Forecasting Electricity Consumption based on Nested LSTM and Attention Mechanism Approach with Cuckoo Search Optimizer

Dean Tirkaamiana & Nurhadi Siswanto
Industrial and Systems Engineering Department
Faculty of Industrial Technology and Systems Engineering
Institut Teknologi Sepuluh Nopember
Surabaya, Indonesia
deantirka@gmail.com, siswanto@ie.its.ac.id

Abstract

This paper presents electricity forecasting by integration of Nested Long Short Term Memory (NLSTM) as the another type of LSTM, Attention Mechanism (AM) as the technique for Neural Network and optimized by Cuckoo Search Optimizer (CSO) as the metaheuristic. It is important because of the electricity forecasting issue. Electricity is the crucial energy, due to the many sectors which is require it. Estimation of electricity consumption can enhance the effectiveness of generation and supply. The shortage and outage of electricity is the thing that should be avoided through the electricity forecasting. In order to create better forecasting result, the forecasting by NLSTM-AM will be combined. NLSTM-AM has several parameters which influence the network performance. Those parameters will be optimized by CSO. To discover the performance of this proposed method, the RMSE of this method will be compared by other algorithms. In this paper there are two datasets which will be used, those are Taiwan electricity consumption and Turkey electricity consumption. It is proved that NLSTM-AM-CSO has the smallest RMSE rather than other algorithms.

Keywords
Forecasting, Nested Long-Short Term Memory, Attention Mechanism, Genetic Algorithm, Cuckoo Search Optimizer.

1. Introduction
Electricity is the crucial energy, due to the many sectors which is require it. The shortage and outage of electricity is the thing that should be avoided. In order to prevent that, the generation and supply of electricity should be effective. Estimation of electricity consumption can enhance the effectiveness of generation and supply. The more advance prediction, the more important it is in planning, analysis and operation of power system to ensure an uninterrupted, reliable, secure and economic supply of electricity (Bianco *et al.* 2010). Also, electricity forecasting assist to make decision related to the purchasing, load switching and infrastructure development (Singh *et al.* 2013). For specific period such as short-term, short-term electricity demand forecasting has big importance for daily control, scheduling, operation, planning, and stability of the power system (Chapagain 2020). Narayan and Hipel (2017) conducted forecasting for short-term load electricity which represented as time series by Long-Short Term Memory (LSTM) approach. LSTM network, an extension of Recurrent Neural Network (RNN). In LSTM approach, there is LSTM cell as the solution of RNN problem related to the gradient vanishing problem. Much research about LSTM have been developed, and it leads to the several types of LSTM (Liu *et al.* 2017). One of them is Nested LSTM (NLSTM). NLSTM is an extension of the original LSTM structure (Yan *et al.* 2022). Not only the several types of LSTM as a proof of the improvement of LSTM research development, but also the integration of LSTM with other mechanism such as Attention Mechanism (AM). Ran *et al.* (2019) conduct research with combine LSTM with AM, to forecast traffic prediction. There are several parameters in NLSTM, which should be determined before running the model. Those parameters should be optimized to create the better model. Gandomi *et al.* (2013) stated that metaheuristic algorithm is influential tools for modeling and optimization. These algorithms imitate effective processes in nature such as chemical and physical processes, and biological system. One of algorithms is Cuckoo Search Optimizer (CSO). The obligate brood parasitism of some cuckoo species, which involves them incubating their eggs in host bird nests, served as the model for the CSO algorithm. Wang *et al.* (2017) conducted research related to the comparison of

CSO and Particle Swarm Optimizer. Also, Ghosh *et al.* (2016) proved that CSO algorithm has better performance than Genetic Algorithm (GA). Therefore, in this research, the electricity consumption will be forecasted by NLSTM and Attention Mechanism optimized by CSO algorithm.

2. Literature

2.1 Long Short-Term Memory (LSTM)

Standard Recurrent Neural Network is not able to span more than 5-10 time steps due to the back-propagated error signals that lead to either develop and decrease with every time step. So, over many time steps, the error usually explodes or vanishes. LSTM as the gradient-based method is the vanishing problem (Staudemeyer and Morris 2019). LSTM architecture contains memory block, it is a group of recurrently connected sub-networks. Memory block has a role to preserve information over time and control the information flow over non-linear gating units (Houdt *et al.* 2020). There are three gates with their own assignment which are input, forget, and output gates. Input gates regulate the signals from the network to the memory cell by grading them properly. Forget gates can understand which should be the internal state of the memory cell when the stored information is no longer required. Output gates has ability to find out how to control access the content of memory cell.

2.2 Nested Long Short-Term Memory (NLSTM)

LSTM is extended as NLSTM, which nests another LSTM cell in one LSTM cell (Yan *et al.* 2022). In NLSTM there is a perception that memories which are not related at the current time-step might still be worth remembering. Along with this perception, there is temporal hierarchy of memories. (Moniz and Krueger 2017). The architecture of NLSTM is shown in Figure 1.

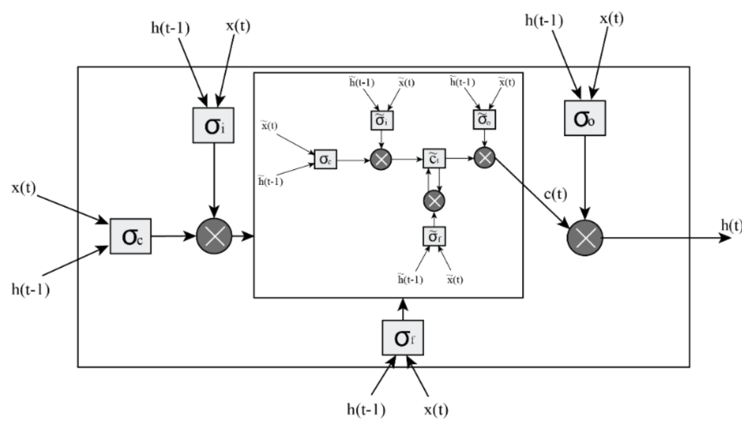

Figure 1. NLSTM architecture

The mathematical expression of outer NLSTM as follows,

$$\boldsymbol{i_t} = \boldsymbol{\sigma}(\boldsymbol{W_{ih}h_{t-1}} + \boldsymbol{W_{ix}x_t} + \boldsymbol{b_i}), \quad (1)$$

$$\boldsymbol{f_t} = \boldsymbol{\sigma}\left(\boldsymbol{W_{fh}h_{t-1}} + \boldsymbol{W_{fx}x_t} + \boldsymbol{b_f}\right), \quad (2)$$

$$\boldsymbol{\tilde{c}_t} = \boldsymbol{tanh}(\boldsymbol{W_{\tilde{c}h}h_{t-1}} + \boldsymbol{W_{\tilde{c}x}x_t} + \boldsymbol{b_{\tilde{c}}}), \quad (3)$$

$$\boldsymbol{o_t} = \boldsymbol{\sigma}(\boldsymbol{W_{oh}h_{t-1}} + \boldsymbol{W_{ox}x_t} + \boldsymbol{b_o}), \text{ and} \quad (4)$$

$$\boldsymbol{c_t} = \boldsymbol{\widetilde{h}_t} \quad (5)$$

$_t$ = Current state,

$_{t-1}$ = Previous state,

x = Input,

h = Recurrent information,

o = Output,

W = Weight,

b = Bias,

i_t = Input gate,

f_t = Forget gate,

c_t = Cell state, and

$\tilde{c}_t$ = Cell input state.

the equation in the inner NLSTM as follows:

$$\tilde{\boldsymbol{h}}_{t-1} = \boldsymbol{f}_t \cdot \boldsymbol{c}_{t-1} \tag{6}$$

$$\tilde{\boldsymbol{x}}_t = \boldsymbol{i}_t \cdot \tilde{\boldsymbol{\sigma}}_c\ (\boldsymbol{tanh}(\boldsymbol{W}_{\tilde{c}h}\boldsymbol{h}_{t-1} + \boldsymbol{W}_{\tilde{c}x}\boldsymbol{x}_t + \boldsymbol{b}_{\tilde{c}}), \tag{7}$$

$$\tilde{\boldsymbol{f}}_t = \tilde{\boldsymbol{\sigma}}_f(\widetilde{\boldsymbol{W}}_{fx}\tilde{\boldsymbol{x}}_t + \widetilde{\boldsymbol{W}}_{fh}\tilde{\boldsymbol{h}}_{t-1} + \tilde{\boldsymbol{b}}_f), \tag{8}$$

$$\tilde{\boldsymbol{i}}_t = \tilde{\boldsymbol{\sigma}}_i(\widetilde{\boldsymbol{W}}_{ix}\ \tilde{\boldsymbol{x}}_t + \widetilde{\boldsymbol{W}}_{ih}\tilde{\boldsymbol{h}}_{t-1} + \tilde{\boldsymbol{b}}_i), \tag{9}$$

$$\tilde{\boldsymbol{c}}_t = \tilde{\boldsymbol{f}}_t \cdot \tilde{\boldsymbol{c}}_{t-1} + \tilde{\boldsymbol{i}}_t \cdot \tilde{\boldsymbol{\sigma}}_c\ (\widetilde{\boldsymbol{W}}_{cx}\tilde{\boldsymbol{x}}_t + \widetilde{\boldsymbol{W}}_{ch}\tilde{\boldsymbol{h}}_{t-1} + \tilde{\boldsymbol{b}}_c), \tag{10}$$

$$\tilde{\boldsymbol{O}}_t = \tilde{\boldsymbol{\sigma}}_o(\widetilde{\boldsymbol{W}}_{ox}\tilde{\boldsymbol{x}}_t + \widetilde{\boldsymbol{W}}_{oh}\tilde{\boldsymbol{h}}_{t-1} + \tilde{\boldsymbol{b}}_o), \text{ and} \tag{11}$$

$$\tilde{\boldsymbol{h}}_t = \tilde{\boldsymbol{O}}_t \cdot \tilde{\boldsymbol{\sigma}}_h(\tilde{\boldsymbol{c}}_t). \tag{12}$$

$$\tilde{\boldsymbol{h}}_{t-1} = \boldsymbol{f}_t \cdot \boldsymbol{c}_{t-1} \tag{13}$$

$$\tilde{\boldsymbol{x}}_t = \boldsymbol{i}_t \cdot \tilde{\boldsymbol{\sigma}}_c\ (\boldsymbol{tanh}(\boldsymbol{W}_{\tilde{c}h}\boldsymbol{h}_{t-1} + \boldsymbol{W}_{\tilde{c}x}\boldsymbol{x}_t + \boldsymbol{b}_{\tilde{c}}), \tag{14}$$

And c_t in the outer memory cell becomes:

$$\boldsymbol{c}_t = \tilde{\boldsymbol{h}}_t, \tag{15}$$

$$\boldsymbol{h}_t = \boldsymbol{o}_t \cdot \mathbf{tanh}(\boldsymbol{c}_t), \tag{16}$$

2.3 Attention Mechanism

Humans do not typically process all the information they are presented with at once, which is a crucial characteristic of perception. Rather, when and when it is necessary, people prefer to selectively focus on a portion of the information while ignoring other perceptible information at the same time (Niu *et al.* 2021). Attention mechanism is the technique with this similar perception. Attention mechanism is a basic method which has ability for encoding sequence data according to each element's importance score. (Hu 2019).

There are two types of Attention Mechanism which are global and local. Global attention is used due to the hidden states of encoder when the context vector c_t are examined (Luong *et al. 2015)*. The equation of c_t, as follow (Bahdanau *et al.* 2014):

$$c_t = \sum_s a_{ts}\bar{h}_s \tag{17}$$

A variable-length alignment vector (a_t) has a size that equal to the number of time steps on the source side. a_t is obtained by conduct a comparison of current target hidden state (h_t) and each source hidden state ($\bar{h}_s$) as below:

$$a_t(s) = \text{align}(h_t, \bar{h}_s), \tag{18}$$

$$= \frac{\exp\left(\mathrm{score}(h_t, \bar{h}_s)\right)}{\sum_{s'} \exp\left(\mathrm{score}(h_t, \bar{h}_{s'})\right)} \tag{19}$$

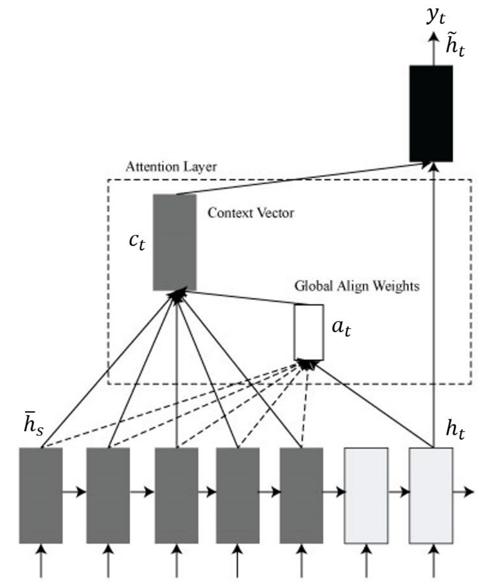

Figure 2. Global attention model

“Score” is cited as a content-based function, it is considered three different alternatives such as:

$$\mathrm{score}(h_t, \bar{h}_s) = \begin{cases} h_t^{\top} \bar{h}_s & \text{dot} \quad (21) \\ h_t^{\top} W_a \bar{h}_s & \text{general} \quad (22) \\ v_a^{\top} \tanh\left(W_a \bar{h}_{[h_t, \bar{h}_s]}\right) & \text{concat} \quad (23) \end{cases}$$

2.4. Cuckoo Search Optimizer

The CS algorithm was inspired by the obligate brood parasitism of some cuckoo species by laying their eggs in the nests of host birds. Some cuckoos have involved in such a way that female parasitic cuckoos can imitate various colours and patterns of the eggs of a few chosen host species (Yang & Deb 2009). The mechanism of CSO is showed in Figure 3 based on Shair (2015). The population is initialized which is reflected by cuckoos having eggs. Then, it places eggs in distinct area and several eggs are detected and eliminated. This process is modelled by Levi Flight in the mechanism of CSO metaheuristic algorithm process. Then, there is two considerations if population is less than max value, the cuckoo in will be eliminated. But, if the population is less than max value then survival eggs in nest. This process is reflected by comparison of fitness value in the CSO metaheuristic algorithm process. Next, there is consideration if the condition or if generation is fulfilled then this algorithm process is finish. But if it has not fulfilled yet then this process will start from beginning.

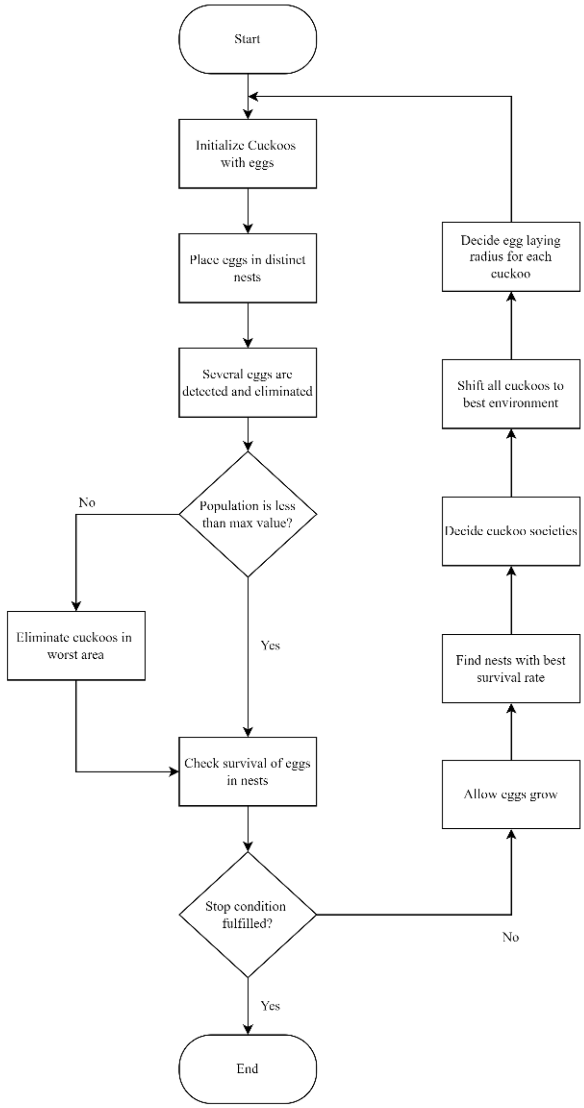

Figure 3. Flowchart of CSO

3. Methods

3.1 Data Preprocessing

The data are processed by normalization which transform those to be in range [0,1]. The equation as follows,

$$x_{(normalized)} = \frac{x - \min(x)}{\max(x) - \min(x)} \tag{24}$$

where,
$x =$ Original value,
$\min(x)$ = Minimal value of all the attribute values, and
$\max(x)$ = Maximal value of all the attribute values.

3.2 The Proposed Forecasting Algorithm

The integration of NLSTM and attention mechanism is applied in this study to make forecasting. The performance of NLSTM-AM will be evaluated based on the Root Mean Squared Error. The equation as follows,

$$RMSE = \sqrt{\sum_{1}^{n} \frac{(T_i - Y_i)^2}{n}}, \tag{25}$$

where,
T_i = Actual value
Y_i = Prediction value
n = Number of value

Parameter of NLSTM should be determined properly with the purpose of having better performance. There are five parameters which should be optimized such as number of NLSTM cell, window size, learning rate, dropout rate, and batch size. The optimization will be assisted by proposed CSO. The name of proposed CSO is Fluctuate CSO.

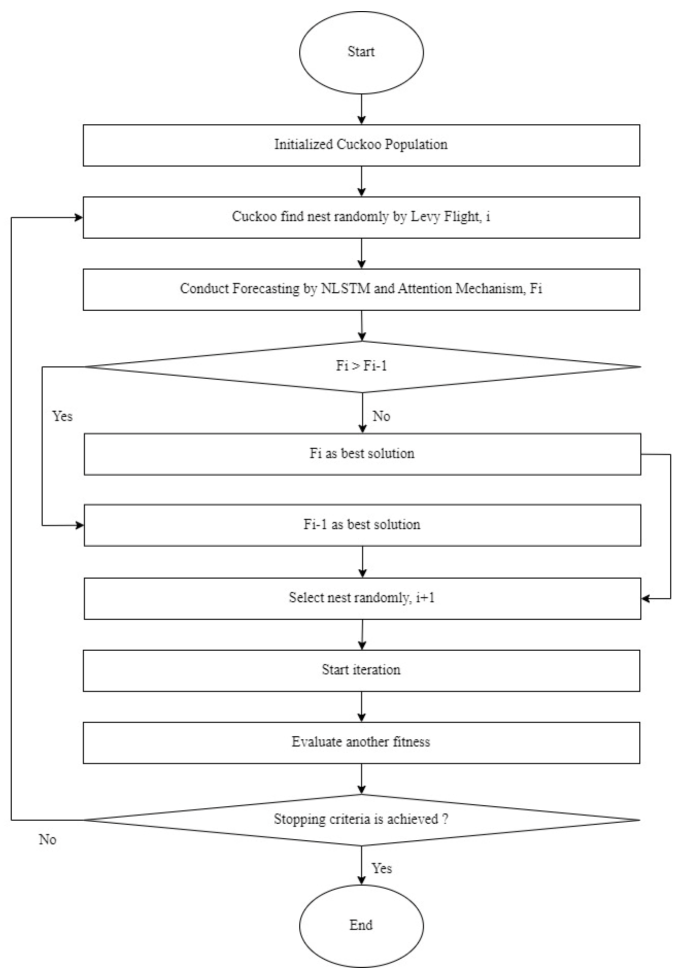

Figure 4. Flowchart of Fluctuate CSO

The procedure as follows:

(1) Initialize Cuckoo population

At the beginning of the network parameter optimization, a prospect nest matrix is randomly created with size N_{pop} x N_{var} is created. Upper and lower limits are determined as the limitation of these values for every cuckoo in different iterations. Real cuckoo has behavior that they lay their eggs within maximum distance from their nest. The maximum range is known as Egg Laying Radius (ERL).

(2) Find nest randomly by Lévy flight

The cuckoo randomly chooses nest using Levy flight. Levy flight is performed by the equation as follows:

$$x_i^{(t+1)} = x_i^{(t)} + \alpha \oplus \text{Lévy}(\lambda), \tag{26}$$

where $\alpha > 0$ and α is the step size which will be determined in this research. In the past, α is set up as 0.01. Lévy flight basically provides a random walk and the random step length is drawn by a Lévy distribution with the equation as follows:

$$\text{Lévy} \sim t^{-\lambda}, \tag{27}$$

where,
t = iteration order (for this stage, set t = 0)
λ = random number between 0 and 1

In this research, lambda value is the parameter of levy flight
Then, from the cuckoo algorithm, it can provide the network parameters of NLSTM-attention mechanism model including NLSTM cell, window size, learning rate, dropout rate, and batch size. After this step, the NLSTM-Fluctuate CSO stage will be performed.

(3) Evaluate fitness

RMSE from first stage is considered as the fitness solution (F_i) in CSO. Compare the fitness solution with those have been obtained in the previous iteration (F_{i-1}). The minimum value of fitness (F) will be the best nest.

(4) Start iteration

The fraction of (p_a) of worst nest is the probability that the host find out the cuckoo is not its egg. And as it is known if the host find out, then the host will eliminate the cuckoo from his nest. For dynamic CSO, the probability will be different for each iteration with range [0,0.025]. From this the best nest will contain the cuckoo which still survive and put its eggs and generate new nest by Levy Flight. Then, the NLSTM-attention mechanism is performed is repeated with different parameters.

(5) Evaluate another fitness

From step 5, we can obtain new nest and also new fitness solution and in this step, it is considered as F_{i+1}. Then, compare between F and F_{i+1}. The F will be replaced if F_{i+1} has smaller value. Update the best nest based on the best F.

(6) Repeat the CSO process.

This process is repeated until stopping criteria is achieved. In this study, the stopping criterion is the number of iterations. When step 4 is repeated, then F will be F_{i-1}.

4. Data Collection

There are two datasets which are used with the ratio 80:20 for training and testing dataset in this research as follow,

A) Dataset 1 - Individual household electric voltage data in France
It is shown as in figure 6, from 2000 data, 1200 data is selected as the data training while the rest which is 800 data as the data testing.

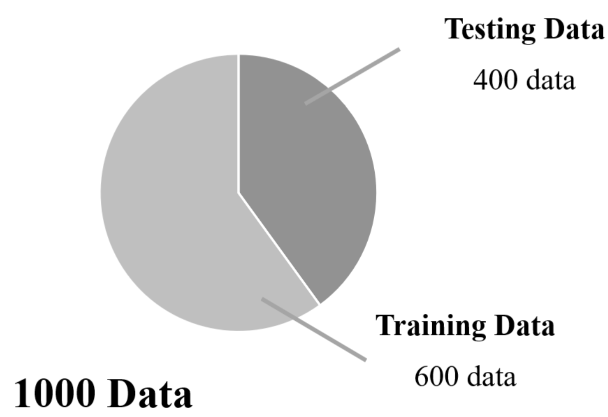

Figure 5. Training and testing for individual household electric consumption data

B) Dataset 2 - Turkey hourly energy consumption data
Figure 7 represents the allocation of the Turkey hourly energy consumption data. From the 2000 data, 1080 data is selected as the training data, while the rest which is 720 data as the testing data.

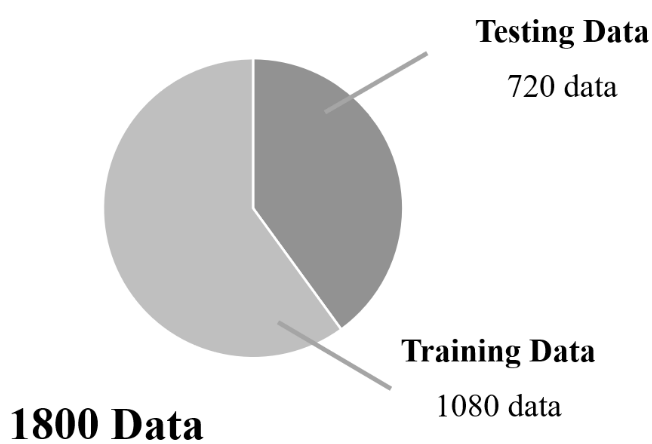

Figure 6. Training and testing for Turkey hourly energy consumption data

5. Results and Discussion

With the purpose to prove the NLSTM-AM-Fluctuate CSO is better performance, the RMSE of this algorithm is compared with several algorithms which are LSTM, NLSTM, NLSTM-AM, NLSTM-AM-Particle Swarm Optimization (PSO), NLSTM-AM-GA, NLSTM-AM-CSO. All algorithms are replicated with 30 times. The RMSE for 30 replications of Dataset 1 as follows,

Table 1. RMSE values of Dataset 1 by forecasting algorithms

Rep	LSTM	NLSTM	NLSTM-AM	NLSTM-AM-PSO	NLSTM-AM-GA	NLSTM-AM-CSO	NLSTM-AM-Fluctuate CSO
1	10.20	10.22	10.2	4.1	5.9	3.4	3.7
2	10.20	10.22	7.7	4.4	3.4	3.7	3.5
3	10.22	10.17	6.9	7.7	7.5	4.5	4.7
4	10.22	10.21	10.2	4.2	4.7	7.9	3.7
5	10.21	10.22	10.2	3.5	5.3	6.7	5.4
6	10.20	10.21	10.2	4.2	4.2	3.6	4.3
7	10.90	10.19	10.0	4.3	6.7	4.0	4.5
8	10.21	10.22	6.4	3.5	4.0	3.6	3.8
9	10.21	10.21	10.2	3.5	8.7	3.8	4.6
10	10.22	10.22	10.0	3.7	3.4	3.5	7.7
11	10.20	10.14	5.8	3.9	4.0	4.0	4.1
12	10.20	10.20	6.5	5.0	4.3	7.0	3.9
13	10.21	10.22	10.2	3.8	4.0	5.1	3.7
14	10.20	10.22	7.2	4.9	6.0	3.6	3.4
15	10.22	10.18	10.2	3.7	4.0	7.7	3.6
16	10.20	10.18	9.9	8.2	3.5	6.3	5.9
17	10.19	10.22	10.2	5.1	3.4	3.4	5.4
18	10.21	10.20	10.2	6.3	6.6	3.9	3.5
19	10.21	10.22	10.2	3.5	7.1	7.0	4.2
20	10.20	10.22	10.2	4.1	3.7	5.3	3.4
21	10.21	10.21	10.2	3.5	3.4	3.6	3.5
22	10.21	10.28	10.2	4.1	7.1	5.4	5.4
23	10.20	10.22	10.2	3.5	6.3	3.4	3.9
24	10.21	10.20	9.9	4.4	4.1	4.0	3.4
25	10.21	10.21	10.2	4.9	4.2	3.4	3.5
26	10.21	10.21	10.2	3.8	8.0	3.8	4.1
27	10.21	10.22	10.2	3.5	4.5	4.1	4.1
28	10.21	10.21	10.1	3.4	4.5	4.3	3.9
29	10.20	10.18	5.6	4.3	4.3	4.3	5.6
30	10.20	10.22	5.6	3.6	8.7	3.7	3.4
Average	**10.23**	**10.21**	**9.3**	**4.3**	**5.2**	**4.6**	**4.2**

From table 1, it is shown the RMSE of testing data average using LSTM is 10.2, for NLSTM is 10, for NLSTM-AM is 10.2, for NLSTM-AM-PSO is 4.3, for NLSTM-AM-GA is 5.2, for NLSTM-AM-CSO is 4.6, for NLSTM-AM-Fluctuate CSO is 4.2. It means that the NLSTM-AM-Fluctuate CSO has smallest RMSE. The result for Dataset 2 as follows,

Table 2. RMSE values of Dataset 2 by forecasting algorithms

Rep	LSTM	NLSTM	NLSTM-AM	NLSTM-AM-PSO	NLSTM-AM-GA	NLSTM-AM-CSO	NLSTM-AM-Fluctuate CSO
1.	13.04	9.40	9.26	4.1	5.9	4.8	4.8
2.	13.04	9.46	8.64	4.44	3.4	4.9	4.9
3.	13.04	9.36	12.65	7.67	7.5	4.8	4.9
4.	13.04	9.41	12.12	4.18	4.7	4.8	4.9
5.	13.04	9.06	9.25	3.47	5.3	4.8	4.8
6.	13.04	8.64	9.22	4.17	4.2	4.9	4.9
7.	13.04	9.43	9.01	4.29	6.7	4.9	4.6
8.	13.04	8.93	8.48	3.53	4	4.8	4.9
9.	13.03	13.03	9.01	3.54	8.7	4.9	4.7
10.	13.04	9.11	9.03	3.67	3.4	4.9	4.9
11.	13.04	8.43	9.30	3.85	4	4.9	4.5
12.	13.04	12.97	9.24	5	4.3	4.8	4.8
13.	13.04	9.29	9.49	3.82	4	5.0	4.5
14.	13.04	9.44	12.48	4.87	6	5.0	4.9
15.	13.04	9.29	9.46	3.7	4	4.9	4.9
16.	13.04	9.29	12.71	8.2	3.5	4.7	4.9
17.	13.04	9.39	8.85	5.14	3.4	4.8	5.0
18.	13.04	9.51	9.01	6.25	6.6	4.9	4.9
19.	13.04	9.42	9.42	3.49	7.1	5.0	4.8
20.	12.97	9.47	9.29	4.11	3.7	4.8	4.7
21.	13.04	12.97	8.66	3.53	3.4	4.9	4.9
22.	13.01	9.54	8.99	4.08	7.1	5.0	4.9
23.	13.04	9.12	9.48	3.46	6.3	5.0	4.9
24.	13.04	9.54	9.12	4.4	4.1	5.0	4.9
25.	13.03	8.58	9.50	4.87	4.2	4.9	5.0
26.	13.04	9.45	9.42	3.83	8	4.8	5.0
27.	13.04	13.01	9.28	3.47	4.5	4.8	4.9
28.	13.04	9.48	9.46	3.411	4.5	4.8	4.9
29.	12.97	9.55	9.43	4.3	4.3	5.0	5.0
30.	13.01	9.55	9.39	3.6	8.7	4.9	5.0
Average	**13.03**	**9.77**	**9.62**	**4.35**	**5.18**	**4.87**	**4.86**

From table 2, it is shown the RMSE of testing data average using LSTM is 13.03, for NLSTM is 9.77, for NLSTM-AM is 9.62, for NLSTM-AM-PSO is 4.35, for NLSTM-AM-GA is 5.18, for NLSTM-AM-CSO is 4.87, for NLSTM-AM-Fluctuate CSO is 4.86. It means that the NLSTM-AM-Fluctuate CSO has smallest RMSE.

6. Conclusion

This research discovers the integration of NLSTM and Attention Mechanism to make forecasting. In order to optimize the parameter those algorithms are combined with Fluctuate CSO. This algorithm is compared with several algorithm such as LSTM, NLSTM, NLSTM-AM, NLSTM-AM-PSO, NLSTM-AM-GA, and NLSTM-CSO for electricity consumption. This algorithm has lowest RMSE which are 4.2 and 4.86. Based on the result, it can be concluded that the integration of NLSTM, Attention Mechanism, and Fluctuate CSO has better performance than other algorithms.

References

Bahdanau, D., Cho, K., & Bengio, Y., *Neural Machine Translation by Jointly Learning to Align and Translate,2014*.

Bianco, V., Manca, O., Nardini, S., & Minea, A. A., Analysis and forecasting of nonresidential electricity consumption in Romania. *Applied Energy*, *87*(11), 3584–3590, 2010. https://doi.org/10.1016/j.apenergy.2010.05.018

Chapagain, K., Kittipiyakul, S., & Kulthanavit, P., Short-Term Electricity Demand Forecasting: Impact Analysis of Temperature for Thailand. *Energies*, *13*(10), 2498, 2020. https://doi.org/10.3390/en13102498

Gandomi, A. H., Yang, X.-S., Talatahari, S., & Alavi, A. H. (2013). Metaheuristic Algorithms in Modeling and Optimization. In *Metaheuristic Applications in Structures and Infrastructures* (pp. 1–24). Elsevier. https://doi.org/10.1016/B978-0-12-398364-0.00001-2

Ghosh, T. K., Das, S., Barman, S., & Goswami, R. , *A Comparison Between Genetic Algorithm and Cuckoo Search Algorithm to Minimize the Makespan for Grid Job Scheduling* (pp. 141–147), 2017. https://doi.org/10.1007/978-981-10-2525-9_14

Hu, D. , *An Introductory Survey on Attention Mechanisms in NLP Problems* (Vol. 1038, pp. 432–448),2020. https://doi.org/10.1007/978-3-030-29513-4_31

Liu, Y., Wang, Y., Yang, X., & Zhang, L. , Short-term travel time prediction by deep learning: A comparison of different LSTM-DNN models. *2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC)*, 1–8,2017. https://doi.org/10.1109/ITSC.2017.8317886

Luong, M.-T., Pham, H., & Manning, C. D. , *Effective Approaches to Attention-based Neural Machine Translation,2015*.

Moniz, J. R. A., & Krueger, D. (2018). *Nested LSTMs*.

Narayan, A., & Hipel, K. W., Long short term memory networks for short-term electric load forecasting. *2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, 2573–2578, 2017. https://doi.org/10.1109/SMC.2017.8123012

Niu, Z., Zhong, G., & Yu, H., A review on the attention mechanism of deep learning. *Neurocomputing*, *452,2021*, 48–62. https://doi.org/10.1016/j.neucom.2021.03.091

Ran, X., Shan, Z., Fang, Y., & Lin, C., An LSTM-Based Method with Attention Mechanism for Travel Time Prediction. *Sensors*, *19*(4), 861 ,2019. https://doi.org/10.3390/s19040861

Shair, E. F., Cuckoo Search Approach for Cutting Stock Problem. *International Journal of Information and Electronics Engineering*, *5*(2),2015. https://doi.org/10.7763/IJIEE.2015.V5.518

Singh, A. K., Ibraheem, Khatoon, S., Muazzam, Md., & Chaturvedi, D. K. (2012). Load forecasting techniques and methodologies: A review. *2012 2nd International Conference on Power, Control and Embedded Systems*, 1–10. https://doi.org/10.1109/ICPCES.2012.6508132

Staudemeyer, R. C., & Morris, E. R. (2019). *Understanding LSTM -- a tutorial into Long Short-Term Memory Recurrent Neural Networks*.

van Houdt, G., Mosquera, C., & Nápoles, G., A review on the long short-term memory model. *Artificial Intelligence Review*, *53*(8), 5929–5955, 2020. https://doi.org/10.1007/s10462-020-09838-1

Wang, T., Meskin, M., & Grinberg, I. (2017). Comparison between particle swarm optimization and Cuckoo Search method for optimization in unbalanced active distribution system. *2017 IEEE International Conference on Smart Energy Grid Engineering (SEGE)*, 14–19. https://doi.org/10.1109/SEGE.2017.8052769

Yan, K., Zhou, X., & Chen, J. (2022a). Collaborative deep learning framework on IoT data with bidirectional NLSTM neural networks for energy consumption forecasting. *Journal of Parallel and Distributed Computing*, *163*, 248–255. https://doi.org/10.1016/j.jpdc.2022.01.012

Yan, K., Zhou, X., & Chen, J. , Collaborative deep learning framework on IoT data with bidirectional NLSTM neural networks for energy consumption forecasting. *Journal of Parallel and Distributed Computing*, *163*, 248–255,2022. https://doi.org/10.1016/j.jpdc.2022.01.012

Yang, X.-S., & Suash Deb., Cuckoo Search via Lévy flights. *2009 World Congress on Nature & Biologically Inspired Computing (NaBIC)*, 210–214, 2009. https://doi.org/10.1109/NABIC.2009.5393690