Integra: Journal of Integrated Mathematics and Computer Science
Journal Homepage: https://integrajimcs.com/index.php/integra/index
integra
Journal of Integrated Mathematics and Computer Science

Research Paper

Classifying Electronic Health Records with Long Short-Term Memory and Gated Recurrent Unit Networks

Dian Kurniasari[1*], Zida Bunga Sobara[1], Riza Sawitri[1]

[1] *Faculty of Mathematics and Natural Science, Universitas Lampung, Bandar Lampung, 35141, Indonesia*
[*] **Corresponding author**: dian.kurniasari@fmipa.unila.ac.id

Keywords
Electronic Health Records, Treatment Classification, LSTM, GRU

Abstract
Electronic Health Records (EHR) are a complete digital store of medical information which contains valuable information about a person's health status. These data are an important tool for health care professionals to decide the management of the patient, i.e. the need for hospitalization (inpatient care) or outpatient treatment. The importance of EHR data underscores the need for hospitals to act quickly and develop appropriate follow-up for patients. The aim of this study is to assess and compare the performance of two deep learning models, Long Short Term Memory (LSTM) and Gated Recurrent Unit (GRU) for accurate classification of EHR data to predict whether a patient is hospitalized (inpatient or outpatient) based on laboratory test results. This study aims to compare these models to identify the best way to classify patient treatment needs. The data set used in this study is EHR data from a private hospital in Indonesia which contains haematocrit, haemoglobin, erythrocytes, leukocytes, thrombocytes, MCH, MCHC, MCV, age, gender, and source (inpatient or outpatient). The data was pre-processed by labelling, variable selection and handling missing and duplicate values. The data set is split for training and testing the models, LSTM and GRU. Performance evaluation is based on accuracy, training time and other relevant metrics to evaluate both the classification precision and computational efficiency of the models. The classification accuracies of the two models were close to each other with LSTM being 75.57% and GRU being 75.34%. However, GRU showed better training efficiency, executing faster than LSTM model. In the cases where the training time efficiency is the priority, the GRU model is the better choice for its faster speed. However, LSTM performs slightly better in terms of accuracy, especially in classifying outpatient cases, making it more appropriate for applications where accuracy is of high importance.

Received: 2 May 2026, Accepted: 12 July 2026
https://doi.org/10.26554/integra.20263262

1. INTRODUCTION

Health information systems are essential to decision-making at all levels of healthcare governance. One of the important elements of these systems is the Electronic Health Record (EHR) which has been widely adopted in the healthcare industry in Indonesia as part of the digital transformation. EHRs are electronic copies of patients' health records that include important information such as demographics, medical history, medications, diagnostic tests, immunizations, allergies, radiology images, lab results, and treatment plans. The shift from paper-based to electronic records is seen as more efficient, offering cost savings and enhancing care quality [1, 2]. Furthermore, EHRs provide valuable insights into disease management, as they enable the comprehensive digital collection of health data, which is essential for informed medical decision-making.

EHR has been applied in various contexts, including med-

ical concept extraction, patient trajectory modelling, disease prediction, and data-driven clinical decision support. The data derived from these records often serve as a critical resource for physicians in determining appropriate treatment plans, whether inpatient or outpatient, ensuring that decisions align with each patient's clinical condition. As a result, classifying patients based on EHR data is crucial for enabling informed clinical decisions. Traditionally, EHR analysis employs basic statistical methods or machine learning algorithms like logistic regression, Support Vector Machines (SVM), and Decision Trees. For instance, Sadikin and Nurhaida [3] reported classification accuracies of 63.39% for Decision Trees, 68.95% for Gaussian Naïve Bayes, and 71.58% for Random Forest on EHR datasets. Additionally, Rosita et al. [4] found that neural networks achieved the highest F1-score of 67% for predicting inpatient versus outpatient care needs. However, these methods often struggle with high-dimensional data and rely on statistical assumptions, highlighting the need for more robust approaches that account for complex, nonlinear interactions in medical data analysis.

Over the past decade, Deep Learning (DL) has demonstrated remarkable performance across a range of applications, including natural language processing, medical image analysis, and genomics. The adoption of DL for analyzing EHR has grown significantly, with key architectures like Feed-Forward Neural Networks (FFNN), Convolutional Neural Networks (CNN), and Recurrent Neural Networks (RNN) widely applied to model and analyze EHR data [5]. Since EHR data typically represents a patient's medical history as a sequence of visits, with each visit capturing specific medical events, DL models designed for sequential data, such as RNN, are frequently employed in this context [6].

RNNs are a class of DL models specifically designed to process sequential data. Unlike feed-forward neural networks, RNNs can retain information from previous inputs through their internal states, making them well-suited for tasks like natural language processing and time series forecasting [7]. RNNs were first introduced in the 1970s by Werbos with the Backpropagation Through Time (BPTT) algorithm. However, they faced practical challenges, such as the vanishing gradient problem. Significant progress was made with the development of Long Short-Term Memory (LSTM) by Hochreiter and Schmidhuber [8] and Gated Recurrent Units (GRU) by Cho et al. [9], offering a simpler architecture with comparable performance.

Smagulova and James [10] state that LSTM was developed to address the vanishing gradient issue commonly encountered in RNNs, especially when dealing with long sequences, which limits RNNs' ability to capture long-term dependencies. GRU, a simplified variant of LSTM, offers a more streamlined approach. Its relative simplicity often results in faster computation across multiple datasets. However, determining which method performs better remains inconclusive. Like LSTM, GRU also tackles the vanishing gradient problem in RNN [11]. Furthermore, both LSTM and GRU are capable of analyzing various data types, including population data, lab results, medical images, and clinical records.

The utilization of LSTM and GRU methodologies has been extensively observed across diverse domains, including healthcare, economics, and business. For instance, Ashfaq et al. [12] employed the LSTM model to forecast unscheduled readmissions within 30 days, leveraging EHR data. Additionally, they introduced cost-sensitive formulations of MSTM networks that integrate expert features and contextual embeddings of clinical concepts. The findings of this research underscore the potential for significant annual cost savings should the model be implemented in real-world hospital settings. Yang et al. [13] evaluate the performance discrepancies between two DL models by taking into account factors such as dataset size for training, the nature of long and short texts, and a quantitative assessment across five metrics: processing speed, accuracy, recall, F1 score, and AUC. The dataset utilized originates from Yelp Inc. The findings indicate that GRU outperforms LSTM in terms of processing speed, being 29.29% quicker with the same dataset. While GRUs demonstrate superior performance with long texts and smaller datasets, LSTMs perform better under other circumstances. Furthermore, the performance-to-computational cost ratio for GRUs surpasses that of LSTMs.

Men et al. [14] applied an LSTM model to forecast a range of future disease diagnoses by analyzing EHR datasets from prominent hospitals in Southern China. This dataset encompassed over 5 million clinical visits recorded between July 2007 and July 2016, including patient demographic information, clinical variables, and diagnostic outcomes. Evaluation results indicated that the proposed LSTM model significantly surpassed several traditional and DL approaches in predicting future disease diagnoses. Notably, the F1 scores improved from 78.9% and 86.4% for conventional and DL models, respectively, to 88.0% with the LSTM approach. Moctezuma et al. [15] present a novel method for classifying sleep stages that leverages a computationally efficient permutation-based channel selection technique alongside the capabilities of DL, particularly the GRU model. This study evaluates various combinations of electroencephalography (EEG) channels using datasets from the International Institute for Integrative Sleep Medicine (WPI-IIIS) at the University of Tsukuba, Japan. The results indicate a significant decline in performance when fewer than three channels are utilized. In comparison, a random selection of three channels yielded predictions that were comparable to or exceeded those obtained using the three channels recommended by the American Academy of Sleep Medicine (AASM). The GRU effectively captures essential temporal information from EEG data, proving adept at identifying patterns associated with each sleep stage.

Saha et al. [16] introduced a GRU-based model, termed mGRU-CP, aimed at analyzing patient embeddings and predicting survival probabilities by leveraging EHR data from the COVID-19 Research Data Commons. Their research compared the performance of mGRU-CP against four advanced baseline methods using three distinct sets of patient features. The experimental findings demonstrate that mGRU-CP either matches or exceeds the performance of the baseline methods. Furthermore, the patient embeddings generated offer valuable insights

into mortality risks, thereby equipping healthcare professionals with computational tools for forecasting patient outcomes and optimizing future health resource allocation.

Despite their widespread application, both models remain underutilized in the classification of patient EHR from laboratory examinations when compared to other domains. Consequently, this study employs EHR data derived from laboratory test results at a private hospital in Indonesia, utilizing the LSTM and GRU methods to identify the optimal model for each approach based on accuracy outcomes.

2. RESEARCH METHODS

This study focuses on analyzing an Electronic Health Record (EHR) dataset to classify patients as either inpatients or outpatients based on clinical variables. The dataset includes variables such as Hematocrit, Hemoglobin, Erythrocyte, and others, with the Source variable acting as the binary target for classification. Data pre-processing steps, including labelling, variable selection, and integrity checks, are applied to prepare the dataset for model training. Two deep learning models, LSTM and GRU, are employed for the classification task. The methodology is illustrated in Figure 1, which provides a visual representation of the research process.

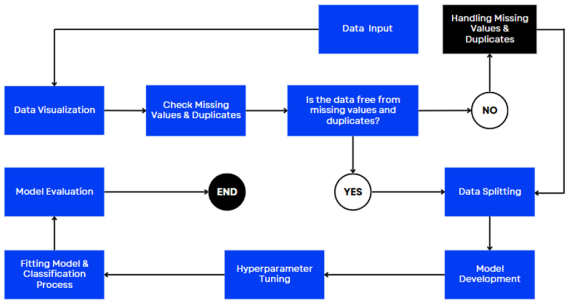

Figure 1. Research Method

2.1 Dataset Description and Pre-Processing

The EHR dataset comprises eleven variables: Hematocrit, Hemoglobin, Erythrocyte, Leucocyte, Thrombocyte, Mean Corpuscular Hemoglobin (MCH), Mean Corpuscular Hemoglobin Concentration (MCHC), Mean Corpuscular Volume (MCV), Age, Gender, and Source. The Source variable serves as a binary target, with "in" indicating inpatient status and "out" signifying outpatient status. The Gender variable is coded as "F" for females and "M" for males. A sample of the EHR data is presented in Table 1.

Hematocrit, hemoglobin, and erythrocyte levels are critical indicators of blood health and can signify the presence of anemia, frequently necessitating hospitalization. In contrast, leukocytes and thrombocytes play a pivotal role in identifying infections or clotting disorders, which may also impact clinical decisions. The variables MCH, MCHC, and MCV offer further insights into the specific type of anemia a patient may be experiencing. Additionally, age and gender are influential factors that can modify health risks, while the variable 'Source' acts as a classification target to ascertain the patient's treatment status. By comprehensively analyzing these variables, classification models can be developed to distinguish between inpatient and outpatient requirements more precisely.

The initial phase of data analysis involves pre-processing the data, encompassing tasks such as labelling, variable selection, data cleaning, data transformation, and checking for missing values [17]. In this research, the pre-processing process will commence with labelling. Data labelling is essential for identifying raw text and appending information labels, thereby furnishing the context required for machine learning [18]. Labeling will be applied to the "Sex" and "Source" columns. The "Sex" variable is categorized into two labels: F (Female) and M (Male). Utilizing the Sci-kit Learn library, precisely the LabelEncoder function, these categorical values are transformed into numerical representations, with F assigned a value of 0 and M a value of 1. Likewise, the "Source" column is encoded into a binary format, where the label "in" is designated a value of 0 and "out" is assigned a value of 1. This process is crucial for streamlining the modelling phase, as classification models typically perform better with numerical data.

The subsequent step involves the identification of variables to distinguish between feature variables and response variables. The selected features include all relevant independent variables such as Haematocrit, Haemoglobins, Erythrocytes among others. The Source column, on the other hand, is the dependent variable that we want to predict. This process is important to ensure that the model is trained only by variables that provide relevant information to the model to improve model performance [19, 20, 21]. The variable definition syntax defines and classifies features and targets. This provides a clear structure to the data.

After the variable selection process, EHR data was analysed further by using visualization techniques to explain the distribution of inpatients and outpatients. The graphical representation of the data was done with the help of the bar chart in Figure 2 and Table 2 which helps in comparison of the number of patients in both categories.

The visualization results indicate that the EHR data is relatively balanced, with a more significant number of outpatients (class 1) compared to inpatients (class 0). This observation is significant, as class imbalances can adversely impact the performance of classification models, potentially leading to biased predictions and diminished accuracy for minority classes [22, 23]. Thus, this factor must be taken into account in the subsequent stages of analysis. The data reveals a difference of 844 patients between the two classes.

Following the visualization phase, a critical aspect of preprocessing involves assessing the presence of missing values and duplicate entries. Missing values refer to instances where crucial information for data analysis is unrecorded, hidden, or lost [24, 25]. This assessment is essential for ensuring data integrity prior to model training, as unaddressed missing values

Table 1. Sample of EHR Data

Haematocrit	Haemoglobins	Erythrocyte	Leucocyte	Thrombocyte	Mch	Mchc	Mcv	Age	Sex	Source
35.1	11.8	4.65	6.3	310	25.4	33.6	75.5	1	F	out
43.5	14.8	5.39	12.7	334	27.5	34.0	80.7	1	F	out
33.5	11.3	4.74	13.2	305	23.8	33.7	70.7	1	F	out
39.1	13.7	4.98	10.5	366	27.5	35.0	78.5	1	F	out
30.9	9.9	4.23	22.1	333	23.4	32.0	73.0	1	M	out
...	...	...	...	...	...	...	...	...	...	...
...	...	...	...	...	...	...	...	...	...	...
32.8	10.4	3.49	8.1	72	29.8	31.7	94.0	92	F	in
33.7	10.8	3.67	6.7	70	29.4	32.0	91.8	92	F	in
33.2	11.2	3.47	7.2	235	32.3	33.7	95.7	93	F	out
31.5	10.4	3.15	9.1	187	33.0	33.0	100.0	98	F	in
33.5	10.9	3.44	5.8	275	31.7	32.5	97.4	99	F	out

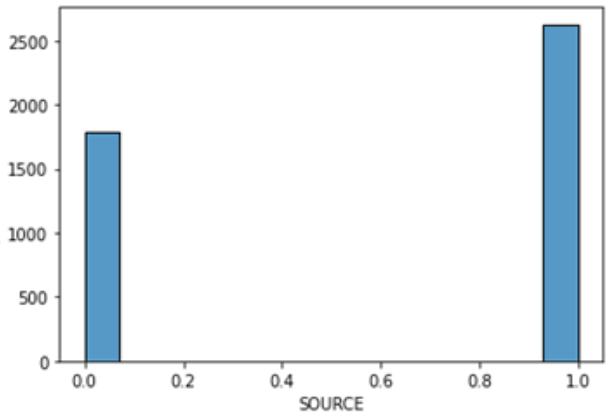

Figure 2. Data visualisation

Table 2. Comparison of Patient Count

Source	Number Patients
0	1784
1	2628

can significantly compromise the model's accuracy [26, 27]. Furthermore, verifying the absence of duplicate data guarantees the uniqueness of the dataset, preventing any disruptions in the model training process. The findings regarding missing values and duplicate data are summarized in Table 3, which indicates that no variables contain missing or duplicate entries.

2.2 Deep Learning Models

The DL model developed in this study includes two types of RNNs, namely the LSTM and GRU models. LSTM is a variation of RNN and has extra memory cells which allow to store information for longer time [28]. It has three specific gates: the input gate, the forget gate, and the output gate. One of the main advantages of LSTM is its ability to remember information from

Table 3. Results of Missing Value and Duplicate Data Analysis

Variable	Count of Missing Values	Count of Duplicate Entries
Hematocrit	0	0
Hemoglobin	0	0
Erythrocyte	0	0
Leucocyte	0	0
Thrombocyte	0	0
MCH	0	0
MCHC	0	0
MCV	0	0
Age	0	0
Sex	0	0
Source	0	0

previous time steps via this gating mechanism.

Each gate has a special function in the information processing sequence. The forget gate decides which information to keep or discard from the previous cell state, the input gate controls the incorporation of new information and the output gate generates an output based on the hidden and cell states [29, 30]. The values of these gates are calculated using the Equations 1, 2 and 3.

$$f_t = \sigma\left(W_f \cdot x_t + U_f \cdot h_{t-1} + b_f\right) \quad (1)$$

$$i_t = \sigma\left(W_i \cdot x_t + U_i \cdot h_{t-1} + b_i\right) \quad (2)$$

$$o_t = \sigma\left(W_o \cdot x_t + U_o \cdot h_{t-1} + b_o\right) \quad (3)$$

where, f_t is the forget gate, i_t is the input gate and o_t is the output gate. W is the weight assigned to each gate in relation to the input, U is the weight associated with the previous output for each gate, and b is the bias value associated with each gate. Figure 3 presents a visual representation of the LSTM model.

On the contrary, the GRU is a simpler version of the LSTM

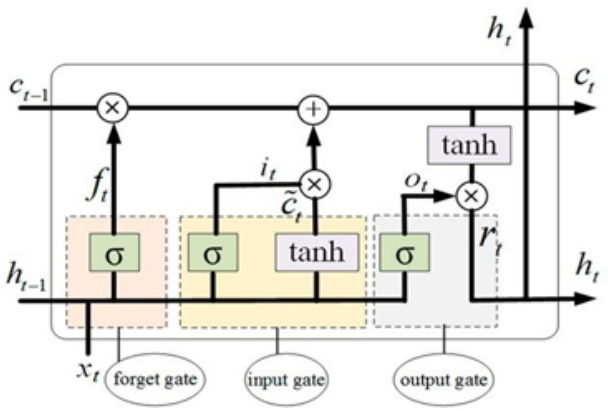

Figure 3. LSTM Architecture [29]

model which is mainly characterized by the lack of a cell state [31]. With the GRU only the hidden state is used to pass information to the next time steps. It has two gates: a reset gate and an update gate. The reset gate determines the amount of previous information to forget, and the update gate determines the flow of information from previous activations, deciding how much information should be retained for the next timestep [32]. The values of the reset gate can be derived from Equations 4 and 5:

$$r_t = \sigma\left(W_r \cdot x_t + U_r \cdot h_{t-1} + b_r\right) \tag{4}$$

$$z_t = \sigma\left(W_z \cdot x_t + U_z \cdot h_{t-1} + b_z\right) \tag{5}$$

Generally, GRU exhibits superior processing speeds compared to LSTM networks due to their reduced number of parameters. This property makes the model easy to implement and computationally efficient [33]. The architecture of the GRU is visualized in Figure 4.

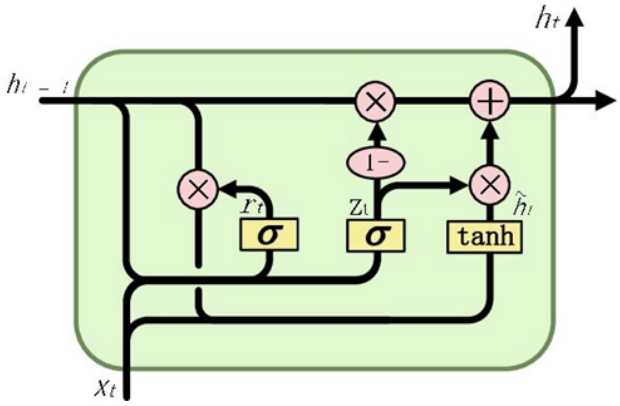

Figure 4. GRU Architecture [32]

In this process, the classification models are built using the training data and then tested using the test data. The data was collected by a method called train test split. This method helps to separate data into different sets to avoid the problem of overfitting. Overfitting is when the model is too specific to the training data and does not generalize well to the new data [34]. Usually, the dataset is partitioned into two segments with varying ratios, such as 10:90, 20:80, 30:70, 40:60, 50:50, 60:40, 70:30, 80:20, and 90:10, where the training dataset serves to construct the model, while the test dataset is employed to evaluate its predictive performance [35]. However, Muraina [36] argued that the training set should be the majority of the data.

The data set was divided for the purpose of this study into a 90% training set and a 10% testing set. The idea that a bigger training data set enables the model to learn from existing patterns is the basis of this decision. The results of train-test split with quantities of training and testing data are shown in Table 4.

Table 4. Comparison of Patient Quantition

Data	Total
Train	3970
Test	442

Furthermore, to find the best parameter combination and to avoid the manual trial-and-error approach, critical parameters, such as the number of LSTM units and batch size, are first determined through a hyperparameter tuning process using GridSearchCV. This approach performs an exhaustive search of the hyperparameter space by individually testing each combination and validating the associated configurations. Many researchers [37, 38, 39, 40]], suggest that the hyperparameter tuning is an important step in improving the model performance that can be evaluated in the evaluation phase. The specified parameter values for the two models are shown in Table 5.

Table 5. Parameter Values for Hyperparameter Optimisation

LSTM or GRU	Batch Size
64	16
128	32
256	64

The model evaluation was conducted with the following metrics: Accuracy, Recall, Precision and F1 Score. These metrics are obtained from Equations 6, 7, 8, and 9. They calculate this using value like True Positive (TP), True Negative (TN), False Positive (FP) and False Negative (FN) from the confusion matrix. The confusion matrix is a complete summary of performance from the evaluation of test data as Ting [41] states. This two-dimensional matrix is a representation of one dimension as the actual class and the other as the predicted class from the model.

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{FN} + \text{TN}} \tag{6}$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{7}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{8}$$

$$\text{F1 Score} = 2 \times \frac{(\text{Recall} \times \text{Precision})}{(\text{Recall} + \text{Precision})} \tag{9}$$

3. RESULT AND DISCUSSIONS

3.1 Result

In this study, the aim is to find the best model for classifying inpatients and outpatients by two methods LSTM and GRU. This process begins with hyperparameter tuning to find the best parameters for each model. The LSTM model is subject to systematic tests of combinations of hyperparameters, each of which is validated. The results show that the best performing LSTM has 128 units and a batch size of 64, reaching a maximum accuracy of 75.57%. The hyperparameter tuning process of the LSTM model took 1 hour, 4 minutes, and 27 seconds. With the implementation of the early stopping technique, training ended at the 87^{th} epoch when this accuracy was reached.

On the other hand, the GRU model used in this study has a faster hyper tuning process which took 47 minutes and 24 seconds. The best GRU model has 64 units with a batch size of 32, with a slightly lower accuracy of 75.34%. The GRU achieves accuracy comparable to the LSTM, as the evaluation metrics for the GRU model show an average accuracy of 76.5%, recall of 71% and F1 score of 72%. Thus, the GRU model is more time efficient in the training phase, while the accuracy performance of both models is similar. The results are presented in detail in Figure 5 and Tables 6-8.

Table 6. Comparative Evaluation of Hyperparameter Tuning Results for LSTM and GRU Models

LSTM/GRU Unit	Batch Size	LSTM Accuracy	GRU Accuracy
64	16	0.734761	0.732746
64	32	0.739798	0.739547
64	64	0.733249	0.737783
128	16	0.739295	0.733753
128	32	0.735768	0.737280
128	64	0.742317	0.735516
256	16	0.735516	0.735013
256	32	0.732997	0.739043
256	64	0.740806	0.738287

After identifying the best LSTM and GRU models, a confusion matrix was used to evaluate the models' classification performance. This matrix shows the predictions of the models against the actual conditions including TP, TN, FP and FN. TP and TN contain important information in the field of patient classification, which is crucial for patient care. FP and FN, on the other hand, can have serious consequences. For example, the wrong classification of a patient who should be treated as outpatient as requiring inpatient care may lead to additional costs of inpatient services and the inconveniences of wrong treatment. In addition, this scenario requires the family of the patient to go to hospitals which further adds to the emotional and financial burden. On the other hand, misclassifying a patient who requires hospitalization as an outpatient can impede their recovery by depriving them of necessary medical attention.

Table 7. Performance Metrics of the LSTM Model

Class	Precision (%)	Recall (%)	F1-Score (%)
0	81	50	62
1	74	92	82
Average	77.5	71	72

Table 8. Performance Metrics of the GRU Model

Class	Precision (%)	Recall (%)	F1-Score (%)
0	79	51	62
1	74	91	82
Average	76.5	71	72

3.2 Discussion

The results indicate that both the LSTM and GRU models exhibit comparable effectiveness in classifying inpatients and outpatients. Specifically, their accuracy rates are nearly identical, with the LSTM achieving 75.57% and the GRU 75.34%. However, when evaluating other factors, particularly training efficiency, the GRU model demonstrates a clear advantage. The hyper tuning process for the GRU requires 47 minutes and 24 seconds, whereas the LSTM takes 1 hour, 4 minutes, and 27 seconds. Furthermore, the prediction runtime for the GRU model is significantly shorter at 41 seconds, in contrast to the LSTM, which requires 2 minutes and 56 seconds.

Based on the evaluation of additional metrics, including precision, recall, and F1-score, the LSTM model demonstrates a marginal advantage in several areas, particularly in precision for class 0, achieving 81% compared to 79% for the GRU. Furthermore, the LSTM model exhibited remarkable performance in class 1, especially regarding recall, with scores of 92% for LSTM versus 91% for GRU. These results suggest that the LSTM model is more effective in retaining information, resulting in fewer classification errors for inpatients.

Considering different factors such as accuracy, time efficiency, and evaluation metrics, the GRU has a better training efficiency and speed, while the LSTM has a marginal advantage in some accuracy metrics. If the goal is to get quick results with similar performance then GRU is the way to go. However, LSTM may be more advantageous in cases where a bit more precision is required, especially in outpatient classification. Both models provide a good compromise, but in practice, GRUs are usually more efficient and still deliver accurate results, being thus the best

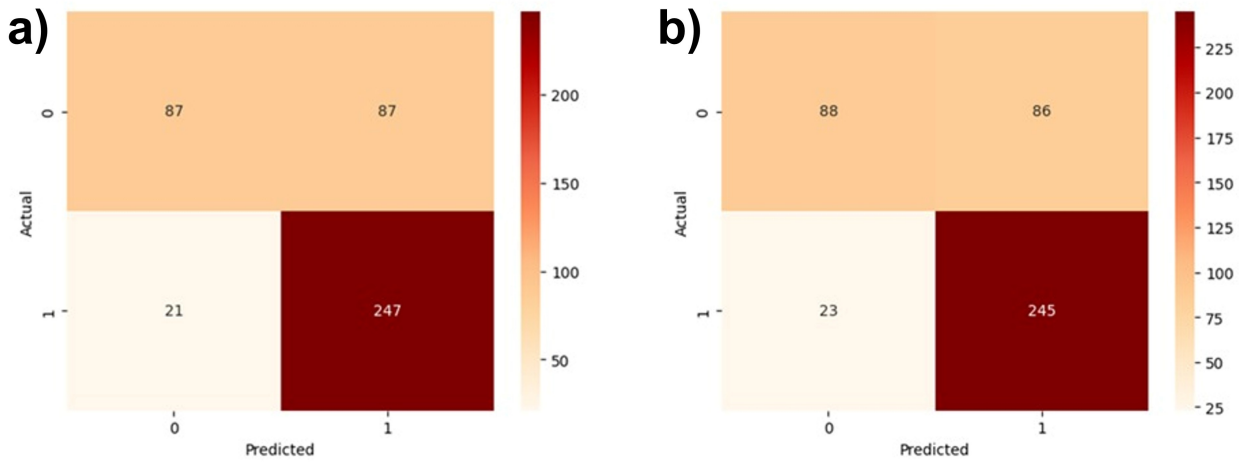

Figure 5. Comparison of the Confusion Matrix Model for the (a) LSTM and (b) GRU

option for patient classification in health systems that require quick and precise responses.

This study supports previous studies that showed that LSTM and GRU models are effective for sequential data such as EHR. Some research suggests that LSTMs are often good at picking up complex patterns, especially those that happen over long periods of time. Other research shows that GRU are more computationally efficient, and can reach similar results to LSTM in more simple cases. These results support the current view that GRUs are more advantageous for time efficient applications, while LSTMs are more suitable for more complex cases that require high accuracy.

4. CONCLUSIONS

The results indicate that both the LSTM and GRU models exhibit comparable effectiveness in classifying inpatients and outpatients. Specifically, their accuracy rates are nearly identical, with the LSTM achieving 75.57% and the GRU 75.34%. However, when evaluating other factors, particularly training efficiency, the GRU model demonstrates a clear advantage. The hyper tuning process for the GRU requires 47 minutes and 24 seconds, whereas the LSTM takes 1 hour, 4 minutes, and 27 seconds. Furthermore, the prediction runtime for the GRU model is significantly shorter at 41 seconds, in contrast to the LSTM, which requires 2 minutes and 56 seconds.

Based on the evaluation of additional metrics, including precision, recall, and F1-score, the LSTM model demonstrates a marginal advantage in several areas, particularly in precision for class 0, achieving 81% compared to 79% for the GRU. Furthermore, the LSTM model exhibited remarkable performance in class 1, especially regarding recall, with scores of 92% for LSTM versus 91% for GRU. These results suggest that the LSTM model is more effective in retaining information, resulting in fewer classification errors for inpatients. Considering various factors such as accuracy, time efficiency, and evaluation metrics, the GRU demonstrates superior training efficiency and speed, while the LSTM exhibits a marginal advantage in specific accuracy metrics. If the primary objective is to achieve rapid results with comparable performance, the GRU is the preferred option. But in cases where a little more precision is needed, especially in classifying outpatients, LSTM might be more useful. Both models achieve a good balance, but in practice GRUs are usually more efficient while still providing accurate results, thus making them a better choice for patient classification in health systems that require fast and accurate responses.

This study is consistent with the previous studies that have shown that both LSTM and GRU models are able to deal with the sequential data such as EHR. Some research suggests that LSTMs are often good at picking up complex patterns, especially those that happen over long periods of time. On the other hand, some studies show that GRU can be more computationally efficient and can achieve similar results to LSTM in less complex scenarios. Hence, our results support the common believe that GRUs are more beneficial for time sensitive applications and LSTM is more suitable for complex application with high accuracy requirements.

5. ACKNOWLEDGEMENT

XX

REFERENCES

[1] Bambang Tutuko, Siti Nurmaini, Muhammad Naufal Rachmatullah, Annisa Darmawahyuni, and Firdaus Firdaus. A Deep Learning Approach to Integrate Medical Big Data for Improving Health Services in Indonesia. *Computer Engineering and Applications Journal*, 9(1):17–28, 2020.

[2] Ismail Keshta and Ammar Odeh. Security and Privacy of Electronic Health Records: Concerns and Challenges. *Egyptian Informatics Journal*, 22(2):177–183, 2021.

[3] Mujiono Sadikin, Ida Nurhaida, and Ria Puspita Sari. Exploratory Study of Some Machine Learning Techniques to

Classify the Patient Treatment. *International Journal of Advanced Computer Science and Applications*, 12(2), 2021.

[4] Rizka Rosita, Dwika Ananda Agustina Pertiwi, and Oktaria Gina Khoirunnisa. Prediction of Hospital Intensive Patients Using Neural Network Algorithm. *Journal of Soft Computing Exploration*, 3(1):8–11, 2022.

[5] Jose Roberto Ayala Solares, Francesca Elisa Diletta Raimondi, Yajie Zhu, Fatemeh Rahimian, Dexter Canoy, Jenny Tran, Ana Catarina Pinho Gomes, Amir H. Payberah, Mariagrazia Zottoli, Milad Nazarzadeh, et al. Deep Learning for Electronic Health Records: A Comparative Review of Multiple Deep Neural Architectures. *Journal of Biomedical Informatics*, 101:103337, 2020.

[6] Laila Rasmy, Jie Zhu, Zhiheng Li, Xin Hao, Hong Thoai Tran, Yujia Zhou, Firat Tiryaki, Yang Xiang, Hua Xu, and Degui Zhi. Simple Recurrent Neural Networks Is All We Need for Clinical Events Predictions Using EHR Data. *arXiv*, abs/2110.00998, 2021.

[7] Ibomoiye Domor Mienye, Theo G. Swart, and George Obaido. Recurrent Neural Networks: A Comprehensive Review of Architectures, Variants, and Applications. *Information*, 15(9):517, 2024.

[8] Sepp Hochreiter and Jürgen Schmidhuber. Long Short-Term Memory. *Neural Computation*, 9(8):1735–1780, 1997.

[9] Kyunghyun Cho, Bart van Merriënboer, Dzmitry Bahdanau, and Yoshua Bengio. On the Properties of Neural Machine Translation: Encoder–Decoder Approaches. In *Proceedings of the Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation (SSST-8)*, pages 103–111, 2014.

[10] Kamilya Smagulova and Alex Pappachen James. Overview of Long Short-Term Memory Neural Networks. In *Deep Learning Classifiers with Memristive Networks: Theory and Applications*, pages 139–153. Springer, 2019.

[11] Guizhu Shen, Qingping Tan, Haoyu Zhang, Ping Zeng, and Jianjun Xu. Deep Learning with Gated Recurrent Unit Networks for Financial Sequence Predictions. *Procedia Computer Science*, 131:895–903, 2018.

[12] Awais Ashfaq, Anita Sant'Anna, Markus Lingman, and Sławomir Nowaczyk. Readmission Prediction Using Deep Learning on Electronic Health Records. *Journal of Biomedical Informatics*, 97:103256, 2019.

[13] Shudong Yang, Xueying Yu, and Ying Zhou. LSTM and GRU Neural Network Performance Comparison Study: Taking Yelp Review Dataset as an Example. In *2020 International Workshop on Electronic Communication and Artificial Intelligence (IWECAI)*, pages 98–101. IEEE, 2020.

[14] Lu Men, Noyan Ilk, Xinlin Tang, and Yuan Liu. Multi-Disease Prediction Using LSTM Recurrent Neural Networks. *Expert Systems with Applications*, 177:114905, 2021.

[15] Luis Alfredo Moctezuma, Yoko Suzuki, Junya Furuki, Marta Molinas, and Takashi Abe. GRU-powered Sleep Stage Classification with Permutation-Based EEG Channel Selection. *Scientific Reports*, 14(1):17952, 2024.

[16] Arpita Saha, Maggie Samaan, Bo Peng, and Xia Ning. A Multi-Layered GRU Model for COVID-19 Patient Representation and Phenotyping from Large-Scale EHR Data. In *Proceedings of the 14th ACM International Conference on Bioinformatics, Computational Biology, and Health Informatics*, pages 1–6. ACM, 2023.

[17] Christo El Morr, Manar Jammal, Hossam Ali-Hassan, and Walid El-Hallak. *Machine Learning for Practical Decision Making*. Springer, 2022.

[18] Eunjung Kwon, Hyunho Park, Sungwon Byon, and Kyu-Chul Lee. Improving Text Classification Performance through Data Labeling Adjustment. In *2022 13th International Conference on Information and Communication Technology Convergence (ICTC)*, pages 2277–2279. IEEE, 2022.

[19] Anwar Ul Haq, Defu Zhang, He Peng, and Sami Ur Rahman. Combining Multiple Feature-Ranking Techniques and Clustering of Variables for Feature Selection. *IEEE Access*, 7:151482–151492, 2019.

[20] Amir Moslemi. A Tutorial-Based Survey on Feature Selection: Recent Advancements on Feature Selection. *Engineering Applications of Artificial Intelligence*, 126:107136, 2023.

[21] Dipti Theng and Kishor K. Bhoyar. Feature Selection Techniques for Machine Learning: A Survey of More than Two Decades of Research. *Knowledge and Information Systems*, 66(3):1575–1637, 2024.

[22] Wuxing Chen, Kaixiang Yang, Zhiwen Yu, Yifan Shi, and C. L. Philip Chen. A Survey on Imbalanced Learning: Latest Research, Applications and Future Directions. *Artificial Intelligence Review*, 57(6):137, 2024.

[23] Preston Billion Polak, Joseph D. Prusa, and Taghi M. Khoshgoftaar. Low-Shot Learning and Class Imbalance: A Survey. *Journal of Big Data*, 11(1):1, 2024.

[24] Roderick J. A. Little and Donald B. Rubin. *Statistical Analysis with Missing Data*. John Wiley & Sons, 3 edition, 2019.

[25] Ardalan Mirzaei, Stephen R. Carter, Asad E. Patanwala, and Carl R. Schneider. Missing Data in Surveys: Key Concepts, Approaches, and Applications. *Research in Social and Administrative Pharmacy*, 18(2):2308–2316, 2022.

[26] Mustafa Alabadla, Fatimah Sidi, Iskandar Ishak, Hamidah Ibrahim, Lilly Suriani Affendey, Zafienas Che Ani, Marzanah A. Jabar, Umar Ali Bukar, Navin Kumar Devaraj, Ahmad Sobri Muda, et al. Systematic Review of Using Machine Learning in Imputing Missing Values. *IEEE Access*, 10:44483–44502, 2022.

[27] Jiaxu Peng, Jungpil Hahn, and Ke-Wei Huang. Handling Missing Values in Information Systems Research: A Review of Methods and Assumptions. *Information Systems Research*, 34(1):5–26, 2023.

[28] Navin Kumar Manaswi. Basics of TensorFlow. In *Deep Learning with Applications Using Python: Chatbots and Face, Object, and Speech Recognition with TensorFlow and Keras*, pages 1–30. Apress, 2018.

[29] Mehrnaz Faraz, Hamid Khaloozadeh, and Milad Abbasi. Stock Market Prediction-Based on Autoencoder Long Short-Term Memory Networks. In *2020 28th Iranian Conference on Electrical Engineering (ICEE)*, pages 1–5. IEEE, 2020.

[30] Nan Luo, Xiaojing Zhong, Luxin Su, Zilin Cheng, Wenyi Ma, and Pingsheng Hao. Artificial Intelligence-assisted Dermatology Diagnosis: From Unimodal to Multimodal. *Computers in Biology and Medicine*, 165:107413, 2023.

[31] Gissel Velarde, Pedro Brañez, Alejandro Bueno, Rodrigo Heredia, and Mateo Lopez-Ledezma. An Open Source and Reproducible Implementation of LSTM and GRU Networks for Time Series Forecasting. *Engineering Proceedings*, 18(1):30, 2022.

[32] Rui-Qing Yan, Wei Liu, Meng Zhu, Yi-Jing Wang, Cong Dai, Shuo Cao, Kang Wu, Yu-Chen Liang, Xian-Chuan Yu, and Meng-Fei Zhang. Real-Time Abnormal Light Curve Detection Based on a Gated Recurrent Unit Network. *Research in Astronomy and Astrophysics*, 20(1):007, 2020.

[33] Divish Rengasamy, Mina Jafari, Benjamin Rothwell, Xin Chen, and Grazziela P. Figueredo. Deep Learning with Dynamically Weighted Loss Function for Sensor-Based Prognostics and Health Management. *Sensors*, 20(3):723, 2020.

[34] Xue Ying. An Overview of Overfitting and Its Solutions. In *Journal of Physics: Conference Series*, volume 1168, page 022022. IOP Publishing, 2019.

[35] Quang Hung Nguyen, Hai-Bang Ly, Lanh Si Ho, Nadhir Al-Ansari, Hiep Van Le, Van Quan Tran, Indra Prakash, and Binh Thai Pham. Influence of Data Splitting on Performance of Machine Learning Models in Prediction of Shear Strength of Soil. *Mathematical Problems in Engineering*, 2021:4832864, 2021.

[36] Ismail Muraina. Ideal Dataset Splitting Ratios in Machine Learning Algorithms: General Concerns for Data Scientists and Data Analysts. In *Proceedings of the 7th International Mardin Artuklu Scientific Research Conference*, pages 496–504, 2022.

[37] Li Yang and Abdallah Shami. On Hyperparameter Optimization of Machine Learning Algorithms: Theory and Practice. *Neurocomputing*, 415:295–316, 2020.

[38] Md Riyad Hossain and Douglas Timmer. Machine Learning Model Optimization with Hyperparameter Tuning Approach. *Global Journal of Computer Science and Technology: D Neural & Artificial Intelligence*, 21(2), 2021.

[39] Sebastian Simon, Nikolay Kolyada, Christopher Akiki, Martin Potthast, Benno Stein, and Norbert Siegmund. Exploring Hyperparameter Usage and Tuning in Machine Learning Research. In *2023 IEEE/ACM 2nd International Conference on AI Engineering – Software Engineering for AI (CAIN)*, pages 68–79. IEEE, 2023.

[40] Justus Ilemobayo, Olamide Durodola, Oreoluwa Alade, Opeyemi J. Awotunde, Adewumi T. Olanrewaju, Olumide Falana, Adedolapo Ogungbire, Abraham Osinuga, Dabira Ogunbiyi, Ark Ifeanyi, et al. Hyperparameter Tuning in Machine Learning: A Comprehensive Review. *Journal of Engineering Research and Reports*, 26(6):388–395, 2024.

[41] Kai Ming Ting. Confusion Matrix. In Claude Sammut and Geoffrey I. Webb, editors, *Encyclopedia of Machine Learning*, pages 209–210. Springer, 2011.

