Evaluating Graph Neural Networks for Epidemic Source Detection: A Benchmark Study

Martin Sterchi [1,2] Lorenz Hilfiker [3]

[1]University of Applied Sciences Northwestern Switzerland [2]University of Zürich [3]University of Bern

Introduction

Epidemic source detection has been actively researched since Shah and Zaman [1] introduced the problem on static networks in 2010. More recently, several studies have explored using Graph Neural Networks (GNNs) for source identification [2–5].

Problem formulation

We consider static, undirected networks on which a continuous-time *susceptible-infectious-recovered* (SIR) infection process unfolds. The SIR parameters are known, and a full snapshot of node states is observed at a known time T after the process begins.

Goal: Infer the **single source** of the epidemic based on the network and the observed node states (Fig. 1). This constitutes a **graph prediction** problem.

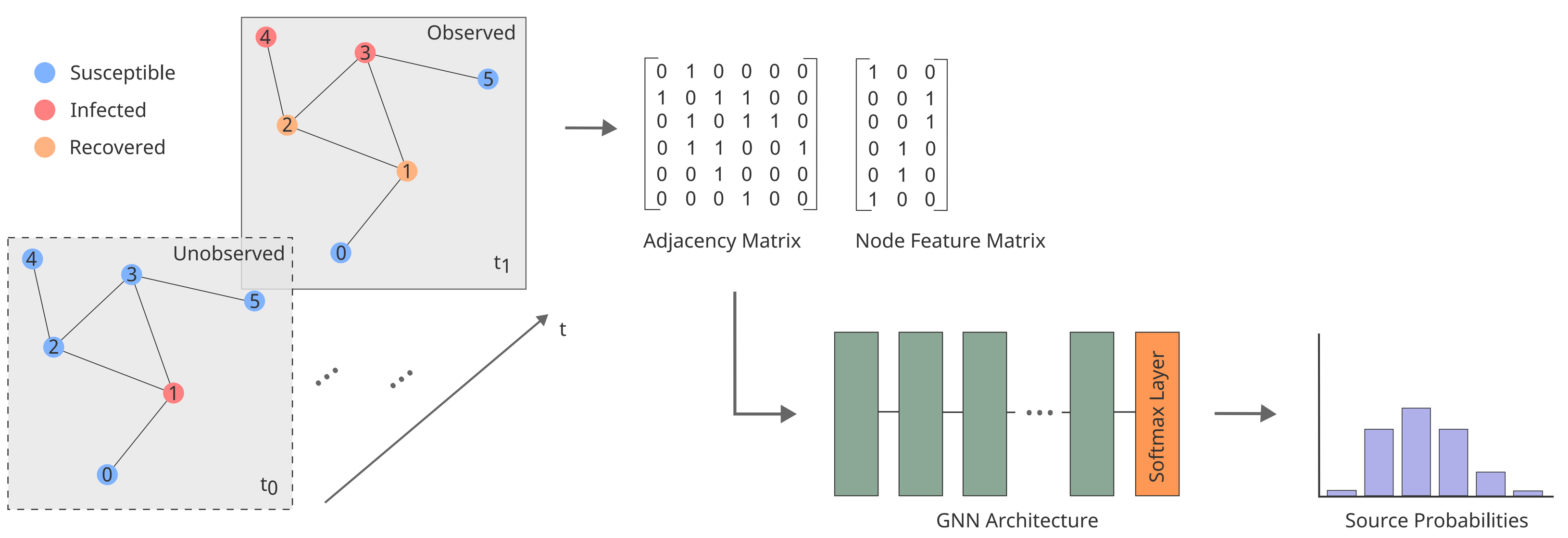

Figure 1. At time $t_1 = t_0 + T$ we observe a full snapshot of the epidemic which translates into the two inputs for the GNN, the adjacency matrix of the graph and the one-hot encoded epidemic states of all nodes. The final output of the GNN is a source probability distribution.

Research question

Recent GNN-based approaches to source detection offer limited insight into how these models compare with traditional methods. This raises the central research question of our work: How do GNNs perform relative to traditional source detection methods, and what is their true potential for this task?

Our GNN architecture

Message-passing layers

Our architecture is primarily based on L graph convolution layers [6] that, for any node v, can be defined as follows:

$$\mathbf{h}_v^{(l)} = \text{ReLU}\left(\mathbf{B}_l\, \mathbf{h}_v^{(l-1)} + \mathbf{W}_l \sum_{u \in \mathcal{N}(v)} \mathbf{h}_u^{(l-1)}\right) \qquad l = 1 \ldots, L$$

The initial embedding of a node is the one-hot encoded representation of its epidemic state, i.e., $\mathbf{h}_v^{(0)} \in \{0,1\}^3$. The final convolutional layer returns the embedding matrix $\mathbf{H}^{(L)} \in \mathbb{R}^{|V| \times s}$, where s denotes the embedding dimension of the final layer.

Output layer

The output layer constitutes of a linear transformation followed by a softmax activation, i.e., $\hat{\mathbf{y}} = \text{Log-Softmax}\left[\mathbf{H}^{(L)}\mathbf{w}\right]$, with $\hat{\mathbf{y}}$ representing the (log-) source probability distribution over $|V|$ nodes.

Training setup

- Simulation of 500 training instances per node in the graph.
- Split in 90% training and 10% validation instances.
- Loss function: negative log likelihood loss.
- Mini-batch gradient descent (batch size: 128).
- Optimizer: Adam (learning rate: 0.001).
- Dropout rate up to layer $L-1$ is 0.05, for layer L it is 0.2.
- Early stopping based on validation loss with a patience of 5 epochs.

Benchmark methods

- Random selection of the source (RANDOM).
- Jordan centrality (CENTRALITY) [7].
- Untrained GNN (GNN-UNTRAINED).
- Soft margin estimator (SME) [8].
- Factorized likelihood method utilizing Monte Carlo estimates of node state probabilities (MCMF) [9].

References

[1] Shah, D. et al., *SIGMETRICS Perform. Eval. Rev.* 38, 1 (2010). 10.1145/1811099.1811063
[2] Dong, M. et al., *CIKM*, (2019). 10.1145/3357384.3357994
[3] Shah, C. et al., *arXiv*, (2020). https://arxiv.org/abs/2006.11913
[4] Sha, H. et al., *IEEE DSAA*, (2021). 10.1109/DSAA53316.2021.9564188
[5] Ru, X. et al., *AAAI*, (2023). 10.1609/aaai.v37i8.26152
[6] Morris, Ch. et al., *AAAI*, (2019). 10.1609/aaai.v33i01.33014602
[7] Zhu, K. et al., *IEEE TNET* 24, 1 (2016). 10.1109/TNET.2014.2364972
[8] Antulov-Fantulin, N. et al., *Phys. Rev. Lett.* 114, 248701 (2015). 10.1103/PhysRevLett.114.248701
[9] Sterchi, M. et al., *Scientific Reports* 13, 1 (2023). 10.1038/s41598-023-38282-8

[10] Haraldsdottir, S. et al., *J Acquir Immune Defic Syndr* 5, 4 (1992).

Results

Detection performance overall

We simulate 200 test outbreaks per node in the graph. The test outbreaks have a basic reproduction number of $R_0 \approx 2$ and the snapshots are taken when 40% of the nodes are infected, on average. The outbreaks in the train and test set have the same characteristics.

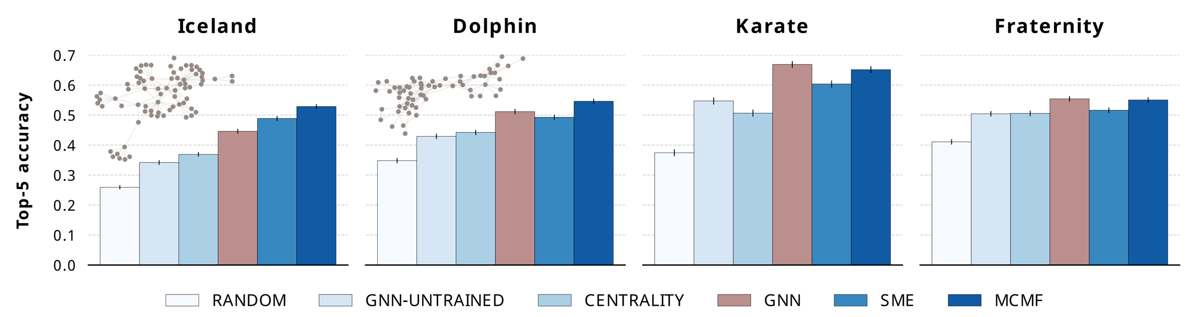

Figure 2. **Top-5 accuracy** of GNN and benchmark methods for the four empirical networks.

Name	$\|V\|$	$\|E\|$	$\langle k \rangle$	diam(G)
Iceland [10]	75	114	3.04	6
Dolphin [2]	62	159	5.13	8
Karate [2]	34	77	4.53	5
Fraternity [4]	58	967	33.34	3

Table 1. Overview of networks.

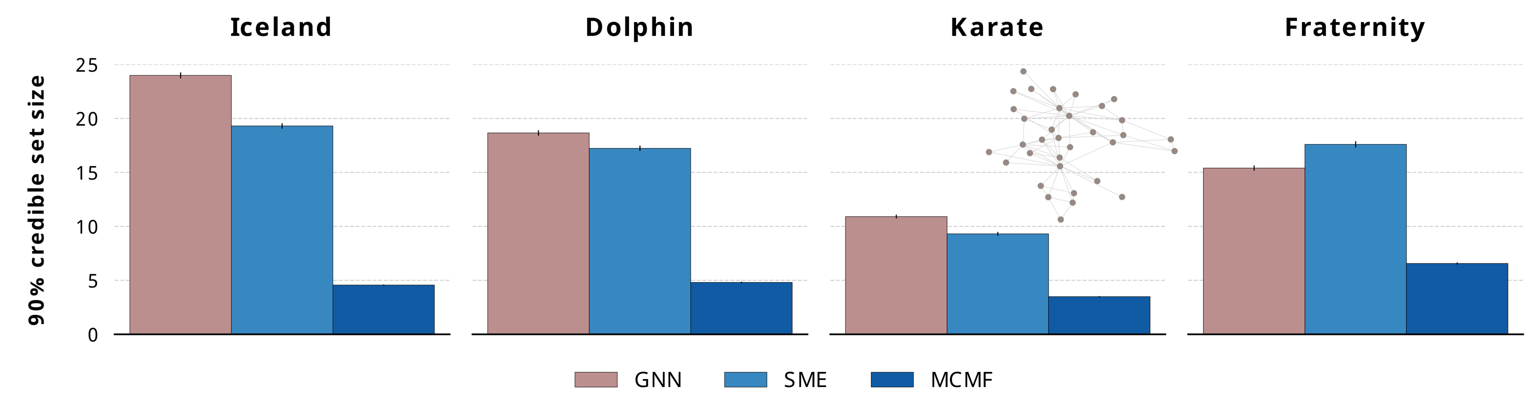

Figure 3. **90% credible set size** for the methods that output a distribution.

Detection performance by outbreak size

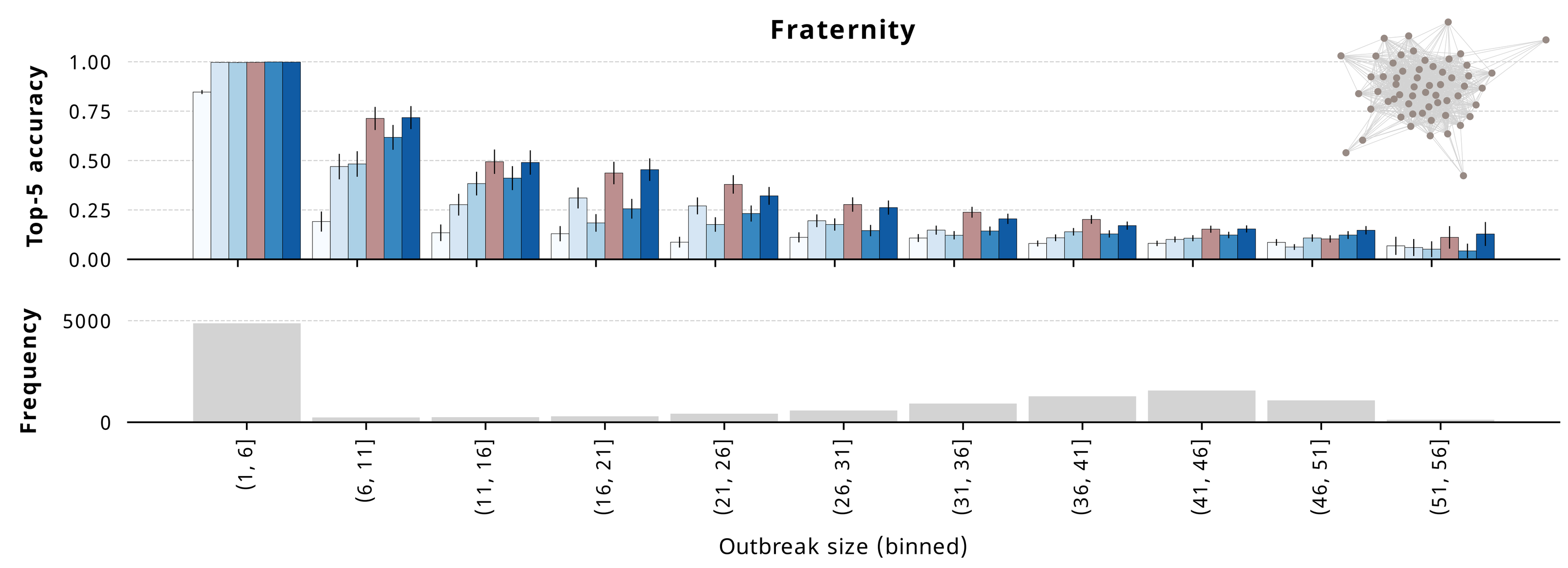

Figure 4. **Top-5 accuracy** of all methods for the Fraternity network, categorized by outbreak size.

Single oubreak scenario on Karate network

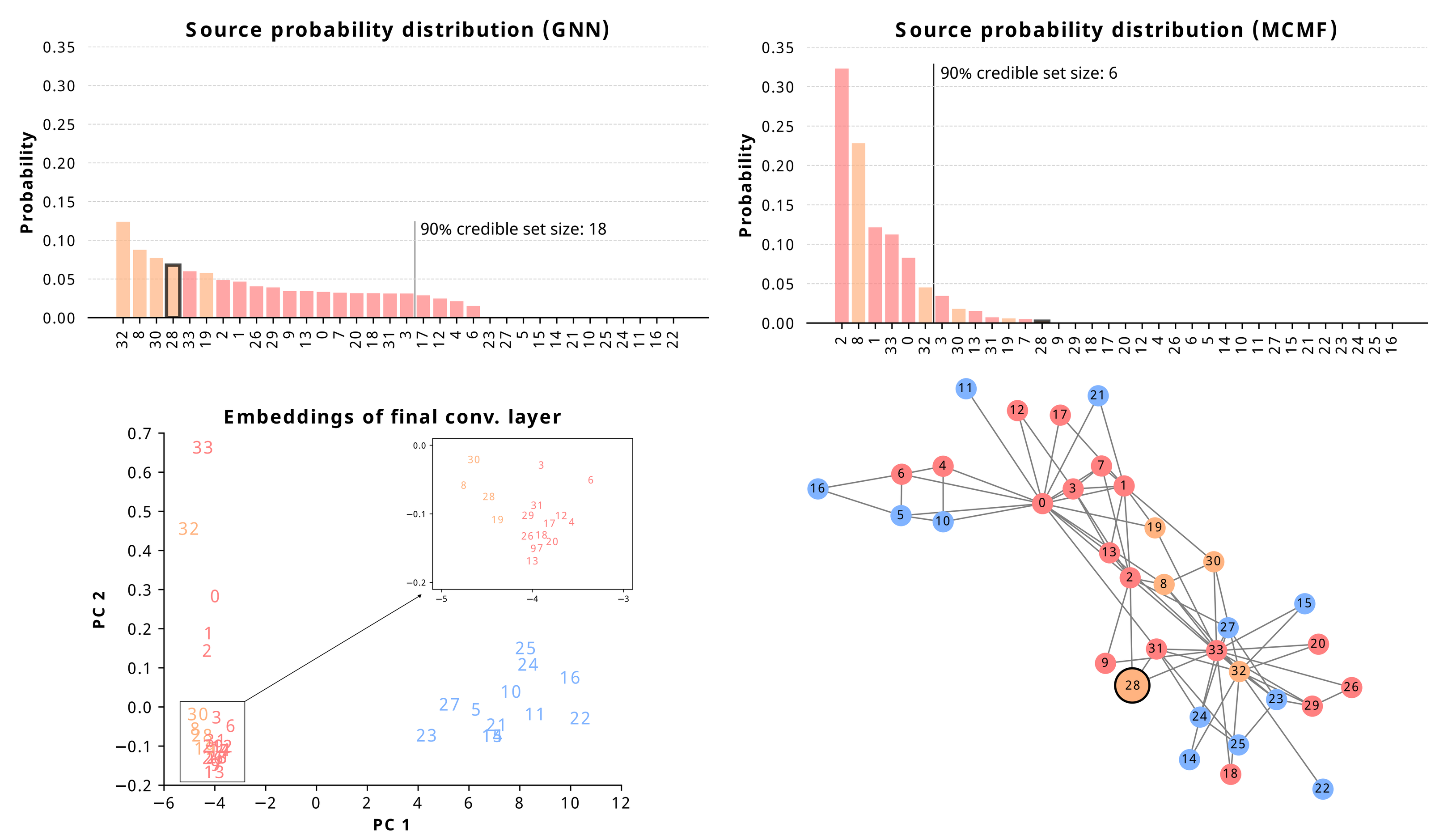

Figure 5. **Top left**: Output distribution of GNN. **Top right**: Output distribution of MCMF. **Bottom left**: First two principal component scores of the final convolutional layer embeddings of GNN. **Bottom right**: Karate network with nodes colored according to epidemic state. Node 28 is the true source.

Conclusions

GNNs perform on par with, or slightly better than, traditional methods. However, their full potential may remain untapped, as the GNN design space has yet to be systematically explored. A key drawback is that GNNs tend to produce larger credible sets.

Contact

Martin Sterchi
martin.sterchi@fhnw.ch
www.martinsterchi.ch

Lorenz Hilfiker
lorenz.hilfiker@gmail.com