Predicting Himalayan Glacier Melting with Spatio-Temporal Graph Neural Networks Using Satellite Imagery

Presented by:

Hari Samhita

23BAI1473

Date: 15 June 2025

Project Summary

Deep Learning Pipeline

Developed a novel deep learning pipeline specifically designed for predicting glacier melt progression over time. This pipeline utilizes satellite imagery, treating each GIF as a rich spatio-temporal data source.

Visual and Spatio-Temporal Reasoning

Our predictions are based on a dual approach: analyzing visual pixel-level changes within image sequences and incorporating sophisticated spatio-temporal reasoning. This ensures a comprehensive understanding of melt dynamics.

Hybrid Model Architecture

The core of our model combines a ResNet50 Convolutional Neural Network (CNN) for effective spatial encoding of image features, and a GraphSAGE Graph Neural Network (GNN) for robust temporal modeling of the sequential data.

Dataset & Input Format

Data Specifics

- **Data Type**: Satellite time-lapse GIFs of **THE GANGOTRI GLACIER**, capturing dynamic changes.
- **Resolution**: All frames uniformly resized to 128x128 pixels for consistent input.
- **Number of Samples**: Over 169 GIFs manually collected from NASA Worldview Sentinel Hub, providing a rich dataset.
- **Format**: Each GIF is meticulously treated as a temporal sequence of individual image frames, preserving chronological information.

Our dataset comprises a comprehensive collection of satellite time-lapse GIFs focusing on the Gangotri glacier of the Himalayas. Each GIF is pre-processed to a standardized 128x128 pixel resolution, ensuring uniformity across the temporal sequences. This meticulous preparation allows our model to effectively learn from the subtle pixel-level changes indicative of glacier melt.

Step 1 – Frame Extraction

GIF Input

The process begins with the raw satellite time-lapse GIFs, serving as the initial input for our data pipeline.

Frame Extraction

Each GIF is systematically decomposed into its constituent individual frames using the **imageio** library, enabling frame-by-frame analysis.

Preprocessing & Storage

Each extracted frame is converted to RGB format and resized to 128x128 pixels, then stored in an ordered sequence (e.g., **frame_000.jpg, frame_001.jpg**).

Temporal Nodes Output

The output of this stage is N distinct frames, each subsequently treated as a temporal node in the spatio-temporal graph structure.

Step 2 – Feature Extraction with ResNet50

Pretrained ResNet50 Backbone

Leveraged a pretrained ResNet50 Convolutional Neural Network as the foundational backbone for robust spatial feature extraction from each glacier frame.

- Removed final classification layer to focus on rich feature representation.
- Resulting output: a 2048-dimensional embedding for each frame, capturing its spatial characteristics.

Preprocessing & Augmentations

Light augmentations, including horizontal flips and slight rotations, were applied during the preprocessing phase to enhance model robustness and generalization.

- This helps the model learn invariant features, reducing sensitivity to minor variations in input.

TorchVision Weights Used: ResNet50_Weights.IMAGENET1K_V1

Step 3 – Melt via Pixel Difference

Pixel-Wise Difference

Melt progression is quantified by computing the mean pixel-wise difference between consecutive frames (frame_t and frame_{t-1}). This method effectively captures visual changes over time.

Normalization

The raw pixel differences are then normalized to a range of [0, 1]. This normalization ensures consistency and facilitates its use as the target melt signal for model training.

Target Melt Signal

The normalized mean pixel difference serves as our empirical proxy for glacier melt at each time step, allowing for supervised learning of melt dynamics.

$$melt_t = mean(abs(frame_t - frame_{t-1}))$$

Step 4 – Spatio-Temporal Graph Creation

1

Node Representation

Each individual frame extracted from the satellite GIF is conceptualized and implemented as a distinct node within the spatio-temporal graph.

2

Bidirectional Temporal Edges

Temporal relationships between consecutive frames are captured by establishing bidirectional edges. For T frames, this results in 2*(T-1) total edges, linking each frame to its immediate predecessor and successor.

3

Exclusion of Spatial Edges

For this specific pipeline, spatial edges (connections between different regions within the same frame or across frames at the same time step) were not required, simplifying the graph structure.

4

Torch_Geometric Data Structure

The graph is efficiently constructed and managed using **torch_geometric.data.Data**, which is optimized for handling graph structures within the PyTorch ecosystem, specifying edge indices, node features (x), and target labels (y).

Model Architecture – STGNN

GraphSAGE GNN Implementation

A 3-layer GraphSAGE Graph Neural Network forms the core of our spatio-temporal model, designed to aggregate information across the temporal graph structure.

Layer Progression

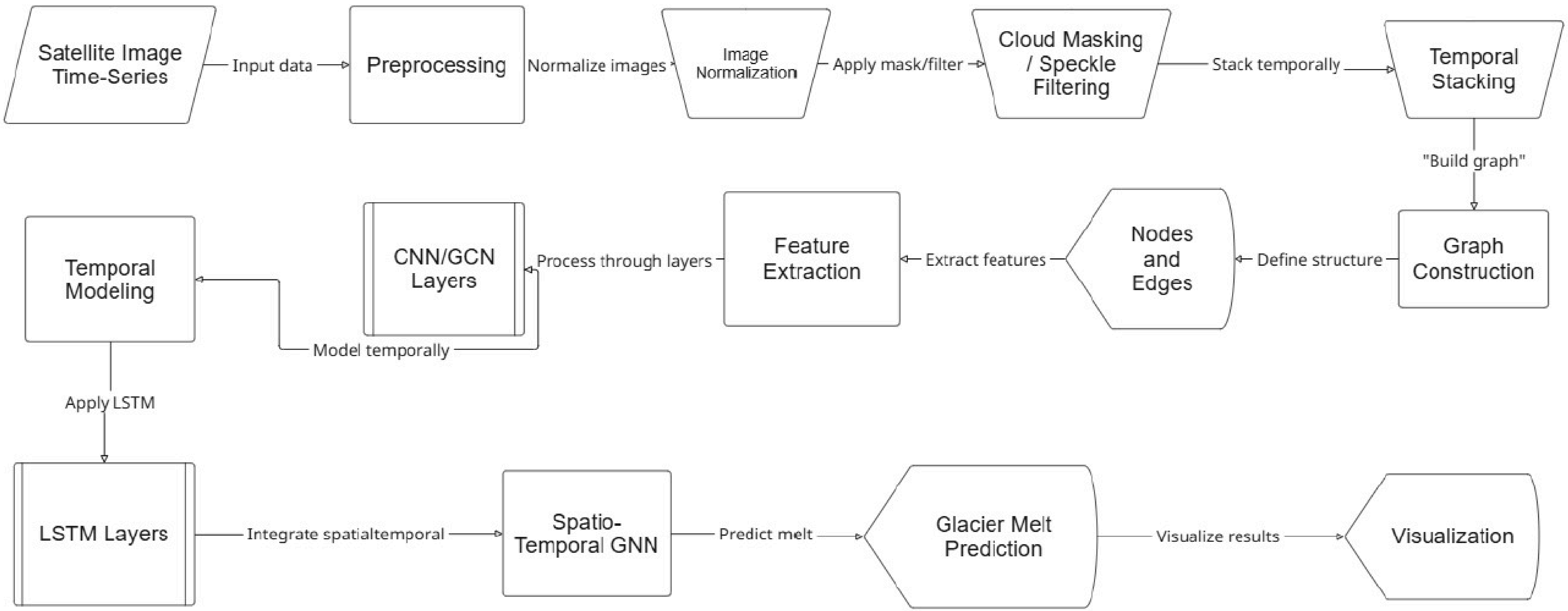
The network consists of three sequential **SAGEConv** layers, progressively reducing dimensionality: **2048 → 256 → 128 → 1**. Each layer is followed by a ReLU activation function to introduce non-linearity.

Input and Output

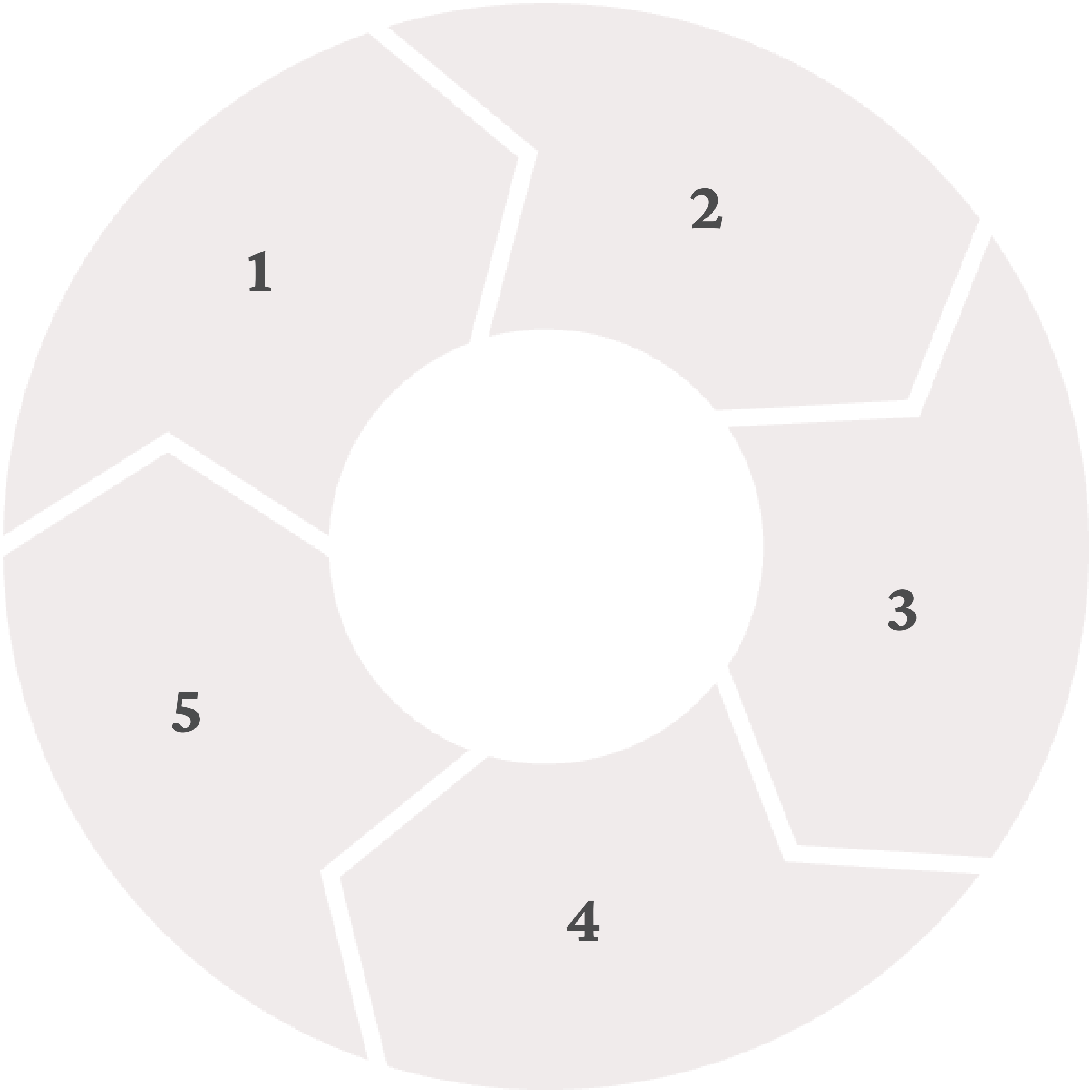
The input to the GNN are the ResNet-extracted spatial features (X) for each frame. The model's output provides melt predictions for every frame in the sequence, aligned with the temporal nodes.

Frameworks Used: PyTorch, torchvision, torch_geometric

Architecture Diagram



Training Setup



Optimizer: Adam

The Adam optimizer was chosen for its efficiency and robust performance in deep learning tasks, facilitating effective weight updates during training.

Learning Rate: 0.003

An initial learning rate of 0.003 was set, balancing between rapid convergence and stability during the optimization process.

Loss Function: MSELoss

Mean Squared Error Loss (MSELoss) was employed as the objective function, quantifying the difference between predicted and actual melt values.

Epochs: 600

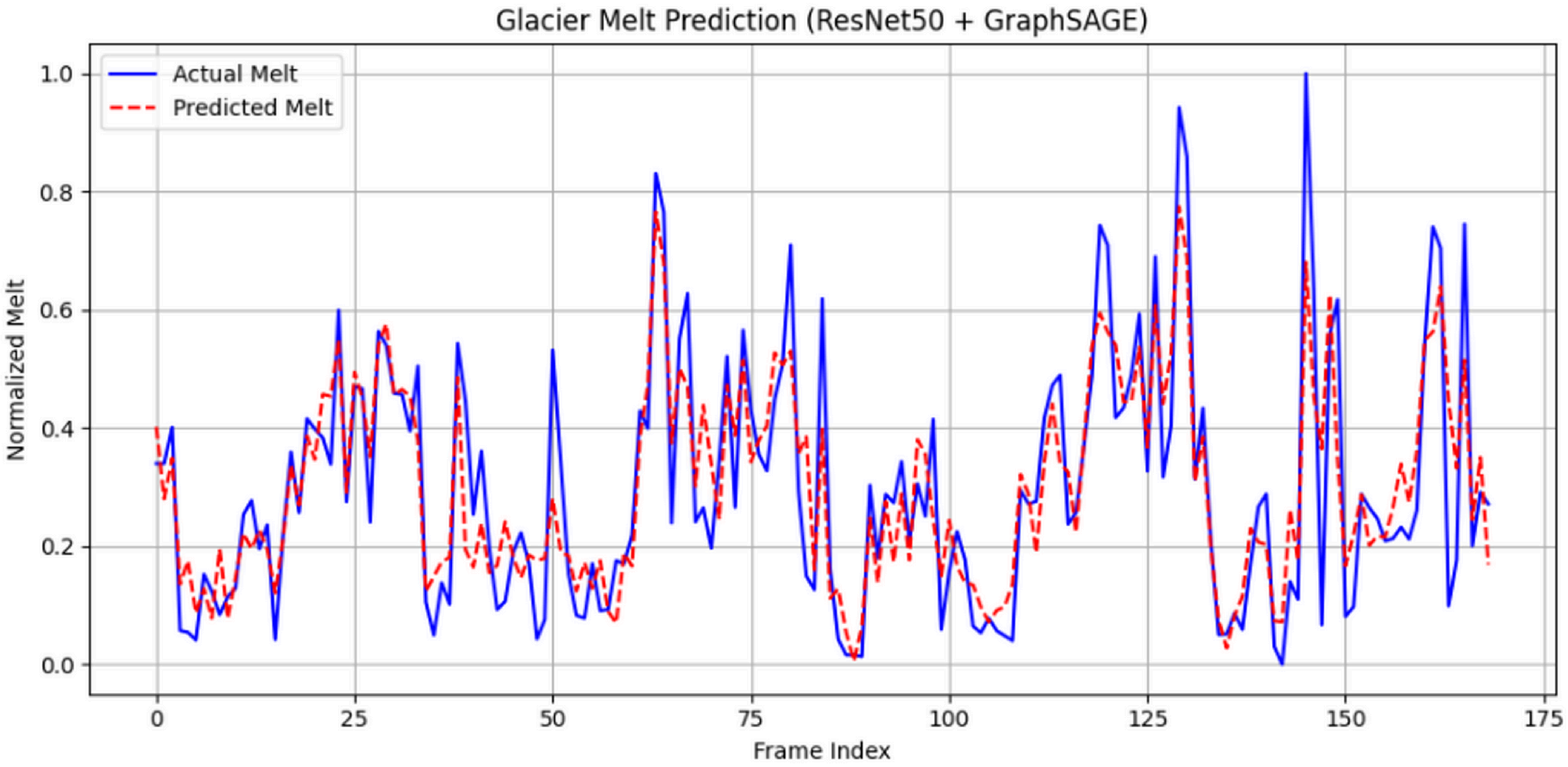
The model was trained over 600 epochs, allowing sufficient iterations for the network to learn complex patterns and converge towards optimal parameters.

Batch Size: Full Sequence

Due to the graph-structured input, the batch size was set to process the full sequence (entire GIF as one graph) per sample, maintaining temporal coherence.

Loss values were printed every 50 epochs to continuously monitor the model's convergence and training progress.

Evaluation & Plot



The plot illustrates a critical aspect of our evaluation: comparing the model's predicted melt against the actual melt (derived from pixel differences) across the temporal sequence. The red dashed line represents the model's output, while the blue line signifies the ground truth melt. The high degree of visual alignment between these two lines confirms that our spatio-temporal graph neural network is capable of accurately tracking the complex progression of glacier melting over time, validating the effectiveness of our combined CNN-GNN architecture.

Future Work & Expansion

Our future work focuses on expanding the scope and enhancing the capabilities of our glacier melt prediction model. These planned improvements will lead to a more robust and generalizable system, addressing current limitations and opening new avenues for research in glaciology.

Dataset Expansion

Expand the dataset to cover a broader range of geographical regions and extend across more years, enhancing model robustness.

Spatial Edge Connections

Incorporate sophisticated spatial edge connections, such as pixel clusters, to capture more intricate local interactions within glacier images.

Metadata Integration

Integrate external metadata like weather patterns and temporal information as additional node features, enriching the contextual understanding of melt.

Advanced GNN Architectures

Experiment with advanced GNN architectures such as Graph Attention Networks (GAT) or Temporal GCNs for richer dynamic modeling.

Conclusion

Glacier Melt Pipeline

Developed a prediction pipeline using satellite images and a CNN + Spatio-Temporal GNN.

Data Simulation

Successfully simulated and predicted melt trends.

Foundation for Monitoring

Provides a foundation for climate monitoring and early warning systems.

References

- Papers on Spatio-Temporal GNNs
- NASA Worldview / Sentinel datasets
- ResNet18 architecture
- Graph Neural Networks tutorials (PyTorch Geometric)