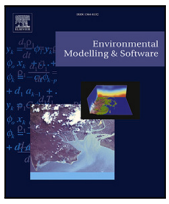
Environmental Modelling and Software 199 (2026) 106938

Contents lists available at ScienceDirect

Environmental Modelling and Software

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



AGFormer: Adaptive Spatiotemporal graph informed transformer for multi-reservoir inflow forecasting☆

Ming Fan [a,*,1], Pengfei Hu [a,b,1], Xiaoxue Han [b], Wei Zhang [a], Hyun Kang [a], Yue Ning [b], Dan Lu [a]

[a] Computational Sciences and Engineering Division, Oak Ridge National Laboratory, Oak Ridge, TN, 37831, USA
[b] Department of Computer Science, Stevens Institute of Technology, Hoboken, NJ, 07030, USA

ARTICLE INFO

Keywords:
Multi-reservoir inflow forecasting
Adaptive graph learning
Graph attention network
Transformer
Semi-supervised pretraining

ABSTRACT

Accurate reservoir inflow forecasting is crucial for effective water resource management, yet most machine learning models focus on single-reservoir prediction and overlook spatial dependencies among hydrologically connected reservoirs. We propose AGFormer (Adaptive Graph-Informed Transformer), an end-to-end framework that integrates adaptive graph learning with temporal sequence modeling for multi-reservoir inflow forecasting. A shared encoder and graph attention mechanism generate reservoir-specific embeddings, which are then processed by the Transformer-based encoder–decoder for multi-step inflow forecasting. We also introduce a pretraining paradigm to learn robust temporal embeddings from misaligned historical records. Evaluated on 30 reservoirs in the Upper Colorado River Basin, AGFormer achieves superior seven-day-ahead forecasts, with NSE > 0.75 for 20 reservoirs—outperforming Encoder–Decoder LSTM, GCN+LSTM, and Transformer baselines. Adaptive graph learning captures dynamic inter-reservoir dependencies, and feature attribution aligns with snowmelt-driven hydrology. Incorporating forecasted meteorological inputs further enhances accuracy, demonstrating AGFormer's potential to support reservoir management under dynamic hydrological conditions.

1. Introduction

Accurate reservoir inflow forecasting is essential for effective water resource management, enabling operators to balance competing demands for flood control, hydropower generation, irrigation, municipal water supply, and ecosystem protection (Yousefi et al., 2022; Apaydin et al., 2020; Yang et al., 2019). The accuracy of these forecasts directly influences reservoir operation efficiency and the reliability of regional water systems under varying hydroclimatic conditions (Fan et al., 2023c). As climate variability intensifies and extreme events become more frequent, advancing multi-step inflow prediction methods is essential for developing adaptive and data-informed water management strategies (Yousefi et al., 2023).

Process-based hydrological models, which simulate physical interactions such as precipitation and runoff, have traditionally been the primary approach to inflow prediction (Bennett et al., 2016). While these models provide valuable insights into underlying hydrological mechanisms and can achieve high accuracy when properly calibrated for specific catchments, they require extensive meteorological and physiographic data inputs along with site-specific parameter calibration (Kratzert et al., 2019). With the growing availability of large hydrological datasets, data-driven approaches have emerged as a variable alternative for inflow forecasting. These methods learn direct mappings from hydrometeorological inputs to streamflow without requiring explicit representation of physical processes (Kratzert et al., 2018).

Early applications of machine learning (ML) to inflow forecasting include support vector machines, decision tree ensembles, random forest, and nonlinear regression models, frequently augmented with signal processing techniques such as wavelet transforms or ensemble methods like bootstrap aggregation (Allawi et al., 2018; Nourani et al., 2021; Latif and Ahmed, 2023). Despite their computational efficiency

* Corresponding author.
E-mail address: fanm@ornl.gov (M. Fan).

[1] These authors contributed equally to this work.

https://doi.org/10.1016/j.envsoft.2026.106938
Received 12 November 2025; Received in revised form 14 February 2026; Accepted 27 February 2026
Available online 27 February 2026

and ease of implementation, conventional ML models exhibited limitations in capturing the nonlinear temporal dependencies characteristic of hydrological systems, resulting in deteriorating performance for multi-step-ahead forecasting scenarios (Bernardes, Jr. et al., 2022).

Deep learning (DL) methods have significantly advanced hydrological forecasting by providing more advanced capabilities to model complex temporal patterns in inflow dynamics. Convolutional neural networks (CNNs) and recurrent neural networks (RNNs), particularly long short-term memory (LSTM) networks, have demonstrated substantial improvements in predictive accuracy compared to traditional ML approaches (F. Li et al., 2024; Khorram and Jehbez, 2023). Among these architectures, the encoder–decoder LSTM (ED-LSTM) has been widely adopted for multi-step hydrological forecasting applications due to its strong temporal modeling capabilities (Kao et al., 2020; Fan et al., 2023a,b). The success of encoder–decoder frameworks stems from their ability to compress historical information into a latent representation (encoding phase) and subsequently decode this representation into future predictions, making them particularly suitable for sequence-to-sequence forecasting tasks. Recent research has extended these concepts through attention mechanisms and Transformer-based architectures, including dynamic Transformers (Xu et al., 2023) and temporal fusion Transformers (Muniz et al., 2025), further demonstrating the effectiveness of encoder–decoder paradigms for long-term hydrological prediction (Zhao et al., 2024).

Despite these advances, most existing ML and DL models are limited to single-reservoir forecasting, capturing only temporal dynamics while overlooking critical spatial dependencies among hydrologically connected systems. Reservoirs within the same river basin are influenced by shared weather patterns and upstream-downstream dynamics. Ignoring these interdependencies can compromise predictive accuracy and limit forecasting skill at the regional scale. Therefore, moving from site-specific models toward regional models that exploit both temporal and spatial dependencies holds substantial potential for enhancing predictive performance.

Graph neural networks (GNNs) provide a natural framework for modeling spatiotemporal dependencies in hydrological systems. They explicitly capture spatial relationships by treating reservoirs as nodes and their connections as edges, while simultaneously learning temporal patterns from the time-series data associated with each node. This structure allows explicit learning of correlated inflow patterns and connectivity effects across sites (Sun et al., 2021; Liu et al., 2022, 2023; Akkala et al., 2025). However, early applications have shown mixed results. For example, Sun et al. (2021) evaluated several recurrent GNN architectures for streamflow prediction on the Catchment Attributes and Meteorology for Large-Sample Studies (CAMELS) dataset and found that models constructed using standard GNN layers performed worse than a baseline LSTM network. Similarly, Liu et al. (2023) used a GNN to explore the impact of adding different spatial connections, such as hydrological–hydrological, hydrological–meteorological, and meteorological–meteorological, for streamflow forecasting and observed only marginal improvements over LSTM benchmarks.

The limited success of existing GNN approaches stems from two fundamental challenges related to architectural design and data utilization. First, from an architectural perspective, GNN performance is highly sensitive to graph topology. Graphs that are too dense may introduce irrelevant dependencies and cause over-smoothing, while overly sparse graphs can hinder effective information propagation (Wu et al., 2020; Longa et al., 2023). Although recent studies have explored dynamic graph learning to represent evolving connectivity, they often depend on computationally expensive fully connected base graphs or random dropout strategies that lack hydrological interpretability (Sun et al., 2022; Jiang et al., 2024). A critical gap therefore remains in developing architectures that can adaptively learn sparse and interpretable structures from data, retaining only the most informative hydrological connections while pruning irrelevant noise. Second, from a data utilization perspective, a major barrier to regional multi-reservoir forecasting lies in the heterogeneity of historical records. Multi-reservoir models typically require temporally aligned datasets, meaning that only dates common to all sites can be used for training. This requirement forces researchers to discard valuable long-term records from older reservoirs to accommodate sites with shorter observational histories, substantially limiting both the volume of training data and the model's exposure to historical hydroclimatic variability. To date, few frameworks provide an effective mechanism to exploit these temporally misaligned and heterogeneous records within a unified training pipeline.

To address these limitations, we propose AGFormer, an end-to-end DL framework for multi-step inflow forecasting in interconnected reservoir systems. AGFormer is designed to learn both spatial inter-reservoir dependencies and temporal dynamics directly from data, without relying on a fixed river topology or a fully connected graph. AGFormer employs an adaptive graph learning mechanism to model evolving inter-reservoir connectivity driven by changing hydrometeorological conditions. Specifically, we initialize a physically plausible superset graph using static geographic constraints, and then use a graph attention network (GAT) to estimate edge importance weights that quantify the predictive relevance of each candidate connection. During early training, edges with persistently low attention (aggregated across time steps, heads, and layers) are progressively masked using conservative thresholds, yielding a sparse and interpretable connectivity structure that reduces redundancy and mitigates over-smoothing. The resulting spatially contextualized embeddings are subsequently processed by a Transformer encoder–decoder to model temporal dependencies and generate multi-step inflow forecasts. To address limited and uneven historical records across reservoirs, we further develop a semi-supervised pretraining strategy that leverages temporally misaligned observations by pretraining exclusively on valid temporal windows prior to the common overlapping period, providing a robust initialization for downstream supervised training.

We evaluate AGFormer using inflow records from the Upper Colorado River Basin and compare against representative baselines, including ED-LSTM, GCN+LSTM, and Transformer architectures. Beyond forecasting accuracy, AGFormer supports interpretation through edge-level connectivity patterns and feature-level attributions, offering practical insights into dominant inter-reservoir influences and basin-wide dynamics. The key contributions of this work are:

1. **Adaptive spatiotemporal learning.** We introduce an attention-based adaptive graph learning strategy that refines a physically plausible superset graph into a sparse, time-varying and interpretable connectivity structure, and integrate it with Transformer-based temporal modeling for multi-reservoir inflow forecasting.
2. **Semi-supervised pretraining.** We propose a pretraining strategy that leverages temporally misaligned multi-reservoir records by using all valid historical windows prior to the overlapping period, enabling robust representation learning from heterogeneous records.
3. **Built-in interpretability.** We provide interpretable spatial dependencies through learned edge importance patterns and quantify feature contributions via attribution analysis, facilitating insight into dominant hydrometeorological influences.

2. Methods

2.1. Problem setup and notation

We consider inflow forecasting over a system of N interconnected reservoirs. For each reservoir $i \in \{1, \dots, N\}$ and day t, let $\mathbf{x}_{i,t} \in \mathbb{R}^F$ denote the F-dimensional vector of hydrometeorological inputs, and let $y_{i,t} \in \mathbb{R}$ denote the corresponding reservoir inflow. Given an input window of length T days ending at forecast initialization time t, the task is to predict inflow at lead times $h = 1, \dots, K$:

$$\{\mathbf{x}_{i,t-T:t-1}\}_{i=1}^{N} \longrightarrow \{\hat{y}_{i,t+h}\}_{i=1}^{N}, \quad h = 1, \dots, K. \tag{1}$$

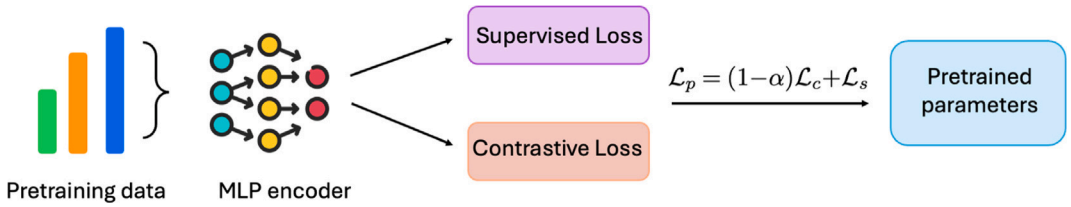


Fig. 1. Pretraining framework for multi-reservoir inflow forecasting. A shared temporal MLP encoder is trained with both supervised and contrastive losses to capture general hydrological patterns while preserving reservoir-specific characteristics, enabling effective use of misaligned historical records. The resulting pretrained parameters provide a strong initialization for downstream AGFormer training.

In this study, we use $N = 30$ reservoirs, $T = 30$ input days, and $K = 7$ lead times.

For compactness, we stack the inputs across reservoirs and time into a tensor $X \in \mathbb{R}^{N\times T\times F}$, where the feature vector at reservoir i and time t is given by $X_{i,t,:} = \mathbf{x}_{i,t}$ with $\mathbf{x}_{i,t} \in \mathbb{R}^F$.

2.2. Pretraining framework for multi-reservoir inflow forecasting

Multi-reservoir inflow forecasting requires temporal alignment of records across reservoirs, however, historical observations are often highly imbalanced. Some reservoirs have records spanning several decades, while others may only contain a decade of data. Standard training approaches typically restrict learning to the common overlapping period shared by all reservoirs, discarding a substantial portion of valuable information from longer records. To address this limitation, we introduce a semi-supervised pretraining framework designed to accommodate heterogeneous and temporally misaligned historical records.

The proposed framework is illustrated in Fig. 1. We employ a universal temporal Multilayer Perceptron (MLP) encoder that maps misaligned daily hydrometeorological records into a shared latent feature space. Because the encoder is shared across all reservoirs, it learns temporal representations that capture common hydrological dynamics despite differences in record length or overlap. Each input sample corresponds to a moving window (30 days in our case) of continuous records from a single reservoir. Because the encoder operates on local temporal windows rather than absolute calendar dates, samples from reservoirs with different record lengths can be used jointly. Misaligned datasets are therefore handled naturally by training on all valid temporal segments preceding the overlapping period for each reservoir, enabling the encoder to learn transferable temporal representations without requiring synchronized timestamps across sites.

The encoder is optimized jointly using supervised and contrastive objectives. The supervised component minimizes the mean squared error between predicted and observed inflows, while the contrastive component (InfoNCE loss; Oord et al., 2018) enforces reservoir-specific consistency—encouraging representations from the same reservoir to cluster together while separating those from different reservoirs. This hybrid training enables the model to capture both shared hydrological dynamics and reservoir-specific patterns. Formally, the overall training objective is a weighted combination of the supervised and contrastive terms:

$$\mathcal{L}_p = (1-\alpha)\mathcal{L}_c + \alpha\mathcal{L}_s, \tag{2}$$

where $\mathcal{L}_c$ denotes the contrastive InfoNCE loss, $\mathcal{L}_s$ is the supervised MSE loss, and α is a tunable weight balancing the two components. With $\alpha = 0.8$, the objective places primary emphasis on the supervised regression term $\mathcal{L}_s$, while using $\mathcal{L}_c$ as an auxiliary regularizer to encourage reservoir-consistent representations. This semi-supervised formulation enables the model to exploit temporally mismatched records that would otherwise be excluded. Additional details of the pretraining strategy and objective, including the full loss formulation and sampling procedure, are provided in Appendix.

After pretraining, the learned parameters are used to initialize AGFormer. This initialization captures general hydrological patterns across the basin while retaining reservoir-specific characteristics through the contrastive objective. Consequently, reservoirs with limited or missing records still benefit from a robust initialization rather than random weights. Overall, this framework effectively leverages heterogeneous observations, accelerates model convergence, and enhances forecasting robustness across diverse hydrological settings.

2.3. AGFormer architecture overview

AGFormer is a graph-based spatiotemporal forecasting framework designed to capture basin dynamics through a local–spatial–temporal hierarchy. As illustrated in Fig. 2, the pipeline consists of four sequential stages: (i) a shared feature extractor that maps raw inputs to latent node embeddings (local representation), (ii) an adaptive graph module that performs message passing to incorporate inter-reservoir context (spatial interaction), (iii) a Transformer-based temporal module that models temporal dependencies using the spatially enriched sequences (temporal dynamics), and (iv) a multi-step forecasting head that outputs K-day-ahead inflow predictions.

2.3.1. Data flow and tensor interfaces

Let $\mathbf{X} \in \mathbb{R}^{N\times T\times F}$ denote the model input. The shared feature extractor maps $\mathbf{X}$ to latent embeddings $\mathbf{H} \in \mathbb{R}^{N\times T\times d}$. The adaptive graph module performs message passing at each day t to produce spatially enriched embeddings $\tilde{\mathbf{H}} \in \mathbb{R}^{N\times T\times d}$. Finally, for each reservoir i, the temporal module processes the sequence $\tilde{\mathbf{H}}_i \in \mathbb{R}^{T\times d}$ and outputs a latent vector $\mathbf{Z}_i$, which is decoded into multi-step forecasts $\hat{y}_{i,t+1:t+K}$.

2.4. Shared feature extractor $\mathcal{F}_\theta(\cdot)$

The feature extractor maps raw hydrometeorological variables into a latent representation with dimension d using an MLP shared across reservoirs:

$$\mathbf{h}_{i,t} = \mathcal{F}_\theta(\mathbf{x}_{i,t}) = \mathrm{MLP}(\mathbf{x}_{i,t};\theta) \in \mathbb{R}^d. \tag{3}$$

Stacking embeddings over all reservoirs and time yields

$$\mathbf{H} = [\mathbf{h}_{:,1}, \ldots, \mathbf{h}_{:,T}] \in \mathbb{R}^{N\times T\times d}. \tag{4}$$

Interface to the graph module. The tensor $\mathbf{H}$ serves as the input node-feature sequence for the adaptive graph module, which models inter-reservoir information exchange at each time step.

Initialization via pretraining. The feature extractor parameters θ are initialized using our semi-supervised pretraining procedure designed to exploit heterogeneous, temporally misaligned historical records.

2.5. Adaptive graph learning $\mathcal{G}_\lambda(\cdot)$

2.5.1. Graph semantics: Physical entities vs. message passing

Before introducing the learning mechanism, we clarify the meaning of nodes and edges in AGFormer. Nodes correspond to physical reservoirs, each associated with a state embedding derived from local

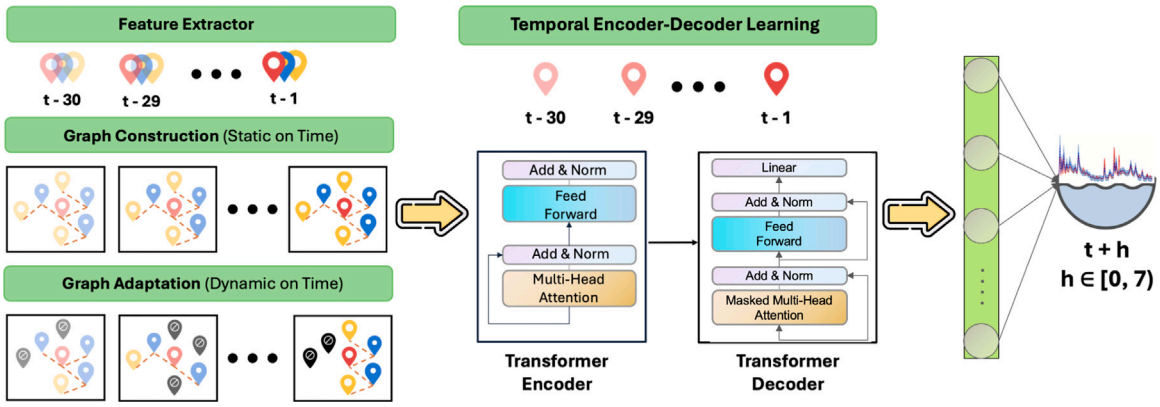


Fig. 2. Schematic workflow of AGFormer. Given an input window of length T (here $T = 30$), daily hydrometeorological variables for N reservoirs are indexed as $\{t-T,\dots,t-1\}$ relative to the forecast initialization time t. Location-pin icons represent reservoirs (graph nodes). The feature extractor maps raw inputs to latent embeddings $\mathbf{H} \in \mathbb{R}^{N\times T\times d}$, which are passed to the graph attention module. Graph construction provides an initial candidate adjacency used by the graph module and subsequent pruning. Graph attention produces spatially enriched embeddings $\tilde{\mathbf{H}} \in \mathbb{R}^{N\times T\times d}$, and the sequence for each reservoir is then processed by the Transformer encoder–decoder to generate multi-step inflow forecasts $\{\hat{y}_{t+h}\}_{h=1}^{7}$. Inside the Transformer, the decoder uses causal (triangular) masking in self-attention to prevent attending to future decoder positions, and "Add & Norm" denotes a residual connection followed by layer normalization.

hydrometeorological conditions. Edges, in contrast, represent message-passing pathways that encode predictive dependencies between reservoirs rather than physical water routing. While learned dependencies often align with upstream–downstream routing, they may also reflect shared hydroclimatic forcing and basin-wide correlations. We include self-loops as part of the standard message-passing formulation so that each reservoir retains its own information when aggregating messages. Hydrologically, this corresponds to local persistence and memory effects (e.g., catchment storage and recession), which make recent inflow informative for near-term evolution.

We emphasize that the learned graph represents functional (predictive) connectivity. Because our targets are regulated inflows, this connectivity may reflect not only hydrological routing but also shared meteorological forcing and operational coupling, and should not be interpreted as a direct reconstruction of the physical river network.

2.5.2. Base graph construction

We initialize a directed candidate graph using geographic proximity and an elevation-consistency constraint as a simple proxy for downstream direction. The spatial location of reservoir i is given by $\mathbf{p}_i = (\mathrm{lat}_i, \mathrm{lon}_i)$ with elevation l_i. Pairwise distances are computed using the Haversine distance function (Qi et al., 2022):

$$d_{ij} = D(\mathbf{p}_i, \mathbf{p}_j), \qquad \delta_{ij} = \begin{cases} 1, & l_j < l_i, \\ 0, & \text{otherwise}, \end{cases} \tag{5}$$

where $D(\cdot)$ returns the scalar distance d_{ij} between reservoirs i and j, and $\delta_{ij} \in \{0,1\}$ is a binary directional indicator based on elevation.

For each reservoir i, we first select its k nearest reservoirs by distance to form a candidate set C_i. We then retain only elevation-consistent downstream candidates to form the neighbor set $\mathcal{N}_i = \{j \in C_i : \delta_{ij} = 1\}$. The adjacency matrix $\mathbf{A} \in \{0,1\}^{N\times N}$ is defined by $A_{ij} = 1$ if $j \in \mathcal{N}_i$ and $A_{ij} = 0$ otherwise, such that $\mathcal{N}_i$ directly specifies the outgoing edges of node i.

We emphasize that elevation consistency does not guarantee true hydrological connectivity. Therefore, this procedure intentionally defines a conservative superset of plausible connections, which is subsequently refined by the adaptive pruning mechanism during training. The resulting $\mathbf{A}$ is a binary candidate adjacency that defines the initial graph topology with allowable edges. During message passing, the node features vary by day, and the graph attention module produces time-varying edge weights $\alpha_{ij,t}$ on this topology.

2.5.3. Graph attention propagation

Given node embeddings $\mathbf{h}_{i,t}$ at day t, we apply a two-layer graph attention network (GAT) to aggregate information from neighbors:

$$\begin{aligned} e_{ij,t} &= \mathrm{LeakyReLU}\big(\mathbf{a}^{\top}[\mathbf{W}\mathbf{h}_{i,t} \| \mathbf{W}\mathbf{h}_{j,t}]\big), \\ \alpha_{ij,t} &= \frac{\exp(e_{ij,t})}{\sum_{j'\in\mathcal{N}_i}\exp(e_{ij',t})}, \\ \tilde{\mathbf{h}}_{i,t} &= \sigma\left(\frac{1}{M}\sum_{m=1}^{M}\sum_{j\in\mathcal{N}_i}\alpha_{ij,t}^{(m)}\mathbf{W}^{(m)}\mathbf{h}_{j,t}\right), \end{aligned} \tag{6}$$

where M is the number of attention heads, $\mathbf{W}^{(m)} \in \mathbb{R}^{d\times d}$ and $\mathbf{a} \in \mathbb{R}^{2d}$ are learnable parameters, $[\cdot\|\cdot]$ denotes concatenation, and σ is a ReLU activation.

Interface to the temporal module. Applying Eq. (6) across all $t = 1,\dots,T$ yields the spatially enriched tensor $\tilde{\mathbf{H}} \in \mathbb{R}^{N\times T\times d}$. For each reservoir i, we extract the sequence $\tilde{\mathbf{H}}_i \in \mathbb{R}^{T\times d}$, which is then passed to the temporal encoder–decoder.

2.5.4. Dynamic graph adaptation via monotonic pruning

To refine the initial connectivity, we adaptively prune edges using temporally averaged attention. For each edge (i,j), we compute the mean attention at day t by averaging over graph layers and attention heads:

$$\bar{\alpha}_{ij,t} = \frac{1}{LM}\sum_{\ell=1}^{L}\sum_{m=1}^{M}\alpha_{ij,t}^{(\ell,h)}, \qquad \bar{\alpha}_{ij} = \frac{1}{T}\sum_{t=1}^{T}\bar{\alpha}_{ij,t}. \tag{7}$$

Edges with $\bar{\alpha}_{ij} < \tau$ are masked, updating the adjacency matrix $\mathbf{A}$. Following a stabilization strategy, pruning is restricted to early training and the final topology is fixed for the remainder of optimization (see Section 3.4 for the schedule and thresholds used in this study).

2.6. Temporal encoder–decoder $\mathcal{T}_\omega(\cdot)$

After graph-based feature extraction, each reservoir i is represented by a temporally ordered sequence of spatially contextualized embeddings $\tilde{\mathbf{H}}_i \in \mathbb{R}^{T\times d}$. We model temporal dependencies by applying a Transformer-based encoder–decoder module $\mathcal{T}_\omega(\cdot)$ independently to each reservoir:

$$\mathbf{Z}_i = \mathcal{T}_\omega(\tilde{\mathbf{H}}_i), \qquad \mathbf{Z}_i \in \mathbb{R}^{K\times d}, \tag{8}$$

where K denotes the number of forecast lead times. The hth row of $\mathbf{Z}_i$, denoted by $\mathbf{z}_{i,h} \in \mathbb{R}^d$, is the lead-time-specific latent state used to generate the inflow prediction $\hat{y}_{i,t+h}$, for $h = 1,\dots,K$.

Unlike the graph module, which aggregates across reservoirs at a fixed time step, the temporal module operates along the time dimension for each reservoir. Using a Transformer also maintains an attention-based modeling paradigm across both spatial and temporal components, improving robustness to dynamically evolving feature representations induced by adaptive graph learning and avoids mixing attention-based spatial modeling with LSTM-based recurrent temporal mechanisms. Standard Transformer components, including multi-head self-attention, position-wise feedforward layers, add-and-norm operations, and the output projection layers, follow Vaswani et al. (2017) and are not repeated here.

2.7. Multi-step forecasting head

Given the lead-time-specific latent states $\mathbf{Z}_i = [\mathbf{z}_{i,1}, \ldots, \mathbf{z}_{i,K}]^\top \in \mathbb{R}^{K\times d}$, a shared forecasting head maps each $\mathbf{z}_{i,h}$ to the corresponding inflow prediction:

$$\hat{y}_{i,t+h} = \mathrm{MLP}(\mathbf{z}_{i,h}), \qquad h = 1, \ldots, K. \tag{9}$$

In our experiments, $K = 7$.

2.8. End-to-end training and inference

AGFormer is trained end-to-end using a mean squared error loss aggregated over reservoirs and forecast lead times:

$$\mathcal{L} = \frac{1}{NK}\sum_{i=1}^{N}\sum_{h=1}^{K}\left(\hat{y}_{i,t+h} - y_{i,t+h}\right)^2. \tag{10}$$

During training, dynamic graph refinement is interleaved with parameter optimization: attention scores are accumulated (averaged over the input window and across attention heads/layers) and edges are pruned according to the early-stage schedule described in Section 3.4. This allows the model to progressively focus on the most informative inter-reservoir connections. During inference, the final pruned graph topology is fixed and the model produces K-step forecasts for all reservoirs in a single forward pass.

End-to-end procedure (summary). For each training batch: (1) compute latent embeddings $\mathbf{H} = \mathcal{F}_\theta(\mathbf{X})$; (2) apply graph message passing to obtain $\tilde{\mathbf{H}} = \mathcal{G}_\lambda(\mathbf{H}, \mathbf{A})$; (3) compute $\mathbf{Z}_i = \mathcal{T}_\omega(\tilde{\mathbf{H}}_i)$ and forecasts $\{\hat{y}_{i,t+h}\}_{h=1}^{K}$ for all reservoirs; (4) update parameters by minimizing Eq. (10); and (5) update $\mathbf{A}$ via early-stage pruning when applicable.

2.9. AGFormer with future meteorology

To integrate future meteorological information, we modified the final prediction layer of the AGFormer framework, as illustrated in Fig. 3. Future meteorological data were obtained from the ERA5 reanalysis dataset, which provides globally consistent medium-range weather forecasts (Muñoz-Sabater et al., 2021). Instead of directly mapping the Transformer decoder's latent representation to inflow predictions, we enhanced this representation by incorporating forecasted meteorological conditions. Specifically, the hidden state, which encodes learned historical spatiotemporal patterns, was concatenated with corresponding temperature and precipitation forecast vectors for the prediction horizon. The resulting augmented feature vector was then passed through the final MLP layer to generate inflow forecasts. This modification allows the model to combine its learned representation of historical spatiotemporal patterns with exogenous meteorological forecasts, effectively improving predictive skill into longer horizons.

2.10. Integrated Gradients for model interpretation

To interpret the predictions of AGFormer, we employ the Integrated Gradients (IG) method, a gradient-based attribution technique that quantifies the contribution of each input feature to the model output. Rather than relying on local gradients at a single input point, IG attributes feature importance by accumulating gradients along a continuous path from a reference input to the observed input, providing a stable and axiomatically grounded explanation of model behavior.

Formally, the IG attribution for input feature i is computed by integrating the gradients of the model output with respect to that feature along a straight-line path from a baseline input x' to the actual input x:

$$\hat{\boldsymbol{\Phi}}_i(x) = (x_i - x_i') \times \sum_{k=1}^{n} \frac{\partial F(x' + \frac{k}{n} \times (x - x'))}{\partial x_i} \times \frac{1}{n} \tag{11}$$

where: F denotes the AGFormer model, k indexes the Riemann sum steps, and n is the number of steps (set to 200 in this study). The baseline input x' is chosen as the zero vector, following the standard IG formulation in Sundararajan et al. (2017).

3. Data preparation and training details

3.1. Study area and data

The Upper Colorado River Basin, encompassing portions of Wyoming, Utah, Colorado, and New Mexico, serves as a major headwater region for the Colorado River. In this region, hydrologic conditions are primarily driven by winter snow accumulation and subsequent spring melt; however, increasing climate variability has introduced additional challenges, including altered snowpack levels, shifts in runoff timing, and more frequent droughts. Reservoirs within the Upper Colorado River Basin provide essential services, including municipal and agricultural water supply for nearly 40 million people, hydropower production, irrigation, flood mitigation, and recreational opportunities (Fan et al., 2022, 2023b).

In this study, we select 30 reservoirs distributed across the Upper Colorado River Basin. Reservoirs were selected based on record completeness (allowing at most ten missing daily values) and spatial representativeness across a range of elevations (Fig. 4). For the few short gaps that remained (≤10 days per reservoir), missing inflow values were filled using a moving-average scheme to obtain continuous daily sequences.

Historical inflow observations were retrieved from the U.S. Bureau of Reclamation's water operations archive (Fan et al., 2022). It is important to note that these inflow records reflect regulated reservoir inflows derived from operations-based records in managed systems, and may therefore embed signatures of upstream reservoir operations (e.g., storage and release decisions) in addition to natural hydroclimatic forcing. Key characteristics of the reservoirs, such as name, record length, elevation, and storage capacity, are listed in Table 1. The dataset spans a wide range of hydrologic settings, with inflow records varying from 13 to 30 years. Meteorological forcings were obtained from the PRISM-based AN81d dataset (Daly and Bryant, 2013), which provides gridded daily precipitation and temperature fields at 4 km resolution. For each reservoir, daily time series were generated by averaging the grid cells overlapping the reservoir's contributing area, covering the period from January 1, 1982, to December 31, 2011.

3.2. Evaluation metrics

To evaluate predictive performance, we employ the Nash–Sutcliffe Efficiency (NSE) coefficient, a widely used measure in hydrologic modeling. NSE compares simulated inflows against their observed values by examining both the magnitude of deviations and the degree to which

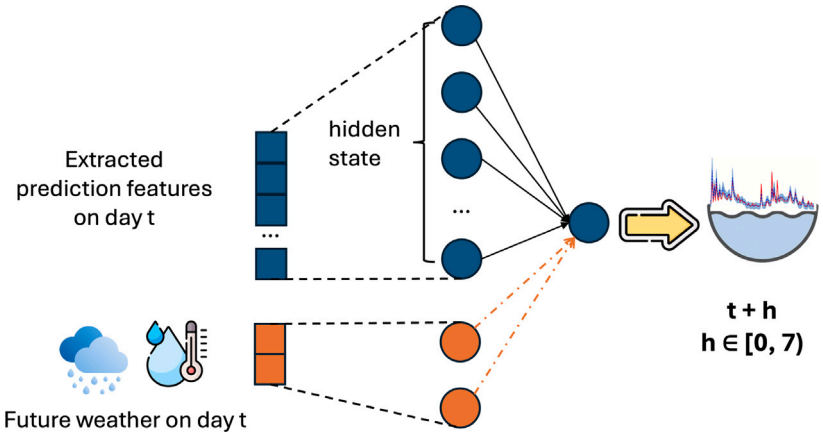


Fig. 3. Schematic illustration of the modified prediction module for incorporating future meteorological forecasts. The latent feature vector extracted by the Transformer decoder is concatenated with corresponding temperature and precipitation forecasts. The resulting augmented feature vector is then passed through an MLP to generate inflow predictions.

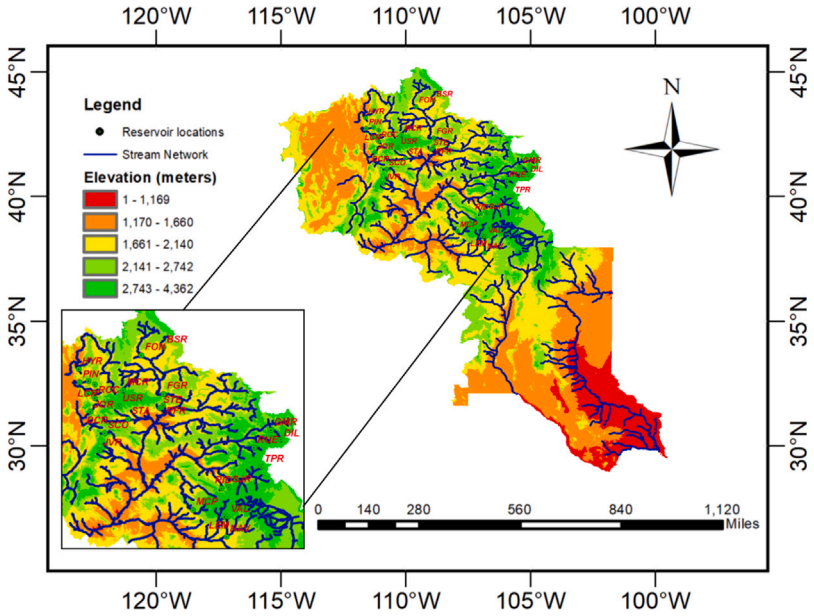


Fig. 4. Geographic distribution of the 30 study reservoirs across the Upper Colorado River Basin. Reservoirs span a wide range of elevations and geographic settings, providing broad spatial coverage for inflow forecasting.

model predictions capture variability in the observed data. The metric is defined as

$$\mathrm{NSE}(\hat{y}_t, y_t) = 1 - \frac{\sum_{t=1}^{n}(y_t - \hat{y}_t)^2}{\sum_{t=1}^{n}(y_t - \bar{y})^2}, \tag{12}$$

where y_t represents the observed inflow at time t, $\hat{y}_t$ is the corresponding prediction, $\bar{y}$ denotes the mean of the observations, and n is the number of samples. NSE values range from $-\infty$ to 1, with 1 indicating perfect agreement between predicted and observed inflows. Following the guidelines proposed by Moriasi et al. (2007), model performance can be interpreted based on the NSE values as follows: predictions are considered *very good* when $NSE > 0.75$; *good* when $0.65 < NSE \leq 0.75$; *satisfactory* when $0.50 < NSE \leq 0.65$; *acceptable* when $0.40 < NSE \leq 0.50$; and *unsatisfactory* when $NSE \leq 0.40$.

3.3. Baseline models

To evaluate the contribution of adaptive (time-varying) graph learning, we benchmark AGFormer against a compact set of representative DL baselines that cover the key modeling choices relevant to this study: (i) temporal sequence models without spatial message passing and (ii) spatiotemporal graph models with a fixed topology. This baseline set is intentionally designed to isolate the impact of learning dynamic inter-reservoir connectivity, rather than to provide an exhaustive comparison across all spatiotemporal forecasting architectures.

- **ED-LSTM** A classical sequence-to-sequence model based on LSTM units. The architecture consists of an encoder that ingests past observations of inflow, precipitation, and temperature, followed by a decoder that predicts future inflows (Fan et al., 2023b). While parameter sharing across reservoirs is allowed, each reservoir is essentially modeled as an independent univariate system. Thus, ED-LSTM captures temporal dynamics but lacks any mechanism for cross-reservoir interaction.
- **Transformer** A self-attention based model where recurrence is replaced with multi-head attention (Vaswani et al., 2017). In our implementation, daily input features for each reservoir are passed through 4 encoder layers with 8 attention heads, and forecasts are generated via a decoder of equal depth. Although the Transformer excels at learning long-range temporal dependencies,

Table 1
Information of the 30 reservoirs analyzed in this study.

Initials	Names	Data start year	Data length (years)	Elevation (m)	Storage (10^6 m^3)
BSR	Big Sandy Reservoir	1990	22	2060	47
CAU	Causey Reservoir	1999	13	1745	11
CRY	Crystal Reservoir	1982	30	2251	32
DCR	Deer Creek Reservoir	1987	25	1653	188
DIL	Dillon Reservoir	1985	27	2751	317
ECH	Echo Reservoir	1982	30	1691	91
ECR	East Canyon Reservoir	1992	20	1749	61
FGR	Flaming Gorge Reservoir	1982	30	1828	4674
FON	Fontenelle Reservoir	1990	22	1976	426
GMR	Green Mountain Reservoir	1982	30	2406	189
HYR	Hyrum Reservoir	1999	13	1427	23
JOR	Jordanelle Reservoir	1997	15	1636	395
JVR	Joes Valley Reservoir	1996	16	2129	77
LCR	Lost Creek Reservoir	1998	14	1829	15
LEM	Lemon Reservoir	1982	30	2478	50
MCP	Mcphee Reservoir	1991	21	2073	470
MCR	Meeks Cabin Reservoir	1998	14	2647	40
NAV	Navajo Reservoir	1986	26	1801	1724
PIN	Pineview Reservoir	1990	22	1495	136
RFR	Red Fleet Reservoir	1989	23	1721	32
RID	Ridgway Reservoir	1990	22	2101	105
ROC	Rockport Reservoir	1982	30	1807	75
RUE	Ruedi Reservoir	1982	30	2349	126
SCO	Scofield Reservoir	1996	16	2338	91
SJR	Silver Jack Reservoir	1992	20	2725	17
STA	Starvation Reservoir	1982	30	1700	206
STE	Steinaker Reservoir	1982	30	1655	41
TPR	Taylor Park Reservoir	1982	30	2847	1375
USR	Upper Stillwater Reservoir	1991	21	2445	40
VAL	Vallecito Reservoir	1986	26	2318	160

it also treats each reservoir as isolated, thereby ignoring the spatial connectivity inherent to the basin.

- **GCN + LSTM** A hybrid architecture that combines graph convolutional layers with sequence modeling (Yu et al., 2017). First, node features (reservoir states) are transformed using a graph convolution defined by a static adjacency matrix. The spatial features are then processed by an ED-LSTM to produce forecasts. This design incorporates spatial information, but the graph structure is fixed and uniform, meaning that temporal variability in reservoir interactions is not captured.

In addition to these external baselines, we include an internal ablation (AGFormer with a fixed graph, i.e., no adaptation) to directly quantify the contribution of the adaptive graph refinement mechanism under an otherwise identical architecture.

3.4. Training details

We forecast 7-day-ahead reservoir inflow for 30 reservoirs across the Upper Colorado River Basin, using the preceding 30 days of hydrometeorological variables as model input. Historical record lengths vary substantially across reservoirs, ranging from 13 to 30 years. To exploit these heterogeneous records without using any data from the common 13-year evaluation window, we pretrain AGFormer's feature extractor on all available windows that occur strictly prior to the start of the 13-year overlapping period shared by all reservoirs.

For supervised forecasting, we then train and evaluate AGFormer and all baseline models on the same 13-year overlapping period to ensure a consistent train/test split: the first 10 years are used for training and the remaining 3 years for testing. Because the semi-supervised pretraining objective is designed specifically for AGFormer's shared feature extractor and graph-based aggregation, applying the same procedure to ED-LSTM or Transformer baselines would require substantial architectural modifications (e.g., introducing shared cross-reservoir encoders and explicit spatial aggregation), thereby redefining these baselines. We therefore keep the baselines in their standard formulations and report AGFormer results both with and without pretraining to isolate the contribution of the pretraining stage.

To leverage the longer, non-overlapping records, we pretrain a two-layer MLP (128 units per layer) to generate daily reservoir embeddings, which serve as initialization for AGFormer's feature extractor $\mathcal{F}_\theta$. This procedure allows the MLP encoder to learn transferable temporal representations from heterogeneous datasets rather than relying solely on the limited overlapping period. The pretrained weights of $\mathcal{F}_\theta$ are then used to initialize the end-to-end AGFormer model, improving convergence and robustness, particularly for reservoirs with shorter records.

The full AGFormer architecture consists of three main components: (1) the feature extractor $\mathcal{F}_\theta$, (2) the adaptive graph module $\mathcal{G}_\lambda$, and (3) the temporal encoder–decoder $\mathcal{T}_\omega$. The embedding dimensions for the three main components $\mathbf{h}_{i,t}, \tilde{\mathbf{h}}_{i,t}, \mathbf{z}_i$ are 128, 128, and 64, respectively. For graph construction, daily graphs G_t were initialized using the $k = 5$ nearest neighbors for each node, with self-loops allowed for all 30 reservoirs. The graph module $\mathcal{G}_\lambda$ is a two-layer GAT, each layer employing four attention heads and 128 hidden units in total. Edge refinement was applied during the first three epochs: edges with mean attention weights below thresholds $\tau \in [0.1, 0.2, 0.3]$ were progressively masked. This progressive pruning strategy serves as a warm-up phase, preventing premature removal of potentially informative edges while attention weights are still stabilizing. Restricting pruning to the early epochs also ensures that the graph topology converges quickly and remains fixed for the majority of training (You et al., 2020; Chen et al., 2021), allowing the temporal encoder to learn on a stable and optimized spatial structure without disruption from continuous topological changes. Empirically, we observed that both the learned graph structure and validation performance stabilized after the initial three epochs, with no consistent benefit from continued pruning in later training stages.

The temporal predictor $\mathcal{T}_\omega$ adopts a Transformer encoder–decoder architecture, each composed of two stacked blocks. Each block employs multi-head attention with four heads and includes a feed-forward sublayer with a hidden dimension of 256. Dropout with a rate of 0.2 is

Table 2
Overall and per-day NSE for all models. Numbers in parentheses show the standard deviation over five independent runs.

Model	Overall	Day 1	Day 2	Day 3	Day 4	Day 5	Day 6	Day 7
ED–LSTM	0.8947 (1.65)	0.9568 (0.94)	0.9387 (2.96)	0.9149 (1.12)	0.8927 (0.87)	0.8710 (1.01)	0.8527 (0.60)	0.8363 (0.67)
Transformer	0.8790 (0.43)	0.9170 (2.16)	0.9161 (1.91)	0.8951 (2.44)	0.8812 (1.21)	0.8618 (2.71)	0.8502 (1.71)	0.8320 (2.28)
GCN+LSTM	0.8829 (0.46)	0.9409 (2.82)	0.9228 (2.75)	0.9018 (1.65)	0.8820 (1.17)	0.8620 (0.35)	0.8440 (2.88)	0.8266 (1.99)
AGFormer	**0.9145** (2.60)	**0.9620** (1.12)	**0.9505** (1.88)	**0.9331** (2.56)	**0.9176** (1.95)	**0.8978** (0.95)	**0.8803** (0.38)	**0.8600** (2.77)
AGFormer (High-flow top 5%)	0.9113 (3.20)	0.9702 (1.51)	0.9560 (2.24)	0.9347 (3.01)	0.9156 (2.47)	0.8916 (1.32)	0.8692 (0.73)	0.8418 (3.48)

applied in both the graph module $\mathcal{G}_\lambda$ and the Transformer $\mathcal{T}_\omega$ to enhance regularization.

AGFormer and the baseline models were trained on the combined dataset of 30 reservoirs. Optimization was performed using Adam with an initial learning rate of 10^{-3}, reduced by a factor of 0.5 after each epoch. A batch size of 4 was used throughout. All experiments were implemented in Python 3.10.17 and PyTorch 2.5.1, with graph operations supported by PyTorch Geometric 2.6. Training was executed on a high-performance node equipped with three AMD 64-core CPUs and eight AMD MI250X GPUs.

4. Results

In this section, we evaluate AGFormer against three baseline models (ED-LSTM, GCN+LSTM, and Transformer) over a seven-day forecast horizon for all 30 reservoirs. We first compare overall predictive accuracy, then analyze the adaptive graph learning capabilities, assess the contribution of different hydrometeorological variables, and finally examine the impact of future meteorological information on forecasting accuracy.

4.1. Overall model performance

Table 2 summarizes the mean NSE across the 30 reservoirs over the seven-day forecast horizon (mean over five runs; standard deviation in parentheses). AGFormer attains the highest overall NSE of 0.9145, exceeding the three baseline methods by approximately 2.0–3.5%. The performance gap is modest at Day 1 but widens from Day 5 onward. Specifically, AGFormer sustains NSE values of 0.8803 on Day 6 and 0.8600 on Day 7, whereas the best baseline declines to 0.8527 and 0.8363, respectively. Among the baselines, ED-LSTM performs best, highlighting the importance of modeling temporal dependencies. In contrast, the lower scores of GCN+LSTM and the Transformer suggest that increased architectural complexity alone does not guarantee improved performance and further motivate topology-aware design. Overall, these results indicate that combining learned spatial dependencies with temporal modeling is effective for multi-reservoir inflow forecasting.

To assess model behavior during operationally critical extremes, we report a regime-specific metric in addition to the overall NSE. Specifically, we compute High-flow NSE by restricting the evaluation to days whose observed inflow falls within the top 5% of the test-period distribution for each reservoir. NSE is then computed using only this high-flow subset, following the same definition as Eq. (12) but applied to the filtered samples. The results show that AGFormer maintains strong skill in this peak-flow regime, indicating improved robustness during event-driven conditions.

Fig. 5 summarizes NSE performance across all reservoirs and forecast lead times. As expected, forecast accuracy declines with increasing lead time for all models. However, AGFormer consistently outperforms the baselines, maintaining more reservoirs in the "very good" category (NSE > 0.75) throughout the forecast horizon. On day one, all models achieved strong performance, with most reservoirs classified as "very good". By day four, AGFormer retained 28 reservoirs in this category, compared with 27 for ED-LSTM, 23 for GCN+LSTM, and 18 for the Transformer. By day seven, AGFormer still achieved "very good" performance on 20 reservoirs, while ED-LSTM dropped to 15, GCN+LSTM to 11, and Transformer to only 10, with another 10 falling to "satisfactory" or "acceptable". Overall, AGFormer shows greater robustness to increasing lead time, with its first "satisfactory" prediction not appearing until day six. This indicates stronger generalization and stability compared to both sequence-only (ED-LSTM, Transformer) and graph-enhanced (GCN+LSTM) baselines.

To provide a reservoir-level perspective, Fig. 6 presents the day-by-day NSE performance for each of the 30 reservoirs. The results show that forecast skill varies considerably across sites, while AGFormer delivers the most consistent performance. Specifically, AGFormer achieves "very good" forecasts across all seven days in 20 of the 30 reservoirs, compared to 15 for ED-LSTM, 11 for GCN+LSTM, and 10 for the Transformer. The baseline models show less stability across individual sites. For example, the Transformer model never reached the "very good" category for the HYR reservoir, where predictions were limited to the "good" and "satisfactory" ranges. A similar pattern was observed for the LEM and SCO reservoirs, where forecasts remained predominantly "satisfactory". The GCN+LSTM model exhibited the greatest performance variability: forecasts for the STA reservoir were consistently "unsatisfactory" across the full forecast horizon, and performance for the JOR and RFR reservoirs fell entirely within "good", "satisfactory", or "acceptable" ranges. In contrast, AGFormer provided "very good" or at least "good" forecasts for these more challenging reservoirs, demonstrating its robustness and ability to adapt across diverse hydrological conditions.

The comparison results indicate that architectural differences in handling spatiotemporal dependencies play a crucial role. ED-LSTM and Transformer treat each reservoir as an independent unit, overlooking the hydrological connectivity within the basin. While these models capture temporal dependencies, their inability to represent spatial interactions limits their accuracy in interconnected systems. The GCN+LSTM baseline represents a step forward by incorporating a fixed graph to model spatial relationships, but its predefined connectivity cannot capture evolving inter-reservoir interactions driven by hydroclimatic variability. This rigidity likely contributes to its inconsistent performance for reservoirs such as STA and JOR. By contrast, AGFormer's more stable performance stems from its core innovation: adaptive graph learning. By updating the inter-reservoir relationships at each time step, AGFormer captures evolving spatiotemporal dependencies and prioritizes the most informative connections under different hydrometeorological conditions.

4.2. Adaptive graph learning and connectivity analysis

4.2.1. Graph construction sensitivity analysis

To further investigate how adaptive connectivity contributes to model robustness, we analyze the sensitivity of AGFormer and the GCN+LSTM model to different graph construction strategies, as illustrated in Fig. 7. The results of experiments vary with the number of nearest reservoirs used to define the initial graph structure. AGFormer maintains stable performance across different initializations, while the GCN+LSTM model shows a sharp decline in NSE and a rise in MSE when the graph topology is perturbed. This contrast demonstrates that conventional GNNs are highly sensitive to predefined connectivity patterns. Static connectivity assumptions in the GCN+LSTM model reduce adaptability and may introduce subjective biases, leading to representation challenges under non-stationary hydrologic conditions (Sun et al.,

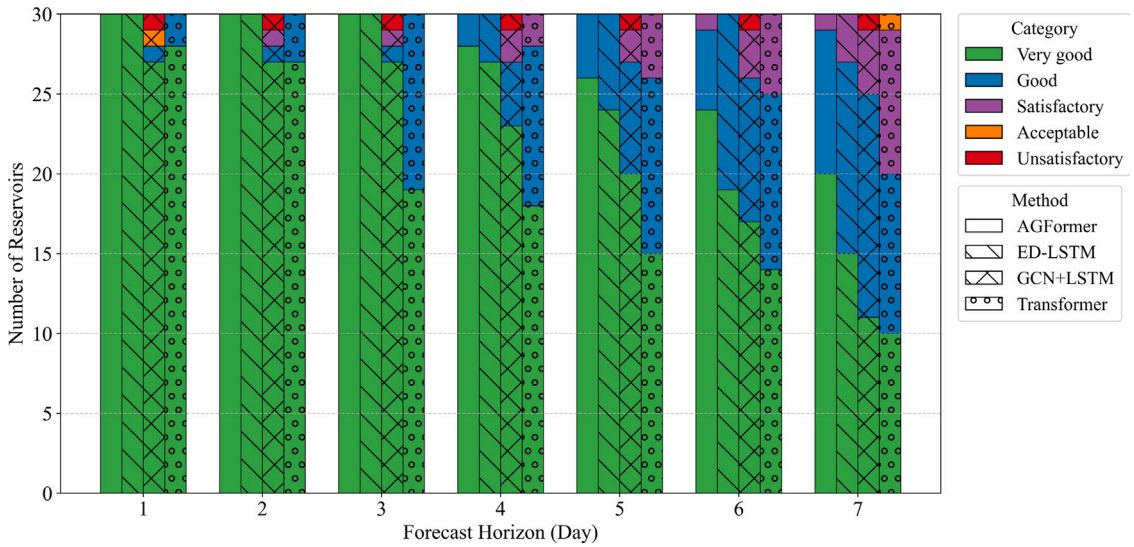


Fig. 5. Forecast performance based on NSE categories for the developed AGFormer model and three baseline models (ED-LSTM, GCN+LSTM, Transformer) over a seven-day forecast horizon. (Very good: NSE > 0.75; Good: 0.65 < NSE ≤ 0.75; Satisfactory: 0.5 < NSE ≤ 0.65; Acceptable: 0.4 < NSE ≤ 0.5; and Unsatisfactory: NSE ≤ 0.4).

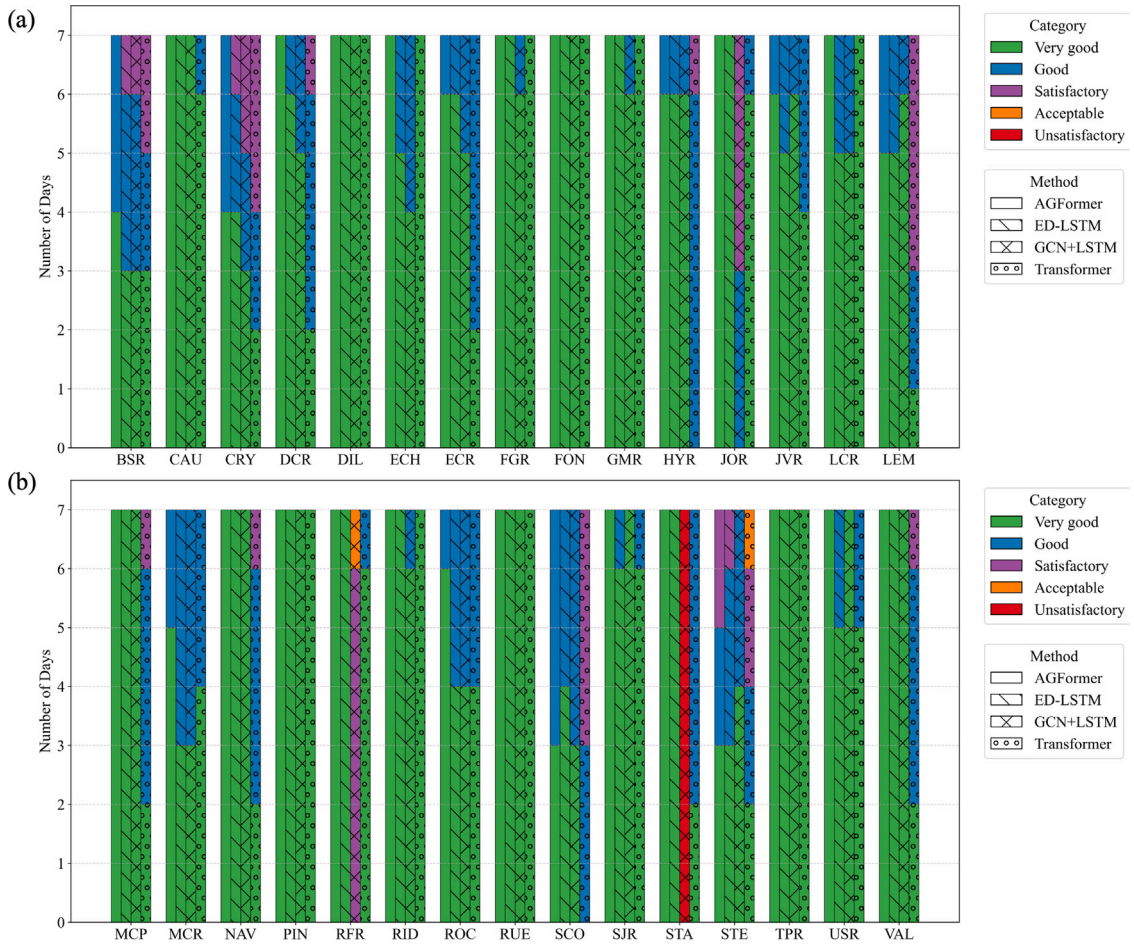


Fig. 6. Comparison of forecast performance for each of the 30 reservoirs over the seven-day forecast horizon, categorized by NSE ratings. Each bar group represents a reservoir, showing the number of forecast lead times where each model (AGFormer, ED-LSTM, GCN+LSTM, Transformer) achieved "very good", "good", "satisfactory", "acceptable", or "unsatisfactory" performance.

2022). In contrast, AGFormer's adaptive graph learning dynamically updates inter-reservoir relationships to reflect evolving hydrologic interactions. When this adaptive mechanism is disabled, as illustrated in Fig. 7, AGFormer's performance declines, confirming that flexible, data-driven connectivity is essential for robust and accurate spatiotemporal forecasting in complex river systems.

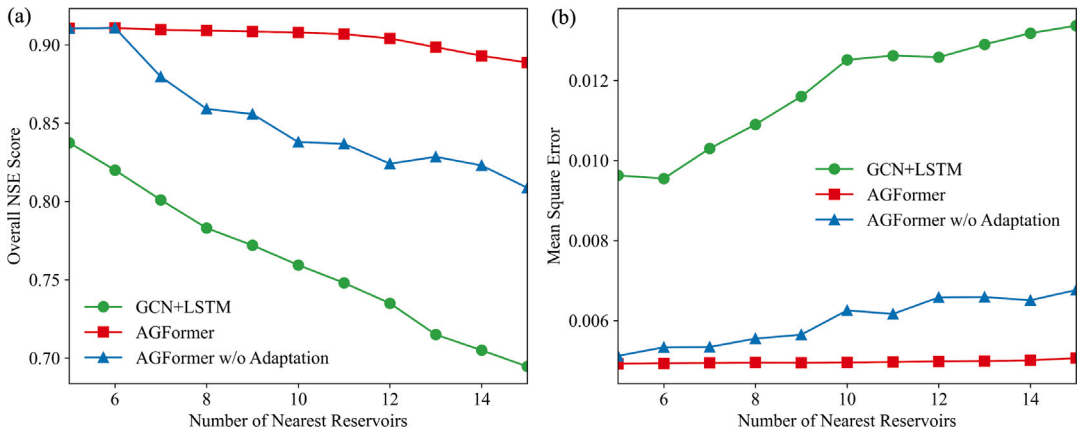


Fig. 7. Sensitivity of model performance to graph construction, measured by (a) NSE and (b) Mean Squared Error (MSE). The x-axis represents the number of nearest reservoirs used to define the initial graph structure. The NSE and MSE metrics are averaged over all 30 reservoirs and seven forecast horizons. The green line represents the GCN+LSTM baseline, the red line shows AGFormer with adaptive graph learning, and the blue line depicts AGFormer with the adaptive mechanism disabled.

4.2.2. Graph learning mechanism interpretation

Fig. 8 illustrates AGFormer's adaptive graph learning process in a six-reservoir subgraph, illustrating how the model automatically discovers optimal connectivity patterns for multi-reservoir inflow forecasting. The model begins from a physically plausible candidate graph (constructed from geographic constraints) that intentionally includes redundant connections. During training, attention scores then guide the refinement process: connections with consistently low weights are pruned, reflecting their limited predictive contribution. This sparsification is also evident at the basin scale: across all 30 reservoirs, the total number of edges (including self-loops) decreases over the first three epochs from 3090 to 2866, then to 2243, and finally to 1912. In the six-reservoir example, after the first epoch, four edges linked to reservoir ECH are pruned, and by the third epoch, the graph has adapted further, with edge counts varying across time lags (e.g., 7 at $t-1$, 8 at $t-15$, and 6 at $t-30$). This temporal variability suggests that not all topologically feasible connections contribute equally to forecasting skill.

These learned structures provide a data-driven view of functional connectivity that complements, but does not replace, interpretations based on river-network topology. In AGFormer, nodes represent reservoirs, while edges are message-passing pathways whose attention weights measure the predictive contribution of one reservoir to another under the current latent state, rather than physical water routing. Because attention is computed on high-dimensional embeddings that mix multiple drivers, edge weights need not vary monotonically with river distance or travel time and may reflect shared meteorological forcing or nonlocal hydroclimatic coherence in addition to upstream–downstream routing. We therefore caution against interpreting attention weights as direct estimates of routing strength or travel time.

A particularly noteworthy result is the progressive isolation of reservoir ECH, which retains only its self-loop connection after two epochs. This suggests that ECH's inflows are more strongly autocorrelated than spatially correlated within this subgraph. The final learned edge weights, represented by varying line thickness in the figure, encode the relative importance of inter-reservoir dependencies. Stronger weights likely correspond to more informative relationships, while weak or pruned edges indicate minimal predictive importance. These learned structures offer a data-driven perspective on connectivity, complementing physical assumptions derived from river network topology.

Multi-reservoir forecasting faces inherent challenges when relying on static graph structures, as reservoir interactions are dynamic and influenced by cascading operational effects (J. Li et al., 2024). Upstream releases directly alter downstream inflows, creating time-varying dependencies that fluctuate with management schedules, which are often uncertain or unavailable to forecasters. Moreover, adjacent sub-basins frequently exhibit correlated runoff behavior due to shared catchment characteristics, suggesting that valuable information exchange extends beyond the strict river topology (Weiler et al., 2003). These complexities motivate a data-driven approach to learning connectivity rather than relying on fixed, predefined river networks. AGFormer addresses these challenges by learning time-varying connection strengths through adaptive attention mechanisms, enabling the model to capture both upstream–downstream dependencies and shared hydrometeorological patterns across sub-basins. This adaptive graph learning strategy improves inflow prediction robustness in complex, non-stationary hydrologic systems.

4.3. Hydrometeorological variable importance analysis

In addition to analyzing adaptive spatial connectivity, we conducted feature attribution analysis to quantify the influence of individual hydrometeorological variables on forecasting performance. We applied IG, a gradient-based attribution method that assigns importance scores to input features according to their contributions to model predictions (Fan et al., 2023a).

Fig. 9 shows the aggregated IG scores across all 30 reservoirs, reflecting the relative importance of each input variable over the forecast horizon. Historical inflow clearly emerges as the dominant predictor, accounting for roughly 80% of total attribution, consistent with the strong autocorrelation structure commonly observed in hydrological time series. Among the meteorological drivers, temperature ranks second, followed by precipitation. This ordering aligns with the basin's snowmelt-dominated hydrological regime: in the Upper Colorado River Basin, nearly 70% of annual runoff originates from snowpack melt (Akkala et al., 2025), making temperature a key control on the timing and magnitude of inflows. The variable ranking is broadly consistent with the physical hydrology of the basin, where temperature-driven snowmelt plays a central role in runoff generation. This consistency suggests that AGFormer is capturing patterns that align with known hydrological processes.

4.4. Enhancement through future meteorological information

Forecast accuracy in snowmelt-dominated basins depends not only on historical hydrometeorological conditions but also on future weather dynamics, particularly precipitation and temperature variability (Fan et al., 2025). To evaluate whether such information can enhance multi-step inflow forecasting, we tested a modified version of AGFormer that explicitly incorporates forecasted meteorological variables into its prediction module.

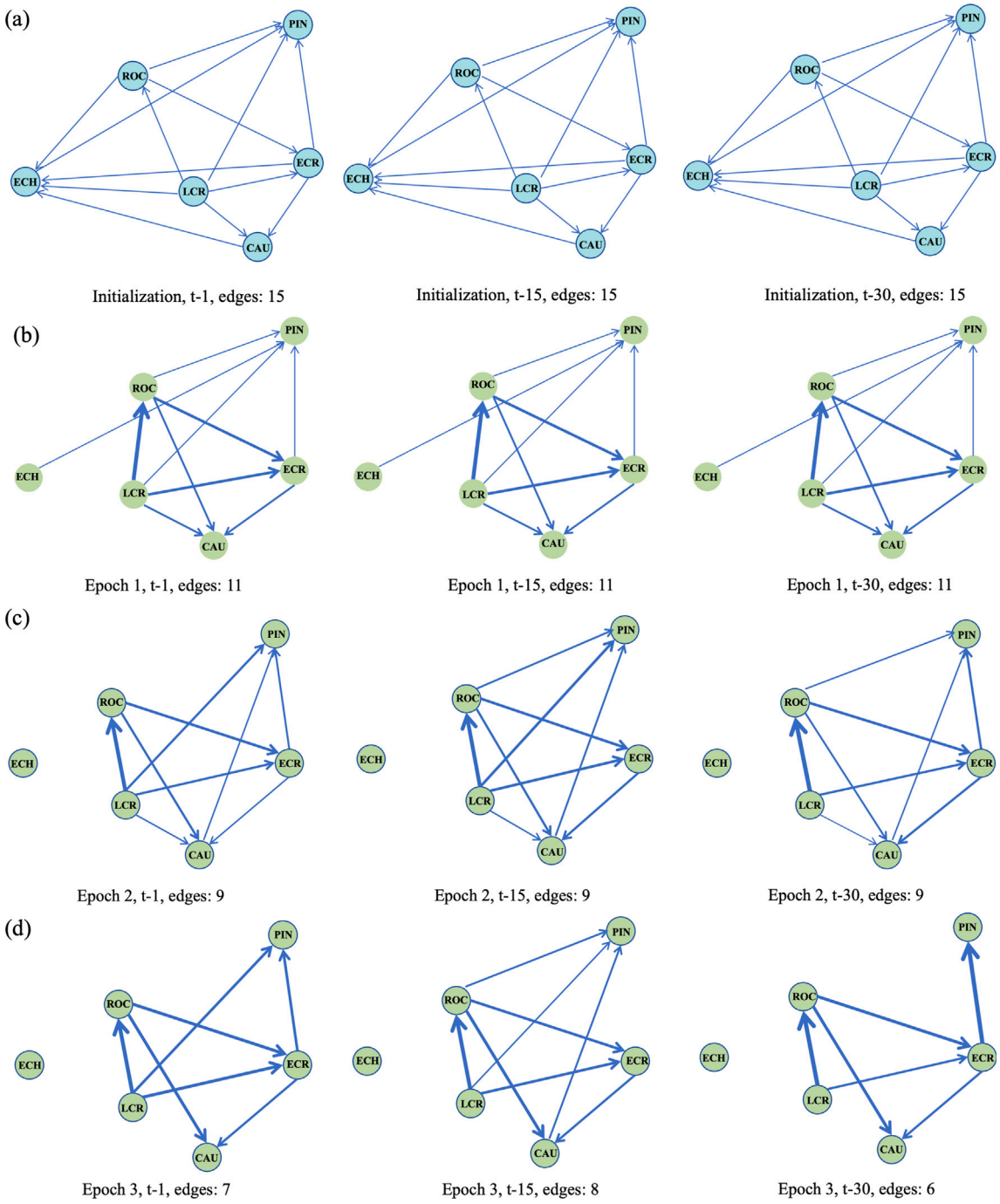


Fig. 8. Demonstration of the adaptive graph learning and edge pruning process on a subgraph of six reservoirs. The panels show the graph structure at different training stages: (a) initial elevation-based connectivity, and the adapted graph after (b) epoch 1, (c) epoch 2, and (d) epoch 3. The width of the edges is linearly correlated with the learned attention weights, indicating connection strength. Self-loops are not shown for clarity. Note learned edges reflect data-driven predictive (functional) dependencies and are not intended to represent physical hydraulic connectivity or river-network topology.

The inclusion of future meteorological forecasts demonstrates a substantial improvement in multi-step prediction accuracy, particularly at longer lead times, as shown in Fig. 10. During the first three forecast days, both the original and modified AGFormer models achieve "very good" performance across all reservoirs. Differences emerge at longer horizons: while the original model shows a gradual decline, with an increasing number of reservoirs classified as "good" or "satisfactory" by day seven, the meteorologically enhanced version sustains "very good" accuracy for almost all reservoirs. Only one reservoir drops into the "good" category on day seven, and no reservoirs fall below this level. These results indicate that access to future temperature and precipitation information helps the model maintain forecasting skill where reliance on historical patterns alone becomes insufficient. Because ERA5 is a retrospective reanalysis product rather than an operational forecast, the "future meteorology" experiment represents a perfect-forecast upper bound and is not directly indicative of real-time forecasting performance.

A more detailed per-reservoir analysis as illustrated in Fig. 11, confirms that improvements are systematic. Reservoirs that previously exhibited pronounced performance declines at longer horizons show the largest gains. For example, forecasts for reservoir STE, which fell to "satisfactory" in days six and seven with the original model, are

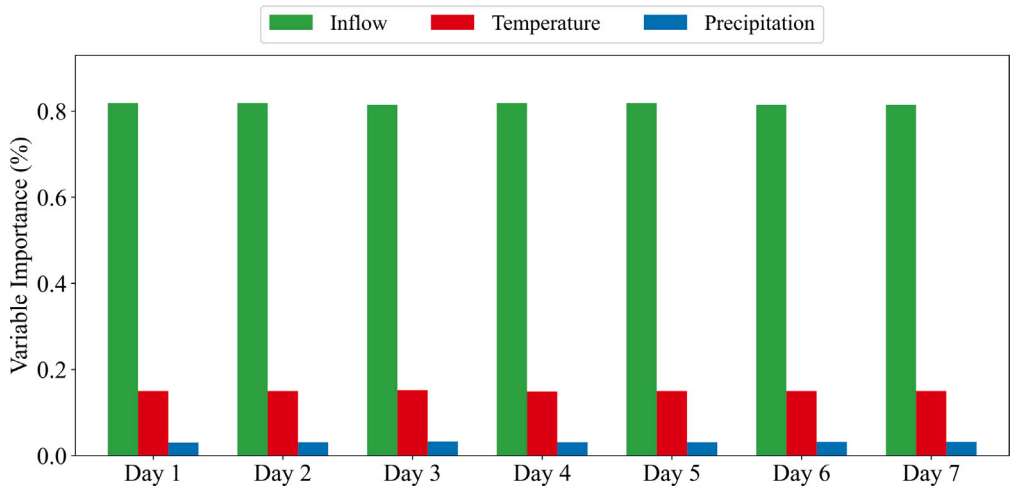


Fig. 9. Aggregated importance scores of hydrometeorological input variables for all 30 reservoirs, as determined by the IG method. The scores represent the relative contribution of each variable (inflow, temperature, precipitation) to the model's inflow forecasts.

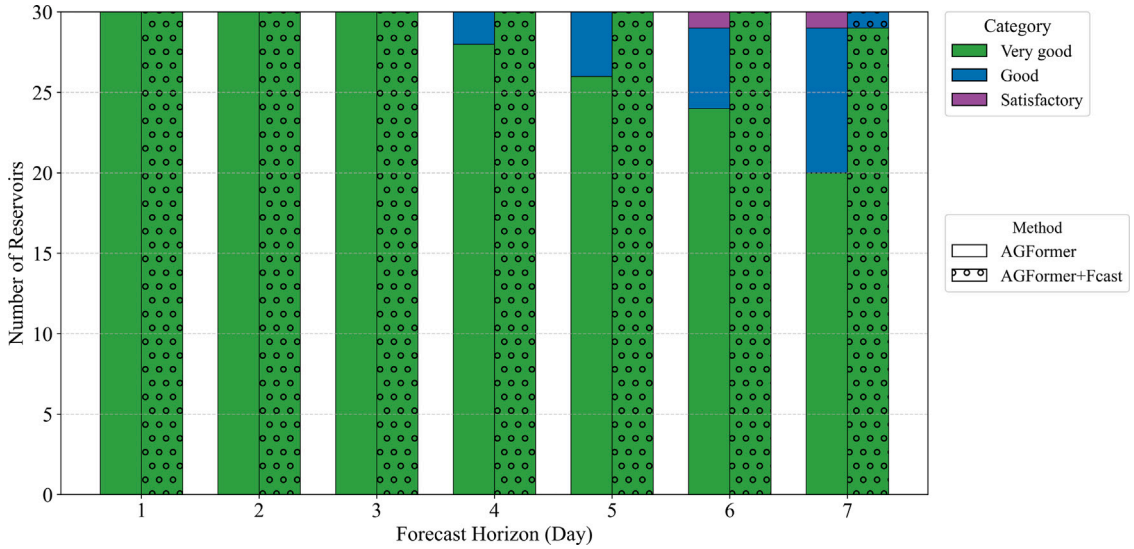


Fig. 10. Comparison of forecast performance between the original AGFormer and the modified version with future meteorological inputs. Categories are based on NSE thresholds: "very good", "good", and "satisfactory".

upgraded to "very good" through day six and "good" on day seven with the enhanced version. Similar patterns are observed for SCO, BSR, and CRY, which previously experienced multiple days of "good" performance, achieve consistent "very good" ratings throughout the forecast period when enhanced with future meteorological data. Across all reservoirs, the modified AGFormer eliminates all "satisfactory" ratings and reduces the total number of "good" ratings from 21 (in the original model) to just one.

These findings suggest that coupling learned spatiotemporal dynamics with meteorological forecasts provides a consistent advantage, particularly for medium-range horizons where the influence of evolving meteorological drivers becomes more pronounced. This enhanced forecasting capability has potential to support more adaptive and informed reservoir operations for flood control, drought mitigation, and hydropower optimization.

5. Discussion

This section examines the experimental findings to characterize the learned graph structures and to discuss limitations and operational implications of the proposed framework.

5.1. Interpreting the adaptive graph: Functional vs. Structural connectivity

A key contribution of AGFormer is its ability to infer inter-reservoir relationships from data rather than relying on a fixed, predefined river topology. Analysis of the learned attention weights highlights an important distinction between structural connectivity (the physical river network) and functional connectivity (predictive dependence used for message passing).

While the learned graph generally follows upstream–downstream relationships, the model sometimes assigns lower weights to physically adjacent reservoirs and higher weights to more distant ones. This suggests that the model prioritizes relationships that are informative for forecasting over simple geographic proximity. In particular, when multiple upstream reservoirs exhibit highly similar inflow dynamics, the attention mechanism tends to concentrate weight on a single representative reservoir while down-weighting others that provide redundant information. This reduction of redundant connections helps mitigate over-smoothing and illustrates that the learned connectivity reflects predictive relevance rather than a direct reconstruction of the static river network.

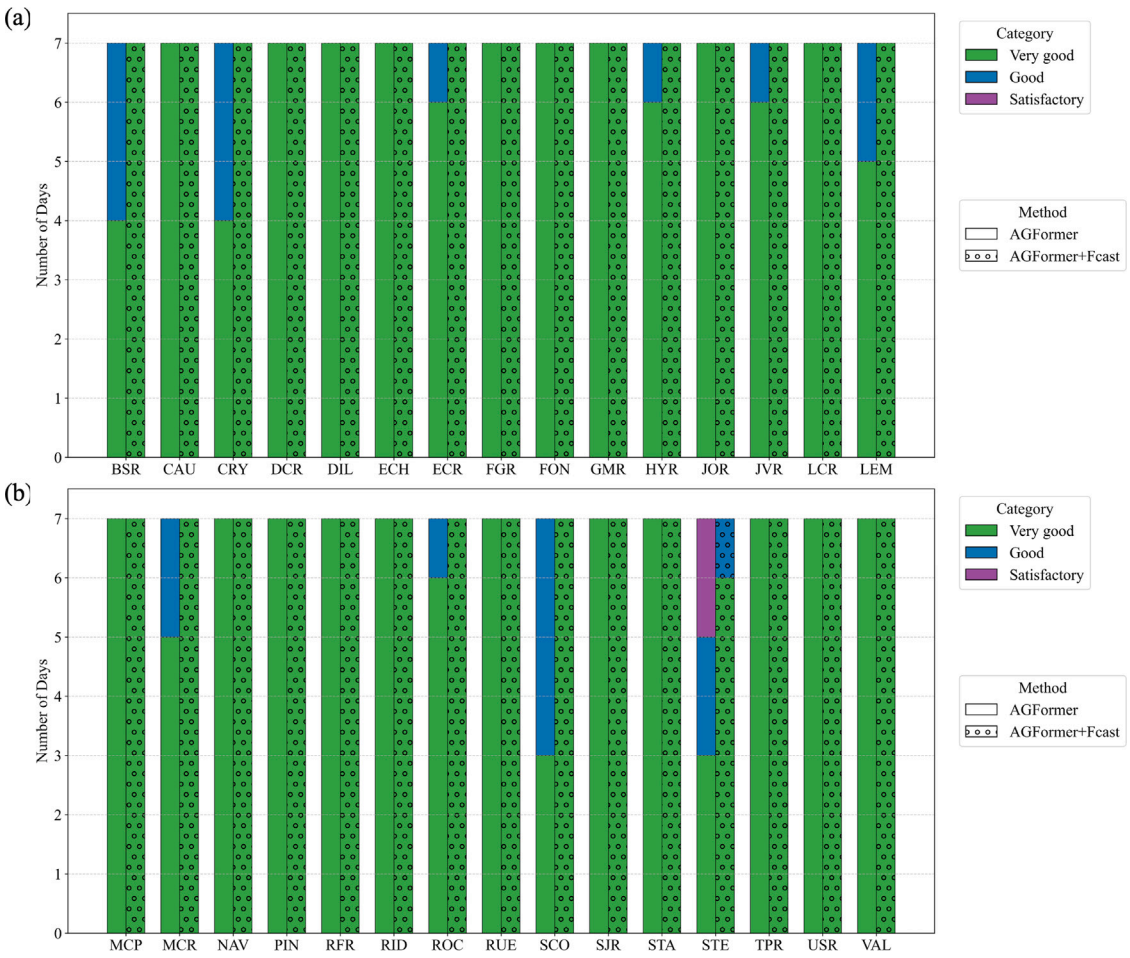


Fig. 11. Per-reservoir forecast performance comparison over seven days between the original AGFormer and the enhanced version with meteorological forecasts. Each bar group represents one of the 30 reservoirs, showing the number of days forecasts fell into each NSE performance category.

5.2. Regulated operations and data-driven connectivity

In the Upper Colorado River Basin, reservoir interactions are influenced by cascade dam operations, where upstream release decisions can modify downstream inflows and disrupt natural rainfall–runoff relationships. Although AGFormer does not ingest explicit operational control variables (e.g., release schedules or rule curves), its time-varying edge weights can reflect changes in inter-reservoir coupling under different system states. In this sense, the adaptive graph may capture aggregate operational influences indirectly through patterns embedded in regulated inflow records, contributing to robust performance in strongly regulated sub-basins. Incorporating explicit operational datasets, when available, is a promising direction to further improve interpretability and performance in highly managed systems.

5.3. Limitations and operational considerations

Despite its advantages, several limitations should be considered when applying AGFormer in operational settings.

Reliance on reanalysis data. This study uses ERA5 reanalysis as a proxy for future meteorological forcing, effectively assuming perfect knowledge of future weather conditions. In real-time forecasting scenarios based on Numerical Weather Prediction models, forecast uncertainty — particularly at longer lead times — is expected to reduce predictive skill. Consequently, results obtained with future meteorological inputs should be interpreted as an upper bound on achievable performance.

Monotonic graph pruning. The current adaptive graph strategy employs monotonic pruning based on temporally averaged attention weights and restricts pruning to the early training stage to stabilize the learned topology. While this design suppresses noisy connections and improves training stability, it may permanently remove edges that are generally weak but become critical during rare, high-impact events such as flash floods. Future work will explore event-aware or non-monotonic adaptation strategies (e.g., pooling or percentile-based aggregation of temporal attention) to preserve connections important for extremes, even if their average contribution is low.

Physical validation and constraints. A fully physical validation of the learned connectivity (e.g., correlation with routed travel times or strict upstream–downstream ordering) would require consistent basin-wide routing metadata and/or naturalized flow records that remove operational regulation signals. Because such data are not uniformly available for all reservoirs in this study, we interpret the learned graph primarily as functional (predictive) connectivity. Future work will incorporate external hydrographic constraints, when available, both to quantitatively validate the learned connectivity and to explore physically informed regularization that encourages consistency with known routing while retaining flexibility to capture nonlocal hydroclimatic dependencies.

Limited meteorological forcings. This study uses precipitation and temperature as meteorological inputs because they are consistently available across all reservoir catchments and throughout the analysis period. While these variables capture key controls on water input and snowmelt timing, additional snow- and energy-related drivers (e.g., snow-water equivalent, radiation, and humidity) may provide more physically direct information in snowmelt-dominated basins. Future work will assess

the benefits of incorporating SWE products and energy-balance proxies to improve physical interpretability and potentially enhance forecasting skill in snowmelt-sensitive catchments.

Operational data availability. Consistent basin-wide operational data sets (e.g., releases and rule curves) are not publicly available for all sites, and incorporating such information remains an important direction for future work.

6. Conclusions

In this study, we proposed AGFormer, a novel end-to-end DL framework for multi-step inflow forecasting in interconnected reservoir systems. The key contributions of this work are twofold: (i) an adaptive edge pruning mechanism that learns sparse, time-varying reservoir connectivity, and (ii) a semi-supervised pretraining strategy that enables effective use of temporally misaligned observations. Together, these design choices allow AGFormer to capture evolving inter-reservoir dependencies and shared hydroclimatic influences, overcoming the limitations of static graph representations. As a result, the proposed framework delivers more accurate and robust multi-reservoir forecasts under complex and non-stationary hydroclimatic conditions.

Our evaluation across 30 reservoirs in the Upper Colorado River Basin shows that AGFormer achieves stronger predictive skill than baseline models, maintaining 20 reservoirs in the "very good" category ($NSE > 0.75$) at day seven compared to 15, 11, and 10 for ED-LSTM, GCN+LSTM, and Transformer, respectively. The interpretable attention mechanism demonstrates how AGFormer dynamically prunes less informative connections, allowing the model to focus on inter-reservoir relationships that improve forecasts. Feature attribution analysis further indicates that AGFormer captures patterns consistent with the basin's snowmelt-driven hydrology, where historical inflow dominates, followed by temperature and precipitation. Furthermore, incorporating future meteorological forecasts improves long-horizon accuracy, eliminating all "satisfactory" performance instances and reducing "good" ratings from 21 to 1 across all reservoirs and forecast days.

Overall, AGFormer advances multi-reservoir inflow forecasting by combining predictive accuracy with interpretability in large reservoir networks. The adaptive graph structure provides insights into time-varying connectivity patterns, while variable importance analysis highlights connections to key hydrometeorological drivers. Although these interpretations are derived from data-driven mechanisms and should be viewed with caution, they nonetheless help reveal how the ML model organizes and prioritizes information across reservoirs. While our experiments focused on the Upper Colorado River Basin, further validation across basins with different climatic and hydrological characteristics is necessary to establish broader applicability. In addition, although pretraining helps address data imbalance among reservoirs, challenges remain in data-sparse regions where observations are highly limited or irregular. Future work will investigate domain adaptation and transfer learning strategies to improve robustness under diverse hydroclimatic settings.

CRediT authorship contribution statement

Ming Fan: Writing – review & editing, Writing – original draft, Visualization, Validation, Supervision, Software, Resources, Project administration, Methodology, Investigation, Funding acquisition, Formal analysis, Data curation, Conceptualization. **Pengfei Hu:** Writing – review & editing, Visualization, Validation, Software, Methodology, Investigation, Formal analysis, Conceptualization. **Xiaoxue Han:** Writing – review & editing, Methodology, Formal analysis. **Wei Zhang:** Writing – review & editing, Formal analysis, Data curation. **Hyun Kang:** Writing – review & editing, Formal analysis, Data curation. **Yue Ning:** Writing – review & editing, Investigation, Funding acquisition, Formal analysis. **Dan Lu:** Writing – review & editing, Resources, Project administration, Methodology, Funding acquisition, Formal analysis.

Software and data availability

- **Name of software:** AGFormer
- **Developers:** Ming Fan and Pengfei Hu
- **Contact:** fanm@ornl.gov, humphreyhuu@gmail.com
- **Date first available:** January 12, 2026
- **Programming language:** Python
- **Source code:** https://github.com/patrickfan/AGFormer
- **Documentation:** Detailed installation, testing, and deployment instructions are available at https://github.com/patrickfan/AGFormer/blob/main/README.md
- **Sample data:** https://github.com/patrickfan/EDLSTM-UQ-Explainability

Declaration of competing interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Acknowledgments

This research was supported by the Seedling Project, funded by the U.S. Department of Energy (DOE) Water Power Technologies Office, and by Dan Lu's Early Career Project, funded by the DOE Biological and Environmental Research Office. Additional support was provided through collaboration between the U.S. Air Force Life Cycle Management Center (LCMC) and Oak Ridge National Laboratory (ORNL). High-performance computing resources were provided by the Frontier system at the Oak Ridge Leadership Computing Facility, a DOE Office of Science User Facility. ORNL is managed by UT-Battelle, LLC, for the US DOE under contract DE-AC05-00OR22725. This work was also supported in part by the U.S. National Science Foundation under grants 2047843 and 2437621.

Appendix. Pretraining strategy for reservoir embeddings

A.1. Heterogeneous data handling

Multi-reservoir inflow forecasting typically requires temporal alignment across sites, yet historical records are heterogeneous: some reservoirs span decades, while others contain only a few years of data. Training solely on the strictly overlapping period would therefore discard a substantial portion of valuable observations. To leverage these heterogeneous records, we pretrain a shared MLP encoder to learn reservoir-level temporal embeddings. The heterogeneity arises primarily from differences in record start dates across reservoirs (e.g., some begin in 1982 while others begin in 1999), whereas end dates are aligned. Accordingly, for reservoirs with longer historical coverage, pretraining uses all valid temporal windows occurring strictly prior to the start of the 13-year overlapping period shared by all reservoirs. No data from the overlapping period are used in any form during pretraining.

A.2. Sampling strategy

During pretraining, we form mini-batches by sampling fixed-length windows (length T) from each reservoir's available sequence. To handle temporal misalignment, we precompute all valid windows from all reservoirs within the pretraining period, place them into a global pool, and shuffle them to form mini-batches. This strategy ensures that each mini-batch contains a diverse mixture of reservoirs and that data-scarce reservoirs contribute throughout pretraining. To ensure well-defined positive pairs in the contrastive loss (Section A.3), mini-batches are constructed such that each reservoir appearing in a batch contributes at least two windows. Overlapping windows are allowed; the contrastive objective is defined by reservoir identity rather than by non-overlapping window constraints.

A.3. Contrastive consistency loss ($\mathcal{L}_c$)

Each sampled window is associated with a reservoir identifier. We adopt a supervised contrastive formulation in which windows from the same reservoir form positive pairs, while windows from different reservoirs within the mini-batch serve as negatives. Let $\mathcal{B}$ denote the index set of windows in the current mini-batch and let $\mathbf{e}_i$ denote the ℓ_2-normalized embedding of window $i \in \mathcal{B}$. For an anchor i, we define the positive index set $\mathcal{P}(i) \subset \mathcal{B} \setminus \{i\}$ as the indices of other windows in the batch drawn from the same reservoir. Our batch construction ensures $\mathcal{P}(i) \neq \emptyset$ by including at least two windows per reservoir whenever the reservoir appears in a mini-batch. In the loss below, $\mathbf{e}_p$ refers to the embedding of a positive sample with index $p \in \mathcal{P}(i)$, while $\mathbf{e}_q$ refers to the embedding of any comparison sample with index $q \in \mathcal{B} \setminus \{i\}$ (including both positives and negatives).

The InfoNCE-style loss is

$$\mathcal{L}_c = -\frac{1}{|\mathcal{B}|} \sum_{i \in \mathcal{B}} \log \frac{\sum_{p \in \mathcal{P}(i)} \exp\left(\langle \mathbf{e}_i, \mathbf{e}_p \rangle / \kappa\right)}{\sum_{q \in \mathcal{B} \setminus \{i\}} \exp\left(\langle \mathbf{e}_i, \mathbf{e}_q \rangle / \kappa\right)}, \tag{13}$$

where $\langle \cdot, \cdot \rangle$ denotes cosine similarity and κ is a similarity scale parameter (analogous to temperature in contrastive learning). In our experiments, we set $\kappa = 0.1$ to control the sharpness of the similarity distribution.

A.4. Supervised predictive loss ($\mathcal{L}_s$)

To ensure the embeddings retain direct predictive information for inflow forecasting, we include an auxiliary supervised regression term aligned with the main task. For each window, the encoder produces hidden states $\{\mathbf{h}_t\}_{t=1}^{T}$. We compute a pooled representation

$$\bar{\mathbf{h}} = \frac{1}{T} \sum_{t=1}^{T} \mathbf{h}_t, \tag{14}$$

and map it to the multi-step inflow target using a lightweight linear predictor $g_\psi(\cdot)$:

$$\hat{\mathbf{y}} = g_\psi(\bar{\mathbf{h}}), \tag{15}$$

where $\mathbf{y} \in \mathbb{R}^K$ is the observed multi-step inflow target over the same forecast horizon as the main task (here $K = 7$ days). The supervised loss is defined as the mean squared error:

$$\mathcal{L}_s = \frac{1}{K} \|\hat{\mathbf{y}} - \mathbf{y}\|_2^2 . \tag{16}$$

A.5. Total pretraining objective

The total pretraining objective combines the supervised term (Eq. (16)) and the contrastive term (Eq. (13)) via a weighted sum:

$$\mathcal{L}_{\text{pre}} = \alpha \mathcal{L}_s + (1 - \alpha)\mathcal{L}_c, \tag{17}$$

where α controls the balance between predictive learning and representation regularization. In our experiments, we set $\alpha = 0.8$, placing greater emphasis on the supervised regression objective while using the contrastive loss as a regularizer to encourage reservoir-consistent representations.

A.6. Sensitivity analysis on loss weight α

To assess robustness to the weighting choice, we conducted a sensitivity analysis by varying $\alpha \in \{0.2, 0.4, 0.6, 0.8, 1.0\}$. For each value of α, we repeated pretraining at least 10 times using different random seeds and report the mean ± standard deviation of the final training loss and validation loss. Table 3 summarizes the results. The validation loss is lowest at $\alpha = 0.8$, while performance is generally stable for $\alpha \in [0.6, 0.8]$. These results support using $\alpha = 0.8$ in this study: the supervised term remains dominant, and the contrastive term provides complementary regularization that improves generalization.

Table 3
Sensitivity analysis of the pretraining loss weight α (10+ runs, mean ± standard deviation).

α	Train loss	Val loss
0.2	0.005512 ± 0.000087	0.006426 ± 0.000094
0.4	0.004718 ± 0.000079	0.006282 ± 0.000083
0.6	0.003556 ± 0.000068	0.005187 ± 0.000076
0.8	0.003147 ± 0.000061	0.004840 ± 0.000069
1.0	0.002901 ± 0.000065	0.005512 ± 0.000074

Table 4
High-flow NSE comparison under the top 5% inflow regime between AGFormer with adaptive pruning and the fixed-graph ablation.

	Full model	w/o pruning
Day 1	0.9702	0.7800
Day 2	0.9560	0.6027
Day 3	0.9347	0.4065
Day 4	0.9156	0.2592
Day 5	0.8916	0.1641
Day 6	0.8692	0.1139
Day 7	0.8418	0.0557

A.7. Ablation on pretraining results

Fig. 12 compares AGFormer with and without pretraining on five data-scarce reservoirs (CAU, HYR, JOR, LCR, and MCR), each with only 13–15 years of historical observations. Pretraining yields consistent improvements in NSE scores across all forecast horizons, with the largest gains at longer lead times. For example, at Day 7, the NSE for HYR increases from 0.7378 to 0.7648. These improvements result from the encoder's ability to exploit the full heterogeneous dataset, rather than being constrained by the overlapping period across reservoirs. Overall, the results demonstrate that pretraining effectively enhances forecast skill for reservoirs with sparse records.

A.8. Extreme-regime evaluation and structural diagnostics

Extreme-regime evaluation. We perform an additional evaluation targeting extreme high-inflow conditions, defined as the top 5% inflow values for each reservoir. The objective is to assess whether adaptive monotonic pruning preserves connections that remain predictive during rare but operationally critical events.

Because extreme samples are limited, models are trained using the full 10-year dataset without regime filtering to maintain training stability and consistency with the main experiment. Extreme-regime analysis is conducted only at evaluation time.

We compare High-flow NSE between AGFormer with adaptive pruning and its fixed-graph ablation. As reported in Table 4, AGFormer maintains strong skill under the top 5% regime (NSE > 0.84 at Day 7), while the fixed-graph variant degrades rapidly with lead time (NSE < 0.1 by Day 7). The widening gap with forecast horizon indicates that adaptive pruning contributes to stabilizing multi-step forecasts during high-flow events.

Structural diagnostics. To characterize how pruning modifies graph structure, we examine node degree distributions and edge-retention rates across pruning stages ($\tau = 0.1, 0.2, 0.3$). Node degree describes local connectivity per reservoir, while the retention rate quantifies the fraction of edges preserved relative to the initial graph, indicating global sparsity. These metrics provide complementary views of structural evolution. Both metrics are computed for each reservoir and each evaluation sample. The reported distributions summarize these values across all reservoirs and all extreme-flow samples at a given pruning stage.

As shown in Figs. 13 and 14, pruning progresses from conservative filtering to stronger sparsification. Early stages retain most edges and

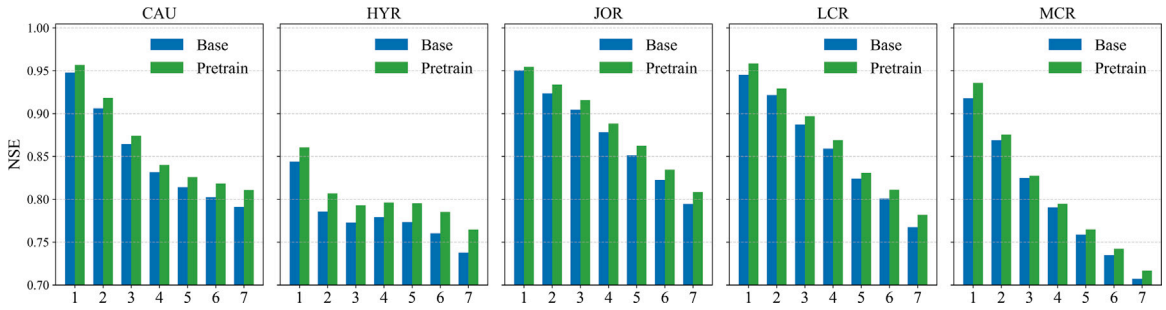


Fig. 12. Comparison of AGFormer inflow forecasting performance with and without pretraining on five data-scarce reservoirs (CAU, HYR, JOR, LCR, and MCR).

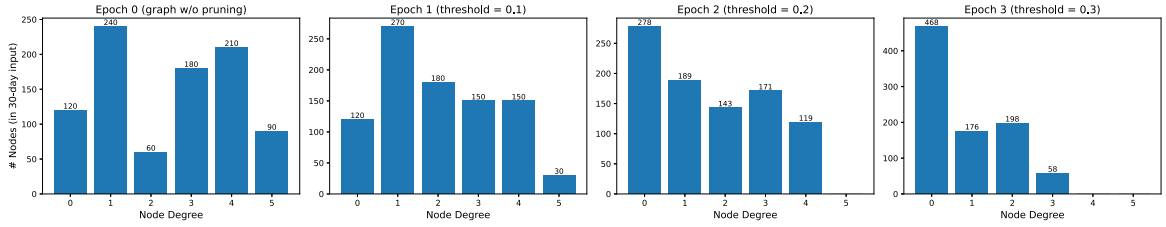


Fig. 13. Node degree distribution under extreme high-flow conditions across three pruning stages ($\tau = 0.1, 0.2, 0.3$), showing progressive sparsification and concentration of connectivity.

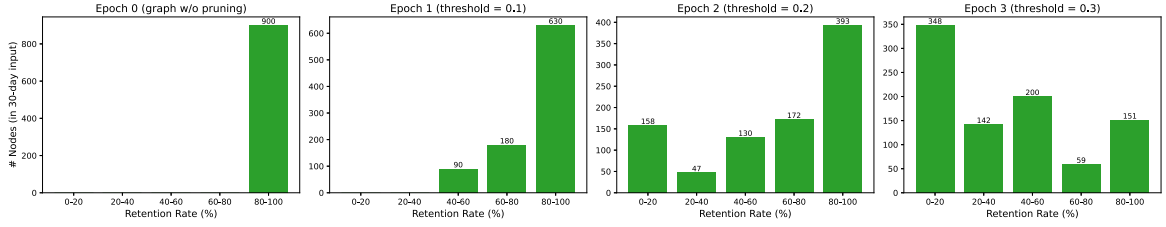


Fig. 14. Edge-retention rate distributions across pruning stages under extreme high-flow conditions. Lower retention bins at later stages indicate systematic removal of low-attention edges.

moderate node degrees, whereas later stages shift toward lower retention and reduced degrees. Because pruning is monotonic and attention-guided, edge removal is systematic rather than random, concentrating connectivity on persistently informative links.

Overall, the joint evolution of degree and retention statistics indicates progressive yet structured sparsification. Even under extreme inflow conditions, the resulting topology remains coherent. This supports the interpretation that monotonic pruning functions as a structural selection mechanism that filters weak dependencies while preserving stable pathways relevant for forecasting.

Data availability

We have included the software availability section in the manuscript.

References

Akkala, A., Boubrahimi, S.F., Hamdi, S.M., Hosseinzadeh, P., Nassar, A., 2025. Spatio-temporal graph neural networks for streamflow prediction in the upper colorado basin. Hydrology 12 (3), 60.

Allawi, M.F., Jaafar, O., Mohamad Hamzah, F., Abdullah, S.M.S., El-Shafie, A., 2018. Review on applications of artificial intelligence methods for dam and reservoir-hydro-environment models. Environ. Sci. Pollut. Res. 25, 13446–13469.

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.

Bennett, J.C., Wang, Q., Li, M., Robertson, D.E., Schepen, A., 2016. Reliable long-range ensemble streamflow forecasts: Combining calibrated climate forecasts with a conceptual runoff model and a staged error model. Water Resour. Res. 52 (10), 8238–8259.

Bernardes, Jr., J., Santos, M., Abreu, T., Prado, Jr., L., Miranda, D., Julio, R., Viana, P., Fonseca, M., Bortoni, E., Bastos, G.S., 2022. Hydropower operation optimization using machine learning: A systematic review. AI 3 (1), 78–99.

Chen, T., Sui, Y., Chen, X., Zhang, A., Wang, Z., 2021. A unified lottery ticket hypothesis for graph neural networks. Adv. Neural Inf. Process. Syst. 34, 16737–16749.

Daly, C., Bryant, K., 2013. The PRISM Climate and Weather System—An Introduction, vol. 2, PRISM Climate Group, Corvallis, OR.

Fan, M., Liu, S., Lu, D., 2023a. Advancing subseasonal reservoir inflow forecasts using an explainable machine learning method. J. Hydrol.: Reg. Stud. 50, 101584.

Fan, M., Liu, S., Lu, D., Gangrade, S., Kao, S.-C., 2023b. Explainable machine learning model for multi-step forecasting of reservoir inflow with uncertainty quantification. Environ. Model. Softw. 170, 105849.

Fan, M., Lu, D., Gangrade, S., 2025. Enhancing multi-step reservoir inflow forecasting: A time-variant encoder–decoder approach. Geosciences 15 (8), 279.

Fan, M., Zhang, L., Liu, S., Yang, T., Lu, D., 2022. Identifying hydrometeorological factors influencing reservoir releases using machine learning methods. In: 2022 IEEE International Conference on Data Mining Workshops. ICDMW, IEEE, pp. 1102–1110.

Fan, M., Zhang, L., Liu, S., Yang, T., Lu, D., 2023c. Investigation of hydrometeorological influences on reservoir releases using explainable machine learning methods. Front. Water 5, 1112970.

Jiang, J., Chen, C., Wang, L., Hou, H., Deng, C., Ju, Y., Zhu, X., 2024. Heterogeneous spatio-temporal series forecasting using dynamic graph neural networks for flood prediction. In: ICC 2024-IEEE International Conference on Communications. IEEE, pp. 1963–1968.

Kao, I.-F., Zhou, Y., Chang, L.-C., Chang, F.-J., 2020. Exploring a long short-term memory based encoder-decoder framework for multi-step-ahead flood forecasting. J. Hydrol. 583, 124631.

Khorram, S., Jehbez, N., 2023. A hybrid CNN-LSTM approach for monthly reservoir inflow forecasting. Water Resour. Manag. 37 (10), 4097–4121.

Kratzert, F., Klotz, D., Brenner, C., Schulz, K., Herrnegger, M., 2018. Rainfall–runoff modelling using long short-term memory (LSTM) networks. Hydrol. Earth Syst. Sci. 22 (11), 6005–6022.

Kratzert, F., Klotz, D., Shalev, G., Klambauer, G., Hochreiter, S., Nearing, G., 2019. Towards learning universal, regional, and local hydrological behaviors via machine learning applied to large-sample datasets. Hydrol. Earth Syst. Sci. 23 (12), 5089–5110.

Latif, S.D., Ahmed, A.N., 2023. A review of deep learning and machine learning techniques for hydrological inflow forecasting. Environ. Dev. Sustain. 25 (11), 12189–12216.

Li, J., Dao, V., Hsu, K., Analui, B., Knofczynski, J.D., Sorooshian, S., 2024. Improving cascade reservoir inflow forecasting and extracting insights by decomposing the physical process using a hybrid model. J. Hydrol. 630, 130623.

Li, F., Ma, G., Ju, C., Chen, S., Huang, W., 2024. Data-driven forecasting framework for daily reservoir inflow time series considering the flood peaks based on multi-head attention mechanism. J. Hydrol. 645, 132197.

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.

Liu, G., Ouyang, S., Qin, H., Liu, S., Shen, Q., Qu, Y., Zheng, Z., Sun, H., Zhou, J., 2023. Assessing spatial connectivity effects on daily streamflow forecasting using Bayesian-based graph neural network. Sci. Total Environ. 855, 158968.

Longa, A., Lachi, V., Santin, G., Bianchini, M., Lepri, B., Lio, P., Scarselli, F., Passerini, A., 2023. Graph neural networks for temporal graphs: State of the art, open challenges, and opportunities. arXiv preprint arXiv:2302.01018.

Moriasi, D.N., Arnold, J.G., Van Liew, M.W., Binger, R.L., Harmel, R.D., Veith, T.L., 2007. Model evaluation guidelines for systematic quantification of accuracy in watershed simulations. Trans. ASABE 50 (3), 885–900.

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.

Muñoz-Sabater, J., Dutra, E., Agustí-Panareda, A., Albergel, C., Arduini, G., Balsamo, G., Boussetta, S., Choulga, M., Harrigan, S., Hersbach, H., et al., 2021. ERA5-land: A state-of-the-art global reanalysis dataset for land applications. Earth Syst. Sci. Data 13 (9), 4349–4383.

Nourani, V., Paknezhad, N.J., Tanaka, H., 2021. Prediction interval estimation methods for artificial neural network (ANN)-based modeling of the hydro-climatic processes, a review. Sustainability 13 (4), 1633.

Oord, A.v.d., Li, Y., Vinyals, O., 2018. Representation learning with contrastive predictive coding. arXiv preprint arXiv:1807.03748.

Qi, W.-y., Chen, J., Li, L., Xu, C.-Y., Li, J., Xiang, Y., Zhang, S., 2022. Regionalization of catchment hydrological model parameters for global water resources simulations. Hydrol. Res. 53 (3), 441–466.

Sun, A.Y., Jiang, P., Mudunuru, M.K., Chen, X., 2021. Explore spatio-temporal learning of large sample hydrology using graph neural networks. Water Resour. Res. 57 (12), e2021WR030394.

Sun, A.Y., Jiang, P., Yang, Z.-L., Xie, Y., Chen, X., 2022. A graph neural network approach to basin-scale river network learning: The role of physics-based connectivity and data fusion. Hydrol. Earth Syst. Sci. Discuss. 2022, 1–35.

Sundararajan, M., Taly, A., Yan, Q., 2017. Axiomatic attribution for deep networks. In: International Conference on Machine Learning. PMLR, pp. 3319–3328.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I., 2017. Attention is all you need. Adv. Neural Inf. Process. Syst. 30.

Weiler, M., McGlynn, B.L., McGuire, K.J., McDonnell, J.J., 2003. How does rainfall become runoff? A combined tracer and runoff transfer function approach. Water Resour. Res. 39 (11).

Wu, Z., Pan, S., Chen, F., Long, G., Zhang, C., Yu, P.S., 2020. A comprehensive survey on graph neural networks. IEEE Trans. Neural Networks Learn. Syst. 32 (1), 4–24.

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.

Yang, S., Yang, D., Chen, J., Zhao, B., 2019. Real-time reservoir operation using recurrent neural networks and inflow forecast from a distributed hydrological model. J. Hydrol. 579, 124229.

You, H., Li, C., Xu, P., Fu, Y., Wang, Y., Chen, X., Baraniuk, R.G., Wang, Z., Lin, Y., 2020. Drawing early-bird tickets: Towards more efficient training of deep networks. In: International Conference on Learning Representations.

Yousefi, M., Cheng, X., Gazzea, M., Wierling, A.H., Rajasekharan, J., Helseth, A., Farahmand, H., Arghandeh, R., 2022. Day-ahead inflow forecasting using causal empirical decomposition. J. Hydrol. 613, 128265.

Yousefi, M., Wang, J., Fandrem Høivik, Ø., Rajasekharan, J., Hubert Wierling, A., Farahmand, H., Arghandeh, R., 2023. Short-term inflow forecasting in a dam-regulated river in southwest Norway using causal variational mode decomposition. Sci. Rep. 13 (1), 7016.

Yu, B., Yin, H., Zhu, Z., 2017. Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting. arXiv preprint arXiv:1709.04875.

Zhao, X., Wang, H., Bai, M., Xu, Y., Dong, S., Rao, H., Ming, W., 2024. A comprehensive review of methods for hydrological forecasting based on deep learning. Water 16 (10), 1407.