Supplement of Hydrol. Earth Syst. Sci., 30, 2079–2092, 2026
https://doi.org/10.5194/hess-30-2079-2026-supplement

Hydrology and
Earth System
Sciences
Open Access

Supplement of

A GNN routing module is all you need for LSTM Rainfall–Runoff models

Hamidreza Mosaffa et al.

Correspondence to: Hamidreza Mosaffa (h.mosaffa@reading.ac.uk)

GNN Architectures for River Routing

To explicitly model basin-scale runoff routing, we evaluate four distinct Graph Neural Network (GNN) architectures. Each architecture defines a different mechanism for aggregating hydrological information from upstream subbasins and propagating it downstream along the river network. Let $h_i^{(l)} \in \mathbb{R}^d$denote the latent feature embedding of subbasin $i$at GNN layer l, $\mathcal{N}(i)$the set of upstream neighbors of node i, and $\sigma(\cdot)$a nonlinear activation function (e.g., ReLU).All four architectures share the same LSTM-generated initial node embeddings $h_i^{(0)}$(Section 3.1.1) and differ only in how they propagate information through the river network graph during the routing phase**.**

S1. Graph Convolutional Network (GCN)

The Graph Convolutional Network (GCN; Kipf and Welling, 2016) approximates a spectral convolution on the river network graph. Information is propagated by averaging feature representations from upstream neighbors, weighted by the graph's normalized connectivity structure. The update rule is given by:

$$h_i^{(l+1)} = \sigma\left(\sum_{j \in \mathcal{N}(i) \cup \{i\}} \frac{1}{\sqrt{\tilde{d}_i \tilde{d}_j}} W^{(l)} h_j^{(l)} \right),$$

where $W^{(l)}$is a learnable weight matrix and $\tilde{d}_i$denotes the degree of node i, including self-loops. From a hydrological perspective, GCN assumes that the influence of upstream subbasins is determined primarily by the network topology, treating all tributaries as having a structurally fixed contribution weight normalized by node degree. This symmetric normalization ensures that nodes with many upstream connections (e.g., downstream confluences) do not receive disproportionately large signals, simulating a simplified routing scenario where flow from each tributary contributes proportionally based on network structure rather than hydrological state.

S2. Graph Attention Network (GAT)

The Graph Attention Network (GAT; Veličković et al., 2017) extends GCN by introducing a learnable attention mechanism that assigns adaptive weights to upstream neighbors. This is particularly relevant for river routing, where tributaries may contribute unequally depending on their current hydrological conditions. The node update is defined as:

$$h_i^{(l+1)} = \sigma\left(\sum_{j \in \mathcal{N}(i) \cup \{i\}} \alpha_{ij}^{(l)} W^{(l)} h_j^{(l)} \right),$$

where the attention coefficient α_{ij}represents the learned importance of upstream subbasin $j$to subbasin i. These coefficients are computed as:

$$\alpha_{ij} = \frac{\exp\left(\mathrm{LeakyReLU}\left(a^\top [W h_i \parallel W h_j]\right)\right)}{\sum_{k \in \mathcal{N}(i) \cup \{i\}} \exp\left(\mathrm{LeakyReLU}(a^\top [W h_i \parallel W h_k])\right)},$$

where $\parallel$denotes vector concatenation and a is a learnable attention vector. Hydrologically, GAT allows the model to dynamically emphasize dominant upstream signals (e.g., a major tributary experiencing a flood event) while down-weighting less influential contributions, providing a flexible representation of flow routing.

S3. GraphSAGE

GraphSAGE (Hamilton et al., 2017) is an inductive GNN framework that explicitly separates neighbor aggregation from node update. It summarizes upstream information using a chosen aggregation function and then combines this summary with the local node state. The update rule is:

$$h_i^{(l+1)} = \sigma\left(W^{(l)} \cdot \mathrm{CONCAT}\left(h_i^{(l)}, \mathrm{AGG}\left(\{h_j^{(l)} \mid j \in \mathcal{N}(i)\}\right)\right)\right),$$

where AGG denotes an aggregation operator (e.g., mean pooling). In the context of river routing, GraphSAGE first aggregates upstream runoff information to represent total incoming flow conditions and then fuses this information with the local subbasin state, enabling flexible and scalable routing representations.

S4. Chebyshev Spectral GCN (ChebNet)

The Chebyshev Spectral Graph Convolutional Network (ChebNet; Defferrard et al., 2016) extends spectral graph convolutions by approximating filters using Chebyshev polynomials of the graph Laplacian, avoiding costly eigen decomposition. The update rule is given by:

$$h^{(l+1)} = \sigma\left(\sum_{k=0}^{K-1} \theta_k^{(l)}\, T_k(\tilde{L})\, h^{(l)}\right),$$

where $T_k(\cdot)$denotes the Chebyshev polynomial of order k, $\tilde{L}$is the scaled normalized Laplacian of the river network, and $\theta_k^{(l)}$are learnable coefficients. By controlling the polynomial order K, ChebNet explicitly defines the spatial receptive field of routing, allowing information to propagate across multiple upstream subbasins within a single layer. This makes ChebNet particularly suited for capturing long-range upstream dependencies in large river basins.