

BOND: Benchmarking Unsupervised Outlier Node Detection on Static Attributed Graphs

Kay Liu^{1,*}, Yingtong Dou^{1,8,*}, Yue Zhao^{2,*}, Xueying Ding², Xiyang Hu²,
Ruitong Zhang³, Kaize Ding⁴, Canyu Chen⁵, Hao Peng³, Kai Shu⁵,
Lichao Sun⁶, Jundong Li⁷, George H. Chen², Zhihao Jia², Philip S. Yu¹

¹University of Illinois Chicago ²Carnegie Mellon University ³Beihang University

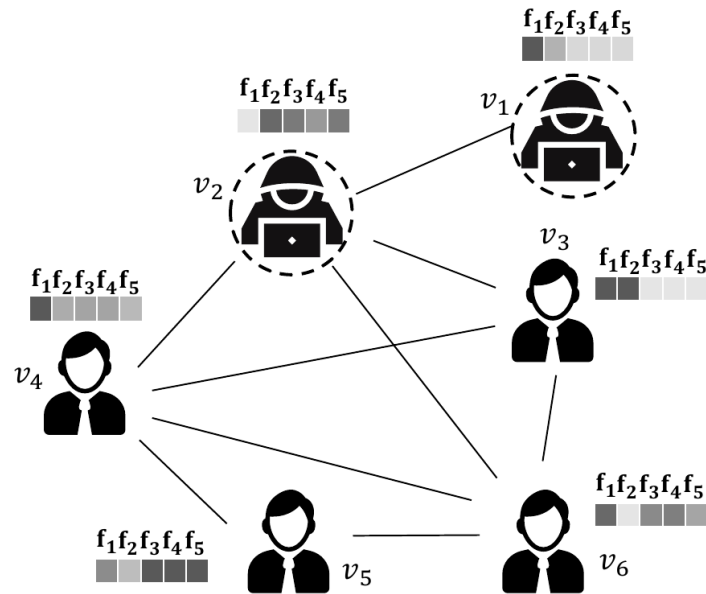
⁴Arizona State University ⁵Illinois Institute of Technology ⁶Lehigh University

⁷University of Virginia ⁸Visa Research

NeurIPS 2022 Datasets and Benchmarks Track



Unsupervised Outlier Node Detection on Static Attributed Graphs



https://github.com/kaize0409/GCN_AnomalyDetection/raw/master/framework.png

Outlier detection on graphs has broad applications:

- Financial fraud detection on transaction networks
- Fake account detection on social networks
- Spam review detection on e-commerce websites

Limitation of existing works:

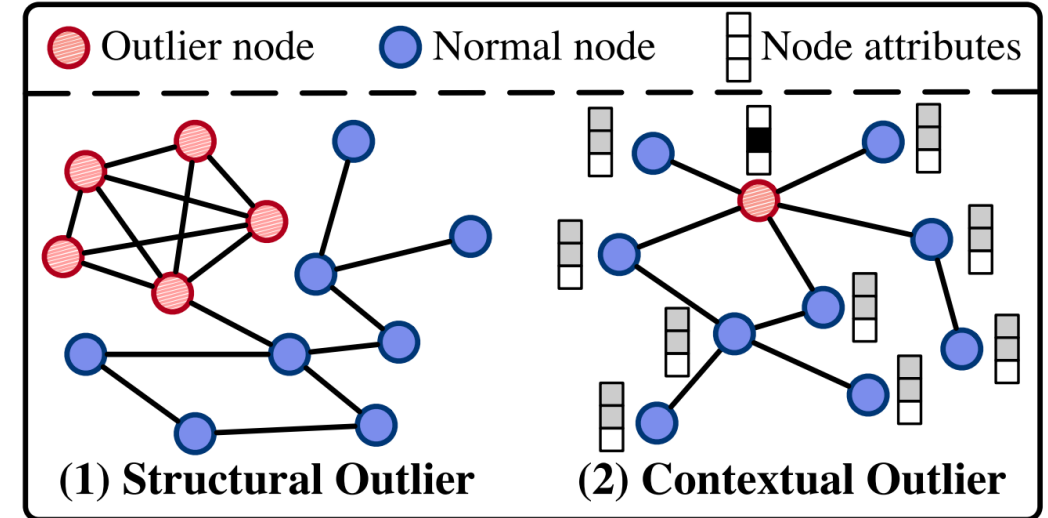
- Lack of a comprehensive benchmark
- Many algorithms only evaluated on synthetic outliers
- Limited analyses of efficiency in both time and space

Key Contributions of BOND

- **The first comprehensive benchmark** for *unsupervised outlier node detection*
- **Consolidated taxonomy** of outlier nodes (structural and contextual)
- **Systematic performance flaw** found in existing deep methods
- **Evaluation** of both *detection quality* and *computational efficiency*
- **Reproducible and accessible benchmark toolkit!**

Taxonomy of Outliers on the Node Level

- **Structural outliers** are densely connected nodes in contrast to sparsely connected regular nodes.
- **Contextual outliers** are nodes whose attributes are significantly different from their neighbor nodes.



3 Types of Benchmarked Datasets

- **Organic dataset:** real graph with organic (real) outliers

Weibo, Reddit, Disney, Books, Enron, DGraph

- **Injected datasets:** real graph with synthetic outliers

Cora, Amazon, Flickr

- **Generated datasets:** generated graph with synthetic outliers

Gen_Time, Gen_100, Gen_500, Gen_1000, Gen_5000, Gen_10000



<https://github.com/pygod-team/data>

14 Benchmarked Algorithms

- **Non-Graph**

- Similarity based: LOF (2000)
- Tree based: IF (2012)
- MLP+AE: MLPAE (2014)

- **Non-Deep**

- Cluster: SCAN (2007)
- MF: Radar (2017), ANOMALOUS (2018)

- **Deep Graph**

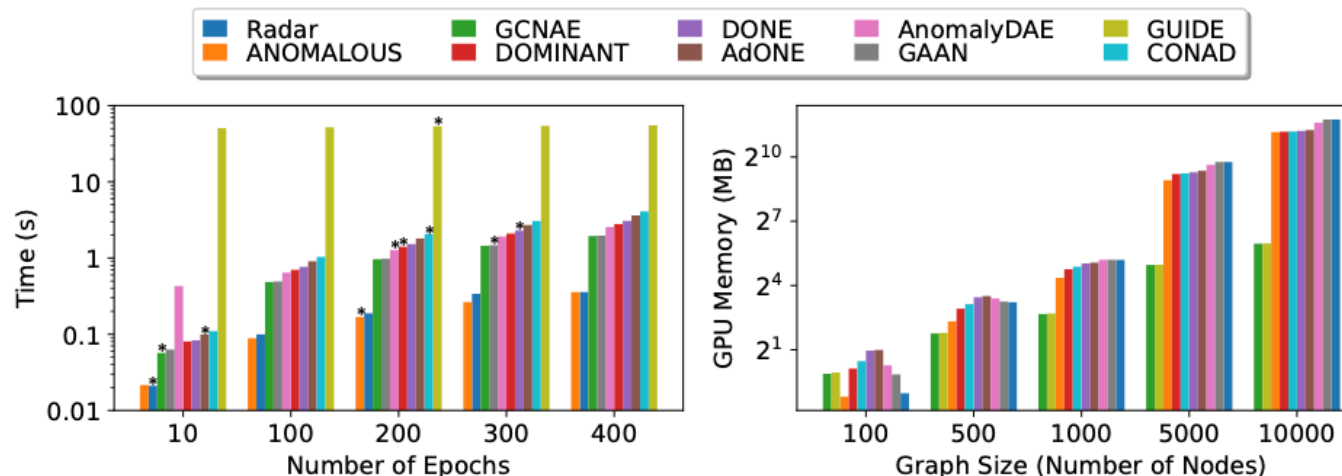
- GAN: GAAN (2020)
- MLP+AE: DONE (2020), AdONE (2020)
- GNN+AE: GCNAE (2016), DOMINANT (2019), AnomalyDAE (2020), GUIDE (2021), CONAD (2022)



<https://pygod.org>

Result Highlight

- **How effective are the algorithms on detecting synthetic and organic outliers?**
 - No outlier node detection method is universally the best on all datasets
 - Most deep methods evaluated fail to detect organic outliers
- **How do algorithms perform under two types of outliers (structural and contextual)?**
 - No method achieves high detection accuracy for both structural and contextual outliers
- **How efficient are algorithms in terms of time and space?**



BOND Resource



PyGOD: <https://pygod.org>

Benchmark: <https://pygod.org/tree/main/benchmark>

Data: <https://github.com/pygod-team/data>

Docs: <https://docs.pygod.org>

Email: benchmark@pygod.org

Presenter: Kay Liu
University of Illinois Chicago

Email: zliu234@uic.edu

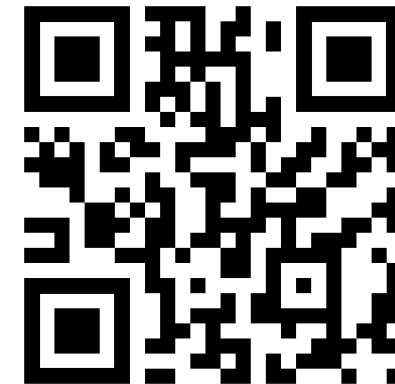
Homepage: <https://kayzliu.com/>

Wechat: kayzliu

Twitter: [@kayzliu](https://twitter.com/kayzliu)



pygod



kayzliu