Temporal Graph Neural Tangent Kernel with Graphon-Guaranteed

Katherine Tieu*, Dongqi Fu*, Yada Zhu, Hendrik Hamann, Jingrui He

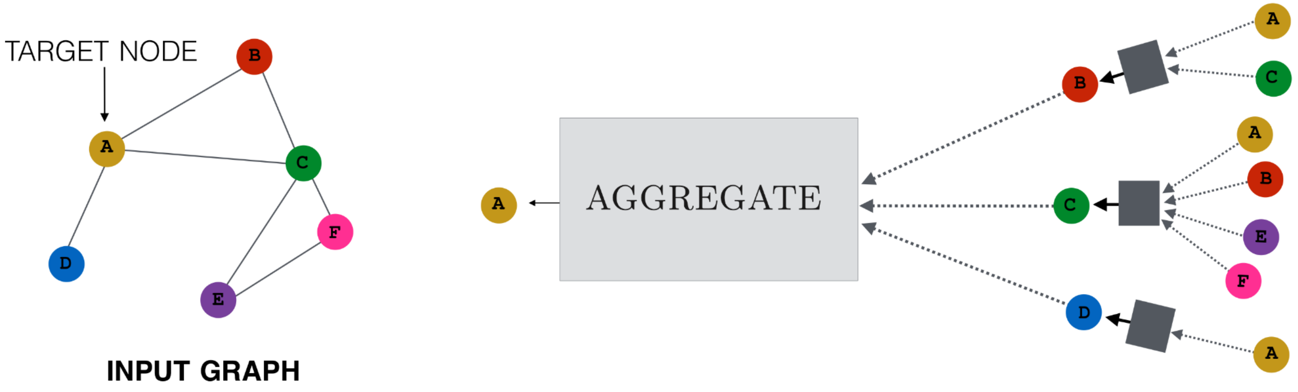
Presenter: Katherine Tieu (kt42@illinois.edu)

Github: https://github.com/kthrn22/TempGNTK

Limitation of Graph Neural Networks

- On the one hand, GNNs needs **complex neural architectures** for the powerful expressivity.
    - For example, in order to capture higher-order information of graphs, GNNs needs multiple layers of Message Passing, consisting of nonlinear aggregation and self-update modules.

- On the other hand, graph kernels enjoy the explicit formula and can be convex, leading to solid theoretical results, although their specific form is often **hand-crafted and may not be powerful** enough to support complicated application scenarios.



Graph Neural Tangent Kernel

GNTK [1] introduces a way to fuse Graph Neural Networks (GNNs) with Graph Kernels (GKs)

- GKs are easy to train and have provable theoretical guarantees
- GNNs has multi-layer architecture that can capture higher-order information of graphs, but they are hard to train
- GNTKs: graph kernels corresponding to infinite-width GNNs trained by gradient descent
  - Inherits advantages from both GNNs and GKs: expressive power of GNNs and easy-training of GKs
  - Establishes provable theoretical guarantees

[1] Du et al: Graph Neural Tangent Kernel: Fusing Graph Neural Networks with Graph Kernels. NeurIPS 2019

Motivation

- However, in the real world, the graph topology and features are inevitably evolving over time, e.g., the user connections and interests in social networks.

- This temporal evolution brings new challenges to GNTK as to how the similarity of temporal graphs is measured and how the corresponding kernel matrix is derived.

- To be more specific, how can we design a temporal graph neural tangent kernel, which
  - not only has a superior representation ability than **temporal graph neural networks** [1, 2]
  - but also inherits the expression simplicity and analysis rigorousness of **graph neural tangent kernels** [3, 4]?

[1] Rossi et al: Temporal Graph Networks for Deep Learning on Dynamic Graphs, arXiv 2020
[2] Cong et al: Do We Really Need Complicated Model Architectures For Temporal Networks? ICLR 2023
[3] Du et al: Graph Neural Tangent Kernel: Fusing Graph Neural Networks with Graph Kernels. NeurIPS 2019
[4] Krishnagopal et al: Graph Neural Tangent Kernel: Convergence on Large Graphs. ICML 2023

Proposed Temp-G^{3}NTK

- Desired Temp-G^{3}NTK value:

$$K(G^{(t)}, G^{'(t)}) = \sum_{v \in V^{(t)}} \sum_{v' \in V^{'(t)}} \boldsymbol{\Theta}^{(L)}(G^{(t)}, G^{'(t)})_{vv'}$$

- An iterative process to compute $\boldsymbol{\Theta}^{(l)}$ corresponding to the l-th MLPs layer:

1. $\boldsymbol{\Theta}^{(l)}(G^{(t)}, G^{'(t)})_{uu'} = \boldsymbol{\Theta}^{(l-1)}(G^{(t)}, G^{'(t)})_{uu'} \cdot \dot{\boldsymbol{\Sigma}}^{(l)}(G^{(t)}, G^{'(t)})_{uu'} + \boldsymbol{\Sigma}^{(l)}(G^{(t)}, G^{'(t)})_{uu'}$
2. $\boldsymbol{\Lambda}^{(l)}(G^{(t)}, G^{'(t)})_{uu'} = \begin{pmatrix} \boldsymbol{\Sigma}^{(l-1)}(G^{(t)}, G^{'(t)})_{uu'} & \boldsymbol{\Sigma}^{(l-1)}(G^{(t)}, G^{(t)})_{uu} \\ \boldsymbol{\Sigma}^{(l-1)}(G^{'(t)}, G^{(t)})_{u'u} & \boldsymbol{\Sigma}^{(l-1)}(G^{'(t)}, G^{'(t)})_{u'u'} \end{pmatrix}$
3. $\boldsymbol{\Sigma}^{(l)}(G^{(t)}, G^{'(t)})_{uu'} = \mathbb{E}_{(a,b)\sim\mathcal{N}(0,\boldsymbol{\Lambda}^{(l)}(G^{(t)},G^{'(t)})_{uu'})}[\sigma(a)\cdot\sigma(b)] =$

$$= \frac{\pi - \arccos(\boldsymbol{\Sigma}^{(l-1)}(G^{(t)}, G^{'(t)})_{uu'})}{2\pi} + \frac{\sqrt{1 - (\boldsymbol{\Sigma}^{(l-1)}(G^{(t)}, G^{'(t)})_{uu'})^2}}{2\pi}$$

- At initialization, $l = 0$:

$$\boldsymbol{\Theta}^{(0)}(G^{(t)}, G^{'(t)})_{uu'} = \boldsymbol{\Sigma}^{(0)}(G^{(t)}, G^{'(t)})_{uu'} = \mathbf{h}_u(t)^T \mathbf{h}'_u(t)$$

Theory of Temp-G^{3}NTK

- Temp-G^{3}NTK satisfies **symmetric** (Theorem 5.1) and **positive semi-definite** (Theorem 5.2).

- Generalization error of Temp-G^{3}NTK is **upper bounded** by a **data-dependent term** (Theorem 5.3)
  - Generalization error:

$$\sum_{t=1}^{T} \mathbb{E}[\ell(f_{kernel}(G^{(t)})y) \,|\, \{G^{(1)}, \ldots, G^{(t-1)}\}] - \ell(f_{kernel}(G^{(t)}), y)$$

    the expected value is taken over all $G^{(t)}$ that is drawn from $\mathbb{P}_t(\,.\,|\, G^{(1)}, \ldots, G^{(t-1)})$. The summation shows the total generalization error of all timestamps, as we want to investigate how Temp-G^{3}NTK operates on each timestamps.
  - Data-dependent upper bound: $\mathcal{O}\left(\sup_t \mathbf{y}^T [\mathbf{K}^{(t)}_{train}]^{-1} \mathbf{y} \cdot \mathrm{tr}(\mathbf{K}^{(t)}_{train}) \right)$ obtained from training graphs' information.

- Under the setting of **growing temporal graphs** (the number of nodes grows with respect to time), Temp-G^{3}NTK converges to a limit object (Theorem 5.4), which is the graphon NTK value.

Performance of Temp-G^{3}NTK

Graph-level Classification

- Given n training temporal graphs and their associated label, and Temp-G^{3}NTK returns label for a testing graph
- Baselines: (1) Graph Kernels, (2) Neural Representation Learning methods, and (3) Temporal GNNs

Table 2: Comparison of Temporal Graph Classification Accuracy.

Method	Infectious	DBLP	Facebook	Tumblr
WL-Subtree [42]	0.600 ± 0.044	0.520 ± 0.068	0.650 ± 0.075	0.570 ± 0.121
Shortest Path [5]	0.670 ± 0.075	0.560 ± 0.049	0.560 ± 0.086	0.580 ± 0.143
Random Walk [46]	0.670 ± 0.073	0.530 ± 0.058	0.590 ± 0.093	0.580 ± 0.112
Graph2Vec [33]	0.565 ± 0.081	0.539 ± 0.031	0.538 ± 0.028	0.547 ± 0.071
NetLSD [45]	0.625 ± 0.061	0.558 ± 0.035	0.535 ± 0.011	0.552 ± 0.046
GL2Vec [6]	0.545 ± 0.051	0.562 ± 0.030	0.538 ± 0.031	0.558 ± 0.080
GraphMixer [8]	0.500 ± 0.000	0.563 ± 0.011	0.561 ± 0.023	0.509 ± 0.508
TGN [39]	0.520 ± 0.019	0.580 ± 0.003	0.559 ± 0.018	0.517 ± 0.025
EvolveGCN [35]	0.521 ± 0.093	0.400 ± 0.089	0.516 ± 0.075	0.395 ± 0.089
Temp-G^{3}NTK (Ours)	$\mathbf{0.740 \pm 0.058}$	$\mathbf{0.600 \pm 0.063}$	$\mathbf{0.700 \pm 0.138}$	$\mathbf{0.630 \pm 0.068}$

Node-level Classification

- Given temporal graphs that has node labels change w.r.t time, and Temp-G^{3}NTK returns the label for a queried node at a queried time
- Baselines: SOTAs from Temporal Graph Learning Benchmark TGB [1]

Table 4: NDCG Score for Node Property Prediction on the tgbn-trade Dataset.

Method	Validation	Test
DyGFormer [49]	$\mathbf{0.408 \pm 0.006}$	$\mathbf{0.388 \pm 0.006}$
TGN [39]	0.395 ± 0.002	0.374 ± 0.001
DyRep [44]	0.394 ± 0.001	0.374 ± 0.001
Temp-G^{3}NTK (Ours)	0.397 ± 0.039	0.380 ± 0.008

[1] Huang et al: Temporal Graph Benchmark for Machine Learning on Temporal Graphs. NeurIPS 2023

Thanks!

Katherine Tieu*, Dongqi Fu*, Yada Zhu, Hendrik Hamann, Jingrui He

TL;DR: **Getting the Temporal Graph Neural Representations without Graph Neural Networks !**

Github: https://github.com/kthrn22/TempGNTK