Rev. Roum. Sci. Techn.– Électrotechn. et Énerg.
Vol. **70**, *3*, pp. 295–300, Bucarest, 2025

SYNTHETIC OPERATIONAL DATA GENERATION FOR DEEP LEARNING APPLICATIONS IN POWER TRANSMISSION LINES

MUHAMMAD WAQAS[1], FEZAN RAFIQUE [2], LING FU[1], RUIKUN MAI[1]

Keywords: Conditional tabular generative adversarial network (CTGAN); Fault detection; Principal component analysis (PCA); Synthetic data generation.

Deep learning (DL)-based protection algorithms for power transmission lines require large volumes of operational data for accurate training. However, such data is often complex to access due to confidentiality, restrictions, and proprietary limitations. This paper proposes a synthetic data generation method that combines principal component analysis (PCA) with a conditional tabular generative adversarial network (CTGAN). PCA reduces the dimensionality of high-frequency time-series data, allowing CTGAN to operate efficiently while retaining essential statistical characteristics. The generated synthetic data shows strong correlation with real data and effectively augments limited datasets. Validation using an LSTM-based fault classification model demonstrated an improvement from 50.93% to 86.07% accuracy. Additional validation using sub-synchronous oscillation data demonstrates broader applicability. The proposed method is scalable and supports DL training in data-scarce scenarios.

1. INTRODUCTION

Power transmission lines are vital elements of power systems responsible for transmitting bulk energy from power sources to consumers. However, owing to their constructional design and extensive geographical coverage, these transmission lines are persistently exposed to the risk of faults, which in turn can have significant operational and economic implications [1]. The timely and accurate detection of these faults holds utmost importance to prevent their potentially disastrous effects and minimize resultant damages. Therefore, several fault detection techniques are proposed in the literature [2]. Recently, Artificial Intelligence (AI), which encompasses Machine Learning (ML) and Deep Learning (DL), has garnered significant interest for designing protection algorithms due to its strong pattern recognition capabilities. Several ML/DL based fault detection algorithms are reported in the literature with superior accuracy [3]. However, designing ML/DL algorithms requires an extensive volume of training data. Availability of a comprehensive dataset for training the ML/DL algorithm is a challenging undertaking. Therefore, this paper attempts to introduce an effective method for generating synthetic data, which can supplement the training process of the ML/DL model with actual data.

ML/DL is a combination of computer hardware and software arrangement that utilizes a training dataset to learn an objective [4], generally with backpropagation (BP), and can be deployed for real-world tasks [5]. The utilization of AI for fault diagnosis is documented in the literature after the emergence of the artificial neural network (ANN) [3]. Typically, a training dataset with annotated outputs is employed to optimize the node weights via the back-propagation algorithm, a technique known as supervised learning [6], which stands as the prevailing method

The recent trend in AI is to use DL for designing fault detection algorithms. DL is a sub-branch of ML [7], which utilizes the intrinsic patterns in raw data for discovering the representations needed for detection or classification tasks [8]. Traditional ML involves designing a feature extractor, and the features are used for pattern recognition in data [9]. DL has integrated the feature learning and decision making into a single algorithm through multiple processing layers [10], most of which can learn non-linear input-output mappings. The modules in the stack modify their input to enhance the specificity and consistency of the representation. By having numerous non-linear layers, ranging from 5 to 20 layers deep, the system can execute complex functions of its inputs that are highly responsive to subtle details while remaining unresponsive to significant, irrelevant changes, such as environmental factors, position, illumination, and surrounding objects [8]. Most popular DL architectures include convolutional neural networks (CNN), long short-term memory (LSTM) units, etc. [10].

Several DL studies have been reported in the literature that utilize deep models for transmission line fault diagnostics. A DL architecture utilizing capsule networks (CN) for fault identification in EHV transmission lines was proposed in [11]. The model has achieved a notable accuracy of 99.7%. However, the training process requires 38,115 examples. Similarly, another CNN-based study [12] utilized a CNN with self-attention, which requires a training dataset of 228,690 examples. [13] used a transfer learning approach with CNN, which allows relatively less training time and data, *i.e.*, 25000 examples of faults. [14] utilized LSTM for featureless robust fault detection; this method is trained on 27,000 training examples. All these methods have strong performance, but they are trained using simulated datasets; therefore, getting such a large volume of training data was not a problem. However, if the training is to be performed on real telemetry data, obtaining such a large volume of data with equal contribution of each shunt fault is challenging. The availability and accessibility of transmission line data pose significant challenges since it is often sensitive and proprietary [15]. Therefore, in this paper, we attempt to introduce a first-of-its-kind synthetic data generation method that produces synthetic data with a strong correlation to the original data and can augment the training process alongside real data.

In the context of the literature review, this paper proposes a computationally friendly synthetic data generation method that can produce look-alike copies of the real data using a hybrid of principal component analysis (PCA) and conditional tabular generative adversarial network (CTGAN) introduced in [16]. CTGAN, by design, does not model temporal dependencies inherent in time series data. However, applying PCA to high-resolution time series data can compress temporal patterns into a lower-dimensional representation that retains key statistical features. CTGAN

[1] School of Electrical Eng., Southwest Jiaotong University, Chengdu, China.
[2] Department of Electrical Eng., NED University of Eng. & Tech, Karachi, Pakistan.
E-mails: muhammad.waqas@my.swjtu.edu.cn, fezan@neduet.edu.pk, lingfu@swjtu.edu.cn, mairk@swjtu.edu.cn

DOI: 10.59277/RRST-EE.2025.70.3.2

can then generate synthetic data in this reduced space. This reduces the computational overhead of CTGAN. The generated data can help train DL models with limited real data. The key contributions of this work lie in the use of synthetic data to support DL applications in power system fault diagnosis, particularly when access to real telemetry data is limited. While synthetic data generation has been explored in other fields, its application to power system operational data, especially for training fault classification models, is limited. Most prior studies rely on simulated data, whereas this work focuses on generating synthetic data that statistically mimics actual telemetry data.

Furthermore, to address the computational burden associated with generating high-resolution time-series data using CTGAN, we introduce the use of PCA as a dimensionality reduction step before data synthesis. Although PCA and GANs have been independently used in other domains, the specific combination of PCA and CTGAN for generating synthetic power system data has not been previously reported. This hybrid approach enables efficient and scalable synthetic data generation while maintaining correlation with the original data distribution. The method is validated on two distinct power system scenarios, highlighting its practical applicability and generalization potential.

This article is organized as follows: section 2 covers the brief theoretical background of CTGAN and PCA, section 3 describes the proposed method, results are illustrated in section 4, and conclusions are listed in section 5.

2. **THEORETICAL BACKGROUND**

2.1 GENERATIVE ADVERSARIAL NETWORKS

Generative Adversarial Networks (GANs) are a class of ML algorithms that have gained significant attention in recent years due to their ability to generate synthetic data that resembles the original data [7]. GANs have been applied in a wide range of applications, including image and video synthesis, natural language processing, and data generation. The GAN architecture comprises two neural networks: the generator and the discriminator, which operate in opposition to each other. The generator is trained to generate synthetic data from noise that resembles the original data, while the discriminator is trained to distinguish between the real and synthetic data [17]. Figure 1 represents a general arrangement of GANs. The competition between the two networks during training results in the generator learning to create increasingly realistic data. This synthetic data can be used to increase the diversity of the training dataset and improve the accuracy and robustness of the ML model [18] while the original data remains protected [19].

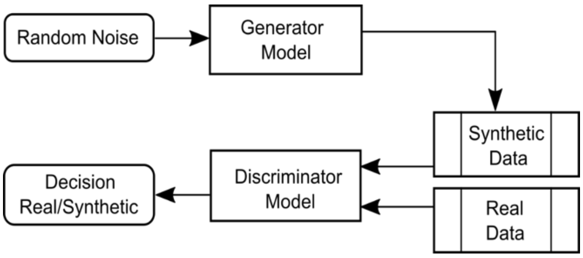

Fig. 1 – General structure of GAN generator is trained from random noise; the Discriminator uses synthetic and real data to distinguish between them.

2.2 CONDITIONAL TABULAR GENERATIVE ADVERSARIAL NETWORK

CTGAN is a special type of GAN, designed to generate synthetic tabular data while preserving the statistical properties of the original data. As transmission lines' operational data falls into the category of time series data, they can be treated as tabular data and augmented using CTGAN. Proposed by [16], CTGAN introduces new techniques specifically for tabular data, such as "*mode-specific normalization*", "*conditional-generator*", and "*training-by-sampling*". These techniques enable CTGAN to significantly outperform other methods for generating tabular data. For a numerical feature x, mode-specific normalization is applied as

$$x' = \frac{x - \mathrm{mode}(x)}{\sigma}, \qquad (1)$$

where mode(x) is the most frequent value in the column and σ is the standard deviation. Mode-specific normalization is a technique used to normalize each column of the input data based on its specific mode. The mode is defined as the value that occurs most frequently in a column. Traditional normalization techniques, such as min-max normalization or z-score normalization, treat all columns equally, regardless of their distribution. However, in some datasets, the columns may have different distributions and normalizing them in the same way may result in the loss of important information. Mode-specific normalization addresses this issue by normalizing each column based on its mode. This normalization helps stabilize training on skewed or multi-modal distributions. In CTGAN, the generator is conditioned on discrete variables, such as class labels, to ensure that the generated samples reflect the structural characteristics of the original data. This conditioning helps the model learn class-specific patterns. To address data imbalance, CTGAN employs a training-by-sampling strategy, where training batches are formed by uniformly sampling across the discrete modes. This ensures better generalization and stable learning across all categories. The generator G and discriminator D in CTGAN are trained using a conditional adversarial loss; the objective function can be given by

$$\min_G \max_D E_{x\sim P_{\mathrm{data}}}\left[\log D\,(x|c)\right] + E_{z\sim P_z}\left[\log\left(1 - D(G(z|c)|c)\right)\right] \qquad (2)$$

where z is a noise vector and c is a conditioning variable that guides generation for each class or mode. *G(z/c)* is the generator that produces synthetic samples from a noise vector z, conditioned on a category or mode c. *D(x/c)* is the discriminator to distinguish real samples x from generated ones, also conditioned on c. $E_{x\sim P_{\mathrm{data}}}\left[\log D\,(x|c)\right]$ denotes the expected value over real data, encouraging the discriminator to identify real inputs correctly. $E_{z\sim P_z}\left[\log\left(1 - D(G(z|c)|c)\right)\right]$ represents the expected value over synthetic data, guiding the generator to produce outputs that can deceive the discriminator, c typically includes metadata like class labels or categorical context, enabling class-conditional generation

2.3 PRINCIPAL COMPONENT ANALYSIS

PCA is a statistical technique that reduces dataset dimensionality by transforming the original variables into fewer uncorrelated principal components. These components are linear combinations of the original variables, ordered by the amount of variance they explain. By retaining the top components that capture most of the variance, PCA preserves key information while simplifying the data [20]. Mathematically, given a zero-mean data matrix $\mathbf{X} \in R^{n\times p}$, where *n* is the number of observations and *p* is the number of variables. PCA seeks a projection matrix $\mathbf{W} \in R^{p\times k}$ that maps **X** into a lower-dimensional space: **Z** =**XW,** where **W** consists of the top k eigenvectors of the covariance matrix $\Sigma = \frac{1}{n}\mathbf{X}^{\mathrm{T}}\mathbf{X}$.

3. PROPOSED MODEL

Since power system operational data is sampled at a high frequency, the data volume generated through such a faster sampling rate is massive. Applying CTGAN directly to the real data will require enormous computing and memory resources, making it a cumbersome process. Therefore, it is proposed to utilize PCA for dimensionality reduction. The proposed method is explained with the flow chart in Fig. 2. Firstly, data is normalized using min-max scaling. The scaled high-dimensional timeseries data $\mathbf{X} \in R^{n\times p}$ is transformed into a lower-dimensional space through PCA,

$$Z_{\mathbf{real}} = X_{\mathbf{real}}W. \quad (3)$$

where $\mathbf{W} \in R^{p\times k}$ contains the top k eigenvectors of the covariance matrix Σ . This is then followed by the generation of synthetic data using CTGAN. The lower-dimensional space $\mathbf{Z}_{\mathbf{real}}$ is used to train the CTGAN model. After training, the generator *G*(*z*|*c*) produces synthetic samples

$$\boldsymbol{Z}_{\mathbf{syn}} = G(z|c). \quad (4)$$

The lower-dimensional synthetic data is then transformed to the original feature space through inverse PCA and inverse scaling,

$$\mathbf{X}_{\mathbf{syn}} = \mathbf{Z}_{\mathbf{syn}}\mathbf{W}^{T}. \quad (5)$$

To evaluate the effectiveness of the proposed synthetic data generation method, the end-to-end LSTM model introduced by Rafique et al. in [14] was adopted. It is a fault classification model that utilizes end-to-end learning through LSTM, as adopted. The original model is trained on 27,000 examples and classifies faults into five categories: no fault (NF), line to ground (LG), line to line (LL), line to line to ground (LLG), and Line to Line to line (LLL). For this study, only 270 real examples per class were selected from the original dataset to simulate a limited data scenario. The dataset was split into 80% for training and 20% for testing.

Initially, the model was trained using only the 270 real examples per class. However, the validation loss diverged, indicating that the dataset was insufficient for generalization (see Fig. 3a). To address this, synthetic data was generated using the proposed PCA–CTGAN method and combined with the real data. Separate models were trained with increasing amounts of synthetic data (1,500, 3,000, 4,500, 6,000, and 9,000 synthetic examples per class), each alongside the original 270 real examples. The training progressions are shown in Fig. 3(b)–(f).

4. RESULTS

This section presents the results of the trained models using the proposed synthetic data technique.

4.1 FAULT CLASSIFICATION USING END-TO-END LSTM

The objective of this evaluation is not to propose a new ML-based fault classification model, but to demonstrate that the proposed synthetic data generation method can support existing DL architectures in data-scarce scenarios. In this study, the LSTM model from [14] serves as a reference to evaluate how model performance changes when real data is supplemented with synthetic data generated using the proposed PCA–CTGAN approach. This allows a controlled evaluation of the data augmentation strategy without introducing architectural biases.

Six models were trained using different combinations of real and synthetic data. All models were evaluated using real test data only to ensure consistent benchmarking. The training hyperparameters were kept identical across all models and followed those specified in [14]. A comprehensive confusion matrix for each trained model is presented in Fig. 4. These matrices are generated using real test data, with fault impedance values randomly ranging from 0.1 Ω to 250 Ω to reflect practical operating conditions. As seen in Fig. 4(a), the model trained solely on 270 real examples per class exhibits poor generalization, with high misclassification across all fault types, particularly between LG, LLG, and LLL faults. As synthetic data is progressively added (as shown in Fig. 4(b) to Fig. 4(f)), the classification accuracy improves noticeably across all classes, and the misclassification rates decrease significantly. The best results are obtained when the model is trained with 9,000 synthetic samples per class, as shown in Fig. 4(f), where the diagonal dominance in the matrix confirms strong class-specific learning. This trend illustrates the potential value of the proposed PCA-CTGAN-based synthetic data generation approach in enhancing ML models when real data is limited. These results are also summarized in Table 1.

Table 1

Performance comparison of models with real and synthetic data.

Model	Real Examples Each Class	Synthetic Examples Each Class	Accuracy	True Positives (out of 1500)
1	270	0	50.93%	764
2	270	1500	62.87%	960
3	270	3000	58.67%	880
4	270	4500	85.40%	1281
5	270	6000	80.67%	1210
6	270	9000	86.07%	1291

4.2 SUB-SYNCHRONOUS OSCILLATIONS

To demonstrate the general applicability of the proposed synthetic data generation approach, the method was further tested on sub-synchronous oscillations (SSO) caused in the grid due to the integration of doubly fed induction generators. PMU data of phase A is used to train the CTGAN. The dataset used in this example is introduced in [21], and a screenshot of it is shown in Fig. 5. The dataset is publicly available at *IEEE Dataport* [22]. Duration of the first two seconds is used for synthetic data generation. Figure 6 illustrates that the synthetic signals closely resemble the real SSO waveforms in terms of both amplitude and frequency characteristics. This result confirms that the method can successfully learn and reproduce statistically consistent patterns even in non-classification, waveform-level applications. More importantly, it highlights the flexibility of the approach in supporting a range of machine learning tasks in power systems through data augmentation with realistic synthetic data, particularly when access to operational records is constrained.

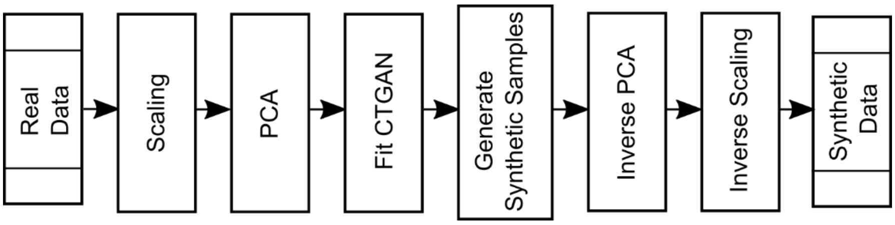

Fig. 2 – Proposed synthetic data generation method, PCA reduces dimensionality of raw data (left), enabling CTGAN to synthesize realistic samples (right).

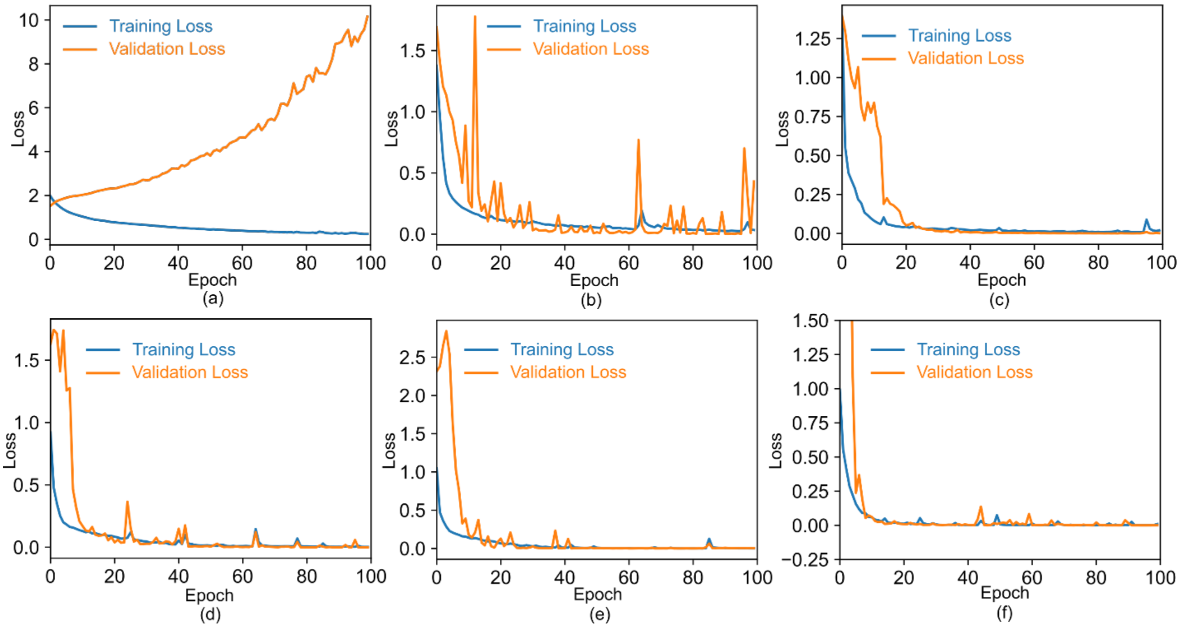

Fig. 3 – Training progress of end-to-end model with real and synthetic data (a) using 270 examples of real data (b) 270 real, 1500 synthetic, (c) 270 real, 3000 synthetic, (d) 270 real, 4500 synthetic, (e) 270 real, 6000 synthetic, (f) 270 real, 9000 synthetic.

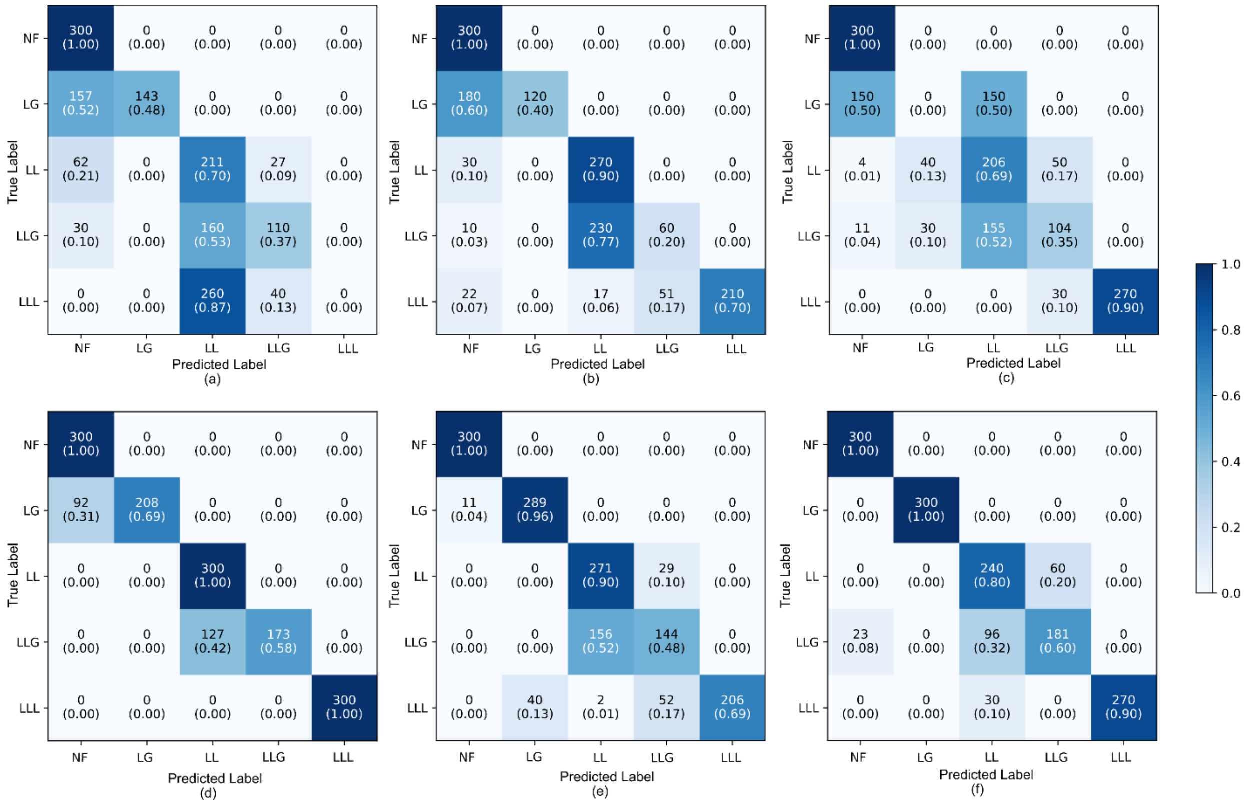

Fig. 4 – Confusion matrix for fault classification (a) Model 1, (b) Model 2, (c) Model 3, (d) Model 4, (e) Model 5, (f) Model 6.

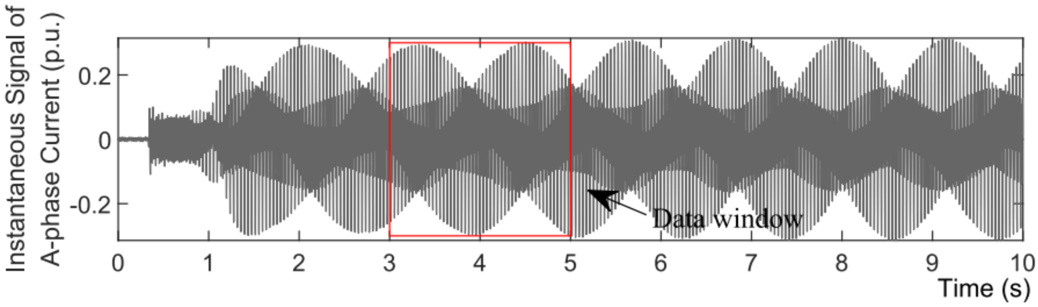

Fig. 5 – Screenshot of instantaneous current PMU data, during sub-synchronous oscillations, adapted from [21].

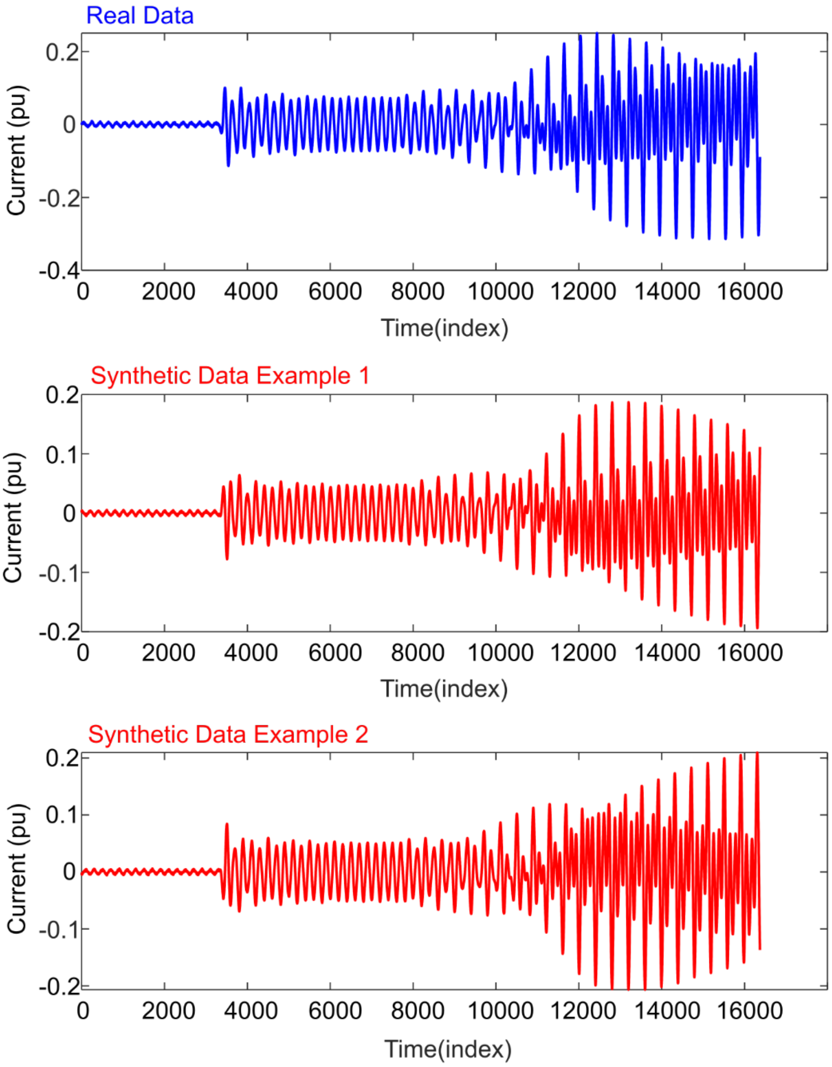

Fig. 6 – Training and synthetic data for sub-synchronous oscillations.

5. CONCLUSIONS

This paper presented an innovative hybrid approach combining PCA and CTGAN to generate synthetic power system operational data for training ML models. Our work makes significant contributions by introducing the first PCA-CTGAN framework specifically designed for power system timeseries data, effectively addressing the critical challenge of limited real-world datasets while overcoming the computational barriers of processing high-resolution measurements. The method's effectiveness is demonstrated through substantial improvements in fault detection performance, where augmentation with synthetic data boosts LSTM model accuracy from 50.93 % to 86.07 %, a remarkable 35.14 % enhancement that validates the approach's ability to compensate for data scarcity. By incorporating PCA as a preprocessing step, we achieved efficient dimensionality reduction, making the solution practical for real-world high-frequency power system applications. Beyond fault detection, the method's strength is further confirmed by its successful application to sub-synchronous oscillation scenarios, suggesting a broader potential for power system monitoring and protection tasks. These advances offer power utilities a practical solution to overcome data limitations while maintaining model performance, with promising extensions to renewable integration and other emerging grid challenges. The reproducible framework, validated on publicly available datasets, sets a foundation for future research in synthetic data generation for power systems.

CREDIT AUTHORSHIP CONTRIBUTION STATEMENT

Muhammad Waqas – investigation, methodology, writing original draft
Fezan Rafique - data curation; investigation; methodology; software; analysis; visualization; writing original draft
Ling Fu: resources; supervision; validation; writing review & editing
Ruikun Mai: resources; supervision; validation; writing review & editing references.

Received on 9 September 2024

REFERENCES

1. M.S. Uddin, et al., *On the protection of power system: Transmission line fault analysis based on an optimal machine learning approach*, Energy Reports, **8**, pp. 10168–10182 (2022).
2. K. Chen, C. Huang, J. He, *Fault detection, classification and location for transmission lines and distribution systems: a review on the methods*, High Voltage, **1**, *1*, pp. 25–33 (2016).
3. A. Mukherjee, P. K. Kundu, A. Das, *Transmission line faults in power system and the different algorithms for identification, classification and localization: a brief review of methods*, Journal of The Institution of Engineers (India): Series B, **102**, *4*, pp. 855–877 (2021).
4. C. Janiesch, P. Zschech, K. Heinrich, *Machine learning and deep learning*, Electronic Markets, **31**, *3*, pp. 685–695 (2021).
5. D.E. Rumelhart, G.E. Hintont, *Learning Representations by Back–Propagating Errors*, Cognitive Modeling, The MIT Press, **2**, pp. 3–6 (2002).
6. C. González García, E. Núñez–Valdez, V. García–Díaz, C. Pelayo G–Bustelo, J. M. Cueva–Lovelle, *A review of artificial intelligence in the internet of things*, International Journal of Interactive Multimedia and Artificial Intelligence, **5**, *4*, p. 9 (2019).
7. A. Gulli, P. Sujit, Deep Learning with Keras, Packt Publishing Ltd (2017).
8. Y. Lecun, Y. Bengio, G. Hinton, *Deep learning*, Nature, **521**, *7553*, pp. 436–444 (2015).
9. R. Wason, *Deep learning: Evolution and expansion*, Cogn Syst Res, **52**, pp. 701–708 (2018).
10. Y. Bengio, Y. Lecun, G. Hinton, *Deep learning for AI*, Commun ACM, **64**, *7*, pp. 58–65 (2021).
11. S.R. Fahim, S.K. Sarker, S.M. Muyeen, S.K. Das, I. Kamwa, *A deep learning based intelligent approach in detection and classification of transmission line faults*, International Journal of Electrical Power & Energy Systems, **133**, p. 107102 (2021).
12. S.R. Fahim, Y. Sarker, S.K. Sarker, M.R.I. Sheikh, S.K. Das, *Self attention convolutional neural network with time series imaging based feature extraction for transmission line fault detection and classification*, Electric Power Systems Research, **187**, p. 106437 (2020).
13. F. Rafique, L. Fu, M. Hassan Ul Haq, R. Mai, *Automatic features extraction by transfer learning for transmission line protection*, Rev. Roum. Sci. Techn. – Électrotechn. Et Énerg., **68**, *4*, pp. 339–344 (2023).
14. F. Rafique, L. Fu, R. Mai, *End–to–end machine learning for fault detection and classification in power transmission lines*, Electric Power Systems Research, **199**, p. 107430 (2021).
15. B.P. Bhattarai, et al., *Big data analytics in smart grids: state–of–the–art, challenges, opportunities, and future directions*, IET Smart Grid, **2**, *2*, pp. 141–154 (2019).
16. L. Xu, M. Skoularidou, A. Cuesta–Infante, K. Veeramachaneni, *Modeling tabular data using conditional GAN*, Adv Neural Inf Process Syst, **32** (2019).
17. I. Goodfellow, et al., *Generative adversarial networks*, Commun ACM, **63**, *11*, pp. 139–144 (2020).
18. I. Goodfellow, et al., Generative Adversarial Nets, Advances in neural information processing systems, Curran Associates, Inc. (2014).
19. T. Karras, T. Aila, S. Laine, J. Lehtinen, *Progressive growing of GANs for improved quality, stability, and variation*, 6th International Conference on Learning Representations, ICLR, pp. 1–26 (2018).
20. H. Abdi, L.J. Williams, *Principal component analysis*, Wiley Interdiscip Rev Comput Stat, **2**, *4*, pp. 433–459 (2010).
21. F. Zhang, J. Li, J. Liu, W. Gao, J. He, *An improved interpolated DFT–based parameter identification for sub–/super–synchronous oscillations with Synchrophasors*, IEEE Transactions on Power Systems, **38**, *2*, pp. 1714–1727 (2022).
22. F. Zhang, *Simulated synchrophasors in SSOs and phasor measurement data recorded during a subsynchronous oscillation incident in an actual power system*, IEEE DataPort, 255, 15, pp.:6989-6994 (2025).