Learning and Clustering on Temporal Graphs: Principles, Primitives, and Pooling

Background

PRINCIPLES

Graph convolutional networks [1] employ message-passing functions on a graph operator (the **Laplacian**) that is skewed on high-degree nodes. This destroys the spectrum of the graph and does not allow faithful reconstruction and clustering, while other operators (the **Hashimoto** and **Bethe-Hessian**) are detectability-optimal down to the theoretical threshold in studied artificial scenarios – with *empirical* implications to real-world data and analysis [2,3].

PRIMITIVES

However, attaining these principled thresholds is currently limited in *theory* and in *practice* [4], leaving two gaps in the state of the art: no sparse GPU eigensolvers are fully implemented for asymmetric matrices (the **Hashimoto**), and the regularization for its symmetric surrogate (the **Bethe-Hessian**) is not complete for temporal and metadata-attributed graphs [5].

POOLING

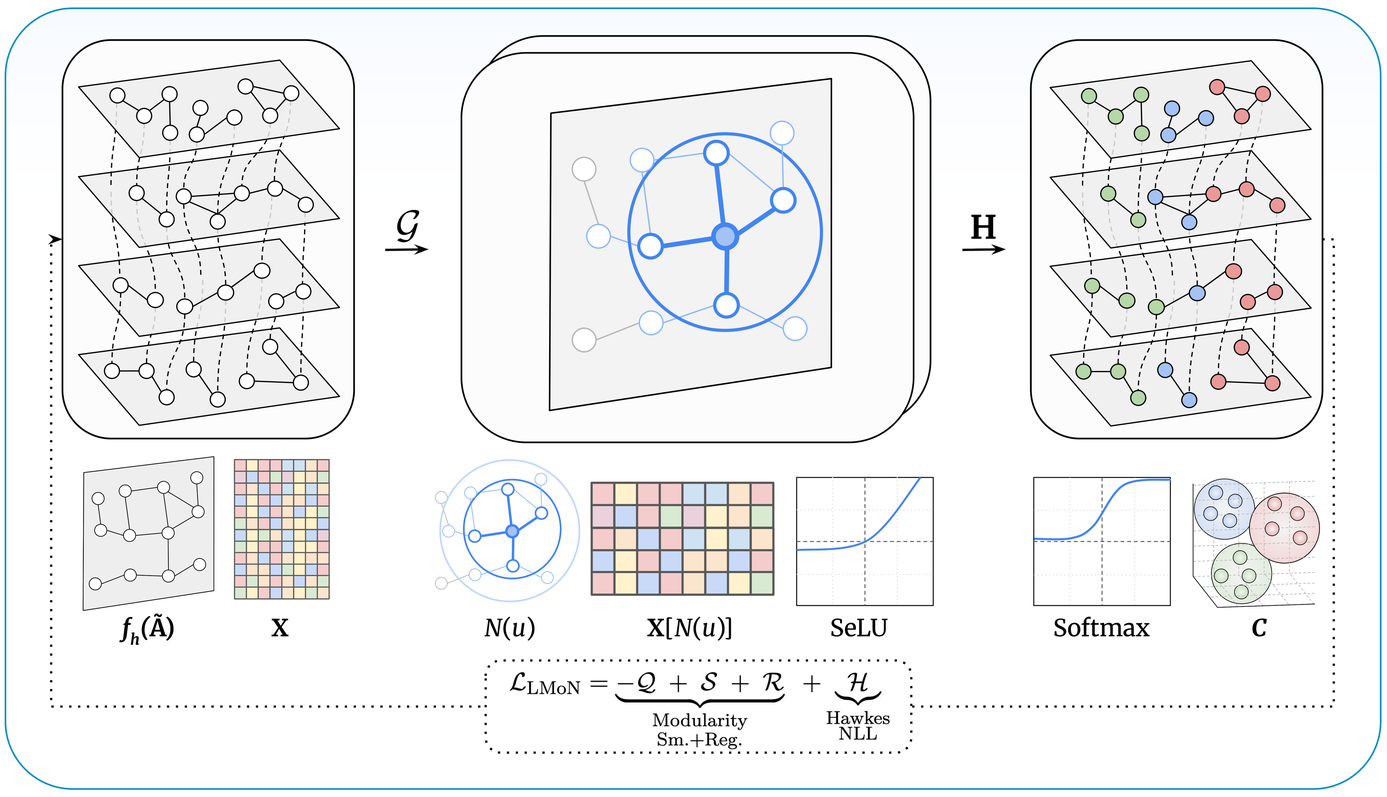
The same issue underlies *learning*: graph neural networks (GNNs) work by message-passing, which filters node signals by aggregation of features using the **Laplacian** eigenbasis of a graph, without explicit eigendecomposition. For the task of *clustering* nodes in a graph (*community detection* in network science or *pooling* in GNNs) [6,7], this introduces a *bottleneck in the operator*: machine learning models can not learn what is not there to be learned.

Our work

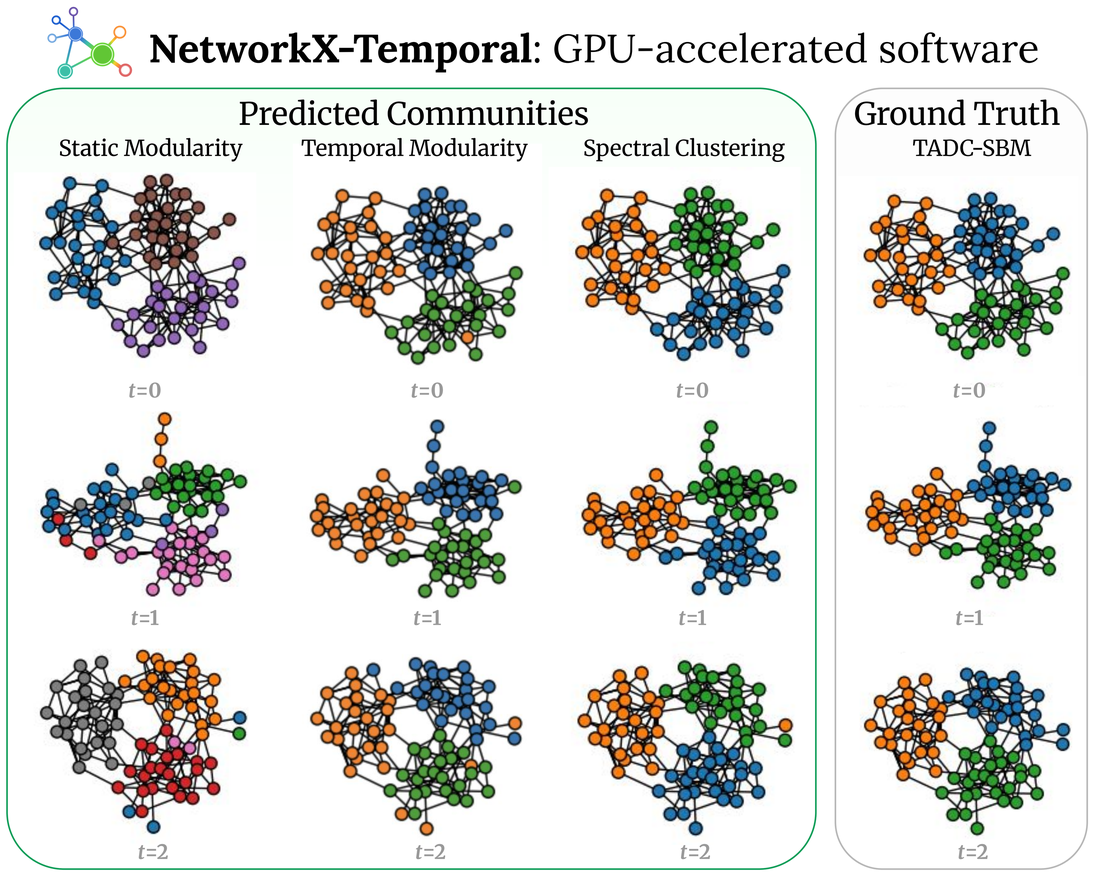
In a graph, node clusters (*communities*) can be *detected* depending on the amount of connections within and between them: methods attaining this threshold (Almeida-Thouless line or Kesten-Stigum bound) may provide *principled pooling primitives* for efficient learning on large-scale temporal graphs, with coarse-grained representations that preserve the graph spectrum.

Our work targets the regime where detection beyond the theoretical threshold is deemed *possible*, but not known to be *efficient*.

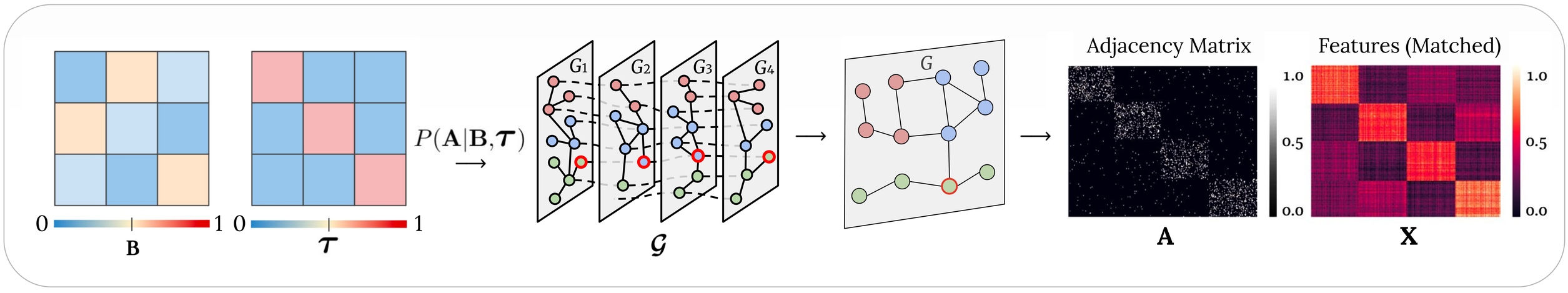
LMoN: neural optimization of longitudinal modularity



NetworkX-Temporal: GPU-accelerated software



TADC-SBM: a time-varying, attributed, degree-corrected stochastic block model for generating benchmark graphs



References:

[1] Semi-Supervised Classification with Graph Convolutional Networks. **Kipf, T. & Welling, M.**, ICLR, 2017.
[2] Detectability Thresholds and Optimal Algorithms for Community Structure in Dynamic Networks. **Ghasemian et al.**, PRX, 2016.
[3] Efficient Neural Community Detection on Time-Varying Attributed Graphs. **Passos, N.A.R.A.**, Thesis: University of Pisa, 2026.
[4] Accelerating Dynamic Graph Clustering on GPU Architectures with cuGraph. **Passos, N.A.R.A. et al.**, Euro-Par: FRAME, 2026.
[5] Community detection in sparse time-evolving graphs with a dynamical Bethe-Hessian. **Dall'Amico, L. et al.**, arXiv, 2020.
[6] Graph Clustering with Graph Neural Networks. **Tsitsulin et al.**, JMLR, 2023.
[7] Understanding Pooling in Graph Neural Networks. **Grattarola et al.**, IEEE TNNLS, 2021.

Preprint and code: