Accepted for presentation in 2021 European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (**ECML-PKDD**), September 2021

Precise Weather Parameter Predictions for Target Regions via Neural Networks

Yihe Zhang[1], Xu Yuan[1], Sytske K. Kimball[2], Eric Rappin[3], Li Chen[1], Paul Darby[1], Tom Johnsten[2], Lu Peng[4], Boisy Pitre[1], David Bourrie[2], and Nian-Feng Tzeng[1]

[1]University of Louisiana at Lafayette
[2]University of South Alabama
[3]Western Kentucky University
[4]Louisiana State University

Abstract. Flexible fine-grained weather forecasting is a problem of national importance due to its stark impacts on economic development and human livelihoods. It remains challenging for such forecasting, given the limitation of currently employed statistical models, that usually involve the complex simulation governed by atmosphere physical equations. To address such a challenge, we develop a deep learning-based prediction model, called Micro-Macro, aiming to precisely forecast weather conditions in the fine temporal resolution (i.e., multiple consecutive short time horizons) based on both the atmospheric numerical output of WRF-HRRR (the weather research and forecasting model with high-resolution rapid refresh) and the ground observation of Mesonet stations. It includes: 1) an Encoder which leverages a set of LSTM units to process the past measurements sequentially in the temporal domain, arriving at a final dense vector that can capture the sequential temporal patterns; 2) a Periodical Mapper which is designed to extract the periodical patterns from past measurements; and 3) a Decoder which employs multiple LSTM units sequentially to forecast a set of weather parameters in the next few short time horizons. Our solution permits temporal scaling in weather parameter predictions flexibly, yielding precise weather forecasting in desirable temporal resolutions. It resorts to a number of Micro-Macro model instances, called modelets, one for each weather parameter per Mesonet station site, to collectively predict a target region precisely. Extensive experiments are conducted to forecast five important weather parameters at two Mesonet station sites. The results exhibit that our Micro-Macro model can achieve high prediction accuracy, outperforming almost all compared counterparts on five parameters of interest.

1 Introduction

Weather forecasting in the temporal domain is a critical problem of national importance, closely tied to the economic development and human livelihoods. However, accurate forecasting remains open and quite challenging, especially in the context of precise and fine-grained prediction over multiple temporal resolutions. Such a short-term and fine temporal resolution prediction relates tightly to

agriculture, transportation, water resource management, human health, emergency responses, and urban planning, essential for taking such timely actions as generating society-level emergency alerts on convection initiation, producing real-time weather guidance for highways and airports, among others.

To date, the most prominent and widely used national forecasting model is called Weather Research and Forecasting (WRF) with HRRR (High Resolution Rapid Refresh) [3]. It provides prediction for the weather parameters that cover the United States continent. However, it is an hourly prediction model, which can only coarsely forecast the weather parameters in the resolution of one hour, failing to capture finer time granularity needs (say, the 5- or 10-minute time horizon) in weather forecasting. This is largely due to the high computation requirement and voluminous data outputs associated with this model, involving complex simulation of physical governing atmospheric flows [16]. Its prediction accuracy is far from satisfaction, as a result of the employed statistical models, whose capability of extracting fine-grained weather patterns is limited. Meanwhile, more than three dozen of regional Mesonet networks exist under the U.S. National Mesonet Program, with each network involving tens or hundreds of observational stations for gathering near-surface weather measurements periodically. Mesonet Stations provide site-specific real datasets in finer temporal granularity (typically in minutes). For example, our experimental evaluation makes use of datasets gathered by the SA Mesonet, which covers South Alabama by 25 observational stations to gather data once in every minute [2].

Recent advances in machine learning technologies have promoted weather forecasting into a new era. Many studies have attempted to leverage the neural network-centric techniques in weather forecasting, producing promising results. These techniques include, but are not limited to, the deep neural network (DNN), convolutional neural network (CNN), long short-term memory network (LSTM), generative adversarial network (GAN), and autoEncoder, for predicting such weather parameters as precipitation [14, 18, 27], wind direction and speed [4, 10, 15, 19], solar radiation [5, 12], air quality [28], weather changes [13, 26], and many others. However, known parameter forecasting models developed so far cannot yield accurate enough predictions in fine-grained temporal resolution over flexible time horizons.

This paper aims to develop a new forecasting model, termed Micro-Macro, for effective and precise prediction on weather parameters in the fine-gained temporal resolution, by taking both micro inputs from Mesonet Stations [1] and macro inputs from Weather Research and Forecasting (WRF) with HRRR (High Resolution Rapid Refresh) [3] computation outputs, for the first time. We leverage the prominent deep learning technologies that take the existing massive atmospheric data sets (resulting from WRF-HRRR numerical prediction) and surface observation data (gathered via existing Mesonet networks) as the input to produce fine-grained weather forecasting in the temporal domain for target regions of interest. Specifically, the developed model includes three components: 1) an Encoder which processes the time sequence data to capture the temporal domain variation of weather conditions, 2) a Periodical Mapper which extracts

the periodical pattern of the time sequence data, and 3) a Decoder which predicts a sequence of values corresponding to different time points. Specifically, each LSTM unit in the Encoder can learn the key features from inputs and then outputs its hidden state to the next LSTM unit, which can continue to learn the key features from both the previous input and the current input, in terms of time sequence characteristics. This results in a dense vector, including rich information for the weather condition's variation in the temporal domain out of the atmospheric output and surface observation. Meanwhile, a Periodical Mapper can capture the periodical pattern of the data and generate a dense vector for enhancing the learning of temporal data patterns. Both dense vectors are used by the Decoder to forecast the weather parameters in the next few continued time horizons. This model incorporates the near surface observation and the atmospheric numerical output, which are complementary with each other to let our model better use relevant past measurements for forecasting, significantly improving prediction accuracy.

We conduct experiments to predict a set of weather parameters. Our experimental results show that the developed Micro-Macro model instances, dubbed modelets, outperform almost all the compared solutions in forecasting temperature, humidity, pressure, wind direction and wind speed.

2 Related Work

Abundant applications of machine learning techniques to weather forecasting exist. This section reviews the recent advances in such applications, which mostly follow two lines of work.

The first line aims to explore whether the neural network is capable of simulating the physical principles of atmosphere systems. In particular, Dueben *et al.* [11] employed two neural networks, i.e., Global NN and Local NN, to simulate the dynamics of a simple global atmosphere model at 500 hPa geopotential. The results concluded that prediction outcomes by the neural network models can be better than those of the coarse-resolution atmosphere models for a short duration under the 1-hour time scale. Scher [21] applied the CNN structure with autoEncoder setup to learn the simplified general circulation models (GCMs), which can predict the weather parameters up to 14 days. Weyn *et al.* [25] leveraged the CNN with LSTM structure to achieve a 14-day lead time forecasting as well. Vlachas *et al.* [22] employed the LSTM model to reduce the order space of a chaotic system. However, known proposed solutions along this line all just focused on developing prediction models for simulated or simplified climate environments, without taking into account the real-world conditions, which tend to be rather complex. Their applicability and effectiveness on real environments are still questionable, given their complex conditions in practice. For example, the actual measurements from Mesonet stations are highly dependent on local conditions. In addition, their solutions cannot be applied to fine-grained predictions with flexible time horizons in the desirable temporal resolution.

The other line of work aims to leverage the neural network to develop new models for the real-world weather parameters prediction. For example, [19] leveraged the LSTM and fully connected neural networks to predict the wind speed at an offshore site, by capturing its rapidly changing features. Grover *et al.* [13] combined the discriminatively trained predictive models with a deep neural network to predict the atmospheric pressure, temperature, dew point, and winds. [27] proposed a convolutional LSTM model to predict precipitation. Pan *et al.* [18] employed the CNN with delicately selected stacked frames for precipitation forecasting. [14] proposed a model with the autoEncoder structure to predict rainfalls. [4] forecasted the hurricane trajectories via an RNN structure. [12] and [5] employed the LSTM structures to predict the solar radiation and photovoltaic energy, respectively. [28] proposed a deep fusion network to predict air quality. [26] developed a deep-CNN model on a cubed sphere for predicting several basic atmospheric variables on a global grid. However, all aforementioned work still cannot predict weather parameters accurately in fine granularity over flexible time horizons, for a desirable temporal resolution. Hence, accurate weather prediction and fine-grained temporal resolution across flexible time horizons remains an open and challenging problem.

3 Pertinent Background

In this section, we describe Mesonet near surface observation and WRF-HRRR (Weather Research and Forecasting High Resolution Rapid Refresh model) prediction model to illustrate their limitations in precise weather forecasting.

Mesonet [1] is a national supported program that comprises a set of automated weather stations located at some specific areas in the USA. Its towers aim to gather meteorological- and soil- measurements relevant to local weather phenomena. Each station monitors tens of atmospheric measurements, including temperature, rainfall, wind speed, and others, once per minute for every day since its establishment.

WRF with HRRR prediction: The WRF model takes actual atmospheric conditions (i.e., from observations and analyses) as its input to produce outputs that serve a wide range of meteorological applications across national scales. WRF with HRRR weather forecast modeling system is nested in the Rapid Refresh model for predicting weather parameters that cover the United States continent with a resolution of 3 km for a total of 1059×1799 geo-grids. The prediction outputs are produced hourly, over the next consecutive 18 hours. In each geo-grid, there are up to 148 parameters, representing the temperature, pressure, among many others, to signify the predicted weather condition. A 1059×1799 matrix is employed to keep each parameter's outputs, with each entry mapping to one geolocation of the United States map.

However, both Mesonet and WRF-HRRR have their respective limitations. For Mesonet, the involved stations are only for gathering the current near-surface measurements, unable to predict future values. For WRF-HRRR, its prediction accuracy is far from satisfaction, besides its hourly scale prediction to limit its

suitability for meteorological applications that requires high temporal resolutions (say, 5 *min*, 15 *min*, or 30 *min*).

4 Learning-based Modelets for Weather Forecasting

This paper aims to develop learning-based meteorology (abbreviated as Meteo) modelets, for correctly and concurrently predicting multiple weather parameters in a flexible and fine-temporal resolution, based on the inputs of both minute-level near-surface observations from Mesonet and WRF hourly atmospheric numerical outputs, referring respectively as the Micro and the Macro datasets. We take the Micro dataset as the main input and screen a set of relevant parameters in Macro dataset for incorporation to predict target weather parameters correctly. Our goal is to extract the temporal variation features from the previous measurements to precisely predict the weather condition in the next few time horizons (e.g., next T min, $2T$ mins, etc.). It is challenging as the two data sources have different scales in the temporal domains. To address such a challenge, the prominent machine learning technology is leveraged to learn the temporal sequence patterns from both datasets that can capture variation of weather conditions to predict specific parameters. A new Meteo modelet, named Micro-Macro, is developed to permit temporal downscaling and upscaling in weather parameter predictions flexibly, arriving at precise weather forecasting in desirable temporal resolutions. We will first outline a Micro model by just relying on the Micro dataset as the input for prediction. Then, we describe our Micro-Macro model which takes both Micro and Macro datasets as the input for precisely forecasting weather parameters via separate modelets (i.e., model instances) in the temporal domain.

4.1 Micro Model

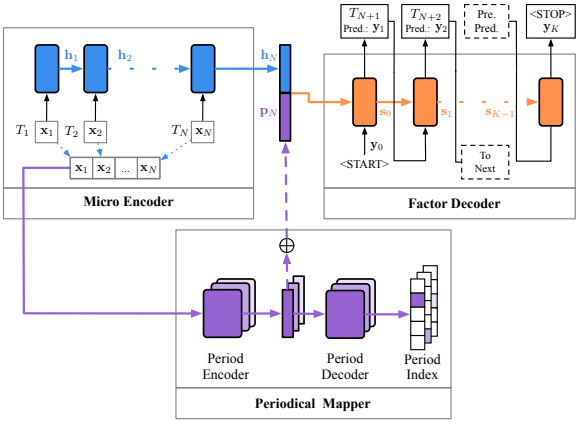


Fig. 1: Structure of Micro model.

Most atmospheric data has the noticeable temporal sequence patterns and periodical patterns, whereas weather conditions (i.e., parameters) change continuously with time. To capture such patterns for forecasting in continuous T-minute horizons, we leverage a structure with an Encoder, a Decoder, and a Periodical Mapper, with the first two both include the (LSTM) networks and the last one is in the neural network structure, to capture the time sequence patterns and periodical patterns, respectively. The structure is shown in Fig. 1. Notably, although the encoder-decoder LSTM model has been widely applied to sequence tasks, e.g., language translation [9] and question answering [7], the physical meaning in each entry for the input vectors is not well explored. This results in the loss of affluent element-wise features, only to encode all features into a dense vector, which cannot work effectively here. The customized design is desired under our application context. The details of three components are illustrated as follows.

Micro Encoder. It comprises one LSTM network, to encode the temporal sequence data in a certain period into one single dense vector, representing the temporal feature variation. To forecast weather condition in next continuous T-*min* horizons, we consider the past $N \times T$ minutes surface observation from Mesonet as a sequence of data frames, with each one including T-*min* observed weather condition to serve as the input. Here, N represents the number of selected T-*min* intervals. The LSTM unit will learn the key features and update its corresponding hidden state vector (denoted as $\mathbf{h}_{t-1}$). Such a vector together with the next data frame is input to the next LSTM unit to produce a new hidden state vector $\mathbf{h}_t$, which can be logically modeled as follows:

$$\mathbf{h}_t = LSTM_h(\mathbf{h}_{t-1}, \mathbf{x}_t) \ , \tag{1}$$

where $LSTM_h$ represents a series of steps to generate the next hidden states and $\mathbf{x}_t$ denotes the data frame in time slot t. In the end, a dense vector $\mathbf{h}_N$ is generated, including the aggregated temporal patterns variation from N inputs.

Periodical Mapper. This design is used to process the input data sequence $\mathbf{x} = \{\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_t, ..., \mathbf{x}_N\}$ for extracting the periodical patterns, comprising two core components: Period Encoder and Period Decoder. Each weather parameter i has a Period Encoder, with its dense vector $\mathbf{p}_{(i)}$. In the end, the sequence data $\mathbf{x}$ is encoded into a dense vector $\mathbf{p}_N$, by summarizing the dense vector from all M weather parameters, yielding:

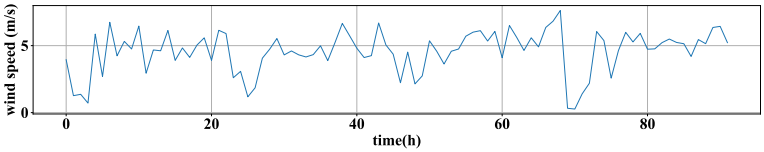
$$\mathbf{p}_{(i)} = P_{e,i}(\bar{\mathbf{x}}_{(i)}), \ \mathbf{p}_N = \sum_{i=1}^{M_i} \mathbf{p}_{(i)} \ . \tag{2}$$

where $\bar{\mathbf{x}}_{(i)}$ is a vector with entries from the i-th weather parameter value of $\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_t, ...,$ and $\mathbf{x}_N$, $P_{e,j}(\cdot)$ represents a Period Encoder corresponding to the i-th weather parameter, which is a neural network structure.

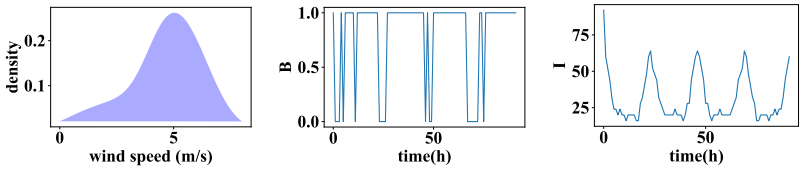
The Period Decoder decodes each dense vector $\mathbf{p}_{(i)}$ to a periodical index vector $\mathbf{p}_{\mathbf{o}(i)}$, expressed as

$$\mathbf{p}_{\mathbf{o}(i)} = P_{d,i}(\mathbf{p}_{(i)}) \ , \tag{3}$$

where $P_{d,i}$ is also a neural network structure. If the input temporal sequence $\bar{\mathbf{x}}_{(i)}$ matches a periodical pattern, the corresponding entry will be 1 and all other entries will be 0.



(a) Wind speed example in 96 hours



(b) Wind speed density (c) Binary pattern (d) Periodical pattern

Fig. 2: Example on periodical pattern discovery.

In the training phase, we derive the periodical index vector from the historical weather records. We run a toy example to explain this step. For example, Fig. 2(a) shows the wind speed within 96 hours, taken from the Mesonet observation dataset. We first need to find a reference point, which shall help discover the periodical pattern of the weather records. Since the data distribution is unknown, we leverage Kernel Density Estimation [8] to find the density of observation values, with the density likelihood to yield:

$$\hat{f}_h(X_i) = \frac{1}{nh} \sum_{j=1}^{n} \Phi(\frac{X_i - X_{ij}}{h}) \ , \tag{4}$$

where $\hat{f}_h(X_i)$ is the density function of measurement X_i. X_{ij} is the j-th observed value of X_i corresponding to a weather parameter. n is the total number of data points and h is an empirical parameter which is set to 0.85 in our experimental evaluation. Φ denotes the normal distribution. By maximizing Eqn. (4), we get the density distribution as shown in Fig. 2(b) and pick up the largest density point of $\hat{f}_h(X_i)$ as the reference point, i.e., 5.03. We then consider the area that covers top-15% density values as the reference area R_i. A binary sequence B_i of measurement X_i is then calculated. That is, if the observed value $X_{ij} \in R_i, 0 < j \leq n$, we set $B_{ij} = 1$, otherwise $B_{ij} = 0$, as shown in Fig. 2(c). Afterwards, we conduct the Discrete Fourier Transform (DFT) [24] on the sequence B_i to transform them to n complex numbers, denoted as $D_i : [D_{i1}, D_{i2}, .., .D_{ij}, ..., D_{in}]$. Then we calculate the periodogram $F_{ij} = \|D_{ij}\|^2$ for each complex number to get F_i. By taking Inverse Discrete Fourier Transform (IDFT) [17] on F_i, we derive the Periodic Correlation I_i [6], as shown in Fig. 2(d).

In the curve of I_i, we identify all peak values. Each interval between two neighboring peak values is denoted as one period. We equally divide each time

period into $P = 24 * 60/T$ time slots and label the time slots from 1 to $24 * 60/T$ sequentially as the periodical indices. We then label the input sequence $\mathbf{x}$ with an index, according to $\mathbf{x}$'s timestamp on I_i. When training, we use Mean Square Loss as Period Decoder's loss.

Factor Decoder. The Factor Decoder is to predict a set of particular weather parameters in the next few time horizons. It includes a set of LSTM units, to predict the weather parameter at consecutive time intervals, denoted as $T_{N+1}, T_{N+2}, \cdots$, following the previous $N \times T$ minutes. The first LSTM unit takes the dense vector $\mathbf{h}_N$ and $\mathbf{p}_N$ as its input for predicting the vector of weather parameter $\mathbf{y}_1$ in the next interval T_{N+1} as follows,

$$\mathbf{y}_1 = LSTM_o(\mathbf{y}_0, \langle \mathbf{h}_N, \mathbf{p}_N \rangle) \,, \tag{5}$$

where $LSTM_o$ denotes a series of steps to calculate outputs and $\mathbf{y}_0$ is an empty output vector, whereas $\langle \mathbf{h}_N, \mathbf{p}_N \rangle$ denotes concatenation of $\mathbf{h}_N$ and $\mathbf{p}_N$. For the prediction in each of the remaining time intervals, we take both the hidden state vector $\mathbf{s}_k$ and the previous predicted vector $\mathbf{y}_k$ as inputs to update the current LSTM state. The new hidden state $\mathbf{s}_{k+1}$ can be logically expressed as: $\mathbf{s}_{k+1} = LSTM_s(\mathbf{s}_k, \mathbf{y}_k)$. Note that, we retake the pervious output as new input to update the new hidden state. The next output is given by $y_{k+1} = LSTM_o(\mathbf{y}_k, \mathbf{s}_{k+1})$.

In the training phase, each $(N \times T)$-minute data will be used as inputs and the data from subsequent M time intervals will be used for labeling. Here, M represents the number of time horizons that we aim to predict. For example, to predict a weather parameter, say temperature, we consider a set of relevant parameters in $N \times T$ minutes as the features and label the temperature values in the following time intervals of $T_{N+1}, T_{N+2}, \cdots, T_{N+M}$. As the surface observation data are generated once in every minute, we average the values of each parameter over T minutes as the features. Similarly, for labeling, we take the averaged temperature value within each T minutes. The N data frames (corresponding to the $(N \times T)$-minute past measurements) and the labeled temperature values (in M subsequent intervals) are inputted to Micro Encoder. At the Decoder, we start from the first LSTM unit and predict a set of weather parameters at the time interval of T_{N+1}. Both the hidden state from this LSTM network and the predicted value of T_{N+1} are then input to the second LSTM for predicting T_{N+2}. This step continues until all values for the next M time horizons are predicted.

4.2 Micro-Macro Model

As the number of observed parameters at Mesonet is limited, it is insufficient for forecasting just based on the Micro datset. Hence, we incorporate the Macro dataset as a complementary input to the model for better forecasting. Given the Macro dataset is hourly generated and surface observation is updated in each minute, how to integrate such two data sources is still a challenging problem, as it requires downscaling the atmospheric output.

The structure of Micro-Macro model is shown in Figure 3, which is similar to that of the Micro model, with a difference in the input that includes an additional Macro Encoder. In Macro Encoder, we divide each hour into $60/T$

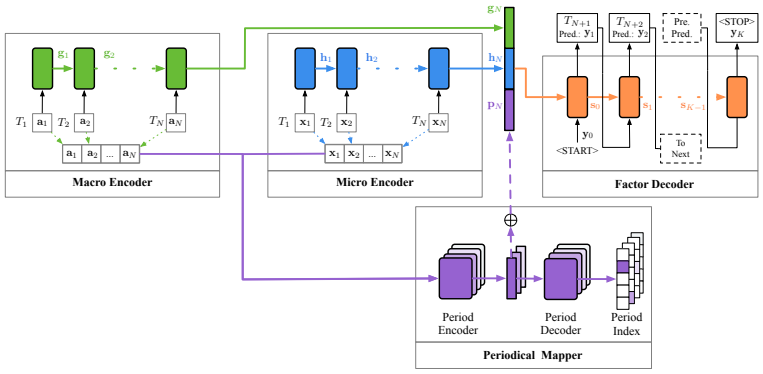


Fig. 3: The structure of Micro-Macro model.

time frames and use this hourly output from Macro dataset to represent the first time frame's value. The values of all remaining time frames are indicated as "Empty". All hourly datasets are processed in the same way. When inputting to the Encoder, if a frame has an empty value, the corresponding LSTM unit in the Macro Encoder takes only the hidden state vector from the previous unit as the input to self-update its hidden state vector; otherwise, it executes in the same way as in Micro Encoder. The Macro Encoder outputs a dense vector, denoted by $\mathbf{g}_N$, as depicted in Figure 3. To extract the time sequence features from both Micro and Macro datasets, we concatenate the dense vectors ($\mathbf{h}_N$, $\mathbf{g}_N$, and $\mathbf{p}_N$) from the Micro Encoder, Macro Encoder and Periodic Selector, i.e., $\mathbf{h} = \langle \mathbf{h}_N, \mathbf{g}_N, \mathbf{p}_N \rangle$. The decoder in the Micro-Macro model is similar to that in the Micro model. It takes the concatenated dense vector $\mathbf{h}$ as its input to perform forecasting for subsequent time horizons sequentially. Notably, in both training and prediction phases, the Micro-Macro model takes the data of the same geo-grid from Micro and Macro datasets at an identical time interval.

5 Experiment

5.1 Setting

Datasets. We take the near surface observation from SA Mesonet [2] and the WRF-HRRR [3] atmospheric numerical output as our experimental datasets, which are called as Micro and Macro datasets, respectively. The Micro dataset includes 26 automated weather stations for monitoring the real-time meteorological phenomena. The monitored weather conditions include temperature, rainfall, wind speed and direction, soil temperature and humidity, once in every minute. SA Mesonet stations Elberta and Atmore are selected for our experiments, with the former located closer to the Gulf Shore and the latter one away from the shore. In total, ten Micro-Macro model instances (called modelets) are involved, one for a weather parameter at each station site. We take the ground observation from years 2017 and 2018 as the training dataset, while taking the observation from 2019 as the test dataset. Macro dataset is the predicted output

Table 1: Parameter information

Parameter	Measurement	Mounting Height	Measuring Range
TEMP	Air Temperature	2 m	-40 to 60°C
HUMI	Relative Humidity	2 m	0 to 100%
PRES	Atmospheric Pressure	1.5m	600 to 1060mb
WSPD	Wind Speed	2 m	0 to 100 m/s
WDIR	Wind Direction	2 m	0 to 360°

from WRF-HRRR model. The numerical output in the years 2017, 2018, and 2019, corresponding to the stations of Atmore and Elberta, are taken to conduct our experiments. To forecast temperature, humidity, pressure, wind speed, and wind direction (see details in Table 1), we select their respective most relevant parameters from Micro dataset and ten most important parameters from the Macro dataset. Table 2 lists the most relevant parameters selected from Micro dataset for training the weather measurements of temperature, humidity, pressure, wind speed, and wind direction, respectively. Table 3 lists 10 most important parameters that are selected from Macro dataset.

Table 2: Relevant parameters from Micro dataset

Predictions	Measurement parameters
TEMP	Vitel_100cm_d, IRTS_Body, SoilCond, SoilWaCond_tc, Vitel_100cm_b, eR, wfv, Vitel_100cm_a, SoilCond_tc, RH_10m
HUMI	Temp_C, Vitel_100cm_d, Vitel_100cm_a, Vitel_100cm_b, WndSpd_Vert_Min, SoilT_5cm,Pressure_1, AirT_2m, AirT_10m, PTemp, IRTS
PRES	RH_10m, SoilCond, Temp_C, Vitel_100cm_d, AirT_1pt5m, IRTS_Trgt, PTemp, Vitel_100cm_b, SoilSfcT, AirT_10m
WSPD	WndSpd_2m_WVc_1, WndSpd_10m, WndSpd_2m_Max, WndSpd_Vert_Tot, WndSpd_2m_Std, QuantRadn, WndSpd_2m_WVc_2, WndSpd_Vert, WndSpd_10m_Max, WndDir_2m
WDIR	WndSpd_2m_WVc_3, WndSpd_10m_Max, WndSpd_2m_Max, WndSpd_2m_WVc_2, WndSpd_2m_Std, AirT_1pt5m, WndSpd_2m_WVc_4, AirT_10m, Pressure_2

Compared solutions. We compare our results with the following ones: *1) Observation:* We take the ground observation monitored in 2019 from Mesonet at stations Atmore and Elberta, respectively, to inspect our results; *2) WRF-HRRR:* The predicted atmospheric numerical output in 2019 from WRF-HRRR model; *3) SVR [20]:* A regression model based on support vector machine; *4) SNN-Micro [11]:* A neural network model which takes the Micro dataset for training; *5) SNN-both [11]:* A neural network model that takes the aligned data from both Micro and Macro datasets for training; *6)* DUQ_{512} *[23]:* A deep uncertainty quantification model which has one GRU layer with 512 hidden nodes; and *7)* $DUQ_{512-512}$ *[23]:* A deep uncertainty quantification model which has two GRU layers with 512 hidden nodes in each layer.

Table 3: Relevant parameters from Macro dataset

Feature ID	Description
9	250hpa U-component of wind (m/s)
10	250hpa V-component of wind (m/s)
55	80 meters U-component of wind (m/s)
56	80 meters V-component of wind (m/s)
61	Ground moisture (%)
71	10 meters U-component of wind (m/s)
72	10 meters V-component of wind (m/s)
102	Cloud base pressure (Pa)
105	Cloud top pressure (Pa)
116	1000m storm relative helicity (%)

Experiment setup. We take data from the first season in 2017 and 2018 for training, and predict the weather conditions in the same season in 2019. We target five weather parameters of interest, i.e., temperature, humidity, pressure, wind speed, and wind direction, for prediction. The time is divided with a sequence of $T = 5$-min intervals. We take each set of 60 minutes' (i.e., $N = 12$) data as the features, and label the weather parameter values in the following 30 minutes, with each 5 minutes as one time interval and the averaged value as the label. For prediction, we also take past 60 minutes' measurements as the input to forecast the next 6 continuous time intervals' values. As SNN-Micro and SNN-both cannot conduct the sequence of prediction, we only let it predict the next time interval immediately after every 60 minutes' measurement. Their parameters are listed in Table 4.

Table 4: Parameters of SNN networks

	SNN-Micro	SNN-Both
Input size	10	20
Hidden 1	200 neurons	200 neurons
Hidden 2	100 neurons	100 neurons
Hidden 3	30 neurons	30 neurons
Output	1 neuron	1 neuron

Each LSTM in the Micro model includes 256 hidden states, whereas every Encoder and Decoder of the Micro-Macro model has 256 and 512 hidden states, respectively. Root Mean Squared Error (RMSE) is employed to gauge the prediction error: $\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2}$, where $\hat{\mathbf{Y}}$ and $\mathbf{Y}$ denote the vectors of predicted and observed values, respectively. n is the number of data values.

5.2 Overall Performance

We conduct multiple experiments to forecast the values of various weather parameters of interest at different time points in the first season of 2019. Table 5 shows the averaged RMSE of our Micro-Macro model for forecasting the next 30-minute weather conditions on temperature (TEMP), humidity (HUMI), pres-

Table 5: RMSE values of our modelets at Atmore and Elberta stations

		0 to 5 min	5 to 10 min	10 to 15 min	15 to 20 min	20 to 25 min	25 to 30 min
Atmore	TEMP	0.502	0.531	0.564	0.601	0.632	0.670
	HUMI	4.431	4.507	4.552	4.707	5.122	5.802
	PRES	1.087	1.133	1.139	1.156	1.184	1.235
	WSPD	0.396	0.552	0.572	0.658	0.709	0.833
	WDIR	4.426	4.534	4.605	4.682	4.873	5.037
Elberta	TEMP	0.424	0.468	0.471	0.475	0.479	0.485
	HUMI	1.852	1.873	1.893	1.905	1.933	2.015
	PRES	1.075	1.213	1.245	1.309	1.452	1.607
	WSPD	0.492	0.528	0.556	0.584	0.614	0.656
	WDIR	5.124	5.251	5.531	5.670	5.834	5.902

sure (PRES), wind speed (WSPD), and wind direction (WDIR) at the two representative SA Mesonet stations of Atmore and Elberta, when comparing to ground observations. Clearly, our modelets achieve very small RMSE values for predicting temperature, pressure, and wind speed. Although the RMSE values appear relatively larger for wind direction prediction at both stations, the fact that wind direction is amenable to sudden changes frequently and that its wide measurement range (of 0 to 360 degrees, as listed in Table 1) renders the larger prediction discrepancies unavoidable. Such marked prediction discrepancies also bring down the prediction accuracy of humility (which is sensitive to wind speed and direction) accordingly, in particular for the inland station (of Atmore).

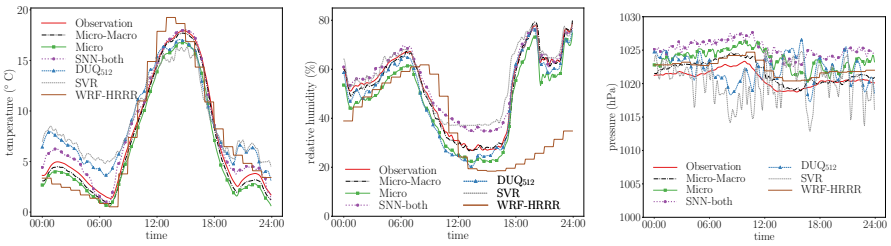
5.3 Comparing to Other Methods

Table 6: RMSE values of different methods for 5-minute prediction

	Atmore					Elberta				
	TEMP	HUMI	PRES	WSPD	WDIR	TEMP	HUMI	PRES	WSPD	WDIR
WRF-HRRR	2.412	20.471	1.648	1.112	24.765	1.633	14.296	1.554	1.412	28.305
SVR	3.581	20.507	5.209	1.306	23.871	1.734	22.953	6.752	1.887	23.050
SNN-Micro	0.668	9.137	5.373	0.354	4.782	1.381	4.387	4.927	0.265	**4.872**
SNN-both	0.619	7.611	4.959	**0.330**	**4.135**	0.804	4.250	4.337	**0.264**	5.027
DUQ_{512}	0.812	5.668	2.714	0.592	5.252	0.645	3.524	3.513	0.541	5.721
$DUQ_{512-512}$	0.657	5.354	2.667	0.585	5.303	0.632	3.326	3.225	0.489	5.353
Micro-Macro	**0.502**	**4.431**	**1.087**	0.396	4.426	**0.424**	**1.852**	**1.075**	0.492	5.124

We next compare the results from our Micro-Macro model to those from other methods on forecasting temperature, humidity, pressure, wind speed, and wind direction. Table 6 shows the prediction results of RMSE (in comparsion to ground observation) obtained from different methods for 5-minute prediction. We can see our model outperforms all other models, with RMSE values of only 0.502, 4.431, 1.087 at Atmore, and with RMSE values of only 0.424, 1.852, 1.075 at Elberta, on the forecasting of temperature, humidity and pressure, respectively. On predicting wind speed and direction, Micro-Macro model beats WRF-HRRR, SVR, DUQ_{512}, and $DUQ_{512-512}$. SNN-Micro and SNN-both have similar prediction performance as our Micro-Macro model on predicting the two

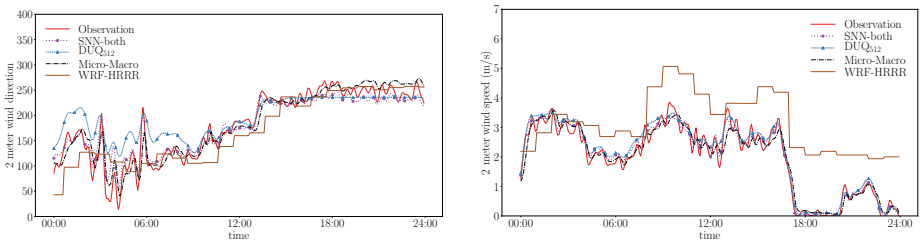
parameters, but notably, they cannot conduct a sequence prediction for subsequent multiple time intervals. SVR performs the worst on predicting all parameters at both stations. WRF-HRRR also performs poorly on all parameters but pressure, which has better accuracy than all other models except for our Micro-Macro model. This demonstrates the necessity and importance of developing new meteorological modelets for nationwide use in lieu of WRF-HRRR.



(a) temperature (b) humidity (c) pressure

Fig. 4: Prediction of temperature, humidity, and pressure at Elberta station.

For prediction result illustration, we randomly select one day in the first season of 2019 for forecasting its weather conditions, starting from 00:00am to 11:59pm. Figures 4(a), (b), and (c) exhibit the comparative results from our modelets versus those from the ground observation, WRF-HRRR output, Micro, SNN-both, DUQ_{512} and SVR, respectively for forecasting temperature, humidity, and pressure at Elberta station. We observe the curves of our modelets are most close to those from ground observation. This demonstrates that our modelets can continuously provide the best prediction results for the examined duration (of 24 hours), in comparison to other methods. Figures 5(a) and (b) show the results of forecasting wind direction and speed by Micro-Macro model, SNN-both, DUQ_{512}, and WRF-HRRR output for the same day. Micro-Macro, SNN-both, and DUQ_{512} models exhibit similar forecasting performance, being far better than the WRF-HRRR output.



(a) Wind direction (b) Wind speed

Fig. 5: Prediction of wind direction and wind speed at Elberta station.

5.4 Ablation Study

The ablation study is next conducted to signify the necessity and importance of the *Periodical Mapper* component in our design. We denote Micro$^-$ and Micro-Macro$^-$ as the models precluding the *Periodical Mapper* from the Micro model and Micro-Macro model, respectively, for comparison. The RMSEs of different variants for 5-minute prediction are listed in Table 7.

From this table, we observe that both Micro and Micro-Macro models significantly outperform their respective variants (i.e., Micro$^-$ and Micro-Macro$^-$ respectively) on predicting all five weather parameters at both stations, except that the Micro-Macro model is slightly inferior to the Micro-Macro$^-$ model on predicting wind speed at Elberta station. These results demonstrate that the inclusion of *Periodical Mapper* is important to help elevate the overall prediction performance. In addition, we also observe that our Micro-Macro model greatly outperforms the Micro model, demonstrating the necessity of incorporating both ground observation and the atmospheric numerical output for precise prediction.

Table 7: Results of ablation study

	Atmore					Elberta				
	TEMP	HUMI	PRES	WSPD	WDIR	TEMP	HUMI	PRES	WSPD	WDIR
Micro$^-$	0.620	7.892	2.845	0.603	5.220	1.289	6.034	3.022	0.682	7.711
Micro	0.583	7.279	2.653	0.515	5.122	1.064	5.756	2.985	0.467	7.682
Micro-Macro$^-$	0.526	4.494	1.114	0.497	4.970	0.467	1.860	1.088	**0.447**	5.767
Micro-Macro	**0.502**	**4.431**	**1.087**	**0.396**	**4.426**	**0.424**	**1.852**	**1.075**	0.492	**5.124**

5.5 Abnormal Weather Forecasting

We next validate the ability of our proposed Micro-Macro model for forecasting abnormal weather conditions. Four abnormal weather conditions are considered, i.e., chill, torridity, storm and rainstorm, which are assumed to associate with the lowest temperature, highest temperature, highest wind speed, and highest precipitation, respectively. We take the set of 5-minute intervals in the first season of 2019 that have the lowest 5% temperature, highest 5% temperature, highest 5% wind speed and highest 5% precipitation from the Mesonet ground measurements. Our experiment is conducted to predict each respective weather parameter in 5-minute intervals, with the one hour input.

Table 8: RMSE for abnormal weather prediction

	chill	torridity	storm	rainstorm
WRF-HRRR	3.098	1.534	5.269	1.694
SVR	3.711	1.715	6.311	4.219
$DUQ_{512-512}$	1.322	0.864	2.695	2.907
Micro	0.452	0.779	2.231	2.301
Micro-Macro	**0.311**	**0.642**	**2.045**	**1.637**

Table 8 lists the averaged RMSE values for different methods for forecasting chill, torridity, storm, and rainstorm, corresponding to lowest temperature,

highest temperature, highest wind speed, and highest precipitation, respectively. Our Micro-Macro model clearly outperforms all other methods, with its RMSE values of 0.311, 0.642, 2.045, and 1.637, respectively, in forecasting chill, torridity, storm, and rainstorm. SVR is the poorest performer. WRF-HRRR performs worse than Micro, $\text{DUQ}_{512-512}$, and Micro-Macro, in forecasting chill, torridity, and storm. For rainstorm forecasting, it performs better than all other models except our Micro-Macro model. $\text{DUQ}_{512-512}$ performs worse than both Micro and Micro-Macro models. These results demonstrate the effectiveness of our Micro-Macro model for forecasting abnormal weather conditions.

6 Conclusion

This paper has dealt with a novel deep learning model which takes both the atmospheric numerical output and the ground measurements taken as the inputs for the very first time, dubbed as the Micro-Macro model for precise regional weather forecasting in multiple short-term time horizons. Our model employs the LSTM structure to capture the temporal variation of weather conditions and incorporates two data sources that include most relevant parameters for individual weather parameter forecasting per Mesonet station site via one model instance, called a modelet. A Periodical Mapper is also designed based on the neural network and Fourier Transform to effectively capture the periodical patterns of temporal data. Experimental results demonstrated that our modelets can achieve much better meteorological forecasting with finer time granularity than almost all examined counterparts, to address an urgent need of national importance.

Acknowledgement

This work was supported in part by NSF under Grants 1763620, 1948374, and 2019511. Any opinion and findings expressed in the paper are those of the authors and do not necessarily reflect the view of funding agency.

References

1. Mesonet. `https://www.mesonet.org/`, 2021.
2. South Alabama Mesonet. `http://chiliweb.southalabama.edu/`, 2021.
3. WRF Resources. `http://home.chpc.utah.edu/~u0553130/Brian_Blaylock/wrf.html`, 2021.
4. Sheila Alemany, Jonathan Beltran, Adrian Perez, and Sam Ganzfried. Predicting hurricane trajectories using a recurrent neural network. In *Proceedings of AAAI Conference on Artificial Intelligence*, volume 33, pages 468–475, 2019.
5. Sahar Arshi, Li Zhang, and Becky Strachan. Weather based photovoltaic energy generation prediction using lstm networks. In *Proceedings of International Joint Conference on Neural Networks (IJCNN)*, 2019.
6. Peter Bloomfield, Harry L Hurd, and Robert B Lund. Periodic correlation in stratospheric ozone data. *Journal of Time Series Analysis*, 15(2):127–150, 1994.

7. Antoine Bordes, Sumit Chopra, and Jason Weston. Question answering with subgraph embeddings. In *Proceedings of Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 615–620, 2014.
8. Zdravko I Botev, Joseph F Grotowski, Dirk P Kroese, et al. Kernel density estimation via diffusion. *The Annals of Statistics*, 38(5):2916–2957, 2010.
9. Kyunghyun Cho, Bart van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using rnn encoder–decoder for statistical machine translation. In *Proceedings of Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1724–1734, 2014.
10. Mladen Dalto, Jadranko Matuško, and Mario Vašak. Deep neural networks for ultra-short-term wind forecasting. In *Proceedings of IEEE International Conference on Industrial Technology (ICIT)*, pages 1657–1663, 2015.
11. Peter D Dueben and Peter Bauer. Challenges and design choices for global weather and climate models based on machine learning. *Geoscientific Model Development*, 11(10):3999–4009, 2018.
12. André Gensler, Janosch Henze, Bernhard Sick, and Nils Raabe. Deep learning for solar power forecasting—an approach using autoencoder and lstm neural networks. In *Proceedings of International Conference on Systems, Man, and Cybernetics (SMC)*, pages 2858–2865, 2016.
13. Aditya Grover, Ashish Kapoor, and Eric Horvitz. A deep hybrid model for weather forecasting. In *Proceedings of 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 379–386, 2015.
14. Emilcy Hernández, Victor Sanchez-Anguix, Vicente Julian, Javier Palanca, and Néstor Duque. Rainfall prediction: A deep learning approach. In *Proceedings of International Conference on Hybrid Artificial Intelligence Systems*, pages 151–162, 2016.
15. Qinghua Hu, Rujia Zhang, and Yucan Zhou. Transfer learning for short-term wind speed prediction with deep neural networks. *Renewable Energy*, 85:83–95, 2016.
16. National Center for Atmospheric Research. *Understanding the earth system*, 2019.
17. Alan V Oppenheim, John R Buck, and Ronald W Schafer. *Discrete-time signal processing. Vol. 2.* Upper Saddle River, NJ: Prentice Hall, 2001.
18. Baoxiang Pan, Kuolin Hsu, Amir AghaKouchak, and Soroosh Sorooshian. Improving precipitation estimation using vonvolutional neural network. *Water Resources Research*, 55(3):2301–2321, 2019.
19. Ravi Kumar Pandit, Athanasios Kolios, and David Infield. Data-driven weather forecasting models performance comparison for improving offshore wind turbine availability and maintenance. *IET Renewable Power Generation*, 14(13):2386–2394, 2020.
20. Y Radhika and M Shashi. Atmospheric temperature prediction using support vector machines. *International Journal of Computer Theory and Engineering*, 2009.
21. Sebastian Scher. Toward data-driven weather and climate forecasting: Approximating a simple general circulation model with deep learning. *Geophysical Research Letters*, 45(22):12–616, 2018.
22. Pantelis R Vlachas, Wonmin Byeon, Zhong Y Wan, Themistoklis P Sapsis, and Petros Koumoutsakos. Data-driven forecasting of high-dimensional chaotic systems with long short-term memory networks. *Proceedings of Royal Society A: Mathematical, Physical and Engineering Sciences*, 474(2213):20170844, 2018.
23. Bin Wang, Jie Lu, Zheng Yan, Huaishao Luo, Tianrui Li, Yu Zheng, and Guangquan Zhang. Deep uncertainty quantification: A machine learning approach

for weather forecasting. In *Proceedings of 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 2087–2095, 2019.
24. S Weinstein and Paul Ebert. Data transmission by frequency-division multiplexing using the discrete fourier transform. *IEEE transactions on Communication Technology*, 19(5):628–634, 1971.
25. Jonathan A Weyn, Dale R Durran, and Rich Caruana. Can machines learn to predict weather? using deep learning to predict gridded 500-hpa geopotential height from historical weather data. *Journal of Advances in Modeling Earth Systems*, 11(8):2680–2693, 2019.
26. Jonathan A Weyn, Dale R Durran, and Rich Caruana. Improving data-driven global weather prediction using deep convolutional neural networks on a cubed sphere. *Journal of Advances in Modeling Earth Systems*, 12(9), 2020.
27. SHI Xingjian, Zhourong Chen, Hao Wang, Dit-Yan Yeung, Wai-Kin Wong, and Wang-chun Woo. Convolutional lstm network: A machine learning approach for precipitation nowcasting. In *Proceedings of Advances in Neural Information Processing Systems*, pages 802–810, 2015.
28. Xiuwen Yi, Junbo Zhang, Zhaoyuan Wang, Tianrui Li, and Yu Zheng. Deep distributed fusion network for air quality prediction. In *Proceedings of 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 965–973, 2018.