

# Deep Learning Based Effective Fine-grained Weather Forecasting Model

Pradeep Hewage<sup>\*</sup> [0000-0002-6909-3546], Marcello Trovati [0000-0001-6607-422X], Ella Pereira [0000-0001-7273-3295], and Ardhendu Behera [0000-0003-0276-9000]

*Department of Computer Science, Edge Hill University, Ormskirk, Lancashire, L39 4QP, UK*

## ***Abstract***

It is well-known that Numerical Weather Prediction (NWP) models require considerable computer power to solve complex mathematical equations to obtain a forecast based on current weather conditions. In this article, we propose a novel lightweight data-driven weather forecasting model by exploring temporal modelling approaches of Long Short-Term Memory (LSTM) and Temporal Convolutional Networks (TCN) and compare its performance with the existing classical machine learning approaches, statistical forecasting approaches, and a dynamic ensemble method, as well as the well-established Weather Research and Forecasting (WRF) NWP model. [More specifically](#) Standard Regression (SR), Support Vector Regression (SVR), and Random Forest (RF) [are implemented](#) as the classical machine learning approaches, and Autoregressive Integrated Moving Average (ARIMA), Vector Auto Regression (VAR), and Vector Error Correction Model (VECM) [are implemented](#) as the statistical forecasting approaches. [Furthermore](#), Arbitrage of Forecasting Expert (AFE) [is implemented](#) as the dynamic ensemble method [in this article](#). Weather information is captured by time-series data and thus, we explore the state-of-art LSTM and TCN models, which is a specialised form of Recurrent Neural Network (RNN) for weather prediction. The proposed deep model consists of a number of layers that use surface weather parameters over a given period of time for weather forecasting. The proposed deep learning networks with LSTM and TCN layers are assessed in two different regressions, namely Multi-Input Multi-Output (MIMO) and Multi-Input Single-Output (MISO). Our experiment shows that the proposed lightweight model produces better results compared to the well-known and complex WRF model, demonstrating its potential for efficient and accurate weather forecasting up to 12 hours.

Keywords:

Long Short-Term Memory, Temporal Convolutional Networks, Weather Prediction, WRF, Neural Network, Time-series Data Analysis.

---

<sup>\*</sup> Corresponding author. Tel. +44 1695575171 ext.1872; e-mail: 23147458@go.edgehill.ac.uk

# 1 Introduction

Weather forecasting refers to the scientific process of predicting the state of the atmosphere based on specific time frames and locations [1]. Numerical Weather Prediction (NWP) utilises computer algorithms to provide a forecast based on current weather conditions by solving a large system of non-linear mathematical equations, which are based on specific mathematical models. More specifically, these models define a coordinate system, which divides the earth into a 3-dimensional grid. The weather parameters such as winds, solar radiation, the phase change of water, heat transfer, relative humidity, and surface hydrology are measured within each grid and their interaction with neighbouring grids to predict atmospheric properties for the future [2].

Meteorology adopted a more quantitative approach with the advancement of technology and computer science, and forecast models became more accessible to researchers, forecasters, and other stakeholders. Many NWP systems were developed in recent years, such as Weather Research and Forecasting (WRF) model, where increasing high-performance computing power has facilitated the enhancement and the introduction of regional or limited area models [3]. As a consequence, the WRF model became the world's most-used atmospheric NWP model due to its higher resolution rate, accuracy, open-source nature, community support, and a wide variety of usability within different domains [4], [5].

According to [1], data-driven computer modelling systems can be utilised to reduce the computational power of NWPs. In particular, Artificial Neural Network (ANN) can be used for this purpose due to their adaptive nature and learning capabilities based on prior knowledge. This feature makes the ANN techniques very appealing in application domains for solving highly nonlinear phenomena. Deep models for multivariate time-series forecasting often use Recurrent Neural Networks (RNN) and Temporal Convolutional Networks (TCN). [Recently, a variant of RNN called Long Short Term Memory \(LSTM\) has attached considerable attention due to its superior performance.](#) Such models have attracted considerable attention due to their superior performance [6]–[8]. Deep networks often use stacked neural networks and include several layers as part of the overall composition known as nodes. The computation takes place at the node level since it allows the combination of data input through a set of coefficients. Subsequently, the activation function gets established on the basis of input-weight products while signal progresses through the network [9]. Regression technique is often employed to develop and evaluate Neural network models for accurate weather prediction as the weather information is captured by time-series data consisting of real numbers [10].

This article presents developing and evaluating a lightweight and novel weather forecasting system using modern neural networks. Figure 1 depicts a general overview of the research discussed in this article. More specifically, a suitable machine learning model is proposed by exploring temporal modelling approaches of LSTM and TCN, and compare its performance with classical machine learning approaches, statistical forecasting models, and a dynamic ensemble method. Secondly, we use the proposed model for short-term weather prediction and compare the model accuracy with the well-established WRF model. Finally, we reform the model for long-term weather forecasting, and analyse the model accuracy and compared the performance to the state-of-art WRF model.

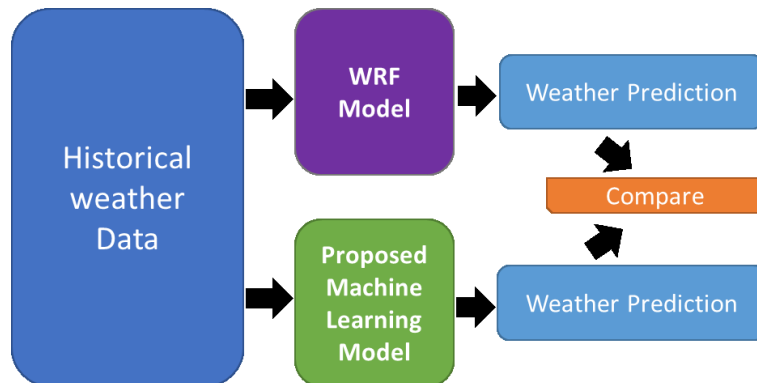


Fig. 1. Overview of the research

In this study, we investigate LSTM and TCN over RNN since there is an inherent issue of the vanishing gradient problem with the RNN [6]. The LSTM and TCN can overcome this vanishing gradient issue, but it can easily use up the high capacity of memory [8], [11]. The rest of the article is organised as follows: Section 2 focuses on related work, and Section 3 discusses the research aims and objectives. In Section 4, we present the WRF model and its challenges, and Section 5 discusses the sequence modelling and prediction. In Sections 6 and Section 7, we discuss the methodology and results. Finally, Section 8 concludes the article.

## 2 Related work

Numerical Weather Prediction (NWP) concept was proposed by Lewis Fry Richardson in 1922, and practical use of NWP began in 1955 after the development of programmable computers [1]. Neural Networks based weather forecasting has been evolved significantly in the last three decades. Before the year 2000, the Model Output Statistics (MOS) was the most widely used approach to improve the numerical models' ability to forecast by relating model outputs to observational data [12]–[14]. A mixed statistical or dynamic technique for the weather forecasting was introduced by [15] in 1983. The work in [16] added a new perception to dynamic modelling in 1991. These approaches have limitations and challenges such as massive computational requirements, lack of design methodologies for selecting the model architecture and parameters, and time-consuming to prediction resulting less reliability as the difference between the current time and the forecast time increases [13], [16], [17].

Artificial Neural Network based minimum temperature prediction system was introduced in 1991 using the backpropagation algorithms [18], [19]. This concept considerably reduced the computational requirements of MOS directing an effective forecast [16]. A snowfall and rainfall forecasting model was introduced in 1995 from weather radar images with ANN [20]. The results show that the ANN is more effective than the traditional cross-correlation method, and the persistence prediction method is producing a substantial reduction in prediction error. In 1998, Oishi et al. developed a severe rainfall prediction method using AI [21]. The development method was unique as it is introduced inference (i.e. Knowledge-based) rather than using numerical simulations. A Multi-Polynomial High Order Neural Network (M-PHONN) based rainfall prediction model was developed by Hui Qi and Ming Zhang in 2001 [22]. This new model has features such as increasing the speed, accuracy, and the robustness of the rainfall estimate. Therefore, this model could be used to complement the already established Auto-Estimator algorithms.

A multilayer perceptron network was trained with the backpropagation algorithm with momentum for temperature forecasting in 2002 [23]. The results were very encouraging and clearly demonstrated the potential for future weather forecasting applications. In the same year, a comparative was carried out analysing different neural network models for daily maximum and minimum temperature, and wind speed [24]. The results show that the Radial Basis Function Network (RBFN) produced the most accurate forecast compared to the Elman Recurrent Neural Network (ELNN) and Multi-Layered Perceptron (MLP) networks. In 2005, a rough set of fuzzy neural network was introduced to forecast weather parameters; dew temperature, wind speed, temperature, and visibility [25]. This model has several fuzzy rules, and their initial weights were estimated with a deeper network for weather forecasting. Moreover, M. Hayati and Z. Mohebi proposed a successful model for temperature forecasting based on MLP.

A feature-based neural network model was introduced in 2008 to predict maximum temperature, minimum temperature, and relative humidity [26]. Neural Network features are extracted over different periods as well as from the time-series weather parameter itself. In particular, feedforward ANN is utilised in this approach with backpropagation for supervised learning. The prediction results have a high degree of accuracy, and this modelling is recommended as an alternative to traditional meteorological approaches by [27]–[29]. In 2012, a Backpropagations Neural Network (BPN) was implemented for temperature forecasting [30], [31]. This network has successfully identified the non-linear structural relationship between various input weather parameters. Furthermore, a new hybrid model was introduced in 2014 to forecast the temperature which is based on an Ensemble of Neural Networks (ENN) [32], and the results suggested that including image data would improve the prediction results. In the same year, a deep neural network based feature representation for weather prediction model was developed for the temperature and dew point prediction [33].

In 2015, eight different novel regression tree structures were applied to short-term wind speed prediction [34]. The author also compared the best regression tree approach against other AI approaches such as Support Vector Regression (SVR), MLP, extreme learning machines, and multi-linear regression approach. The best regression tree yields the best results for wind speed prediction. In the same year, a deep neural network was introduced for ultra-short-term wind forecasting with success [35]. Deep learning with LSTM layers has been introduced to precipitation nowcasting by Shi et al. [11]. The experimental results show that the LSTM network has the ability to capture spatiotemporal correlations and can be used to precipitation nowcasting. In the same year, a model was developed to predict the temperate in Nevada using a deep neural network with stacked denoising auto-encoders with higher accuracy of 97.97% compared to traditional neural networks (94.92%) [36]. In 2016, the multi-stacked deep learning LSTM approach was utilised to forecasting weather parameters temperature, humidity, and wind speed [37]. The author suggested that the model could be used to predict other weather parameters based on the effectiveness and accuracy of the results.

Traditional machine learning methods were analysed for radiation forecasting in 2017 [38]. The author concluded that the SVR, regression trees, and forests have produced a promising outcome for radiation forecasting. In 2018, the Backpropagation Neural (BPN) network's performance compared with linear regression and regression tree for temperature forecasting [39]. As a result, a significant better temperature yields the BPN. In 2018, a short-term local rain and temperature forecasting model was developed using deep neural network [40]. The author concluded that the deep neural networks yield the highest accuracy for rain prediction among several machine learning methods. In the same year, the neural network approach is utilised to create models to predict sea surface temperature and soil moisture [41], [42].

The selected state-of-the-art deep learning approaches for weather forecasting and their contributions and differences with the previous approaches are discussed in Table 1.

Deep learning approach for weather forecasting	Contribution and difference with the previous approaches
Deep neural networks for ultra-short-term wind forecasting [35]	Results show that carefully selection of deep neural networks outperforms shallow ones. The model accepts a single input parameter and predicts a single parameter, and the model is limited to very short-term forecasting (less than an hour).
Weather forecasting using deep learning techniques [43]	<b>Recurrent</b> neural network <b>is used</b> for prediction of the rainfall with adequate accuracy level. The model uses a single input single output and is used for short-term forecasting.
Short-term local weather forecast using dense weather station by deep neural network [40]	Deep neural network is used to predict rain and temperature. The researches use four input parameters and predict one parameter at a given time. This model <b>is</b> able to predict data accurately up to an hour.
Convolutional LSTM Network: A Machine Learning Approach for Precipitation Nowcasting [11]	Formulated precipitation nowcasting as a spatiotemporal sequence forecasting problem. The proposed model is a Single-input single-output and able to produce a state-of-the-art performance for up to 6 hours.
Forecasting the weather of Nevada: A deep learning approach [36]	This model accepts four input parameters and predicts one output as temperature. <b>Results indicated that stacked denoising auto-encoder deep learning model predicts accurate long-term temperature.</b>
Sequence to Sequence Weather Forecasting with Long Short-Term Memory Recurrent Neural Networks [37]	<b>Multi-stacked</b> LSTMs <b>are used</b> to map sequences of weather values of the same length. This model uses three input parameters and it predicts one parameter at a time.
A Deep Learning Methodology Based on Bidirectional Gated Recurrent Unit for Wind Power Prediction [44]	Contributed the bidirectional gated recurrent network for <b>wind power</b> forecasting. The model used wind direction and wind speed as inputs and predicted the results more accurately up to 6 hours.

Table 1: Existing deep learning approaches and their contributions

The above existing weather forecasting models are able to predict up to maximum three weather parameters. Besides, weather forecasting is an entirely non-linear process, and each parameter often depends upon one more other parameters [13], [45], [46]. These larger numbers of interrelated parameters work together, aiming for an accurate weather forecast in a more reliable NWP such as met office and WRF models [4], [47]. A maximum of up to four input weather parameters is considered in the existing AI-based forecasting models.

Based on the related work, it is evident that:

- There is no identified attempt to compare an AI-based weather prediction with a well-established and existing weather forecasting model such as WRF;
- There has been little or no attempt to compare traditional machine learning approaches with cutting-edge deep learning technologies for weather forecasting;
- Most of the existing approaches use less than four interrelated input parameters for neural network-based weather forecasting model;
- A complete AI-based weather forecasting model with up to 10 input/output weather parameters is yet to be explored.

### 3 Research aim and objectives

The work presented in this article aimed to develop a weather forecasting model to address the above-mentioned drawbacks using state-of-the-art deep models by establishing the following objectives.

1. To propose an efficient Neural Network-based weather forecasting model by exploring temporal modelling approaches of LSTM and TCN, and compare its performance with the existing approaches;
2. Use the proposed neural network model for short-term weather prediction and compare the results with WRF model prediction;
3. Fine-tune the proposed model for long-term weather forecasting;
4. Compare the model performances for long-term forecasting with the WRF model prediction.

Our approach is targeted to develop deep neural networks to solve the regression problem of weather forecasting. We propose two different regression models to assess proposed deep learning models, namely Multi-Input Multi-Output (MIMO) and Multi-Input Single-Output (MISO). In this article, we addressed the above objectives in detail in various sections. Objective 1, an effective neural network-based weather forecasting model is proposed and compared its performance with existing approaches in Section 7.1. Objective 2, the proposed model is used to short-term weather forecasting and compared its performance with the WRF model predictions in Section 7.2. For Objective 3 and Objective 4, the proposed model is fine-tuned for long-term forecasting and compared the results with the WRF model predictions in Section 7.3.

### 4 Weather Research and Forecasting (WRF) model

The WRF model was developed by Norwegian physicist Vilhelm Bjerknes in the latter part of the 1990s as part of a collaborative partnership with many environmental and meteorology organisations. The model involves solving of various thermodynamic equations so that numerical weather-based predictions can be made mainly through different vertical levels [48], [49]. The primary role of the WRF is to carry out analysis focusing on climate time scale via linking physics data between land, atmosphere and ocean. The WRF model is currently the world's most-used atmospheric model since its initial public release in the year 2000 [5].

In order to investigate the model for real cases, it is necessary to install and configure WPS (WRF Pre-processing System), WRF ARW (Advanced Research WRF model), and Post Processing software. The WRF post-processing is not described in this article, as the main objective is to collect historical weather data for prediction and analyses. Interested researchers can refer to [50] for further details. The WRF ARW and the WPS share common routines, like WRF I/O API. Therefore, the successful compilation of the WPS depends upon the successful compilation of the WRF ARW model [4].

The WRF model needs to run in two different modes to extract time-series data. Firstly, historical weather data are collected and subsequently, predicted weather data is identified for evaluation purposes. For each instance,

the model runs in a single domain mode and utilises different “*namelist.wps*” and “*namelist.input*” files to configure the WPS and WRF-ARW components [17]. GRIBbed Binary or General Regularly-distributed Information in Binary, often use as GRIB data, which is a concise data format commonly used in meteorology to store historical and forecast weather data [17], [51]. According to [52], Global Forecast System (GFS) GRIB data provides 0.25 degrees resolution and available to download every three hours freely. Therefore, the GFS three-hourly data are selected for this project, with a horizontal resolution set to 10km.

One of the primary challenges in the WRF is its requirement for massive computational power to solve the equations that describe the atmosphere. Furthermore, atmospheric processes are associated with highly chaotic dynamical systems, which causes a limited model's accuracy. As a consequence, the model forecast capabilities are less reliable as the difference between the current time and the forecast time increases [1], [53]. In addition, the WRF is a large and complex model with different versions and applications, which lead to the need for greater understanding of the model, its implementation and the different option associated with its execution [5]. The GFS 0.25 degrees dataset is the freely available highest resolution dataset for the WRF model. This allows the user to forecast weather data at a horizontal resolution about 27km [51], [52]. This implies that the user can predict data with increased accuracy up to 27km. The model calculates the lesser resolution data based on results obtained. Thus, the model obtains better results for long-range forecast and not for a selected geographical region, such as a farm, school, places of interest, and so on [5], [17], [54].

Based on the above discussion, we propose a novel lightweight weather prediction model that could run on a standalone PC for accurate weather prediction and could easily be deployed in a selected geographical region.

## 5 Sequence modelling and Prediction

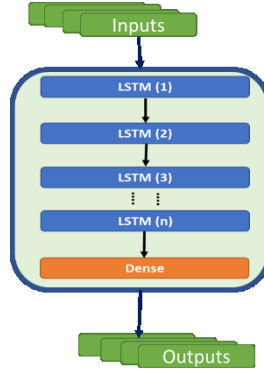
The modelling task has been highlighted before defining a network structure which involves time-series weather data sequence  $x_0, \dots, x_T$  and wish to predict some corresponding outputs  $y_0, \dots, y_T$  at each time. As presented in Table 2, there are 10 different weather parameters in data at a given time  $t$ ,  $x_t = [p_1, \dots, p_{10}]$ . The aim is to predict the value  $y_t$  at time  $t$ , which is constrained to only previously observed inputs:  $x_0, \dots, x_{t-1}$ . Therefore, the sequence modelling network can be defined as a function  $\mathcal{F} : \mathcal{X}^{T+1} \rightarrow \mathcal{Y}^{T+1}$  that produces the mapping  $\hat{y}_0, \dots, \hat{y}_T = \mathcal{F}(x_0, \dots, x_T)$ , if it satisfies the causal constraints, i.e.  $y_t$  only depends on  $x_0, \dots, x_t$  and not on any future inputs  $x_{t+1}, \dots, x_T$ . The main idea of learning in the sequence modelling is to find a network  $\mathcal{F}$  which minimizes the loss ( $\ell$ ) between the actual outputs and the predictions,  $\ell(y_0, \dots, y_T, \mathcal{F}(x_0, \dots, x_T))$  in which the sequences and predictions are drawn according to some distribution.

The WRF model with GFS- GRIB data can produce a large amount of historical weather data. Recurrent Neural Networks (RNN), LSTM, and TCN are extremely expressive models which are appropriate in such a scenario. These networks have attracted considerable attention due to their superior performance based on ability to learn highly complex vector-to-vector mapping [55], [56]. The LSTM/TCN is a specialised form of RNN that is designed for sequence modelling [55], [57]. Highly dimensional hidden states  $\mathbf{H}$  are the basic building blocks of RNN which are updated with non-linear activation function  $\mathcal{F}$ . At a given time  $t$ , the hidden state  $\mathbf{H}_t$  is updated by  $\mathbf{H}_t = \mathcal{F}(\mathbf{H}_{t-1}, x_t)$ . The structure of  $\mathbf{H}$  works as the memory of the network. The state of the hidden layer at a given time is conditioned on its previous state. The RNN is extremely deep as they are maintained a vector activation through time at each timestep. This will result in high training time-consuming due to the exploding and the vanishing gradient problems [6]. The development of LSTM and TCN architectures have been addressed the gradient vanishing issue with RNN [58]. Therefore, the LSTM and TCN deep learning architectures are used in this study.

### 5.1 Proposed Deep Model with Long Short-Term Memory (LSTM) layers

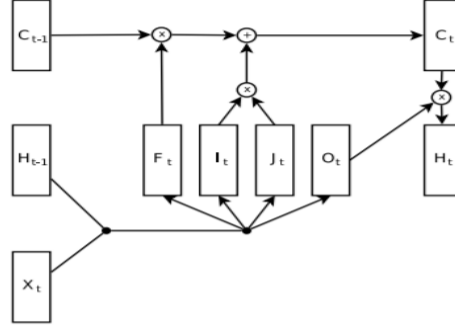
The proposed model is based on LSTM networks and uses temporal weather data to identify the patterns and produces weather predictions. As discussed in Section 5, we experiment with the state-of-the-art LSTM, which is a specialised form of RNN, and it is widely applied to handle temporal data. The key concepts of the LSTM have the ability to learn long-term dependencies by incorporating memory units. These memory units allow the network to learn, forget previously hidden states, and update hidden states [6], [9], [59].





**Fig. 2.** Proposed Weather Prediction Model using LSTM

Figure 2 shows the deep learning model consisting of stacked LSTM layers for weather forecasting using surface weather parameters. Table 2 describes the surface weather parameters, which are used as the input parameters. The model provides outputs, which are the predicted weather parameters.



**Fig. 3.** LSTM memory cell [6]

Figure 3 shows the LSTM memory architecture used in our model. More specifically, the proposed model has the input vector  $X_t = [p_1, p_2, \dots, p_9, p_{10}]$  at a given time step  $t$ , which consists of 10 different ( $p_1 \dots p_{10}$ ) weather parameters. In a given time  $t$ , the model updates the memory cells for long-term  $C_{t-1}$  and short-term  $H_{t-1}$  recall from the previous timestep  $t - 1$  via:

$$\begin{aligned}
 I_t &= \tanh(w_{xi}X_t + w_{hi}H_{t-1} + b_i) \\
 J_t &= \text{sigm}(w_{xj}X_t + w_{hj}H_{t-1} + b_j) \\
 F_t &= \text{sigm}(w_{xf}X_t + w_{hf}H_{t-1} + b_f) \\
 O_t &= \tanh(w_{xo}X_t + w_{ho}H_{t-1} + b_o) \\
 C_t &= C_{t-1} \odot F_t + I_t \odot J_t \\
 H_t &= \tanh(C_t) \odot O_t
 \end{aligned} \tag{1}$$

The notations of Equation 1 are:  $w_*$ -weight matrices,  $b_*$ - biases,  $\odot$ - element-wise vector product,  $I_t$ - input gate and  $J_t$ - input moderation gate contributing to memory,  $F_t$ - forget gate, and  $O_t$ -output gate as a multiplier between memory gates. To allow the LSTM to make complex decisions over a short period of time, there are two types of hidden states, namely  $C_t$  and  $H_t$  [6], [60]. The LSTM has the ability to selectively consider its current inputs or forgets its previous memory by switching the gates  $I_t$  and  $F_t$ . Similarly, the output gate  $O_t$  learns how much memory cell  $C_t$  needs to be transferred to the hidden state  $H_t$ . Compared to the RNN, these additional memory cells give the ability to learn enormously complex and long-term temporal dynamics with the LSTM.

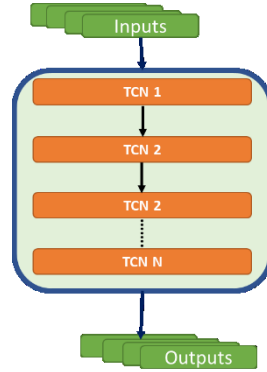
In this work, we propose two types of deep models to solve the regression problem involving weather forecasting, namely Multi-Input Multi-Output (MIMO) and Multi-Input Single-Output (MISO).

### 5.1.1 MIMO-LSTM and MISO-LSTM

In the MIMO, all the weather parameters (i.e. 10 surface weather parameters in this study) are fed into the network, which is expected to predict the same number of parameters (i.e. 10 parameters in this study) as the output. Therefore, only one model is required for weather forecasting. In MISO approach, all of the weather parameters (i.e. 10 surface weather parameters in this study) are fed into the network, which is expected to predict a single parameter. Whereas, in the MISO, 10 different models are required as each of them is trained to predict a particular weather parameter.

## 5.2 Proposed Deep Model with Temporal Convolutional Network (TCN) layers

The main characteristic of the TCN is that the network can take a sequence of any length as inputs and map it to an output sequence of the same length, just similar to the RNN categories. These networks involve causal convolutions and initially developed to examine long-range patterns using a hierarchy of temporal convolutional filters [8], [61], [62]. TCN architecture is quite simple and is informed by recent generic convolutional architectures for sequential data. Figure 4 shows the general deep learning with stacked TCN architecture.

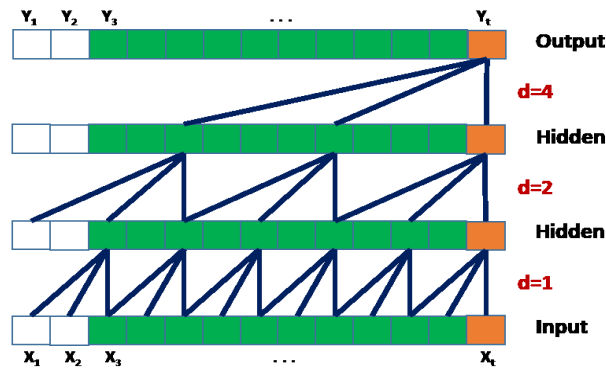


**Fig. 4.** Deep learning with stacked TCN layers

This TCN architecture has no skip connections across layers, conditioning, context stacking or gated activations, and autoregressive prediction and a very long memory. This use dilated convolutions that enable an exponentially large receptive field, allowing very deep networks and very long effective history [63]. For instance, the dilation convolution operation  $F$  for a 1-D sequence of a given weather parameter  $p^1$ , i.e.  $p = (p_0^1, \dots, p_t^1)$  and a filter  $f : \{0, \dots, k-1\}$ , on element  $s = p_{\hat{t}}^1$  (where  $\hat{t} = 0, \dots, t$ ) of the sequence is defined as:

$$F(s) = (p *_d f)(s) = \sum_{i=0}^{k-1} f(i) \cdot p_{s-d \cdot i} \quad (2)$$

The notations of Equation 2 are:  $d$ - dilation factor,  $k$ - filter size, and  $s - d \cdot i$  accounts for the direction of the past. The TCN consists of stacked units of one dimensional convolution with activation functions [7]. The architectural elements in a TCN with configurations dilations dilation factors  $d = 1, 2$ , and 4 are shown in the Figure 5. The input to the TCN is  $x_t$  and output  $y_t$ . The  $x_t$  contains 10-dimensional weather parameter



**Fig. 5.** Architectural elements in a typical TCN layer with causal convolution and different dilation factors.



The main purpose of the dilation to introduce a fixed step between every adjacent filter taps, and larger dilations and larger filter sizes  $k$  enable effectively expanding the receptive field [8], [62]. The increment of  $d$  exponentially increase the depth of the network in these convolutions and this guarantees that there is some filters that hits each input within the effective history [62].

### 5.2.1 MIMO-TCN and MISO-TCN

Similar to LSTM in section 5.1.1 and section 5.1.2, we also use the TCN in our proposed MIMO and MISO models.

## 5.3 Proposed model for weather forecasting

As discussed in Section 5.1 and Section 5.2, the LSTM and TCN deep learning approaches are proposed for weather forecasting. The MIMO and MISO are the two types of deep models to solve the regression problem. Therefore, proposed models for weather forecasting are MIMO-LSTM, MISO-LSTM, MIMO-TCN, and MISO-TCN. Deep learning models are discussed in [11], [35], [40], [43], [44] are single input single output models. The MISO are experimented in [36], [37] and a MIMO is discussed in [64]. All these models can be accepted up to four input parameters at a given time. Increased number of input parameters will increase the forecasting accuracy of an NWP model by distinguishing interrelationships among parameters [17], [50]. Our proposed model uses ten input parameters which has not been explored in the past for neural network-based weather forecasting. Subsequently, the research discusses in this article is explored for both MIMO and MISO.

Moreover, [44] uses the bidirectional recurrent network with weather-related input parameters successfully to predict the wind power up to 6 hours. Therefore, bidirectional LSTM experiments in long-term forecasting and compare with the proposed model. Most of the researches discussed in Table 1 are attempted to forecasting a single or few parameters for a specific purpose rather developing a complete weather forecasting model. Our proposed model explores to complete AI-based fine-grained weather forecasting model.

We use Keras as a tool to implement both LSTM and TCN deep learning networks [59], [65]–[67].

## 6 Methodology

This is an empirical-based study and is focused on analysing the quantitative temporal weather data. There are 10 surface weather parameters utilised in this research for weather prediction. These weather parameters are identified by considering their usefulness in precision farming. Moreover, these surface parameters can be captured at a chosen location using various sensors using a local weather station.

### 6.1 Surface weather parameters

The surface weather parameters are observed and reported in for monitoring and forecasting purposes [68]. In our previous study, we defined 10 surface weather parameters for the forecasting, which can be extruded from GRIB data using the WRF model [67]. Those 10 surface parameters, as shown in Table 2.

Variable	Description	Measuring Unit
TSK	Skin temperature or surface temperature	°K
PSFC	Surface pressure	Pa
U10	X component of wind at 10m	m/s
V10	Y component of wind at 10m	m/s
Q2	2- meter specific humidity	Kg/Kg
Rainc	Convective rain (Accumulated precipitation)	mm
Rainnc	Non-convective rain	mm
Snow	Snow water equivalent	Kg/m <sup>2</sup>
TSLB	Soil temperature	°K
SMOIS	Soil Moisture	m <sup>3</sup> /m <sup>3</sup>

**Table 2.** Surface weather parameters (10 identified parameters used by our model)

The surface parameters of wind direction and wind speed can be calculated from the WRF surface variables  $U_{10}$  and  $V_{10}$  [4]. Table 2 shows the surface weather parameters which are utilised in this research. The  $XLAT$ -Reference Latitude and  $XLONG$ -Reference Longitude parameters are used with each data point for the location identification.

## 6.2 Data collection and preparation

As described in Sections 4, the GRIB data is used to run the WRF model. A total of 12 weather parameters is extracted from the period of January 2018 to May 2018. This is used as the training dataset to train the proposed models. Similarly, the parameters in June 2018 data are used to test the network. This is to test different trained deep models to identify the best model for forecasting. The parameters in July 2018 are considered as the validation dataset, which is used as the ground truth to compare perdition from the best model. The WRF model is being run in forecast mode using the same format GRIB data for the month of July 2018 to evaluate the overall prediction performance of the WRF model.

The training data set has been normalised to keep each value in between -1 and 1, and the same maximum and minimum variable values are used to normalise the testing and the evaluation data set. We apply a sliding window of seven days temporal resolution on each dataset as input to the model and the temporal resolution of next 3 hours data as the model's output. By using this sliding window method, the size of our training dataset is ~6.5GB with a sample size of 675,924, and the testing dataset is ~1.19GB with a sample size of 114,450.

## 6.3 Model details

There are six different configurations are considered for both MIMO-LSTM and MISO-LSTM models. Figure 2 depicts the general architecture of the proposed model. Each configuration has a different number of layers, and each layer consists of a different number of nodes. Each configuration is experimented with:

- Fixed learning rate (LR) and adaptive learning rate [69]. In the fixed learning rate, we set  $LR=0.01$ . In the adaptive learning rate method, the LR (initial  $LR=0.1$ ) is reduced to half of the current LR in every 20 epochs to find the optimal model with best LR.
- Adam [70] and SGD [71] optimizers to minimise a given cost function[59], [65].

The MIMO-TCN and MISO-TCN approaches have experimented with different configurations and controls, such as;

- Filter sizes: 32, 64, 128, 256, and 512
- Stacked TCN layers: 1, 2, 3, and 4 and
- With different activation functions such as 'linear' and 'tanh'

According to [8], [62], [73], the following controls are kept constant within these experiments as these do not impact on final results significantly in the regression model for time-series data; kernel size: 2, dilations: 7, where dilation values are: 1, 2, 4, 8, 16, 32, 64, batch size-64, and dropout rate-0, learning rate- 0.01.

## 6.4 Evaluation metric

The proposed deep regression models are evaluated using the most common metrics of Mean Squared Error (MSE), which is calculated as:

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_a - y_b)^2 \quad (3)$$

Where  $y_a$  is the actual expected output,  $y_b$  is the model's prediction, and  $n$  is number of samples.

## 6.5 Baseline approaches

Performances of the proposed LSTM and TCN models are compared with the following three types of baseline approaches. These approaches do not consider the temporal information rather count as another dimension in multivariate weather data.

- Classic machine learning approaches  
Standard Regression (SR), Support Vector Regression (SVR), and Random Forest (RF)
- Statistical machine learning approaches  
Autoregressive Integrated Moving Average (ARIMA), Vector Auto Regression (VAR), and Vector Error Correction Model (VECM)
- A dynamic ensemble method  
Arbitrage of Forecasting Expert (AFE)

We use both linear and RBF (Radial Basis Function) kernels for SVR in our experiments and use the grid search algorithm technique to optimize both  $C$  and  $\gamma$  parameters. In linear kernel, the parameter  $C$  is selected among the range  $[0.01 - 10000]$  with multiples of 10. In RBF kernel, the parameters  $C$  is selected as above but  $\gamma$  is selected among the range  $[0.0001, 0.001, 0.01, 0.1, 0.2, 0.5, 0.6, 0.9]$ . For RF [73], we select number of trees as  $[100, 250, 500]$ . For ARIMA model, we use the parameters  $p=2$ ,  $d=0$ , and  $q=1$  [74]. For VAR and VECM, the auto option is selected for weather forecasting [75], [76]. The given software package is used for the AFE [77].

The baseline performances are compared with the proposed LSTM and TCN networks. These models are evaluated using the testing dataset to select the optimal model or a model with the **least** MSE, which can be used as a tool for future forecasting. The selected optimal is used to forecast the weather parameters for the validation dataset (Model Prediction), and the model predicted values are evaluated with respect to the ground truth. Similarly, the WRF model has been run in forecast mode using the same format GRIB data for the month of July 2018 (WRF Prediction). These WRF predicted values are evaluated with respect to the ground truth. Then, we compare the model prediction and WRF prediction to determine the possibility to use the proposed model for short-term weather forecasting (i.e. 3-hour prediction). Then, the optimal model is re-tuned for long-term weather forecasting, such as 6, 9, 12, 24, and 48 hours. Similar to the short-term forecasting, we compare the model predictions and WRF predictions to determine up to what extent the proposed model can be used for weather forecasting.

## 7 Results and discussion

There are three types of results, namely: 1) a comparison of various machine learning techniques, statistical forecasting approaches, and a dynamic ensemble method with the proposed approach for weather forecasting, 2) performance of short-term weather forecasting, and 3) performance of long-term weather forecasting using the proposed model. More specifically, the short-term weather forecasting refers to 3-hours weather prediction, and long-term weather forecasting refers to 6-hours, 9-hours, 12-hours, 24-hours, and 48-hours weather predictions.

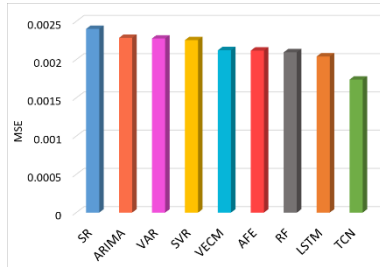
### 7.1 Comparison of machine learning techniques for short-term weather forecasting

As described in Section 6.5, we examine the classic machine learning approaches (i.e. SR, SVR, RF), statistical forecasting approaches (i.e. ARIMA, VAR, and VECM), and a dynamic ensemble method (i.e. AFE). Finally, we compare these performances with the proposed deep models (i.e. MISO-LSTM, MISO-TCN, MIMO-LSTM, MIMO-TCN) consisting of cutting-edge networks such as LSTM and TCN layers. As described in sections 5.1 and 5.2, these models are evaluated using two different regression types, namely MISO and MIMO.

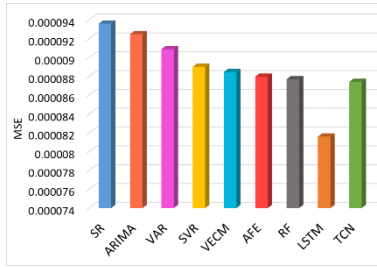
We evaluate the MISO models to determine the MISO-optimal with the least MSE for weather prediction. Table 3 and Figure 6 represent the comparison of machine learning approaches for MISO. As per information from Table 3 and Figure 6, the MISO-LSTM provides better performance with the least MSE for 6 parameters out of 10. Thus, the LSTM combined model with 10 parameters (i.e. MISO-LSTM) has been selected as the MISO proposed model.

Parameter	SR	SVR	ARIMA	VAR	VECM	AFE	RF	LSTM	TCN
<b>TSK</b>	0.002401549	0.002254852	0.002284599	0.002276563	0.002121585	0.002117985	0.002095814	0.002041361	<b>0.001738656</b>
<b>PSFC</b>	9.359E-05	8.90012E-05	9.2467E-05	9.08745E-05	8.8452E-05	8.79532E-05	8.76859E-05	<b>8.16E-05</b>	8.74041E-05
<b>U10</b>	0.005820971	0.005620015	0.00568859	0.005689786	0.005425125	0.005325658	0.00486305	<b>0.002748407</b>	0.004384032
<b>V10</b>	0.009827752	0.008465238	0.00920003	0.008902459	0.008625459	0.008612125	0.007865233	<b>0.003732091</b>	0.007427616
<b>Q2</b>	0.00698015	0.006901244	0.006976667	0.006885608	0.006841126	0.006827854	0.006795488	<b>0.006379222</b>	0.006752483
<b>Rainc</b>	0.004125379	0.003956207	0.004072365	0.004006562	0.003756566	0.003654545	0.003465855	<b>0.002799961</b>	0.003260107
<b>Rainnc</b>	0.021597916	0.019257844	0.016020204	0.015784525	0.013299866	0.013198986	0.009548721	<b>0.000502061</b>	0.001895714
<b>Snow</b>	1.65547E-06	9.87E-07	1.18746E-06	9.98926E-07	5.98926E-07	5.13656E-07	3.72155E-07	1.74E-07	<b>1.34E-07</b>
<b>TSLB</b>	0.000934762	0.000847989	0.000904632	0.000869562	0.000853657	0.000796566	0.000762486	0.000724035	<b>0.000376134</b>
<b>SMOIS</b>	0.000359895	0.000285655	0.000327851	0.000302515	0.000273252	0.000271652	0.000249451	0.00024636	<b>9.98907E-05</b>

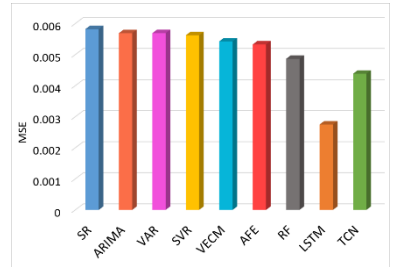
Table 3. Comparison of machine learning approaches for MISO. Lower MSE is better and is shown in **bold**



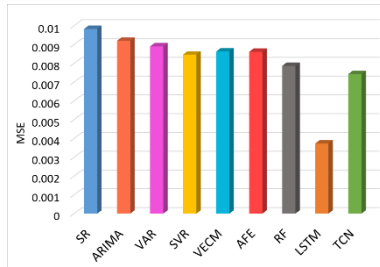
a) TSK



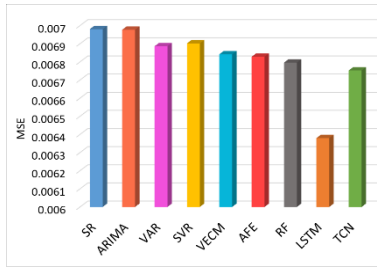
b) PSFC



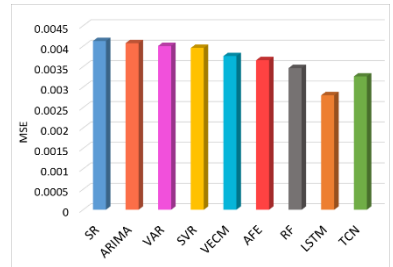
c) U10



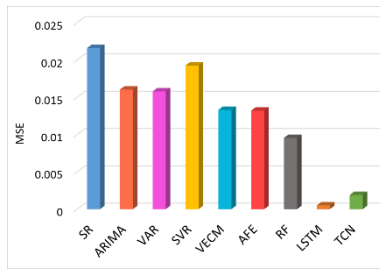
d) V10



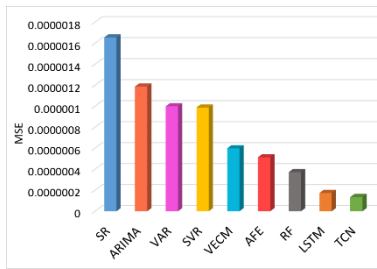
e) Q2



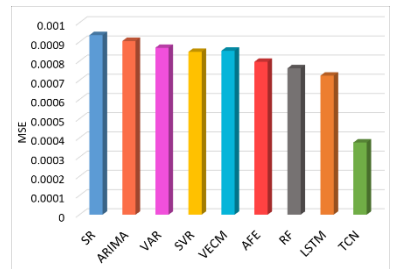
f) Rainc



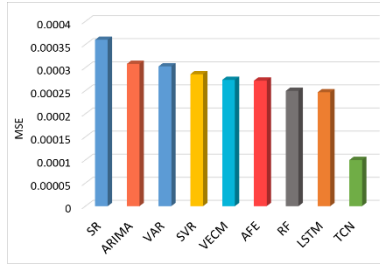
g) Rainnc



h) Snow



i) TSLB



j) SMOIS

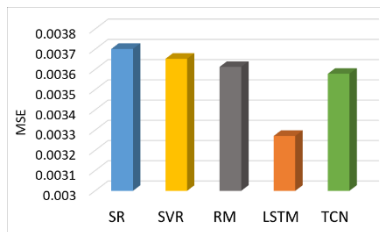
**Fig. 6.** MISO analysis of different approaches to predicting different weather parameters

(SR- Standard Regression, ARIMA- Autoregressive Integrated Moving Average, VAR-Vector Autoregression, SVR- Support Vector Regression, VECM- Vector Error Correction Model, AFE- Arbitrage of Forecasting Experts, RF- Random Forest, LSTM- Long Short-Term Memory, TCN- Temporal Convolutional Network)

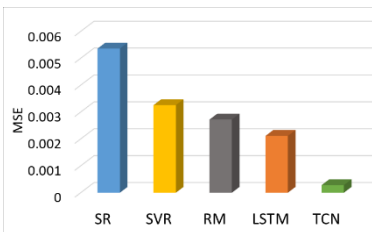
Similarly, we evaluate the MIMO models to determine the MIMO-optimal with the least MSE for weather prediction. Table 4 and Figure 7 represent the comparison of machine learning approaches for MISO. We do not consider the approaches ARIMA, VAR, VECM, and AFE in MIMO. Therefore, we compare SR, multi-output SVR [78], and RF with the proposed deep models MIMO-LSTM and MIMO-TCN. The results are subsequently evaluated via the Mean Squared Error. This is used to assess the best model (i.e. least MSE) after comparing the performance of all models.

Parameter	SR	SVR	RF	LSTM	TCN
<b>TSK</b>	0.003701561	0.003652545	0.003612458	<b>0.003271054</b>	0.003578392
<b>PSFC</b>	0.005358824	0.00325658	0.002720456	0.002112675	<b>0.000279068</b>
<b>U10</b>	0.008420962	0.006994518	0.006890641	<b>0.005394089</b>	0.00632667
<b>V10</b>	0.015627757	0.012985601	0.012056545	<b>0.006311009</b>	0.010195208
<b>Q2</b>	0.009980163	0.009979542	0.009929825	0.009881492	<b>0.006578324</b>
<b>Rainc</b>	0.006125415	0.005231452	0.005095621	<b>0.002878811</b>	0.004785024
<b>Rainnc</b>	0.021599896	0.016958456	0.016332367	<b>0.003070845</b>	0.015204848
<b>Snow</b>	1.65518E-05	9.72462E-06	7.27815E-06	2.39E-06	<b>1.60078E-06</b>
<b>TSLB</b>	0.004349349	0.003756588	0.003665241	0.003427306	<b>0.000485899</b>
<b>SMOIS</b>	0.000979024	0.000782515	0.000771265	<b>0.000648767</b>	0.000756974

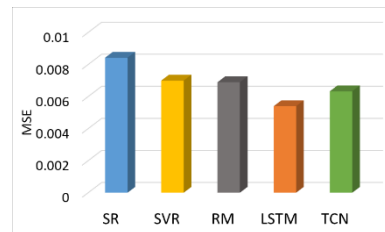
Table 4. Comparison of machine learning approaches for MIMO. Lower MSE is better and is shown in **bold**



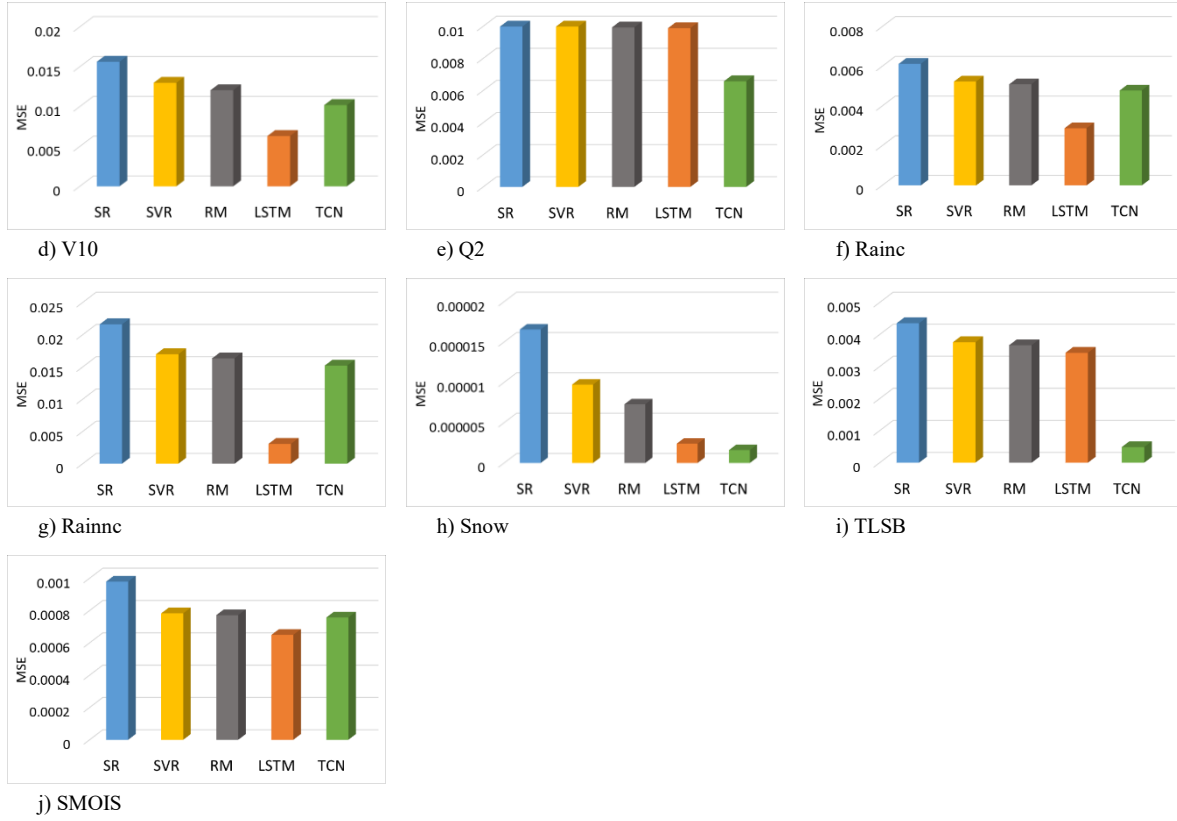
a) TSK



b) PSFC



c) U10



**Fig. 7.** MIMO analysis of different approaches to predicting different weather parameters (SR- Standard Regression, SVR- Support Vector Regression, RM- Random Forest, LSTM- Long Short-Term Memory, and TCN- Temporal Convolutional Network)

As per Table 4 and Figure 7, the MIMO-LSTM provides high accuracy output with least MSE for 6 parameters out of 10. Therefore, the MIMO-LSTM has been selected as the proposed model (i.e. MIMO-optimal).

In both MIMO and MISO, the LSTM and the TCN produce high performance with smaller errors compared to the classic machine learning approaches and statistical forecasting approaches as presented in Figure 6 and Figure 7. The reason is that the selected parameters do not follow a linear path within selected sequential timeslots [79], [80] and there is a non-linear interrelationship among parameters [6], [56], [81]. Besides, the sequential information is not encoded by the classic machine learning approaches and statistical forecasting models. The LSTM and TCN encode both multivariate and sequential information by taking them into another dimension in the input data [6], [62], [82].

## 7.2 Proposed models for short-term weather prediction

The least MSE for the MIMO is identified in the configuration with three LSTM layers, with 128, 512, and 256 number of nodes, respectively (i.e. MIMO-optimal model). We use the SGD optimiser with a fixed learning rate of 0.01 to optimise the MSE regression loss function. The model is trained for 230 epochs. In MISO, all these 10 models have different configurations with a different number of LSTM layers and nodes, activation functions, and optimisers (i.e. MISO-optimal). Table 5 and Figure 8 graphically represents the comparison of MSE in each variable for both MIMO-optimal and MISO-optimal.

Parameter	MIMO-optimal	MISO-optimal
TSK	3.27E-03	2.04E-03
PSFC	2.11E-03	8.16E-05
U10	5.39E-03	2.75E-03
V10	6.31E-03	3.73E-03
Q2	9.88E-03	6.38E-03
Rainc	2.87E-03	2.79E-03
Rainnc	3.07E-03	5.02E-04
Snow	2.39E-06	1.74E-07
TSLB	3.42E-03	7.24E-04
SMOIS	6.48E-04	2.46E-04

Table 5. MSE comparison

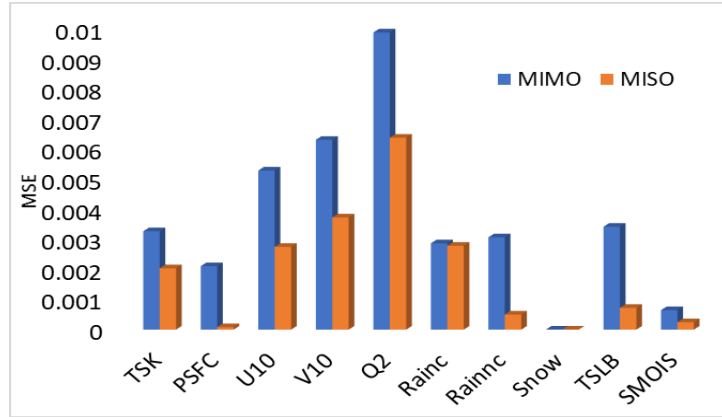


Fig.8. Comparison of MIMO and MISO

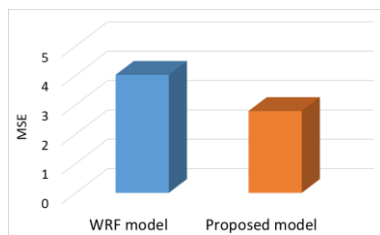
Table 5 shows the comparison of MSE in each variable for both MIMO and MISO. Figure 8 graphically represents these values to get an idea of whether to use the MIMO model or the MISO combined model to use as the best model for future predictions.

According to Figure 8, there is no major gap between MSE values for each variable when compare the MIMO-optimal and MISO-optimal. These differences are less than 0.04 for each variable. These error figures are significantly smaller. Moreover, the MISO-optimal requires 10 different models for the prediction of 10 different weather parameters. Therefore, we consider the MIMO-optimal (i.e. MIMO-LSTM) model as a tool for future forecasting since it is easier to handle and less time and power consumption (only one model to run) than running 10 different models of MISO-optimal.

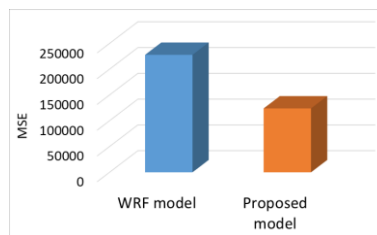
Parameter	Mean Squared Error	
	WRF Model	Proposed model (MIMO-LSTM)
TSK	4.0209727	<b>2.7882845</b>
PSFC	227869.02	<b>123881.22</b>
U10	10.540705	<b>5.327054</b>
V10	12.0824	<b>4.6248293</b>
Q2	1.1117266e <sup>-6</sup>	<b>7.716598e<sup>-7</sup></b>
Rainc	15.942339	<b>0.11341145</b>
Rainnc	18.627722	<b>0.83847433</b>
Snow	<b>0.0</b>	0.016857434
TSLB	8.140333	<b>2.6088953</b>
SMOIS	<b>8.523197e<sup>-5</sup></b>	0.000024246839

Table 6: Comparison of the proposed deep model with the WRF forecasting model for 3-hour prediction.

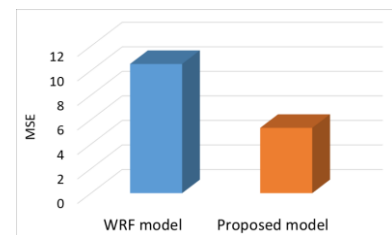
As described in Section 6.5, the validation dataset is utilised to get weather prediction using the proposed model. Similarly, the WRF model is run in forecast mode using the July 2018 data to compare results. Both WRF and model predicted values are compared with respect to the ground truth and calculated the MSE. Table 6 and Figure 9 represent the MSE comparison values for each variable.



a) TSK

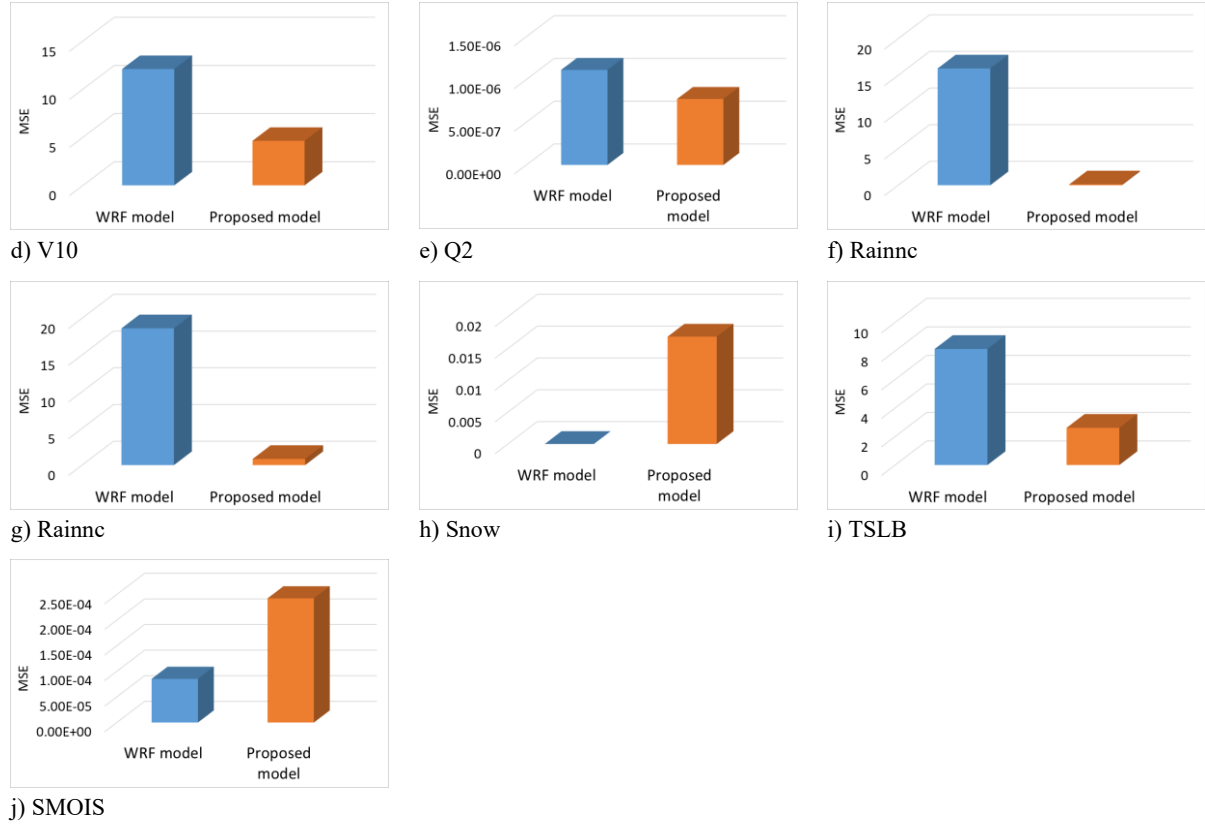


b) PSFC



c) U10

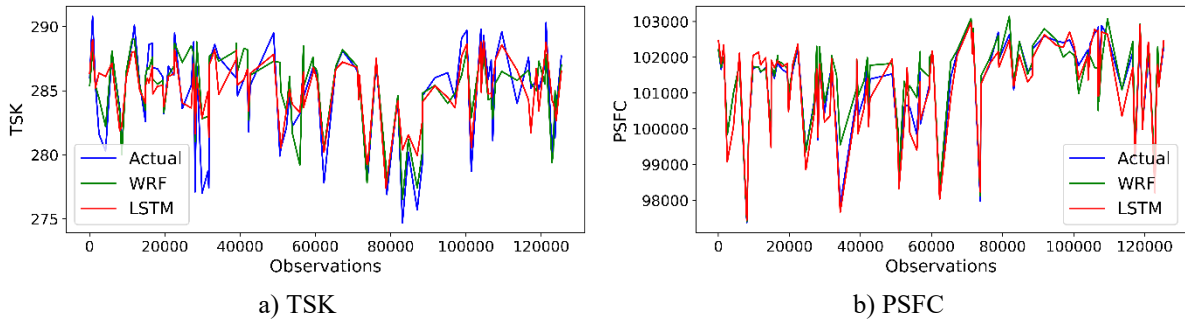


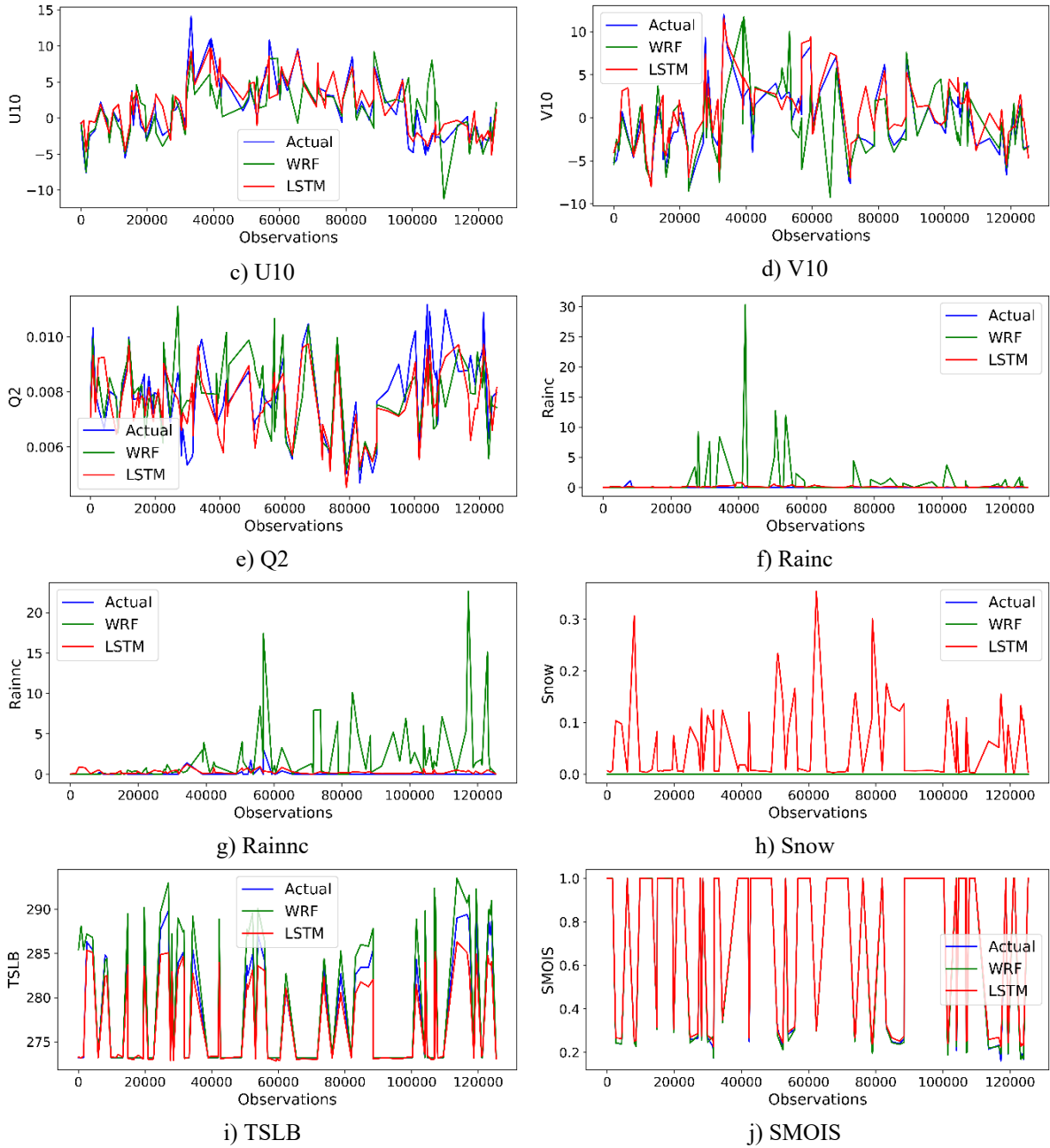


**Fig. 9.** Analysis of weather prediction of the WRF model and Proposed deep learning LSTM model.

When comparing Table 6 and Figure 9, the proposed deep model (i.e. MIMO-LSTM) provides comparatively best results (**bolded** in the table) on eight occasions out of 10. The WRF model provides the best results for the Snow and Soil Moisture (SMOIS) variables. On both occasions, these error figures are quite small. For example, MSE for the variable snow is  $0.0168574 \text{ kg/m}^2$ . This is quite a small and therefore, negligible. Similarly, the SMOIS has got a minimal and negligible error value. Figure 9k shows an overall comparison of both models.

As there are 125,373 samples in the July 2018 evaluation data, the proposed deep model and the WRF model will produce a similar number of samples as the predicted data. It is difficult to visualise all of these predictions because of the large sample size and therefore, a random sample of the 100 samples has been taken from the test set to compare with the respective ground truth. Figure 10 shows a comparison of the proposed deep model's predictions verses the WRF model predictions. For each graph, the ground truth, WRF prediction, and the proposed deep model's predictions are represented by each line with blue, green, and red colours, respectively.





**Fig. 10.** Comparison of WRF prediction vs the MIMO-LSTM model prediction for 100 random data samples with respect to the ground truth

As per Figure 10, the red line-chart (Deep model prediction) follows closely to the blue line-chart (ground truth) compared to the green-chart (WRF prediction). The WRF prediction is widely diverted in the parameters Rainc and Rainnc compared to the actual values. The deep model prediction is diverted in the parameter snow compared to the actual values. According to Figure 10 h), the highest snow prediction is 0.24 kg/m<sup>2</sup>. This is quite a small figure and can be negligible. Overall, the deep learning model provides a better short-term (up to 3 hours) prediction compared to the WRF model.

### 7.3 Proposed model for Long-term Weather forecasting

As described in Section 7.2, the proposed model (i.e. MIMO-LSTM) can be utilised for short-term weather forecasting, and it yields more accurate results compare to the well-known WRF model. In this section, our study is focused on exploring long-term weather prediction using the same historical weather data with 10 surface weather parameters.

### 7.3.1 Selection of an appropriate technique

As discussed in Section 7.1, the proposed model provides better performance compared to other machine learning techniques. Therefore, we use the same deep learning model with the LSTM layers for the long-term weather forecasting with the following variations. All these three variants use the same configuration and controls, which are comparable to the proposed MIMO-LSTM model.

- Load the MIMO-LSTM optimal model weights (3-hour) and fine-tune models for the long-term forecasting (shortened form: LSTM LW)
- Train models for each time frame without loading the optimal model weights (shortened form: LSTM WL). That is train the model at the beginning of the training dataset and new labels.
- We have also experimented with Bi-directional LSTM (Bi-LSTM). Compared to the LSTM, the Bi-LSTM has used two layers; one layer performs the operations following the forward direction (time-series data) of the data sequence, and the other layer applies its operations on in the reverse direction of the data sequence [83].

The following Table 7 shows the comparison of these three variations for each timeslot. As shown in Table 7, the Bi-LSTM provides slightly better results compared to the LSTM LW except for the timeslot 3-hour. The LSTM WL produces weaker results compared to the both LSTM LW. The reason is that the LSTM LW used its optimal weight, which is already configured to retrain and yield a prediction. Moreover, this is re-tune the model which is matched to the new dataset [58]. The Bi-LSTM is also trained the model at the beginning similar to the LSTM WL. However, the Bi-LSTM provides more accurate results due to the ability to preserve the past and future values [83].

The only drawback of the Bi-LSTM is that time taken to training, testing, and predicting data [84]. This is less efficient compared to the LSTM LW. Moreover, as can be observed in Table 7, there is a slight gap in the overall figures of MSE in both LSTM LW and Bi-LSTM. Therefore, we have selected the LSTM LW method for long-term forecasting for an effective and efficient outcome.

Parameter	LSTM LW	LSTM WL	Bi-LSTM
TSK	N/A	0.003271054	0.002371392
PSFC	N/A	0.002112675	0.001007641
U10	N/A	0.005394089	0.008889356
V10	N/A	0.006311009	0.010825
Q2	N/A	0.009881492	0.00885295
Rainc	N/A	0.002878811	0.004197211
Rainnc	N/A	0.003070845	0.025307791
Snow	N/A	2.39E-06	1.06E-06
TSLB	N/A	0.003427306	0.001056143
SMOIS	N/A	0.000648767	0.000677912
Overall	N/A	0.003699844	0.006318646

a) 3 Hour

Parameter	LSTM LW	LSTM WL	Bi-LSTM
TSK	0.004679656	0.003785324	0.002954833
PSFC	0.00337704	0.005435103	0.002543765
U10	0.016287696	0.01789222	0.015199178
V10	0.022693845	0.032980144	0.026619522
Q2	0.016228491	0.017330563	0.014454748
Rainc	0.007961646	0.007261488	0.006792006
Rainnc	0.08320849	0.087723635	0.0691833
Snow	2.13871E-06	1.39469E-05	1.86E-06
TSLB	0.002113115	0.002125454	0.002216928
SMOIS	0.001027248	0.001156121	0.00075886
Overall	0.015757935	0.017570399	0.014072499

b) 9 Hour

Parameter	LSTM LW	LSTM WL	Bi-LSTM
TSK	0.003225559	0.003989982	0.003520491
PSFC	0.012482793	0.010315491	0.007714262
U10	0.026440082	0.026202237	0.024926782
V10	0.03660787	0.042136274	0.036013693
Q2	0.026067492	0.030222168	0.02755576
Rainc	0.08263268	0.07865509	0.078575564
Rainnc	0.15932418	0.16492906	0.158401
Snow	4.65178E-07	0.000137658	0.000442552
TSLB	0.004401047	0.004503616	0.005910429
SMOIS	0.001600785	0.001434334	0.001202147
Overall	0.035278295	0.03625259	0.034456268

c) 24 Hour

Parameter	LSTM LW	LSTM WL	Bi-LSTM
TSK	0.004480547	0.005869389	0.003708232
PSFC	0.018504778	0.013365718	0.016115312
U10	0.045134	0.037737582	0.03475978
V10	0.04253545	0.04715329	0.042574175
Q2	0.050479617	0.04151997	0.038551033
Rainc	0.061815947	0.068089165	0.059418406
Rainnc	0.16204703	0.16313162	0.15197921
Snow	3.72323E-06	0.000231712	6.40E-05
TSLB	0.007845704	0.012153346	0.005880864
SMOIS	0.00158867	0.001342647	0.001104945
Overall	0.039443548	0.038059445	0.03541559

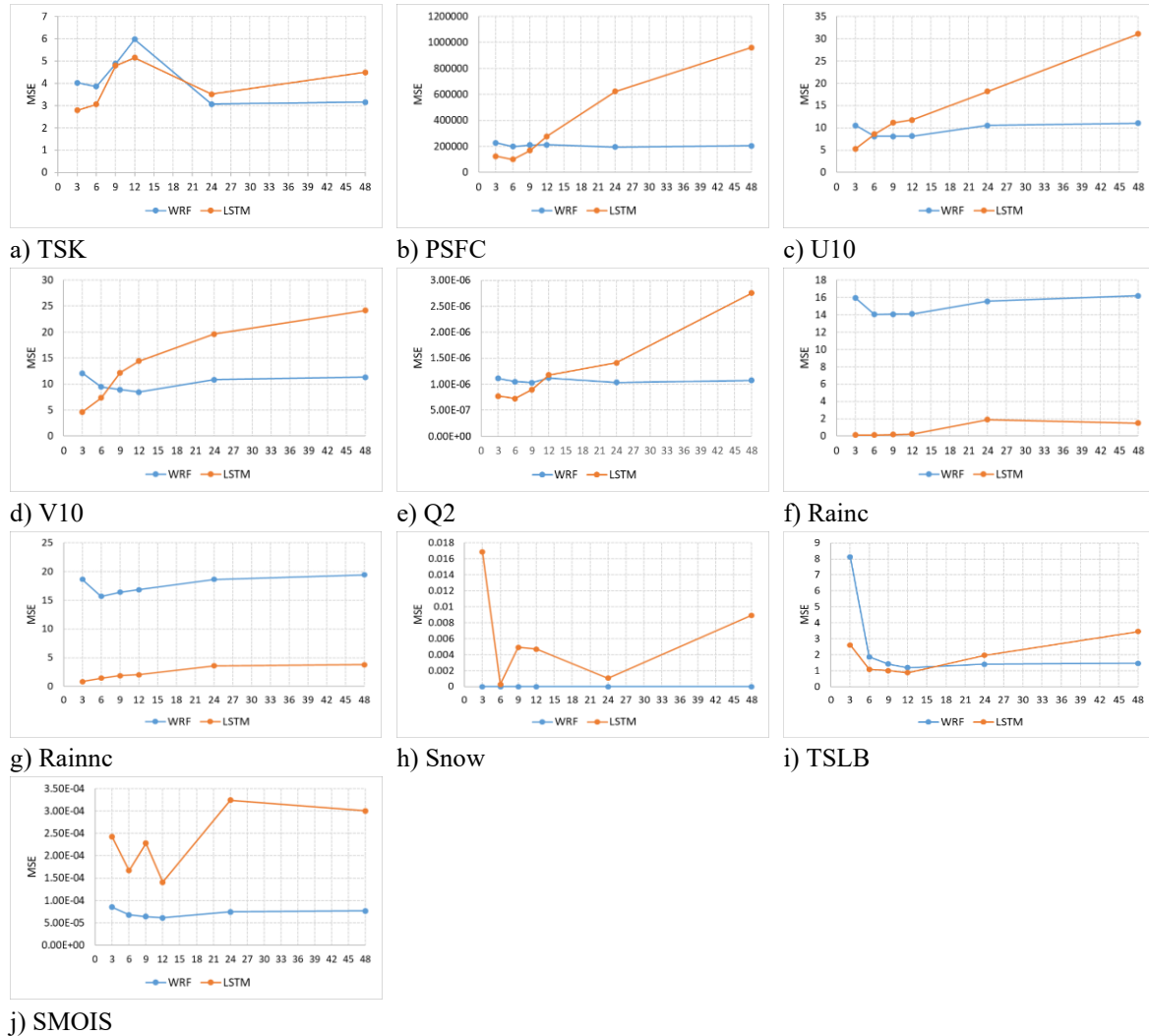
f) 48 Hour

Table 7. Comparison of LSTM LW, LSTM WL, and Bi-LSTM. Only included the results for 3, 9, 24, and 48 hours. The other tables are included in the supplementary section.

### 7.3.2 Long-Term weather forecasting

The proposed model (i.e. MIMO-LSTM) consists of three LSTM layers with other controls. As described in the section 7.3.1 the LSTM with loading the optimal weight method is used for the long-term weather prediction. Therefore, the optimal model is re-tuned (i.e. load optimal model weight and re-train models) for timeslots 3-hour, 6-hour, 9-hour, 12-hour, 24-hour, and 48-hour. While re-tuning, the optimal models are found in different epochs such as 80, 10, 10, 10, and 10 for timeslots 6, 9, 12, 24, and 48 hours, respectively.

Similar to the short-term weather forecasting, the optimal model for each timeslot is used to forecast the weather parameters for the July 2018 data (model prediction), and the model predicted values are evaluated with respect to the ground truth. The WRF model has been run in forecast mode using the same format GRIB data for the month of July 2018 (WRF prediction) based on the same conditions as model prediction (i.e. input seven days data and predict weather parameters for timeslot 6, 9, 12, 24 and 48). The WRF predicted values are evaluated with respect to the ground truth. Finally, compare the model prediction and WRF prediction to determine what extent the deep learning model can be used for weather forecasting. Figure 11 shows a comparison of MSE values related to the proposed model and the WRF model for each time slot.

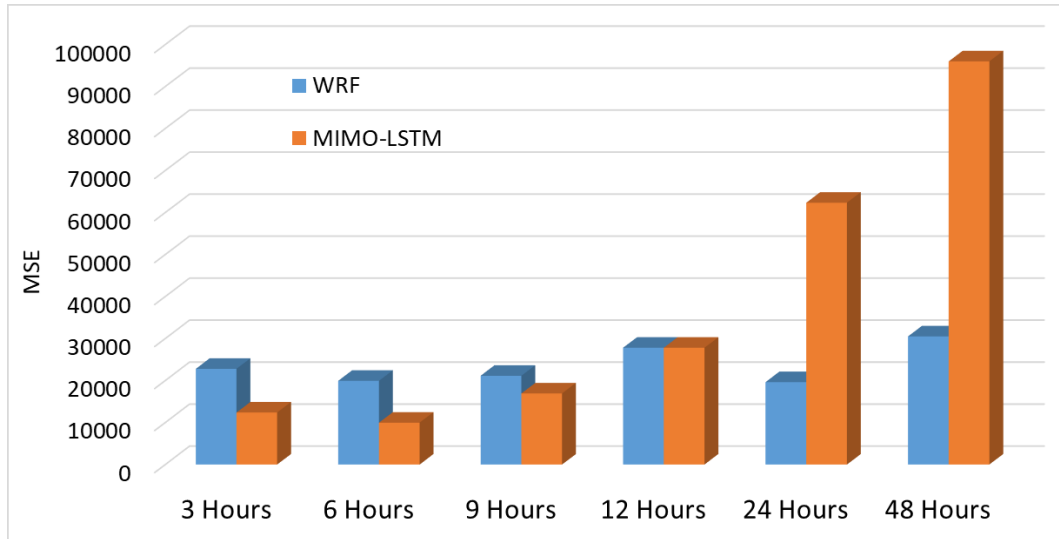


**Fig. 11.** Compare proposed MIMO-LSTM model prediction with WRF prediction for long-term forecasting. The MSE values are calculated with respect to the ground truth in both WRF and LSTM models.

According to the results presented in Figure 11, it is obvious that the WRF model produces better forecasting results for the very long-term compared to the deep learning model. The reason is that the WRF model is combined with many other climate models [4], [85], [86] and data is coming to the system globally [4], [52]. The deep

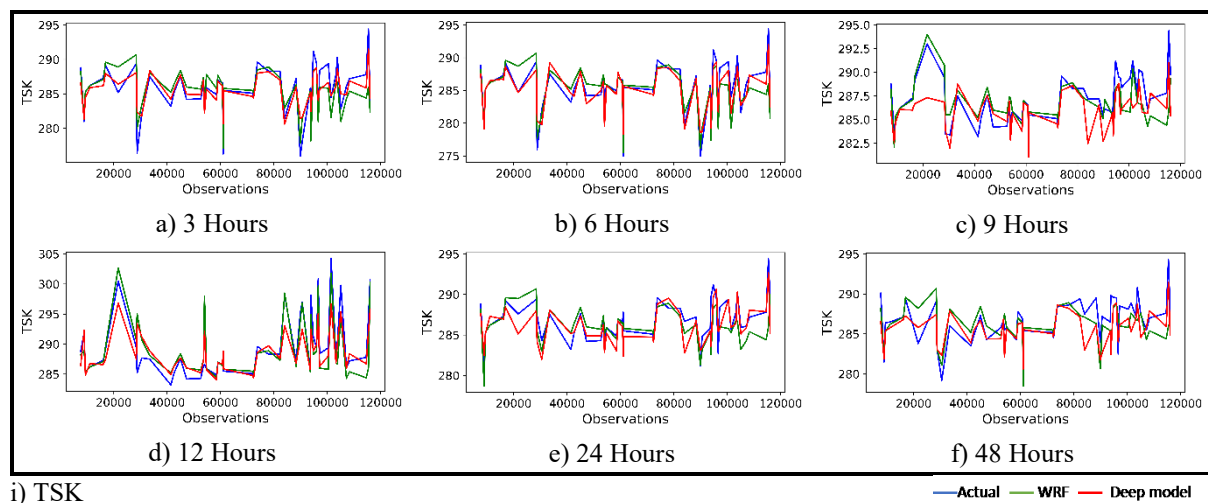
learning model has predicted these outputs based on five months of training data. We could receive better results if we increase the size of the training dataset [59]. The Rainc and Rainnc parameters show much better results in the deep learning model compared to the WRF model for long-term forecasting. The experiments of [40] already proved that the deep learning neural networks yield the highest accuracy for rain prediction.

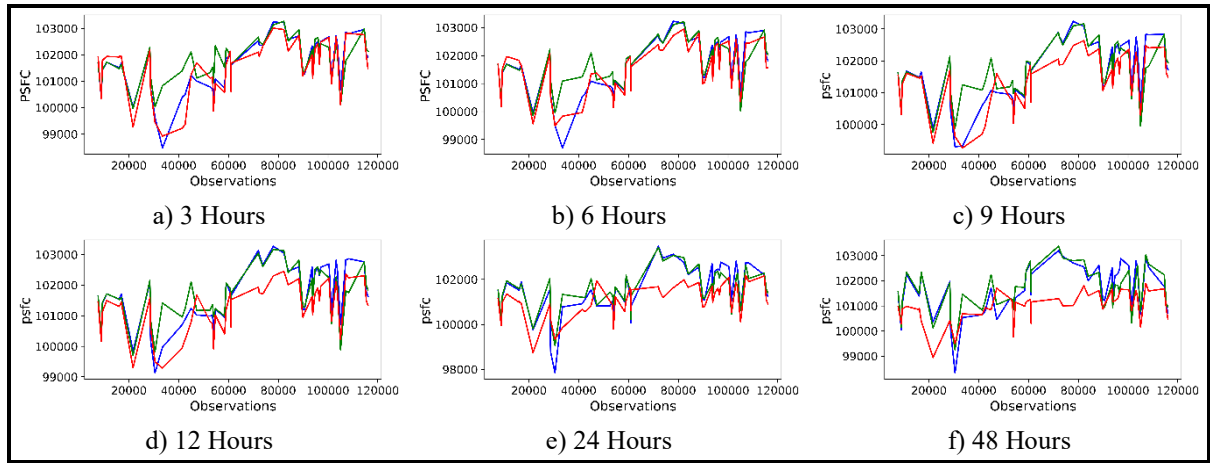
Contrarily, the SMOIS and snow parameters show weak results in deep learning compared to the WRF model at all timeslots. Simply, these error patterns are rather low (maximum error: Snow-0.016kg/m<sup>2</sup>, SMOIS- 0.00035 m<sup>3</sup>/m<sup>3</sup>) and can be negligible. This could be resolved by increasing the size of the sample data. All other occasions, the deep learning model provide more accurate prediction compared to the WRF model up to some extent, than the WRF model produces better prediction compared to the deep learning model. Figure 12 shows the comparison of overall error values of the WRF model and proposed deep learning model.



**Fig. 12.** Comparison of overall MSE for each timeslot

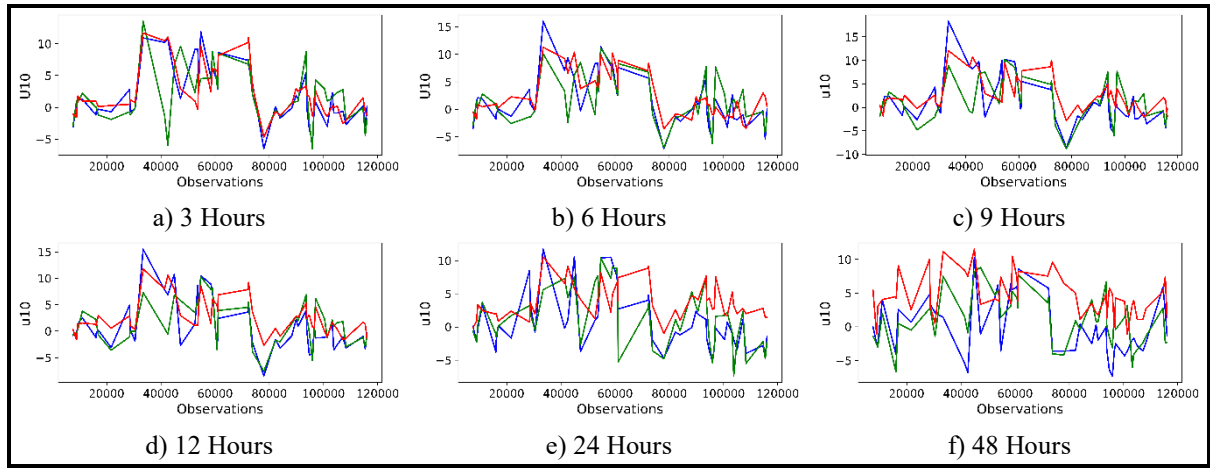
As indicated in Figure 12, the deep learning model produces better predictions compared to the WRF model prediction up to 12 hours overall. Therefore, we can use deep learning with LSTM model up to 12 hours of weather forecasting much accurately compared to the well-recognised WRF model. The comparison of WRF prediction vs the LSTM model prediction for 50 random data samples with respect to the ground truth is shown in Figure 13. For each graph, the ground truth, WRF prediction, and the proposed deep model's predictions are represented by each line with blue, green, and red colours, respectively.





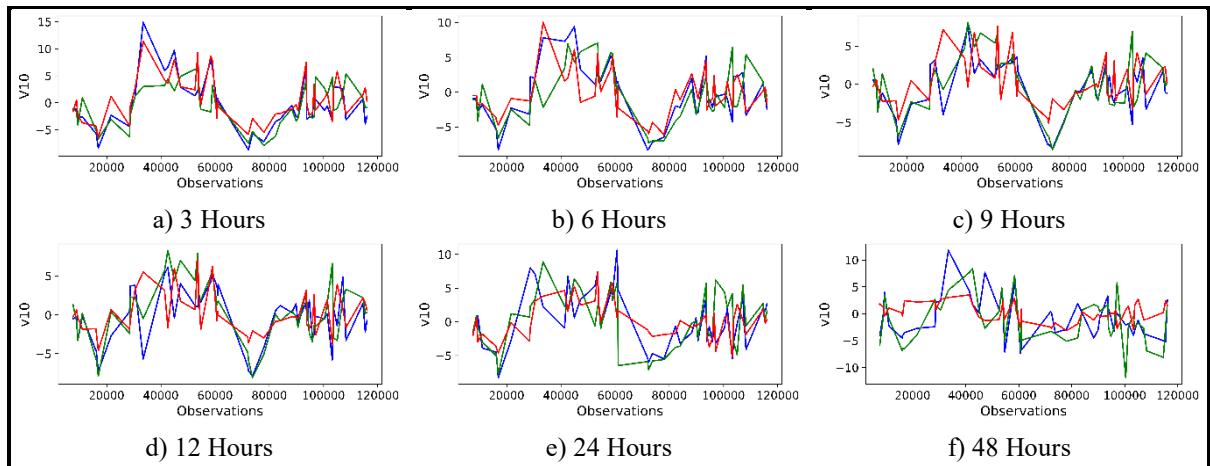
ii) PSFC

— Actual — WRF — Deep model



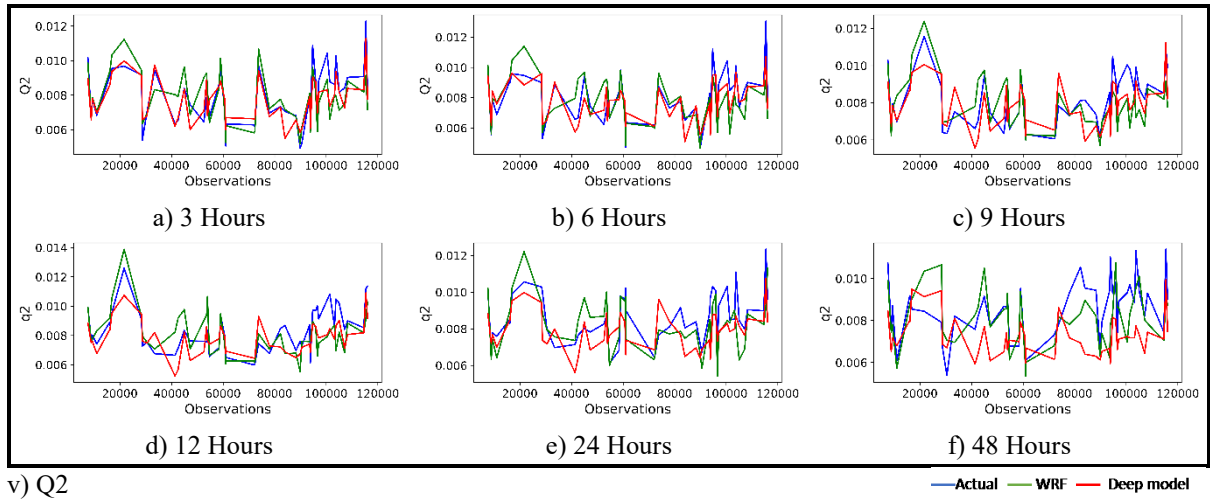
iii) U10

— Actual — WRF — Deep model

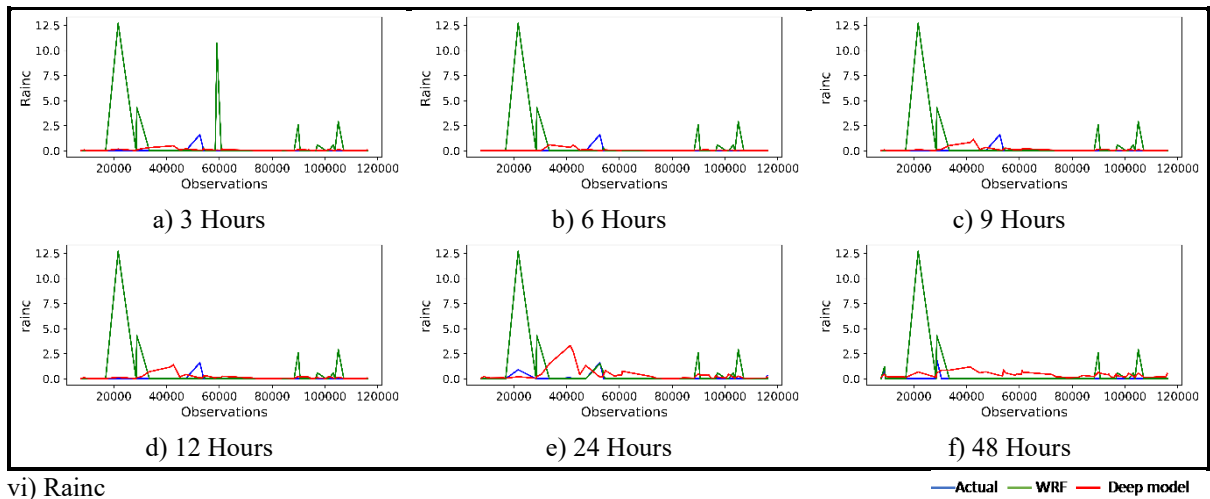


iv) V10

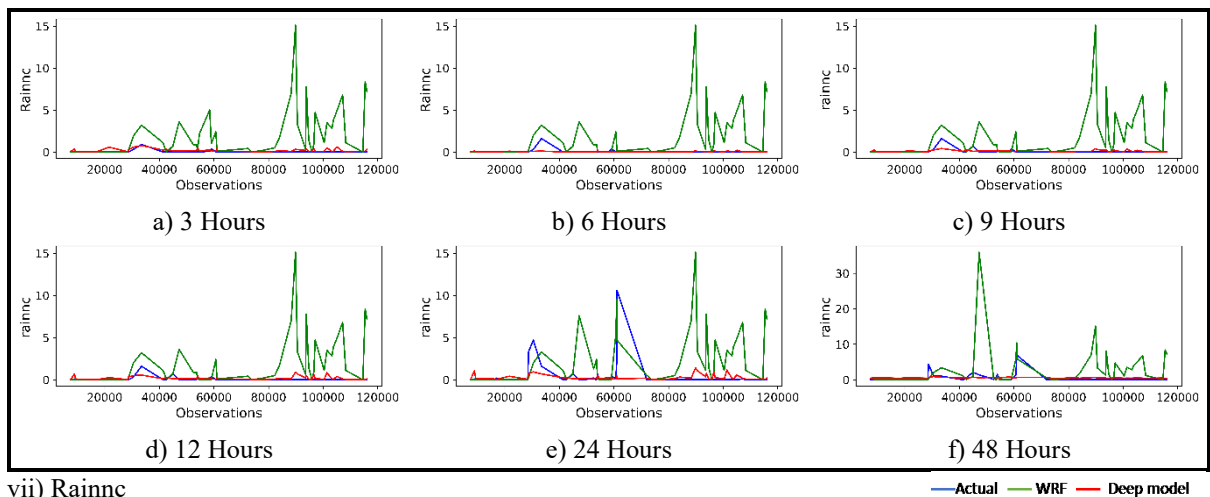
— Actual — WRF — Deep model



v) Q2

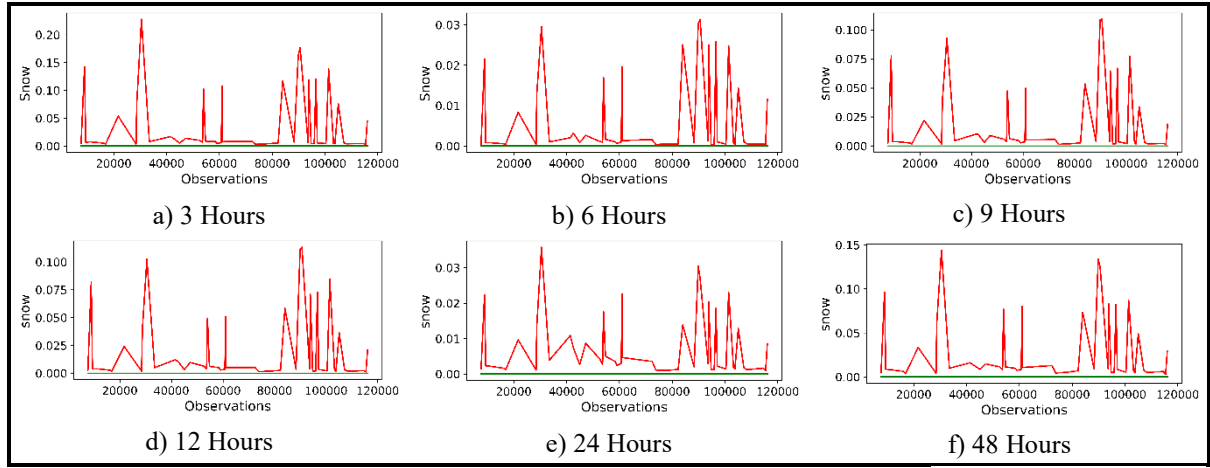


vi) Rainc



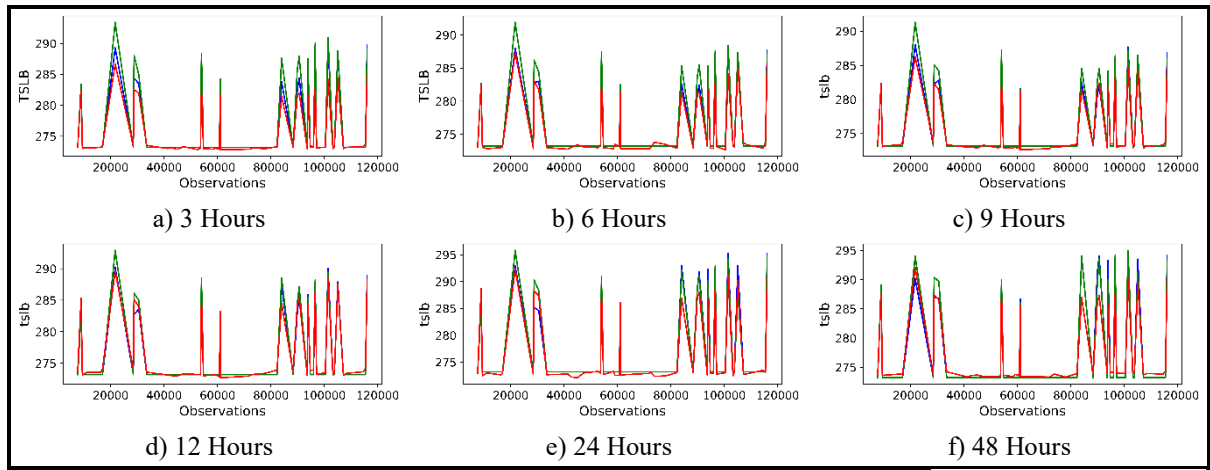
vii) Rainnc





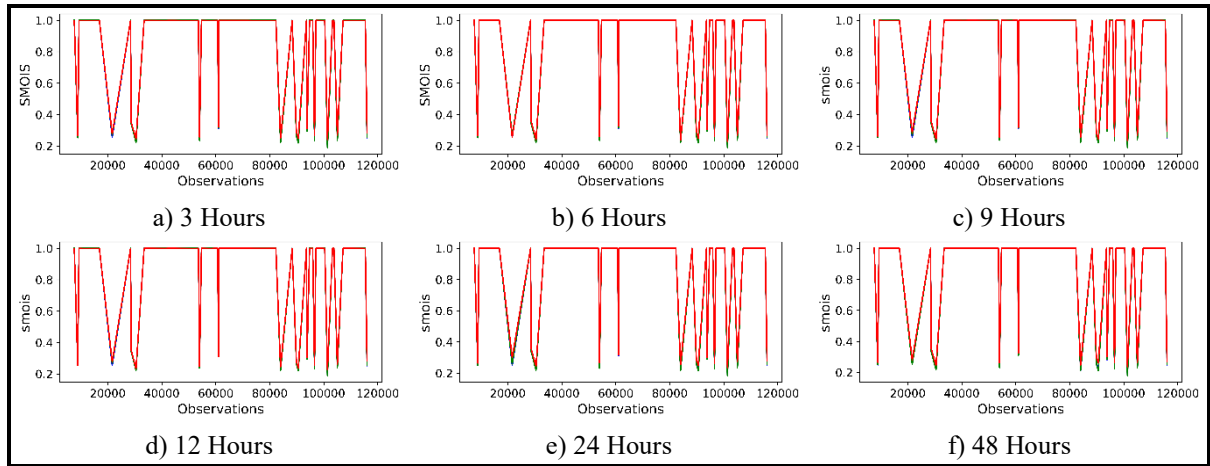
viii) Snow

— Actual — WRF — Deep model



ix) TSLB

— Actual — WRF — Deep model



x) SMOIS

— Actual — WRF — Deep model

**Fig. 13.** Comparison of WRF prediction vs the LSTM model prediction for 50 random data samples with respect to the ground truth

As per Figure 13, the red line-chart (deep model prediction) followed closely to the blue line-chart (ground truth) up to some extent and diverted when time increases in many parameters. The green line-chart (WRF model prediction) also diverted from the blue line-chart when time increased, but this diversion is relatively small

compared with the red line-chart. As shown in Figure 13 (vi) and 12 (vii), the rainc and rainnc values are accurate in the deep learning model compared to the WRF model for up to 48 hours. As discussed earlier, The WRF model produces a better prediction for the Snow and SMOIS parameters. As shown in Figure 13 (x), the difference is negligible for the parameter SMOIS. As shown in Figure 13 (viii), the maximum snow values are shown in the 3 hours line-chart. This value is equal to  $0.24 \text{ kg/m}^2$  and this is a relatively negligible figure. Overall, the deep learning model delivers a better forecasting prediction compared to the WRF model for up to 12 hours.

## 7.4 Applicability of the new model

As described in Section 7.3, the proposed model can be used for weather prediction. Even, this model generates more accurate predictions compared to the well-recognised WRF model for up to 12 hours. We use historical weather data to evaluate and validate these models. The only issue is we still use the WRF model to extract GRIB data to use as input for the new model (we use GFS GRIB data). On the other hand, it requires a minimum of three hours of access GFS data after taking the atmospheric measurements. This includes the time taken to upload data to the website [4], [87]. In addition, the WRF model also taken the time to extract the GFS data depend on the computer system. Hence, the input data which are used in the new model are not the current atmospheric measurement data (i.e. older more than 3 hours). Therefore, it is not practicable to use WRF data with the new model, and it will be highly beneficial to consider the use of local weather station data for weather forecasting.

## 8 Conclusion and Future work

In this article, we demonstrate that the proposed lightweight deep model can be utilised for weather forecasting up to 12 hours for 10 surface weather parameters. The model outperformed the state-of-the-art WRF model for up to 12 hours. The proposed model could run on a standalone computer, and it could easily be deployed in a selected geographical region for fine-grained short to medium-term weather prediction. Furthermore, the proposed model is able to overcome some challenges within the WRF model, such as the understanding of the model and its installation, as well as its execution and portability. In particular, the deep model is portable and can be easily installed into a Python environment for effective results [17], [59]. This process is highly efficient compared to the WRF model.

This research is carried out using ten different surface weather parameters, and an increased number of inputs would probably lead to enhanced results. For example, there are 36 different pressure levels defined in the WRF model [17]. Only the pressure at two meters is considered within this research. There is a possibility to increase the accuracy of the results if we introduce all 36 possible pressure levels to the proposed model. However, it will increase the model complexity requiring a large number of parameters to estimate. Furthermore, January to May weather data is utilised for training the deep model, and the increase in the size of training dataset could help towards improved results in a deep learning network [59], [88].

Besides, we used the MIMO approach within this research to predict weather data. Table 4 and Figure 7 shows that the MISO approach produces better MSE values compared to the MIMO. Therefore, there is a huge potential that the MIMO approach will increase the accuracy of the results; even this method is less efficient compared to the MIMO. Besides, the Bi-LSTM yields high accuracy long-term prediction compared to the LSTM, as presented in Table 6. Therefore, we could get more accurate results if we use Bi-LSTM; even this method is not efficient due to high time-consumption.

These experiments show that we can apply the neural network approach for weather prediction. Based on the geographical appearance of location (such as the top of a mountain, land covered by several mountains, the slope of the land, etc.) the regional weather forecasting may not be accurate. As a solution, we could develop a lightweight (neural network based) short-term weather forecasting system for the community of users utilising weather station data. These are our future experimentation.

## 9 Acknowledgement

This research work is partly supported by Clive Blaker and Rich Kavanagh (Precision Decisions Ltd). We are grateful to Prof Mark Anderson (Former Professor at Edge Hill University) for his valuable inputs and support at the initial stage of this study. We are also grateful to Dr Alan Gadian (National Centre for Atmospheric Sciences, University of Leeds) for his valuable support to identify weather parameters.

## 10 References

- [1] M. Hayati and Z. Mohebi, ‘Application of Artificial Neural Networks for Temperature Forecasting’, *Int. J. Electr. Comput. Eng.*, vol. 1, no. 4, p. 5, 2007.
- [2] P. Lynch, *The emergence of numerical weather prediction: Richardson’s dream*. Cambridge: Cambridge Univ. Press, 2006.
- [3] L. Oana and A. Spataru, ‘Use of Genetic Algorithms in Numerical Weather Prediction’, in *2016 18th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing (SYNASC)*, Timisoara, Romania, Sep. 2016, pp. 456–461, doi: 10.1109/SYNASC.2016.075.
- [4] NCAR/UCAR, ‘WRF Model Users Site’, 2019. <http://www2.mmm.ucar.edu/wrf/users/> (accessed Jan. 21, 2019).
- [5] J. G. Powers *et al.*, ‘The Weather Research and Forecasting Model: Overview, System Efforts, and Future Directions’, *Bull. Am. Meteorol. Soc.*, vol. 98, no. 8, pp. 1717–1737, Jan. 2017, doi: 10.1175/BAMS-D-15-00308.1.
- [6] R. Jozefowicz, W. Zaremba, and I. Sutskever, ‘An Empirical Exploration of Recurrent Network Architectures’, *Int. Conf. Mach. Learn.*, pp. 2342–2350, 2015.
- [7] T. S. Kim and A. Reiter, ‘Interpretable 3D Human Action Analysis with Temporal Convolutional Networks’, *ArXiv170404516 Cs*, Apr. 2017, Accessed: Feb. 22, 2019. [Online]. Available: <http://arxiv.org/abs/1704.04516>.
- [8] C. Lea, M. D. Flynn, R. Vidal, A. Reiter, and G. D. Hager, ‘Temporal Convolutional Networks for Action Segmentation and Detection’, in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, HI, Jul. 2017, pp. 1003–1012, doi: 10.1109/CVPR.2017.113.
- [9] G. E. Hinton, ‘Reducing the Dimensionality of Data with Neural Networks’, *Science*, vol. 313, no. 5786, pp. 504–507, Jul. 2006, doi: 10.1126/science.1127647.
- [10] T. Choi, C. Hui, and Y. Yu, ‘Intelligent time series fast forecasting for fashion sales: A research agenda’, in *2011 International Conference on Machine Learning and Cybernetics*, Jul. 2011, vol. 3, pp. 1010–1014, doi: 10.1109/ICMLC.2011.6016870.
- [11] X. Shi, Z. Chen, H. Wang, D.-Y. Yeung, W. Wong, and W. WOO, ‘Convolutional LSTM Network: A Machine Learning Approach for Precipitation Nowcasting’, in *Advances in Neural Information Processing Systems 28*, C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, Eds. Curran Associates, Inc., 2015, pp. 802–810.
- [12] N. US Department of Commerce, ‘Model Output Statistics (MOS)’. [https://www.weather.gov/mdl/mos\\_home](https://www.weather.gov/mdl/mos_home) (accessed Jul. 05, 2019).
- [13] H. R. Glahn and D. A. Lowry, ‘The Use of Model Output Statistics (MOS) in Objective Weather Forecasting’, *J. Appl. Meteorol.*, vol. 11, no. 8, pp. 1203–1211, Dec. 1972, doi: 10.1175/1520-0450(1972)011<1203:TUOMOS>2.0.CO;2.
- [14] F. Woodcock, ‘Australian Experimental Model Output Statistics Forecasts of Daily Maximum and Minimum Temperature’, *Mon. Weather Rev.*, vol. 112, no. 10, pp. 2112–2121, Oct. 1984, doi: 10.1175/1520-0493(1984)112<2112:AEMOSF>2.0.CO;2.
- [15] S. Kruizinga and A. H. Murphy, ‘Use of an Analogue Procedure to Formulate Objective Probabilistic Temperature Forecasts in The Netherlands’, *Mon. Weather Rev.*, vol. 111, no. 11, pp. 2244–2254, Nov. 1983, doi: 10.1175/1520-0493(1983)111<2244:UOAAPT>2.0.CO;2.
- [16] R. E. Abdel-Aal and M. A. Elhadidy, ‘Modeling and Forecasting the Daily Maximum Temperature Using Abductive Machine Learning’, *Weather Forecast.*, vol. 10, no. 2, pp. 310–325, Jun. 1995, doi: 10.1175/1520-0434(1995)010<0310:MAFTDM>2.0.CO;2.

- [17] C. Skamarock *et al.*, ‘A Description of the Advanced Research WRF Version 3’, 2008, doi: 10.5065/D68S4MVH.
- [18] C. N. Schizas, S. Michaelides, C. S. Pattichis, and R. R. Livesay, ‘Artificial neural networks in forecasting minimum temperature (weather)’, in *1991 Second International Conference on Artificial Neural Networks*, Nov. 1991, pp. 112–114.
- [19] D. E. Rumelhart and J. L. McClelland, ‘The PDP Perspective’, in *Parallel Distributed Processing: Explorations in the Microstructure of Cognition: Foundations*, MITP, 1987.
- [20] K. Ochiai, H. Suzuki, K. Shinozawa, M. Fujii, and N. Sonehara, ‘Snowfall and rainfall forecasting from weather radar images with artificial neural networks’, in *Proceedings of ICNN’95 - International Conference on Neural Networks*, Nov. 1995, vol. 2, pp. 1182–1187 vol.2, doi: 10.1109/ICNN.1995.487781.
- [21] S. Oishi, S. Ikebuchi, and T. Kojiri, ‘Severe rainfall prediction method using artificial intelligence’, in *SMC’98 Conference Proceedings. 1998 IEEE International Conference on Systems, Man, and Cybernetics (Cat. No.98CH36218)*, Oct. 1998, vol. 5, pp. 4820–4825 vol.5, doi: 10.1109/ICSMC.1998.727615.
- [22] H. Qi and M. Zhang, ‘Rainfall estimation using M-PHONN model’, in *IJCNN’01. International Joint Conference on Neural Networks. Proceedings (Cat. No.01CH37222)*, Jul. 2001, vol. 3, pp. 1620–1624 vol.3, doi: 10.1109/IJCNN.2001.938403.
- [23] M. Jaruszewicz and J. Mandziuk, ‘Application of PCA method to weather prediction task’, in *Proceedings of the 9th International Conference on Neural Information Processing, 2002. ICONIP ’02.*, Nov. 2002, vol. 5, pp. 2359–2363 vol.5, doi: 10.1109/ICONIP.2002.1201916.
- [24] I. Maqsood, M. R. Khan, and A. Abraham, ‘Intelligent Weather Monitoring Systems Using Connectionist Models’, vol. 10, p. 21, 2002.
- [25] K. Li and Y. S. Liu, ‘A rough set based fuzzy neural network algorithm for weather prediction’, in *2005 International Conference on Machine Learning and Cybernetics*, Aug. 2005, vol. 3, pp. 1888–1892 Vol. 3, doi: 10.1109/ICMLC.2005.1527253.
- [26] S. Mathur, A. Kumar, and M. Ch, *International Journal of Computational Intelligence - A Feature Based Neural Network Model for Weather Forecasting*. 2008.
- [27] K. Abhishek, M. P. Singh, S. Ghosh, and A. Anand, ‘Weather Forecasting Model using Artificial Neural Network’, *Procedia Technol.*, vol. 4, pp. 311–318, Jan. 2012, doi: 10.1016/j.protcy.2012.05.047.
- [28] B. S. P. Reddy, K. V. Kumar, B. M. Reddy, and N. Raja, ‘ANN Approach for Weather Prediction using Back Propagation’, 2012.
- [29] G. Shrivastava, S. Karmakar, M. Kumar Kowar, and P. Guhathakurta, ‘Application of Artificial Neural Networks in Weather Forecasting: A Comprehensive Literature Review’, *Int. J. Comput. Appl.*, vol. 51, no. 18, pp. 17–29, Aug. 2012, doi: 10.5120/8142-1867.
- [30] R. Nayak, P. S. Patheja, and A. A. Wao, ‘An artificial neural network model for weather forecasting in Bhopal’, in *IEEE-International Conference On Advances In Engineering, Science And Management (ICAESM -2012)*, Mar. 2012, pp. 747–749.
- [31] K. Abhishek, M. P. Singh, S. Ghosh, and A. Anand, ‘Weather Forecasting Model using Artificial Neural Network’, *Procedia Technol.*, vol. 4, pp. 311–318, Jan. 2012, doi: 10.1016/j.protcy.2012.05.047.
- [32] A. Ahmadi, Z. Zargar, A. Mohebi, and F. Taghavi, ‘Hybrid model for weather forecasting using ensemble of neural networks and mutual information’, in *2014 IEEE Geoscience and Remote Sensing Symposium*, Jul. 2014, pp. 3774–3777, doi: 10.1109/IGARSS.2014.6947305.
- [33] J. N. K. Liu, Y. Hu, J. J. You, and P. W. Chan, ‘Deep Neural Network Based Feature Representation for Weather Forecasting’, *Proc. Int. Conf. Artif. Intell. ICAI*, p. 7, 2014.
- [34] A. Troncoso, S. Salcedo-Sanz, C. Casanova-Mateo, J. C. Riquelme, and L. Prieto, ‘Local models-based regression trees for very short-term wind speed prediction’, *Renew. Energy*, vol. 81, pp. 589–598, Sep. 2015, doi: 10.1016/j.renene.2015.03.071.
- [35] M. Dalto, J. Matuško, and M. Vašak, ‘Deep neural networks for ultra-short-term wind forecasting’, in *2015 IEEE International Conference on Industrial Technology (ICIT)*, Mar. 2015, pp. 1657–1663, doi: 10.1109/ICIT.2015.7125335.

- [36] M. Hossain, B. Rekabdar, S. J. Louis, and S. Dascalu, 'Forecasting the weather of Nevada: A deep learning approach', in *2015 International Joint Conference on Neural Networks (IJCNN)*, Jul. 2015, pp. 1–6, doi: 10.1109/IJCNN.2015.7280812.
- [37] M. Akram and C. El, 'Sequence to Sequence Weather Forecasting with Long Short-Term Memory Recurrent Neural Networks', *Int. J. Comput. Appl.*, vol. 143, no. 11, pp. 7–11, Jun. 2016, doi: 10.5120/ijca2016910497.
- [38] C. Voyant *et al.*, 'Machine learning methods for solar radiation forecasting: A review', *Renew. Energy*, vol. 105, pp. 569–582, May 2017, doi: 10.1016/j.renene.2016.12.095.
- [39] A. Sharaff and S. R. Roy, 'Comparative Analysis of Temperature Prediction Using Regression Methods and Back Propagation Neural Network', in *2018 2nd International Conference on Trends in Electronics and Informatics (ICOEI)*, May 2018, pp. 739–742, doi: 10.1109/ICOEI.2018.8553803.
- [40] K. Yonekura, H. Hattori, and T. Suzuki, 'Short-term local weather forecast using dense weather station by deep neural network', in *2018 IEEE International Conference on Big Data (Big Data)*, Dec. 2018, pp. 1683–1690, doi: 10.1109/BigData.2018.8622195.
- [41] K. Patil and M. C. Deo, 'Basin-Scale Prediction of Sea Surface Temperature with Artificial Neural Networks', in *2018 OCEANS - MTS/IEEE Kobe Techno-Oceans (OTO)*, May 2018, pp. 1–5, doi: 10.1109/OCEANSKOBE.2018.8558780.
- [42] N. J. Rodríguez-Fernández *et al.*, 'SMOS Neural Network Soil Moisture Data Assimilation', in *IGARSS 2018 - 2018 IEEE International Geoscience and Remote Sensing Symposium*, Jul. 2018, pp. 5548–5551, doi: 10.1109/IGARSS.2018.8519377.
- [43] A. G. Salman, B. Kanigoro, and Y. Heryadi, 'Weather forecasting using deep learning techniques', presented at the 2015 international conference on advanced computer science and information systems (ICACSIS), 2015, pp. 281–285.
- [44] Y. Deng, H. Jia, P. Li, X. Tong, X. Qiu, and F. Li, 'A Deep Learning Methodology Based on Bidirectional Gated Recurrent Unit for Wind Power Prediction', presented at the 2019 14th IEEE Conference on Industrial Electronics and Applications (ICIEA), 2019, pp. 591–595.
- [45] J. B. Elsner and A. A. Tsonis, 'Nonlinear Prediction, Chaos, and Noise', *Bull. Am. Meteorol. Soc.*, vol. 73, no. 1, pp. 49–60, Jan. 1992, doi: 10.1175/1520-0477(1992)073<0049:NPCAN>2.0.CO;2.
- [46] J. W. Taylor and R. Buizza, 'Neural network load forecasting with weather ensemble predictions', *IEEE Trans. Power Syst.*, vol. 17, no. 3, pp. 626–632, Aug. 2002, doi: 10.1109/TPWRS.2002.800906.
- [47] Met Office, 'How weather forecasts are created', *Met Office*, 2019. [https://www.metoffice.gov.uk/weather/learn-about/how-forecasts-are-made/\\_index\\_](https://www.metoffice.gov.uk/weather/learn-about/how-forecasts-are-made/_index_) (accessed Feb. 21, 2019).
- [48] D. J. Stensrud, *Parameterization schemes: keys to understanding numerical weather prediction models*. 2007.
- [49] 'Weather Research and Forecasting Model | MMM: Mesoscale & Microscale Meteorology Laboratory', 2019. <https://www.mmm.ucar.edu/weather-research-and-forecasting-model> (accessed May 14, 2019).
- [50] UCAR, 'WRF Model Updates', 2019. <http://www2.mmm.ucar.edu/wrf/users/graphics/WRF-post-processing.htm> (accessed Feb. 01, 2019).
- [51] Noaa, 'Reading GRIB Files', 2017. [http://www.cpc.ncep.noaa.gov/products/wesley/reading\\_grib.html](http://www.cpc.ncep.noaa.gov/products/wesley/reading_grib.html) (accessed Jan. 23, 2019).
- [52] N. C. for E. P. W. S. S. D. of Commerce, 'NCEP GFS 0.25 Degree Global Forecast Grids Historical Archive'. Jan. 26, 2015, doi: <https://doi.org/10.5065/D65D8PWK>.
- [53] S. S. Baboo and I. K. Shereef, 'An Efficient Weather Forecasting System using Artificial Neural Network', *Int. J. Environ. Sci. Dev.*, pp. 321–326, 2010, doi: 10.7763/IJESD.2010.V1.63.
- [54] A. Routray, U. C. Mohanty, K. K. Osuri, S. C. Kar, and D. Niyogi, 'Impact of Satellite Radiance Data on Simulations of Bay of Bengal Tropical Cyclones Using the WRF-3DVAR Modeling System', *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 4, pp. 2285–2303, Apr. 2016, doi: 10.1109/TGRS.2015.2498971.
- [55] J. L. Elman, 'Finding Structure in Time', *Cogn. Sci.*, vol. 14, no. 2, pp. 179–211, Mar. 1990, doi: 10.1207/s15516709cog1402\_1.

- [56] A. Graves, ‘Supervised Sequence Labelling’, in *Supervised Sequence Labelling with Recurrent Neural Networks*, A. Graves, Ed. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 5–13.
- [57] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, ‘Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling’, *ArXiv14123555 Cs*, Dec. 2014, Accessed: Jul. 04, 2019. [Online]. Available: <http://arxiv.org/abs/1412.3555>.
- [58] S. Hochreiter and J. Schmidhuber, ‘Long Short-Term Memory’, *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997, doi: 10.1162/neco.1997.9.8.1735.
- [59] A. Gulli and S. Pal, *Deep Learning with Keras*. Packt Publishing Ltd, 2017.
- [60] A. Behera, A. Keidel, and B. Debnath, ‘Context-driven Multi-stream LSTM (M-LSTM) for Recognizing Fine-Grained Activity of Drivers’, *Pattern Recognit.*, p. 16, 2018.
- [61] J. Yang, M. N. Nguyen, P. P. San, X. L. Li, and S. Krishnaswamy, ‘Deep Convolutional Neural Networks on Multichannel Time Series for Human Activity Recognition’, in *Twenty-Fourth International Joint Conference on Artificial Intelligence*, Jun. 2015, Accessed: Jul. 04, 2019. [Online]. Available: <https://www.aaai.org/ocs/index.php/IJCAI/IJCAI15/paper/view/10710>.
- [62] S. Bai, J. Z. Kolter, and V. Koltun, ‘An Empirical Evaluation of Generic Convolutional and Recurrent Networks for Sequence Modeling’, *ArXiv180301271 Cs*, Mar. 2018, Accessed: Feb. 27, 2019. [Online]. Available: <http://arxiv.org/abs/1803.01271>.
- [63] F. Yu and V. Koltun, ‘Multi-Scale Context Aggregation by Dilated Convolutions’, *ArXiv151107122 Cs*, Nov. 2015, Accessed: May 03, 2019. [Online]. Available: <http://arxiv.org/abs/1511.07122>.
- [64] D. N. Fente and D. K. Singh, ‘Weather Forecasting Using Artificial Neural Network’, in *2018 Second International Conference on Inventive Communication and Computational Technologies (ICICCT)*, Apr. 2018, pp. 1757–1761, doi: 10.1109/ICICCT.2018.8473167.
- [65] Keras, ‘Home - Keras Documentation’, 2019. <https://keras.io/> (accessed Jan. 28, 2019).
- [66] A. Krizhevsky, I. Sutskever, and G. E. Hinton, ‘ImageNet Classification with Deep Convolutional Neural Networks’, in *Advances in Neural Information Processing Systems 25*, F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, Eds. Curran Associates, Inc., 2012, pp. 1097–1105.
- [67] P. Hewage, A. Behera, M. Trovati, and E. Pereira, ‘Long-Short Term Memory for an Effective Short-Term Weather Forecasting Model Using Surface Weather Data’, in *Artificial Intelligence Applications and Innovations*, 2019, pp. 382–390.
- [68] Federal Meteorological, ‘Surface Weather Observations and Reports’, p. 98, 11 2017.
- [69] J. Brownlee, ‘Using Learning Rate Schedules for Deep Learning Models in Python with Keras’, *Machine Learning Mastery*, Jun. 21, 2016. <https://machinelearningmastery.com/using-learning-rate-schedules-deep-learning-models-python-keras/> (accessed Jan. 28, 2019).
- [70] J. Brownlee, ‘Gentle Introduction to the Adam Optimization Algorithm for Deep Learning’, *Machine Learning Mastery*, Jul. 02, 2017. <https://machinelearningmastery.com/adam-optimization-algorithm-for-deep-learning/> (accessed Jan. 28, 2019).
- [71] S. Ruder, ‘An overview of gradient descent optimization algorithms’, *ArXiv160904747 Cs*, Sep. 2016, Accessed: Jan. 28, 2019. [Online]. Available: <http://arxiv.org/abs/1609.04747>.
- [72] C. Pelletier, G. I. Webb, and F. Petitjean, ‘Temporal Convolutional Neural Network for the Classification of Satellite Image Time Series’, *Remote Sens.*, vol. 11, no. 5, p. 523, Jan. 2019, doi: 10.3390/rs11050523.
- [73] M. N. Wright and A. Ziegler, ‘ranger: A fast implementation of random forests for high dimensional data in C++ and R’, *ArXiv Prepr. ArXiv150804409*, 2015.
- [74] M. Rahman, A. S. Islam, S. Y. M. Nadvi, and R. M. Rahman, ‘Comparative study of ANFIS and ARIMA model for weather forecasting in Dhaka’, presented at the 2013 International Conference on Informatics, Electronics and Vision (ICIEV), 2013, pp. 1–6.
- [75] C.-S. Sun, Y.-N. Wang, and X.-R. Li, ‘A vector autoregression model of hourly wind speed and its applications in hourly wind speed forecasting’, *Proc.-Chin. Soc. Electr. Eng.*, vol. 28, no. 14, p. 112, 2008.
- [76] S. Jaupllari and O. Zoto, ‘An assessment of demand for imports through the VECM model’, *J. Knowl. Manag. Econ. Inf. Technol.*, vol. 3, no. 6, pp. 1–17, 2013.
- [77] V. Cerqueira, L. Torgo, F. Pinto, and C. Soares, ‘Arbitrage of forecasting experts’, *Mach. Learn.*, vol. 108, no. 6, pp. 913–944, Jun. 2019, doi: 10.1007/s10994-018-05774-y.

- [78] M. Sanchez-Fernandez, M. de-Prado-Cumplido, J. Arenas-Garcia, and F. Perez-Cruz, ‘SVM Multiregression for Nonlinear Channel Estimation in Multiple-Input Multiple-Output Systems’, *IEEE Trans. Signal Process.*, vol. 52, no. 8, pp. 2298–2307, Aug. 2004, doi: 10.1109/TSP.2004.831028.
- [79] C. M. Bishop, ‘Pattern Recognition and Machine Learning’, *Springer*, 2006.
- [80] M. McCREA *et al.*, ‘Standard regression-based methods for measuring recovery after sport-related concussion’, *J. Int. Neuropsychol. Soc.*, vol. 11, no. 1, pp. 58–69, Jan. 2005, doi: 10.1017/S1355617705050083.
- [81] S. Kavitha, S. Varuna, and R. Ramya, ‘A comparative analysis on linear regression and support vector regression’, in *2016 Online International Conference on Green Engineering and Technologies (IC-GET)*, Nov. 2016, pp. 1–5, doi: 10.1109/GET.2016.7916627.
- [82] D. Basak, S. Pal, and D. Patranabis, ‘Support vector regression’, *Neural Inf. Process.-Lett. Rev.*, vol. 11, no. 10, pp. 203–224, 2007.
- [83] K. A. Althelaya, E. M. El-Alfy, and S. Mohammed, ‘Evaluation of bidirectional LSTM for short-and long-term stock market prediction’, in *2018 9th International Conference on Information and Communication Systems (ICICS)*, Apr. 2018, pp. 151–156, doi: 10.1109/IACS.2018.8355458.
- [84] H. Salehinejad, S. Sankar, J. Barfett, E. Colak, and S. Valaee, ‘Recent Advances in Recurrent Neural Networks’, *ArXiv180101078 Cs*, Dec. 2017, Accessed: Jul. 04, 2019. [Online]. Available: <http://arxiv.org/abs/1801.01078>.
- [85] Z. Liu, S. Liu, F. Hu, J. Li, Y. Ma, and H. Liu, ‘A comparison study of the simulation accuracy between WRF and MM5 in simulating local atmospheric circulations over Greater Beijing’, *Sci. China Earth Sci.*, vol. 55, no. 3, pp. 418–427, Mar. 2012, doi: 10.1007/s11430-011-4310-2.
- [86] M. A. Hernández-Ceballos, C. A. Skjøth, H. García-Mozo, J. P. Bolívar, and C. Galán, ‘Improvement in the accuracy of back trajectories using WRF to identify pollen sources in southern Iberian Peninsula’, *Int. J. Biometeorol.*, vol. 58, no. 10, pp. 2031–2043, Dec. 2014, doi: 10.1007/s00484-014-0804-x.
- [87] National Centers for Environmental Information, ‘Global Forecast System (GFS)’, 2019. <https://www.ncdc.noaa.gov/data-access/model-data/model-datasets/global-forecast-system-gfs> (accessed Jul. 19, 2019).
- [88] P. Hewage *et al.*, ‘Temporal convolutional neural (TCN) network for an effective weather forecasting using time-series data from the local weather station’, *Soft Comput.*, Apr. 2020, doi: 10.1007/s00500-020-04954-0.