Website Visitors Forecasting using Recurrent Neural Network Method

I Putu Bagus Arya Pradnyana*1, Wayan Firdaus Mahmudy2, Achmad Basuki3
1,2,3Faculty of Computer Science, University of Brawijaya Malang, Indonesia
1bagusarya@student.ub.ac.id, 2wayanfm@ub.ac.id, 3abazh@ub.ac.id
*Corresponding Author

Received 15 February 2021; accepted 04 July 2021

Abstract. The number of visitors and content accessed by users on a site shows the performance of the site. In this study, using two data new visitors, and first time visitors of the journal website. The testing using new visitor data and first time visitors from 2018 to 2019 with vulnerable time per month. Therefore, forecasting needs to be done to find out how many users a website will come. This study applies the Long Short Term Memory (LSTM) method which is a development of the Recurrent Neural Network (RNN) method. LSTM has the advantage that there is an architecture of remembering and forgetting the output to be processed back into the input. In addition, the ability of another LSTM is to be able to maintain errors that occur when doing Backpropagation so that it does not allow errors to increase. This study compares two methods, namely LSTM and Backpropagation. Results of the mean square error (MSE) for the LSTM method at first time visit data is 0.0184, and for new visitor data is 0.0521. The Backpropagation result for first time visitor data is 0.1542, and for new visitor data is 0.1424. The computational experiment prove that the LSTM produces better result in term of the (MSE) comparable to those achieved by Backpropagation Neural Network method.

1 Introduction

In this modern era, the growth of internet usage is very high. Internet users are now mainly in the search for information very much. Every time an internet user visits a website will surely leave a mark on the website. These tracks can be collected, and used for various purposes such as tracking user behavior, recommending products to internet users when visiting the next website, and optimizing the usefulness of the website [4].

So far, forecasting is an approach commonly used to help humans in making decisions. The purpose of this research is to use forecasting to determine user behavior when visiting the website. Also, one of the factors that influence the research objectives is that information on the number of visitors to a website is needed [8].

The journal website is a useful tool for the requirements for advancement of each level of position for lecturers, researchers, teachers, lecturers, engineers, and other functional functions. There are several problems that journal managers have because they have not implemented online scientific journal management [11]. One of the components of journal accreditation is the number of visitors to the journal website. Therefore, forecasting is necessary to see how many users measure whether a promotion is necessary to increase the journal website visitors. The number of visitors to the journal website is one of the providers for journal accreditation[11].

One of the websites that is frequently visited is the official site of the Universitas Negeri Malang, which is one of the media providers of information for the academic community [15]. Forecasting is an attempt to predict the situation in the future by testing the situation in the past. Forecasting problems usually use time-series data [10].
In this study, it focuses on forecasting journal website visitors. Using two visitor data. In this study, it was conducted by comparing two methods, namely LSTM and Backpropagation. The contribution of this study is to prove the best forecasting method seen from the MSE results.

2 Related Works

In forecasting research that used Neural Networks, there are many methods used in the following examples. In the research of Alfiyatin [2] and Oktanisa [12] discusses inflation forecasting in Indonesia. For E research using the method (ELM optimization with the PSO method), and for the I method (SVR optimization with the GA method) The results in research E forecasting with ELM and ELM optimized with PSO, the results are almost the same with the difference of error 0.0000019. The results in the research I forecast with SVR and SVR optimized with GA results better SVR optimized GA but requires a longer process than SVR. In the study, Meilia [9] discusses forecasting electricity consumption in Indonesia. The method used is the method (ELM with GA method optimization) The results of the EW study show that the ELM method can be GA-optimized to optimize the weighting of the ELM.

In a study, Sari's [14] research shows that the Backpropagation method compared to the Sugeno FIS method for forecasting, in Indonesia shows the forecasting results with Backpropagation better than Sugeno FIS with an RMSE of 0.204. In a study conducted by Haviluddin [6], that the accuracy of the network traffic activity prediction model can be optimized using GA on a multi-layer perceptron. Convergence and early permutation problems can be optimized using backpropagation on the GA operator. The results showed that the model prediction performance was superior to the traditional Multi-layer perceptron. Network traffic forecasting when using a combination of Backpropagation and GA that must be considered in the determination of the optimal Neural Network. In research by Havil and Dengen [5], it can be predicted that daily network traffic can be predicted. SARIMA, NARX, and BPNN models are used to test the performance of the prediction time series. The results showed that nonlinear time series modeling and complex predictive tasks can be used in all models. Backpropagation results have better predictive accuracy than the SARIMA and NARX models measured using MSE. With the results of SARIMA error of 0.064190, NARX error of 0.006717, and BPNN error of 0.009424 [5].

From some previous studies, the use of the Multi-layer perceptron method is widely used for forecasting. One Neural Network algorithm that can also be used in forecasting is the Recurrent Neural Network. RNN is one of the neural network methods used for forecasting. The RNN method is very good for research because it can train using time series data [13]. This method is good for predicting data with time series data types. According to Berradi's research [3], the number of features is reduced using the PCA method, and then the RNN method is used to predict the Total Maroc share price from the Casablanca exchange. MSE obtained by RNN with PCA is smaller than MSE obtained by RNN without PCA with results with PCA with MSE of 0.00596 and without PCA with MSE as of 0.011835 [3].

Based on previous research, the RNN method obtained quite good results in overcoming the problem of forecasting, so in this study, the RNN method was used to forecast website visitors. RNN can study dependencies between sequential data input or time series. The ability of sequential dependency learning makes the RNN method very popular and widely used. In this study using the Recurrent Neural Network method for forecasting website visitors. It is expected to get good results in the field of forecasting.

3 Backpropagation

In this study, the Backpropagation method is used as a comparison method. Because the Backpropagation method is often used in various fields. The Backpropagation learning algorithm is to minimize the error rate by means of adjusting the output difference and does it weigh based on the desired target. Backpropagation including a multilayer network which is a single layer of network development [14]. Backpropagation architecture can be seen in Fig. 1.
Fig 1. Backpropagation Architecture

In general the process stages in Backpropagation is divided into 4 stages[14]:
1. The initial weights Initialization
2. Perform calculations feedforward
3. Do the calculation of backpropagation
4. Calculate the weights and new bias

1. The initial weights Initialization

Random initial weights from 0 to 1. With the number of inputs you want.

2. Perform calculations feedforward

First calculation where variable $z_{inj}$ is signal entered hidden layer. Do calculation with variable $v_{0j}$ is bias in hidden layers plus sigma n to i as 1 and multiply variable $x_i$ is input consisting of neurons and variable $v_{ij}$ is weight on hidden layers. Each unit is hidden ($z_j, j = 1,2,3,..., p$) the weight of the input signal is added by Equation 1.

$$z_{inj} = v_{0j} + \sum_{i=1}^{n} x_i v_{ij}$$

(1)

Second calculation where variable; $z_j$ is the result of the hidden layer activation function. Variable $z_{inj}$ is signal entered hidden layer. Use the activated sigmoid binary function to calculate the output signal from a hidden unit with Equation 2.

$$z_j = \frac{1}{1 + e^{-z_{inj}}}$$

(2)

Third calculation where variable $y_{ink}$ is input output signal, Variable $w_{0k}$ is weight bias to the output layer, The $z_i$ and $w_{ij}$ variables are the result of the hidden layer activation function and the hidden layer weight, respectively. Where ($Y_k, k = 1, ..., m$) will add the weighted input signals including the bias, with Equation 3.

$$y_{ink} = w_{0k} + \sum_{i=1}^{n} z_j w_{ij}$$

(3)
Fourth calculation where Variable $y_k$ is output layer activation function. Variable $y_{in}$ is input output signal. Equation 4 is used to calculate the output signal from the output unit using the binary sigmoid activation function.

$$y_k = \frac{1}{1 + e^{-y_{in}}}$$  

(4)

2. Do the calculation of backpropagation

First calculation where variable $\delta_k$ is output layer correction factor. Variable $t_k$ is target data. Variable $y_k$ is training output. Based on the error in each unit of output calculate the factor $\delta$ unit of output ($y_k, k = 1, 2, ..., n$) with Equation 5.

$$\delta_k = (t_k - y_k)y_k(1 - y_k)$$  

(5)

Second calculation where variable $\Delta W_{jk}$ is factor changes in the output layer weight. The variables $\alpha$ and $\delta_k$ are the learning speed and output layer correction factor, respectively. Variable $z_j$ is the result of the hidden layer activation function. Calculating the weight change factor $W_{jk}$ will change the weight of $W_{jk}$ with Equation 6.

$$\Delta W_{jk} = \alpha \delta_k z_j$$  

(6)

Third calculation where variable $\Delta W_{0k}$ is change factor of output layer bias. The variables $\alpha$ and $\delta_k$ are the learning speed and output layer correction factor, respectively. Calculates the change factor of the $W_{0k}$ bias which will change the $W_{0k}$ bias with Equation 7.

$$\Delta W_{0k} = \alpha \delta_k$$  

(7)

Fourth calculation where variable $\delta_{inj}$ is hidden weight delta. Variable $\delta_k$ is hidden layer correction factor. Variable $w_{ij}$ is hidden layer weight to the output layer. Calculating delta weight of hidden units with Equation 8.

$$\delta_{inj} = \sum_{k=1}^{m} \delta_k w_{ij}$$  

(8)

Fifth calculation where variable $\delta_j$ is hidden unit error correction factor. Variable $\delta_{inj}$ is hidden weight delta. Variable $z_j$ is hidden layer activation factor. Calculate hidden unit error correction factors with Equation 9.

$$\delta_j = \delta_{inj} z_j(1 - z_j)$$  

(9)

Sixth calculation where variable $\Delta v_{jk}$ is correction of hidden layer weights. Variable $\alpha$ is learning rate. Variable $\delta_j$ is hidden unit error correction factor. Variable $x_j$ is input value. Calculate the correction of hidden layer weights with Equation 10.

$$\Delta v_{jk} = \alpha \delta_j x_j$$  

(10)

Seventh calculation where variable $\Delta v_{0j}$ is correction of hidden layer bias. Variable $\alpha$ is learning rate. Variable $\delta_j$ is hidden unit error correction factor. Equation 11 is used to calculate the hidden layer bias correction.

$$\Delta v_{0j} = \alpha \delta_j$$  

(11)

3. Calculate the weights and new bias

First calculation where the variables $v_{ij}$ (new) and $v_{ij}$ (old) are the new weight input layer for the hidden layer and the weight from the input layer to the hidden layer, respectively. Variable $\Delta v_{ij}$ is correction of hidden layer weights. Equation 12 is used to calculate the new weight from the input layer to the hidden layer.

$$v_{ij}(new) = v_{ij}(old) + \Delta v_{ij}$$  

(12)

Second calculation variables $v_{0j}$ (new) and $v_{0j}$ (old) respectively are only able to insert layer into hidden layer and bias from input layer to hidden layer. Variable $\Delta v_{0j}$ is correction of hidden layer bias. Equation 13 is used to calculate the bias from the new input layer to the hidden layer.

$$v_{0j}(new) = v_{0j}(old) + \Delta v_{0j}$$  

(13)
Third calculation where variable $w_{jk}(\text{new})$ and variable $w_{jk}(\text{old})$ are the new weight from the hidden layer to the output layer and the old weight from the hidden layer to the output layer, respectively. The variable $\Delta w_{jk}$ is the weight correction for the output layer. Equation 14 is used to calculate the new weight from the hidden layer to the output layer.

$$w_{jk}(\text{new}) = w_{jk}(\text{old}) + \Delta w_{jk}$$

(14)

Fourth calculation where variable $w_{ok}(\text{new})$ and variable $w_{ok}(\text{old})$ are are biased to the output layer and the long hidden layer to the output layer, respectively. The variable $\Delta w_{ok}$ is correction of the output layer bias. Equation 15 is used to calculate the bias of the new hidden layer to the output layer.

$$w_{0k}(\text{new}) = w_{0k}(\text{old}) + \Delta w_{0k}$$

(15)

4 Reccurent Neural Network (RNN)

Reccurent Neural Network is a good Neural Network used for forecasting with time series data. Recurrent Neural Network neuron values in the previously hidden layer will be reused as input data. at its core (called a cell) a loop occurs. This means that the output of this cell will be the input again [1]. Recurrent Neural Network architecture can be seen in Fig. 2.

![Recurrent Neural Network Architecture](image)

Figure 2 shows the input, recurrent hidden, and output layers that RNN has. N input units are the vector sequence through time t that is $x_t = (x_1, x_2, ..., x_N)$. Meanwhile, the recurrent hidden layer is directly connected to the input layer. Where $M$ the hidden layer units are $h_t = (h_1, h_2, ..., h_M)$. In the RNN method, the output process will refer to the previous computation for each element sequentially. RNN has a memory that contains previously recorded information generated [13].

The RNN training process is very similar to training on ordinary neural networks. Using the Backpropagation Algorithm, with a slight twist. Because the parameters are shared equally (evenly) at each time step on the network, the gradient for each output depends not only on the calculation of the current time step but also on the previous time step [13]. The training process for the Recurrent Network is divided into three namely:

1. Forward propagation which involves the hidden state calculation process and activation functions,
2. Backward propagation to find the gradient value based on the loss function value of the forward propagation process,
3. Weight Update is changing the weight value of Wxh, Whh, and Who using the learning rate and the gradient value of the results of backward propagation.

5 Long Short Term Memory (LSTM)

In this research, the method used was LSTM as the proposed method. Because the LSTM method is one of the upgraded RNN methods. The RNN method used is LSTM where the development method of the RNN follows its explanation. Long Short Term Memory networks (LSTM) is an evolution of the RNN architecture, which was first introduced by Hochreiter & Schmidhuber (1997). Until this research was conducted many researchers continued to develop LSTM architecture in various fields such as speech recognition and forecasting [1].

LSTM uses memory cells and gate units to manage memory at each input. As explained in the previous section, to manage memory in each neuron, LSTM has a memory cell and a gate unit [8].

![LSTM Architecture](image)

In Fig 3 shows the workflow of memory cells in each LSTM neuron working. There are four processes of activation functions at each input to neurons, hereinafter referred to as gates units. Gates units are forgotten gates, input gates, cell gates, and output gates [7].

LSTM Training

Forget gates information on each input data will be processed and selected which data will be stored or discarded in memory cells. The output using the simoid activation function is a value of 1 then the data is stored and the value is 0 then the data is discarded. [8]. With the Equation 16:

\[
f_t = \sigma(W_f[h_{t-1}, x_t] + b_f)
\]

Where:
- \( f_t \) = Forget gate
- \( \sigma \) = Learning rate
- \( W_f \) = Weight matrix of forget gate
- \( h_{t-1} \) = Output values at time points ti 1 and ti
- \( x_t \) = Input value
- \( b_f \) = Bias of forget gate

Input gates, consists of the sigmoid activation function and the tanh activation function, each of which is useful for updating values and creating new value vectors that will be stored in memory cells. With the Equation 17 and Equation 18:

\[
i_t = \sigma(W_i[h_{t-1}, x_t] + b_i)
\]

\[
c_t = \tanh(W_c[h_{t-1}, x_t] + b_i)
\]

Where:
\[ i_t = \text{Input gate} \]
\[ C_t = \text{Memory cell} \]
\[ \sigma = \text{Learning rate} \]
\[ W_f = \text{Weight matrix of forget gate} \]
\[ h_{t-1} = \text{Output values at time points t\_1 and t\_1} \]
\[ x_t = \text{Input value} \]
\[ b_i = \text{Bias of input gate} \]

Cell gate will replace the value in the previous memory cell with a new memory cell value. Where this value is obtained from combining the values contained in the forget gate and input gate. With the Equation 19:
\[ C_t = f_t \cdot c_{t-1} + i_t \cdot \tilde{C}_t \]  
(19)

Where:
\[ C_t = \text{Cell Gate} \]
\[ f_t = \text{Forget gate} \]
\[ \tilde{C}_t = \text{Memory cell} \]
\[ i_t = \text{Input gate} \]
\[ W_f = \text{Weight matrix of forget gate} \]
\[ c_{t-1} = \text{Cell status at time points t\_1 and t\_1} \]

The output gate is a sigmoid activation function and a tanh activation function, each of which is useful for determining the value of which part of the memory cell will be issued and placing the value in the memory cell. Lastly the value is issued by multiplying the two values. With the Equation 20 and 21:
\[ o_t = \sigma (W_o [h_{t-1}, x_t] + b_o) \]  
(20)

\[ h_t = o_t \cdot \text{tanh}(C_t) \]  
(21)

Where:
\[ o_t = \text{Output gate} \]
\[ \sigma = \text{Learning rate} \]
\[ W_o = \text{Weight matrix of output gate} \]
\[ h_{t-1} = \text{Output values at time points t\_1 and t\_1} \]
\[ x_t = \text{Input value} \]
\[ b_o = \text{Bias of output gate} \]

6 Experiments and Results

Using UM journal website visitor data. There are 2 journals used in this study. First is the journal with the journal2.um.ac.id/index.php/keds link and the second is the journal link journal.um.ac.id/index.php/jptpp. With 4 input variables from the first journal, namely, pageview, session, visitor, and new visitor. With 4 input variables from the second journal namely page loads, unique visits, first time visits, and return visits. With vulnerable times per month.

This study will use the variable first time visits. The data used for the train is 80% and testing is 20% of the data used.

Data collection in this study was taken from the journal website of Malang State University. The research variable in this thesis is the website of the Malang State University journal. The results of this experiment are estimates of website visitors. Website visitor data shows how big the website visitors are from 2018 to 2019. Based on the data that will be used in this study will test the training of the two methods with the following parameters:

For LSTM parameter:
- Input: 15
- Output : 1
- Batch: 1
- Epoch: 100
- Learning rate: 0.001
For Backpropagation parameter:
- Input: 15
- Output: 1
- Epoch: 100
- Hidden layer: 1
- Learning rate: 0.001

In this study, the performance of the LSTM and Backpropagation methods for epoch 100 will be tested. Training will be tested 10 times because of the random value at the initialization of the weight value of the two methods. With Epoch 100 the MSE results from the LSTM and BP method can be seen in Table 1.

| Training | First time visits LSTM | New visitor | First time visits BP | New visitor |
|----------|------------------------|-------------|----------------------|------------|
| 1        | 0.0131                 | 0.0522      | 0.0916               | 0.0680     |
| 2        | 0.0173                 | 0.0552      | 0.1494               | 0.2768     |
| 3        | 0.0155                 | 0.0519      | 0.0700               | 0.0850     |
| 4        | 0.0270                 | 0.0523      | 0.0451               | 0.0904     |
| 5        | 0.0134                 | **0.0501**  | 0.1058               | 0.2354     |
| 6        | 0.0278                 | 0.0515      | **0.0129**           | 0.1171     |
| 7        | 0.0226                 | 0.0525      | 0.0463               | **0.0828** |
| 8        | 0.0139                 | 0.0510      | 0.3896               | 0.2738     |
| 9        | 0.0152                 | 0.0523      | 0.4286               | 0.0555     |
| 10       | 0.0184                 | 0.0525      | 0.2030               | 0.1398     |
| **Average** | **0.0184** | **0.0521** | **0.1542**           | **0.1424** |

Computation Time | 139 ms | 150 ms | 158 ms | 189 ms

In table 1, you can see the MSE results from the LSTM and Backpropagation methods with 10 times of training. It can be seen that the LSTM method on the first time visit data from the 1st training got 0.0131 results, while the Backpropagation on the first time visit data from the 6th training got 0.0129 results showing Backpropagation is better than LSTM, but the LSTM method is better than Backpropagation in terms of an average of 10 training times.

It can be seen that the LSTM method on new visitor data from the 5th training got 0.0501 results, while Backpropagation on the first time visit data from the 7th training got 0.0828 results showing LSTM is better than Backpropagation. The results above show that MSE from LSTM is better than Backpropagation.

The LSTM result for first time visits data is 0.0184, and for new visitor data is 0.0521 and Backpropagation for time visitor data is 0.1542, and for new visitor data is 0.1424. It can be seen the results of the LSTM training method more accurate than Backpropagation.

With computation time 139ms for LSTM, and 150ms for Backpropagation at first time visits data. Computation time 158ms for LSTM and 189ms for Backpropagation at new visitor data. For comparison of the computation the LSTM method performs computation more efficiently than Backpropagation.

From the comparison of the LSTM and Backpropagation methods on the data of the first time and new visitors, it can be seen from the LSTM method that the average MSE results from the second data are better than the Backpropagation method. In terms of computation time, the LSTM method is faster than the Backpropagation method. Therefore, the LSTM method is better than the Backpropagation method, because the LSTM training process uses two activation functions while Backpropagation is only one activation function.

7 Conclusions

This paper proposed the forecasting of journal website visitors using the LSTM and Backpropagation methods. By testing training 10 times using Epoch 100.
From the test results using the data of new visitors and first time visitors from 2018 to 2019 with vulnerable times per month. It can be seen the value of MSE from new visitors is better than first time visitors with MSE of 0.0184 for first time visitors, and for New Visitor data is 0.0521. And for the Backpropagation method the MSE value is First time visit 0.1542, and for New Visitor data is 0.1424. It can be concluded that the LSTM method is better than the Backpropagation method.

From these results, it can be seen that the data used are still lacking results so it causes less than optimal results in the field of prediction. It is hoped that in the future it can use more data.

References
1. H. Abbasimehr, M. Shabani, and M. Yousefi, "An optimized model using LSTM network for demand forecasting," Computers & Industrial Engineering, vol. 143, p. 106435, 2020/05/01/ 2020.
2. A. Alfiyatin, A. M. Rizki, W. Mahmudy, and C. Fajri, "Extreme Learning Machine and Particle Swarm Optimization for Inflation Forecasting," International Journal of Advanced Computer Science and Applications, vol. 10, 2019.
3. Z. Berradi and M. Lazaar, "Integration of Principal Component Analysis and Recurrent Neural Network to Forecast the Stock Price of Casablanca Stock Exchange," Procedia Computer Science, vol. 148, pp. 55-61, 2019/01/01/ 2019.
4. U. Gunter and I. Onder, "Forecasting city arrivals with Google Analytics," Annals of Tourism Research, vol. 61, pp. 199-212, 2016.
5. Haviluddin and N. Degen, "Comparison of SARIMA, NARX and BPNN models in forecasting time series data of network traffic," in 2016 2nd International Conference on Science in Information Technology (ICSITech), 2016, pp. 264-269.
6. H. Haviluddin and R. Alfred, A genetic-based backpropagation neural network for forecasting in time-series data, 2015.
7. Y. Huang, S. Liu, and L. Yang, "Wind speed forecasting method using EEMD and the combination forecasting method based on GPR and LSTM," Sustainability, vol. 10, p. 3693, 2018.
8. X. Li, R. Law, G. Xie, and S. Wang, "Review of tourism forecasting research with internet data," Tourism Management, vol. 83, p. 104245, 2021/04/01/ 2021.
9. V. Meilia, B. D. Setiawan, and N. Santoso, "Extreme Learning Machine Weights Optimization Using Genetic Algorithm In Electrical Load Forecasting," Journal of Information Technology and Computer Science, vol. 3, p. 77, 2018.
10. D. C. Montgomery, C. L. Jennings, and M. Kulahci, Introduction to Time Series Analysis and Forecasting: Wiley, 2011.
11. A. Nurdiani, Pedoman Akreditasi Jurnal Ilmiah: Direktorat Jenderal Penguatan Riset dan Pengembangan Kementerian Riset, Teknologi, dan Pendidikan Tinggi, 2018.
12. I. Oktanisa, W. Mahmudy, and G. Maski, "Inflation Rate Prediction in Indonesia using Optimized Support Vector Regression Model," Journal of Information Technology and Computer Science, vol. 5, p. 104, 2020.
13. H. Salehinejad, S. Sankar, J. Barfett, E. Colak, and S. Valae, "Recent Advances in Recurrent Neural Networks," 2017.
14. N. Sari, W. Mahmudy, and A. Wibawa, "Backpropagation on neural network method for inflation rate forecasting in Indonesia," vol. 8, pp. 69-87, 2016.
15. S. Syahrial, M. Kharul, and M. Nunung, "Analisa Statistik Pengunjung Situs Resmi Universitas Syiah Kuala (Www.unsyiah.ac.id)," Jurnal Rekayasa Elektrika, vol. 9, pp. 49-54, 2010 2010.