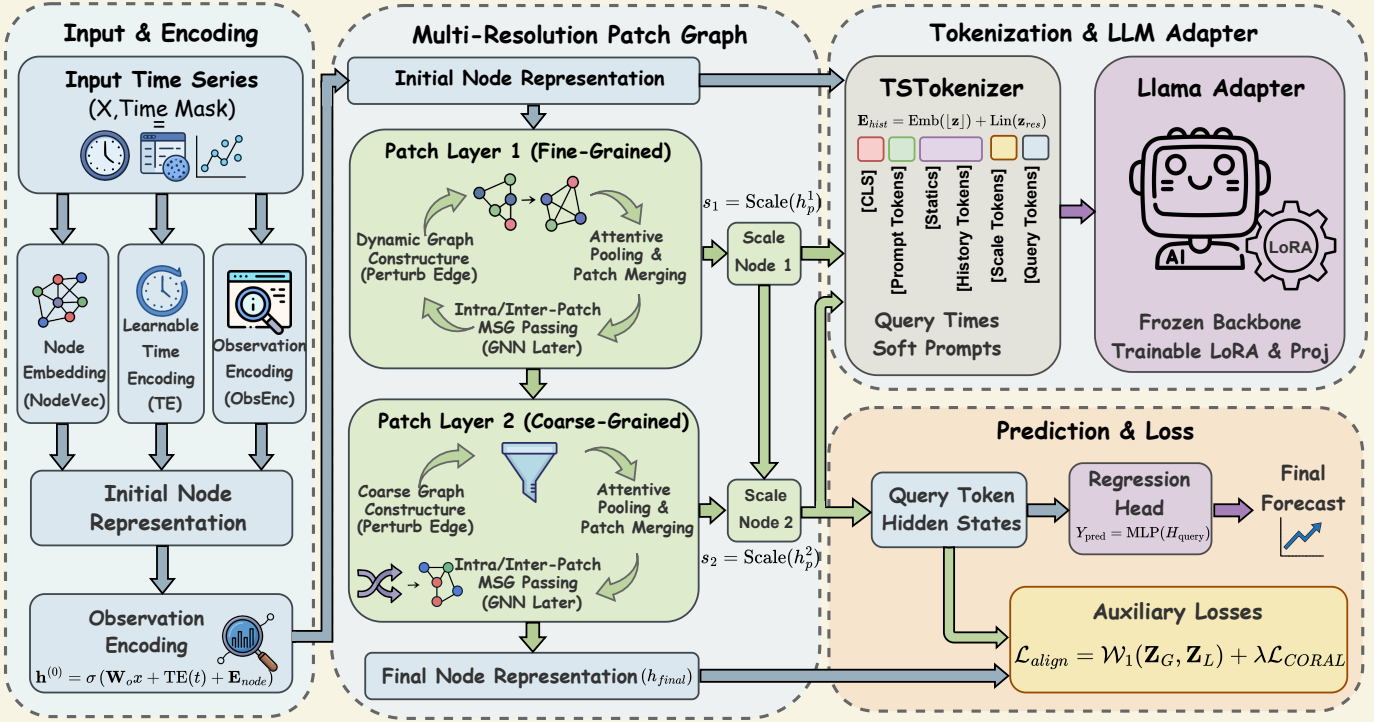
Overview of PatchGNN-LLM Architecture
Input & Encoding
Input Time Series
(X,Time Mask)
Node Embedding (NodeVec)
Learnable Time Encoding (TE)
Observation Encoding (ObsEnc)
Initial Node Representation
Observation Encoding
$\mathbf{h}^{(0)} = \sigma(\mathbf{W}_o x + \mathrm{TE}(t) + \mathbf{E}_{node})$
Multi-Resolution Patch Graph
Initial Node Representation
Patch Layer 1 (Fine-Grained)
Dynamic Graph Constructure (Perturb Edge)
Attentive Pooling & Patch Merging
Intra/Inter-Patch MSG Passing (GNN Later)
$s_1 = \mathrm{Scale}(h_p^1)$
Scale Node 1
Patch Layer 2 (Coarse-Grained)
Coarse Graph Constructure (Perturb Edge)
Attentive Pooling & Patch Merging
Intra/Inter-Patch MSG Passing (GNN Later)
$s_2 = \mathrm{Scale}(h_p^2)$
Scale Node 2
Final Node Representation (h_{final})
Tokenization & LLM Adapter
TSTokenizer
$\mathbf{E}_{hist} = \mathrm{Emb}([\mathbf{z}]) + \mathrm{Lin}(\mathbf{z}_{res})$
[CLS]
[Prompt Tokens]
[Statics]
[History Tokens]
[Scale Tokens]
[Query Tokens]
Query Times
Soft Prompts
Llama Adapter
AI
LoRA
Frozen Backbone
Trainable LoRA & Proj
Prediction & Loss
Query Token Hidden States
Regression Head
$Y_{\mathrm{pred}} = \mathrm{MLP}(H_{\mathrm{query}})$
Final Forecast
Auxiliary Losses
$\mathcal{L}_{align} = \mathcal{W}_1(\mathbf{Z}_G, \mathbf{Z}_L) + \lambda \mathcal{L}_{CORAL}$