Check for
updates

Long-Term Energy Forecasting System Based on LSTM and Deep Extreme Machine Learning

Cherifa Nakkach*, Amira Zrelli and Tahar Ezzedine

Laboratory of Systems Communications Syscom, National Engineering School of Tunis University of Tunis-El Manar, Tunis, 7000, Tunisia
*Corresponding Author: Cherifa Nakkach. Email: cherifa.nakkach@enit.utm.tn

Abstract: Due to the development of diversified and flexible building energy resources, the balancing energy supply and demand especially in smart buildings caused an increasing problem. Energy forecasting is necessary to address building energy issues and comfort challenges that drive urbanization and consequent increases in energy consumption. Recently, their management has a great significance as resources become scarcer and their emissions increase. In this article, we propose an intelligent energy forecasting method based on hybrid deep learning, in which the data collected by the smart home through meters is put into the pre-evaluation step. Next, the refined data is the input of a Long Short-Term Memory (LSTM) network, which captures the spatio-temporal correlations from the sequence and generates the feature maps. The output feature map is passed into a Deep Extreme Machine Learning network (with seven hidden layers) for learning, which provides the final prediction. Compared to existing techniques, the LSTM-DELM model offers better prediction results. The simulation values demonstrate the superior performance of the proposed model.

1 Introduction

Building sector is considered as one of the huge energy consuming entities nowadays, accounting for about 40% of international energy consumption [1]. Indeed, energy consumption studies problems have become a central aim of studies in recent decades. As energy is one of the highly important resources for industrial production, anticipating energy consumption is an important task for macro planning of industrial and energy sectors [2]. Energy efficiency has become a challenge of high priority and importance achieving a cleaner and more sustainable way of life. As it is a massively important resource, its demand is increasing daily. Therefore, saving energy is not just important to promote a green atmosphere for future sustainability but also necessary for household consumers and power generation companies. Predicting the energy consumption for buildings is a great way for building managers to choose all types of equipment wisely. Moreover, this is an efficient and crucial way to

avoid wasting building energy and improve energy efficiency. As a sub-area of energy consumption, building energy consumption accounts for a significant proportion [2]. Researchers have developed several powerful simulation tools, such as Energy Plus [3], DEST [4], and eQuest [5], relying on the fundamental concepts of the thermodynamic equilibrium equation and the heat equation. Hence, these tools can accurately compute the building energy consumption with exact boundary conditions and offer important insights into energy consumption models with physical meanings. Still, there are drawbacks related to these various tools. They react exceptionally to boundary conditions that are based on expertise. Furthermore, the simulations take a lot of time, which refutes the main idea of rapid and precise computing in terms of big data. As data mining transforms several disciplines, data-driven approaches for predicting building energy consumption have become mainstream [6]. Generally, smart homes have different and huge energy necessitates managing all modules and present new advances in an organized manner by making optimal application of all these energy sources. Thus, an efficient predictive concept for energy management in a smart home is needed. Moreover, due to the increase in data and processing power of computers, machine learning and deep learning have been applied in different fields [7] such as construction [8], security [9–11], medical [12], economics [13] and image recognition [14]. The recent efficiencies of deep learning in multiple fields have attracted scientists to monitor energy consumption, and its promising potential is demonstrated by the plethora of proposed models and the increasing diversity of research. In this context, deep learning plays an important role in the domain of energy consumption. It offers many accurate models to achieve this goal. Deep learning methods can predict both linear and nonlinear correlations, making it a useful choice for energy prediction. Deep neural networks (DNN) improve forecasting accuracy and reduce operating costs by reducing the number of errors. In forecasting research, DNN have been considered widely for their performance in solving nonlinear relationship problems.

Hence, this article aims to propose a hybrid model for reducing energy consumption and ensuring its efficient use. The major contributions of the proposed model are underlined as follows:

1) An acquisition and refinement layer is employed, which refines the data by substituting past values, normalizing, and organizing the data into a continuous windowing sequence.
2) A LSTM network is designed for energy forecasting that extracts the most discriminating spatial-temporal features from the power load sequence and makes a block of feature map.
3) DELM network has been applied for sequence learning, which obtains the spatiotemporal features (feature map) from the LSTM network. The DELM network is adequate for learning the sequence patterns and provides an effective energy prediction.

We prove through experiments that the proposed model shows excellent results and outperforms the state of the art by recording the lowest value of MSE and RMSE on the most demanding data set. It reveals the suitability of the method to effectively manage the energy infrastructure and ensure the saving of large amounts of energy waste. By applying our proposed method, we achieved accurate and quick calculation results. The objective is to provide maximum user comfort at the cost of minimum energy consumption. With regard to the rest of the paper, Section 2 discusses related works, which treat models used for energy prediction. Section 3 provides an overview of the proposed system of energy forecasting. Section 4 presents the performance evaluation of the model proposed and discusses the main results. Finally, Section 5 concludes the paper.

2 Related Work

A detailed overview of the most important methods for optimizing and predicting energy consumption was carried out in [15]. In [16], a comparison between PSOANN and a simple ANN

with GA-ANN succeeded (GA was implemented for the same goal as PSO). Two databases have been applied to predict hourly electricity consumption. An optimization strategy was developed in [17]. It was called teach-learn based optimization. It employs evolutionary algorithms with BPNN to improve convergence speed and prediction performance. A basic aggregated model was designed along with five improved versions. A short-term energy consumption prediction model was endowed by [18]. It used a multilayer perceptron (MLP), a scaled conjugate gradient and Levenberg-Marquardt backpropagation algorithms. An energy forecasting technique for smart homes proposed in [19]. They introduced artificial neural networks to forecast the energy consumption in different time intervals daily or weekly. The dynamic neural network and the Energy Plus program were useful in predicting the building energy load and the Taguchi method to measure the effect of parameters on the load [20]. In [21], the authors presented a comparison of ANN, SVM and ARIMA models to forecast heating and cooling energy consumption in Chinese office buildings in Shanghai, Nanjing and Changsha. All models were evolved with IBM SPSS Modeller. The results showed that the ANN is better than the SVM and that both outperformed the ARIMA with a MAPE of 0.15%–0.11percentage, 0.21%–0.18% and 0.41%–0.33%, respectively (test samples in the first and second month). Another effective hybrid approach in [22] used an autoregressive integrated moving average (ARIMA) to forecast energy consumption. An IoT-based home energy management method for energy consumption optimization of electronic devices in smart homes in [23] used a hybrid artificial neural network and particle swarm optimization algorithms. The model presented the early design of energy efficient buildings. Besides simple neural networks, other algorithms for energy consumption prediction have also been successful, such as the extreme learning machine-based model in [24], the Bat algorithm, and fuzzy logic control to improve energy consumption utilizing temperature, lighting, and air quality in [25]. Through efficient prediction, various approaches provide rough directions and features for energy consumption prediction, such as weather information and power data. Due to their disorganized data usage, these methods are also less effective in predicting. The literature discussed above indicates that most prediction methods rely on statistical features and utilize machine learning techniques to predict energy consumption. Due to its high accuracy and performance in energy and electric systems, we introduce a hybrid deep learning model based on LSTM and DELM to overcome several challenges. As a result of weather conditions and occupant behavior, the collected energy consumption data can contain abnormalities and redundancies, leading to incorrect energy predictions. So, consumer and supplier power management fails. Resolving this issue requires removing noise and handling missing values. When it comes to smart grid planning and electricity marketing, statistical machine learning techniques like linear regression and clustering have limited effectiveness. Several LSTM layers are added for feature extraction. In this paper, we propose a method that predicts power consumption using the LSTM-DELM sequential learning algorithm. By combining forward and backward interpretations at the end, the LSTM-DELM learns the input sequence, and then predicts output power by concatenating them. Experiments have also shown that the proposed method is superior to existing approaches. By using our method, MSE and RMSE are the lowest values obtained, proving that it predicts accurately energy consumption.

3 Proposed System

In this section, the novel hybrid model applied to predict energy is detailed. First, we present the models employed in the feature extraction and learning task. Then, a great concise description of our own system is developed. . In this study, we have followed these steps in order to obtain results of prediction of energy as presented in Fig. 1.
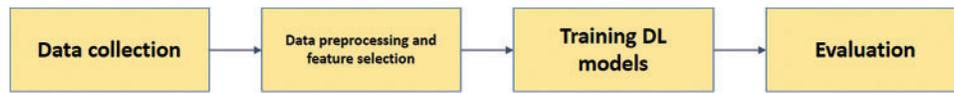
Figure 1: Methodology of prediction

The proposed energy prediction system is described in Fig. 2. It involves of three steps:

1. Step1: For data management and energy usage response, the first step is to collect data from sensors and smart meters. Further refinement is performed so that user behavior can be stabilized and removed.
2. Step 2: LSTM is used to extract features from the energy data.
3. Step 3: This step involves predicting future energy consumption using the input data.
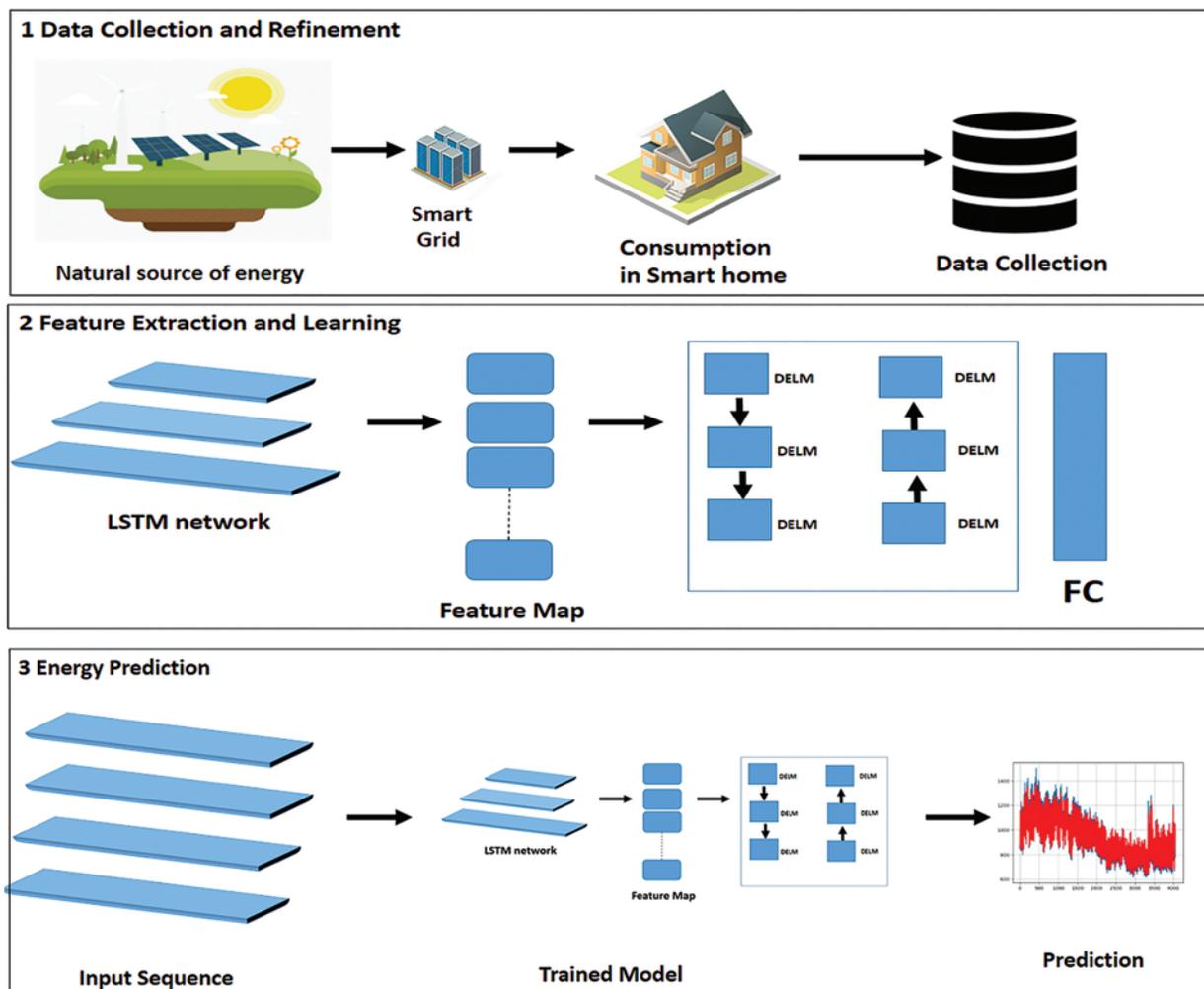
Figure 2: The proposed energy prediction system

3.1 Data Collection and Refinement

This part provides an explanation of data collection from sources such as gauges and installed sensors. Data pre-processing is also explained. To collect data, wires are run across the building floors in a single edge with the motherboard, and the meter with few sensors is installed to read and measure energy usage above the building structure, where the data is usually collected at minute resolutions. Normally, climate conditions affect the data collected by sensors and meters, occupant behavior, redundancy, broken wire or short circuit, leading to noise in the variable values. Solving this problem is crucial for accurate forecasts; Therefore, we refine the data before the actual processing.

3.2 Feature Extraction and Learning
3.2.1 Sequence Modeling Through LSTM

As a dimensionality reduction technique, feature selection can be useful for reducing the dimensions of time series meteorological datasets [26]. By providing only relevant features as input to the prediction model, the feature selection improves the capability and reduces the complexity [27]. To model the sequential data, a form of recurrent neural networks (RNNs) such as LSTM was chosen as it is the most stable and powerful network that understands and processes the long-range information. LSTM has the ability to learn long-term sequence information.

This section defines the internal design of LSTM in sequence analysis. LSTM is a complex version of RNN, with more layers inside. It proposes special units called memory blocks in the recurring layer to surmount vanishing/exploding gradient problems. Fig. 3 presents the structure of a memory block. Memory cells are integrated in the self-connected memory blocks. Three special multiplicative units, called gates, are introduced to store temporal sequences. The three gates have these functions as follows:

Input gate: This gate controls the information flow into the memory cell.

Output gate: This gate controls the information flow into the remainder of the networks.

Forget gate: This gate selects the cell state at the prior moment and retains a portion of the information into the current moment.
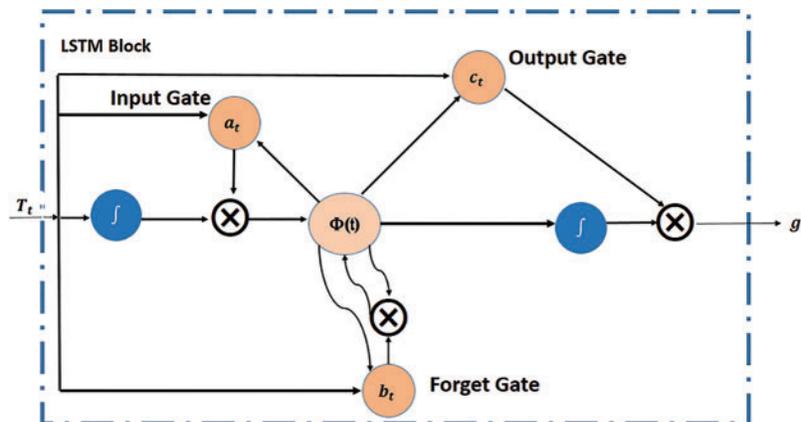
Figure 3: Structure of a memory block

These gates are equal to multiplying the previous information by a number ranging from 0 to 1. If the number is zero, all previous information flow is discarded. If the number is 1, the entire flow of information is kept. The three gates all use sigmoid functions that make the data limited to the range of 0–1. The sigmoid function is presented as:

$$\omega(x) = \frac{1}{1 + e^{-x}} \tag{1}$$

a_t, b_t, c_t are respectively the input gate, forget gate, and output gate, ϕ_t is the intermediate value during when calculating,

$$a_t = \varphi(S_a T_t + V_a g_{t-1} + bi_t) \tag{2}$$

$$\phi_t = tanh(S_z T_t + V_z gt - 1 + bi_z) \tag{3}$$

$$b_t = \varphi(S_b T_t + V_b g_{t-1} + bi_b) \tag{4}$$

$$\phi_t = a_t * \phi_t + b_t * \phi_{t-1} \tag{5}$$

$$c_t = \varphi(S_c T_t + V_c g_{t-1} + bi_c) \tag{6}$$

$$g_t = c_t * tanh(\phi_t) \tag{7}$$

where S_i, V_i, S_z, V_z, B_b, V_b, S_c and V_c are weight matrices, bi_t, b_z, bi_b and bi_c are bias vectors. T_t is the current input. g_t and g_{t-1} are the outputs at the current time t and the previous time t − 1, respectively. The hyperbolic tangent function is presented as follows:

$$tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \tag{8}$$

3.2.2 Deep Extreme Machine Learning Network

Deep extreme learning networks receive the features obtained from these LSTM layers. It is a variation of the Extreme Learning Machine (ELM). It has been adopted in many domains. The traditional algorithm based on artificial neural networks necessitates more training samples; slower learning times, and can lead to overfitting of a learning model [28]. Deep ELM, as presented in Fig. 4 is one of the latest and highly important iterations in the ELM evolution. DELMs are based on the concept of MLELMs with the application of KELM as the output layer [29]. The number of hidden layers relies on the problem under study. Here we adopted seven hidden layers in the proposed model. We used trial and error to choose the number of nodes in the hidden layers. We adopted the process described by [30], for the calculation method of hidden layers of the Deep Extreme Learning Machine (DELM) network.

The concept of ELM was first specified in [31]. The ELM is widely adopted in many fields for classification and regression purposes because an ELM learns very rapidly and it is computationally efficient. The ELM model is presented in Fig. 5. It includes an input layer, a single hidden layer, and an output layer.
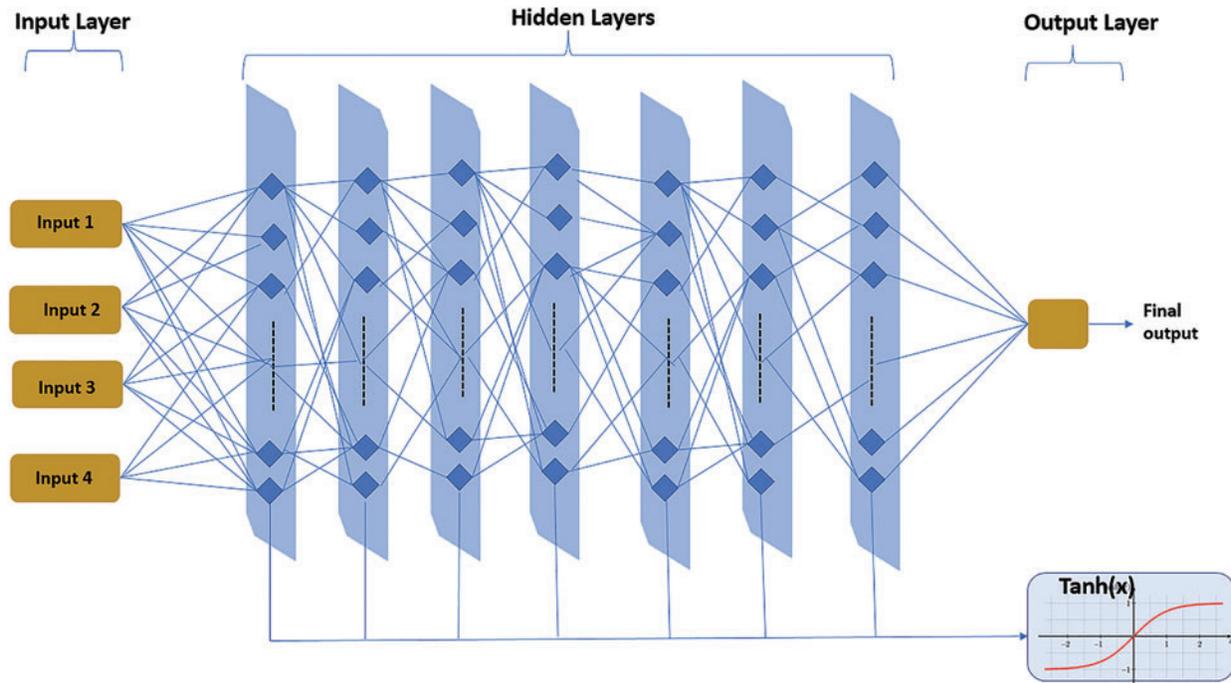
Figure 4: The DELM model
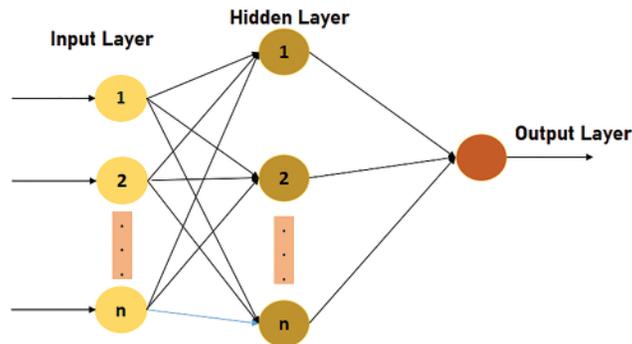
Figure 5: The ELM structure

Where p represents input layer nodes, q represents hidden layer nodes, and r indicates output layer nodes. First of all, consider $\{E, F\} = \{E_i, F_i\}$ where (i = 1, 2, 3 … p), $E = [e_{i1}, e_{i2}, e_{i3}, \ldots, e_{ip}]$ is the output feature and the matrix $F = [f_{i1}, f_{i2}, f_{i3}, \ldots, f_{ip}]$. The matrix E and F are presented as Eqs. (9) and (10):

$$E = \begin{bmatrix} e_{11} & e_{12} & .. & e_{1p} \\ e_{21} & e_{22} & .. & e_{2p} \\ e_{31} & e_{32} & .. & e_{3p} \\ e_{41} & e_{42} & .. & e_{4p} \\ . & . & . & . \\ . & . & . & . \\ e_{q1} & e_{q2} & .. & e_{qp} \end{bmatrix} \tag{9}$$

$$F = \begin{bmatrix} f_{11} & f_{12} & .. & f_{1p} \\ f_{21} & f_{22} & .. & f_{2p} \\ f_{31} & f_{32} & .. & f_{3p} \\ f_{41} & f_{42} & .. & f_{4p} \\ . & . & . & . \\ . & . & . & . \\ . & . & . & . \\ f_{k1} & f_{k2} & .. & f_{kp} \end{bmatrix} \tag{10}$$

where q and k indicate the output and input matrix dimensions, respectively. The ELMs randomly assign weights between input layers and the hidden layer. The weights t are represented among the ith neurons of the input layer and jth neurons output hidden layer. ELM considers weights between the output layer and the hidden layer of neurons, which is expressed in Eq. (11):

$$T = \begin{bmatrix} T_{11} & T_{12} & .. & T_{1q} \\ T_{21} & T_{22} & .. & T_{2q} \\ T_{31} & T_{32} & .. & T_{3q} \\ T_{41} & T_{42} & .. & T_{4q} \\ . & . & . & . \\ . & . & . & . \\ T_{l1} & T_{l2} & .. & T_{lq} \end{bmatrix} \tag{11}$$

$$\Omega = \begin{bmatrix} \Omega_{1_1} & \Omega_{1_2} & .. & \Omega_{1_k} \\ \Omega_{2_1} & \Omega_{2_2} & .. & T_{2k} \\ \Omega_{3_1} & \Omega_{3_2} & .. & \Omega_{3_k} \\ \Omega_{4_1} & \Omega_{4_2} & .. & \Omega_{4_k} \\ . & . & . & . \\ . & . & . & . \\ \Omega_{q_1} & \Omega_{q_2} & .. & \Omega_{q_k} \end{bmatrix} \tag{12}$$

The ELM randomly sets the bias of the hidden layer as Eq. (13):

$$F = \begin{bmatrix} f_1, & f_2, & f_3, & \ldots, & f_q \end{bmatrix}^T \tag{13}$$

$$S = \begin{bmatrix} s_1, & s_2, & s_3, & \ldots, & s_Q \end{bmatrix}_{k*p} \tag{14}$$

Each column of the matrix S is presented as follows:

$$s = \begin{bmatrix} s_{1j} \\ s_{2j} \\ s_{3j} \\ . \\ . \\ . \\ s_{kj} \end{bmatrix} = \begin{bmatrix} \sum_i^q \Omega_{r1} g\left(t_r a_1 + f_r\right) \\ \sum_i^q \Omega_{r2} g\left(t_r a_1 + f_r\right) \\ \sum_i^q \Omega_{r3} g\left(t_r a_1 + f_r\right) \\ . \\ . \\ . \\ \sum_i^q \Omega_{rk} g\left(t_r a_1 + f_r\right) \end{bmatrix}_{i=1,2,3,\ldots,Q} \tag{15}$$

Transpose of the matrix T is denoted by μ:

$$\mu = \begin{bmatrix} s_1, & s_2, & s_3, & \ldots, & s_Q \end{bmatrix}_T \tag{16}$$

Taking Eqs. (6) and (7) into account, we obtain (9):

$$D\Omega = \mu \tag{17}$$

$$\Omega = D^+\mu \tag{18}$$

The regularization term Ω is used to make the network more generalized and stable. With the combination of the benefits of ELMs and deep learning (DL), we obtain the second layer as expressed in Eq. (11):

$$D_1 = S\Omega \tag{19}$$

where Ω^+ the generalized inverse of matrix Ω. The output of hidden layer 2 is obtained using Eq. (18) and the activation function inverse.

$$h(T_1 S + F_1) = D_1 \tag{20}$$

In Eq. (20), the variables T_1 S, F_1 and D_1 are weight matrices of the initial 2 hidden layers, the bias of the neurons of the first hidden layer, the output of the initial hidden layer, and the projected output of the 2nd hidden layer.

$$T_{DE} = h^{-1}(D_1)D_E^+ \tag{21}$$

D_E^+ is the inverse of D_E, and the activation function h(x) is used in Eq. (22). Therefore, given any suitable activation function h(x), the output of the second hidden layer will be as follows:

$$D_2 = H(T_{DE}D_E) \tag{22}$$

The update of the weight matrix Ω among the second and third hidden layers is performed in Eq. (23), where D_2^+ represents the inverse of D_2. so, the predicted value of the third layer is defined as:

$$\Omega_{new} = D_2^+ S \tag{23}$$

$$D_3 = S\Omega_{new}^+ \tag{24}$$

where Ω_{new}^+ new represents the inverse of the weight matrix Ω_{new}. Then, the DELM represents the matrix $T_{DE} = [F_2, T_2]$. The output of the third layer is obtained using Eqs. (9)–(10).

$$D_3 = h^{-1}(D_2 T_2 + F_2) = h(T_{DE1}D_{E1}) \tag{25}$$

$$T_{DE1} = \Omega^{-1}(D_3)D_{E1}^+ \tag{26}$$

where D_2 is the anticipated output of hidden layer 2, T_2 represents the weight between the second and third hidden layers, and F_2 is the bias of the neurons of the third hidden. D_{E1}^+ is the inverse of D_{E1}, and h^{-1} represents the inverse of the activation function h(x). The logistic sigmoid function in Eq. (27) is used, and the third hidden layer output is obtained as follows:

$$g(x) = \frac{1}{1 + e^{-x}} \tag{27}$$

$$D_3 = h(T_{DE}D_{E1}) \tag{28}$$

The resultant weight matrix between the third hidden layer and the output layer is represented in Eq. (29). The projected output of the third hidden layer is expressed as follows:

$$\Omega_{new} = D_4^T \left(\frac{1}{\beta} + D_4^T D_4\right)^{-1} S \tag{29}$$

$$D_4 = S\Omega_{new}^+ \tag{30}$$

Ω_{new}^{+} is the inverse of the weight matrix Ω_{new}. Here, the DELM represents the matrix $T_{DE} = [F_2, T_2]$. The output of the fourth layer is calculated using Eqs. (31)–(32).

$$D_4 = h^{-1}(D_3 T_3 + F_3) = h(T_{DE2} D_{E2}) \tag{31}$$

$$T_{DE2} = \Omega^{-1}(D_4) D_{E2}^{+} \tag{32}$$

In Eq. (28), D_3 is the predicted output of the third hidden layer, the weights between the third and fourth hidden layers are represented as T_3 and F_3 is the bias of the neurons of the third hidden layer. D_{E1}^{+} is the inverse of D_{E1} and h^{-1} is the inverse of the activation function h(x). The logistic sigmoid function is used. The output of the third and fourth layers is calculated as Eq. (33):

$$D_4 = h(T_{DE2} D_{E2}) \tag{33}$$

Now, the output weight matrix among the fourth and the fifth layer is determined using Eq. (34). The projected output of the fifth layer is presented in Eq. (35). The predicted output of the DELM network is defined in Eq. (36).

$$\Omega_{new} = D_5^{T} \left(\frac{1}{\beta} + D_5^{T} D_5 \right)^{-1} S \tag{34}$$

$$D_5 = S \Omega_{new}^{+} \tag{35}$$

Ω_{new}^{+} is the inverse of the weight matrix Ω_{new}. Here, the matrix TDE3 = [F3, T3] is presented. The output of the fifth layer is calculated using Eqs. (36)–(37)

$$D_5 = h^{-1}(D_4 T_4 + F_4) = h(T_{DE3} D_{E3}) \tag{36}$$

$$T_{DE3} = \Omega^{-1}(D_3) D_{E3}^{+} \tag{37}$$

The output of the fourth and fifth layers is calculated as Eq. (38)

$$D_5 = h(T_{DE3} D_{E3}) \tag{38}$$

Here, the output weight matrix among the fifth and the sixth layer is determined using Eq. (39). The projected output of the sixth layer is presented in Eq. (40). The predicted output of the DELM network is defined in Eq. (41).

$$\Omega_{new} = D_6^{T} \left(\frac{1}{\beta} + D_5^{T} D_5 \right)^{-1} S \tag{39}$$

$$D_6 = S \Omega_{new}^{+} \tag{40}$$

$$g(x) = D_5 \Omega_{new} \tag{41}$$

In this section, we have discussed the calculation process for the seventh hidden layer of the DELM network. To demonstrate the calculation process of DELM, cycle theory was applied. In order to obtain the hidden layer's parameters and ultimately the final result of the DELM network, Eqs. (26)–(38) are utilized. This process is illustrated in Fig. 6.
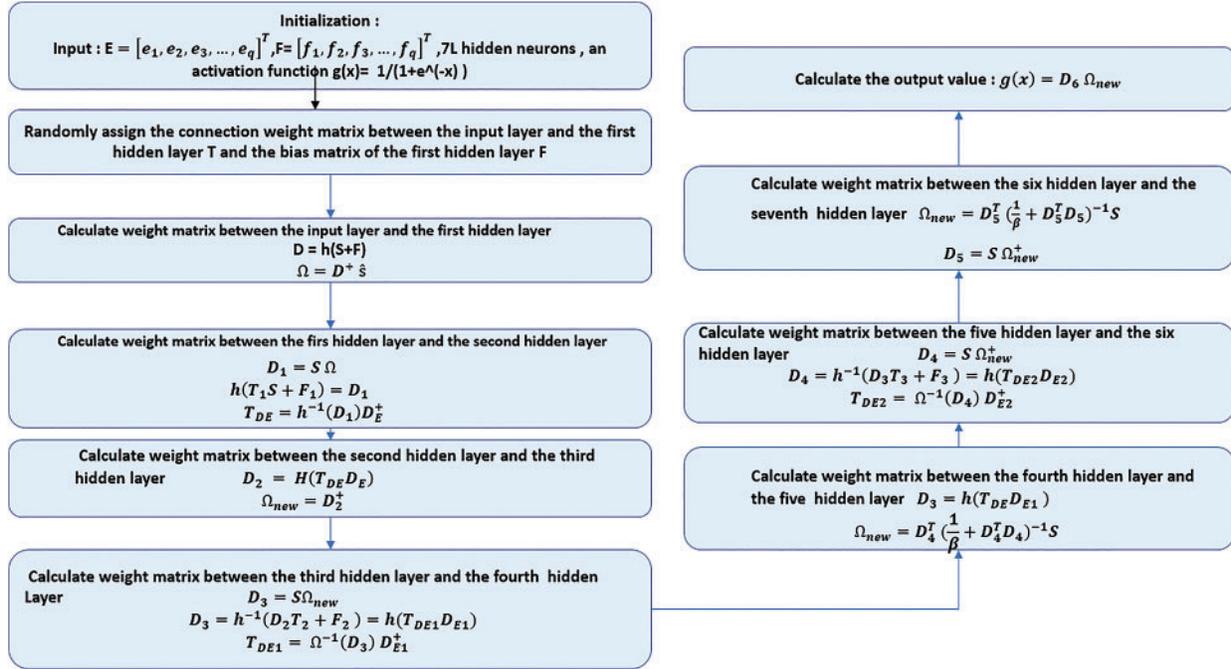
Figure 6: DELM proposed algorithm

4 Discussion

This section describes the detailed experimental evaluation of the proposed model on this dataset [32]. We evaluate and analyze the proposed LSTM-DELM network using different types of experiments to measure its performance using the household performance dataset. It has been collected from 2006 to 2010. It contains 2075259 instances, with 25979 instances containing missing values, accounting for 1.25% of the whole data. In Table 1, the variables used in this dataset are listed with their detailed comments. In addition, the quantitative details of the household electricity dataset are provided. The data set is divided into two parts:

- Training set: 70% of the data is selected as the training set.
- Test set: 30% of the data set is selected and used as the test set for evaluation of the detector's performance. Training was performed over 100 epochs, with each epoch running through all training images containing 350 steps in each epoch.

Table 1: Dataset [32] with its attribute description

Attribute	Symbol	Unit	Description	Min	Max
Date	D	Days/months/years	1∼30/1∼12/2006∼2010	16.12.2006	26.11.2010
Timing	T	Hours/minutes/seconds	1∼24/1∼60/1∼60	00:00:00	23:59:00
Global_active_power	GAP	Kilowatts	Power consumed in each minute	0.076	11.12
Global_reactive_power	GRP	Kilowatts	Reactive power consumed in each minute	0	1.390

Table 1: Continued

Attribute	Symbol	Unit	Description	Min	Max
Voltage	V	Volts	Voltage in each minute	223.200	254.150
Global_intensity	I	Ampere	Current consumed in each minute	0.200	48.400
Sub_metering_1	S1	Watts per hour	Energy consumed for dishwasher and microwave oven	0	88.000
Sub_metering_2	S2	Watts per hour	Energy consumed in living room over washing machines, drier, and refrigerator	0	88.000
Sub_metering_3	S3	Watts per hour	Energy consumed for air conditioner and water heater	0	31.000

We analyze the results of our proposed method. The proposed method is implemented in Python (version 3.5) using Keras (Deep Learning Framework) powered by TensorFlow. In order to evaluate the forecasting models' performance, these assessment criteria are used: The MAE is based on the absolute error and can show the normal separation between the expected value and the actual value. To analyze the precision of different anticipating standards, RMSE is commonly used as an expectation quality estimation. Because the measurement mathematically enhances errors, huge errors are punished. R^2 measures how closely a model approximates the real information, giving a metric of its consistency.

$$MAE = \frac{1}{n} \sum_{1}^{n} (S_i - T_i) \tag{42}$$

$$MSE = \frac{1}{n} \sum_{1}^{n} (S_i - T_i)^2 \tag{43}$$

$$RMSE = \sqrt{\left(\frac{1}{n} \sum_{1}^{n} (S_i - T_i)^2\right)} \tag{44}$$

where n represents the number of observations, T denotes the actual value, and S denotes the estimated value. Accuracy metrics values are given in Table 2.

Table 2: Accuracy metrics to validate the proposed model

Model	Training time (s)	MSE	R^2	RMSE	MAE
Arima	1.3	7.7841	1.6	2.79	0.683
Sarimax	1.42	5.29	1.5	2.3	0.620
LSTM-DELM	3.1	1.5129	1.1	1.23	0.42

As training time, LSTM-DELM achieves 3.1 with 1.5129 as RMSE. Compared to other models, as illustrated in Fig. 7, it outperforms other ones.
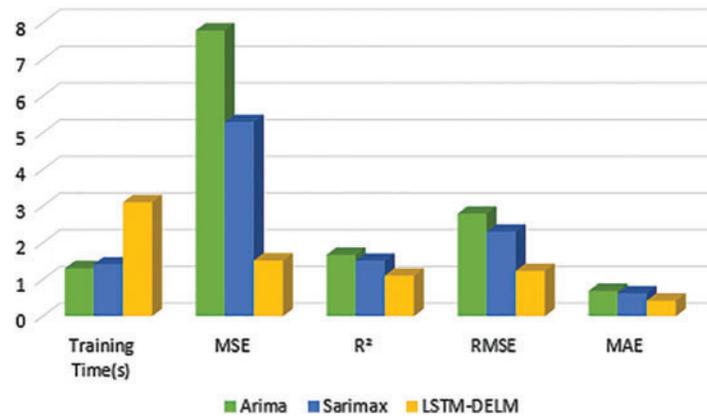
Figure 7: Results of energy prediction

A learning curve is most commonly illustrated by loss over time. Our model error is measured by loss. Therefore, our model's performance will improve as our loss decreases. Fig. 8 illustrates how the learning process is expected to behave.
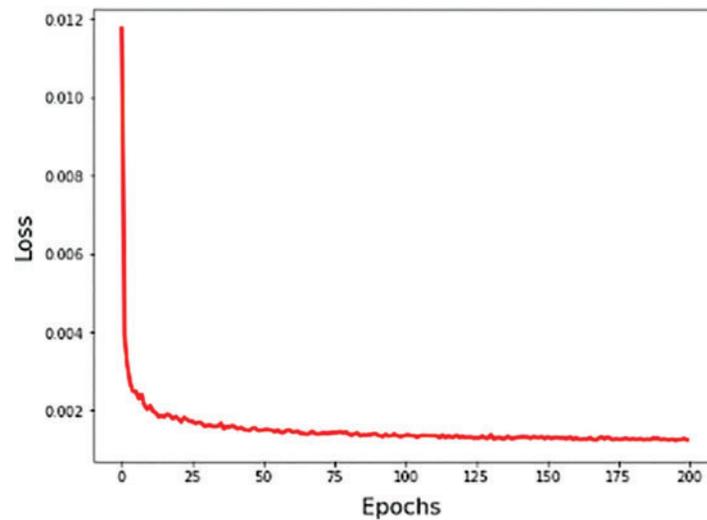
Figure 8: Loss curve

Fig. 9 presents a representation of our model forecasting results on the household power dataset.
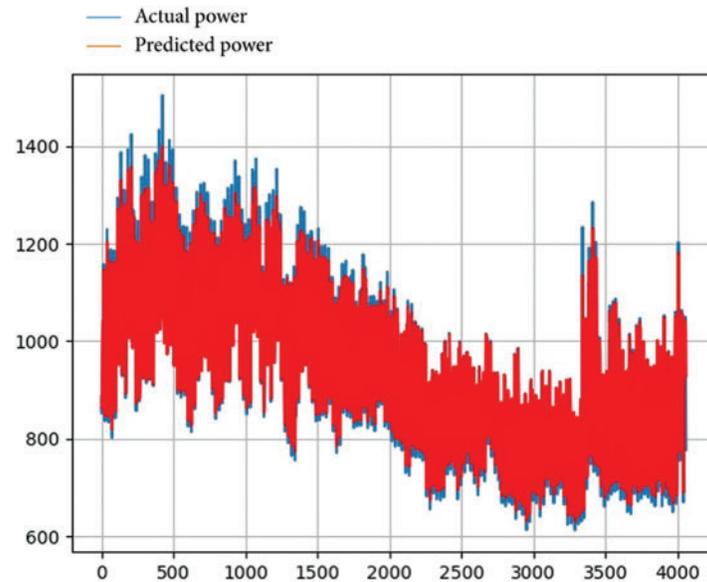
Figure 9: Forecasting energy using LSTM-DELM

5 Conclusions

Deep learning models demonstrated their potent learning and prediction capabilities in time series prediction. We recommend the use of hybrid deep learning models, especially LSTM-DELM models to predict energy consumption in the case of smart building. Focusing on the objective of this research, a new model was proposed in order to predict energy in houses. It's a hybrid approach that combines LSTM with DELM (with seven hidden layers). It was successfully compared in terms of its prediction performance with other statistical models The consequence of the model training and testing shows that our model has shown its strength in predicting energy consumption in terms of MAE (0.42), R-squared (1, 1) and RMSE (1.23). The model proposed with a modified architecture of DELM outperforms other models. It can be applicable in multiple IoT applications such as smart cities, smart homes, smart buildings and smart grids.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] A. Vishwanath, V. Chandan and K. Saurav, "An IoT-based data driven precooling solution for electricity cost savings in commercial buildings," *IEEE Internet of Things Journal*, vol. 6, no. 5, pp. 7337–7347, 2019.

[2] J. Huang, "Energy efficiency improvement of composite energy building energy supply system based on multiobjective network sensor optimization," *Journal of Sensors*, vol. 2022, 2022.

[3] D. B. Crawley, L. K. Lawrie, F. C. Winkelmann, W. F. Buhl, Y. J. Huang, *et al.,* "EnergyPlus: Creating a new-generation building energy simulation program," *Energy and Buildings*, vol. 33, no. 4, pp. 319–331, 2001.

[4] D. Yan, J. Xia, W. Tang, F. Song, X. Zhang *et al.,*"DeST—An integrated building simulation toolkit part I: Fundamentals," *Building Simulation*, vol. 1, no. 2, pp. 95–110, 2008.

[5] Zhu, Y., "Applying computer-based simulation to energy auditing: A case study," *Energy and Buildings*, vol. 38, no. 5, pp. 421–428, 2006.

[6] J. Seo, S. Kim, S. Lee, H. Jeong, T. Kim *et al.,* "Data-driven approach to predicting the energy performance of residential buildings using minimal input data," *Building and Environment*, vol. 214, pp. 108911, 2022.

[7] C. Nakkach, A. Zrelli and T. Ezzdine, "Deep learning algorithms enabling event edtection: A review," in *the 2nd Int. Conf. on Industry 4.0 and Artificial Intelligence (ICIAI 2021)*, Tunisia, Sousse, pp. 170–175, 2022.

[8] K. Mirzaei, M. Arashpour, E. Asadi, H. Masoumi, Y. Bai *et al.,* "3D point cloud data processing with machine learning for construction and infrastructure applications: A comprehensive review," *Advanced Engineering Informatics*, vol. 51, pp. 101501, 2022.

[9] C. Nakkach, A. Zrelli and T. Ezzedine, "Smart border surveillance system based on deep learning methods," in *the Int. Symp. on Networks, Computers and Communications (ISNCC)*, Shenzhen, China, pp. 1–6, 2022.

[10] A. Zrelli, C. Nakkach and T. Ezzedine, "Cyber-security for IoT applications based on ANN algorithm," in *Int. Symp. on Networks, Computers and Communications (ISNCC)*, Shenzhen, China, pp. 1–5, 2022.

[11] C. Nakkach, A. Zrelli and T. Ezzedine, "Plane's detection in border area using modern deep learning methods," in *the IEEE/ACIS 22nd Int. Conf. on Computer and Information Science (ICIS)*, Zhuhai, China, pp. 88–93, 2022.

[12] V. Jasti, A. S. Zamani, K. Arumugam, M. Naved, H. Pallathadka *et al.,* "Computational technique based on machine learning and image processing for medical image analysis of breast cancer diagnosis," *Security and Communication Networks*, vol. 2022, 2022.

[13] Y. Duan, J. W. Goodell, H. Li and X. Li, "Assessing machine learning for forecasting economic risk: Evidence from an expanded Chinese financial information set," *Finance Research Letters*, vol. 46, pp. 102273, 2022.

[14] C. Nakkach, A. Zrelli and T. Ezzdine, "An efficient approach of vehicle vetection based on deep learning algorithms and wireless sensors networks," *International Journal of Software Innovation (IJSI)*, vol. 10, no. 1, pp. 1–16, 2022.

[15] A. S. Shah, H. Nasir, M. Fayaz, A. Lajis and A. Shah, "A review on energy consumption optimization techniques in IoT based smart building environments," *Information*, vol. 10, no no. 3, pp. 108, 2019.

[16] M. Castelli, L. Trujillo, L. Vanneschi and A. Popovič, "Prediction of energy performance of residential buildings: A genetic programming approach," *Energy and Buildings*, vol. 102, pp. 67–74, 2015.

[17] K. Li, X. Xie, W. Xue, X. Dai, X. Chen *et al.,* "A hybrid teaching-learning artificial neural network for building electrical energy consumption prediction," *Energy and Buildings*, vol. 174, pp. 323–334, 2018.

[18] F. Wahid and D. H. Kim, "Short-term energy consumption prediction in Korean residential buildings using optimized multi-layer perceptron," *Kuwait Journal of Science*, vol. 44, no. 2, pp. 67–77, 2017.

[19] M. Collotta and G. Pau, "An innovative approach for forecasting of energy requirements to improve a smart home management system based on BLE," *IEEE Transactions on Green Communications and Networking*, vol. 1, no. 1, pp. 112–120, 2017.

[20] S. Sholahudin and H. Han, "Simplified dynamic neural network model to predict heating load of a building using taguchi method," *Energy*, vol. 115, pp. 1672–1678, 2016.

[21] D. Zhao, M. Zhong, X. Zhang and X. Su, "Energy consumption predicting model of VRV (Variable refrigerant volume) system in office buildings based on data mining," *Energy*, vol. 102, pp. 660–668, 2016.

[22] S. Goudarzi, M. H. Anisi, N. Kama, F. Doctor, S. A. Soleymani *et al.,* "Predictive modelling of building energy consumption based on a hybrid nature-inspired optimization algorithm," *Energy and Buildings*, vol. 196, pp. 83–93, 2019.

[23] Y. H. Lin and Y. C. Hu, "Electrical energy management based on a hybrid artificial neural network-particle swarm optimization-integrated two-stage non-intrusive load monitoring process in smart homes," *Processes*, vol. 6, no. 12, pp. 236, 2018.

[24] S. Naji, A. Keivani, S. Shamshirband, U. J. Alengaram, M. Z. Jumaat *et al.,* "Estimating building energy consumption using extreme learning machine method," *Energy*, vol. 97, pp. 506–516, 2016.

[25] M. Fayaz and D. Kim, "Energy consumption optimization and user comfort management in residential buildings using a bat algorithm and fuzzy logic," *Energies*, vol. 11, no. 1, pp. 161, 2018.

[26] S. S. Subbiah, S. K. Paramasivan, K. Arockiasamy, S. Senthivel and M. Thangavel, "Deep learning for wind speed forecasting using bi-lstm with selected features," *Intelligent Automation & Soft Computing*, vol. 35, no. 3, pp. 3829–3844, 2023.

[27] S. Salcedo-Sanz, L. Cornejo-Bueno, L. Prieto, D. Paredes and R. García-Herrera, "Feature selection in machine learning prediction systems for renewable energy applications," *Renewable and Sustainable Energy Reviews*, vol. 90, pp. 728–741, 2018.

[28] J. Cheng, Z. Duan and Y. Xiong, "QAPSO-BP algorithm and its application in vibration fault diagnosis for a hydroelectric generating unit," *J. Vib. Shock*, vol. 34, no. 23, pp. 177–181, 2015.

[29] S. Ding, N. Zhang, X. Xu, L. Guo and J. Zhang, "Deep extreme learning machine and its application in EEG classification," *Mathematical Problems in Engineering*, vol. 2015, 2015.

[30] D. Xiao, B. Li and Y. Mao, "A multiple hidden layers extreme learning machine method and its application," *Mathematical Problems in Engineering*, vol. 2017, 2017.

[31] G. B. Huang, Q. Y. Zhu and C. K. Siew, "Extreme learning machine: Theory and applications," *Neurocomputing*, vol. 70, no. 1–3, pp. 489–501, 2006.

[32] G. Hebrail and A. Berard, "Individual household electric power consumption data set," in: É. d. France(Ed.), *UCI Machine Learning Repository*, 2012.