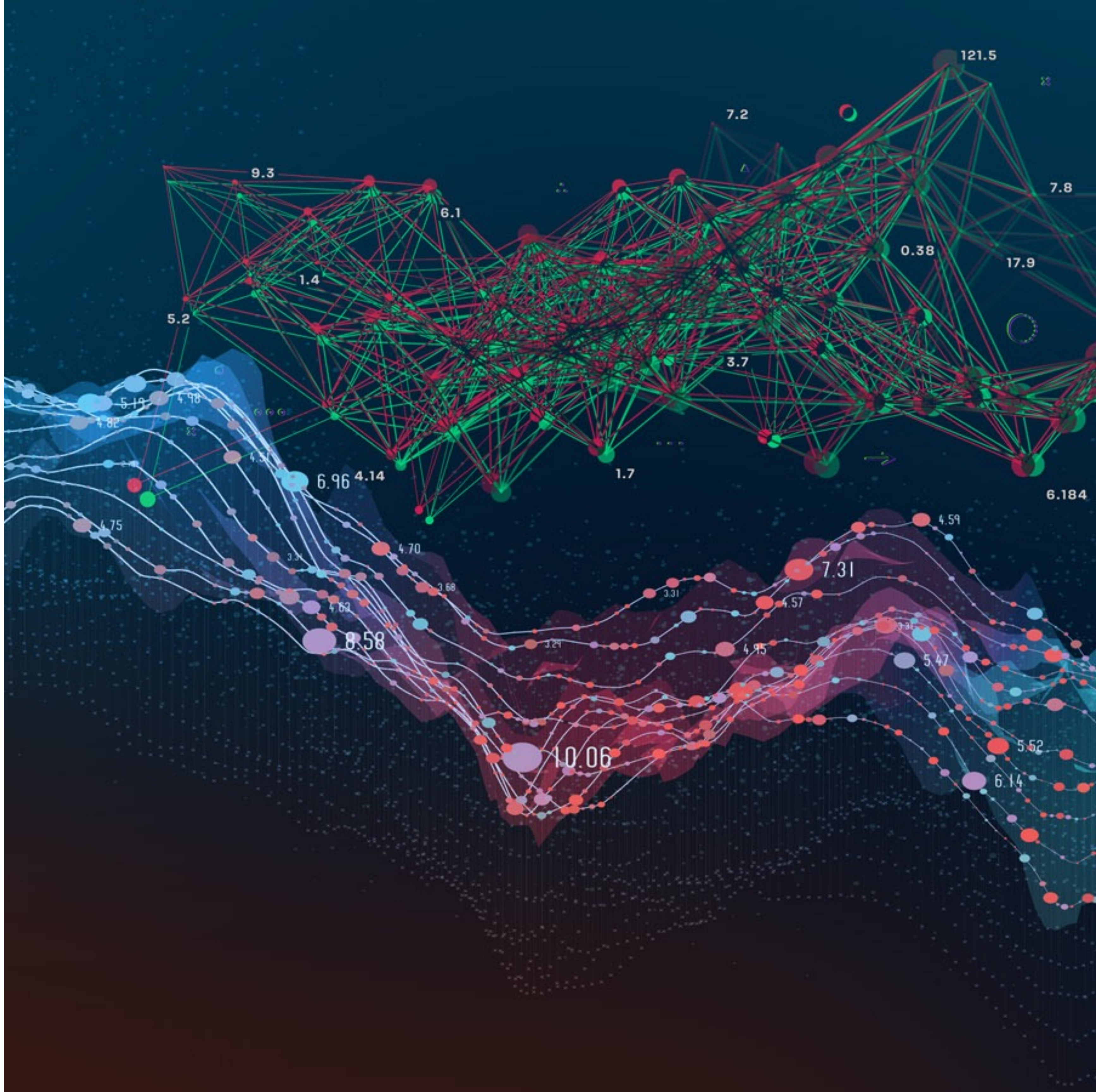
Self-Supervised Learning of Contextual Embeddings for Link Prediction in Heterogeneous Cyber Networks

Sutanay Choudhury

March 28, 2021

Advanced Computing Mathematics and Data Division

Work with Ping Wang (Virginia Tech), Khushbu Agarwal (PNNL), Colby Ham (PNNL), Chandan Reddy (Virginia Tech)



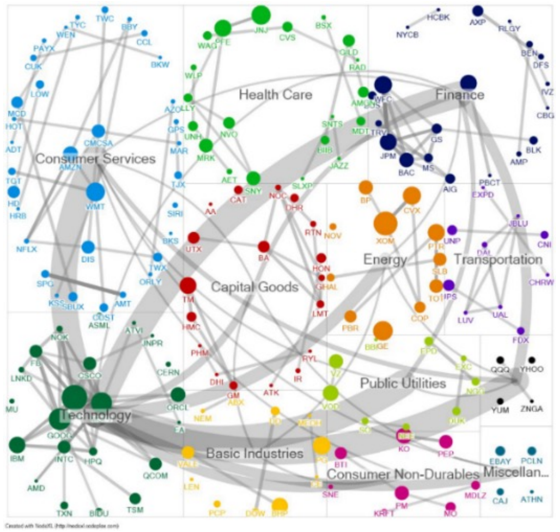
Key Insights from this Talk

- https://github.com/pnnl/SLICE

- Developing graph-based ML methods
  - Moving from a single embedding per node paradigm to contextual embedding learning
- If you are a cyber-security researcher/practitioner
  - Consider using self-supervised learning-based link prediction as a key method
  - **29% boost in F1-score for a 7-day intrusion detection dataset**
- If you are interested in accelerating graph-based ML:
  - What does it mean to interleave GNNs and Transformers?
  - How to scale up context generation?

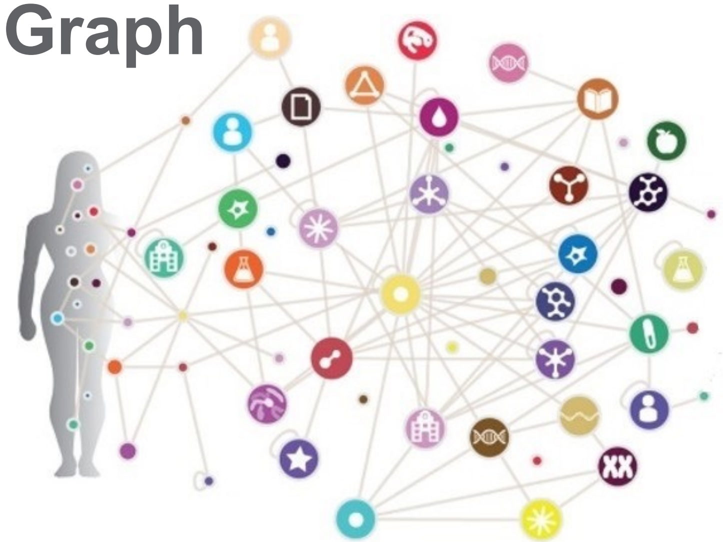
Introduction

Co-authorship Network



Social Network

Clinical Knowledge Graph



- **Heterogeneous networks:**
  - Integrate different data sources, build relations between them.
  - Allows us to further discover underlying correlations with link prediction task.
- **Existing link prediction methods:**
  - Provide a **static** embedding for each entity that is agnostic to any specific context.
  - Without considering **contextual information** of the downstream task.

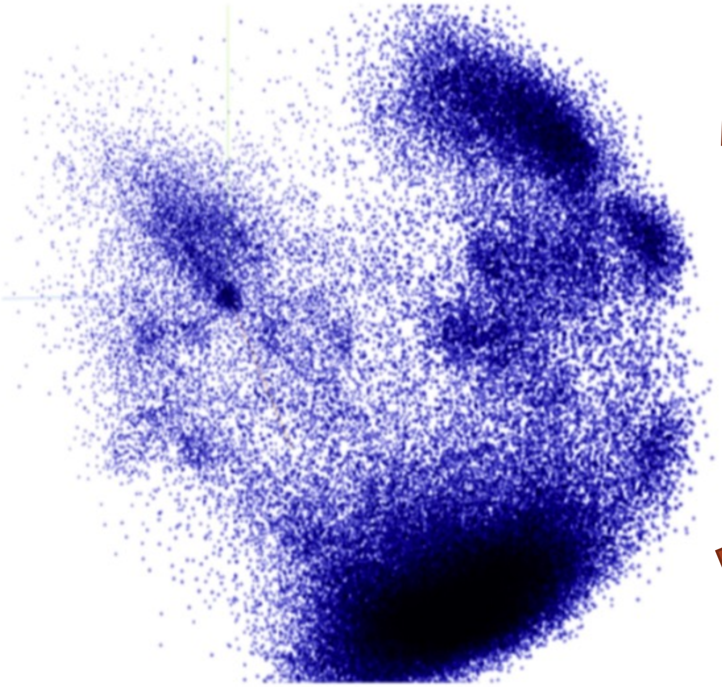
Image source: https://www.biorxiv.org/content/10.1101/2020.05.09.084897v1; https://www.smrfoundation.org/2013/12/11/university-of-maryland-computer-science-class-cmsc734-student-projects-put-nodexl-to-work-finding-insights-in-diverse-networks/; https://www.wired.com/2012/04/facebook-disease-friends/

Pacific Northwest
NATIONAL LABORATORY

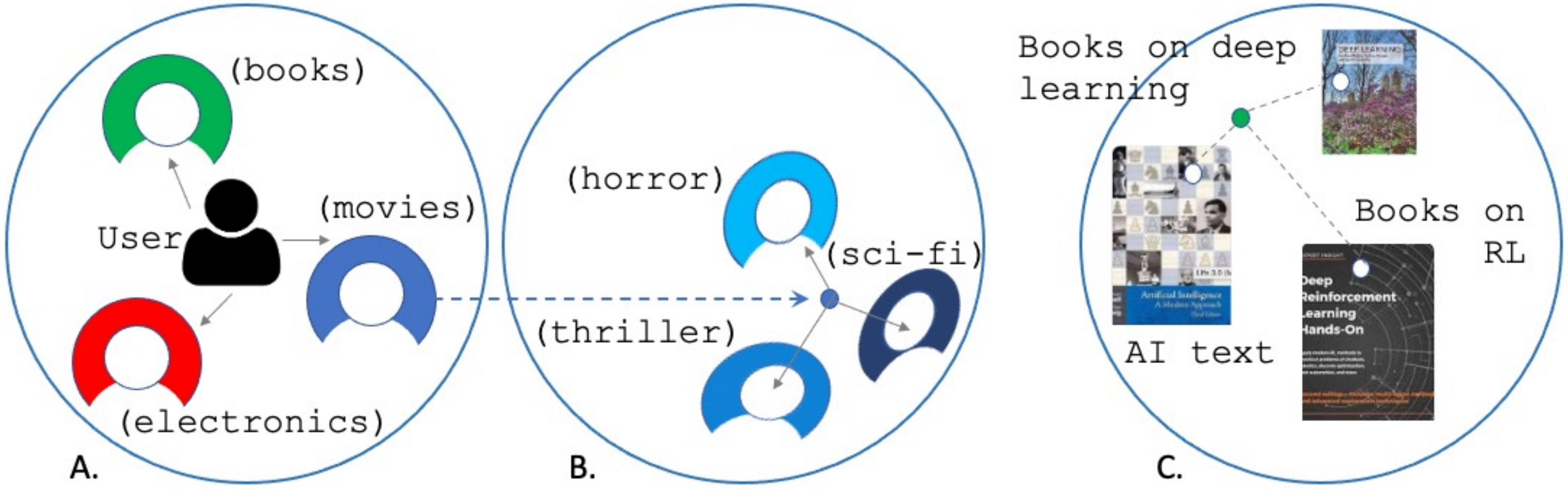
The Importance of Embedding Learning

Embedding learning is a critical step for graph-based machine-learning

Each node in a graph is mapped to a point in vector space



Is a Single Embedding Enough?

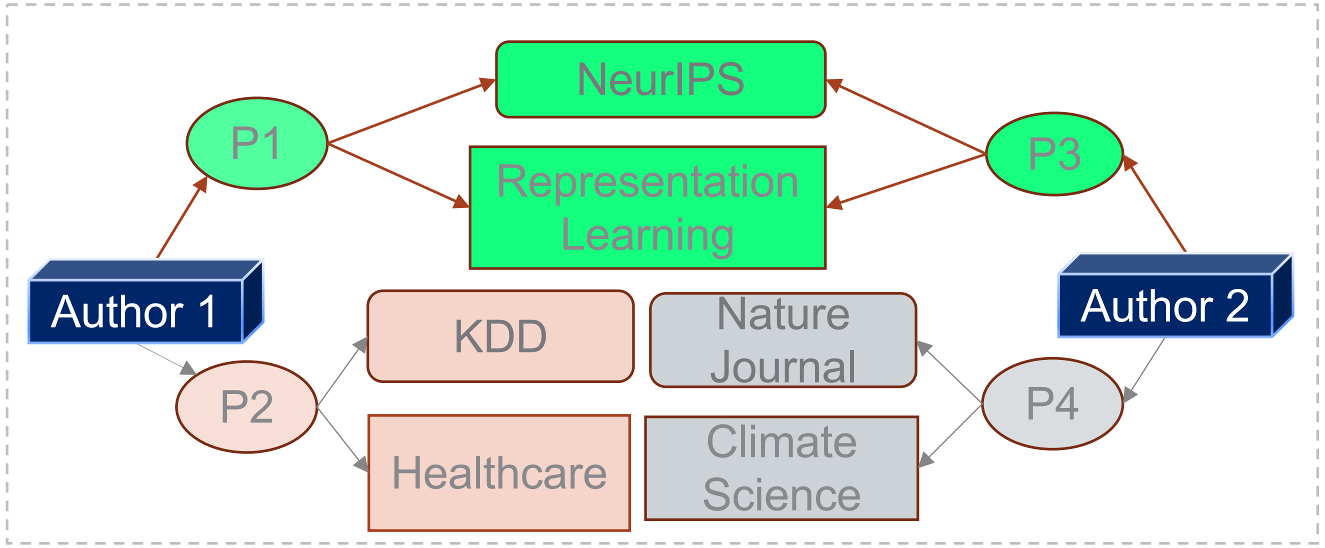


Entities exhibit diverse behavior in heterogeneous networks that are reflected via their diverse associations

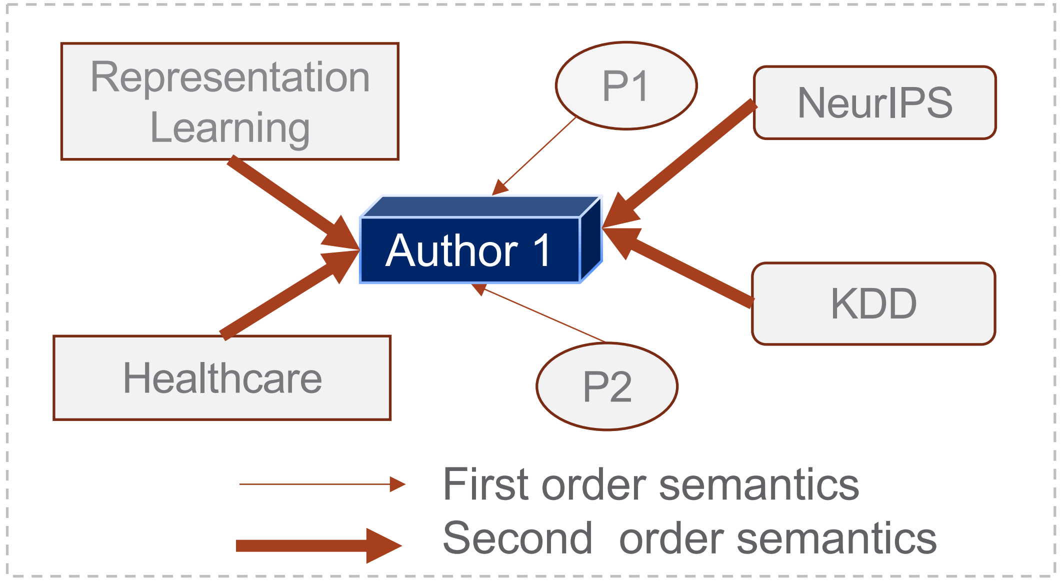
Can we do better by recognizing this heterogeneity and eliminating any bias they introduce?

Example Motivation from an Academic Network

Academic Network with authors publish on diverse topics



State-of-the-art methods aggregate global semantics for authors based on all papers

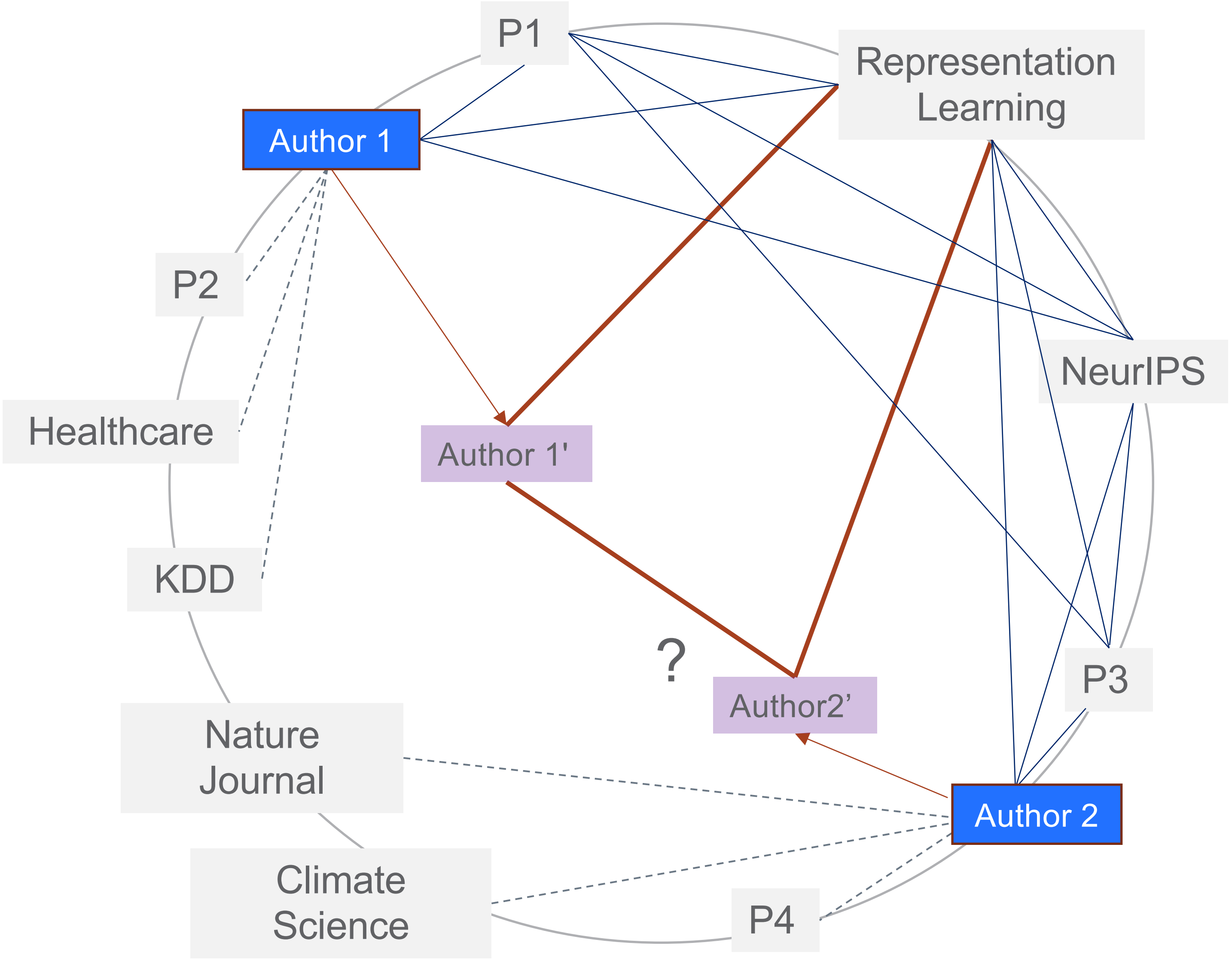


We propose Contextual Embedding Learning

Start with a **global representation, and move** the **embedding** of a set of nodes in vector space based on a **task context**

Recent work have focused on clustering-based approaches that assigns multiple embeddings corresponding to clusters or communities (see ”related work” in [1])

Such methods are limited by the need to pre-assign fixed size clusters to all, as well by the complexity of clustering heterogeneous networks

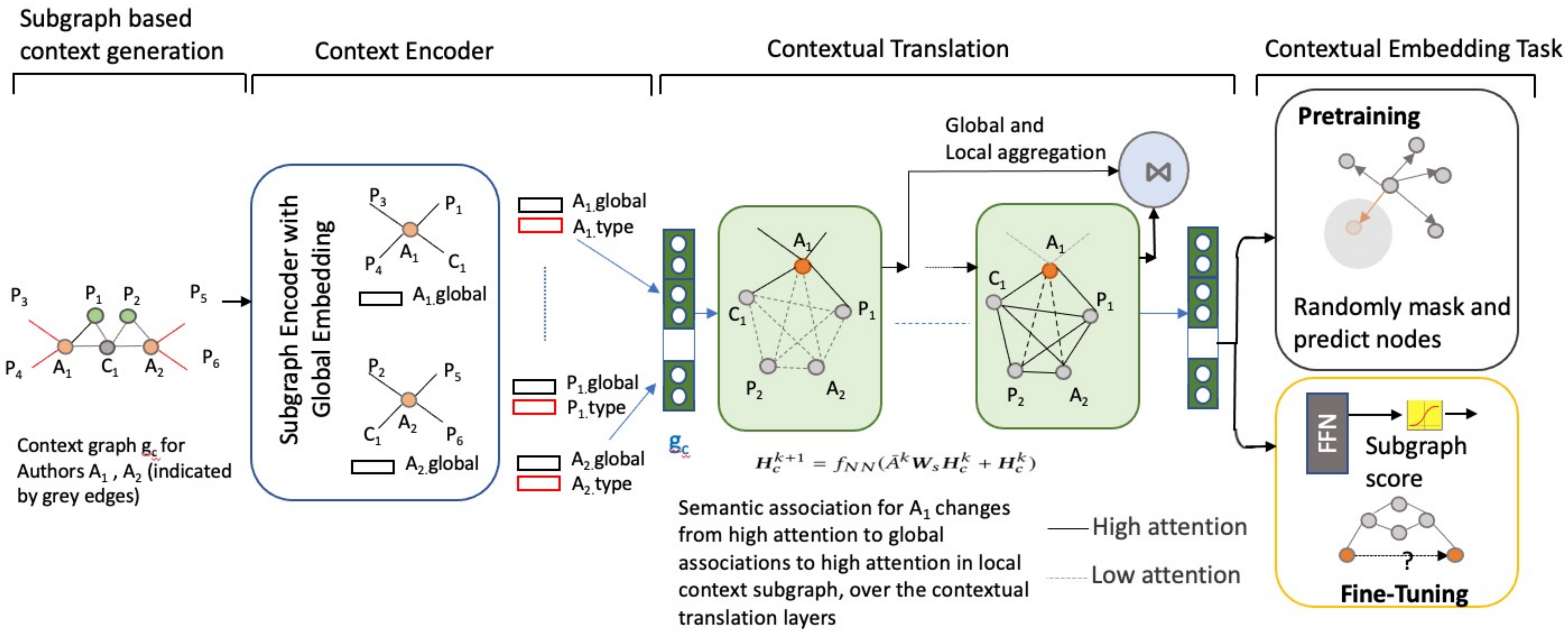


1. Wang, P., Agarwal, K., Ham, C., Choudhury, S. and Reddy, C.K., 2020. Self-Supervised Learning of Contextual Embeddings for Link Prediction in Heterogeneous Networks. *WebConf 2021*.

Our Contributions

Define Contextual Subgraphs	• Contextual embeddings are learnt based on task-specific subgraphs. • Node representations will be dynamically changing with different subgraphs.
Self-supervised Learning Approach	• Learn higher-order semantic associations by simultaneously capturing the global information and local context. • Two training stages: pre-training and fine-turning.
Performance Evaluation	• Compare with static and contextual embedding learning methods. • Demonstrate the interpretability, effectiveness of contextual translation.

Neural Architecture



Context Generation and Representation

- **Context Generation:** generate context for each node or a node-pair.
  - Shortest Path: consider the shortest path between two nodes.
  - Random strategy: BFS based star graph; random walks with a certain depth.
- **Context Representation:**
  - Subgraph g_c is encoded as $g_c = \left(v_1, v_2, \ldots, v_{|V_c|}\right)$. Here, $|V_c|$ is the number of nodes in g_c.
  - Global embeddings of nodes in g_c are represented as $H_c = \left(h_1, h_2, \ldots, h_{|V_c|}\right)$. Where, h_i is the low-dimensional representation of node i that considers various information in the global graph, such as the structures and attributes.
  - We mainly consider the pre-trained node embeddings from node2vec, which is a random walk-based skip-gram methods.

Contextual Translation

- Semantic association matrix $\bar{A}$:
    - Given two nodes v_i and v_j in the context, the corresponding entry $\bar{A}_{ij}^k$ can be computed as follows.

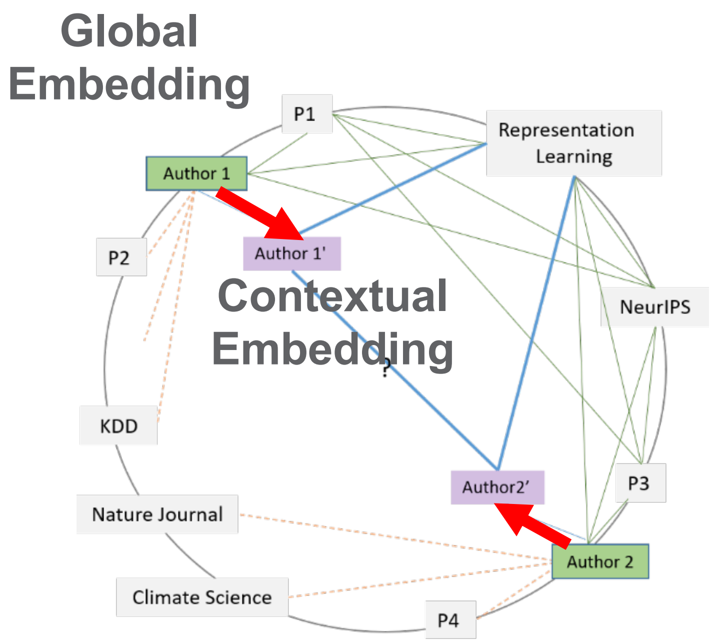
$$\bar{A}_{ij}^k = \frac{\exp\left(\left(W_1 h_i^k\right)^T \left(W_2 h_j^k\right)\right)}{\sum_{t=1}^{|V_C|} \exp\left(\left(W_1 h_i^k\right)^T \left(W_2 h_t^k\right)\right)}$$

- Contextual Translation: Apply multiple translation layers; in $k+1$ layer, $\bar{A}^k$ is updated as follows:

$$H_c^{k+1} = f_{NN}(W_s H_c^k \bar{A}^k + H_c^k)$$

- The node embeddings from different layers ($K\ in\ total$) are aggregated as the contextual embedding:

$$\tilde{h}_i = h_i^1 \oplus h_i^2 \oplus \cdots \oplus h_i^K$$



Contextual Translation
- **Maintain global relations**
- **Learn local context**

Contextual Learning Tasks

- **Self-supervised Contextual Node Prediction in Pre-training:**
  - Generate context subgraphs for each node in the network via random walks and randomly mask nodes in each subgraph for prediction.
  - **Objective**: maximize the probability of observing the masked node based on the context.
- **Fine-tuning with Supervised Contextual Link Prediction:**
  - Generate context subgraphs for each node-pair and perform the binary link prediction.
  - **Objective**: maximizing the prediction score of positive edges and minimizing the score for negative edges.
  - The probability of the edge between two nodes is calculated as the similarity score between their contextual embeddings.

Experiments

Datasets used:

- Amazon (E-commerce): co-viewing and co-purchasing links between products.
- DBLP (Academic): relationships between papers, authors, venues and terms.
- Freebase (Knowledge Base): links between people and their demographic features.
- Twitter (Social Networks): links between tweets users.
- Healthcare[1] (MIMIC III): relations between patients and their diagnosed medical conditions, procedures and medications received during each hospital admission.

Dataset	Amazon	DBLP	Freebase	Twitter	Healthcare
# Nodes	10,099	37,791	14,541	9,990	4,683
# Edges	129,811	170,794	248,611	294,330	205,428
# Relations	2	3	237	4	4
# Training (positive)	126,535	119,554	272,115	282,115	164,816
# Development	14,756	51,242	35,070	32,926	40,612
# Testing	29,492	51,238	40,932	65,838	40,612

1. Codes for generating the Healthcare network based on MIMIC III is available at https://github.com/pnnl/SLICE

SLiCE Outperforms Most Recent Methods

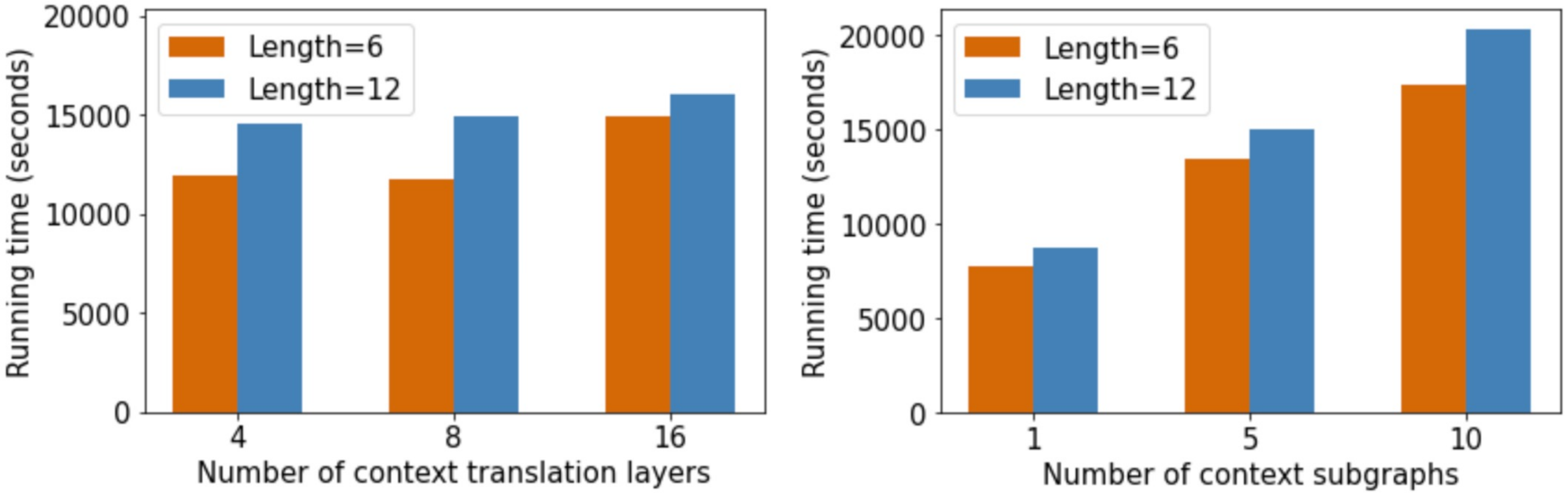
- Experiments performed on NVIDIA Tesla P100 GPU

Type	Methods	micro-F1 score					AUCROC				
		Amazon	DBLP	Freebase	Twitter	Healthcare	Amazon	DBLP	Freebase	Twitter	Healthcare
Static	TransE	50.28	49.60	47.78	50.60	48.42	50.53	49.05	48.18	50.26	49.80
	RefE	51.86	49.60	50.25	48.55	47.96	51.74	48.50	50.41	49.28	50.73
	node2vec	88.06	86.71	83.69	72.72	71.92	94.48	93.87	89.77	80.48	79.42
	metapath2vec	88.86	44.58	77.18	66.73	62.64	95.42	38.41	84.33	72.16	69.11
Contextual	GAN	85.47	OOM	OOM	85.01	81.94	92.86	OOM	OOM	92.39	89.72
	GATNE-T	89.06	57.04	OOM	68.16	58.02	94.74	58.44	OOM	72.07	73.40
	RGCN	65.03	28.84	OOM	63.46	56.73	74.77	50.35	OOM	64.35	46.15
	CompGCN	83.42	40.10	65.39	40.75	39.84	90.14	34.04	72.01	39.86	38.03
	HGT	65.77	53.32	OOM	53.13	76.54	68.66	50.85	OOM	59.32	82.36
	asp2vec	94.89	78.82	90.02	88.29	85.46	98.51	92.51	**96.61**	95.00	92.97
	SLiCE$_{w/o\ GF}$	67.01	66.02	66.31	67.07	60.88	62.87	57.52	55.31	66.69	63.11
	SLiCE$_{w/o\ FT}$	94.99	89.34	90.01	82.19	81.58	98.66	96.07	96.33	90.38	89.51
	SLiCE (Ours)	**96.00***	**90.70***	**90.26**	**89.30***	**91.64***	**99.02***	**96.69***	96.41	**95.73***	**94.94***

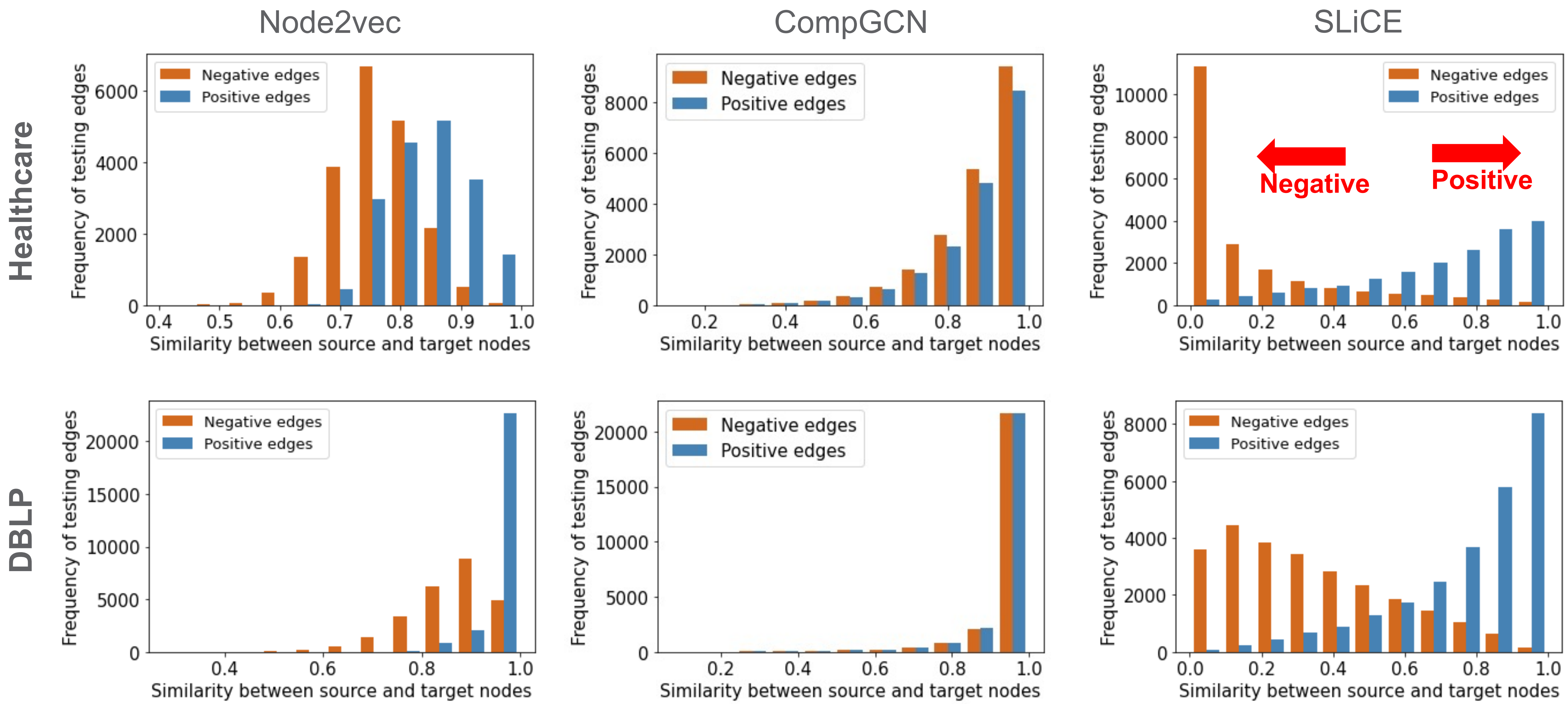
- The symbol "OOM" indicates out of memory.
- The symbol * indicates that the improvement is statistically significant over the best baseline by two-sided t-test with p-value 10^{-10}.

Computational Complexity

- Doubling the context length does not raise run time proportionately
- Approximately linear to the number of edges (coverage of graph matters more)



The Effect of Contextualization



Similar results are obtained on Amazon, Freebase and Twitter datasets

Motivation for Link Prediction in Cyber Security

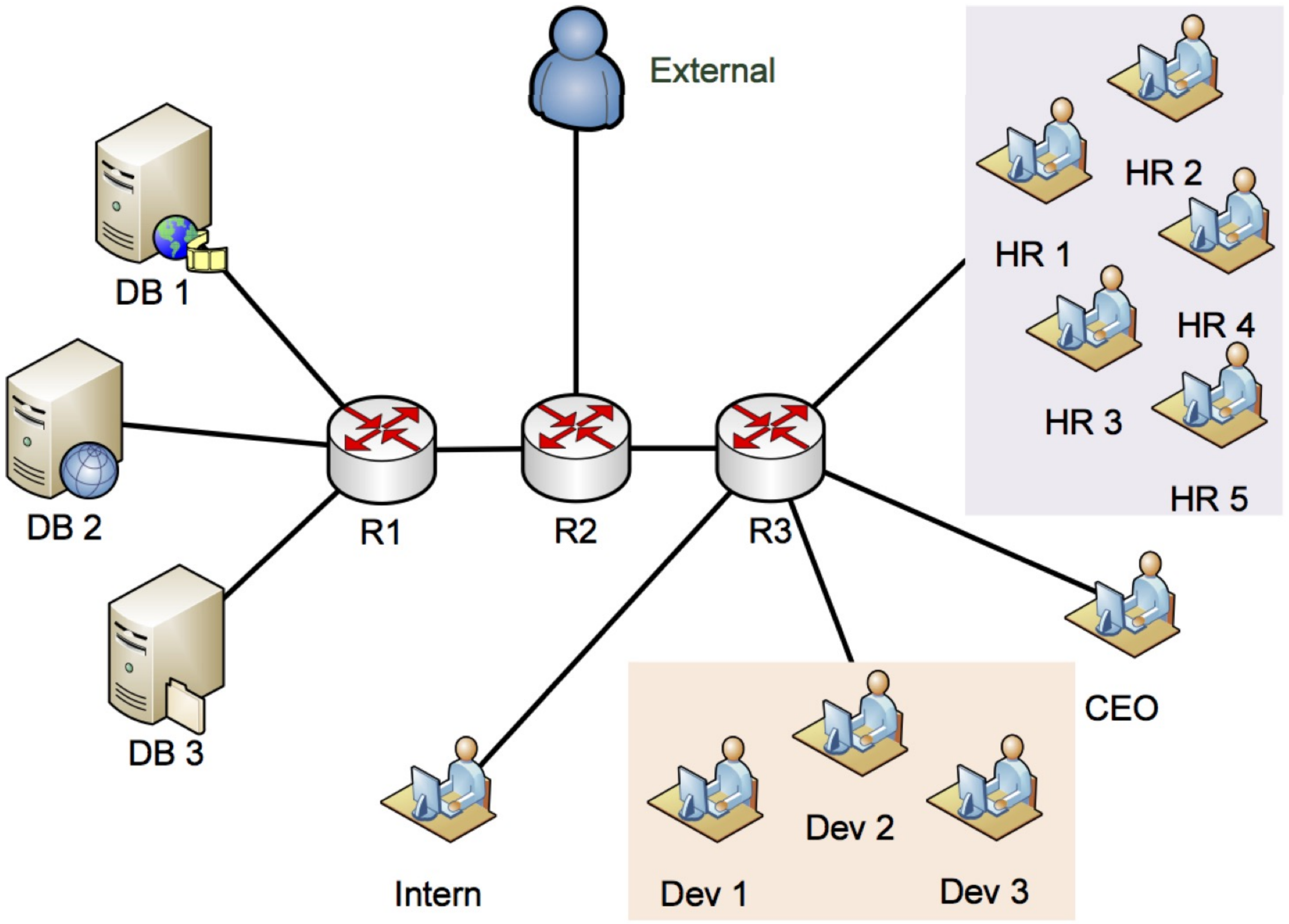
Need

We see some periodic communication between Dev1 and CEO's machine

Can we explain why this is anomalous?

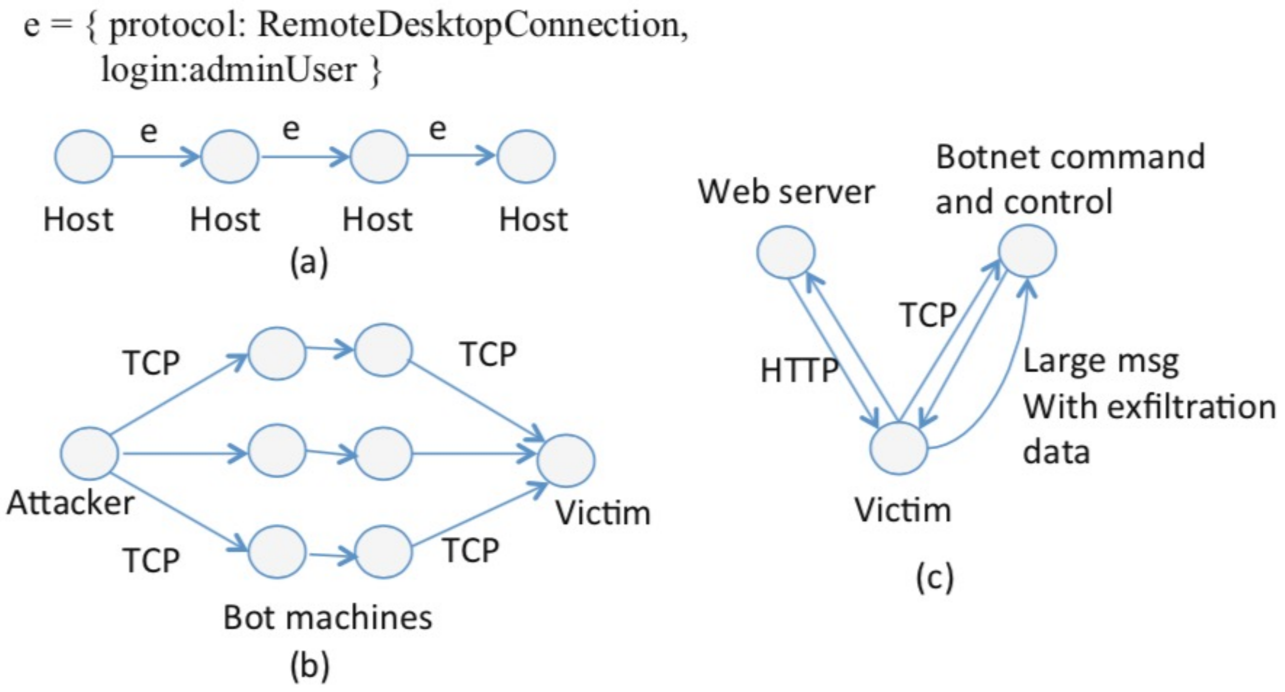
Underneath Context Graph

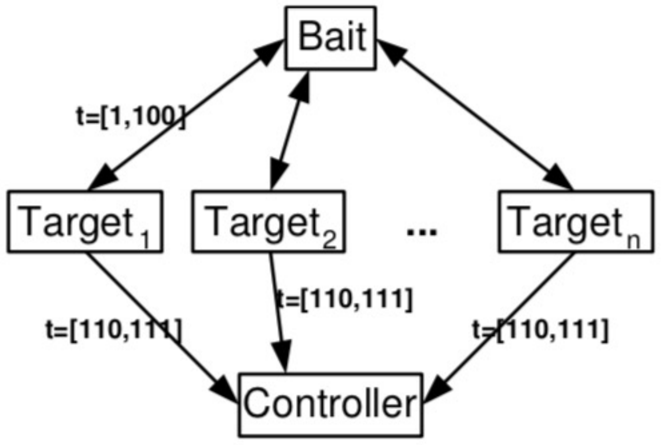
Dev1 frequent connects to service DB2. CE0's machine frequently connects to DB2.

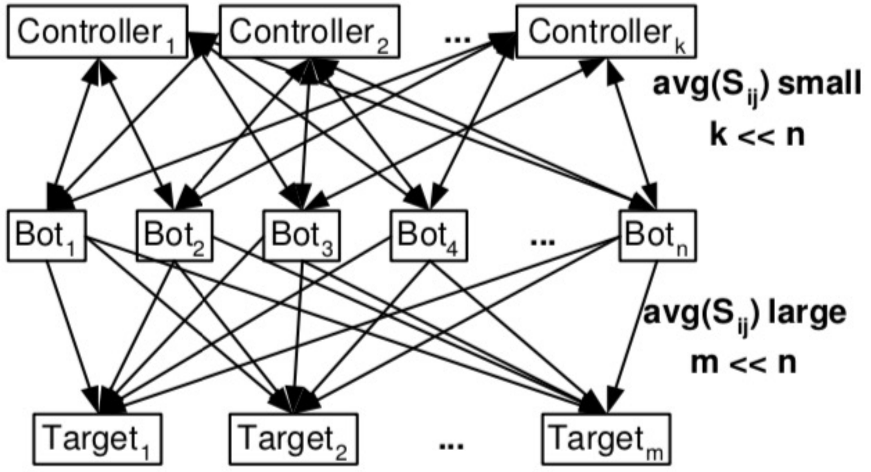


Subgraph Patterns for Attack Detection

- Well studied and motivated in the literature
- Determine if two machines are connected in attacker-victim, controller-target relationship





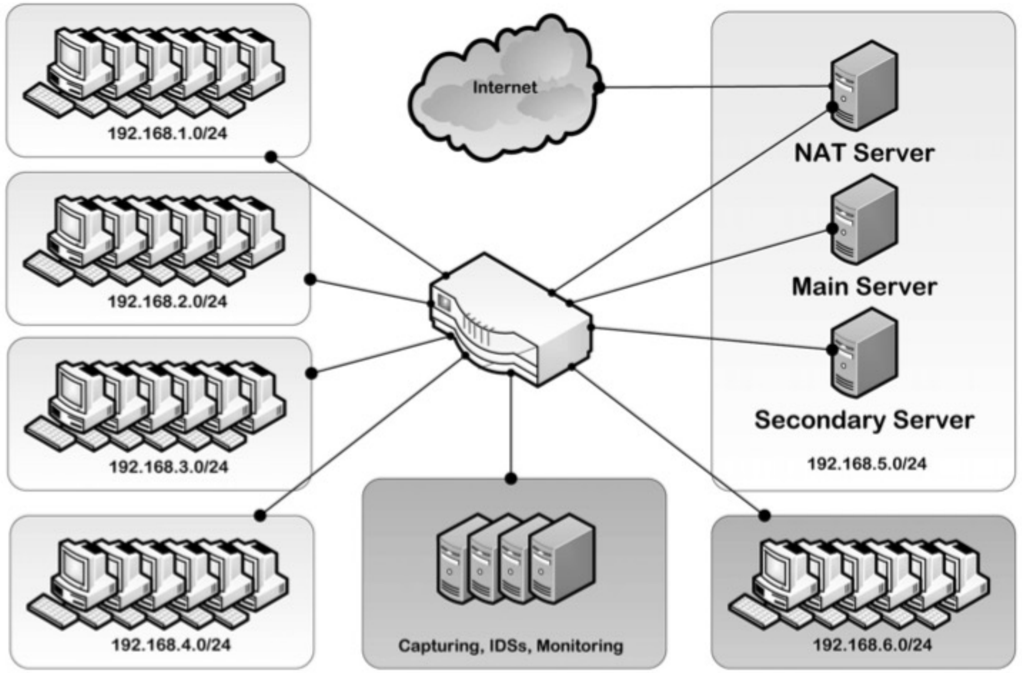


1. Joslyn, C., Choudhury, S., Haglin, D., Howe, B., Nickless, B. and Olsen, B., 2013, June. Massive scale cyber traffic analysis: a driver for graph database research. In *First International Workshop on Graph Data Management Experiences and Systems* (pp. 1-6).

2. Choudhury, S., Holder, L., Chin, G., Agarwal, K. and Feo, J., 2015. A selectivity-based approach to continuous pattern detection in streaming graphs. EDBT

Example Case Study from Intrusion Detection

- Intrusion Detection dataset from University of New Brunswick [1]
- We build a graph representation from the network traffic data [2]



Dataset	# nodes	# edges	Description
Day 1	5357	12887	Normal activity
Day 2	2631	5614	Normal activity
Day 3	3052	5406	Infiltrating attack and normal activity
Day 4	8221	12594	HTTP denial of service attack and normal activity
Day 5	24062	32848	Distributed denial of service attack using Botnet
Day 6	5638	13958	Normal activity
Day 7	4738	11492	Brute force SSH attack and normal activity

1. A. Shiravi, H. Shiravi, M. Tavallaee, and A. A. Ghorbani, “Toward developing a systematic approach to generate benchmark datasets for intrusion detection,” *Computers & Security*, vol. 31, 2012.

2. Chen, P.Y., Choudhury, S. and Hero, A.O., 2016, March. Multi-centrality graph spectral decompositions and their application to cyber intrusion detection. In 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (pp. 4553-4557). IEEE.

Significant Performance Boost from SLiCE

- We split each day's graph into training, validation and test partition

Dataset	# nodes	# edges	Description
Day 1	5357	12887	Normal activity
Day 2	2631	5614	Normal activity
Day 3	3052	5406	Infiltrating attack and normal activity
Day 4	8221	12594	HTTP denial of service attack and normal activity
Day 5	24062	32848	Distributed denial of service attack using Botnet
Day 6	5638	13958	Normal activity
Day 7	4738	11492	Brute force SSH attack and normal activity

	Node2Vec			SLiCE				
	ROCAUC	F1	AUC	ROCAUC	F1	Gain (%)	AUC	Gain (%)
Day 1	0.799	0.7116	0.7524	0.9643	0.8988	26.30691	0.9579	27.3126
Day 2	0.8518	0.7766	0.8104	0.948	0.8696	11.97528	0.9378	15.72063
Day 3	0.8479	0.7728	0.8248	0.9499	0.8726	12.91408	0.9431	14.34287
Day 4	0.8016	0.6815	0.7961	0.9677	0.9169	34.54145	0.9723	22.1329
Day 5	0.7327	0.6202	0.7494	0.9811	0.9604	54.85327	0.9871	31.71871
Day 6	0.7888	0.7032	0.776	0.9667	0.9193	30.73094	0.9581	23.46649
Day 7	0.785	0.6974	0.7737	0.9661	0.9216	32.14798	0.9558	23.53625
					Average Gain (F1)		Average Gain (AUC)	
						29.06713		22.60435

- Rigorous validation under more realistic settings needed before celebration but the outperformance above other baselines is very promising

Key Insights from this Talk

- Developing graph-based ML methods
  - Moving from a single embedding per node paradigm to contextual embedding learning
  - Where can we push further?
    - ✓ **Support Node and Edge Attributes**

- If you are a cyber-security researcher/practitioner
  - Consider using link prediction as a key method
    - ✓ **How often you need to re-train?**
  - Develop domain-informed pre-training ideas
    - ✓ **Integrate existing cyber knowledge bases**

- If you are interested in accelerating graph-based ML:
  - What does it mean to interleave GNNs and Transformers?
    - ✓ **Support heterogeneous networks with attributes (DataFrames)**
  - How to scale up context generation?
    - ✓ **Support message-passing models for with optimizations for sparsity**

Pacific Northwest
NATIONAL LABORATORY

Thank you

https://github.com/pnnl/SLICE