CISPA HELMHOLTZ CENTER FOR INFORMATION SECURITY

Link Stealing Attacks Against Inductive Graph Neural Networks

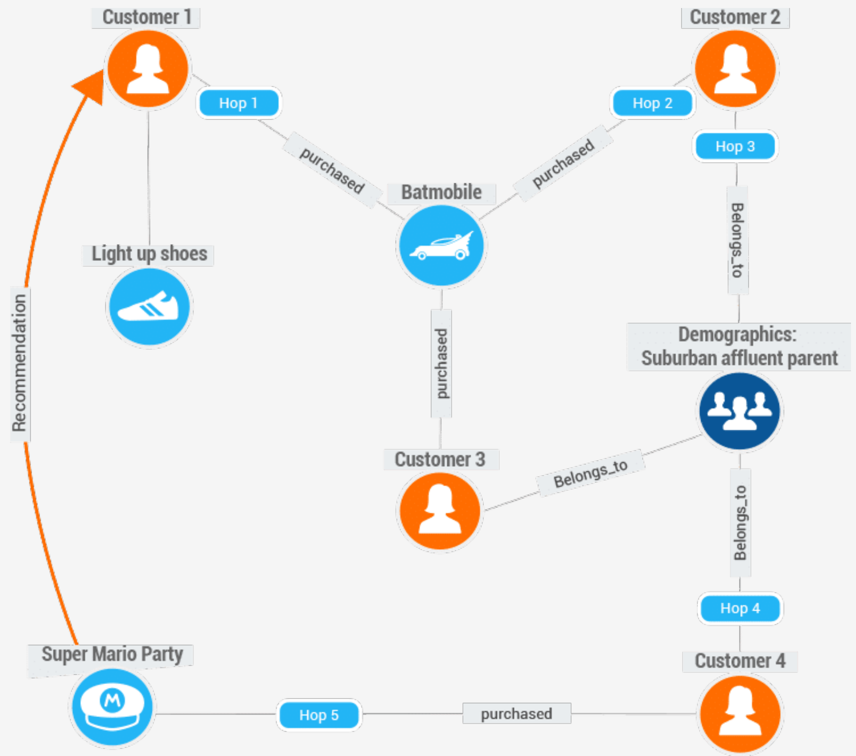
Yixin Wu,[1] *Xinlei He*,[2] *Pascal Berrang*,[3] *Mathias Humbert*,[4] *Michael Backes*,[1] *Neil Zhenqiang Gong*,[5] *Yang Zhang*[1]

[1] CISPA Helmholtz Center for Information Security

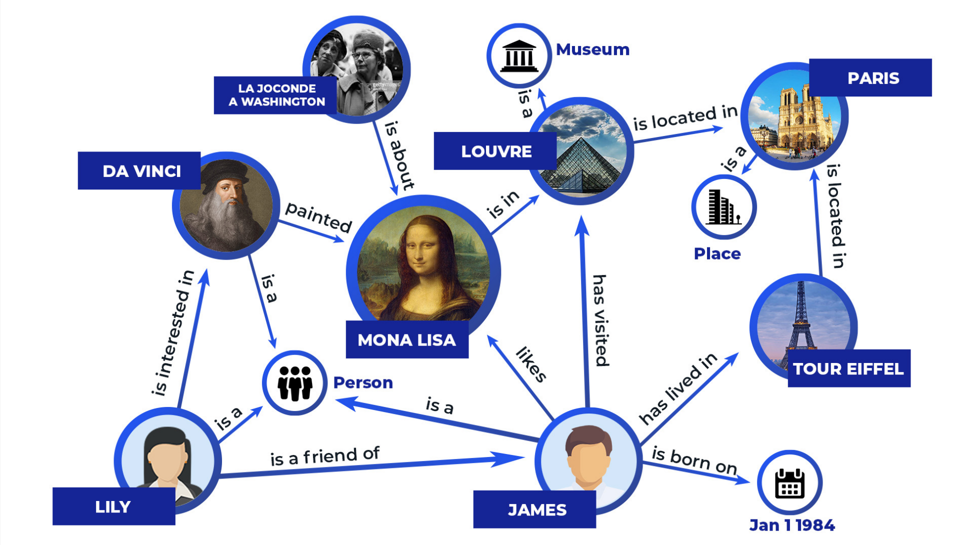
[2] Hong Kong University of Science and Technology

[3] University of Birmingham [4] University of Lausanne [5] Duke University

Prevalence of Graph-Structured Data



Recommendation System



Knowledge Graph

Social Network

Image source: https://www.tigergraph.com/solutions/recommendation-engine/;
https://yashuseth.wordpress.com/2019/10/08/introduction-question-answering-knowledge-graphs-kgqa/
https://medium.com/analytics-vidhya/social-network-analytics-f082f4e21b16

Graph Neural Networks (GNNs)

- Leverage neighbor information among nodes to learn embeddings to perform downstream tasks

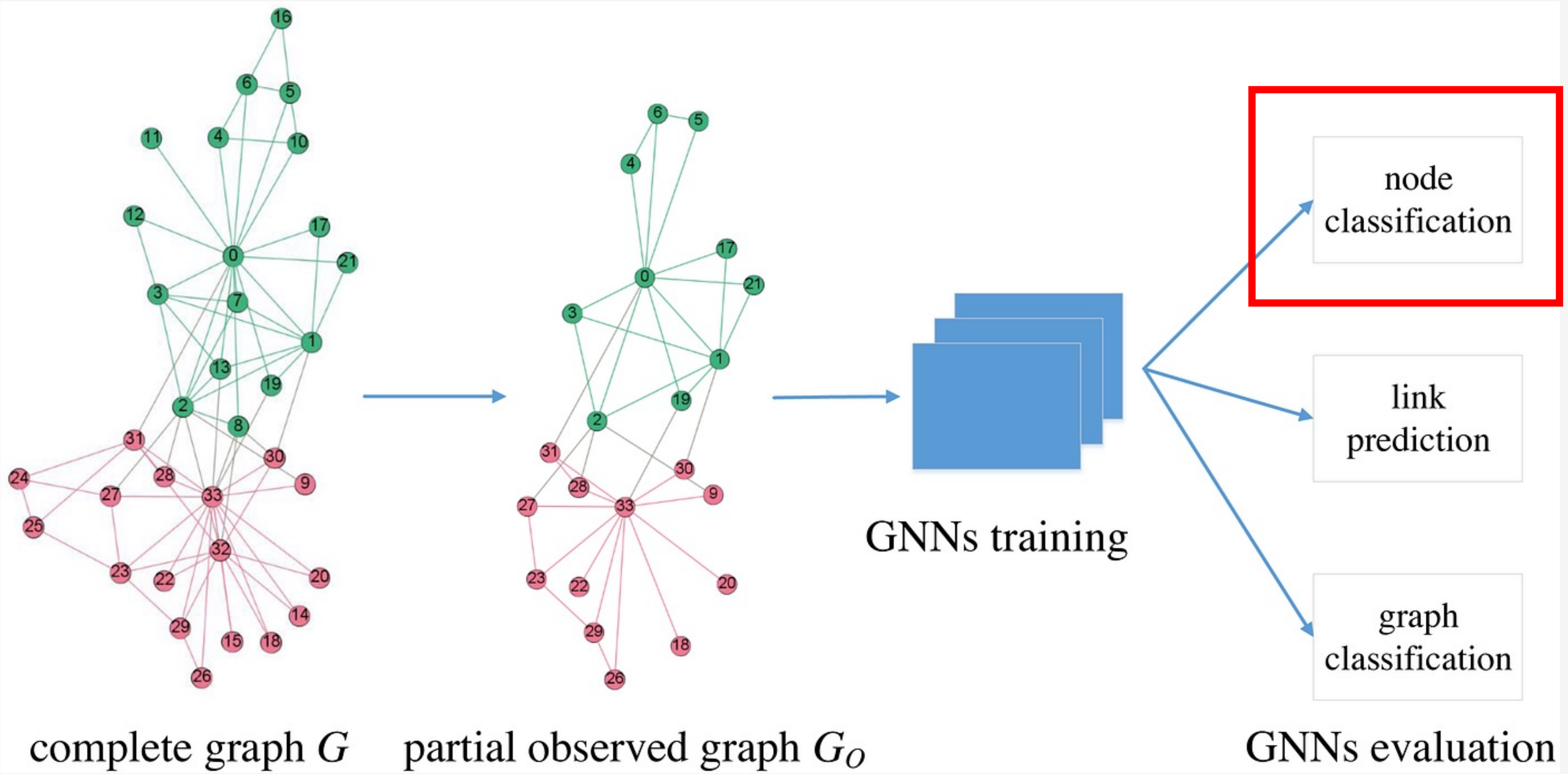
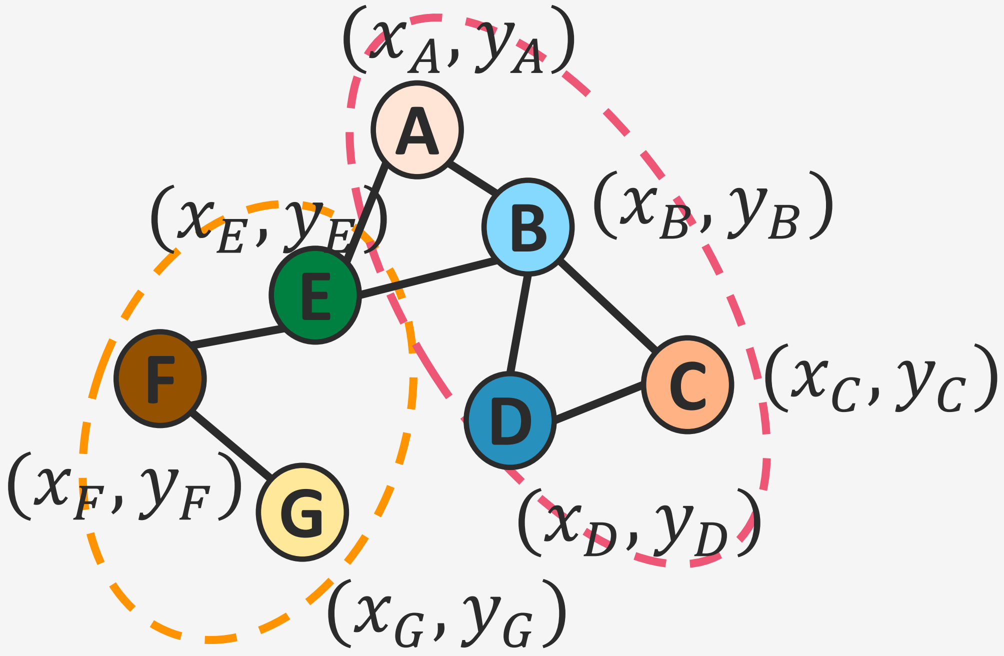


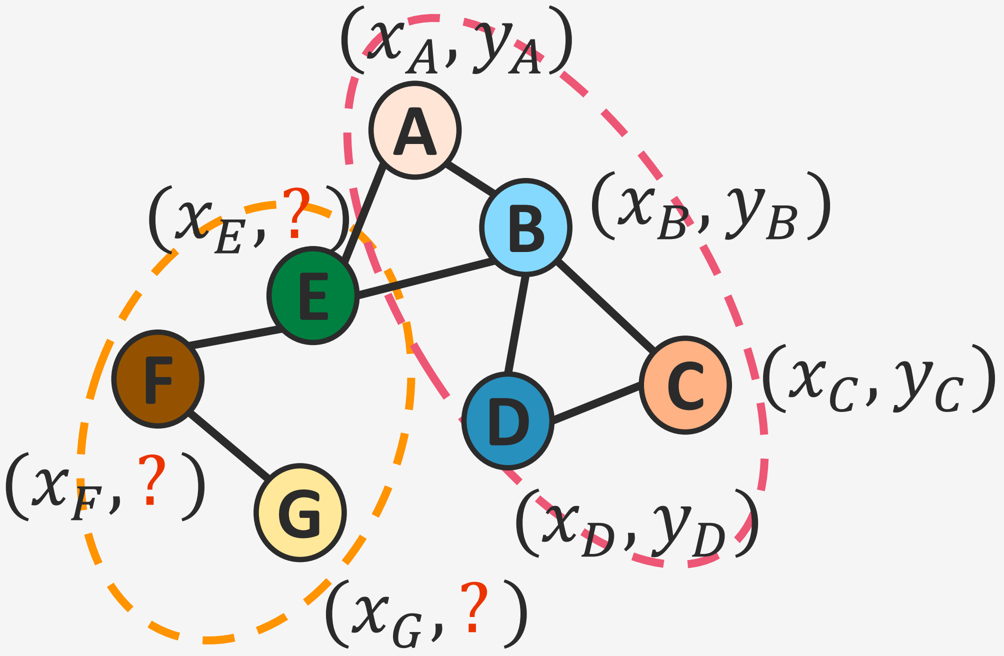
 Image source: https://medium.com/@shekhawatsamvardhan/types-of-neural-networks-graph-neural-networks-a82ae13638ac

Transductive Setting



Train/Test Split

Training



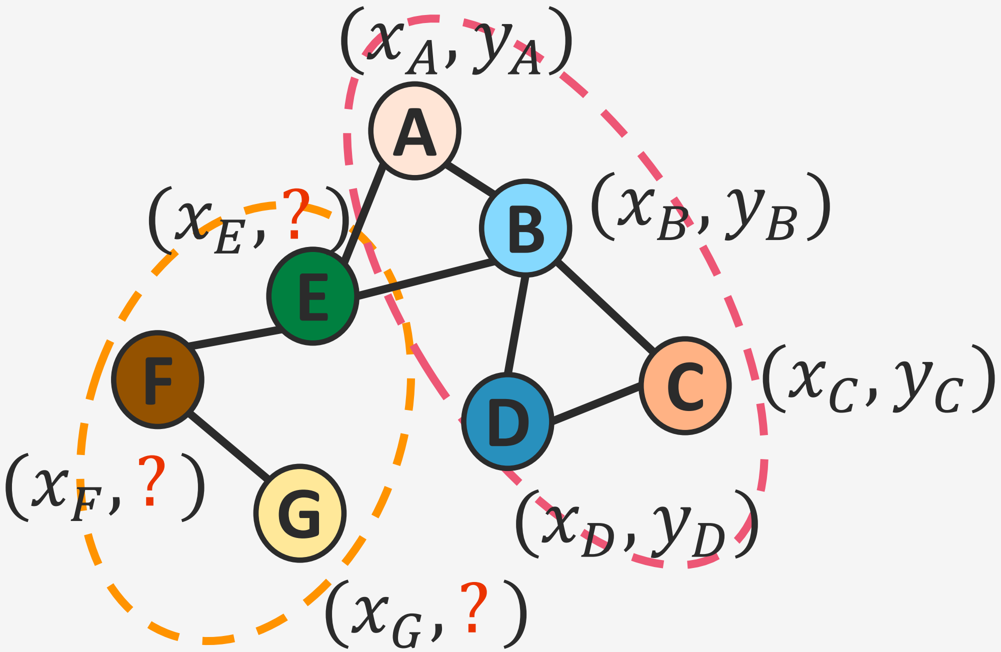
During training, **the entire graph** including node attributes and edges can been observed

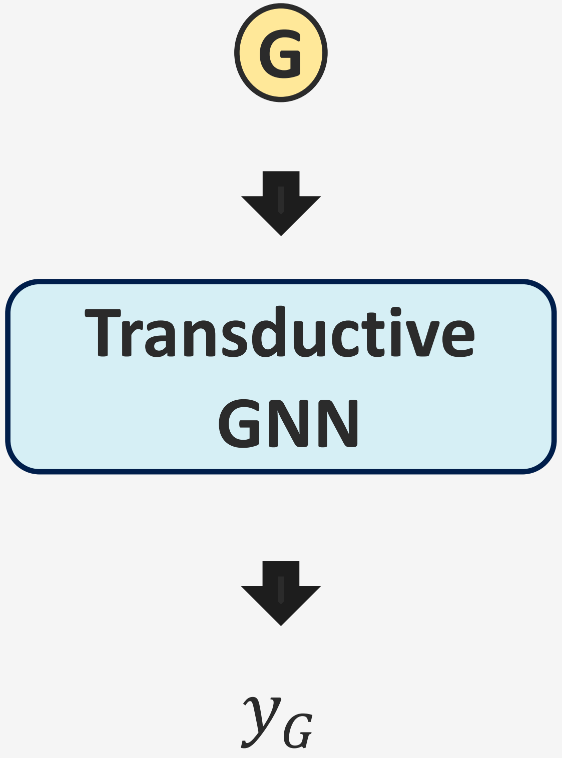
Transductive GNN

Transductive Setting

Testing

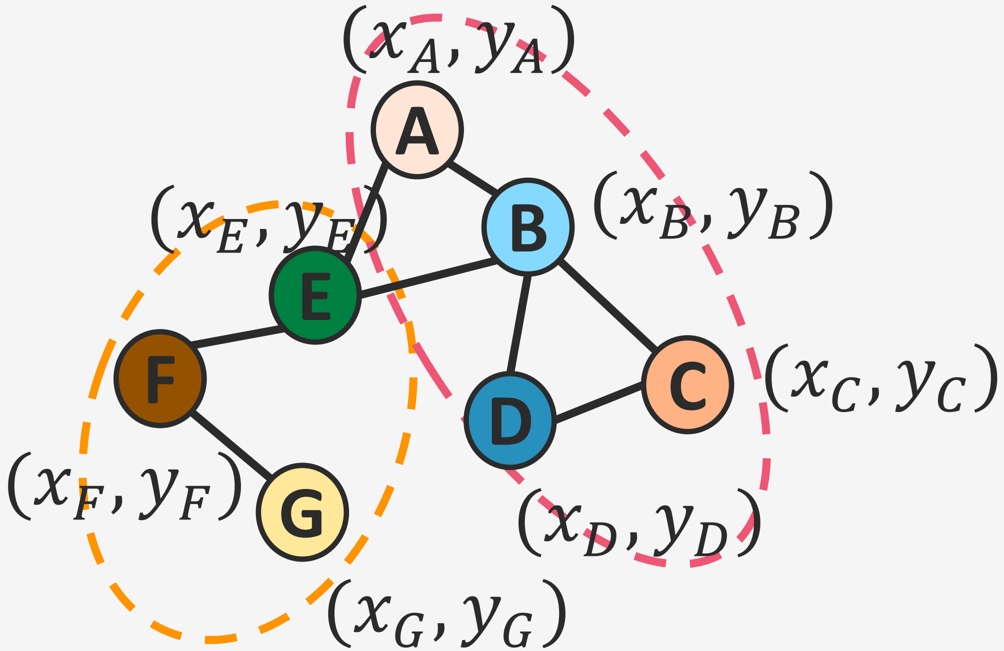
Graph Observed During Training





During inference, users feed the **identifiers of unlabeled seen nodes** into GNN to obtain prediction results

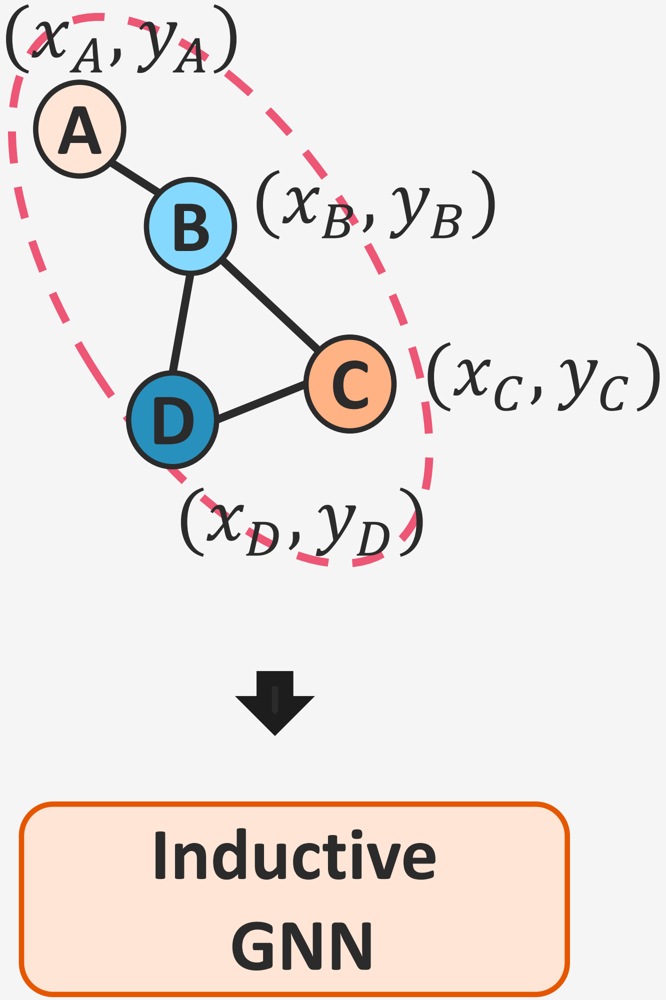
Inductive Setting



Train/Test Split

During training, **only the training graph** can be observed

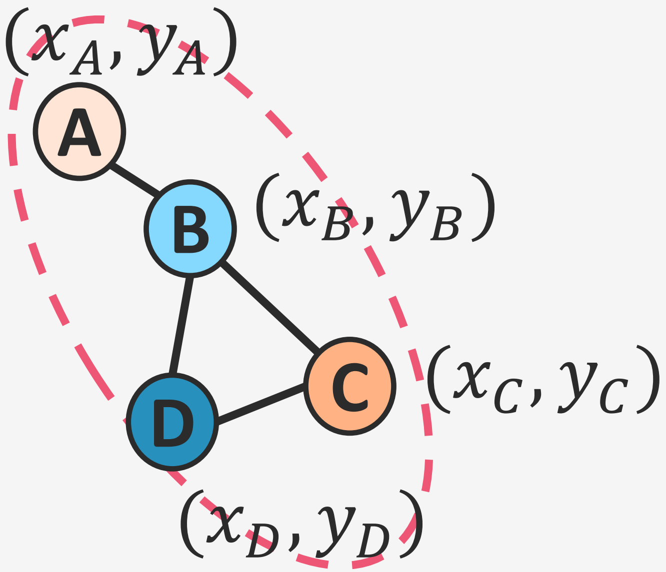
Training

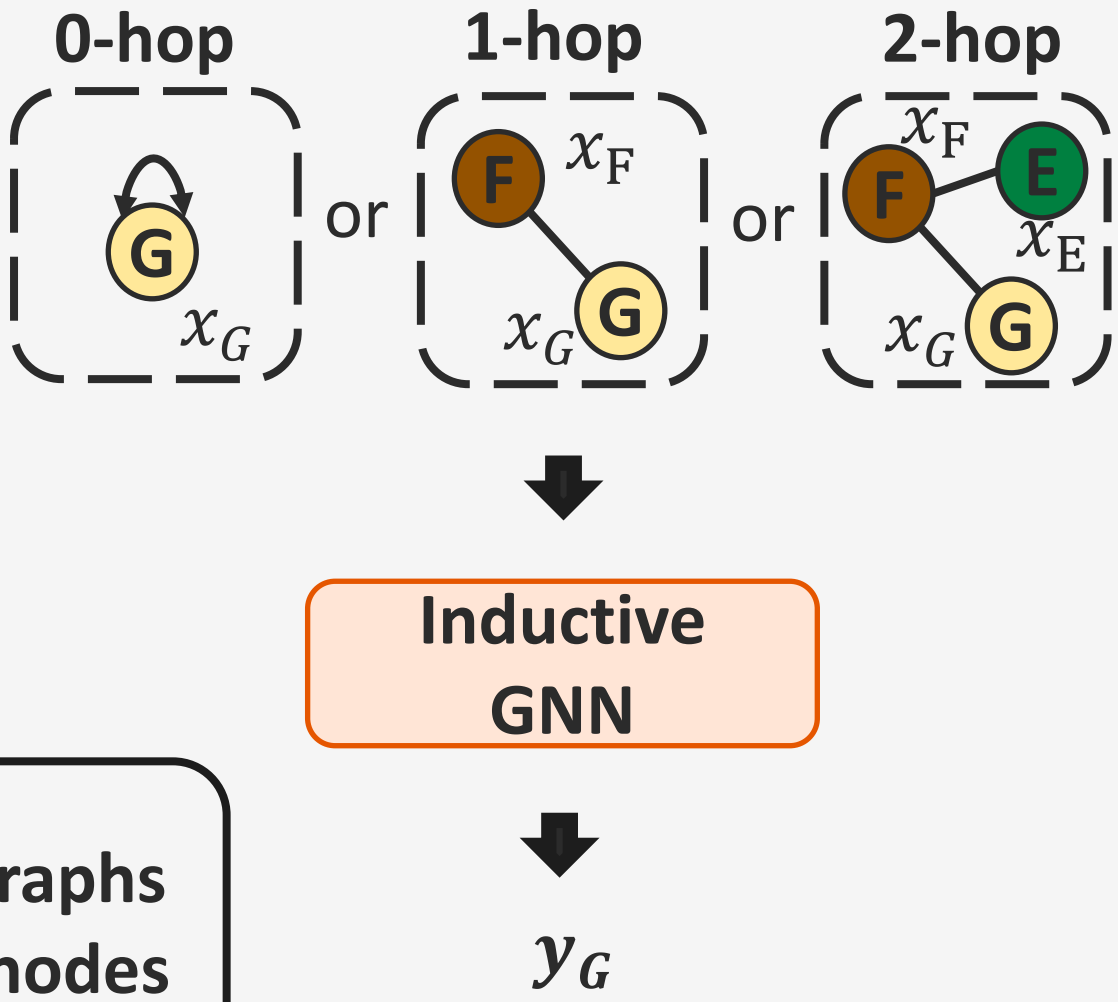


Inductive Setting

Testing

Graph Observed During Training





During inference, users **construct subgraphs** to obtain prediction results of **unseen nodes**

Difference Between Two Settings

Transductive GNN

“Memorize” the training graph

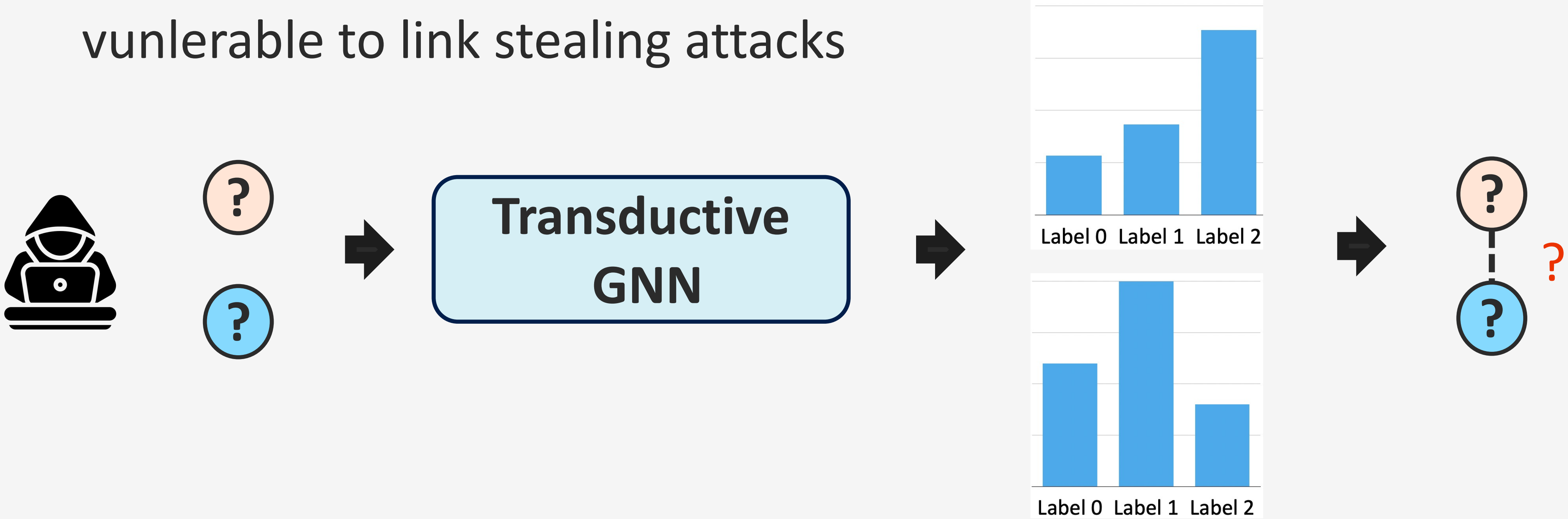
Inductive GNN

Learn a generalizable embedding function

Inductive GNNs are more generalizable and flexible for dynamic real-world applications, e.g., social network and recommendation system

Link Stealing Attacks on Transductive GNNs

- Previous work[1] demonstrates that the transductive GNNs are vunlerable to link stealing attacks



Given two nodes used to train a black-box GNN, can we predict whether they are linked?

[1] "Stealing Links from Graph Neural Networks," Xinlei He, Jinyuan Jia, Michael Backes, Neil Zhenqiang Gong, Yang Zhang; USENIX Security 2021

Challenges From The Differences

Transductive GNN

"Memorize" the training graph

Inductive GNN

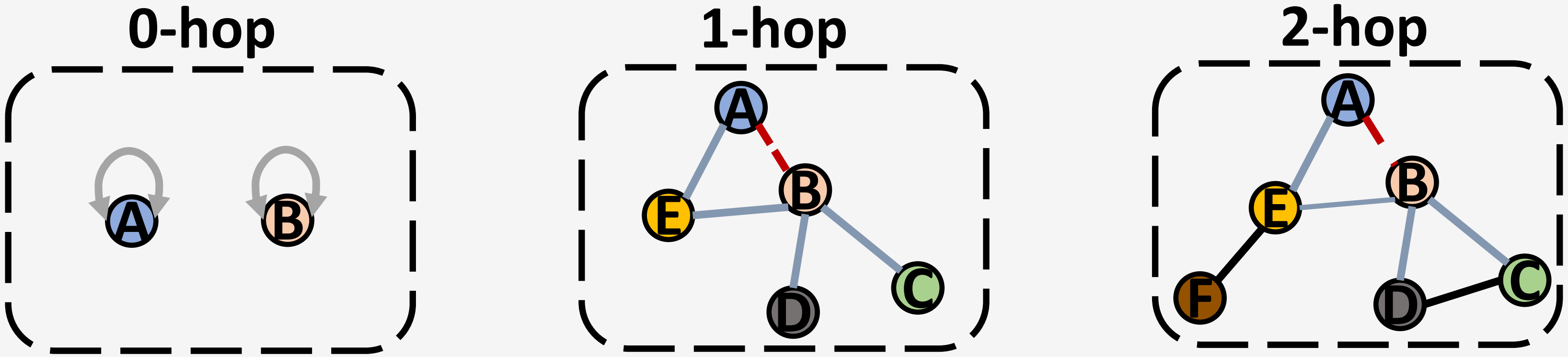
Learn a generalizable embedding function

Inductive GNNs might not learn specific information of training graph as transductive GNNs

Challenges From The Differences

Inductive GNN

Query with own constructed subgraphs



The adversary relies on limited and incomplete neighbor information, as the information of the link they intend to infer is missing

Are inductive GNNs vulnerable?

Transductive GNN

Query with the **identifiers of unlabeled nodes** and obtain fixed node embeddings learned during training

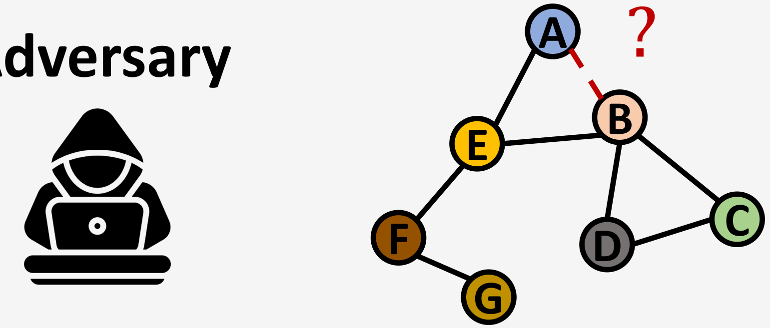
Inductive GNN

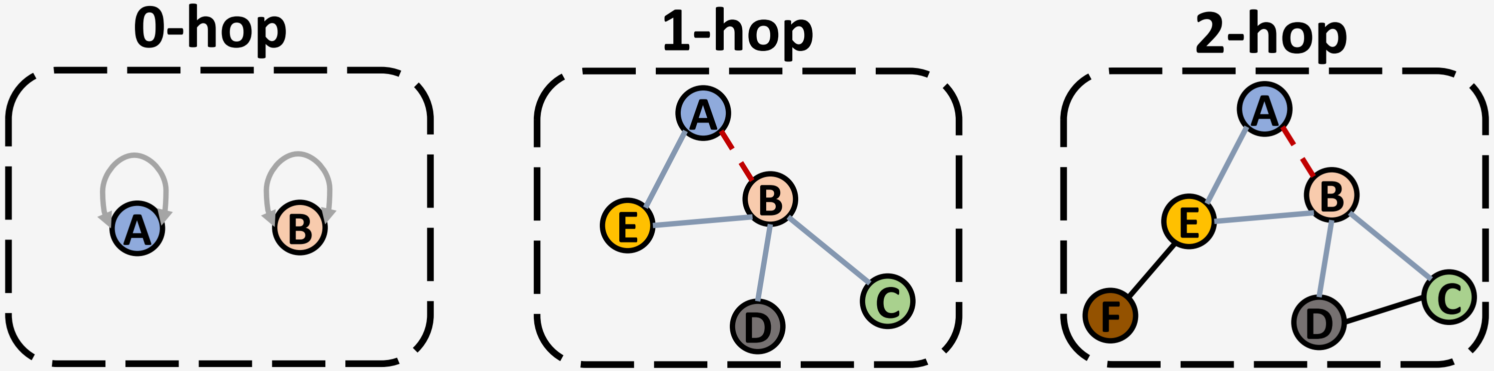
Query with own constructed subgraphs and obtain node embeddings based on the subgraphs

Given the above two challenges, are inductive GNNs vulnerable to link stealing attacks?

Link Stealing Attacks on Inductive GNNs

Adversary





Adversary can have either these three types of features

Node Features

A x_A
B x_B
→ Pairwise Operation

Posterior Features

GNN → Pairwise Operation

Graph Features

- Preferential Attachment
- Jaccard Coefficient
- Common Neighbors

Evaluation Results

Dataset	Posterior-Only Attack			Combined Attack						
	A0	A1	A2	A3	A4	A5	A6	A7	A8	A9
Cora	0.859	0.849	0.849	0.876	0.876	0.875	0.882	0.884	0.908	**0.909**
Pubmed	0.768	0.806	0.809	0.889	0.895	0.897	0.881	0.882	**0.939**	**0.939**
DBLP	0.781	0.821	0.822	0.834	0.873	0.872	0.879	0.903	0.924	**0.929**
Photo	0.877	0.898	0.898	0.892	0.916	0.915	0.967	**0.968**	0.946	0.946
CS	0.817	0.838	0.845	0.869	0.890	0.893	0.955	**0.956**	0.941	0.940
LastFM	0.850	0.869	0.867	0.883	0.909	0.911	0.919	0.921	0.929	**0.930**

- The proposed attacks with no (A0; 0-hop only) or limited (A1; 1-hop query) neighbor information can achieve good performance
- More information achieves better performance

Conclusion

- We propose in total 10 link stealing attacks against inductive GNNs
- No neighbor information (0-hop query) still enables well-performing link stealing attacks
- More information achieves better attack performance
- High robustness of the proposed attacks; better performance than traditional link prediction (baseline), showing inductive GNNs indeed leak privacy information

Contact

Thanks!

Yixin Wu

CISPA Helmholtz Center for Information Security

@yxoh28

https://yxoh.github.io/