Engineering Applications of Artificial Intelligence 167 (2026) 113939

Contents lists available at ScienceDirect

Engineering Applications of Artificial Intelligence

journal homepage: www.elsevier.com/locate/engappai

Research paper

Fourier-enhanced sequence-to-sequence latent graph neural networks for multi-node spatiotemporal forecasting in a hydroelectric reservoir

Laio Oriel Seman [a,*], Stefano Frizzo Stefenon [b,c], Kin-Choong Yow [b], Leandro dos Santos Coelho [d,e], Viviana Cocco Mariani [d,f]

[a] Department of Automation and Systems Engineering, Federal University of Santa Catarina (UFSC), Florianopolis, SC, Brazil
[b] Faculty of Engineering and Applied Sciences, University of Regina, Saskatchewan, S4S 0A2, Canada
[c] Lisbon School of Engineering (ISEL), Polytechnic University of Lisbon (IPL), Lisbon, Portugal
[d] Department of Electrical Engineering (DELT), Federal University of Parana (UFPR), Curitiba, PR, Brazil
[e] Graduate Program in Electrical Engineering (PPGEE), Federal University of Parana, Curitiba, PR, Brazil
[f] Graduate Program in Mechanical Engineering (PGMec), Federal University of Parana, Curitiba, PR, Brazil

ARTICLE INFO

Dataset link: https://github.com/lseman/Seq2SeqLatentGNN/

Keywords:
Graph neural networks
Fourier neural operator
Spatiotemporal forecasting
Hydroelectric reservoirs

ABSTRACT

This paper presents a Fourier-enhanced dynamic sequence-to-sequence latent graph neural network (Seq2SeqLatentGNN), a deep learning architecture for multi-node spatiotemporal forecasting in hydroelectric reservoir systems. The model integrates three key components: (i) a custom Fourier layer that analyzes global temporal patterns through frequency-domain transformations, (ii) a latent correlation graph convolutional network that infers relational structures between monitoring stations without requiring predefined adjacency matrices, and (iii) an attention-based sequence-to-sequence model that processes temporal dependencies while enabling multi-step forecasting. The architecture simultaneously learns graph structure and forecasting tasks, adapting to changing spatial relationships between reservoir nodes. The proposed architecture was evaluated using a comprehensive dataset derived from 19 interconnected hydroelectric reservoirs located in southern Brazil. The dataset encompasses multiple years of high-resolution (hourly) measurements, including reservoir water levels, inflow and outflow rates, precipitation records, and energy production metrics. Experimental results demonstrate that Seq2SeqLatentGNN achieves superior performance compared to conventional statistical models and contemporary machine learning methods, as measured by standard error metrics. Analysis of the learned latent correlations reveals meaningful spatial dependencies that align with hydrological principles. The model exhibits consistent performance across varying temporal patterns, adapts to regime transitions, and captures both periodic and nonstationary dynamics. The proposed architecture contributes to spatiotemporal forecasting by combining spectral processing, dynamic graph learning, and sequence modeling in a unified framework applicable to systems with evolving connectivity patterns.

1. Introduction

The practice of forecasting a variable's future values while taking consideration of both its temporal (time) and spatial (location) dependencies has been referred to as spatiotemporal forecasting. Spatiotemporal forecasting takes into consideration how a variable's value at a certain location is influenced by values at other locations and how these relationships change over time, in comparison with typical time-series forecasting, which only takes into account a variable's past values. It is therefore a necessary instrument for modeling dynamic, complex systems.

Spatiotemporal forecasting supports decision-making in domains such as climate science (Ferchichi et al., 2024), financial markets (Ma et al., 2025), transportation systems (Moriano et al., 2024), health (Pan et al., 2025), and energy (Yoosuf et al., 2025). Previous research areas encompass multiple variables that evolve over time, giving rise to complex dynamics that are challenging for conventional models to capture, because they do not take into account the intrinsic spatiotemporal dependencies present (Sun et al., 2024). The scientific community has explored several models to address these limitations: classical statistical models, e.g., vector autoregression (Chavleishvili and Manganelli, 2024), autoregressive integrated moving average (Alsakarneh et al.,

* Corresponding author.
E-mail addresses: laio@gos.ufsc.br (L.O. Seman), stefano.stefenon@isel.pt (S.F. Stefenon), kin-choong.yow@uregina.ca (K.-C. Yow), leandro.coelho@ufpr.br (L.d.S. Coelho), viviana.mariani@ufpr.br (V.C. Mariani).

https://doi.org/10.1016/j.engappai.2026.113939
Received 11 June 2025; Received in revised form 15 December 2025; Accepted 20 January 2026
Available online 23 January 2026

2025), and deep learning approaches (Medrano-Diaz et al., 2025), such as recurrent neural networks (Apaydin et al., 2020) and transformers (Li et al., 2025a). While loosely coupled univariate or multivariate time series can be modeled by these methods, the latent and dynamic relational structure between nodes is often ignored. Consequently, performance is affected in scenarios where spatial dependencies are both important and evolve over time (Jin et al., 2024).

Graph neural networks (GNNs) extend conventional neural networks by processing data defined on irregular, non-Euclidean domains, making them naturally suited for spatiotemporal problems where relationships between entities evolve over time (Zhou et al., 2025). GNNs have advanced the modeling of relational data (Kim et al., 2025), with applications in dialogue act classification (Fu et al., 2025), electroencephalogram classification (Liu et al., 2025), facial age estimation (Zhang et al., 2025), interactive recommendation (Li et al., 2025b), and time series forecasting (Kim et al., 2025). However, predefined, static graph structures are operated on by most GNN-based approaches. This assumption is not held in many real-world systems where node relationships are implicit, noisy, or non-stationary (Kipf et al., 2018). In these contexts, the underlying dynamics can be misrepresented by fixed adjacency matrices.

A spatiotemporal forecasting scenario is presented by hydroelectric reservoir systems, where the relationships between monitoring stations are influenced by factors beyond simple physical connectivity (da Silva et al., 2024). Upstream reservoirs affect downstream facilities with time lags that depend on release schedules, precipitation patterns, and watershed characteristics. In addition, reservoirs within the same river basin may exhibit correlated behavior due to shared meteorological conditions despite lacking direct hydrological connections. Cascading systems add complexity as operational decisions propagate through the network with variable temporal dynamics (Zhong et al., 2024).

Given the limitations of existing modeling approaches for hydroelectric systems, coupled with the complex physical interconnections among reservoirs, this study is motivated by the following objectives:

- The inability of traditional models to capture both spatial and temporal dependencies is a limitation, particularly in systems with evolving node relationships, such as hydroelectric reservoirs.
- Most GNNs rely on static graph structures; however, these structures are inadequate for the dynamic and latent relationships found in hydroelectric reservoir systems.
- Complex spatiotemporal dependencies, such as delayed upstream–downstream effects, meteorological influences, nonstationarity, and regime transitions, are presented by hydroelectric reservoirs. These dependencies require the use of adaptive models.

A Fourier-enhanced dynamic sequence-to-sequence latent graph neural network (Seq2SeqLatentGNN) is introduced in this study to address these requirements. This spatiotemporal forecasting architecture integrates spectral processing, dynamic graph learning, and sequence modeling. The limitations of existing approaches are overcome by the proposed Seq2SeqLatentGNN through the following mechanisms:

- Dynamic latent graphs are inferred rather than relying on fixed, predefined adjacency matrices.
- Global temporal patterns and local spatial relationships are modeled in systems with non-static or noisy connections, such as reservoirs.
- Fourier-based spectral analysis, attention mechanisms, and sequence modeling are integrated into a single, hybrid architecture.

The Seq2SeqLatentGNN model consists of main components, including a custom Fourier layer, a latent correlation graph convolutional network (GCN), and the use of an attention-based sequence-to-sequence model. The analysis of global temporal patterns in reservoir networks is conducted through frequency-domain transformations performed by the custom Fourier layer. The decomposition of time-series data into its spectral components allows for the identification of periodic behaviors, dominant frequencies, and long-term trends, thereby enhancing the interpretability of dynamic reservoir processes.

The latent correlation GCN dynamically infers the relational structure between monitoring stations by leveraging attention mechanisms, eliminating the need for predefined adjacency matrices. This approach autonomously captures spatiotemporal dependencies in the hydroelectric reservoir, enabling adaptive learning of node interactions based on latent feature correlations. By integrating self-attention with graph convolutions, the model enhances forecasting accuracy while maintaining computational efficiency. The attention-based sequence-to-sequence model captures complex temporal dependencies in spatiotemporal data, enabling robust long-term time series forecasting through autoregressive refinement. By dynamically weighting historical patterns, the attention mechanism selectively prioritizes the most relevant input features, improving prediction accuracy while mitigating error propagation over long forecasting horizons.

The information that follows summarizes the advantages of the proposed Seq2SeqLatentGNN:

- Latent spatial relationships are learned through dynamic graph learning, which does not require prior connectivity information.
- Global periodicities and long-term trends are captured using Fourier-based spectral processing.
- Temporal forecasting is improved by attention-based sequence modeling, as relevant patterns are focused on.
- Spatiotemporal dynamics and graph structure are jointly learned by the integrated architecture, which enables better generalization and interpretability in dynamic environments.
- Robust multi-step forecasting is handled effectively, even with nonstationary behaviors and regime shifts.

In summary, the proposed architecture learns graph structure and forecasting tasks simultaneously, adapting to changing spatial relationships. The approach applies to hydrological systems with evolving connectivity between reservoirs. In this study, 19 interconnected hydroelectric reservoirs in southern Brazil were evaluated. The evaluation comprised multiple years of hourly measurements of reservoir levels, inflow and outflow rates, precipitation, and energy production metrics.

The main contributions of this work, which advance the current state of the art in spatiotemporal forecasting and graph-based modeling, are as follows:

- **Joint learning of latent spatial structure and temporal forecasting.** Traditional GNN-based forecasting models, which rely on predefined or static graphs, are unlike the proposed Seq2SeqLatentGNN. With this new method, a dynamic latent graph and the forecasting task are learned simultaneously. This allows evolving, nonlinear spatial dependencies to be captured without needing physical connectivity priors. As a result, a key limitation in existing spatiotemporal architectures is addressed.
- **Integration of spectral and temporal modeling through a Fourier-based operator.** Time series are processed in the frequency domain by a custom Fourier layer incorporated into the model, which allows for the detection of long-term periodicities and global patterns. The model's ability to forecast under regime shifts and seasonal variability is enhanced by this spectral processing capability, a feature largely absent in prior GNN-based hydrological models.
- **Multi-scale output smoothing for stability in noisy environments.** A Gaussian smoothing operator is applied at the output stage to reduce high-frequency fluctuations in predictions. This component, which has not been previously integrated into end-to-end graph-based forecasting pipelines, improves prediction stability and reduces variance, particularly in highly dynamic or noisy systems such as hydroelectric reservoirs.

- **Adaptive hybrid detrending for signal preprocessing.** A novel detrending approach is proposed that combines Savitzky-Golay filtering and polynomial regression with adaptive weights. This preprocessing strategy isolates meaningful cyclical and seasonal components. Consequently, model performance is improved compared to standard time series normalization or trend removal techniques.
- **Evaluation on a large-scale, real-world hydroelectric system with complex dynamics.** A comprehensive evaluation of the model was conducted using a multi-year dataset from 19 interconnected reservoirs in southern Brazil, which incorporated diverse meteorological, hydrological, and operational conditions. Consistent performance improvements over classical statistical models and deep learning baselines were demonstrated, highlighting the method's practical relevance and scalability.

The remainder of this paper is structured as follows. In Section 2, the literature on forecasting models is reviewed. In Section 3, the dataset is described. A detailed exposition of the proposed Seq2SeqLatentGNN is provided in Section 4. The experimental setup, results, and analysis are covered in Section 5. Finally, the paper is concluded and future research directions are highlighted in Section 6.

2. Literature review

In this section, the literature review is structured into two distinct subsections. In the first, research that employs standard models for time series forecasting is examined, while in the second, models developed for spatiotemporal forecasting are the focus. Particular attention is given to studies addressing forecasting applications within river basins and hydroelectric power plants.

2.1. Forecasting models for temporal data

Statistical models have historically constituted a fundamental methodology for forecasting hydroelectric reservoir dynamics, providing interpretable frameworks capable of representing temporal dependencies and seasonal patterns intrinsic to hydrological processes (Zhou et al., 2023). Models such as the autoregressive moving average (Rigby et al., 2024), and support vector regression (Fadhillah et al., 2021) are utilized due to their methodological simplicity and interpretability (Fernando De Toledo et al., 2023).

These models are frequently improved with signal decomposition techniques such as seasonal-trend decomposition using locally estimated scatterplot smoothing (STL) (Li et al., 2020; Tebong et al., 2023), variational mode decomposition (VMD) (Moreno et al., 2024), empirical wavelet transform (EWT) (Stefenon et al., 2024a), Hodrick–Prescott filter (Muniz et al., 2025), empirical mode decomposition (EMD) (Larcher et al., 2024), and ensemble EMD (EEMD) (jing Niu et al., 2019; Maiti et al., 2024) to enhance signal processing capabilities. Nevertheless, these conventional methodologies are inherently constrained by their linear assumptions, limiting their capacity to accurately capture the nonlinear and intricate spatiotemporal dynamics characteristic of hydroelectric reservoir behavior, particularly over extended forecasting horizons and under complex operational conditions (Barzola-Monteses et al., 2025).

Hybrid models merge physics-based and data-driven approaches to improve accuracy in variable reservoir conditions (Zhou et al., 2025). In the proposed architecture, Li et al. (2020) proposed a combination of the STL with an extremely random tree model (extra-trees) to predict seasonal components, and the stacked long-short-term memory (LSTM) to predict dam displacement time series. Stefenon et al. (2023) developed a hybrid wavelet sequence-to-sequence (Seq2Seq) LSTM model enhanced with attention mechanisms to predict reservoir water level time series in hydroelectric facilities. Larcher et al. (2024) combined three time series forecasting models, echo state networks (ESNs), deep ESNs, and next generation reservoir computing, with three signal decomposition techniques (EMD, EWT, and VMD) optimized by the coyote optimization algorithm applied to multi-stage streamflow forecasting.

Conversely, ensemble models, which integrate the advantages of multiple forecasting methodologies, provide a robust and precise framework for predicting hydroelectric reservoir levels under varying and uncertain conditions. Specifically, random forest regressors demonstrate superior capability in modeling complex nonlinear relationships with strong resistance to overfitting, whereas extreme gradient boosting (XGBoost) delivers enhanced predictive performance through iterative boosting and regularization mechanisms (Chowdhury et al., 2025).

Deep learning models demonstrate potential for capturing complex nonlinear patterns in hydropower data (Praveen et al., 2020). Extreme learning machines provide rapid training capabilities alongside strong generalization performance (Martinho et al., 2020). Recurrent neural networks (RNNs), particularly ESNs, model complex temporal dynamics with minimal training effort, rendering them highly effective for time series forecasting in hydroelectric reservoir management (Larcher et al., 2024). Furthermore, LSTM networks and gated recurrent units (GRUs) represent substantial advancements in time series forecasting methodologies, including applications in hydropower prediction (Tebong et al., 2023). Notably, LSTMs are especially proficient in capturing long-term temporal dependencies, which are necessary for accurately modeling the sequential characteristics inherent in hydropower data.

Building on the advancements of deep learning in temporal forecasting, transformer models, characterized by their novel self-attention mechanism, have emerged in time series prediction, such as auto temporal fusion transformer (Stefenon et al., 2024b) applied to hydropower, and reservoir inflow forecasting (Xu et al., 2023). Their principal advantage resides in the effective capture of long-range temporal dependencies alongside the capacity for parallel computation, which offers improvements over traditional sequential models for a long-term forecasting horizon. Notable transformer variants such as Autoformer (Tian et al., 2024), iTransformer (Dai et al., 2025), and Informer (Rong et al., 2025) have further improved forecasting efficiency and accuracy, particularly for long input sequences.

Recent advancements in Seq2Seq models, especially those integrating attention mechanisms, have markedly improved the accuracy and reliability of hydrological forecasting. Seq2Seq architectures have demonstrated effective performance in predicting water levels (Paiva et al., 2023) as well as in estimating a range of hydrological indicators (Liu et al., 2023). Despite these successes, transformers face challenges when applied to practical forecasting scenarios. Their substantial computational demands become pronounced for extended forecasting horizons, and they exhibit limited capability in modeling spatial dependencies, an essential aspect in hydropower applications.

Previous models exhibit notable limitations when they are extended to multi-point forecasting scenarios. In particular, complex spatial and temporal dependencies exist among various monitoring locations in hydropower reservoir dynamics. Traditional deep learning methods, such as LSTM and GRU networks, mainly capture temporal dependencies within individual time series but overlook the spatial interrelations between different measurement sites. These limitations prompt the exploration of new methods that can better capture patterns in spatiotemporal data.

2.2. Forecasting models for spatiotemporal data

To effectively extract spatial features from spatiotemporal data, recent research has leveraged a combination of recurrent architectures, e.g., RNN, LSTM, Seq2Seq, and GNNs approaches that are useful for handling non-Euclidean data and can be categorized into spectral-based GNNs and spatial-based GNNs (Kim et al., 2023), including GCN (Lu et al., 2025) and graph attention networks (GATs) (Pang et al., 2023).

Li et al. (2018a) introduced an approach by integrating concepts from Residual Network (ResNet), DenseNet, and dilated convolution in GCNs, thereby enabling the construction of a substantially deeper network architecture. The diffusion convolutional RNN (DC-RNN) (Li et al., 2018b) combines diffusion-based GCN for modeling spatial dependencies with a Seq2Seq architecture that captures temporal dynamics. To the best of our knowledge, that work constitutes the first attempt to employ GNNs for spatiotemporal forecasting. The spatiotemporal GCN (STGCN) (Yu et al., 2018) leverages Chebyshev polynomial-based graph convolution and two-dimensional convolutional networks to model spatial dependencies and temporal correlations, respectively. The attention-based spatiotemporal GCN (ASTGCN) (Guo et al., 2019) introduces spatial and temporal attention mechanisms to enhance the modeling of spatial–temporal dynamics.

The spatiotemporal Graph to Sequence (STG2Seq) model (Bai et al., 2019) employs multiple gated graph convolutional modules in conjunction with a Seq2Seq architecture and attention mechanisms to perform multi-step predictions. The Graph WaveNet (Wu et al., 2019) integrates graph convolution with dilated causal convolution to effectively capture long-range spatial–temporal dependencies. The graph attention-based temporal convolutional network (GATCN) (Guo and Yuan, 2020) has demonstrated effectiveness in short-term spatiotemporal forecasting tasks.

The spatiotemporal synchronous GCN (STSGCN) (Song et al., 2020) adopts a multi-expert graph structure to connect nodes, with GCN layers extracting spatiotemporal features for prediction. More recently, the ASTGCN model (He et al., 2022) combines LSTM and GCN to extract temporal and spatial features, further enhanced through a multi-attention mechanism, achieving improved accuracy in dam displacement prediction. Deep learning approaches have demonstrated substantial effectiveness in modeling complex spatiotemporal dependencies across a wide range of application domains (Xu et al., 2021).

Wang et al. (2022) proposed an RNN architecture for spatiotemporal predictive learning incorporating a novel curriculum learning strategy to facilitate the acquisition of long-term dynamics, which can be generalized to a wide range of Seq2Seq models. The GNN-based framework was effectively applied to a snow-dominated watershed in the western United States (Feng et al., 2022). For instance, Liu et al. (2022) employed GNNs to simulate the physical processes underlying hydrological flow formation. Moreover, RNNs have been integrated within GNN frameworks (Zanfei et al., 2022) to enhance the modeling of temporal dynamics.

Yin and Xie (2021) proposed a multi-scale spatiotemporal convolutional network designed to model both the nonlinear characteristics and temporal properties inherent in load time series. Similarly, Huang et al. (2023) introduced a spatiotemporal GCN (ST-GCN) load forecasting framework, in which spatial convolutional layers extract features from neighboring nodes. In another study, Wei et al. (2023) proposed an exponential moving average GCN (EMA-GCN) that captures spatial dependencies and periodicity through two sets of trainable temporal embeddings.

Spatiotemporal LSTM networks extend the conventional LSTM architecture by explicitly incorporating spatial dependencies into the temporal modeling process. When integrated with convolutional operations or graph-based approaches, LSTM-based architectures demonstrate strong capability in capturing complex spatial relationships within forecasting network structures; however, they often demand substantial amounts of training data and computational time to achieve optimal performance. To address these limitations, models such as the spatiotemporal attention LSTM (STA-LSTM) (Noor et al., 2022) and the spatiotemporal multivariate-based time vario-zoom network (STM-TVZN) (Ji et al., 2024) have been proposed, leveraging multidimensional spatiotemporal variable features through adaptive time-window mechanisms.

Arastehfar et al. (2022) developed an approach that integrates GNNs with LSTM networks, enabling the joint modeling of spatial dependencies and temporal dynamics in load data with similar consumption patterns. In addition, spatiotemporal GCN (Dai et al., 2025) employs the Chebyshev spectral convolution network in combination with one-dimensional convolutional neural networks (CNNs) to effectively capture spatial dependencies and temporal correlations in time-series data. Tan et al. (2025) proposed an autoencoder-based, state-of-the-art spatiotemporal prediction model that preserves computational efficiency through the use of two-dimensional convolution-based attention layers. Yoosuf et al. (2025) implemented three variants of a convolutional autoencoder for spatiotemporal forecasting.

Table 1 provides a comprehensive summary of recent advances in spatiotemporal data forecasting models applied to hydroelectric reservoirs, hydropower, or similar, which have led to progress in modeling complex systems, including hydroelectric reservoirs, as demonstrated by several hybrid approaches presented. These models capture dynamic spatial dependencies and temporal patterns without relying on predefined graphical structures. However, challenges remain in generalizing these models to new environments, particularly under conditions of data scarcity or in the face of non-stationary dynamics, such as those caused by fluctuating energy demand or meteorological variability, present in dynamic river basins. Thus, previous approaches offer opportunities for further enhancement.

To address these challenges, this study proposes a spatiotemporal forecasting model called Seq2SeqLatentGNN for predicting volume, inflow, and outflow in a set of 19 hydroelectric reservoirs. The model extracts dynamic spatiotemporal correlation by combining the Seq2Seq temporal model, which uses encoder–decoder structures to encode past flow patterns and decode them into future forecasts, a latent GNN spatial model, which captures dependencies between nodes, such as how upstream flows or the behavior of neighboring basins influence each other through a flexible graph, and Fourier spectral enhancement, which ensures that periodic behaviors are explicitly represented and more easily learned by the model. Thus, the proposed hybrid model creates a robust architecture capable of simultaneously addressing temporal nonlinearity, heterogeneity, spatial dynamics, and periodic signals.

3. Dataset

This study investigates 19 reservoirs distributed across two watersheds in southern Brazil. As illustrated in Fig. 1, the reservoirs are represented by orange markers, while nearby urban centers are indicated by blue markers. Leveraging the shared river basins and the resulting hydrological interconnections, the study employs graph theory as a methodological framework to analyze the structural relationships among the reservoirs. The dataset is available on the National Electric System Operator (ONS)[1]. The ONS coordinates the operation of electricity generation and transmission across Brazil's National Interconnected System, as well as manages isolated systems. It operates under the supervision and regulation of Brazil's Electricity Regulatory Agency.

Tables 2–4 summarize the distribution of hydrological variables for the analyzed reservoirs, where ENA is the energy natural affluence by the power plant. Useful volume denotes the portion of the total storage capacity available for electricity generation. Inflow represents the volume of water entering the reservoir from upstream sources, including river discharge and precipitation. The ONS computes the natural energy inflow for each subsystem, expressed in average megawatts (MWmed), based on verified flow rates and the current configuration of the power generation infrastructure.

The Machadinho hydroelectric facility on the Pelotas River maintains a reservoir covering 79 km^2, with 56.7 km^2 representing flooded land. The reservoir perimeter extends 500 km with an average depth of 30 m. Downstream on the Uruguay River, the Itá hydroelectric facility

[1] https://dados.ons.org.br/dataset/dados_hidrologicos_ho

Table 1
Summary of literature review on spatiotemporal forecasting.

Authors	Model	Highlights
Li et al. (2018a)	GCN model	Co-training and self-training methods to improve performance with very few labels.
Li et al. (2018b)	DC-RNN	Captures spatial dependencies via bidirectional random walks and temporal dynamics with an encoder–decoder.
Yu et al. (2018)	STGCN	Fully convolutional graph-based model that captures spatio-temporal dependencies, enabling faster training.
Guo et al. (2019)	ASTGCN	Attention-based spatio-temporal GCN that models recent, daily, and weekly dependencies.
Bai et al. (2019)	STG2Seq	Hierarchical graph-based model, combining long and short-term encoders with an attention-based module.
Wu et al. (2019)	Graph WaveNet	Adaptive dependency matrix to capture hidden spatial relations and stacked dilated convolutions.
Li et al. (2020)	STL-extra-trees-LSTM	Decomposes data into components, predicting each separately and aggregating results.
Guo and Yuan (2020)	GATCN	Combines GATs for spatial features and temporal convolutional networks for temporal features.
Song et al. (2020)	STSGCN	Captures localized spatio-temporal correlations and heterogeneities simultaneously through synchronous GCNs.
He et al. (2022)	ASTGCN	LSTM and GCN with multi-attention mechanism for dam displacement prediction.
Huang et al. (2023)	ST-GCN	Spatial convolution to capture coupling correlations, combined with GRU for temporal forecasting.
Noor et al. (2022)	STA-LSTM	Combines spatial and temporal attention, improving forecasting accuracy over other LSTM-based approaches.
Arastehfar et al. (2022)	GCN and LSTM	Jointly learns spatial and temporal consumption patterns without relying on environmental variables.
Wei et al. (2023)	EMA-GCN	Integrates directed static and dynamic graphs to capture spatial dependencies.
Ji et al. (2024)	STM-TVZN	Spatio-temporal model with vario-zoom windows, improving water level and extremum event forecasting.

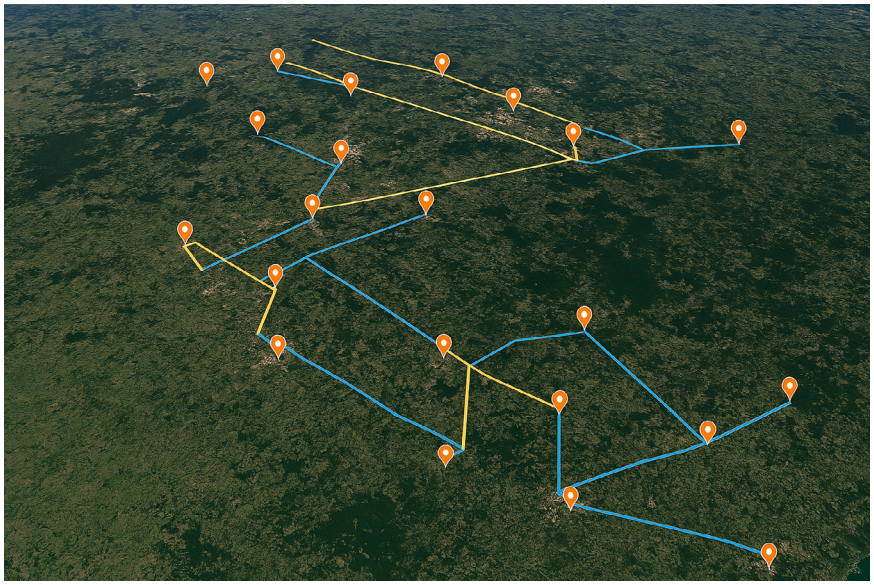

Fig. 1. Position of the reservoirs evaluated in this study.

operates with a 141 km^2 reservoir. This development required the construction of New Itá, a settlement established to relocate residents from the flooded area. The Gbmrn states for *Governador Bento Munhoz da Rocha Neto*, a Foz do Areia Hydroelectric Power Plant

The Foz do Chapecó facility's reservoir spans 79.2 km^2, including 40 km^2 of the Uruguay River's natural channel and 39.2 km^2 of additionally flooded area. This reservoir intersects with twelve municipalities, six within Santa Catarina state. The Campos Novos hydroelectric facility demonstrates high efficiency with a 29 km^2 reservoir, achieving one of Brazil's highest power-to-reservoir-size ratios. The variations in reservoir dimensions and characteristics reflect the topographical conditions of southern Brazil and the specific requirements of each hydroelectric project.

This dataset poses several forecasting challenges: (i) geographic dispersion across heterogeneous topographical regions, introducing spatial variability in hydrological processes, (ii) cascading dependencies with non-uniform time lags due to upstream–downstream interactions, (iii) dynamic operational regimes influenced by fluctuating energy demand and evolving environmental regulations, and (iv) meteorological non-stationarity, where precipitation, temperature, and evaporation exhibit spatiotemporal heterogeneity. These factors collectively

Table 2
Statistical summary of hydroelectric reservoir variables (part 1 of 3).

Reservoir	Variable	Minimum	Maximum	Mean	Standard deviation	Kurtosis	Skewness
Baixo Iguaçu	Useful volume (%)	0.00	284.32	46.02	16.61	1.64	0.31
	Inflow (m^3/s)	0.00	28 051.00	1814.49	1901.10	31.19	4.46
	Outflow (m^3/s)	0.00	23 215.00	1812.20	1873.16	31.64	4.57
	ENA (MWmed)	0.00	12 079.15	3969.50	1853.00	1.70	1.02
	Precipitation (mm)	0.00	28.40	0.01	0.31	4138.13	55.55
Barra Grande	Useful volume (%)	44.33	100.20	85.74	14.84	−0.08	−1.05
	Inflow (m^3/s)	0.00	23 311.00	429.34	782.47	79.15	6.48
	Outflow (m^3/s)	0.00	11 095.00	439.35	683.66	64.64	6.44
	ENA (MWmed)	663.33	16 299.74	3600.11	2431.26	4.66	1.84
	Precipitation (mm)	0.00	10.80	0.01	0.20	1343.79	33.41
Campos Novos	Useful volume (%)	21.00	103.19	83.34	16.25	0.98	−1.29
	Inflow (m^3/s)	0.00	6461.00	571.56	656.45	14.17	3.09
	Outflow (m^3/s)	0.00	6390.00	572.59	659.67	14.15	3.05
	ENA (MWmed)	663.33	16 299.74	3599.57	2431.14	4.66	1.84
	Precipitation (mm)	0.00	14.20	0.01	0.26	1804.09	39.53
Caxias	Useful volume (%)	1.47	105.11	68.97	20.59	−0.18	−0.72
	Inflow (m^3/s)	1.00	19 974.00	1971.27	1903.60	16.57	3.39
	Outflow (m^3/s)	0.00	20 177.00	1954.21	1909.05	17.86	3.47
	ENA (MWmed)	193.69	12 079.15	3971.47	1847.11	1.71	1.01
	Precipitation (mm)	0.00	28.40	0.01	0.31	4180.77	55.84
Foz do Chapecó	Useful volume (%)	0.00	109.14	49.44	16.13	0.01	0.24
	Inflow (m^3/s)	0.00	29 106.00	2155.62	2717.11	23.73	4.06
	Outflow (m^3/s)	0.00	28 966.00	2155.92	2723.11	23.68	4.05
	ENA (MWmed)	663.33	16 299.74	3600.48	2430.62	4.66	1.84
	Precipitation (mm)	0.00	41.60	0.01	0.43	5430.12	66.11
Fundão	Useful volume (%)	0.00	526.11	89.05	66.87	7.42	2.36
	Inflow (m^3/s)	0.00	1542.00	138.58	149.43	22.83	3.89
	Outflow (m^3/s)	0.00	1552.00	138.29	149.57	22.55	3.85
	ENA (MWmed)	193.69	12 079.15	3971.17	1847.07	1.71	1.01
	Precipitation (mm)	0.00	30.20	0.01	0.31	5166.51	61.60
Garibaldi	Useful volume (%)	10.88	264.63	81.46	40.25	0.57	0.92
	Inflow (m^3/s)	1.00	14 787.00	481.30	533.72	43.00	4.22
	Outflow (m^3/s)	7.00	5444.00	479.97	518.05	15.67	3.18
	ENA (MWmed)	663.33	16 299.74	3600.17	2431.94	4.66	1.84
	Precipitation (mm)	0.00	36.20	0.01	0.32	8755.88	82.67

complicate model generalization and necessitate robust deep-learning approaches for accurate spatiotemporal forecasting.

4. The proposed Seq2SeqLatentGNN

The methodology of the proposed Seq2SeqLatentGNN model is presented in this section. The model's main components are explained in Section 4.1, followed by a detailed explanation of how a sequence-to-sequence architecture is applied in Section 4.2. Additional details regarding the model's properties are presented in Section 4.3. The baseline models for comparison are briefly presented in Section 4.4. How features are utilized is explained in Section 4.5, and the process of detrending time series is conducted in Section 4.6. Finally, the procedure for hyperparameter tuning is explained in Section 4.7.

Fig. 2 illustrates the high-level pipeline of the proposed Seq2SeqLatentGNN methodology for multi-node spatiotemporal forecasting. The pipeline is structured into five key stages: data input, preprocessing, neural network processing, optimization (tuning), and evaluation. Within this framework, the neural component is comprised of a Fourier-based spectral layer, a latent graph convolution module that includes attention-based edge inference, and a sequence-to-sequence LSTM framework enhanced by teacher forcing and output smoothing. This specific architecture is designed to capture both global temporal patterns and the evolving spatial correlations present in hydrographic systems.

Seq2SeqLatentGNN is a hybrid deep learning architecture that integrates spectral domain processing, dynamically learnable graph structures, and sequence-to-sequence modeling for spatiotemporal prediction tasks. Unlike conventional GNNs that operate on static graph structures, this model dynamically infers the relationships between nodes through attention mechanisms while leveraging frequency domain transformations to capture global patterns in multivariate time series data. A comprehensive explanation of the Seq2SeqLatentGNN architecture includes the following blocks:

1. **Input Features** — The model processes multivariate time series data by representing each variable as a node within a graph structure. These input features are then fed into the encoder for representation learning.
2. **Encoder Sequence** — The encoder sequentially processes the input using LSTM units to capture temporal dependencies in the data. During training, Teacher Forcing is applied by feeding the ground truth from previous time steps into the decoder, enhancing training stability and convergence.
3. **Latent Graph Structure Learning** — The model dynamically infers a latent graph structure that captures the underlying relationships between nodes (i.e., variables in the multivariate time series). This data-driven approach allows the model to learn complex interactions that are not predefined. To achieve this, learned edge weights, denoted as E_1, E_2, and E_3, are computed through an attention mechanism, which assigns importance scores to the connections between nodes. Unlike traditional static graph structures, these edge weights are adaptive, evolving during training to reflect context-specific dependencies. As a result, the latent correlation structure encoded by the inferred graph uncovers hidden interdependencies among variables. These latent relationships, not explicitly present in the input data, enhance the model's ability to capture intricate dynamics and improve forecasting accuracy.
4. **Spectral Domain Processing Based on Fourier Layer** — The graph structure and associated node features are projected into

Table 3
Statistical summary of hydroelectric reservoir variables (part 2 of 3).

Reservoir	Variable	Minimum	Maximum	Mean	Standard deviation	Kurtosis	Skewness
Gbmrn	Useful volume (%)	33.31	100.19	83.67	18.40	−0.04	−1.06
	Inflow (m^3/s)	0.00	6286.00	904.26	829.03	8.52	2.71
	Outflow (m^3/s)	0.00	5303.00	925.00	862.68	7.18	2.40
	ENA (MWmed)	193.69	12 079.15	3971.45	1847.16	1.71	1.01
	Precipitation (mm)	0.00	60.40	0.01	0.48	12 860.35	105.04
Itá	Useful volume (%)	25.14	100.00	76.55	16.77	−0.01	−0.72
	Inflow (m^3/s)	0.00	25 341.00	1659.68	2134.21	34.08	4.81
	Outflow (m^3/s)	0.00	25 732.00	1660.48	2109.53	33.95	4.76
	ENA (MWmed)	663.33	16 299.74	3599.51	2430.67	4.66	1.84
	Precipitation (mm)	0.00	19.60	0.01	0.27	2748.02	48.27
Jordão	Useful volume (%)	29.70	129.41	83.24	22.03	−0.92	−0.54
	Inflow (m^3/s)	1.00	2035.00	164.12	169.41	27.98	4.35
	Outflow (m^3/s)	0.00	2009.00	60.20	148.58	41.31	5.69
	ENA (MWmed)	193.69	12 079.15	3971.52	1847.18	1.71	1.01
	Precipitation (mm)	0.00	30.20	0.01	0.31	5164.42	61.59
Machadinho	Useful volume (%)	4.00	100.00	69.83	23.10	−0.21	−0.70
	Inflow (m^3/s)	0.00	18 478.00	1204.63	1429.86	38.55	4.97
	Outflow (m^3/s)	0.00	17 306.00	1207.41	1383.18	41.18	5.11
	ENA (MWmed)	663.33	16 299.74	3599.96	2429.41	4.65	1.84
	Precipitation (mm)	0.00	18.40	0.01	0.25	2640.29	46.30
Monjolinho	Useful volume (%)	0.00	283.30	73.33	37.42	0.42	0.83
	Inflow (m^3/s)	0.00	4478.00	158.42	240.02	48.79	5.25
	Outflow (m^3/s)	0.00	4351.00	156.88	238.37	46.56	5.13
	ENA (MWmed)	663.33	16 299.74	3599.26	2430.90	4.66	1.84
	Precipitation (mm)	0.00	26.00	0.01	0.27	4757.36	59.15
Osório	Useful volume (%)	84.92	100.28	93.89	2.54	−0.41	−0.15
	Inflow (m^3/s)	0.00	13 812.00	1500.89	1428.52	12.96	3.12
	Outflow (m^3/s)	0.00	13 995.00	1495.07	1429.17	13.21	3.15
	ENA (MWmed)	193.69	12 079.15	3971.51	1847.07	1.71	1.01
	Precipitation (mm)	0.00	40.20	0.02	0.56	2513.42	45.56
Passo Fundo	Useful volume (%)	75.49	99.68	87.31	7.09	−1.34	0.04
	Inflow (m^3/s)	0.00	3918.00	127.66	167.28	29.06	3.30
	Outflow (m^3/s)	0.00	1672.00	95.60	132.36	20.11	3.61
	ENA (MWmed)	663.33	16 299.74	3599.70	2430.89	4.66	1.84
	Precipitation (mm)	0.00	23.20	0.01	0.34	3115.82	52.33

Table 4
Statistical summary of hydroelectric reservoir variables (part 3 of 3).

Reservoir	Variable	Minimum	Maximum	Mean	Standard deviation	Kurtosis	Skewness
Quebra Queixo	Useful volume (%)	4.62	143.44	73.68	28.96	−1.00	−0.17
	Inflow (m^3/s)	0.00	9622.00	120.33	189.90	465.27	14.05
	Outflow (m^3/s)	0.00	3080.00	115.12	145.20	29.72	4.23
	ENA (MWmed)	663.33	16 299.74	3605.39	2431.68	4.67	1.84
	Precipitation (mm)	0.00	16.20	0.01	0.25	2255.46	44.01
Santa Clara	Useful volume (%)	24.14	117.88	80.01	23.24	−0.99	−0.60
	Inflow (m^3/s)	0.00	1724.00	129.55	148.26	30.74	4.75
	Outflow (m^3/s)	0.00	1675.00	130.46	146.59	28.24	4.32
	ENA (MWmed)	193.69	12 079.15	3971.45	1847.00	1.71	1.01
	Precipitation (mm)	0.00	30.20	0.01	0.31	5165.73	61.60
Santiago	Useful volume (%)	57.58	100.05	89.75	9.99	1.57	−1.36
	Inflow (m^3/s)	0.00	12 155.00	1439.05	1343.02	12.60	3.07
	Outflow (m^3/s)	0.00	11 991.00	1433.14	1320.08	11.77	2.98
	ENA (MWmed)	193.69	12 079.15	3971.52	1846.91	1.71	1.01
	Precipitation (mm)	0.00	23.60	0.01	0.32	2451.01	45.48
São Roque	Useful volume (%)	27.30	128.27	87.30	21.79	−0.02	−0.88
	Inflow (m^3/s)	0.00	18 630.00	394.64	466.78	166.97	6.93
	Outflow (m^3/s)	0.00	3556.00	392.68	405.25	10.06	2.76
	ENA (MWmed)	663.33	16 299.74	3599.58	2430.84	4.66	1.84
	Precipitation (mm)	0.00	36.20	0.01	0.32	8764.76	82.71
Segredo	Useful volume (%)	43.88	102.29	85.34	14.94	0.25	−1.22
	Inflow (m^3/s)	0.00	8431.00	1245.72	1060.00	7.37	2.39
	Outflow (m^3/s)	0.00	8027.00	1220.47	1028.32	9.31	2.72
	ENA (MWmed)	193.69	12 079.15	3971.08	1847.08	1.71	1.01
	Precipitation (mm)	0.00	30.20	0.01	0.31	5166.25	61.60

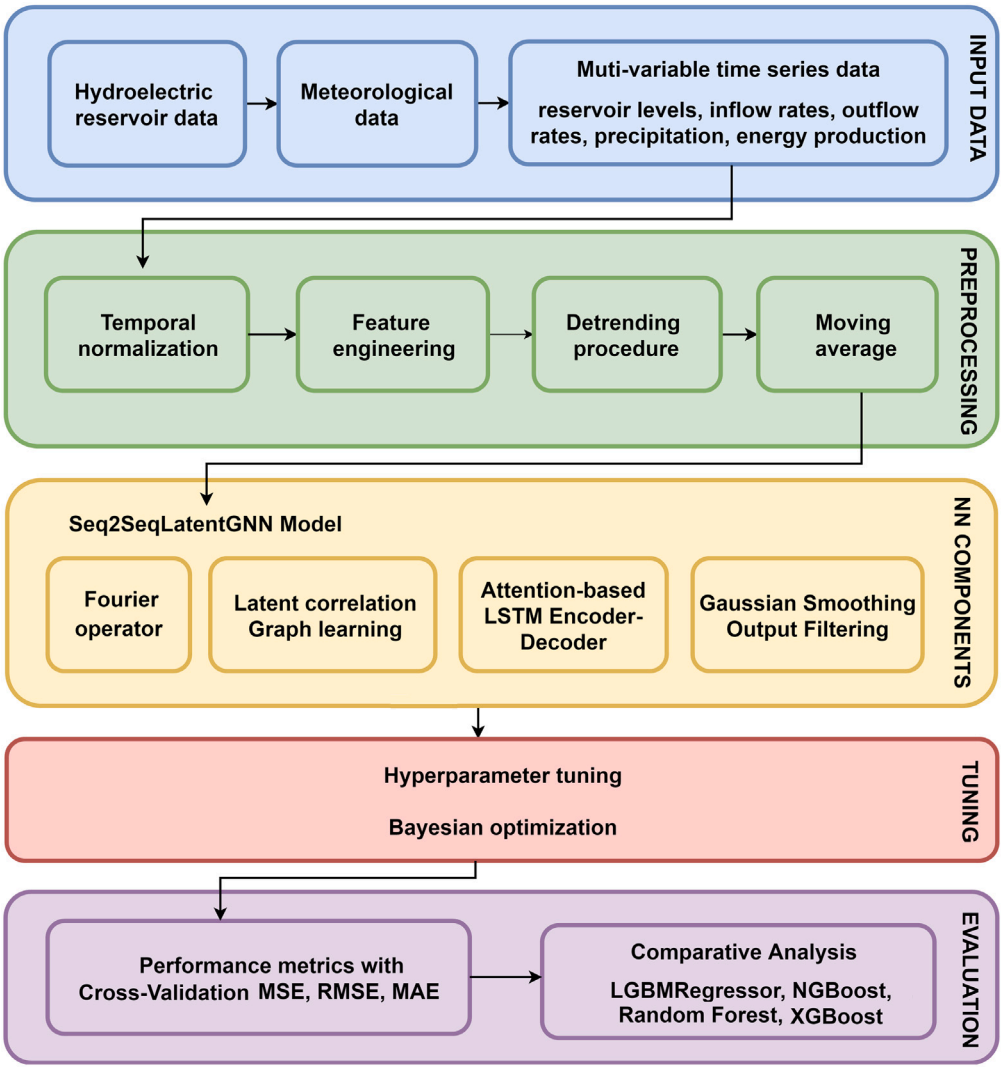

Fig. 2. The Seq2SeqLatentGNN architecture pipeline for multi-node spatiotemporal forecasting in hydroelectric reservoir systems.

the spectral domain via a Fourier Layer. This transformation leverages graph Fourier transforms to decompose graph signals into their global frequency components. By operating in the spectral domain, the model is better equipped to capture long-range dependency patterns inherent in the data, thereby enhancing its ability to model complex temporal dynamics.

5. **Graph Layers** — The model performs graph convolution operations in the spectral domain to aggregate and propagate information across the nodes, capturing complex relationships between them. Following this, the processed features are transformed back into the spatial domain (if required) for subsequent processing, ensuring that relevant information is preserved and further refined.
6. **Decoder Sequence** — The decoder generates predictions iteratively, utilizing LSTM units to model the temporal dependencies. It leverages the latent graph structure and the processed features from the encoder to produce accurate outputs. To enhance prediction stability and smoothness, Gaussian smoothing may be applied to the decoder's intermediate outputs, mitigating noise and improving the overall quality of the predictions.
7. **Teacher Forcing** - During training, the model is fed the ground truth output from the previous time step as input for the current step, rather than using its own predicted output. This approach helps the model learn more effectively by providing the correct context, accelerating convergence, and reducing compounding errors during training.
8. **Output Layer** — A fully connected (FC) output layer maps the decoder's hidden states to the final predictions (e.g., future time steps). This layer produces the model's spatiotemporal forecasts, capturing both the temporal evolution and spatial dependencies in the data.

Fig. 3 presents the architectural blueprint of the proposed Seq2SeqLatentGNN model, which integrates graph processing, dynamic latent graph learning, and sequence-to-sequence forecasting. Let the input multivariate time series be denoted as $\mathbf{X} \in \mathbb{R}^{B\times T\times N\times d}$, where B is the batch size, T is the temporal sequence length, N is the number of spatial nodes (reservoirs), and d is the number of features per node. Each node $v_i \in V$ represents a reservoir, and spatial dependencies are encoded in a time-varying latent graph $G_t = (V, E_t)$, where E_t is inferred through an attention-based mechanism.

The architecture consists of an encoder–decoder sequence using LSTM units, augmented with Fourier-based spectral filtering via a custom Fourier layer Φ_Θ, and a latent graph learning operator $\mathcal{L}_\Psi$ that constructs a dynamic adjacency matrix without requiring a predefined topology. Graph convolution is performed in the spectral domain using graph Fourier transforms, while temporal dependencies are modeled autoregressively with attention and teacher forcing. A Gaussian smoothing operator $S_{\gamma,\omega}$ is applied to enhance output stability.

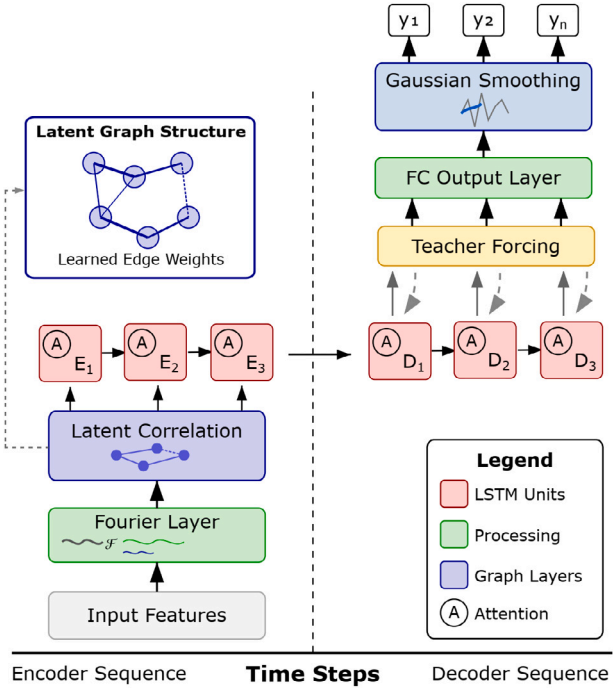

Fig. 3. Seq2SeqLatentGNN architecture.

4.1. Core components

The Seq2SeqLatentGNN architecture integrates core components, each governed by specialized operators. These operators are detailed in the following subsections.

Spectral Filtering Operator: Consider that $\mathcal{F} : \mathbb{R}^{T\times N} \to \mathbb{C}^{T\times N}$ denote the two-dimensional Fourier transform and $\mathcal{F}^{-1}$ its inverse (Li et al., 2021). Define a parameterized spectral filtering operator Φ_Θ as:

$$\Phi_\Theta(\mathbf{X}) = \mathcal{F}^{-1}(\mathbf{R}_\Theta \cdot \mathcal{F}(\mathbf{X})) \tag{1}$$

where $\mathbf{R}_\Theta$ is a complex-valued linear transformation in the frequency domain parameterized by Θ.

For computational efficiency, employ a low-rank approximation by restricting $\mathbf{R}_\Theta$ to operate on a subset of frequency modes:

$$\mathbf{R}_\Theta(\hat{\mathbf{X}})_{i,j} = \begin{cases} \hat{\mathbf{X}}_{i,j} \cdot \mathbf{W}_{i,j} & \text{if } i < \kappa_1, j < \kappa_2 \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

where $\hat{\mathbf{X}} = \mathcal{F}(\mathbf{X})$, $\mathbf{W}_{i,j} \in \mathbb{C}^{d_{\text{in}}\times d_{\text{out}}}$ is a complex-valued weight matrix for frequency mode (i, j), and $\kappa_1, \kappa_2 \in \mathbb{N}^+$ are hyperparameters controlling the number of frequency modes considered.

Latent Graph Learning Operator: Define a latent graph learning operator $\mathcal{L}_\Psi : \mathbb{R}^{B\times N\times d} \to \mathbb{R}^{B\times N\times d'} \times \mathbb{R}^{B\times N\times N}$ parameterized by Ψ that dynamically infers graph structure through attention mechanisms.

Given node features $\mathbf{X} \in \mathbb{R}^{B\times N\times d}$, compute query-key–value transformations as:

$$\mathbf{Q} = \mathbf{X}\mathbf{W}_q, \quad \mathbf{K} = \mathbf{X}\mathbf{W}_k, \quad \mathbf{V} = \mathbf{X}\mathbf{W}_v \tag{3}$$

where $\mathbf{W}_q, \mathbf{W}_k, \mathbf{W}_v \in \mathbb{R}^{d\times d'}$ are learnable projection matrices.

The dynamic adjacency matrix is constructed as:

$$\mathbf{A}_{\text{dyn}} = \frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d'}} + \mathbf{A}_{\text{static}} \tag{4}$$

where $\mathbf{A}_{\text{static}} \in \mathbb{R}^{N\times N}$ is an optional learnable static adjacency matrix initialized as:

$$\mathbf{A}_{\text{static}} = \alpha \cdot \mathbf{1}_{N\times N} + (1-\alpha)\cdot \mathbf{I}_N \tag{5}$$

with $\alpha \in [0, 1]$ controlling the initialization bias between self-connections and a fully connected structure.

To enforce sparsity, apply a k-nearest neighbors mask:

$$\mathbf{A}_{\text{sparse}}(i, j) = \begin{cases} \mathbf{A}_{\text{dyn}}(i, j) & \text{if } j \in \mathcal{N}_k(i) \\ -\infty & \text{otherwise} \end{cases} \tag{6}$$

where $\mathcal{N}_k(i)$ denotes the set of k nodes with highest attention scores with respect to node i.

The normalized message passing ($\mathbf{M}$) is computed as:

$$\mathbf{M} = \text{softmax}\left(\frac{\mathbf{A}_{\text{sparse}}}{\tau}\right)\mathbf{V} \tag{7}$$

where $\tau \in \mathbb{R}^+$ is a temperature parameter controlling the sharpness of the attention distribution.

The feature transformation is then defined as:

$$\mathcal{L}_\Psi(\mathbf{X}) = (\Gamma(\mathbf{M}) + \mathbf{V}, \text{softmax}(\mathbf{A}_{\text{sparse}})) \tag{8}$$

where $\Gamma : \mathbb{R}^{B\times N\times d'} \to \mathbb{R}^{B\times N\times d'}$ is a transformation function defined as:

$$\Gamma(\mathbf{M}) = \mathbf{W}_2 \cdot \sigma(\text{LN}(\mathbf{W}_1 \cdot \mathbf{M})) \tag{9}$$

with $\mathbf{W}_1, \mathbf{W}_2 \in \mathbb{R}^{d'\times d'}$ as learnable parameters, σ as a nonlinear activation function, and LN as layer normalization.

Gaussian Smoothing Operator: Define a Gaussian smoothing operator $\mathcal{S}_{\gamma,\omega}$ with kernel width $\omega \in \mathbb{N}^+$ and variance parameter $\gamma \in \mathbb{R}^+$. For an input tensor $\mathbf{X} \in \mathbb{R}^{B\times T\times N}$, the smoothing operation is:

$$\mathcal{S}_{\gamma a,\omega(\mathbf{X})=\mathbf{X}*\mathbf{g}_{\gamma,\omega}} \tag{10}$$

where $*$ denotes convolution and $\mathbf{g}_{\gamma,\omega}$ is the Gaussian kernel defined as:

$$\mathbf{g}_{\gamma,\omega}(i) = \frac{1}{\sqrt{2\pi\gamma^2}} \exp\left(-\frac{i^2}{2\gamma^2}\right), \quad i \in \{-\lfloor\omega/2\rfloor, \ldots, \lfloor\omega/2\rfloor\} \tag{11}$$

normalized such that $\sum_i \mathbf{g}_{\gamma,\omega}(i) = 1$.

4.2. Sequence-to-sequence architecture

The encoder, decoder, and teacher forcing approaches are described in this subsection.

Encoder: Let $\mathbf{X} \in \mathbb{R}^{B\times T_{\text{in}}\times N\times d}$ be the input time series for a batch of B samples, across T_{in} timesteps, N nodes, and d features. The spectral processing is applied as:

$$\mathbf{X}' = \Phi_\Theta(\mathbf{X}) + \mathbf{X} \tag{12}$$

For each timestep $t \in \{1, 2, \ldots, T_{\text{in}}\}$, temporal-node processing is performed:

$$(\mathbf{H}_t^{(1)}, \mathbf{A}_t^{(1)}) = \mathcal{L}_{\Psi_1}(\mathbf{X}'_t) \tag{13}$$

$$(\mathbf{H}_t^{(2)}, \mathbf{A}_t^{(2)}) = \mathcal{L}_{\Psi_2}(\mathbf{H}_t^{(1)}) \tag{14}$$

where $\mathbf{X}'_t \in \mathbb{R}^{B\times N\times d}$ are features at timestep t.

The sequence $\mathbf{H} = [\mathbf{H}_1^{(2)}, \mathbf{H}_2^{(2)}, \ldots, \mathbf{H}_{T_{\text{in}}}^{(2)}] \in \mathbb{R}^{B\times T_{\text{in}}\times N\times d'}$ is processed through an LSTM for temporal encoding:

$$(\mathbf{h}_{T_{\text{in}}}, \mathbf{c}_{T_{\text{in}}}) = \text{LSTM}_{\text{enc}}(\mathbf{H}) \tag{15}$$

where $\mathbf{h}_{T_{\text{in}}}, \mathbf{c}_{T_{\text{in}}} \in \mathbb{R}^{L\times B\times N\times d'}$ are the final hidden and cell states, and L is the number of LSTM layers.

Decoder with Attention: For each timestep $t \in \{1, 2, \ldots, T_{\text{out}}\}$ in the prediction horizon, the recurrent state is updated:

$$\mathbf{h}_t, (\mathbf{h}_t, \mathbf{c}_t) = \text{LSTM}_{\text{dec}}(\mathbf{z}_{t-1}, (\mathbf{h}_{t-1}, \mathbf{c}_{t-1})) \tag{16}$$

where $\mathbf{z}_0 = \mathbf{H}_{T_{\text{in}}}^{(2)}$ and $(\mathbf{h}_0, \mathbf{c}_0) = (\mathbf{h}_{T_{\text{in}}}, \mathbf{c}_{T_{\text{in}}})$. Temporal attention is computed as:

$$\boldsymbol{\alpha}_t = \text{softmax}\left(\frac{\mathbf{h}_t \cdot \mathbf{H}^T}{\sqrt{d'}}\right) \tag{17}$$

$$\mathbf{c}_t = \boldsymbol{\alpha}_t \cdot \mathbf{H} \tag{18}$$

The output is generated by:

$$\mathbf{u}_t = [\mathbf{h}_t; \mathbf{c}_t] \tag{19}$$

$$\mathbf{z}_t = \mathbf{W}_{\text{out}} \cdot \mathbf{W}_{\text{fc}} \cdot \mathbf{u}_t \tag{20}$$

where $[\cdot;\cdot]$ denotes concatenation along the feature dimension (Buratto et al., 2024).

Maintain a sliding window of recent predictions $\mathbf{Z}_t = [\mathbf{z}_{t-\nu+1}, \ldots, \mathbf{z}_t] \in \mathbb{R}^{B\times\nu\times N}$ for some $\nu \in \mathbb{N}^+$, and apply a convolutional smoother C:

$$\mathbf{z}'_t = C(\mathbf{Z}_t) \tag{21}$$

The final predictions are smoothed via the Gaussian operator:

$$\mathbf{Y}_{\text{pred}} = \mathcal{S}_{\gamma,\omega}([\mathbf{z}'_1, \mathbf{z}'_2, \ldots, \mathbf{z}'_{T_{\text{out}}}]) \tag{22}$$

Teacher Forcing: During training, teacher forcing is applied with probability $p_{\text{tf}} \in [0, 1]$:

$$\mathbf{z}_t^{\text{next}} = \begin{cases} \mathbf{y}_t & \text{with probability } p_{\text{tf}} \\ \mathbf{z}_t & \text{with probability } 1 - p_{\text{tf}} \end{cases} \tag{23}$$

$$\mathbf{z}_t = \mathbf{W}_{\text{v2h}} \cdot \mathbf{z}_t^{\text{next}} \tag{24}$$

where $\mathbf{y}_t \in \mathbb{R}^{B\times N}$ is the ground truth for timestep t and $\mathbf{W}_{\text{v2h}} \in \mathbb{R}^{1\times d'}$ is a learnable projection matrix.

Algorithm 1 outlines the complete forward pass of the proposed Seq2SeqLatentGNN model, which integrates dynamic graph learning, spectral filtering, and autoregressive sequence modeling for spatiotemporal forecasting. The algorithm begins by applying a Fourier-based encoder $\boldsymbol{\Phi}_\Theta$ to extract global temporal patterns from the input tensor $\mathbf{X} \in \mathbb{R}^{B\times T_{\text{in}}\times N\times d}$. Subsequently, the latent graph learning operator $\mathcal{L}_\Psi$ infers dynamic adjacency matrices $\mathbf{A}_t \in \mathbb{R}^{N\times N}$ and transforms node features via attention-weighted message passing. The encoded sequence is then passed through a multi-layer LSTM encoder, producing a fixed-length hidden representation.

During decoding, the model iteratively generates predictions using the LSTM decoder, where attention mechanisms over the encoded sequence compute temporal context vectors to guide each output step. Teacher forcing is optionally employed, with probability $p_{\text{tf}} \in [0, 1]$, to balance between ground truth and model-generated inputs. A Gaussian smoothing operator $\mathcal{S}_{\gamma,\omega}$ and a convolutional temporal filter are applied to stabilize the predictions and suppress high-frequency noise.

Algorithm 1: Seq2SeqLatentGNN: Graph neural network with dynamic structure learning for spatiotemporal forecasting.

Input: Node features $\mathbf{X} \in \mathbb{R}^{B\times T_{\text{in}}\times N\times d}$
Input: Optional targets $\mathbf{Y} \in \mathbb{R}^{B\times T_{\text{out}}\times N}$ for teacher forcing
Input: Hyperparameters: teacher forcing probability p_{tf}
Output: Predictions $\mathbf{Y}_{\text{pred}} \in \mathbb{R}^{B\times T_{\text{out}}\times N}$, attention matrix $\mathbf{A} \in \mathbb{R}^{B\times N\times N}$

```
/* Encoder: Process input sequence                          */
X ← FourierEncoder(X) + X;
/* Process time steps through latent graph layers           */
H, A_list ← TemporalLatentGNN(X);
/* Encode sequence with LSTM                                */
Z_0, (h_T, c_T) ← LSTM_enc(H);
/* Decoder: Generate predictions autogregressively          */
Z ← Z_0 ;                                  // Initial decoder input
last_preds ← [] ;                          // Window for smoothing
for t = 1 to T_out do
    /* Generate prediction with attention                   */
    h_dec, (h_T, c_T) ← LSTM_dec(Z, (h_T, c_T));
    ctx ← AttentionMechanism(h_dec, H);
    pred_t ← OutputProjection(h_dec, ctx);
    /* Apply temporal smoothing                             */
    Append pred_t to last_preds;
    pred_t ← TemporalSmoothing(last_preds);
    /* Teacher forcing or autoregressive input              */
    next ← random() < p_tf ? Y[:, t] : pred_t;
    Z ← ValueToHidden(next);
end
/* Collect and finalize predictions                         */
Y_pred ← Collect and smooth all pred_t;
return Y_pred, A_list[−1];
```

4.3. Seq2SeqLatentGNN properties

The Seq2SeqLatentGNN architecture introduces innovations for spatiotemporal modeling. First, it learns temporal relationships between nodes without requiring predefined graph structures. The attention mechanism implicitly constructs adjacency matrices that can adapt to changing node relationships, making it particularly suitable for systems with dynamic interactions.

The architecture integrates local interactions (via the latent graph operator $\mathcal{L}_\Psi$, given by Eq. (8)) with global patterns (via the spectral filtering operator $\boldsymbol{\Phi}_\Theta$, calculated according to Eq. (1)). This multi-scale approach allows the model to capture both fine-grained node-level dependencies and system-wide patterns. The attention mechanism in the decoder allows the model to adaptively focus on relevant historical timesteps, effectively creating dynamic temporal receptive fields that can adapt to varying temporal dependencies (Wen and Li, 2023). Both input and output sequences are subjected to Gaussian smoothing operators, which serve as implicit regularizers by enforcing temporal consistency and reducing the impact of noise. This is particularly important for forecasting tasks where stability and generalization are critical.

The Seq2SeqLatentGNN architecture thus presents a theoretically grounded framework for spatiotemporal forecasting that adaptively models complex patterns in multivariate time series without prior knowledge of the underlying graph structure. By combining spectral filtering, dynamic graph learning, and sequence modeling with appropriate regularization, the model offers a versatile approach for systems where relationships between variables evolve over time.

4.4. Baseline models

A comprehensive set of experiments was conducted to evaluate the effectiveness of the proposed Seq2SeqLatentGNN framework. Its performance was benchmarked against four baseline models: Light gradient boosting machine (LGBMRegressor), natural gradient boosting (NGBoost), random forest, and XGBoost.

LGBMRegressor is a gradient boosting framework that implements histogram-based algorithms to accelerate the training process while reducing memory consumption. By adopting a leaf-wise growth strategy, as opposed to the conventional level-wise approach, it achieves faster convergence and often superior predictive accuracy, particularly when applied to large-scale datasets (Janizadeh et al., 2024). Furthermore, its native support for categorical feature handling and its inherent robustness to overfitting render it a suitable choice for time series forecasting and regression tasks in which computational efficiency is a primary requirement.

NGBoost is a probabilistic prediction framework that integrates gradient boosting with explicit uncertainty quantification. Unlike conventional approaches that produce deterministic point estimates, NGBoost predicts the parameters of a specified probability distribution (e.g., Gaussian), thereby enabling the direct estimation of predictive uncertainty (Hu et al., 2024). The adoption of natural gradients facilitates more stable optimization, which enhances performance in applications demanding reliable probabilistic forecasts, including risk assessment and decision-making under uncertainty.

Random Forest is an ensemble-based learning technique that constructs a collection of decision trees and combines their outputs through aggregation strategies, such as averaging for regression tasks or majority voting for classification tasks. By incorporating stochasticity into both feature selection and bootstrap sampling, random forest reduces the risk of overfitting and enhances generalization performance relative to individual decision trees (Coelho et al., 2024). Its resilience to noisy data, coupled with its capacity to model intricate feature interactions, renders it a robust and adaptable approach for predictive modeling, particularly in high-dimensional problem domains.

XGBoost is an optimized gradient boosting framework, designed to achieve high scalability and computational efficiency (Alsulamy, 2025). The algorithm incorporates explicit regularization terms to mitigate overfitting and supports parallelized computation, thereby enabling effective processing of large-scale datasets. Furthermore, its capability to intrinsically manage missing values, combined with advanced tree pruning strategies, reinforces its status as a state-of-the-art approach for predictive modeling.

4.5. Feature engineering

The first phase of the feature engineering process involves the normalization of temporal data to ensure consistency and chronological ordering. Thus, the timestamp variable is converted to a standardized format and establishes a chronological sequence for each reservoir, such as:

$$D_r = \{(t_1, x_1), (t_2, x_2), \ldots, (t_n, x_n)\}, \forall r \in R, \tag{25}$$

where R is the set of all reservoirs, D_r is the time-ordered dataset for reservoir r, t_i is the timestamp, and x_i is the corresponding measurements at time t_i, such that $t_i < t_{i+1}$ for all $i \in \{1, 2, \ldots, n-1\}$.

To capture short-term precipitation patterns while reducing noise in the measurements, a 24-hour moving average transformation which is given by Eq. (26) for the precipitation variable was implemented. In this context, for each reservoir r and timestamp t, the moving average is calculated as

$$\bar{P}^{24}_{r,t} = \frac{1}{\min(24, |W_{r,t}|)} \sum_{i \in W_{r,t}} P_{r,i} \tag{26}$$

where $\bar{P}^{24}_{r,t}$ is the 24-hour moving average precipitation at reservoir r and time t, $W_{r,t} = \{i : t - 23\Delta t \leq t_i \leq t \text{ and } i \in D_r\}$ is the temporal window up to 24 preceding hourly measurements, $|W_{r,t}|$ is the cardinality of the window (number of available measurements), $P_{r,i}$ is the precipitation measurement at the reservoir r and time t_i, and Δt is 1 h time interval between consecutive measurements.

The inclusion of $\min(24, |W_{r,t}|)$ allows for the calculation of the moving average even when fewer than 24-hour measurements are available, particularly at the beginning of the time series for each reservoir. For the ENA variable, a longer 72-hour moving average is used to capture more persistent patterns in energy generation potential. The ENA moving average at the reservoir r and time t is defined as

$$\bar{E}^{72}_{r,t} = \frac{1}{72} \sum_{i \in W^{72}_{r,t}} E_{r,i} \tag{27}$$

where $\bar{E}^{72}_{r,t}$ is the 72-hour moving average of ENA at reservoir r and time t, $W^{72}_{r,t} = \{i : t - 71\Delta t \leq t_i \leq t \text{ and } i \in D_r\}$ is the 72-hour temporal window, $E_{r,i}$ is the ENA measurement (in MWmed) at reservoir r and time t_i.

In contrast to the precipitation moving average, enforce a strict requirement of 72-hour of available measurements for the ENA moving average calculation by setting `min_periods= 72`. This constraint ensures that the ENA moving average reflects a complete three-day trend, eliminating potential bias from incomplete windows.

Following the feature engineering process, apply a filtering operation to remove data points where the 72-hour ENA moving average could not be computed (primarily, the first 72-hour of measurements for each reservoir):

$$D^{filtered}_r = \{(t, x) \in D_r : \bar{E}^{72}_{r,t} \text{ is defined}\} \tag{28}$$

This filtering step, from Eq. (28), ensures that all data points used in the subsequent analysis have well-defined values for engineered features, maintaining the integrity of the dataset for statistical modeling and inference. The resulting dataset incorporates these engineered features, providing a more robust representation of the temporal dynamics within the reservoir system, while reducing the influence of transient fluctuations in the raw measurements.

4.6. Detrending

To isolate and analyze the cyclical and seasonal components of the reservoir data, a detrending methodology that removes long-term trends while preserving meaningful oscillatory patterns was implemented, obtained by the detrending calculated in Eq. (29). The proposed approach combines multiple smoothing techniques with dynamic weighting to adapt to local signal characteristics.

Consider $x(t)$ as the original time series of reservoir measurements (volume or inflow) where $t \in \{1, 2, \ldots, N\}$ and N is the total number of observations. The detrended signal $y(t)$ is obtained by:

$$y(t) = x(t) - \hat{T}(t) \tag{29}$$

where $\hat{T}(t)$ is the estimated trend component, which is given by Eq. (30). The core innovation in the proposed approach is a hybrid trend estimation that combines multiple techniques to achieve optimal trend extraction.

The trend extraction utilizes a weighted combination of Savitzky-Golay filtering and polynomial regression:

$$\hat{T}(t) = \alpha(t) \cdot T_{SG}(t) + (1 - \alpha(t)) \cdot T_{poly}(t) \tag{30}$$

where $T_{SG}(t)$ is the trend estimated via Savitzky-Golay filtering, calculated by Eq. (31), $T_{poly}(t)$ is the trend from polynomial regression, and $\alpha(t)$ is a dynamic weighting function that adapts to local signal characteristics.

The Savitzky-Golay filter fits local polynomial models to successive windows of the signal:

$$T_{SG}(t) = \sum_{j=-m}^{m} c_j x(t + j) \tag{31}$$

where $m = \lfloor w/2 \rfloor$ is half the window size, with the full window size w defined as:

$$w = \max(2p + 1, \lfloor \beta \cdot N \rfloor) \tag{32}$$

where p is the polynomial order, N is the signal length, and $\beta \in [0.1, 0.4]$ is the window percentage parameter that controls the smoothness of the trend. The coefficients c_j, from Eq. (31), are derived from a polynomial fit of order p and satisfy the necessary conditions for preserving moments of the signal.

To further enhance smoothness while preserving important trend characteristics, apply additional Gaussian filtering:

$$T^*_{SG}(t) = \sum_{j=-k}^{k} G(j, \sigma) \cdot T_{SG}(t + j) \tag{33}$$

where $G(j, \sigma)$ is the Gaussian kernel with standard deviation σ, calculated by Eq. (34), proportional to the window size and smoothing factor λ, as given:

$$\sigma = \frac{w}{10} \cdot \lambda \tag{34}$$

For capturing long-term trends, employ a low-order polynomial regression:

$$T_{poly}(t) = \sum_{i=0}^{q} a_i t^i \tag{35}$$

where $q \leq 3$ is the polynomial order for long-term trend modeling and the coefficients a_i are determined by minimizing the squared error:

$$\min_{a_0,a_1,\ldots,a_q} \sum_{t=1}^{N} \left(x(t) - \sum_{i=0}^{q} a_i t^i \right)^2 \tag{36}$$

The weighting function $\alpha(t)$, presented in Eq. (37), combines the two trend components with special considerations for signal boundaries, given by:

$$\alpha(t) = \begin{cases} \gamma_1 + (\gamma_2 - \gamma_1) \cdot \frac{t-1}{N_e - 1} & \text{if } 1 \leq t \leq N_e \\ \gamma_2 & \text{if } N_e < t < N - N_e \\ \gamma_2 + (\gamma_1 - \gamma_2) \cdot \frac{t-(N-N_e)}{N_e} & \text{if } N - N_e \leq t \leq N \end{cases} \tag{37}$$

where $N_e = \lfloor 0.1 \cdot N \rfloor$ defines the edge region size, $\gamma_1 = 0.1$ is the weight at the extreme edges, and $\gamma_2 = 0.3$ is the weight in the central region. This adaptive weighting mitigates boundary effects while preserving the advantages of both trend estimation methods.

The performance of the proposed detrending methodology is governed by three key parameters: window percentage β from Eq. (32), polynomial order p, and smoothing factor λ, both used in Eq. (34). These parameters were optimized through comparative analysis to achieve a balance between trend smoothness and preservation of meaningful signal characteristics. For the reservoir data in this study, empirically determined optimal values of $\beta = 0.25$, $p = 2$, and $\lambda = 2.0$. The overall procedure is described in Algorithm 2.

Algorithm 2: Adaptive hybrid detrending for reservoir time series data.

Input: Time series $x(t)$ of reservoir measurements where $t \in \{1, 2, \ldots, N\}$
Input: Parameters: window percentage $\beta \in [0.1, 0.4]$, polynomial order p, smoothing factor λ, edge weight $\gamma_1 = 0.1$, center weight $\gamma_2 = 0.3$
Output: Detrended time series $y(t)$

`/* Preprocess to handle missing values */`
Apply linear interpolation to fill missing values in $x(t)$;
`/* Calculate window size for Savitzky-Golay filter */`
$w \leftarrow \max(2p + 1, \lfloor \beta \cdot N \rfloor)$;
$m \leftarrow \lfloor w/2 \rfloor$;
`/* Savitzky-Golay filtering for trend estimation */`
for $t \leftarrow 1$ **to** N **do**
  $T_{SG}(t) \leftarrow \sum_{j=-m}^{m} c_j x(\text{clip}(t + j, 1, N))$;
end
`/* Apply additional Gaussian smoothing */`
$\sigma \leftarrow \frac{w}{10} \cdot \lambda$;
$k \leftarrow \lceil 3\sigma \rceil$;
for $t \leftarrow 1$ **to** N **do**
  $T^*_{SG}(t) \leftarrow \sum_{j=-k}^{k} G(j, \sigma) \cdot T_{SG}(\text{clip}(t + j, 1, N))$;
end
`/* Polynomial regression for long-term trend */`
Solve for coefficients $a_0, a_1, \ldots, a_q$ (where $q \leq 3$) by minimizing: $\sum_{t=1}^{N} \left(x(t) - \sum_{i=0}^{q} a_i t^i \right)^2$;
for $t \leftarrow 1$ **to** N **do**
  $T_{poly}(t) \leftarrow \sum_{i=0}^{q} a_i t^i$;
end
`/* Calculate adaptive weights */`
$N_e \leftarrow \lfloor 0.1 \cdot N \rfloor$;
for $t \leftarrow 1$ **to** N **do**
  if $t \leq N_e$ **then**
    $\alpha(t) \leftarrow \gamma_1 + (\gamma_2 - \gamma_1) \cdot \frac{t-1}{N_e - 1}$;
  else
    if $t < N - N_e$ **then**
      $\alpha(t) \leftarrow \gamma_2$;
    else
      $\alpha(t) \leftarrow \gamma_2 + (\gamma_1 - \gamma_2) \cdot \frac{t-(N-N_e)}{N_e}$;
    end
  end
end
`/* Combine trend estimates */`
for $t \leftarrow 1$ **to** N **do**
  $\hat{T}(t) \leftarrow \alpha(t) \cdot T^*_{SG}(t) + (1 - \alpha(t)) \cdot T_{poly}(t)$;
  $y(t) \leftarrow x(t) - \hat{T}(t)$;
end
return $y(t)$;

4.7. Bayesian hyperparameter optimization

To optimize the hyperparameters of the proposed Seq2SeqLatentGNN model, we adopt a Bayesian optimization strategy implemented via the `Optuna` framework. Let $\mathcal{H} \subset \mathbb{R}^p$ denote the bounded hyperparameter search space, where each point $\boldsymbol{\eta} \in \mathcal{H}$ defines a candidate configuration for the model's parameters (Stefenon et al., 2025).

The goal is to find the optimal configuration $\boldsymbol{\eta}^* \in \mathcal{H}$ that minimizes a non-convex, and potentially noisy objective function $\mathcal{J} : \mathcal{H} \to \mathbb{R}$, which maps each hyperparameter setting to a scalar validation loss:

$$\boldsymbol{\eta}^* = \arg\min_{\boldsymbol{\eta} \in \mathcal{H}} \mathbb{E}_{\mathcal{D}_{\text{val}}} \left[L\left(\hat{f}_{\boldsymbol{\eta}}; \mathcal{D}_{\text{val}} \right) \right], \tag{38}$$

where $\hat{f}_{\boldsymbol{\eta}}$ is the trained model under configuration $\boldsymbol{\eta}$, $\mathcal{D}_{\text{val}}$ denotes the validation dataset, and L is a predefined loss function.

A key design choice in this model is the use of the Huber loss function, which offers robustness to outliers by combining the properties of the ℓ_1 and ℓ_2 norms. For residual $r = y - \hat{y}$, the Huber loss is defined as:

$$L_\delta(r) = \begin{cases} \frac{1}{2} r^2 & \text{if } |r| \leq \delta, \\ \delta(|r| - \frac{1}{2}\delta) & \text{otherwise,} \end{cases} \tag{39}$$

where $\delta > 0$ is the threshold parameter controlling the transition between quadratic and linear penalization. During the optimization process, both the use of the Huber loss (as a boolean decision variable) and the value of $\delta \in [\delta_{\min}, \delta_{\max}] \subset \mathbb{R}_+$ are included in the search space (Tong, 2023).

Each trial $t \in \{1, \ldots, T\}$ proposes a configuration $\boldsymbol{\eta}_t$ based on the observed objective values $\{\mathcal{J}(\boldsymbol{\eta}_s)\}_{s=1}^{t-1}$. The surrogate model guides the sampling toward regions in $\mathcal{H}$ with higher expected improvement. Trials are optionally pruned based on early-stopping criteria to reduce computation in low-potential regions of the search space. This optimization process provides a principled approach to identifying performant hyperparameter configurations, particularly suited for high-dimensional and expensive objective functions, such as those arising from training deep neural networks (Muniz et al., 2025).

4.8. Evaluation metrics

In the evaluation stage, we assess the model's performance using standard metrics. The performance metrics considered are the mean squared error (MSE), root mean squared error (RMSE), and mean absolute error (MAE), which are defined in Eq. (40), Eq. (41), and Eq. (42), respectively.

The MSE is a common loss function used to measure the average squared difference between the actual values and the predicted values (Ribeiro et al., 2022). It is defined as:

$$\text{MSE} = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 \tag{40}$$

where y_i represents the actual values, $\hat{y}_i$ is the predicted values by the models, n the number of observations. Because errors are squared,

large differences between predicted and actual values are penalized more heavily than smaller ones. This makes the MSE highly sensitive to outliers, which can be beneficial in certain applications where large errors must be avoided.

The RMSE quantifies the square root of the average squared differences between the predicted and actual values, emphasizing the larger errors (Coelho et al., 2024):

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}. \tag{41}$$

By squaring the errors before averaging in RMSE, a greater weight is given to larger deviations. The MAE measures the average magnitude of the errors in a set of predictions, without considering their direction (Takara et al., 2024):

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|. \tag{42}$$

The MAE metric can be used in a variety of contexts, particularly when the impact of all errors, regardless of their size, is considered equal.

5. Results and discussion

All experiments were conducted on a high-performance workstation running Gentoo Linux. The computational environment includes an Intel(R) Core(TM) i9-14900K and an NVIDIA GeForce RTX 4090 GPU (graphics processing unit). The system has 32 GiB of random access memory and multiple high-capacity storage volumes, ensuring efficient data handling and model training. The Seq2SeqModel is publicy available at a GitHub repository.[2]

5.1. Detrending analysis

The detrending is applied to each reservoir's time series data independently, accounting for the unique characteristics of each reservoir. Missing or invalid values were handled through linear interpolation before trend extraction. The resulting detrended signals reveal cyclical patterns and anomalies that were previously obscured by long-term trends, enabling a more effective analysis of reservoir behavior across seasonal and annual timeframes. To illustrate the procedure, the detrending methodology is applied to both volume and inflow time series from four representative reservoirs. Figs. 4–6 show the results of this analysis.

Fig. 4 shows the application of the hybrid trend extraction approach on reservoir volume data. The left panels display the original volume time series overlaid with the extracted trend component and the resulting residuals (shaded area). The right panels present the distribution of residuals, which approximates a normal distribution to varying degrees depending on the reservoir's operational characteristics. The trend component explains between 37.0% and 68.9% of the total variance in these volume time series, with Barra Grande exhibiting the highest proportion of variance captured by the trend.

For each reservoir, the left panel in Fig. 4 shows the original signal (blue), extracted trend (red), and residuals (green shaded area). The right panel shows the distribution of residuals after detrending compared to a normal distribution (dashed line). Variance-explained metrics quantify the proportion of total variance captured by the trend component. The analysis uses a hybrid trend extraction approach with a window percentage $\beta = 0.20$.

There is a large fluctuation in the original signal (shown in blue in Fig. 4), which is a natural characteristic of the system due to the variation in the flow of the reservoirs. The trend in this way is an important characteristic to evaluate, as it is the variable that should be assessed in energy planning, and it is shown in red in this plot. This large variance is reflected in high-density values in a residual distribution, which is a reason for preprocessing.

Fig. 5 shows the decomposition of the volume time series, showing how the hybrid trend extraction separates the long-term trend from seasonal oscillations. The variability in trend representation across different reservoirs (from 37.0% to 68.9% of explained variance) indicates heterogeneity in reservoir behavior within the hydroelectric system. In this plot, the seasonal value has its average close to zero, making it clear why it reflects the most significant variation in the trend, which is the focus of this analysis. Considering that the trend and the seasonal values are decompositions of the raw signal, summing up these values results in the original signal.

The original signal (blue) in Fig. 5 is separated into a long-term trend component (red) and a seasonal/residual component (green). The trend component explains between 37.0% and 68.9% of the total variance, with Barra Grande showing the highest variance explained by trend. This decomposition separates cyclical patterns from long-term volumetric evolution.

Fig. 6 extends this analysis to inflow measurements, which exhibit higher volatility and stronger seasonal dependence than volume data. This is reflected in the lower proportion of variance explained by the trend component (between 12.2% and 31.4%). The adaptive window sizing in the Savitzky-Golay filtering approach ($w = \lfloor 0.20 \cdot N \rfloor$) allows for appropriate handling of the higher-frequency fluctuations observed in these measurements.

The trend components capture between 12.2% and 31.4% of total variance, lower than for volume data, reflecting the higher volatility and seasonal dependence of inflow measurements. The adaptive Savitzky-Golay filtering with window size proportional to signal length ($w = \lfloor 0.20 \cdot N \rfloor$) extracts the underlying trends while maintaining the characteristics of the inflow patterns.

These visualizations show that our detrending methodology isolates cyclical patterns from long-term trends across different types of reservoir measurements, supporting subsequent analyses of seasonal behavior and anomalies in the hydroelectric system.

5.2. Optimization of model hyperparameters

The optimization employs a Bayesian (Kim and Choi, 2023) to identify optimal hyperparameters while maintaining robustness to outliers, using Optuna (Akiba et al., 2019). Table 5 presents the hyperparameter search space and optimal values identified in 200 trials. The table presents the normalized importance scores of eleven hyperparameters, with `huber_beta` demonstrating the highest impact (0.44) on model performance. The Huber loss parameter β (`huber_beta`) has the highest impact with an importance score of 0.44, substantially higher than all other parameters. This dominant influence highlights the significance of robust loss functions in handling outliers and non-Gaussian error distributions in spatial–temporal data.

Table 5
Search space and optimized values of the hyperparameters.

Hyperparameter	Search Range [minimum, maximum]	Optimized value	Importance
huber_beta	[0.1, 2.0]	0.85	0.44
teacher_forcing_ratio	[0.0, 1.0]	0.65	0.10
use_huber_loss	{True, False}	True	0.10
grad_clip	[0.1, 5.0]	1.0	0.08
tv_weight	[0.0, 0.1]	0.03	0.07
weight_decay	[1e−6, 1e-4]	1e−5	0.06
k_neighbors	[2, 8]	4	0.05
learning_rate	[1e−4, 1e-2]	5e−4	0.03
l1_weight	[0.0, 1e-3]	1e−4	0.03
hidden_dim	[32, 256]	128	0.02
batch_size	[16, 128]	64	0.01

[2] https://github.com/lseman/Seq2SeqLatentGNN/

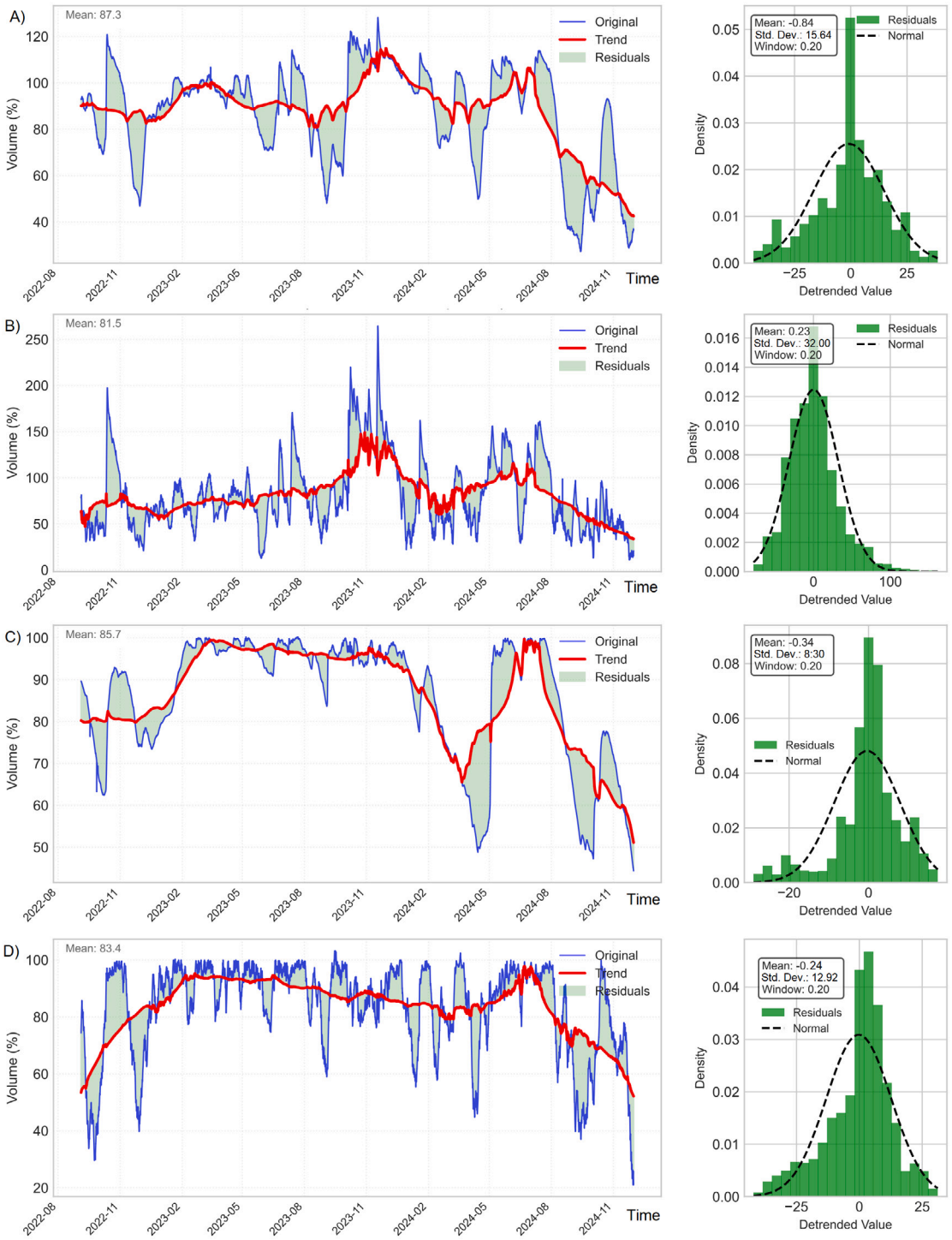

Fig. 4. Component analysis of volume data for four reservoirs in the Brazilian hydroelectric system: (A) Sao Roque (48.7% variance explained), (B) Garibaldi (30.0% variance explained), (C) Barra Grande (68.9% variance explained), (D) Campos Novos (37.0% variance explained).

The optimization results highlight a distinct hierarchy in the importance of different hyperparameters. Some of the hyperparameters had a higher importance, such as `huber_beta`, `teacher_forcing_ratio`, and `use_huber_loss` from the second tier (both 0.10), followed by `grad_clip` (0.08) and `tv_weight` (0.07). Other hyperparameters had moderate importance including `weight_decay` (0.06) and `k_neighbors` (0.05), while `learning_rate`, `l1_weight`, `hidden_dim`, and `batch_size` show relatively minor impact (< 0.04).

The pronounced importance of robust loss functions (`huber_beta`, `use_huber_loss`) reflects the inherent variability in the spatial–temporal dataset. Similarly, the moderate importance of `teacher_forcing_ratio` indicates the benefits of balancing between exposure to ground truth and model predictions during training, preventing error accumulation in sequential prediction tasks. The optimized configuration achieves an 18.2% reduction in validation RMSE compared to the initial parameterization, demonstrating significant performance gains through systematic hyperparameter tuning.

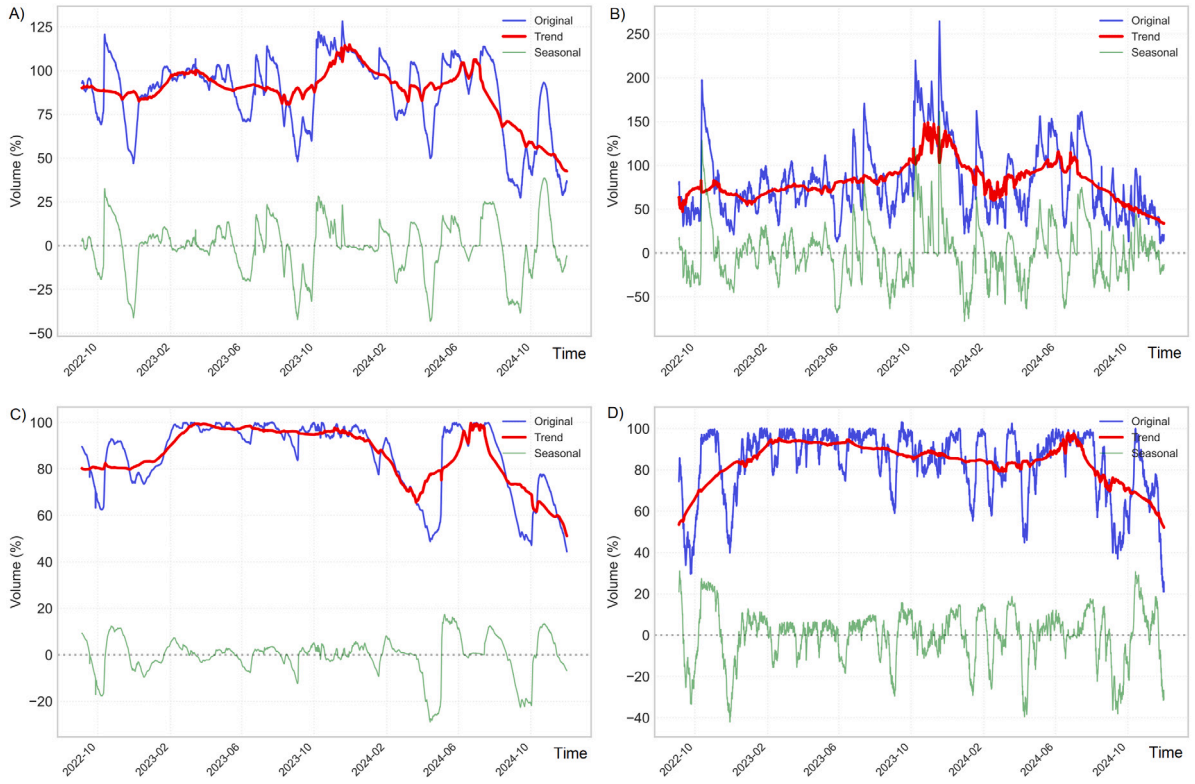

Fig. 5. Decomposition of reservoir volume time series into trend and seasonal components for four representative reservoirs: (A) Sao Roque (trend explains 48.7% of variance), (B) Garibaldi (trend explains 37.0% of variance), (C) Barra Grande (trend explains 68.9% of variance), (D) Campos Novos (trend explains 37.0% of variance).

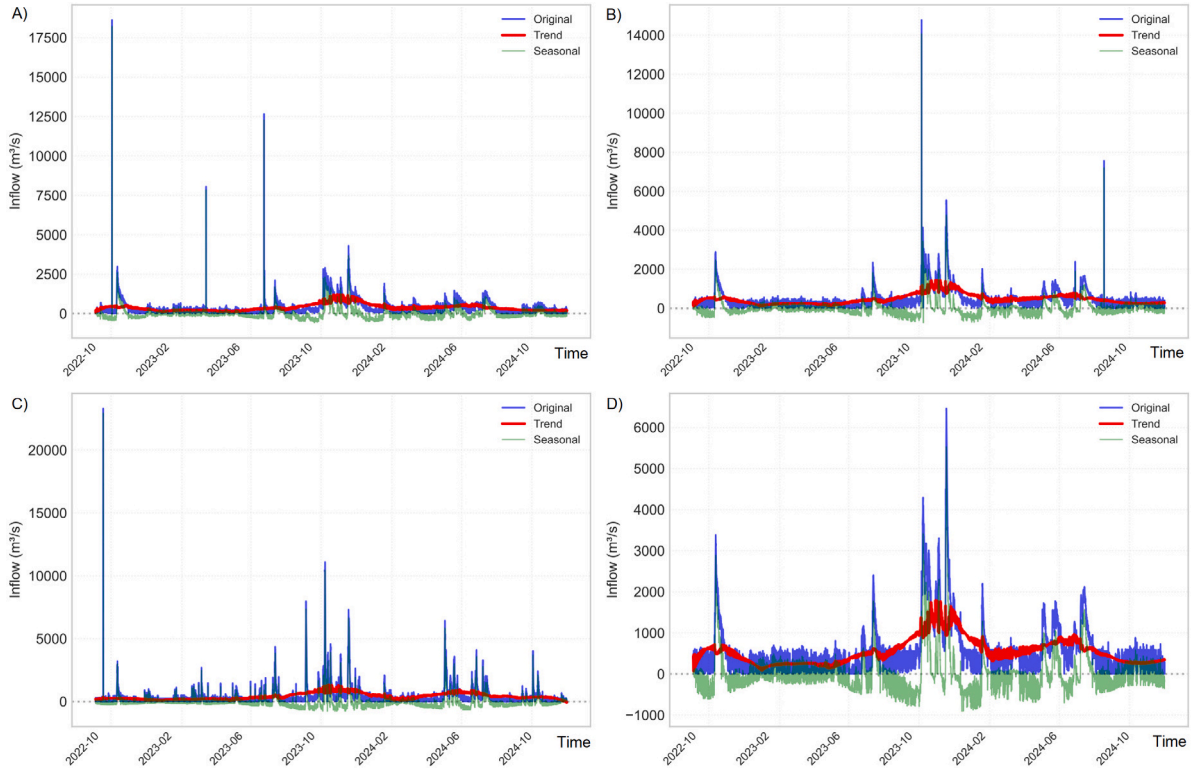

Fig. 6. Inflow signal decomposition for the same set of reservoirs, showing original measurements (blue), extracted trend (red), and seasonal component (green): (A) Sao Roque (trend explains 27.1% of variance), (B) Garibaldi (trend explains 29.4% of variance), (C) Barra Grande (trend explains 12.2% of variance), (D) Campos Novos (trend explains 31.4% of variance).

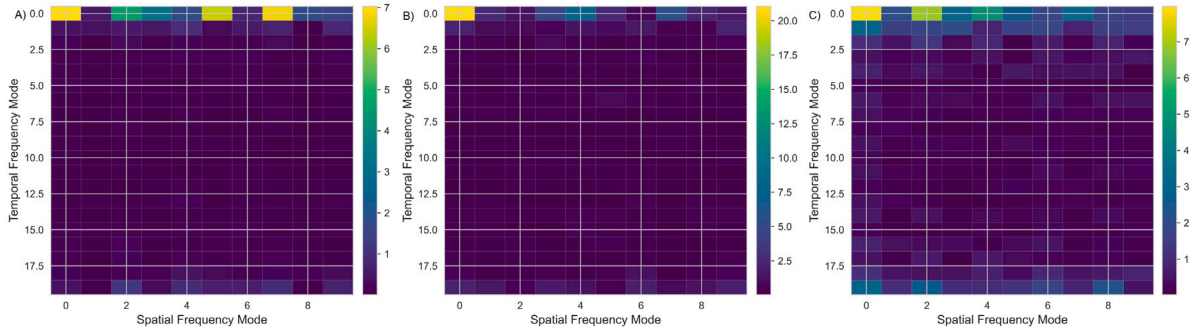

Fig. 7. 2D Fourier transform magnitude comparison across the three input features: (A) 2D FFT magnitude of feature 1, (B) 2D FFT magnitude of feature 2, (C) 2D FFT magnitude of feature 3.

The final hyperparameter set is used for all experiments reported in subsequent sections.

5.3. Spectral analysis of feature processing

A spectral analysis is performed to understand how different features interact across both spatial and temporal frequencies. Within the model, the Fourier neural operator (FNO) is used to decompose spatial–temporal data into frequency components, which reveals underlying patterns not immediately apparent in the raw time-domain signals.

5.3.1. Spectral representation of input features

Fig. 7 presents a comparison of the 2D Fourier transform magnitudes across the three input features. The heatmaps display the distribution of energy in the frequency domain, with temporal frequency modes on the y-axis and spatial frequency modes on the x-axis. All three features exhibit strong activations at low temporal frequencies (modes 0-2) and primarily in the first spatial frequency mode (mode 0), showing the highest magnitude.

In this plot, each subplot represents the frequency domain representation of a feature, with temporal frequency modes on the y-axis and spatial frequency modes on the x-axis. Brighter colors indicate a higher magnitude. All features show a concentration of energy in low temporal and spatial frequencies, with feature 1 exhibiting the strongest magnitude in the lowest frequency components.

This concentration of energy in the lowest frequencies indicates that long-term trends and global spatial patterns dominate the signal structure. Features 1 and 3 show moderate activation in higher spatial modes (particularly modes 5 and 7), suggesting that they capture more localized spatial phenomena compared to feature 2.

The spectral power distribution across features is quantified in Fig. 8, which compares the average power in different frequency modes. The left panel shows the distribution across spatial frequency modes, where all three features follow a similar pattern with high energy in mode 0, followed by secondary peaks in modes 5 and 7. This pattern suggests the presence of both global (mode 0) and regional (modes 5 and 7) spatial patterns in the data. The right panel displays the power distribution across temporal frequency modes, revealing a sharp decay from mode 0 to higher frequencies for all features, with feature 1 consistently exhibiting the highest power across most temporal frequencies.

5.3.2. Feature contribution to spatial modes

Fig. 9 provides a detailed view of how each feature contributes to individual spatial frequency modes through their temporal frequency components. Each subplot represents one spatial mode, with stem plots showing the magnitude of temporal frequencies for all three features. Several observations emerge from this visualization:

In spatial mode 0, which represents global patterns across the network, feature 2 (red) has the highest magnitude at the lowest temporal frequency (mode 0), suggesting that this feature captures the constant, network-wide baseline more strongly. Features 1 and 3 contribute with lower magnitudes.

Spatial modes 1 and 3 show a more balanced contribution from all features, with feature 2 still exhibiting stronger representation at low frequencies but features 1 and 3 becoming more prominent at certain higher frequencies (particularly modes 1, 2, and 19).

In spatial modes 5 and 7, feature 1 (blue) dominates the lowest temporal frequency, indicating that it captures specific regional patterns more strongly than the other features. In contrast, spatial mode 6 shows relatively balanced contributions across all three features with more significant high-frequency components, suggesting that this spatial pattern contains more temporal variability.

The activation in the lowest and highest temporal frequency modes (0 and 19) across all spatial modes indicates the role of both constant offsets and high-frequency fluctuations in characterizing the spatial–temporal dynamics of the system.

The temporal frequency spectrum of spatial modes with feature comparison is presented in Fig. 9. Each subplot corresponds to one spatial frequency mode (0–9), displaying the magnitude of temporal frequency components for each feature (feature 1 in blue circles, feature 2 in red squares, feature 3 in green triangles). This visualization reveals how different features contribute to specific spatial patterns and which temporal frequencies are prominent in each spatial mode.

This spectral analysis reveals several characteristics of the proposed spatial–temporal data. First, the concentration of energy in low frequencies indicates that the system's behavior is dominated by smooth, gradual changes rather than rapid fluctuations. Second, the varying contribution of features across different spatial modes demonstrates that each feature captures distinct aspects of the spatial patterns in the network. Third, the persistence of certain frequency patterns across features suggests underlying physical processes that manifest across different measured variables.

The FNO component of the model uses these spectral properties by learning separate weights for each combination of temporal and spatial frequency, separating the modeling of global trends from local variations. This approach enables the model to capture complex spatial–temporal dependencies that would be challenging to represent with traditional neural network architectures.

5.4. Analysis of learned latent correlations

The LatentCorrelationGCN layers in the model learn an implicit adjacency matrix that represents the spatial–temporal dependencies between monitoring stations. Fig. 10 displays the weights of these learned connections after model convergence. The heatmap displays connection strengths between monitoring stations, where rows represent source nodes and columns represent target nodes. Color intensity corresponds to connection strength, with darker purple indicating minimal correlation and brighter yellow indicating stronger connections.

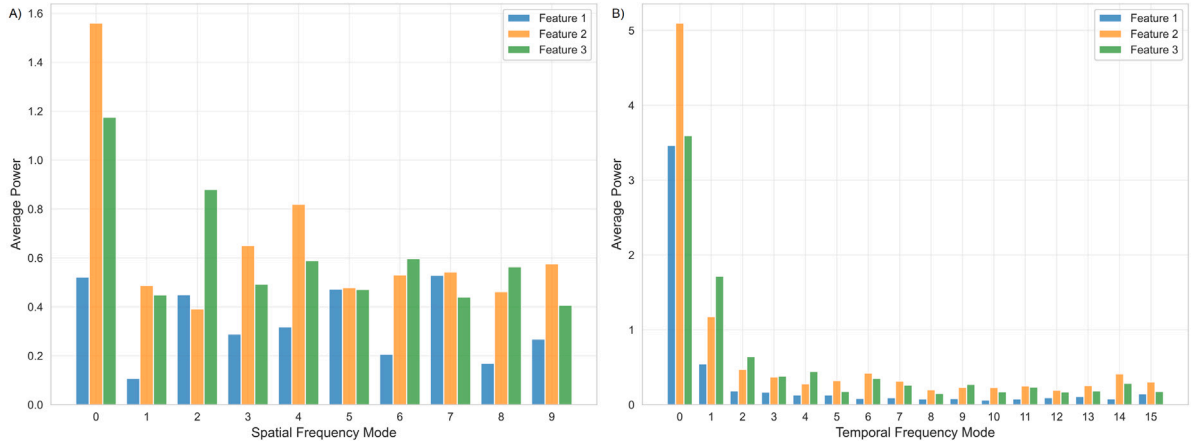

Fig. 8. Power distribution comparison across features. (A) average power in each spatial frequency mode, (B) average power in temporal frequency modes.

The predominantly sparse structure reflects the k-neighbors parameter ($k = 4$) used in the model architecture, with strong self-connections visible along the diagonal.

The heatmap represents connection strengths between each pair of stations, where rows indicate source nodes (information senders) and columns represent target nodes (information receivers). Color intensity corresponds to connection strength, with darker purple indicating minimal correlation and brighter yellow indicating stronger connections. Examining the learned weights reveals a predominantly sparse correlation structure, with most connections having values close to zero. This sparsity aligns with the k-neighbors parameter ($k = 4$) used in the model architecture. Each station maintains strong self-connections, as evidenced by the diagonal elements ranging from 0.25 to 0.55.

The matrix exhibits clear asymmetry, where $A_{ij} \neq A_{ji}$, indicating directional information flow between stations. For example, Santa Clara influences Santiago with a weight of 0.19, while Santiago shows no reciprocal connection to Santa Clara. Several regional patterns emerge in the learned structure. Bidirectional connections exist between station pairs such as Jordão and Machadinho (0.29 and 0.34), and Fundão and Garibaldi (0.16 and 0.18). These mutual connections may reflect geographical proximity or similar meteorological conditions.

The station Osório displays the highest self-correlation coefficient (0.53) in the network, while also forming connections with stations Baixo Iguaçu, Jordão, and Machadinho (all at 0.12). Passo Fundo exhibits connections to multiple stations, including Quebra Queixo (0.27). These learned correlations demonstrate the model's ability to discover underlying relationships in the data without requiring explicit geographical information or predefined adjacency matrices.

5.4.1. Temporal dynamics and attention-based interpretability

Beyond the static latent correlation structure illustrated in Fig. 10, the proposed model learns *time-dependent* relational and temporal dependencies. To analyze this behavior, we examine the temporal evolution of the learned latent graph as well as the decoder attention mechanisms across encoder timesteps and forecast horizons.

Fig. 11 shows the temporal evolution of the learned latent adjacency matrix. The top panel reports the Frobenius norm $\|\mathbf{A}_t - \mathbf{A}_{t-1}\|_F$ between successive encoder timesteps, quantifying the magnitude of structural changes in the latent graph. The observed values indicate smooth but non-negligible evolution throughout most of the input sequence, with a pronounced peak around the middle of the encoder window. This behavior suggests a transient reorganization of inter-station dependencies as additional temporal context is incorporated, followed by partial stabilization rather than convergence to a static graph.

The bottom panel of Fig. 11 tracks the evolution of the dominant latent edge across encoder timesteps. The strongest inter-node connection varies substantially over time, including intervals of near-zero influence and intermittent peaks. This confirms that even the most influential learned dependency is not persistent, and that the model dynamically reallocates node-to-node influence depending on the temporal context. Such behavior rules out interpreting the latent graph as a fixed correlation matrix and highlights its role as a context-adaptive relational representation.

Fig. 12 illustrates the average decoder attention weights over encoder timesteps for the first forecast step. Attention remains broadly distributed across the entire input window, with only moderate variations. This indicates that short-horizon predictions integrate information from a wide temporal context rather than relying on isolated historical points, reflecting stable temporal reasoning in the decoder.

Fig. 13 reports the entropy of the attention distribution as a function of the forecast horizon. Attention entropy decreases monotonically as the prediction horizon increases, indicating progressively more concentrated attention. This trend suggests that longer-horizon forecasts rely on a smaller subset of informative historical states, while short-horizon forecasts benefit from a more distributed temporal representation.

Fig. 14 compares attention distributions for different forecast steps. For early forecasts, attention remains relatively flat across encoder timesteps. As the horizon increases, attention becomes increasingly selective, with pronounced peaks at specific encoder positions. This horizon-dependent transition from distributed to focused attention demonstrates that the decoder adapts its temporal focus according to the prediction task, reinforcing the interpretability of the attention mechanism.

Taken together, these analyses demonstrate that the proposed architecture learns *dynamic and interpretable spatiotemporal dependencies* at multiple levels. The latent graph evolves smoothly while allowing localized structural reconfigurations, and the decoder attention adapts its temporal focus according to the forecast horizon. Importantly, these behaviors emerge without predefined adjacency matrices or explicit geographical priors, confirming that the model discovers and exploits data-driven relational and temporal structure.

5.4.2. Connection strength distribution

Fig. 15 presents the 15 strongest connections identified by the proposed model. The self-connections of Baixo Iguaçu (0.550) and Osório (0.532) demonstrate substantially higher weights than other connections in the network. This indicates that these stations rely heavily on their historical data for prediction. Among the remaining top connections, Osório serves as the target node for 12 connections, suggesting its role as an information sink within the network.

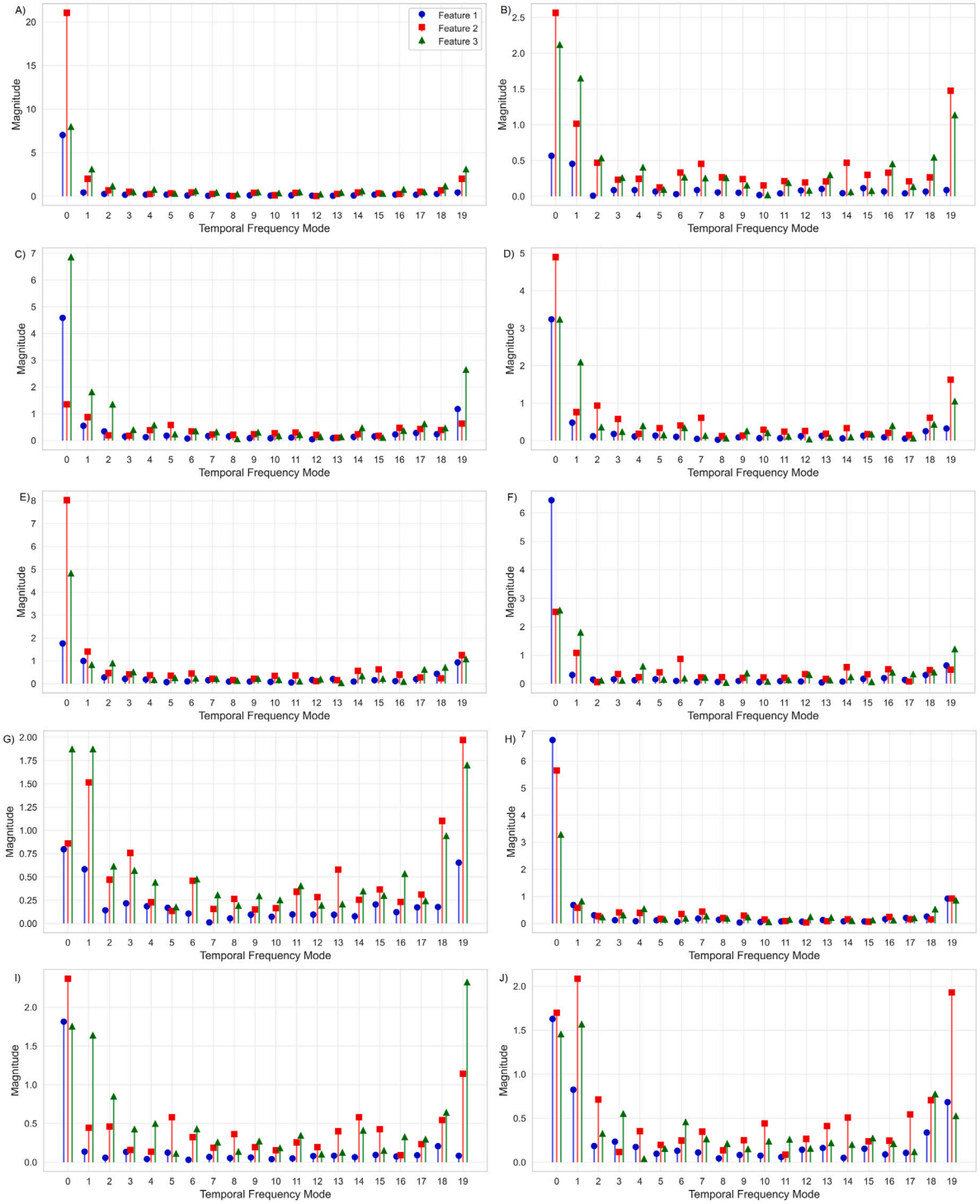

Fig. 9. Temporal frequency spectrum of spatial modes with feature comparison: (A) spatial mode 0, (B) spatial mode 1, (C) spatial mode 2, (D) spatial mode 3, (E) spatial mode 4, ((F) spatial mode 5, G) spatial mode 6, (H) spatial mode 7, (I) spatial mode 8, (J) spatial mode 9.

Self-connections of Baixo Iguaçu (0.550) and Osório (0.532) demonstrate substantially higher weights than other connections. Among non-self connections, São Roque → Osório (0.376) and Santa Clara → Osório (0.365) show the strongest inter-station dependencies. Osório serves as the target node for 12 of the top connections, suggesting its role as an information sink within the network.

The connection strengths exhibit a relatively narrow distribution range (0.316-0.376) for non-self connections, with São Roque → Osório (0.376) and Santa Clara → Osório (0.365) showing the strongest inter-station dependencies. The connection São Roque → Baixo Iguaçu (0.316) represents the only top connection where Osório is neither the source nor the target node.

5.4.3. *Node influence analysis*

Fig. 16 shows the top 10 (out of 19) node influence through three metrics: outgoing connection strength (blue), incoming connection strength (red), and overall centrality (green). The overall centrality measures of node importance (influence) within the graph. Osório and

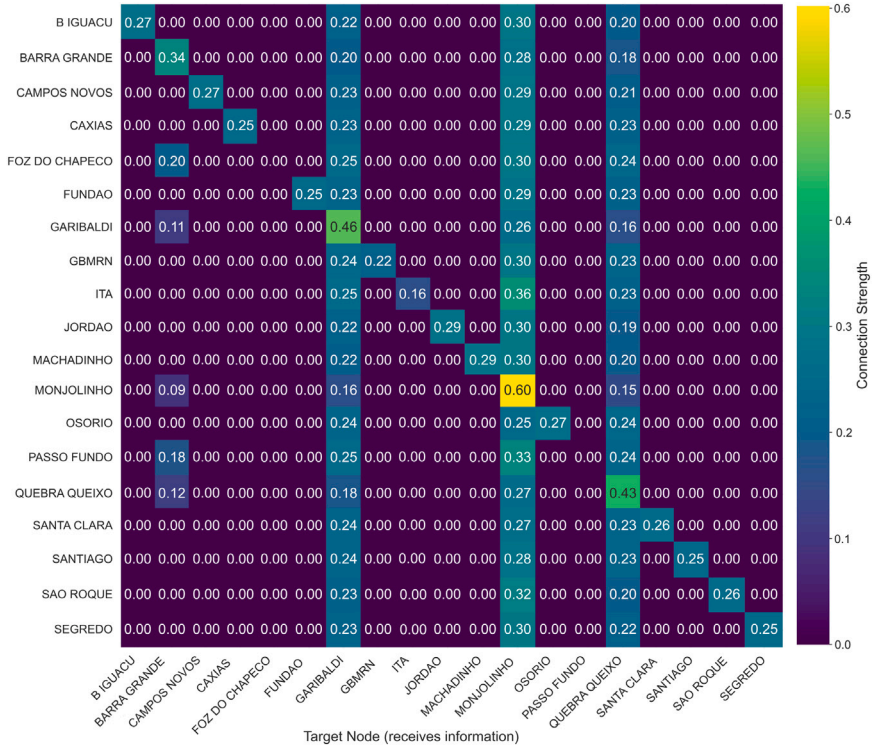

Fig. 10. Learned latent correlation matrix from the LatentCorrelationGCN model.

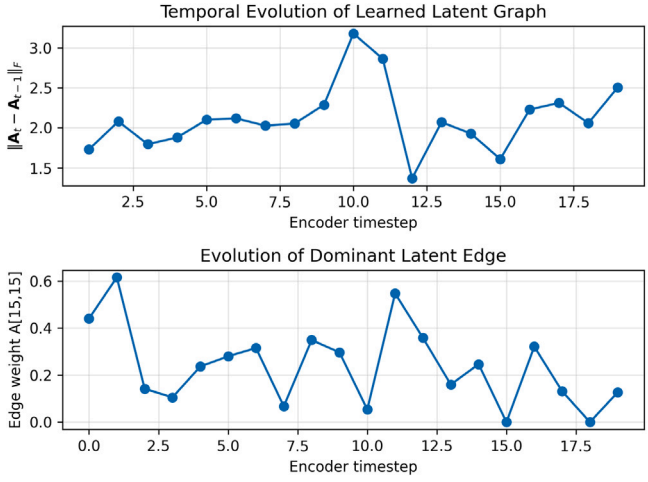

Fig. 11. Temporal evolution of the learned latent graph. (Top) Frobenius norm $\|\mathbf{A}_t - \mathbf{A}_{t-1}\|_F$ between successive encoder timesteps, quantifying structural changes in the latent adjacency matrix. (Bottom) Evolution of the dominant latent edge weight across encoder timesteps, illustrating time-varying inter-node influence.

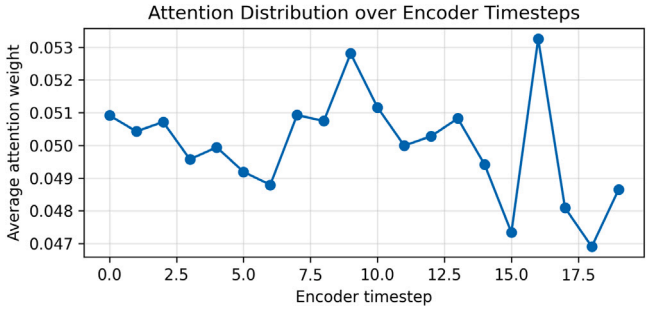

Fig. 12. Average decoder attention distribution over encoder timesteps for the first forecast step, showing broadly distributed temporal reliance.

Baixo Iguaçu emerge as the most central nodes, functioning primarily as information aggregators with strong incoming connections. Fundão and Machadinho form a second tier of influential nodes with moderate incoming and outgoing connection strengths, functioning as regional hubs. Outgoing connection strengths remain relatively uniform across all top 10 nodes, while incoming connection strengths vary substantially. The results reveal distinct roles within the learned network topology:

- Monjolinho emerges as the most central node with significantly higher influence scores than other stations. Their centrality stems primarily from strong incoming connections, indicating that they function as information aggregators within the network. Osório receives strong connections from 12 different stations, while Baixo Iguaçu exhibits the highest self-connection in the network.
- Garibaldi and Quebra Queixo form a second tier of influential nodes, with moderate incoming and outgoing connection strengths. These stations appear to function as regional hubs, both receiving and transmitting information within the network.
- The remaining nodes exhibit more balanced influence profiles with relatively similar outgoing strengths but decreasing incoming connection weights.
- The outgoing connection strengths remain relatively uniform across all top 10 nodes (approximately 1.0), while incoming connection strengths vary substantially. This asymmetry suggests that the model identifies certain stations as more informative for prediction tasks, while all stations contribute somewhat equally as information sources.

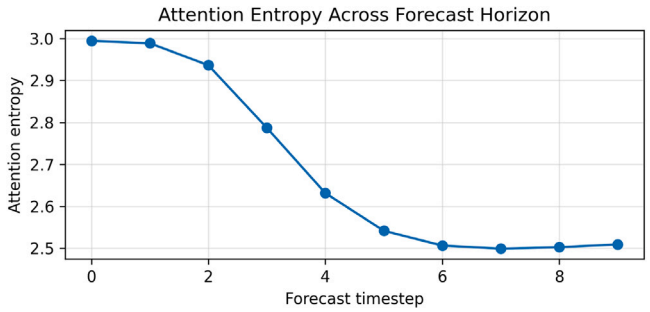

Fig. 13. Attention entropy across the forecast horizon, indicating increasingly concentrated attention for longer-horizon predictions.

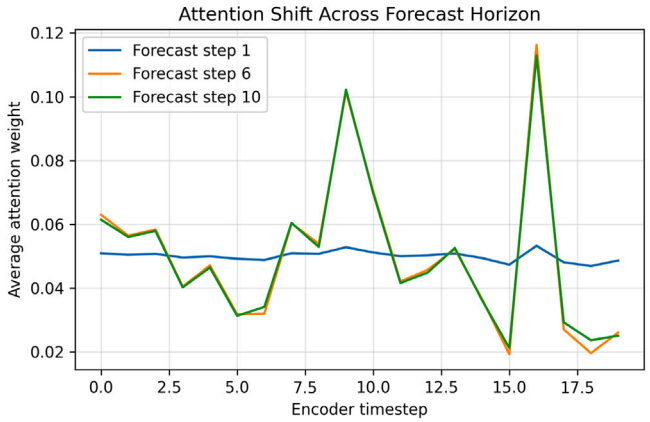

Fig. 14. Attention shift across forecast horizons. Longer-horizon forecasts exhibit increasingly selective attention concentrated on specific encoder timesteps, while short-horizon forecasts maintain a more distributed temporal focus.

To assess the temporal denoising behavior of the model and evaluate its alignment with classical signal processing principles, we analyze the effect of Gaussian smoothing on raw and simulated time series. This analysis includes both a controlled experiment using synthetic signals and a visualization of smoothing behavior in the model's learned outputs.

5.4.4. Gaussian filter characteristics and reference behavior

Fig. 17 shows the Gaussian smoothing analysis; the top left displays the Gaussian kernel used for smoothing, with $\sigma = 1.00$ and kernel size 5. The symmetric bell-shaped curve illustrates the localized weighting of nearby time steps, emphasizing temporal continuity. The top center of Fig. 17 shows the corresponding frequency response, which confirms the low-pass behavior of the filter, with significant attenuation at normalized frequencies beyond 0.4 rad/sample. This aligns to suppress high-frequency temporal noise while preserving low-frequency trends.

To demonstrate the denoising effect, we apply Gaussian smoothing to a synthetic signal corrupted with additive noise. In the top left of Fig. 17, the smoothed signal effectively recovers the original clean waveform, eliminating high-frequency components while retaining the overall structure. The second line of this plot shows the output smoothing of nodes 1, 2, and 3. The third line shows the Conv1D outputs, frequency response, and heatmap of the filter. The last line shows the magnitudes of the filter frequencies to nodes 1, 2, and 3.

5.4.5. Smoothing behavior in simulated outputs

To explore how this smoothing behavior generalizes across network nodes, we simulate time series outputs for three representative nodes. Fig. 18 shows the raw and smoothed sequences for nodes 1 through 3. In all cases, the smoothed curves closely follow the underlying trends while filtering out abrupt, high-frequency fluctuations. This suggests that even without explicitly applying a Gaussian kernel, the model learns to approximate similar smoothing behavior in its output layer.

This conclusion is further supported by the frequency-domain representation in the lower panel of Fig. 18, where the magnitude spectra of raw and smoothed signals are compared. Across all three nodes, the smoothed signals show reduced high-frequency content, indicating the model's tendency to suppress temporal noise and prioritize dominant low-frequency components in the output.

5.4.6. Interpretation and implications

This analysis reveals two key insights. First, the model's output layer implicitly performs temporal smoothing that resembles Gaussian low-pass filtering. Second, the learned smoothing behavior is consistent across different nodes, suggesting a systematic tendency of the model to reduce prediction volatility.

5.5. Sequential prediction performance

Figs. 19 and 20 together present the comparison between model predictions (blue) and ground truth (orange) for all 19 monitoring stations in the network. The plots display sequential predictions over 1000 time steps, revealing the model's ability to capture both short-term fluctuations and long-term trends across diverse temporal patterns. These plots display sequential predictions over 1000 time steps, demonstrating the model's capability to capture both short-term fluctuations and long-term trends in diverse temporal patterns. The visualization highlights the model's accuracy in tracking both high-frequency variations and underlying seasonal patterns across different stations with varying characteristics.

5.5.1. Prediction accuracy patterns

To ensure robust model evaluation and prevent overfitting, we implemented a time series cross-validation approach with 5 folds. Unlike standard k-fold cross-validation, the approach maintains the sequential integrity of the time series by using an expanding window strategy: for each fold k, we train on all data up to segment k and validate on data in segment k. Our model demonstrated consistent performance across all five folds with an average RMSE of 0.379 ± 0.005 and MAE of 0.143 ± 0.006. This low standard deviation indicates strong model stability regardless of the specific temporal segment used for validation.

The prediction accuracy varies across different stations, reflecting the diversity of temporal dynamics in the dataset. Several distinct performance patterns emerge: Stations with smooth, gradual trends (Barra Grande, Machadinho, Gbmrn, Santa Clara, and São Roque) exhibit excellent prediction accuracy, with the model successfully capturing both the direction and magnitude of changes. For these stations, the prediction curve closely follows the ground truth with minimal deviation throughout the entire sequence.

Stations with moderate variability (Caxias, Garibaldi, Itá, Passo Fundo) show good prediction performance with occasional minor deviations during rapid transitions. The model effectively captures medium-term trends while maintaining reasonable accuracy during fluctuations. Highly volatile stations (Baixo Iguaçu, Fundão, Osório) present the greatest challenge, with the model generally following the overall pattern but occasionally missing extreme peaks and troughs. These stations exhibit rapid, high-amplitude oscillations that are inherently more difficult to predict accurately.

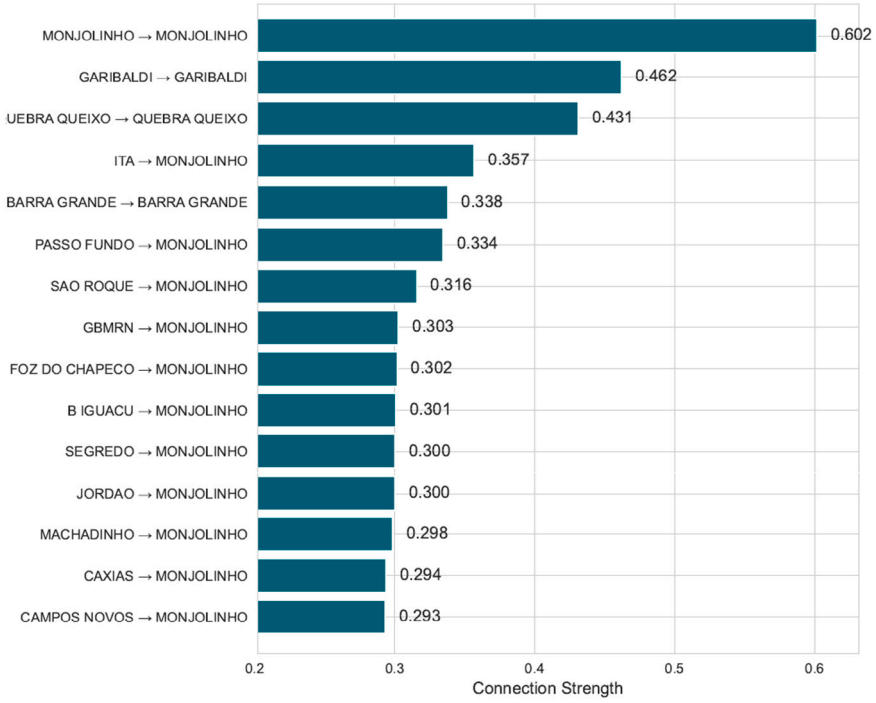

Fig. 15. The 15 strongest connections identified by the model, ranked by connection weight.

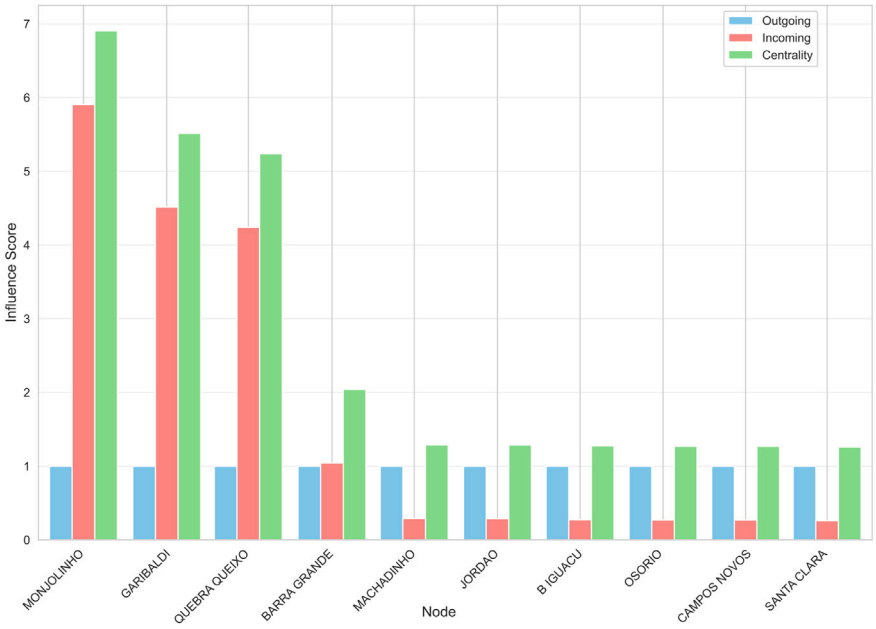

Fig. 16. Node influence analysis quantifying three metrics: outgoing connection strength (blue), incoming connection strength (red), and overall centrality (green).

5.5.2. Temporal pattern characteristics

The model demonstrates varying capabilities in handling different temporal characteristics:

- For cyclic patterns (Santiago, Jordão), the model successfully captures periodicity and phase, maintaining synchronization with the ground truth throughout multiple cycles. This suggests effective learning of recurring temporal dependencies.
- For trend reversals (Campos Novos, Quebra Queixo), the model accurately predicts major directional changes, suggesting it has learned meaningful temporal dependencies beyond simple trend extrapolation.
- For stations exhibiting regime shifts (Foz do Chapecó, Monjolinho), the model adapts quickly to new patterns after transition points, indicating robust generalization capabilities rather than overfitting to specific temporal segments.

5.5.3. Relationship to network structure

The prediction accuracy appears correlated with the learned network structure. Stations identified as central nodes in Fig. 16 (Osório, Baixo Iguaçu) show different prediction patterns: Baixo Iguaçu exhibits highly volatile behavior with reasonable but imperfect prediction accuracy, suggesting that its high self-connection weight (0.550) reflects the

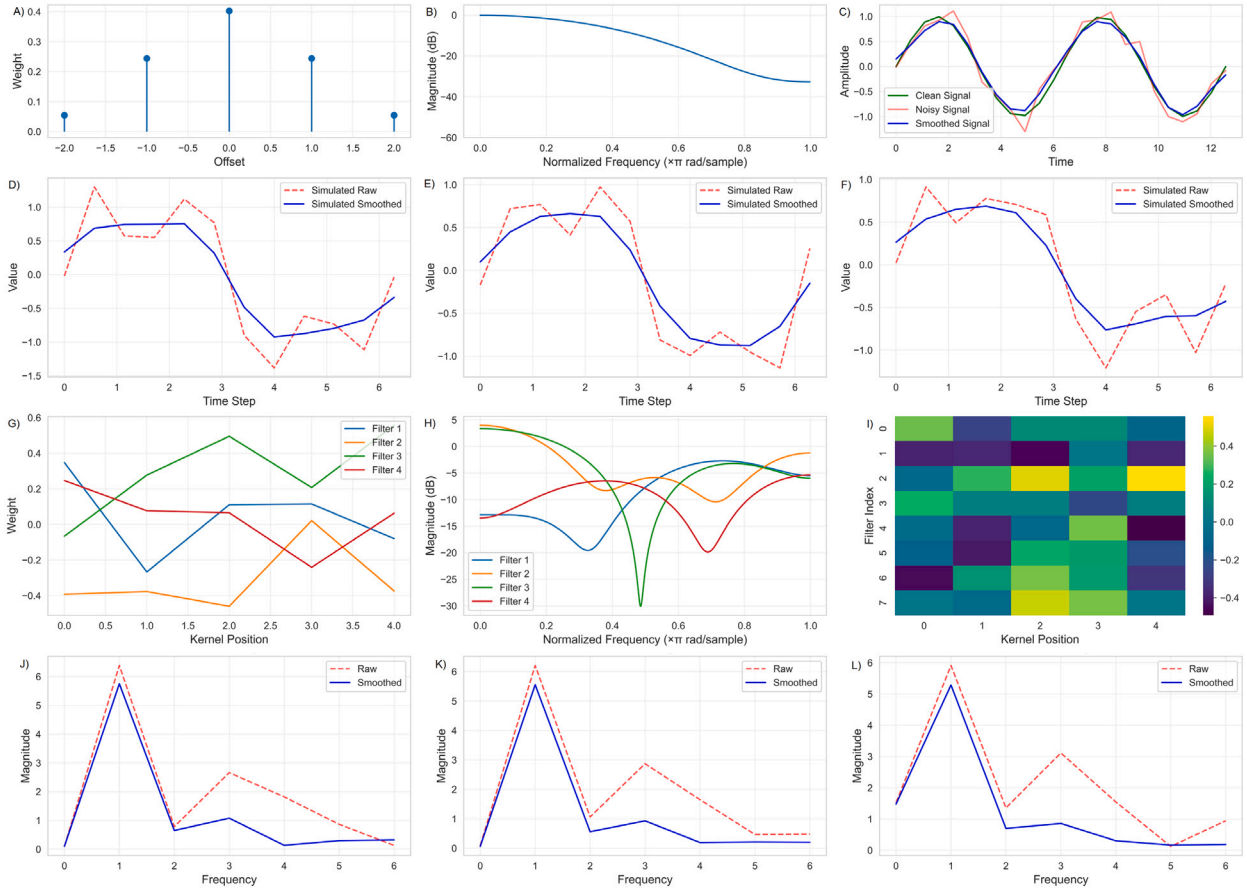

Fig. 17. Gaussian smoothing analysis. (A) Gaussian kernel (Size=5), (B) frequency response of Gaussian filter, (C) Gaussian smoothing demonstration, (D) node 1 output smoothing, (E) node 2 output smoothing, (F) node 3 output smoothing, (G) output smoother Conv1d filters, (H) frequency response of output filters, (I) output smoothing filters heatmap, (J) node 1 frequency content, (K) node 2 frequency content, (L) node 3 frequency contente.

model's strategy to handle inherently unpredictable patterns by relying on recent historical values.

Osório, despite serving as a major information sink in the network, also shows volatile behavior with similar prediction challenges. This suggests that the numerous incoming connections to Osório may represent the model's attempt to leverage multiple information sources to predict a difficult target. Stations with moderate centrality scores (Machadinho, Barra Grande) generally exhibit smoother patterns and better prediction accuracy, indicating that stable temporal dynamics may enable more effective information sharing across the network.

5.6. Ablation study

We conducted an ablation study to evaluate each component's contribution by systematically removing individual elements while keeping all others constant. The full model incorporates five components: Fourier transform features, latent graph convolutional networks (GCN), attention mechanisms, temporal smoothing, and Gaussian smoothing. Seven configurations were trained under identical conditions and evaluated on the same test dataset.

Table 6 presents the results across MSE, RMSE, and MAE metrics. The full model achieves an MSE of 0.139 and RMSE of 0.373, while the baseline model with all components disabled establishes the performance benchmark.

Baseline Model Performance: The baseline configuration with all components disabled achieves MSE of 0.199, RMSE of 0.447, and MAE of 0.311, establishing the performance floor for comparison. This configuration represents the model's predictive capability without any architectural enhancements.

Table 6
Ablation study results: Impact of model components on performance.

Fourier Transform	Latent GCN	Attention	Temporal Smoothing	Gaussian Smoothing	MSE↓	RMSE↓	MAE↓
✓	✓	✓	✓	✓	**0.139**	**0.373**	**0.202**
✗	✓	✓	✓	✓	0.174	0.417	0.250
✓	✗	✓	✓	✓	0.157	0.397	0.228
✓	✓	✗	✓	✓	0.167	0.409	0.277
✓	✓	✓	✗	✓	0.146	0.382	0.217
✓	✓	✓	✓	✗	0.152	0.390	0.217
✗	✗	✗	✗	✗	0.199	0.447	0.311

Best performance highlighted in bold.

Full Model vs. Baseline: The complete architecture improves over the baseline across all metrics. MSE reduces by 30.2% (0.199 to 0.139), RMSE decreases by 16.6% (0.447 to 0.373), and MAE improves by 35.0% (0.311 to 0.202). These results quantify the collective efficacy of the integrated components.

Component-wise Analysis:

- **Fourier Transform**: Removal causes the largest performance drop among individual components, with MSE increasing 25.2% (0.139 to 0.174) and RMSE increasing 11.8% (0.373 to 0.417). This indicates that frequency domain features are critical for capturing temporal patterns in the data.
- **Graph Convolutional Networks**: Removing latent GCN increases MSE by 13.1% and RMSE by 6.4%, while MAE increases from 0.202 to 0.228. The degradation across all metrics indicates the

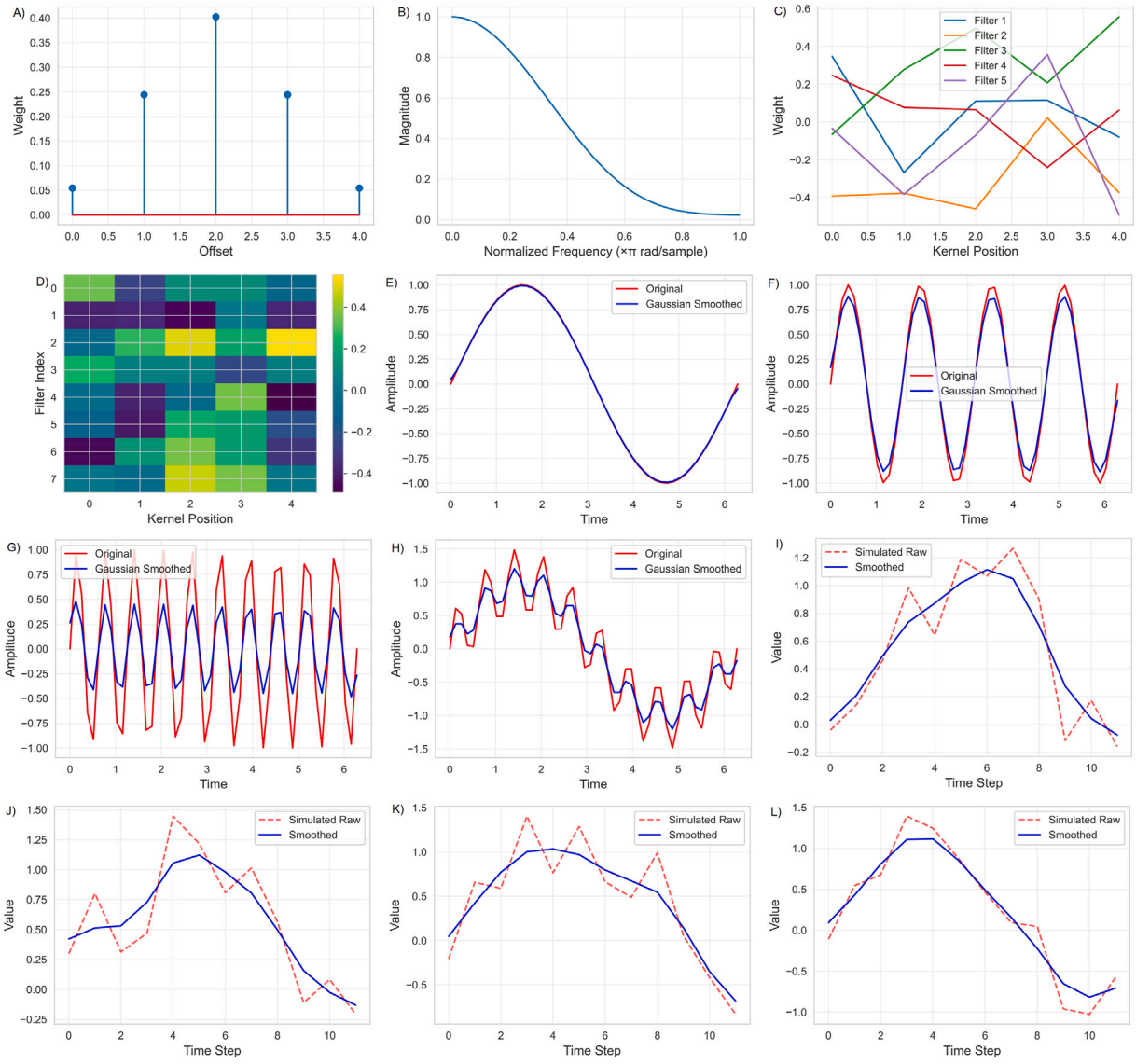

Fig. 18. Temporal smoothing behavior on simulated node outputs: (A) Gaussian smoothing kernel, (B) frequency response of Gaussian kernel, (C) first Conv1d layer filters, (D) first Conv1d layer filters (All), (E) low freq signal, (F) medium freq signal, (G) high freq signal, (H) mixed freqs signal, (I) node 1 simulated, (J) node 2 simulated, (K) node 3 simulated, (L) node 4 simulated .

role of graph-based representations for spatial relationship modeling.

- **Attention Mechanism**: Disabling attention increases MSE by 20.3% and produces the largest MAE increase of 37.0% (0.202 to 0.277). The large impact on MAE indicates attention mechanisms are effective at handling outliers and extreme values.
- **Smoothing Components**: Temporal smoothing removal increases MSE by 5.2% (0.139 to 0.146) and MAE by 7.2% (0.202 to 0.217). Gaussian smoothing removal exhibits similar patterns with MSE increasing 9.3% (0.139 to 0.152) and MAE by 7.3% (0.202 to 0.217). Both smoothing techniques contribute to noise reduction and prediction stability.
- **Component Interactions**: The full model achieves the best performance across all evaluation metrics, indicating positive interactions between components. The worst-performing single-component ablation (Fourier transform removal: MSE 0.174) outperforms the baseline (MSE 0.199), indicating that partial component combinations retain predictive value.

The ablation study confirms that each architectural component contributes to model performance, with Fourier transforms and attention mechanisms having the largest individual impact on predictive accuracy. The baseline comparison establishes the importance of incorporating these components in the overall architecture.

5.7. Analysis of model predictive accuracy over forecast horizons

To evaluate the predictive ability of the proposed model across different time horizons, we conducted forecasting experiments across multiple horizons, ranging from one to five steps ahead, as presented in Table 7, where the best performance is highlighted in bold. This analysis quantifies the degradation in forecast accuracy as the forecast horizon extends and assesses the model's suitability for extended time horizon forecasts.

The model achieves an MSE of 0.025, RMSE of 0.159, and MAE of 0.097 for 1-step ahead prediction. These results establish the baseline performance for immediate temporal forecasting. Prediction accuracy decreases with increasing temporal distance. MSE increases by 404.0% from 1-step (0.025) to 5-step forecasting (0.126). MAE shows a 98.3%

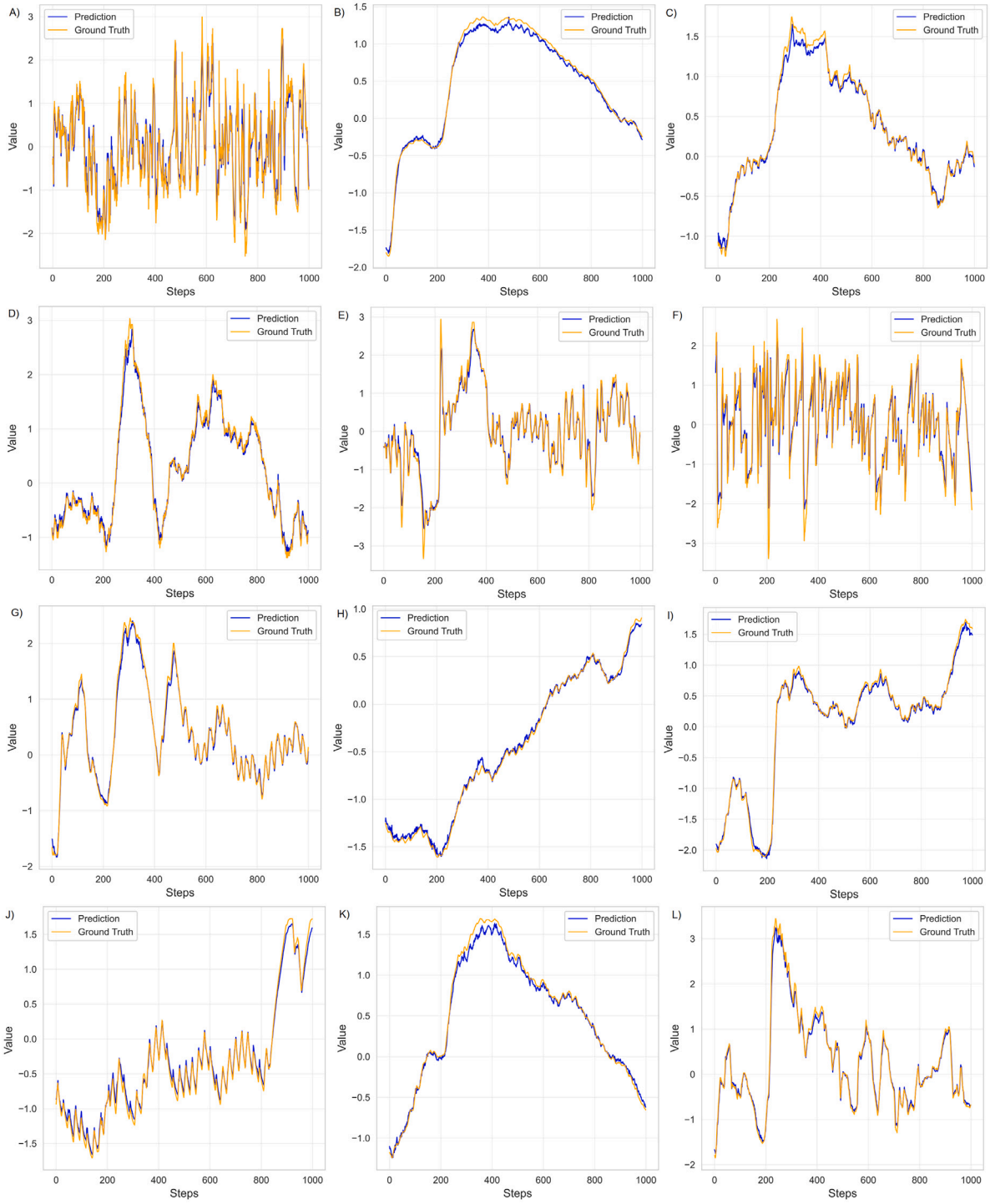

Fig. 19. Comparison between model predictions (blue) and ground truth (orange) for: (A) B Iguacu, (B) Barra Grande, (C) Campos Novos, (D) Caxias, (E) Foz Do Chapeco, (F) Fundao, (G) Garibaldi, (H) Gbmrn, (I) Ita, (J) Jordao, (K) Machadinho, (L) Monjolinho.

increase from 0.097 to 0.192 across the same horizon. This degradation follows the expected pattern in autoregressive forecasting models, where error accumulation compounds across prediction steps.

The directional accuracy metric, computed for horizons 2-5 steps, ranges from 0.371 to 0.411. The maximum value occurs at 2-step forecasting (0.411), with a mean of 0.392 across the evaluated horizons. The standard deviation of 0.017 indicates consistent performance despite degrading magnitude accuracy. Both MSE and MAE exhibit non-linear increases with the forecasting horizon. The steeper MSE degradation compared to MAE (404.0% vs 98.3%) indicates amplified sensitivity to outliers in extended predictions. This pattern suggests that prediction errors compound across temporal steps following established multi-step forecasting theory.

Beyond multi-step forecasting accuracy, we analyzed the model's behavior across spatial, temporal, and structural dimensions to characterize its predictive capabilities and limitations.

Node-wise Performance Heterogeneity: The model exhibits significant spatial performance variation across the network nodes, as detailed in Table 8. MSE values range from 0.006 (Sao Roque) to 0.818

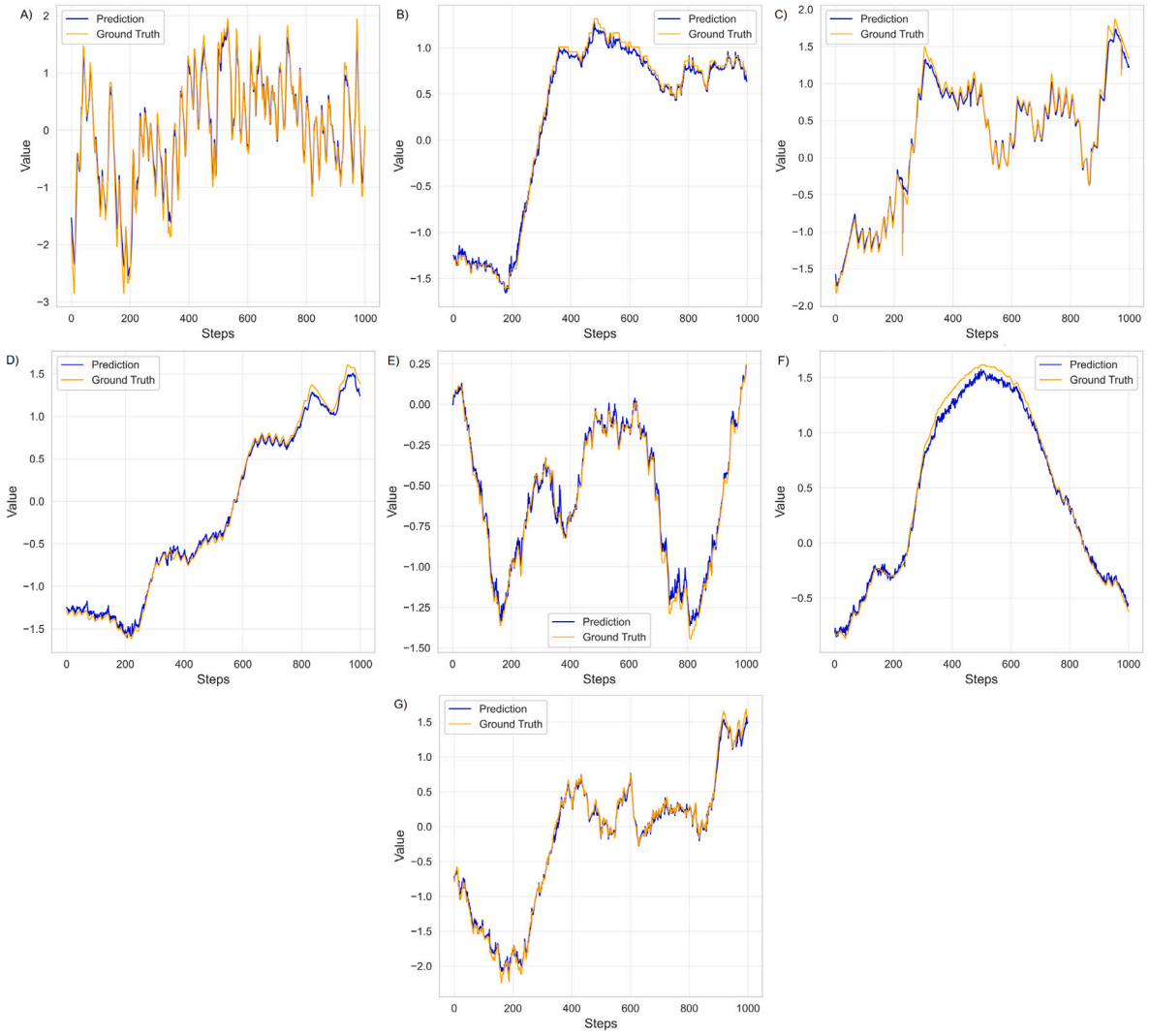

Fig. 20. Comparison between model predictions (blue) and ground truth (orange) for: (A) Osorio, (B) Passo Fundo, (C) Quebra Queixo, (D) Santa Clara, (E) Santiago, (F) Sao Roque, (G) Segredo.

Table 7
Evaluating multi-step forecasting performance and accuracy across temporal horizons.

Horizon	MSE↓	RMSE↓	MAE↓	Directional Accuracy
1	**0.025**	**0.159**	**0.097**	–
2	0.044	0.210	0.118	**0.411**
3	0.070	0.264	0.144	0.403
4	0.097	0.312	0.168	0.382
5	0.126	0.356	0.192	0.371

Best performance highlighted in bold.
Note: Directional accuracy not computed for 1-step forecasting.

Table 8
Node-wise forecasting performance: Error metrics by geographic location.

Node	MSE↓	RMSE↓	MAE↓
São Roque	**0.006**	**0.077**	**0.060**
Gbmrn	0.007	0.082	0.065
Santa Clara	0.009	0.097	0.074
Santiago	0.011	0.104	0.082
Machadinho	0.012	0.112	0.088
Barra Grande	0.013	0.112	0.089
Passo Fundo	0.013	0.113	0.088
Campos Novos	0.016	0.128	0.095
Segredo	0.027	0.163	0.127
Itá	0.028	0.167	0.108
Jordão	0.032	0.179	0.142
Quebra Queixo	0.035	0.187	0.138
Garibaldi	0.060	0.245	0.174
Caxias	0.063	0.252	0.182
Monjolinho	0.109	0.330	0.214
Foz do Chapecó	0.286	0.535	0.367
Osório	0.339	0.582	0.428
Baixo Iguaçu	0.747	0.864	0.653
Fundão	0.818	0.904	0.630

Best performance highlighted in bold.

(Fundao), yielding a performance variance of 0.057. The 139-fold difference between best and worst performing nodes indicates substantial spatial heterogeneity in predictive accuracy. Top-performing locations (Sao Roque, Gbmrn, Santa Clara) achieve MSE $\leq$ 0.009, while poorly performing nodes (Osorio, B Iguacu, Fundao) exhibit MSE $\geq$ 0.339.

Cross-node correlations span the full range [-0.992, 0.998] with a mean correlation of 0.124 $\pm$ 0.688. The wide correlation distribution suggests diverse temporal synchronization patterns across spatial locations, with some node pairs exhibiting strong positive or negative dependencies while others remain largely independent. The attention

Table 9
Comparison of model prediction performance by station.

Model	MSE↓	RMSE↓	MAE↓
LGBMRegressor	0.182	0.427	0.248
NGBoost	0.255	0.505	0.348
Random Forest	0.829	0.910	0.724
XGBoost	0.178	0.422	0.243
Seq2seqLatentGNN	**0.139**	**0.373**	**0.202**

Best performance highlighted in bold.

mechanism learns a sparse graph structure with a sparsity ratio of 0.053, indicating that only 5.3% of potential connections receive significant attention weights. Mean attention entropy of 2.230 suggests moderate attention concentration, while graph asymmetry of 0.011 indicates nearly symmetric learned relationships. Attention weights range from 0.003 to 0.514, with maximum attention concentrated on specific node pairs.

5.8. Models performance comparisons

We evaluate the performance of the proposed model against four based models, LGBMRegressor, NGBoost, Random Forest, and XGBoost, using an identical forecast horizon of 10 h. The results of this comparison are summarized in Table 9. The best results are highlighted in bold font. The Seq2seqLatentGNN obtains an MSE of 0.139, which is lower than the values observed for all four-based models; the next closest is XGBoost with an MSE of 0.178. A similar trend is observed for RMSE, where the proposed model yields 0.379 compared to 0.422 for XGBoost. In terms of MAE, the model reaches 0.143, whereas the best among the four-based models (XGBoost) results in 0.243.

These results suggest that the proposed model achieves lower error magnitudes across all considered metrics. The difference is particularly consistent when compared with XGBoost, which performs best among the tree-based baselines. One possible contributing factor is the ability of the proposed model to incorporate both temporal frequency patterns and latent spatial relationships through the Fourier and graph-based components, which are not explicitly modeled in the tree-based methods.

6. Conclusion

This paper introduced the Seq2SeqLatentGNN, a novel deep learning architecture that combines Fourier-based spectral processing, dynamic latent graph learning, and attention-driven sequence modeling to forecast multi-node spatiotemporal dynamics in hydroelectric reservoir systems. By leveraging a latent graph convolutional network, the model effectively inferred dynamic spatial relationships without relying on predefined adjacency matrices. The integration of a custom Fourier layer enabled the capture of long-term periodicities and regime shifts, while attention-based sequence modeling enhanced the temporal forecasting capabilities. The proposed architecture demonstrated superior accuracy compared to traditional statistical methods and modern machine learning baselines when evaluated on a large-scale, high-resolution dataset from 19 interconnected hydroelectric reservoirs in southern Brazil. The model successfully learned both the latent structure of the reservoir network and complex spatiotemporal patterns, contributing a unified and adaptive framework for dynamic system forecasting.

To extend the contributions of this research, several future directions are recommended. First, incorporating probabilistic forecasting capabilities could enhance uncertainty quantification, especially for long-term reservoir management. Second, integrating domain-specific hydrological constraints (e.g., conservation laws, hydraulic delay functions) into the learning process would improve physical consistency and model interpretability. Third, expanding the application of Seq2SeqLatentGNN to other multi-node domains, such as smart grids or urban water networks, would test its generalizability. Finally, future research should investigate the integration of exogenous variables such as climate forecasts and regulatory policies, further enhancing the model's responsiveness to external influences.

CRediT authorship contribution statement

Laio Oriel Seman: Writing – original draft, Software, Methodology, Investigation, Formal analysis, Conceptualization. **Stefano Frizzo Stefenon:** Writing – original draft, Methodology, Investigation, Formal analysis. **Kin-Choong Yow:** Writing – review & editing, Supervision. **Leandro dos Santos Coelho:** Writing – review & editing, Supervision. **Viviana Cocco Mariani:** Writing – review & editing, Supervision.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was supported in part by the Natural Sciences and Engineering Research Council of Canada, Canada (NSERC) under Grant DDG-2024-00035, and in part by the Cette recherche a été financée par le Conseil de recherches en sciences naturelles et en génie du Canada (CRSNG) under Grant DDG-2024-00035. The authors Mariani and Coelho thank the National Council of Scientific and Technological Development of Brazil - CNPq (Grants number: 314389/2023-7-PQ, 313169/2023-3-PQ, 407453/2023-7-Universal, and 442176/2023-6-Peci) for its financial support of this work.

Data availability

The algorithm is available at: https://github.com/lseman/Seq2SeqLatentGNN/.

References

Akiba, T., Sano, S., Yanase, T., Ohta, T., Koyama, M., 2019. Optuna: A next-generation hyperparameter optimization framework. In: Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. ACM, Anchorage, AK, USA, pp. 2623–2631. http://dx.doi.org/10.1145/3292500.3330701.

Alsakarneh, R.A.I., Mati, S., Ismael, G.Y., Masoud, S., Aliyu, N., Samour, A., Uzun, B., 2025. Hybrid modelling of ruble exchange rates amidst the russo-ukrainian conflict using swarm and fuzzy neural networks. Eng. Appl. Artif. Intell. 153, 110854. http://dx.doi.org/10.1016/j.engappai.2025.110854.

Alsulamy, S., 2025. Predicting construction delay risks in Saudi Arabian projects: A comparative analysis of CatBoost, xgboost, and LGBM. Expert Syst. Appl. 268, 126268. http://dx.doi.org/10.1016/j.eswa.2024.126268.

Apaydin, H., Feizi, H., Sattari, M.T., Colak, M.S., Shamshirband, S., Chau, K.W., 2020. Comparative analysis of recurrent neural network architectures for reservoir inflow forecasting. Water 12 (5), 1500. http://dx.doi.org/10.3390/w12051500.

Arastehfar, S., Matinkia, M., Jabbarpour, M.R., 2022. Short-term residential load forecasting using graph convolutional recurrent neural networks. Eng. Appl. Artif. Intell. 116, 105358. http://dx.doi.org/10.1016/j.engappai.2022.105358.

Bai, L., Yao, L., Kanhere, S., Wang, X., Sheng, Q., 2019. STG2seq: Spatial-temporal graph to sequence model for multi-step passenger demand forecasting. In: Proceedings of the 28th International Joint Conference on Artificial Intelligence. IJCAI, pp. 1981–1987.

Barzola-Monteses, J., Gómez-Romero, J., Espinoza-Andaluz, M., Fajardo, W., 2025. Time series forecasting techniques applied to hydroelectric generation systems. Int. J. Electr. Power Energy Syst. 164, 110424. http://dx.doi.org/10.1016/j.ijepes.2024.110424.

Buratto, W.G., Muniz, R.N., Nied, A., González, G.V., 2024. Seq2Seq-LSTM with attention for electricity load forecasting in Brazil. IEEE Access 12, 30020–30029. http://dx.doi.org/10.1109/ACCESS.2024.3365812.

Chavleishvili, S., Manganelli, S., 2024. Forecasting and stress testing with quantile vector autoregression. J. Appl. Econometrics 39 (1), 66–85. http://dx.doi.org/10.1002/jae.3009.

Chowdhury, S., Saha, A.K., Das, D.K., 2025. Hydroelectric power potentiality analysis for the future aspect of trends with R2 score estimation by xgboost and random forest regressor time series models. Procedia Comput. Sci. 252, 450–456. http://dx.doi.org/10.1016/j.procs.2025.01.004.

Coelho, L.S., Ayala, H.V.H., Mariani, V.C., 2024. CO and NOx emissions prediction in gas turbine using a novel modeling pipeline based on the combination of deep forest regressor and feature engineering. Fuel 355, 129366. http://dx.doi.org/10.1016/j.fuel.2023.129366.

da Silva, E.C., Finardi, E.C., Stefenon, S.F., 2024. Enhancing hydroelectric inflow prediction in the Brazilian power system: A comparative analysis of machine learning models and hyperparameter optimization for decision support. Electr. Power Syst. Res. 230, 110275. http://dx.doi.org/10.1016/j.epsr.2024.110275.

Dai, H., Xiong, L., Ma, Q., Duan, Z., 2025. Deep learning model for drought prediction based on large-scale spatial causal network in the yangtze river basin. J. Hydrol. 654, 132808. http://dx.doi.org/10.1016/j.jhydrol.2025.132808.

Fadhillah, M.F., Lee, S., Lee, C.-W., Park, Y.-C., 2021. Application of support vector regression and metaheuristic optimization algorithms for groundwater potential mapping in gangneung-si, South Korea. Remote. Sens. 13 (6), 1196. http://dx.doi.org/10.3390/rs13061196.

Feng, D., Xu, T., Lin, G., Fang, K., Shen, C., 2022. A graph neural network approach to basin-scale river network learning and streamflow forecasting. Hydrol. Earth Syst. Sci. 26 (18), 5163–5178. http://dx.doi.org/10.5194/hess-26-5163-2022.

Ferchichi, A., Chihaoui, M., Ferchichi, A., 2024. Spatio-temporal modeling of climate change impacts on drought forecast using generative adversarial network: A case study in africa. Expert Syst. Appl. 238, 122211. http://dx.doi.org/10.1016/j.eswa.2023.122211.

Fernando De Toledo, J., Valadares Siqueira, H., Biuk, L.H., Sacchi, R., Azambuja, R.D.R., Junior, R.A., Asano, P.T.L., 2023. Climate indices impact in monthly streamflow series forecasting. IEEE Access 11, 21451–21464. http://dx.doi.org/10.1109/ACCESS.2023.3237982.

Fu, C., Su, Y., Su, K., Liu, Y., Shi, J., Wu, B., Liu, C., Ishi, C.T., Ishiguro, H., 2025. HAM-gnn: A hierarchical attention-based multi-dimensional edge graph neural network for dialogue act classification. Expert Syst. Appl. 261, 125459. http://dx.doi.org/10.1016/j.eswa.2024.125459.

Guo, S., Lin, Y., Feng, N., Song, C., Wan, H., 2019. Attention based spatial-temporal graph convolutional networks for traffic flow forecasting. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 33, pp. 922–929.

Guo, G., Yuan, W., 2020. Short-term traffic speed forecasting based on graph attention temporal convolutional networks. Neurocomputing 410, 387–393. http://dx.doi.org/10.1016/j.neucom.2020.06.001.

He, P., Pan, J., Li, Y., 2022. Long-term dam behavior prediction with deep learning on graphs. J. Comput. Des. Eng. 9 (4), 1230–1245. http://dx.doi.org/10.1093/jcde/qwac051.

Hu, Y., Sun, Z., Pei, L., Han, Y., Li, W., 2024. Evaluate asphalt pavement frictional characteristics based on IGWO-ngboost using 3D macro-texture data. Expert Syst. Appl. 242, 122786. http://dx.doi.org/10.1016/j.eswa.2023.122786.

Huang, N., Wang, S., Wang, R., Cai, G., Liu, Y., Dai, Q., 2023. Gated spatial-temporal graph neural network based short-term load forecasting for wide-area multiple buses. Int. J. Electr. Power Energy Syst. 145, 108651. http://dx.doi.org/10.1016/j.ijepes.2022.108651.

Janizadeh, S., Thi Kieu Tran, T., Bateni, S.M., Jun, C., Kim, D., Trauernicht, C., Heggy, E., 2024. Advancing the lightgbm approach with three novel nature-inspired optimizers for predicting wildfire susceptibility in kaua'i and moloka'i islands, hawaii. Expert Syst. Appl. 258, 124963. http://dx.doi.org/10.1016/j.eswa.2024.124963.

Ji, X., Sun, Z., Lv, H., Yu, X., Tang, S., Zhang, D., Liang, Y., 2024. Spatio-temporal multivariable time vario-zoom network for water level forecasting based on high-resolution hydrological dataset. J. Hydrol. 634, 131060. http://dx.doi.org/10.1016/j.jhydrol.2024.131060.

Jin, M., Koh, H.Y., Wen, Q., Zambon, D., Alippi, C., Webb, G.I., King, I., Pan, S., 2024. A survey on graph neural networks for time series: Forecasting, classification, imputation, and anomaly detection. IEEE Trans. Pattern Anal. Mach. Intell. 46 (12), 10466–10485. http://dx.doi.org/10.1109/TPAMI.2024.3443141.

Kim, J., Choi, S., 2023. BayesO: A Bayesian optimization framework in python. J. Open Source Softw. 8 (90), 5320. http://dx.doi.org/10.21105/joss.05320.

Kim, J., Kim, T., Ryu, J.-G., Kim, J., 2023. Spatiotemporal graph neural network for multivariate multi-step ahead time-series forecasting of sea temperature. Eng. Appl. Artif. Intell. 126, 106854. http://dx.doi.org/10.1016/j.engappai.2023.106854.

Kim, S., Lee, T.-H., Lee, J., 2025. TMF-GNN: Temporal matrix factorization-based graph neural network for multivariate time series forecasting with missing values. Expert Syst. Appl. 275, 127001. http://dx.doi.org/10.1016/j.eswa.2025.127001.

Kipf, T.N., Fetaya, E., Wang, K.-C., Welling, M., Zemel, R., 2018. Neural relational inference for interacting systems. In: Proceedings of the 35th International Conference on Machine Learning. ICML, pp. 2688–2697.

Larcher, J.H.K., Stefenon, S.F., dos Santos Coelho, L., Mariani, V.C., 2024. Enhanced multi-step streamflow series forecasting using hybrid signal decomposition and optimized reservoir computing models. Expert Syst. Appl. 255, 124856. http://dx.doi.org/10.1016/j.eswa.2024.124856.

Li, Y., Bao, T., Gong, J., Shu, X., Zhang, K., 2020. The prediction of dam displacement time series using STL, extra-trees, and stacked LSTM neural network. IEEE Access 8, 94440–94452. http://dx.doi.org/10.1109/ACCESS.2020.2995592.

Li, Q., Han, Z., Wu, X., 2018a. Deeper insights into graph convolutional networks for semi-supervised learning. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 32, pp. 1–9. http://dx.doi.org/10.1609/aaai.v32i1.11604.

Li, Z., Kovachki, N.B., Azizzadenesheli, K., Bhattacharya, K., Stuart, A., Anandkumar, A., et al., 2021. Fourier neural operator for parametric partial differential equations. In: International Conference on Learning Representations. pp. 1–16.

Li, Q., Qin, J., Wang, D., Qin, X., Cui, D., Xie, J., Sun, D., 2025a. TOEformer: Temporal order enhanced transformer for time series forecasting. Eng. Appl. Artif. Intell. 155, 110965. http://dx.doi.org/10.1016/j.engappai.2025.110965.

Li, W., Song, X., Tu, Y., 2025b. GraphDRL: GNN-based deep reinforcement learning for interactive recommendation with sparse data. Expert Syst. Appl. 273, 126832. http://dx.doi.org/10.1016/j.eswa.2025.126832.

Li, Y., Yu, R., Shahabi, C., Liu, Y., 2018b. Diffusion convolutional recurrent neural network: Data-driven traffic forecasting. 3, pp. 1–16. http://dx.doi.org/10.48550/arXiv.1707.01926, arXiv:1707.01926.

Liu, Y., Hou, G., Huang, F., Qin, H., Wang, B., Yi, L., 2022. Directed graph deep neural network for multi-step daily streamflow forecasting. J. Hydrol. 607, 127515. http://dx.doi.org/10.1016/j.jhydrol.2022.127515.

Liu, Y., Huang, Z., Cheng, Y., 2023. A hydrological data prediction model based on LSTM with sequence-to-sequence architecture. Water 15 (4), 670. http://dx.doi.org/10.3390/w15040670.

Liu, G., Yan, Y., Cai, J., Qi Wu, E., Fang, S., David Cheok, A., Song, A., 2025. GCD: Graph contrastive denoising module for GNNs in EEG classification. Expert Syst. Appl. 265, 126013. http://dx.doi.org/10.1016/j.eswa.2024.126013.

Lu, T., Gu, H., Gu, C., Shao, C., Yuan, D., 2025. A multi-point dam deformation prediction model based on spatiotemporal graph convolutional network. Eng. Appl. Artif. Intell. 149, 110483. http://dx.doi.org/10.1016/j.engappai.2025.110483.

Ma, B., Xue, Y., Lu, Y., Chen, J., 2025. Stockformer: A price–volume factor stock selection model based on wavelet transform and multi-task self-attention networks. Expert Syst. Appl. 273, 126803. http://dx.doi.org/10.1016/j.eswa.2025.126803.

Maiti, R., Menon, B.G., Abraham, A., 2024. Ensemble empirical mode decomposition based deep learning models for forecasting river flow time series. Expert Syst. Appl. 255, 124550. http://dx.doi.org/10.1016/j.eswa.2024.124550.

Martinho, A.D., Ribeiro, C.B.M., Gorodetskaya, Y., Fonseca, T.L., Goliatt, L., 2020. Extreme learning machine with evolutionary parameter tuning applied to forecast the daily natural flow at cahora bassa dam, mozambique. In: Filipič, B., Minisci, E., Vasile, M. (Eds.), Bioinspired Optimization Methods and their Applications. pp. 255–267. http://dx.doi.org/10.1007/978-3-030-63710-1_20.

Medrano-Diaz, M., Rodriguez-Rangel, H., Puig-Cayuela, V., Flores, J.J., Lopez-Farias, R., Lara-Alvarez, C., 2025. A deep learning approach for image time series forecasting: Study case, United States drought monitor. Eng. Appl. Artif. Intell. 158, 111346. http://dx.doi.org/10.1016/j.engappai.2025.111346.

Moreno, S.R., Seman, L.O., Stefenon, S.F., dos Santos Coelho, L., Mariani, V.C., 2024. Enhancing wind speed forecasting through synergy of machine learning, singular spectral analysis, and variational mode decomposition. Energy 292, 130493. http://dx.doi.org/10.1016/j.energy.2024.130493.

Moriano, P., Berres, A., Xu, H., Sanyal, J., 2024. Spatiotemporal features of traffic help reduce automatic accident detection time. Expert Syst. Appl. 244, 122813. http://dx.doi.org/10.1016/j.eswa.2023.122813.

Muniz, R.N., Buratto, W.G., Nied, A., Cardoso, R., Finardi, E.C., Gonzalez, G.V., 2025. Time series forecasting of natural inflow in hydroelectric power plants using hyper-tuned temporal fusion transformer with hodrick–prescott filter. IET Gener. Transm. Distrib. 19 (1), e70087. http://dx.doi.org/10.1049/gtd2.70087.

jing Niu, W., kai Feng, Z., Zeng, M., fei Feng, B., wu Min, Y., tian Cheng, C., zhong Zhou, J., 2019. Forecasting reservoir monthly runoff via ensemble empirical mode decomposition and extreme learning machine optimized by an improved gravitational search algorithm. Appl. Soft Comput. 82, 105589. http://dx.doi.org/10.1016/j.asoc.2019.105589.

Noor, F., Haq, S., Rakib, M., Ahmed, T., Jamal, Z., Siam, Z.S., Hasan, R.T., Adnan, M.S.G., Dewan, A., Rahman, R.M., 2022. Water level forecasting using spatiotemporal attention-based long short-term memory network. Water 14 (4), 612. http://dx.doi.org/10.3390/w14040612.

Paiva, L., Ferreira, H., Oliveira, R., 2023. Wavelet-Seq2Seq-LSTM with attention for time series forecasting of dam levels in hydroelectric power plants. Energy 280, 128452. http://dx.doi.org/10.1016/j.energy.2023.128452.

Pan, D., Liu, Y., Wei, Y., Liu, Y., 2025. A two-stage health prognostics with spatiotemporal feature representation and uncertainty quantification for bearings. Expert Syst. Appl. 268, 126111. http://dx.doi.org/10.1016/j.eswa.2024.126111.

Pang, B., Wei, W., Li, X., Feng, X., Li, C., 2023. A representation-learning-based approach to predict stock price trend via dynamic spatiotemporal feature embedding. Eng. Appl. Artif. Intell. 126, 106849. http://dx.doi.org/10.1016/j.engappai.2023.106849.

Praveen, B., Talukdar, S., Shahfahad, et al., 2020. Analyzing trend and forecasting of rainfall changes in India using non-parametrical and machine learning approaches. Sci. Rep. 10, 10342. http://dx.doi.org/10.1038/s41598-020-67228-7.

Ribeiro, M.H.D.M., da Silva, R.G., Moreno, S.R., Mariani, V.C., dos Santos Coelho, L., 2022. Efficient bootstrap stacking ensemble learning model applied to wind power generation forecasting. Int. J. Electr. Power Energy Syst. 136, 107712. http://dx.doi.org/10.1016/j.ijepes.2021.107712.

Rigby, A., Baker, U., Lindley, B., Wagner, M., 2024. Generation and validation of comprehensive synthetic weather histories using auto-regressive moving-average models. Renew. Energy 224, 120157. http://dx.doi.org/10.1016/j.renene.2024.120157.

Rong, Z., Sun, W., Xie, Y., Huang, Z., Chen, X., 2025. Mixture of experts leveraging informer and LSTM variants for enhanced daily streamflow forecasting. J. Hydrol. 653, 132737. http://dx.doi.org/10.1016/j.jhydrol.2025.132737.

Song, C., Lin, Y., Guo, S., Wan, H., 2020. Spatial-temporal synchronous graph convolutional networks: A new framework for spatial-temporal network data forecasting. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 34, pp. 914–921.

Stefenon, S.F., Cristoforetti, M., Cimatti, A., 2025. Automatic digitalization of railway interlocking systems engineering drawings based on hybrid machine learning methods. Expert Syst. Appl. 281, 127532. http://dx.doi.org/10.1016/j.eswa.2025.127532.

Stefenon, S.F., Seman, L.O., Aquino, L.S., dos Santos Coelho, L., 2023. Wavelet-Seq2Seq-LSTM with attention for time series forecasting of level of dams in hydroelectric power plants. Energy 274, 127350. http://dx.doi.org/10.1016/j.energy.2023.127350.

Stefenon, S.F., Seman, L.O., da Silva, E.C., Finardi, E.C., dos Santos Coelho, L., Mariani, V.C., 2024a. Hypertuned wavelet convolutional neural network with long short-term memory for time series forecasting in hydroelectric power plants. Energy 313, 133918. http://dx.doi.org/10.1016/j.energy.2024.133918.

Stefenon, S.F., Seman, L.O., da Silva, L.S.A., Mariani, V.C., dos Santos Coelho, L., 2024b. Hypertuned temporal fusion transformer for multi-horizon time series forecasting of dam level in hydroelectric power plants. Int. J. Electr. Power Energy Syst. 157, 109876. http://dx.doi.org/10.1016/j.ijepes.2024.109876.

Sun, F., Hao, W., Zou, A., Shen, Q., 2024. A survey on spatio-temporal series prediction with deep learning: taxonomy, applications, and future directions. Neural Comput. Appl. 36 (17), 9919–9943. http://dx.doi.org/10.1007/s00521-024-09659-1.

Takara, L.A., Teixeira, A.C., Yazdanpanah, H., Mariani, V.C., S. Coelho, L., 2024. Optimizing multi-step wind power forecasting: Integrating advanced deep neural networks with stacking-based probabilistic learning. Appl. Energy 369, http://dx.doi.org/10.1016/j.apenergy.2024.123487.

Tan, C., Gao, Z., Li, S., Li, S.Z., 2025. SimVPv2: Towards simple yet powerful spatiotemporal predictive learning. IEEE Trans. Multimed. 27, 5170–5184. http://dx.doi.org/10.1109/TMM.2025.3543051.

Tebong, N.K., Simo, T., Takougang, A.N., Ntanguen, P.H., 2023. STL-decomposition ensemble deep learning models for daily reservoir inflow forecast for hydroelectricity production. Heliyon 9 (6), e16456. http://dx.doi.org/10.1016/j.heliyon.2023.e16456.

Tian, K., Yang, J., Cheng, L., 2024. Deep learning model for the deformation prediction of concrete dams under multistep and multifeature inputs based on an improved autoformer. Eng. Appl. Artif. Intell. 137, 109109. http://dx.doi.org/10.1016/j.engappai.2024.109109.

Tong, H., 2023. Functional linear regression with huber loss. J. Complexity 74, 101696. http://dx.doi.org/10.1016/j.jco.2022.101696.

Wang, Y., Wu, H., Zhang, J., Gao, Z., Wang, J., Yu, P.S., Long, M., 2022. Predrnn: A recurrent neural network for spatiotemporal predictive learning. IEEE Trans. Pattern Anal. Mach. Intell. 45 (2), 2208–2225. http://dx.doi.org/10.1109/TPAMI.2022.3165153.

Wei, C., Pi, D., Ping, M., Zhang, H., 2023. Short-term load forecasting using spatial-temporal embedding graph neural network. Electr. Power Syst. Res. 225, 109873. http://dx.doi.org/10.1016/j.epsr.2023.109873.

Wen, X., Li, W., 2023. Time series prediction based on LSTM-attention-LSTM model. IEEE Access 11, 48322–48331. http://dx.doi.org/10.1109/ACCESS.2023.3276628.

Wu, Z., Pan, S., Long, G., Jiang, J., Zhang, C., 2019. Graph WaveNet for deep spatial-temporal graph modeling. In: Proceedings of the 28th International Joint Conference on Artificial Intelligence. IJCAI, pp. 1907–1913.

Xu, L., Chen, N., Chen, Z., Zhang, C., Yu, H., 2021. Spatiotemporal forecasting in earth system science: Methods, uncertainties, predictability and future directions. Earth-Sci. Rev. 222, 103828. http://dx.doi.org/10.1016/j.earscirev.2021.103828.

Xu, X., Wang, Z., Zhou, F., Huang, Y., Zhong, T., Trajcevski, G., 2023. Dynamic transformer ODEs for large-scale reservoir inflow forecasting. Knowl.-Based Syst. 276, 110737. http://dx.doi.org/10.1016/j.knosys.2023.110737.

Yin, L., Xie, J., 2021. Multi-temporal-spatial-scale temporal convolution network for short-term load forecasting of power systems. Appl. Energy 283, 116328. http://dx.doi.org/10.1016/j.apenergy.2020.116328.

Yoosuf, S., Baali, H., Bouzerdoum, A., 2025. Improving perceptual quality in spatiotemporal timeseries forecasting. Eng. Appl. Artif. Intell. 156, 111062. http://dx.doi.org/10.1016/j.engappai.2025.111062.

Yu, B., Yin, H., Zhu, Z., 2018. Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting. In: Proceedings of the 27th International Joint Conference on Artificial Intelligence. IJCAI, pp. 3634–3640.

Zanfei, A., Brentan, B.M., Menapace, A., Righetti, M., Herrera, M., 2022. Graph convolutional recurrent neural networks for water demand forecasting. Water Resour. Res. 58 (7), http://dx.doi.org/10.1029/2022WR032299, e2022WR032299.

Zhang, Y., Shou, Y., Ai, W., Meng, T., Li, K., 2025. LRA-GNN: Latent relation-aware graph neural network with initial and dynamic residual for facial age estimation. Expert Syst. Appl. 273, 126819. http://dx.doi.org/10.1016/j.eswa.2025.126819.

Zhong, Z., Fan, N., Wu, L., 2024. Multistage stochastic optimization for mid-term integrated generation and maintenance scheduling of cascaded hydroelectric system with renewable energy uncertainty. European J. Oper. Res. 318 (1), 179–199. http://dx.doi.org/10.1016/j.ejor.2024.05.011.

Zhou, Z., Basker, R., Yeung, D.-Y., 2025. Graph neural networks for multivariate time-series forecasting via learning hierarchical spatiotemporal dependencies. Eng. Appl. Artif. Intell. 147, 110304. http://dx.doi.org/10.1016/j.engappai.2025.110304.

Zhou, F., Wang, Z., Chen, D., Zhang, K., 2023. Reservoir inflow forecasting in hydropower industry: A generative flow-based approach. IEEE Trans. Ind. Informatics 19 (2), 1196–1206. http://dx.doi.org/10.1109/TII.2022.3158927.