applied sciences

MDPI

Article

Artificial Jellyfish Optimization with Deep-Learning-Driven Decision Support System for Energy Management in Smart Cities

A. Al-Qarafi [1], Hadeel Alsolai [2], Jaber S. Alzahrani [3], Noha Negm [4], Lubna A. Alharbi [5], Mesfer Al Duhayyim [6], Heba Mohsen [7], M. Al-Shabi [8] and Fahd N. Al-Wesabi [4,*]

[1] Department of Information Systems, College of Computer Science and Engineering, Taibah University, Medina 42353, Saudi Arabia; aqarafi@taibahu.edu.sa
[2] Department of Information Systems, College of Computer and Information Sciences, Princess Nourah Bint Abdulrahman University, P.O. Box 84428, Riyadh 11671, Saudi Arabia; haalsolai@pnu.edu.sa
[3] Department of Industrial Engineering, College of Engineering at Alqunfudah, Umm Al-Qura University, Mecca 24382, Saudi Arabia; jszahrani@uqu.edu.sa
[4] Department of Computer Science, College of Science & Art at Mahayil, King Khalid University, Abha 62529, Saudi Arabia; nohawesabi@gmail.com
[5] Department of Computer Science, College of Computers and Information Technology, Tabuk University, Tabuk 47512, Saudi Arabia; lualharbi@ut.edu.sa
[6] Department of Computer Science, College of Sciences and Humanities-Aflaj, Prince Sattam Bin Abdulaziz University, Al-Kharj 16278, Saudi Arabia; m.alduhayyim@psau.edu.sa
[7] Department of Computer Science, Faculty of Computers and Information Technology, Future University in Egypt, New Cairo 11835, Egypt; hmohsen@fue.edu.eg
[8] Department of Management Information System, College of Business Administration, Taibah University, Medina 42353, Saudi Arabia; mshaby@taibahu.edu.sa
* Correspondence: falwesabi@kku.edu.sa

check for updates

Citation: Al-Qarafi, A.; Alsolai, H.; Alzahrani, J.S.; Negm, N.; Alharbi, L.A.; Al Duhayyim, M.; Mohsen, H.; Al-Shabi, M.; Al-Wesabi, F.N. Artificial Jellyfish Optimization with Deep-Learning-Driven Decision Support System for Energy Management in Smart Cities. *Appl. Sci.* **2022**, *12*, 7457. https://doi.org/10.3390/app12157457

Academic Editors: Gianni Pantaleo and Pierfrancesco Bellini

Received: 19 June 2022
Accepted: 22 July 2022
Published: 25 July 2022

Abstract: A smart city is a sustainable and effectual urban center which offers a maximal quality of life to its inhabitants with the optimal management of their resources. Energy management is the most difficult problem in such urban centers because of the difficulty of energy models and their important role. The recent developments of machine learning (ML) and deep learning (DL) models pave the way to design effective energy management schemes. In this respect, this study introduces an artificial jellyfish optimization with deep learning-driven decision support system (AJODL-DSSEM) model for energy management in smart cities. The proposed AJODL-DSSEM model predicts the energy in the smart city environment. To do so, the proposed AJODL-DSSEM model primarily performs data preprocessing at the initial stage to normalize the data. Besides, the AJODL-DSSEM model involves the attention-based convolutional neural network-bidirectional long short-term memory (CNN-ABLSTM) model for the prediction of energy. For the hyperparameter tuning of the CNN-ABLSTM model, the AJO algorithm was applied. The experimental validation of the proposed AJODL-DSSEM model was tested using two open-access datasets, namely the IHEPC and ISO-NE datasets. The comparative study reported the improved outcomes of the AJODL-DSSEM model over recent approaches.

Keywords: smart cities; energy management; decision support systems; deep learning; prediction models; hyperparameter optimization

1. Introduction

The word "smart city" means an urban system targeting satisfying efficacy and stability phenomena [1] inside crucial fields and implementation zones, such as energy and environmental management, mobility, administrative services, etc. A smart city is comprised of various distinct functional environments, substructures, and networks that can be optimized and enhanced via the application of developed solutions [2]. There is a demand to assess the present conditions of the city (via data arising from sensor networks located in the metropolitan regions), and decisions should be made in accordance with particular goals and targets. This means the advancement of intensely linked substructures, emerging alongside the smart city atmosphere [3,4]. Based on these methods, there exist decision support systems (DSSs) and computational methodologies. A DSS, broadly implemented in numerous sectors and fields to guide the automation of decisional functions, understands and interprets the diverse necessities to be encountered, considering the relative merits and demerits of the constituting components [5]. DSSs have been broadly researched and employed in a wide range of application zones, starting from clinical DSSs to management and business, including smart cities [6].

Figure 1 illustrates the process of energy management in smart cities. The energy in the smart city environment can be optimally managed to satisfy resource availability, system cost, geolocation characteristics, energy prices, regulatory constraints, environmental benefits, etc. The power deployment and impact of smart technologies, regarding the dynamic optimization of grid operations and resources, automation, analytics, and information exchange, are major difficulties for industrial units to understand the prerequisites for computational intelligence (CI) patterns in brilliant decision-support techniques [7,8]. CI has several branches which are unconstrained to neural networks, such as expert systems, fuzzy systems, artificial immune systems, swarm intelligence [9], evolutionary computing, and numerous hybrid models, which are compositions of two or more branches. Additionally, CI is a successor of artificial intelligence (AI), and by means of future computing, approaches smart grid functions in energy management. Energy management is a global problem with significant consequences [10]. High power surges and environmental factors necessitate the transition of electric power grids and smart grids towards the direction of higher rational energy consumption (ECM).

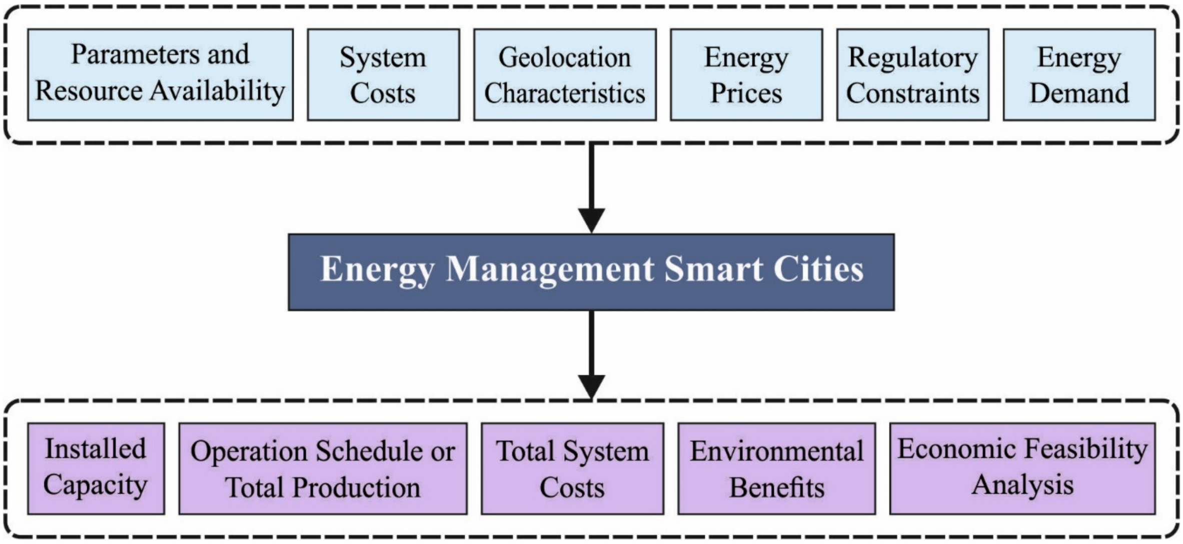
Parameters and Resource Availability
System Costs
Geolocation Characteristics
Energy Prices
Regulatory Constraints
Energy Demand
Energy Management Smart Cities
Installed Capacity
Operation Schedule or Total Production
Total System Costs
Environmental Benefits
Economic Feasibility Analysis

Figure 1. Process of energy management in smart cities.

The suitable regulation of power generation and utilization promises its effective exploitation, which requires a smart grid to maintain consistent power transmission among users and producers for balancing the respective energy status [11]. In this regard, load estimation approaches are necessary for allowing the estimation of effective power utilization to neglect extra expenses and hikes from the loss ratio, since a million pounds vanish annually because of energy wastage [11]. Subsequently, the precise and dependable load forecasting (LF) method is necessary for perfect energy management. An intelligent data-driven LF method frequently compiled numerous real-life IoT-application-like smart constructions in the day ahead of estimations, developing suitable energy needs in smart grids that decrease the likelihood of serious energy shortfalls and endorse optimal utilization. Such techniques can be classified into two categories: they are machine learning (ML) or statistical techniques, and deep learning (DL) techniques.

This study introduced an artificial jellyfish optimization with deep-learning-driven decision support system (AJODL-DSSEM) model for energy management in smart cities. The proposed AJODL-DSSEM model initially performed data preprocessing at the initial stage to normalize the data. Additionally, the AJODL-DSSEM model involved an attention-based convolutional neural network bidirectional long short-term memory (CNN-ABLSTM) model for the prediction of energy. Moreover, the AJO algorithm was applied for the hyperparameter adjustment of the CNN-ABLSTM model. The experimental validation of the proposed AJODL-DSSEM model was tested using two open access datasets, namely the IHEPC and ISO-NE datasets.

The rest of the paper is arranged as follows: Section 2 offers the related work and Section 3 introduces the proposed model. Next, Section 4 provides experimental validation and Section 5 reports the conclusions.

2. Related Works

This section offers a detailed survey of energy management schemes in the smart city environment. Shreenidhi et al. [12] presented an effective load-scheduling model called the two-stage deep dilated multi-kernel convolutional network (DDMKC)-modified elephant herd optimization algorithm (MEHOA) model for managing and shoring the load, and reducing the electricity bill. Here, the proposed model exploited demand response (DR) pricing information to precisely predict the future pricing signal to make optimal decisions and achieve the minimal degree of discomfort. Lotfi et al. [13] analyzed the coordination among home energy management systems (HEMSs), and EV parking lot management systems (PLEMSs). The EMS coordinated the partial sharing of individual EV schedules with no communication of private data.

Elsisi et al. [14] projected a DL-based person recognition scheme using YOLOv3 architecture to calculate the number of people in a certain region. Consequently, the function of air conditioners was optimally accomplished in a smart building. The presented algorithm improved the decision making regarding the consumption of energy. For confirming the efficiency and efficacy of the suggested manner, intensive test scenarios were inspired by a smart building by considering the existence of air conditioners. Vázquez-Canteli et al. [15] projected a combined simulation environment which incorporated TensorFlow, CitySim, and a faster constructing energy simulator, a platform for efficiently implementing innovative ML algorithms.

In [16], the role of IoT in fusing green energy resources into a smart electrical grid was presented using a multiobjective-distributed dispatching algorithm (MODDA). Effectual energy management involved a trade-off of the cost connected to ECM and the utility function. Therefore, the changes amongst ECM and the utility function should be recognized. Ullah et al. [17] scheduled new appliances for university campuses to decrease the cost of ECM and the possible peak-to-average-power ratio. The study presented two nature-inspired approaches, such as the sine-cosine algorithm (SCA) and the multi-verse optimization (MVO) technique, to resolve the optimization issue.

In [18], the authors proposed a multi-scale LSTM-based DL technique which was able to forecast short-term PVGF for effective management. The algorithm concentrated on two dissimilarly scaled LSTM models for overcoming the shortcomings devised from the irregular factor. In [19], a special edition of an RNN, for example, the LSTM, was briefly discussed. We presented ANNdotNET that provided a user-friendly ML architecture with the ability to import information from the smart grid of smart cities. The ANNdotNET is a cloud solution that is interconnected by other IoT devices for information providing, gathering, and feeding effective methods for energy management for smart city cloud solutions. Li et al. [20] conducted a big data analysis (BDA) on the large volumes of data produced in the smart city IoT, constructing the smart city alteration to efficient and safe data processing to the direction of fine governance. Directing the multiple source information gathered from the smart city, the DL approach, utilizing BDA, was developed and offered the distributed parallelism approach of CNNs.

After reviewing the existing studies, we noticed that the energy management performance for smart cities has yet to be increased. Though DL models are available in the literature for energy prediction, the predictive results need to be improved. At the same time, the parameters related to the DL models increased due to the incessant deepening of the model, which resulted in model overfitting. In addition, various hyperparameters had a significant impact on the efficiency of the CNN model, particularly the learning rate. The learning rate parameter for obtaining better performances must be modified. Hence, we applied the AJO algorithm for the hyperparameter tuning of the CNN-ABLSTM model.

3. The Proposed Model

In this study, a novel AJODL-DSSEM algorithm was established for the prediction of energy in the smart city environment. At the initial stage, the proposed AJODL-DSSEM model mainly accomplished data preprocessing at the initial stage to normalize the data. Apart from data preprocessing, the AJODL-DSSEM model involved the CNN-ABLSTM model for the prediction of energy. Finally, the AJO algorithm was applied for the hyperparameter adjustment of the CNN-ABLSTM model, which in turn helped in achieving improved prediction performance.

3.1. Design of CNN-ABLSTM-Based Predictive Model

For the effective prediction of ECM in smart cities, the pre-processed data was passed into the CNN-ABLSTM model. The CNN had pooling, convolution, and fully connected (FC) layers. The CNN captured hidden features in the input data by implementing convolution as well as pooling functions. Afterward, the extracting features were combined and fed into the FC layer. Lastly, several activation functions were employed for introducing non-linearity to the resultant neurons. The convolutional layer was a vital part of the CNN. All the convolution layers maintained several convolution kernels that were convolved with the input data to capture hidden features and develop feature maps. The feature map endured a nonlinear activation function for generating the results of the convolution layer. The convolution layer is formulated as:

$$c_i = f(w_{\mathrm{i}} * x_i + b_i) \tag{1}$$

where x_i signifies the input of the convolutional layer, c_i represents the i^{th} resultant feature map, w_i denotes the weighted matrix, $'t$ implies the dot products, b_i represents the bias vector, and $f(\cdot)$ stands for the activation function. The ReLU function has been widely selected as the activation function of CNNs. In the mathematical process, ReLU is determined as:

$$c_i = f(h_i) = \max\,(0,\ h_i) \tag{2}$$

where h_i denotes the element of feature maps attained in the convolution functions. Max pooling is the most utilized pooling approach. It can be understood by computing the maximal value of the allocated region from the feature maps based on Equations (3) and (4):

$$\gamma(c_i,\ c_{i-1}) = \max\,(c_i,\ c_{i-1}) \tag{3}$$

$$p_i = \gamma(c_i,\ c_{i-1}) + \beta_i \tag{4}$$

where $\gamma(\cdot)$ signifies the max-pooling sub-sampling function, β_i indicates the bias, and p_i stands for the result of the max-pooling layer. Lastly, the feature maps attained with convolution and pooling functions were fed into the FC layer; then, the layer computed the last resultant vector, as demonstrated under:

$$y_j = f(t_i p_i + \delta_i) \tag{5}$$

where y_i denotes the last resultant vector, δ_i represents the bias, and r_i implies the weighted matrix.

The proposed architecture is a structure of two branches. One branch used a CNN to capture the spatial properties of the data, and the other conducted the feature selections by utilizing a two-layer BiLSTM model with an attention mechanism.

LSTM NNs are variants of RNNs and solve the gradient vanishing problems of RNNs. LSTM adds a memory cell structure from the neural node of the hidden state of RNNs for storing the previous data and adds a three-gate infrastructure: forget, output, and input gates, to control the utilization of the previous data. By forgetting the unused data and memorizing the original data from the cell state, LSTM transfers valuable data from the subsequent computation time [21]. The computation equation is shown in the subsequent formulae:

$$i_\tau = o(W_i \cdot [h_{\tau-12} x_\tau] + b_i) \tag{6}$$

$$f_T = o\left(W_f \cdot [h_{\tau-12} x_\tau] + b_f\right) \tag{7}$$

$$o_\tau = o(W_o \cdot [h_{\tau-12} x_\tau] + b_o) \tag{8}$$

$$h_\tau = 0_\tau \odot tanh(c_\tau) \tag{9}$$

$$c_\tau = f_T \odot c_{t-1} + i_t \odot \widetilde{c} \tag{10}$$

$$\widetilde{c} = tanh(W_c \cdot [h_{\tau-1}, x_t] + b_c) \tag{11}$$

$$o(x) = \frac{1}{1 + e^{-\chi}} \tag{12}$$

$$tanh\ (x) = \frac{e^{\chi} - e^{-\chi}}{e^{\chi} + e^{-\chi}} \tag{13}$$

where $\widetilde{c_t}$ refers to the temporary state and c_t denotes the present state. i_t, f_t, and 0_t signify output, input, and forget gates, respectively, x_τ signifies the present input, and $h_{\tau-1}$ denotes the hidden state of the earlier time. W_i, W_f, and W_o characterize the connection weight of the three gates, b specifies the offset, and σ and $\tan\text{h}$ symbolize the activation functions. Because LSTM only learns the abovementioned dataset of sequential time, BiLSTM makes further progress to LSTM; for example, it reverses and forwards LSTM networks and presents the contextual dataset of sequential time. At this point, $\chi_1, \chi_2, \ldots \chi_t$ signifies the series of inputs, $\overrightarrow{h}_t$ and $\overleftarrow{h_t}$ symbolize the forward and reverse output calculated at each moment, respectively, and they were evaluated for attaining the concluding output y_t. Assuming the forward output $\overrightarrow{h}_t$ at t time, the computational equation of forward and reverse directions was consistent with LSTM. The forward and backward temporary cell states, $\overrightarrow{\widetilde{c_t}}$ and $\overleftarrow{\widetilde{c_t}}$, $\overrightarrow{l_t}$ and $\overleftarrow{l_t}$ input gates, $\overrightarrow{f_t}$ and $\overleftarrow{f_t}$ forget gates, and $\overrightarrow{o_t}$ and $\overleftarrow{o_t}$ output gates were evaluated. Figure 2 depicts the framework of the BiLSTM technique.

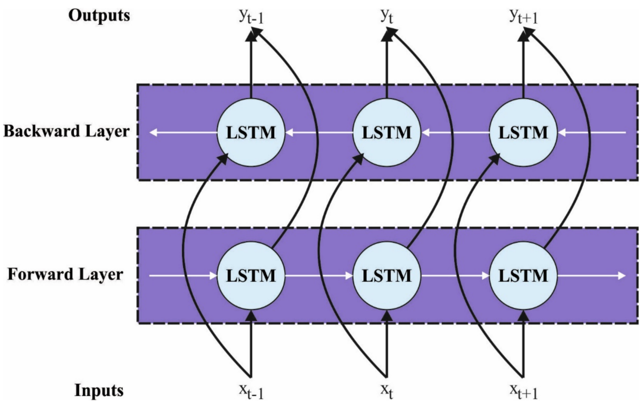

Figure 2. Structure of BiLSTM model.

The final output y_t at t time was:

$$y_t = \left[\overrightarrow{h_t},\ \overleftarrow{h_t}\right] \tag{14}$$

With Equation (14), we evaluated the output at every moment, as well as accomplished the concluding output $Y = [h_0,\ h_1,\ h_t]$. In an ABLSTM network with an attention process, the attention technique proceeded to benefit the final cell state of BiLSTM and produced a position with the cell state of input utilizing the hidden layer of BiLSTM. Next, the correlation among the resultant layer and these candidate in-between states were calculated. In the learning procedure, the connected data were noted, and the irrelevant data were suppressed for enhancing the accuracy and efficacy of the forecast [22]. The resultant A of the attention layer from the attentive BiLSTM network was created, based on the subsequent Equations (15)–(17):

$$M = tanh\ (Y) \tag{15}$$

$$\alpha = softmax\left(w_a^T M\right) \tag{16}$$

$$A = Y\alpha^T \tag{17}$$

where y represents the matrix and signifies the features captured with the BiLSTM technique as the aforementioned matrix $y = [y_1,\ y_2,\ \ldots,\ y_t]$. α signifies the vector and denotes the attention weighted to the feature y. w_a implies the weighted co-efficient matrix of the attention layer. T demonstrates the transpose function.

3.2. Hyperparameter Optimization

In this study, the hyperparameters of the CNN-ABLSTM model, such as learning rate, batch size, and the number of epochs, were optimally chosen for the use of the AJO algorithm. The AJO algorithm was simulated for the performance of jellyfish (JF) in the ocean. The AJO behavior of searching for food in the ocean consisted of movement inside the swarm or following the ocean current and utilizing a time-control model for switching between these movements [23].

Primarily, we observed a chaotic map with a random method to discover the optimal initialized method that precisely distributed the solution in the searching space to prevent getting stuck in local minima and to speed up the convergence. After observation, the JF were implemented in the logistic map, arithmetically defined as the following:

$$\overrightarrow{X}_{i+1} = \eta \overrightarrow{X}_i (1 - X_i),\ 0 \leq \overrightarrow{X}_o \leq 1 \tag{18}$$

where $\overrightarrow{X}_i$ refers to a vector that comprised the logistic chaotic values of i^{th} JF. $\overrightarrow{X}_0$ indicates a primary vector of JF 0, randomly created within [0, 1]. This vector was an initial point that was dependent upon creating the logistic chaotic value to the remainder of JF. η was allocated to a value of four. After being initialized, every solution was observed and the one with optimal fitness values was selected as the position with food $\overrightarrow{X}$. Then, the present location of every jellyfish was updated towards either the ocean current or motion inside the swarm, depending upon the time-control strategy for switching between the two movements. Mathematically, the ocean current can be defined as follows [24]:

$$\overrightarrow{X}_i(t+1) = \overrightarrow{X}_i(t) + \overrightarrow{r} * \left(\overrightarrow{X'} - \beta * r_1 * \mu\right) \tag{19}$$

where $\overrightarrow{X'}$ represents the jellyfish having the current best position among the whole population, $\overrightarrow{r}$ represents a vector randomly generated within [0, 1], $*$ indicates the element-wise vector multiplication, $\beta > 0$ denotes the distribution co-efficient that depends on the sensitivity analysis, $\beta = 3$ μ represents the mean of the population, and r_1 indicates an arbitrary value within [0, 1]. Figure 3 illustrates the behaviors involved in jellyfish.

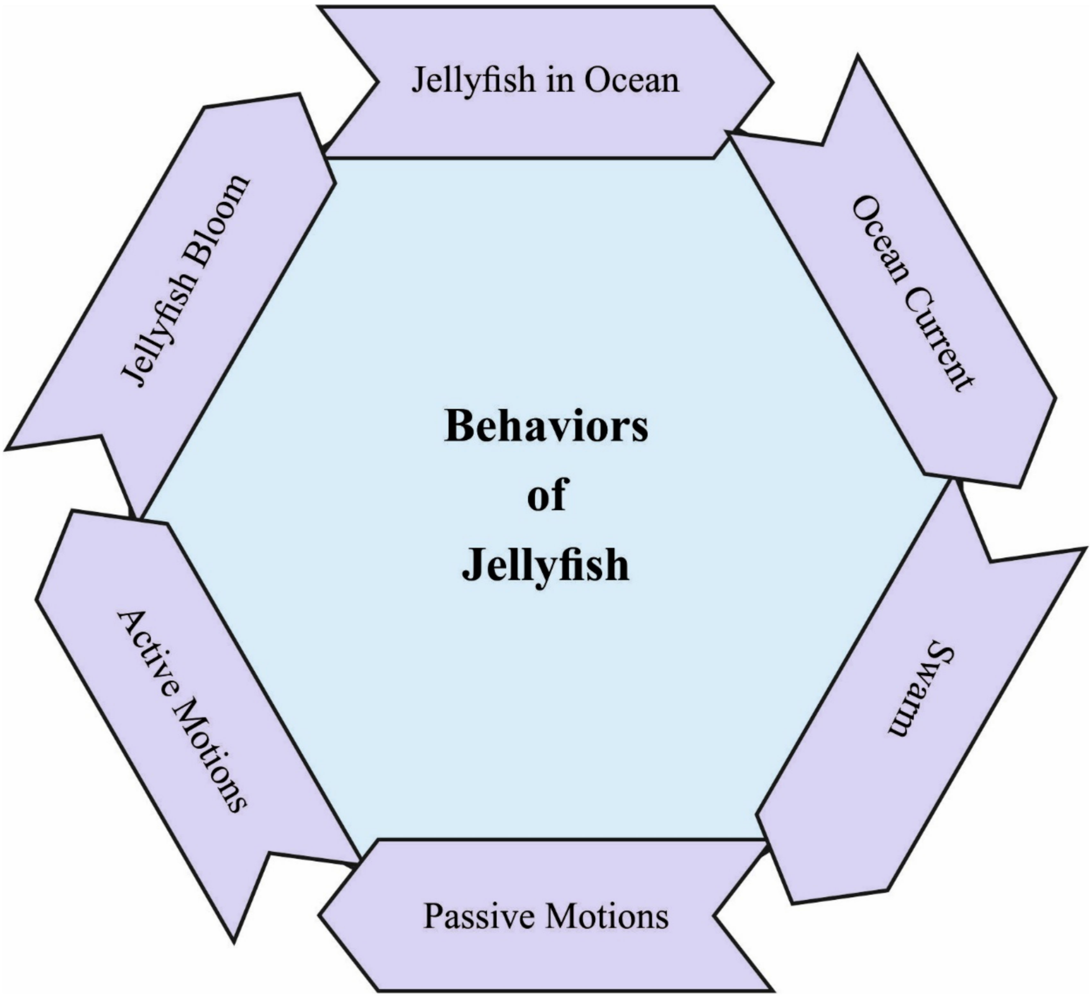

Figure 3. Behaviors of jellyfish.

The movement inside the JF swarm is classified into active and passive motions. In the passive motion, the JF moves nearby the location, and the novel position is given as follows:

$$\overrightarrow{X}_i(t+1) = \overrightarrow{X}_i(t) + r_3 * \gamma * (U_b - L_b) \tag{20}$$

where r_3 indicates an arbitrary value within [0, 1], and $\gamma > 0$ denotes the length of motion near the present position. u_b and L_b characterize the upper as well as lower limits of searching space o, respectively. The mathematical expression of the active motion is given as:

$$\overrightarrow{X}_i(t+1) = \overrightarrow{X}_i(t) + \overrightarrow{r} * \overrightarrow{D} \tag{21}$$

where $\overrightarrow{r}$ denotes a vector that comprises arbitrary values lying within [0, 1]. $\overrightarrow{D}$ was utilized for determining the way of motion of the present JF with the following generation, and the motion was often toward the position of optimal food and given as follows:

$$\overrightarrow{D} = \begin{cases} \overrightarrow{X}_i(t) - \overrightarrow{X}_j(t), & if\ f\left(\overrightarrow{X}_i\right) < f\left(\overrightarrow{X}_j\right) \\ \overrightarrow{X}_j(t) - \overrightarrow{X}_i(t), & otherwise \end{cases} \tag{22}$$

where j represents the index of JF designated in a random fashion, and f designates the fitness function. The time-control model was utilized for switching between the ocean current, passive and active motions, and comprised a constant c_0. A mathematical expression of the time-control mechanism is given as:

$$c(t) = \left(1 - \frac{t}{t_{\max}}\right) * (2 * r - 1) \tag{23}$$

where t refers to the present evaluation, $t_{\max}$ indicates the maximal evaluation, and r represents an arbitrary value lying within [0, 1] as illustrated in Algorithm 1.

Algorithm 1: Pseudocode of AJO algorithm

```
Begin
    Determine the objective function $f(X)$, $X = (x_1, \ldots, x_d)^T$
    Fix the searching space, population size $(nPop,)$ and maximal iteration $(\text{Max}_{int})$
    Initialize the population of JF, $X_i(i = 1, 2, \ldots, np_{op})$, utilizing a logistic chaotic
    map
    Compute the quantity of food at all $X_i, f(X_i)$
    Define the JF at place presently with most food $(X^*)$
    Initializing time: $t = l$
    Repeat
      For $i = 1$ : nPop do
      Compute the time control $c(t)$ utilization
      If $c(t) \geq 0.5$: the JF follows the ocean current
        (1) Define the ocean current
        (2) Novel place of JF was determined
      Else: the JF moves inside a swarm
        If rand(0,1) $> (l - c(t))$: the JF displays type A motion (passive motion)
            (1) Novel place of JF was determined
        Else: JF displays type $B$ motion (active motion)
            (2) Define the direction of JF
            (3) Novel place of JF was determined
        End if
    End if
    Verify the boundary condition and compute the quantity of food at novel place
    Upgrade the place of JF $(X_i)$ and place of JF presently with the food $(X^*)$
    End for $i$
    Upgrade the time: $t = t + 1$
    Still end condition was met (e.g., $t > (\text{Max}_{int})$)
    Output the optimal outcomes and visualize (JF bloom)
End
```

This study established an AJO technique for a suitable selection of network weights from the CNN-ABLSTM method with a minimized mean square error (MSE). The MSE mathematical model is determined as:

$$MSE = \frac{1}{T} \sum_{j=1}^{L} \sum_{i=1}^{M} \left(y_j^i - d_j^i\right)^2, \tag{24}$$

where M and L represent the resultant values of layers and data, respectively, and y_j^i and d_j^i signify the attained and the appropriate magnitudes to the j^{th} unit from the resultant layer of networks from the time t.

4. Results and Analysis

The proposed model was simulated using a Python 3.6.5 tool with packages, namely tensorflow-gpu==2.2.0, scikit-learn, matplotlib, seaborn, pyqt5, prettytable, numpy, pandas, and openpyxl.

4.1. Dataset Details

In this section, the experimental validation of the AJODL-DSSEM model was tested using two open access datasets, namely the IHEPC [25] and ISO-NE datasets [26]. The IHEPC dataset encompasses 2,075,259 readings collected in a house located in Sceaux, Paris. The dataset collected power consumption for four years (from 16 December 2006 to 26 November 2010) in a home in France. The dataset holds nine attributes, such as data, time, global active power, global reactive power, voltage, global intensity, and sub metering one, two, and three. Next, the ISO-NE dataset collected hourly time-series data from 2012 to 2016, for a total of five years (43,915 samples), and was employed for model training. Similarly, one year (2017) of hourly data (8783 samples) was used for testing purposes. The dataset comprises a total of 14 features, where a feature called "SYSLOAD" was undertaken as the target label, and the dry bulb column represented the temperature in degrees Fahrenheit, among other data-time features.

4.2. Result Analysis

Table 1 offers a comprehensive predictive outcome of the AJODL-DSSEM model on two datasets. Figure 4 reports a brief result analysis of the AJODL-DSSEM model under different cases of the IHEPC dataset. The figure implied that the AJODL-DSSEM model attained enhanced performance in all aspects. For instance, with the autumn season, the AJODL-DSSEM model obtained an RMSE, MAE, and MAPE of 0.291, 0.270, and 0.349, respectively. Furthermore, with the spring season, the AJODL-DSSEM technique reached an RMSE, MAE, and MAPE of 0.271, 0.218, and 0.330, respectively. In addition, with winter the season, the AJODL-DSSEM methodology obtained an RMSE, MAE, and MAPE of 0.319, 0.280, and 0.302, respectively.

Table 1. Result analysis of AJODL-DSSEM technique with various measures under two datasets.

Label	**RMSE**	**MAE**	**MAPE**
IHEPC Dataset			
Autumn	0.291	0.270	0.349
Summer	0.330	0.281	0.335
Spring	0.271	0.218	0.330
Winter	0.319	0.280	0.302
Average	**0.303**	**0.262**	**0.329**
ISO-NE Dataset			
Autumn	0.413	0.333	0.256
Summer	0.453	0.364	0.241
Spring	0.480	0.422	0.218
Winter	0.479	0.416	0.231
Average	**0.456**	**0.384**	0.237

Figure 5 demonstrates a detailed result analysis of the AJODL-DSSEM approach under distinct cases of the ISO-NE dataset. The figure exposed that the AJODL-DSSEM technique attained improved performance under all aspects. For instance, with the autumn season, the AJODL-DSSEM model achieved an RMSE, MAE, and MAPE of 0.413, 0.333, and 0.256, respectively. Moreover, with the spring season, the AJODL-DSSEM algorithm obtained an RMSE, MAE, and MAPE of 0.480, 0.422, and 0.218, respectively. Furthermore, with the winter season, the AJODL-DSSEM methodology reached an RMSE, MAE, and MAPE of 0.479, 0.416, and 0.231, respectively.

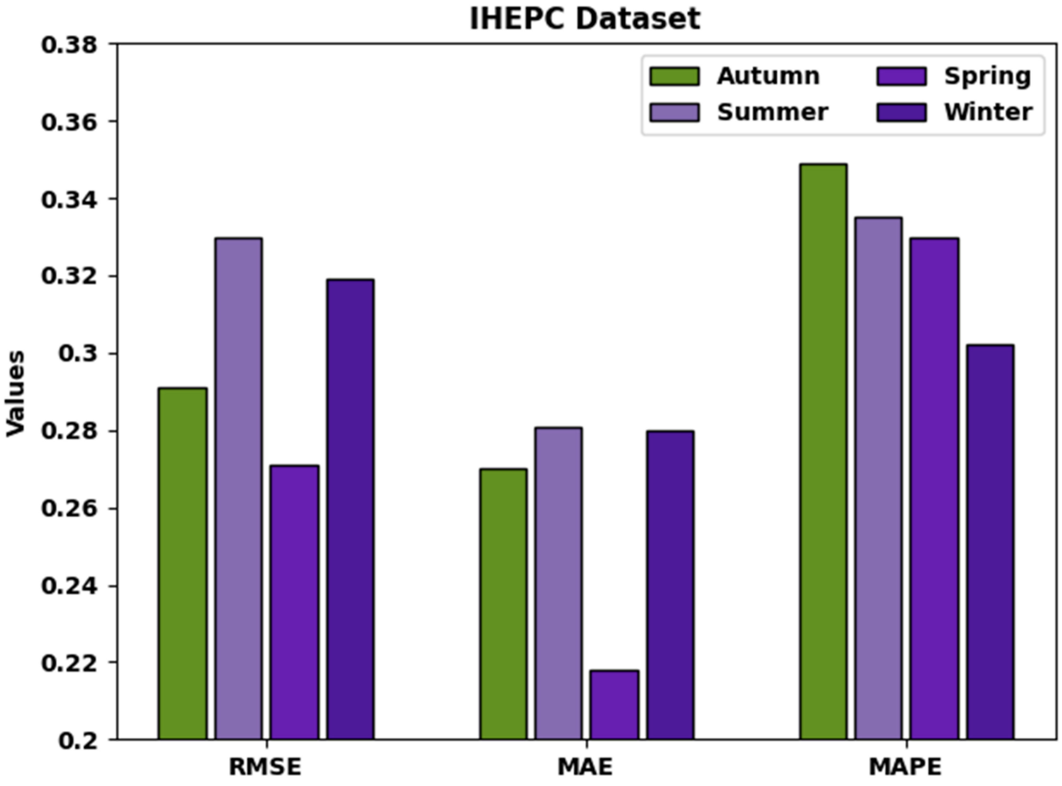

Figure 4. Result analysis of AJODL-DSSEM technique under IHEPC dataset.

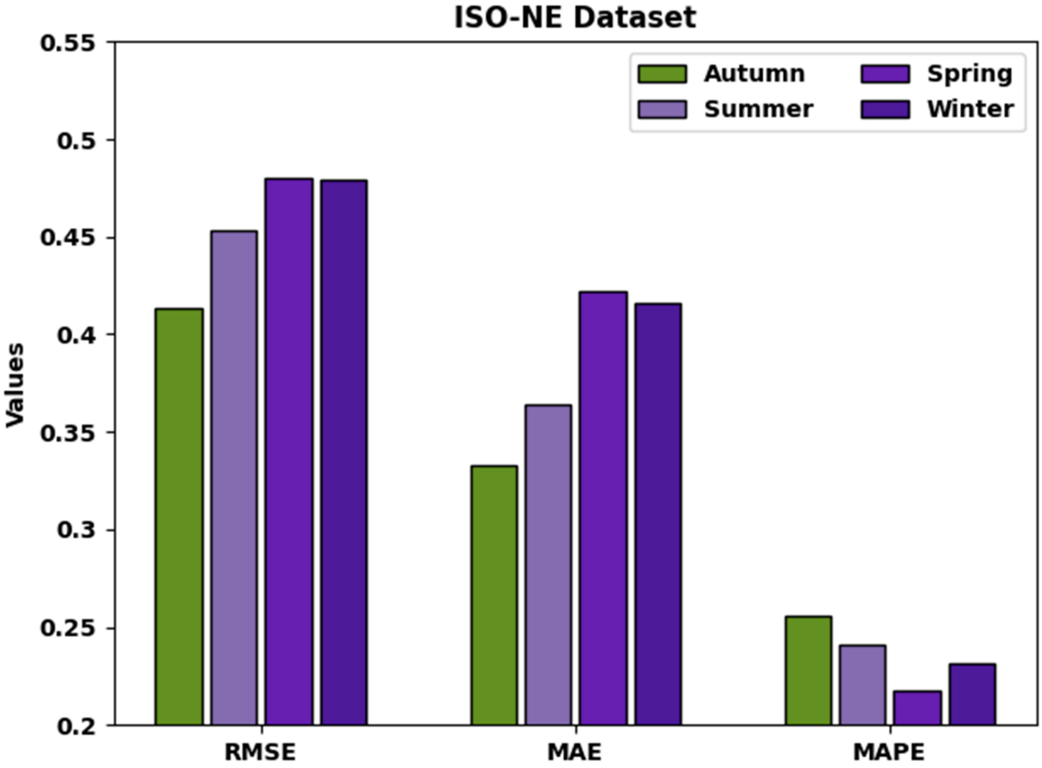

Figure 5. Result analysis of AJODL-DSSEM technique under ISO-NE dataset.

Table 2 and Figure 6 illustrate the actual vs. predicted global active power of the AJODL-DSSEM model under distinct time steps on the IHEPC dataset. The results indicated that the AJODL-DSSEM model predicted the values much closer to the actual values. For instance, with a time step of 20 h and actual value of 4.308, the IHEPC dataset obtained a predicted value of 4.383. Furthermore, with a time step of 80 h and actual value of 1.532, the IHEPC dataset reached a predicted value of 1.387. In addition, with a time step of 160 h and actual value of 0.182, the IHEPC dataset attained a predicted value of 0.222. In addition, with a time step of 200 h and actual value of 0.478, the IHEPC dataset obtained a predicted value of 0.415.

Table 2. Global active power analysis of AJODL-DSSEM technique under distinct time steps on IHEPC dataset.

Global Active Power—IHEPC Dataset		
Time Steps (h)	**Actual**	**Predicted**
0	1.053	0.890
20	4.308	4.383
40	0.334	0.223
60	0.422	0.580
80	1.532	1.387
100	0.321	0.302
120	0.283	0.187
140	1.368	1.309
160	0.182	0.222
180	1.961	1.762
200	0.478	0.415

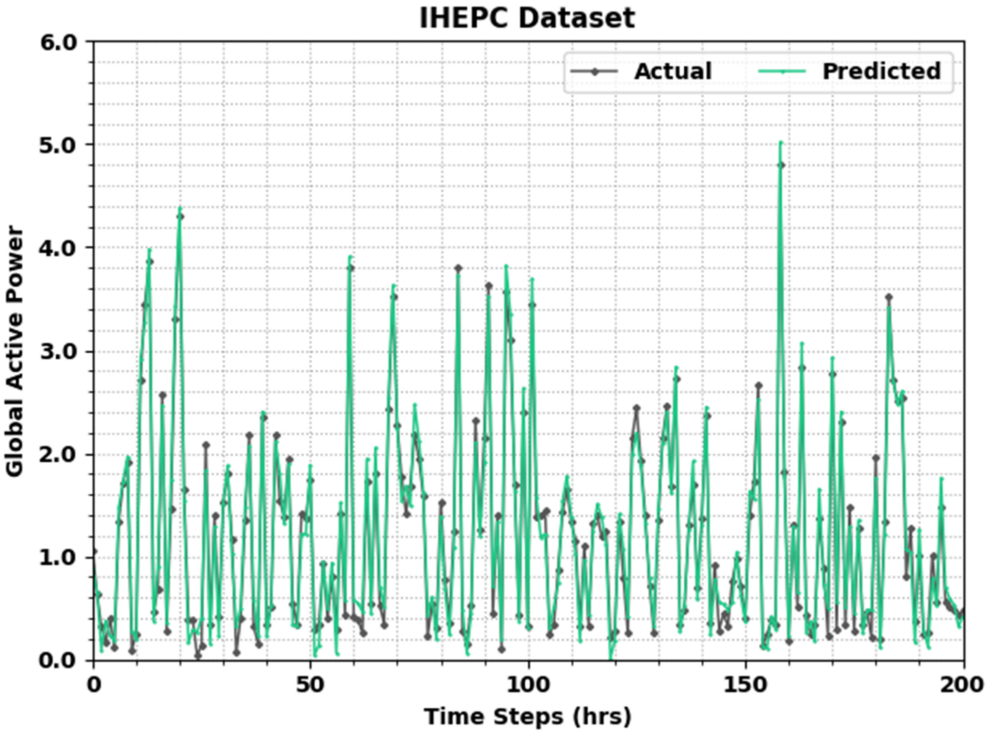

Figure 6. Global active power analysis of AJODL-DSSEM technique under IHEPC dataset.

Table 3 and Figure 7 demonstrate the actual vs. predicted system load of the AJODL-DSSEM algorithm under distinct time steps on the ISO-NE dataset. The outcomes showed that the AJODL-DSSEM methodology predicted the values much closer to the actual values. For instance, with a time step of 20 h and actual value of 0.401, the IHEPC dataset achieved a predicted value of 0.394. Furthermore, with a time step of 40 h and actual value of 0.309, the IHEPC dataset reached a predicted value of 0.322. In addition, with a time step of 80 h and actual value of 0.125, the IHEPC dataset obtained a predicted value of 0.139. In addition, with a time step of 100 h and actual value of 0.406, the IHEPC dataset obtained a predicted value of 0.420.

Table 3. System load analysis of AJODL-DSSEM technique under distinct time steps on ISO-NE dataset.

System Load—ISO-NE Dataset		
Time Steps (h)	**Actual**	**Predicted**
0	0.341	0.345
10	0.190	0.199
20	0.401	0.394
30	0.198	0.201
40	0.309	0.322
50	0.131	0.133
60	0.266	0.278
70	0.285	0.302
80	0.125	0.139
90	0.345	0.365
100	0.406	0.420

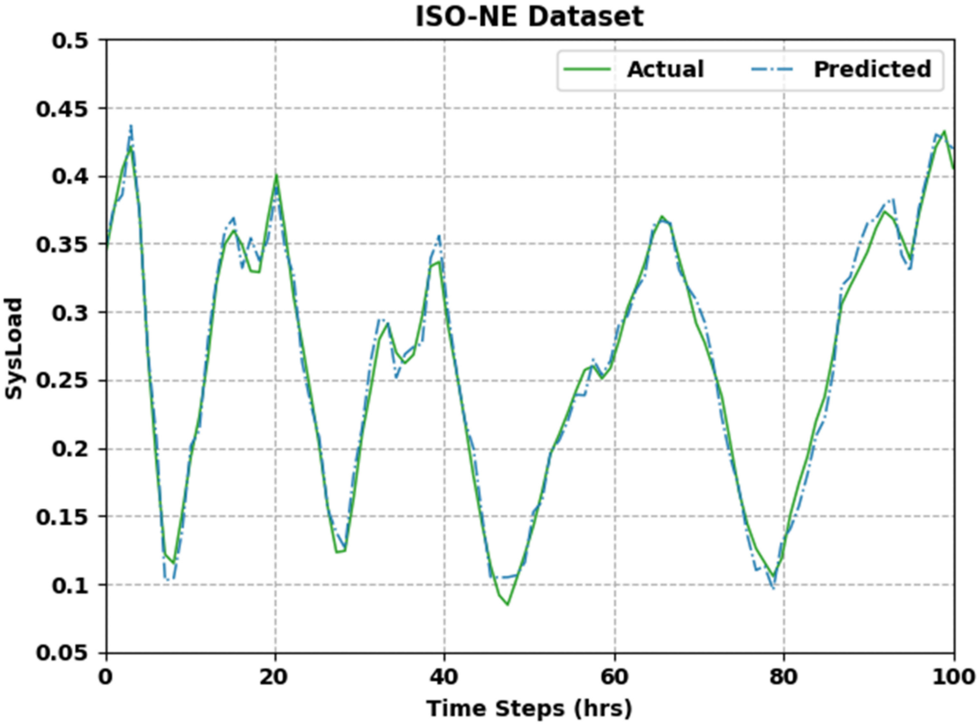

Figure 7. System load analysis of AJODL-DSSEM technique under ISO-NE dataset.

5. Discussion

A comparative study of the AJODL-DSSEM model with recent models: the GRU [25], Bi-GRU [26], LSTM [27], Bi-LSTM [28], CNN-LSTM [29], CNN-GRU [30], and energy-net [31] models on the IHEPC dataset, is portrayed in Table 4. Figure 8 compares the MSE, RMSE, and MAE inspections of the AJODL-DSSEM model on the IHEPC dataset. The figure implied that the IHEPC dataset showed effectual outcomes with minimal values of MSEs, RMSEs, and MAEs. With respect to MSE, the AJODL-DSSEM algorithm obtained a reduced MSE of 0.092, whereas the GRU, Bi-GRU, LSTM, Bi-LSTM, CNN-LSTM, CNN-GRU, and energy-net models obtained increased MSEs of 0.270, 0.251, 0.413, 0.422, 0.431, 0.243, and 0.125, respectively. Moreover, in terms of the RMSE, the AJODL-DSSEM methodology obtained a lower RMSE of 0.303, whereas the GRU, Bi-GRU, LSTM, Bi-LSTM, CNN-LSTM, CNN-GRU, and energy-net methodologies obtained improved RMSEs of 0.518, 0.501, 0.643, 0.647, 0.662, 0.493, and 0.354, respectively.

Table 4. Comparative analysis of AJODL-DSSEM technique with existing approaches under IHEPC dataset.

IHEPC Dataset				
Models	**MSE**	**RMSE**	**MAE**	**MAPE (%)**
GRU [24]	0.270	0.518	0.389	65.200
Bi-GRU [25]	0.251	0.501	0.372	63.900
LSTM [26]	0.413	0.643	0.409	67.800
Bi-LSTM [27]	0.422	0.647	0.392	65.300
CNN-LSTM [28]	0.431	0.662	0.403	50.900
CNN-GRU [29]	0.243	0.493	0.348	46.400
Energy-Net [30]	0.125	0.354	0.287	39.200
AJODL-DSSEM	0.092	0.303	0.262	32.900

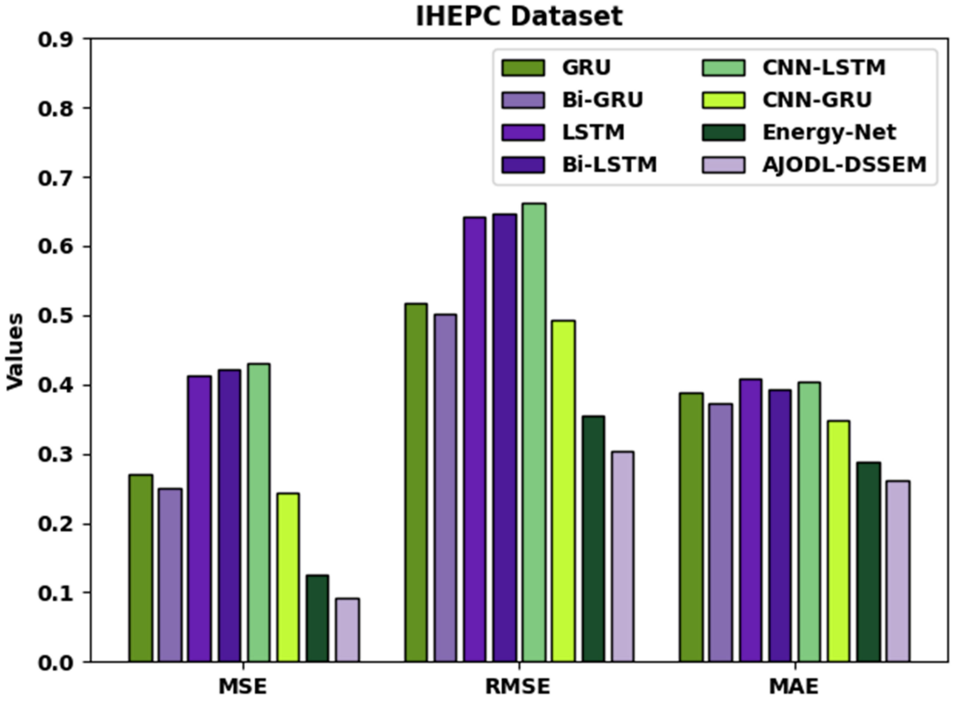

Figure 8. Comparative analysis of AJODL-DSSEM algorithm under IHEPC dataset.

Figure 9 demonstrates the MAPE analysis of the AJODL-DSSEM method on the IHEPC dataset. The figure exposed that the IHEPC dataset showed effectual outcomes with minimal values of MAPE. In terms of the MAPE, the AJODL-DSSEM technique obtained a lower MAPE of 32.9%, whereas the GRU, Bi-GRU, LSTM, Bi-LSTM, CNN-LSTM, CNN-GRU, and energy-net systems reached improved MAPEs of 65.2%, 63.9%, 67.8%, 65.3%, 50.9%, 46.4%, and 39.2%, respectively.

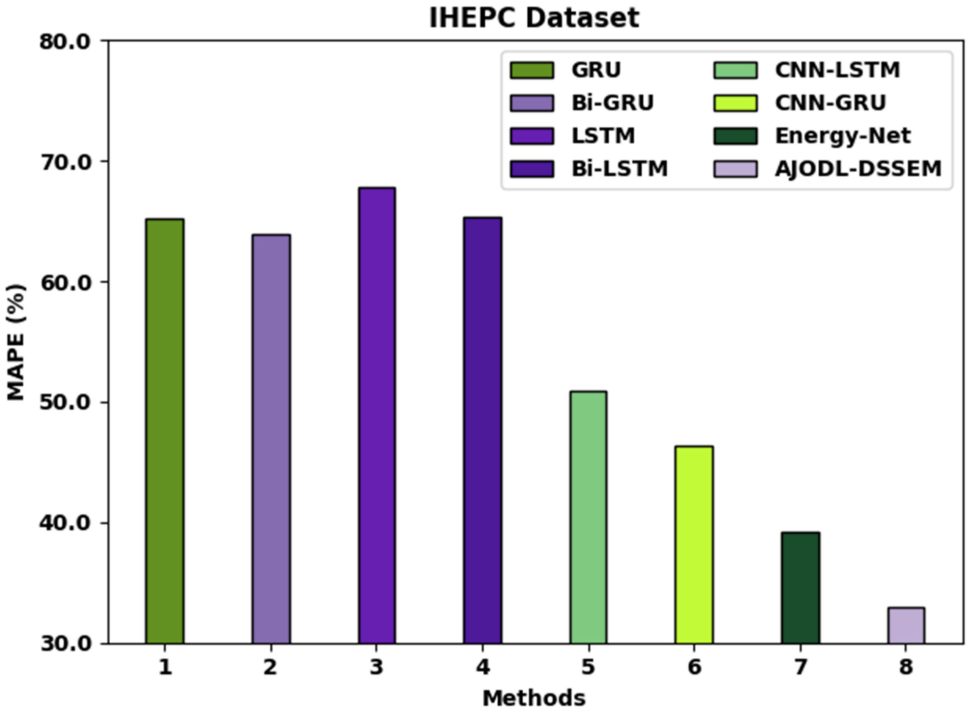

Figure 9. MAPE analysis of AJODL-DSSEM algorithm under IHEPC dataset.

A comparative study of the AJODL-DSSEM algorithm with recent approaches to the ISO-NE dataset is depicted in Table 5. Figure 10 illustrates the MSE, RMSE, and MAE examinations of the AJODL-DSSEM approach on the ISO-NE dataset. The figure exposed that the IHEPC dataset obtained effectual outcomes with lesser values of MSEs, RMSEs, and MAEs. In terms of the MSE, the AJODL-DSSEM algorithm obtained a decreased MSE of 0.208, whereas the GRU, Bi-GRU, LSTM, Bi-LSTM, CNN-LSTM, CNN-GRU, and energy-net approaches obtained increased MSEs of 0.619, 0.501, 0.792, 0.557, 0.456, 0.379, and 0.286, respectively.

Table 5. Comparative analysis of AJODL-DSSEM algorithm with recent methodologies under ISO-NE dataset.

ISO-NE Dataset				
Models	**MSE**	**RMSE**	**MAE**	**MAPE (%)**
GRU [24]	0.619	0.794	0.513	49.200
Bi-GRU [25]	0.501	0.713	0.461	60.900
LSTM [26]	0.792	0.891	0.552	65.400
Bi-LSTM [27]	0.557	0.746	0.534	62.300
CNN-LSTM [28]	0.456	0.681	0.434	40.900
CNN-GRU [29]	0.379	0.617	0.488	34.100
Energy-Net [30]	0.286	0.535	0.414	29.300
AJODL-DSSEM	0.208	0.456	0.384	23.700

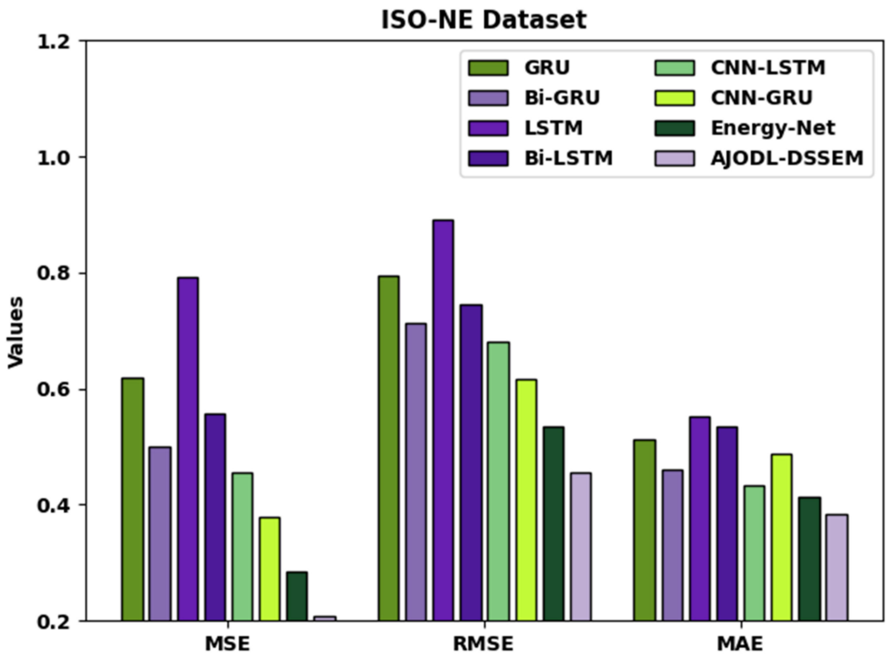

Figure 10. Comparative analysis of AJODL-DSSEM technique under ISO-NE dataset.

With respect to the RMSE, the AJODL-DSSEM system obtained an RMSE of 0.384, whereas the GRU, Bi-GRU, LSTM, Bi-LSTM, CNN-LSTM, CNN-GRU, and energy-net techniques obtained improved RMSEs of 0.794, 0.713, 0.891, 0.746, 0.681, 0.617, and 0.535, respectively.

Figure 11 illustrates the MAPE inspection of the AJODL-DSSEM approach on the ISO-NE dataset. The figure showed that the IHEPC dataset outperformed effectual outcomes with minimal values of MAPEs. In terms of the MAPE, the AJODL-DSSEM system obtained a decreased MAPE of 23.7%, whereas the GRU, Bi-GRU, LSTM, Bi-LSTM, CNN-LSTM, CNN-GRU, and energy-net methodologies obtained enhanced MAPEs of 49.2%, 60.9%, 65.4%, 62.3%, 40.9%, 34.1%, and 29.3%, respectively.

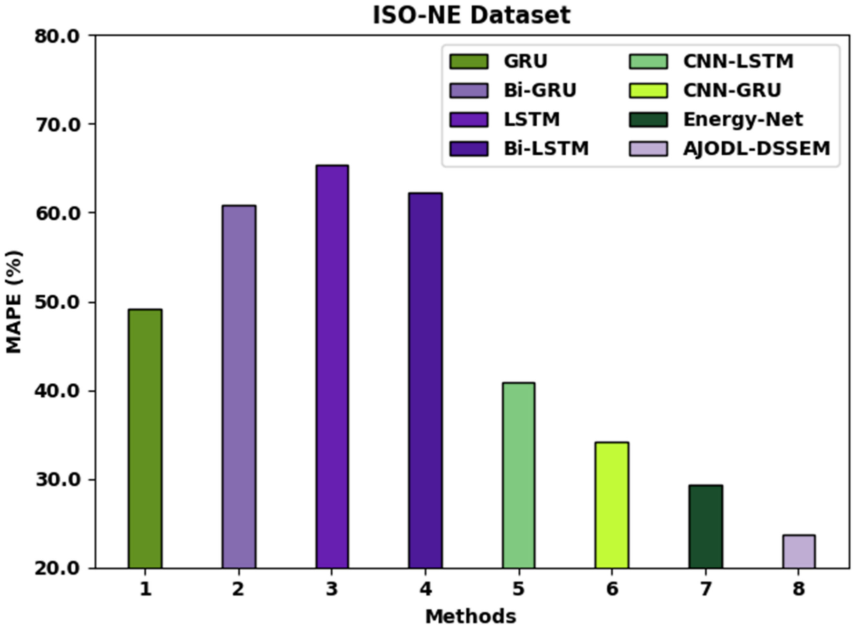

Figure 11. MAPE analysis of AJODL-DSSEM technique under ISO-NE dataset.

From the detailed results and discussion, the AJODL-DSSEM model resulted in enhanced prediction outcomes over existing models.

6. Conclusions

In this study, a novel AJODL-DSSEM model was established for the prediction of energy in the smart city environment. The proposed AJODL-DSSEM model mainly accomplished data preprocessing at the initial stage to normalize the data. Further, the AJODL-DSSEM model involved a CNN-ABLSTM model for the prediction of energy. Lastly, the AJO algorithm was applied for the hyperparameter adjustment of the CNN-ABLSTM model. The experimental validation of the proposed AJODL-DSSEM model was tested using two open access datasets, namely the IHEPC and ISO-NE datasets. The comparative study reported the enhanced outcomes of the AJODL-DSSEM model over recent approaches. Thus, the AJODL-DSSEM model can be employed for energy-management-related decision making in the real-time smart city environment. The proposed model can be useful for optimal resource allocation in the smart city environment. It can also assist stakeholders and policymakers in the design of energy solutions for smart cities by providing strategies for the effective modeling and management of energy systems. It is helpful for the stakeholders to understand urban dynamics and evaluate the influence of energy policy alternatives. In the future, feature selection and outlier detection approaches can be integrated into the proposed model to boost the predictive performance. Moreover, the proposed model can be tested on real-time large-scale datasets in the future.

Author Contributions: Conceptualization, A.A.-Q. and H.A.; methodology, J.S.A.; software, H.M.; validation, A.A.-Q., N.N. and L.A.A.; formal analysis, M.A.D.; investigation, F.N.A.-W.; resources, H.A.; data curation, M.A.D.; writing—original draft preparation, F.N.A.-W. and J.S.A.; writing—review and editing, L.A.A.; visualization, M.A.-S.; supervision, M.A.D.; project administration, M.A.D.; funding acquisition, H.A. All authors have read and agreed to the published version of the manuscript.

Funding: The authors extend their appreciation to the Deanship of scientific research at King Khalid University for funding this work through the Large Groups Project under the grant number (42/43), the Princess Nourah bint Abdulrahman University Researchers supporting project number (PNURSP2022R303), Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia. The authors would like to thank the Deanship of scientific research at Umm Al-Qura University for supporting this work by grant code: (22UQU4340237DSR18).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data sharing is not applicable to this article, as no datasets were generated during the current study.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Calvillo, C.F.; Sánchez-Miralles, A.; Villar, J. Energy management and planning in smart cities. *Renew. Sustain. Energy Rev.* **2016**, *55*, 273–287. [CrossRef]
2. Liu, Y.; Yang, C.; Jiang, L.; Xie, S.; Zhang, Y. Intelligent edge computing for IoT-based energy management in smart cities. *IEEE Network* **2019**, *33*, 111–117. [CrossRef]
3. Mahapatra, C.; Moharana, A.K.; Leung, V. Energy management in smart cities based on internet of things: Peak demand reduction and energy savings. *Sensors* **2017**, *17*, 2812. [CrossRef]
4. Sirohi, P.; Al-Wesabi, F.N.; Alshahrani, H.M.; Maheshwari, P.; Agarwal, A.; Dewangan, B.K.; Hilal, A.M.; Choudhury, T. Energy-efficient cloud service selection and recommendation based on qos for sustainable smart cities. *Appl. Sci.* **2021**, *11*, 9394. [CrossRef]
5. Alsubaei, F.S.; Al-Wesabi, F.N.; Hilal, A.M. Deep learning-based small object detection and classification model for garbage waste management in smart cities and iot environment. *Appl. Sci.* **2022**, *12*, 2281. [CrossRef]
6. Al-Qarafi, A.; Alrowais, F.; Alotaibi, S.; Nemri, N.; Al-Wesabi, F.N.; Duhayyim, A.; Marzouk, R.; Othman, M.; Al-Shabi, M. Optimal machine learning based privacy preserving blockchain assisted internet of things with smart cities environment. *Appl. Sci.* **2022**, *12*, 5893.
7. Kamienski, C.A.; Borelli, F.F.; Biondi, G.O.; Pinheiro, I.; Zyrianoff, I.D.; Jentsch, M. Context design and tracking for IoT-based energy management in smart cities. *IEEE Internet Things J.* **2017**, *5*, 687–695. [CrossRef]

8. Petrović, N.; Roblek, V.; Nejković, V. Mobile Applications and Services for Next-Generation Energy Management in Smart Cities. *Sustain. Dev.* **2020**, *1*, 2.
9. Laroui, M.; Dridi, A.; Afifi, H.; Moungla, H.; Marot, M.; Cherif, M.A. Energy management for electric vehicles in smart cities: A deep learning approach. In Proceedings of the 2019 15th International Wireless Communications & Mobile Computing Conference (IWCMC), IEEE, Tangier, Morocco, 24–28 June 2019; pp. 2080–2085.
10. Shreenidhi, H.S.; Ramaiah, N.S. A two-stage deep convolutional model for demand response energy management system in IoT-enabled smart grid. *Sustain. Energy Grids Netw.* **2022**, *30*, 100630.
11. Lotfi, M.; Almeida, T.; Javadi, M.S.; Osório, G.J.; Monteiro, C.; Catalão, J.P. Coordinating energy management systems in smart cities with electric vehicles. *Appl. Energy* **2022**, *307*, 118241. [CrossRef]
12. Elsisi, M.; Tran, M.Q.; Mahmoud, K.; Lehtonen, M.; Darwish, M.M. Deep learning-based industry 4.0 and Internet of Things towards effective energy management for smart buildings. *Sensors* **2021**, *21*, 1038. [CrossRef] [PubMed]
13. Vázquez-Canteli, J.R.; Ulyanin, S.; Kämpf, J.; Nagy, Z. Fusing TensorFlow with building energy simulation for intelligent energy management in smart cities. *Sustain. Cities Soc.* **2019**, *45*, 243–257. [CrossRef]
14. Xiaoyi, Z.; Dongling, W.; Yuming, Z.; Manokaran, K.B.; Antony, A.B. IoT driven framework based efficient green energy management in smart cities using multi-objective distributed dispatching algorithm. *Environ. Impact Assess. Rev.* **2021**, *88*, 106567. [CrossRef]
15. Ullah, I.; Hussain, I.; Uthansakul, P.; Riaz, M.; Khan, M.N.; Lloret, J. Exploiting multi-verse optimization and sine-cosine algorithms for energy management in smart cities. *Appl. Sci.* **2020**, *10*, 2095. [CrossRef]
16. Kim, D.; Kwon, D.; Park, L.; Kim, J.; Cho, S. Multiscale LSTM-based deep learning for very-short-term photovoltaic power generation forecasting in smart city energy management. *IEEE Syst. J.* **2020**, *15*, 346–354. [CrossRef]
17. Hrnjica, B.; Mehr, A.D. Energy demand forecasting using deep learning. In *Smart Cities Performability, Cognition, & Security*; Springer: Cham, Switzerland, 2020; pp. 71–104.
18. Li, X.; Liu, H.; Wang, W.; Zheng, Y.; Lv, H.; Lv, Z. Big data analysis of the internet of things in the digital twins of smart city based on deep learning. *Future Gener. Comput. Syst.* **2022**, *128*, 167–177. [CrossRef]
19. Siami-Namini, S.; Tavakoli, N.; Namin, A.S. The performance of LSTM and BiLSTM in forecasting time series. In Proceedings of the 2019 IEEE International Conference on Big Data (Big Data) IEEE, Los Angeles, CA, USA, 9–12 December 2019; pp. 3285–3292.
20. Shan, L.; Liu, Y.; Tang, M.; Yang, M.; Bai, X. CNN-BiLSTM hybrid neural networks with attention mechanism for well log prediction. *J. Pet. Sci. Eng.* **2021**, *205*, 108838. [CrossRef]
21. Chou, J.S.; Truong, D.N. A novel metaheuristic optimizer inspired by behavior of jellyfish in ocean. *Appl. Math. Comput.* **2021**, *389*, 125535. [CrossRef]
22. Abdel-Basset, M.; Mohamed, R.; Chakrabortty, R.K.; Ryan, M.J.; El-Fergany, A. An improved artificial jellyfish search optimizer for parameter identification of photovoltaic models. *Energies* **2021**, *14*, 1867. [CrossRef]
23. Individual Household Electric Power Consumption Data Set. Available online: https://archive.ics.uci.edu/ml/datasets/individual+household+electric+power+consumption (accessed on 12 March 2022).
24. England, I.N. Available online: https://www.iso-ne.com/system-planning/system-forecasting/load-forecast/ (accessed on 12 March 2022).
25. Han, T.; Muhammad, K.; Hussain, T.; Lloret, J.; Baik, S.W. Efficient Deep Learning Framework for Intelligent Energy Management in IoT Networks. *IEEE Internet Things J.* **2020**, *8*, 3170–3179. [CrossRef]
26. Lv, P.; Liu, S.; Yu, W.; Zheng, S.; Lv, J. EGA-STLF: A Hybrid Short-Term Load Forecasting Model. *IEEE Access* **2020**, *8*, 31742–31752. [CrossRef]
27. Tan, M.; Yuan, S.; Li, S.; Su, Y.; Li, H.; He, F. Ultra-short-term industrial power demand forecasting using LSTM based hybrid ensemble learning. *IEEE Trans. Power Syst.* **2019**, *35*, 2937–2948. [CrossRef]
28. Chitalia, G.; Pipattanasomporn, M.; Garg, V.; Rahman, S.J.A.E. Robust short-term electrical load forecasting framework for commercial buildings using deep recurrent neural networks. *Appl. Energy* **2020**, *278*, 115410. [CrossRef]
29. Sajjad, M.; Khan, Z.A.; Ullah, A.; Hussain, T.; Ullah2, W.; Lee, M.Y.; Baik, S.W. A novel CNN-GRU-based hybrid approach for short-term residential load forecasting. *IEEE Access* **2020**, *8*, 143759–143768. [CrossRef]
30. Kim, T.-Y.; Cho, S.-B.J.E. Predicting residential energy consumption using CNN-LSTM neural networks. *Energy* **2019**, *182*, 72–81. [CrossRef]
31. Abdel-Basset, M.; Hawash, H.; Chakrabortty, R.K.; Ryan, M. Energy-net: A deep learning approach for smart energy management in iot-based smart cities. *IEEE Internet Things J.* **2021**, *8*, 12422–12435. [CrossRef]