Network Intrusion Detection System with Time-Based Sequential Cluster Models Using LSTM and GRU

Ravi Vendra Rishika [1)], Baskoro Adi Pratomo [2,*)], and Shintami Chusnul Hidayati [3)]

[1, 2, 3)] Department of Informatics, Institut Teknologi Sepuluh Nopember, Surabaya, Indonesia

E-mail: 6025211015@student.its.ac.id[1)], baskoro@its.ac.id[2)], and shintami@its.ac.id[3)]

ABSTRACT

Technological development and the growth of the internet today have a positive and revolutionary impact in various areas of human life, such as banking, health, science, and more. The presence of open data and open APIs also facilitates the exchange of data and information between entities without the restrictions imposed by different regions and geographical areas. However, information openness not only has a positive impact but also makes data vulnerable to data theft, viruses, and various other types of cyberattacks. The large-scale data exchange that occurs across the network poses a challenge in detecting unusual anomalous activity and new cyber attack methods. Therefore, the existence of an Intrusion Detection System (IDS) is urgently essential. The IDS helps system administrators detect cyber attacks and network anomalies, thus minimizing the risk of data leaks and intrusions. Scientists came up with a new way to use time-based sequentially clustered data sets in the Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU) models to make the new method. This IDS model was implemented using the CIC-IDS 2018 data set, which has more than 4 million data lines. We use the LSTM and GRU models' abilities and unique qualities to sort and figure out different attacks in IDS. This is done by putting sequential data sets in order by time and grouping them by destination ports and protocols, like TCP and UDP. The model was evaluated using the accuracy, precision, recall, and F-1 scores matrix, and the results showed that the time-based sequential clustered models in LSTM and GRU have an accuracy of up to 97.21%. This implies that the future IDS models can effectively utilize this new approach.

Keywords: CIC-IDS 2018, cyber security, data clustering, Gated Recurrent Unit (GRU), Intrusion Detection System (IDS), Long Short-Term Memory (LSTM), sequential model

1. Introduction

The internet is one of the most significant discoveries and breakthroughs in the last three decades. Today, the internet is used in various lines of human life, ranging from information technology, education, health, agriculture, and so on. The presence of the internet has changed the paradigm and civilization of humans massively. The data and information revolution that occurred in the 21st century cannot be separated from the presence of the internet [1]. Even the presence of the internet and the availability of adequate bandwidth are directly proportional to the increase in per capita income of a country [2].

As the internet grows and more technologies like smart systems and the Internet of Things (IoT) are used to control different devices, the risk of cyberattacks and intrusions also rises [3]. Quoting Derek Manky, Fortinet's global security analyst, "Every minute, we are seeing about half a million attack attempts that are happening in cyberspace" [4]. Therefore, the presence of security devices such as the Internet Detection System (IDS) is very crucial as an early detector of attacks or anomalies in the network. IDS is a device in the form of hardware or software that has the ability to check data and information traffic on a network, analyze each packet that passes through the network infrastructure, and detect intrusions or anomalies that occur in the network or system [5].

IDS is one of the most basic network security system devices and must be owned by every system because its presence is very crucial and helps administrators to take preventive steps in network security [6]. IDS plays a very

* Corresponding author.
Received: August 7th, 2024. Revised: December 26th, 2024. Accepted: January 1st, 2025.

Available online: February 25th, 2025.

DOI: 10.12962/j24068535.v23i1.a1241

important role in a network system, regardless of the sector, agency, or organization in which the network is located [7][8]. Not only in on-premises network architecture, IDS is also crucial in cloud computing architecture. Cloud computing is increasingly popular today with the presence of major players such as Amazon Web Services (AWS), Google Cloud Platform (GCP), Microsoft Azure, Alibaba Cloud, and IBM Cloud. However, data and information security issues are also challenges in cloud computing [9]. IDS plays a very important role, so it needs to be able to do a few specific things, like adapting to intrusions and meeting strict requirements. The analysis of more data traffic necessitates the construction of a stronger IDS architecture [10]. Various studies have been conducted to build a better IDS. An intrusion detection system (IDS) inspired and based on the human immune system was created to find attacks on computer networks. An IDS was used in networks by combining misuse detection with genetic network programming (GNP) [11] [12]. A study of the literature also looked at how well SNORT, ALAD, PHAD, LERAD, and NETAD could find attacks [13]. A suggested IDS model using N-Gram and Cosine Similarity was also brought up to make research on the IDS [14] better.

One of the latest algorithms that is quite popular among many researchers related to IDS is Long Short-Term Memory (LSTM) [15] and Gated Recurrent Unit (GRU) [16]. GRU is a development of the LSTM architecture where GRU has a simpler number of gates than LSTM [17]. Unlike the previous Feedforward Neural Network (FNN) and Recurrent Neural Network (RNN) architectures that do not have feedback capabilities and experience the main problem of vanishing gradients, both LSTM and GRU have the ability to remember previous data output results as input for the next data. This is possible with the feedback gate, and both algorithms are able to overcome the vanishing gradient problem that occurs in FNN and RNN [18]. This also makes it possible to produce better predictions for sequential data sets that are related between nodes, for example, rainfall data sets, weather forecasts, stock price fluctuations, and so on. The GRU architecture, with its two gates, update and reset, is simpler than the LSTM architecture, which has three gates: input, forget, and output. Due to these characteristics, the The GRU model training process is relatively faster, and the parameters required in modeling are also fewer [19]. Various studies also have been conducted on both LSTM and GRU in various research fields, such as improved lip reading using GRU [20], which also uses GRU to detect the lip movement. On the other hand, LSTM is also being used to predict wind speed forecasting [19]. Several studies have been conducted to show the capability of both LSTM and GRU.

In the realm of IDS research, a popular dataset used by many researchers in modeling is KDDCup-99 [21]. KDDCup-99 is a popular dataset in the IDS model creation process. This dataset has 4,898,431 labeled data and is a development of the DARPA98 dataset [22]. In addition to the KDDCup-99 dataset, a popular dataset used by many researchers is CIC-IDS 2018. The Canadian Institute of Cybersecurity built and published this dataset in 2018, updating the previous version from 2017. This dataset has a total of 10 files with the extension .csv, with a total of more than 10,000,000 rows of data and consists of 80 attribute columns [23]. Several studies have used this dataset, one of which is the study conducted by Liu et al., where they comprehensively compared several angles between the CIC-IDS 2017 and CIC-IDS 2018 datasets [24].

From several studies that have been conducted previously, the majority of studies use the original CIC-IDS 2018 data set without going through any processing or stages such as sorting or grouping (clustering) of the data set. The LSTM and GRU methods used in this study work better with sequential or time-series data sets [19]. Using sequential data sets based on time can help LSTM and GRU make better models.

As a result, this study looks into IDS modeling using the LSTM and GRU method approaches. It does this by using the CIC-IDS 2018 data set, which has been sorted by time attributes and grouped or clustered by destination ports and protocols. The three attributes used, namely time, destination port, and protocol, are some of the attributes owned by the CIC-IDS 2018 dataset.

Putting in place sorting and clustering of datasets based on time, destination port, and protocol attributes should make the IDS model built more accurate and faster, allowing it to be used as a real IDS model in the future.

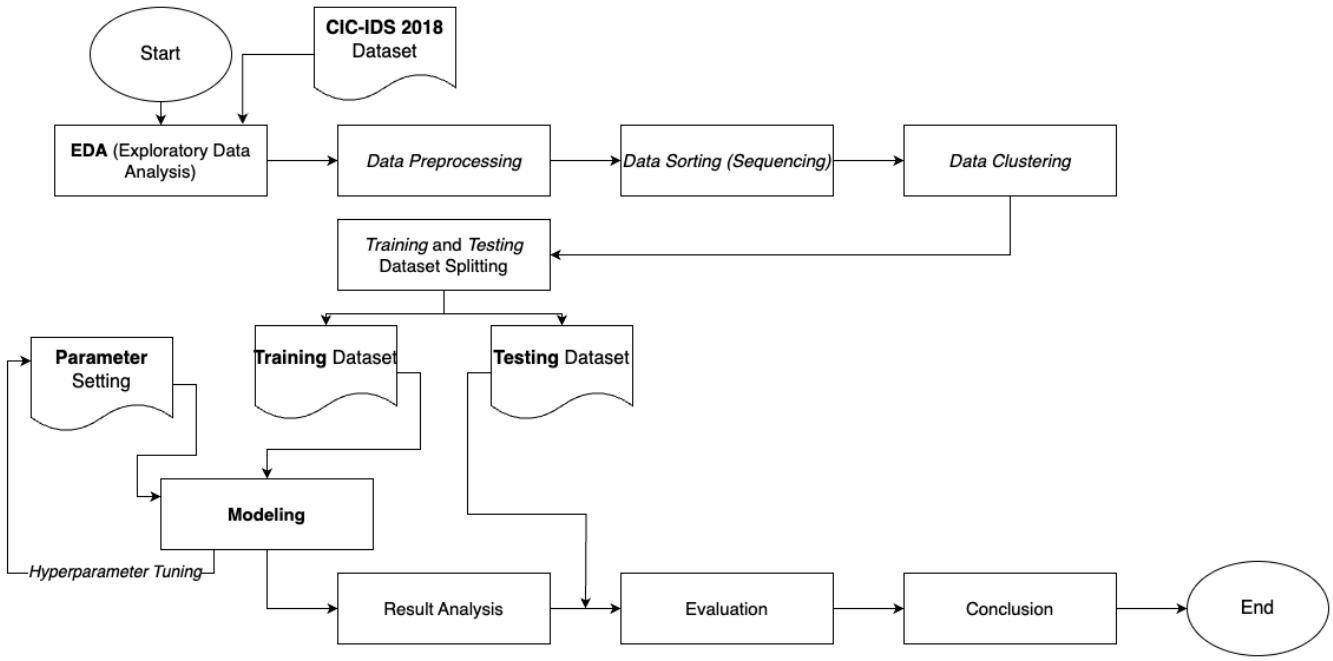

Fig. 1: Research stages.

2. Research Methodology

The CIC-IDS 2018 dataset is used in this study, which is sorted by time attributes and clustered by destination port and protocol attributes. The LSTM and GRU methods are used for IDS modeling. The stages carried out in this study include data preparation, exploratory data analysis (EDA), data preprocessing, data sorting, data clustering, separation of training and testing data sets, and modeling using LSTM and GRU. The stages of sorting and clustering data based on time attributes, destination ports, and protocols are crucial points raised in this study.

The last stage carried out is to evaluate the model built using the accuracy matrix (accuracy, precision, recall, and F1 score) and performance based on time. All stages carried out in this study are written using the Python programming language. Fig. 1 shows the stages carried out in this study.

2.1. Problem Identification

The experimental stages carried out in this study were to analyze and evaluate the role of data sorting and clustering on the results of IDS modeling using the LSTM and GRU methods.

2.2. Literature Review

Literature review of previous studies related to IDS, LSTM, GRU, and CIC-IDS 2018. Additionally, it pertains to the data sorting and clustering stages implemented in this study. The purpose of this literature review stage is to find and identify topics that are similar or related to the research being conducted, so that it can sharpen this research and enrich the treasury of previous research on IDS.

2.3. Data Analysis and Processing

A. Data Preparation

This stage is the stage of preparing data, namely the CIC-IDS 2018 dataset. This dataset is very large, consisting of 10 files with the extension CSV, consisting of 80 attribute columns and having more than 10,000,000 rows of data. [23].

B. Exploratory Data Analysis (EDA)

This stage is the initial stage that aims to analyze the CIC-IDS 2018 dataset and identify dataset patterns. In addition, it can also show the distribution of data. In the CIC-IDS 2018 dataset, Fig. 2 and Fig. 3 show examples of how labels and protocols are spread out.

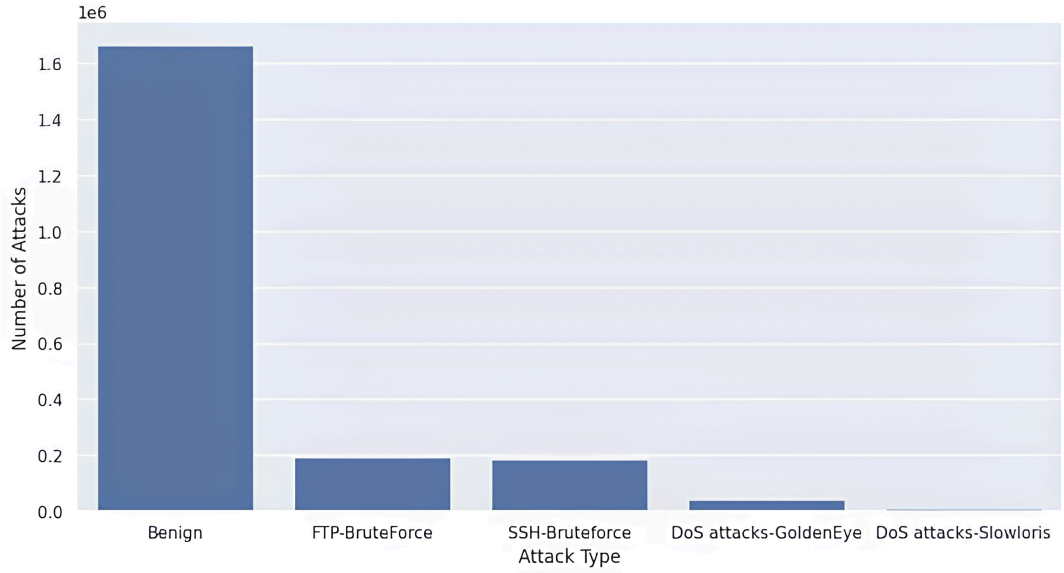

Fig. 2: Distribution of label attribute on CIC-IDS 2018 dataset.

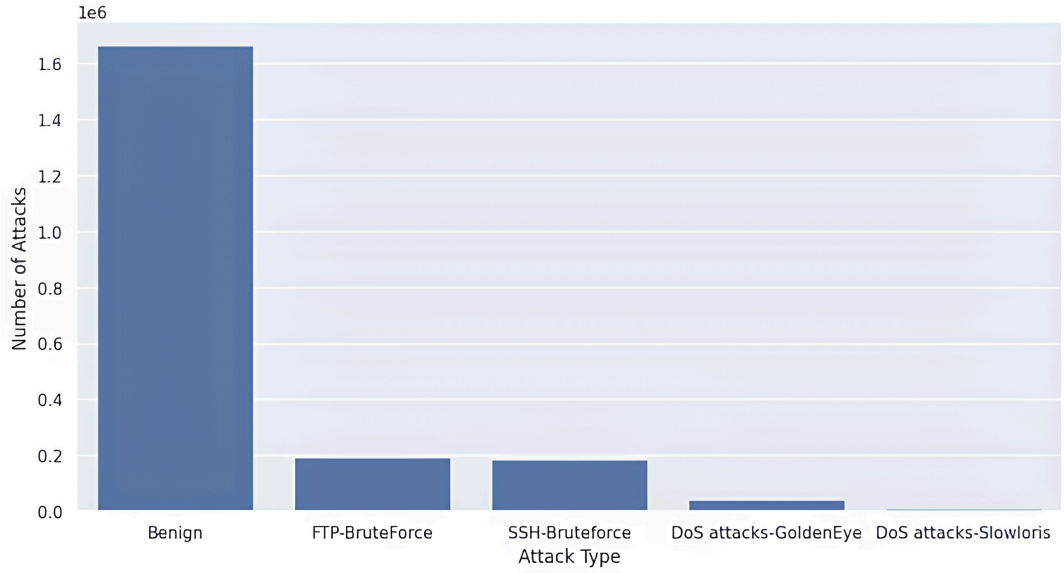

Fig. 3: Distribution of protocol attribute on CIC-IDS 2018 dataset.

C. Data Preprocessing

The data preprocessing stage is a very important stage in every study that uses machine learning modeling. This stage includes the imputation stage, namely to clean up missing value data, outliers, noise, NaN (not a number), or infinite numbers. The existence of missing or inappropriate data will affect the quality and performance of the resulting model. On top of the imputation steps mentioned above, this study also changes categorical attribute values to numerical ones so that these attributes can be used in LSTM or GRU modeling. This value conversion can also facilitate analysis and improve the quality and accuracy of the model being built.

D. Data Sorting

In this study, the data sorting stage is a crucial point because this stage will produce a sequential data set based on the time attribute (timestamp). In the CIC-IDS 2018 data set, the timestamp attribute represents the time when the attack was recorded by the system. However, in this data set, the time attribute is random, so it requires a data sorting process to produce sequential and sequential data [23].

This process is carried out because the LSTM and GRU methods used in this study have advantages over sequential or time-series data sets [19][25].

	Dst_Port_Protocol	Total
0	5317	531135
1	806	366368
2	4436	247753
3	33896	165567
4	226	118095
5	4456	97471
6	216	39426
7	00	30445
8	31286	11989
9	535517	11982

Fig. 4: CIC-IDS-2018 data clustering based on destination port and protocol attribute.

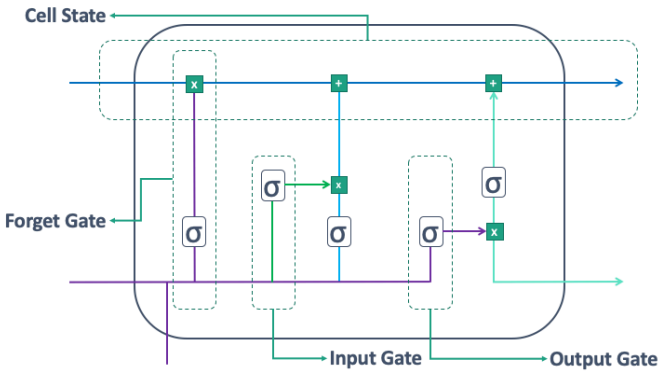

Fig. 5: LSTM gate diagram.

E. Data Clustering

This study is mostly about how to use the CIC-IDS 2018 data set's sorting and clustering stages based on time attributes, destination ports, and protocols. The clustering stage carried out is done by splitting the CIC IDS 2018 data set based on the destination port and protocol attributes.

This stage will later produce several data sets that are grouped based on the destination port and protocol attributes. In this study, the data sets used in the modeling are the three data sets that have the highest amount of data, as shown in Fig. 4. This data set will then be entered into the LSTM and GRU modeling that is built.

F. Long Short-Term Memory (LSTM)

In addition to the data analysis and processing processes that have been discussed in the previous sub-chapter, Long Short-Term Memory (LSTM) is the main highlight in this study. LSTM is an algorithm created by Hochreiter & Schmidhuber in 1997 [11]. This algorithm is a family of RNNs that were later developed and popularized by many researchers in the world. Like RNN, LSTM consists of modules with repeated processing paths [19].

LSTM is present as a solution to the vanishing gradient problem that occurs in RNN modeling [18]. LSTM is also able to show good accuracy results and is widely used in prediction models, including speech modeling and natural language [26][27]. Fig. 5 shows a schematic diagram of the LSTM architecture.

Basically, LSTM consists of three gates, namely forget, input, and output gates. The three determine which information should be forwarded as output [15]. The combination of these gates allows LSTM to store information from previous data to produce output values in the next data series. The advantage of this LSTM model is that it can be trained to store information for a fairly long period without consuming resources or time.

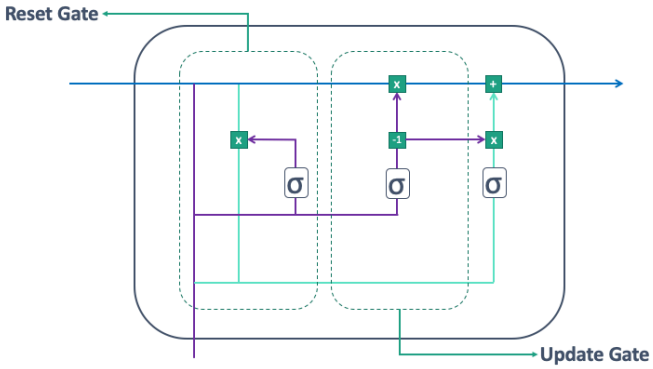

Fig. 6: GRU gate diagram.

Table 1: Parameter Setting on LSTM and GRU Modeling.

Item	Value
Metode	LSTM, GRU
Activation Function	Sigmoid, ReLU, Tanh, Linear, Softmax
Loss Function	Binary crossentropy, Binary focal crossentropy, Hinge
Optimizer	Adam
Lookback (Timesteps)	10
Epoch	10
Training: Testing	70%:30%

G. Gated Recurrent Unit (GRU)

In addition to LSTM, Gated Recurrent Unit (GRU) is also a method used and is a highlight in this study. GRU is a new algorithm that was born in 2014 [16]. This algorithm is a family of RNNs and a development of LSTM with a simpler architecture [19]. GRU is able to show good results on small data sets and is widely used in prediction models, speech modeling, and natural language, in line with LSTM [26][27]. Fig. 6 shows a schematic diagram of the GRU architecture.

Basically, GRU consists of two gates, namely update and reset gates. Both determine which information should be forwarded as output. The combination of these gates also allows GRU, as in LSTM, to store information from previous data to produce output values in the next data series. The advantage of this GRU model is that it can be trained to store information for a fairly long period without consuming resources or time [16].

H. Model Parameter Setting

This study requires several parameters, particularly for the LSTM and GRU modeling stage. Table 1 shows several parameters set in this study.

I. Model Evaluation

At this stage, an evaluation of the modeling results will be carried out. In research using machine learning, there are several evaluation methods, including evaluation matrices and confusion matrices. The confusion matrix is able to measure how well the model is built and also predict the wrong value of a model [28]. In the confusion matrix, there are 4 labels, namely True Positive (TP), False Positive (FP), True Negative (TN), and False Negative (FN).

A confusion matrix helps understand the performance and accuracy of a model and also helps determine the right evaluation metrics for a model; some evaluation metrics include $accuracy$, $precision$, $recall$, and $F1\text{-}score$. The model's predictions are judged by how accurate they are by comparing the total TP and TN values to all the data that is available [29].

$$accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{1}$$

Precision is the comparison between the TP value and all optimistic prediction results [30].

$$precision = \frac{TP}{TP + FP} \tag{2}$$

Recall is the comparison between the TP value and all actual positive values [30].

$$recall = \frac{TP}{TP + FN} \tag{3}$$

$F1\text{-}score$ is an evaluation metric that describes the balance between $precision$ and $recall$. This metric is able to see the negative potential generated by the FP and FN values [31]. Some of the metrics mentioned are benchmarks for how well the modeling is built.

$$F1\text{-}score = \frac{2 \times (precision \times recall)}{precision + recall} \tag{4}$$

3. Results and Discussion

3.1. Dataset

The steps used in this study to sort and group the CIC-IDS 2018 dataset will result in several separate datasets that are sorted by time attributes and grouped by destination port and protocol attributes. Fig. 4 shows a dataset that has been clustered based on destination port and protocol attributes. Taking the top 3 datasets with the most data is the next step.

The datasets that have the most data, as shown in Fig. 4, are as follows:

- Port 53/protocol 17 dataset (UDP/DNS).
- Port 80/protocol 6 dataset (TCP/HTTP).
- Port 443/protocol 6 dataset (TCP/HTTPS).-

We will then use the three datasets to construct IDS modeling using the LSTM and GRU methods.

3.2. LSTM

Following the steps of sorting and clustering the data set based on time, destination port, and protocol, the next step is to use the sorted and clustered data set to build an LSTM model, as explained in subsection 3.1. Modeling using the LSTM method on these 3 data sets uses several parameters as shown in Table 1. The results of modeling using the LSTM method produced can be seen and shown in Table 2.

3.3. GRU

After performing the modeling stages using the LSTM method, the next step is modeling using the GRU method, using the same data set as that used in the previous LSTM modeling. The parameters used in this GRU modeling also refer to Table 1. Table 3 displays the results of modeling using the GRU method.

3.4. Evaluation Matrix Results for Modeling

The model evaluation used a confusion matrix, evaluation metrics (accuracy, precision, recall, and F1 score), and performance based on time in milliseconds (ms). We will average the percentage of evaluation metrics with the mean weight. The average weight in a confusion matrix is a method for calculating the average performance of a classification model based on the number of samples in each class or label. This takes into account and shows the important role of each class by considering the number of examples included in that class. The confusion matrix also represents the results of the evaluation of the classification model created using test data and is used to provide a clearer picture of the accuracy of the model in predicting or classifying a data set.

From the stages of research and experiments that have been carried out, Table 4 shows the results of the evaluation of the IDS model built using the LSTM method, while Table 5 shows the results of the evaluation of the model built using the GRU method. From the experimental results shown in Tables 4 and 5, it shows that IDS

Table 2: Average Result Of LSTM Modeling Using 3 Different Sequenced and Clustered Datasets.

No	Activation Function	Loss Function	Training Accuracy	Validation Accuracy	Waktu (ms)
1	Sigmoid	Binary Crossentropy	97.08	97.14	2.65
2	Sigmoid	Binary Focal Crossentropy	97.07	97.15	2.76
3	Sigmoid	Hinge	97.06	97.13	2.57
4	ReLU	Binary Crossentropy	88.40	88.46	2.57
5	ReLU	Binary Focal Crossentropy	88.41	88.48	2.76
6	ReLU	Hinge	88.40	88.45	2.56
7	Tanh	Binary Crossentropy	97.21	97.27	3.19
8	Tanh	Binary Focal Crossentropy	97.20	97.27	2.27
9	Tanh	Hinge	97.21	97.28	3.11
10	Linear	Binary Crossentropy	96.18	96.24	2.61
11	Linear	Binary Focal Crossentropy	96.17	96.24	2.69
12	Linear	Hinge	96.17	96.25	2.53
13	Softmax	Binary Crossentropy	11.32	11.34	2.76
14	Softmax	Binary Focal Crossentropy	11.32	11.35	2.84
15	Softmax	Hinge	11.31	11.34	2.69

Table 3: Average Result Of GRU Modeling Using 3 Different Sequenced and Clustered Datasets.

No	Activation Function	Loss Function	Training Accuracy	Validation Accuracy	Waktu (ms)
1	Sigmoid	Binary Crossentropy	97.18	97.22	2.65
2	Sigmoid	Binary Focal Crossentropy	97.18	97.22	2.69
3	Sigmoid	Hinge	97.19	97.23	2.57
4	ReLU	Binary Crossentropy	86.91	86.98	2.65
5	ReLU	Binary Focal Crossentropy	86.92	86.98	2.69
6	ReLU	Hinge	86.92	86.98	2.61
7	Tanh	Binary Crossentropy	97.22	97.28	3.15
8	Tanh	Binary Focal Crossentropy	97.21	97.28	3.27
9	Tanh	Hinge	97.22	97.29	3.15
10	Linear	Binary Crossentropy	97.10	97.17	2.65
11	Linear	Binary Focal Crossentropy	97.10	97.17	2.72
12	Linear	Hinge	97.11	97.17	2.65
13	Softmax	Binary Crossentropy	11.32	11.35	2.69
14	Softmax	Binary Focal Crossentropy	11.32	11.35	2.72
15	Softmax	Hinge	11.32	11.35	2.61

modeling using the LSTM and GRU methods that utilize 3 subsets of CIC-IDS 2018 data that have been sorted by time attributes and clustered using destination port and protocol attributes shows good results in terms of accuracy metrics. The metric values in the table indicate that the modeling built and also the data sorting and clustering process played a role in producing good values. In addition, the IDS model built can also be applied to detect and classify attacks or intrusions that occur in a network system.

4. Discussion

In this study, the ratio of training and testing data sets used is 70:30, with 70% allocated for training data and 30% for testing data. We used this ratio for all LSTM and GRU modeling in 90 experiments that combined two

Table 4: Evaluation Result for LSTM Modeling.

Activation Function	Loss Function	Accuracy	Precision	Recall	F1 Score
Sigmoid	Binary Crossentropy	97.14	97.21	97.14	96.17
Sigmoid	Binary Focal Crossentropy	97.15	97.22	97.15	96.17
Sigmoid	Hinge	97.13	97.21	97.13	96.18
ReLU	Binary Crossentropy	88.46	89.20	88.46	85.75
ReLU	Binary Focal Crossentropy	88.48	89.21	88.48	85.77
ReLU	Hinge	88.45	89.20	88.45	85.75
Tanh	Binary Crossentropy	97.27	97.28	97.27	96.46
Tanh	Binary Focal Crossentropy	97.27	97.28	97.27	96.46
Tanh	Hinge	97.28	97.28	97.28	96.48
Linear	Binary Crossentropy	96.24	96.27	96.24	94.11
Linear	Binary Focal Crossentropy	96.24	96.27	96.24	94.11
Linear	Hinge	96.25	96.27	96.25	94.11
Softmax	Binary Crossentropy	11.34	12.20	11.34	9.53
Softmax	Binary Focal Crossentropy	11.35	12.20	11.35	9.53
Softmax	Hinge	11.34	12.20	11.34	9.53

Table 5: Evaluation Result for GRU Modeling.

Activation Function	Loss Function	Accuracy	Precision	Recall	F1 Score
Sigmoid	Binary Crossentropy	97.22	97.35	97.22	96.36
Sigmoid	Binary Focal Crossentropy	97.22	97.35	97.22	96.36
Sigmoid	Hinge	97.23	97.36	97.23	96.37
ReLU	Binary Crossentropy	86.98	89.23	86.98	82.64
ReLU	Binary Focal Crossentropy	86.98	89.24	86.98	82.64
ReLU	Hinge	86.98	89.23	86.98	82.64
Tanh	Binary Crossentropy	97.28	97.72	97.28	96.46
Tanh	Binary Focal Crossentropy	97.28	97.73	97.28	96.46
Tanh	Hinge	97.29	97.73	97.29	96.48
Linear	Binary Crossentropy	97.17	96.35	97.17	95.01
Linear	Binary Focal Crossentropy	97.17	96.35	97.17	95.01
Linear	Hinge	97.17	96.35	97.17	95.01
Softmax	Binary Crossentropy	11.35	12.20	11.35	9.53
Softmax	Binary Focal Crossentropy	11.35	12.20	11.35	9.53
Softmax	Hinge	11.35	12.20	11.35	9.53

algorithm types, three groups of data, five activation functions (sigmoid, ReLU, tanh, linear, and softmax), and three loss functions (binary cross-entropy, binary focal cross-entropy, and hinge).

This study uses LSTM, GRU, and the CIC-IDS 2018 data sets in a new way to model IDS. The methods and stages used are innovative. Many earlier studies, on the other hand, only used the CIC-IDS 2018 data set for the EDA process. To deal with NaNs, infinite numbers, missing values, and outliers, they used features or preprocessing. However, those prior studies did not execute clustering based on destination ports and protocols or sequential sorting based on time. For LSTM and GRU to be able to remember information from earlier data sets, they need to be fed sequential or time-series data sets. In this study, the sorting and clustering processes are very important because they are expected to make the final IDS model more accurate.

Of the 5 activation functions used, 3 activation functions, namely sigmoid, tanh, and linear, showed accuracy values above 96%. 1 activation function, namely ReLU, showed an accuracy value above 88%. While softmax only achieved the lowest accuracy value at 11%. The poor results produced by softmax make it possible to carry out a hyperparameter tuning process that is more appropriate for the characteristics of softmax in further research.

Of the 3 loss functions used, hinge tends to produce the best accuracy value and shorter computing time when compared to binary cross-entropy and binary focal cross-entropy, regardless of the activation function combined.

By combining activation functions and loss functions, this study shows that the combination of tanh-hinge and sigmoid-hinge has a better accuracy value than other combinations.

The tanh-hinge combination has an accuracy value of 97.21% in LSTM and GRU modeling, while the sigmoid-hinge combination has an accuracy value of 97.13% in LSTM modeling and 97.19% in GRU modeling. However, sigmoid-hinge outperforms tanh-hinge for the computation time factor, which reaches 2.57 ms in LSTM and GRU modeling.

5. Conclusion

Using the CIC-IDS 2018 dataset, which was grouped by destination ports and protocols and sorted by time in the LSTM and GRU methods, a new approach to IDS modeling was put through a number of steps and research methods. The results show that the new modeling is very accurate and works well.

The results of this study demonstrate that LSTM and GRU modeling using the CIC-IDS 2018 dataset, which has been clustered based on destination port and protocol and sequentially sorted by time, achieves an accuracy rate of 97.21% with a relatively short processing time of approximately 2-3 ms for each data entry into the model. So, we can say that this modeling is a good candidate for use and implementation as an IDS model in the future to find and classify cyberattacks in a network system.

The modeling stages carried out in this study are sorting the data set by time, clustering the data set by destination port and protocol, breaking the data set into small pieces of 10 rows of data using the Python library, and then processing the data set into LSTM and GRU modeling. Some suggestions for future research can refer to the sequence of stages and parameter values used in this study.

The research conducted is also still limited to using only 4 of the total 10 files contained in the CIC-IDS 2018 data set, where the 4 files have approximately 4 million rows of data. Although this study uses a sufficient number of rows of data, it can be expanded by using all of the CIC-IDS 2018 data set. The CIC-IDS 2018 data set used has a very large size and amount of data so that the research and IDS modeling process using this data set requires quite large computer resources. The limitations of the computer resources used in this study mean that only a maximum of 4 files can be used as a data set.

This research still has several shortcomings, and continuous research can be carried out as a form of improvement or development in the future. In this study, clustering of the data set is still limited to using only two attributes from the CIC-IDS 2018 data set: the destination port and protocol. This data set has 80 attributes in total. So that in the future further research can still be carried out related to clustering using other attributes.

The IDS modeling process in this study is also still limited to using 4 of the total 10 files in the CIC-IDS 2018 data set, so that further research can be carried out on IDS modeling using all files contained in the data set, although this requires quite large computer resources. The use of a training and testing data set ratio of 70:30 and several parameter values used in this study are also still random, where they can be developed and changed to other more optimal parameter values by utilizing several optimization techniques in further research.

In addition, the IDS modeling process using the CIC-IDS 2018 data set requires quite massive computer resources due to the very large size of the data set, so the research development niche related to computer resources for subsequent IDS modeling is still very open.

CRediT Authorship Contribution Statement

Ravi V. Rishika: Conceptualization, Methodology, Software, Investigation, Resources, Data Curation, Writing – Original Draft, Visualization, Funding Acquisition. **Baskoro A. Pratomo:** Resources, Writing – Review & Editing, Supervision. **Shintami C. Hidayati:** Validation, Formal analysis, Resources, Writing – Review & Editing, Supervision, Project Administration.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Declaration of Generative AI and AI-assisted Technologies in The Writing Process

The authors used generative AI to improve the writing clarity of this paper. They reviewed and edited the AI-assisted content and take full responsibility for the final publication.

References

[1] L. Chen and W. Liu, "The Effect of Internet Access on Body Weight: Evidence from China," *SSRN Electronic Journal*, 2022, doi: 10.2139/ssrn.4067120.

[2] N. Czernich, O. Falck, T. Kretschmer, and L. Woessmann, "Broadband Infrastructure and Economic Growth," *SSRN Electronic Journal*, 2009, doi: 10.2139/ssrn.1516232.

[3] N. S. Bhati and M. Khari, "Comparative Analysis of Classification Based Intrusion Detection Techniques," in *2021 5th International Conference on Information Systems and Computer Networks (ISCON)*, IEEE, Oct. 2021, pp. 1–6. doi: 10.1109/iscon52037.2021.9702411.

[4] H. Taylor, "Biggest cybersecurity threats in 2016," *URL: http://www.cnbc.com/2015/12/28/biggest-cybersecurity-threats-in-2016.html*, 2015.

[5] N. S. Bhati, M. Khari, V. Garc\ia-D\iaz, and E. Verdú, "A review on intrusion detection systems and techniques," *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, vol. 28, no. Supp2, pp. 65–91, 2020, doi: 10.1142/S0218488520400140.

[6] H. Debar, M. Dacier, and A. Wespi, "Towards a taxonomy of intrusion-detection systems," *Computer networks*, vol. 31, no. 8, pp. 805–822, 1999, doi: 10.1016/S1389-1286(98)00017-6.

[7] M. Ektefa, S. Memar, F. Sidi, and L. S. Affendey, "Intrusion detection using data mining techniques," in *2010 International Conference on Information Retrieval & Knowledge Management (CAMP)*, 2010, pp. 200–203. doi: 10.1109/INFRKM.2010.5466919.

[8] M. D. Holtz, B. M. David, and R. T. de Sousa Júnior, "Building scalable distributed intrusion detection systems based on the mapreduce framework," *Revista Telecommun*, vol. 13, no. 2, p. 22, 2011.

[9] S. Ghribi, A. M. Makhlouf, and F. Zarai, "C-dids: A cooperative and distributed intrusion detection system in cloud environment," in *2018 14th International Wireless Communications & Mobile Computing Conference (IWCMC)*, 2018, pp. 267–272. doi: 10.1109/IWCMC.2018.8450478.

[10] T. Verwoerd and R. Hunt, "Intrusion detection techniques and approaches," *Computer communications*, vol. 25, no. 15, pp. 1356–1365, 2002, doi: 10.1016/S0140-3664(02)00037-3.

[11] P. Widulinski and K. Wawryn, "A human immunity inspired intrusion detection system to search for infections in an operating system," in *2020 27th International Conference on Mixed Design of Integrated Circuits and System (MIXDES)*, 2020, pp. 187–191. doi: 10.23919/MIXDES49814.2020.9155771.

[12] Y. Gong, S. Mabu, C. Chen, Y. Wang, and K. Hirasawa, "Intrusion detection system combining misuse detection and anomaly detection using genetic network programming," in *2009 ICCAS-SICE*, 2009, pp. 3463–3467.

[13] A. Garg and P. Maheshwari, "A hybrid intrusion detection system: A review," in *2016 10th International Conference on Intelligent Systems and Control (ISCO)*, 2016, pp. 1–5. doi: 10.1109/ISCO.2016.7726909.

[14] B. A. Pratomo and R. M. Ijtihadie, "Sistem Deteksi Intrusi Menggunakan N-Gram Dan Cosine Similarity," *JUTI: Jurnal Ilmiah Teknologi Informasi*, pp. 108–116, 2016, doi: 10.12962/j24068535.v14i1.a516.

[15] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997, doi: 10.1162/neco.1997.9.8.1735.

[16] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," *arXiv preprint arXiv:1412.3555*, 2014.

[17] V. B. Kumar, V. M. Nookesh, B. S. Saketh, S. Syama, and J. Ramprabhakar, "Wind speed prediction using deep learning-LSTM and GRU," in *2021 2nd International Conference on Smart Electronics and Communication (ICOSEC)*, 2021, pp. 602–607. doi: 10.1109/ICOSEC51865.2021.9591886.

[18] S. Squartini, A. Hussain, and F. Piazza, "Preprocessing based solution for the vanishing gradient problem in recurrent neural networks," in *Proceedings of the 2003 International Symposium on Circuits and Systems, 2003. ISCAS'03.*, 2003, p. V–V. doi: 10.1109/ISCAS.2003.1206412.

[19] Y.-D. Syu *et al.*, "Ultra-short-term wind speed forecasting for wind power based on gated recurrent unit," in *2020 8th International electrical engineering congress (iEECON)*, 2020, pp. 1–4. doi: 10.1109/iEECON48109.2020.229518.

[20] N. Zulfa, N. Suciati, and S. C. Hidayati, "Improved Lip-reading Language Using Gated Recurrent Units," *JUTI: Jurnal Ilmiah Teknologi Informasi*, pp. 120–127, 2021, doi: 10.12962/j24068535.v19i2.a1080.

[21] F. W. L. W. P. A. Stolfo Salvatore and P. Chan, "KDD Cup 1999 Data." 1999.

[22] J. W. Haines, R. P. Lippmann, D. J. Fried, E. Tran, S. Boswell, and M. A. Zissman, "DARPA intrusion detection system evaluation: Design and procedures," 2001.

[23] Y. E. Tadesse and Y.-J. Choi, "Cse-cic-ids2018 and nslkdd image dataset," *IEEE Dataport*, 2023.

[24] L. Liu, G. Engelen, T. Lynar, D. Essam, and W. Joosen, "Error prevalence in nids datasets: A case study on cic-ids-2017 and cse-cic-ids-2018," in *2022 IEEE Conference on Communications and Network Security (CNS)*, 2022, pp. 254–262. doi: 10.1109/CNS56114.2022.9947235.

[25] J. Kim, S. Kim, H. Wimmer, and H. Liu, "A cryptocurrency prediction model using LSTM and GRU algorithms," in *2021 IEEE/ACIS 6th International Conference on Big Data, Cloud Computing, and Data Science (BCD)*, 2021, pp. 37–44. doi: 10.1109/BCD51206.2021.9581397.

[26] M. Pavithra, K. Saruladha, and K. Sathyabama, "GRU based deep learning model for prognosis prediction of disease progression," in *2019 3rd International Conference on Computing Methodologies and Communication (ICCMC)*, 2019, pp. 840–844. doi: 10.1109/ICCMC.2019.8819830.

[27] K. Zor and K. Buluş, "A benchmark of GRU and LSTM networks for short-term electric load forecasting," in *2021 International Conference on Innovation and Intelligence for Informatics, Computing, and Technologies (3ICT)*, 2021, pp. 598–602. doi: 10.1109/3ICT53449.2021.9581373.

[28] L.-E. Pommé, R. Bourqui, R. Giot, and D. Auber, "Relative confusion matrix: Efficient comparison of decision models," in *2022 26th International Conference Information Visualisation (IV)*, 2022, pp. 98–103. doi: 10.1109/IV56949.2022.00025.

[29] A. Jierula, S. Wang, T.-M. OH, and P. Wang, "Study on Accuracy Metrics for Evaluating the Predictions of Damage Locations in Deep Piles Using Artificial Neural Networks with Acoustic Emission Data," *Applied Sciences*, vol. 11, no. 5, p. 2314, Mar. 2021, doi: 10.3390/app11052314.

[30] P. Fränti and R. Mariescu-Istodor, "Soft precision and recall," *Pattern Recognition Letters*, vol. 167, pp. 115–121, Mar. 2023, doi: 10.1016/j.patrec.2023.02.005.

[31] S. A. Hicks *et al.*, "On evaluation metrics for medical applications of artificial intelligence," *Scientific Reports*, vol. 12, no. 1, Apr. 2022, doi: 10.1038/s41598-022-09954-8.