Received 7 March 2024; revised 28 May 2024, 8 June 2024, and 25 June 2024; accepted 8 July 2024. Date of publication 10 July 2024; date of current version 19 July 2024. The review of this article was arranged by Associate Editor Kyriaki-Nefeli Malamaki.

Digital Object Identifier 10.1109/OJIA.2024.3426281

FLACON: A Deep Federated Transfer Learning-Enabled Transient Stability Assessment During Symmetrical and Asymmetrical Grid Faults

MOHAMED MASSAOUDI [1,2] (Member, IEEE), HAITHAM ABU-RUB [2,3] (Fellow, IEEE), AND ALI GHRAYEB [2,3] (Fellow, IEEE)

[1]Department of Electrical Engineering, Texas A&M University, College Station, TX 77840 USA
[2]Department of Electrical Engineering, Texas A&M University at Qatar, Doha 23874, Qatar
[3]College of Science and Engineering, Hamad Bin Khalifa University, Doha 34110, Qatar

CORRESPONDING AUTHOR: MOHAMED MASSAOUDI (e-mail: mohamed.massaoudi@qatar.tamu.edu)

This work was supported by the Qatar National Research Fund (QNRF is a member of Qatar Foundation) under Grant NPRP12C-33905-SP-213 and Grant NPRP12C-33905-SP-220 and in part by the Qatar National Library through open access.

ABSTRACT Transient stability assessment (TSA) is critical to the reliable operation of a power system against severe fault conditions. In practice, TSA based on deep learning is preferable for its high accuracy but often overlooks challenges in maintaining data privacy while coping with network topology changes. This article proposes an innovative focal loss-based multihead attention convolutional network (FLACON) for accurate post-disturbance TSA under both symmetrical and asymmetrical smart grid faults. The proposed approach effectively incorporates cross-domain deep federated transfer learning (FTL) to leverage local operating data for TSA in a decentralized fashion. By introducing convolutional layers alongside multi-head attention mechanisms, the FLACON framework significantly improves learning efficiency across geographically distributed datasets. To address the challenge of class imbalance, the model integrates a balance factor-enhanced focal loss function. The FTL architecture enables decentralized model training across various clients, thus preserving data privacy and reducing the burden of communication overhead. To avoid the constant adjustment of hyperparameters, the FLACON employs an inductive transfer learning approach for hyperparameter tuning of the pre-trained model, markedly decreasing training time. Extensive experiments on datasets from the IEEE 39-bus system and the IEEE 68-bus system demonstrate FLACON's exceptional accuracy of 98.98% compared to some competitive alternatives.

INDEX TERMS Federated transfer learning (FTL), multihead attention (MHA) mechanisms, power grid faults, symmetrical and asymmetrical faults, transient stability assessment (TSA).

NOMENCLATURE

Vector/Matrix

$\mathbf{W}_k^*$	Trainable weight matrices for key.
$\mathbf{W}_q^*$	Trainable weight matrices for query.
$\mathbf{W}_v^*$	Trainable weight matrices for value.
$\mathbf{X}_{\text{in}}$	Input embeddings.
Θ	Global model parameters.
C_k	Client nodes.
K	Key matrix in attention mechanism.
Qu	Query matrix in attention mechanism.
Va	Value matrix in attention mechanism.

Variables

$\delta_{\max}$	Maximal phase angle discrepancy.
θ	Bus angle.
F	Frequency.
P	Active power.
Q	Reactive power.

© 2024 The Authors. This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

t	Time variable.
V	Bus voltage.
x	State variables of the system in $\mathbb{R}^n$.

Abbreviations

3 PB	Three phase balanced fault.
Acc	Accuracy.
ANN	Artificial neural network.
CNN	Convolutional neural network.
CPU	Central processing unit.
DL	Deep learning.
DLL	Double line to ground fault.
DTL	Deep transfer learning.
F1	F1-score.
FFN	Feedforward neural network.
FL	Federated learning.
FN	False negative.
FP	False positive.
FTL	Federated transfer learning.
GPU	Graphics processing unit.
HFL	Horizontal federated learning.
IL	Inference latency.
LL	Line to line fault.
LSTM	Long short term memory.
MCS	Monte Carlo simulations.
MHA	Multihead attention.
PDF	Probability distribution function.
PMUs	Phasor measurement units.
Prec	Precision.
R	Recall.
RAM	Random access memory.
ReLU	Rectified linear unit.
SLG	Single line to ground fault.
t-SNE	t-Distributed stochastic neighbor embedding.
TL	Transfer learning.
TN	True negative.
TP	True positive.
TSA	Transient stability assessment.
TSI	Transient stability index.

I. INTRODUCTION

The rapid shifts in renewable portfolio standards lead to unprecedented changes in the power infrastructure, causing fluctuations in energy generation [1]. These fluctuations, along with the integration of power electronic devices, challenge system stability and grid control. The grid instability can cause system collapse within seconds, often due to changes in generator output power and power angles. As it is widely stipulated, sufficiently accurate transient stability assessment (TSA) is pivotal to avoid loss-of-synchronism and ultimately blackouts [2]. When subjected to severe disturbances, such as grid faults, real-time TSA evaluates a power system's ability to recover after a severe malfunction. TSA models promote the system's ability to withstand possible large disturbances [3]. Nonetheless, the stability of power systems is associated with a substantial array of differential-algebraic equations that present challenges for direct integration into optimization models for system operations [4]. When uncertainties are factored in, the resulting optimization model can become exceedingly complex and high-dimensional. Traditional model-based analytical approaches fall short in these scenarios, particularly as the integration of a significant quantity of renewable energy resources into the intelligent cyber-physical grid progresses.

Data-driven methodologies, especially those employing deep learning (DL) have made significant strides in predicting stability status under various grid conditions [5]. However, the majority of the existing artificial neural network (ANN) architectures employ a database stored at a centralized power system control center leading to a single point of control and decision-making [6]. This centralized approach can be impractical due to the computational burden and vulnerability to privacy exposure associated with transmitting sensitive information from distributed phasor measurement units (PMUs) to a central processing facility [7]. While existing data-driven models are capable of detecting instabilities, they frequently fail to account for the distributed nature of power grids and the localized characteristics of data arising from geographically dispersed regions [8]. Centralized models also suffer from substantial delays in real-time prediction due to the overheads associated with large-scale data transmission, storage, and analysis [9]. Federated learning (FL) has been proposed to promote a decentralized approach to model training [10]. However, it often presupposes that data is independently and identically distributed, which does not hold for PMU data. This assumption overlooks the unique data distributions inherent to specific regions, influenced by their operational grid characteristics. Consequently, regional changes in data distribution can lead to a degradation of model performance when applied globally, necessitating frequent retraining of models for local accuracy—an often time-prohibitive task for rapid, short-term instability predictions [11].

The transient stability of power systems has been thoroughly studied in the literature. For instance, Hijazi et al. [12] proposed a one-to-one transfer learning (TL) for TSA amidst changing topologies. The integration of 1-D convolutional neural network (CNN) and long short term memory (LSTM) to form a ConvLSTM model acknowledges the spatial-temporal intricacies of PMU measurements but omits asymmetrical faults. Moreover, the occurrence of rare yet critical events in real-world scenarios is much underrepresented than in normal operating conditions leading to imbalanced datasets, in which the proposed model did not perform adequately. To address this gap, a recent study in [13] offered a novel corrective for this imbalance, enhancing LSTM efficacy by using multiple layers. Nevertheless, concerns linger over computational efficiency and potential overfitting. Moreover, the asymmetrical faults that are most frequently occurring in real power systems are neglected. Li et al. [14] addressed sample collection hurdles via an instance-transfer extreme learning machine. Unfortunately, the leap from simulation results to real-world application remains vast, compounded by latency issues inherent in centralized data processing critical to TSA's timeliness. Across these studies, the pursuit of robust, adaptable, and swift TSA methodologies is palpable, but the convergence of theory with the on-the-ground realities of power systems demands further development [15]. Despite obvious advances, the design of neural network structures still lack a solid theoretical foundations, as transient stability landscape is constantly evolving, requiring neural network models to be continually updated and adapted to reflect these changes.

To enhance the model's adaptability to significant shifts in topologies or operational conditions, this study introduces an active TL approach. To the best of the author's knowledge, this article offers the first attempt to solve the TSA task in a coupled approach where TL-based TSA and horizontal federated learning (HFL) are simultaneously computed. In summary, the main contributions of this article are listed as follows.

1) A novel TSA method based on a multifaceted architecture that synergizes the strengths of CNNs and multihead attention (MHA) mechanisms is introduced. The proposed method uniquely relies on the MHA mechanism to boost the model generalization in solving the TSA problem under both symmetrical and asymmetrical faults in power systems.
2) A customized focal loss based on an α-balanced variant is designed to deal with the category-imbalanced problem in transient stability classification. This approach adjusts the focus dynamically, improving model sensitivity and specificity for minority classes which are critical in predicting instability within power systems.
3) An effective federated deep neural network is proposed to conserve the privacy-preserving characteristics of TSA in a decentralized manner. By addressing the TSA challenge of privacy exposure, this work is the first attempt, to our knowledge, to employ HFL with deep transfer learning (DTL) to model focal loss-based multihead attention convolutional network (FLACON), presenting an architecture based on HFL (HFL-FLACON). The growing complexity and interconnectivity of power grids render them susceptible to failure at single critical junctures and to potential security breaches. The proposed model demonstrated its efficiency and trustworthiness through a HFL environment, ensuring that failures in other areas do not affect the operation of local data analysis.
4) A highly efficient DTL-based fine-tuning method is employed to further improve the scalability of models trained by the HFL algorithm when the power system undergoes enduring topological changes, which is novel in TSA methodologies. This method enables the model to quickly adjust to new operating conditions without extensive retraining, ensuring continuous reliability and responsiveness of TSA applications. Furthermore, it compensates for inadequate model generalization, enabling application to other power systems.

The rest of this article is organized as follows. Section II provides detailed discussions about the problem statement of TSA. The proposed architecture is outlined in Section III. In Section IV, the results from the simulation case studies are explained. Finally, Section V concludes this article.

II. PROBLEM FORMULATION

The dynamics of power systems can be encapsulated by stochastic differential and algebraic equations expressed as

$$\dot{x} = f(x, y) \tag{1}$$

$$0 = g(x, y) \tag{2}$$

$$x = \{x_i | i = 1, 2, \ldots, n\}, \quad x(t_0) = x_0 \tag{3}$$

$$y = \{y_b | b = 1, 2, \ldots, m\}, \quad y_b = [|y_b|, \angle\theta_b]^T. \tag{4}$$

The state variables, represented by $x \in \mathbb{R}^n$, evolve according to a set of differential equations, and their initial conditions are denoted by x_0. The time variable t spans an interval $[t_0, T]$. The algebraic variables, symbolized by y, include the states of the system. Assuming the availability of both raw PMU measurements and PMU-derived dynamic state characteristics at generator buses, the following five types of quantities are considered as the raw inputs for TSA: voltage magnitude (V), voltage angle (θ), frequency (F), active power (P), and reactive power (Q). The values of V, θ, and F are collected from terminal buses throughout the network while P and Q are associated with generation units and are measured at the output of these units. These quantities are denoted collectively as $\{V, \theta, F, P, Q\}$. The functions $f(x)$ and $g(x)$ encapsulate the system's nonlinear differential and algebraic equations, respectively, with n and m being the number of generators and buses. The transient stability status is determined based on the maximal phase angle discrepancy $\delta_{\max}$, which is extracted from the state vector x. The interrelation of state vector x and the algebraic variable vector V is given by the nonlinear differential-algebraic system of equations as

$$x(t_0 + \Delta t) = x_0 + \int_{t_0}^{t_0+\Delta t} f(x, V, t)\, dt \tag{5}$$

$$0 = g(x(t_0 + \Delta t), V(t_0 + \Delta t), t). \tag{6}$$

The largest phase angle difference $|\delta_{\max}|$ is determined as

$$|\delta_{\max}| = \max\left\{|\delta_i(t) - \delta_j(t)| \quad \forall i, j \in \{1, \ldots, n\}, t \in [t_0, T]\right\} \tag{7}$$

where $|\delta_{\max}|$ denotes the utmost phase angle variation between any two generators over the given time span. This measure is crucial for linking the system's stability status to the phase angle differences among the generators. The transient stability index (TSI) is a widely recognized metric for gauging the transient stability of power systems. The TSI is calculated as follows [16]:

$$\text{TSI} = 100 \times \frac{360 - \delta_{\max}}{360 + \delta_{\max}} \tag{8}$$

where $\delta_{\max}$ is the peak rotor angle difference between any two generators throughout dynamic simulations. A TSI exceeding zero signifies system stability and is denoted by a label of 1, while a negative TSI indicates potential instability, which is marked with a label of -1. The TSI is formulated as [16]

$$y = \begin{cases} 1 & \text{(Stable), TSI} > 0 \\ -1 & \text{(Unstable), TSI} \leq 0 \\ \text{Unknown} & \text{otherwise.} \end{cases} \tag{9}$$

The transient stability evaluation function $M(\cdot)$, which correlates TSI with the state vector $\zeta = [P_{\mathrm{R}}^{u}, P_{\mathrm{L}}, Q_{\mathrm{L}}, P_{\mathrm{G}}]^{T}$, can be articulated as $TSI = M(\zeta)$, where P_{R}^{u}, P_{L}, Q_{L}, and P_{G} correspond to the stochastic active power of renewable generations, such as photovoltaic and wind systems, the active power of loads, the reactive power of loads, and the active power of generators, respectively. To investigate the effects of uncertainty on system stability, Monte Carlo simulations are utilized to sample a multitude of potential outcomes $Y = \{\mathrm{TSI}_1, \mathrm{TSI}_2, \ldots, \mathrm{TSI}_N\}$ based on a probability distribution function (PDF) associated with the uncertain factors from the sample space $X = \{\xi_1, \xi_2, \ldots, \xi_N\}$.

TSA-based DL primarily aims to determine the system's stability status by analyzing transient responses shortly after disturbances. The process can be formally described as

$$M : \Psi \rightarrow \hat{y} \in \{\text{Stable, Unstable}\} \tag{10}$$

where Ψ encompasses the complete set of transient responses recorded by devices, such as PMUs, and $\hat{y}$ denotes the predicted transient stability status. The accuracy of M is critically dependent on the precision with which Ψ captures the system's transient behavior, implying that definitive stability evaluations are contingent upon accurate post-fault data.

This study leverages the capabilities of the PowerWorld simulator's transient stability module to conduct detailed simulations of symmetrical and asymmetrical grid faults. By varying model configurations based on the fault type, the simulator enables precise adjustments in impedance calculations, which are essential for diagnosing and simulating the specific impacts of different fault conditions on the power system. This approach ensures that each type of fault is accurately represented in terms of its effect on the system's impedance characteristics, focusing particularly on positive sequence faults, as illustrated in Fig. 1. The figure displays the sequence network driving point impedances, Z_0 and Z_2, across four different fault scenarios. For the three-phase balanced (3 PB) fault, Z_2 is utilized as the sequence impedance. In the single line to ground (SLG) fault, Z_2 and Z_0 are combined with the fault impedance, Z_{fault}, which is calculated as three times the fault impedance value. The line to line (LL) fault simply uses Z_2 combined with Z_{fault}. Lastly, the double line to ground fault incorporates Z_2 and Z_0, again multiplied by three times Z_{fault}, demonstrating how each type of fault requires specific adjustments in the impedance modeling to accurately reflect the fault conditions in the power system.

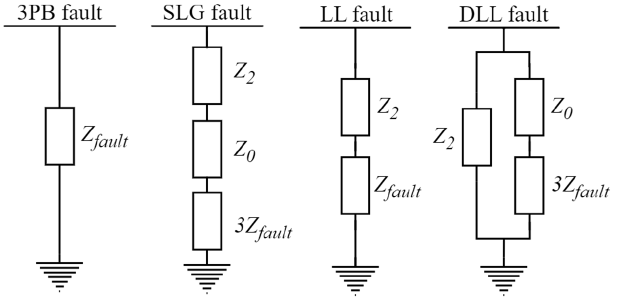

FIGURE 1. Effective impedance models for different types of electrical faults.

III. PROPOSED ARCHITECTURE

This section delves into the innovative structural designs that underpin the proposed approach to tackle the TSA problem.

A. CONVOLUTIONAL MHA NETWORK

A training dataset $D_i = \{B_j^i, C_j^i\}$ is established, where B_j^i represents the input batch and C_j^i the corresponding labels. Within this framework, a CNN model processes each input B_j^i to transform it into a k-channel signal. This transformation is defined by

$$B_j^i \xrightarrow{\text{1-D CNN}} (B_j^i, N)^{(1)}, \ldots, (B_j^i, N)^{(k)}. \tag{11}$$

Here, $B_{j,k}^i \in \mathbb{R}^{N \times m}$ denotes each channel of the CNN output, which serves as an embedded representation for the subsequent transformer layers. Each channel resulting from the convolution encapsulates localized features of the input data. These are extracted through the application of convolutional filters K, where $K = \{K_1, K_2, \ldots, K_j\}$ and each $K_n \in \mathbb{R}^{1\times 4}$. The convolution operation can be formulated as

$$S_1 = \phi\left(\mathrm{BN}\left(K * B_j^i\right) + K\right) \tag{12}$$

where $*$ denotes the convolution operation. ϕ signifies the activation function, specifically the rectified linear unit (ReLU). This function introduces nonlinearity to the network and S_1 represents the activated feature map output of the 1-D CNN. The transformation of the signal through the network layers, including self-attention and feedforward neural network (FFN) layers, can be generalized as

$$\begin{aligned} S_i'(t) = S_i(t-1)' + \text{Attention}\left(S_i(t-1)'\right) \\ + \text{FFN}\left(S_i(t-1)'\right). \end{aligned} \tag{13}$$

Here, $S_i'(t)$ represents the output of the tth transformer block for the ith sample. The attention function is computed through a MHA mechanism, and FFN denotes the position-wise FFN. The MHA mechanism can be further expanded as follows:

$$\text{Attention}(Q_u, K, V_a) = \text{softmax}\left(\frac{Q_u K^T}{\sqrt{d_k}}\right) V_a. \tag{14}$$

Here, Q_u, K, and V_a are the query, key, and value matrices derived from the input, and d_k is the scaling factor based on the dimensionality of the keys. These matrices are generated from the input embeddings, $\mathbf{X}_{\text{in}}$, through a series of linear transformations. The inputs, denoted as $\mathbf{X}_{\text{in}}$, are projected into different subspaces through three linear transformations to obtain queries Q_u, K, and V_a with a trainable weight matrices, specifically $\mathbf{W}_q^*$, $\mathbf{W}_k^*$, and $\mathbf{W}_v^*$, respectively. These projections are expressed as follows:

$$Q_u = \text{dense}(\mathbf{X}_{\text{in}}) = \mathbf{X}_{\text{in}}\mathbf{W}_q^* \tag{15}$$

$$K = \text{dense}(\mathbf{X}_{\text{in}}) = \mathbf{X}_{\text{in}}\mathbf{W}_k^* \tag{16}$$

$$V_a = \text{dense}(\mathbf{X}_{\text{in}}) = \mathbf{X}_{\text{in}}\mathbf{W}_v^*. \tag{17}$$

Finally, the FFN within each Transformer block applies two linear transformations with a ReLU function as

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

where W_1 and W_2 represent the weight matrices for the first and second linear transformations, respectively. The vectors b_1 and b_2 correspond to the bias terms for these transformations.

B. FEDERATED TRANSFER LEARNING (FTL)

DTL integrated into FL paradigms presents a groundbreaking approach to achieving TSA. The initiation of this sophisticated learning cycle involves the distribution of the global model parameters, denoted as Θ, to a selected ensemble of client nodes $C_{k\,k=1}^{\,K}$. Each client C_k engages in rigorous local optimization epochs as

$$w_{t+1}^k = w_t - \eta\nabla\ell(w_t; D_k) \tag{19}$$

where w_{t+1}^k are the updated local weights at client C_k, η represents the learning rate, and $\nabla\ell(w_t; D_k)$ denotes the gradient of the loss function ℓ with respect to the weights w_t based on the local data D_k. This local optimization is a critical step that enables each node to individually adapt to the specific characteristics of its data while contributing to the collective learning objective. Subsequently, these local updates are securely transmitted and aggregated on a central server to update the global model. The aggregation process is a weighted sum where each client's contribution is proportional to its data volume as

$$w_{t+1} = w_t - \sum_{k=1}^{K} \frac{|D_k|}{\sum_{j=1}^{K}|D_j|}\Delta w_{t+1}^k \tag{20}$$

where $\Delta w_{t+1}^k = \eta\nabla\ell(w_t; D_k)$ is the product of the learning rate and the gradient of the loss function for client C_k, and $|D_k|$ is the size of the local dataset. This federated update ensures that the global model, Θ_{t+1}, reflects a comprehensive learning trajectory shaped by the collective experience of all nodes as

$$\Theta_{t+1} = \Theta_t - \eta\left(\sum_{k=1}^{K} \frac{|D_k|}{\sum_{j=1}^{K}|D_j|}\nabla\ell(\Theta_t; D_k)\right). \tag{21}$$

The iterative nature of this process refines the model's capacity to predict grid stability with each epoch, leveraging the distributed computation for both scalability and privacy. The DTL is further enhanced by the transformer encoder's capability to process sequential data with self-attention mechanisms, captured by the transformer encoder function τ applied to the input data X as $X' = \tau(X; \Theta_{\text{transformer}})$, where $\Theta_{\text{transformer}}$ includes the parameters of the MHA and feedforward layers within the transformer encoder. The proposed method leverages HFL, wherein the dataset $D = \{D_1, D_2, \ldots, D_n\}$ is partitioned horizontally across n clients. Each client possesses a subset D_i that shares the same feature space yet contains distinct samples. In this framework, every client independently trains a local model M_i with parameters ϕ_i on their respective dataset D_i to minimize a loss function $\ell(D_i, \phi_i)$. The objective of this decentralized approach is to update the global model parameters ϕ through an aggregation function $A(\phi_1, \phi_2, \ldots, \phi_n)$, thereby iteratively refining the global model over T rounds or until convergence. To better articulate the HFL process, we integrate and reformulate the update mechanism as follows:

$$\phi^{(t+1)} = A\left(\phi_1^{(t)}, \phi_2^{(t)}, \ldots, \phi_n^{(t)}\right). \tag{22}$$

Here, $\phi^{(t+1)}$ signifies the global model parameters updated for the subsequent iteration $t+1$, where $A(\cdot)$ represents the aggregation function. This function synthesizes the updated parameters $\phi_i^{(t)}$ from each client i at the current iteration t, with the goal of achieving convergence through T iterative rounds. This aggregation process is vital for enhancing the model's ability to predict grid stability effectively by pooling the strengths of local models trained on diverse subsets of the overall dataset.

C. PROPOSED MODEL

The FLACON network is designed for TSA in power systems [17]. The model adeptly captures both localized and extended patterns within sequences, ensuring a nuanced understanding of temporal dynamics. At the initial stage, convolutional layers process the input data. For an input sequence X, a series of convolutional operations are applied through layers $\{\text{Conv1D}_k\}_{k=1}^N$, where N represents the number of such layers. These operations are defined as

$$Y_k = f(W_k * X + b_k). \tag{23}$$

Here, $*$ represents the convolutional operation, W_k is the kernel, b_k is the bias, and f is the ReLU activation function. Subsequently, the Transformer encoder layers, through the self-attention mechanism, process the resultant feature maps. Within the MHA framework, each attention head h_i computes as

$$\text{head}_i = \text{Attention}\left(Q_u W_i^{Q_u}, KW_i^K, V_a W_i^{V_a}\right). \tag{24}$$

The outputs from all attention heads are concatenated and then linearly transformed as

$$O = \text{Concat}\left(\text{head}_1, \ldots, \text{head}_h\right) W^O. \tag{25}$$

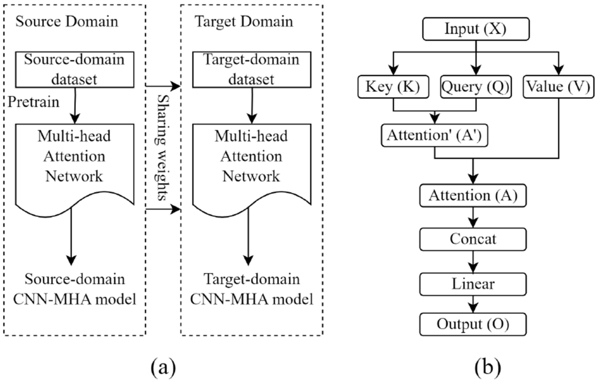

FIGURE 2. (a) DTL (b) MHA mechanisms in domain-adaptive CNN-MHA model.

Each Transformer encoder layer incorporates these attention computations, alongside FFNs, normalization, and skip connections as

$$H' = \text{LayerNorm}(X + \text{MultiHead}(Q_u, K, V_a)) \tag{26}$$

$$O' = \text{LayerNorm}(H' + \text{FFN}(H')). \tag{27}$$

The network employs a focal loss function during optimization to concentrate learning. In essence, the FLACON is primed for tasks where the discernment of complex temporal sequences is crucial. By adopting a decentralized approach, it eliminates the need for central data aggregation, thereby mitigating concerns over privacy and computational efficiency. The FL aspect of FLACON allows for localized model training on distributed datasets, ensuring that the unique characteristics of regional data are captured and utilized effectively. This approach reduces the latency associated with centralized data processing and enhances the responsiveness of the system to dynamic grid conditions. Moreover, the integration of TL enables the model to benefit from pretrained MHA network as seen in Fig. 2. The TL through adaptive fine-tuning provides a head start in learning and adapts more rapidly to the unique features of electrical grid data.

The MHA mechanism allows for a more nuanced understanding of the relationship between different regions of the input data, providing insights into how specific features influence the overall stability assessment. The FLACON architecture also addresses class imbalance through a balance factor-based focal loss function, enhancing the model's ability to learn from underrepresented classes in the dataset. This feature is particularly beneficial for TSA, where certain types of faults may occur infrequently but have significant implications for grid stability.

In the training of the model, a focal loss function, denoted as ℓ_{focal}, is employed to effectively address the challenge of class imbalance [18], which is a common issue in TSA. This specialized loss function is designed to fine-tune the model's sensitivity toward rare yet critical fault events. The ℓ_{focal} is expressed as

$$\begin{aligned}\ell_{\text{focal}}&(y, \hat{y}; \alpha, \gamma)\\ &= -\alpha(1-\hat{y})^{\gamma}\log(\hat{y}) - (1-\alpha)\hat{y}^{\gamma}\log(1-\hat{y})\end{aligned} \tag{28}$$

where y represents the true label, $\hat{y}$ denotes the predicted probability, α serves to balance the significance of positive and negative classes, and γ aims to lessen the loss contribution from straightforward examples, thereby concentrating training efforts on more challenging cases.

Further expanding on ℓ_{focal} framework, we define the ground truth class as $y \in \{+1, -1\}$, with p representing the estimated probability for the class labeled $y = 1$. For ease of explanation, the posterior probability p_t is introduced as follows:

$$p_t = \begin{cases} p & \text{if } y = 1 \\ 1-p & \text{if } y = -1. \end{cases} \tag{29}$$

By adjusting the weight of each sample's contribution to the total loss based on the classification error, the revised focal loss function is given by $L_{\text{Focal}}(p_t) = -\alpha_t(1 - p_t)^{\gamma}\log(p_t)$ [19]. Here, α_t and γ continue to play crucial roles. α_t adjusts the balance between classes, while γ acts as a focusing parameter that modulates the loss, reducing the impact of easily classified examples and thus directing the model's focus toward misclassified or hard-to-classify samples. The local models $f_i(x)$ are trained on the local data of each operator, and then the models are aggregated to form a global model $F(x)$ using FL techniques. The global model is then used for TSA across the entire power system. For notational convenience, the FL environment is computed as

$$f_i(x) = \sum_{j=1}^{N_i} w_{ij} h_j(x; \phi_{ij}) \tag{30}$$

$$F(x) = \sum_{i=1}^{K} w_i f_i(x) \tag{31}$$

$$\text{FL:} w_i^{(t+1)} = w_i^{(t)} - \eta \nabla L_i\left(w_i^{(t)}, F^{(t)}\right) \tag{32}$$

where $f_i(x)$ is the local model at operator i, w_{ij} is the weight of tree j in the ensemble for operator i, $h_j(x; \phi_{ij})$ is the jth decision tree in the ensemble with parameters ϕ_{ij}, $F(x)$ is the global model, K is the number of power system operators, w_i is the weight of operator i in the global model, η is the learning rate, $L_i(w_i, F)$ is the loss function for operator i with weights w_i and global model F.

Fig. 3 illustrates the schematic of the FL approach used for TSA. These clients represent different nodes within a power grid or distributed computing systems that handle local data processing. The clients are also responsible for hyperparameter optimization, which, once refined, are sent back to the central server. This iterative process ensures that the local models are not only well-tuned to their specific datasets but also contribute to the optimization of the global model through

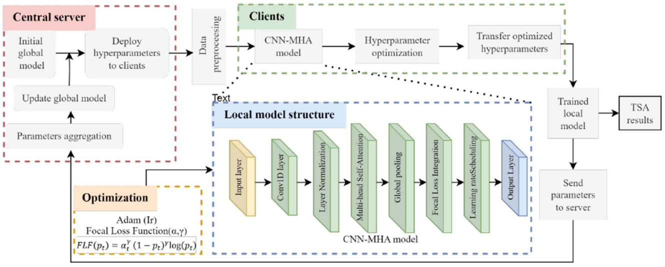

FIGURE 3. Flowchart of the Proposed Approach for TSA.

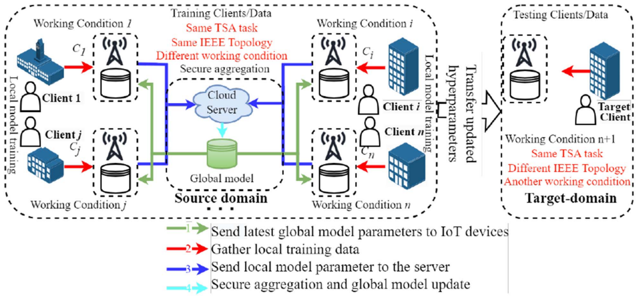

FIGURE 4. Proposed FLACON-based FTL.

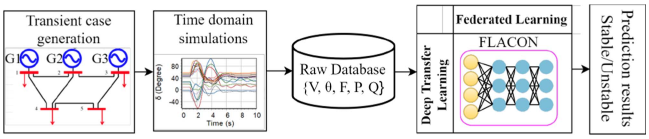

FIGURE 5. Methodology of this study.

Algorithm 1: FTL-Based FLACON Training Process.

```
Input: Distributed datasets $\{D_k\}$ across clients $\{C_k\}$
Initialize global model parameters $\Theta$
Define model architecture with Conv1D, MHA, and
    Dense layers
Conv1D layer: $X_{\text{conv}} = \text{ReLU}(\text{Conv1D}(X))$
MHA layer: $X_{\text{mha}} = \text{softmax}(\frac{Q_a K^T}{\sqrt{d_k}})V_a$
Dense layer: $X_{\text{dense}} = \sigma(W_d X + b_d)$
Focal loss: $L_{\text{focal}}(y, \hat{y}) = -\alpha(1 - \hat{y})^{\gamma} \log(\hat{y})$
Define fine-tuning with DTL: $X_{\text{new}} = \text{DTL}(X_{\text{old}}, D_{\text{new}})$
for each client $C_k$ in parallel do
        Initialize local model with parameters $\phi_k$ from $\Theta$
    Segment data into sequences $D_{k,\text{seq}}$ using sliding
    window
    Divide $D_{k,\text{seq}}$ into $D_{k,\text{train}} = \{(\mathbf{S}_i^{k,\text{train}}, y_i^{k,\text{train}})\}$,
    validation $D_{k,\text{val}} = \{(\mathbf{S}_i^{k,\text{val}}, y_i^{k,\text{val}})\}$, and testing
    $D_{k,\text{test}} = \{(\mathbf{S}_i^{k,\text{test}}, y_i^{k,\text{test}})\}$
end for
repeat
    for each client $C_k$ in parallel do
        Train local model on $D_{k,\text{train}}$ using focal loss and
        optimizer Adam
        Evaluate local model on $D_{k,\text{val}}$
        Calculate gradients $\nabla L_k(\phi_k)$ with respect to $\ell_{\text{focal}}$
    end for
    Aggregate gradients $\nabla L(\Theta) = \sum_k \nabla L_k(\phi_k)$ on the
    server
    Update global model parameters $\Theta$ using aggregated
    gradients
    Apply fine tuning to adapt the global model to new
    data distribution $D_{\text{new}}$
    Broadcast updated $\Theta$ to all clients $C_k$
until convergence or maximum number of iterations
    reached
Output: Global trained model with fine-tuned
    parameters optimized for TSA in distributed settings
```

parameter aggregation. Fig. 4 depicts the architecture of a FTL-based FLACON model for TSA. As seen in the figure, the framework involves various clients (Client 1 to Client n), each working under different conditions but on the same TSA task and IEEE topology. The architecture includes a target domain with a different IEEE topology and working conditions. This suggests that the global model's learned features can be transferred to a new domain that shares the same TSA task but has different characteristics. The training process of the proposed algorithm is summarized in Algorithm 1. The algorithm operates iteratively until it meets the predefined requirements.

Fig. 5 encompasses several key processes for TSA using the FLACON model. The process starts with the generation of N transient cases $\{(X_n, y_n)\}_{n=1}^{N}$. These cases are then subjected to time-domain simulations, where the dynamics of the power system response over time are analyzed. The outcomes of these simulations are stored in a raw database containing various parameters such as voltage (V), power (P), and reactive power (Q). These data are then fed into the FLACON model through a FL framework. The final output of the model is the prediction results, which assess the transient stability of the power system under different conditions. Simulation results demonstrate that the entire process can be completed within a few cycles across various test systems, assuming PMU delays are disregarded.

D. ONLINE ASSESSMENT

Based on the discussions in Sections III-A–III-C, the FLACON model is trained in advance in the offline mode for enabling real-time TSA. During online monitoring, the proposed algorithm continuously processes incoming pseudomeasurements of $\{V, \theta, F, P, Q\}$ from PMUs and relevant dynamic state estimation from individual generators, making immediate stability predictions. The model parameters are dynamically updated using both mini-batch and periodic full updates to ensure relevance and accuracy. This approach leverages the robustness of the FLACON architecture and the efficiency of FL ensuring that the model adapts swiftly to

Algorithm 2: Real-time TSA-Based FLACON.

```
Input: Real-time data $\{S_t\}$ from PMUs
Initialize model parameters $\Theta$ from pre-trained global FLACON model
Online Data Processing:
while new data $S_t$ received do
    Pre-process $S_t$ (e.g., normalization, feature extraction)
    Predict stability using current model parameters: $y_t = \text{FLACON}(S_t, \Theta)$
    Display or record the prediction $y_t$
    Quick Update Check:
    if data batch sufficient for mini-batch update then
        Update $\Theta$ using mini-batch gradient descent with focal loss:
        $\Theta = \Theta - \eta \nabla L_{\text{focal}}(\Theta, S_{\text{mini-batch}})$
        Reset mini-batch data collector
    end if
    Periodic Full Model Update:
    if time for periodic update then
        Collect larger set of recent data $D_{\text{update}}$
        Retrain model on $D_{\text{update}}$ to refine $\Theta$
        Update global model parameters and redistribute if in federated setting
    end if
end while
Output: Continuously updated model parameters $\Theta$, real-time stability predictions
```

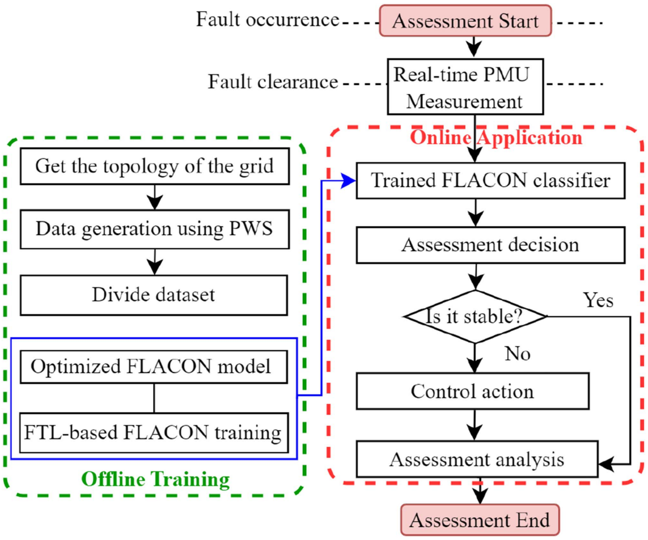

FIGURE 6. Block diagram of the proposed FLACON-based TSA.

new data and changing grid conditions, providing an effective solution for real-time TSA. Algorithm 2 provides a detailed step-by-step process for implementing the FLACON model in real-time, incorporating both mini-batch updates for swift adaptation and periodic full updates for comprehensive model refinement.

The proposed algorithmic approach ensures that the FLACON model remains up-to-date and accurate, leveraging the benefits of FL to efficiently process and learn from incoming data streams. Fig. 6 outlines the implementation of the FLACON model, incorporating both offline training and online application procedures. This setup ensures that the FLACON model, through continuous updates and real-time data processing, remains accurate and responsive to changing grid conditions, thereby providing an effective solution for real-time TSA. The robust architecture of FLACON coupled with the efficiency of HFL facilitates rapid adaptation to new data and dynamic grid conditions, underscoring the model's applicability in practical settings.

IV. CASE STUDY

This section demonstrates the feasibility and effectiveness of the proposed method through multiple simulation results. Two bus systems with different scales are discussed: IEEE 39-bus and 68-bus power systems. Both systems and transient contingencies were modeled in PowerWorld Simulator (Version 23). All simulations in this article are implemented on a computer with an Intel Core i7-9750H 4.5 GHz CPU and 16 GB RAM. The proposed method and comparative models are implemented using Python version 3.7.1 with TensorFlow framework. All comparison results are based on 10 repeated simulation trials for each IEEE bus system to obtain a mean value of recognition metrics.

A. DATA SOURCE

The asymmetrical fault scenarios encompassed instances where a single line, two lines, or both lines and the ground were compromised across various segments of the transmission lines (comprising 25%, 50%, and 75% of their total length). The premise of this investigation is the rectification of all faults within 12 cycles by severing the connection of the affected line or the busbar. During each unique fault scenario, the simulation was conducted over a span of 10 s, and the integration was measured in steps of 0.01 s. The assessment of the system's stability was carried out upon completion of each simulation. When the faults were isolated, measurements, such as bus voltages, angles, and frequencies, constituted a 1-D initial input feature. The resulting simulation data bank designated 50% of the stable scenarios and 25% of the unstable ones for training purposes. The rest, which amounts to 25%, were earmarked for validation purposes.

The t-distributed stochastic neighbor embedding (t-SNE) is used to map the high-dimension space into a 2-D space. Fig. 7 illustrates the 2-D projection using the t-SNE algorithm for the tested IEEE bus systems.

Fig. 7 illustrates that the stable and unstable samples are interspersed in the original feature space. However, in Fig. 7(a), we can see that the samples gradually separate into two distinct clusters. This separation leads to visually identify unstable cases in the representation space as compared to the

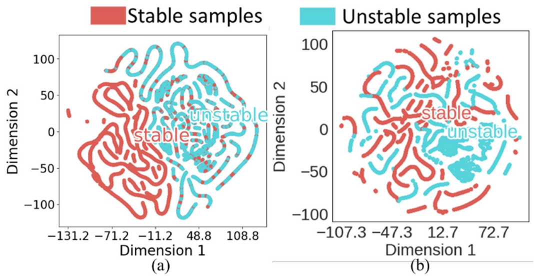

FIGURE 7. 2-D visualization of sampling strategy behaviors with t-SNE. (a) 39-bus system and (b) 68-bus system.

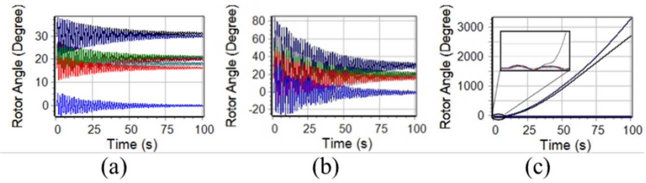

FIGURE 8. Rotor angle response for varying fault severities: (a) Definitively stable case with less severe fault, (b) marginally stable case with very severe fault, and (c) transient unstable case.

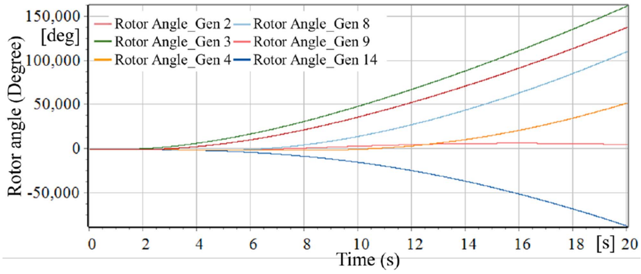

FIGURE 9. Synchronous generator rotor angle dynamics on the IEEE 68-bus system, where bus 3 experienced a SLG fault leading to an unstable contingency.

original feature space. As we can see from Fig. 7(b), there is a considerable overlap among samples, resulting in difficult separability of the samples. This shows that the model training is challenging to avoid any overfitting issues with topological changes.

Fig. 8(a) and (b) demonstrates the rotor angle of generators for less severe and more severe stable scenarios when the New England 39-bus system has suffered a short-circuit fault, respectively, while Fig. 8(c). illustrates an unstable scenario when the New England 39-bus system has suffered a short-circuit fault on the critical bus. To better understand the dynamic response of the power system to different types of faults, Fig. 9 illustrates the rotor angle instability in the IEEE 68-bus system when an SLG fault occurs at bus 3 and the fault is cleared after 0.12 s. According to the figure, the angle variations show a dramatic and continuous increase or decrease in angle, ultimately resulting in a loss of transient stability as the SLG fault influences the synchronization of the generators. When rotor angle instability occurs, the divergence of the generators' rotor angles widens over time. Fig. 10 displays a scenario where a 3 PB fault occurs at bus 6, also within the IEEE 68-bus system and the fault is cleared after 0.12 s. In Fig. 10, the angles post-disturbance trajectory display fluctuations initially but tend to stabilize and converge toward a steady state, suggesting that the system remains stable and the generators are able to maintain synchronization post-fault.

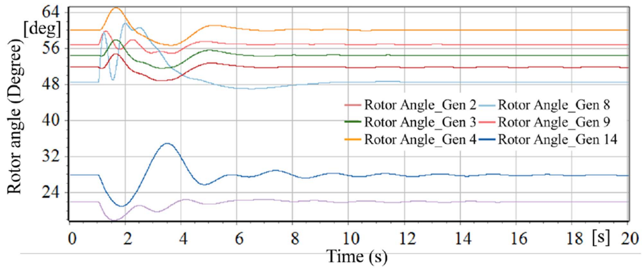

FIGURE 10. Case analysis of a stable contingency on the IEEE 68-bus system, where bus 6 experienced a 3 PB fault.

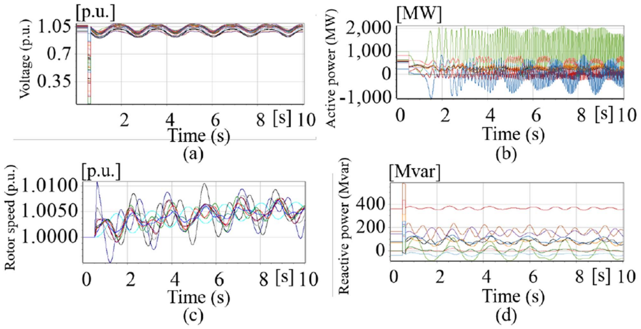

FIGURE 11. Time-domain simulation results on the IEEE 39-bus system: (a) bus voltages with a 3 PB fault on bus 10, (b) generator active power variations with LL fault on bus 10, (c) rotor speed variation with a 3 PB fault on bus 20, and (d) generator reactive power with a 3 PB fault on bus 36.

Time-domain simulations are performed for the IEEE 39-bus system using PowerWorld software and depicted in Fig. 11. Fig. 11(a) displays the bus voltages when a 3 PB fault occurs at bus 10, maintaining relatively stable voltages with minor fluctuations. Fig. 11(b) illustrates generator active power variations, which exhibit significant spikes and variability in response to an LL fault on bus 10, indicating unstable power output during the fault. Fig. 11(c) shows the rotor speed variations under a 3 PB fault on bus 20, where speeds oscillate within a narrow range, suggesting a moderate impact on generator stability. Finally, Fig. 11(d) depicts the generator's reactive power in response to a 3 PB fault on bus 36, with the power levels showing slight undulations but remaining generally stable. Together, these graphs provide insights into how different types of faults impact the stability and performance of the power system components.

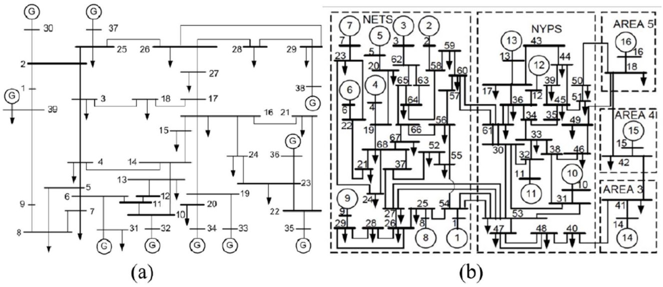

FIGURE 12. One-line diagram for (a) IEEE 39-bus system and (b) IEEE 68-bus system.

B. EVALUATION MEASURES

In the current study, the efficiency of the introduced technique is assessed through several metrics including accuracy (Acc), precision (Prec), recall (R), and F1-score (F1). These metrics were derived from the model's counts of accurate and inaccurate predictions: true positive (TP), true negative (TN), false positive (FP), and false negative (FN). The equations that define these metrics are altered as [20]

$$\mathrm{Acc} = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{TN} + \mathrm{FP} + \mathrm{FN}} \times 100\% \tag{33}$$

$$\mathrm{Prec} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}} \times 100\% \tag{34}$$

$$R = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} \times 100\% \tag{35}$$

$$F1 = 2 \times \frac{\mathrm{Precision} \times \mathrm{Recall}}{\mathrm{Precision} + \mathrm{Recall}} \times 100\%. \tag{36}$$

C. SIMULATION RESULTS

In this section, the experimental results are derived from simulations conducted on the widely recognized New England 39-bus system, frequently showcased in TSA studies [21]. As shown in Fig. 12(a), the one-line schematic of the proposed system comprises 10 power generators, 19 load points, 12 transformers, and 34 power transmission lines. To evaluate the topology changes from the source to target domains, the IEEE 68-bus test system is used and displayed in Fig. 12(b). This system has 16 synchronous generators, 86 transmission lines, and 5 areas, which represent the reduced News England test system linked with the New York power system [22]. Table 1 provides a high-level overview of the FLACON's configuration. According to the table, the Transformer block includes layers, such as layer normalization, MHA, and two dense layers with intervening dropout to prevent overfitting as shown in Fig. 2(b).

Fig. 13 displays the training and validation loss and accuracy over epochs for a transferred model. According to Fig. 13, the training and validation loss drop sharply in the initial epochs, which indicates rapid learning. A steep decline in loss suggests that the model is effectively learning patterns from the data. Both training and validation accuracy quickly reach high levels and remain stable throughout the remaining epochs. This rapid improvement suggests that the model quickly learns to capture the essential features relevant to the task of stability assessment in the initial training phase. High accuracy on validation data suggests that the model's predictions are mostly correct and it has learned the underlying patterns in the data well. The model's performance is stable across different subsets of the data, implying reliable predictions when applied to unseen measurement data.

TABLE 1. Summary of the FLACON Model Architecture

Layer (type)	Output Shape	Parameters
Input Layer	(None, 128, 1)	0
Conv1D (x5)	(None, 128, 64)	256
Global Max Pooling 1D	(None, 128)	0
Transformer Block (repeated 4 times)		
Layer Normalization	(None, 1, 128)	256
Multi-Head Attention	(None, 1, 128)	Varies
Add & Layer Normalization	(None, 1, 128)	0
Dense	(None, 1, 256)	Varies
Dropout	(None, 1, 256)	0
Dense	(None, 1, 128)	Varies
Add	(None, 1, 128)	0
Global Max Pooling 1D	(None, 128)	0
Dense (x2)	(None, 128)	16,512
Dropout	(None, 128)	0
Dense	(None, 1)	129

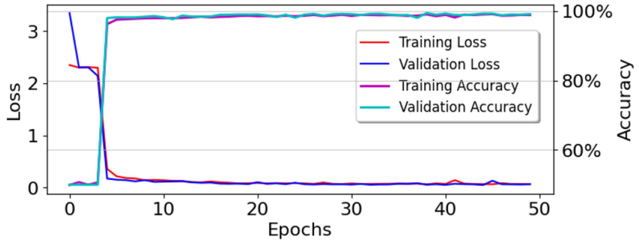

FIGURE 13. Accuracy and loss of the transferred model.

Table 2 summarizes the performance of the FLACON model under different federated client configurations. With a single client, the model achieves an accuracy of 98.98% and a recall of 98.03%. As the number of clients increases to four, the accuracy slightly decreases to 98.96% and further dips to 98.56% accuracy with six clients. An eight-client setup results in a lower accuracy of 98.46% and a recall of 99.06%, indicating a potential tradeoff between model performance and client number. On average, the FLACON model maintains a high accuracy of 98.97% and a recall of 98.48% across configurations, showcasing robustness in federated settings and illustrating that an increase in the number of clients can lead to variations in performance, with a two-client configuration yielding the best results.

TABLE 2. Performance Metrics for FLACON Model Across Different Client Configurations

Clients	1		2		4		6		8		Average	
Confusion Matrix	Stable	Unstable	Stable	Unstable	Stable	Unstable	Stable	Unstable	Stable	Unstable	Stable	Unstable
	6878	4	6878	4	6857	25	6878	4	6735	147	6845	37
	135	6743	13	6865	117	6761	193	6685	64	6814	104	6774
Metrics	Acc	Recall	Acc	Recall	Acc	Recall	Acc	Recall	Acc	Recall	Acc	Recall
Values (%)	98.98	98.03	99.87	99.81	98.96	98.29	98.56	97.19	98.46	99.06	98.97	98.48

TABLE 3. Source Metrics in Source and Target Domains

Client	Domain	Acc	Prec	Recall	F1-score
Client 1	Source domain	86.90	78.43	100	87.91
	Target domain	98.98	99.94	98.03	98.97
Client 2	Source domain	82.37	72.99	100	84.39
	Target domain	99.87	99.94	99.81	99.87

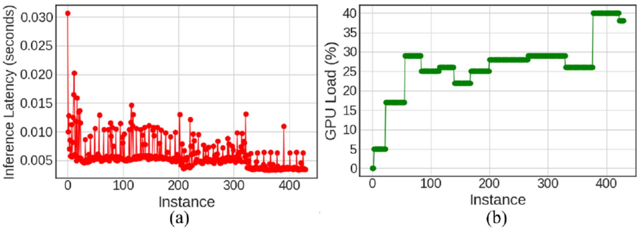

FIGURE 14. (a) Model IL/time, and (b) graphics processing unit load for different instances.

Table 3 presents a comparative analysis of the FLACON's performance in both source and target domains across two client configurations within a FTL framework for electrical TSA. In the source domain, Client 1 achieves an accuracy of 86.90%, a precision of 78.43% resulting in an F1-score of 87.91%. Interestingly, the same client shows a significant improvement in the target domain, with accuracy and precision nearly perfect at 98.98% and 99.94%, respectively, recall at 98.03%, and an F1-score of 98.97%. Client 2 exhibits a lower performance in the source domain with an accuracy of 82.37% and precision of 72.99%, but like Client 1, sees substantial gains in the target domain, achieving near-perfect accuracy and precision at 99.87% and 99.94%, respectively, along with a recall of 99.81% and an F1-score of 99.87%. These results underscore the effectiveness of FTL in enhancing model performance from source to target domain. The inference latency (IL) is the amount of time it takes for a model to process input and return an output. The IL directly impacts the speed at which a DL model can analyze data and provide insights or decisions. It can be expressed as $L = T_{\text{out}} - T_{\text{in}}$, where L, T_{in} and T_{out} represent the IL, the time at which an input data point enters the model and the time at which the output is produced by the model, respectively. In our TSA application, the average IL is measured over multiple data points to get a more accurate understanding of the model's performance. This can be defined as $L_{\text{avg}} = \frac{1}{n}\sum_{i=1}^{n}(T_{\text{out}_i} - T_{\text{in}_i})$, where L_{avg} and n represent the average IL and the total number of data points. T_{out_i} and T_{in_i} denote the times at which the output is produced and the input is received, respectively, for the ith data point.

Fig. 14 illustrates two key performance indicators for the FLACON model in a real-world scenario: IL (a) and graphics processing unit (GPU) load (b). Fig. 14(a) shows the time taken for the model to make a prediction (inference) for each instance. According to the figure, the IL is quite erratic, with a lot of variability between different instances. The variability could be due to several factors, such as differing input data complexities and background computational tasks. Fig. 14(b) shows the percentage of GPU utilization during the inference process. The GPU load appears to be increasing in a step-like fashion, which may suggest that the model's computational demands increase with more instances being processed, or it could be a result of how the inference batches are being managed. The GPU load not being at or near 100% could also mean that there is still computational headroom to increase the batch size for inferences, potentially reducing inference time through parallel processing.

This study benchmarks four DL architectures: LSTM, CNN, ANN, and CNN-LSTM models. Using a random search method for hyperparameter optimization, the LSTM and CNN models feature layers with units/filters ranging from 32 to 128, incorporating dropout rates of 0.5. The ANN model uses two dense layers and a dropout of 0.5, while the CNN-LSTM model integrates CNN and LSTM layers for enhanced feature extraction. All models employ the Adam optimizer and binary crossentropy loss functions. Moreover, all the compared models use 50 epochs for their training phase and early stopping function on the validation set for optimal accuracy and reliable assessment. Furthermore, multiple recent emerging TSA models on the IEEE 68-bus system from the literature are exploited in the comparative study including random forest LSTM (RF-LSTM) [23], graph inception neural network (GINN) [24], and ensemble support vector machines (SVMs) [25].

Table 4 shows the performance results of the classifiers for the TSA. Overall, the FLACON model excels with near-perfect accuracy at 99.87% and an F1-score at 99.09%, likely benefiting from its attention mechanism's ability to focus on different sequence parts. Conversely, the simple ANN struggles significantly, with an accuracy of 47.51%, underscoring its inadequacy for the TSA task. Meanwhile, the CNN and CNN-LSTM models show mixed results; the former balances precision and recall reasonably well, while the latter's high precision at 81.47% is diminished by a low F1-score at 46.21%, suggesting a recall deficiency. The RF-LSTM

TABLE 4. Performance Comparison of Different Methods on the IEEE 68-Bus System

Metrics	Acc (%)	Prec (%)	F1-score (%)
FLACON	**99.87**	**99.94**	**99.09**
CNN	82.64	79.63	82.66
LSTM	79.61	61.06	82.37
CNN-LSTM	64.32	81.47	46.21
ANN	47.51	47.51	64.42
RF-LSTM [23]	98.98	92.45	95.14
GINN [24]	98.40	99.27	99.02
Ensemble SVMs [25]	97.31	97.01	97.40

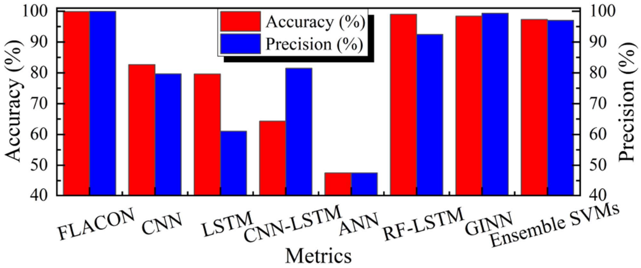

FIGURE 15. Flowchart of model comparison based on the accuracy and precision metrics.

TABLE 5. Accuracy and Training Time Comparison With Previous Studies

Proposed	Previous studies				
FLACON	FC [24]	SLODL [27]	SLODC [28]	RCNN [29]	HDLM [26]
Accuracy (%)					
99.87	97.37	99.09	99.01	98.92	99.25
Training time (minutes)					
4.95	14.49	1841.68	35.11	7.80	14.54

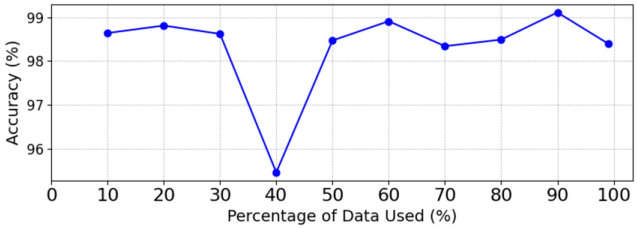

FIGURE 16. Accuracy versus percentage of data used.

model exhibits strong performance metrics with an accuracy of 98.98% and an F1-score of 95.14%, though it does not surpass FLACON. Similarly, the GINN closely rivals FLACON, delivering an accuracy of 98.40%, precision of 99.27%, and an F1-score of 99.02%, showing only a slight lag in accuracy. In addition, ensemble SVMs maintain robust performance as well, with all metrics above 97%, yet they still fall short of FLACON's results. The proposed FLACON model surpasses all the existing methods by a significant margin.

Fig. 15 provides a visual representation of various models' performance based on accuracy and precision metrics. According to the figure, the FLACON model leads with near-perfect scores in both accuracy and precision, closely mirrored by the GINN and RF-LSTM models, which also exhibit high performance in both metrics. Overall, this comparative analysis clearly highlights the superiority of FLACON, GINN, RF-LSTM, and ensemble SVMs in terms of these two critical metrics.

For the TSA classification problem, the FLACON model is compared with other benchmarks from the recent literature. The competitive models include the hierarchical DL machine (HDLM) [26], fault cluster (FC) [27], supervised learning of overcomplete dictionaries-based LSTM (SLODL) [28], supervised learning of overcomplete dictionaries-based CNN (SLODC) [28], and heatmap representation-based CNN (RCNN) [29]. Table 5 presents the accuracy and training time comparison with previous studies.

According to Table 5, the FLACON model achieves an exceptionally high accuracy of 99.87%. The HDLM is the next best performer at 99.25%, followed closely by SLODL at 99.09%. The proposed model outperforms other models, such as the FC and SLODC, at 97.37% and 99.01%, respectively. This high accuracy highlights FLACON's robustness and advanced feature extraction capabilities by integrating focal loss and attention mechanisms, making it highly effective in TSA. Moreover, the FLACON model demonstrates a significant advantage in training time, requiring only 4.951 min. This is substantially faster than all other methods. The next fastest is RCNN at 7.80 min. Notably, SLODL has an extremely long training time of 1841.68 min (about 30.7 h). The proposed FLACON model not only demonstrates superior accuracy and short training time but also introduces advanced features. These features address critical issues, such as decentralized training, overlooked by the benchmark models in dynamic and privacy-sensitive smart grid environments. Fig. 16 depicts the model sensitivity to data size.

From Fig. 16, the model accuracy generally improves with an increase in data usage, peaking at 99.11% when 90% of the data are utilized. However, there is a notable exception at 40% data usage where accuracy dips to 95.46%, suggesting that at certain thresholds, the model may require more data to maintain or improve its predictive performance. The highest accuracies are achieved with smaller (10%–20%) and larger (60%–90%) data portions, indicating that the model can perform well both with limited data and as more data becomes available.

V. CONCLUSION

This study introduced FLACON-based TSA framework by harnessing the synergy of CNN and MHA mechanisms, fortified by FTL. The integration of a focal loss function tailored to balance the class distribution enhances the robustness of the proposed FLACON model, enabling it to deliver high predictive accuracy in the complex domain of TSA under symmetrical and asymmetrical grid faults. The decentralized training approach, pivotal in preserving data privacy and minimizing communication overhead, represented a significant advancement over traditional centralized DL methods. The inductive TL methodology further refines the FLACON by streamlining the hyperparameter tuning process, which results in a considerable reduction in training time without compromising the model's performance. Empirical evaluations, conducted using the New England 10-machine 39-bus testing system and the IEEE New York/New England 68-bus system, underscore the model's superior accuracy, which peaks at 98.98%, highlighting its efficacy. The FLACON model, therefore, stands as a scalable, efficient, and secure solution, offering notable improvements in TSA. Future work focuses on solving other types of grid instabilities and exploring the model interpretability for computer-aided stability analysis.

REFERENCES

[1] D. Arnold et al., "Adaptive control of distributed energy resources for distribution grid voltage stability," *IEEE Trans. Power Syst.*, vol. 38, no. 1, pp. 129–141, Jan. 2023.

[2] N. Hatziargyriou et al., "Definition and classification of power system stability revisited & extended," *IEEE Trans. Power Syst.*, vol. 36, no. 4, pp. 3271–3281, Jul. 2021.

[3] Y. Cheng, N. Yu, B. Foggo, and K. Yamashita, "Online power system event detection via bidirectional generative adversarial networks," *IEEE Trans. Power Syst.*, vol. 37, no. 6, pp. 4807–4818, Nov. 2022.

[4] M. Savastianov, K. Smedley, and J. Cao, "Power system recovery from momentary cessation with transient stability improvement," *IEEE Trans. Power Syst.*, vol. 39, no. 4, pp. 6014–6025, Jul. 2024.

[5] M. S. Massaoudi, H. Abu-Rub, and A. Ghrayeb, "Navigating the landscape of deep reinforcement learning for power system stability control: A review," *IEEE Access*, vol. 11, pp. 134298–134317, 2023.

[6] T. Su, Y. Liu, J. Zhao, and J. Liu, "Probabilistic stacked denoising autoencoder for power system transient stability prediction with wind farms," *IEEE Trans. Power Syst.*, vol. 36, no. 4, pp. 3786–3789, Jul. 2021.

[7] W. Y. B. Lim et al., "Decentralized edge intelligence: A dynamic resource allocation framework for hierarchical federated learning," *IEEE Trans. Parallel Distrib. Syst.*, vol. 33, no. 3, pp. 536–550, Mar. 2022.

[8] A. Ahmed, S. Basumallik, A.K. Srivastava, Y. Wu, and S. Choudhury, "Federated synchrophasor data prediction, aggregation and inference using deep learning: A case of proactive control for short-term stability," *IEEE Trans. Power Del.*, vol. 39, no. 2, pp. 823–834, Apr. 2024.

[9] M. Massaoudi, A. Ghrayeb, M. Begovic, and T. Huang, "Leveraging explainable extremely randomized trees model for poisoning attack detection in power grid stability assessment," in *Proc. 4th Int. Conf. Smart Grid Renewable Energy*, 2024, pp. 1–6.

[10] C. Ren et al., "QFDSA: A quantum-secured federated learning system for smart grid dynamic security assessment," *IEEE Internet Things J.*, vol. 11, no. 5, pp. 8414–8426, Mar. 2024.

[11] C. Ren, T. Wang, H. Yu, Y. Xu, and Z. Y. Dong, "EFedDSA: An efficient differential privacy-based horizontal federated learning approach for smart grid dynamic security assessment," *IEEE Trans. Emerg. Sel. Topics Circuits Syst.*, vol. 13, no. 3, pp. 817–828, Sep. 2023.

[12] M. Hijazi, P. Dehghanian, and S. Wang, "Transfer learning for transient stability predictions in modern power systems under enduring topological changes," *IEEE Trans. Autom. Sci. Eng.*, early access, Jun. 1, 2023, doi: 10.1109/TASE.2023.3277536.

[13] Q. Chen, H. Wang, and N. Lin, "Imbalance correction method based on ratio of loss function values for transient stability assessment," *CSEE J. Power Energy Syst.*, early access, May 6, 2022, doi: 10.17775/CSEEJPES.2021.00290.

[14] F. Li, Q. Wang, Y. Tang, Y. Xu, and J. Dang, "Hybrid analytical and data-driven modeling based instance-transfer method for power system online transient stability assessment," *CSEE J. Power Energy Syst.*, early access, Apr. 30, 2021, doi: 10.17775/CSEEJPES.2020.03880.

[15] J. Kim, H. Lee, S. Kim, and J. H. Park, "Transient stability assessment using deep transfer learning," *IEEE Access*, vol. 11, pp. 116622–116637, 2023.

[16] G. Wang et al., "A physical mechanism enabled neural network for power system dynamic security assessment," *CSEE J. Power Energy Syst.*, early access, Jun. 27, 2023, doi: 10.17775/CSEEJPES.2022.08800.

[17] K. Wang, J. He, and L. Zhang, "Attention-based convolutional neural network for weakly labeled human activities' recognition with wearable sensors," *IEEE Sensors J.*, vol. 19, no. 17, pp. 7598–7604, Sep. 2019.

[18] J. Tian et al., "Synergetic focal loss for imbalanced classification in federated xgboost," *IEEE Trans. Artif. Intell.*, vol. 5, no. 2, pp. 647–660, Feb. 2024.

[19] S.-Y. Wang, Z. Qu, and L.-Y. Gao, "Multi-spatial pyramid feature and optimizing focal loss function for object detection," *IEEE Trans. Intell. Veh.*, vol. 9, no. 1, pp. 1054–1065, Jan. 2024.

[20] M. Massaoudi, S. S. Refaat, A. Ghrayeb, and H. Abu-Rub, "Short-term dynamic voltage stability status estimation using multilayer neural networks," in *Proc. IEEE Texas Power Energy Conf.*, 2023, pp. 1–6.

[21] P. Sun, L. Huo, X. Chen, and S. Liang, "Rotor angle stability prediction using temporal and topological embedding deep neural network based on grid-informed adjacency matrix," *J. Modern Power Syst. Clean Energy*, vol. 12, no. 3, pp. 695–706, May 2024.

[22] J. Ma, S. Wang, Y. Qiu, Y. Li, Z. Wang, and J. S. Thorp, "Angle stability analysis of power system with multiple operating conditions considering cascading failure," *IEEE Trans. Power Syst.*, vol. 32, no. 2, pp. 873–882, Mar. 2017.

[23] X. Zhang et al., "Power system transient stability control method based on deep learning hybrid model," in *Proc. IEEE/IAS Ind. Commercial Power System Asia*, 2021, pp. 1447–1451.

[24] S. K. Azman, Y. J. Isbeih, M. S. El Moursi, and K. Elbassioni, "A unified online deep learning prediction model for small signal and transient stability," *IEEE Trans. Power Syst.*, vol. 35, no. 6, pp. 4585–4598, Nov. 2020.

[25] Y. Zhou, J. Wu, Z. Yu, L. Ji, and L. Hao, "A hierarchical method for transient stability prediction of power systems using the confidence of a svm-based ensemble classifier," *Energies*, vol. 9, no. 10, 2016, Art. no. 778.

[26] L. Zhu, D. J. Hill, and C. Lu, "Hierarchical deep learning machine for power system online transient stability prediction," *IEEE Trans. Power Syst.*, vol. 35, no. 3, pp. 2399–2411, May 2020.

[27] S. M. Mazhari, N. Safari, C. Chung, and I. Kamwa, "A hybrid fault cluster and thé venin equivalent based framework for rotor angle stability prediction," *IEEE Trans. Power Syst.*, vol. 33, no. 5, pp. 5594–5603, Sep. 2018.

[28] R. T. Dabou, I. Kamwa, J. Tagoudjeu, and C. F. Mugombozi, "Supervised learning of overcomplete dictionaries for rapid response-based dynamic stability prediction," *IEEE Trans. Power Syst.*, vol. 37, no. 6, pp. 4912–4924, Nov. 2022.

[29] A. Gupta, G. Gurrala, and P. Sastry, "An online power system stability monitoring system using convolutional neural networks," *IEEE Trans. Power Syst.*, vol. 34, no. 2, pp. 864–872, Mar. 2019.

MOHAMED MASSAOUDI (Member, IEEE) received the Ph.D. degree in electronics engineering from the National Institute of Applied Sciences and Technology (INSAT), University of Carthage, Carthage, Tunisia, in 2022.

He has eight years of hands-on experience in applying deep learning and machine learning strategies to tackle real-world problems. During his work with Texas A&M University at Qatar, Doha, Qatar, he is the Lead Author of more than 40 peer-reviewed journal and conference publications and one book chapter. His research interests include machine learning and deep learning techniques for power system stability, energy management, cybersecurity in smart grids, and innovative prediction models.

Dr. Massaoudi was the recipient of the Outstanding Student Research Excellence Award in 2021, the Thomas W. Powell'62 and Powell Industries Inc., Fellowship award in 2024, and the Richard E. Ewing Award for Excellence in 2024 for his research contributions. His h-index is 13 and his work has been cited more than 800 times.

HAITHAM ABU-RUB (Fellow, IEEE) received the Ph.D. degree in electrical engineering from the Technical University of Gdansk, Gdansk, Poland, in 1995, and the Ph.D. degree in humanities from Gdansk University, Gdansk, in 2004.

He has worked with many universities in many countries including Poland, Palestine, USA, Germany, and Qatar. Since 2006, he has been with Texas A&M University at Qatar. For five years, he was the Chair of the Electrical and Computer Engineering Program with Texas A&M University at Qatar, Doha, Qatar, and is currently working as the Managing Director of Smart Grid Center. He has authored or coauthored more than 600 journal and conference papers, five books, and six book chapters. He has supervised many research projects on smart grid, power electronics converters, and renewable energy systems. His main research interests include electric drives, power electronic converters, renewable energy, and smart grid.

Dr. Abu-Rub was the recipient of many prestigious national and international awards and recognitions, such as the American Fulbright Scholarship and the German Alexander von Humboldt Fellowship. He is the Coeditor-in-Chief for IEEE TRANSACTIONS ON INDUSTRIAL ELECTRONICS.

ALI GHRAYEB (Fellow, IEEE) received the Ph.D. degree in electrical engineering from The University of Arizona, Tucson, AZ, USA, in 2000.

He is currently a Professor with the Department of Electrical and Computer Engineering, Texas A&M University at Qatar, Doha, Qatar. Prior to his current position, he was a tenured Professor with the Electrical and Computer Engineering Department, Concordia University, Montreal, QC, Canada. He has coauthored two books and authored or coauthored more than 250 journal and conference papers. His research interests include wireless and mobile communications, physical layer security, massive MIMO, visible light communications, smart grid, artificial intelligence and machine learning. Dr. Ghrayeb was an Instructor or Co-Instructor in many technical tutorials at several major IEEE conferences, the Executive Chair of the 2016 IEEE WCNC Conference, a Member of the IEEE ComSoc Conferences Council, the IEEE GITC Committee, and the IEEE WCNC Steering Committee. He was in different editorial capacities on a number of IEEE transactions journals. He is currently with the IEEE ComSoc Awards Committee.