Optimized Deep Networks Structure to Improve the Accuracy of estimator algorithm in Deep Networks learning

Hamideh Rezaei Nezhad ¹, Farshid Keynia ²*, Amir Sabbag Molahosseini ³

1- Department of Computer, Kerman Branch, Islamic Azad University, Kerman, Iran, https://orcid.org/0000-0002-7726-7171, Email: rezaei.iauqeshm@gmail.com

2- Department of Energy Management and Optimization, Institute of Science and High Technology and Environmental Sciences, Graduate University of Advanced Technology, Kerman, Iran, https://orcid.org/0000-0002-9027-7315, Email: f.keynia@kgut.ac.ir

3- Department of Computer, Kerman Branch, Islamic Azad University, Kerman, Iran, https://orcid.org/0000-0002-6024-3202, Email: Sabbagh@iauk.ac.ir

*Corresponding Author: Farshid Keynia, Department of Energy Management and Optimization, Institute of Science and High Technology and Environmental Sciences, Graduate University of Advanced Technology, Kerman, Iran, Email: f.keynia@kgut.ac.ir

Abstract

The present study considers decreasing prediction error for the types of time series and the uncertainty in estimation parameters, improving the structure of the deep neural network and increasing response speed in the proposed neural network method; besides, the competitive performance and the collaboration among the neurons of deep neural network are also increased. Selected data is related to Qeshm weather (suitable weather conditions to study our purpose) prediction during 2016 onwards. In this study, for the purpose of analyzing the prediction issue of power consumption of domestic expenses in the indefinite and severe fluctuation mode, we decided to combine two methods of Long Short-Term Memory and Convolutional Neural Networks. For the training of the deep network, the BP algorithm is used. The results indicate that Gated Recurrent Unit networks compared other models (MLP, CNN and DNN) produce more realistic results, and also two-way networks obtained better results on test data compared Long Short-Term Memory networks. RMSE prediction are more realistic than the LSTM model on test and training data against the significant data. A GRU network has two gates of rt readjust and Zt forgetting, which helps to assure that long term dependencies of gradient fading will not occur.

¹ present address: School of Electronics, Electrical Engineering and Computer Science, Queen's University Belfast, Belfast, UK, https://orcid.org/0000-0002-6024-3202, Email: a.sabbaghmolahosseini@qub.ac.uk
Keywords: Optimization algorithm, Time series, Estimation, Prediction, Convolutional Neural Network, Long Short-Term Memory.

1. Introduction

At its simplest, a neural network with some level of complexity, usually at least two layers, qualifies as a deep neural network (DNN), or deep net for short. Deep nets process data in complex ways by employing sophisticated math modeling. The optimization algorithm is formed based on training and learning and the process of training and learning in a class that was introduced for the first time by Rao et al. [1].

The uniformly minimum variance unbiased estimator is one of the most fundamental and important estimation methods in classical statistics, but its existence and characterization are usually challenging to investigate when one moves beyond exponential families. In the past several decades, many shrinkage estimation, regression and variable selection methods were proposed. For instance, the James–Stein estimator dominates the maximum likelihood estimator in terms of expected total squared loss beyond two-dimensional Gaussian models [2].

Deep learning is part of a broader family of machine learning methods, which is based on artificial neural networks with representation learning. A new efficient optimization method, called ‘Teaching–Learning-Based Optimization (TLBO)’, is for the optimization of mechanical design problems. This method works on the effect of influence of a teacher on learners. Teaching Learning Based Optimization (TLBO) algorithm utilized the ability of students in the classroom and the teaching of class teachers to improve the educational level of the class. Teachers and students are the two main elements of the TLBO algorithm. Teaching–learning-based optimization (TLBO) is a powerful metaheuristic algorithm for solving complex optimization problems pertaining to the global optimum. Many TLBO variants have been presented to improve the local optima avoidance capability and to increase the convergence speed. Due to the rapid development of engineering requirements, global optimization has attracted a lot of interest. In general, there are two classes of methods for solving global optimization problems, i.e., nature-inspired methods and deterministic methods. Because the deterministic method often fails in solving complex global optimization problems, nature-inspired methods are more popular. Meta-heuristic algorithms (MAs), which are instinctively immune from non-smooth behaviors, are the most widely used nature-inspired methods. In the past ten years, many state-
of-the-art MAs, such as quila optimizer (AO), reptile search algorithm (RSA), moth flame optimization (MFO), arithmetic optimization algorithm (AOA), whale optimization algorithm (WOA), TLBO, sine cosine algorithm (SCA), etc., have been developed. Because of its simple structure, no specific parameters, and strong practicability, TLBO has been widely used in many engineering applications, such as personalized recommender systems, controller design, wind power forecasting, parameters estimate, neural network training and so on. Accordingly, the teacher phase and student phase create two important and fundamental parts of this algorithm. The algorithm output is the scores of students and their level of knowledge in which the quality and ability of teachers in this field are so consequential, TLBO algorithm is a modern optimization algorithm based on the population in which this population is the same members of the class.

The deep neural network is one of the types of feedforward neural networks, the connection pattern among its neurons is inspired by the visual cortex of animals’ brain. The basis of the architecture of this network is based on the following concepts:

1- Separated connections
2- Common weights
3- The deletion of some neurons using the layers with the name of the merged layer

Neural networks have significant features which are considered as follows:

**Learning capability:** Learning is the capability for regulating network parameters (synaptic weight) for receiving information to be reminded in future of this purpose that if the network is trained for a specific status and a small change in network environmental situation (special condition) occurs, the network will be efficient for the new situation with brief training. There are weight changes, which are called learning laws.

**Generalization capability:** Learning, in general, manages a comprehensive range of the set of inputs and outputs. The network has learning capability and generalization power to the whole set, and also it can perform correctly for the patterns of undefined input-output.

**Parallel processing:** The general duty of processing is distributed among smaller and independent processors, which causes an increase in the speed of processing.
Studying variables in the research:

**Dependent variable:** The weights of deep neural network layers

**Independent variable:** The data of time series for training the data network

The present study considers finding a suitable answer to the following questions:

1. Which part of the deep network structure can be improved by optimization algorithms?

2. Does the estimator based on the smart method such as learning have any better performance towards classic estimators?

3. Do CNN and LSTM methods cause the decrease of the prediction error of power consumption of household charges in indefinite and severe fluctuation mode?

In this regard, the following hypotheses are considered:

- All-weather features such as the least humidity, the most humidity, humidity average, etc. will be available.

- There will not be any dependence among the dimensions

- Convolution neural network has three inputs

An optimization algorithm based on training and learning is formed based on the process of training and learning in a class. A deep neural network is one of the types of feedforward neural networks whose connection pattern among its neurons is inspired by the visual cortex of animals' brain. The present study considers decreasing prediction error for the types of time series and the uncertainty in estimation parameters, improving the structure of the deep neural network and increasing response speed in the proposed neural network method; besides, the competitive performance and the collaboration among the neurons of deep neural network are also increased.

2. Literature Review

2.1 Analysis algorithm of main components
PCA technique is the best method for decreasing data dimensions in linear mode. It means that less important coefficients obtained from this transform are deleted and lost data is lower towards other methods. PCA-CNN (Principal Component Analysis-Convolutional Neural Network) is a development method of the Convolutional Neural Network method which gives special treatment to the dimension reduction process in the input data. The dimension reduction process is carried out by utilizing the PCA method so that the data processing process becomes faster without losing important information so that better method performance is obtained. The PCA-CNN method is implemented on a dataset of the Situbondo district which is classified into five land cover classes. Of course, PCA application is not limited to the decrease of dimensions, and it is used in other fields such as pattern recognition and facial recognition. In this method, new coordinates are defined for data, and the data is noted based on these new coordinates. The first axis should be in the direction in which data variance will be maximum (it means in the direction which data separation is more). Gorgoglione et al. [3], adopted principal component analysis (PCA) to assess the effect of rainfall, watershed, and drainage network characteristics on urban nutrient runoff in poorly gauged areas. The second axis should be perpendicular to the first axis so that data variance will be maximum. Hence, the next axes perpendicular to all previous axes is such that the data have the most separation [4].

2.1.1 Accuracy of estimator algorithm in Deep Networks learning

An estimation algorithm is an algorithm which produces an estimate for some value (such as the length of the stream). Similarly, an approximation algorithm is an algorithm which given an optimization problem finds a solution which approximately optimizes the objective function of the problem. DOA estimation, also named spatial spectrum estimation, estimates the direction angle of the spatial signal reaching the array reference element by processing the received signal of the array. The traditional DOA estimation methods are mainly based on beamforming [5].

Convolutional neural network (CNN)-based deep learning (DL) is a powerful, recently developed image classification approach. With origins in the computer vision and image processing communities, the accuracy assessment methods developed for CNN-based DL use a wide range of metrics that may be unfamiliar to the remote sensing (RS) community. Array signal processing, also known as spatial domain signal processing, is an important branch of the signal processing field, widely used in radar signals, underwater sonar, wireless communication,
Mainly, to process the signals received by the array, enhance the useful signals needed, suppress useless interference and noise, and obtain important parameters, the estimation of the direction of arrival (DOA) is one of the important research contents of array signal processing. DOA estimation, also named spatial spectrum estimation, estimates the direction angle of the spatial signal reaching the array reference element by processing the received signal of the array [7].

Estimation of distribution algorithms (EDAs) are a novel class of evolutionary optimization algorithms that were developed as a natural alternative to genetic algorithms in the last decade. The principal advantages of EDAs over genetic algorithms are the absence of multiple parameters to be tuned (e.g. crossover and mutation probabilities) and the expressiveness and transparency of the probabilistic model that guides the search process. In addition, EDAs have been proven to be better suited to some applications than GAs, while achieving competitive and robust results in the majority of tackled problems. EDAs belong to the class of evolutionary algorithms. The main difference between EDAs and most conventional evolutionary algorithms is that evolutionary algorithms generate new candidate solutions using an implicit distribution defined by one or more variation operators, whereas EDAs use an explicit probability distribution encoded by a Bayesian network, a multivariate normal distribution, or another model class. Similarly as other evolutionary algorithms, EDAs can be used to solve optimization problems defined over a number of representations from vectors to LISP style S expressions, and the quality of candidate solutions is often evaluated using one or more objective functions [8].

The convolutional neural network (CNN) is a common deep learning algorithm, which is a feedforward neural network that can reduce the number of parameters to a large extent by local connectivity and weight sharing. A typical CNN model usually consists of several convolutional and pooling layers connected alternately, ending with a fully connected layer [9]. Deep learning applications are used in industries from automated driving to medical devices. Automated Driving: Automotive researchers are using deep learning to automatically detect objects such as stop signs and traffic lights. In addition, deep learning is used to detect pedestrians, which helps decrease accidents. LeNet is the first successful application of CNNs and was developed by Yann Lecun in the 1990s that was used to read zip codes, digits, etc. The latest work is called
LeNet-5 which a 5-layer CNN that reaches 99.2% accuracy on insolated character recognition [10].

2.2 Convolution neural network

The neural network appears as a practical technology that is successfully utilized in different fields of classification, estimation, speech recognition, drug recognition, image, and signal processing. There are three types of layers that make up the CNN which are the convolutional layers, pooling layers, and fully-connected (FC) layers. When these layers are stacked, a CNN architecture will be formed. The most important advantage of the neural network is self-adaptive and self-organizing capability and proactive operation, etc. In created neural network models, progressive multilayer networks are generally used with Back Propagation training algorithm (BP) for classification, which is among the supervised training methods. The structure of this network is included in an input layer, a middle layer, and an output layer. There are one or some processor elements (neuron) in each layer which are related to all neurons of the next layer with weighted connections. The vector of input data of the model is mapped to the neurons of the first layer, and in this layer, there is no process, and the neurons of the output layer are mapped to the output vector of the model. The number of neurons of input and output layers depends on the number of input and output variables of the model. Still, the selection of the number of neurons of the middle layer is determined as trial and error.

2.3 Advantages of CNNs

One of the main advantages of CNNs is that they can learn from raw pixel data, without requiring any manual feature engineering or preprocessing. This means that they can automatically discover and adapt to the most salient characteristics of the images, such as edges, shapes, colors, textures, and objects. This also reduces the dimensionality and complexity of the input data, making the training and inference faster and more efficient. Another advantage of CNNs is that they can exploit the spatial and hierarchical structure of the images, by using filters that preserve the local connectivity and context of the pixels, and by building more abstract and
high-level representations as they go deeper into the network. This allows them to capture the variability and diversity of the images, and to generalize well to new and unseen data.

Wood, investigated a machine learning (ML) approach for three objectives, one of which is predicting the location of targets. This work performed two-dimensional positioning with a circular object using the KNearest Neighbors (KNN) algorithm and a homogeneous non-magnetic wall. Common methods used in TWR for locating targets, estimating wall parameters such as permittivity, wall thickness, etc., separately [11].

Deep Networks (DNS) consists of two steps: the generation step and the pruning step. In the generation step, the network generates hidden layers layer by layer until accuracy reaches the threshold. The network uses a pruning algorithm based on Hebb’s rule or Pearson’s correlation for adaptation in the pruning step. Experimental results show that compared with traditional neural network topology optimization algorithms, GA-DNS can generate neural networks with higher construction efficiency, lower structure complexity, and higher classification accuracy. Structural optimization has over the past decades qualified as an important tool in the design process. The method can be grouped into topology, size and shape optimization. The objective of the optimization can be to minimize the stresses weight or compliance for a given amount of material and boundary conditions. A wide range of algorithms is used to build the optimal neural network structure. The first of these algorithms is the tiled constructing algorithm. The idea of the algorithm is to add new layers of neurons in a way that input training vectors that have different respective initial values, would have a different internal representation in the algorithm. Another prominent representative is the fast superstructure algorithm [12]. According to this algorithm new neurons are added between the output layers. The role of these neurons is the correction of the output neurons error. In general, a neural network that is based on this algorithm has the form of a binary tree.

2.4 Research literature

A neural network is a series of algorithms that endeavors to recognize underlying relationships in a set of data through a process that mimics the way the human brain operates. In this sense, neural networks refer to systems of neurons, either organic or artificial in nature [13].
The preliminary theoretical base for contemporary neural networks was independently proposed by Bain, [14] and Brush, [15]. In their work, both thoughts and body activity resulted from interactions among neurons within the brain.

Deep neural networks can recognize voice commands, identify voices, recognize sounds and graphics and do much more than a neural network. Deep learning networks utilize "Big Data" along with algorithms in order to solve a problem, and these deep neural networks can solve problems with limited or no human input.

For Bain, every activity led to the firing of a certain set of neurons. When activities were repeated, the connections between those neurons strengthened. According to his theory, this repetition was what led to the formation of memory. The general scientific community at the time was skeptical of Bain's theory because it required what appeared to be an inordinate number of neural connections within the brain. It is now apparent that the brain is exceedingly complex and that the same brain “wiring” can handle multiple problems and inputs.

James's theory was similar to Bain's, however, he suggested that memories and actions resulted from electrical currents flowing among the neurons in the brain. His model, by focusing on the flow of electrical currents, did not require individual neural connections for each memory or action.

Sherrington, conducted experiments to test James's theory. He ran electrical currents down the spinal cords of rats. However, instead of demonstrating an increase in electrical current as projected by James, Sherrington found that the electrical current strength decreased as the testing continued over time. Importantly, this work led to the discovery of the concept of habituation [16].

Li et al., after surveying the reasons of overfitting and unsuitable generalization of neural networks, by applying the changes in neural network and using a class of RBF structuring delay of neural networks succeeded to create the neural network with high accuracy and with the lower number of neurons in the hidden layer of neural network and tested its results in the actual world [17].
Lahmiri, by applying Discrete Wavelet Transform (DWT) and the division of price time series into two parts of major and minor, concludes that, in fact, the major part has lower frequency and separation, and it is suitable for long-term prediction. After noted transformations, using feedback neural networks, he considered the prediction of the stock price, and by investigating his theory in 15 databases, he concludes that his proposed algorithm has better performance than RW and ARMA models [18].

Ticknor, by considering daily prices and technical analysis indicators as the input of the neural network, predicts the total price of the next day. By noting presented complexities in the trend of stock price changes and the problems for its prediction, for preventing overfitting and overtraining, suggests that neural network is controlled by the Bayesian algorithm. For the models with high complexities, some fines should be determined to prevent overfitting and overtraining. He tested the accuracy of his claim in the stocks of Microsoft and Goldman [19].

Kara et al., by noting this issue that the prediction for the changes of stock prices is a considerable problem and in the case of being accurate, it will be so beneficial, he declares that the expansion of mathematical models of this magnitude for reasons of intrinsic complexities of the stock market is difficult. Using technical analysis indicators as input, he investigates the performance of two algorithms of neural network classification and supports vector machines. By comparing the results, he found the performance of neural networks is better and more suitable [20].

Zhang et al., developed an SVM-based method for two-dimensional locating under a homogeneous wall and a circular metal cylinder object. They also attempted to estimate the wall parameters using the same method, which is based on SVM [21].

In the article of Kose & Arslan, for training ANFIS neural network, a method is proposed, which has the capability of updating that is easier and faster than the Gradient method. In this method, which is called chaotic particle swarm intelligence, there is no need for learning rate, and the proposed learning algorithms are a combination of descending Gradient method and Squared difference method with swarm intelligence algorithms such as genetic algorithm and Ant Colony algorithm, etc [22].
In the following article, Ma et al., a combined neural-fuzzy method is presented based on monthly different charge patterns for the prediction of consuming charge of distribution companies. In the following, the mathematical equations related to the calculations, hypotheses, the type of neural network, and applied fuzzy logic and proposed algorithm are presented. This expert system is based on the different status and times by considering decided input data using each one of the patterns [23].

A fully connected deep LSTM network is proposed in Zhu, [24] to recognize action with a framework composed of three LSTM and two feed-forward layers, incorporating the co-occurrence regularization into the loss function, so exploring the conjunctions of discriminative joints and different co-occurrences for several actions. In Li et al., [17], a deep LSTM framework, based on RNN, is proposed to better localize the start and end of action with a regression module, to automatically extract the features. This joint classification–regression RNN considers the sequence frame by frame and does not require a sliding window approach. A hierarchical approach is also presented in Zhang, [25], where they propose three exploration fusion methods based on multilayer LSTM.

Some authors use a combination of CNN and LSTM to extract spatiotemporal information, but differently from our approach by merging the individual scores obtained from the CNN and the LSTM. Also in this case, contrary to the method proposed by us, they consider all the joints of the skeleton, extrapolating also other information of distance and trajectory between the joints and the poses.

Effective utilization from the present data in the signals of neural network trainer is among the most efficient cases on network performance. In the proposed method of Qiu et al., a suitable selection of training signals is made through its separation using the experimental mode method. The results of handled simulations indicate that using the proposed method leads to positive and considerable results [26].

3. Research Method

The considerable dataset is related to Qeshm weather (suitable weather conditions to study our purpose) prediction from 2016 onwards. These datasets have nine features. In this dataset, the
features such as year, solar day, solar month, dry temperature average, etc. is used for the prediction of humidity average and the objective of the research is the estimation of pollution rate of CO₂. In this research, using an analysis algorithm of major data components, present space to a new space is conveyed, and the dependence among the dimensions is lost. The accuracy in the estimation process is increased. In this study, Convolution neural network is used for the estimation of the pollution rate of CO₂ and for improving the accuracy of the Convolution neural network, optimization algorithm based on training and learning in the training process is used to determine the weights and bias of Convolution neural network in optimized mode. Dimensional reduction is a process carried out to simplify the existing variables to be fewer without losing the information contained in the initial data. One of the methods used in dimension reduction is Principal Component Analysis (PCA). The workings of PCA is to change the initial variable as many as n variables are reduced to k new variables called Principal Component (PC). Assuming that Convolution neural network includes three inputs in the input layer, four neurons in the hidden layer and one neuron in the output layer, the process of neural network training by optimization algorithm based on training and learning includes two major stages including Primary preparation stage and repetition stage, which are explained in the following with the details for neural network training (Figure 1).

The stage of primary preparation in the optimization algorithm based on training and learning

In this stage, a population of the students is made in which each student is made from two parts of decision-making (status) variables and objective function. Decision variables are considered for each student of neural network weights, and error squared mean considered for training data as the objective function. (Figure 2) shows the status of a student, which is, in fact, the weights of the Convolution neural network.

In this stage, the status of all students is randomizing the value, and the squared mean of neural network error is calculated for each member of the population.

Repetition stage in the optimization algorithm based on training and learning

In this stage, the following operations are repeated to fulfill the finishing situation.

Mean calculation
In this step, the status of students is averaged, including the values of weights of the Convolution neural network.

Recognition of teacher

In this step, the best student is selected in terms of objective function value as the teacher. In fact, the teacher is a member of the students' population that has the least amount of error squared mean.

Training phase (teacher phase)

In this step, the new status of each student is calculated based on current situations of the student, teacher, and mean. In fact, the teacher causes the values of weights in the Convolution neural network to change. (Figure 3) indicates the status of the student before and after the teacher phase. The new status of the student is evaluated. It means training inputs are applied to it, and training outputs are calculated and squared mean error is calculated and is considered as a new objective function value of the student. If the error squared mean of the student in new status is better than the previous status, it is stored, and otherwise, its storage is rejected.

Student phase

In this step, the new status of each student (At any level and base), which is randomly selected based on current situations of student and another student, is calculated. In fact, the student phase causes the values of weights in the Convolution neural network to change. (Figure 4) Student status before and after the learning phase. The new status of the student is evaluated. It means training inputs are applied to it, and training outputs are calculated and squared mean error is calculated and is considered as a new objective function value of the student. If the squared mean error of the student in new status is better than the previous status, it is stored, otherwise, its storage is rejected.

The above operation will be repeated to fulfill the finishing situation. The output of the optimization algorithm is based on the training and learning of students, which has the best values of weights in the Convolution neural network, so that squared mean error is the least value [27].
4. Data Analysis

The considered dataset is related to weather prediction of the country from 2016 onwards. These datasets have nine features. The following (Table 1) is a part of the significant dataset in which the range of data changes in each column is different. In this dataset, we used the features such as year, solar day, solar month, dry temperature average, etc. for the prediction of humidity average. A general explanation of this dataset is displayed in the following table.

4.1 Prediction of maximum humidity

First, an LSTM model with 52 blocks was applied for predicting the maximum humidity. Obtained results by LSTM model on proposed data for the prediction of the maximum humidity are as follows:

The following (Table 2) shows the RMSE value for training and test data:

4.2 Combined model

In the combined proposed approach, the Convolution network output is given to an LSTM network.

A Convolution layer with 128 4*4 kernel
A-Max pooling layer with 2*2 kernel
A Convolution layer with 128 2*2 kernel
A Max pooling layer with 2*2 kernel
An LSTM network with 10 LSTM blocks
Dropout=0.5
A layer connected with ten neurons
Last layer with one neuron for prediction

The structure of considerable CNNLSTM model is as follows (Figure 5):
Obtained results by CNNLSTM model on proposed data are as follows (Figure 6):

The following (Table 3) indicates the RMSE amount of CNNLSTM model for training and test data.

Following diagram indicates the prediction results of CNNLSTM model on training and test data (Figure 7):

As observed GRU model towards two other models obtained lower RMSE amount on considerable dataset. On the other hand, the Bidirectional LSTM model against LSTM obtains a lower RMSE amount (Figure 8).

**4.3 Evaluation of model efficiency**

For evaluating model efficiency, we applied two criteria of mean absolute percentage error and Cosine proximity. The following (Table 4) indicates the results of proposed models based on these two criteria:

The amount of cosine proximity is so small that it will be similar for each model, but in terms of map, the GRU model has a lower amount towards other models. The results of models for the prediction of maximum temperature are as follows (Table 5):

For evaluating model efficiency, we applied two criteria of mean absolute percentage error and Cosine proximity. The following (Table 6) indicates the results of proposed models based on these two criteria:

**4.4 The implementation of maximum temperature prediction**

In the next part, the results of each model for the maximum temperature prediction is brought. The following (Table 7) shows RMSE amount of different models for training and test data:

5. **Proposed Method**

5.1 **Recurrent neural networks**
The present study considers decreasing prediction error for the types of time series and the uncertainty in estimation parameters, improving the structure of the deep neural network and increasing response speed in the proposed neural network method. Recurrent neural networks are a type of forwarding network which adds time concept to the model, which is defined through edges in adjacent steps. In these types of networks, in each step, the status is related to current input and the previous step. If \( X_t \) is current input and \( h^t \) is the previous status of the network which was caught from the hidden node of the network, network output is calculated through the following equation: following equation (1):

\[
y' = \text{Soft} \max(w^y h^t + b^y)
\]  

(1)

And \( h^t \) is calculated through the following equation (2):

\[
h^t = \delta(w^{hx} x^t + w^{hh} h^{(t-1)} + b^h)
\]  

(2)

In these equations, \( w^{yh}, w^{hh}, \) and \( w^{hx} \) are learnable weight matrices, and \( b^y \) and \( b^h \) are bias values that allow each neuron to learn the offsets. In these networks, in the time of back propagation over a long time, the gradient will disappear. LSTM networks are a solution to prevent this problem.

For the improvement of LSTM networks, two following methods are used:

**Bidirectional LSTM networks:** LSTM networks, which are summarized for Long Short Term Memory, are a particular type of recurrent neural network which has the capability of learning long term dependencies. The purpose of designing LSTM networks was to solve the problem of long term dependence. It is important to note that remembering information for long term intervals is default and common behavior of LSTM networks, and their structures are such that they learn very far information, a feature which is hidden in their structure.

**5.2 GRU Networks**

GRU architecture or Gated Recurrent Unit is presented to solve the deficiencies of traditional recurrent neural networks such as gradient fading and also the decrease of present overhead in LSTM architecture. GRU is generally considered as a changed version of LSTM because both these architectures use the same design, and they obtain equally excellent results. This type of
architecture uses the concepts as the name of the Update gate and Reset gate. These two are generally gate of two vectors that by using them, the decision is made as to whether the information is conveyed to the output or not. The unique point about these gates is that these gates can be trained until the information related to past steps is kept without any change during the different time steps.

Recurrent networks act with GRU cells, such as the LSTM mechanism, and the only difference is that instead of three gates, it uses two gates. As a result, we will have relatively good improvement in terms of speed, and its results are almost the same as the LSTM model. Conceptually speaking, a GRU network has two gates of $r_t$ readjust and $Z_t$ forgetting, which helps to assure that long term dependencies of gradient fading will not occur. The following equation (3) obtains the values of these gates:

$$z_t = \sigma(W_z x_t + U_z h_{t-1} + b_z)$$

$$r_t = \sigma(W_r x_t + U_r h_{t-1} + b_r)$$

(3)

The following (Table 8) shows the RMSE value for training and test data:

This model obtained lower RMSE value towards the LSTM model. Also, the results of its prediction are more realistic than the LSTM model on test and training data against the significant data.

5.3 Bidirectional LSTM Networks

Back Two-way neural networks (BLSTM) connect two hidden layers from different directions to the same output. Using this form of relative deep learning, the output layer concurrently can receive the data from the previous modes (back) and future ones (forth). BLSTMs do not require their input data. Also, their input data are available in the previous status. The following (Table 9) shows the RMSE value for training and test data:

6. Discussion and Conclusion
In general, the humidity is considered in two modes of absolute humidity and relative humidity. The absolute humidity is the amount of steam that is present in the unit of air volume. The amount of steam that requires the saturation of a definite volume of air will be more with the increase in temperature. The relative humidity is the present humidity amount in air volume with a specific temperature to the maximum humidity, which that air can have the same temperature. Temperature is a physical and relative quantity that determines the amount of cold and warmth, and it is measurable with the thermometer. If two things have different temperatures, the warmth is conveyed from the item with a higher temperature to the item with a lower temperature until the temperature of two will be balanced.

The estimation of fault impedance and distance values due to the occurrence of a single line to ground fault (SLGF) and balanced fault in the underground distribution system is proposed. A single measurement of three-phase voltage and current waveforms are measured at the main substation. The DWT is used to extract the features from the measured voltage and current waveforms. Then, the correlation between the extracted features of voltage and current waveforms is obtained using the cross-product analysis. Subsequently, the ANN is utilized to estimate the fault impedance and distance values. A thorough investigation is carried out to analyze the effect of different types of ANN learning algorithms as well as its ANN parameters such as learning rate (lr), momentum constants (mc) and the number of neurons in a hidden layer.

In the present research, in section 4, using different methods, the maximum prediction of humidity, and the temperature is investigated. The models used were the samples of recurrent neural networks, which among these networks, GRU, back two-way LSTM, and LSTM, and the proposed combined CNN+LST model were used. Among the criteria which were applied for evaluating the models are RMSE, Mse, Mae, Mean Absolute Percentage Error, and Cosine Proximity.

Around, LSTM started to revolutionize speech recognition, outperforming traditional models in certain speech applications. In 2009, a Connectionist Temporal Classification (CTC)-trained LSTM network was the first RNN to win pattern recognition contests when it won several competitions in connected handwriting recognition. In 2014, the Chinese company Baidu used
CTC-trained RNNs to break the 2S09 Switchboard Hub5'00 speech recognition dataset benchmark without using any traditional speech processing methods [28].

LSTM also improved large-vocabulary speech recognition and text-to-speech synthesis and was used in Google Android. In 2015, Google's speech recognition reportedly experienced a dramatic performance jump of 49% through CTC-trained LSTM [29].

The maximum temperature and humidity are predicted using GRU and LSTM. For comparing the obtained values in these predictions, the RMSE method is used, which is known as a sufficient amount. Root Mean Square Error (RMSE) is an index that shows the effective amount of error. In other words, this index indicates the difference between predicted value by model and the actual value of a variable. This instrument for determining the model is selected as the model close to reality.

In the method of GRU networks, instead of three gates, two gates are used for the prediction of the maximum amount of humidity, but a relatively good speed of improvement is observed. Of course, it is almost the difference as the LSTM model, and they are not very different. In general, GRU networks towards other models produce more realistic results, and also back two-way networks produce better results on test data towards LSTM networks.

References


13. Cuntz, H. “Forest of synthetic pyramidal dendrites grown using cajal's laws of neuronal branching”. PLoS Computational Biology. 6(8), ev06.i08-17. (2010). DOI: 10.1371/image.pcbi.v06.i08


Biographies

Hamideh Rezaei Nezhad
E-mail address: rezaei.iauqeshm@gmail.com
https://orcid.org/0000-0002-7726-7171
Hamideh Rezaei Nezhad is currently Ph.D student a member of the academic staff and faculty member of Qeshm Islamic Azad University, I received a bachelor's degree in software engineering from Baft Islamic Azad University and master's degree in software engineering from Qeshm International University, currently I'm a doctoral student in computer major, My main specialty is Estimation and forecasting; Deep learning; Processing complex systems; Data mining; Software engineering and artificial intelligence,.. etc.

Farshid Keynia (Correspondent author)
E-mail address: f.keynia@kgut.ac.ir
https://orcid.org/0000-0002-9027-7315
Farshid Keynia is currently an Department of Energy Management and Optimization, Institute of Science and High Technology and Environmental Sciences, Graduate University of Advanced Technology, Kerman, Iran, He received the B.Sc. degree in Electrical Engineering from S. B. University, Kerman, Iran, in 1996, and the M.Sc. degree in Electrical Engineering from Semnan University, Semnan, Iran, in 2001.
He is received the Ph.D. degree at the Electrical Engineering Department of Semnan University, Semnan, Iran, in 2010.
His researches focus on Electricity Market Features, short-term and mid-term price and load forecasting in deregulated electricity markets, and feature selection and classification algorithms. He also work on meta heuristic multi optimization methods.

Amir Sabbagh Molahosseini
E-mail address: a.sabbaghmolahosseini@qub.ac.uk
https://orcid.org/0000-0002-6024-3202
Amir Sabbagh Mola Hoseini is Amir Sabbagh Molahosseini received a B.Sc. degree from Shahid Bahonar University of Kerman, Iran in 2005, M.Sc. and Ph.D. degrees (Hons.) in computer engineering from the Science and Research Branch of Azad University, Tehran, Iran, in 2007 and 2010, respectively. He is currently with the School of Electronics, Electrical Engineering and Computer Science, Queen's University Belfast, UK. His research interests are Computer Arithmetic/Architecture and Approximate/Transprecise Computing.

Figure 1. Proposed algorithm flowchart for neural network training

Figure 2. Weights of Convolution neural network as student status

Figure 3. New status of the student after teacher phase

Figure 4. New status of the student after the learning phase in the student phase
**Figure 5.** The structure of considerable CNNLSTM model

**Figure 6.** Diagram related to mse and mae values on training data

**Figure 7.** In this model, blue color shows actual data; yellow color indicates model prediction on training data, and green color shows model prediction on test data.

**Figure 8.** The prediction of model humidity with RMSE criterion

**Table 1.** General explanation of considered dataset

**Table 2.** The test and training root mean square error (RMSE)

**Table 3.** RMSE amount of CNNLSTM model for training and test data

**Table 4.** Proposed models based on mean absolute percentage error and Cosine proximity

**Table 5.** Prediction of maximum temperature for training and test data

**Table 6.** Proposed models based on mean absolute percentage error and Cosine proximity

**Table 7.** RMSE amount of different models for training and test data

**Table 8.** The test and training root mean square error (RMSE)

**Table 9.** The test and training root mean square error (RMSE)
Figure 1.
Figure 2.
Figure 3.
Figure 4.

Figure 5.
Figure 8.

Table 1.

<table>
<thead>
<tr>
<th>Solar year</th>
<th>Solar month</th>
<th>Solar day</th>
<th>Dry temperature average celsius</th>
<th>Absolute maximum temperature celsius</th>
<th>Absolute minimum temperature celsius</th>
<th>Humidity average</th>
<th>Maximum humidity</th>
<th>Minimum humidity</th>
</tr>
</thead>
<tbody>
<tr>
<td>1395</td>
<td>1</td>
<td>1</td>
<td>23.2</td>
<td>26.2</td>
<td>20.7</td>
<td>68.5</td>
<td>60</td>
<td>79</td>
</tr>
<tr>
<td>1395</td>
<td>1</td>
<td>2</td>
<td>22.9</td>
<td>27.5</td>
<td>14.8</td>
<td>71.3</td>
<td>62</td>
<td>82</td>
</tr>
<tr>
<td>1395</td>
<td>1</td>
<td>3</td>
<td>21.4</td>
<td>23.4</td>
<td>21.4</td>
<td>81.9</td>
<td>74</td>
<td>87</td>
</tr>
<tr>
<td>1395</td>
<td>1</td>
<td>4</td>
<td>22.5</td>
<td>25.7</td>
<td>18.2</td>
<td>83.6</td>
<td>76</td>
<td>94</td>
</tr>
<tr>
<td>1395</td>
<td>1</td>
<td>5</td>
<td>22.6</td>
<td>26.5</td>
<td>19.0</td>
<td>65.9</td>
<td>59</td>
<td>78</td>
</tr>
<tr>
<td>1395</td>
<td>1</td>
<td>6</td>
<td>22.6</td>
<td>28.2</td>
<td>17.4</td>
<td>69.3</td>
<td>51</td>
<td>80</td>
</tr>
<tr>
<td>1395</td>
<td>1</td>
<td>7</td>
<td>22.1</td>
<td>27.6</td>
<td>15.6</td>
<td>72.3</td>
<td>60</td>
<td>82</td>
</tr>
<tr>
<td>1395</td>
<td>1</td>
<td>8</td>
<td>22.8</td>
<td>30.1</td>
<td>15.6</td>
<td>66.3</td>
<td>37</td>
<td>87</td>
</tr>
</tbody>
</table>
Table 2.

<table>
<thead>
<tr>
<th>RMSE</th>
<th>Train</th>
<th>Test</th>
</tr>
</thead>
<tbody>
<tr>
<td>9.42</td>
<td>8.64</td>
<td></td>
</tr>
</tbody>
</table>

Table 3.

<table>
<thead>
<tr>
<th>RMSE</th>
<th>Train</th>
<th>Test</th>
</tr>
</thead>
<tbody>
<tr>
<td>13.35</td>
<td>14.94</td>
<td></td>
</tr>
</tbody>
</table>

Table 4.

<table>
<thead>
<tr>
<th>Model</th>
<th>Mean absolute percentage error</th>
<th>Cosin proximity</th>
</tr>
</thead>
<tbody>
<tr>
<td>LSTM</td>
<td>24.8340</td>
<td>-1.0000e+00</td>
</tr>
<tr>
<td>Bidirectional LSTM</td>
<td>21.6581</td>
<td>-1.0000e+00</td>
</tr>
<tr>
<td>GRU</td>
<td>21.4872</td>
<td>-1.0000e+00</td>
</tr>
<tr>
<td>CNN+LSMT</td>
<td>63.12</td>
<td>-1.0000e+00</td>
</tr>
</tbody>
</table>

Table 5.

<table>
<thead>
<tr>
<th>Model</th>
<th>Train</th>
<th>Test</th>
</tr>
</thead>
<tbody>
<tr>
<td>LSTM</td>
<td>2.31</td>
<td>2.31</td>
</tr>
<tr>
<td>Bidirectional LSTM</td>
<td>2.12</td>
<td>2.17</td>
</tr>
<tr>
<td>GRU</td>
<td>1.98</td>
<td>2.08</td>
</tr>
<tr>
<td>CNN+LSMT</td>
<td>1.93</td>
<td>2.11</td>
</tr>
</tbody>
</table>
### Table 6.

<table>
<thead>
<tr>
<th>Model</th>
<th>Mean absolute percentage error</th>
<th>Consine proximity</th>
</tr>
</thead>
<tbody>
<tr>
<td>LSTM</td>
<td>462428.4772</td>
<td>-9.9846e⁻⁰¹</td>
</tr>
<tr>
<td>Bidirectional LSTM</td>
<td>452009.3791</td>
<td>-9.9846e⁻⁰¹</td>
</tr>
<tr>
<td>GRU</td>
<td>450656.1325</td>
<td>-9.9846e⁻⁰¹</td>
</tr>
<tr>
<td>CNN+LSMT</td>
<td>632595.9525</td>
<td>-9.9846e⁻⁰¹</td>
</tr>
</tbody>
</table>

### Table 7.

<table>
<thead>
<tr>
<th>Model</th>
<th>Train</th>
<th>Test</th>
</tr>
</thead>
<tbody>
<tr>
<td>LSTM</td>
<td>2.31</td>
<td>2.31</td>
</tr>
<tr>
<td>Bidirectional LSTM</td>
<td>2.12</td>
<td>2.17</td>
</tr>
<tr>
<td>GRU</td>
<td>1.98</td>
<td>2.08</td>
</tr>
<tr>
<td>CNN+LSMT</td>
<td>1.93</td>
<td>2.11</td>
</tr>
</tbody>
</table>

### Table 8.

<table>
<thead>
<tr>
<th>RMSE</th>
<th>Train</th>
<th>Test</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>8.62</td>
<td>7.84</td>
</tr>
</tbody>
</table>

### Table 9.

<table>
<thead>
<tr>
<th>RMSE</th>
<th>Train</th>
<th>Test</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>9.44</td>
<td>8.59</td>
</tr>
</tbody>
</table>