CoDy: Counterfactual Explainers for Dynamic Graphs

Zhan Qu *, Daniel Gomm *, Michael Färber

TECHNISCHE UNIVERSITÄT DRESDEN · ScaDS.AI DRESDEN LEIPZIG · KIT Karlsruher Institut für Technologie · UNIVERSITY OF AMSTERDAM · CWI Centrum Wiskunde & Informatica

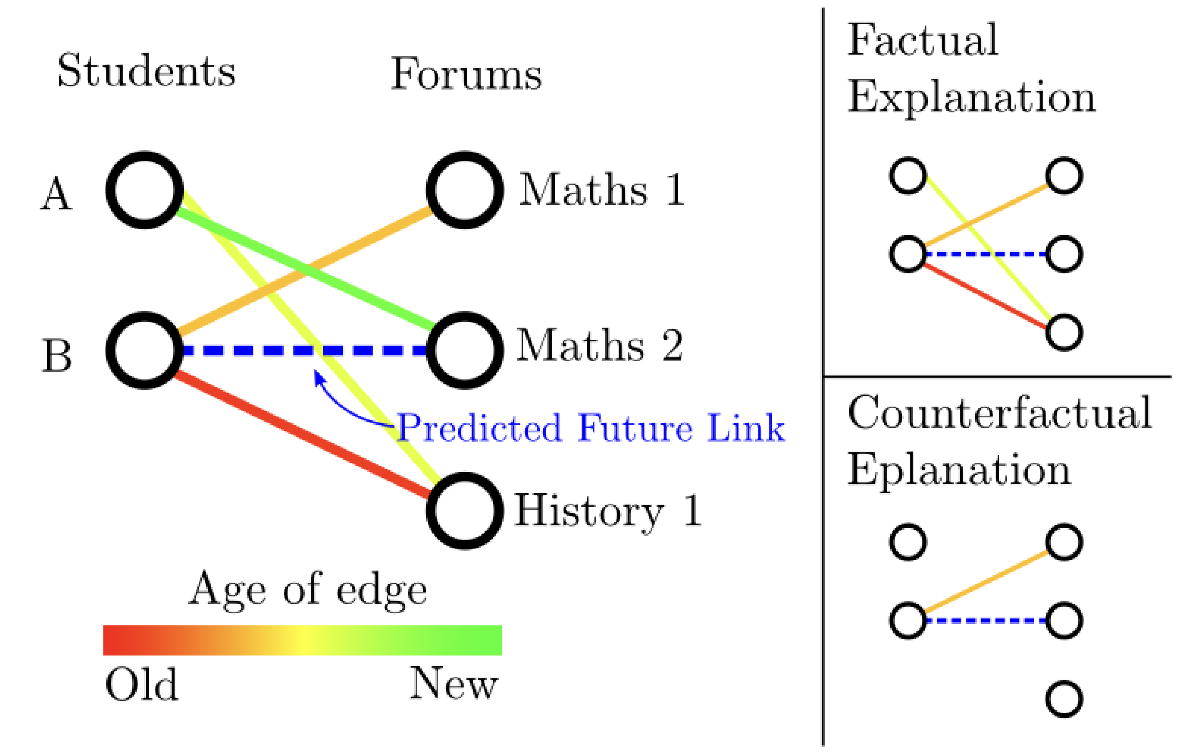
ICML International Conference On Machine Learning

Introduction

- Temporal Graph Neural Networks (TGNNs) are powerful for modeling dynamic systems where relationships and features change over time.
- Current explainability methods mostly cater to:
  - Static Graphs
  - Discrete-time Dynamic Graphs
  - Factual Explanations

Counterfactual Explanations

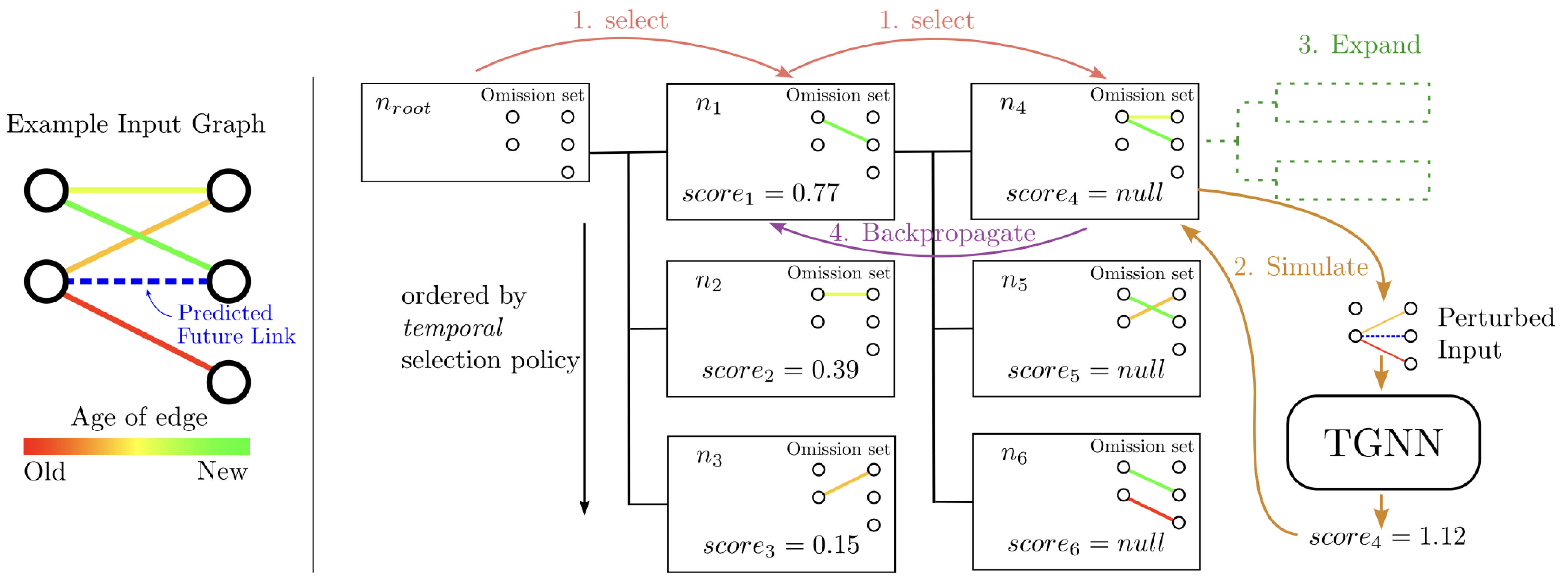
- Counterfactual Explanations: What minimal change to the input would alter the prediction.
  - More intuitive and actionable.
  - Help identify model biases and establish causal links



Contributions

1. First counterfactual explanation method for Temporal Graph Neural Networks
2. GreeDy: a baseline for counterfactual explanaitions in dynamic graphs
3. Evaluation framework for dynamic graph explanations
4. CoDy outperforms counterfactual and factual baselines

Framework



Search Policies

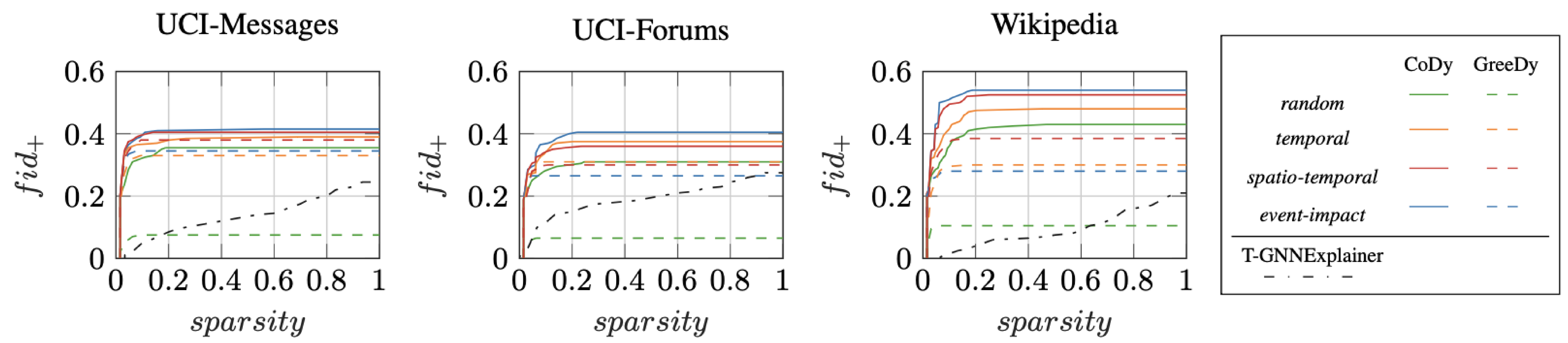


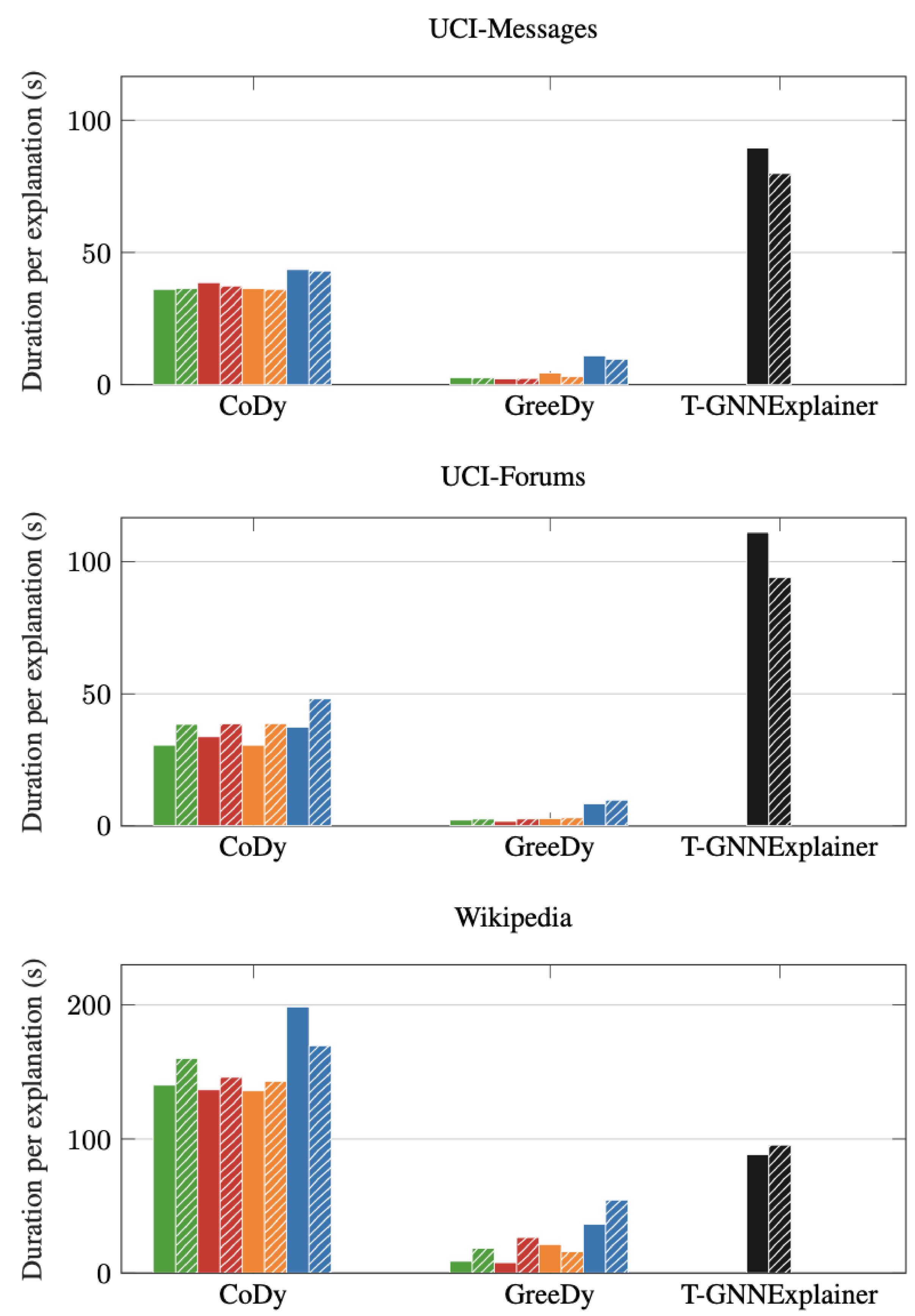
Figure 3. Cumulative fid_+ score relative to an upper $sparsity$ limit for incorrect predictions with TGN as target model. PGExplainer is excluded since it is assessed with a fixed sparsity.

Experiments

Table 1. Results for the $AUFSC_+$, $AUFSC_-$, and *char* scores of different explanation methods applied to the TGN model. Results are reported for three datasets: UCI-Messages (msg.), UCI-Forums (for.), and Wikipedia (wiki.). The best result for each experimental setting is shown in **bold**, and the second best is underlined.

	$AUFSC_+$						$AUFSC_-$						*char*					
	Correct			Incorrect			Correct			Incorrect			Correct			Incorrect		
Dataset	msg.	for.	wiki.	msg.	for.	wiki.	msg.	for.	wiki.	msg.	for.	wiki.	msg.	for.	wiki.	msg.	for.	wiki.
PGExplainer	0.02	0.03	0.03	0.08	0.04	0.11	0.39	0.35	0.67	0.61	0.67	0.54	0.05	0.07	0.09	0.17	0.08	0.22
T-GNNExplainer	0.05	0.03	0.01	0.14	0.19	0.10	0.45	0.36	0.61	0.53	0.49	0.43	0.17	0.08	0.09	0.39	0.40	0.34
GreeDy-*rand.*	0.02	0.05	0.04	0.07	0.06	0.10	0.33	0.27	0.53	**0.95**	**0.97**	**0.91**	0.04	0.08	0.09	0.14	0.12	0.19
GreeDy-*temp.*	0.13	0.41	0.08	0.32	0.30	0.29	0.52	0.58	0.72	**0.95**	0.95	0.87	0.22	0.50	0.17	0.49	0.47	0.45
GreeDy-*spa-temp.*	**0.19**	**0.44**	0.12	0.37	0.29	0.37	0.64	0.60	0.76	0.93	0.91	0.85	0.31	0.53	0.23	0.54	0.46	0.54
GreeDy-*evnt-impct*	0.10	0.28	0.07	0.34	0.26	0.27	0.62	0.61	0.67	**0.95**	0.96	0.88	0.18	0.39	0.14	0.51	0.42	0.43
CoDy-*rand.*	0.10	0.30	0.12	0.34	0.30	0.41	0.63	0.59	0.82	0.91	0.92	0.82	0.19	0.43	0.24	0.52	0.47	0.58
CoDy-*temp.*	0.13	0.36	0.11	0.38	0.36	0.46	0.64	0.58	0.83	0.92	0.93	0.84	0.23	0.49	0.22	0.55	0.54	0.62
CoDy-*spa-temp.*	**0.19**	0.43	**0.16**	0.39	0.35	0.50	**0.67**	**0.63**	**0.84**	0.92	0.90	0.82	**0.31**	**0.54**	**0.30**	0.57	0.52	0.65
CoDy-*evnt-impct*	0.16	0.38	0.14	**0.40**	**0.39**	**0.52**	0.65	0.61	0.82	0.92	0.90	0.85	0.27	0.50	0.27	**0.58**	**0.57**	**0.68**

Efficiency



Conclusion

- CoDy sets a new benchmark in explaining TGNNs
- It excels at identifying concise, necessary, and sufficient explanations—especially for incorrect predictions where it reveals model limitations.
- The spatio-temporal and event-impact policies are the most effective
- CoDy adapts its search dynamically, avoiding local optima—unlike GreeDy, which is faster but less flexible. In real-world use:
  - CoDy is ideal when accuracy and insight are critical.
  - GreeDy is a good alternative when speed is the priority.