ISEIS

Journal of Environmental Informatics Letters 6(2) 114-124 (2021)

Journal of Environmental Informatics Letters

www.iseis.org/jeil

Outdoor Relative Humidity Prediction via Machine Learning Techniques

R. T. Zarinkamar[1], and R. V. Mayorga[1*]

[1] *Faculty of Engineering and Applied Science, University of Regina, Regina, Saskatchewan, S4S 0A2 Canada*

Received 06 August 2021; revised 10 September 2021; accepted 03 October 2021; published online 15 November 2021

ABSTRACT. In an environmental control system, relative humidity (RH) is a very important factor because of its direct impact on humans or even animals and plants. However, there are few studies focused on prediction of humidity variables. The main objective of this paper is to show the capability of machine learning algorithms for RH prediction. In this study, a Long-Short Term Memory (LSTM) and four popular machine learning algorithms; namely, Multi-Layer Perceptron (MLP), Random Forest (RF), k-Nearest Neighbor (KNN) and Support Vector Machine for Regression (SVMR) are presented for a year period of time-series relative humidity to predict for a particular case in an Italian city. In order to have precise performance, data pre-processing is done before running the models. This thorough examination proves the positive effect of all Machine Learning-based algorithms in time-series relative humidity prediction based on predictive accuracy. Over the different metrics, LSTM indicates the best performance among all considered algorithms.

Keywords: deep learning, LSTM, machine learning, relative humidity, weather prediction

1. Introduction

Today, using data for decision-making has become a fundamental part of all enterprises throughout the world including weather forecasting, transportation, healthcare, etc. Since the merging of mathematics and computer science, many computational predictive models have been improved to help industries to make better decisions in dynamic competitive environments. For this purpose, to analyze the data, Machine Learning is an efficient data analysis technique to help industries to predict without much human intervention (Salman et al., 2018). Supervised Machine Learning techniques try to identify the outcomes by exploring the relationships and patterns among variables. Those techniques can be used by adjusting an algorithm to make the minimum error between the observed values and predicted values. So, they can learn and perform accurately just by having a sufficient amount of data (Talebizarinkamar, 2020).

One application of Machine Learning techniques is to forecast the weather. Weather forecasting is very crucial for human health, agricultural industries, wildfire, water resources management, etc. Machine Learning techniques can use enough observed weather data to find the pattern between input data i.e., the features that affect the weather, and output data. There are many Machine Learning techniques for predicting the weather (Moon et al., 2019; Md Abul Ehsan et al., 2019; Diez-Sierra and del Jesus, 2020) but the accuracy depends on many factors such as the amount of data, input factors, and output factors. The purpose of this study is to find the relative humidity (RH) of the weather in an Italian city using several predictive techniques to find the best method.

Humidity is the concentration of moisture in the air. Therefore, both climate and weather are affected by humidity because it is a significant factor of the hydrological cycle (Gunawardhana et al., 2017). Climate Scientists indicate that the capacity of holding moisture of the atmosphere increases 7% for every 1 °C of warming causing extreme rainfall and severe floods (Trenberth et al., 2003; Schaller et al., 2016). Different fields, including medicine, ecology, agriculture and hydrology, generally measure relative humidity (Gunawardhana et al., 2017).

The relative humidity is the ratio of the actual amount of vapor in the air to the actual amount of vapor that can exist in the air at a certain temperature. The value of the relative humidity is always a percentage. The relative humidity is one of the most significant parameters needed for evapotranspiration estimation. When the moisture is less in the atmosphere, the evaporation from the soil and open bodies of water is enhanced (Webster and Sherman, 1995). Also, humidity is an important factor for plants to remain wet for a longer duration (Sentelhas et al., 2008). It affects the plants' nutrient concentration and photosynthetic rates (Butler and Tibbitts, 1979).

In the medical field, relative humidity affects the physiolog-ical processes; low relative humidity can cause serious problems such as making the nasal system susceptible to penetration of viruses, nosebleeds, respiratory problems and eye irritation (Arundel et al., 1986). Humans are very sensitive to humidity; our bodies rely on the air to dispose of the moisture and sweat from our body. Sweat is our bodies' attempt to keep cool and maintain the best temperature. If the relative humidity is 100%, sweat on the body cannot evaporate. As a result, our body feels hotter. On the other hand, if the relative humidity is 0%

* Corresponding author. Tel.: 306-585-4726; fax: 306-585-4855.
E-mail address: Rene.Mayorga@uregina.ca (R. V. Mayorga).

ISSN: 2663-6859 print/2663-6867 online
 doi:10.3808/jeil.202100074

sweat can evaporate easily and our body feels cooler in the medical field, relative humidity affects the physiological processes; low relative humidity can cause serious problems such as making the nasal system susceptible to penetration of viruses, nosebleeds, respiratory problems and eye irritation (Arundel et al., 1986). Humans are very sensitive to humidity; our bodies rely on the air to dispose of the moisture and sweat from our body. Sweat is our bodies' attempt to keep cool and maintain the best temperature. If the relative humidity is 100%, sweat on the body cannot evaporate. As a result, our body feels hotter. On the other hand, if the relative humidity is 0% sweat can evaporate easily and our body feels cooler than the actual temperature. In relation to climate change, when the temperature and humidity are high, the rate of evaporation for cooling the body goes down and as a result our core body temperature can rise to a harmful level (Sherwood and Huber, 2010).

Yau and Hasbi (2013) claimed that buildings consume more than 40% of all produced energy in the world, which causes more than 30% of global greenhouse gas emissions. Energy consumption in a high-demand sector subjected to temperature and humidity changes in the environment is one of the major issues (Wang et al., 2010; Ren et al., 2011).

Planning for climate adaptation with mitigation measures and energy demand estimation needs ground-level climate projections. Future projections of Relative humidity with a high temporal resolution can be produced using robust Machine Learning techniques. The goal is to understand and solve these problems through Machine Learning methods.

Machine Learning methods provide more precise predictions by creating sophisticated models by exploring the structure and patterns of the climate data. One drawback is they can be computationally complex and must be calibrated (Cramer et al., 2017). To this end, this study, for the first time, implements a Long-Short Term Memory (LSTM) model as well as four classical Machine Learning models, namely, Multi-Layer Perceptron (MLP), Random Forest (RF), K-Nearest Neighbor (KNN) and Support Vector Machine for Regression (SVMR), to predict outdoor relative humidity. These models use algorithms that can achieve higher performance than many other algorithms because of their ability to learn nonlinear and complex relationships. In particular, the LSTM algorithm is highly effective for time-series prediction, and they have never been used in specific studies on relative humidity prediction.

1.1. Literature Review

In the last few years, Machine Learning regression algorithms have turned out to be highly efficient for prediction in weather-related areas. However, there are few papers focused on the prediction of relative humidity compared to other climate factors. Gunawardhana et al. (2017) used Large-scale general circulation models (GCMs) to downscale the minimum air temperature to predict relative humidity. In the study by Molano-Jimenez et al. (2018), applied machine learning algorithms are used to predict relative humidity in an indoor environment. Our proposed approach uses a similar approach; however, in a larger outdoor environment. Furthermore, our study tests different structures of an Artificial Neural Network (ANN) i.e., a different number of hidden layers, to find the best architecture for prediction. Also, the conclusion of their study is that a single numeric metric without any comparative visualization of predicted value vs actual values. Indeed, it's important to mention that selected temperature and CO_2 concentration are considered as input features. Yang (2019) used a Fourier-series-based model to predict hourly air humidity fluctuation.

On the other hand, there are many papers that use Machine Learning algorithms for the prediction of weather-related variables. Wu (2009) proposed a K-Nearest Neighbor (KNN) non-parametric based estimation of regression for rainfall prediction. Lee et al. (2017) presented K-Nearest Neighbor for climate analysis. Their study revealed that the proposed KNN-based model can be used as an alternative forecasting tool for a meteorological application in achieving greater forecasting accuracy and increasing prediction quality.

Paniagua-Tineo et al. (2011) presented the Support Vector Machine for Regression (SVMR) approach for daily maximum temperature prediction. Their paper indicated that the SVMR model gives an accurate estimation of the temperature for the next 24 hours. On the other hand, Ortiz-García et al. (2012) showed how the SVMR algorithm obtains high accurate results for short-term (6-hour later) temperature prediction. Rasouli et al. (2012) indicate how three nonlinear Machine Learning models, namely the SVMR method, the Bayesian Neural Net-work (BNN) and the Gaussian Process (GP), outperformed Multiple Linear Regression (MLR) for streamflow forecasting.

In the Granata (2019) and Granata et al. (2020) study, Random Forest and K-Nearest Neighbors (KNN) performed better than the Additive Regression of Decision Stump and Multilayer Perceptron (MLP). The MLP model, which is a popular feed-forward Artificial Neural Networks model, has already been applied in many studies concerning weather prediction (Maqsood et al., 2005; Dutot et al., 2007; Hayati and Mohebi, 2007; Granata et al., 2020).

However, all Machine Learning models mentioned above assume that the data is non-sequential, and that each data point is independent of the others. As a result, the inputs are analyzed in isolation, which can cause problems if there are dependencies in the data (Kratzert et al., 2018). Recurrent neural networks (RNNs) are a special type of neural network architecture that have been specifically designed to understand temporal dynamics by processing the input in its sequential order. In recent years, the LSTM method, which is a modern recurrent neural network architecture, has been proposed in many studies. The LSTM is powerful at forecasting time series data, and it is capable of remembering long-term information due to its mechanisms and recurrent structure. Chen et al. (2018) and Liu et al. (2018) applied LSTM to forecast wind speed. Qu et al. (2016) indicate how the LSTM model outperforms the Backpropagation Neural Network and the Support Vector Machine (SVMR) models for weather prediction. Qing and Niu (2018) show that a developed LSTM model is 18.34% more accurate than multi-Layered feed-forward neural networks using back-propagation algorithm (BPNN) in terms of root mean square error (RMSE) for hourly day-ahead solar irradiance prediction.

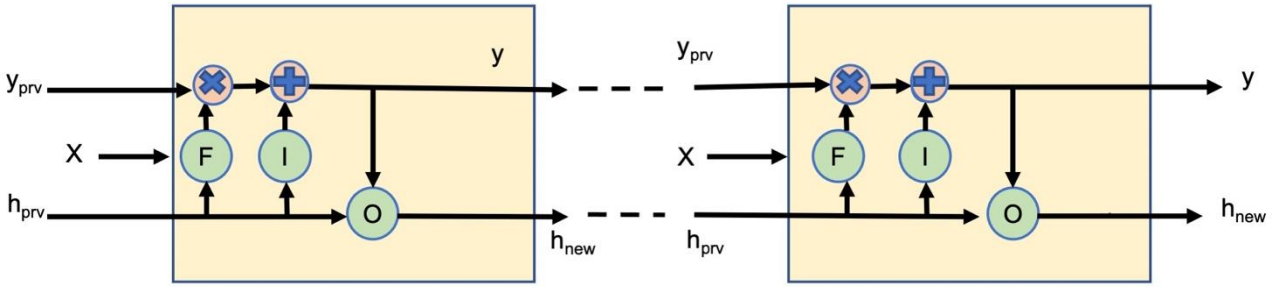


Figure 1. The architecture of a LSTM algorithm, where F, I and O denote as forget, input and output gate, respectively (Kratzert et al., 2018).

The classical Machine Learning models mentioned above take the data one-by-one and assume all data are independent, which may cause inaccuracy on sequential data calculations. So, this study considered the LSTM model to evaluate its performance when there are dependencies in the data. In addition, unlike most studies that estimate indoor relative humidity, this study predicts outdoor relative humidity for the next 24 hours. One day ahead could be enough to take action against low relative humidity, which may cause diseases, more energy consumptions, more greenhouse gas emissions. Furthermore, unlike the current literature, this study develops and tunes all five machine learning models, including long-short term memory (LSTM), Multi-Layer Perceptron (MLP), Support Vector Machine for Regression (SVMR), Random Forest (RF), and K-Nearest Neighbor (KNN) to find the best performance. These models are powerful to learn from highly nonlinear and complex weather-related data. To the authors' knowledge, this is the first study that uses the LSTM algorithm as well as four other Machine Learning algorithms for relative humidity prediction. Also, in addition to the numeric metric, this study evaluates the estimation of each model by comparative visualization. Furthermore, it is important to mention that our study also takes into account many more weather-related variables and selects input features by the Pearson correlation and feature importance method.

2. Machine Learning Methods

Machine learning algorithms are very popular for prediction. Most of them are supervised which means that they need labeled samples to learn and adapt themselves. In this section, five Machine Learning models for relative humidity prediction are introduced, including Long-Short Term Memory (LSTM), Multi-Layer Perceptron (MLP), Random Forest (RF), K-Nearest Neighbor (KNN) and Support Vector Machine for Regression (SVMR).

2.1. Long-Short Term Memory

The Recurrent Neural Network (RNN) is a suitable Deep Learning method for modeling sequential data due to its ability to remember the analysis that was previously done. Bengio et al. (1994) indicated that the traditional RNN could not remember more than 10 sequences. It means the model can only use a short period of 10 hours of relative humidity data as an input to estimate the value of the next hour. Also, another weakness of RNN model is that the gradient of the loss function in the model can decay dramatically after long time (vanishing) or it can be accumulated exponentially during the time (exploding). The LSTM is a specific type of RNN, introduced to overcome the weakness of the traditional RNN to learn long-term dependencies (Kratzert et al., 2018). The LSTM contains memory blocks composed of four elements; namely, the memory cell (which is responsible for holding the data as well as three gates), the input, output and forget gates (Hochreiter and Schmidhuber, 1997). The input and output gates are responsible for writing into the memory cell and reading and sending back data to the recurrent network. The forget gate introduced by (Gers et al., 1999) is trying to find out how much of the previous data should be deleted or maintained. In fact, by executing these gates, LSTM is able to remember what the network needs and ignores useless data.

To describe how the LSTM works, the predicted values (outputs) for a specific time step come from the input $X = [X_1, X_2, \ldots, X_n]$ consisting of the last n consecutive time steps of independent variables (in our case temperature and absolute humidity) and is processed sequentially.

The input data, denoted as X, the previous output denoted ad y_{prv} and the internal state denoted as h are used in the node's calculations as shown in Figure 1. Then, the results provide an output value and updates the stats. Here the gate parameters control the flow information and determine which information should be used and which should be ignored for further calculations.

Regarding the training of the model, on each an iteration step of LSTM's training, weights and biases — the adaptable parameters of the algorithm — are updated based on a given loss function. So, this study regards the literature applied mean squared error (MSE). On the other hand, the model's hyperparameters are the parameters which cannot be achieved from the data, and they should be set before the beginning of the learning process. The specific hyperparameters (architecture) of the LSTM algorithm, i.e., the length of neurons (5, 10, 15, 20, 50, and 100), the number of layers (1, 2, and 3), the length of inputs and dropout rates are achieved manually through a number of experiments from given relative humidity data in an Italian city. Throughout the experiments, a one, two or three-layer LSTM model, with each layer having a different number of

neurons, is used. To prevent the model from over-fitting, the dropout (10% in this study) is added between the layers (Srivastava et al., 2014). Dropout ignores the random neurons in the neural network during the training process in order to force the network to learn robustly. Regarding the activation function, the tanh function (default function) is used for inner state. Also, for optimizing the object function, the ADAM optimizer is used. Another important hyperparameter in the LSTM model is the length of input sequence (time steps), which corresponds to the number of hours of relative humidity input data to predict the next relative humidity value, i.e., the algorithm needs to know the previous N data to predict the $N + 1$ value (Kratzert et al., 2018). The study tests the values at 24, 168 and 720 in order to capture at least the dynamics of a full day, week and month cycle to predict the relative humidity for the next (next 24 hours) day.

2.2. Multi-Layer Perceptron

A Multilayer Perceptron (MLP) is a class of feed-forward Artificial Neural Network with at least three layers, i.e., an input layer, a hidden layer, and an output layer (Aish et al., 2015), as shown in Figure 2.

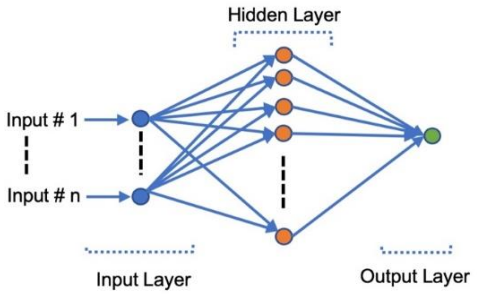


Figure 2. A multilayer perceptron (MLP) with three hidden layers (Gardner and Dorling, 1998).

Each node is a neuron that uses a nonlinear activation function except the input layer. For training, MLP usually utilizes a supervised learning model, called back-propagation. Back-propagation is a technique applied in Artificial Neural Networks to compute a gradient that is required in the computation of the weights to be used in the network (Aish et al., 2015). This method is capable of distinguishing data which is not linearly separable (Cybenko, 1989). Applications include prediction, speech recognition, image recognition and machine translation (Fan et al., 2016). The proposed study develops an MLP model for a range of one-year time series forecasting relative humidity using Grid Search (a function to find the best parameters with respect to the data) with 5-fold cross-validation to find the best estimator of MLP for prediction. The model is optimized by the mean squared error loss function and uses the Adam version of stochastic gradient descent. Furthermore, the model is tested by a range of hidden layers [1, 2, …, 25] and a range of alpha [10^ range of (-7, -1)] in 1500 training iterations to be calibrated.

2.3. Random Forest

The Random Forest is an efficient algorithm for both regression and classification, which builds multiple decision trees and combines them to get accurate results. These trees are grown simultaneously to reduce the variance and bias of the model (Breiman, 2001). The algorithm draws multiple bootstrapped samples from the main dataset to use them for each decision tree. Then the algorithm selects a small number of predictors randomly. This process continues until the model finds the accurate result by aggregation of the prediction of the trees (Ahmad et al., 2018):

$$Y_{RF} = \frac{1}{T}\sum_{t=1}^{T} Y_t \tag{1}$$

where T is the number of trees and Y_t is the result of the t^{th} tree.

This process of finding the average of decision trees continues until the best results are achieved (Ahmad et al., 2018). RF can be used in both classification and regression models. Figure 3 shows a random forest with two trees.

The Random Forest needs to tune three hyper-parameters, which includes the number of trees in the forest, attribute selec-

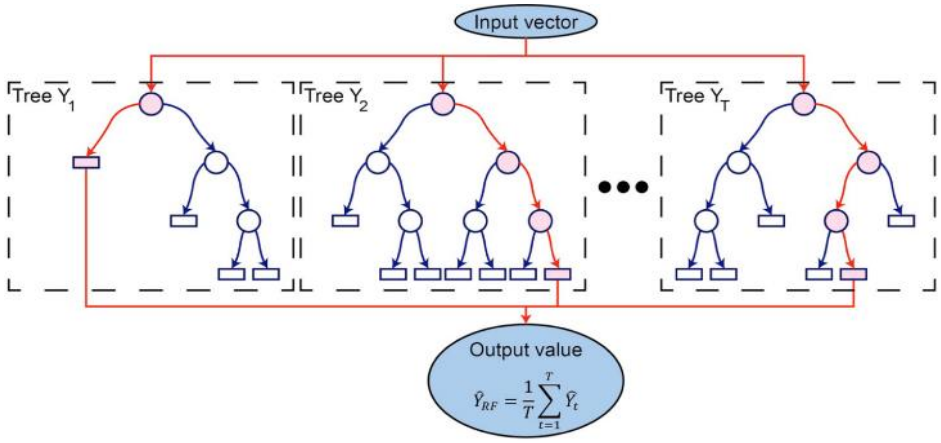


Figure 3. The functional design of a Random Forest model (Bienvenido-Huertas et al., 2020).

tion, which denotes the number of random features during the process at each node, and the number of minimum samples to split a node. In this study, the Random Forest model is developed using Grid Search function with 5-fold cross-validation to find the best hyperparameters' values. The model is tested by maximum depth: [110, 120, 130, 140, 150], minimum samples leaf: [3, 4, 5], minimum samples split: [2, 3, 5, 8], and number of estimators: [50, 100, 300, 1000]. The best structure of model would be used for the relative humidity prediction.

2.4. Support Vector Machine for Regression

The Support Vector Machine is a powerful supervised learning algorithm introduced by (Boser et al., 1992). Initially, the Support Vector Machine algorithm was applied to classification tasks. The goal of this technique was to find a hyperplane in an *N*-dimensional space to distinguish classes of data points. There could be many hyperplanes, but the Support Vector Machine tries to find a plane with the maximum margin. Then Drucker et al. (Drucker·et al., 1997) introduced the Support Vector Machine for the Regression (SVMR) method in 1997. The SVMR uses an epsilon-sensitive function and considers the points which are within the boundary lines. The best fit line in this algorithm is a hyperplane that has a maximum number of points (Sermpinis et al., 2017). Figure 4 shows the margin of tolerance between two decision boundaries which the SVMR tries to take only the points which are in the boundary.

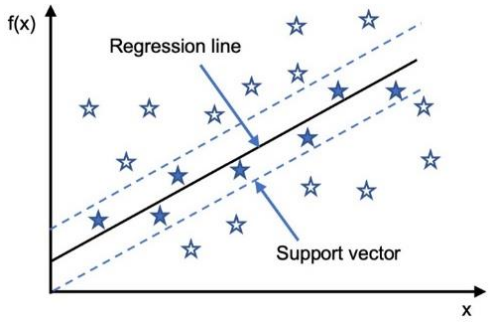


Figure 4. Support Vector regression with decision boundary (Kleynhans et al., 2017).

Support Vector Regression has adaptable hyperparameters which need to be tuned based on the data, including, (a) the kernel function, (b) penalty parameter (*C* for the error). This study developed an SVMR model using Grid Search with 5-fold cross validation to find the *C* parameter for the different values {0.001, 0.1, 1, 10, 100}, and linear and RBF kernel to calibrate the SVMR algorithm for a better performance prediction.

2.5. K-Nearest Neighbor

Perhaps one of the simplest algorithms in Machine Learning is the nearest neighbor technique. In the K-Nearest Neighbor (KNN) technique, the algorithm assumes that similar things are near to each other without making any assumption for data distribution (non-parametric) (Cover and Hart, 1967). In this technique, *k* indicates the number of training data points which are the neighbors of the numerical target. In one simple way, KNN regression tries to compute the average value of *k*'s nearest neighbors to the target point. It means that we have to find *k*'s nearest neighbors to the target point. Then, the average of those *k* numerical values is the prediction of the target. Figure 5 shows how the KNN algorithm works when $k = 4$. There are some weight functions for calculating the average of the neighbor's value. In this study, "uniform" and "distance" functions were used. The uniform function assumes that all neighbors have the same weight but in the distance function, the nearest neighbor has the highest weight, and the furthest neighbor has the lowest. Different number of neighbors; $k = 1, 2, 3, 4, 10, 15, 20, 30, 50, 70$ and 100, uniform and distance function are tested for calibration by Grid Search with 5-fold cross validation in this study.

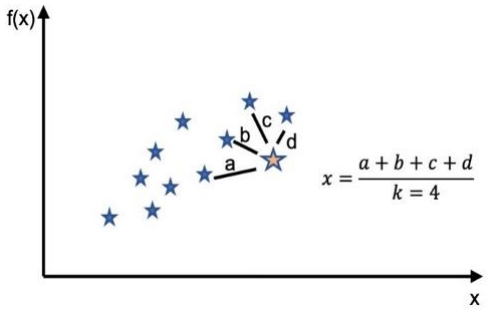


Figure 5. A K-nearest neighbor for regression's scheme in calculation.

3. Data and Methodology

This section describes the dataset, Pearson feature selection, and also metrics for evaluating the models. The implementation of all models considered here use the Keras module in Python programming language. The process includes the following steps as shown in Figure 6. The implementation was carried out on a laptop with 2.6 GHz 6-core 9th generation Intel Core i7 processor and 16 GB 2400 MHz DDR4 memory.

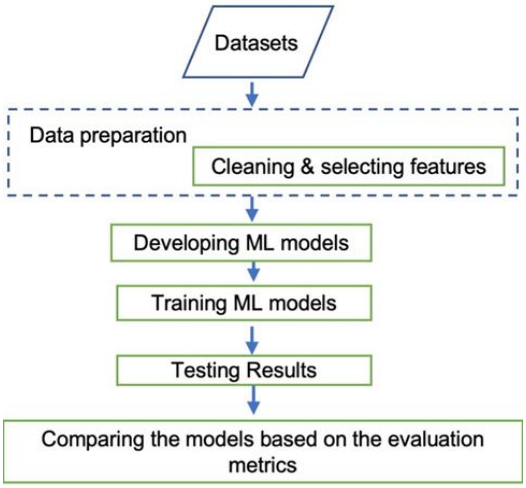
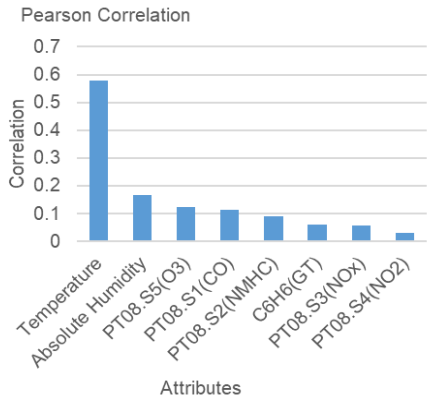


Figure 6. The whole process from data preparation to the comparison of all models.

3.1. Data Description

The used dataset was published at the repository of the University of California at Irvine which was used in an urban pollution monitoring study (De Vito et al., 2009). It includes 9358 observations of hourly measurements from five metal oxide chemical sensors which are embedded in an Air Quality Chemical Multisensory device, located at road level in a significantly polluted Italian city (UCI, 2019). A year of (March 2004 to February 2005) recorded data represents the longest freely available measurements of on-field air quality chemical devices. Ground truth hourly averaged concentrations for CO, Non-Metanic Hydrocarbons, Benzene, total Nitrogen Oxides (NO_x) and Nitrogen Dioxide (NO_2) are obtained from a co-located reference certified analyzer. While previous studies used different numbers and types of variables for relative humidity prediction e.g., temperature and CO_2 concentration were selected by Molano-Jimenez et al. (2018) work. Also, Gunawardhana et al. (2017), used Large-scale general circulation models (GCMs) to downscale the minimum air temperature to predict the relative humidity.

For the implementation of Machine Learning models, feature selection process plays a key role. The number of features could vary from two to many, and many of them may have less correlation with the target variable i.e., it means that the effect of these variables for prediction is unimportant. First of all, the study dropped the columns (attributes) with more than 50% NaN (not a number) values. Then, missing values for such a feature were replaced by the feature daily mean. In total, after removing the missing values, the dataset contained 8991 data samples.



Notes: PT08.S1 (tin oxide) hourly averaged sensor response (nominally CO targeted), PT08.S2 (titania) hourly averaged sensor response (nominally NMHC targeted), PT08.S3 (tungsten oxide) hourly averaged sensor response (nominally NO_x targeted), PT08.S4 (tungsten oxide) hourly averaged sensor response (nominally NO_2 targeted), PT08.S5 (indium oxide) hourly averaged sensor response (nominally O_3 targeted), C_6H_6(GT): True hourly averaged Benzene concentration.

Figure 7. Pearson correlation for all the variables in relative humidity prediction.

The study used the Filter method for feature selection. In the Filter method, the model takes only the subset of the relevant features. Pearson correlation is widely used for this model, where the correlation of the independent variables was computed with the relative humidity feature. Figure 7 ranked the Pearson correlation of all features with relative humidity.

Normally, the Pearson feature selection is used for linear relationships, but a weak linear correlation does not mean that there is no coupling relation between the variables (Huang et al., 2020) used similar approach for complex models.

To analyze the data in-depth, our study also used a feature importance technique alongside of Pearson feature selection. Feature importance refers to a class of techniques for assigning scores to input features to a predictive model that indicates the relative importance of each feature when making a prediction.

This study based on (Vens and Costa, 2011) research used Random Forest Feature Importance which can score features from the most relevant feature to the least relevant for relative humidity prediction. Figure 8 shows the result of Random Forest feature importance. It is shown that temperature and absolute humidity have the greatest scores with the values of almost 0.7 and 0.3, respectively, and the converse, PT08.S1, PT08.S2, PT08.S3, PT08.S4, and C_6H_6(GT) achieved scores with zero values which means that they are not effective in relative humidity prediction. Temperature and absolute humidity have the highest correlation based on Pearson and Random Forest feature importance, and here the study is going to select these two features to feed the Machine Learning models.

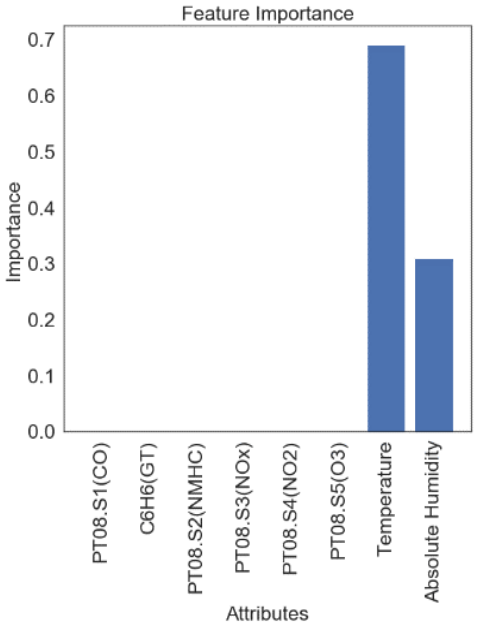


Figure 8. Scoring all variables based on Random Forest feature importance.

Splitting proportions in datasets depend on the amount of dataset variations. Here because of the sufficient amount of data for the Machine Learning models, the study splits the data to 80 and 20% for training and testing, respectively (Sharma et al., 2011). The study used 80% of the data for Grid Search function to train and find the best hyperparameters using 5-fold

cross-validation (Diez-Sierra and del Jesus, 2020). The Grid Search function evaluates all possible combinations of hyperparameters' values on the training datasets to find the best combination to improve the performance of the model. Also, *K*-fold cross-validation divides the data into *K* equal size of subsets. The model uses *K*-1 subset for the training dataset, whereas the only remaining subset is used for validation. This process continues until all subsets are used for validation, and at the last stage, the average of all subsets' values is considered as the final result. Then 20% of whole datasets are used to evaluate the performance of the models. Also, it should be mentioned that the feature values were normalized before applying the Machine Learning algorithms to avoid variables in greater value ranges dominating those with smaller ranges.

3.2. Model Evaluation

In order to evaluate the models, Mean Absolute Error (MAE), Root Mean Square Error (RMSE) and R^2 (determination coefficient) are computed. These metrics are so popular for regression model evaluation and they provide an excellent estimation of the model accuracy (Ortiz-García et al., 2012; Li et al., 2014; Liu et al., 2015; Burgan and Aksoy, 2018; Qing and Niu, 2018; Salman et al., 2018; Granata 2019; Granata et al., 2020). The Mean Absolute Error (MAE) shows the average difference between the prediction and the observed values. The Root Mean Square Error is the sample standard deviation of the differences between observed and foretold values. The coefficient of determination (R^2) is adopted to calculate the correlation between the predicted values and actual values and is always between 0% and 100%. In other words, it shows that how the predicted outcomes fit the actual values. In general, the higher the R^2, the better the model fits the data. These metrics are defined below:

(1) Mean absolute error:

$$MAE = \frac{1}{n}\sum_{i=1}^{n} |\ predicted\ value\ -\ actual\ value\ | \quad (2)$$

(2) Root mean square error:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n} (predicted\ value\ -\ actual\ value)^2} \quad (3)$$

(3) R^2 (determination coefficient):

$$R^2 = 1 - \frac{\sum_{i=1}^{n} (predicted\ value\ -\ actual\ value)^2}{\sum_{i=1}^{n} (Average\ of\ actual\ values\ -\ actual\ value)^2} \quad (4)$$

R^2 can be less than zero which indicates that the predictions are worse than random, and the highest value for R^2 is 100%; in general, the higher the R^2, the better the model fits the data.

4. Results and Discussion

The considered algorithms use the 8991 hourly observations for the period of a year. The models have been developed using 2 features as an outcome of feature selection; temperature and absolute humidity, which have the highest feature importance and correlation with the relative humidity (Shown in Figures 7 and 8). This is the first study that used and compared Machine Learning algorithms for relative humidity prediction.

The LSTM's prediction for the next 24 hours, with different values of hour time steps for looking back, are presented in Table 1. The results indicate that this model performs very well and does not suffer from overfitting. Due to verifying the ability of LSTM to remember historical data, the MAE and RMSE of LSTM with the different time steps (the values are 24, 168 and 720 in order to capture at least the dynamics of a full day, week and month cycle) are compared. For the aforementioned time steps, the training and testing errors are almost the same, which proves that implementation of this model does not suffer from overfitting. On the other hand, the results show that N = 720 has the best testing performance for MAE and RMSE with values of 2.702 and 3.932, respectively. Also, the determination coefficient for LSTM is 0.962% which shows how well the prediction values fit the actual observed values. This highly accurate performance could be due to its particular architecture which enables it to ignore the unnecessary information.

Table 1. Comparison of the Performance of the LSTM Based on Different Time Steps

	N = 24		N = 168		N = 720	
	Train	Test	Train	Test	Train	Test
MAE	3.761	3.626	3.281	3.355	2.682	2.702
RMSE	4.92	4.96	4.744	4.757	3.904	3.932
R^2	0.922	0.921	0.944	0.942	0.963	0.962

Table 2. Comparison of the Performance of the Different Regression Models for a Year Period

	MLP		RF		SVMR		KNN	
	Train	Test	Train	Test	Train	Test	Train	Test
MAE	4.999	5.111	5.390	5.391	4.632	4.615	5.610	5.497
RMSE	6.181	6.335	6.931	6.928	5.887	5.960	6.989	7.005
R^2	0.854	0.852	0.823	0.823	0.875	0.869	0.821	0.819

Also, the Table 2 shows the performance of MLP, RF, SVMR, and KNN. As the Table 2 shows, the SVMR has the best performance among all the models which is not close to the LSTM result. The MAE and RMSE for training and testing datasets indicate that SVMR model avoids overfitting with a test error of 4.615 and 5.960, respectively. The determination coefficient for SVMR is 0.869% which is not good enough compared to the LSTM. On the other hand, the SVMR, MLP, RF and KNN performed quite reasonably with a low value of errors. The worst model among these algorithms is KNN, because of the higher training and testing errors. The MAE and RMSE of KNN for the testing datasets are 5.497 and 7.005, respectively. In total, the results suggest that LSTM achieved the best performance

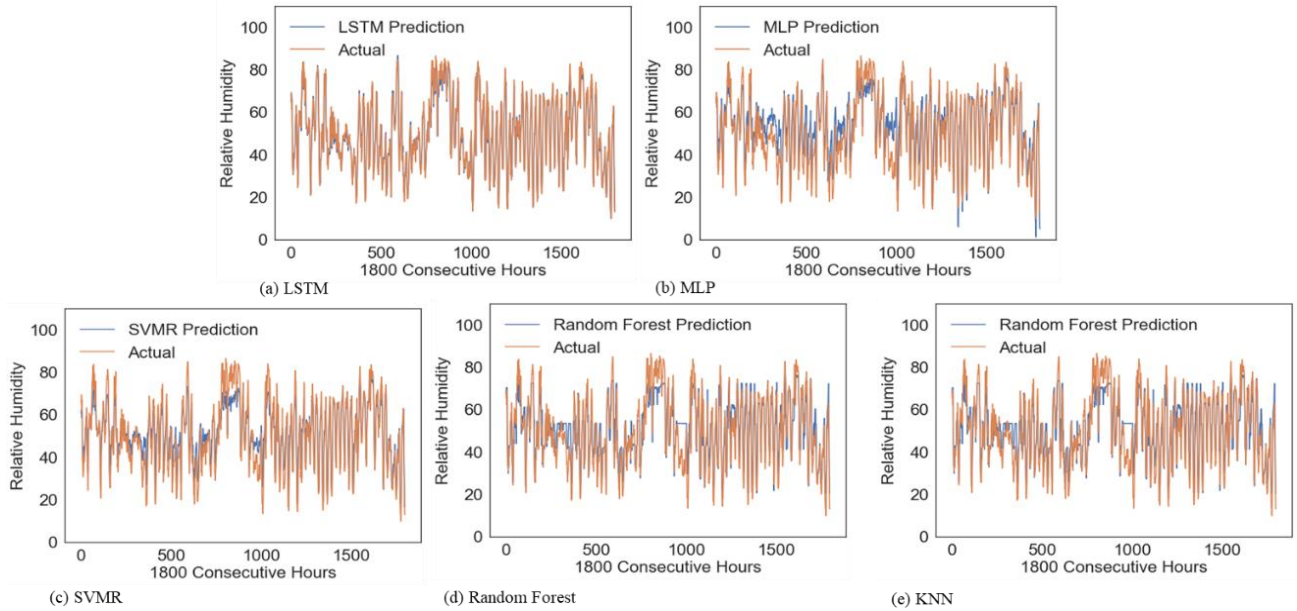


Figure 9. The hourly relative humidity time series for both actual values and predicted values for all models in a year period.

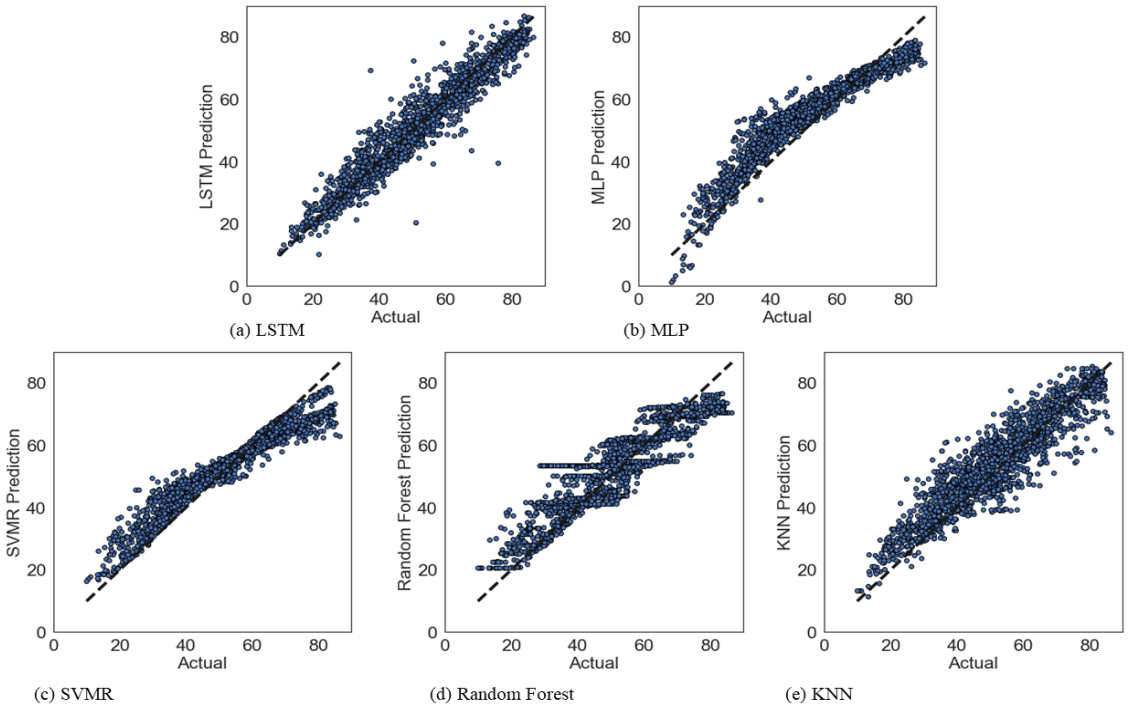


Figure 10. Comparison of correlation between predicted values and actual values for all models with scatter plot.

among the algorithms across the datasets. Although the errors for SVMR, MLP, RF and KNN are higher than the LSTM algorithm, the results of prediction are reasonable. To understand the accuracy of the predicted values, actual values and predicted values are plotted in Figure 9. It illustrates the plot for hourly relative humidity values predicted in testing data by all Machine Learning models vs actual values. From this plot it could be concluded that LSTM is very strong in nonlinear generalization, and it could be effective in hourly relative humidity prediction. The predicted values almost overlap on actual values in Figure 9(a).

To understand the accuracy of the predicted values, the relating actual values to predicted values should be plotted. The Figure 10 shows the correlation between predicted values and actual values for three different Machine Learning models for regression. These scatter plot says that if the predicted value

would be on the dashed line or very near the dashed line, the prediction is almost the same as actual value. Otherwise, the prediction is far from actual values. Each point in the plot represents a prediction for each house. Predicted points in the Figure 10(a) show that LSTM performed better than the others, and except some points, almost all other points are near the fitted line.

Machine Learning algorithms have shown that they have a strong ability in relative humidity prediction but, according to Tables 1 and 2, different algorithms have different error and determination coefficients. The results of the other previous studies in the literature are not directly comparable to our study since climate factors (here relative humidity) and the dataset are different. As an example, in (Gunawardhana et al., 2017) work, the outcomes are monthly-based predictions, whereas this study used a year term period for prediction.

5. Conclusions

In climate change studies, relative humidity is a very important factor in micro-climate because of its direct impact on humans or even animals and plants. However, a less focused feature among the weather variables is the relative humidity. In this study, a Deep Learning LSTM recurrent neural network and four Machine Learning models were applied to predict relative humidity in an Italian city, namely Long-Short Term Memory (LSTM), Multi-Layer Perceptron (MLP), Support Vector Machine for Regression (SVMR), Random Forest (RF), and K-Nearest Neighbor (KNN). Then, the results of all models were compared with both numeric metrics and visualization. The results proved that all the Machine Learning algorithms considered performed quite well, but the LSTM technique had the best performance, with MAE and the RMSE values of 2.702 and 3.932, respectively. The SVMR model achieved the second place, and the KNN model perfomed the worst among all models.

In the time-series dataset, the points are dependent on the other points. Traditional machine learning models typically cannot analyze this kind of data perfectly because they accept sequentially input after input and produce individual calculation for every hour. It's important to note that the traditional models do not remember the data just analyzed. The LSTM overcomed the traditional Machine Learning models because its capability to learn long-term dependencies. The LSTM is able to remember what the network needs and ignores useless data.

The current study feed the supervised ML models with limited data. Thus, in the future, big sequential data can be used for a developed LSTM and other Machine Learning approaches to estimate the outdoor relative humidity more accurately.

Acknowledgement. This research was supported by a grant (RGPIN-2019-07269) from the Natural Sciences and Engineering Research Council of Canada (NSERC).

References

Ahmad, M.W., Reynolds, J. and Rezgui, Y. (2018). Predictive modelling for solar thermal energy systems: A comparison of support vector regression, random forest, extra trees and regression trees. *Journal of Cleaner Production*, 203, 810-821. https://doi.org/10.1016/j.jclepro.2018.08.207

Aish, A.M., Zaqoot, H.A. and Abdeljawad, S.M. (2015). Artificial neural network approach for predicting reverse osmosis desalination plants performance in the Gaza Strip. *Desalination*, 367, 240-247. https://doi.org/10.1016/j.desal.2015.04.008

Indirect health effects of relative humidity in indoor environments. (1986) *Environmental Health Perspectives*, 65, 351-361. https://doi.org/1 0.1289/ehp.8665351

Bengio, Y., Simard, P. and Frasconi, P. (1994). Learning long-term dependencies with gradient descent is difficult. *IEEE Transactions on Neural Networks*, 5(2), 157-166. https://doi.org/10.1109/72.279181

Bienvenido-Huertas, D., Rubio-Bellido, C., Pérez-Ordóñez, J.L. and Oliveira, M.J. (2020). Automation and optimization of in-situ assessment of wall thermal transmittance using a Random Forest algorithm. *Building and Environment*, 168, 106479. https://doi.org/10.1016/j. buildenv.2019.106479

Boser, B.E., Guyon, I.M. and Vapnik, V.N. (1992). A training algorithm for optimal margin classifiers. *Proceedings of the Fifth Annual Workshop on Computational Learning Theory*, Pittsburgh, 144-152. http://doi.org/10.1145/130385.130401

Breiman, L. (2001). Random forests. *Machine Learning*, 45(1), 5-32. https://doi.org/10.1023/a:1010933404324

Burgan, H.I. and Aksoy, H. (2018). Annual flow duration curve model for ungauged basins. *Hydrology Research*, 49(5), 1684-1695. https://doi.org/10.2166/nh.2018.109

Butler, L.K. and Tibbitts, T.W. (1979). Stomatal mechanisms determining genetic resistance to ozone in *Phaseolus vulgaris* L. *Journal of American Society Horticultural Science*, 104(2), 213-216.

Chen, J., Zeng, G.Q., Zhou, W., Du, W. and Lu, K.D. (2018). Wind speed forecasting using nonlinear-learning ensemble of deep learning time series prediction and extremal optimization. *Energy Conversion and Management*, 165, 681-695. https://doi.org/10.1016/j.enconman.2018.03.098

Cover, T.M. and Hart, P.E. (1967). Nearest neighbor pattern classification. *IEEE Transactions on Information Theory*, 13(1), 21-27. https://doi.org/10.1109/TIT.1967.1053964

Cramer, S., Kampouridis, M., Freitas, A.A. and Alexandridis, A.K. (2017). An extensive evaluation of seven machine learning methods for rainfall prediction in weather derivatives. *Expert Systems with Applications*, 85, 169-181. https://doi.org/10.1016/j.eswa.2017.05.029

Cybenko, G. (1989). Approximation by superpositions of a sigmoidal function. *Mathematics of Control, Signals and Systems*, 2(4), 303-314. https://doi.org/10.1007/BF02551274

De Vito, S., Piga, M., Martinotto, L. and Francia, G.D. (2009). CO, NO_2 and NO_x urban pollution monitoring with on-field calibrated electronic nose by automatic bayesian regularization. *Sensors and Actuators, B: Chemical*, 143(1), 182-191. https://doi.org/10.1016/j.snb.2009.08.041

Diez-Sierra, J. and del Jesus, M. (2020). Long-term rainfall prediction using atmospheric synoptic patterns in semi-arid climates with statistical and machine learning methods. *Journal of Hydrology*, 586, 124789. https://doi.org/10.1016/j.jhydrol.2020.124789

Drucker, H., Burges, C.J.C., Kaufman, L., Smola, A. and Vapnik, V. (1997). Support vector regression machines. *Advances in Neural Information Processing Systems* (NIPS). MIT Press, Cambridge, MA.

Dutot, A.L., Rynkiewicz, J., Steiner, F.E. and Rude, J. (2007). A 24-h forecast of ozone peaks and exceedance levels using neural classifiers and weather predictions. *Environmental Modelling and Software*, 22(9), 1261-1269. https://doi.org/10.1016/j.envsoft.2006.08.002

Ehsan, B.M.A., Begum, F., Ilham, S.J. and Khan, R.S. (2019). Advanced wind speed prediction using convective weather variables through machine learning application. *Applied Computing and Geosciences*, 1, 100002. https://doi.org/10.1016/j.acags.2019.100002

Fan, X.H., Wang, L. and Li, S.S. (2016). Predicting chaotic coal prices using a multi-layer perceptron network model. *Resources Policy*, 50, 86-92. https://doi.org/10.1016/j.resourpol.2016.08.009

Gardner, M.W. and Dorling, S.R. (1998). Artificial neural networks (the multilayer perceptron)—a review of applications in the atmospheric sciences. *Atmospheric Environment*, 32(14-15), 2627-2636. https://doi.org/10.1016/S1352-2310(97)00447-0

Gers, F.A., Schmidhuber, J. and Cummins, F. (1999). Learning to forget: Continual prediction with LSTM. *IEE Conference Publication*, 2, 850-855. https://doi.org/10.1049/cp:19991218

Granata, F. (2019). Evapotranspiration evaluation models based on machine learning algorithms—A comparative study. *Agricultural Water Management*, 217, 303-315. https://doi.org/10.1016/j.agwat.2019.03.015

Granata, F., Gargano R. and de Marinis, G. (2020). Artificial intelligence based approaches to evaluate actual evapotranspiration in wetlands. *Science of the Total Environment*, 703, 135653. https://doi.org/10.1016/j.scitotenv.2019.135653

Gunawardhana, L.N., Al-Rawas, G.A. and Kazama, S. (2017). An alternative method for predicting relative humidity for climate change studies. *Meteorological Applications*, 24(4), 551-559. https://doi.org/10.1002/met.1641

Hayati, M. and Mohebi, Z. (2007). Application of artificial neural networks for temperature forecasting. *Proceedings of World Academy of Science, Engineering and Technology*, 22, 275-279. http://papersim.com/wp-content/uploads/Neural_Networks__Temperature_Forecasting_2007.pdf.

Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, 9(8), 1735-1780. https://doi.org/10.1162/neco.1997.9.8.1735

Huang, Y., Yang, L. and Fu, Z. (2020). Reconstructing coupled time series in climate systems using three kinds of machine-learning methods. *Earth System Dynamics*, 11(3), 835-853. https://doi.org/10.5194/esd-11-835-2020

Kleynhans, T., Montanaro, M., Gerace, A. and Kanan, C. (2017). Predicting top-of-atmosphere thermal radiance using MERRA-2 atmosphereic data with deep learning. *Remote Sensing*, 9(11), 1133. https://doi.org/10.3390/rs9111133

Kratzert, F., Klotz, D., Brenner, C., Schulz, K. and Herrnegger, M. (2018). Rainfall-runoff modelling using Long Short-Term Memory (LSTM) networks. *Hydrology and Earth System Sciences*, 22(11), 6005-6022. https://doi.org/10.5194/hess-22-6005-2018

Lee, T., Ouarda, T.B.M.J. and Yoon, S. (2017). KNN-based local linear regression for the analysis and simulation of low flow extremes under climatic influence. *Climate Dynamics*, 49(9-10), 3493-3511. https://doi.org/10.1007/s00382-017-3525-0

Li, L.J., Hu, Q.H., Wu, X.Q. and Yu, D.R. (2014). Exploration of classification confidence in ensemble learning. *Pattern Recognition*, 47(9), 3120-3131. https://doi.org/10.1016/j.patcog.2014.03.021

Liu, H., Mi, X.W. and Li, Y.F. (2018). Smart multi-step deep learning model for wind speed forecasting based on variational mode decomposition, singular spectrum analysis, LSTM network and ELM. *Energy Conversion and Management*, 159, 54-64. https://doi.org/10.1016/j.enconman.2018.01.010

Liu, H., Tian, H.Q., Li, Y.F. and Zhang, L. (2015). Comparison of four Adaboost algorithm based artificial neural networks in wind speed predictions. *Energy Conversion and Management*, 92, 67-81. https://doi.org/10.1016/j.enconman.2014.12.053

Maqsood, I., Khan, M.R., Huang, G.H. and Abdalla, R. (2005). Application of soft computing models to hourly weather analysis in southern Saskatchewan, Canada. *Engineering Applications of Artificial Intelligence*, 18(1), 115-125. https://doi.org/10.1016/j.engappai.2004.08.019

Molano-Jimenez, A., Orjuela-Cañón, A.D. and Acosta-Burbano, W. (2018). Temperature and relative humidity prediction in swine livestock buildings. *2018 IEEE Latin American Conference on Computational Intelligence (LA-CCI)*, 1-4. https://doi.org/10.1109/LA-CCI.2018.8625245

Moon, S.H., Kim, Y.H., Lee, Y.H. and Moon, B.R. (2019). Application of machine learning to an early warning system for very short-term heavy rainfall. *Journal of Hydrology*, 568, 1042-1054. https://doi.org/10.1016/j.jhydrol.2018.11.060

Ortiz-García, E.G., Salcedo-Sanz, S., Casanova-Mateo, C., Paniagua-Tineo, A. and Portilla-Figueras, J.A. (2012). Accurate local very short-term temperature prediction based on synoptic situation Support Vector Regression banks. *Atmospheric Research*, 107, 1-8. https://doi.org/10.1016/j.atmosres.2011.10.013

Paniagua-Tineo, A., Salcedo-Sanz, S., Casanova-Mateo, C., Ortiz-García, E.G., Cony, M.A. and Hernández-Martín, E. (2011). Prediction of daily maximum temperature using a support vector regression algorithm. *Renewable Energy*, 36(11), 3054-3060. https://doi.org/10.1016/j.renene.2011.03.030

Qing, X.Y. and Niu, Y.G. (2018). Hourly day-ahead solar irradiance prediction using weather forecasts by LSTM. *Energy*, 148, 461-468. https://doi.org/10.1016/j.energy.2018.01.177

Qu, X.Y., Kang, X.N., Zhang, C., Jiang, S. and Ma X.D. (2016). Short-term prediction of wind power based on deep long short-term memory. *2016 IEEE PES Asia-Pacific Power and Energy Engi-neering Conference (APPEEC)*, 1148-1152. https://doi.org/10.1109/APPEEC.2016.7779672

Rasouli, K., Hsieh, W.W. and Cannon, A.J. (2012). Daily streamflow forecasting by machine learning methods with weather and climate inputs. *Journal of Hydrology*, 414, 284-293. https://doi.org/10.1016/j.jhydrol.2011.10.039

Ren, Z.G., Chen, Z.D. and Wang, X.M. (2011). Climate change adaptation pathways for Australian residential buildings. *Building and Environment*, 46(11), 2398-2412. https://doi.org/10.1016/j.buildenv.2011.05.022

Salman, A.G., Heryadi, Y., Abdurahman, E. and Suparta, W. (2018). Single layer & multi-layer long short-term memory (LSTM) model with intermediate variables for weather forecasting. *Procedia Computer Science*, 135, 89-98. https://doi.org/10.1016/j.procs.2018.08.153

Schaller, N., Kay, A.L., Lamb, R., Massey, N.R., van Oldenborgh, G.J., Otto, F.E.L., Sparrow, S.N., Vautard, R., Yiou, P., Ashpole, I., Bowery, A., Crooks, S.M., Haustein, K., Huntingford, C., Ingram, W.J., Jones, R.G., Legg, T., Miller, J., Skeggs, J., Wallom, D., Weisheimer, A., Wilson, S., Stott, P.A. and Allen, M.R. (2016). Human influence on climate in the 2014 southern England winter floods and their impacts. *Nature Climate Change*, 6(6), 627-634. https://doi.org/10.1038/nclimate2927

Sentelhas, P.C., Dalla Marta, A., Orlandini, S., Santos, E.A., Gillespie, T.J. and Gleason, M.L. (2008). Suitability of relative humidity as an estimator of leaf wetness duration. *Agricultural and Forest Meteorology*, 148(3), 392-400. https://doi.org/10.1016/j.agrformet.2007.09.011

Sermpinis, G., Stasinakis, C., Rosillo, R., and de la Fuente, D. (2017). European exchange trading funds trading with locally weighted support vector regression. *European Journal of Operational Research*, 258(1), 372-384. https://doi.org/10.1016/j.ejor.2016.09.005

Sharma, N., Sharma, P., Irwin, D. and Shenoy, P. (2011). Predicting solar generation from weather forecasts using machine learning. *2011 IEEE International Conference on Smart Grid Communications* (*SmartGridComm)*, 528-533. https://doi.org/10.1109/SmartGridComm.2011.6102379

Sherwood, S.C. and Huber, M. (2010). An adaptability limit to climate change due to heat stress. *Proceedings of the National Academy of Sciences of the United States of America*, 107(21), 9552-9555. https://doi.org/10.1073/pnas.0913352107

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I. and Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15, 1929-

1958.

Talebizarinkamar, R. (2020). *A Machine Learning Approach for Lung and Bronchus Cancer Survival Prediction*. Master Thesis, Department of Industrial Systems Engineering, University of Regina, SK, Canada.

Trenberth, K.E., Dai, A., Rasmussen, R.M. and Parsons, D.B. (2003). The changing character of precipitation. *Bulletin of the American Meteorological Society*, 84(9), 1205-1218. https://doi.org/10.1175/BAMS-84-9-1205

UCI. Air quality data set. https://archive.ics.uci.edu/ml/datasets/Air+quality (accessed August 1, 2019).

Vens, C. and Costa, F. (2011). Random forest based feature induction. *2011 IEEE 11th International Conference on Data Mining*, 744-753. https://doi.org/10.1109/ICDM.2011.121

Wang, X.M., Chen, D. and Ren, Z.G. (2010). Assessment of climate change impact on residential building heating and cooling energy requirement in Australia. *Building and Environment* 45(7), 1663-1682. https://doi.org/10.1016/j.buildenv.2010.01.022

Webster, I.T. and Sherman, B.S. (1995). Evaporation from fetch-limited water bodies. *Irrigation Science*, 16(2), 53-64. https://doi.org/10.1007/BF00189161

Wu, J.S. (2009). A novel artificial neural network ensemble model based on k-nearest neighbor nonparametric estimation of regression function and its application for rainfall forecasting. *2009 International Joint Conference on Computational Sciences and Optimization*, 2, 44-48. https://doi.org/10.1109/CSO.2009.307

Yang, Z.C. (2019). Hourly ambient air humidity fluctuation evaluation and forecasting based on the least-squares Fourier-model. *Measurement*, 133, 112-123. https://doi.org/10.1016/j.measurement.2018.10.002

Yau, Y.H. and Hasbi, S. (2013). A review of climate change impacts on commercial buildings and their technical services in the tropics. *Renewable and Sustainable Energy Reviews*, 18, 430-441. https://doi.org/10.1016/j.rser.2012.10.035