Practical Machine Learning Methods for Power-System Operations and Planning

Amir Exir, P.E., M.Eng (EE)

M.S. in Artificial Intelligence Candidate

College of Natural Sciences

The University of Texas at Austin

November 2025

https://www.amirexirpe.com

amir.exir@utexas.edu

Table of Contents

Abstract —The power system electrical grid is having exponential load growth as large flexible loads such as data centers and crypto mining and hydrogen facilities surge. Meanwhile, the grid's infrastructure is aging and system upgrades cannot keep pace with accelerating demand. At the same time, the rise in extreme-weather events have further increased the stress on the grids. The uncertainties in the timing of load growth and system upgrades make forecasting, contingency assessment more complex. At the same time, utilities receive massive amounts of data from SCADA, PMUs, relay logs, PI historians, and network models elements that traditional deterministic frameworks struggle to keep up with this complexity and the flood of system alarms during weather events make alarm analysis significantly more difficult for system operators. In this project, I explored several machine-learning approaches that can support these challenges. The study includes short-term load forecasting using XGBoost on PJM demand data, waveform-based fault classification using SVM, Decision Tree, and Random Forest models, and topology-aware alarm prediction with a graph neural network trained on synthetic cases generated in pandapower. I also developed retrieval-augmented assistants to help engineers navigate ERCOT planning and operational documents more efficiently. Across all four applications, the models showed strong predictive performance and reduced the amount of manual effort required in typical engineering workflows. These results suggest that ML can complement physics-based tools by improving accuracy, speeding up information retrieval, and helping engineers manage the increasing complexity of modern power-system operations.

1. Introduction and research background:

Power system grids need massive digital transformation as its going through rapid changes that make the traditional workflows harder to manage. The widespread deployment of PMUs, smart meters, and remote terminal units has increased both the observability and the complexity of system operations. During emergency events such as Energy Emergency Alerts or hurricanes, the volume of incoming data can overwhelm operators' screens. Traditional deterministic approaches struggle with this level of complexity, especially when dealing with rising load and demand variability, renewable intermittency from solar and wind generation, and the rapid interconnection of very large data center, artificial intelligence, and crypto loads [31]. Recent studies also show that real-time SCADA alarm detection using deep neural networks can assist operators during alarm floods by automatically identifying and classifying alarms into in terms of relevance and importance and retrieving relevant historical events [32].

By using alternative methods, such as machine learning techniques, engineers can extract structure and find patterns from enormous datasets from the field to EMS systems or historian software like PI or Edna, and capture nonlinear dependencies to support real-time operation or system planning decision-making. Looking at prior work in forecasting [16], fault detection [13], and dynamic analysis [9] shows clear benefits, but integrating these techniques into power-system workflows requires careful attention to data quality and accuracy, feature engineering, model interpretability, and validation, and alignment with physics-based tools such as PSS®E and Power World, TARA, and pandapower.

Recently there is an increasing interest in using foundation models and agentic AI and LLMs to merge power system domain knowledge and performing power system analysis, following procedural workflows, and by using power system software to execution analysis in a single decision-making framework. Power Agent and Grid Agent are conceptual multiagent systems described in industry illustrates how LLMs can support tasks such as simulation setup, steady state contingency analysis, and interconnection workflow management [1][32]. This means that the idea is not to replace engineers but to provide structured AI assistants that automate repetitive tasks and increase efficiency and support human decision-making using power system assistants.

On the other hand, utilities, transmission providers, generation developers, independent system operators, and reliability coordinators are now dealing with large flexible loads such as semiconductor plants, AI training clusters, and cryptocurrency facilities that require uninterruptible and highly reliable power. These loads introduce steep ramps and sharp changes in demand, and they often cluster in regions with higher transfer capacity. This creates additional uncertainty because traditional models do not capture all of the detailed behavior of these loads under different contingencies [2], [21]–[28]. In Texas, the magnitude of this challenge is especially visible in ERCOT's most recent Long-Term Load Forecast (LTLF). The LTLF is an hourly forecast for the next ten years based on economic projections and historical weather from 2008–2022, and it incorporates Transmission Service Provider (TSP) submitted large-load additions such as data centers, crypto facilities, hydrogen production, and industrial electrification. ERCOT's adjusted forecast shows peak demand rising sharply from historical levels in the 70–85 GW range to more than 120–145 GW by 2031, with data centers representing the largest share of new growth. Several of the 2025–2031 scenario cases also indicate that hourly system load may increase faster than transmission expansion timelines. These projections highlight why forecasting, anomaly detection, and topology-aware alarm prediction are becoming essential traditional deterministic tools struggle when load growth is both steep and dominated by nontraditional, flat 24/7 "tech" loads. Figure 1 to 4 show ERCOT and TSPs Load Breakdown and Forecasts [37].

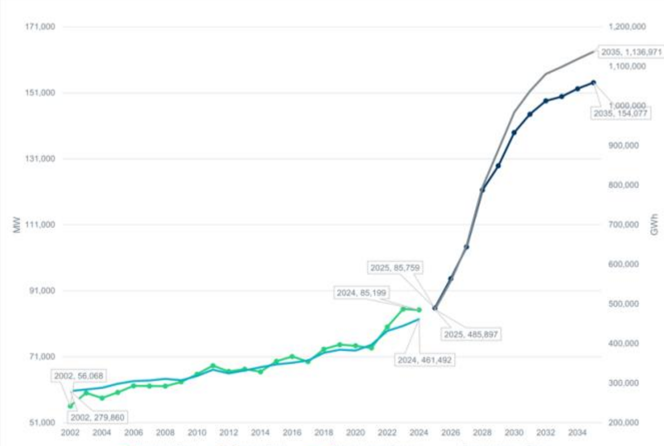
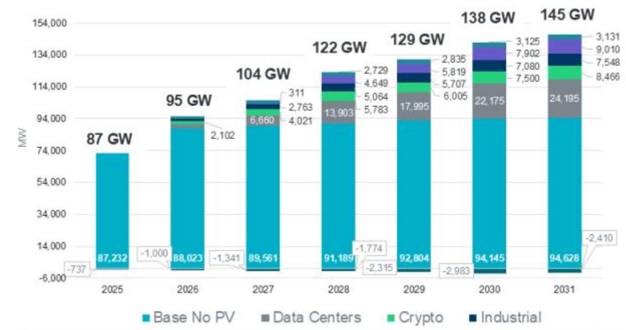


Figure 1: ERCOT Adjusted Load Forecasts [37].



	2025	2026	2027	2028	2029	2030	2031
Data Centers	0	2,433	6,660	13903.35	17,995	22,175	24,195
Crypto	0	2,102	4,021	5782.964	6,005	7,500	8,466
Industrial	0	1,387	2,763	5064.178	5,707	7,080	7,548
Hydrogen	0	5	311	4649.1	5,819	7,902	9,010
Oil and Gas	0	1,699	2,320	2728.529	2,835	3,125	3,131
PV	-737	-1,000	-1,341	-1773.73	-2,315	-2,983	-2,410

Figure 2: ERCOT Adjusted Load Breakdown [37].

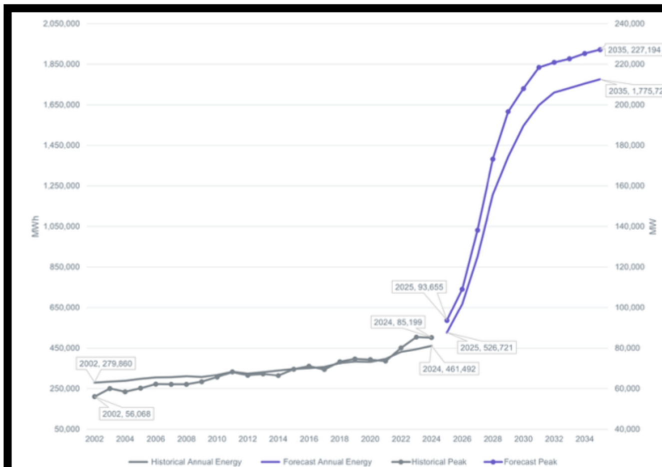
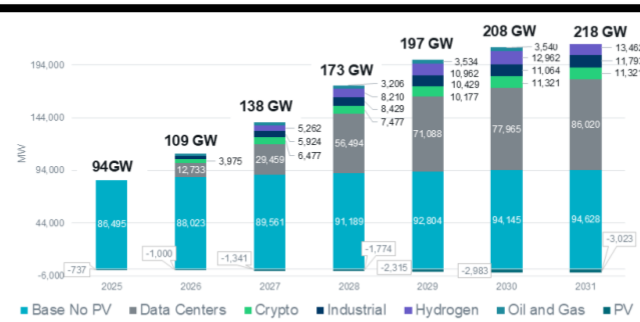


Figure 3: TSP Large Load Forecast [37].



	2025	2026	2027	2028	2029	2030	2031
Data Centers	3,717	12,733	29,459	56,494	71,088	77,965	86,020
Crypto	1,868	3,975	6,477	7,477	10,177	11,321	11,321
Industrial	973	2,757	5,924	8,429	10,429	11,064	11,793
Hydrogen	5	357	5,262	8,210	10,962	12,962	13,462
Oil and Gas	1,334	2,186	2,851	3,206	3,534	3,540	3,606
PV	-737	-1,000	-1,341	-1,774	-2,315	-2,983	-3,023

Figure 4: TSP Large Load Breakdown [37].

2. Research and methods

The four machine-learning applications explored in this project are related to power-system operation and planning. Each method reflects a different class of ML workflow, ranging from time-series prediction to pattern classification, graph-based learning, and retrieval-augmented language-model pipelines.

2.1 Short-Term Load Forecasting

To forecast future short–term load of AEP/PJM based on historical data, I used gradient-boosted decision trees to model short-term variations in hourly electricity demand. The method relies on engineered temporal features such as lagged load values, calendar variables, and rolling averages. XGBoost was selected because it handles nonlinear patterns, missing data, and mixed feature types efficiently. The model was trained using a standard 80/20 train-test split, with hyperparameters tuned using grid search. During tuning, I ran a 3-fold cross-validation grid search over a range of depth, learning-rate, subsample, and tree-count values. The best combination was max_depth = 7, learning_rate = 0.1, n_estimators = 300, subsample = 0.8, and colsample_bytree = 1.0. These settings were then used to retrain the model on the full training split for the final evaluation.

2.2 Fault Classification Using Three-Phase Waveform Features

To classify electrical faults, each three-phase waveform was transformed into a set of statistical and sequence-component features. This included RMS values, peak magnitudes, zero- and negative-sequence quantities, and simple harmonic descriptors. Three supervised algorithms were tested: Support Vector Machine (SVM), Decision Tree, and Random Forest. The dataset was split into training and testing subsets, and model performance was evaluated using accuracy and F1-score.

2.3 Graph Neural Network for Voltage and Thermal Alarm Prediction

To model topology-aware alarm prediction, I developed a graph-based learning pipeline in which each simulated power-flow scenario is converted into a graph representation using pandapower and PyTorch Geometric. For voltage-related predictions, buses are treated as nodes and transmission lines as edges, with node features including voltage magnitude, real and reactive injections, loading indicators, and neighborhood connectivity. For thermal-overload prediction, a line-graph construction is used: each transmission line becomes a node, and edges connect electrically adjacent lines, allowing the model to learn spatial and electrical relationships relevant to overload propagation.

Four GNN architectures were implemented to compare different message-passing strategies: Graph Convolutional Network (GCN) [10], Graph Attention Network (GAT) [11], Graph Isomorphism Network (GIN), and a Graph Transformer. All models were trained with the Adam optimizer, ReLU activation, and minibatch training across multiple simulated contingency scenarios. Voltage labels were assigned using discretized |V| ranges, and thermal labels were defined using four load-percentage classes. This design allows the models to learn topology-dependent alarm patterns directly from the network structure and scenario-dependent electrical conditions, consistent with recent work on GNN-based operational risk assessment in power grids [5]–[7], [19].

2.4 Retrieval-Augmented Assistants for ERCOT Documentation

To support engineers navigating ERCOT's extensive technical documentation, I developed a retrieval-augmented generation (RAG) workflow. The approach uses similarity-based retrieval to surface relevant text from ERCOT Planning Guides, Nodal

Protocols, Dynamic and Steady-State Working Group (DWG & SSWG) manuals, and Resource Integration materials, and then synthesizes concise answers using a language model grounded in the retrieved passages. This provides a structured way to query large technical documents and serves as the foundation for the specialized assistants described in the results.

3. Materials and Data Sources

For this project I used four main data sources:

3.1 PJM Hourly Load Dataset

Hourly electricity-consumption data from PJM (2004–2018) were obtained from Kaggle [29]. The dataset includes multiple load zones and is widely used for short-term forecasting studies. Before modeling, several preprocessing steps were required to ensure data consistency and suitability. Missing values were filled using simple interpolation, and clear outliers were removed to prevent distortion of the training process. The time series was resampled into hourly intervals and enriched with engineered temporal features such as lagged load values (t–1, t–24, and t–168) to capture short-term and weekly demand patterns. Additional calendar-based variables (including hour of day, day of week, and a weekday/weekend indicator) were added to reflect routine behavioral cycles in electricity usage. When temperature data were available, they were merged with the load time series, as weather is a strong driver of demand across the PJM region.

3.2 Fault Classification Dataset

Synthetic three-phase voltage and current waveforms representing several fault types (single-line-to-ground (LG), line-to-line (LL), double-line-to-ground (LLG), and three-phase faults (LLL)) were obtained from the Electrical Fault Detection and Classification dataset on Kaggle [30]. Each sample contains six raw time-series signals (Va, Vb, Vc, Ia, Ib, Ic which are the voltage and currents in each phase) generated under controlled fault scenarios.

In order to prepare the data for supervised learning, the raw waveforms were first normalized to place voltage and current magnitudes on consistent scales. Each waveform was then converted into a structured feature vector using statistically and physically meaningful descriptors. These included peak and RMS magnitudes, minimum and maximum values, and basic harmonic or shape-based attributes. Symmetrical components (particularly zero-sequence and negative-sequence values) were also computed to capture phase imbalance patterns that are characteristic of different fault types.

After feature extraction, all samples were labeled according to their corresponding fault type, shuffled, and split into training and testing subsets. This preprocessing transformed the raw waveform data into compact numerical representations suitable for models such as SVM, Decision Tree, and Random Forest, enabling accurate fault-type classification.

3.3 Synthetic Grid-Simulation Data for GNN Training

I trained the GNN models by a large collection of synthetic contingency scenarios that I generated using pandapower on a test system containing 118 buses and 346 transmission lines. Each scenario consisted of a solved AC power flow with randomized load perturbations, generator dispatch variations, and a full set of N-1 line-outage events. For every solved case I assigned two sets of labels: bus-level voltage-alarm classes derived from discretized voltage-magnitude ranges, and line-level thermal-loading categories based on the percentage of rated current.

To support topology-aware learning, the data were represented in two complementary graph formats. The first was a bus graph in which buses served as nodes connected by their incident transmission lines. The second was a line graph in which each transmission line was treated as a node connected to other lines that shared a terminal bus. All node features were taken directly from the power-flow outputs, including voltage magnitudes, real and reactive injections, line loadings, and structural connectivity information. This dataset enabled consistent training and evaluation of multiple GNN architectures for both voltage-alarm and thermal-overload prediction tasks.

3.4 ERCOT Technical Documents for RAG Evaluation

The RAG systems were built using publicly available ERCOT documents including the Planning Guide, Nodal Protocols, DWG and SSWG procedure manuals, and Resource Integration guidelines. All files were processed into paragraph-level text and then split into chunks with slight overlaps to prevent loss of cross-section references. These chunks were embedded with OpenAI's text-embedding-3-large model, producing dense vector representations stored in a FAISS similarity-search index. This dataset enabled evaluation of retrieval quality and supported the deployment of multiple domain-specific assistants.

4. Results

4.1 Short-Term Load Forecasting

Load Visualization

The time-series plot in figure 5 shows the raw hourly AEP load data for 2017–2018. This visualization highlights the daily and seasonal structure of electricity demand, including winter peaks, summer peaks, and intraday variability. It also demonstrates the need for feature engineering such as lagged variables and calendar effects.

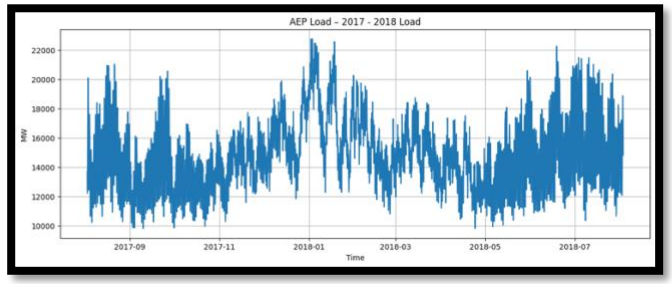


Figure 5: Raw hourly AEP load data for 2017–2018.

Hourly Load Pattern

The boxplot in Figure 6 shows how the load changes over a typical day. The blue boxes represent the middle range of values for each hour, while the circles above them are unusually high days (mostly hot summer afternoons where demand spikes). Early morning hours have the lowest and most stable loads. As the day progresses, demand builds, and the spread gets wider. The late-afternoon period (around 3 to 8 PM) has the highest values and also the most outliers. Seeing this pattern makes it clear why features like hour-of-day and lagged values are useful for the forecasting model.

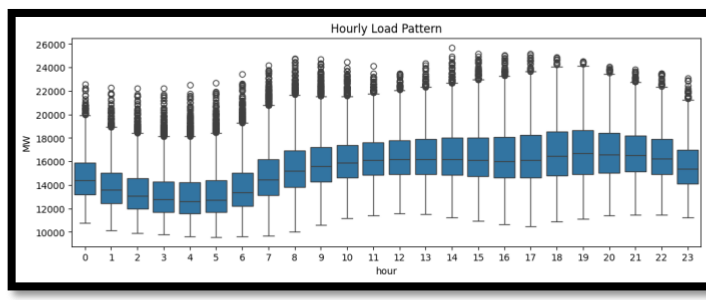


Figure 6: Load changes over a typical day.

Weekday vs. Weekend Load Patterns

To test the intuition that people use electricity differently on weekdays and weekends, I looked at the daily structure of electricity demand by comparing the average hourly load for the two groups. Figure 7 shows a clear difference between the patterns. On weekdays, the load ramps up sharply after about 6 AM as commercial and industrial activity begins, reaching a broad afternoon and early-evening peak. Weekends follow a flatter shape, with lower early-morning demand and a smaller peak because most workplace and school activity is absent. Across the dataset, the average weekday load was 15,936.69 MW, compared with 14,405.99 MW on weekends. These differences show why calendar features such as hour-of-day and weekend indicators help the XGBoost model capture routine behavioral patterns in electricity demand.

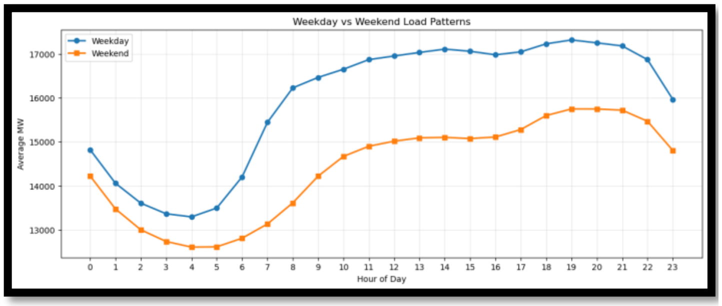


Figure 7: Weekday vs. Weekend Load Patterns

Model Prediction Visualization

The following plot in figure 8 compares XGBoost predictions with the actual load values for the first 500 hours of the test set. The model captures the daily and weekly load patterns closely, including peaks, valleys, and weather-driven variations. This visual alignment reinforces the strong quantitative performance observed in the RMSE and R^2 metrics.

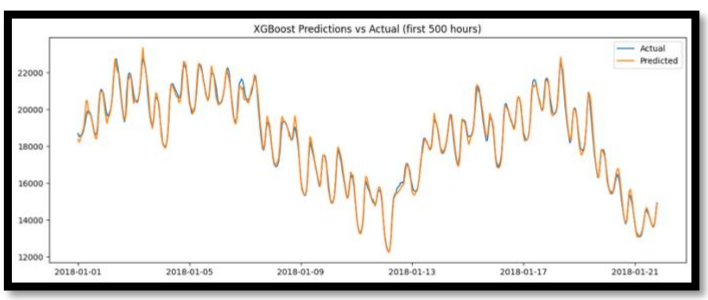


Figure 8: XGBoost predictions with the actual load values

Feature Importance Analysis

To know what are the most critical criterial in order to find out how the model makes predictions, we can look at figure 9 that shows the feature-importance rankings from the trained XGBoost model. The most influential predictors include the hour of the day, month, and short-term lagged load values (t–1 and t–24). Rolling averages (24h and 168h) also contribute substantially. These importance scores confirm that load forecasting is driven primarily by strong temporal and seasonal patterns, which XGBoost captures effectively.

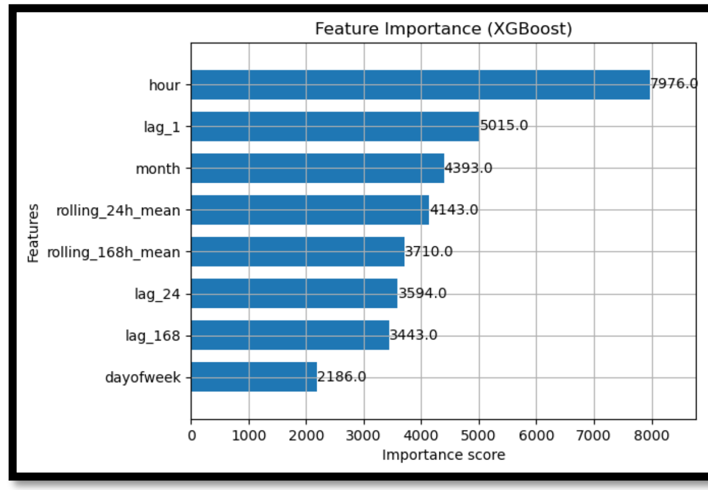


Figure 9: The feature-importance rankings from the trained XGBoost model.

24-Hour Ahead Forecast Demonstration

To illustrate how the trained model performs in an operational setting, a 24-hour ahead forecast in figure 10 was generated using the most recent historical data. The XGBoost model accurately tracks the daily load cycle, capturing both the morning ramp and the evening peak with realistic timing and amplitude. This strong visual alignment is reflected in the quantitative results: the model achieved an R^2 of approximately 0.9951 and an RMSE of 176.95 MW, indicating that it explains nearly all the variance in hourly demand while maintaining a relatively small prediction error. These results are consistent with findings in the literature showing that gradient-boosted tree models often outperform neural networks on structured electricity-demand datasets [16]–[18], and they demonstrate that XGBoost is well suited for practical next-day operational forecasting or as an input to downstream reliability assessments.

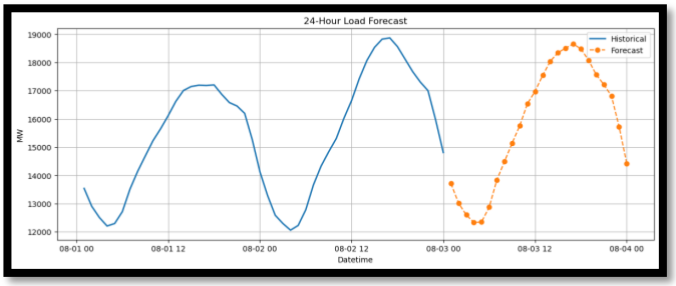


Figure 10: a 24-hour ahead forecast

Practical Interpretation

These forecasts offer a transparent balance of accuracy and interpretability. With the rapid growth of large loads, such models help planners evaluate impacts on congestion, capacity planning, and local-area security.

4.2 Fault Classification Using Three-Phase Measurements

The fault-classification task was performed using synthetic three-phase voltage and current waveforms representing several standard fault types, including single-line-to-ground (LG), line-to-line (LL), double line-to-ground (LLG), and three-phase faults (LLL). Each waveform was converted into statistical and sequence-component features such as peak and RMS magnitude, negative- and zero-sequence values, phase imbalance, and basic harmonic descriptors. Three supervised models (SVM, Decision Tree, and Random Forest) were tested to compare how well different algorithms handle these extracted features.

Figure 11 shows the confusion matrix for the best-performing model, Random Forest. The model achieved an accuracy of 0.93 and an F1-score of 0.93, and the confusion matrix confirms that most misclassifications occur between fault types with similar electrical signatures. These results are consistent with prior studies showing that ensemble methods tend to perform well for waveform-based condition-monitoring tasks [13].

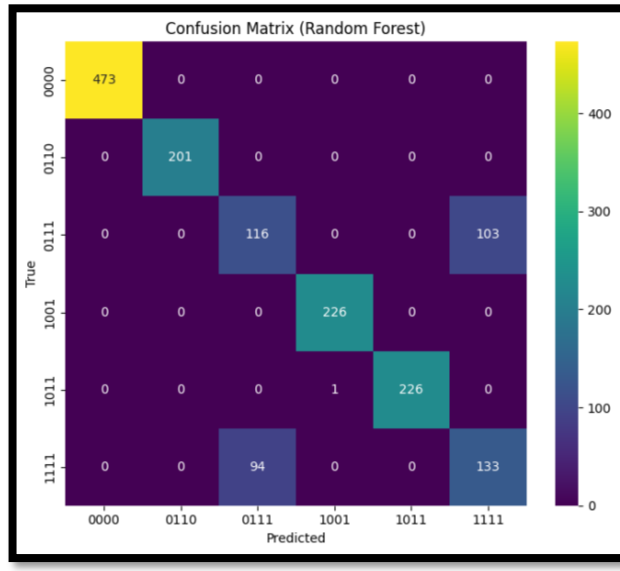


Figure 11: The confusion matrix for the best-performing model

4.3 Graph Neural Network (GNN) for Alarm Prediction

The alarm-prediction task evaluated the ability of several graph-based neural network architectures to detect voltage and thermal issues directly from the network topology. Four models were tested: a Graph Convolutional Network (GCN) [10], a Graph Attention Network (GAT) [11], a Graph Isomorphism Network (GIN) [38], and a Transformer-based GNN that applies global attention for message passing. All models were trained using the Adam optimizer with ReLU activations and identical train–test splits to ensure a fair comparison.

For the voltage-alarm classification problem, the Transformer model achieved the highest performance, reaching an accuracy of 94.53% and an F1-score of 96.48%. The GIN and GAT architectures also performed well, while the baseline GCN showed lower accuracy. These results align with recent findings showing that attention-based GNNs capture long-range electrical dependencies more effectively than convolution-based approaches when assessing grid operating conditions [5]–[7], [19].

A similar trend was observed in the thermal-loading task, which classified each transmission line into one of four loading categories. The Transformer-based GNN again performed best, achieving 96.51% accuracy and an 89.52% macro-F1 score. Both GAT and GIN delivered competitive results, while the GCN model struggled more with the multi-class nature of thermal-limit prediction. This is consistent with the fact that thermal overloads depend on both local and nonlocal interactions across the network, which attention-based architectures capture more effectively.

Overall, the results demonstrate that topology-aware learning significantly improves the prediction of voltage and thermal alarms compared with traditional linear baselines. The strong performance of the Transformer architecture suggests that global attention mechanisms provide a powerful way to model evolving grid topologies and identify operational risk patterns under contingency conditions.

Figure 12 illustrates the training behavior of the Transformer-based GNN for voltage-alarm classification. The loss curves show smooth convergence for both training and validation sets, while the validation accuracy stabilizes above 94%. The F1-score approaches 0.96 and Macro F1 remains consistent across epochs. This behavior confirms stable optimization and strong generalization, matching the numerical performance reported in Tables I and II.

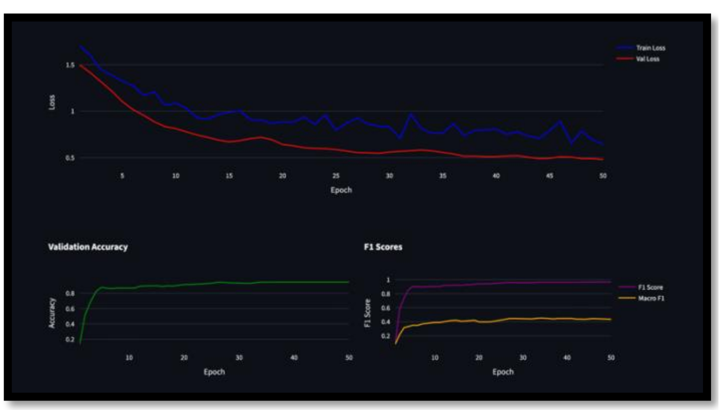


Figure 12: Transformer-based GNN for voltage-alarm classification.

Model	Accuracy	Precision	Recall	F1 Score
GCN	85.42%	92.60%	85.42%	88.73%
GAT	83.91%	92.44%	83.91%	87.81%
GIN	87.05%	93.68%	87.05%	90.01%
Transformer	**94.53%**	**98.66%**	**94.53%**	**96.48%**

Table I: Voltage-Alarm Classification Performance

Model	Accuracy	Precision	Recall	F1 Score
GCN	72.04%	84.14%	72.04%	76.73%
GAT	76.34%	89.05%	76.34%	80.01%
GIN	78.44%	90.10%	78.44%	82.02%
Transformer	**96.51%**	**97.87%**	**96.51%**	**96.91%**

Table II: Thermal-Alarm Classification Performance

4.4 RAG-Based ERCOT AI Assistants

ERCOT's technical documents span hundreds of pages across multiple manuals, so I created four specialized RAG-based assistants targeting the Planning Guide, DWG/SSWG procedures, Resource Integration materials, and Nodal Protocols. Each assistant allows users to ask natural-language questions and returns grounded answers linked to the most relevant sections of the documents. In practice, these tools greatly reduce the time engineers spend searching for modeling rules, PMCR and DCP requirements, and validation procedures, providing a faster and more intuitive way to interact with ERCOT's regulatory material.

5. Conclusion

Table III summarizes the performance of the four machine-learning applications explored in this study. XGBoost delivered the strongest results on structured load-forecasting data, Random Forest achieved high accuracy on waveform-based fault classification, the GNN captured topology-dependent alarm patterns more effectively than linear baselines, and the RAG system significantly reduced search time for ERCOT documentation. These findings show that different ML techniques are suited to different operational and planning tasks and can complement the traditional physics-based tools used in power-system engineering.

Across all four applications, ML proved useful but not standalone. Structured models like XGBoost perform extremely well for short-term load forecasting; fault classification benefits from physically meaningful features such as sequence components; GNNs leverage network connectivity to improve alarm prediction; and RAG assistants reduce the manual workload involved in navigating ERCOT documentation. These methods cannot replace engineering judgment or physics-based simulations, but they can meaningfully enhance forecasting, alarm detection, and information retrieval as grid complexity increases.

Looking ahead, foundation models and agentic workflows may support future EMS/SCADA integration, automated model validation, and adaptive planning. Ensuring that these tools remain safe, interpretable, and aligned with operational needs will require continued collaboration between ML researchers and power-system engineers.

Task	**Technique**	**Metric**	**F1**	**Key Observation**
Load Forecasting	XGBoost	R^2 = 0.995, RMSE = 176 MW	—	Captures seasonal and daily load patterns
Fault Classification	Random Forest	Accuracy = 0.93	0.93	Performs well with waveform-based features
Voltage Alarm Prediction	Transformer-GNN	Accuracy = 94.53 %	96.48 %	Strongest performer; learns voltage-dependent topology patterns
Thermal Alarm Prediction	Transformer-GNN	Accuracy = 96.51 %	96.91 %	Excels on multi-class thermal overload prediction across 346 lines
ERCOT Document Search	RAG (Embedding+GPT)	-	-	Helps surface relevant ERCOT rules and reduces manual PDF searching

Table III: Summary of Models Performance

References

[1] Q. Zhang and L. Xie, “PowerAgent: A Road Map Toward Agentic Intelligence in Power Systems: Foundation Model, Model Context Protocol, and Workflow,” IEEE Power & Energy Magazine, vol. 23, no. 5, pp. 93–101, Sep.–Oct. 2025, doi: 10.1109/MPE.2025.3579718.

[2] I. Kamwa, “Energy-Hungry Tech: Large Flexible Loads in the Age of AI and Bitcoin,” IEEE Power & Energy Magazine, vol. 23, no. 5, pp. 4–9, Sep.–Oct. 2025, doi: 10.1109/MPE.2025.3584184.

[3] J. V. Milanović, “PES Publications: Challenges, Opportunities, and Progress,” IEEE Power & Energy Magazine, vol. 23, no. 5, pp. 20–23, Sep.–Oct. 2025, doi: 10.1109/MPE.2025.3584185.

[4] S. Zhang et al., “Grid Modernization in the Age of Artificial Intelligence and Cloud: Synergies Between Grid Modernization, Artificial Intelligence, and Cloud,” IEEE Power & Energy Magazine, vol. 23, no. 5, pp. 102–116, Sep.–Oct. 2025, doi: 10.1109/MPE.2025.3565467.

[5] M. Dezvarei, K. Tomsovic, J. S. Sun, and S. M. Djouadi, “Graph Neural Network Framework for Security Assessment Informed by Topological Measures,” arXiv:2301.12988 [cs.LG], 2023.

[6] C. Kim, K. Kim, P. Balaprakash, and M. Anitescu, “Graph Convolutional Neural Networks for Optimal Load Shedding under Line Contingency,” 2019 IEEE Power & Energy Society General Meeting (PESGM), Atlanta, GA, USA, 2019, pp. 1–5, doi: 10.1109/PESGM40551.2019.8973468.

[7] Y. Zhang, P. M. Karve, and S. Mahadevan, “Graph Neural Networks for Power Grid Operational Risk Assessment Under Evolving Grid Topology,” arXiv:2405.07343 [cs.LG], 2024.

[8] L. Thurner et al., “Pandapower: An Open-Source Python Tool for Convenient Modeling, Analysis, and Optimization of Electric Power Systems,” IEEE Trans. Power Syst., vol. 33, no. 6, pp. 6510–6521, 2018, doi: 10.1109/TPWRS.2018.2829021.

[9] M. Sun, N. Zhang, and Z. Y. Dong, “Dynamic Security Assessment for Power Systems Using Deep Learning,” IEEE Trans. Power Syst., vol. 36, no. 4, pp. 3270–3280, 2021, doi: 10.1109/TPWRS.2020.3043209.

[10] T. N. Kipf and M. Welling, “Semi-Supervised Classification With Graph Convolutional Networks,” in Proc. ICLR, 2017.

[11] P. Veličković et al., “Graph Attention Networks,” in Proc. ICLR, 2018.

[12] A. Mosavi, M. Salimi, and M. M. Ardehali, “Domain Knowledge-Based Machine Learning Models for Power-System Operation and Planning,” Energies, vol. 13, no. 15, p. 3862, 2020, doi: 10.3390/en13153862.

[13] H. Khorasgani, A. K. S. Jardine, and D. Banjevic, “Condition-Based Maintenance for Power-System Components Using Machine-Learning Approaches,” IEEE Trans. Power Syst., vol. 32, no. 5, pp. 3907–3916, 2017, doi: 10.1109/TPWRS.2016.2638899.

[14] Y. Zhang, X. Li, and L. Wang, “Data-Driven Model Reduction for Power System Dynamic Studies,” IEEE Trans. Power Syst., vol. 37, no. 4, pp. 3088–3099, 2022, doi: 10.1109/TPWRS.2021.3126245.

[15] S. J. Pan and Q. Yang, “A Survey on Transfer Learning,” IEEE Trans. Knowl. Data Eng., vol. 22, no. 10, pp. 1345–1359, 2010, doi: 10.1109/TKDE.2009.191.

[16] T. Chen and C. Guestrin, “XGBoost: A Scalable Tree Boosting System,” in Proc. 22nd ACM SIGKDD Conf. Knowledge Discovery and Data Mining, 2016, pp. 785–794, doi: 10.1145/2939672.2939785.

[17] G. Dudek, “Pattern-Based Forecasting of Electricity Demand,” Applied Energy, vol. 162, pp. 190–199, 2016, doi: 10.1016/j.apenergy.2015.10.071.

[18] S. Fan, R. J. Hyndman, and K. Zhou, “Probabilistic Forecasting of Electricity Demand: A Review,” Int. J.

Forecasting, vol. 37, no. 4, pp. 1528–1550, 2021, doi: 10.1016/j.ijforecast.2021.01.004.

[19] J. Zhou et al., “Graph Neural Networks: A Review of Methods and Applications,” AI Open, vol. 1, pp. 57–81, 2020, doi: 10.1016/j.aiopen.2020.01.001.

[20] H. F. Hamann et al., “Foundation Models for the Electric Power Grid,” Joule, vol. 8, no. 12, pp. 3245–3258, 2024, doi: 10.1016/j.joule.2024.11.002.

[21] Flexible Load and the Electricity Grid: A Bitcoin Story, IEEE Power & Energy Magazine, 2024.

[22] M. Sahni, W. Bojorquez, J. Nolan, C. Boyer, and D. Vedullapalli, “Buckle Up, Texas: A Large Load Tsunami in the Lone Star State,” IEEE Power and Energy Magazine, vol. 23, no. 5, pp. 68–81, Sept.–Oct. 2025, doi: 10.1109/MPE.2025.3571028.

[23] Texas Loads Ride Toward Grid Stability: Voltage Ride Through of Large Power Electronic Loads, IEEE Power & Energy Magazine, 2024.

[24] Resource Adequacy and Forecasting: An ISO Perspective on Challenges Associated With Large Flexible Loads, IEEE Power & Energy Magazine, 2024.

[25] Emerging Loads Modeling for Transmission Reliability Studies, IEEE Power & Energy Magazine, 2024.

[26] Economic Impacts of Price-Sensitive Loads in ERCOT: Quantifying Effects of Large Flexible Loads, IEEE Power & Energy Magazine, 2024.

[27] Future Industrial Demand: Large, Elastic, and Concentrated Guest Editorial, IEEE Power & Energy Magazine, 2024.

[28] Reliability Implications Integrating Large Loads Into Bulk Power: Hot Topic, IEEE Power & Energy Magazine, 2024.

[29] PJM Interconnection, “Hourly Energy Consumption Dataset,” Kaggle, 2020. [Online]. Available: https://www.kaggle.com/datasets/robikscube/hourly-energy-consumption

[30] E. Sathya Prakash, “Electrical Fault Detection and Classification Dataset,” Kaggle, 2019. [Online]. Available: https://www.kaggle.com/datasets/esathyaprakash/electrical-fault-detection-and-classification

[31] M. Sahni, W. Bojorquez, J. Nolan, C. Boyer and D. Vedullapalli, "Buckle Up, Texas: A Large Load Tsunami in the Lone Star State," in IEEE Power and Energy Magazine, vol. 23, no. 5, pp. 68-81, Sept.-Oct. 2025, doi: 10.1109/MPE.2025.3571028.

[32] C. Liu, G. Dalzell, E. Kranz, X. Yu, A. Kelly, and M. Almashor, “Real-Time Machine Learning for Power Grid SCADA Alarm Event Detection Decision Support,” in Proceedings of IECON 2024 – 50th Annual Conference of the IEEE Industrial Electronics Society, Chicago, IL, USA, 2024, pp. 1–6, doi: 10.1109/IECON55916.2024.10905542.

[33] X. Hua, R. Chaoyu, G. Zhixing et al., “Stochastic load flow calculation method based on clustering and sampling,” Transactions of China Electrotechnical Society, vol. 35, no. 23, pp. 4940–4948, 2020.

[34] X. Hu, Z.-W. Liu, G. Wen, X. Yu, and C. Liu, “Voltage control for distribution networks via coordinated regulation of active and reactive power of dgs,” IEEE Transactions on Smart Grid, vol. 11, no. 5, pp. 4017–4031, 2020.

[35] C. Balducelli, L. Lavalle, and G. Vicoli, “Novelty detection and management to safeguard information-intensive critical infrastructures,” International Journal of Emergency Management, vol. 4, no. 1, pp. 88– 103, 2007.

[36] 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 (TISSEC), vol. 14, no. 1, pp. 1–33, 2011.

[37] ERCOT, “Long-Term Load Forecast,” Electric Reliability Council of Texas, 2025. Accessed: Nov. 26,

2025. [Online]. Available: https://www.ercot.com/gridinfo/load/forecast

[38] K. Xu, W. Hu, J. Leskovec, and S. Jegelka, "How Powerful Are Graph Neural Networks?" Proc. ICLR, 2019.

Appendix A – Code and Application Links

The following applications and demonstration tools were developed by Amir Exir to support and visualize the methods presented in this study:

ERCOT All-in-One AI Assistant: https://amirexir-por-chatbot-ercot-all-in-oneercot-assistant-app-ahgre0.streamlit.app/

ERCOT DWG & SSWG Assistant: https://portfolio-m4bbup6umxumtfibeqkrrz.streamlit.app/

ERCOT Planning Guides Assistant: https://portfolio-evxbz642ppnqfsen8txlic.streamlit.app/

ERCOT Resource Integration Assistant: https://portfolio-ewyxjs3snfqvos8pq2ior9.streamlit.app/

ERCOT Nodal Protocols Assistant: https://portfolio-mkfjrirugw3bjzy6jfrjx4.streamlit.app/

ERCOT Load Dashboard: https://portfolio-w5bmjqktrj969skqfxavxm.streamlit.app/

PSS/E API Automation Assistant: https://portfolio-8yebxmxquvaavwwg5va3tp.streamlit.app/

PSS/E Multi-Agent Automation Bot: https://portfolio-4co3lvfwtpzsl3ivbuhvou.streamlit.app/

Fault Classifier App: https://portfolio-xmc8hpyiryrkj8acggxe6k.streamlit.app/

Hourly Load Forecast App (AEP / PJM): https://portfolio-yq4typvbncex55yh5cfdmr.streamlit.app/

Power Grid GNN Predictor: https://ai-in-power-system-electrical-engineering-hw6ktvbujtbw5zygqxxj.streamlit.app/