A Cross-System Pre-Trained Framework for False Data Injection Attack Detection in Power Systems

Chenhan Xiao*, **Mostafa Mohammadpourfard**†, **Yang Weng***, **Suhas Pol**†
*Department of Electrical, Computer and Energy Engineering, Arizona State University, AZ, USA
†National Wind Institute, Texas Tech University, TX, USA
Email: {cxiao20, yang.weng}@asu.edu; {mostafa.mohammadpour, Suhas.Pol}@ttu.edu

***Abstract*—False Data Injection Attacks (FDIAs) pose a critical threat to power system operation, yet most existing detection methods must be retrained for each grid and require abundant system-specific attack data that are limited in practice. This paper proposes a pre-trained, topology-conditioned FDIA detection framework that generalizes across heterogeneous power systems and adapts to new grids with lightweight fine-tuning. A graph neural network (GNN) encoder represents each grid as a node–edge graph with fixed-dimensional features, enabling the model to process systems of arbitrary size and embed physical and measurement dependencies in a unified latent space. A Transformer module then captures the temporal evolution of measurement patterns, while a vector-quantized codebook provides a shared library of reusable FDIA templates and enhances transferability across systems. Experimental results on benchmark grids demonstrate that the proposed framework achieves high detection accuracy, requires significantly fewer attack labels for adaptation, and remains robust under varying topologies.**

***Index Terms*—False data injection attack, graph neural network, Transformer model, vector-quantized codebook**

I. Introduction

The growing digitalization of modern power systems, enabled by cloud-integrated monitoring, wide-area communication, and third-party analytics platforms [1], [2], has improved grid observability while simultaneously broadening the attack surface. Adversaries can exploit communication links, sensor gateways, or data-management infrastructures, turning the grid's intelligence into a liability, as demonstrated by incidents such as the 2015 Ukraine blackout [3], the 2018 compromise of U.S. grid assets [4], and coordinated energy-sector attacks in 2022 [5]. Among emerging threats, False Data Injection Attacks (FDIAs) are particularly concerning. Recent work shows that attackers can construct malicious yet physically coherent measurements [6]–[8] that bypass conventional residual-based Bad Data Detection (BDD) schemes [9]. Notably, such attacks can remain stealthy even without explicit knowledge of system topology by approximating the Jacobian matrix or the measurement manifold. These manipulated measurements can bias state estimation, degrade situational awareness, and trigger costly operational failures [9].

In response, researchers have developed a wide range of data-driven FDIA detection models that move beyond traditional BDD. Early approaches, including k-nearest neighbors, decision trees, and anomaly detectors, classify measurements as normal or attacked using labeled clean data [10]. With the rise of high-resolution sensing and deep learning techniques, more advanced architectures have emerged. RNN and LSTM models can exploit temporal correlations [11], CNN-based detectors capture local spatial patterns [12], and generative models learn the manifold of feasible system states [13]. These deep models can handle noisy measurements and more sophisticated FDIAs. However, despite promising results in controlled settings, most existing FDIA detectors face two key limitations: (1) they are trained independently for each system, so a model developed for one grid cannot be deployed on another without full retraining. (2) utilities often lack sufficient FDIA samples for system-specific training. Although normal operating data are abundant, attack instances vary across grids and are often limited, making system-specific deep learning approaches difficult to operationalize in practice.

To address these limitations, we observed an opportunity: despite different grid size and topology, FDIA designed to evade BDD share consistent structural traits across grids, including alignment with physical constraints, exploitation of BDD vulnerabilities, and low-dimensional, topology-dependent perturbation directions [14]. These insights motivate a pre-trained, topology-conditioned FDIA detector that learns a universal measurement representation from diverse systems and then adapts to a target grid through lightweight fine-tuning.

To realize this objective, we first construct a topology-conditioned encoder based on graph neural networks (GNNs). At each time step, the power grid is represented as a graph whose nodes (buses) and edges (branches) carry fixed-size feature vectors that encode both topology information (which is typically available to system defenders) and measurement values on buses and branches. This GNN representation accommodates heterogeneous system sizes by requiring only a fixed topology template (e.g., impedance and reactance) and a fixed measurement template (e.g., voltage magnitude, bus injections, branch flows) rather than a fixed number of buses; grids that lack certain measurements simply use masked feature entries. The GNN then performs message passing [15] over the physical topology and aggregates bus embeddings through permutation-invariant pooling, yielding a fixed-dimensional graph representation. This enables a single encoder to operate across grids of different sizes while embedding system-specific physical constraints and measurement dependencies in a unified latent space.

On top of the GNN embeddings, we deploy a Transformer module [16] to model the temporal signatures of FDIA behavior. Stealthy FDIAs can evolve gradually and exhibit temporal coordination rather than instantaneous anomalies [17], self-

979-8-3315-8138-1

attention provides a flexible mechanism to identify long-range dependencies in the latent state trajectory. To account for the diverse styles of FDIAs encountered across different systems, we utilized a vector-quantized codebook [18] that stores a finite set of reusable FDIA templates. For each input sequence, the GNN embeddings are temporally pooled into a scenario-level summary that is matched to the closest codeword, and this codeword is fused back into each time-step embedding via a lightweight Feature-wise Linear Modulation modulation. The Transformer thus receives embeddings that incorporate both local temporal variation and a global descriptor of the underlying attack pattern, enabling robust discrimination and facilitating efficient adaptation to new grids during fine-tuning.

Overall, our pre-trained FDIA detection framework enables rapid deployment across heterogeneous power grids, substantially reduces reliance on system-specific attack labels, and provides robustness to varying topologies and operating conditions. The remainder of the paper is organized as follows. Section II discusses the FDIA formulation. Section III details the proposed GNN-based encoder, Transformer module, and codebook integration. Section IV presents experimental validations and Section V concludes the paper.

II. System Modeling of FDIA

We consider the AC measurement model $\mathbf{z} = \mathbf{h}(\mathbf{x}) + \mathbf{e}$, where $\mathbf{z}$ is the system measurement, $\mathbf{x}$ is the system state, $\mathbf{h}(\cdot)$ is the nonlinear AC measurement function, and $\mathbf{e} \sim \mathcal{N}(0, \boldsymbol{R})$ is measurement noise. Traditional residual-based BDD compares $\mathbf{z}$ with $\mathbf{h}(\hat{\mathbf{x}})$ obtained from state estimation $\hat{\mathbf{x}} = \arg\min_{\mathbf{x}} (\mathbf{z}-\mathbf{h}(\mathbf{x}))^\top \boldsymbol{R}^{-1} (\mathbf{z}-\mathbf{h}(\mathbf{x}))$ and raises an alarm when the residual exceeds a threshold.

In model-based FDIAs [9], [19], an attacker with knowledge of $\mathbf{h}(\cdot)$ constructs manipulated measurements $\mathbf{z}_a$ as

$$\mathbf{z}_a = \mathbf{z} + \mathbf{h}(\hat{\mathbf{x}} + \mathbf{c}) - \mathbf{h}(\hat{\mathbf{x}}), \tag{1}$$

where $\mathbf{c}$ is an arbitrary perturbation in the state space. This modification shifts $\mathbf{z}$ along a direction *parallel to the measurement manifold*, thereby preserving the estimator residual. Consequently, if the original $\mathbf{z}$ passes the BDD, the corrupted $\mathbf{z}_a$ will also pass. Model-free FDIAs [6]–[8] show that even without explicit access to $\mathbf{h}(\cdot)$, an adversary can learn approximate manifold directions from historical or intercepted data, enabling similarly stealthy, parallel-manifold perturbations.

Across both settings, stealthy FDIAs follow a *universal geometric structure*: attack perturbations $\mathbf{z}_a - \mathbf{z}$ remain on or near the measurement manifold, altering measurements in directions that preserve the estimator residual. This shared structure motivates developing a *pre-trained*, cross-system FDIA detector that can internalize these universal manifold-aligned perturbation patterns and transfer effectively to new grids with minimal fine-tuning.

III. Pre-Trained, Topology-Conditioned FDIA Detection Framework

This section presents the proposed cross-system FDIA detection framework, whose overall architecture is shown in Fig. 1. The model is deployed in two stages: a pre-training stage, where the framework learns universal FDIA representations from multiple power systems and diverse attack scenarios, and a fine-tuning stage, where it adapts to a target grid using only a small number of system-specific samples. Structurally, the framework consists of two major components. The first is a topology-conditioned GNN encoder (Section III-A) that maps measurements and network topology into a unified latent representation and naturally accommodates grids of arbitrary size. The second is a Transformer-based temporal module (Section III-B) enhanced by a vector-quantized FDIA-pattern codebook, which models temporal evolution and captures reusable FDIA templates shared across systems.

A. Topology-Conditioned GNN Encoder

A key requirement for cross-system FDIA detection is an architecture that operates on grids of arbitrary size and topology without structural changes. We note that although the number of buses and branches varies across systems, the *types* of measurements available at each bus or branch are typically drawn from a finite and fixed set (e.g., voltage magnitude, power injection, branch flow), as detailed in Section II. This observation suggests that, by representing each bus and branch using a consistent feature template, the model need only adapt to different graph structures rather than different input dimensions. GNNs naturally satisfy this requirement: they ingest graphs with any number of nodes and edges while sharing parameters across all message-passing operations, making them well suited for heterogeneous power-system topologies.

At each time step, we represent the power grid as a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$. Each bus $i \in \mathcal{V}$ and each branch $(i, j) \in \mathcal{E}$ are described by fixed-dimensional feature vectors:

$$\begin{aligned} \mathbf{n}_i &= \left[V_i,\ \theta_i,\ P_i,\ Q_i\right] \in \mathbb{R}^{F_n}, \\ \mathbf{e}_{ij} &= \left[r_{ij},\ x_{ij},\ b_{ij},\ P_{ij},\ Q_{ij},\ \text{status}_{ij}\right] \in \mathbb{R}^{F_e}, \end{aligned} \tag{2}$$

where $\mathbf{n}_i$ contains nodal measurements (voltage magnitude V_i, angle θ_i, real/reactive injections P_i/Q_i), and $\mathbf{e}_{ij}$ contains branch parameters (resistance r_{ij}, reactance x_{ij}, susceptance b_{ij}), power flows (P_{ij}, Q_{ij}), and a line-status indicator.

For grids where certain measurements are unavailable, we apply an explicit masking scheme. Let $\mathbf{m}_i \in \{0,1\}^{F_n}$ and $\mathbf{m}_{ij} \in \{0,1\}^{F_e}$ denote node-level and edge-level masks indicating feature availability. The masked node and edge features are constructed as

$$\begin{aligned} \tilde{\mathbf{n}}_i &= \mathbf{m}_i \odot \mathbf{n}_i + (1 - \mathbf{m}_i) \odot \mathbf{n}_{\text{fill}}, \\ \tilde{\mathbf{e}}_{ij} &= \mathbf{m}_{ij} \odot \mathbf{e}_{ij} + (1 - \mathbf{m}_{ij}) \odot \mathbf{e}_{\text{fill}}, \end{aligned} \tag{3}$$

where $\odot$ denotes elementwise multiplication, and $\mathbf{n}_{\text{fill}}$ and $\mathbf{e}_{\text{fill}}$ are predefined fill vectors (e.g., zeros or nominal values). This masking mechanism enforces a uniform feature template across heterogeneous grids while preserving information about missing measurements.

To extract system-aware representations, we employ a GNN encoder with D message-passing layers [15]. Let $\mathbf{h}_i^{(0)} = \tilde{\mathbf{n}}_i$ denote the initial embedding for bus i. At each layer, bus embeddings are updated by aggregating information from

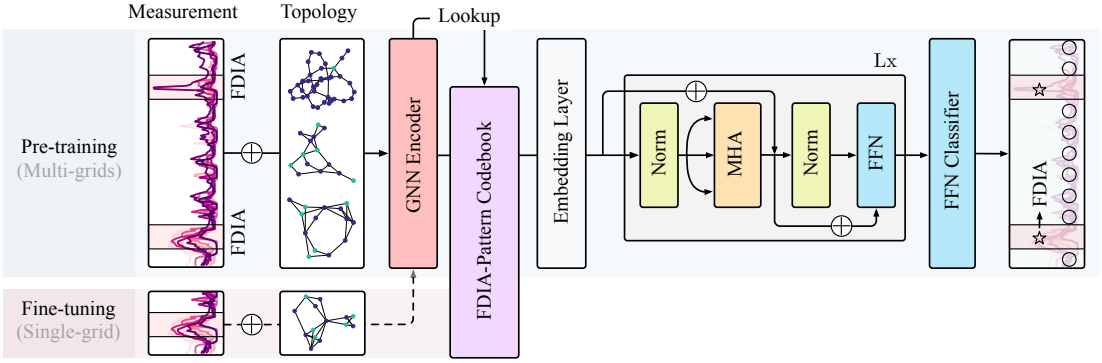

Fig. 1. Overall architecture of the proposed pre-trained FDIA detection framework.

neighboring buses and their associated branch features. A generic message-passing update is given by

$$\mathbf{h}_i^{(d+1)} = \phi_{\text{upd}}\Big(\mathbf{h}_i^{(d)}, \sum_{j\in\mathcal{N}(i)} \phi_{\text{msg}}\big(\mathbf{h}_j^{(d)}, \tilde{\mathbf{e}}_{ij}\big)\Big), \tag{4}$$

where $d = 0, \ldots, D-1$, $\mathcal{N}(i)$ denotes the neighbors of bus i, and ϕ_{msg} and ϕ_{upd} are shared multilayer perceptrons (MLPs). Because the message-passing parameters are shared across all nodes and edges, the encoder naturally scales to grids with arbitrary numbers of buses and branches.

After D message-passing layers, we obtain node embeddings $\{\mathbf{h}_i^{(D)}\}_{i\in\mathcal{V}}$. To further obtain a fixed-size representation for the entire grid at time t, we employ a permutation-invariant mean pooling operation, $\mathbf{h}_t = \text{mean-pool}(\{\mathbf{h}_i^{(D)} : i \in \mathcal{V}\})$. This graph-level embedding $\mathbf{h}_t \in \mathbb{R}^{d_h}$ aggregates system-wide measurement and topology information into a unified latent space, forming the foundation for downstream temporal modeling and FDIA detection.

B. Transformer-Based Temporal Modeling with Codebook-Based FDIA Pattern Conditioning

Given the graph-level embeddings produced by the GNN encoder in Section III-A, a simple MLP classifier would be insufficient: stealthy FDIAs rarely manifest as isolated spikes; instead, they evolve gradually and exhibit coordinated temporal behavior designed to avoid triggering traditional residual-based or rule-based detectors [17]. To capture these temporal dependencies, the embedding sequence $\{\mathbf{h}_t\}_{t=1}^T$ is processed by a Transformer module [16].

However, we note that Transformer-based temporal modeling alone is not sufficient: FDIA behavior can differ markedly across systems and scenarios, so a single temporal decoder may not generalize across heterogeneous attack styles. To model this diversity in a transferable manner, we introduce a vector-quantized codebook $E = [\mathbf{co}_1, \ldots, \mathbf{co}_K] \in \mathbb{R}^{K\times d_c}$ [18], which serves as a compact library of characteristic attack and nominal patterns learned during pre-training.

Before feeding the sequence $\{\mathbf{h}_t\}_{t=1}^T$ into the Transformer blocks, we compute a scenario-level summary $\mathbf{s} = \frac{1}{T}\sum_{t=1}^T \mathbf{h}_t$, and select its nearest codeword,

$$k^* = \arg\min_k \|\mathbf{s} - \mathbf{co}_k\|_2^2, \qquad \mathbf{q} = \mathbf{co}_{k^*}. \tag{5}$$

The codebook is initialized by applying K-means clustering to early batches of scenario embeddings, ensuring that each codeword begins near a frequently occurring FDIA or nominal pattern. We choose the codebook size K within the range $[32, 256]$, and prune unused codewords during training if certain indices receive no assignments. During training, the codebook is then refined using the exponential moving average (EMA) update rule from [18]. This EMA refinement allows each codeword to gradually track the moving average of all scenario embeddings assigned to it, ensuring stable updates and preventing codeword collapse while enabling the codebook to represent recurring FDIA and nominal patterns across heterogeneous grids.

To inject scenario-level information into each time step, we modulate the embeddings using FiLM:

$$\gamma = \text{MLP}_W(\mathbf{q}),\ \beta = \text{MLP}_\beta(\mathbf{q}),\ \hat{\mathbf{h}}_t = \gamma \odot \mathbf{h}_t + \beta, \tag{6}$$

where $\odot$ denotes elementwise multiplication. This operation conditions the entire sequence on the inferred FDIA pattern, enabling the Transformer to jointly exploit local temporal dynamics and global attack-style cues.

Each embedding $\hat{\mathbf{h}}_t$ is then augmented with a positional encoding $\mathbf{p}_t$ [16], $\tilde{\mathbf{h}}_t = \hat{\mathbf{h}}_t + \mathbf{p}_t$, to preserve temporal order. The resulting sequence is passed through a stack of L Transformer decoder blocks (see Fig. 1). For $\ell = 1, \ldots, L$, the ℓ-th Transformer block performs the following pre-normalized residual updates:

$$\begin{aligned} \tilde{\mathbf{u}}_t^\ell &= \mathbf{u}_t^{\ell-1} + \text{MHA}_\ell\big(\text{LN}(\mathbf{u}_{\le t}^{\ell-1})\big), \\ \mathbf{u}_t^\ell &= \tilde{\mathbf{u}}_t^\ell + \text{FFN}_{\tanh,\ell}\big(\text{LN}(\tilde{\mathbf{u}}_t^\ell)\big), \end{aligned} \tag{7}$$

where the initial hidden state are $\mathbf{u}_t^0 = \tilde{\mathbf{h}}_t$, and $\text{MHA}_\ell(\cdot)$ denotes masked (causal) multi-head self-attention, $\text{LN}(\cdot)$ is layer normalization, and $\text{FFN}_{\tanh,\ell}(\cdot)$ is a position-wise feed-forward network with $\tanh$ activation. Causal masking ensures that each time step only attends to its past, preserving the chronological order required for real-time FDIA detection.

To summarize, the proposed framework operates in two stages. In the pre-training stage, the GNN encoder, Transformer, and codebook are jointly trained on multiple heterogeneous systems to learn topology-aware representations, temporal dependencies, and reusable FDIA pattern codes; this stage is performed entirely offline. During deployment, a lightweight fine-tuning stage using abundant normal operating data and a small number of system-specific FDIA samples adapts the model to the target grid's measurement distribution and attack characteristics, enabling rapid and data-efficient transfer to new systems.

IV. Numerical Results

Dataset. A collection of benchmark transmission networks from MATPOWER [20], including the IEEE 14-, 30-, 39-, 57-, 118-, and 300-bus systems, is used to construct the heterogeneous pre-training dataset. For each system, dynamic load trajectories are generated by embedding real residential demand profiles from the Duquesne Light Company (Pittsburgh) dataset into the load buses, followed by random scaling to have operating-point variability [21]. At every time step, AC power flow equations are solved to obtain measurements $\mathbf{z}_t$.

To simulate stealthy FDIAs, we generate temporally evolving attack sequences that remain consistent with the AC measurement manifold. Following Eq. (1), corrupted measurements are constructed as $\mathbf{z}_{a,t} = \mathbf{z}_t + \mathbf{h}(\hat{\mathbf{x}}_t + \mathbf{c}_t) - \mathbf{h}(\hat{\mathbf{x}}_t)$, where $\mathbf{c}_t$ denotes a smoothly varying attack trajectory. Instead of using a fixed perturbation, $\mathbf{c}_t$ is sampled from a family of slowly changing processes (i.e., low-pass–filtered noise, piecewise-linear ramps, sinusoidal drifts), ensuring that attacks evolve gradually and avoid triggering simple residual-based or abrupt-change detectors. Attack duration, target buses, and attack amplitude are randomly generated. For pre-training, we generate a total of 300,000 time steps across all systems, with a nominal-to-attack ratio of 80% nominal and 20% FDIA. This yields a rich dataset with diverse attack patterns for large-scale pre-training and downstream evaluation.

Implementation Details. The GNN encoder uses $D = 4$ message-passing layers, each with hidden dimension $d_h = 128$ and ReLU activations. The Transformer temporal module consists of $L = 6$ decoder blocks with four attention heads and model dimension 128; positional encodings follow the sinusoidal formulation in [16]. The vector-quantized codebook contains $K = 128$ codewords of dimension $d_c = 128$, initialized using K-means clustering and updated using EMA during pre-training. FiLM modulation layers use two-layer MLPs with hidden size 64. The overall model is trained using the Adam optimizer with learning rate 10^{-4}, batch size 64, and dropout rate 0.1. Pre-training is performed for 5,000 epochs on a Google Colab environment with 4 A100 GPUs, after which the model is fine-tuned on each target system using substantially fewer samples (as detailed in Section IV-B).

A. Few-Sample Adaptation: Pre-Training vs. From Scratch

We first demonstrate that a model pre-trained across heterogeneous grids can be adapted to a new system using far fewer labeled FDIA samples than a model trained from scratch. To ensure a fair comparison, the target test system is excluded from the pre-training set. We compare (i) the proposed pre-trained model followed by lightweight fine-tuning, and (ii) an identical architecture trained from scratch using only data from the target system. For each target grid, we vary the number of labeled FDIA samples available for fine-tuning (for the pre-trained model) or full training (for the from-scratch baseline). Both models are evaluated on the same held-out test set.

Fig. 2 reports the detection F1-score as a function of the number of available attack samples. Across all tested networks, the pre-trained model consistently requires far fewer labeled attacks to reach the same accuracy achieved by a from-scratch model. For instance, on the IEEE 118-bus system, the pre-trained model matches the performance of the from-scratch baseline while using only 11.5% of the attack samples. Comparable improvements are observed across systems of varying sizes and topologies. These results confirm that pre-training effectively captures system-invariant FDIA structure, enabling data-efficient transfer to new grids where labeled attack samples are scarce.

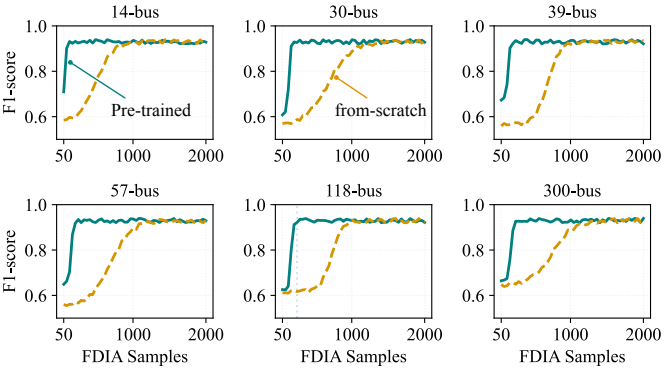

Fig. 2. F1-score versus number of attack samples for the pre-trained and train-from-scratch models.

B. Fine-Tuning Performance Across Systems

We next evaluate the overall FDIA detection performance of the proposed pre-trained model across all six target systems. Each system is held out during pre-training to ensure unbiased transfer evaluation. For every target grid, we fine-tune the pre-trained model and report metrics under two complementary settings, summarized in Table I. (1) *Fixed fine-tuning budget*. In the first setting, each system is fine-tuned using 500 labeled FDIA samples for 500 epochs. We report the resulting detection rate (DR; true positive rate) and false alarm rate (FAR; false positive rate). (2) *Required attack samples for F1 = 0.9*. In the second setting, we measure how many labeled FDIA samples are required for the fine-tuned model to reach a target F1-score of 0.9, while fixing the fine-tuning duration to 500 epochs. This setting directly quantifies data efficiency: systems that traditionally require large labeled attack datasets can now be adapted using significantly fewer samples.

TABLE I
Fine-tuning performance across all test systems.

System	**Fixed 500 Samples**		**Samples for F1 = 0.9**
	DR (%)	FAR (%)	# FDIA Samples
IEEE 14-bus	97.83	2.77	70
IEEE 30-bus	97.15	2.49	90
IEEE 39-bus	96.54	3.15	130
IEEE 57-bus	95.94	3.44	170
IEEE 118-bus	95.23	3.93	230
IEEE 300-bus	94.60	4.24	260

As shown in Table I, across all grids, the pre-trained model achieves high detection rates with low false alarm rates under the fixed 500-sample budget. Furthermore, the number of FDIA samples required to reach an F1-score of 0.9 is consistently small. These results further demonstrate the model's strong cross-system transferability and data efficiency during deployment.

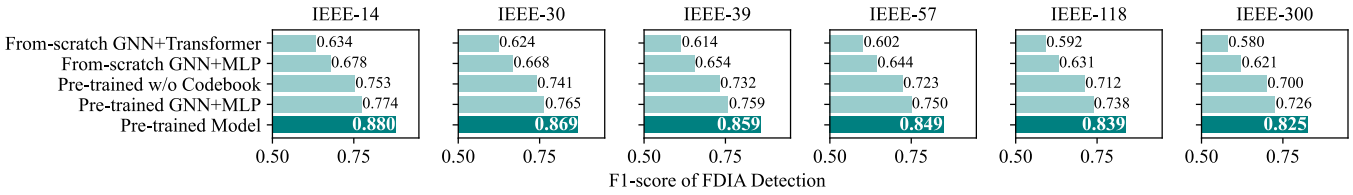

Fig. 3. F1-scores for ablation variants of the proposed FDIA detector across IEEE test systems.

C. Ablation Study

We evaluate the contribution of each component in the proposed framework by comparing the full pre-trained model with four reduced variants. To stress-test robustness in low-label regimes, all variants are fine-tuned or trained using only 100 FDIA samples. (i) Pre-trained GNN+MLP: the Transformer is replaced by a six-layer MLP classifier. (ii) Pre-trained without Codebook: the vector-quantized codebook and FiLM modulation are removed. (iii) GNN+MLP (from scratch): the same GNN+MLP architecture as in (i), but trained entirely from scratch. (iv) GNN+Transformer (from scratch): the same architecture as in (ii), but trained entirely from scratch.

Fig. 3 reports the F1-scores of all ablation variants across the six IEEE test systems. The full pre-trained model achieves the highest F1-score on every grid. Removing either major component leads to a clear performance degradation: replacing the Transformer with an MLP lowers F1-scores by approximately 10–15% on average, while removing the codebook further reduces performance, confirming the importance of both temporal modeling and cross-system pattern conditioning.

The two from-scratch baselines show the largest performance drop, reflecting the difficulty of learning FDIA structure using only 100 labeled attacks. The GNN+MLP variant performs moderately but consistently below its pre-trained counterpart, while the GNN+Transformer baseline suffers the most severe degradation, highlighting that a high-capacity temporal model cannot be reliably trained under such limited supervision. Overall, these results show that both the architectural components (Transformer + codebook) and the multi-grid pre-training stage are essential. Together, they provide strong performance, stability across system sizes, and substantial robustness in low-sample regimes for FDIA detection.

V. Conclusion

This work presented a deployable, cross-system FDIA detection framework that integrates topology-aware GNN encoding, Transformer-based temporal modeling, and codebook-driven FDIA pattern conditioning into a unified pre-trainable architecture. By learning the universal, manifold-aligned structure of stealthy attacks across heterogeneous transmission grids, the framework supports rapid adaptation to new systems with minimal labeled attack data—an essential requirement for practical utility in power-grid operations. Experiments on six IEEE benchmark networks demonstrate consistently superior detection performance over training-from-scratch baselines, delivering high detection rates and low false alarms while requiring an order of magnitude fewer FDIA samples.

References

[1] Bidgely Inc., "Energy disaggregation services," https://www.bidgely.com, accessed: 2025-07-02.

[2] Verdigris Technologies, "Ai-powered energy monitoring," https://verdigris.co, accessed: 2025-07-02.

[3] J. Styczynski and N. Beach-Westmoreland, "When The Lights Went Out: A Comprehensive Review Of The 2015 Attacks On Ukrainian Critical Infrastructure," 2019. [Online]. Available: https://www.boozallen.com/s/insight/thought-leadership/lessons-from-ukranians-energy-grid-cyber-attack.html

[4] S. Tatum, "US accuses Russia of cyberattacks on power grid," CNN, 2018. [Online]. Available: https://www.cnn.com/2018/03/15/politics/dhs-fbi-russia-power-grid/index.html

[5] C. for Strategic and I. Studies, "Significant Cyber Incidents," CSIS, Accesssed June 2024. [Online]. Available: https://www.csis.org/programs/strategic-technologies-program/significant-cyber-incidents

[6] H. Yang, X. He, Z. Wang, R. C. Qiu, and Q. Ai, "Blind false data injection attacks against state estimation based on matrix reconstruction," *IEEE Transactions on Smart Grid*, vol. 13, no. 4, pp. 3174–3187, 2022.

[7] H. Yang and Z. Wang, "A false data injection attack approach without knowledge of system parameters considering measurement noise," *IEEE Internet of Things Journal*, 2023.

[8] M. H. Shahriar, A. A. Khalil, M. A. Rahman, M. H. Manshaei, and D. Chen, "iAttackGen: Generative synthesis of false data injection attacks in cyber-physical systems," in *Conference on Communications and Network Security*. IEEE, 2021, pp. 200–208.

[9] Y. Liu, P. Ning, and M. K. Reiter, "False data injection attacks against state estimation in electric power grids," *ACM Transactions on Information and System Security*, vol. 14, no. 1, pp. 1–33, 2011.

[10] M. Ozay, I. Esnaola, F. T. Y. Vural, S. R. Kulkarni, and H. V. Poor, "Machine learning methods for attack detection in the smart grid," *IEEE transactions on neural networks and learning systems*, vol. 27, no. 8, pp. 1773–1786, 2015.

[11] Y. He, G. J. Mendis, and J. Wei, "Real-time detection of false data injection attacks in smart grid: A deep learning-based intelligent mechanism," *IEEE Transactions on Smart Grid*, vol. 8, no. 5, pp. 2505–2516, 2017.

[12] J. Wang, H. Chen, Y. Si, Y. Zhu, T. Zhu, S. Yin, and B. Liu, "Fdia localization and classification detection in smart grids using multimodal data and deep learning technique," *Computers and Electrical Engineering*, vol. 119, p. 109572, 2024.

[13] M. Mohammadpourfard, C. Xiao, and Y. Weng, "Performance guaranteed deep learning for detection of cyber-attacks in dynamic smart grids," *IEEE Transactions on Power Systems*, pp. 1–12, 2025.

[14] C. Xiao, N. Costilla-Enriquez, and Y. Weng, "Guaranteed false data injection attack without physical model," *IEEE Open Access Journal of Power and Energy*, vol. 12, pp. 429–441, 2025.

[15] J. Gilmer, S. S. Schoenholz, P. F. Riley, O. Vinyals, and G. E. Dahl, "Neural message passing for quantum chemistry," in *International conference on machine learning*. Pmlr, 2017, pp. 1263–1272.

[16] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems*, 2017.

[17] Y. Xu, "A review of cyber security risks of power systems: from static to dynamic false data attacks," *Protection and Control of Modern Power Systems*, vol. 5, no. 1, p. 19, 2020.

[18] A. Van Den Oord, O. Vinyals *et al.*, "Neural discrete representation learning," *Advances in neural information processing systems*, vol. 30, 2017.

[19] G. Hug and J. A. Giampapa, "Vulnerability assessment of ac state estimation with respect to false data injection cyber-attacks," *IEEE Transactions on Smart Grid*, vol. 3, no. 3, pp. 1362–1370, 2012.

[20] R. D. Zimmerman, C. E. Murillo-Sánchez, and D. Gan, "Matpower: A matlab power system simulation package," *Manual, Power Systems Engineering Research Center, Ithaca NY*, vol. 1, pp. 10–7, 1997.

[21] C. Xiao, Y. Liao, and Y. Weng, "Privacy-preserving line outage detection in distribution grids: An efficient approach with uncompromised performance," *IEEE Transactions on Power Systems*, 2024.