mathematics

Review

Literature Review on Integrating Generalized Space-Time Autoregressive Integrated Moving Average (GSTARIMA) and Deep Neural Networks in Machine Learning for Climate Forecasting

Devi Munandar [1,*] ![ORCID]**, Budi Nurani Ruchjana** [1] ![ORCID]**, Atje Setiawan Abdullah** [2] **and Hilman Ferdinandus Pardede** [3] ![ORCID]

1   Department of Mathematics, Faculty of Mathematics and Natural Sciences, Universitas Padjadjaran, Bandung 40132, Indonesia; budi.nurani@unpad.ac.id
2   Department of Computer Science, Faculty of Mathematics and Natural Sciences, Universitas Padjadjaran, Bandung 40132, Indonesia; atje.setiawan@unpad.ac.id
3   Research Center for Artificial Intelligence and Cybersecurity, National Research and Innovation Agency (BRIN), Jakarta Pusat 10340, Indonesia; hilm003@brin.go.id
*   Correspondence: devi19010@mail.unpad.ac.id

Abstract: The issue of climate change holds immense significance, affecting various aspects of life, including the environment, the interaction between soil conditions and the atmosphere, and agriculture. Over the past few decades, a range of spatio-temporal and Deep Neural Network (DNN) techniques had been proposed within the field of Machine Learning (ML) for climate forecasting, using spatial and temporal data. The forecasting model in this paper is highly complex, particularly due to the presence of nonlinear data in the residual modeling of General Space-Time Autoregressive Integrated Moving Average (GSTARIMA), which represented nonstationary data with time and location dependencies. This model effectively captured trends and seasonal data with time and location dependencies. On the other hand, DNNs proved reliable for modeling nonlinear data that posed challenges for spatio-temporal approaches. This research presented a comprehensive overview of the integrated approach between the GSTARIMA model and DNNs, following the six-stage Data Analytics Lifecycle methodology. The focus was primarily on previous works conducted between 2013 and 2022. The review showed that the GSTARIMA–DNN integration model was a promising tool for forecasting climate in a specific region in the future. Although spatio-temporal and DNN approaches have been widely employed for predicting the climate and its impact on human life due to their computational efficiency and ability to handle complex problems, the proposed method is expected to be universally accepted for integrating these models, which encompass location and time dependencies. Furthermore, it was found that the GSTARIMA–DNN method, incorporating multivariate variables, locations, and multiple hidden layers, was suitable for short-term climate forecasting. Finally, this paper presented several future directions and recommendations for further research.

Keywords: integration; GSTARIMA; deep neural network; data analytics lifecycle; forecasting; climate

MSC: 68T07; 68T09

1. Introduction

The climate is a long-term average weather condition in a specific area or zone, and it is determined by the climatic system of that region, such as the atmosphere. A spatio-temporal model can observe and analyze various aspects based on location and time, and it is integrable with other research fields for examining climate phenomena and developing decision-making strategies. This model focuses on the sequence of events that can be observed and identified according to their location and time. In the spatio-temporal model,

the Box–Jenkins spatial model and time series are combined [1]. One commonly studied model in this context is the Space-Time Autoregressive (STAR) model, which incorporates a spatial lag operator, representing the effect of nearest neighbors on a particular spatial location through weights. It was the first spatio-temporal model to be introduced, but it is only applicable to homogeneous locations, assuming the same parameters for each area [2]. To address this limitation, the Generalized Space-Time Autoregressive (GSTAR) model is developed, which is a natural extension of the STAR model. It allows autoregressive parameters to vary across locations, making it suitable for analyzing heterogeneous sample-site characteristics [3]. Furthermore, the Generalized Space-Time Autoregressive Integrated Moving Average (GSTARIMA) model is specifically designed for analyzing nonstationary spatio-temporal data [4].

Climate and air pollution are closely related, as climate change can significantly impact air quality. Previous research, such as that focused on the GSTARMA model, has focused on improving predictions of pollutant datasets within spatio-temporal frameworks [5]. These predictions are crucial for understanding the economic and societal impacts of air pollution [6]. Furthermore, the influence of rainfall, as a climate factor, can be estimated using GSTAR to plan rice planting seasons, which vary across different sites. GSTARIMA is employed to analyze the distribution of yields and determine pricing based on the transfer function [7–9]. These applications extend beyond agriculture, encompassing various sectors [10]. Moreover, unobserved locations can be predicted using GSTAR–Kriging [11,12]. When modeling temperature and forecasting nonlinear time-series data with spatial dependency, the STARMA–GARCH hybrid model outperforms STARMA in terms of modeling efficiency and forecasting accuracy [13]. Lastly, the effect of exogenous climatic factors on disease prediction during seasons with spatio-temporal variation can be examined using the Seasonal Difference Space-Time Autoregressive Integrated Moving Average (SD–STARIMA) approach. This approach is specifically designed to analyze spatio-temporal series data exhibiting seasonal distribution features [14], and can even generate a statistical spatio-temporal model [15] for forecasting photovoltaic plant electricity output in the near future [16].

In previous research, Deep Neural Networks (DNNs) have been used as part of Deep Learning (DL), which is a widely recognized approach for integrating climatological data. The supervised Convolutional Neural Network (CNN) algorithm technique has been proven to be highly effective in handling complex climate and environmental data, particularly nonlinear data; an example is the CNN forecast system for the prediction of the El Niño model [17]. This has achieved a training accuracy of up to 94% [18], making it particularly suitable for datasets containing multiple time series of high-dimensional climate variables and multidimensional spatial series with undetermined sequences [19]. The accurate prediction of climate variables is crucial for social and economic activities, and the CNN–LSTM hybrid model outperforms traditional Machine Learning (ML) approaches in predicting rainfall for the next three hours [20]. This hybrid model has also proven to be effective in analyzing air pollution using Deep Learning [21,22] and in large storms such as typhoons [23]. Additionally, wind speed and meteorological conditions can be predicted using the multilayer perceptron technique [24,25]. The characteristics of big data, including large volumes, various variables, and rapid data growth, significantly impact predictions in regional tourism management. ML, particularly the random forest algorithm, plays a vital role in increasing tourist visits by providing accurate predictions [26]. Meteorological forecasting, which includes enormous and complicated datasets, is intrinsically related to mitigating environmental consequences on daily activities [27].

The integration of the Space-Time and Neural Network (NN) models as part of ML is an emerging trend. These two models are being developed both independently and in hybrid forms to analyze climate data. To forecast air quality and pollution, GSTAR has been integrated with the NN model [28], and the Generalized Regression Neural Network (GRNN) model has been employed to predict solar radiation, enabling the identification of outlier data [29]. Nonlinear patterns can be generated and effectively addressed using

spatio-temporal techniques in conjunction with NNs [30]. When downscaled weather and climate data simulations of summer monsoon rainfall are conducted, a deep convolutional architecture with data-type dependencies is selected for forecasting, as no single optimum architecture exists for all variables [31]. DNN has a significant impact on the predictive analysis of nonlinear data because cluster techniques provide relatively accurate results [32]. Figure 1 shows the rapid growth of research on spatio-temporal and DL in the context of big climate data.
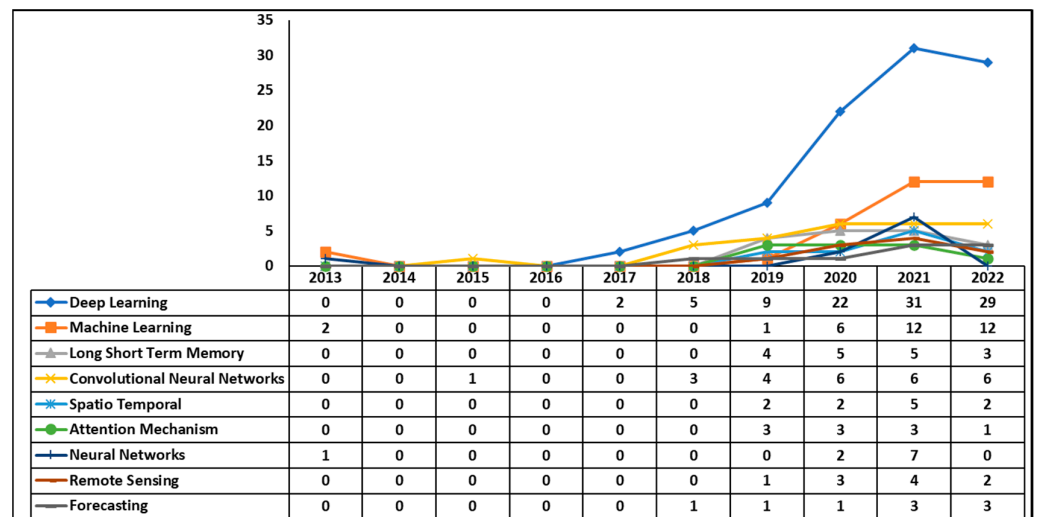

	2013	2014	2015	2016	2017	2018	2019	2020	2021	2022
Deep Learning	0	0	0	0	2	5	9	22	31	29
Machine Learning	2	0	0	0	0	0	1	6	12	12
Long Short Term Memory	0	0	0	0	0	0	4	5	5	3
Convolutional Neural Networks	0	0	1	0	0	3	4	6	6	6
Spatio Temporal	0	0	0	0	0	0	2	2	5	2
Attention Mechanism	0	0	0	0	0	0	3	3	3	1
Neural Networks	1	0	0	0	0	0	0	2	7	0
Remote Sensing	0	0	0	0	0	0	1	3	4	2
Forecasting	0	0	0	0	0	1	1	1	3	3

Figure 1. Cumulative research topics regarding spatio-temporal, neural network, and machine learning models on climate data from 2013 to 2022.

The purpose of this research is to compile previous findings on forecasting spatio-temporal and DNN models applied to climate data. It aims to cover topics such as spatial models for stationary and nonstationary data, methods for estimating location and time parameters using various approaches, supervised ML algorithms, and the potential for integrating spatio-temporal and DNN models in climate forecasting. The findings from this research will contribute to future climate forecasting efforts, with a focus on multivariate time-series big data and the use of different DNN algorithms to enhance the accuracy and timeliness of predictions. It provides an overview of published papers on integrated Generalized Space-Time ARIMA with DNN forecasting. The examination of climatic big data datasets employs the Data Analytics Lifecycle to gain insights. To guide the analysis, the following three research questions (RQ) have been developed:

- RQ1: How does the integration of the GSTARIMA–DNN model using the ML technique work?
- RQ2: How does the integration of the GSTARIMA–DNN model utilizing ML contribute to climate data forecasting?
- RQ3: How does the GSTARIMA–DNN model compare to the GSTARIMA model in forecasting climate data?

This literature review is organized into five sections, with the first providing an overview of its purpose. In the second section, information from previous research on spatio-temporal modeling and DL in the context of climate is systematically identified. The third focuses on traditional mathematical and statistical modeling methods. The fourth explores the potential of integrating spatio-temporal and DNN models with big data concepts for future climate forecasting. Lastly, the fifth section provides a conclusion by analyzing the developed model and its ramifications.

2. Materials and Methods

2.1. Literature Review and Information Analysis

This research focused on the implementation of GSTARIMA model with ML using a DNN. Although there is considerable research on of GSTARIMA and NN and their implementation, it remained incomplete. The integration of GSTARIMA–DNN using an ML approach had not been extensively explored. To highlight the novelty of this research, a literature search was conducted using search engines such as Scopus, Web of Science, EBSCO, Dimensions, and other academic research sources (Other Sources). The search followed the Preferred Reporting Items for Systematic Reviews and Meta-Analyses (PRISMA) approach.

PRISMA serves as an evidence-based minimum reporting standard for systematic reviews and meta-analyses. It is also beneficial for peer reviewers and editors [33] when critically assessing published systematic reviews. In addition, a bibliometric approach was employed, and despite its limited exploration, the method held significant theoretical and empirical value [34]. Bibliometrics involves studying the scientific literature [35], utilizing mathematics and statistics as connection tools. This acted as a supporting instrument for parameterizing and evaluating scientific outputs [36]. The approach incorporated the latest developments in scientific disciplines by reviewing related experts, cited publications, journals, and countries. It enabled mapping papers based on scientific knowledge; analyzing authors, journals, institutions, articles, and countries using keyword searches and several citations; identifying novel research topics [37]; and conducting comprehensive literature reviews [38]. Additionally, it served as a medium for carrying out the complex literature review method in scientific fields [39].

The bibliographic survey in this research involved searching for the published literature that supported spatio-temporal modeling, namely GSTAR, and research on DNNs in ML. The search focused on peer-reviewed journals published in international languages. The results were saved in various formats (CSV, BIB, RIS, CIW), with a dedicated format reader.

The determination of keyword A in Table 1 of the main paper corresponded to the research topic of spatio-temporal and NN models. This was supported by keywords in the paper and aligned with the research title on the Space-Time model [3,4,40]. Keyword B referred to the concept of a DNN using the MLP and CNN algorithms for analyzing time-series data in climate research [41–43]. Keyword C encompassed the concept of the Data Analytics Lifecycle methodology, the use of climate variables that impacted the environment, and the derived theory of the backpropagation algorithm [44–46].

Table 1. The mining of databases spatio-temporal and DNN using a machine learning approach.

Code	Keywords	Scopus	Web of Science	Dimensions	EBSCO	Total
A	("GSTAR" OR "Generalized Space-Time Autoregressive" OR "Spatio Temporal") AND ("Machine Learning" OR "Deep Learning" OR "Multivariate Time Series")	4556	2833	2312	1182	10,883
B	("Neural Network" OR "Deep Neural Network" OR "Feed Forward Neural Network" OR "Multilayer Perceptron" OR "Convolutional Neural Network" OR "Autoregressive Integrated Moving Average")	796,162	489,068	226,657	163,955	1,675,842
C	("Data Analytics Lifecycle" OR "Climate" OR "Weather" OR "Function Derivative Approximation" OR "Ordinary Differential Equation")	1,108,037	1,032,136	619,507	585,205	3,344,885
D	A AND B AND C	250	141	138	46	575

A literature search was conducted to identify papers that included the main keywords, such as "Spatio Temporal", "Generalized Space-Time Autoregressive", "Machine Learning", "Multivariate Time Series", "Data Analytics Lifecycle", and "Deep Learning". These keywords were obtained from several research papers discussing the basic concepts and their application to the research object. The search engine was used to find papers based on keywords and applied inclusion restrictions such as year, language, and article type. All the papers obtained were sourced from the specified database portals.

2.2. Dataset Analysis

Figure 2 shows the systematic literature process using the PRISMA flow diagram, which consisted of three stages, namely Identification, Screening, and Included. In the Identification stage, articles were obtained from four search databases using code D, as well as other academic research sources that supported this investigation.
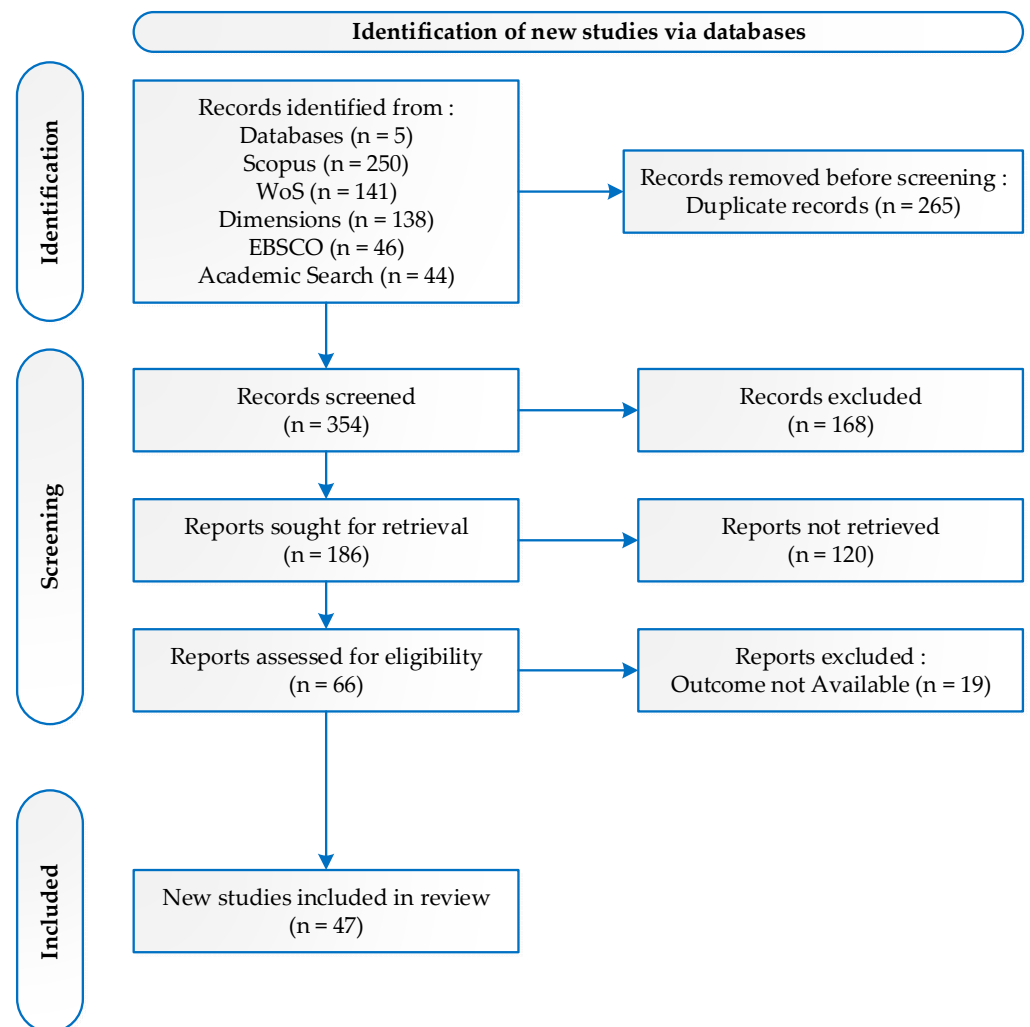

Figure 2. Systematic literature review selection was used in this research.

The analysis using the PRISMA method began with determining keywords related to the topic of this proposed research. The keywords were generated from the keywords of several papers, which are the main studies described above. After obtaining the keywords shown in Table 1, in stage 2, the search (to process code D) was based on four databases and papers on academic references (Table 1). In stage 3, 575 papers were obtained through an inclusion search strategy by limiting the paper years to 2013–2022, the paper types to articles and proceedings, and the language used to English. Stage 4 removed duplicates

of papers from the four search databases using reference management software; we used Mendeley to analyze them. After removing 265 duplicated articles, the screening stage led to 354 articles. From these, 168 articles were excluded as they did not match the title and supporting abstract. In the final stage for appropriate topics, we selected 47 papers as research references by conducting content analysis to find gaps in the research that we would conduct.

At the Included stage, 47 articles were obtained and analyzed for this research. Figure 3 shows the data analysis findings, divided into 4 clusters comprising 25 items. The search keywords related to DL had 11 occurrences (cases) in 2021.
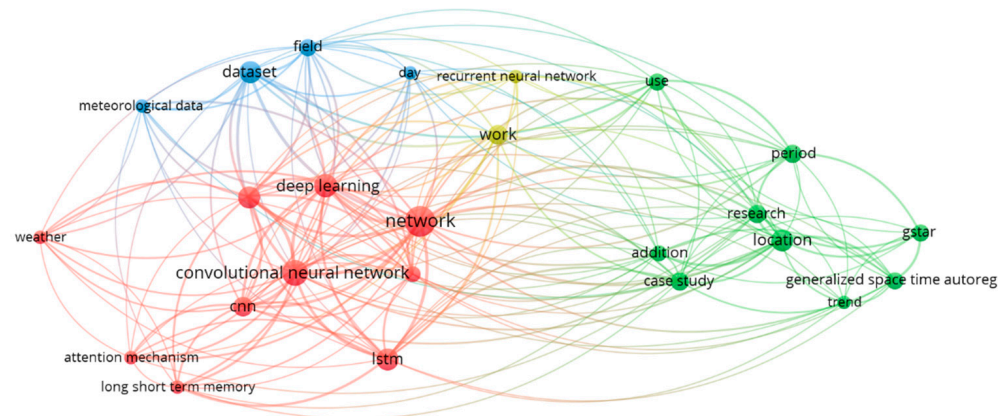

Figure 3. Network visualization article search analysis with VOSViewer.1.6.15.

Meanwhile, in the case of GSTAR research, there were six occurrences in 2018; CNN had thirteen, and Meteorological Data had a total of four in the 2021 publication. The visualization revealed that GSTAR research still had gaps and distant clusters from DL and CNN, which formed the basis for the analysis conducted in this research. Despite the availability of datasets, utilizing GSTARIMA for spatio-temporal modeling to evaluate data based on location and time was still frequent, particularly when integrated with DNN.

2.3. Theoretical Background

2.3.1. Space-Time Autoregressive (STAR)

Time-series models that incorporated univariate and multivariate periods could be observed in the ARIMA and Vector Autoregressive (VAR) models. On the other hand, the Space-Time model combined elements of location and time in a multivariate time series. It was first introduced by Pfeifer and Deutsch [2] and is known as STAR. The STAR model assumes the same parameters for all locations and was used for homogeneous locations. The STAR (*p*) model with order (1) was defined using a spatial lag operator to express the effect of the closest location on a particular spatial lag using weights. This could be formulated as follows:

$$Z_t = \sum_{k=1}^{p} \sum_{l=0}^{\lambda_k} \left[\varphi_{kl} \mathbf{W}^{(l)} Z_{t-k} \right] + e_t \tag{1}$$

where:

λ_k: the spatial order of the autoregressive of the *k*th order

Z_t: vector with size $(n \times 1)$ at time-*t*

Z_{t-k}: vector with size $(n \times 1)$ at a time $(t-k)$

φ_{kl}: the STAR parameter at time-*k* and spatial lag-*l*

$\mathbf{W}^{(l)}$: matrix weight size $(n \times n)$ on spatial lag-*l* (with *l* = 1, 2, . . .), and the weights selected for $w_{ii} = 0$ and $\sum_{i \neq j} w_{ij} = 1$

e_t: error vector with size $(n \times 1)$ at time-*t*, assuming $e_t \overset{iid}{\sim} N(0, \sigma^2 \mathbf{I})$

2.3.2. Generalized Space-Time Autoregressive (GSTAR)

GSTAR, originally developed by Borovkova [3] as a natural generalization of the STAR model, allows autoregressive parameters to vary per location. The GSTAR model (p_{λ_p}) applied to heterogeneous sample-site characteristics and was formulated when the differencing and the Moving Average vector were 0, respectively. The given formula was different in the aspect of the *phi* parameter, of which STAR was a scalar vector quantity while GSTAR was a matrix:

$$Z_t = \sum_{k=1}^{p} \sum_{l=0}^{\lambda_p} \mathbf{\Phi}_{kl} \mathbf{W}^{(l)} Z_{t-k} + e_t \tag{2}$$

where $\mathbf{\Phi}_{kl}$ is the diagonal matrices for the AR parameter at lag-k and spatial lag-l of size $(n \times n)$; $diag(\varphi_{kl}^1, \varphi_{kl}^2, \ldots \varphi_{kl}^n)$.

2.3.3. Generalized Space-Time Autoregressive Integrated Moving Average (GSTARIMA)

The expansion of GSTAR by adding a Moving Average (MA) element to GSTARMA (p_{λ_p}, q_{v_q}), with a differencing of 0, produced:

$$Z_t = \sum_{k=1}^{p} \sum_{l=0}^{\lambda_p} \mathbf{\Phi}_{kl} \mathbf{W}^{(l)} Z_{t-k} + e_t \tag{3}$$

If $\mathbf{Z}_t$ was an observation vector that was not stationary and the differencing process was applied to make $\nabla \mathbf{Z}_t = (1 - B)^d \mathbf{Z}_t$ stationary, the GSTARIMA ($p_{\lambda_p}, d, q_{v_q}$) model could be defined as:

$$\nabla Z_t = \sum_{k=1}^{p} \sum_{l=0}^{\lambda_p} \mathbf{\Phi}_{kl} \mathbf{W}^{(l)} \nabla Z_{t-k} - \sum_{k=1}^{p} \sum_{l=0}^{v_q} \Theta_{kl} \mathbf{W}^{(l)} e_{t-k} + e_t \tag{4}$$

where:

∇Z_t: observation vector with $\nabla Z_t = [\nabla Z_{1,t}, \nabla Z_{1,t}, \ldots, \nabla Z_{n,t}]$ at the time-$t = 1, 2, \ldots, T$ with size ($n \times 1$)

Θ_{kl}: the diagonal matrices for the MA parameter lag-k and spatial lag-l are of size $(n \times n)$; $diag(\theta_{kl}^1, \theta_{kl}^2, \ldots \theta_{kl}^n)$

p: autoregressive vector order (AR)

q: Moving Average vector order (MA)

λ_p: the spatial order-p of the autoregressive

v_q: the spatial-q order of the moving average

2.3.4. Machine Learning (ML)

ML is a decision-making technology adopted through Artificial Intelligence (AI), which is used in all areas of life for basic research and practical applications. The ML approach could be defined as the exploration of computational methods to test the validity of new knowledge and discover novel ways to organize existing knowledge [47].

In addition, ML offers various techniques for problem-solving, particularly in predicting the future. In this technology era, it plays crucial roles in real-time applications, such as business analytics, education, pharmaceuticals/molecular biology, manufacturing, crime detection, financial support, and marketing. The technique is predominantly employed for tackling complex problems, regardless of whether they involve structured or unstructured data. AI facilitates ML in learning from past information by adapting to and extracting valuable insights from large datasets (big data). The incorporation of ML features in data analysis [48] is very important for model development. This technique has permeated various software-based sectors and applications.

In ML, supervised learning is a potent method for classifying labeled/tagged data using learning algorithms, such as regression techniques (Linear, Logistic, Polynomial)

and classification (Random Forest, Support Vector Machine, Linear Discriminant Analysis, K-Nearest Neighbor, NN Classifier), even using ensemble algorithms for algorithms in the learning model [49]. This algorithm focused on qualitative data used in forecasting, considering the labels assigned to the data attributes. Supervised learning-based classification methods were utilized to build the best predictive model.

On the other hand, unsupervised learning categorized and clustered unlabeled data based on similarity. Through this technique, it became possible to discover hidden layers and patterns [50]. This type of learning aided in diverse clustering applications, encompassing numerical and categorical data distribution as well as Dimensionality Reduction (Wavelet Transform, PCA Method).

The semi-supervised learning algorithm combined supervised and unsupervised approaches known as reinforcement learning. In general, this technique addressed problems on a large scale [51]. Reinforcement learning constituted the third paradigm of the ML technique, which differed from other learning methods. It could shape data based on past experiences even when the data were lacking. Learning techniques within this framework were based on sequential decision-making [52].

2.3.5. Multilayer Perceptron (MLP)

The perceptron, which is the basic concept of the NN model, enabled the construction of more complex artificial neuron hierarchies. Figure 4 shows the architecture of the Multilayer Perceptron (MLP). The network had three input signals ($\mathbf{x} \in \mathbb{R} \times \mathbb{R} \times \mathbb{R}$), referred to as the input layer. Input variables were not mapped to neurons but represented real number values. In addition, the network had two output neurons arranged in the output layer. The input and output layers were visible because they were directly connected to the model. All layers in between were called hidden layers and could contain an arbitrary number of neurons. In Figure 4, there are two hidden layers, each with four neurons. The layer with all neurons connected to the preceding ones could be referred to as fully connected. This network topology, denoted as {3, 4, 4, 2} referred to a feedforward NN since the information flowed from the input to the output layer without loops. To mathematically describe NNs, proper notation and indexing schemes were required, with j neurons ($j = 1, 2, \ldots, m_l$) and l layer ($l = 1, 2, \ldots, L$) consisting of m_l neurons from the sum $v_j^{(l)}$ and the activation output $y_j^{(l)} = \varphi(v_j^{(l)})$.
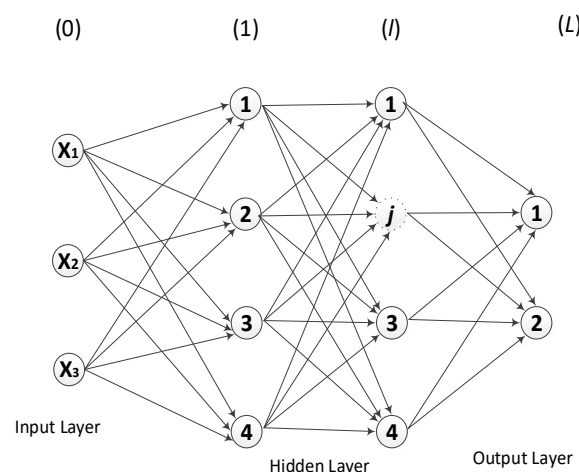

Figure 4. Architecture of multilayer perceptron [53].

The weight $w_{ij}^{(l)}$ represents that the index j is a neuron measuring layer l, and the second index-i shows the output neuron in the previous layer $l - 1$. The first layer $l = 0$ is connected to the input value $y^{(0)} = x_i$. The weight $w_{j0}^{(l)}$ captures the bias value, ensuring a

constant output value $y_0^{l-1} = \varphi(v_0^{(l-1)}) = 1$ is performed for each layer. With this definition, the sum output at layer *l* is calculated as follows:

$$v_j^{(l)} = \sum_{i=0}^{m_{l-1}} w_{ij}^{(l)} y_i^{(l-i)} \tag{5}$$

The vector notation for the weights in layer *l* is obtained as:

$$\mathbf{w}^{(l)} = \left(w_{10}^{(l)}, w_{11}^{(l)}, \dots, w_{m_l, m_{l-1}}^{(l)} \right)^T \tag{6}$$

Also, the vector containing all the weights in their entirety is calculated as:

$$\mathbf{w} = \left(\mathbf{w}^{(1)}, \mathbf{w}^{(2)}, \dots, \mathbf{w}^{(L)} \right)^T \tag{7}$$

Equation (6) states that $w_{10}^{(l)}$ is the weight in the lth hidden layer, the data input-1, and the neuron-0. Moreover, $w_{11}^{(l)}$ is the weight on the hidden layer to-*l*, input data to-1 and neuron-1, continuing until weight-*l*, hidden layer-*ml*, and neuron(*ml*-1). Equation (7) states the weight vector for each hidden layer (1), (2), . . . , (L).

2.3.6. Convolutional Neural Network (CNN)

A CNN refers to a specialized type of MLP widely used for computer vision tasks such as image classification and time-series analysis. The network is effective in qualitative data analysis, particularly when dealing with numerical data for training and testing predictions of historical data. A CNN functions by sliding a window over the data matrix. In this review, convolution layers with multiple filters were applied, followed by an activation function for classification, and the stages are shown in Figure 5. The CNN's architecture consisted of three layers, namely convolution, pooling, and fully connected. To obtain the highest level of accuracy, the existing parameters were tested. In essence, the convolution layer modified the weights of the filtered neuron layer to produce output. The defined filters moved horizontally and vertically across each input vector matrix, generating feature maps. Padding with a value of 0 was applied to maintain the maximum length of the input. Equation (8) presents the process in the convolutions layer:

$$x_j^l = f\left(\sum_{i=1,\dots,s} x_i^{l-1} * k_{ij}^l + b_j^l \right), j = 1, \dots, M \tag{8}$$

x_j^l is a dependent variable of a th convolution, * is a convolution operator that multiplies the input by the kernel. x_i^{l-1} is the ith input vector, k_{ij}^l is the kernel received from *j* through filtration combined with the *i*th feature map, while b_j^l is related to the *j*th input filter.
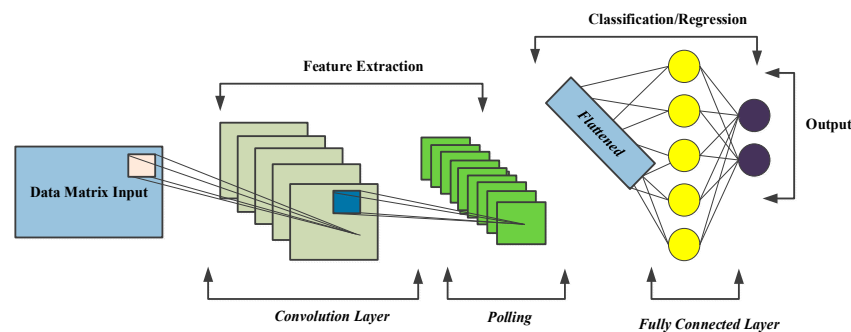

Figure 5. The Architecture of Convolutional Neural Networks (CNNs).

M represents the output result of a feature, while the function $f(x)$ serves as the activation function used in the CNN. It includes the Rectified Linear Unit (ReLU) in the convolutional layer and the sigmoid function in the output classification [54].

$$f(x) = \max(0, x), \; f'(x) = \begin{cases} 0, \; x < 0 \\ 1, \; x > 0 \end{cases} \tag{9}$$

The pooling layer was an integral part of the CNN, serving to reduce the input dimensionality by decreasing the number of parameters from the convolution layer. The pooling method selected the maximum value from each vector as a feature using max pooling. The equation for this layer was as follows:

$$x_j^l = f\left(\psi_j^i(x_i^{l-1}) + b_j^l\right), j = 1, \dots, M \tag{10}$$

Variable representation with value pooled as a feature of the input, f was the activation function in the input processing. Meanwhile, ψ_i^l represented the bias set multiplied by b_j^l which corresponded to the jth input bias from the convolution filter results. M indicated the output value containing the classification features required to achieve the desired model output.

$$x_j^l = f\left(\sum\nolimits_{i=1,\dots,s} x_i^{l-1} w_{ij}^l + b_j^l\right), j = 1, \dots, M \tag{11}$$

The CNN included a fully connected layer, x_j^l as shown in Equation (11). The kernel contained in the convolution layer equation was replaced by the multiplication of the weights w_{ij}^l, which represented the process of obtaining the jth output value associated with the ith input variable. Nonlinear activation functions were necessary for this layer to produce the desired output as predictive classes. This layer was the final process for generating classification classes from the CNN [55]. The use of the CNN architecture aimed to facilitate and avoid trial and error by leveraging well-established and tested models, such as LeNet-5, utilized for image detection [56], or VGGNet, introduced by Simonyan and Andrew Zisserman from the University of Oxford in 2014.

2.3.7. Data Analytics Lifecycle

The Data Analytics Lifecycle was designed to address the challenge of big data and Data Science, including large data volumes, diverse data structures, and rapid data growth. This lifecycle consisted of six stages, which could occur simultaneously in certain cases. In most conditions, the analysis could progress both forward and backward, allowing for an iterative approach that accommodated new information as it became available [57]. This enabled problem-solving and moving through the process iteratively, also facilitating the operationalization of research goals. The Data Analysis Lifecycle established best practices for the analytical process, spanning from discovery to the completion of the research work.

The overview of the Data Analysis Lifecycle, spanning six stages, is presented as follows:

- Stage 1—Discovery (Problem Formulation): This stage involved conducting a literature review to prepare for problem analysis in research. It entailed gathering resources such as references, technology, time, and data. The important activities in this stage included creating a problem framework as an analytical challenge to be addressed in the next stage and formulating initial hypotheses to test and explore the data.
- Stage 2—Data Preparation: Data pre-processing was carried out in this stage, involving initial data analysis. It encompassed processes such as data cleaning, extraction, transformation, and integration, preparing the data to be collected in the database repository as a prerequisite for model preparation.
- Stage 3—Model Planning: This stage focused on planning the model by determining the methods, techniques, and research flow to be followed during the model-building stage.

- Stage 4—Model Building: At this stage, the research was directed towards developing datasets for training purposes, testing, and producing output models. Consideration was given to whether the existing device supported running the model efficiently, such as fast hardware and parallel processing capabilities.
- Stage 5—Communicating Results: This stage involved testing the data model and its changes with the user or in a laboratory setting, to determine whether the output aligned with the development criteria. If the model did not meet the criteria, an evaluation was conducted, and the process could return to the previous stage for further refinement.
- Stage 6—Operationalizing (Operationalization): This stage entailed submitting the final report, directions, codes, and technical documents. In addition, it could involve implementing the model as a pilot project to ensure a broader application.

The Data Analysis Lifecycle could be repeated from steps 1 to 5 if further improvements were required. The evaluation of the modeling process, from steps 6 to 1, was indicated by dotted lines, highlighting the possibility of revisiting certain stages if the modeling results did not meet the desired criteria.

3. Results

A total of 47 appropriate research studies were selected for additional investigation based on the selection criteria. Table 2 shows the distribution of spatio-temporal and DL methods used in the selected papers. The most widely applied models were DNN and spatio-temporal models—location-based models offering flexibility in multivariate time-series modeling. The traditional DNN model was specifically developed to solve nonlinear time-series data problems. Hybrid combinations such as ConvLSTM, CNN–BiLSTM, ConvGRU, and CNN–SVM were special variants of DL capable of learning long-term dependencies as well as handling nonlinear and nonstationary data problems. The hybrid model had the key advantage of achieving maximum prediction accuracy with minimal error. The CNN and LTSM models had different working approaches in completing the modeling process, with CNN excelling in speed and parallelization, while LSTM operated sequentially.

Regarding the data used, satellite data predominated in the analysis, which could be obtained from various sources such as NASA, ECMWF, NOAA, ENSO, and Himawari-8. Satellite data proved to be highly effective, specifically in areas without manual measurement sensors. Converting satellite images into data matrices enabled complex analysis and minimized missing values at measurement locations.

Table 2. Selected publications of spatio-temporal and DNN models based on ML approach for climate and environmental datasets.

Reference		Model	Dataset	Location	Content Performance Analysis				Application
					R^2	RMSE	MAPE	Accuracy	
(Zheng et al., 2023)	[19]	DN	ECMWF (https://atmosphere.copernicus.eu/ (access on 22 May 2023)	-	-	-	-	RNN = 80.32% SB-DNN = 80.06	Climate prediction
(Jiao et al., 2022)	[58]	SP, DN	Solar Radiation, 17 locations, 48,989 data	Hawaii	-	CNN + LSTM = 0.013, GSINN = 0.0047	2.86 1.06	-	Solar radiation prediction
(Manley et al., 2022)	[26]	ML, BD	Climate data temporal, ~750 locations (2005–2017)	California, USA	0.924	-	-	-	Predict the pattern of suitable recreation in summer
(Nikezić et al., 2022)	[59]	SP, DN, BD	Image Satellite of Aerosol (NASA), 21 months	Earth surface	-	0.3199	-	90%	Aerosol movement forecasting
(X. Zhang et al., 2022)	[27]	SP, ML	LAPS = 2018–2021, ERA5 1979–2021	China and Southeast Asia	-	LAPS Temp = 0.37, v-wind = 0.96 u-wind = 0.88 RH = 0.79 ERA5 Temp = 0.32, v-wind = 0.96 u-wind = 0.87 RH = 0.76	-	-	Weather prediction
(W. Li et al., 2022)	[20]	SP, DN	Rainfall, temp, RH, wind, dewpoint (2013–2019)	Gansu, China	-	-	-	Rainfall (response) variable = 85%	Rainfall prediction
(Kumar et al., 2022)	[13]	SP	NASA: temp max, temp avg (1981–2020)	Bihar, India in 8 locations	-	-	STARMA temp avg = 10.54, temp max = 3.08 STARMA—GARCH temp avg = 10.36%, Temp max = 3.06%	-	Comparison of STARMA–GARCH and STARMA

Table 2. *Cont.*

Reference		Model	Dataset	Location	Content Performance Analysis				Application
					R^2	RMSE	MAPE	Accuracy	
(Zou et al., 2022)	[60]	DN	Image data of wind speed and direction, air density	Croatia (Bruska and Jelinak)	-	Bayesian CNN–BiLSTM dan Vine–GMCM (Bruska = 0.093, Jelinak = 0.100)	-	-	Weather prediction
(Marco et al., 2022)	[61]	SP, ML, BD	Image data, 60 flood events from 1995 to 2020	Venice, Italia (188 rainfall stations)	-		-	Logistic Regression = 0.860, Neural Network = 0.843, Random Forest = 0.844	Prediction of flood mitigation due to climate change
(Y. Li et al., 2022)	[62]	ML	Weather data from 1980 to 2020	Tuojiang river basin, China	-	CNN–RF = 6.29, CNN–SVM = 16.12, CNN = 10.30, RF = 8.12, SVM = 9.20	-	CNN–RF = 0.97, CNN–SVM = 0.85, CNN = 0.94, RF = 0.96, SVM = 0.95	Evaporation predictions affecting the water, carbon, and energy cycles
(Kong et al., 2022)	[63]	SP, RN	Weather data (2015–2015) 1 h interval, prediction every 3 h	Beijing, China (226 observation station)	-	DeepSTF = 2.41, CNNseq2seq = 2.50, AttnSeq2seq = 2.54	-	DeepSTF = 70.03%, CNNseq2seq = 68.41%, AttnSeq2seq = 67.45%	Deep Spatio-Temporal Forecasting (DeepSTF)
(Y. Zhang et al., 2022)	[64]	DN, RN	River flow data (2012 to 2017) 1 h interval.	Humber River, Ontario, Canada	-	LSTM = 8.48, ConvLSTM = 8.73, CNN–LSTM = 9.24, STA–LSTM = 7.99	-	-	Early Warning Flood forecasting
(Orescanin et al., 2022)	[65]	ML	Rainfall Satellite-borne passive microwave (PMW) monthly data (2017–1018)	Orbit di Laut Atlantic	-	-	-	Bay.ResNet56 = 90% Bay.ResNet38 = 93%	Rainfall prediction
(Anshuka et al., 2022)	[66]	SP, DN	Image data NOAA 1980 to 2020	Southwest Pacific. 30,000 measuring stations		Train multivariate SST = 0.49, Test Multivariate SST = 0.6		Mean = 0.75 for 22 locations	Extreme rainfall prediction
(Suhartono et al., 2021)	[67]	GS	CO, PM_{10} from 14 January 2017 to 14 February 2018	Surabaya, Indonesia	-	ARIMA = 0.22, MGSTAR = 4.99	ARIMA = 29.51%, MGSTAR = 116.94%	-	Air pollution prediction

Table 2. *Cont.*

Reference		Model	Dataset	Location	Content Performance Analysis				Application
					R^2	RMSE	MAPE	Accuracy	
(Böhm et al., 2021)	[68]	ML	Numerical conversion image satellite data. 2017–2019	Chili	-	Each location < 40% detected fog frequency	-	-	Fog detection for freshwater sources in desert areas
(Christoforou et al., 2021)	[69]	DN	Daily wind speed data from 5 locations from 1 January 2013 to 31 December 2014	Greece	-	WRF = 2.3, DCNN = 0.997 to 1.803	WRF = 26.14%, DSTNN = 14%	-	Prediction of wind speed for electricity consumption
(Kong et al., 2022)	[63]	SP, DN	Weather data (2015 to 2017) data interval of 1 h	Beijing, China	-	Temp = 2.41	-	Temp = 70.03%, RH = 70.34%, Wind = 84.44%, Wind breeze = 77.05%	Weather Prediction
(Silva et al., 2021)	[70]	SP, DN	Wind speed, temperature, and pluviometry in 2013 to 2016	Brazil	-	LR = 21.73 MLP (10 neuron) = 4.15	-	-	Detect the spread of dengue fever using climate data
(Guillaumin et al., 2021)	[71]	DN	CO_2 gas satellite data, for ~7000 days (20 years)	Image of Earth's sea surface	85.5%	-	-	-	Predict the distribution of CO_2
(Steffenel et al., 2021)	[72]	SP, DN	Ozone data from 1980 to 2019, 6 h interval (ERA5) (58,500 observations)	South America, South Africa, and New Zealand	-	Min = 55.63 Max = 134.83	-	-	Ozone prediction
(Kimura et al., 2021)	[73]	SP, DN	Climate data (1984 to 2020)	Tokachi River, Hokkaido, Japan	LR = 0.744, LSTM = 0.839, LSTM (add data) = 0.871, LSTM (TL) = 0.853	LR = 2.96, LSTM = 2.027, LSTM (add data) = 1.807 LSTM (TL) = 1.933	-	-	Predict the correlation of air temperature with surface water temperature
(Geng et al., 2021)	[74]	SP, DN, RN	ENSO: (CMIP5) (1864 to 2004) for training, (GODAS) (1994 to 2010) for validation	ENSO3.4, Pacific	-	CNN = 0.5603, DC–LSTM = 0.5558	-	-	El Niño and Southern Oscillation (ENSO) Forecasting
(Kumar et al., 2021)	[31]	DN, BD	Rainfall of ERA5 data (1975 to 2009)	India	DeepSD = 67 SRCNN- = 68	-	-	-	Rainfall prediction

Table 2. *Cont.*

Reference		Model	Dataset	Location	Content Performance Analysis				Application
					R^2	RMSE	MAPE	Accuracy	
(Liu et al., 2021)	[75]	SP, ML	Climate data (GRACE dan USGS) (2007 to 2016)	Northeastern United States	$R_{GR} \geq 046$ (t1 = 0.85, t2 = 0.85, t3 = 0.80). t = month	-	-	-	Prediction of groundwater level
(Al-Shargabi et al., 2021)	[76]	DN	Cold energy, heat energy	Qasim Region, Saudi Arabia	DNN, LM algoritma (layer = 2, neuron = 20). 0.99 (train) 0.99 (test)	DNN, LM algoritma (layer = 2, neuron = 20). 0.119 (Heat) 3.604 (Cool)	-	-	Prediction of energy consumption due to climate change
(Adewoyin et al., 2021)	[77]	DN	ERA5 as a target, E-OBS as input	UK	-	all seasons = 3.081, winter = 3.570, spring = 2.504. summers = 2.991, autumns = 3.215	-	-	Climate modeling for flood anticipation due to extreme rains
(Sulistyono et al., 2020)	[10]	GS	Rainfall (2005 to 2015)	East Java—Indonesia	-	Cross-correlation weight = 10.471. cross-covariance weight = 10.433	-	-	Precipitation forecasting
(Akbar et al., 2020)	[5]	GM	CO gas from January to December 2018	Surabaya, Indonesia	-	GSTARMA– OLS = 0.20, GSTAR–OLS = 0.22	-	-	Comparison of GSTAR and GSTARMA
(Rajakumari et al., 2020)	[78]	ML	NO_2, SO_2, and O_3 for 10 years, 6 h intervals	-	-	RNN = 3.378, ARIMA = 2.006	-	-	Air pollution gas prediction model
(Huang et al., 2020)	[79]	DN, RN	Hotspots = 3240 data, coal gas = 1464 data, Lorenz = 3000 data		-	Hot spots = 3.3834, coal gas = 0.0493 Lorentz = 0.0756	Hot spots = 0.0419, coal gas = 0.1094 Lorentz = 0.0135	-	Anticipation of nonlinear growth predictions on energy and weather data
(Chirayath et al., 2020)	[80]	SP, DN	Image data Coral reefs at sea level	Fiji	-	-	-	84.3%	Biodiversity and Ecological predictions

Table 2. *Cont.*

Reference		Model	Dataset	Location	Content Performance Analysis				Application
					R^2	RMSE	MAPE	Accuracy	
(Ziyabari et al., 2020)	[81]	SP, DN	Solar radiation from 2000 to 2017, 30-min intervals (National Solar Radiation Database)	Philadelphia, Pennsylvania	-	ResNet/LSTM (Adam, ReLU) = 0.068	-	-	Predictions on Photovoltaic (PV)
(Zhang et al., 2020)	[82]	SP, ML	Hurricane data (Japanese Himawari-8 satellite)	Beijing-Tianjin-Hebei, China	-	Predictions per 30 min ConvLSTM = 9.232, TrajGRU = 9.117, ConvGRU = 10.24	-	-	Distribution of storm event information
(Chen et al., 2019)	[23]	DN	Weather data area ~1000 km	Western Pacific (WP), Eastern Pacific (EP), dan North Atlantic (NA).	-	-	-	WP = 0.852 EP = 0.780 NA = 0.759	Typhoon intensity forecasting
(Ding et al., 2019)	[83]	SP, RN	Weather data, 3 h interval from May 2002 to January 2018	Stream of the Lech River, Austria	-	FC = 85.74 SVM = 78.82 LSTM = 74.96 STA–LSTM = 66.02		FC = 0.633 SVM = 0.720 LSTM = 0.750 STA–LSTM = 0.807	Forecasting floods in watersheds
(Pusporani et al., 2019)	[84]	ML, GS	air pollution, 2018	Surabaya, Indonesia	-	MGSTAR = 11.37 MGSTAR–FFNN = 5.49 MGSTAR–DLNN = 4.9			Forecasting linear and nonlinear air pollution data
(Thongniran et al., 2019)	[85]	SP, DN	Radar data in coastal bays from 2014 to 2016	Thailand	-	CNN–GRU (U) = 4.509, CNN–GRU (V) = 7.405	-	-	Prediction of sea surface
(Wilms et al., 2019)	[86]	SP, DN	GEFCom dataset. Wind power targets	Australia (10 locations)	Conv LSTM524 = 0.7588, Conv LSTM254 = 0.7688	Conv LSTM524 = 0.1697, Conv LSTM254 = 0.1661	-	-	Forecasting wind power on turbines
(Cui et al., 2019)	[87]	SP, ML	Soil moisture (EC-TEMP sensor) from 2002 to 2015	Tibetan Plateau	0.71	0.05	-	-	Tibetan Plateau humidity forecast
(Zhao et al., 2018)	[14]	SP	HFRS from 2005 to 2014	Hubei Province, China	-	Luotian = 0.004 Zhongxiang = 0.003 Yicheng = 0.001	Luotian = 10.3 Zhongxiang = 13.2 Yicheng = 9.12		Hemorrhagic fever with renal syndrome (HFRS)

Table 2. *Cont.*

Reference		Model	Dataset	Location	Content Performance Analysis				Application
					R²	RMSE	MAPE	Accuracy	
(Saikhu et al., 2018)	[88]	SP, DN	Rainfall, 1983 to 2016	Surabaya, Indonesia	-	RNN Train = 48.69, Test = 94.46	-	-	Rainfall prediction
(Andayani et al., 2018)	[9]	GM, GX	Price of rice (BPS) from January 2007 to December 2014	Java Island, Indonesia	-	GSTARIMA–X = 287.316, GSTAR = 313.872	GSTARIMA–X = 3.059, GSTAR = 2.752		Comparison predictions of GSTARIMA and GSTARIMA–X
(Abdullah et al., 2018)	[11]	GS	Monthly rainfall (1981 to 2016)	West Java, Indonesia	-	-	Majalengka-Kuningan = 8.97%, Majalengka-Ciamis = 12.51%, Kuningan-Ciamis = 7.72%,	-	Rainfall prediction
(Astuti et al., 2017)	[89]	GS	CPO export volume from January 2004 to August 2015	Sumatera, Indonesia	-	-	MSE (uniform weight = 9.30 × 103, Distance weight = 9.66 × 106)	-	Crude Palm Oil (CPO) export volume prediction
(Handajani et al., 2017)	[8]	GS	Rainfall, 2004 to 2015	Central Java, Indonesia	-	Sragen = 155.16 Karanganyar = 179.11 Klaten = 141.70	-	-	Rainfall prediction
(Ippoliti, 2001)	[90]	GS	Sulfur Dioxide (SO₂) from 1 January 1999 to 31 December 1999	Milan, Italia	28 December 1999 to 31 December 1999 are 0.983, 0.855, 0.775, 0.802	-	-	-	Prediction online monitoring of Sulfur Dioxide

Notes: SP = Spatio-Temporal, ML = Machine Learning, DN = Deep Neural Network, RN = Recurrent Neural Network, BD = Big Data, GS = Generalized Space-Time Autoregressive, GM = Generalized Space-Time Autoregressive Moving Average, GX = Generalized Space-Time Autoregressive Integrated Moving Average—Exogenous.

Furthermore, spatio-temporal and DNN models were employed in several articles across various domains, as shown in Table 2. For example, Jiao et al. (2022) [58] utilized three models (CNN, LSTM, GSINN) to forecast solar datasets for 17 locations. Although all models performed well, GSINN yielded the highest performance analysis value. Andayani et al. (2018) [9] employed a spatio-temporal model to compare rainfall at three different locations using GSTARIMA–X and GSTAR. The results showed that GSTARIMA–X exhibited a small error value, highlighting the significant influence of the Moving Average on the modeling. Furthermore, Cui et al. (2019) [87] applied a spatio-temporal hybrid model with ML to forecast soil moisture for detecting the growing season on the Tibetan plateau, achieving a satisfactory R^2 value. Traditional models such as STAR, GSTAR, and GSTARIMA, developed from scratch, also played a promising role in modeling location and time dependencies with stationary and nonstationary data. On the other hand, DL, as a subset of ML, was widely used and reliable for achieving high forecasting accuracy. All the application models employed appropriate methods to select data, although there were variations across different models and contexts (Kumar et al., 2022) [13]; due to the geographical dependency of the time-series data and their nonlinear features, temperature modeling and forecasting is challenging. The authors designed a hybrid model of monthly maximum temperatures and temperature ranges called the Space-Time Autoregressive Moving Average Generalized Autoregressive Conditional Heteroscedasticity (STARMA–GARCH) model. The fitted STARMA model residual was checked for nonlinear behavior. GARCH modeling is, therefore, necessary. To capture the dynamics of monthly maximum temperatures and temperature ranges, the STARMA–GARCH hybrid model was applied.

4. Discussion

4.1. Gaps in the Literature

The analysis results in this research showed an interesting area for further investigation. The GSTARIMA model produced residuals, which were an integral part of the proposed model. The DNN process, whether using an MLP or a CNN, played a crucial role in minimizing residual values. In general, traditional spatio-temporal prediction models [8,11,89,90] or hybrids with NNs tended to use variables from a limited number of locations, or NN modeling employed a single layer [78,87,88].

The model used diverse climate data sources, prioritizing the integration of DL models without considering location parameters [20,23,58–60,79]. The performance analysis of DL integration yielded quite good results based on empirical simulation. However, climatic conditions in a particular location could vary due to the influence of other locations. This research did not extensively explore this gap, such as the influence of temperature, humidity, wind speed, solar radiation, and soil surface humidity, which could affect rainfall patterns across different locations. Climate analysis used more than single-variable constraints, such as predicting air pollution, rainfall, and wind speed separately [10,65,71,74,81]. Although this approach enhanced model accuracy, addressing other complex parameters was necessary to provide different solutions during general climate forecasting for adjacent locations that exhibited a significant correlation.

In the equatorial regions, the spatio-temporal hybrid model and NN employed stationary data, emphasizing the GSTAR model and NN with a single hidden layer [30]. Computation time and the analysis of specific variables played a notable role. The integration of the spatio-temporal model with DNN required a fast computation time, considering the hidden layer and selecting the appropriate regression activation function to improve accuracy compared to previous research. Fine-tuning the model was essential to mitigate trial and error during the integration process and optimize research time. For example, in the CNN model, the GSTARIMA residual input adopted the architecture of the output layer, which consisted of a fully connected layer. A nonlinear activation function was necessary to generate prediction classes as output. While this layer served as the final step in classifying CNN outputs [55], the use of the architecture was based on traditional tested models, such as LeNet-5 used in image detection [56], or VGGNet

introduced by Simonyan and Andrew Zisserman from the University of Oxford in 2014. This architecture was primarily employed in image processing and could hypothetically enhance the accuracy of the regression DNN.

4.2. Conceptual Model

4.2.1. Data Analytics Lifecycle for Climate Dataset

The proposed model concept integrated the GSTARIMA model with a DNN research flow using the Data Analysis Lifecycle methodology shown in Figure 6. The process began with the research problem formulation stage, encompassing problem identification, determining data sources as research indicators, and formulating initial hypotheses based on selected data samples to integrate the GSTARIMA–DNN model. Furthermore, the data preparation stage involved pre-processing, which included acquiring and preparing the data sources for analysis. The prepared data were then used as a model and stored in a repository database, constituting a significant volume of big data (such as climate data volume distributed by EarthData). The model planning stage involved providing data inputs for GSTARIMA as spatio-temporal modeling, considering multiple sample locations with various variables. Supervised ML algorithms were selected to fill in the input for the spatio-temporal model. Subsequently, in the model development stage, GSTARIMA modeling was performed using the Box–Jenkin procedure to produce a residual model based on the diagnostic test results. These findings served as the input data for the DNN process.
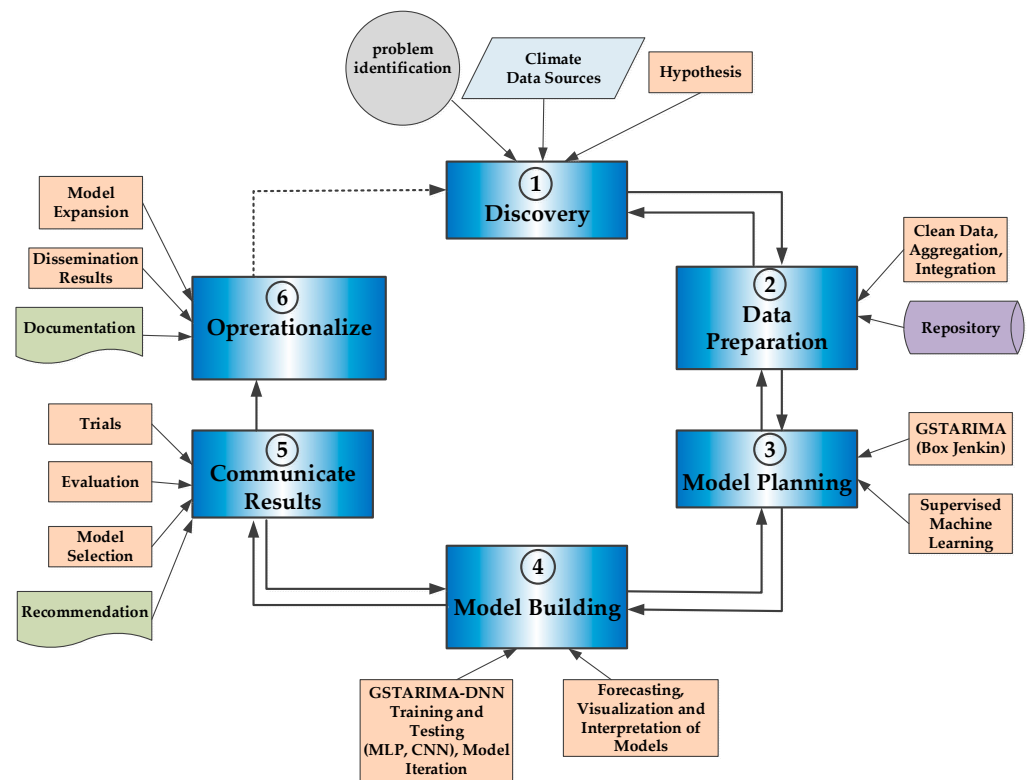

Figure 6. Overview of Data Analytics Lifecycle methodology of the GSTARIMA–DNN model for climate datasets forecasting.

The standard selection of the architecture (Multilayer Perceptron or CNN) determined accuracy and error values. The training and testing process of the model optimized accuracy and minimized errors by updating the GSTARIMA–DNN model integration architecture, including the hidden layer and the number of neurons, and adjusting the activation function used to determine the weight for the next layer. This ensured that the obtained results aligned with the initial target. The stage of communicating the results involved the

integration trial of the GSTARIMA–DNN model. The results were assessed based on their adherence to the initial hypothesis, leading to the formulation of a mathematical modeling theorem. Recommendations were required to select a model according to the research findings. The final stage was operationalization, which entailed implementing the model in a broader domain, documenting research results, reporting, and disseminating scientific papers in journals or other scientific meetings. The Data Analytics Lifecycle was carried out continuously to improve each stage until the model achieved the desired accuracy according to the final target

4.2.2. Integration of GSTARIMA with DNN for Forecasting

Based on the gap analysis and previous research reviews of existing models, GSTARIMA was integrated with DNN in a conceptual model, utilizing multiple layers and complex algorithms. Figure 7 shows the framework diagram for GSTARIMA model planning as part of the model building stage. As stated in the introduction, to answer the first research question, we explored how to integrate the two different models regarding GSTARIMA and DNN. The process began by inputting pre-processed climate data results comprising eight variables: rainfall, temperature, humidity, air pressure, wind speed, solar radiation, soil surface moisture, and root moisture.
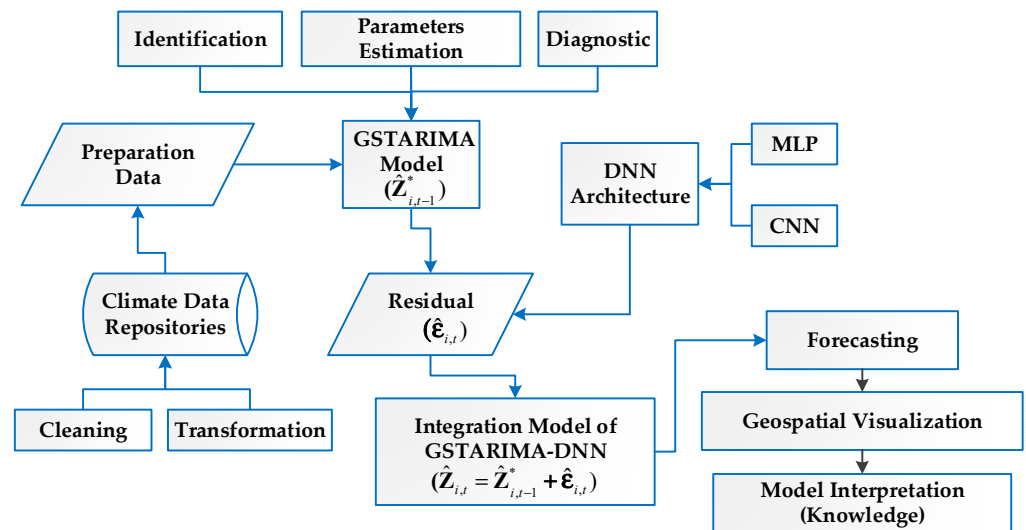

Figure 7. Framework diagram of GSTARIMA–DNN model integration process for forecasting.

Descriptive statistics were then obtained to examine the overall data average for each variable, standard deviation, maximum and minimum values, and data correlation between variables in the preparation data. The next stage involved testing data stationarity using Autocorrelation and Partial Autocorrelation functions. If the test results indicated nonstationarity in the average, a first-level differencing process was carried out on the identification process. On the other hand, if the data were stationary, the spatial weight matrix calculation was conducted using a distance weighting matrix. By plotting the Spatial Autocorrelation and Spatial Partial Autocorrelation functions, data were analyzed to observe autocorrelation and partial autocorrelation relationships, accounting for spatial dependence between locations. For parameter estimation of the GSTARIMA model, maximum likelihood was employed to obtain parameters for each location variable. Subsequently, the GSTARIMA model went through a diagnostic test to ensure no correlation existed among the residuals. The final stage of GSTARIMA modeling yielded residuals, which served as input for the DNN process in determining its architecture.

Figure 7 shows the process of obtaining the value of e_t for the GSTARIMA model using Equation (4). The error vector symbolized by e_t followed independently and identically distributed (iid) in the multivariate normal distribution $N_M(0, \sigma^2 \mathbf{I}_M)$, while the element

matrix $\mathbf{W}^{(l)}$ denoted the spatial lag weight l. The values $\hat{\varepsilon}_t^*$ could be obtained for N locations and Z variables, generating the GSTARIMA residual.

Following residual modeling, the forecasting process employed a DNN. The design of the DNN architecture significantly influenced the performance of the model analysis. Configuring the hidden layer, the number of neurons, and updating the weights through backpropagation greatly affected the MSE and MAPE values. The residual of the GSTARIMA model representing nonlinear data served as the input for the DNN. The residual input comprised a multivariate vector of climate variables. The process utilized the DNN algorithm with a multilayer architecture, such as an MLP or a CNN. The value $\hat{Z}_{i,t}$ represented a combination of GSTARIMA results $\hat{Z}_{i,t-1}^*$ and the DNN calculation of the nonlinear residual value $\hat{\varepsilon}_{i,t}^*$. Selecting an appropriate activation function for DNN regression modeling greatly aided in fine-tuning the integration model. GSTARIMA–DNN integration model enabled short-term forecasting for the next year and long-term forecasting for 5 to 10 years. The interpretation of the integration model was visualized using geospatial thematic maps, such as choropleth, heat, [91] or dot density maps, generating evaluating insights for climate forecasting. The end of the modeling process was the interpretation of forecasting, which generates knowledge for the short and long term.

Furthermore, the integration model obtained was applied to climate data with the abovementioned eight parameters. The hypothesis was based on each residual climate variable obtained from the GSTARIMA results. Nonlinear data were processed using MLP or CNN algorithms separately. The results were returned to the initial GSTARIMA model and compared without using a DNN to obtain the Mean Absolute Percentage Error (MAPE) resulting from the two models. The main element of this integration model is the residual, which is assumed to be independent and identically distributed (iid) and normal with constant variance. Although the residuals obtained from the GSTARIMA model are not linear, Deep Neural Networks anticipate non-constant (nonlinear) data. If the residual is not linear, then it is assumed that the DNN algorithm (we use MLP and DNN) makes the residual stable due to the involvement of the Moving Average (MA) element and further validation using MAPE values.

5. Conclusions

This paper proposed a systematic review integrating the spatio-temporal (GSTARIMA) model and DNNs for forecasting climate datasets. The review shows that hybrid and traditional models utilizing spatio-temporal and DL techniques have achieved high accuracy in performance analysis. Incorporating spatially sourced data from satellites and temporal data facilitates the development of intricate models for representing climate and environmental phenomena across multiple regions of the world. Notably, DNNs, representing ML, have shown promising outcomes and can provide reliable climate forecasting using multivariate time-series data. Through this research, the proposal is to integrate the GSTARIMA model with a DNN, capitalizing on the respective strength of each model. GSTARIMA reliably handles nonstationary data properties characterized by location dependence and seasonal patterns. However, the complexity of the model is complemented by a DNN, which adeptly captures nonlinear data patterns from GSTARIMA residuals. This integration is achieved by utilizing algorithms such as Multilayer Perceptron or CNN, which encompass multiple hidden layers and diverse activation functions. As a part of ML, the spatio-temporal aspect of DNNs is anticipated to assume a critical role in the future of statistical and mathematical climate forecasting.

Author Contributions: Conceptualization, D.M. and B.N.R.; methodology, B.N.R. and A.S.A.; validation, D.M., B.N.R. and A.S.A.; software. D.M., B.N.R. and H.F.P.; formal analysis, D.M., B.N.R., A.S.A. and H.F.P.; investigation, D.M., B.N.R. and A.S.A.; resources, D.M.; writing—original draft preparation, D.M.; writing—review and editing, B.N.R. and H.F.P.; supervision, B.N.R. All authors listed have made a substantial, direct, and intellectual contribution to the work and approved publication. All authors have read and agreed to the published version of the manuscript.

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

References

1. Box, G.E.P.; Jenkins, G.M. *Time Series Analysis Forecasting and Control*; Holden-Day Inc.: Oakland, CA, USA, 1976.
2. Pfeifer, P.; Deutsch, S. A Three-Stage Iterative Procedure for Space-Time Modeling. *Technometrics* **1980**, *22*, 35–47. [CrossRef]
3. Borovkova, S.A.; Lopuhaa, H.P.; Ruchjana, B.N. Generalized STAR Model with Experimental Weights. In Proceedings of the 17th International Workshop on Statistical Modeling, Trieste, Italy, 8–12 July 2002; pp. 139–147.
4. Min, X.; Hu, J.; Zhang, Z. Urban Traffic Network Modeling and Short-Term Traffic Flow Forecasting Based on GSTARIMA Model. In Proceedings of the IEEE Conference on Intelligent Transportation Systems, Proceedings, ITSC, Funchal, Portugal, 19–22 September 2010; pp. 1535–1540.
5. Akbar, M.S.; Setiawan; Suhartono; Ruchjana, B.N.; Prastyo, D.D.; Muhaimin, A.; Setyowati, E. A Generalized Space-Time Autoregressive Moving Average (GSTARMA) Model for Forecasting Air Pollutant in Surabaya. In Proceedings of the Journal of Physics: Conference Series, Surabaya, Indonesia, 19 October 2020; Volume 1490.
6. Hu, J.; Wang, S.; Mao, J. Short Time PM2.5 Prediction Model for Beijing-Tianjin-Hebei Region Based on Generalized Space Time Autoregressive (GSTAR). In Proceedings of the IOP Conference Series: Earth and Environmental Science, Ancona, Italy, 1–2 October 2019; Volume 358.
7. Jamilatuzzahro; Caraka, R.E.; Herliansyah, R.; Asmawati, S.; Sari, D.M.; Pardamean, B. Generalized Space Time Autoregressive of Chili Prices. In Proceedings of the 2018 International Conference on Information Management and Technology, ICIMTech, Jakarta, Indonesia, 3–5 September 2018; pp. 291–296.
8. Handajani, S.S.; Pratiwi, H.; Susanti, Y.; Subanti, S.; Respatiwulan; Hartatik. Rainfall Model on Area of Rice Production in Sragen, Karanganyar and Klaten by Using Generalized Space Time Autoregressive (GSTAR). In Proceedings of the Journal of Physics: Conference Series, Surakarta, Indonesia, 6–7 December 2017; Volume 855.
9. Andayani, N.; Sumertajaya, I.M.; Ruchjana, B.N.; Aidi, M.N. Comparison of GSTARIMA and GSTARIMA-X Model by Using Transfer Function Model Approach to Rice Price Data. In Proceedings of the IOP Conference Series: Earth and Environmental Science, Bogor, Indonesia, 19–20 October 2018; Volume 187.
10. Sulistyono, A.D.; Hartawati; Iriany, A.; Suryawardhani, N.W.; Iriany, A. Rainfall Forecasting in Agricultural Areas Using GSTAR-SUR Model. In Proceedings of the IOP Conference Series: Earth and Environmental Science, Yogyakarta, Indonesia, 30–31 July 2020; Volume 458.
11. Abdullah, A.S.; Matoha, S.; Lubis, D.A.; Falah, A.N.; Jaya, I.G.N.M.; Hermawan, E.; Ruchjana, B.N. Implementation of Generalized Space Time Autoregressive (GSTAR)-Kriging Model for Predicting Rainfall Data at Unobserved Locations in West Java. *Appl. Math. Inf. Sci.* **2018**, *12*, 607–615. [CrossRef]
12. Prasetiyowati, S.S.; Sibaroni, Y.; Carolina, S. Prediction and Mapping of Air Pollution in Bandung Using Generalized Space Time Autoregressive and Simple Kriging. In Proceedings of the 2020 International Conference on Data Science and Its Applications, ICoDSA, Bandung, Indonesia, 5–6 August 2020.
13. Kumar, R.R.; Sarkar, K.A.; Dhakre, D.S.; Bhattacharya, D. A Hybrid Space–Time Modelling Approach for Forecasting Monthly Temperature. *Environ. Model. Assess.* **2022**, *28*, 317–330. [CrossRef]
14. Zhao, Y.; Ge, L.; Zhou, Y.; Sun, Z.; Zheng, E.; Wang, X.; Huang, Y.; Cheng, H. A New Seasonal Difference Space-Time Autoregressive Integrated Moving Average (SD-STARIMA) Model and Spatiotemporal Trend Prediction Analysis for Hemorrhagic Fever with Renal Syndrome (HFRS). *PLoS ONE* **2018**, *13*, e0207518. [CrossRef]
15. Xu, L.; Chen, N.; Chen, Z.; Zhang, C.; Yu, H. Spatiotemporal Forecasting in Earth System Science: Methods, Uncertainties, Predictability and Future Directions. *Earth-Sci. Rev.* **2021**, *222*, 103828. [CrossRef]
16. Agoua, X.G.; Girard, R.; Kariniotakis, G. Short-Term Spatio-Temporal Forecasting of Photovoltaic Power Production. *IEEE Trans. Sustain. Energy* **2018**, *9*, 538–546. [CrossRef]
17. Ham, Y.G.; Kim, J.H.; Luo, J.J. Deep Learning for Multi-Year ENSO Forecasts. *Nature* **2019**, *573*, 568–572. [CrossRef]
18. Chattopadhyay, A.; Hassanzadeh, P.; Pasha, S. Predicting Clustered Weather Patterns: A Test Case for Applications of Convolutional Neural Networks to Spatio-Temporal Climate Data. *Sci. Rep.* **2020**, *10*, 1317. [CrossRef]
19. Zheng, J.; Wang, Q.; Liu, C.; Wang, J.; Liu, H.; Li, J. Relation Patterns Extraction from High-Dimensional Climate Data with Complicated Multi-Variables Using Deep Neural Networks. *Appl. Intell.* **2023**, *53*, 3124–3135. [CrossRef]

20. Li, W.; Gao, X.; Hao, Z.; Sun, R. Using Deep Learning for Precipitation Forecasting Based on Spatio-Temporal Information: A Case Study. *Clim. Dyn.* **2022**, *58*, 443–457. [CrossRef]

21. Zhang, Q.; Han, Y.; Li, V.O.K.; Lam, J.C.K. Deep-AIR: A Hybrid CNN-LSTM Framework for Fine-Grained Air Pollution Estimation and Forecast in Metropolitan Cities. *IEEE Access.* **2022**, *10*, 55818–55841. [CrossRef]

22. Qi, Y.; Li, Q.; Karimian, H.; Liu, D. A Hybrid Model for Spatiotemporal Forecasting of PM 2.5 Based on Graph Convolutional Neural Network and Long Short-Term Memory. *Sci. Total Environ.* **2019**, *664*, 1–10. [CrossRef] [PubMed]

23. Chen, R.; Wang, X.; Zhang, W.; Zhu, X.; Li, A.; Yang, C. A Hybrid CNN-LSTM Model for Typhoon Formation Forecasting. *Geoinformatica* **2019**, *23*, 375–396. [CrossRef]

24. Velo, R.; López, P.; Maseda, F. Wind Speed Estimation Using Multilayer Perceptron. *Energy Convers. Manag.* **2014**, *81*, 1–9. [CrossRef]

25. Deo, R.C.; Ghorbani, M.A.; Samadianfard, S.; Maraseni, T.; Bilgili, M.; Biazar, M. Multi-Layer Perceptron Hybrid Model Integrated with the Firefly Optimizer Algorithm for Windspeed Prediction of Target Site Using a Limited Set of Neighboring Reference Station Data. *Renew. Energy* **2018**, *116*, 309–323. [CrossRef]

26. Manley, K.; Egoh, B.N. Mapping and Modeling the Impact of Climate Change on Recreational Ecosystem Services Using Machine Learning and Big Data. *Environ. Res. Lett.* **2022**, *17*, 054025. [CrossRef]

27. Zhang, X.; Jin, Q.; Yu, T.; Xiang, S.; Kuang, Q.; Prinet, V.; Pan, C. Multi-Modal Spatio-Temporal Meteorological Forecasting with Deep Neural Network. *ISPRS J. Photogramm. Remote Sens.* **2022**, *188*, 380–393. [CrossRef]

28. Toharudin, T.; Caraka, R.E.; Yasin, H.; Pardamean, B. Evolving Hybrid Generalized Space-Time Autoregressive Forecasting with Cascade Neural Network Particle Swarm Optimization. *Atmosphere* **2022**, *13*, 875. [CrossRef]

29. Hiben, Y.G.; Kahsay, M.B.; Lauwaert, J. Hourly Solar Radiation Estimation Using Data Mining and Generalized Regression Neural Network Models. In Proceedings of the American Solar Energy Society National Solar Conference 2020 Proceedings, Online, 24–25 June 2020; pp. 155–164.

30. Setyowati, E.; Suhartono; Prastyo, D.D. A Hybrid Generalized Space-Time Autoregressive-Elman Recurrent Neural Network Model for Forecasting Space-Time Data with Exogenous Variables. In Proceedings of the Journal of Physics: Conference Series, Makassar, Indonesia, 9–10 October 2021; Volume 1752.

31. Kumar, B.; Chattopadhyay, R.; Singh, M.; Chaudhari, N.; Kodari, K.; Barve, A. Deep Learning–Based Downscaling of Summer Monsoon Rainfall Data over Indian Region. *Theor. Appl. Climatol.* **2021**, *143*, 1145–1156. [CrossRef]

32. Su, X.; Li, T.; An, C.; Wang, G. Prediction of Short-Time Cloud Motion Using a Deep-Learning Model. *Atmosphere* **2020**, *11*, 1151. [CrossRef]

33. Moher, D.; Liberati, A.; Tetzlaff, J.; Altman, D.G. Preferred Reporting Items for Systematic Reviews and Meta-Analyses: The PRISMA Statement. *BMJ* **2009**, *339*, b2535. [CrossRef] [PubMed]

34. Liu, Y.; Avello, M. Status of the Research in Fitness Apps: A Bibliometric Analysis. *Telemat. Inform.* **2020**, *57*, 101506. [CrossRef] [PubMed]

35. Kou, W.-J.; Wang, X.-Q.; Li, Y.; Ren, X.-H.; Sun, J.-R.; Lei, S.-Y.; Liao, C.-Y.; Wang, M.-X. Research Trends of Posttraumatic Growth from 1996 to 2020: A Bibliometric Analysis Based on Web of Science and CiteSpace. *J. Affect. Disord. Rep.* **2021**, *3*, 100052. [CrossRef]

36. Lungu, E.; Tang, A.; Trop, I.; Soulez, G.; Bureau, N.J. Current State of Bibliometric Research on the Scholarly Activity of Academic Radiologists. *Acad. Radiol.* **2020**, *29*, 107–118. [CrossRef] [PubMed]

37. El Mohadab, M.; Bouikhalene, B.; Safi, S. Bibliometric Method for Mapping the State of the Art of Scientific Production in COVID-19. *Chaos Solitons Fractals* **2020**, *139*, 110052. [CrossRef]

38. Rejeb, A.; Simske, S.; Rejeb, K.; Treiblmaier, H.; Zailani, S. Internet of Things Research in Supply Chain Management and Logistics: A Bibliometric Analysis. *Internet Things* **2020**, *12*, 100318. [CrossRef]

39. Chàfer, M.; Cabeza, L.F.; Pisello, A.L.; Tan, C.L.; Wong, N.H. Trends and Gaps in Global Research of Greenery Systems through a Bibliometric Analysis. *Sustain. Cities Soc.* **2020**, *65*, 102608. [CrossRef]

40. Zhang, P.G. Time Series Forecasting Using a Hybrid ARIMA and Neural Network Model. *Neurocomputing* **2003**, *50*, 159–175. [CrossRef]

41. Cho, S.B.; Lee, Y.W. Rice Yield Modeling in China Using Climate Data with Deep Neural Network. In Proceedings of the 40th Asian Conference on Remote Sensing, ACRS 2019: Progress of Remote Sensing Technology for Smart Future, Daejeon, Korea, 14–18 October 2020.

42. Alzubaidi, L.; Zhang, J.; Humaidi, A.J.; Al-Dujaili, A.; Duan, Y.; Al-Shamma, O.; Santamaría, J.; Fadhel, M.A.; Al-Amidie, M.; Farhan, L. Review of Deep Learning: Concepts, CNN Architectures, Challenges, Applications, Future Directions. *J. Big Data* **2021**, *8*, 53. [CrossRef]

43. Gardner, M.W.; Dorling, S.R. Artificial Neural Networks (the Multilayer Perceptron)—A Review of Applications in the Atmospheric Sciences. *Atmos. Environ.* **1998**, *32*, 2627–2636. [CrossRef]

44. Hermawan, E.; Lubis, S.W.; Harjana, T.; Purwaningsih, A.; Risyanto; Ridho, A.; Andarini, D.F.; Ratri, D.N.; Widyaningsih, R. Large-Scale Meteorological Drivers of the Extreme Precipitation Event and Devastating Floods of Early-February 2021 in Semarang, Central Java, Indonesia. *Atmosphere* **2022**, *13*, 1092. [CrossRef]

45. EMC Education Services. *Data Science and Big Data Analytics: Discovering, Analyzing, Visualizing and Presenting Data*; John Wiley & Sons, Inc.: Indianapolis, IN, USA, 2015; ISBN 978-1-118-87613-8.

46. Singh, A.; Kushwaha, S.; Alarfaj, M.; Singh, M. Comprehensive Overview of Backpropagation Algorithm for Digital Image Denoising. *Electronics* **2022**, *11*, 1590. [CrossRef]
47. Witten, I.; Frank, E. *Data Mining: Practical Machine Learning Tools and Techniques*; Morgan Kau: San Francisco, CA, USA, 2005.
48. Nithya, B.; Ilango, V. Predictive Analytics in Health Care Using Machine Learning Tools and Techniques. In Proceedings of the 2017 International Conference on Intelligent Computing and Control Systems (ICICCS), Madurai, India, 15–16 June 2017; pp. 492–499.
49. Xu, L.; Chen, N.; Zhang, X.; Chen, Z.; Hu, C.; Wang, C. Improving the North American Multi-Model Ensemble (NMME) Precipitation Forecasts at Local Areas Using Wavelet and Machine Learning. *Clim. Dyn.* **2019**, *53*, 601–615. [CrossRef]
50. Srivastava, A.; Saini, S.; Gupta, D. Comparison of Various Machine Learning Techniques and Their Uses in Different Fields. In Proceedings of the 2019 3rd International conference on Electronics, Communication and Aerospace Technology (ICECA), Coimbatore, India, 12–14 June 2019; pp. 81–86.
51. Jordan, M.I.; Mitchell, T.M. Machine Learning: Trends, Perspectives, and Prospects. *Science* **2015**, *80*, 255–260. [CrossRef]
52. Mishra, B.K.; Kumar, V.; Panda, S.K.; Tiwari, P. *Handbook of Research for Big Data: Concepts and Technique*; Taylor & Francis: Abingdon, UK, 2022.
53. Gurbuz, S.Z. *Deep Neural Network Design for Radar Applications*; Gurbuz, S.Z., Ed.; SciTech Publishing: Raleigh, NC, USA, 2020.
54. Atkinson, P.M.; Tatnall, A.R.L. Introduction Neuralnetworks in Remote Sensing. *Int. J. Remote Sens.* **1997**, *18*, 699–709. [CrossRef]
55. Dos Santos, C.; Gatti, M. Deep Convolutional Neural Networks for Sentiment Analysis of Short Texts. In Proceedings of the the 25th International Conference on Computational Linguistics: Technical Papers, Dublin, Ireland, 23–29 August 2014; pp. 69–78.
56. Lecun, Y.; Bottou, L.; Bengio, Y.; Haffner, P. Gradient-Based Learning Applied to Document Recognition. In Proceedings of the IEEE; IEEE: Piscataway, NJ, USA, 1998; Volume 86, pp. 2278–2324.
57. Rahul, K.; Banyal, R.K. *Data Life Cycle Management in Big Data Analytics*; Elsevier B.V.: Berlin/Heidelberg, Germany, 2020; Volume 173, pp. 364–371.
58. Jiao, X.; Li, X.; Lin, D.; Xiao, W. A Graph Neural Network Based Deep Learning Predictor for Spatio-Temporal Group Solar Irradiance Forecasting. *IEEE Trans. Ind. Inform.* **2022**, *18*, 6142–6149. [CrossRef]
59. Nikezić, D.P.; Ramadani, U.R.; Radivojević, D.S.; Lazović, I.M.; Mirkov, N.S. Deep Learning Model for Global Spatio-Temporal Image Prediction. *Mathematics* **2022**, *10*, 3392. [CrossRef]
60. Zou, M.Z.; Holjevac, N.; Dakovic, J.; Kuzle, I.; Langella, R.; Di Giorgio, V.; Djokic, S.Z. Bayesian CNN-BiLSTM and Vine-GMCM Based Probabilistic Forecasting of Hour-Ahead Wind Farm Power Outputs. *IEEE Trans. Sustain. Energy* **2022**, *13*, 1169–1187. [CrossRef]
61. Marco, Z.; Elena, A.; Anna, S.; Silvia, T.; Andrea, C. Spatio-Temporal Cross-Validation to Predict Pluvial Flood Events in the Metropolitan City of Venice. *J. Hydrol.* **2022**, *612*, 128150. [CrossRef]
62. Li, Y.; Wang, W.; Wang, G.; Tan, Q. Actual Evapotranspiration Estimation over the Tuojiang River Basin Based on a Hybrid CNN-RF Model. *J. Hydrol.* **2022**, *610*, 127788. [CrossRef]
63. Kong, W.J.; Li, H.C.; Yu, C.; Xia, J.J.; Kang, Y.Y.; Zhang, P.W. A Deep Spatio-Temporal Forecasting Model for Multi-Site Weather Prediction Post-Processing. *Commun. Comput. Phys.* **2022**, *31*, 131–153. [CrossRef]
64. Zhang, Y.; Gu, Z.; Thé, J.V.G.; Yang, S.X.; Gharabaghi, B. The Discharge Forecasting of Multiple Monitoring Station for Humber River by Hybrid LSTM Models. *Water* **2022**, *14*, 1794. [CrossRef]
65. Orescanin, M.; Petkovic, V.; Powell, S.W.; Marsh, B.R.; Heslin, S.C. Bayesian Deep Learning for Passive Microwave Precipitation Type Detection. *IEEE Geosci. Remote Sens. Lett.* **2022**, *19*, 4500705. [CrossRef]
66. Anshuka, A.; Chandra, R.; Buzacott, A.J.V.; Sanderson, D.; van Ogtrop, F.F. Spatio Temporal Hydrological Extreme Forecasting Framework Using LSTM Deep Learning Model. *Stoch. Environ. Res. Risk Assess.* **2022**, *36*, 3467–3485. [CrossRef]
67. Suhartono; Nahdliyah, N.; Akbar, M.S.; Salehah, N.A.; Choiruddin, A. A MGSTAR: An Extension of the Generalized Space-Time Autoregressive Model. In Proceedings of the Journal of Physics: Conference Series, Makassar, Indonesia, 9–10 October 2021; Volume 1752.
68. Böhm, C.; Schween, J.H.; Reyers, M.; Maier, B.; Löhnert, U.; Crewell, S. Toward a Climatology of Fog Frequency in the Atacama Desert via Multispectral Satellite Data and Machine Learning Techniques. *J. Appl. Meteorol. Climatol.* **2021**, *60*, 1149–1169. [CrossRef]
69. Christoforou, E.; Emiris, I.Z.; Florakis, A.; Rizou, D.; Zaharia, S. Spatio-Temporal Deep Learning for Day-Ahead Wind Speed Forecasting Relying on WRF Predictions. *Energy Syst.* **2021**, *14*, 473–493. [CrossRef]
70. Da Silva, C.C.; de Lima, C.L.; da Silva, A.C.G.; Moreno, G.M.M.; Musah, A.; Aldosery, A.; Dutra, L.; Ambrizzi, T.; Borges, I.V.G.; Tunali, M.; et al. Forecasting Dengue, Chikungunya and Zika Cases in Recife, Brazil: A Spatio-Temporal Approach Based on Climate Conditions, Health Notifications and Machine Learning. *Res. Soc. Dev.* **2021**, *10*, e452101220804. [CrossRef]
71. Guillaumin, A.P.; Zanna, L. Stochastic-Deep Learning Parameterization of Ocean Momentum Forcing. *J. Adv. Model. Earth Syst.* **2021**, *13*, e2021MS002534. [CrossRef]
72. Steffenel, L.A.; Anabor, V.; Kirsch Pinheiro, D.; Guzman, L.; Dornelles Bittencourt, G.; Bencherif, H. Forecasting Upper Atmospheric Scalars Advection Using Deep Learning: An O3 Experiment. *Mach. Learn.* **2021**, *112*, 765–778. [CrossRef]
73. Kimura, N.; Ishida, K.; Baba, D. Surface Water Temperature Predictions at a Mid-Latitude Reservoir under Long-Term Climate Change Impacts Using a Deep Neural Network Coupled with a Transfer Learning Approach. *Water* **2021**, *13*, 1109. [CrossRef]
74. Geng, H.; Wang, T. Spatiotemporal Model Based on Deep Learning for Enso Forecasts. *Atmosphere* **2021**, *12*, 810. [CrossRef]

75. Liu, D.; Mishra, A.K.; Yu, Z.B.; Lu, H.S.; Li, Y.J. Support Vector Machine and Data Assimilation Framework for Groundwater Level Forecasting Using GRACE Satellite Data. *J. Hydrol.* **2021**, *603*, 126929. [CrossRef]

76. Al-Shargabi, A.A.; Almhafdy, A.; Ibrahim, D.M.; Alghieth, M.; Chiclana, F. Tuning Deep Neural Networks for Predicting Energy Consumption in Arid Climate Based on Buildings Characteristics. *Sustainability* **2021**, *13*, 12442. [CrossRef]

77. Adewoyin, R.A.; Dueben, P.; Watson, P.; He, Y.; Dutta, R. TRU-NET: A Deep Learning Approach to High Resolution Prediction of Rainfall. *Mach. Learn.* **2021**, *110*, 2035–2062. [CrossRef]

78. Rajakumari, D.K.; Priyanka, V. Air Pollution Prediction in Smart Cities by Using Machine Learning Techniques. *Int. J. Innov. Technol. Explor. Eng.* **2020**, *9*, 1272–1279. [CrossRef]

79. Huang, W.; Li, Y.; Huang, Y. Deep Hybrid Neural Network and Improved Differential Neuroevolution for Chaotic Time Series Prediction. *IEEE Access* **2020**, *8*, 159552–159565. [CrossRef]

80. Chirayath, V.; Li, A.; Torres-Perez, J.; Segal-Rozenhaimer, M.; Van Den Bergh, J. *NASA NeMO-Net—A Neural Multimodal Observation and Training Network for Marine Ecosystem Mapping at Diverse Spatiotemporal Scales*; Institute of Electrical and Electronics Engineers Inc.: Piscataway, NJ, USA, 2020; pp. 3633–3636.

81. Ziyabari, S.; Du, L.; Biswas, S. A Spatio-Temporal Hybrid Deep Learning Architecture for Short-Term Solar Irradiance Forecasting. In Proceedings of the Conference Record of the IEEE Photovoltaic Specialists Conference, Calgary, AB, Canada, 15 June–21 August 2020; Volume 2020, pp. 0833–0838.

82. Zhang, W.; Liu, H.; Li, P.; Han, L. *A Multi-Task Two-Stream Spatiotemporal Convolutional Neural Network for Convective Storm Nowcasting*; Institute of Electrical and Electronics Engineers Inc.: Piscataway, NJ, USA, 2020; pp. 3953–3960.

83. Ding, Y.; Zhu, Y.; Wu, Y.; Jun, F.; Cheng, Z. *Spatio-Temporal Attention Lstm Model for Flood Forecasting*; Institute of Electrical and Electronics Engineers Inc.: Piscataway, NJ, USA, 2019; pp. 458–465.

84. Pusporani, E.; Suhartono; Prastyo, D.D. Hybrid Multivariate Generalized Space-Time Autoregressive Artificial Neural Network Models to Forecast Air Pollution Data at Surabaya. In Proceedings of the AIP Conference Proceedings, Surakarta, Indonesia, 26–28 July 2019; Volume 2194.

85. Thongniran, N.; Vateekul, P.; Jitkajornwanich, K.; Lawawirojwong, S.; Srestasathiern, P. Spatio-Temporal Deep Learning for Ocean Current Prediction Based on HF Radar Data. In Proceedings of the 2019 16th International Joint Conference on Computer Science and Software Engineering (JCSSE), Chonburi, Thailand, 10–12 July 2019; pp. 254–259.

86. Wilms, H.; Cupelli, M.; Monti, A.; Gross, T. *Exploiting Spatio-Temporal Dependencies for RNN-Based Wind Power Forecasts*; Institute of Electrical and Electronics Engineers Inc.: Piscataway, NJ, USA, 2019; pp. 921–926.

87. Cui, Y.K.; Xiong, W.T.; Hu, L.; Liu, R.H.; Chen, X.; Geng, X.Z.; Lv, F.; Fan, W.J.; Hong, Y. Applying a Machine Learning Method to Obtain Long Time and Spatio-Temporal Continuous Soil Moisture over the Tibetan Plateau. In Proceedings of the 2019 IEEE International Geoscience and Remote Sensing Symposium, Yokohama, Japan, 28 July–2 August 2019; pp. 6986–6989.

88. Saikhu, A.; Arifin, A.Z.; Fatichah, C. Non-Linear Spatio-Temporal Input Selection for Rainfall Forecasting Using Recurrent Neural Networks. In Proceedings of the 2018 International Seminar on Intelligent Technology and Its Applications (ISITIA), Bali, Indonesia, 30–31 August 2018; pp. 351–356.

89. Astuti, D.; Ruchjana, B.N.; Soemartini. Generalized Space Time Autoregressive with Exogenous Variable Model and Its Application. In Proceedings of the Journal of Physics: Conference Series, Bali, Indonesia, 25–29 July 2017; Volume 893.

90. Ippoliti, L. On-Line Spatio-Temporal Prediction by a State Space Representation of the Generalised Space Time Autoregressive Model. *Metron* **2001**, *59*, 157–168.

91. Słomska-Przech, K.; Panecki, T.; Pokojski, W. Heat Maps: Perfect Maps for Quick Reading? Comparing Usability of Heat Maps with Different Levels of Generalization. *ISPRS Int. J. Geo-Inf.* **2021**, *10*, 562. [CrossRef]