De Bruijn Graph-Enhanced Time Series Models for Electricity Load Forecasting

Mert Onur Cakiroglu[1], Idil Bilge Altun[1], Shahriar Rahman Fahim[2,*], Hasan Kurban[3],
Mehmet M. Dalkilic[1], Rachad Atat[4], Abdulrahman Takiddin[5], Erchin Serpedin[2], Khalid Qaraqe[3]
[1]Department of Computer Science, Indiana University, Bloomington, IN 47405, USA;
[2]Electrical & Computer Engineering Department, Texas A&M University, College Station, TX 77843, USA;
[3]College of Science and Engineering, Hamad Bin Khalifa University, Doha, Qatar;
[4]Department of Computer Science & Mathematics, Lebanese American University, Beirut, Lebanon;
[5]Department of Electrical & Computer Engineering, Florida State University, Tallahassee, FL 32310, USA;
*Corresponding author email: sr-fahim@tamu.edu

***Abstract*—Effective electricity load prediction enables grid operators to design optimal generation schedules and energy dispatch strategies that minimize risks from demand fluctuations. However, predicting highly volatile loads in large-scale power grids remains a complex challenge due to the dynamic nature of individual bus-level consumption patterns. Traditional forecasting methods primarily focus on temporal trends and often overlook interdependencies between influencing factors, which limits their ability to capture load variations. To address these challenges, this study introduces a methodology that integrates de Bruijn Graphs (dBGs) with advanced time-series forecasting models. By leveraging structural properties, the framework enhances the modeling of sequential dependencies within power grid data. Advanced graph encoding techniques extract salient features from dBGs and help identify overlooked patterns. This study develops four enhanced architectures—FiLMdBG, iTransformerdBG, TimesNetdBG, and DLineardBG—evaluated on the Texas 2,000-bus test system across various forecasting horizons. Empirical results demonstrate that dBG-enhanced models outperform traditional approaches and achieve superior accuracy in both short-term and long-term electricity load forecasting. These findings highlight the potential of dBGs as a key tool for improving power grid management and advancing sustainable energy systems.**

***Index Terms*—Load Forecasting, Time Series, de Bruijn Graphs, Power Systems, Deep Learning**

I. Introduction

As a fundamental aspect of power system planning and operation, effective load prediction enables grid operators to balance supply and demand, mitigate instability risks, and enhance overall resource efficiency [1]. According to the State Grid Corporation of China, even a marginal 1% improvement in forecasting accuracy could prevent nearly 58,000 MW of power from being wasted annually, highlighting its direct impact on efficiency and resource utilization [2]. As power grids continue to modernize, load forecasting has evolved beyond conventional energy management to play a crucial role in maintaining system stability, optimizing frequency regulation, and strengthening cybersecurity measures [3]. Consequently, developing advanced methodologies for electricity load forecasting has become a key focus in power system research.

A. Related Works

The load forecasting methodologies presented in the existing literature are selected based on the complexity of demand patterns, external influencing factors, and the desired level of accuracy. Deep learning has become a powerful tool for load forecasting due to its ability to automatically learn complex patterns and capture temporal dependencies. Recurrent neural networks (RNNs) were among the first deep learning models used for time series forecasting [4], but their effectiveness was hindered by the vanishing gradient problem, limiting their ability to model long-term dependencies. Long short-term memory (LSTM) networks addressed this issue with gating mechanisms to retain essential information over longer sequences [5]. Further advancements have introduced hybrid models combining LSTMs with convolutional neural networks (CNNs) to improve spatiotemporal feature extraction [6]. Despite their adaptability to diverse datasets, deep learning models still struggle to effectively capture long-range dependencies, which remains a key limitation in load forecasting.

Recent advancements in time series forecasting have been driven by transformer-based architectures, which effectively model both short-term variations and long-term dependencies. To evaluate the performance of state-of-the-art models, we employed the Time Series Library (TSLib) [7], a comprehensive repository featuring leading deep-learning frameworks for time-series analysis. The following provides a summary of key models utilized in our study. iTransformer refines traditional transformer designs by applying feed-forward networks to inverted dimensions, enhancing multivariate correlation modeling without requiring significant architectural modifications [8]. On the other hand, TimesNet introduces a novel representation of time-series data by transforming it into 2D tensors, allowing for more effective pattern recognition and improved forecasting performance [7]. DLinear adopts a simple yet efficient decomposition approach, breaking input data into trend and seasonal components. After that, two linear layers are applied to these components separately, and the final prediction is formed after adding the two layers to get the sum [9]. FiLM (Frequency-Improved Legendre Memory Model) enhances historical data retention while filtering out noise through Legendre polynomial projections and Fourier-based noise reduction to improve forecasting as well as computational efficiency [10]. These advanced architectures offer improved forecasting accuracy across diverse time-series applications.

The major contributions of this work are summarized as:

- This work introduces a novel methodology that utilizes de Bruijn Graphs (dBGs) to enhance electricity load time-series representation in large-scale power grids.
- A new technique is proposed to identify substitute relationships within discretized dBG characters, strengthening its representational capabilities.
- The *struct2vec* encoding method is employed to extract hidden load behavior patterns, improving predictive accuracy over traditional forecasting models.
- Four dBG-integrated architectures: FiLM, iTransformer, TimesNet, and DLinear—are developed to improve forecasting accuracy by preserving spatial relationships and capturing long-range dependencies in load variations.
- Extensive evaluation on the Texas 2,000-bus system demonstrates that dBG-enhanced models consistently outperform conventional approaches in various forecasting scenarios.

II. Graph-Based Power System Representation and Load Prediction

A. Modeling of Power System

Power grids inherently exhibit a networked structure, where interconnected buses and power lines form a complex system that can be effectively represented as a graph. Power system behavior is influenced by spatial relationships arising from the physical connectivity between buses and temporal variations driven by factors such as demand fluctuations, seasonal changes, and weather conditions. In this study, the power grid is formulated as a weighted undirected graph, denoted as $G = (V, E, \mathbf{W})$. Here, $V = \{1, 2, \ldots, B\}$ represents the set of buses (nodes), with B being the total number of buses. E denotes the set of power lines (edges) connecting the buses. $\mathbf{W} \in \mathbb{R}^{B \times B}$ is the weighted adjacency matrix, where each entry $\mathbf{W}_{ij}$ reflects the electrical coupling strength between buses i and j, such as line admittance or impedance. Each node in the graph contains attributes including voltage magnitude, active/reactive power demand, and power injection, while edges encode power flow characteristics influenced by transmission line properties.

B. Dataset

The 2,000-bus test case [11] is designed to reflect the geographical footprint of the Electric Reliability Council of Texas (ERCOT), covering a major portion of Texas. The dataset includes eight geographic regions with 1,500 substations and a total system load of 49,776 MW, assuming a fixed power factor of 0.96. Each substation contains a 115 kV bus, while 15% also feature a 345 kV bus to accommodate larger generation and higher loads. Generators operate at 13.8 kV and are linked to the highest voltage bus in their substations, while loads connect to the lowest voltage bus.

Transmission infrastructure consists of 287 lines at 345 kV and 1,813 lines at 115 kV, incorporating both overhead and underground configurations. Underground lines are used for distances under 8 miles when carrying 200 MW or more. The dataset does not replicate any real Texas grid but emulates structural and operational characteristics of real-world systems [11]. The combined 345 kV and 115 kV network is fully connected, structurally reliable, and resilient to single-node failures, as validated through comparisons with Eastern Interconnect statistics [12].

III. Proposed Methodology

De Bruijn Graphs (dBGs) provide a compact and structured representation of discrete sequential data, enabling efficient pattern recognition and analysis. Their construction ensures the presence of Eulerian paths, where each edge is traversed exactly once, making them well-suited for sequence-based modeling tasks [13]. Since dBGs operate on categorical data, any continuous dataset $\mathcal{D}_{\text{raw}}$ must first be discretized. This process, defined as:

$$\text{DISC}(\mathcal{D}_{\text{raw}}, \alpha) \rightarrow \mathcal{D}_{\text{disc}}(\mathcal{D}_{\text{raw}}, \alpha), \tag{1}$$

maps each value in $\mathcal{D}_{\text{raw}}$ to a discrete symbol from a finite alphabet $\Sigma = \{c_1, c_2, \ldots, c_\alpha\}$, ensuring consistent binning across datasets and subsets.

A k-tuple (or k-mer) represents a contiguous sequence of length k within $\mathcal{D}_{\text{disc}}$, forming the edges $\mathcal{E}$ in a k^{th}-order dBG. Each node $\mathcal{V}$ corresponds to a $(k-1)$-tuple, representing the prefix and suffix of k-tuples. Edges $(\mathfrak{s}, \mathfrak{t})$ connect nodes where the suffix of one tuple overlaps with the prefix of another, creating a directed graph that encodes sequential relationships. The edge weight reflects the frequency of k-tuples in $\mathcal{D}_{\text{disc}}$, preserving statistical properties of the dataset. The overall dBG construction process is outlined in Algorithm 1.

Algorithm 1: dBG Construction Algorithm

$\mathcal{D}_{\text{raw}}$: Continuous dataset
α: Alphabet size for discretization
k: Length of k-tuples
DISC: Discretization function
$dBG = (\mathcal{V}, \mathcal{E}, w)$: Constructed de Bruijn Graph
dBGConstruction($\mathcal{D}_{raw}, \alpha, k,$ DISC)
$\mathcal{D}_{\text{disc}} \leftarrow \text{DISC}(\mathcal{D}_{\text{raw}}, \alpha)$
$\mathcal{V} \leftarrow \emptyset, \quad \mathcal{E} \leftarrow \emptyset, \quad w : \mathcal{E} \rightarrow \mathbb{N}$
foreach $\mathfrak{s} \in \mathcal{D}_{disc}$ **do**
  for $i \leftarrow 0$ **to** $|\mathfrak{s}| - k$ **do**
    $\mathfrak{p} \leftarrow \mathfrak{s}[i : i + k - 1]$ $\mathfrak{q} \leftarrow \mathfrak{s}[i + 1 : i + k]$
    $\mathfrak{k} \leftarrow \mathfrak{s}[i : i + k]$
    if $\mathfrak{p} \notin \mathcal{V}$ **then**
      $\mathcal{V} \leftarrow \mathcal{V} \cup \{\mathfrak{p}\}$
    end
    if $\mathfrak{q} \notin \mathcal{V}$ **then**
      $\mathcal{V} \leftarrow \mathcal{V} \cup \{\mathfrak{q}\}$
    end
    if $(\mathfrak{p}, \mathfrak{q}) \notin \mathcal{E}$ **then**
      $\mathcal{E} \leftarrow \mathcal{E} \cup \{(\mathfrak{p}, \mathfrak{q})\}, \quad w(\mathfrak{p}, \mathfrak{q}) \leftarrow 1$
    end
    else
      $w(\mathfrak{p}, \mathfrak{q}) \leftarrow w(\mathfrak{p}, \mathfrak{q}) + 1$
    end
  end
end
return $(\mathcal{V}, \mathcal{E}, w)$

A. Substitute dBG

The substitute de Bruijn Graph (SdBG) enhances the classical dBG by incorporating symbol substitution relationships, crucial for capturing dependencies in sequential data. To model substitutions, a score matrix, $\delta \in \mathbb{R}^{\alpha \times \alpha}$ quantifies the similarity between symbols in the discretized alphabet Σ. The probability of substituting c_1 for c_2 is defined as:

$$\mathcal{P}_{c_1,c_2} = \frac{\sum_{i=0}^{\ell-1} p^i_{c_1,c_2}}{\ell \, \frac{n(n-1)}{2}}. \quad (2)$$

Each edge $e \in \mathcal{E}$ is considered similar to f if $\delta(e[i], f[i]) \geq \theta$ for all $i < k$, where θ is a predefined threshold. Edge weights are updated as:

$$w_e \leftarrow w_e + \kappa \sum_{f \in M_{e,\theta}} w_f \cdot \mathcal{S}_{rel}(e, f), \quad (3)$$

where κ controls the weight adjustment. The SdBG retains the same topology as the original dBG but refines edge weights to capture substitution-based relationships. The overall time complexity remains $\mathcal{O}(\mathcal{E}km)$, where m is the number of Trie edges.

B. de Bruijn Graph Encoding Using Struct2Vec

To encode the Substitute de Bruijn Graph, we utilize Struct2Vec (S2V) [14], which learns node embeddings based on structural identity. S2V constructs a multilayer graph, where each layer encodes structural similarity with edge weights:

$$w_i(v_1, v_2) = e^{-f_i(v_1,v_2)}, \quad (4)$$

where $f_i(v_1, v_2)$ recursively measures node distances based on their i-hop neighborhoods. The learned representations capture hierarchical graph structures and are precomputed to reduce training overhead.

C. FiLM with dBG

FiLMdBG integrates dBG embeddings into the FiLM architecture for time-series forecasting. The input is first normalized using reversible instance normalization (RevIN) [15], which adjusts scale and trend variations as:

$$\hat{X} = \gamma \frac{X - \mu}{\sigma} + \beta + \textsc{Lin}(X_{dBG}), \quad (5)$$

where μ, σ are the mean and standard deviation, γ, β are learnable parameters, and LIN transforms X_{dBG} via an MLP.

The processed data moves through Legendre Projection Units (LPU) that encode historical dependencies and then through the Frequency Enhanced Layer (FEL) that filters noise using Fourier transforms. These components capture temporal patterns at multiple scales and construct a hierarchical representation. The multiscale outputs undergo denormalization via RevIN to generate the final forecast.

D. TimesNet with dBG

TimesNet utilizes the fast Fourier transform (FFT) to analyze time-series data by converting it into the frequency domain. This approach enables the extraction of amplitude components $Amp(\cdot)$, which quantify the contributions of different frequency elements. To improve this transformation in TimesNet-dBG, we integrate the dBG embedding X_{dBG} with the output of an initial 1D convolution layer $Conv(\cdot)$ applied to the original time-series data. The transformation is defined as:

$$\begin{aligned} \mathbf{A}' &= Mean\left(Amp\left(Conv_{1D}\left(X_{1D}\right) \| X_{dBG}\right)\right) \quad (6) \\ \{f_1, \ldots f_k\} &= \arg \mathsf{Top}-k_{f_* \in \{1, \ldots \frac{\ell}{2}\}}(\mathbf{A}') \quad . \end{aligned}$$

By incorporating X_{dBG}, TimesNet-dBG enhances its ability to capture intricate frequency patterns, improving its effectiveness in multi-horizon forecasting tasks.

E. DLinear with dBG

DLinear decomposes time-series data into trend and seasonal components, each processed by separate linear layers. To enhance this structure, DLinear-dBG integrates dBG embeddings X_{dBG}, refining temporal dependencies. The final forecast is computed as:

$$\hat{\mathbf{Y}} = \text{LIN}(\mathbf{Y}_{\text{trend}} + \mathbf{Y}_{\text{seasonal}} + \text{LIN}(X_{dBG})). \quad (7)$$

This integration improves forecasting accuracy by leveraging both sequential trends and graph-based representations.

F. iTransformer with dBG

iTransformer-dBG enhances iTransformer by concatenating dBG embeddings X_{dBG} with the input features:

$$h_0^n = \text{Embedding}(X \| X_{dBG}). \quad (8)$$

The Transformer blocks (TrmBlock) then process these enriched representations, leveraging self-attention for inter-variate dependencies and feed-forward networks for temporal feature extraction. After L stacked layers, the final output is projected to forecast future values, integrating both sequential patterns and graph-based structure.

IV. Experimental Results

This section evaluates the proposed approach across various forecast horizons using a 2,000-bus Texas dataset with one year of hourly load data. Performance is assessed using five key metrics: SMAPE, MAPE, MASE, MSE, and MAE [16], [17].

Table I compares DLinear, FiLM, iTransformer, and TimesNet with and without dBG integration, highlighting the best-performing models in red. While dBG-enhanced models generally outperform their counterparts, DLinear performs better without dBG for 48- and 168-hour horizons, whereas DLineardBG excels at 24- and 96-hour horizons. For FiLMdBG, the model outperforms across all metrics and forecast horizon values, as does iTransformerdBG. TimesNet, on the other hand, only performs better when the forecast horizon is set to 24, while for all other cases, the TimesNet-dBG model achieves better performance. Figure 1 illustrates forecasting performance across all models and horizons, where each row represents a model and each column denotes a forecast horizon.

V. Conclusion and Future Works

This study introduces an extended FiLM model with dBG integration for long-term electricity load forecasting. The evaluation on the ACTIVSg2000-bus system shows that dBG-based embeddings improve forecasting accuracy by preserving sequential dependencies and outperform state-of-the-art benchmarks. The approach adapts to multiple models, including TimesNet, FiLM, iTransformer, and DLinear. Future research could explore dBGs for demand-side management to enhance demand response strategies and minimize utility costs. Additionally, dBG-based embeddings may improve fault detection by identifying anomalies in power grid operation.

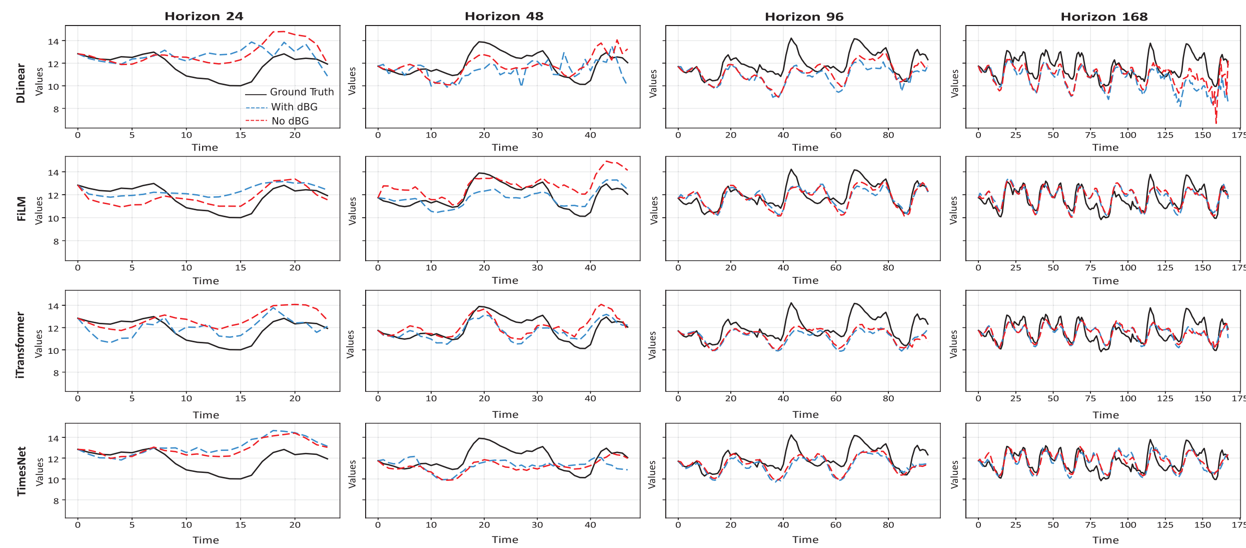

Fig. 1. Comparison of prediction performance across ground truth and models with and without dBG integration.

TABLE I. Performance Comparison Across Different Forecast Horizons With and Without dBG Encoding.

FORECAST HORIZON: 24						
Model	dBG	SMAPE	MAPE	MASE	MSE	MAE
DLinear	✗	23.155	27.037	3.511	147.335	5.592
	✓	23.140	27.312	3.571	139.474	5.455
FiLM	✗	23.573	27.874	3.575	158.327	5.783
	✓	22.536	26.672	3.400	145.387	5.532
iTransfomer	✗	25.522	31.117	3.993	183.035	6.417
	✓	24.517	29.263	3.729	169.490	5.996
TimesNet	✗	24.626	26.869	3.487	153.696	5.588
	✓	23.792	23.792	3.444	164.856	5.930
FORECAST HORIZON: 48						
Model	dBG	SMAPE	MAPE	MASE	MSE	MAE
DLinear	✗	19.218	19.398	2.740	68.939	4.140
	✓	19.448	18.756	2.627	64.912	4.093
FiLM	✗	20.843	20.601	2.879	77.149	4.386
	✓	18.756	18.756	2.627	64.912	4.093
iTransfomer	✗	21.615	21.354	2.944	84.295	4.629
	✓	19.269	19.269	2.647	64.527	4.105
TimesNet	✗	19.887	19.729	2.733	70.992	4.263
	✓	18.916	20.115	2.683	70.606	4.202
FORECAST HORIZON: 96						
Model	dBG	SMAPE	MAPE	MASE	MSE	MAE
DLinear	✗	18.827	19.618	2.761	87.134	4.357
	✓	18.376	18.656	2.667	82.409	4.165
FiLM	✗	18.887	19.148	2.721	93.650	4.369
	✓	18.168	18.193	2.603	88.741	4.179
iTransfomer	✗	19.259	18.815	2.707	89.641	4.298
	✓	18.633	18.633	2.698	90.022	4.271
TimesNet	✗	19.126	19.965	2.740	84.526	4.349
	✓	18.980	19.860	2.712	84.462	4.332
FORECAST HORIZON: 168						
Model	dBG	SMAPE	MAPE	MASE	MSE	MAE
DLinear	✗	18.041	19.669	2.602	68.027	4.056
	✓	18.076	19.033	2.607	68.646	4.062
FiLM	✗	19.226	20.073	2.73	76.174	4.287
	✓	18.811	18.756	2.667	74.179	4.200
iTransfomer	✗	19.859	19.996	2.787	91.372	4.422
	✓	18.578	19.578	2.659	91.724	4.220
TimesNet	✗	20.195	19.884	2.836	95.404	4.530
	✓	18.754	19.329	2.662	79.236	4.214

References

[1] A. Sharma and S. K. Jain, "A novel two-stage framework for mid-term electric load forecasting," *IEEE Transactions on Industrial Informatics*, vol. 20, no. 1, pp. 247–255, 2024.

[2] L. Xiao, J. Wang, X. Yang, and L. Xiao, "A hybrid model based on data preprocessing for electrical power forecasting," *International Journal of Electrical Power & Energy Systems*, vol. 64, pp. 311–327, 2015.

[3] N. Sahani and C.-C. Liu, "Model-based detection of coordinated attacks (dca) in distribution systems," *IEEE Open Access Journal of Power and Energy*, vol. 11, pp. 558–570, 2024.

[4] M. N. Fekri, H. Patel, K. Grolinger, and V. Sharma, "Deep learning for load forecasting with smart meter data: Online adaptive recurrent neural network," *Applied Energy*, vol. 282, p. 116177, 2021.

[5] S. Muzaffar and A. Afshari, "Short-term load forecasts using lstm networks," *Energy Procedia*, vol. 158, pp. 2922–2927, 2019.

[6] S. H. Rafi, S. R. Deeba, E. Hossain *et al.*, "A short-term load forecasting method using integrated cnn and lstm network," *IEEE Access*, vol. 9, pp. 32 436–32 448, 2021.

[7] H. Wu, T. Hu, Y. Liu, H. Zhou, J. Wang, and M. Long, "Timesnet: Temporal 2d-variation modeling for general time series analysis," *arXiv preprint arXiv:2210.02186*, 2022.

[8] Y. Liu, T. Hu, H. Zhang, H. Wu, S. Wang, L. Ma, and M. Long, "itransformer: Inverted transformers are effective for time series forecasting," *arXiv preprint arXiv:2310.06625*, 2023.

[9] A. Zeng, M. Chen, L. Zhang, and Q. Xu, "Are transformers effective for time series forecasting?" *arXiv preprint arXiv:2205.13504*, 2022. [Online]. Available: https://arxiv.org/abs/2205.13504

[10] T. Zhou, Z. Ma, xue wang, Q. Wen, L. Sun, T. Yao, W. Yin, and R. Jin, "FiLM: Frequency improved legendre memory model for long-term time series forecasting," in *Advances in Neural Information Processing Systems*, A. H. Oh, A. Agarwal, D. Belgrave, and K. Cho, Eds., 2022. [Online]. Available: https://openreview.net/forum?id=zTQdHSQUQWc

[11] A. B. Birchfield, T. Xu, K. M. Gegner, K. S. Shetye, and T. J. Overbye, "Grid structural characteristics as validation criteria for synthetic networks," *IEEE Transactions on Power Systems*, vol. 32, no. 4, pp. 3258–3265, 2016.

[12] K. M. Gegner, A. B. Birchfield, T. Xu, K. S. Shetye, and T. J. Overbye, "A methodology for the creation of geographically realistic synthetic power flow models," in *2016 IEEE Power and Energy Conference at Illinois (PECI)*. IEEE, 2016, pp. 1–6.

[13] N. G. De Bruijn, "A combinatorial problem," *Proceedings of the Section of Sciences of the Koninklijke Nederlandse Akademie van Wetenschappen te Amsterdam*, vol. 49, no. 7, pp. 758–764, 1946.

[14] D. R. Figueiredo, L. F. R. Ribeiro, and P. H. P. Saverese, "struc2vec: Learning node representations from structural identity," *CoRR*, vol. abs/1704.03165, 2017. [Online]. Available: http://arxiv.org/abs/1704.03165

[15] T. Kim, J. Kim, Y. Tae, C. Park, J.-H. Choi, and J. Choo, "Reversible instance normalization for accurate time-series forecasting against distribution shift," in *International Conference on Learning Representations*, 2022. [Online]. Available: https://openreview.net/forum?id=cGDAkQo1C0p

[16] S. Makridakis, E. Spiliotis, and V. Assimakopoulos, "Statistical and machine learning forecasting methods: Concerns and ways forward," *PloS One*, vol. 13, no. 3, p. e0194889, 2018.

[17] D. Chicco, M. J. Warrens, and G. Jurman, "The coefficient of determination r-squared is more informative than smape, mae, mape, mse and rmse in regression analysis evaluation," *Peerj computer science*, vol. 7, p. e623, 2021.