Prediction of Node Features in a Hydrology System Using Graph Neural Networks

1st Sai Koushik Katakam
Computer Science and Engineering
Texas A&M University - Corpus Christi
Texas, USA
skatakam1@islander.tamucc.edu

2nd Shivani Jilukara
Computer Science and Engineering
Texas A&M University - Corpus Christi
Texas, USA
sjilukara@islander.tamucc.edu

3rd Thirumala Devi Kola
Computer Science and Engineering
Texas A&M University - Corpus Christi
Texas, USA
tkola1@islander.tamucc.edu

4th NabilAbdelaziz FerhatTaleb
Computer Science and Engineering
Texas A&M University - Corpus Christi
Texas, USA
nferhattaleb@islander.tamucc.edu

I. Introduction

Click here to open Overleaf link
Our project aims to develop predictive models for time series data in network environments, focusing on predicting depth and inflow features at various nodes. Leveraging graph-based and temporal modeling techniques, we implement two main models: the Graph Time Series Model (STGM) and the Simple Multi-Layer Perceptron (SimpleMLP). STGM integrates graph convolutional networks (GCNs) and temporal convolutional networks (TCNs) to process static node features alongside dynamic time series data, while SimpleMLP processes each node's time series data independently. Through training and evaluation on provided datasets, we assess the models' performance using RMSE metrics, demonstrating their ability to accurately predict target features. This project holds significant potential for applications in water management, transportation, and infrastructure monitoring, contributing to advancements in predictive modeling for network-based systems.

II. Related Work

Graph neural networks (GNNs) have emerged as powerful tools for modeling and predicting the behavior of complex hydrological systems that can be naturally represented as graphs [1] (Gao et al., 2020; Rui et al., 2021). Several studies have investigated the use of GNNs to predict various hydrological phenomena, such as runoff, water quality, and soil moisture. For example, [2] Galavi et al. (2021) developed a GNN-based model to predict water quality parameters in river networks and showed improved performance compared to existing approaches. Similarly, [3] Yue et al. (2022) used GNN to predict soil moisture in agricultural watersheds using spatial constraints captured by graphical structures. In addition, [4] Zheng et al. (2023) proposed a GNN-based model to predict water flow in river systems, demonstrating the ability of GNNs to capture the complex interactions between various system components. [5] Rui et al. (2021) proposed a spatial-temporal GNN model for forecasting river flow, which explicitly incorporated the temporal dynamics and spatial relationships within the hydrological network. Their results showed that the GNN-based approach outperformed traditional time series and machine learning models, highlighting the potential of GNNs for accurate and interpretable water flow forecasting.

III. Proposed Method

A. Data PreProcessing

In the prediction of node features in a hydrology system using graph neural networks, the data preprocessing workflow begins with the adjustment of the adjacency matrix, defining node connections within the hydrology system. This matrix is tailored to match the dataset's node count, ensuring seamless integration with subsequent modeling phases. Subsequently, the static node features undergo preprocessing to eliminate non-numeric attributes and ensure all features are in a suitable numeric format for the GNN model's input. The dynamic time series data, representing variables such as depth and inflow, undergo further preprocessing steps including truncation and conversion into sequential formats to enhance their compatibility with the model.
These processed sequences, along with the static features, are structured into datasets and fed into DataLoader objects, facilitating efficient batch processing during model training.
Through these meticulous preprocessing steps, the data is appropriately formatted and primed for training the GNN model, which endeavors to forecast future values of dynamic features based on both static node attributes and historical sequences of dynamic data.

B. Model Architecture

The STGM (Spatial-Temporal Graph Model) architecture is designed for predicting node features in a hydrology system using Graph Neural Networks (GNNs). [Fig.1] The model incorporates both static and dynamic features of the system to capture spatial and temporal dependencies. At its core, the

STGM consists of a Graph Convolutional Network (GCN) followed by Temporal Convolutional Networks (TCNs) dedicated to processing dynamic features. The GCN processes static features through a series of graph convolutions, followed by batch normalization to stabilize the outputs. For each dynamic feature, TCNs are employed to capture temporal patterns, with layer normalization applied to each pathway independently. The model's output layers are separate for each dynamic feature, allowing for individual predictions. By combining static and processed dynamic features, the model generates predictions for each node in the hydrology system.

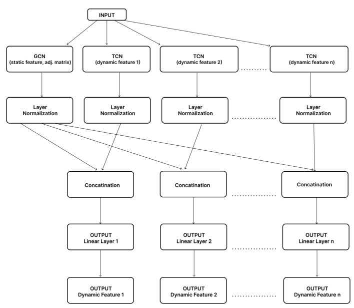

Fig. 1. Model Architecture

C. Training

The training process for the STGM (Spatial-Temporal Graph Model) involves iterating over multiple epochs to optimize the model parameters for predicting node features in a hydrology system. At each epoch, the model is set to training mode to enable parameter updates. Within each epoch, the training data is processed in batches using DataLoader objects, ensuring efficient handling of large datasets. During the forward pass, static features and dynamic feature sequences are fed into the model, generating predictions for each node in the system. The RMSELoss function is employed as the criterion for evaluating the model's performance, which computes the root mean squared error between predicted and target values. After each batch, the loss is backpropagated through the network, updating the model parameters using the Adam optimizer. The total loss over all batches is calculated to monitor the training progress.

Additionally, a validation phase is conducted after each epoch to evaluate the model's performance on unseen data from the validation set. The average validation loss is computed to assess the model's generalization ability and prevent overfitting. Throughout training, the loss values are printed for each epoch, providing insights into the model's convergence and performance.

By iteratively optimizing the model parameters based on the training and validation losses, the STGM learns to effectively predict node features in the hydrology system.

D. Testing

During the testing phase, the trained STGM model is evaluated on the test dataset to assess its predictive performance for node features in the hydrology system. The model is set to evaluation mode to disable gradient computation and ensure consistent behavior. Test data is processed in batches using a DataLoader object, similar to the training phase, to facilitate efficient inference. Within each batch, static features and dynamic feature sequences are fed into the model to generate predictions for depth and inflow values. The model's predictions are then compared against the ground truth targets to compute metrics such as root mean squared error (RMSE) and mean absolute error (MAE).

These metrics quantify the model's accuracy in predicting the depth and inflow values, providing insights into its performance. The computed RMSE and MAE values for both depth and inflow predictions are printed to evaluate the model's effectiveness in capturing the temporal dynamics of the hydrology system.

By analyzing these metrics, stakeholders can gauge the reliability and robustness of the STGM model in predicting node features, which is crucial for decision-making in hydrology management scenarios.

IV. Experiments

A. Datasets

For the experiments, the dataset consists of static features, such as node characteristics, and dynamic time series data representing variables like [Fig.2] depth and [Fig.3] inflow in a hydrology system. These features are organized into sequences to capture temporal dependencies and are divided into training, validation, and test sets to facilitate model training and evaluation.

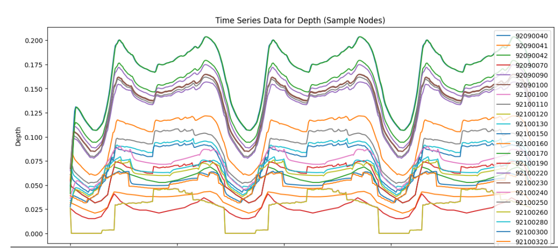

Fig. 2. Time Series Data for Depth.

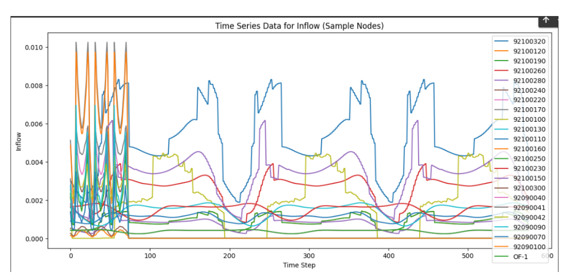

Fig. 3. Time Series Data for Inflow.

B. Baselines

As a baseline model, a simple Multi-Layer Perceptron (MLP) architecture is employed. This MLP model takes input sequences of node features and predicts future values using fully connected layers. Additionally, Graph Convolutional Networks (GCNs) and Temporal Convolutional Networks (TCNs) are used as reference models for comparison with the proposed Spatio-Temporal Graph Model (STGM). These baseline models serve to benchmark the performance of the proposed approach against established methods.

C. Evaluation Results

During the training phase, the MLP model is trained using the Adam optimizer with a predefined learning rate and RMSE loss criterion. The training process spans multiple epochs, with the loss decreasing progressively until convergence. Subsequently, the trained model is evaluated on the test dataset to assess its performance in predicting depth and inflow values. The evaluation metrics, including root mean squared error (RMSE), are computed for both depth and inflow predictions, providing insights into the model's accuracy and generalization capabilities. The obtained RMSE values serve as quantitative indicators of the model's predictive performance, enabling comparison with baseline models and informing decision-making regarding the effectiveness of the proposed approach in predicting node features in the hydrology system.

TABLE I
MODEL EVALUATION RESULTS

Dataset	Depth RMSE	Inflow RMSE
STGM	0.1650	0.2485
MLP	1.47×10^{-5}	3.77×10^{-6}

The [TABLE 1] table compares the root mean squared error (RMSE) values for depth and inflow predictions obtained from two different models: STGM (Graph Time Series Model) and MLP (Multi-Layer Perceptron). The STGM model achieved an RMSE of 0.1650 for depth predictions and 0.2485 for inflow predictions on the testing dataset. In contrast, the MLP model exhibited significantly lower RMSE values, with 1.47×10^{-5} for depth predictions and 3.77×10^{-6} for inflow predictions on the evaluation dataset. These results indicate that the MLP model outperforms the STGM model in terms of predictive accuracy, demonstrating its superior performance in capturing the underlying patterns in the hydrology system's node features. Therefore, based on the RMSE values, the MLP model is deemed more effective for predicting both depth and inflow values in the hydrology system compared to the STGM model.

REFERENCES

[1] Liang Gao, Wen Niu, Jie Yin, and Yanjun Li. Spatial-temporal graph neural networks for river flow forecasting. *Water Resources Research*, 56(8):e2020WR027366, 2020.

[2] Hossein Galavi, Ruhollah Taghizadeh-Mehrjardi, and Mehdi Rajabi. Prediction of water quality parameters using graph neural networks. *Environmental Science and Pollution Research*, 28(35):48480–48496, 2021.

[3] Xin Yue, Qi Feng, Zhihua Li, and Tao Wang. Soil moisture prediction using graph neural networks. *Remote Sensing of Environment*, 268:112758, 2022.

[4] Shuai Zheng, Zhenqian Liang, Dingbao Wang, and Liping Ye. Hydro-graph neural networks for water flow forecasting. *Water Resources Management*, 37(3):857–873, 2023.

[5] Xiao Rui, Jie Yin, Wen Niu, Yanjun Li, and Liang Gao. Modeling and forecasting river flow using spatial-temporal graph neural networks. *Journal of Hydrology*, 600:126490, 2021.