Ticket Prediction using LSTM on a GLPI System

Achmad Nurfaizi, Muhaimin Hasanudin

***Abstract*—GLPI (Gestionnaire Libre de Parc Informatique) is software for asset management, with an additional interface for submitting requests and reporting incidents to computer technicians in the form of tickets. A ticket is a disruption ticket, also called a problem report, that is used in an organization to track the detection, reporting, and resolution of several problems, The number of incoming and unresolved tickets has an impact on intense uncertainty and instability for customers. The data used is primary data, namely ticket data obtained from the GLPI system in the form of raw data, data is processed and recapitulated based on daily customer request tickets. In this study, using the LSTM (long short-term memory) model with the dataset being eight attributes with a total of 2035 records, for optimal data performance, we use min-max scaling sci-kit learn to transform data, extract features, and create models to predict tickets. The parameters used include batch_size = 32, epoch = 100, and learning_rate = 0.001, with the optimizer being Adam. The best validation accuracy value (val_acc) was obtained at the 82nd epoch with a value of 9.695, and the best validation loss value (val_loss) was 0.0044. The results showed that ticket demand increased while the results of the ticket demand model prediction in the following year decreased because it was predicted that many ticket requests from customers had been completed by the team. This clearly shows how good the LSTM method is for the analysis of time series and sequential data.**

***Keywords*— Tickets, disruption tickets, customers, LSTM**

I. Introduction

Currently, information technology plays an important role in the performance and growth of the company. BICOM is an IT solutions company founded in early 2008 that focuses on providing valuable and reliable IT services to customers with the concept of "One Stop IT Solutions". Bicom's services and solutions include managed services, server and network installation, system implementation and integration, web development, and hardware and software supply.PT. Bicom Mitra Solusindo has a project to install IT asset and management tools using open source software. The software used is GLPI (Gestionnaire Libre de Parc Informatique), an IT asset and management system package that provides service desk features, license tracking, and software auditing [1].A ticketing system is a customer service tool that helps companies manage customer cases by documenting customer inquiries and interactions over time, making it easier for customer service staff to resolve issues. Bicom has customers spread across Indonesia, and there are also many ticket requests from customers that need to be resolved immediately.In an IT environment, there are many trouble tickets that occur, such as network problems, hardware failures, and software problems. These trouble tickets need to be dealt with quickly and efficiently to minimize downtime on the system and provide good service to users. Therefore, the use of a trouble ticket prediction model can help identify and predict problems that may occur in a system so that preventive measures can be taken before they occur. In this case, LSTM was chosen as a model to predict trouble tickets because of its ability to process continuous data and predict patterns in time-series data. Thus, this research is expected to contribute to the development of a more efficient and high-quality IT service management system.In this study, the method used is the development of a trouble ticket prediction model using long short-term memory (LSTM), which is a type of neural network model that can process continuous data with long time sequences [2, 3, 4]. In developing this prediction model, the data used is historical data on trouble tickets on the system to be predicted. It is hoped that this research can contribute to the development of a more efficient and high-quality IT service management system so as to increase user satisfaction, reduce system downtime, and cause both financial and non-financial losses. Therefore, a method is needed that can help predict problems before they occur so that preventive actions can be taken in a timely manner and can be used as a reference for further research in the development of trouble ticket prediction models in IT systems using the LSTM method [5].

II. THEORY THING AND PREVIOUS RESEARCH

A. Dataset

The data used in this study are primary data obtained from the GLPI system at PT. Bicom Mitra Solusindo. The data collected is in the form of time series data with a time span of 2018–2022. The data obtained from the GLPI system is the number of tickets issued each day, totaling 2767 records containing 14 variables. The data used as research material is a request ticket with a completion time of one day. The data taken is from 2018 to 2022.

B. Methodology

The following is the methodology for making a ticket prediction model using LSTM:

a) Data collection: Historical trouble ticket data on the IT system to be predicted is collected and stored in a database.

b) Data preprocessing: Data that has been collected will be processed first by preprocessing, namely deleting irrelevant or incomplete data, filling in missing data, and correcting data formats.

c) Data division: The processed data is then divided into training data and test data.

d) Model creation: The trouble ticket prediction model using LSTM was developed by utilizing pre-shared training data. The model is then trained using the appropriate algorithm to improve prediction accuracy.

e) Model evaluation: After the model has been successfully trained, it will be tested using test data to evaluate the accuracy of its predictions. If the prediction accuracy is in accordance with the expected standards, then the model can be used to predict trouble tickets in IT systems.

f) Model optimization: If the prediction accuracy does not meet the expected standards, then the model needs to be optimized using appropriate techniques, such as improving the model structure, adding new features, or changing the algorithm used.

g) Model implementation: After the model has been successfully optimized and meets the expected standards, it can be implemented in IT systems to predict future trouble tickets.

h) Model maintenance: After the model is implemented, it is necessary to carry out regular model maintenance to ensure that the prediction accuracy remains consistent and continues to be improved. This maintenance includes collecting new data, re-evaluating the model, and optimizing the model if needed.

C. GLPI

GLPI stands for Gestionnaire Libre de Parc Informatiquea, which in English means free asset and IT management software. It is software for asset management with an additional interface for submitting requests and also for reporting incidents to IT technicians. It is open source software that is written in PHP and distributed under the GPL (General Public License). Where everyone can freely run, modify, and develop according to needs, which makes the function of an administrator easier. GLPI stands for Gestionnaire Libre de Parc Informatiquea,a which in English means free asset and IT management software. Ite isa software for asset management with an additional interface for submitting requests and also for reporting incidents to IT technicians.s Ith isn open source software that ist written in PHP and distributed under the GPL (General Public License). Where everyone can freely run, modify, and develop according to needs, which makes the function of an administrator easier [1].

D. Ticketing

Ticketing is a trouble ticket, also called a problem report, that is used in an organization to track the detection, reporting, and resolution of several problems [10]. The helpdesk ticketing system is an organized system of trouble tickets that are reported by the user to the support team to solve problems that exist for the user and as a complement to a service function.

E. Long Short-Term Memory (LSTM)

Long Short-Term Memory (LSTM) is a type of recursive neural network architecture developed to overcome the vanishing gradient problem in traditional neural networks [2, 3]. LSTM can maintain the information generated at each layer by adjusting the flow of information at different times, so that information can be accumulated and processed better.LSTM consists of several unit cells, where each unit cell has three "gates", namely the input gate, the forget gate, and the output gate. The input gate is used to determine how much new information will be entered into the cell, the forgotten gate is used to determine how much old information will be retained, and the output gate is used to determine how much information will be generated at the next layer [11, 12].

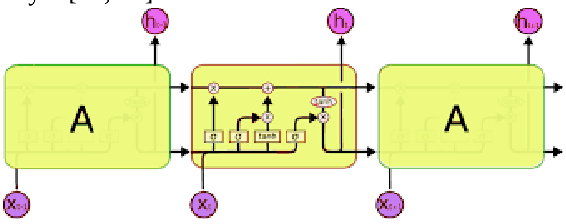


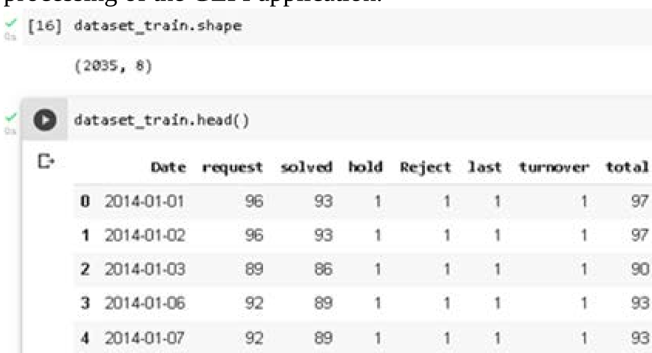
Fig.1. LSTM Architecture [13]

With a gateway, LSTM can select important information and ignore irrelevant information so that it can make more accurate predictions on complex data that has patterns that are hard to find. LSTMs are widely used in applications such as natural language processing, handwriting recognition, and predicting stock price movements.

III. PROPOSED IDEA

3.1 Dataset

The dataset used is primary data at PT Bicom Mitra Solusindo, namely ticketing support data obtained from the GLPI application. The data source from the GLPI application is in the form of raw and detailed data; therefore, the data is processed and the data is recapitulated based on daily data from ticketing or cases from service requests that are at partner work sites (onsite) for computer repairs such as computer networks, hardware, software, IT support, etc., to create GLPI datasets. The GLPI dataset consists of eight (8) attributes, namely date, request, solved, hold, request postponed, reject, last request, turnover (transferred), and total (total requests), with a total of 2035 records. The data was taken from 2014 to 2022 and used as a tracing dataset. Figure 2 is the GLPI training dataset obtained from the processing of the GLPI application.



```
[16] dataset_train.shape

(2035, 8)

dataset_train.head()
```

	Date	request	solved	hold	Reject	last	turnover	total
0	2014-01-01	96	93	1	1	1	1	97
1	2014-01-02	96	93	1	1	1	1	97
2	2014-01-03	89	86	1	1	1	1	90
3	2014-01-06	92	89	1	1	1	1	93
4	2014-01-07	92	89	1	1	1	1	93

Fig. 2. GLPI Training Dataset

whereas the testing dataset consists of eight attributes and sixteen (16) records made for the January 2023 data simulation and tracking. Where the training and testing datasets have the same attributes, only the amount of data differs.

3.2 Data Preprocessing

In this study, the dataset was divided into about 90% training data and about 10% validation and testing data. So that the number of training records is 2035 and the number of testing records is 16 records. The request column is the initial case, while the solved column is the final case of a problem complaint on a certain day. The pending and rejected columns represent pending and rejected cases for a given day. Then we scale the data features so that performance is optimal. In this case, we use MinMaxScaler and scale the dataset to zero and one in the training dataset with the following script

```
from sklearn.preprocessing import MinMaxScale
r
sc = MinMaxScaler(feature_range = (0, 1))
training_set_scaled = sc.fit_transform(traini
ng_set)
```

In making data with Timesteps with LSTM, that is data in a certain format, usually 3D arrays. Start by creating the data in 60 time steps and turning it into an array using NumPy. Next, convert the data into a 3D dimensional array with X_train samples, 60 timestamps, and one feature at each step, like in the script below.

```
X_train = []
y_train = []
for i in range(60, 2035):
    X_train.append(training_set_scaled[i-
60:i, 0])
    y_train.append(training_set_scaled[i, 0])
X_train, y_train = np.array(X_train), np.arra
y(y_train)

X_train = np.reshape(X_train, (X_train.shape[
0], X_train.shape[1], 1))
```

Then, I added an LSTM layer and then some dropout layers to prevent overfitting. Add an LSTM layer with the following arguments:

a) Fifty (50) units, which is the dimension of the output room
b) Return_sequences is true to determine whether to return the last output in the output sequence or the full sequence.
c) Input_shape as the shape of the training dataset

```
regressor = Sequential()
regressor.add(LSTM(units = 50, return_sequenc
es = True, input_shape = (X_train.shape[1], 1
)))
regressor.add(Dropout(0.2))
regressor.add(LSTM(units = 50, return_sequenc
es = True))
regressor.add(Dropout(0.2))
regressor.add(LSTM(units = 50, return_sequenc
es = True))
regressor.add(Dropout(0.2))
regressor.add(LSTM(units = 50))
regressor.add(Dropout(0.2))
regressor.add(Dense(units = 1))
regressor.compile(optimizer = 'adam', loss =
'mean_squared_error')
regressor.fit(X_train, y_train, epochs = 100,
 batch_size = 32)
```

When specifying the dropout layer, I set it to 0.2, meaning that 20% of the layer will be deleted. After that, add a dense layer, which determines the output of 1 unit. Then, we compile the model using the popular Adam optimizer and define the loss as mean squared error. This will calculate the average squared error. Next, we adjusted the model to run at 100 epochs with a batch size of 32. This process could take a few minutes to complete.

3.3 Model Training and Testing

To predict the GLPI problem, several things are needed after loading in the test set, namely combining the training set (the train dataset) and the test set (the test dataset) on axis 0 and giving a time of 60 using the MinMaxScaler to change the new dataset to reshape the dataset as was done before.

```
dataset_total = pd.concat((dataset_train['req
uest'], dataset_test['request']), axis = 0)
inputs = dataset_total[len(dataset_total) -
 len(dataset_test) - 60:].values
inputs = inputs.reshape(-1,1)
inputs = sc.transform(inputs)
X_test = []
for i in range(60, 76):
    X_test.append(inputs[i-60:i, 0])
X_test = np.array(X_test)
X_test = np.reshape(X_test, (X_test.shape[0],
 X_test.shape[1], 1))
predicted_case = regressor.predict(X_test)
predicted_case = sc.inverse_transform(predict
ed_case)
```

The scaled data is then fed to a neural network consisting of a sequential arrangement containing two LSTM layers of 50 units each and a final dense layer consisting of one unit. The results are then used to calculate the metrics in the previous section as well as for the prediction graph shown below in figure 3. After creating the prediction, the author uses inverse_transform to retrieve the ticketing request in a normally readable format. In the last step, the author uses Matplotlib to visualize the predicted results of ticketing requests and actual ticketing requests.

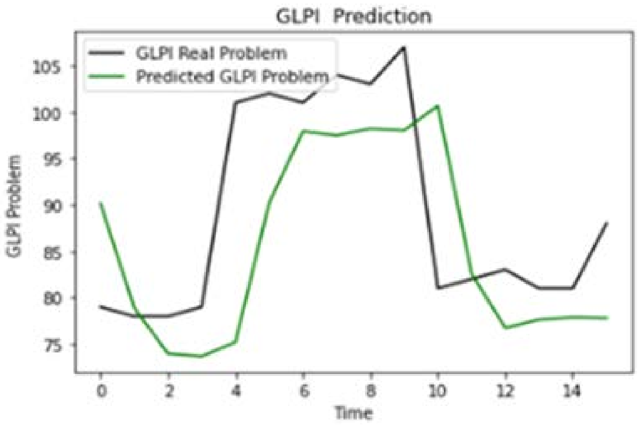


Fig. 3. GLPI Prediction Results

IV. CONCLUSION

The use of LSTM in ticket prediction systems can improve prediction accuracy and assist in decision-making. In implementation, adequate training data and parameters are needed that are adjusted to the case at hand. Based on the results of the GLPI prediction, the demand for real tickets (dataset training) has increased, while the prediction model for ticket demand in 2023 will decrease because it is predicted that many ticket requests from customers have been completed by the Bicom IT team. This clearly shows how good the LSTM method is for analyzing time series and sequential data.

V. ACKNOWLEDGMENTS

This paper is one of the requirements for passing the final assignment at Mercubuana University, Jakarta. Thanks to Mr. Muhaimin Hasanudin, the supervisor who has helped complete this thesis.

REFERENCES

[1] Robby, D. N. R. (2020). IMPLEMENTASI GLPI (GESTIONNAIRE LIBRE DE PARC INFORMATIQUE) UNTUK LAYANAN IT, MANAJEMEN ASET DAN RESERVASI ASET.

[2] Wiranda, L., & Sadikin, M. (2019). Penerapan Long Short Term Memory Pada Data Time Series Untuk Memprediksi Penjualan Produk Pt. Metiska Farma. Jurnal Nasional Pendidikan Teknik Informatika: JANAPATI, 8(3), 184-196.

[3] yuli Astari, Y., Afiyati, A., & Rozaqi, S. W. (2021). Analisis Sentimen Multi-Class pada Sosial Media menggunakan metode Long Short-Term Memory (LSTM). Jurnal Linguistik Komputasional, 4(1), 8-12.

[4] Fitrianah, D., & Jauhari, R. N. (2022). Extractive text summarization for scientific journal articles using long short-term memory and gated recurrent units. Bulletin of Electrical Engineering and Informatics, 11(1), 150-157.

[5] Asih, N. K. (2021). Algoritma J48 untuk Pemodelan Sistem Prediksi Tingkat Kerawanan Banjir dengan Visualisasi Web GIS (Doctoral dissertation, Universitas Mercu Buana Jakarta).

[6] B. Wooten, Building & managing a world class IT help desk. Berkeley, California: McGraw-Hill Osborne Media, 2001.

[7] R. Rico, "Analisis Dan Perancangan Sistem Informasi It- Helpdesk (Studi Kasus : Pt . Lontar Papyrus Pulp & Paper Industry)," J. Ilm. MEDIA SISFO, vol. 10, no. 2, pp. 296–305, 2016.

[8] L. D. Fitrani and R. V. H. Ginardi, "Analysis Improvement of Helpdesk System Services Based on Framework COBIT 5 and ITIL 3rd Version (Case Study: DSIK Airlangga University)," in The 4th International Seminal on Science and Technology, 2018, vol. 0, no. 1, p. 28.

[9] R. M. Bahrudin, M. Ridwan, and H. S. Darmojo, "Penerapan Helpdesk Ticketing System Dalam Penanganan Keluhan Penggunaan Sistem Informasi Berbasis Web," Jutis, vol. 7, no. 1, pp. 71–82, 2019

[10] W. D. Suryono, Saptono, Ristu, and W. Wiranto, "Implementasi Pengembangan Smart Helpdesk di UPT TIK UNS Menggunakan Algoritma Naive Bayes Classifier," in Seminar Nasional Aplikasi Teknologi Informasi (SNATi), 2017, pp. 39–43.

[11] Balakumar, P., Vinopraba, T., & Chandrasekaran, K. (2023). Deep learning based real time Demand Side Management controller for smart building integrated with renewable energy and Energy Storage System. Journal of Energy Storage, 58, 106412.

[12] Zhao, Jinghua & Dalin, Zeng & Liang, Shuang & Kang, Huilin & Liu, Qinming. (2021). Prediction model for stock price trend based on recurrent neural network. Journal of Ambient Intelligence and Humanized Computing. 12. 10.1007/s12652-020-02057-0.

[13] Saboor, K., Saboor, Q. U. A., Han, L., & Zahid, A. S. (2020). Predicting the stock market using machine learning: Long short-term memory. Electronic Research Journal of Engineering, Computer and Applied Sciences, 2(2020), 202-219.

Achmad Nurfaizi
Informatics Engineering Student, Mercu Buana University
141518310039@student.mercubuana.ac.id

Muhaimin Hasanuddin
Faculty of Computer Science, Mercu Buana University, Jakarta, Indonesia
Email: muhaimin.hasanudin@mercubuana.ac.id