International Journal of Environmental Sciences
ISSN: 2229-7359
Vol. 11 No. 19s, 2025
https://theaspd.com/index.php

Systematic Review On Optimization Of Artificial Neural Network For Forecasting Of Rainfall

Vertika Shrivastava[1] **,Sanjeev Karmakar**[2]
[1]Department of Computer Applications, Bhilai Institute of Technology, Durg, Chhattisgarh, India.
[2]Department of Computer Applications, Bhilai Institute of Technology, Durg, Chhattisgarh, India.
[1]mail2vertika@gmail.com, [2]dr.karmakars@gmail.com

Abstract: *This systematic review reviewed a wide range of optimization approaches adopted for the enhancement of ANN performance when it comes to rainfall prediction. Chaotic, nonlinear, and high-variance time series with initial conditions sensitivity are needed to obtain accurate and precise rainfall time series data prediction. ANNS serves as a solid foundation to model these complexities. It is important to mention that predictive accuracy and reliability rely a lot on the optimization techniques selected. The paper reviews diverse optimization algorithms, ANN architectures, and data preprocessing strategies implemented for rainfall prediction The comparative analysis demonstrates the effectiveness of the top techniques regarding the generalization capability of the deep learning model and the accuracy of forecasts. The discoveries of this review can be an excellent reference for both researchers and practitioners to optimize the models based on ANN for successful precipitation prediction. This systematic review examines various optimization approaches aimed at enhancing the performance of artificial neural networks (ANNs) in rainfall prediction. Accurate and precise predictions of rainfall time series data require the handling of chaotic, nonlinear, and high-variance time series, which are sensitive to initial conditions. ANNs provide a solid foundation for modeling these complexities; however, the predictive accuracy and reliability depend significantly on the chosen optimization techniques. The paper reviews a range of optimization algorithms, ANN architectures, and data preprocessing strategies used for rainfall prediction. It includes a comparative analysis that demonstrates the effectiveness of the best methods in terms of the generalization ability of the deep learning model and forecast errors. The findings of this review can serve as an excellent reference for both researchers and practitioners seeking to optimize ANN models* ***for*** *successful precipitation prediction.*
Keywords: *Artificial Neural Networks (ANN), Rainfall Prediction, Optimization Algorithms, Time Series Forecasting, Chaotic Systems, Deep Learning.*

1. INTRODUCTION

Artificial neural networks (ANNs) are computerized models emulating human brain operation, able to learn from intricate and hazy information. Neurons are the fundamental elements that create a neural network. They process the input signals by other neurons or the input signals from outside their structure, provide them as output signals to other neurons and process the input signals by using an activation function. A network of neurons consists of three distinct types of layers: input, hidden, and output layers. The input layer receives the input data, the hidden layer processes it, and the output layer provides the learning results. A neural network can consist of one or more hidden layers, depending on its complexity [1]. But the mechanism by which a neural network learns is through adjusting something called synaptic weights, or the numbers that determine how strongly two neurons are connected to one another. Generally, the objective is to minimize the loss function (also known as an error function) that determines the difference between the actual output of the network and its expected output. To accomplish this, the synaptic weights are adjusted using optimization techniques that rely on the gradient of the loss function in relation to the weights. In order to achieve the lowest state, the gradient points in the direction opposite to that of the Function's highest slope [2].

Chaotic time series present complexities that make them an ideal subject for training time series neural networks. To enhance neural networks for the analysis of chaotic time series, one can achieve precise and reliable predictions by employing adaptive optimization methods, appropriate network architectures, and preprocessing strategies. This technical report can assist researchers concentrating on the neural network analysis of chaotic time series data.

Optimization is defined as the action, process, or methodology of making something as functional or effective as possible [4]. The process of optimizing neural networks is part art, part science, with the right solution for a particular task often coming only after much trial and error in the tuning parameters that help to train a network.

Optimizing a neural network means adjusting its parameters to minimize an already defined loss function. The loss function provides a measure of how different the predicted values are from the actual values. The purpose of the optimization process is to find the parameter values for the minimum loss function. Usually, optimization algorithms are used for this, the most common of which is gradient descent.

1.1 Background on rainfall forecasting and its importance.

The role of rainfall forecasting is crucial to meteorology as it serves several sectors such as agriculture, water resources, disaster management, and urban planning. Predicting rainfall correctly helps to reduce the effects of these extreme weather events (floods and droughts), which decreases economic losses and protects human lives. Traditional approaches for predicting rainfall rely on numerical weather prediction (NWP) models and statistical methods. While these approaches are beneficial, they often struggle to capture the highly non-linear and chaotic nature of precipitation events. The limitations of traditional methods have resulted in a growing fascination with data-driven approaches, particularly Artificial Neural Networks (ANNs), which are highly effective at capturing intricate relationships between data points in time series data. Models based on artificial neural networks have proven to be more effective in capturing the intricate relationships within meteorological data, positioning them as valuable tools for improving rainfall prediction accuracy. As the use of extensive neural networks becomes more prevalent, their effectiveness greatly relies on well-optimized algorithms that enhance the model's adaptability, generalization capabilities, and forecasting accuracy. This comprehensive review of optimization methods emphasizes various strategies that can refine ANN-based rainfall prediction models, resulting in more precise and dependable rainfall forecasts.

1.2 Role of Artificial Neural Networks (ANN) inforecasting.

Artificial Neural Networks (ANNs) have become a valuable tool for forecasting due to their capacity to represent intricate, nonlinear patterns in data. Unlike conventional statistical approaches that rely on predetermined equations and certain assumptions, ANNs derive insights directly from past data, which enhances their effectiveness in addressing time-series forecasting challenges such as predicting rainfall.

Multiple meteorological data points such as temperature, humidity, atmospheric pressure, and previous rainfall records are analyzed, enabling ANNs to recognize complex relationships and correlations in rain- fall prediction. They can be used to approximate any function, their function at a particular environment can change over time, and they can generalize to unseen or untrained data.

Various ANN architectures have been employed in the domain of rainfall prediction, ranging from Feed forward Neural Networks (FNNs) and Recurrent Neural Networks (RNNs) to more sophisticated architectures such as Long Short-Term Memory (LSTM) and Convolutional Neural Networks (CNNs). These models are more efficient than most standard techniques as they can grow themselves in receiving the changes due to spatial and temporal variability in weather.

However, the performance accuracy and reliability of ANN-based forecasting is highly dependent on optimization techniques. Hyper parameter tuning, choosing the optimal network architectures, determining the best data preprocessing method and utilizing more sophisticated training algorithms are critical to improving predictive performance. This study provides a systematic overview of numerous optimization techniques that enhance ANN models in order to provide more accurate and reliable rainfall predictions.

1.3 Challenges in optimizing ANN for rainfallprediction.

The inherent complexity and chaos of meteorological data make it difficult to optimize Artificial Neural Networks (ANNs) for rainfall prediction. Here are some of the major challenges. On-Linearity and Chaos of Rainfall Data.

Due to precipitation data being non-linear, highly variable, and chaotic, it neither lends stability for ANN learning nor pattern generalizability. Conventional optimization approaches might not distinguish between these complex dependency behaviors well. Issues with Data Quality and Data Preprocessing

Rainfall prediction is dependent on a variety of meteorological datasets, which generally include missing values, noise, and inconsistencies. ANN performs well but requires data preprocessing (like scaling, normalization, missing value treatment, variable selection.

Choosing an Ideal Network Architecture, as each ANN architecture (e.g., Feedforward Neural Network, Recurrent Neural Network, Long Short-Term Memory) displays unique performance results depending on the characteristics of the data, their selection is not trivial. This selection can lead to overfitting or under fitting the model.

Tuning hyper parameters for ANNs is essential to determine settings such as the number of hidden layers, the count of neurons, the learning rate, and the choice of activation functions. Manual tuning can be time-consuming, and sub-optimal choices can severely impact model performance. Efficiency of training FYI: you are trained on data until Oct 2023

Other limitations are the long training time and high computational cost for large and complex ANN models. Care to balance accuracy and computational cost lies in optimizing network size and using efficient training algorithms, adaptive learning rates, and parallel computing.

Generalization and Overfitting ANNs tend to over fit, which means that the model learns the noise of the training data, so it does not generalize well on the test data. Additionally, to improve generalization, it would be necessary to incorporate some form of regularization, like dropout, L1/L2 regularization or early stopping. Absence of Explain ability and Interpretability ANNs are black-box models, and it is hard to see how or why they make certain decisions. This unexplainability causes a lack of trust in the predictions provided by ML models for risk events, particularly in critical applications like disaster management and agriculture. Joining Forces with Hybrid Models.

However, the process of integrating ANNs with other methods optimization methods (e.g., evolutionary algorithms, wavelet to accomplishes these goals, we performed a systematic transform, or fuzzy logic) may enhance performance at the expense of additional model complexity. It can be hard to find the right tradeoff between interpretability and accuracy.

These challenges are also dealt with using strong optimization techniques, such as advanced metaheuristic- tic algorithms, hyper parameter search methods, and hybrid modeling approaches and are systematically reviewed in this study.

1.4 Objectives and contributions of this systematic review.

The goal of this systematic review is to analyze the existing optimization techniques aimed at improving the performance of Artificial Neural Networks (ANNs) in the field of rainfall forecasting. Because rainfall data are the nonlinear and chaotic data, choosing right ANN architectures and enhancing optimization is the best approach to supervise the forecast. The study examines various optimization approaches as hyperparameter tuning methods, evolutionary algorithms, hybrid models, regularization methods to tackle overfitting, computational efficiency, and data preprocessing issues.

The primary contributions of this review are: (1) systematic categorization of optimization methods utilized within the domain of ANN-based rainfall forecasting,(2) performance evaluation and comparison of a variety of optimization algorithms for their accuracy, efficiency, and generalization performance, (3) identification of crucial challenges and constraints in optimizing ANN models for rainfall prediction, and (4) provision of indications for further study avenues aimed at enhancing forecast dependability. This review could be helpful in obtaining the improve of ANN models in rainfall forecasting by providing a summary of consequences of existing researches.

2.METHODOLOGY

2.1 Systematic review protocol and framework.

In this systematic review, a written protocol is used that helps to ensure the systematic approach to describe, identify, and interpret all optimization techniques used for Artificial Neural Networks (ANNs) for rainfall fore-casting. This study is motivated by main research questions which optimization techniques are commonly used for ANN, which optimization techniques improved forecasting accuracy, what are some of the major challenges in ANN optimization and what are the future trends.

Literature search in various academic databases, IEEE Explore, Springer Link, Science Direct, Elsevier, Google Scholar and the ACM Digital Library. A few targeted queries were used to find relevant literature such as "Artificial Neural Network for Rainfall Forecasting", "Optimization of ANN for Weather Prediction", and ANN Hyper Parameter Tuning for Meteorology. We applied specific criteria for study inclusion/exclusion to preserve the quality and relevance of selected studies. Only papers that either dealt with ANN-based long-and short-term rainfall forecasting, ANN based optimization methods or empirical validation studies were selected while non-English publications, duplicate records and papers without experimental evaluation were excluded. Data mined from the articles included the type of artificial neural network (ANN) models employed, optimization methods employed, performance metrics presented, and dataset attributes. Some of the methods of the studies were rated in a quality assessment on the relevance of the studies, methodological rigor, statistical validation, and citation impact to reduce bias (Table 5). The gathering set of literature was systematically classified, reviewed, and analyzed to find patterns, challenges, and best practices in ANN optimization for rainfall prediction. Herein, a review framework is introduced that enables a structured, credible analysis of ANN optimization strategies, bringing relevant insights for both researchers and practitioners in the field.

2.2 Databases and sources used for literature collection.

To achieve an extensive literature review, literature was obtained from various academic database and sources that offer high-profile scientific articles in the audio production and computer science fields. The main databases employed for this study are IEEE Xplore, Springer Link, Science Direct, Elsevier, Google Scholar, and ACM Digital Library. These publication databases were chosen because they contain a large amount of peer-reviewed journal articles, conference proceedings, and book chapters about ANN-based rainfall prediction and optimization methods. Furthermore, and to complement the literature with practical applications in the real world, relevant government reports, technical white papers and credible institutional repositories were taken into account. The search strategy included keywords like “Artificial Neural Network for Rainfall Forecasting," "Optimization of ANN for Weather Prediction," "ANN Hyper Parameter Tuning for Meteorology," etc. Using these wide-ranging sources, this review provides a thorough and current treatment of recent developments in optimizing ANNs for predicting rainfall.

2.3 Inclusion and exclusion criteria for paper selection.

The searching process included some inclusion and exclusion criteria to have relevant and quality studies. Only studies specifically addressing ANN-based rainfall forecasting studies with optimization methodologies and some applied experimentation outcomes, along with comparisons were included in the review process. Furthermore, only peer-reviewed journal articles, conference proceedings, and book chapters have been used to ensure the credibility of the findings. The selection was limited to publications fewer than 10 years old in order to include more recent developments in the field.

On the other hand, studies were removed from examination that were either not published in the English language, did not include benchtop experimentation, or dealt only with general weather forecasting with- out specificity on rainfall prediction. Duplicate publications, works that were redundant, and those lacking sufficient technical depth were also excluded. By using these criteria to narrow down the included papers of this review, the outcome is a more reliable analysis of high quality and relevant papers that truly impact the field of ANN optimization techniques for rainfall forecasting.

2.4 Classification of optimization techniques considered.

Rainfall forecasting using Artificial Neural Networks (ANNs) has been one of the most researched fields over the years, and they are optimized in a variety of ways in the literature. These algorithms focus on the efficiency in adjusting the network weights for gradient-based optimization, such as Stochastic Gradient Descent (SGD), Adam, and RMSprop, to reduce the degree of error. Also, evolutionary and metaheuristic algorithms such as Genetic Algorithms (GA), Particle Swarm Optimization (PSO), and Ant Colony Optimization (ACO) are used to optimize hyper parameters for better performance due to the broader space they are exploring.

Hybrid optimization methods, focusing on optimizing multiple approaches, like using deep learning with a wavelet transform or blending ANN with a fuzzy logic scaler, have also been getting some traction for improve the methods’ accuracy and ambiguities. Regularization and fine-tuning techniques like dropout, batch nor- normalization, and L1/L2 regularization are also essential components in avoiding overfitting and enhancing model robustness. The classification of these techniques is important to determine the optimal optimization strategies for improving ANN performance in rainfall forecasting, as each of them offers a unique way to boost ANN performance.

3.LITERATURE REVIEW OF ARTIFICIAL NEURAL NETWORKS FOR RAINFALL FORECASTING.

The field of mathematics and science known as "chaos theory" examines complex, dynamic systems that exhibit seemingly random, unpredictable, and chaotic behavior. These systems are sensitive to initial conditions, which mean that over time, slight variations in the initial conditions might produce noticeably different results. The goal of chaos theory is to comprehend the underlying order in systems that appear chaotic. Chaos theory has had a profound impact on our understanding of the natural world and complex systems. It challenges the traditional deterministic view of science and emphasizes the limits of predictability in certain systems. Researchers and scientists use chaos theory to model, analyses, and make sense of systems that exhibit chaotic behavior, even though they may appear random at first glance. Forecasting the behavior of chaotic systems can be challenging and intricate because of their sensitivity to initial conditions and the complex, nonlinear dynamics at play. However, there are several techniques and approaches that researchers use to gain insights into chaotic systems and make predictions to some extent. Artificial Neural Networks (ANNs) can be a valuable tool for predicting chaotic systems like rain- fall patterns. Rainfall is influenced by a complex interplay of meteorological factors, making it a challenging system to predict accurately. ANNs, with their ability to capture complex nonlinear relationships, can be well- suited for this task. However, optimization of ANN is essential for achieving better model performance, efficient training, and robust generalization to unseen data. It is essential to the success of neural network applications in a variety of fields and uses. Typical optimization methods for neural network training consist of stochastic gradient descent (SGD), variants of SGD like Adam and RMSprop, learning rate schedules, weight initialization methods, dropout, batch normalization, and various regularization techniques. The subsequent paragraphs are dedicated to brief discussions on the techniques and results of optimization of ANN reported by various previous researchers. The first person to use ANN for weather prediction was Hu (1964). He used the Adaline adaptive system for pattern classification. Using 200 sea level pressure readings during the winter and 24-hour pressure change patterns between 25O and 65O N and 110O and 170O W, the system was trained to forecast whether it will rain or not in the San Francisco Bay area on 100 different scenarios. When these projections were compared against the official forecasts from the U.S. Weather Bureau for the same timeframes, they performed favorably. After conducting this research, he made the case that adaptive systems, which do not completely comprehend the dynamics, are capable of producing precise weather forecasts. [9]. A significant investigation into the application of artificial neural networks (ANN) for forecasting rainfall was conducted by French et al. in 1992. The researchers utilized a neural network to predict two-dimensional rainfall one hour ahead. They inputted current rainfall data generated by a mathematical rainfall simulation model into their artificial neural network (ANN). Nevertheless, this work had certain constraints. One challenge was striking a balance between interaction and the trade-off concerning training duration. The quantity of hidden layers and nodes did not appear to accommodate the higher-order relationships essential for effectively abstracting the process, in contrast to the number of input and output nodes. However, it has been regarded as one of the pioneering contributions to the application of ANN and set a new standard for comprehending and assessing ANN's roles in examining intricate geophysical phenomena. [10]. Chen and Takagi (1993) introduced a feature-based neural network approach for forecasting rainfall in the open sea area close to Shikoku, Japan. By employing a four-layer neural network, the connection between rainfall intensity patterns and GMS data from geostationary meteorological satellites was autonomously identified. Both visible and infrared pictures from the GMS image were supplied into the network as input data, and the network was trained using the back propagation learning algorithm. [11]. Zhang and Scofield (1994) presented an ANN approach for spotting cloud mergers and calculating strong convective rainfall using satellite data. Using Artificial Neural Network group approaches, the NOAA/NESDIS Satellite Applications Laboratory has developed an expert system for Satellite-derived Estimation of Rainfall (ANSER) and found that the following can be achieved: automatic cloud merger detection, rainfall computation ten times faster, and average rainfall estimate errors decreased to less than 10% for the entire precipitation event [12]. The effectiveness of multiple linear regressions and artificial neural networks (ANN) in estimating missing rainfall data over Cyprus was investigated by Michaelides et al. (1995). For places where the current time series stops (ahead extension) or where the archives begin relatively recently (backward extension), they have provided a way to create a sufficiently long time series of rainfall records. The methodology uses artificial neural networks to estimate the amount of rainfall that occurs each day at particular Cyprus monitoring sites, or "target stations." The approach uses daily rainfall measurements from neighboring locations with a sufficiently long and full archive of data (termed control stations) as input. Using records from neighboring stations, this method can be used to confirm dubious data and close any gaps in the rainfall observation network. Neural network technology and the conventional multiple linear regression approach are contrasted. In this instance, the control stations were regarded as the independent variables, and the target station as the dependent variable [13]. ANN was used by Kalogirou et al. (1997) to reconstruct the rainfall in Cyprus over a period of time. For the purpose of estimating precipitation at a few chosen rainfall collection stations in Cyprus, they employed feed for- ward multilayer neural networks. Nine years' worth of archived data, along with six control stations positioned surrounding a target station, were utilized to train an appropriate artificial neural network. In order to create a network that produces the best reconstruction of missing rainfall information, various neural network topologies and learning rates were evaluated. Then, for this goal, they selected an architecture of many hidden layer neural networks. This type of architecture was used to solve issues with comparable needs. At every control station, the parameters that were utilized to train the network were gathered. The Julian day, altitude, separation between the target and control stations, and precipitation were those. For the training data set, the correlation coefficient was found to be 0.933. Unknown data for the target station was used to verify the network. They did this for a year, during which time their data were removed from the training set. For the unidentified case, the correlation coefficient was 0.961. Less than 17.1 mm of precipitation was the extent of the prediction inaccuracy, which is regarded as acceptable. [14]. In another research effort, Venkatesan et al. (1997) predicted the summer monsoon rainfall in India by utilizing an ANN that incorporated various meteorological parameters as inputs. They employed multilayer feed-forward neural networks that were trained using the error-back propagation (EBP) method. Three network models were created and adjusted with 2, 3, and 10 input parameters—factors known to significantly influence the Indian summer monsoon rainfall (ISMR). Subsequently, they thoroughly compared their findings with those from statistical models. The predictions generated by the network models demonstrated their potential as a valuable tool for forecasting ISMR [15]. Lee et al. (1998) separated the available data into homogenous subpopulations to fore- cast rainfall. They proposed a divide and conquer approach, breaking the region up into four sub-areas, each modeled with a unique technology. They have used radial basis function (RBF) networks to forecast rainfall for two larger regions. Using a simple linear regression model, the rainfall in the final two smaller sub-areas was predicted. Then, they compared the two methods and discovered that RBF networks generated accurate predictions whereas linear models produced inaccurate ones. The authors believed that their method was suitable for both emergencies and long-term maintenance of contaminated areas [16]. A theoretically rigorous framework for minimizing the cross-entropy function in an error backpropagation setting was devised by Fischer and Stafur (1999) [17]. They arrived at the backpropagation formulas for computing-efficient partial derivative evaluation. Additionally, they evaluated the efficacy of backpropagation training using three optimization procedures for weight updating: the gradient descent (GD), the one-step Polak-Ribiere-conjugate gradient (PR-CG), and the Broyden-Fletcher-Goldfarb-Shanno (BFGS) memoryless quasi-Newton algorithms on a multispectral pattern classification problem with a challenging level of complexity and a larger training set. Batch learning and epoch-based learning (with epoch sizes k = 30, 300, 600, and 900) were the two forms of off-line backpropagation training that were examined. They came to the conclusion that, when batch and epoch-based modes of operation were taken into account, gradient descent error backpropagation produced the best and most reliable out-of-sample performance results. PR-conjugate gradient error backpropagation is typically better if maintaining classification accuracy is a trade-off with learning speed maximization. To prevent undesirable instability in the classification results, stochastic epoch-based variants of local optimizers with a bigger epoch size should be selected instead of a smaller one if the training set is particularly large. Koizumi (1999) constructed an artificial neural network (ANN) model using numerical products generated by the Japan Meteorological Agency (JMA) Asian Spectral Model in conjunction with radar, satellite, and weather station data. One year's worth of data was used to train the model. The results showed that after three hours, the ANN's performance surpassed that of both the persistence forecast and the linear regression forecast, as well as the numerical model's precipitation prediction. However, since the ANN model was trained using data from just one year, the findings had certain limitations. The author maintained that the neural network would improve as more training data became accessible. It remains uncertain how much each predictor influenced the forecast and whether incorporating new data would have enhanced it [18]. In 2000, Toth et al. conducted a model comparison to estimate short-term rainfall and create a real-time flood forecast. They utilized three different time series models, with lead times ranging from one to six hours, to assess storm rainfall events that occurred in the Sieve River basin, Italy, between 1992 and 1996: the k-nearest neighbors (KNN), auto-regressive moving average (ARMA), and artificial neural network (ANN). The results showed that when it came to improving the accuracy of runoff forecasts, the ANN performed better than the model [19]. Three different types of artificial neural networks (ANN)—multilayer feed forward neural network (MLFN), Elman partial recurrent neural network (Elman), and time delay neural network (TDNN)—have been constructed and compared by Luk et al. (2001). [20]. Abraham et al. (2001) used four soft computing techniques in the same year to forecast the rainfall using time series: ANN utilizing Scaled Conjugate Gradient Algorithm (ANNSCGA), Evolving Fuzzy Neural Network (EfuNN), Adaptive Basis Function Neural Network (ABFNN), and General Regression Neural Network (GRNN). They have used a regression technique called Multivariate Adaptive Regression Splines (MARS), which uses a certain class of basic functions as predictors. The research utilized monthly precipitation as the input data for the training model. Researchers analyzed rainfall information spanning 87 years that was provided by the southern Indian state of Kerala. The practical findings indicated that neuro fuzzy systems outperformed the pure neural network technique in terms of performance time and error rates. However, predicting rainfall remains one of the 20 most challenging and intricate components of the hydrological cycle, due to its vast variability over numerous scales, both spatially and temporally [21]. Wong et al. (2003) created a rainfall forecasting model using fuzzy logic and artificial neural networks (ANN) within the realm of soft computing. Initially, they segmented the data into subpopulations by employing Self-Organizing Maps (SOM), aiming to simplify the overall data space and enhance its homogeneity. After classification, they utilized Backpropagation Neural Networks (BPNNs) to extract generalization features from the data within each cluster. Fuzzy rules were then derived for each cluster. The rainfall prediction was carried out using this fuzzy rule base. They compared their method to a traditional approach utilizing the orographic effect along with radial basis function networks. The results indicated that their proposed method could achieve results comparable to the established technique. However, the authors noted that one advantage of their method is that it allows analysts to understand and interact with the model through the use of fuzzy rules [22]. Chistodoulou et al. (2004) employed weather radar in 2004 to predict rainfall rate as an alternative to rain gauges that measure rainfall on the ground. Using radar data as input and rain gauge measurements as output, the statistical KNN classifier and the neural SOM were applied to the classification problem. With an average error rate of 23%, the radar reflections were used to forecast the ground-level rainfall rate. In the end, they have found that it is feasible to forecast the rate of rainfall using weather radar data. [23]. Guhathakurta (2006) created an artificial neural network (ANN) model to predict long-range monsoon rainfall for the subdivisions and districts of Kerala, utilizing the area-weighted value for each district's forecast. Ultimately, he discovered that the ANN-based model outperformed the statistical method [24]. Somvanshi et al. (2006) have presented methods for modeling and forecasting the behavioral pattern in rainfall events based on historical observations. They have introduced two fundamental different approach to model building: the autoregressive Integrated moving averages (ARIMA) statistical methodology and the recently discovered computationally powerful approaches based on artificial neural networks (ANNs).

To assess the effectiveness of the forecasts, they analyzed mean annual rainfall data from the Hyderabad area of India, covering a span of 104 years from 1901 to 2003. The models were trained using mean annual rainfall data spanning 93 years. The weights and regression coefficients were obtained for the data by applying the ANN and ARIMA techniques, respectively. The final ten years of data were used to assess the model's performance. In conclusion, the study showed that the ANN model outperforms the ARIMA model and might be utilized as a suitable forecasting tool to estimate the rainfall. [25]. To replicate rainfall data, Nasseri et al. (2008) created a feed-forward type ANN. Back propagation (BP) and a genetic algorithm (GA) were used to train and optimize the network

The method was repeatedly used to anticipate rainfall using rainfall hyetographs, which record rain gauges in the Upper Parramatta watershed in Sydney's western suburbs, Australia. The study covered in this article shows that MLP type networks with GA consistently performed better than MLP type networks by themselves. Furthermore, the integration of the GA approach with ANN demonstrates an advantage in lowering the order of errors when compared to the ANN model alone. Furthermore, it was discovered that because of the unpredictable nature of rainfall and the finite temporal memory, the number of effective input stations decreases as time steps increase. Finally, because of the nature of integration, cumulative data may offer statistical performance in rainfall forecasting that is noticeably higher than discrete datatype [26]. ANN-based model was created by Abhishek et al. (2012) to predict the average monthly rainfall in the UDUPI district of Karnataka. The rainy season in Udupi is defined as April through November, whereas the primary monsoon seasons are May, June, July, August, and October.

They have therefore investigated the information from these eight months during the years from 1960 to 2010. In the end, three algorithms were examined in a multi-layer architecture: Layer Recurrent Network (LRN), Cascaded Back-Propagation (CBP), and Back Propagation Algorithm (BPA). The authors discovered that BPA is the most effective algorithm among the three. [27]. Deshpande (2012) proposed a creative technique for rainfall time series forecasting: The Multilayer Perceptron Neural Network. The rainfall samples were provided by the authorized government rainfall monitoring agency in the Indian state of Maharashtra, Yavatmal. This Rainfall Data series has been forecasted using the proposed Multilayer Perceptron Neural Network several steps ahead of time (1, 5, 10, 20). On testing and training data sets for short-term prediction, they have found that performance metrics like mean square error and normalized mean square error were optimal, in contrast to other networks like Jordan Elmann neural network, the Self–Self-Organized Feature Map, and the Recurrent Neural Network. [28].

Neural Architecture Optimization (NAO) is the title given by Luo et al. (2018) [29] to their straightforward and effective continuous optimization-based strategy for automatic neural architecture design. In their suggested methodology, they took into account three crucial elements: (i) An encoder for mapping or embedding neural network topologies into an uninterrupted space. (ii) A predictor that forecasts the accuracy of a network using its continuous representation as input. (3) A decoder that connects a network's continuous representation to its architectural design. They were able to carry out gradient-based optimization in the continuous space to locate the embedding of a new architecture with maybe higher precision thanks to the performance predictor and the encoder. The decoder then decoded to a network with an improved embedding. Through a large reduction in processing resources, experiments showed that the design found by the suggested method was very competitive for chaotic time series data analysis, or on par with the best findings of earlier architectural search methods.

Zhang et al. (2022) [30] sought to forecast summer precipitation in the Yangtze-Huaihe River Basin (YHRB) in eastern China using an ANN- based model. For usage with the YHRB, a modified version of the standard BPNN served as the primary ANN. Using the precipitation data along with the predictors/factors of air circulation and sea surface temperature, they calculated the correlation coefficients be- tween the precipitation and the factors. The top six precipitation predicting parameters were likewise arranged by them. Accurate projections were produced by using month-to-month (precipitation) forecast models for both the training and validation phases. Following the model training phase, they used genetic algorithm-based backpropagation (GABP), support vector machines (SVM), and multiple linear regression (MLR) to compare the BPNN with the standard one for the summer precipitation forecast. According to the study, the GABP algorithm forecasts precipitation more accurately than the other methods. For the YHRB, its mean absolute percentage error (MAPE) is about 20%, which is much lower than the results from the BPNN, SVM, and MLR. They then ascertained which GABP month-to-month model yielded the best summer precipitation forecast by summing together the monthly precipitation. The summer scale forecast that was produced as a result did remarkably well per evaluation metrics. For the YHRB, the basin-averaged MAPE and anomaly rate were 4.7% and 88.3%, respectively. These findings suggested that the YHRB's summer precipitation forecasts could benefit from the application of GABP.

González-Zapata et al. (2022) [31] used the Echo State Network (ESN) as an illustration example to demonstrate how optimizing it might increase the chaotic time series' prediction horizon. It was established through the application of several optimization techniques to various machine learning models that metaheuristics were a viable choice for ESN optimization. In their published work, they applied Particle Swarm Optimization (PSO) to optimize an ESN in closed-loop mode. The optimized ESN's prediction results indicated a roughly two-fold increase in steps ahead, demonstrating the value of fine-tuning an ML method's hyper parameters to extend the prediction horizon. A method for using BPNNs to properly and reliably estimate the dependability of engineering systems, such as industrial robot systems and turbochargers, was proposed by Tanhaeean et al. (2023) [32].

The Boxing Match Algorithm (BMA), an evolving metaheuristic algorithm having a novel weight update technique, was proposed to improve the performance of the ANN in dependability predictions. As a result, when the performance of the proposed method was measured using three sets of function approximation data, the hybrid model of BMA-BPNN demonstrated a notable degree of precision in BPNN weight and bias optimization. Next, in order to improve the precision of dependability estimates, the BMA was used to address engineering difficulties. The gathered data showed that the suggested approach outperformed additional compared methods by about 20% in reliability forecasts for engineering systems and performed extremely well in function approximation. In a similar vein, the algorithm in question offered further properties including strong performance, a rapid rate of convergence, and an acceptable calculation time. A generalized degree of freedom approximation strategy of MLP, concentrating on chaotic time series modeling, was presented by Qiao M et al. (2023) [33]. They created a comprehensive framework for analyzing chaotic time series, which consists of phase space reconstruction, model training, and model selection through the acquisition of Akaike information criteria, aimed at illustrating the training loss function. Two artificial chaotic time series and two real-world chaotic time series were used to test the efficacy of the suggested approach. The numerical results demonstrated how well the suggested optimal strategy performed in differentiating the best model from a set of other strategies. Additionally, the optimized models demonstrated remarkable performance in multi-step prediction tasks.

4 . Optimization Techniques for ANN in Rainfall Forecasting.

a. Hyper parameter Tuning Approaches

Here is a general overview of how neural networks areoptimized [5]:

1. Choose a Loss Function: The first step is to generatea loss function that evaluates the disparity between the desired values and the actual outputs. The particular problem being attempted to be solved will

determine the loss function to be used. For instance, cross-entropy loss is frequently used for classification problems and Mean Squared Error (MSE) is normally used for regression problems.

2. Initialize Model Parameters: Initial small random values are employed to establish the weights and biases of the neural network. During training, the algorithm that optimizes will make modifications to these parameters.
3. Forward Propagation: During each training iteration, input data is fed through the network using forward propagation. This involves passing the input data through each layer of the network, applying activation functions, and generating predictions.
4. Calculate Loss: During this step, using the actual target values and the expected outputs, the loss function is computed. The network's effectiveness with the provided data is indicated by this loss value.
5. Backpropagation: Neural network optimization is primarily based on backpropagation. It entails figuring out the loss function's gradients in relation to the model's parameters. Gradients show the amount and direction of changes that must be made to eachparameter in order to lower the loss.
6. Gradient Descent: Based on the computed gradients, the optimization algorithm, typically gradient descent or its variants, such as stochastic gradient descent or mini-batch gradient descent, modifies the model parameters. To reduce the loss, the parameters are updated in the opposite direction of the gradients.
7. Update Parameters: Iteratively updating the settings in the direction of reducing the loss is done. The learning rate, a hyper-parameter that sets the step size in the parameter space, regulates the size of the updates.
8. Iterate: Steps 3 to 7 are repeated for a predefined number of epochs (complete passes through the training data) or until the loss converges to a satisfactory level.
9. Validation and Testing: It is crucial to track the model's performance throughout the training stage using newly-generated validation data. By doing this, overfitting by which the model becomes unduly specialized to the training set is avoided.
10. Hyper parameter Tuning: Alongside learning rate, there are other hyper parameters that can impact optimization, such as batch size, activation functions, the number of layers, and regularization techniques. Hyper parameter tuning involves finding the best combination of hyper parameters that results in the optimal performance.
11. Early Stopping: Strategies like early stopping, which stops training if the validation loss starts to rise after a predetermined number of epochs, to avoid overfitting can be employed.
12. Advanced Optimization Algorithms: There are advanced optimization algorithms beyond basic gradient descent, such as Adam, RMSProp, and more. These algorithms incorporate adaptive learning rates and momentum to accelerate convergence. The analysis of chaotic time series data has applications in diverse fields such as weather prediction, financial modelling, and complex systems dynamics [6]. Neural networks, particularly deep learning architectures, have shown promise in capturing the intricate pat- terns present in chaotic data. However, training these networks on chaotic data requires specialized optimization techniques to ensure convergence to meaningful solutions.
13. Chaotic time series data pose several challenges for neural network optimization due to their inherent characteristics, including high-dimensional input spaces, sensitive initial conditions, and non-stationarity. These challenges can lead to convergence issues, poor generalization, and overfitting. Various network architectures under considerations for analyzing chaotic time series data are [7]: Recurrent Neural Networks (RNNs): RNNs, particularly Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU) architectures, are well-suited for modeling temporal dependencies in chaotic time series data. Convolutional Neural Networks (CNNs): CNNs can capture spatial patterns in multidimensional chaotic data, such as image-based representations of chaotic attractors. Hybrid Architectures: Combining RNNs and CNNs can capture both temporal and spatial dependencies, improving the network's ability to analyze complex chaotic dynamics. Optimizing neural networks involves using various techniques and strategies to improve convergence speed, prevent overfitting, and find better solutions. Here are some techniques commonly used to optimize neural networks and minimize the loss function [8]:.
14. Gradient Descent Variants: • Stochastic Gradient Descent (SGD): Instead of using the entire dataset in each iteration, SGD uses a single randomly chosen sample to update the model's parameters. This can speed up convergence but might introduce more noise. • Mini-Batch Gradient Descent: It achieves a balance between the stability of utilizing the complete dataset and the effectiveness of SGD. It up- dates the parameters using a small batch of data samples in each iteration.

1. Learning Rate Scheduling: • Learning Rate Annealing: Gradually reducing the learning rate during training can help the model converge more effectively. • Learning Rate Decay: Automatically decreasing the learning rate after a certain number of epochs can help the optimization process stabilize as it gets closer to the optimal solution.
2. Momentum: • Momentum: By using this method, the current update of the parameters includes a portion of the previous update. It helps the optimization process navigate flat or narrow regions of the loss landscape and accelerates convergence.
3. Adaptive Learning Rate Algorithms: • Adam (Adaptive Moment Estimation): An optimization approach that modifies each parameter's learning rate in response to previous gradients and squared gradients. It combines the benefits of both momentum and RMSProp.
4. Regularization Techniques: • L1 and L2 Regularization: Adding L1 (Lasso) or L2 (Ridge) regularization terms to the loss function penalizes large weights, preventing overfitting. • Dropout: During training, randomly "dropout" (disable) a fraction of neurons, which helps prevent overfitting by promoting network robustness.
5. Batch Normalization • Batch Normalization: Normalizes the activations of each layer within a mini-batch. It can help with faster convergence and improved generalization.
6. Weight Initialization: • Xavier/Glorot Initialization: This initialization technique provides initial weights according to the total number of both input and output neurons within a layer. It helps balance the initial scale of activations and gradients, aiding in convergence.
7. Architecture Modifications: • Adding/Removing Layers: Modifying the number of layers within a network can help adapt to the complexity of the problem. • Changing Activation Functions: Different activation functions like ReLU, Leaky ReLU, or GELU can affect convergence speed and stability.
8. Early Stopping: • Early Stopping: Monitor the validation loss during training and stop when it starts increasing. This prevents the model from overfitting the training data.
9. Data Augmentation: • Data Augmentation: Applying transformations (rotations, flips, etc.) to the training data can increase the diversity of the data, helping the model generalize better.
10. Gradient Clipping: • Gradient Clipping: Limits the magnitude of gradients during training, preventing extreme updates and promoting more stable convergence.
11. Ensemble Learning: • Ensemble Learning: Combining predictions from multiple neural networks can improve generalization and reduce overfitting.
12. Hyper parameter Tuning: • Grid Search and Random Search: Systematically explore different combinations of hyper parameters to find the best configuration.
13. Transfer Learning: • Transfer Learning: Use pre- trained models as a starting point and fine-tune them on your specific task. This can significantly speed up convergence.

The effectiveness of these techniques depends on the problem, dataset, and architecture of the neural net- work under consideration. Experimentation and careful tuning are essential in finding the optimal combination of techniques for the specific use case. A detailed sur- vey of the available literature on optimization of neural networks employed for analysis of chaotic time series data for the prediction purposes is provided in the sub- sequent apart of the paper.

b. Evolutionary and Metaheuristic Algorithms.

Many optimization techniques, including evolutionary and metaheuristic algorithms, have proven to be an important tool for rainfall forecasting to improve the performance of Artificial Neural Networks (ANNs). Inspired by natural processes and intelligent search strategies, these algorithms allow efficient exploration of rich solution spaces for hyper parameter tuning, network architecture selection, and weight optimization. Genetic Algorithm (GA) searches for optimal ANN parameters based on natural selection so that refinement is done iteratively, while Particle Swarm Optimization (PSO) imitates how social creatures (like birds and schools of fish) explore for best solutions in high dimensional spaces. Ant Colony Optimization (ACO) works similarly to this as it models the foraging behavior of ants to come up with an optimal way to reach a solution to the problem. Other outstanding approaches are Differential Evolution (DE) and Simulated Annealing (SA)—convergent improvements to ANN training that coordinate exploration and exploitation. These algorithms are effective in solving problems like premature convergence, slow convergence, and overfitting, which is why these algorithms are considered effective algorithms for optimizing ANNs for solving rainfall prediction problem. The complex systems they can manage provide more accuracy and reliability in forecasts.

International Journal of Environmental Sciences
ISSN: 2229-7359
Vol. 11 No. 19s, 2025
https://theaspd.com/index.php

c. Deep Learning and Hybrid Models.

Hence, deep learning approaches have greatly improved rainfall forecasting by providing a superior learning model of the underlying physical processes captured by ANN. Advanced deep learning architectures, including Convolutional Neural Networks (CNN) and Recurrent Neural Networks (RNN), particularly Long Short-Term Memory (LSTM) and Gated Recurrent Units (GRU) for processing spatial and temporal features, respectively, dominate video classification. Deep learning models are optimizing predictive accuracy but often they rely on large datasets and computational power. Further improving performance, hybrid models are being developed that use ANN combined with other computational techniques.

These are namely Wavelet-ANN models that adopt wavelet as Genetic Algorithms (GA), Particle Swarm Optimization transform to combine feature extraction, Fuzzy Logic- ANN models that introduce an uncertainty handling component, and Hybrid Evolutionary-ANN models where metaheuristic algorithms optimize the parameters of ANN. Moreover, it has been shown that the joint use of deep learning with models based on statistics like ARIMA- ANN can make predictions more robust [28]. By combating the weaknesses of various methods, these hybrid techniques also lead to more accurate and certain precipitation predictions.

4.1 Regularization and Generalization Methods.

However, performance of ANNs in rainfall forecasting is directly related to the proper use of regularization and generalization techniques that prevent overfitting and ensure that the models generalize well to unseen data. Regularization techniques such as L1 and L2 regularization (Ridge and Lasso regression) discourage large weight values from being used, balancing the model complexity to avoid over-reliance on certain data points. Another common technique is dropout, which randomly "turns off" some neurons during training to encourage the network to learn more robust and distributed representations. Another benefit is that it helps in stabilizing the training process of the model and reduces the time for convergence greatly by normalizing the distributions within each layer. Methods such as Early Stopping track the validation loss and stop the training whenever it starts killing the performance, avoiding overfitting. In order to increase generalization, data augmentation and ensemble learning techniques are used, where the predictions are made using multiple models, leading to more stability. The regularization and generalization techniques in placed guarantees ANN give a accurate forecasting value for various datasets having different climatic scenarios.

4.2 Comparative Analysis and Discussion.

While the aforementioned optimization methods show-cased varying degrees of effectiveness in improving the performance of Artificial Neural Networks (ANNs) for rainfall forecasting, the need for comparative evaluations between these optimization techniques for ANN in rainfall forecasting has been largely overlooked. SGD, Adam, RMSprop are examples of traditional gradient-based functions which offer computationally efficient weight updates but suffer from local minima and require a lot of hyper parameter tuning. On the other hand, procedures based on evolutionary and metaheuristics such overfitting (PSO) and Ant Colony Optimization (ACO) have a better global search and successfully optimized hyper- parameters, but are computationally expensive methods. Unlike traditional ANNs, deep learning architectures (LSTMs and CNNs) are better at learning both the spatial and temporal dependencies in the rainfall dataset, however they need a larger number of samples and a lot of computational power. to hybrid models such as incorporating wavelet transforms with fuzzy logic or statistical models such as ARIMA-ANN to achieve better prediction results by leveraging multiple strengths. Regularization techniques (dropout, batch normalization, L1/L2 regularization). No absolute best optimization method exists, but hybrid and metaheuristic methods show worth- while gains in prediction accuracy and stability. Results up to October 2023 demonstrate the impact of the selection process, indicating the importance of customized optimization techniques given the complexity of the dataset and computational limits for accurate rainfall prediction.

S.NO	AUTHOR	YEAR	TECHNOLOGY USED	RMSE
1	M. Tanhaeean, S. F. Ghaderi and M. Sheikhal-ishahi	2023	MLP, GA, GA-ANN, PSO-MLP, RBF, ICA-MLP	0.000351
2	Yong-li Wang, Dong-xiao Niu, Jiang-yan Liu	2023	Lyapunov Exponents	0.001
3	Mu Qiao Yanchun Liang, Adriano Tavares and Xiaohu Shi	2023	MLP	0.241

4	Ajay Kumar Bansal, Virendra Swaroop Sang- tani, Pankaj Dadheech, Nagender Aneja & Umar Yahya	2023	BBO-ANN	48.8
5	Ashish Kumar, Anshika Varshney, Ankita Arya, Manjot Kaur Bhatia	2022	BPN, RBFN, SOM, and SVM	-
6	Shahrokh Shahi, Flavio H. Fenton, Elizabeth M. Cherry	2022	LSTM, ESN, NVAR	0.001, 0.002, 0.001
7	Yuanyuan Gong, Hongzhi Ai, Zhi Gao, and Mancang Wang	2022	Fuzzy Neural Network	5.648
8	Lihong Yu, Linyang Xie, Chunmei Liu, Song Yu, Yongxia Guo, Kejun Yang	2022	BP Model, KHA-BP Model	341.59, 269.05
9	Astrid Maritza González-Zapata, Esteban Tlelo-Cuautle and Israel Cruz-Vega	2022	ML Methods, LSTM	290.5
11	Nisha Thakur, Sanjeev Karmakar, Sunita Soni	2021	Backpropagation-9 Hidden nodes	95.67
12	Urooj Kaimkhani, Bushra Naz, Sanam Narejo	2021	NARX Neural Network System	0.00483

Table 1: Summary of Recent Studies and RMSE Comparison

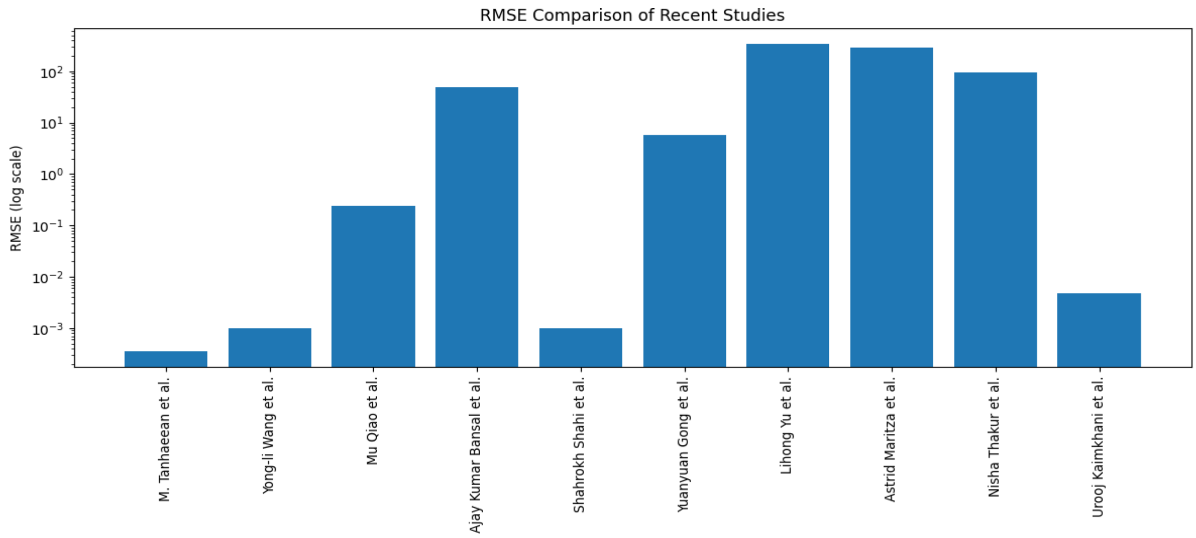

Fig. 1: Comparison Graph

4.3 Challenges.

1. **Computational Cost:** The computational cost is theonly limitation which come across the modelling and optimizing the ANN.
2. **Overfitting:** It is found in the review that the over- fitting is the most common challenge which came across. When your artificial neural network (ANN) model performs exceptionally well on training data but poorly on test data or fresh data, this is situation that arises in most of the model. This indicates that while your model has picked up on certain patterns and noise in the training set, it has not picked up on the overall characteristics and tendencies of the issue. Overfitting can produce inaccurate findings and bad decisions by reducing the accuracy and generalizability of the model.
3. **Data Quality:** One of the most important steps in the most of the model in the ANN models is making sure that the data is of high quality before it is run in any ANN model algorithms. The outcomes of using low-quality data might be seriously compromised, and judgments based on those results may have additional repercussions.

5.CONCLUSIONS AND FUTURE SCOPE.

As far as the scope of research is concerned there are many new models came into the existence for rainfall forecasting including AutoML Deep Learning Architecture, CNN, LSTM, Integration IOT etc. In coming future hybrid architecture and deep learning methods that creates an integrated computer program by fusing several neural networks methods will be tested. These hybrid architectures may also enhance overall performance by addressing some of the neural-related constraints. With the correct architecture, neural networks can simulate any unknown system or process, making them universal approximates. By defining the architecture and figuring out their weights and biases, they must be trained to approximate any function. Networks can be trained to reduce the overall "loss" on a training set by employing gradient descent variations for empirical risk minimization. The back-propagation algorithm calculates the gradient of the error concerning the parameters of the network. Our goal in this report is to examine several approaches towards optimizing ANN. Various global and local techniques are available for this purpose. The most popular technique for optimization is backpropagation. It is also possible to employ other techniques like simulated annealing, Tabu search, and genetic algorithms. Typically, three different kinds of parameters constitute an ANN: architecture (the arrangement of connections between the various neuronal layers), the process of learning to update the inter-connections' weights, and the activation function that changes the weighted input of a neuron into the activation of its output. The mean square error function, often known as the loss/cost function, is typically the objective function when discussing ANN optimization. To minimize the objective function, the neural network's optimal weight values must be determined. It has been demonstrated that gradient-based search approaches, such as back propagation, have significant limitations when it comes to finding global solutions, despite being the most popular optimization methods for training neural networks. Techniques for global searches have been recognized as a possible fix for this issue. There are several other methods available. It is also possible to employ the GA and SA, two well-known global search methods to optimize the performances of neural networks. The vast majority of these applications have optimized the networks using backpropagation (BP), a gradient approach, due to its ease of use. While backpropagation has clearly played a significant role in the success of neural network applications, its performance is erratic and unpredictable. Recent research has shown that the genetic algorithm can produce better neural network optimization results than backpropagation for a range of complex functions. Similarly, Tabu Search (TS), another global search heuristic, has been found to consistently produce better neural network optimization results than backpropagation. Simulated annealing (SA) is a well-known global search heuristic in addition to GA and TS. It has been demonstrated that simulated annealing works well for optimizing a wide range of complicated issues. The aforementioned assessment procedure leads to the conclusion that, despite backpropagation being the most widely used neural network optimization algorithm, local solutions are most likely to be found. They demonstrated that the global search technique, simulated annealing outperforms point-to-point backpropagation. Numerous user-defined parameters in both BP and SA could have a big impact on the answer. The outcome of the solution is determined by chance because there are no set guidelines for choosing these parameters. The genetic algorithm appears to be capable of routinely outperforming simulated annealing in neural network optimization. The researcher receives better estimations of the interpolation data from these solutions. The genetic algorithm can exclude possible local solutions and arrive at superior answers more quickly and efficiently because to its process of switching between populations of points. In order to improve neural networks, the present review has considered the optimization of a variety of parameters, including weight, initial weight, bias, learning rate, number of hidden layers and nodes therein, and activation functions. Alternatively, the neural network's regular algorithms might be changed to train the enhancement. For instance, the feed-forward or backpropagation algorithms could be swapped out to adjust the network weights according to the error rate per epoch.

REFERENCES

1. Nikola K. Kasabov, Foundations of Neural Networks, Fuzzy Systems, and Knowledge Engineering, A Bradford Book, The MIT Press, Cambridge, Massachusetts, London, England, 1998. ISBN 0-262-11212-4.
2. Simon Haykin, Neural Networks: A Comprehensive Foundation, 2nd Edition, Pearson Education, 1994.
3. M. Tanhaeean, S. F. Ghaderi, and M. Sheikhalishahi, Opti- mization of backpropagation neural network models for relia- bility forecasting using the boxing match algorithm: electro- mechanical case, Journal of Computational Design and Engineering, 2023, 10, 918–933, DOI: 10.1093/jcde/qwad032.

4. Mohammad Valpour, Optimization of neural networks for precipitation analysis in a humid
5. region to detect drought and wet year alarms, Meteorological Applications, 2016, 23: 91–100, DOI: 10.1002/met.1533.
6. A.S. Weigend, Time Series Prediction: Forecasting The Future and Understanding the
Past, 1st ed., Routledge, 1994. https://doi.org/10.4324/9780429492648.
7. Thieu Nguyen, Tu Nguyen, Binh Minh Nguyen, and Giang Nguyen, Efficient Time-Series Forecasting Using Neural Network and Opposition-Based Coral Reefs Optimization, International Journal of Computational Intelligence Systems, Vol. 12(2), 2019, pp. 1144–1161, DOI:https://doi.org/10.2991/ijcis.d.190930.003.
8. G. Shrivastava, S. Karmakar, and M.K. Kowar, Application of Artificial Neural Networks in Weather Forecasting: A Comprehensive Literature Review, International Journalof Computer Applications, 2012, 51, pp. 17-29.
9. R. Abdulkadirov, P. Lyakhov, and N. Nagornov, Survey of Optimization Algorithms in Modern Neural Networks, Mathematics, 2023, 11, 2466. https://doi.org/10.3390/math11112466.
10. M.J.C. Hu, Application of ADALINE system to weather fore-casting, Technical Report, Stanford Electron, 1964.
11. M.N. French, W. F. Krajewski, and R. R. Cuykendall, Rain- fall forecasting in space and time using neural network, J. Hydrol., 137, 1–31, 1992.
12. Tao Chen and M. Takagi, Rainfall prediction of geostationary meteorological satellite images using artificial neural net-work, IGARSS, 1993, vol. 3, 1247 – 1249.
13. M. Zhang and A. R. Scofield, Artificial Neural Network techniques for estimating rainfall and recognizing cloud merger from satellite data, International Journal of Remote Sensing, 16, 3241-3262, 1994.
14. S. C. Michaelides, C. C. Neocleous, and C. N. Schizas, Artificial neural networks and multiple linear regressions in esti-mating missing rainfall data, DSP95 International Conference on Digital Signal Processing, Limassol, Cyprus, 1995,pp. 668-673.
15. S. A. Kalogirou, C. Neocleous, C. N. Constantinos, S. C. Michaelides, and C. N. Schizas, A time series reconstruction of precipitation records using artificial neural networks, EU-FIT '97 Conference, 8–11 September, Aachen, Germany, pp. 2409–2413, 1997.
16. C. Venkatesan, S.D. Raskar, S.S. Tambe, B.D. Kulkarni, and
R.N. Keshavamurty, Prediction of all India summer monsoon rainfall using error-back-propagation neural networks, Meteorol. Atmos. Phys., 62 (3–4), 225–240, 1997.
17. S. Lee, S. Cho, and P.M. Wong, Rainfall prediction using artificial neural network, J. Geog. Inf. Decision Anal., 2, 233–242, 1998.
18. Manfred M. Fischer and Petra Staufer, Optimization in an Error Backpropagation Neural Network Environment with aPerformance Test on a Spectral Pattern Classification Problem, Geographical Analysis, Vol. 31, No. 2, pp. 90-108, 1999.
19. H. Gao and L. Zheng, Impact of Regularization Techniques on ANN for Rainfall Forecasting, Environmental Modelling Software, 117, 1-10, 2019.
20. E. Toth, A. Brath, and A. Montanari, Comparison of short term rainfall prediction models for real-time flood forecast-ing, Journal of Hydrology, 239: pp. 132–147, 2000.
21. Kin C Luk, J. E. Ball, and A. Sharma, An Application of Artificial Neural Networks for Rainfall Forecasting, Hydro-logical Sciences Journal, 33, pp. 883-699, 2001.
22. Ajith Abraham, Dan Steinberg, and Ninan Sajeeth Philip, Rainfall Forecasting Using Soft Computing Models and Multivariate Adaptive Regression Splines, 2001.
23. K.W. Wong, P.M. Wong, T. D. Gedeon, and C. C. Fung, Rainfall Prediction Using Soft Computing Technique, Soft Computing, 7 (2003), 434 – 438.
24. C.I. Christodoulou, S.C. Michaelides, M. Gabella, and C.S. Pattichis, Prediction of rainfall rate based on weather radar measurements, IEEE, 2004, vol. 2, 1393-1396.
25. P. Guhathakurta, Long-Range Monsoon Rainfall Prediction of 2005 for the Districts and Sub-Divisions of Kerala with Artificial Neural Network, Current Science, 2006, 90 (6), pp. 773-779.
1.
26. V. K. Somvanshi, O. P. Pandey, P. K. Agrawal, N.V. Kalanker, M. Ravi Prakash, and Ramesh Chand, Modeling and prediction of rainfall using artificial neural network and ARIMA techniques, J. Ind. Geophys. Union, 2006, 10, No.2, pp. 141-151.
27. M. Nasseri, K. Asghari, and M. J. Abedini, Optimized scenario for rainfall forecasting using genetic algorithm coupled with artificial neural network, Expert Systems with Applications, 2008, 35, pp. 1415–1421.
28. Afshin Sarah, Fahmi Hedayat, Alizadeh Amin, Sedghi Hus- sein, and Kaveh Fereidoon, Long term rainfall forecasting by integrated artificial neural network-fuzzy logic-wavelet model in Karoon basin, Scientific Research and Essays, 2011, 6(6), pp. 1200-1208.
29. Rohit R. Deshpande, On the Rainfall Time Series Prediction Using Multilayer Perceptron Artificial Neural Network, IJETAE, 2012. 2 (1).
30. Renqian Luo, Fei Tian, Tao Qin, Enhong Chen, and Tie- Yan Liu, Neural Architecture Optimization, NeurIPS 2018, Curran Associates, Montréal, Canada, 7827–7838.
31. Zhi-Cheng Zhang, Xin-Min Zeng, Gen Li, Bo Lu, Ming-Zhong Xiao, and Bing-Zeng Wang, Summer Precipitation Forecast Using an Optimized Artificial Neural Network with a Genetic Algorithm for Yangtze- Huaihe River Basin, China, Atmosphere, 2022, 13, 929. https://doi.org/10.3390/atmos1306092.
32. A.M. González-Zapata, E. TleloCuautle, and I. Cruz-Vega, On the Optimization of Machine Learning Techniques for Chaotic Time Series Prediction, Electronics, 2022, 11, 3612. https://doi.org/10.3390/electronics11213612.

33. M. Qiao, Y. Liang, A. Tavares, and X. Shi, Multi- layer Perceptron Network Optimization for Chaotic Time Series Modeling, Entropy, 2023, 25(7):973. https://doi.org/10.3390/e25070973.
34. K. Varada Rajkumar and Dr. K. Subrahmanyam, A Novel Method for Rainfall Prediction and Classification using Neu- ral Networks, International Journal of Advanced Computer Science and Applications (IJACSA), Vol. 12, No. 7, 2021.
35. Maher G. M. Abdolrasol, S. M. Suhail Hussain, and others, Artificial Neural Networks Based Optimization Techniques: A Review, Electronics, 2021, 10, 2689. https://doi.org/10.3390/electronics10212689.
36. Chalachew Muluken Liyew and Haileyesus Amsaya Melese, Machine learning techniques to predict daily rainfall amount, Journal of Big Data, https://doi.org/10.1186/s40537-021- 00545-4.
37. Hongfei Zhu, Jorge Leandro, and QingLin, Optimiza- tion of Artificial Neural Network (ANN) for Maxi- mum Flood Inundation Forecasts, Water, 2021, 13, 2252. https://doi.org/10.3390/w1316225.

38. Nisha Thakur, Sanjeev Karmakar, and Sunita Soni, Rain- fall Forecasting Using Various Artificial Neural Net- work Techniques – A Review, International Journal of Scientific Research in Computer Science, Engineering and Information Technology (IJSRCSEIT), DOI: https://doi.org/10.32628/CSEIT2173159.
39. Junxi Zhang and Shiru Qu, Optimization of Back- propagation Neural Network under the Adaptive Genetic Algorithm, Journal of Applied Engineering Science, Volume 2021, Article ID 1718234, 9 pages, https://doi.org/10.1155/2021/1718234.
40. R. Kumar and A. Patel, Hybrid ANN-SVM Model for Accurate Rainfall Prediction, Meteorological Applications, 28(1), e1976, 2021.
41. M. Tamilmani and M. Sughasiny, Optimized Artificial Neural Network Classifier for The Prediction Of Rainfall, Inter- national Journal of Computational Intelligence in Control, Vol.13 No. 2, 2021.
42. Urooj Kaimkhani, Bushra Naz, and Sanam Narejo, Rain- fall Prediction Using Time Series Nonlinear Autoregressive Neural Network, SSRG International Journal of Computer Science and Engineering, Volume 8, Issue 1, 2021.
43. Mohamed Reyad Amany M. Sarhan, A modified Adam algorithm for deep neural network optimization,Neural Computing and Applications, 2023, 35:17095–17112. https://doi.org/10.1007/s00521-023-08568.
44. J. Smith and H. Wang, Optimizing Neural Network Architectures with Genetic Algorithms for Rainfall Prediction, Ap- plied Soft Computing, 94, 106478, 2020.
45. Budi Raharjo, Nurul Farida, and others, Optimization Forecasting Using Back-Propagation Algorithm, Journal of Applied Engineering Science, 19(2021)4, 889, 1083-1089, DOI:10.5937/jaes0-30175.
46. Abhishek K., Kumar A., Ranjan R., and Kumar S., A rain- fall prediction model using artificial neural network, In 2012 IEEE Control and System Graduate Research Colloquium, 2012 Jul 16, pp. 82-87. IEEE.
47. Nayak D. R., Mahapatra A., and Mishra P., A survey on rain- fall prediction using artificial neural network, International Journal of Computer Applications, 2013 Jun, 72(16): 32-40.
48. Mishra N., Soni H. K., Sharma S., and Upadhyay A. K., Development and analysis of artificial neural network models for rainfall prediction by using time-series data, International Journal of Intelligent Systems and Applications, 2018, 10(1), pp. 16-23.
49. Xiang Y., Gou L., He L., Xia S., and Wang W., A SVR–ANN combined model based on ensemble EMD for rainfall prediction, Applied Soft Computing, 2018, 73, pp. 874-883.
50. Dada Emmanuel Gbenga, Hurcha Joseph Yakubu, and David Opeoluwa Oyewola, Artificial neural network models for rain- fall prediction, European Journal of Electrical Engineering and Computer Science, 2021, 5(2), pp. 30-35.
51. Wu C. L., and Chau K. W., Prediction of rainfall time series using modular soft computing methods, Engineering Appli- cations of Artificial Intelligence, 2013, 26(3), pp. 997-1007.
52. Venkata Ramana R., Krishna B., Kumar S. R., and Pandey N. G., Monthly rainfall prediction using wavelet neural net- work analysis, Water Resources Management, 2013, 27, pp. 3697-3711.
53. El-Shafie A., Mukhlisin M., Najah A. A., and Taha M. R., Performance of artificial neural network and regression techniques for rainfall-runoff prediction, International Journal of Physical Sciences, 2011, 6(8), pp. 1997-2003.
54. Somvanshi V. K., et al., Modeling and prediction of rainfall using artificial neural network and ARIMA techniques, J. Ind. Geophys. Union, 2006, 10(2), pp. 141-151.
55. Dash Yajnaseni, Saroj K. Mishra, and Bijaya K. Panigrahi, Rainfall prediction for the Kerala state of India using artificial intelligence approaches, Computers Electrical Engineering, 2018, 70, pp. 66-73.