electronics

Article

Transient Stability Assessment of Power Systems Based on the Transformer and Neighborhood Rough Set

Tianyi Bei, Jianmei Xiao * and Xihuai Wang

Department of Electrical Automation, Logistics Engineering College, Shanghai Maritime University, Shanghai 201306, China; beitianyi@stu.shmtu.edu.cn (T.B.); wxh@shmtu.edu.cn (X.W.)
* Correspondence: jmxiao@shmtu.edu.cn

Abstract: Modern power systems are large in size and complex in features; the data collected by Phasor Measurement Units (PMUs) are often noisy and contaminated; and the machine learning models that have been applied to the transient stability assessment (TSA) of power systems are not sufficiently capable of capturing long-distance dependencies. All these issues make it difficult for data mining-based power system TSA methods to have sufficient accuracy, timeliness, and robustness. To solve this problem, this paper proposes a power system TSA model based on the transformer and neighborhood rough set. The model first uses the neighborhood rough set to deal with the redundant features of the power system trend data and then uses the transformer model to train the TSA model, in which various normalization methods such as Batch Normalization and Layer Normalization are introduced in the process to obtain better evaluation performance and speed up the convergence rate of the model. Finally, the model is evaluated by two evaluation indicators, $F_{1-\text{measure}}$ and accuracy, with values of 99.61% for accuracy and 0.9972 for $F_{1-\text{measure}}$, as soon as the tests on noise contamination and missing data test results on the IEEE39 system show that the NRS-Transformer model proposed in this paper is superior in terms of prediction accuracy, training speed, and robustness.

Keywords: neighborhood rough set; power system transient stability assessment; transformer; normalization methods

check for updates

Citation: Bei, T.; Xiao, J.; Wang, X. Transient Stability Assessment of Power Systems Based on the Transformer and Neighborhood Rough Set. *Electronics* **2024**, *13*, 270. https://doi.org/10.3390/electronics13020270

Academic Editor: Ahmed Abu-Siada

Received: 28 November 2023
Revised: 28 December 2023
Accepted: 5 January 2024
Published: 7 January 2024

Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

The construction of smart grids makes the power system structure more complex, and at the same time, it provides a wealth of data information for the power system transient stability assessment. Due to the implementation of energy savings and emission reduction, the proportion of electricity generated by renewable energy sources in the power grid continues to increase as soon as the system inertia continues to reduce. All these problems make the power system subject to greater perturbation; the transient power angle and the transient voltage are more likely to be unstable [1], and the risk of power outages increase [2], which makes it more difficult to maintain the stability of the power system.

In order to deal with the above problems, research on the transient stability assessment (TSA) of power systems is needed. The TSA can determine in time whether a system can maintain stability in the event of large perturbations. The time-domain simulation method, the transient energy function method, and the machine learning method are currently the most commonly used analysis methods for TSA. The time-domain simulation method [3,4] is the oldest and most mature of the analysis methods because the simulation is time-consuming and the calculation speed is slow [5]. This is inconsistent with the actual situation of the modern power system, which is often in a highly uncertain state and cannot satisfy the requirements of real-time assessments, so it is usually only used for the offline analysis of transient stability problems.

The transient function energy method [6] can obtain key information, such as the stability margin of the system, and significantly improve the calculation speed. However,

the transient function energy method is not well adapted to different models, and it has difficulties constructing energy functions for complex systems. The machine learning method does not establish a complete mathematical model, and it does not construct the energy function of the system; its main idea is to establish a nonlinear mapping relationship between the input features and the stable state of the system. The extensive development of Phasor Measurement Units (PMUs) means that a large amount of power system historical data and real-time data can be introduced into the work of the power system TSA [7,8], so the machine learning method has gradually become an important technical means of the power system TSA [9].

1.1. Related Work

According to AI methods, there are two main types of data mining-based TSAs: shallow learning and deep learning. The first of the former is artificial neural networks [10] (ANNs), and with the development of the method, decision trees [11], support vector machines [12] (SVMs), and other methods have also begun to be used for TSA. While shallow learning as a traditional machine learning method is able to complete the TSA, it has difficulty dealing with the huge amount of time-series data information brought by the modern smart grid and the features in it, and the model effect is often poor and the training is inefficient. Deep learning methods use more complex network structures to improve the ability to process data [13], and the current mainstream algorithms are convolutional neural networks [14] (CNNs), deep belief networks [15] (DBNs), long short-term memory [16] (LSTM) networks, and gated recurrent units [17] (GRUs).

In power system TSAs, the massive amounts of raw data provided by modern smart grids are usually time-series data, which include many time-series variables that reflect the dynamic changes in the power system over the time series, and thus need to select time-series variables based on the features of the time-series data. However, the traditional machine learning model lacks such operations, which therefore limits the performance of the evaluation model. Deep learning algorithms such as long short-term memory (LSTM) networks and gated recurrent units (GRUs) and their variants such as bi-directional long short-term memory (Bi-LSTM) networks and bi-directional gated recurrent units (Bi-GRUs) are time-dependent neural networks that can be learned in one or both directions along the time series and applied to the problem of the classification of sequential data to varying extents. They offer better performance and generalization capabilities compared to the traditional machine learning models. However, the existing networks used in power system TSA work have many limitations, such as the long short-term memory (LSTM) networks and gated recurrent units (GRUs), which are variants of recurrent neural networks (RNNs). Although they can mitigate the "gradient vanishing" and "gradient explosion" problems associated with RNN recurrent structures to a certain extent, this phenomenon cannot be completely eliminated, which limits their performance in TSA work with increasing data volumes. This is similar to the limitations of the pooling layer on convolutional neural networks (CNNs).

The transformer was initially proposed as a deep learning model for processing natural language tasks by Google in 2017 [18]. A self-attention mechanism (self-attention) is used to establish the dependencies between the elements in the input sequence, and encoding and decoding are performed through multi-layer stacking. Compared with traditional recurrent neural networks (RNNs) [19,20] and convolutional neural networks (CNNs), as well as long short-term memory (LSTM) networks and gated recurrent units (GRUs), the transformer can better capture long-distance dependencies and is more effective at processing long texts. Once proposed, the transformer quickly became an important model in the field of natural language processing, and it achieved successful applications in natural language processing [21], computer vision [22], audio processing [23], speech recognition [24], and machine translation [25]. There are many problems with the existing methods, such as the need for RNN to perform multiple loop calculations and the need for CNN to change the number of convolutional layers while capturing long-distance dependencies, which the transformer can avoid due to the matrix computation of the self-attention mechanism.

Most of the existing methods mainly focus on how to design the network structure and optimize the network parameters to achieve higher accuracy and faster speed [26] as modern power systems grow in size, become more intelligent, and provide increased data sample sets and more complex feature information because of the integration of renewable energy sources [27]. Therefore, attribute approximation of the input data is required to weaken the influence of excessive features before inputting into the assessment model, but traditional dimensionality reduction methods such as the principal component analysis (PCA) are not suitable to be applied in the approximation of the input data to the TSA model as the power system data are non-linear, time-varying, and dynamic.

1.2. Challenges and Limitations

A more appropriate approach is needed to deal with the lack of dimensionality of the reduction methods, which cannot be dealt with by the traditional dimensionality reduction methods. In the meantime, the data collected by PMU are often noisy, contaminated, large in size, and complex in features. The TSA model needs to overcome the speed and accuracy limitations caused by this problem, and new model structures are needed to avoid the limitations that CNNs and RNNs face.

1.3. Main Contributions of This Paper

To overcome the above problems, this paper proposes a transient stability assessment method for power systems based on the neighborhood rough set and transformer network. While significantly improving the accuracy of TSA, the method proposed in this paper will speed up training and become more reliable when dealing with noisy, contaminated data. The main contributions are summarized below:

This paper proposes a transient stability assessment method for power systems based on the transformer network and neighborhood rough set. The neighborhood rough set eliminates redundant attributes while keeping the original amount of information unchanged, so it can obtain the optimal feature subset of the original dataset while at the same time using transformer's unique self-attention mechanism to achieve a more comprehensive and selective weighting control of information with different levels of importance. This highlights the impact of the most important input features and avoids the problems of the existing methods, such as gradient vanishing and gradient explosion, and better incorporates the results of the neighborhood rough set processing to capture long-distance dependencies to achieve faster and better model training.

This method is also suitable for cases where the data provided by PMU are contaminated by noise and some data are missing, which is applicable to the reality of more complex data in modern smart grids.

Overall, unlike the existing methods that struggle with making full use of the information in the data when dealing with the massive amounts of data in modern smart grids and are susceptible to problems such as noise pollution, data defacement, and missing data, the NRS-Transformer model proposed in this paper effectively exploits the adaptive weight control of transformers for different features and the attribute approximation ability of the neighborhood rough set test results on the IEEE39 system to show that it provides higher accuracy and better applicability to the reality of more complex data in modern smart grids compared with the existing methods.

1.4. Structure of the Rest of the Paper

The remaining sections of this paper are organized as follows: The basic principles of the neighborhood rough set and transformer are discussed in Section 2. Section 3 introduces the structure of the transformer-based power system transient stability assessment method. Section 4 presents the implementation of a transformer-based transient stability assessment method for power systems along with its results and discussion, while the conclusion of the proposed work is concluded in Section 5.

2. Basic Principles

This section introduces the basic principles of the neighborhood rough set and transformer model, respectively.

2.1. Neighborhood Rough Set

The traditional rough set was proposed by Pawlak in 1982 [28], which is suitable for dealing with imprecise and fuzzy problems and can mine the hidden information in massive data, so it is widely applied to data processing in data mining and other fields.

Defining information systems $\langle U, A, V, f \rangle$, the $U = \{x_1, x_2, \cdots, x_l\}$ criterion for a field of view, the finite set of all samples, $A = \{a_1, a_2, \cdots, a_n\}$ as the combination of a set C of conditional attributes and a set D of decision attributes, both have an amount of M, V is the concatenated set of the domains of all values of A, f as the information function, and $f : U \times A = V$.

For any x_i in U, define its neighborhood as follows:

$$\delta(x_i) = \{x \mid x \in U, \Delta(x, x_i) \leq \delta\} \tag{1}$$

where δ is the radius of the neighborhood, $\delta \geq 0$, Δ is the distance function, and the smaller its value, the greater the similarity there is between the two samples.

Given N a neighborhood relation on U, we have $NAS = \langle U, N \rangle$ as a neighborhood approximation space. By $NAS = \langle U, N \rangle$ and δ, we can define the lower approximations ($\underline{N}X$) and upper approximations ($\overline{N}X$) of any subsets $X \subseteq U$ as follows:

$$\underline{N}X = \{x_i \mid \delta(x_i) \subseteq X, x_i \in U\} \tag{2}$$

$$\overline{N}X = \{x_i \mid \delta(x_i) \cap X \neq \varnothing, x_i \in U\} \tag{3}$$

- The boundary domain of X is as follows:

$$BN(X) = \overline{N}(X) - \underline{N}(X) \tag{4}$$

- The positive domain of X is as follows:

$$Pos(X) = \underline{N}(X) \tag{5}$$

- The negative domain of X is as follows:

$$Neg(X) = U - \overline{N}(X) \tag{6}$$

The boundary domain, positive domain, and negative domain correspond to the corresponding neighborhood particles in the domain U, thus approximating any subset of the neighborhood approximation space by the neighborhood particles.

For information systems $\langle U, A, V, f \rangle$:

B is the subset of conditional attributes, $B \subseteq C$, and the dependence of D on B can be defined as follows [29]:

$$\gamma_B(D) = \frac{|Pos_B(D)|}{|U|} \tag{7}$$

The formula shows that $\gamma_B(D) \in [0, 1]$; the larger its value, the better the ability of division for B.

For all $a \in C$, there are two cases:

1. $a \in C - B$: In this case, to perform attribute pooling to achieve the necessary subset of attributes, it is necessary to eliminate some of the attributes from all the attributes by constantly judging the effect of importance. The importance of attributes a relative to B is as follows:

$$Sig_1(a, B, D) = \gamma_B(D) - \gamma_{B-a}(D) \tag{8}$$

2. $a \in B$: In this case, in order to obtain the necessary subset of attributes for attribute pooling, it is necessary to add some of the attributes by setting the empty set and constantly judging the effect of the importance. The importance of the attribute a relative to B is as follows:

$$Sig_2(a, B, D) = \gamma_{B \cup a}(D) - \gamma_B(D) \tag{9}$$

Using these two methods for attribute approximation, the attribute subset is considered to have the maximum classification ability when the attribute importance is no longer changing and the neighborhood rough set approximation is complete.

2.2. Transformer Model

The transformer model consists of a stack of N sub-modules (transformer block). As shown in Figure 1, the sub-module contains two main parts, the multi-head attention layer and the feed-forward network layer, using layer normalization to prevent gradient degradation [30].

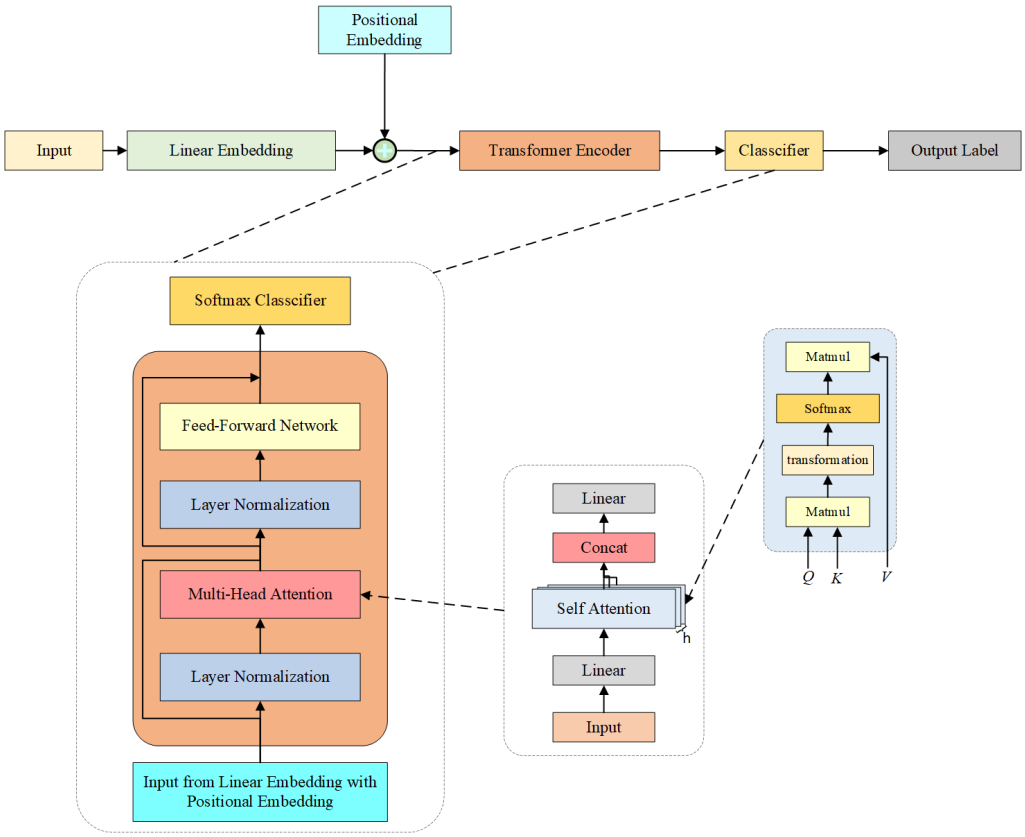


Figure 1. Transformer model.

Position embedding is used to describe the relative positional relationships between features, which are superposed with the embedding layer. The formula for its calculation is as follows:

$$P_E(p_{os}, 2i) = \sin\left(p_{os} / 10000^{2i/d_{model}}\right) \tag{10}$$

$$P_E(p_{os}, 2i+1) = \cos\left(p_{os} / 10000^{2i/d_{model}}\right) \tag{11}$$

where p_{os} is the sequence length, i is the feature vector dimension subscript, and d_{model} is the feature length.

The formula for multi-head attention is as follows:

$$A_{\text{ttention}}(Q, K, V) = F_{\text{softmax}}\left(\frac{QK^{\text{T}}}{\sqrt{d_k}}\right)V \tag{12}$$

$$h_{\text{ead}i} = A_{\text{ttention}}(Q_i, K_i, V_i) \tag{13}$$

$$M_{\text{ultiHead}}(Q, K, V) = C_{\text{oncat}}(h_{\text{ead}l}, h_{\text{ead}2}, \cdots, h_{\text{ead}i})W^{\text{O}} \tag{14}$$

In these formulas, $A_{\text{ttention}}(Q, K, V)$ is the output of self-attention, and $h_{\text{ead}i}$ is the $A_{\text{ttention}}(Q, K, V)$ of ith dimension.

Before inputting into the multi-head attention layer, the input matrix X undergoes three different linear transformations to obtain the query matrix Q, the key matrix K, and the value matrix V. Where W^Q, W^K, and W^V are the three learnable matrices, they will map X as a vector from d_{model} dimensions to d_k dimensional space:

$$Q = W^Q X \tag{15}$$

$$K = W^K X \tag{16}$$

$$V = W^V X \tag{17}$$

These matrices are then fed into self-attention layers, where the number is h.

In the self-attention layer, the input matrices are multiplied and transformed, and the attention weight values are obtained through the SoftMax function, creating h heads parallel to each other, which process attention operations independently, making up the multi-head attention mechanism.

Where W^O is the matrix of h heads of W^Q, W^K, and W^V, which are linearly transformed and then concatenated.

In these formulas, $A_{\text{ttention}}(Q, K, V)$ is the output of self-attention, $h_{\text{ead}i}$ is the $A_{\text{ttention}}(Q, K, V)$ of the ith dimension, and $M_{\text{ultiHead}}(Q, K, V)$, the output of the attention layer, can be obtained by W^O after the process of C_{oncat} connects all of the results of SoftMax ($h_{\text{ead}i}$) together.

The feed-forward neural network layer consists of a two-layer fully connected network with each layer mapping linearly to the input vector and an intermediate hidden layer activated using the Relu function. The feed-forward neural network formulation is as follows.

$$F_{\text{FN}}(x) = \max(0, xW_1 + b_1)W_2 + b_2 \tag{18}$$

x is the output vector after normalization of the attention layer, W is the weight vector, and b is a bias term.

3. The Structure of the Transformer-Based Power System TSA Method

This paper uses transformer as the TSA model and designs the transformer classifier to evaluate whether the power system is transiently stable. Firstly, we constructed the matrix dataset of samples and trained and tested sample sets were obtained using data preprocessing. Subsequently, parameter selection and model training were carried out, and the transformer models that met the requirements were screened with the help of performance evaluation metrics.

The assessment methodology is structured as shown in Figure 2.

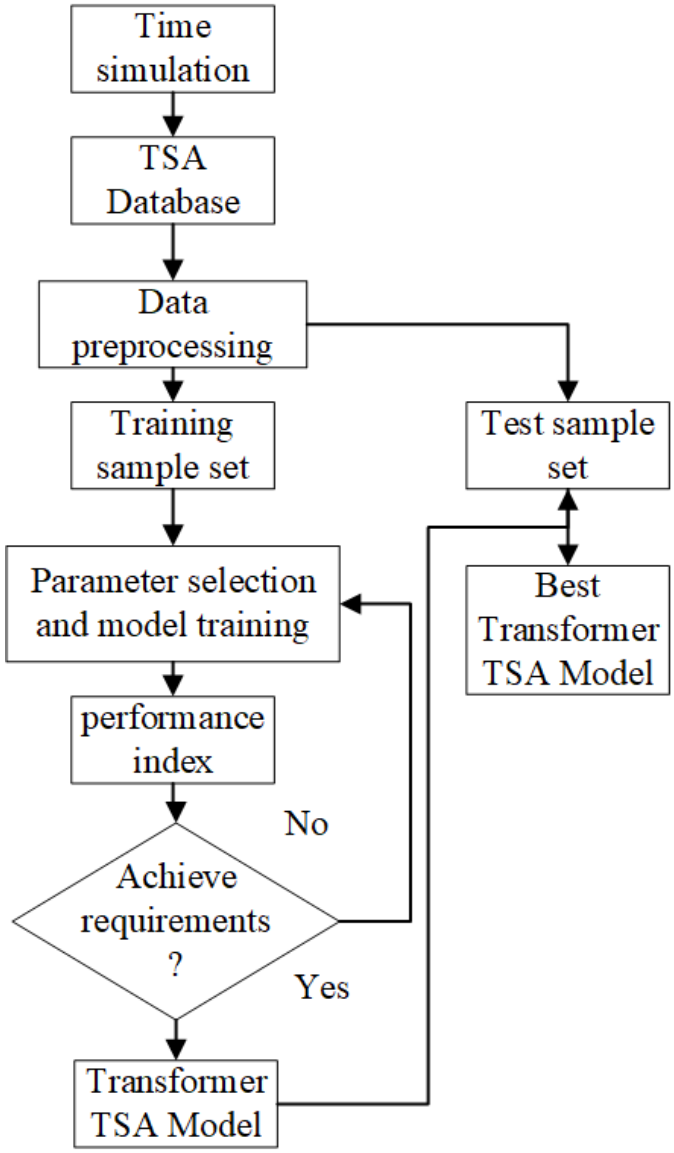


Figure 2. Assessment methodology structure.

3.1. Data Preprocessing

This paper uses the time-domain simulation to generate samples, the fault lines, fault times, and load levels of the power system, which are continuously adjusted to enrich the samples, and then the sample labels are added according to the basis of the transient stability judgment of the power system.

The sample set needs to construct the feature set and its labels. The process is as follows:

Step 1: Construct features. The selection of features will generally be carried out using the a priori knowledge of experts in power system transient analysis or automatic classifiers. The disadvantage of the expert method is its slow speed and the fact that there are errors in the calculations, while automatic classifiers have problems dealing with the huge amount of data.

Due to the data of the bus voltage and phase angle before and after the occurrence of faults having a large amplitude of change over time, it is beneficial to reduce the difficulty of the model in the training process. Meanwhile, it meets the requirements of high transient correlation and grid transient stability and facilitates the training of deep learning models. Therefore, this kind of feature set must be as wide as possible. The results are as follows:

$$\begin{bmatrix} u_{11} & \cdots & u_{1n} & \theta_{11} & \cdots & \theta_{1n} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ u_{m1} & \cdots & u_{mn} & \theta_{m1} & \cdots & \theta_{mn} \end{bmatrix} \tag{19}$$

In this formula, u is the bus voltage, θ is the bus phase angle, n is the number of features, and m is the time dimension.

Step 2: Obtaining labels. Determine whether the system is unstable or not using the power system transient stability judgment of stability basis, giving the sample set labels.

The transient stability of power systems refers to the power angle stability of a system subjected to large disturbances. The rotor formula of motion for a generator is as follows:

$$\begin{cases} \frac{\mathrm{d}\delta}{\mathrm{d}t} = (\omega - 1)\omega_0 \\ \frac{\mathrm{d}\omega}{\mathrm{d}t} = \frac{1}{T_J}(P_T - P_E) \end{cases} \tag{20}$$

In this formula, δ is the power angle, ω_0 is the synchronous electrical angular velocity, ω is the electrical angular velocity, T_J is the generator inertia time constant, and P_T and P_E are the generator mechanical power and electromagnetic power, respectively.

The size of δ can be used to determine whether the system is in an unstable state, and if the maximum power angle difference between generating units $|\Delta\delta_{\max}|$ is larger than 360°, meaning β is larger than 0, then the transient can be considered unstable [31]:

$$\beta = \frac{360^\circ - |\Delta\delta_{\max}|}{360^\circ + |\Delta\delta_{\max}|} \tag{21}$$

The stable data are labeled as 1; unstable data are labeled as 0.

3.2. The Construction of the Transformer Model

The transformer has been widely used in the field of natural language processing, and its unique structure of embedding feature layers and encoder–decoder framework endows it with an excellent ability to process information and cope with the problems of long-distance dependency and training speed.

The problems of the power system transient stability assessment are more likely classification problems, so it is necessary to modify the structure by removing parts of the decoder and forming the classifier with multiple encoders. The whole model can be divided into an input layer, a multi-head attention layer, a feed-forward neural network, an output layer, and a normalization layer after the multi-head attention layer and feed-forward neural network layers.

3.2.1. Input Layer

Feed features into the input layer of the transformer model after selecting them.

3.2.2. Multi-Head Attention Layer

After the features arrive from the input layer to the multi-head attention layer, all the features can be integrated, so the output result of the multi-head attention layer represents the weighted result of all the input features. Therefore, it has a good information processing ability. This is the source of the transformer model's advantage in dealing with long-distance dependencies.

3.2.3. Feed-Forward Neural Network Layer

The feed-forward neural network consists of a fully connected network, where the output of each encoder's multi-attention layer passes through a normalization layer once before it reaches the feed-forward neural network, and then it will be output again passing through a normalization layer and traveling to the next encoder until the output.

3.2.4. Normalization Layer

The normalization layer is interspersed throughout the encoder, serves to speed up convergence, and prevent gradient vanishing or gradient explosion.

3.2.5. Output Layer

The output layer is classified using the SoftMax function, as shown in the following formula:

$$SoftMax(z_i) = \frac{e^{z_i}}{\sum_{i=1}^{k} e^{z_i}} \tag{22}$$

In this formula, z_i is the output value of node i, and k is the number of classification categories.

3.3. Parameter Adjustment and Training of Transformer Model

3.3.1. Hyperparameter Adjustment

The transient stability assessment of power systems is a complex problem, and the hyperparameters of the transformer model, such as the number of hidden size, the number of attention heads, the embedding dimension, the number of encoder layers, the learning rate, and the batch size, can affect the process and results of model training. Most of the research in this area uses the grid search method, which determines the optimal value by finding all of the points in the search range. The advantage of this is that the optimal value within the search range can be found. However, the grid search method requires high computational resources, especially when there are many hyperparameters to be optimized, like in this paper.

The Bayesian optimization method, on the other hand, uses a Gaussian process, which takes into account the previous parameter information and constantly updates the prior. By sacrificing a certain amount of the ability to find the global optimum, it has advantages such as speed, but it does not easily fall into the local optimum.

Suppose $Z = x_1, x_2, \cdots, x_n$ is a set of hyperparameters, Bayesian optimization assumes that a functional relationship exists $f(x)$ between the hyperparameters and the loss function to be optimized, aiming to find $x \in Z$ using the optimization process that can make

$$x^* = \underset{x \in Z}{\operatorname{argmin}} f(x) \tag{23}$$

In this formula, x^* is the optimal set of hyperparameters.

3.3.2. Dropout

Dropout means that the model temporarily discards some neurons with probability so that they do not participate in the update of the network parameters. This can effectively prevent the model from overfitting.

3.3.3. Loss Function

The cross-entropy loss function is used as the loss function to improve the transformer model accuracy by reducing the value of the loss function, and at the same time, in order to improve the model training speed, it adds the Adam optimization algorithm to adaptively regulate the learning rate so that the learning rate can update the values in the direction of the gradient opposite to the loss function.

$$Loss = -y \log \hat{y} - (1 - y) \log(1 - \hat{y}) \tag{24}$$

In this formula, *Loss* represents the value of the loss function, y represents the stable sample, and $\hat{y}$ represents the probability that the sample is predicted to be stable.

3.4. Neighborhood Rough Set Approximations

Unlike principal component analysis methods such as PCA, rough sets can mine datasets for potential informational associations in non-linear and varying problems, enabling the elimination of redundant attributes. Neighborhood rough sets build on this and enhance the ability to deal with continuous problems; thus, the neighborhood rough set-based attribute approximation method can applied to the TSA.

Use two neighborhood rough set attribute approximation methods mentioned in Section 2.1 to work on the attribute approximation according to different situations.

Set the dataset sample to $U = \{x_1, x_2, \cdots, x_i\}$, the conditional attribute to $C = \{c_1, c_2, \cdots, c_j\}$, the input to the input layer, and the decision attributes D represent the results of the transient stability assessments (0 (unstable) and 1 (stable)). Perform attribute approximation until the attribute importance no longer changes.

4. Experimental Procedures, Results, and Analyses

To validate the method in this paper, Matlab/Simulink is used to perform time-domain simulations, and the transformer model was constructed based on the pytorch environment with the PC configured as AMD Ryzen 7 5800H with Radeon Graphics/16.0 GB RAM.

4.1. Data and Evaluation Metrics

4.1.1. IEEE 39 System

The arithmetic example used in this paper is the IEEE39 System; its topology is shown in Figure 3.

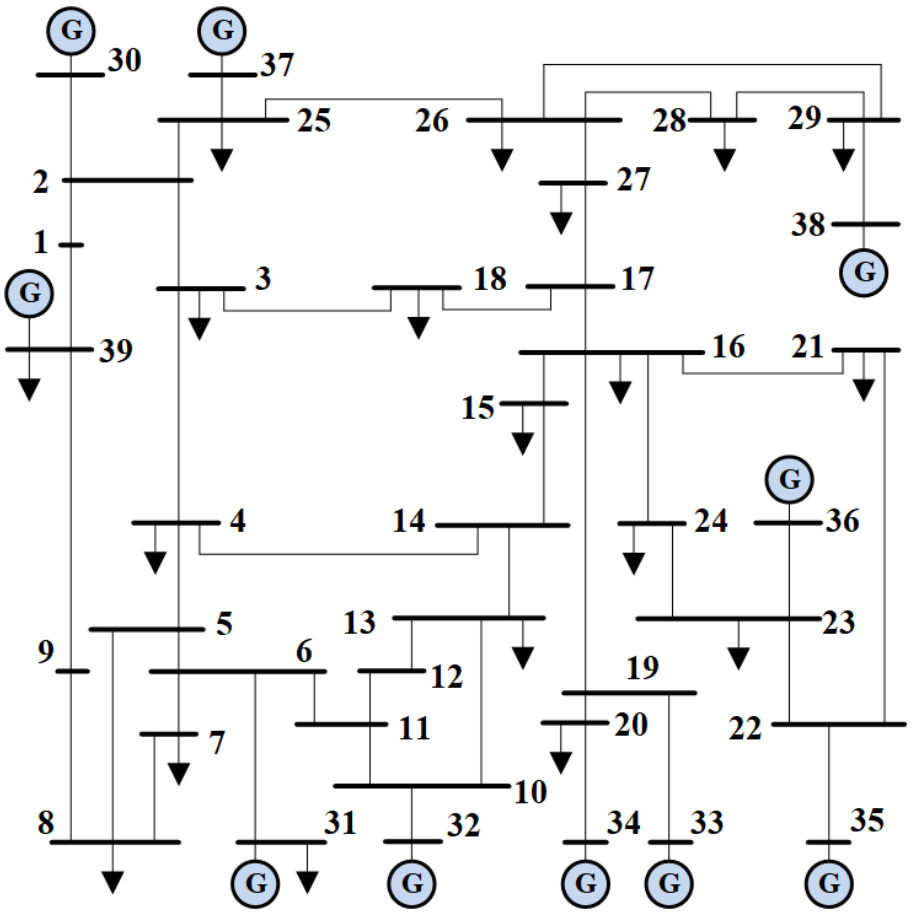


Figure 3. IEEE 39 System.

4.1.2. Construction of Datasets

Generators are modeled in second order, and the load level is incrementally increased from 70 percent to 140 percent in steps of 5 percent. Random short-circuit faults began at 0.1s. The fault time duration incrementally increased from 0.1 s to 0.5 s in steps of 0.1 s. The length of the simulation is 3 s, and a total of 16,035 samples were generated. We used the bus voltage and bus phase angle as a dataset, just as one was presented. Also, to verify whether the bus voltage and bus phase angle can be used as separate datasets for TSA, we established the bus voltage dataset as DATA_A, we established the bus phase angle dataset as DATA_B, and we established both of the two datasets as DATA_C. We also chose one pre-failure moment as t_1, the moment of failure as t_2, and the moment of fault removal as t_3, and used the data from these three moments to form the dataset together, as shown in Table 1:

Table 1. Dataset.

Dataset	Feature	Dimension	Training Set	Validation Set	Test Set
DATA_A	t_1, t_2, t_3, voltage of 39 bus	117	12,000	2000	2034
DATA_B	t_1, t_2, t_3, phase angle of 39 bus	117	12,000	2000	2034
DATA_C	t_1, t_2, t_3, voltage and phase angle of 39 bus	234	12,000	2000	2034

4.1.3. Evaluation Metrics

Finding the optimal transformer model relies on multiple performance evaluation metrics.

Traditional experiments mostly use accuracy (AC) as the evaluation index of machine learning transient stability assessment methods. However, the samples for transient stability assessments have more stable samples than unstable samples, and the stable and unstable samples are unbalanced; therefore, it is an unbalanced sample classification problem, so AC cannot be used as the only evaluation metric.

Therefore, this paper tries to introduce the F1-measure evaluation metrics, which use precision and recall as the main considerations and pay more attention to the measurement results of unbalanced samples. The mathematical expressions for AC, precision, recall, and F_a are as follows:

$$AC = \frac{TP + TN}{TP + TN + FP + FN} \tag{25}$$

$$precision = \frac{TN}{TN + FN} \tag{26}$$

$$recall = \frac{TN}{TN + FP} \tag{27}$$

$$F_a = \frac{(a^2 + 1) \times precision \times recall}{precision + recall} \tag{28}$$

In these formulas, TP is the number of correctly classified stable samples, FN is the number of incorrectly classified stable samples, FP is the number of correctly classified unstable samples, and TN is the number of incorrectly classified unstable samples. When $a = 1$, it means that the weights of *precision* and *recall* are the same, namely $F_{1-\text{measure}}$, which is the most common evaluation metric, and the larger it is, the better effect the model has.

4.2. Performance of the Transformer Model

Firstly, using the training set in Section 4.1.2, after completing the hyperparameter optimization for all models, seven models including the model proposed in this paper were trained, including transformer, Bi-LSTM-Attention, Bi-GRU-Attention, CNN, RNN, DNN, and SVM.

The loss function for Bi-LSTM-Attention and Bi-GRU-Attention was cross-entropy, the dropout was set to 0.5, the dropout of CNN was set to 0.01, and RBF was selected as the kernel function of SVM. The results of the model test are shown in Table 2.

Table 2. Results of the model test.

Model	DATA_A		DATA_B		DATA_C	
	AC/%	$F_{1-\text{measure}}$	**AC/%**	$F_{1-\text{measure}}$	**AC/%**	$F_{1-\text{measure}}$
Transformer	98.31	0.9879	98.76	0.9912	99.09	0.9935
Bi-LSTM-Attention	98.10	0.9865	98.55	0.9896	98.72	0.9908
Bi-GRU-Attention	97.27	0.9803	97.99	0.9856	98.27	0.9876
CNN	93.75	0.9558	96.15	0.9725	96.72	0.9765
RNN	95.92	0.9710	96.02	0.9715	97.43	0.9816
DNN	90.68	0.9352	89.17	0.9268	93.80	0.9559
SVM	86.72	0.8926	90.02	0.9199	89.93	0.9191

As can be seen from the results in Table 2, deep learning algorithms like transformer, Bi-LSTM-Attention, Bi-GRU-Attention, CNN, and RNN tend to have similar performance across the three datasets. Individually occurring datasets do not perform as well as co-occurring datasets in any of these algorithms. The transformer has an AC/% of 98.31% in DATA_A, which is 0.45% lower than in DATA_B, and a 0.0033 difference in $F_{1-\text{measure}}$. Bi-LSTM-Attention has the same trend, a 0.45% difference in AC/%, and 0.0031 in $F_{1-\text{measure}}$, which suggests that the performance of both networks is slightly weaker on the bus voltage dataset than on the bus phase angle dataset. Transformer performs better than Bi-LSTM-Attention overall; in other words, Bi-GRU-Attention has an AC/% of 97.27% in DATA_A; it is 0.72% lower than in DATA_B; and it has a 0.0052 difference in $F_{1-\text{measure}}$, meaning that the performance is more volatile and overall not as good as transformer and Bi-LSTM-Attention.

Because of the structural deficiencies that make it difficult for the model to deal with the problem of gradient vanishing, the feature extraction capabilities of CNN, RNN, and DNN are far inferior to transformer; therefore, the performance on all three datasets falls short of that of transformer. The difference between RNN's performance on the bus voltage dataset and the bus phase angle dataset is relatively small, but a large gap emerges between CNN's handling of the bus voltage dataset and the bus phase angle dataset, with a difference of 2.4% in AC/% between the two datasets, which suggests that CNN exhibits significantly poorer generalization capabilities when dealing with the features of the bus voltage dataset than the bus phase angle dataset. The reverse is true for DNN, as its performance on the bus phase angle dataset is poorer than that on the bus voltage dataset.

SVM as a shallow network does not perform as well as deep learning algorithms, and due to the limitations of the model itself, despite the application of the RBF kernel function, it still performs poorly when dealing with a large number of samples, and it performs rather poorly in the case of voltage–phase–angle co-operation than in the case of phase-angle alone.

Table 3 shows the transformer, Bi-LSTM-Attention, Bi-GRU-Attention, and CNN's training time on the DATA_C. The two tables together show that the transformer proposed in this paper has the highest accuracy and that the training time is less than that of Bi-LSTM-Attention and Bi-GRU-Attention on the DATA_C. The prediction accuracy is also significantly improved, despite the longer time taken in comparison to models such as CNN. With its combined accuracy and training speed, the transformer proposed in this paper is the most suitable for the transient stability assessment of the power system.

Table 3. The training time of some models on the DATA_C.

Model	Training Time/s
Transformer	245
Bi-LSTM-Attention	261
Bi-GRU-Attention	257
CNN	188

4.3. Visualization of the Training Process and Feature Extraction Capabilities

4.3.1. Visualization of the Training Process

In order to visually evaluate the performance of the network, this paper visualizes the training phase of each algorithm based on the performance of the above algorithms on the DATA_C, and the graph shows the AC% curves for the training phase. As can be concluded from Figure 4, the curve in the training phase of the transformer starts to gradually maintain the highest accuracy rate with fewer fluctuations after 100 rounds. Bi-LSTM-Attention and Bi-GRU-Attention are closer in their overall performance, with Bi-LSTM-Attention being slightly better than Bi-GRU-Attention, and both of them are at a higher level than CNN, RNN, and DNN. Due to the obvious lack of a generalization ability shown when dealing with the features of the voltage dataset, the performance of CNN has a slight gap in the AC% compared to that of RNN, especially at the beginning of the training period where the gap is very large, which is gradually narrowed down with the number of training sessions. With a slightly better stability, DNN is the one that rises the slowest and has the lowest accuracy rate among all the algorithms.

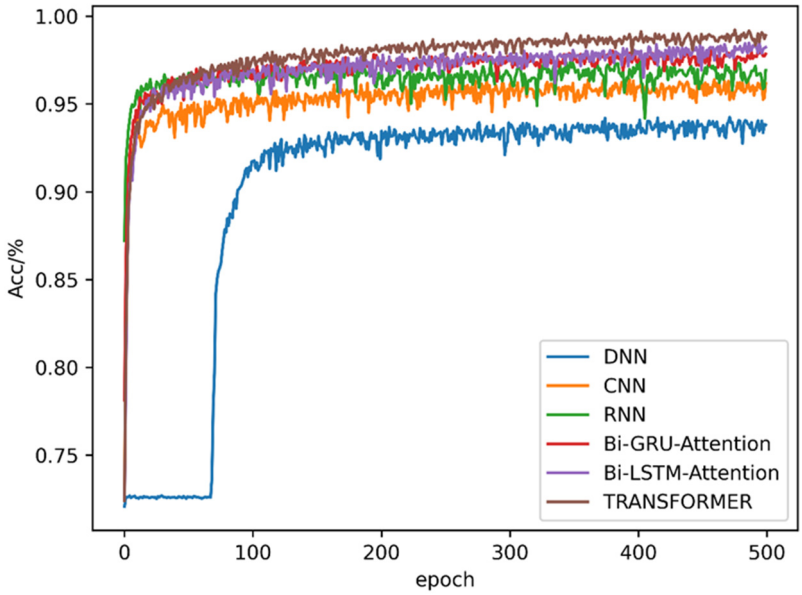


Figure 4. Visualization of the training process.

4.3.2. Visualization of Feature Extraction Capabilities

The transformer's visualization results before and after the extraction of the features can also achieve the same conclusions as Section 4.3.1, using t-SNE dimensionality reduction visualization to analyze the transformer's ability of extracting features, as shown in Figure 5.

In Figure 5a, it can be seen that the original data distribution is very messy. The data distribution before inputting the multi-head attention layer is shown in Figure 5b, and it can be seen that the data distribution is significantly improved and the division is beginning to be clear. Figure 5c reflects the fact that the division of unstable and stable samples is clearer after going through the multi-head attention layer, and the distribution is beginning to tighten. The final data distribution is shown in Figure 5d, which shows that the distribution of the original data has improved significantly after the transformer network. Therefore, the model is effective in dealing with the transient stability assessment of power systems.

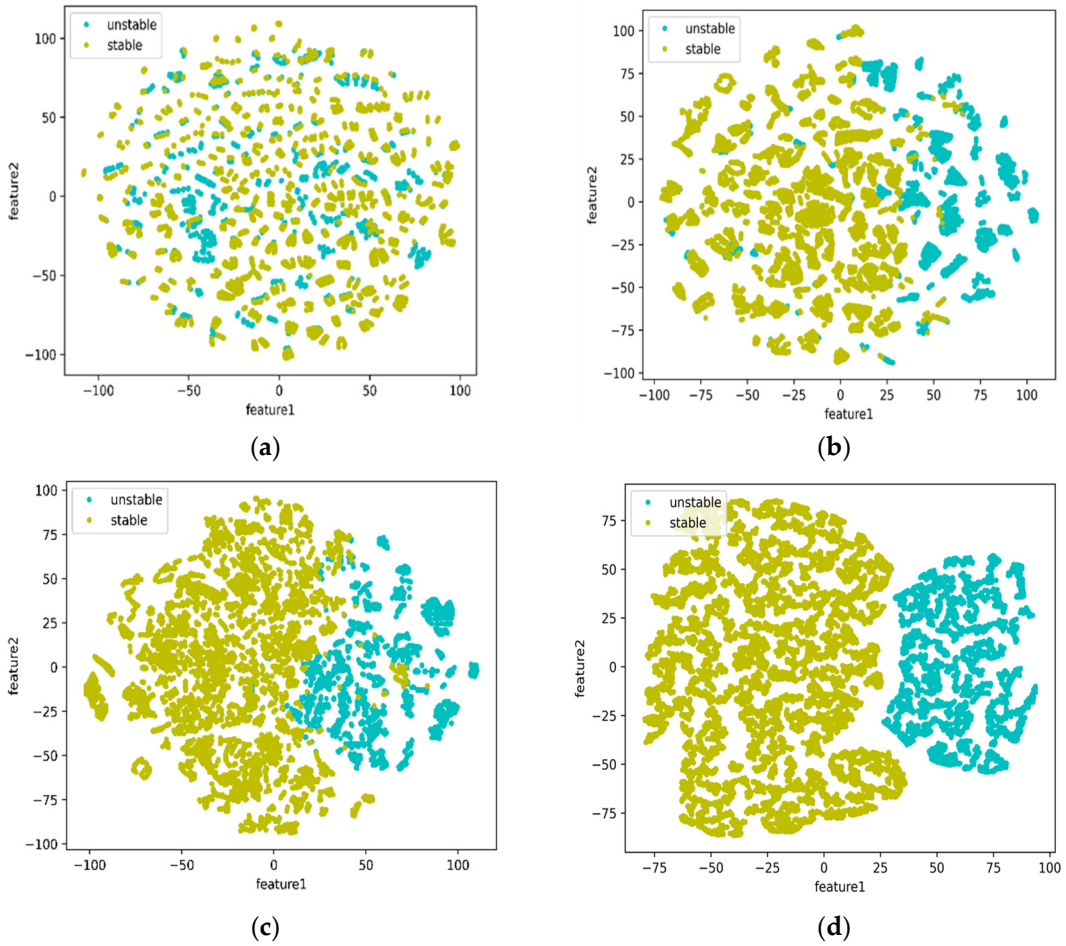


Figure 5. t-SNE dimensionality reduction visualization. (**a**) Original data distribution. (**b**) Before multi-head attention. (**c**) After multi-head attention. (**d**) Output data distribution.

4.4. Impact of Different Normalization Patterns on the Transformer Model

Deep learning models in the training process usually do not input the same distribution of data, so in the training process, there is a need to constantly learn the distribution of data into the model, which prolongs the time of convergence of the model. If the input dataset achieves a normalization operation before processing in the model, the time used for convergence can be reduced, and the impact of the size and dimensional differences on the model's training results can be avoided by enhancing the stability of the data distribution.

However, the above methods can only ensure that the data distribution in the input layer is stable, and for the subsequent network layer, the continuous updating of the parameters may lead to the reappearance of the problems mentioned above or even aggravate them with the process of parameter updating. Therefore, this paper introduces two normalization patterns to cope with the problems mentioned above.

4.4.1. Batch Normalization and Layer Normalization

Batch Normalization and Layer Normalization are currently the two dominant normalization modes. Batch Normalization will normalize each dimension of the model, starting with the features for normalization, first calculating the mean and variance in each batch, and then learning to derive, mathematically shift, and scale the input data, therefore accelerating the convergence without compromising the accuracy of the model. Layer Normalization, on the other hand, will comprehensively evaluate the inputs of all of the dimensions in a single layer of data for cross-sectional normalization.

The formula of Batch Normalization is as follows:

$$\mu_{\mathcal{B}} = \frac{1}{m}\sum_{i=1}^{m} x_i \tag{29}$$

$$\sigma_{\mathcal{B}}^2 = \frac{1}{m}\sum_{i=1}^{m}(x_i - \mu_{\mathcal{B}})^2 \tag{30}$$

$$\hat{x}_i = \frac{x_i - \mu_{\mathcal{B}}}{\sqrt{\sigma_{\mathcal{B}}^2 + \epsilon}} \tag{31}$$

$$\mathrm{BN}(x_i) = \gamma \hat{x}_i + \beta \tag{32}$$

In these formulas, $\mu_{\mathcal{B}}$ is the average value, $\sigma_{\mathcal{B}}^2$ is the variance (statistics), x_i is the ith data, ϵ is the stability constant, and γ and β are learned parameter values.

The formula of Layer Normalization is as follows:

$$\mu_{\mathcal{L}} = \frac{1}{M}\sum_{j=1}^{M} x_j \tag{33}$$

$$\sigma_M^2 = \frac{1}{M}\sum_{j=1}^{M}(x_j - \mu_M)^2 \tag{34}$$

$$LN(x) = \alpha \times \frac{x - \mu_M}{\sqrt{\sigma_M^2 + \varepsilon}} + \lambda \tag{35}$$

M is the number of neurons in a particular layer of the neural network, and x_j is the input to that layer, $\mu_{\mathcal{L}}$ is the average value, σ_M^2 is the variance (statistics), j means that is the jth dimension of the input, x_j and ε are stability constants, and α and λ are learned parameter values.

Layer Normalization operates in a similar way to Batch Normalization, but in order to achieve the normalization of a single training data to all neurons in a layer, the normalized objects of Layer Normalization become different dimensions of the same sample, as Batch Normalization normalizes the features of the same dimensions of a different sample.

4.4.2. Performance Comparison after Adding Different Normalization Patterns

The results of the two normalization patterns and the model without layer normalization are shown in Table 4.

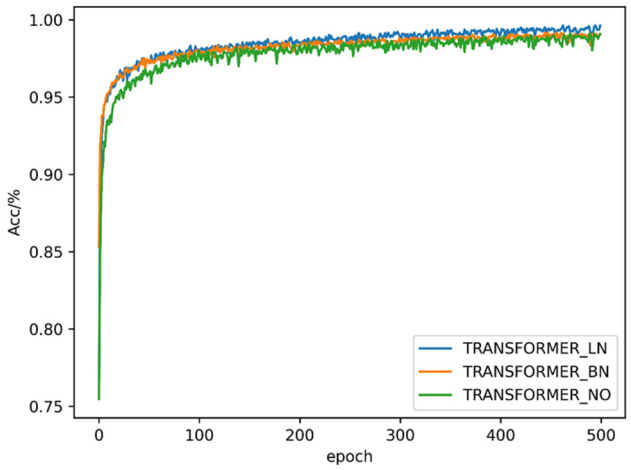
Table 4. Results of the two normalization patterns and the model without layer normalization.

Normalization	AC/%	$F_{1-\mathrm{measure}}$
Batch Normalization	99.14	0.9939
Layer Normalization	99.28	0.9949
No Normalization	99.09	0.9935

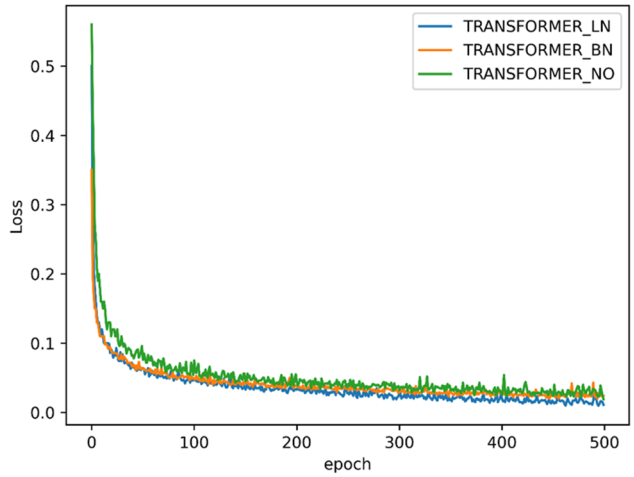
As the table shows, BN is 0.14% lower than LN in AC/%, and 0.1% lower than LN in the comprehensive evaluation $F_{1-\mathrm{measure}}$. The results of the training process visualization are shown in Figure 6.

In the early stage of training, after using Batch Normalization and Layer Normalization, the performance of the transformer model is improved compared to the previous one. As the number of training rounds increases, the gap between the model using Batch Normalization and the model without normalization gradually narrows, and the evaluation metrics also show that there is not much difference in the final effect between them, as the effect of the model using Layer Normalization not only keeps ahead of Batch Normalization

but is also smaller in the curve fluctuation, so the use of the LN layer is better than the use of the BN layer in the transformer model.



(**a**)



(**b**)

Figure 6. The results of the training process visualization. (**a**) The results of ACC. (**b**) The results of LOSS.

4.5. Impact of Neighborhood Rough Sets on Model Training

As can be seen from the evaluation phases reflected in Section 4.2, the transformer model proposed in this paper achieves satisfactory results in TSA; however, in practice, the size of the dataset constructed in this paper is far from good enough, and in practice, the dataset will have more redundant attributes, which will seriously restrict the training speed and accuracy of the model; therefore, the data is further processed to improve the training speed and training effect of the model through attribute approximation.

The neighborhood rough set not only retains the ability to deal with nonlinear problems, but it is also more suitable for dealing with continuous data. Through the neighborhood rough set's approximation work, the redundant attributes in the dataset are approximated, and the model can make better use of the effective attributes in the training process.

Still taking DATA_C as an example, we used the neighborhood rough set for attribute simplification, deleted the six attributes of the bus voltage and bus phase angle data at three moments of bus No. 1, brought the resulting dataset into the model training, and compared the training results with the model not simplified by neighborhood rough set as shown in Table 5. The methodological process is shown in Figure 7.

Table 5. Comparison of training results.

Model	AC/%	$F_{1-measure}$
Transformer	99.09	0.9935
NRS-Transformer	99.61	0.9972

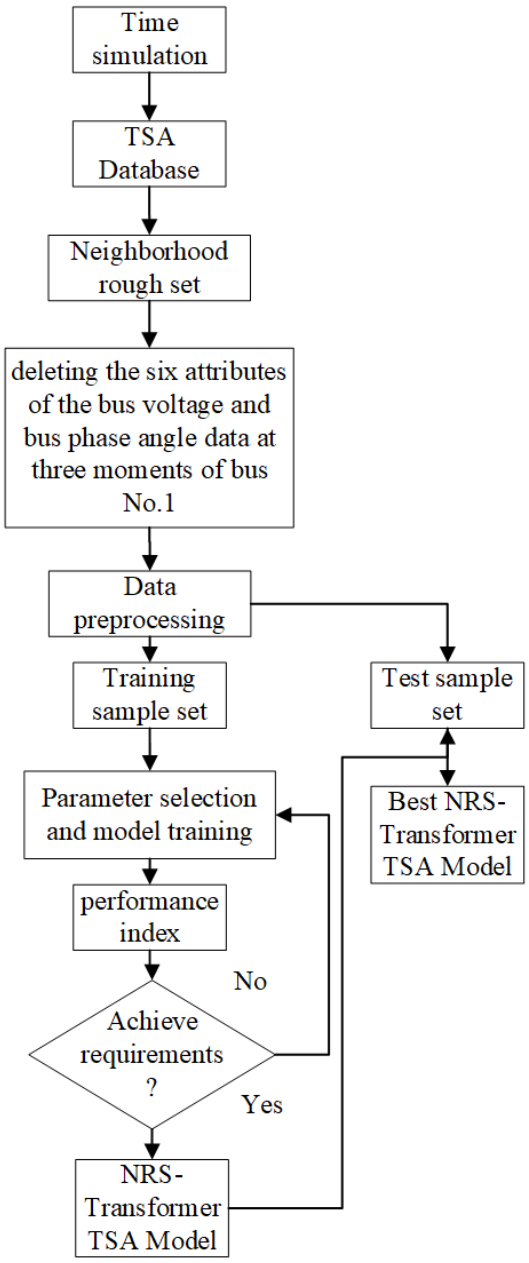


Figure 7. The methodological process of NRS-Transformer.

The training process of the model after introducing the neighborhood rough set and the training process of the model without introducing the neighborhood rough set is shown in Figure 8.

NRS-Transformer achieves an improvement of 0.52% and 0.37% in AC% and $F_{1-measure}$ compared to the transformer. This indicates that the neighborhood rough set attribute approximation effectively removes the negative impact of redundant attributes in the original data on model training, enhances the ability of the model to mine the main attributes, reduces erroneous evaluation results, and effectively improves the model's performance.

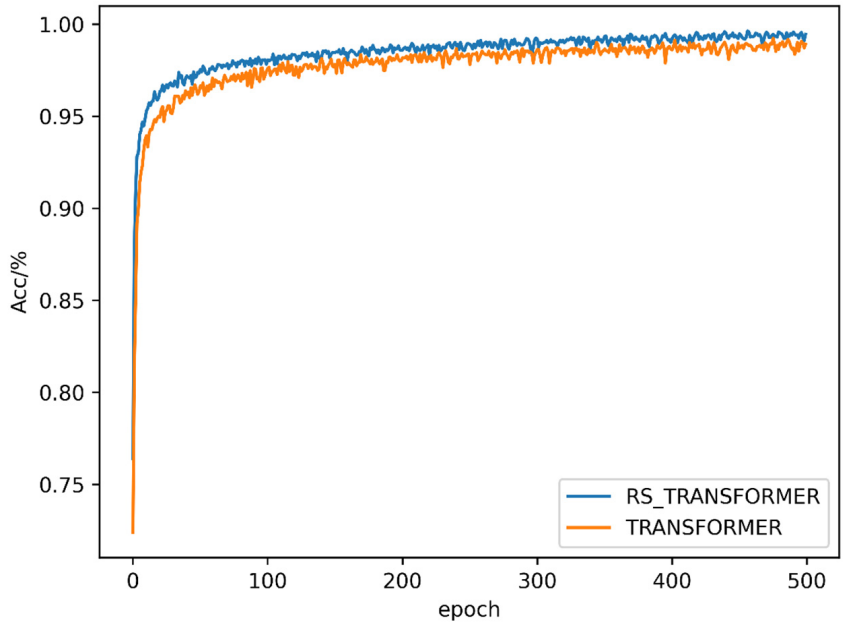


Figure 8. Comparison of training procession.

4.6. Model Performance of Different Models with Noise Contamination and Missing Data

The electrical data used for power system transient stability assessment work in modern power systems are collected by PMU, and ideally, the data are free of error, but noise pollution, data defacement, and missing data are inevitable in practice, so to better simulate the actual situation, the standard deviation of the noise conforming to the Gaussian distribution is added to the dataset as 0.01, 0.015, and 0.02. At the same time, a certain proportion of the data is replaced with random numbers conforming to a normal distribution within 0 to 1 as a feature masking to simulate the case of missing data features in the PMU dataset with proportions of 10%, 15%, and 20%.

Meanwhile, Bi-LSTM-Attention, Bi-GRU-Attention, CNN, RNN, DNN, and SVM are selected as comparison models to compare the performance of the model in the case of noise distribution and missing data. While following the idea of the NRS-Transformer in Section 4.5, the same NRS-Transformer model is trained to validate the effect of the model with neighborhood rough set approximation over the data; the results are shown in Table 6.

Table 6. Model performance of different models with noise contamination and missing data.

Model	AC/%			$F_{1-measure}$		
	Gaussian Noise Standard Deviation/Percentage of Missing Data					
	0.01/10%	0.015/15%	0.02/20%	0.01/10%	0.015/15%	0.02/20%
NRS-Transformer	88.93	87.17	84.90	0.9216	0.9107	0.8961
Transformer	88.48	86.46	84.18	0.9186	0.9068	0.8911
Bi-LSTM-Attention	87.17	84.64	83.15	0.9078	0.8980	0.8852
Bi-GRU-Attention	88.28	85.94	83.92	0.9172	0.9018	0.8932
CNN	79.53	76.41	70.57	0.8637	0.8505	0.8275
RNN	86.74	83.77	81.00	0.9065	0.8891	0.8755
DNN	82.86	78.18	77.24	0.8814	0.8583	0.8564
SVM	75.96	73.17	71.21	0.8194	0.8082	0.7893

Table 7 shows the changes of some models in training time.

Table 7. Changes in the training time of some models.

Model	Original Training Time/s	0.1	0.15	0.2
Transformer	245	314	287	280
Bi-LSTM-Attention	261	412	403	400
Bi-GRU-Attention	257	214	206	202
CNN	188	212	227	212

As shown in Table 6, with the increasing proportion of the noise standard deviation and anomalous features, the performance of each model slips. NRS-Transformer avoids a part of the influence of noise and anomalous features by removing redundant features, so it achieves a certain lead over transformer in all three datasets. Bi-LSTM-Attention does not perform as well as transformer in all three datasets, and Bi-GRU-Attention, due to the relatively simpler structure of the GRU compared to the LSTM, is less influenced by the anomalous data and even achieves a performance close to transformer in the case of less anomalous data and less noise. But with the increase in abnormal data and the increase in noise, its performance also decreases rapidly. In terms of CNN, RNN, DNN, and SVM performance, RNN's performance is the best, followed by DNN. CNN's performance is better in the first two datasets, but when abnormal data and noise are increased, its performance is even lower than SVM, which is less desirable.

As shown in Table 7, among the four models, the training speed of transformer, Bi-LSTM-Attention, and CNN slowed down to varying degrees, with Bi-LSTM-Attention being the most serious. But as more and more anomalous features were added and less information could be extracted from the models, the training speed started to increase instead, while the training speed of Bi-GRU-Attention was not affected but continued to increase.

Comprehensively, the method proposed in this paper can deal with the dataset in the presence of noise pollution and feature anomalies, and it outperforms the other comparative models in the presence of different noise pollution and anomalous features with better robustness.

5. Conclusions

This paper addresses the problem of the transient stability assessment of power systems. A TSA method was proposed based on the transformer and neighborhood rough set, and the following conclusions were drawn based on a series of simulation studies on the IEEE39 system:

- The transformer-based model constructed in this paper, with its multi-head attention layer, exhibits a better ability to mine information from the data than the other networks; it can make better use of the information in the input dataset; and it has a higher performance than the other comparison networks in this paper.
- The transformer-based model constructed in this paper can avoid problems of existing methods such as gradient vanishing and gradient explosion that RNN and its variants cannot avoid, so it can significantly improve the accuracy of TSA as soon as it speeds up the training of the model.
- In this paper, the effects of different normalization patterns on the training results and process of neural networks are verified by introducing two normalization patterns, and the results show that Layer Normalization is more suitable for the model proposed in this paper.
- In this paper, the original dataset is simplified with the help of the neighborhood rough set, and the dataset imported into the transformer model with redundant attributes removed improves the training results as well as optimizes the model's performance during the training process.
- In this paper, the anti-interference ability of the proposed transformer model is verified by a noise test; the model outperforms other comparative models, and the results after using the neighborhood roughness set method also verify the role of neighborhood

roughness set in optimizing the model training and enhancing the anti-interference ability of the model.

Author Contributions: Methodology, T.B.; Software, T.B.; Validation, T.B.; Data curation, T.B.; Writing—original draft, T.B.; Writing—review & editing, J.X. and X.W.; Visualization, T.B.; Supervision, J.X. and X.W. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The data are not publicly available due to the sensitive and critical nature of the data.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Dai, Y.T.; Preece, R.; Panteli, M. Risk assessment of cascading failures in power systems with increasing wind penetration. *Electr. Power Syst. Res.* **2022**, *211*, 108392. [CrossRef]
2. Wei, L.; Yi, C.; Yun, J. Energy drive and management of smart grids with high penetration of renewable sources of wind unit and solar panel. *Int. J. Electr. Power Energy Syst.* **2021**, *129*, 106846. [CrossRef]
3. Stott, B. Power system dynamic response calculations. *Proc. IEEE* **1979**, *67*, 219–241. [CrossRef]
4. Deng, X.D.; Jiang, Z.H.; Sundaresh, L.; Yao, W.X.; Yu, W.P.; Wang, W.K.; Liu, Y.L. A time-domain electromechanical co-simulation framework for power system transient analysis with retainment of user defined models. *Int. J. Electr. Power Energy Syst.* **2021**, *125*, 106506. [CrossRef]
5. Pavella, M.; Ernst, D.; Ruiz-Vega, D. *Transient Stability of Power Systems: A Unified Approach to Assessment and Control*; Springer Science & Business Media: Berlin/Heidelberg, Germany, 2000; Volume 581.
6. Pai, A. *Energy Function Analysis for Power System Stability*; Springer Science & Business Media: Berlin/Heidelberg, Germany, 1989.
7. Ge, H.C.; Guo, Q.L.; Sun, H.B.; Zhao, W.L. A model and data hybrid-driven short-term voltage stability real-time monitoring method. *Int. J. Electr. Power Energy Syst.* **2020**, *114*, 105373. [CrossRef]
8. Samantaray, S.R.; Kamwa, I.; Joos, G. Phasor measurement unit based wide-area monitoring and information sharing between micro-grids. *IET Gener. Transm. Distrib.* **2017**, *11*, 1293–1302. [CrossRef]
9. Kang, Z.; Zhang, Q.; Chen, M.; Gan, D. Research on Network Voltage Analysis Algorithm Suitable for Power System Transient Stability Analysis. *Power Syst. Prot. Control* **2021**, *49*, 32–38.
10. Siddiqui, S.A.; Verma, K.; Niazi, K.R.; Fozdar, M. Real-Time Monitoring of Post-Fault Scenario for Determining Generator Coherency and Transient Stability Through ANN. *IEEE Trans. Ind. Appl.* **2018**, *54*, 685–692. [CrossRef]
11. Vanfretti, L.; Arava, V.S.N. Decision tree-based classification of multiple operating conditions for power system voltage stability assessment. *Int. J. Electr. Power Energy Syst.* **2020**, *123*, 106251. [CrossRef]
12. Mosavi, A.B.; Amiri, A.; Hosseini, H. A learning framework for size and type independent transient stability prediction of power system using twin convolutional support vector machine. *IEEE Access* **2018**, *6*, 69937–69947. [CrossRef]
13. Goodfellow, I.; Bengio, Y.; Courville, A. *Deep Learning*; MIT Press: Cambridge, MA, USA, 2016.
14. Zhu, L.P.; Hill, D.J.; Lu, C. Hierarchical Deep Learning Machine for Power System Online Transient Stability Prediction. *IEEE Trans. Power Syst.* **2020**, *35*, 2399–2411. [CrossRef]
15. Wu, S.; Zheng, L.; Hu, W.; Yu, R.; Liu, B.S. Improved Deep Belief Network and Model Interpretation Method for Power System Transient Stability Assessment. *J. Mod. Power Syst. Clean Energy* **2020**, *8*, 27–37. [CrossRef]
16. Li, B.Q.; Wu, J.Y.; Hao, L.L.; Shao, M.Y.; Zhang, R.Y.; Zhao, W. Anti-Jitter and Refined Power System Transient Stability Assessment Based on Long-Short Term Memory Network. *IEEE Access* **2020**, *8*, 35231–35244. [CrossRef]
17. Chen, Q.F.; Wang, H.Y. Time-adaptive transient stability assessment based on gated recurrent unit. *Int. J. Electr. Power Energy Syst.* **2021**, *133*, 107156. [CrossRef]
18. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention Is All You Need. Advances in Neural Information Processing Systems. 2017, p. 30. Available online: https://pdf-reader-dkraft.s3.us-east-2.amazonaws.com/1706.03762.pdf (accessed on 4 January 2024).
19. Yu, J.J.Q.; Hill, D.J.; Lam, A.Y.S.; Gu, J.T.; Li, V.O.K. Intelligent Time-Adaptive Transient Stability Assessment System. *IEEE Trans. Power Syst.* **2018**, *33*, 1049–1058. [CrossRef]
20. Nguyen, T.H.; Shirai, K. Phrasernn: Phrase recursive neural network for aspect-based sentiment analysis. In Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, Lisbon, Portugal, 17–21 September 2015; pp. 2509–2514.
21. Zhao, Q.; Cai, X.; Chen, C.; Lv, L.; Chen, M. Commented content classification with deep neural network based on attention mechanism. In Proceedings of the 2017 IEEE 2nd Advanced Information Technology, Electronic and Automation Control Conference (IAEAC), Chongqing, China, 25–26 March 2017; pp. 2016–2019.
22. Ranftl, R.; Bochkovskiy, A.; Koltun, V. Vision transformers for dense prediction. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, BC, Canada, 11–17 October 2021; pp. 12179–12188.

23. Yang, B.; Tu, Z.; Wong, D.F.; Meng, F.; Chao, L.S.; Zhang, T. Modeling localness for self-attention networks. *arXiv* **2018**, arXiv:1810.10182.
24. Zhang, Q.; Lu, H.; Sak, H.; Tripathi, A.; McDermott, E.; Koo, S.; Kumar, S. Transformer transducer: A streamable speech recognition model with transformer encoders and rnn-t loss. In Proceedings of the ICASSP 2020—2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Barcelona, Spain, 4–8 May 2020; pp. 7829–7833.
25. Liu, X.; Duh, K.; Liu, L.; Gao, J. Very deep transformers for neural machine translation. *arXiv* **2020**, arXiv:2008.07772.
26. Li, B.Y.; Xiao, J.M.; Wang, X.H. Feature Reduction for Power System Transient Stability Assessment Based on Neighborhood Rough Set and Discernibility Matrix. *Energies* **2018**, *11*, 185. [CrossRef]
27. Tahir, M.F.; Chen, H.Y.; Han, G.Z. A comprehensive review of 4E analysis of thermal power plants, intermittent renewable energy and integrated energy systems. *Energy Rep.* **2021**, *7*, 3517–3534. [CrossRef]
28. Pawlak, Z. Rough sets. *Int. J. Comput. Inf. Sci.* **1982**, *11*, 341–356. [CrossRef]
29. Qian, Y.; Liang, J.; Pedrycz, W.; Dang, C. An efficient accelerator for attribute reduction from incomplete data in rough set framework. *Pattern Recognit.* **2011**, *44*, 1658–1670. [CrossRef]
30. Fang, J.S.; Liu, C.R.; Zheng, L.; Su, C.B. A data-driven method for online transient stability monitoring with vision-transformer networks. *Int. J. Electr. Power Energy Syst.* **2023**, *149*, 109020. [CrossRef]
31. Li, X.; Liu, C.K.; Guo, P.F.; Liu, S.C.; Ning, J. Deep learning-based transient stability assessment framework for large-scale modern power system. *Int. J. Electr. Power Energy Syst.* **2022**, *139*, 108010. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.