. Unlike traditional neural networks, which treat weights as constant values, Bayesian neural networks treat weights as probability distributions. Therefore, approximation methods such as Variational Inference are used to approximate the posterior distribution, which is not analytically tractable in deep neural networks [45].

2.2. Artificial Neural Networks (ANN)

The goal of ANN is to enable machines to learn tasks such as classification and numerical regression [35]. The activation function transforms the linear output into a nonlinear representation, which is mathematically illustrated in Figure 1:

$$z = \sum_{i=1}^{n} w_i x_i + b$$

Then, z is fed into the activation function:

$$y = g(z)$$

where x_i represents the inputs, w_i the weights, b the bias, g the activation function, and y the outputs. Some of the most common activation functions are: The sigmoid function maps its input to the range $[0, 1]$ [36]. Its mathematical form is:

$$f(x) = \sigma(x) = \frac{1}{1 + \exp(-x)}$$

The Tanh function maps its input to the range $[-1, 1]$ [36]. Its mathematical form is:

$$\tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$$

ReLU (Rectified Linear Unit): It outputs zero for negative inputs and retains positive values [36, 4].

$$\mathrm{ReLU}(x) = \max(0, x), \qquad 0 \leq \mathrm{ReLU}(x) < \infty$$

Swish: It is a smooth, non-monotonic activation function that allows small negative values while remaining unbounded above [25]. Its mathematical function is:

$$f(x) = x\,\sigma(x) = \frac{x}{1 + \exp(-x)}$$

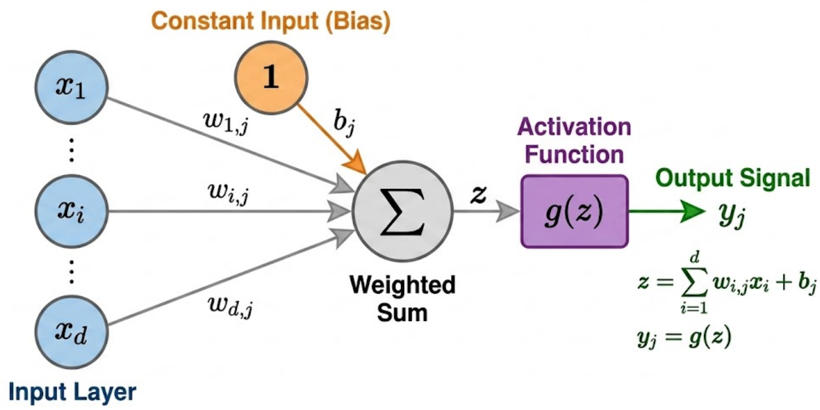

Figure 1. The basic mathematical operations within a single artificial neuron.

Neural networks are organized into interconnected layers [45], as shown in Figure 2, where the loss/error function compares the predicted outputs $\hat{y}$ with the actual targets y. The Mean Squared Error (MSE) is commonly used for regression problems. The Backpropagation algorithm works backward from the output layer to the input layer to calculate the error gradient for each weight and bias [51]. The process of updating the weights is repeated over several training cycles (epochs) until the error is minimized.

Despite the success of conventional neural networks, they treat weights as deterministic quantities, which limits their ability to represent uncertainty. Therefore, Bayesian neural networks emerged, representing weights as probability distributions.

2.3. Motivation for adopting Bayesian Neural Networks – BNNs

The significance of Bayesian Neural Networks (BNNs) is that they can estimate the uncertainty in both the model and its predictions. There are two types of uncertainty [19]: epistemic, which arises from a lack of knowledge or data, and is represented by $P(\theta \mid D)$. This type can be reduced by collecting additional data, as the initial belief about the parameters $P(\theta)$ is updated to an a posteriori belief after data input. This is important in studies with limited data sets. The other type is aleatoric, inherent, or random uncertainty in the data itself (e.g., measurement noise). This type of uncertainty does not go away with more data and is usually modeled by the likelihood function $P(y \mid \theta)$. Bayesian Neural Networks handle uncertainty in both estimated parameters and predictions. These methods have achieved success and widespread acceptance in fields where error estimation and uncertainty are critical, such as industrial and engineering applications, and medical and healthcare applications. However, despite these advantages, training Bayesian neural networks is computationally challenging.

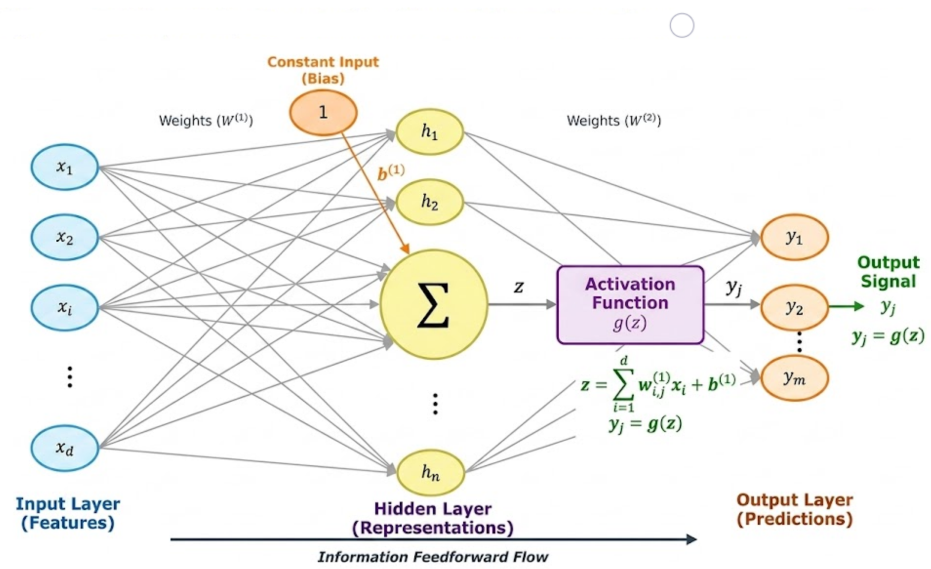

Figure 2. A fully connected feedforward neural network with multiple layers showing information flow and activation function.

2.4. Bayesian Neural Networks (BNNs)

In traditional neural networks (ANNs), the problem of approximating the function relating inputs to outputs is approached from a strictly deterministic perspective. This means the network seeks a fixed, specific value for each weight and parameter (the coefficient vector). However, when moving to the Bayesian perspective (BNNs), we do not estimate a single, fixed value for the coefficients θ, but rather estimate the posterior distribution of the coefficients given the available data. This is denoted by $p(\theta \mid D_x, D_y)$ [40, 45]. To calculate the posterior distribution, the following equation is used:

$$p(\theta \mid D_x, D_y) = \frac{p(D_y \mid D_x, \theta)\, p(\theta)}{\int_{\Theta} p(D_y \mid D_x, \theta')\, p(\theta')\, d\theta'}$$

where D is the dataset and consists of the input matrix $D_x = \{x_i\}_{i=1}^N$ and the corresponding output vector $D_y = \{y_i\}_{i=1}^N$. Thus, $D = \{(x_i, y_i)\}_{i=1}^N$, where N is the sample size. θ represents the neural network parameters. In a traditional ANN, these values are constant after training. In a BNN, however, they are not constant numbers but rather probabilistic variables with distributions. $p(\theta)$ is the prior distribution, representing the belief or prior knowledge about the parameter values before seeing the current data. The likelihood function $p(D_y \mid D_x, \theta)$ expresses the probability of obtaining the output D_y based on the input D_x and the current values of the coefficients θ during the forward pass process. The network transforms the inputs into predictions, and these outputs follow a specified probability distribution, such as the Gaussian likelihood in regression problems, or the binomial likelihood in classification problems. The denominator $\int_{\Theta} p(D_y \mid D_x, \theta')\, p(\theta')\, d\theta'$ is called the Marginal Likelihood. It acts as a normalization factor ensuring that the posterior distribution integrates to 1. Since directly calculating this integral is computationally intractable because neural networks contain thousands or millions of parameters, BNNs use well-known approximation techniques such as Monte Carlo methods, Variational Inference, MCMC, and Monte Carlo Dropout. Furthermore, the prior distribution must be defined—this is a key difference. In BNNs, the model designer must define the parametric form of the prior distribution $p(\theta)$ over the coefficients. Without a prior distribution of weights, Bayesian inference cannot be performed.

2.5. Neutrosophic Concepts

Classical logic represents information in binary form (0 or 1) without accommodating intermediate or uncertain states. Zadeh [58] introduced fuzzy set theory. Atanassov later extended this concept [12] by introducing membership and non-membership degrees to better address ambiguity. The neutrosophic framework was developed by the philosopher and mathematician Florentin Smarandache [54], who treated indeterminacy as an independent component alongside truth and falsity. The neutrosophic framework includes the Truth membership function $T_A(x)$, which measures the degree of truth; the Indeterminacy membership function $I_A(x)$, which measures the degree of doubt, neutrality, or uncertainty; and the Falsity membership function $F_A(x)$, which measures the degree of falsity in the information [6].

3. Methodology

3.1. Study Area

The study area extends from northern Baghdad to the Iraq–Turkey border. It is bounded by the lower-left coordinates $(33.41°\mathrm{N},\ 41.35°\mathrm{E})$ and the upper-right coordinates $(37.24°\mathrm{N},\ 45.91°\mathrm{E})$ (Figure 3).

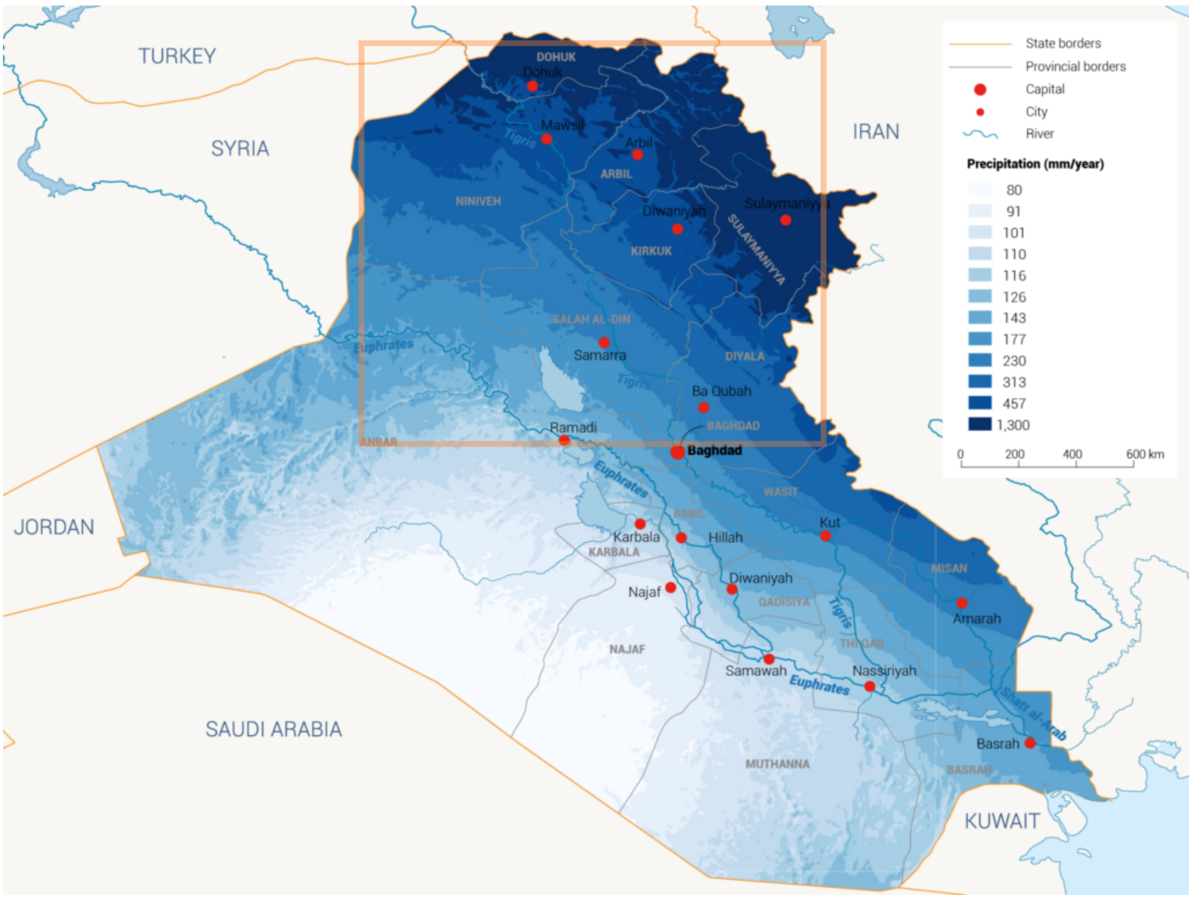

Figure 3. Study area in northern Iraq. Adapted from Fanack Water [28].

3.2. *Dataset Composition*

The data were obtained from the NASA POWER MERRA-2 model (Table 1). The target variable (**PRECTOT**) represents monthly precipitation, while the remaining variables were used as predictor variables.

Table 1. Variables used in the study dataset obtained from NASA POWER.

Variable	**Unit**	**Abbreviation**
Precipitation	mm	PRECTOT
Temperature at 2 meters	°C	T2M
Specific humidity at 2 meters	g/kg	QV2M
Surface soil wetness	Unitless	GWETTOP
Wind speed at 2 meters	m/s	WS2M

3.3. *Data Preprocessing*

Regional monthly precipitation was calculated using a cosine-latitude-weighted average across all grid cells to account for differences in area.

$$\frac{\sum_{i=1}^{N} \cos(\mathrm{Lat}_i)\,\mathrm{Value}_i}{\sum_{i=1}^{N} \cos(\mathrm{Lat}_i)}$$

Data preprocessing involves transforming raw time-series data into features suitable for Bayesian modeling. This preprocessing is carried out through the following basic stages:

3.4. *Data Cleaning and Feature Engineering*

Preprocessing begins with data cleaning and transformation. The data are ordered chronologically to preserve the temporal sequence between successive time steps. New features are then constructed (feature engineering) by combining physical variables. Precipitation does not depend on a single climatic factor, but often results from the simultaneous interaction of several factors. Therefore, new interaction variables were created by combining QV2M, T2M, WS2M, and GWETTOP. This aims to represent the nonlinear relationships between atmospheric and Earth surface elements by formulating complex physical interactions that mimic climatic conditions of the study area.

1. The interaction between QV2M and T2M results in a composite index that reflects the thermo–humidity state of the atmosphere. The feature was calculated as:

$$\mathrm{QV2M\_T2M} = \mathrm{QV2M} \times \mathrm{T2M}$$

Therefore, if QV2M is high and T2M conditions are suitable, this may be more conducive to water vapor condensation and cloud formation.
2. The interaction between QV2M and GWETTOP results in a composite index that reflects land–atmosphere moisture interactions. The feature was calculated as:

$$\mathrm{QV2M\_GWETTOP} = \mathrm{QV2M} \times \mathrm{GWETTOP}$$

GWETTOP can support evaporation, increasing the amount of water vapor available in the near-surface layers.
3. The interaction between T2M and GWETTOP results in a composite index that reflects soil thermal conditions:

$$\mathrm{T2M\_GWETTOP} = \mathrm{T2M} \times \mathrm{GWETTOP}$$

When the soil is moist and the temperature is suitable, evaporation from the Earth's surface can increase, raising humidity in the lower atmosphere.

4. The interaction between WS2M and QV2M yields a composite index that reflects moisture transport conditions.
$$\mathrm{WS2M\_QV2M} = \mathrm{WS2M} \times \mathrm{QV2M}$$
Higher WS2M and QV2M values may indicate the transport of moist air, which can support precipitation.
5. The interaction of WS2M and T2M is captured by the composite index, which reflects the atmospheric mixing condition:
$$\mathrm{WS2M\_T2M} = \mathrm{WS2M} \times \mathrm{T2M}$$
Higher T2M and WS2M often contribute to greater surface moisture loss, and in some cases, this can decrease the probability of precipitation.
6. The ratio of QV2M to T2M represents an index of the moisture-temperature balance:
$$\mathrm{Moisture\_Thermal\_Index} = \frac{\mathrm{QV2M}}{\mathrm{T2M} + 10^{-6}}$$
The small value 10^{-6} in the denominator is to avoid division by zero. The more humid the air is relative to the temperature, the more supportive the conditions are for condensation and precipitation.
7. Evaporation proxy: The evaporation proxy provides a composite index that represents the potential for evaporation:
$$\mathrm{Evapo\_Proxy} = \mathrm{WS2M} \times (\mathrm{T2M} + 273.15) \times (1 - \mathrm{GWETTOP})$$
where 273.15 is the constant for converting Celsius to Kelvin. This index is not a direct measure of actual evaporation but an approximate proxy for identifying conditions that promote moisture loss.
8. The Humidity–Soil Balance Index represents a composite index of soil-atmosphere moisture balance:
$$\mathrm{Humidity\_Soil\_Balance} = \frac{\mathrm{QV2M}}{1 + |\mathrm{GWETTOP} - \mathrm{QV2M}|}$$
When the difference between atmospheric humidity and GWETTOP is small, the index value is relatively high. If the difference is large, the denominator increases and the index value decreases.

3.5. Treatment of Missing Values

Rows containing missing values (NaNs) introduced during lag generation were removed before model training to avoid invalid loss values and parameter updates. This ensures that only complete observations are supplied to the Bayesian neural network. The lag operation is expressed as $\mathrm{Lag}_k(X_t) = X_{t-k}$.

3.6. Neutrosophic Normalization Using Min–Max Scaling

The weather variables were converted into the 3-D neutrosophic space by calculating the truth (T), indeterminacy (I), and falsity (F) components of each observation. In this representation, the truth component describes climatic conditions supporting precipitation, the falsity component reflects opposing conditions, and the indeterminacy component captures discordance among the climatic variables. Thus, the three-dimensional neutrosophic representation explicitly represents climatic uncertainty prior to Bayesian neural network training. For the Min–Max normalized data, the neutrosophic components were calculated as follows:

$$T_n = \alpha \times QV2M + \beta \times GWETTOP + \gamma \times (1 - T2M) + \delta \times (1 - WS2M)$$

$$I_n = \Sigma \times |WS2M - QV2M| + \phi \times |T2M - GWETTOP| + \lambda \times |QV2M - GWETTOP|$$

$$F_n = \Omega \times T2M + \theta \times (1 - GWETTOP) + \Psi \times WS2M$$

$$N_{\mathrm{Score}} = \frac{2 + T_n - I_n - F_n}{3}$$

The score can equivalently be expressed as

$$N_{\text{Score}} = \frac{T_n + (1 - I_n) + (1 - F_n)}{3}.$$

This representation shows that the score is the arithmetic mean of the truth degree and the complements of the indeterminacy and falsity degrees. Since all three terms lie within $[0, 1]$, the resulting score is also bounded within $[0, 1]$, with higher values indicating greater truth and lower indeterminacy and falsity.

Ten candidate weight configurations were sampled from symmetric Dirichlet distributions and evaluated across four temporal windows $k \in \{2, 3, 5, 7\}$. The best configuration was selected primarily according to R^2, with RMSE and MAE as secondary criteria.

3.7. Statistical Standardization Using StandardScaler

Before model training, StandardScaler is used to standardize the features according to:

$$z = \frac{x - \mu}{\sigma} \tag{1}$$

where μ and σ denote the mean and standard deviation, respectively; all features (physical, neutrosophic, or complex interactions) are transformed to have a mean of 0 and a standard deviation of 1. The scaler was fitted on the training data and applied to the test data to prevent data leakage. This standardization improves numerical stability and facilitates convergence in DenseFlipout Bayesian layers.

3.8. Time Windowing

The tabular time series was transformed into causal temporal windows, such that each forecasting instance was constructed exclusively from observations preceding the target time:

$$X_t = [x_{t-k}, x_{t-k+1}, \ldots, x_{t-1}],$$

where k represents the temporal-window length and y_t denotes the target precipitation value at time t.

To characterize short- and medium-term temporal dependencies in monthly precipitation, four candidate window lengths, $k \in \{2, 3, 5, 7\}$ months, were predefined for evaluation. The data were chronologically partitioned into 80% training and 20% testing subsets. Each candidate window was evaluated under identical preprocessing and training settings. Selection was based primarily on the highest R^2, with RMSE and MAE used as supporting error criteria. Among the evaluated candidates, the two-month window achieved the best overall performance and was therefore selected for the final model. The selected model was then retrained using the optimal window and evaluated on the held-out 20% test subset.

4. Hybrid Modeling by Combining Neutrosophic Logic and Bayesian Deep Learning

Figure 4 presents the overall workflow of the proposed hybrid framework. The methodology begins with data loading and preprocessing, followed by neutrosophic feature engineering and the construction of temporal windows. The processed data are then used to train the Bayesian neural network, after which hyperparameter optimization, ensemble retraining, and probabilistic evaluation are performed to generate the final precipitation forecasts.

This study combines neutrosophic logic and Bayesian neural networks into a hybrid forecasting framework. Neutrosophic logic is used to represent uncertainty in climate data, while Bayesian neural networks provide probabilistic forecasting capabilities.

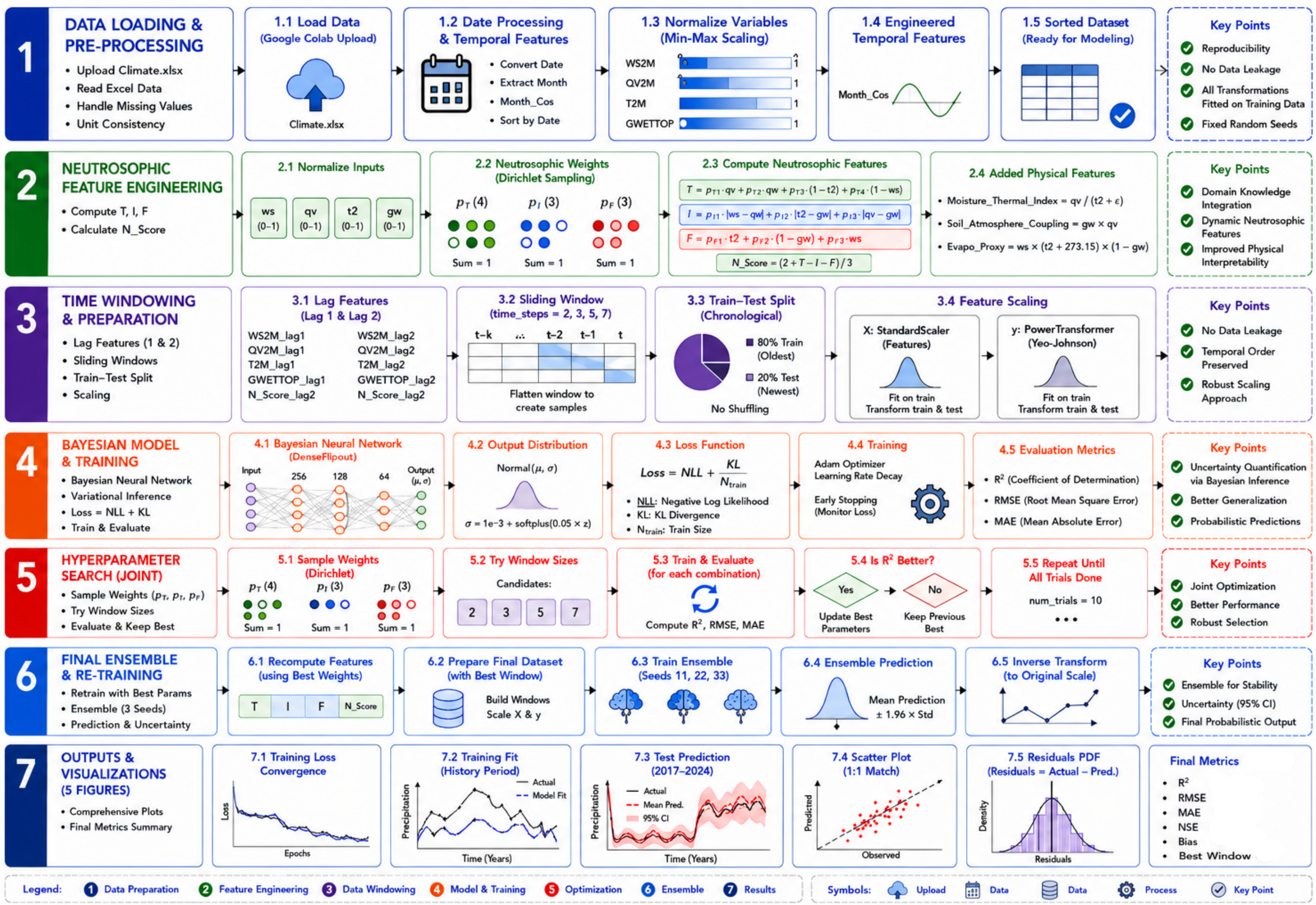

Figure 4. Overall workflow of the proposed hybrid neutrosophic–Bayesian deep learning framework for precipitation forecasting.

4.1. Modeling Using Variational Inference Layers

The BNN structure was built using neutrosophic inputs, while probability distributions rather than deterministic values were assigned to the network weights. The Bayesian setting was employed to extend the forecasting procedure from simple deterministic point estimation to a probabilistic statistical inference framework. Whereas the uncertainty in the input was described using neutrosophic logic, the uncertainty in the model's weights was described using probabilistic layers (Flipout).

The DenseFlipout layers were used to implement variational inference over the network weights within a Bayesian setting. Therefore, the model can represent predictive distributions and improve generalization. This probabilistic approach can improve the accuracy of monthly precipitation predictions and produce probabilistic forecasts. The kernel weights of each DenseFlipout layer were assigned independent standard normal priors, $p(w) = \mathcal{N}(0, 1)$, while the variational posterior was modeled as a fully factorized Gaussian distribution:

$$q(w \mid \theta) = \mathcal{N}(\mu, \sigma^2).$$

The bias terms were treated as deterministic.

The progressively decreasing DenseFlipout architecture (256–128–64) captures complex nonlinear relationships and progressively condenses the learned representation. Swish was used because its smooth, differentiable form supports stable optimization.

4.2. Training Stability and Normalization

Layer normalization was applied after each hidden layer to improve training stability. This is important in Bayesian models because stochastic weight sampling can introduce variability during optimization.

4.3. Probabilistic Output Modeling

The final layer of the model is designed to generate nonlinear probabilistic outputs that characterize heteroscedastic aleatoric uncertainty. Unlike traditional models that predict a single point value, this model dynamically estimates two statistical parameters for each monthly sample:

$$y \sim \mathcal{N}(\mu, \sigma^2)$$

The mean μ represents the most probable precipitation value based on the input climatic data. The standard deviation σ is adjusted via an activation function to ensure positive values. This parameter represents the degree of dispersion, or aleatoric observation-level uncertainty, in the weather conditions for each month. By modeling these two parameters within a normal distribution, the model produces probabilistic rather than deterministic forecasts. This enables the estimation of both expected precipitation and its associated uncertainty, offering a more reliable assessment of forecast performance under changing climatic conditions.

In the final layer, the output corresponding to the standard deviation σ is passed through the Softplus function:

$$f(x) = \ln(1 + e^x)$$

This ensures that the standard deviation remains strictly positive and avoids numerical errors during the calculation of the negative log-likelihood (NLL).

4.4. Loss Function and Bayesian Inference

Model training was guided by a composite loss function that balances accurate prediction with effective regularization. To achieve this, the optimization process uses two fundamental loss components:

Negative Log-Likelihood (NLL): This component evaluates how well the predicted probability distribution aligns with the observed precipitation data. Minimizing the NLL encourages the model to generate accurate estimates of both the expected precipitation and its associated uncertainty.

$$\mathrm{NLL} = -\log p(y \mid x, w)$$

$$L = \mathrm{NLL} + \frac{1}{N}\,\mathrm{KL}(q(w)\,\|\,p(w))$$

where N is the training sample size. Kullback–Leibler (KL) Divergence: This term acts as a Bayesian regularizer by encouraging the learned weight distributions to remain close to their prior distributions. This helps prevent overfitting while maintaining a balance between prior assumptions and information extracted from the data.

This balance enables the model to capture underlying climate patterns rather than simply fitting the observed data, leading to better generalization and more reliable predictions under uncertainty.

4.5. Decomposition and Integration of Total Uncertainty

Predictive uncertainty was separated into two main components:

4.5.1. Aleatoric Uncertainty This component reflects inherent variability and observation noise in the data. It was estimated using the average predicted standard deviation across the three models, and it cannot be reduced by collecting more data.

4.5.2. Epistemic Uncertainty It was estimated from the variability among the predictions produced by the three models. This type reflects the model's limited knowledge about rare or uncommon climatic patterns in the data. These two components were combined in the final stage using the variance-addition formula,

$$\sigma_{\text{total}} = \sqrt{\sigma^2_{\text{aleatoric}} + \sigma^2_{\text{epistemic}}},$$

where $\sigma^2_{\text{aleatoric}}$ and $\sigma^2_{\text{epistemic}}$ denote the aleatoric and epistemic variances, respectively. This combination enabled the construction of a 95% predictive interval.

4.6. Model Selection and Hyperparameter Tuning

The time-window size and neutrosophic weights were jointly selected based on the coefficient of determination R^2, with root mean square error (RMSE) and mean absolute error (MAE) used as supporting criteria. Specifically, the neutrosophic weight vectors were generated using Dirichlet-based randomized sampling, which ensured non-negative coefficients that summed to one within each of the truth, indeterminacy, and falsity functions. The search comprised ten randomized trials, within which the temporal window parameters $k \in \{2, 3, 5, 7\}$ were examined. Each candidate configuration was trained for 120 epochs and evaluated using R^2, RMSE, and MAE. The configuration with the highest R^2 was retained, while RMSE and MAE were used as supporting error measures. The selected configuration, corresponding to $k = 2$, was subsequently retrained for 400 epochs using three independently initialized Bayesian models with random seeds 11, 22, and 33. The final neutrosophic weights are reported in Table 4.

4.7. Ensemble Post-processing

The process combines the predictive means and predictive standard deviations from three independently initialized Bayesian models into a single probabilistic representation.

4.8. Optimization Strategy

The Adam optimizer was used with an initial learning rate of 0.002. Training was conducted for 120 epochs during the hyperparameter search and for 400 epochs during the retraining of the final model, allowing sufficient exploration of the Bayesian loss landscape before convergence.

4.9. Generalization Testing

Model performance was evaluated using a temporally separated final test set (20%) that was excluded from training and model selection. This test set provided an independent basis for evaluating the Bayesian neutrosophic model's ability to reproduce observed precipitation behavior. The baseline BNN was selected as the primary comparator because it preserves the same probabilistic architecture and training configuration, allowing the incremental contribution of the neutrosophic component to be isolated without introducing architectural differences.

5. Results and Discussion

5.1. Descriptive and Exploratory Analysis of Precipitation and Climatic Variables

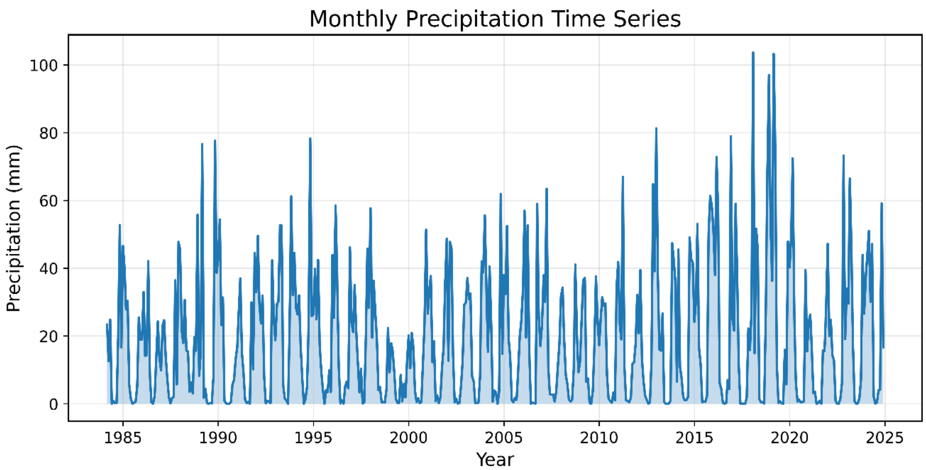

Figure 5. Monthly Precipitation in Northern Iraq, 1984–2024.

Figure 5 clearly shows seasonal fluctuations, with recurring peaks and troughs each year. For the period 1984–2010, the peaks were generally lower. The period 2015–2024 recorded the highest peaks, exceeding 100 mm. Thus, the region exhibits a seasonal pattern together with substantial variability in precipitation amounts. The figure shows numerous zero values, indicating dry periods. The series also appears to exhibit changing variability over time, suggesting possible heteroscedastic behavior. These characteristics motivate the use of the proposed hybrid neutrosophic–Bayesian framework.

From Figure 6, it is evident that the distribution of monthly precipitation is positively skewed. This indicates that most months are characterized by low or zero precipitation. The long right tail extends to higher values, such as 80 or 100 mm, representing fewer months with relatively high precipitation values. It is also evident that half of the months recorded less than 13.76 mm of precipitation. The arithmetic mean (19.24) is greater than the median (13.76), confirming positive skewness.

Figure 7 illustrates a clear seasonal pattern consisting of three periods. The rainy season extends from January to April, with March being the wettest month on average. During the dry season from May to September, precipitation declines beginning in May and approaches zero from June to September. Precipitation returns from October to December, increasing gradually in October and more sharply in November and December to levels above 30 mm. Overall, precipitation is markedly higher during the winter months.

Despite recurring droughts, total annual precipitation has fluctuated considerably over the past 40 years, with a gradual upward trend (Figure 8). However, the peaks in recent years (2015-2024) have been higher than those of the 1980s. In 1999, total precipitation was below 100 mm, representing the driest year in the series. 2018 was an exceptionally wet year, with the total exceeding 430 mm.

Figure 9 shows the correlation matrix for five climatic variables. It is observed that increased precipitation is associated with higher surface GWETTOP (0.77). The figure also shows an inverse relationship between T2M and surface GWETTOP (-0.88). Likewise, there is an inverse relationship between precipitation and T2M (-0.66).

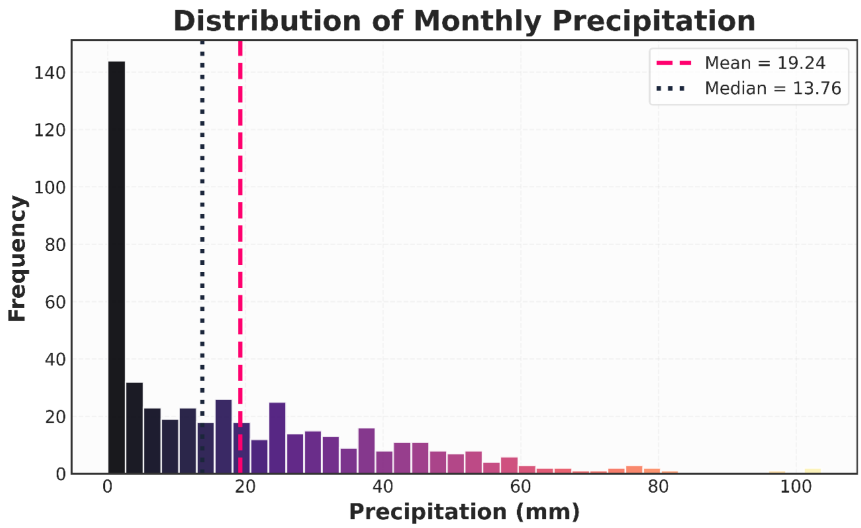

Figure 6. Frequency Distribution of Monthly Precipitation.

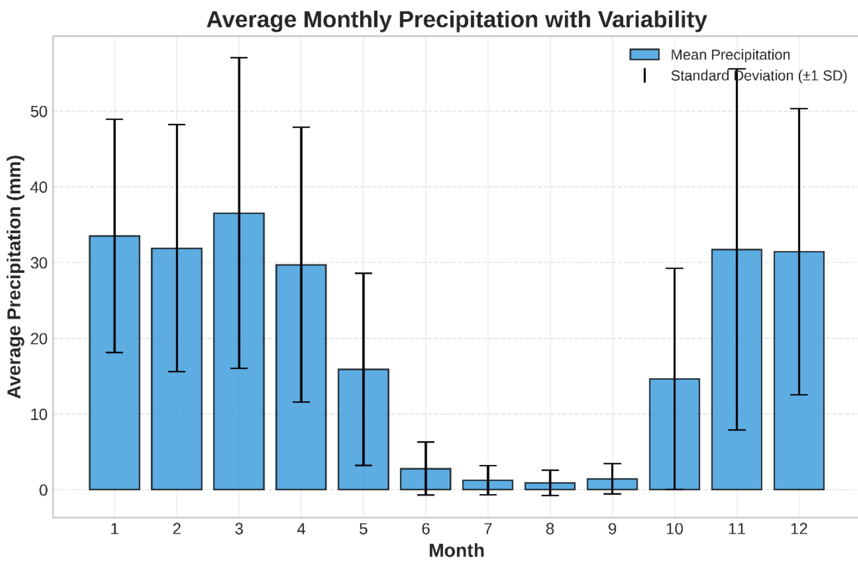

Figure 7. Seasonal Variation in Mean Monthly Precipitation.

Scatter plots were also examined to better understand the relationship between main climatic variables and precipitation. The overall trend, as shown in Figure 10, indicates a decrease in precipitation with increasing T2M, with the most intense rainfall events occurring mainly at low to moderate temperatures. On the other hand, precipitation tends to increase with QV2M, as shown in Figure 11. However, high humidity alone is not sufficient to produce extreme rainfall, suggesting that other atmospheric conditions are also involved. Figure 12 shows that low values of GWETTOP are generally associated with low or no precipitation, while higher soil moisture is associated with higher variability and more extreme events. Lastly, Figure 13 indicates a weaker connection between WS2M and precipitation, with no apparent linear trend in the data.

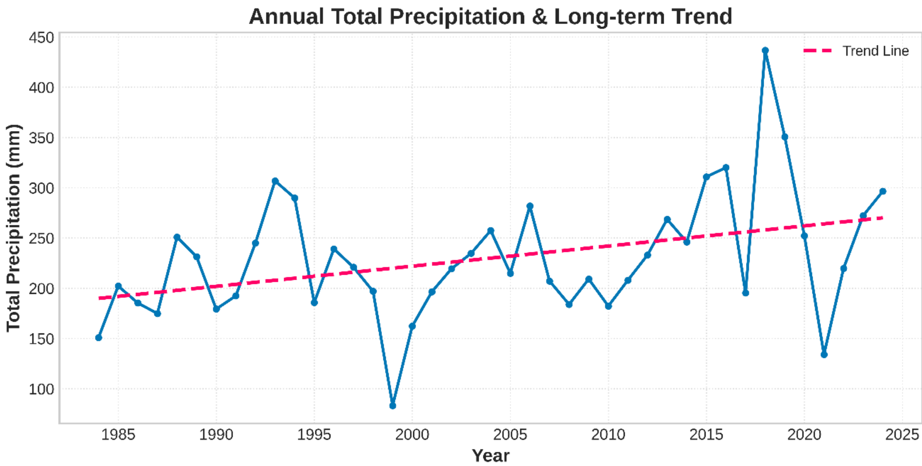

Figure 8. General Trend of Annual Precipitation, 1984–2024.

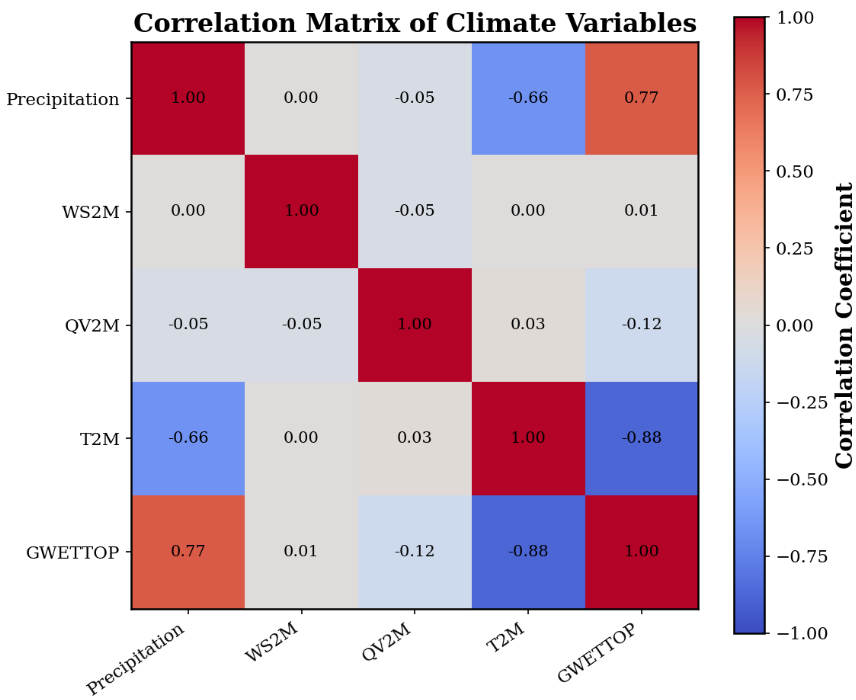

Figure 9. Correlation Matrix for Climatic and Hydrological Variables.

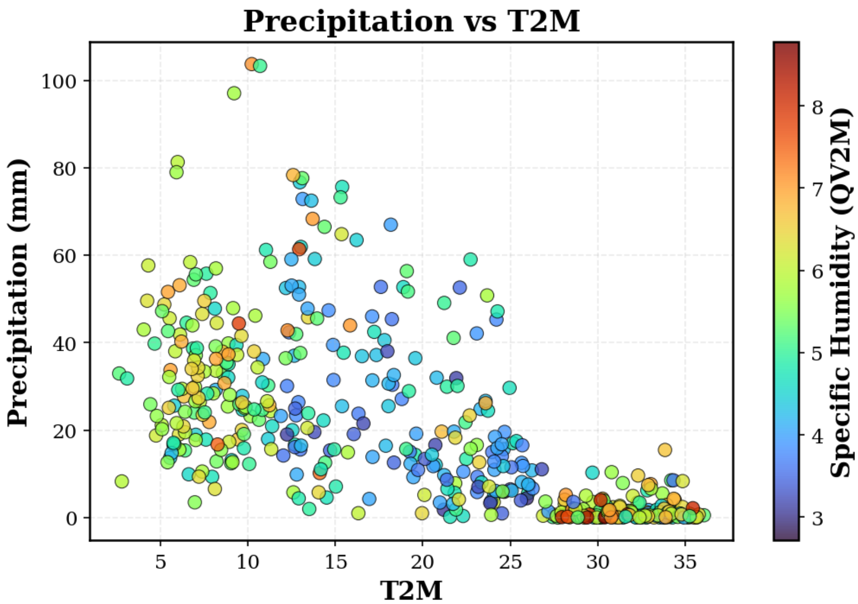

Figure 10. The relationship between T2M and precipitation.

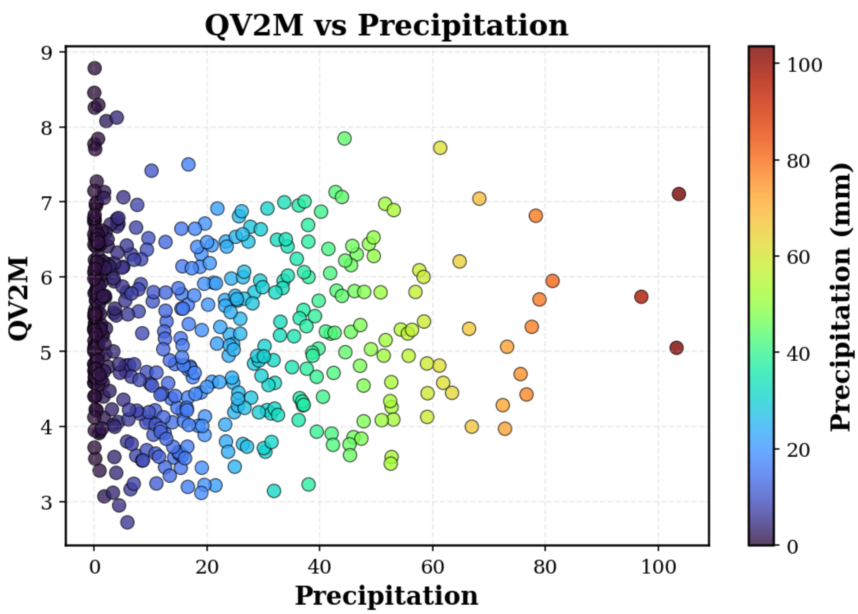

Figure 11. Relationship between QV2M and Precipitation.

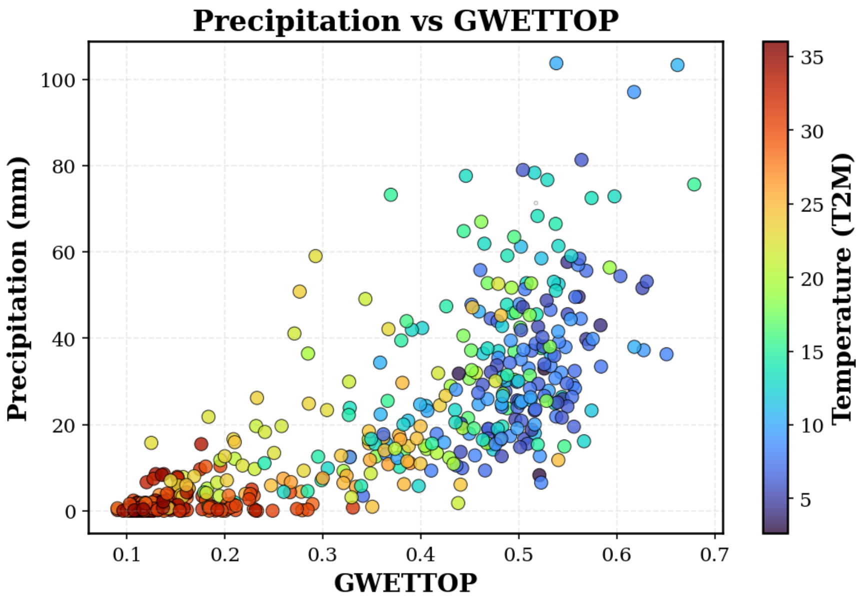

Figure 12. The relationship between GWETTOP and precipitation.

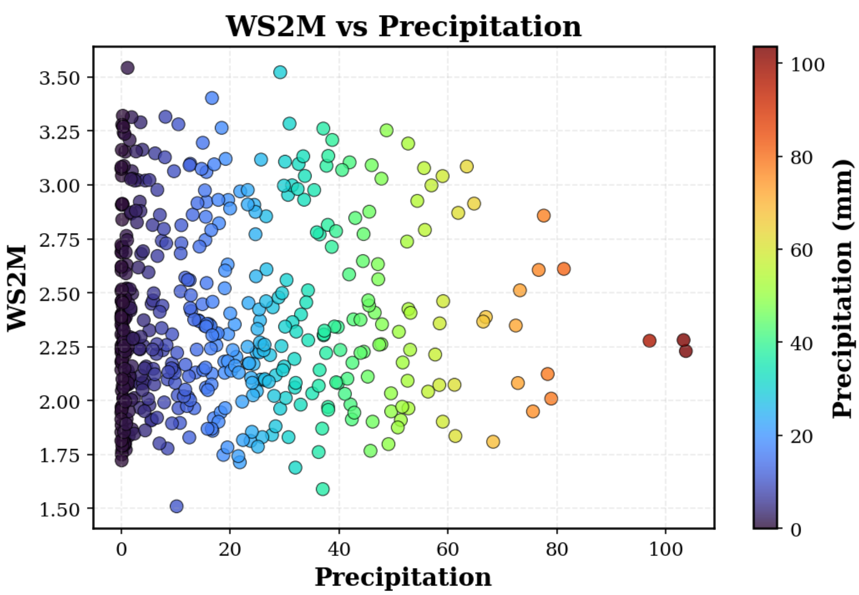

Figure 13. The relationship between WS2M and precipitation.

5.2. Predictive Performance of the Proposed Model

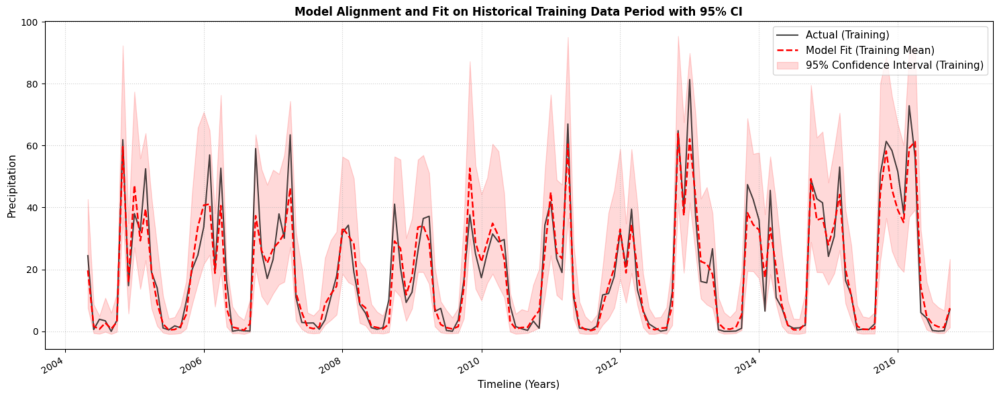

Figure 14. Observed and fitted precipitation values for a selected segment (2004–2016) of the training period using the proposed Neutro-BNN model.

Figure 14 illustrates the agreement between observed and fitted precipitation values over a selected segment (2004–2016) of the training period. The fitted series captures the main seasonal variations, including the pronounced dry periods in 2005, 2008, 2011, and 2015, and tracks the timing of precipitation peaks without a noticeable temporal lag. This agreement indicates that the model learned the dominant temporal patterns in the training data.

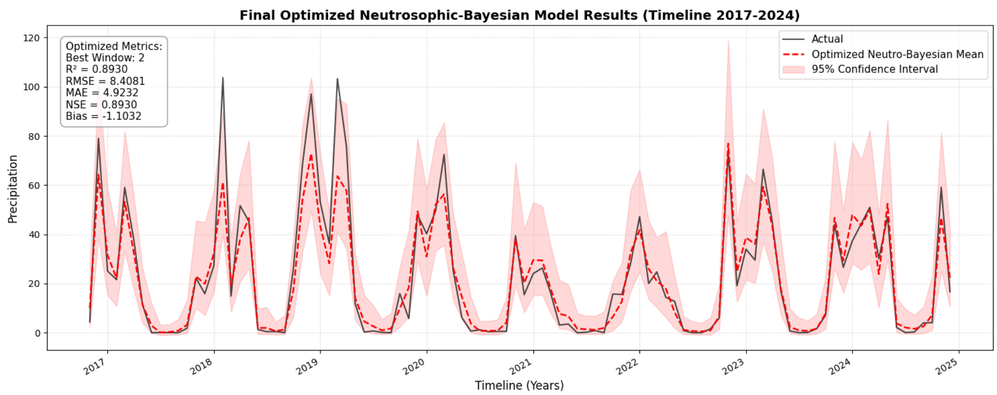

Figure 15. Observed and predicted precipitation during the test period from 2017 to the end of 2024 using the proposed Neutro-BNN model.

Figure 15 presents the out-of-sample predictive performance of the proposed model over the chronologically separated test period from 2017 to the end of 2024. The model explains approximately 89.30% of the variance in observed precipitation and tracks the main seasonal cycles without a noticeable temporal lag. The corresponding RMSE and MAE were 8.4081 and 4.9232, respectively. The 95% predictive intervals are narrower during dry periods and wider around high-precipitation peaks, indicating greater predictive uncertainty during extreme rainfall events.

Figure 16 evaluates the agreement between observed and predicted precipitation using the Neutro-BNN model. Most points are concentrated around the 1:1 line, indicating strong agreement between observed and predicted values. The model performs particularly well for low to moderate precipitation amounts. At the same time, some underestimation is observed for extreme rainfall events, which are naturally less frequent and more difficult to predict accurately. High-precipitation events, defined as observed monthly values exceeding 80 mm, occurred in only 3 of the 98 test observations (3.1%), and all three were underestimated. This pattern reflects the limited representation of extreme events in the monthly series. Future extensions could examine tail-weighted loss functions or dedicated modeling of extreme precipitation events.

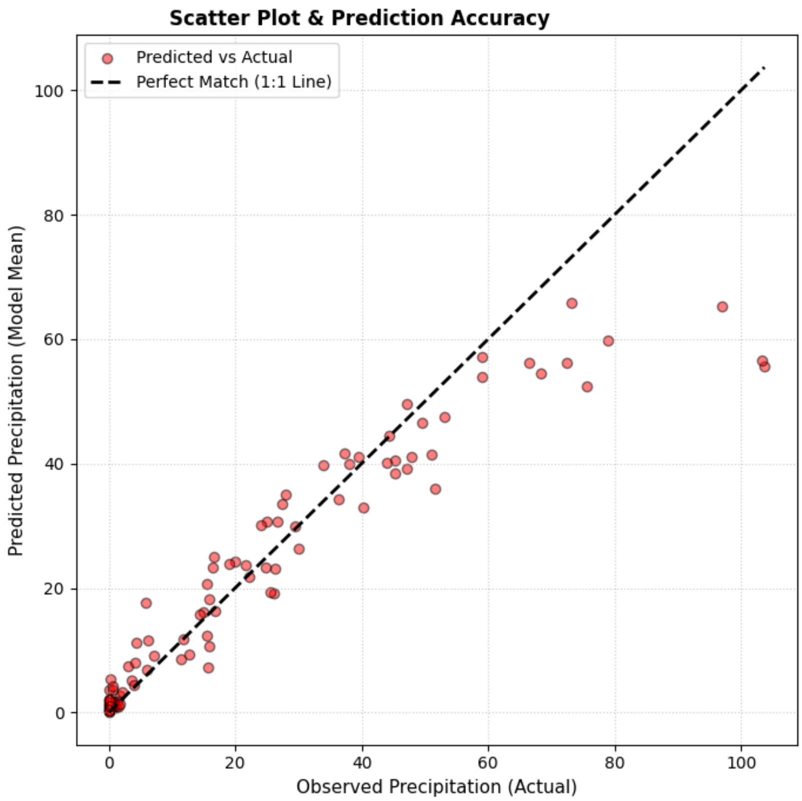

Figure 16. Comparison of observed and predicted precipitation values using the proposed Neutro-BNN model.

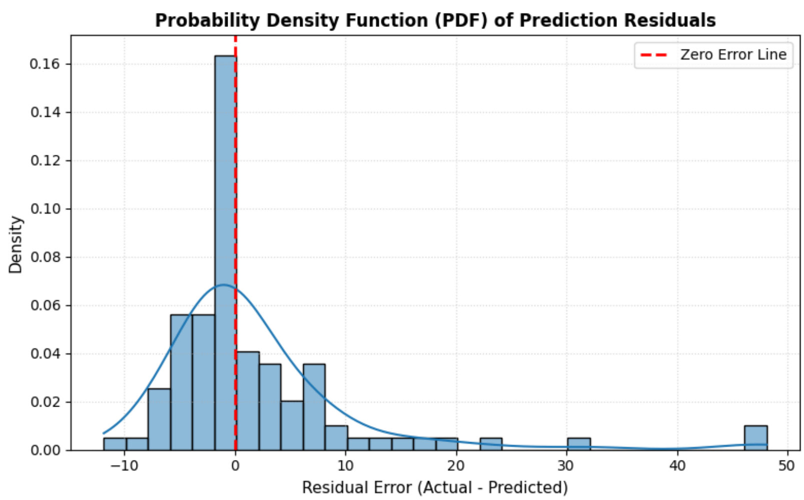

Figure 17. Probability Density Function and Histogram of Residuals.

Figure 17 shows the residual distribution used to assess the predictive performance of the proposed framework. Most residuals are concentrated around zero, indicating small prediction errors for most observations. The distribution is positively skewed, with a small number of positive residuals extending from approximately 10 to 50. Because the residuals were defined as observed minus predicted values, these positive residuals indicate underprediction during rare high-precipitation events.

Despite this rightward tail, most residuals lie between -10 and $+10$ and are concentrated around zero, indicating relatively small errors for most observations. The larger positive residuals are mainly associated with the high-precipitation events discussed above.

5.3. *Targeted Component-Ablation Analysis*

Table 2 presents the targeted component-ablation results for the baseline BNN and the proposed Neutro-BNN under the same evaluation setting.

Table 2. Targeted component-ablation comparison between the baseline BNN and the proposed Neutro-BNN

Model	R^2	**RMSE**	**MAE**	**NSE**	**Bias**
BNN	0.821	10.89	6.10	0.821	-2.61
Neutro-BNN	0.893	8.41	4.92	0.893	-1.10

Compared with the baseline BNN, the proposed Neutro-BNN increased both R^2 and NSE from 0.821 to 0.893, while reducing RMSE from 10.89 to 8.41, MAE from 6.10 to 4.92, and the absolute bias from 2.61 to 1.10. These changes correspond to reductions of 22.77% in RMSE, 19.34% in MAE, and 57.85% in absolute bias. These results represent observed out-of-sample performance improvements relative to the corresponding baseline BNN over the chronologically separated test period. They should not be interpreted as a comprehensive ranking against all contemporary forecasting architectures.

5.4. *Final Model Configuration and Neutrosophic Weights*

Table 3 summarizes the final architecture, training parameters, data configuration, and preprocessing settings adopted for the Bayesian ensemble model.

Table 3. Detailed Architecture and Final Specifications of the Bayesian Ensemble Model

Model Component / Training Parameter	**Final Adopted Configuration**
Hidden Layers Architecture	DenseFlipout(256) $\rightarrow$ DenseFlipout(128) $\rightarrow$ DenseFlipout(64)
Activation Function	Swish
Output Distribution	Normal Distribution Parameterized by μ and σ
Ensemble Size	3 Models
Ensemble Random Seeds	(11, 22, 33)
Initial Learning Rate	0.002
LR Decay	Exponential Decay (Rate: 0.9 per 1000 steps)
Final Training Epochs	400
Final Batch Size	32
Optimizer	Adam Optimizer
Loss Function	Negative Log-Likelihood (NLL) + KL Divergence
Training Samples	390 Samples
Testing Samples	98 Samples
Input Shape	60 features (Flattened Window)
Feature Scaling	StandardScaler
Target Transformation	Power Transformer (Yeo–Johnson)

Table 4 presents the final weights of the neutrosophic functions (T, I, and F) obtained through the optimization process. T2M received the highest weight in the T function, whereas the difference between QV2M and GWETTOP was the main contributor to the I function. These weights reflect the relative contribution of the climatic variables to the construction of the N_Score used in the proposed Bayesian model. In function (F), WS2M recorded the highest relative weight.

Table 4. Neutrosophic function weights used (Time Window = 2)

$T_n = 0.17 \times \mathrm{QV2M} + 0.02 \times \mathrm{GWETTOP} + 0.731 \times (1 - \mathrm{T2M}) + 0.071 \times (1 - \mathrm{WS2M})$
$I_n = 0.19 \times \lvert\mathrm{WS2M} - \mathrm{QV2M}\rvert + 0.07 \times \lvert\mathrm{T2M} - \mathrm{GWETTOP}\rvert + 0.74 \times \lvert\mathrm{QV2M} - \mathrm{GWETTOP}\rvert$
$F_n = 0.24 \times \mathrm{T2M} + 0.311 \times (1 - \mathrm{GWETTOP}) + 0.45 \times \mathrm{WS2M}$

The controlled comparison with the standard BNN isolates the contribution of the neutrosophic inputs under identical experimental settings. Nevertheless, restricting the comparative analysis to this baseline limits a broader assessment of the proposed model across different forecasting approaches. Future studies should therefore benchmark the Neutro-BNN against models such as LSTM, GRU, Random Forest, and climatological baselines using a common experimental protocol.

6. Conclusion

The results showed that integrating neutrosophic logic with Bayesian neural networks improved rainfall prediction performance compared with the traditional Bayesian model, confirming the effectiveness of utilizing representations of truth, indeterminacy, and falsity in processing climate data. The proposed model achieved improved performance metrics: R^2 and NSE increased, whereas RMSE, MAE, and the absolute bias decreased, reflecting improved predictive accuracy and reduced errors and bias. The model showed a good capability to reproduce general rainfall patterns and performed well in both dry and average rainfall periods. However, the rarity of extreme rainfall events made them more difficult to predict. The confidence intervals obtained by the Neutro-BNN model suggest that such uncertainty is condition-dependent, being tighter during stable periods and wider in the presence of extreme values. This is realistic behaviour, in line with the nature of rainfall data. The results suggest that integrating neutrosophic logic and Bayesian deep learning provides a promising framework to improve rainfall forecasting and assist applications related to water resource management and environmental planning in regions with high climatic variability. Future work should benchmark the proposed framework against recent deep learning and ensemble machine learning models under a unified experimental setting.

REFERENCES

1. M. Abdel-Basset, V. Chang, M. Mohamed, and F. Smarandache, "A refined approach for forecasting based on neutrosophic time series," *Symmetry*, vol. 11, no. 4, p. 457, 2019.
2. W. Ahmad, N. Ayub, T. Ali, M. Irfan, M. Awais, M. Shiraz, and A. Glowacz, "Towards short term electricity load forecasting using improved support vector machine and extreme learning machine," *Energies*, vol. 13, no. 11, p. 2907, 2020.
3. S. Al-Dahidi, M. Madhiarasan, L. Al-Ghussain, A. M. Abubaker, A. D. Ahmad, M. Alrbai, M. Aghaei, H. Alahmer, A. Alahmer, P. Baraldi, and E. Zio, "Forecasting solar photovoltaic power production: A comprehensive review and innovative data-driven modeling framework," *Energies*, vol. 17, no. 16, p. 4145, 2024.
4. F. A. Agarap, "Deep learning using rectified linear units," *arXiv preprint arXiv:1803.08375*, 2018.
5. M. Al-Hashimi, H. Hayawi, and M. Q. Y. Alawjar, "Bayesian Optimization for Enhanced Prediction of Earthquakes with Machine Learning Techniques," *Statistics, Optimization and Information Computing*, vol. 15, no. 5, pp. 4305–4322, 2026.
6. M. Al-Hashimi, H. Hayawi, and M. Alawjar, "Predicting Sunspot Numbers Based on Neutrosophic Time Series," *Statistics, Optimization & Information Computing*, vol. 15, no. 6, pp. 5431–5444, 2026.
7. M. M. Al-Hashimi, H. A. Hayawi, and M. Al-Kassab, "A comparative study of traditional methods and hybridization for predicting non-stationary sunspot time series," *International Journal of Mathematics and Computer Science*, vol. 19, no. 1, pp. 195–203, 2024.
8. M. M. Al-Hashimi and A. Hayawi, "Nonlinear model for precipitation forecasting in northern Iraq using machine learning algorithms," *International Journal of Mathematics and Computer Science*, vol. 19, no. 1, pp. 171–179, 2024.
9. M. Al-Hashimi, H. H. Alawjar, and M. Alawjar, "Ensemble method for intervention analysis to predict the water resources of the Tigris River," *Statistics, Optimization & Information Computing*, vol. 14, no. 1, pp. 144–161, 2025.

10. M. Al-Hashimi, M. Alawjar, E. M. N. Mahmood, and H. Hayawi, "An improved robust model for parameter estimation using the fuzzy transform," *Pak. J. Statist.*, vol. 42, no. 3, pp. 205–222, 2026.
11. H. Alghamdi, C. Maduabuchi, K. Okoli, A. Albaker, M. Alobaid, M. Alghassab, E. Makki, and M. Alkhedher, "Bayesian neural networks for solar power forecasts in advanced thermoelectric systems," *Case Studies in Thermal Engineering*, vol. 61, p. 104940, 2024.
12. K. T. Atanassov, *Intuitionistic Fuzzy Sets*. Heidelberg: Physica, 1999.
13. P. Bauer, A. Thorpe, and G. Brunet, "The quiet revolution of numerical weather prediction," *Nature*, vol. 525, no. 7567, pp. 47–55, 2015.
14. Y. Chen, P. Xu, Y. Chu, W. Li, Y. Wu, L. Ni, Y. Bao, and K. Wang, "Short-term electrical load forecasting using the Support Vector Regression (SVR) model to calculate the demand response baseline for office buildings," *Applied Energy*, vol. 195, pp. 659–670, 2017.
15. M. Cobos-Maestre, M. Flores-Soriano, and D. F. Barrero, "BaLNet: A probabilistic Bayesian neural network for solar wind speed forecasting," 2025.
16. F. Dadras Javan, I. A. Campodonico Avendano, B. Najafi, A. Moazami, and F. Rinaldi, "Machine-learning-based prediction of HVAC-driven load flexibility in warehouses," *Energies*, vol. 16, no. 14, p. 5407, 2023.
17. J. F. de Freitas, M. Niranjan, and A. H. Gee, "Hierarchical Bayesian models for regularization in sequential learning," *Neural Computation*, vol. 12, no. 4, pp. 933–953, 2000.
18. Z. Deng, X. Zhang, Z. Li, J. Yang, X. Lv, Q. Wu, and B. Zhu, "Probabilistic prediction of wind power based on improved Bayesian neural network," *Frontiers in Energy Research*, vol. 11, p. 1309778, 2024.
19. A. Der Kiureghian and O. Ditlevsen, "Aleatory or epistemic? Does it matter?" *Structural Safety*, vol. 31, no. 2, pp. 105–112, 2009.
20. P. Di Leo, A. Ciocia, G. Malgaroli, and F. Spertino, "Advancements and challenges in photovoltaic power forecasting: A comprehensive review," *Energies*, vol. 18, no. 8, p. 2108, 2025.
21. M. F. Dixon, I. Halperin, and P. Bilokon, *Machine Learning in Finance*, vol. 1170. New York, NY, USA: Springer International Publishing, 2020.
22. Y. Du, Y. Liu, X. Wang, J. Fang, G. Sheng, and X. Jiang, "Predicting weather-related failure risk in distribution systems using Bayesian neural network," *IEEE Transactions on Smart Grid*, vol. 12, no. 1, pp. 350–360, 2020.
23. H. Eakin, "Seasonal climate forecasting and the relevance of local knowledge," *Physical Geography*, vol. 20, no. 6, pp. 447–460, 1999.
24. E. E. Ebhuoma and D. M. Simatele, "We know our Terrain: Indigenous knowledge preferred to scientific systems of weather forecasting in the Delta State of Nigeria," *Climate and Development*, vol. 11, no. 2, pp. 112–123, 2019.
25. S. Eger, P. Youssef, and I. Gurevych, "Is it time to swish? Comparing deep learning activation functions across NLP tasks," in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pp. 4415–4424, 2018.
26. O. Ejike, D. Ndzi, and M. Z. Shakir, "Comparative study of machine learning-based rainfall prediction in tropical and temperate climates," *Climate*, vol. 13, no. 8, p. 167, 2025.
27. A. Etz, Q. F. Gronau, F. Dablander, P. A. Edelsbrunner, and B. Baribault, "How to become a Bayesian in eight easy steps: An annotated reading list," *Psychonomic Bulletin & Review*, vol. 25, no. 1, pp. 219–234, 2018.
28. Fanack Water, "Iraq Water Report," Available: https://water.fanack.com/iraq/. Accessed: Jun. 13, 2026.
29. Y. Gal and Z. Ghahramani, "Dropout as a Bayesian approximation: Representing model uncertainty in deep learning," in *Proceedings of the International Conference on Machine Learning (ICML)*, pp. 1050–1059, 2016.
30. A. Gelman, J. B. Carlin, H. S. Stern, *et al.*, *Bayesian Data Analysis*. New York, NY, USA: Chapman and Hall/CRC, 1995.
31. T. Gneiting and M. Katzfuss, "Probabilistic forecasting," *Annual Review of Statistics and Its Application*, vol. 1, no. 1, pp. 125–151, 2014.
32. A. Gutierrez-Lopez, I. Cruz-Paz, and M. Muñoz Mandujano, "Algorithm to predict the rainfall starting point as a function of atmospheric pressure, humidity, and dewpoint," *Climate*, vol. 7, no. 11, p. 131, 2019.
33. S. O. Hasoon and M. M. Al-Hashimi, "Hybrid deep neural network and long short term memory network for predicting of sunspot time series," *International Journal of Mathematics and Computer Science*, vol. 17, no. 3, pp. 955–967, 2022.
34. H. Hayawi, M. Al-Hashimi, and M. Alawjar, "Machine learning methods for modelling and predicting dust storms in Iraq," *Statistics, Optimization & Information Computing*, vol. 13, no. 3, pp. 1063–1075, 2025.
35. S. Haykin, *Neural Networks: A Comprehensive Foundation*. Upper Saddle River, NJ, USA: Prentice Hall PTR, 1994.
36. K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on ImageNet classification," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pp. 1026–1034, 2015.
37. I. Intan, A. T. Koswara, A. Amirah, S. Wahyuni, N. Nurdin, S. Salman, and I. Syarif, "Weather forecasting analysis using Bayesian regularization algorithms," *International Journal of Informatics and Computation*, vol. 3, no. 2, pp. 1–14, 2021.
38. M. A. Iqbal, J. M. Gil, and S. K. Kim, "Attention-driven hybrid ensemble approach with Bayesian optimization for accurate energy forecasting in Jeju Island's renewable energy system," *IEEE Access*, vol. 13, pp. 7986–8010, 2025.
39. X. Jin, C. Dai, Y. Tao, and F. G. Zhang, "A comparative study of RNN, CNN, LSTM and GRU for random sea wave forecasting," *Ocean Engineering*, vol. 338, p. 122013, 2025.
40. L. V. Jospin, H. Laga, F. Boussaid, W. Buntine, and M. Bennamoun, "Hands-on Bayesian neural networks—A tutorial for deep learning users," *IEEE Computational Intelligence Magazine*, vol. 17, no. 2, pp. 29–48, 2022.
41. P. Kankanala, S. Das, and A. Pahwa, "AdaBoost^{+}: An ensemble learning approach for estimating weather-related outages in distribution systems," *IEEE Transactions on Power Systems*, vol. 29, no. 1, pp. 359–367, 2013.
42. A. A. Khan, O. Chaudhari, and R. Chandra, "A review of ensemble learning and data augmentation models for class imbalanced problems: Combination, implementation and evaluation," *Expert Systems with Applications*, vol. 244, p. 122778, 2024.
43. X. H. Le, H. V. Ho, G. Lee, and S. Jung, "Application of long short-term memory (LSTM) neural network for flood forecasting," *Water*, vol. 11, no. 7, p. 1387, 2019.
44. M. LÃ¤ngkvist, L. Karlsson, and A. Loutfi, "A review of unsupervised feature learning and deep learning for time-series modeling," *Pattern Recognition Letters*, vol. 42, pp. 11–24, 2014.

45. M. Magris and A. Iosifidis, “Bayesian learning for neural networks: An algorithmic survey,” *Artificial Intelligence Review*, vol. 56, no. 10, pp. 11773–11823, 2023.
46. A. Mahajan, S. Das, W. Su, and V. H. Bui, “Bayesian-neural-network-based approach for probabilistic prediction of building-energy demands,” *Sustainability*, vol. 16, no. 22, p. 9943, 2024.
47. T. N. Palmer, “Predicting uncertainty in forecasts of weather and climate,” *Reports on Progress in Physics*, vol. 63, no. 2, pp. 71–116, 2000.
48. Y. Pang, Y. Wang, and Y. Liu, “Probabilistic aircraft trajectory prediction with weather uncertainties using approximate Bayesian variational inference to neural networks,” in *AIAA Aviation 2020 Forum*, p. 2897, 2020.
49. E. Pinheiro and T. B. Ouarda, “Uncertainty decomposition and quantification of seasonal precipitation forecasting based on Bayesian neural networks,” *Atmospheric Research*, p. 108815, 2026.
50. S. Poornima, M. Pushpalatha, R. B. Jana, and L. A. Patti, “Rainfall forecast and drought analysis for recent and forthcoming years in India,” *Water*, vol. 15, no. 3, p. 592, 2023.
51. D. E. Rumelhart, G. E. Hinton, and R. J. Williams, “Learning representations by back-propagating errors,” *Nature*, vol. 323, no. 6088, pp. 533–536, 1986.
52. D. Salite, “Traditional prediction of drought under weather and climate uncertainty: Analyzing the challenges and opportunities for small-scale farmers in Gaza Province, southern region of Mozambique,” *Natural Hazards*, vol. 96, no. 3, pp. 1289–1309, 2019.
53. F. A. F. Sham, A. El-Shafie, W. Z. W. Jaafar, M. Sherif, and A. N. Ahmed, “Advances in AI-based rainfall forecasting: A comprehensive review of past, present, and future directions with intelligent data fusion and climate change models,” *Results in Engineering*, vol. 27, p. 105774, 2025.
54. F. Smarandache, *A Unifying Field in Logics: Neutrosophic Logic, Neutrosophy, Neutrosophic Set, Neutrosophic Probability*, 3rd ed. Rehoboth, DE, USA: American Research Press, 1999.
55. G. Wang, T. Xu, T. Tang, T. Yuan, and H. Wang, “A Bayesian network model for prediction of weather-related failures in railway turnout systems,” *Expert Systems with Applications*, vol. 69, pp. 247–256, 2017.
56. S. F. A. R. Wissal, L. Amhaimar, A. Khalidi, and B. Talbi, “A hybrid long-term photovoltaic power prediction model integrating a BiLSTM network with residual correction via CatBoost,” *Results in Engineering*, p. 108898, 2025.
57. W. XiangJun and M. M. Al-Hashimi, “The comparison of adaptive neuro-fuzzy inference system (ANFIS) with nonlinear regression for estimation and prediction,” in *2012 International Conference on Information Technology and e-Services*, pp. 1–7, IEEE, Mar. 2012.
58. L. A. Zadeh, “Fuzzy sets,” *Information and Control*, vol. 8, no. 3, pp. 338–353, 1965.
59. Y. Zhang, J. Wang, and X. Wang, “Review on probabilistic forecasting of wind power generation,” *Renewable and Sustainable Energy Reviews*, vol. 32, pp. 255–270, 2014.
60. N. Zhou, B. Shang, M. Xu, L. Peng, and G. Feng, “Enhancing photovoltaic power prediction using a CNN-LSTM-attention hybrid model with Bayesian hyperparameter optimization,” *Global Energy Interconnection*, vol. 7, no. 5, pp. 667–681, 2024.