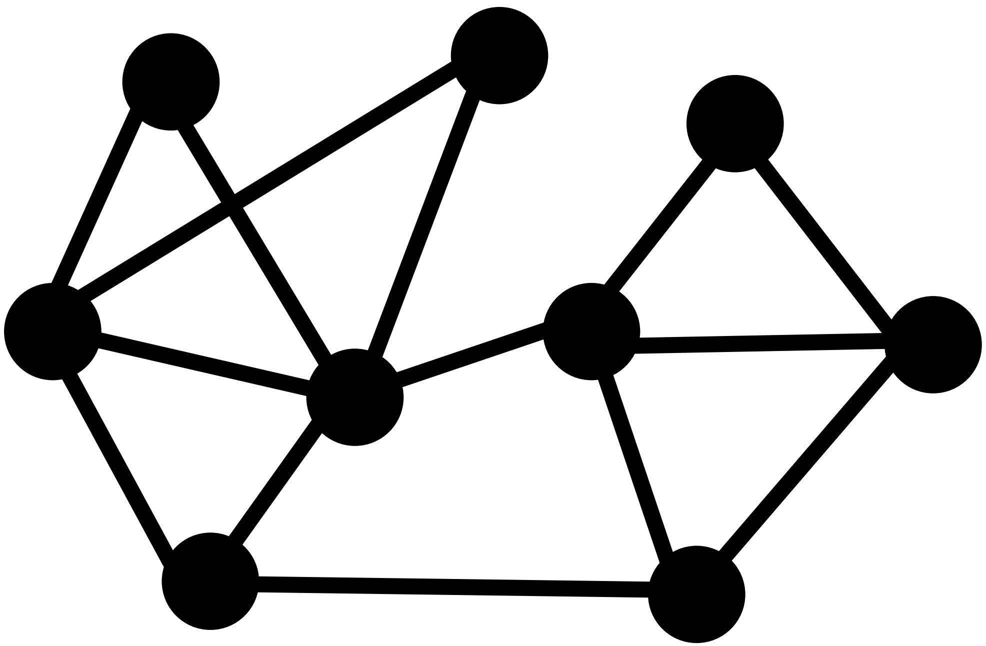
Stanford
Graph Learning
Workshop

Jure Leskovec

Graphs

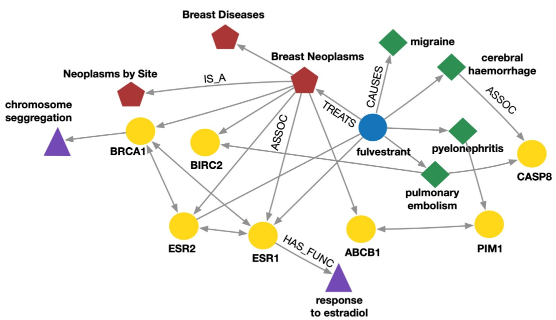
Graphs represent objects and their relationships as nodes and edges



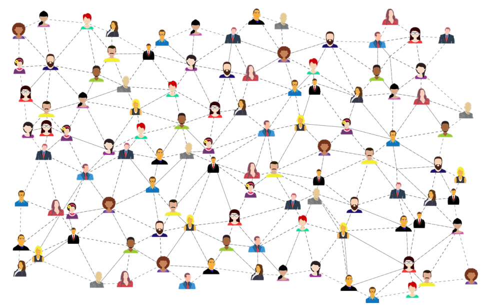
Graphs capture relations, which leads to better models.

Graphs in Many Domains

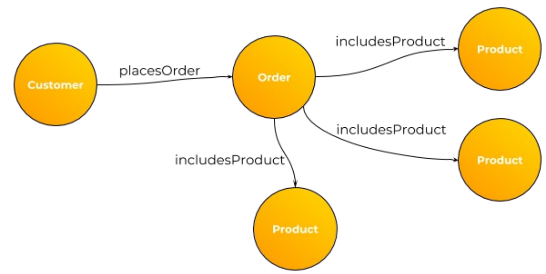
Often most valuable data are graphs:



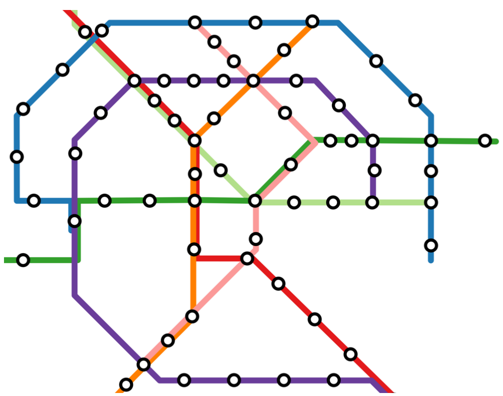
Healthcare



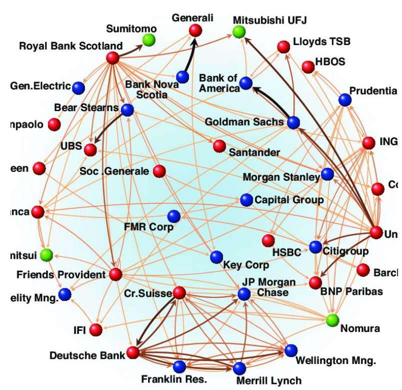
IT industry



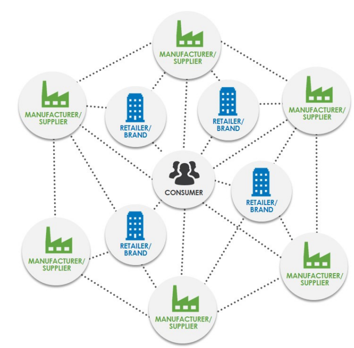
Commerce & Retail



Transportation



Finance & Insurance



Manufacturing

Today We are Showing

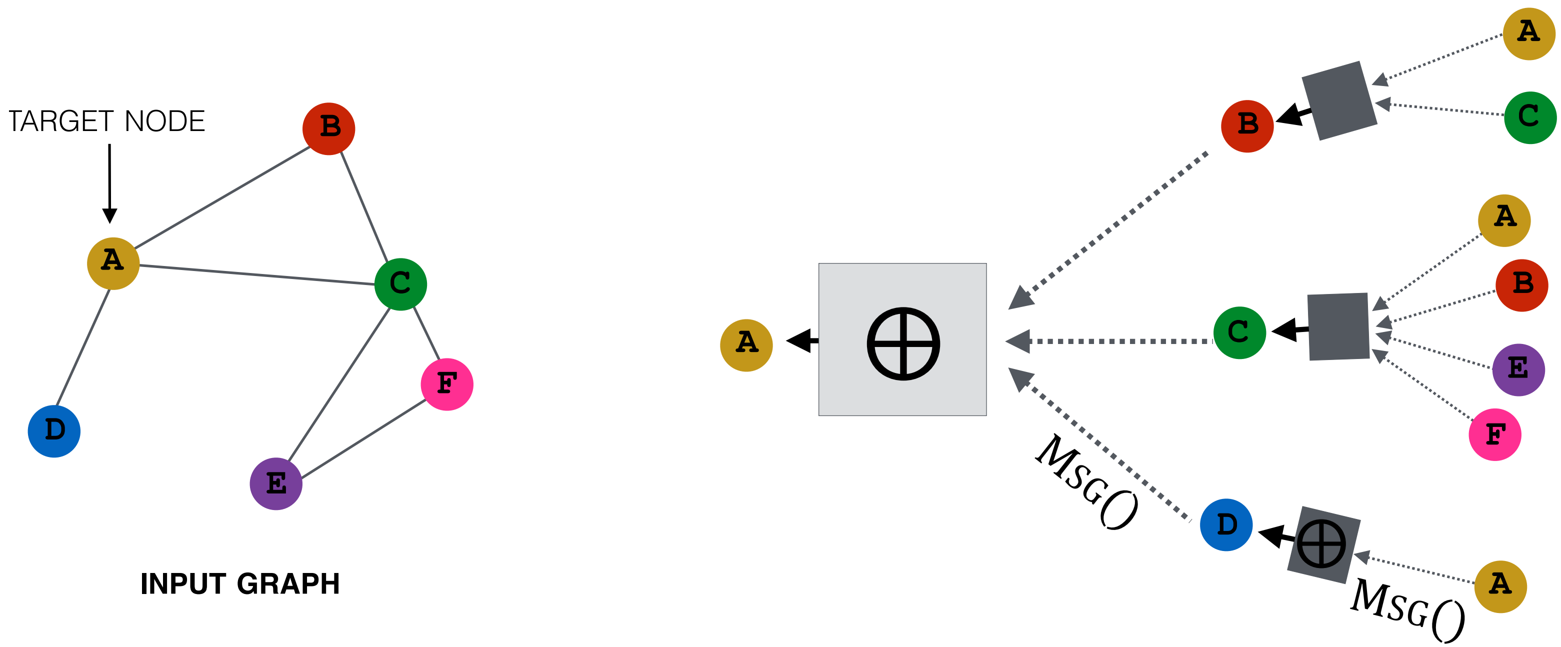
- Research advancements in Graph ML
- PyG ecosystem and partnerships
- Industrial applications
- How these technologies propel advancements of AI and applications of AI to next wave for science and industry
- We are going to announce new tools and partnerships that power these applications

Thank you!

Thank you to all the researchers, developers and partners who join us to make this event amazing.

Welcome to Stanford Graph Learning Workshop!

Deep Learning for Graphs



Each node defines a computation graph

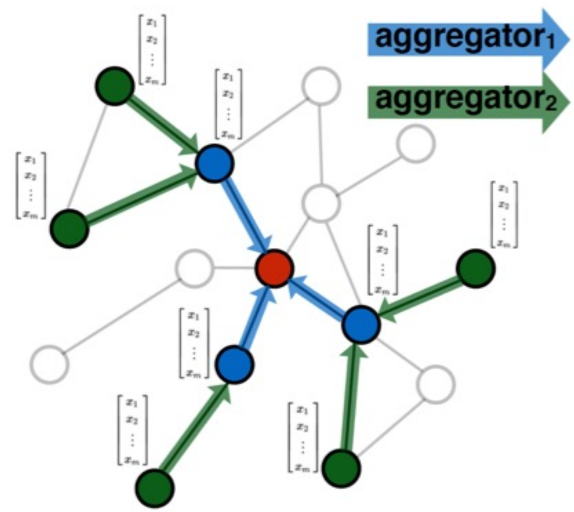
- Each edge is a message function MSG()
- ⊕ is a message aggregation function

Scarselli et al. 2005. [The Graph Neural Network Model](). *IEEE Transactions on Neural Networks*.

Graph Neural Networks

Graph Neural Networks (GNNs) can learn patterns and relations on a giant scale and train predictive models:

- **1)** GNNs learn to combine features from neighboring nodes
- **2)** GNNs learn the graph patterns and relations



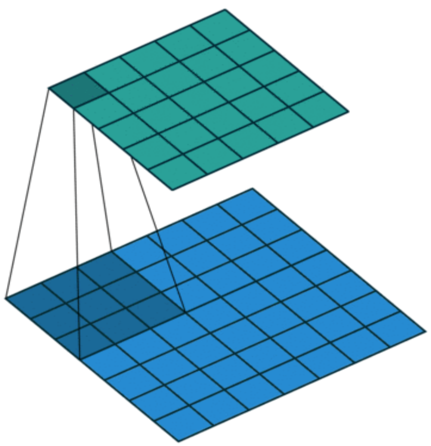
Key Benefits of GNNs

- GNNs adapt to local **shape** of data
  - Other methods assume fixed input (matrix, sequence) while GNNs capture local patterns around each node
- GNNs glue together other Neural Network architectures (CNNs, Transformers) and integrate multimodal data

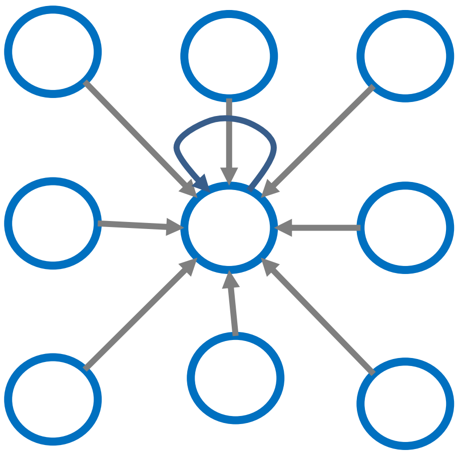
Key Benefits of GNNs

GNNs are extremely general: They subsume CNNs and Transformers as special cases:

- Example: CNN layer with 3x3 filter



Image

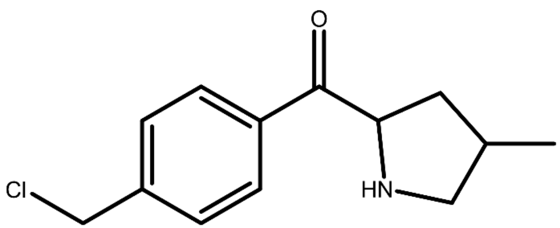


Graph

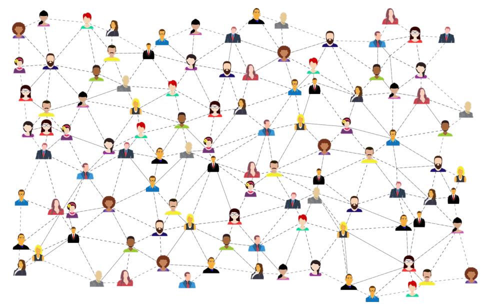
GNN formulation: $h_v^{(l+1)} = \sigma(\mathbf{W_l} \sum_{u \in N(v)} \frac{h_u^{(l)}}{|N(v)|} + B_l h_v^{(l)}), \forall l \in \{0, \dots, L-1\}$

CNN formulation: $h_v^{(l+1)} = \sigma(\sum_{u \in N(v)} \mathbf{W_l^u} h_u^{(l)} + B_l h_v^{(l)}), \forall l \in \{0, \dots, L-1\}$

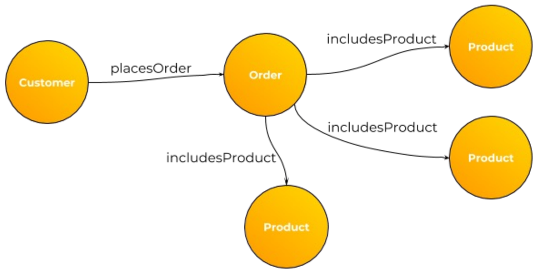
Many Applications of GNNs



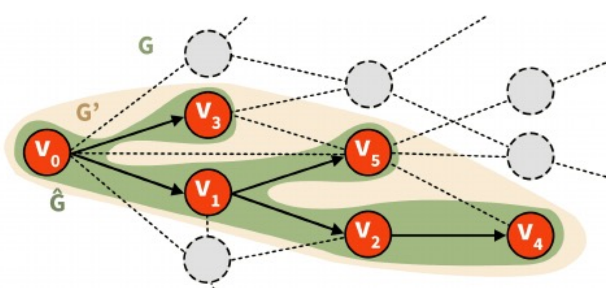
Drug discovery



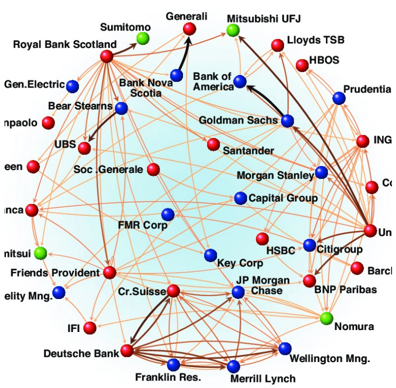
Recommender systems



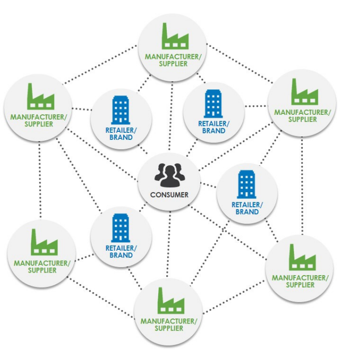
Customer 360 journey analysis



Fake news detection



Fraud detection



Supply chain optimization

The hottest subfield in ML

LOG CONFERENCE

Learning on Graphs Conference

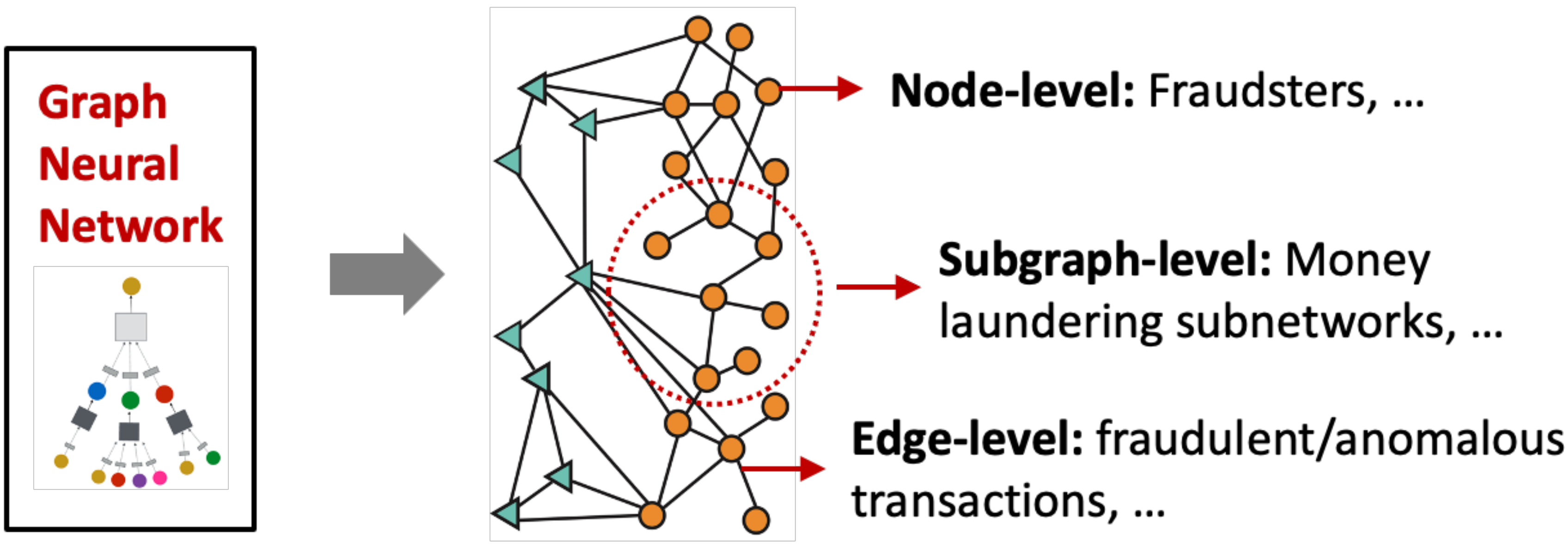
9th – 12th December 2022

https://logconference.org

Dynamic Financial Networks

Financial Networks: Describe financial entities and their connections

Tasks: Fraud detection, Anti-money laundering, Anomaly detection

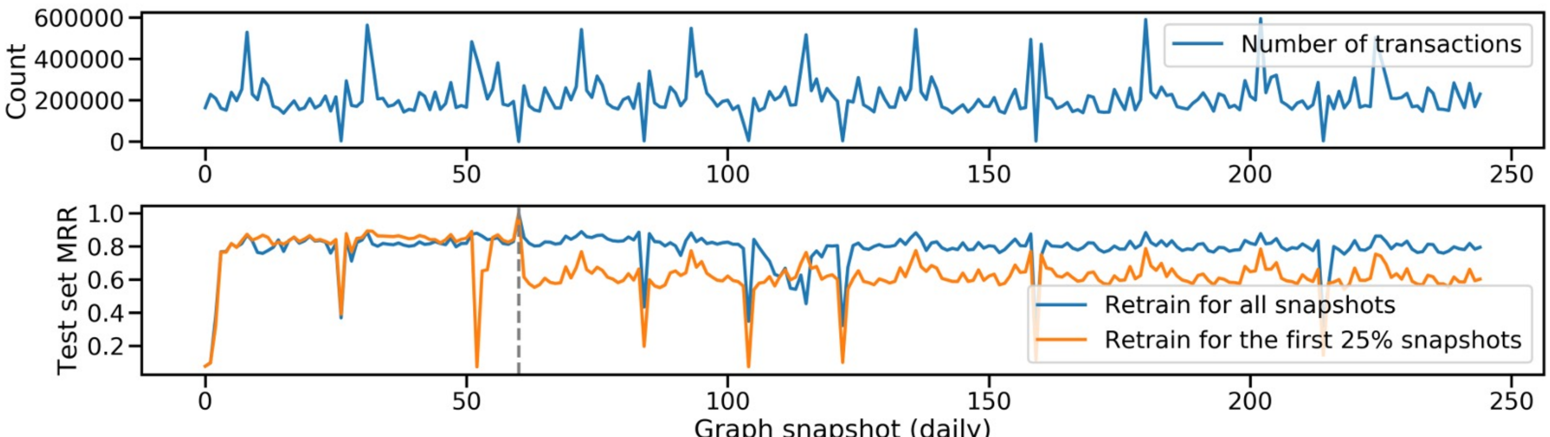


[ROLAND: Graph Learning Framework for Dynamic Graphs](). Jiaxuan You, Tianyu Du, Jure Leskovec. KDD 2022.

Forecasting Transactions

Central Bank of a European country forecasted financial transactions:

- Central Bank was able to process dynamic graphs with tens of millions of transactions:

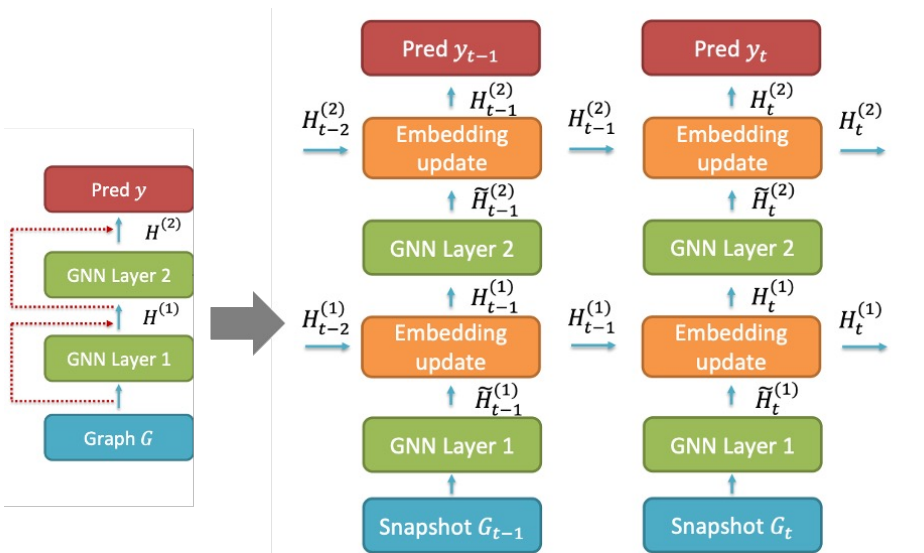


- 43-73% improvement in accuracy
  - Robust performance with changes in transaction pattern

ROLAND: Tool for Dynamic Graphs

We developed ROLAND:

- Easy creation of dynamic GNN from any static GNN
- Scalable and adaptive training



Static GNN

Dynamic GNN

Key idea: Recurrently update node embeddings *at each layer,* by injecting a new module to a static GNN:

Input:

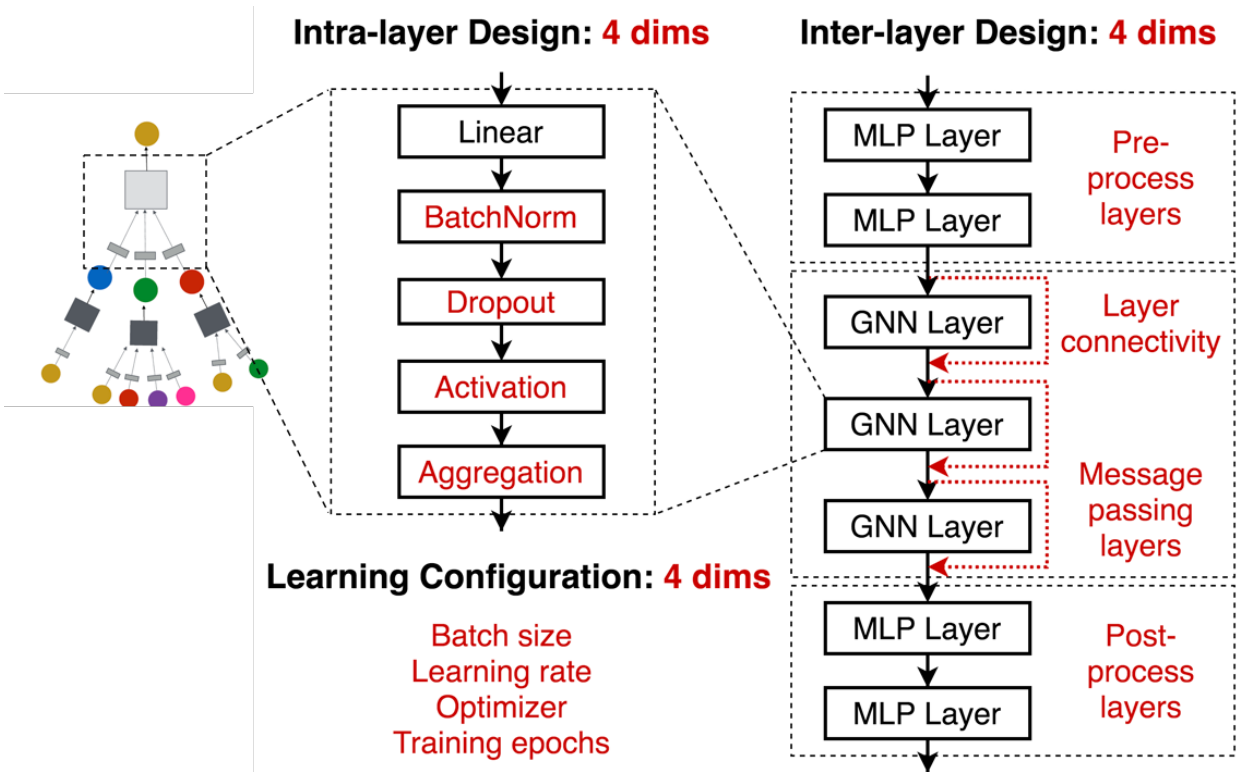
- **Previous embeddings** from the same layer
- **Current embeddings** from the previous layer

Output: Updated embeddings

https://github.com/snap-stanford/roland

ROLAND: Implementation

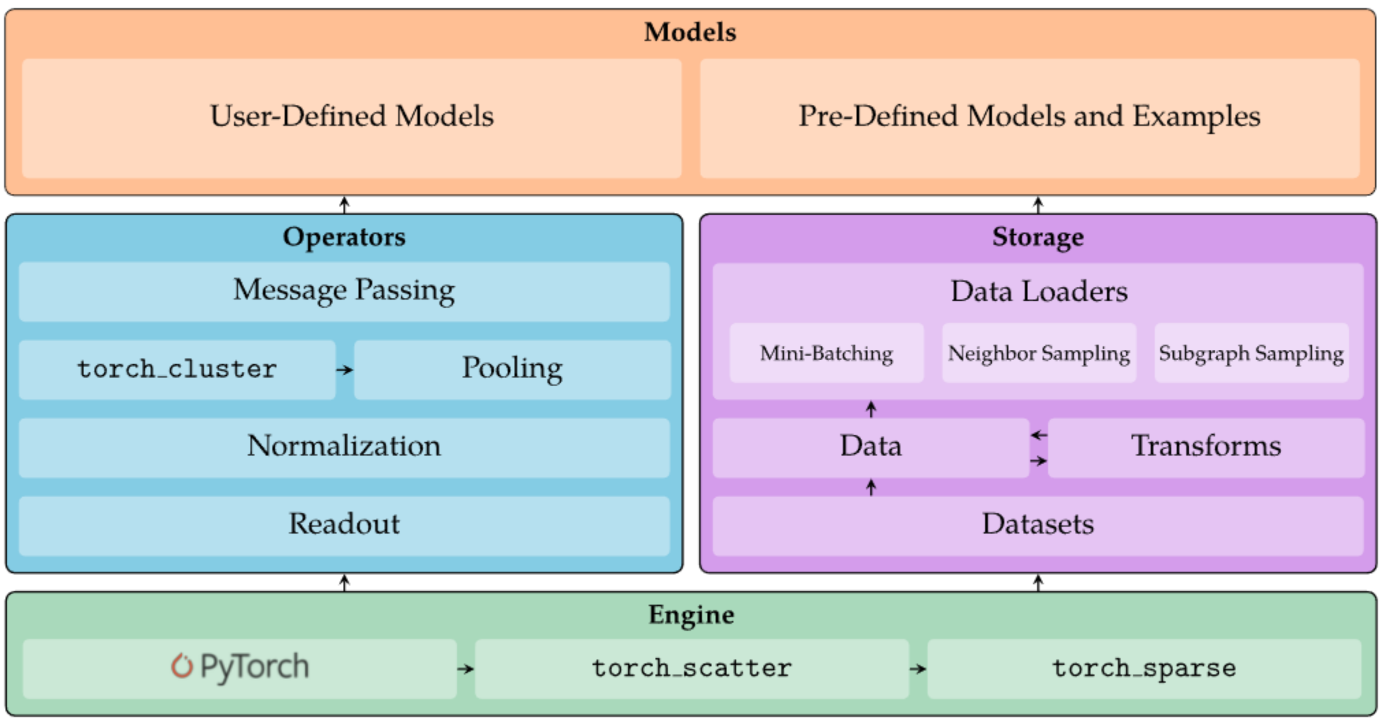
ROLAND **is built with** PyG GraphGym to efficiently explore the GNN design space



[Design Space for Graph Neural Networks](). Jiaxuan You, Rex Ying, Jure Leskovec. NeurIPS 2020

PyG: GNN Library

- PyG provides the state-of-the-art in Graph Representation Learning



PyG 2.1: What's New

- pyg-lib: A low-level GNN engine to further accelerate PyG, joint effort across different partnerships
- Improved GNN design via principled aggregations
- New model milestones: Scalable Link prediction, 1000+ layer deep GNNs, GNNs for heterophily graphs, ...

PyG: GNN Library

PyG is the most used Graph Representation Learning framework

- 80+ GNN architectures
- 200+ benchmark datasets
- Extendable via a message passing interface
- Dedicated sparsity-aware CUDA kernels
- Scalable and easy to use

PyG

```
conda install pyg -c pyg
```

stars 15k forks 2.7k

PyG Powers Products

AIRBUS anomaly and outlier detection

AstraZeneca knowledge graph for drug discovery

Spotify recommender system for audio books and podcasts

amazon fraud detection

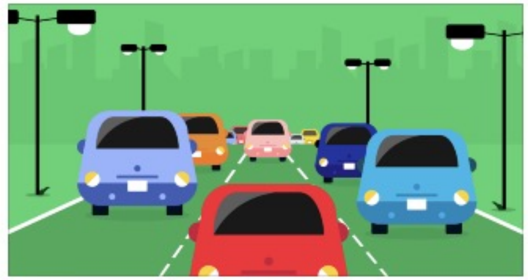
أرامكو السعودية saudi aramco fluid simulations

Graph ML Case Studies

Deep Learning on 3D Meshes

A learned solution to node-level classification on irregular graphs via graph neural networks.

A Anya Fries
Jan 25 · 7 min read



Predicting Los Angeles Traffic with Graph Neural Networks

By Julie Wang, Amelia Woodward, Tracy Cai as part of the Stanford CS224W course project.

Amelia Woodward
Jan 15 · 14 min read

Alexa, Queue That Banger!

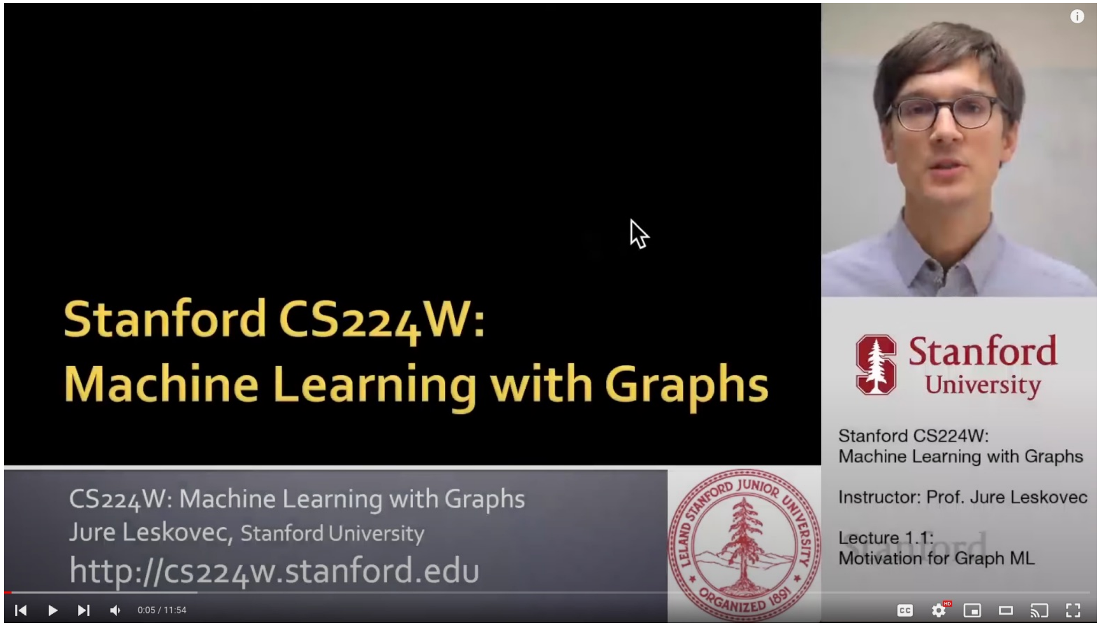
Recommending Spotify Playlist Tracks with Neural Collaborative Filtering Using Graph Machine Learning

Michał Skręta
Jan 15 · 14 min read

Graph ML Tutorials: https://medium.com/stanford-cs224w

- Graph Machine Learning tutorials created by Stanford students of CS224W.

Stanford Graph ML Videos



Stanford CS224W: Machine Learning with Graphs on YouTube

https://youtube.com/playlist?list=PLoROMvodv4rPLKxIpqhjhPgdQy7imNkDn

- Over 700,000 views

PyG: Huge Community

PYG.ORG:

- ~1600 research papers written using PyG
- ~100K monthly downloads
- ~300 external contributors/developers
- ~3k members on Slack (Join us: https://data.pyg.org/slack.html)

team@pyg.org

https://pyg.org

/pyg-team/pytorch-geometric

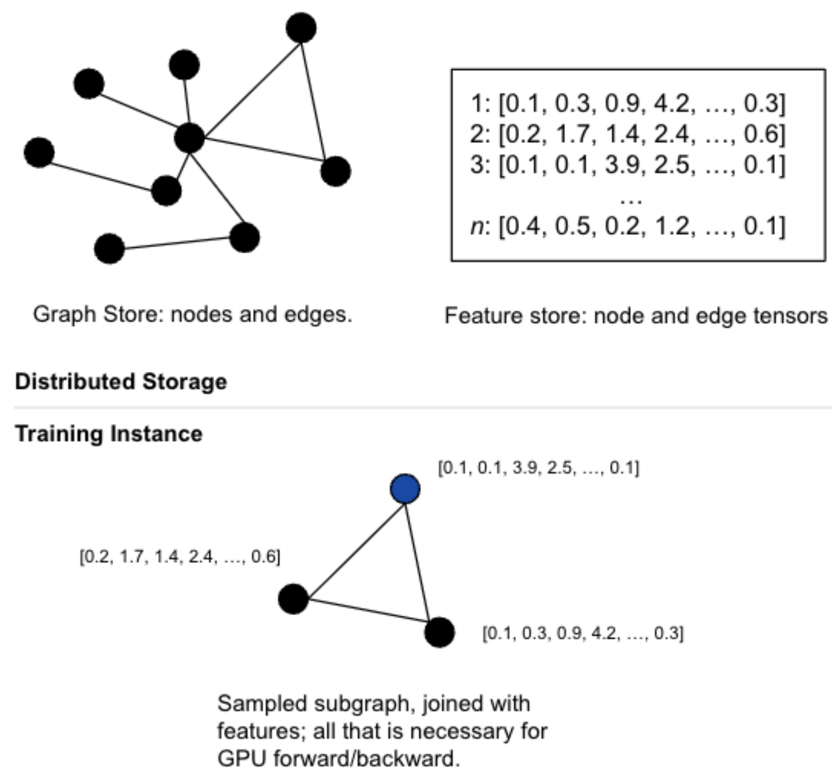
license MIT PRs welcome

Scaling-up Graph Learning

- In real-world applications graph sizes easily reach billions of nodes and edges
- Now with PyG we can learn deep learning models at a giant scale and train a predictive model for a graph

New PyG Abstractions

- Graphs contain both the wireframe and rich features on its nodes.
- Scaling up requires distributing this information out-of-core.
- New PyG graph store and PyG feature store abstractions enable modular scalability.



Partnership with NVIDIA

PyG & NVIDIA

NVIDIA joins as an official partner to accelerate GPU needs:

- GPU-accelerated neighbor sampling via cugraph integration
- GPU-accelerated heterogeneous GNN execution via typed matrix multiply

Partnership with Intel

PyG & intel

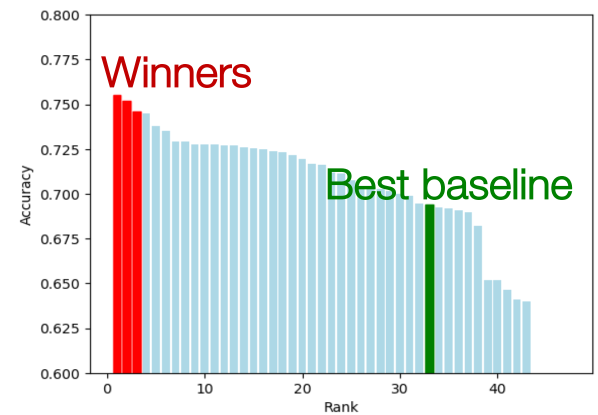
Intel joins as an official partner to accelerate CPU needs:

- Accelerated inference of GNNs on the CPU
- Accelerated neighbor sampling techniques (up to 20x speed-ups)

OGB-LSC-2

In 2021 we run OGB-LSC with over **500+ participating teams**

- Huge leap forward on model performance



We are announcing OGB-LSC-2:

OGB-LSC @ NeurIPS 2022

Large – Scale Challenge

Knowledge Graphs

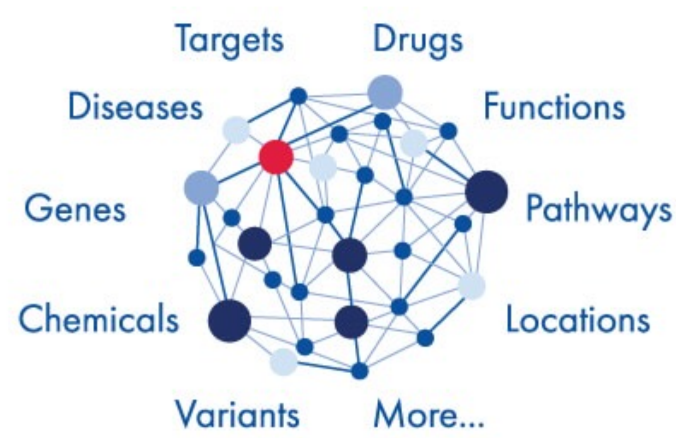
Knowledge Graph stores facts as triples (h, r, t)

- ('Mona Lisa', 'created_by', 'Leonardo da Vinci')
- ('Fulvestrant', 'causes', 'Migrane')
- ('Fulvestrant', 'treats', 'Breast Neoplasm')

- Examples:

WIKIDATA

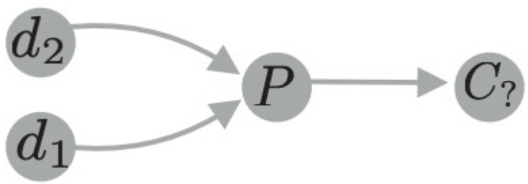
Freebase™

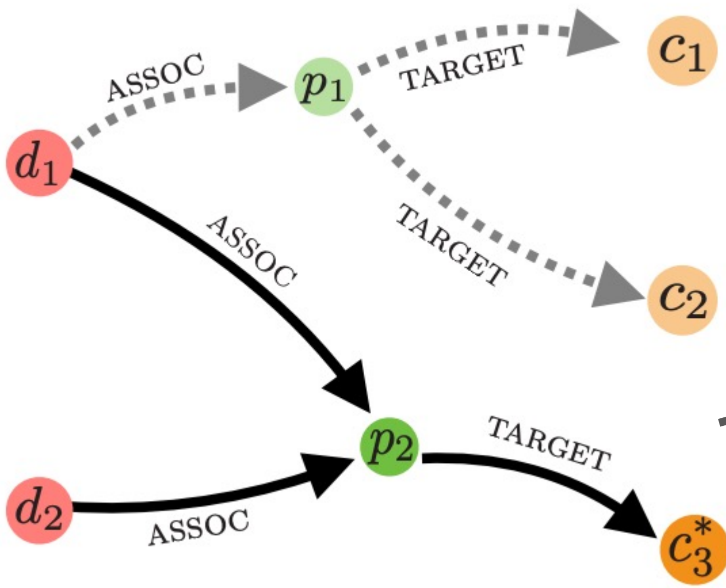


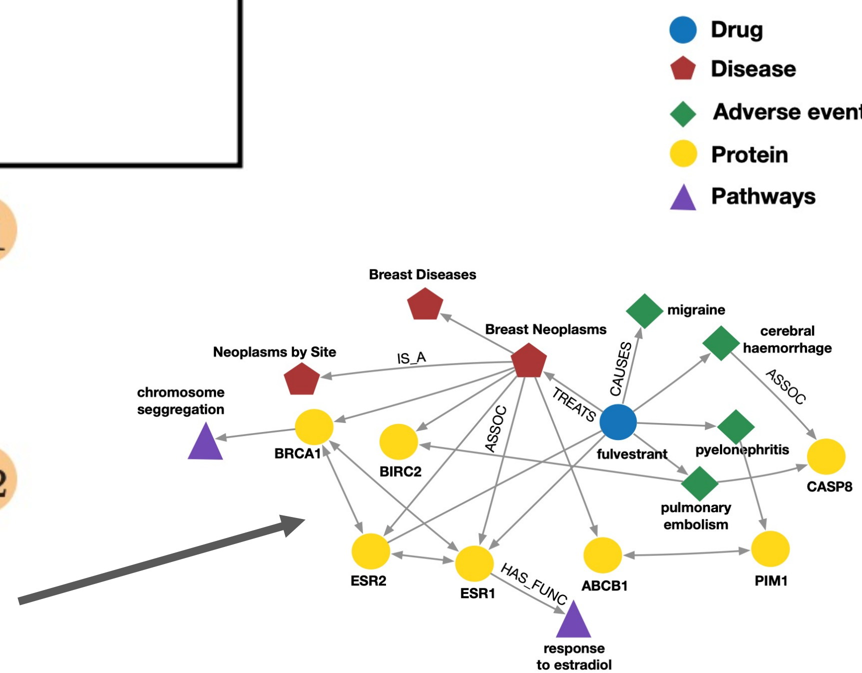
Predictive Queries over KGs

$C_?.\exists P : \textsc{assoc}(d_1, P) \wedge \textsc{assoc}(d_2, P) \wedge \textsc{target}(P, C_?)$

"Predict drugs $C_?$ that might target proteins that are associated with the given disease nodes d_1 and d_2 "





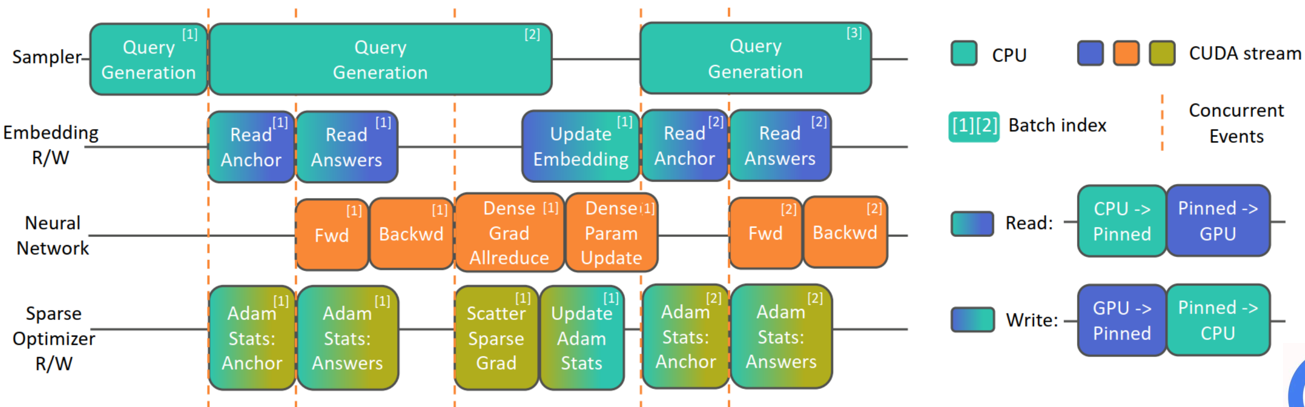


Biomedical knowledge graph

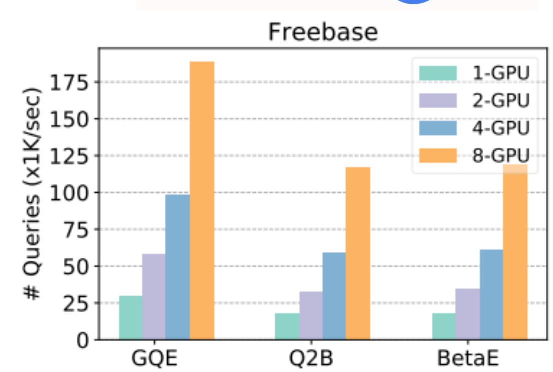
[Embedding Logical Queries on Knowledge Graphs. Hamilton, et al., NeurIPS 2018]
[Query2box: Reasoning over Knowledge Graphs in Vector Space Using Box Embeddings. Ren, et al., ICLR 2020]

KG Reasoning with SMORE

SMORE: Scalable framework for multi-hop Knowledge Graph reasoning



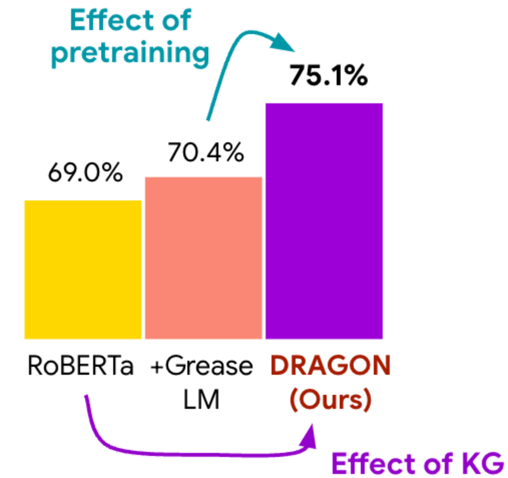
Scales to full Freebase KG: 86M nodes, 338M edges

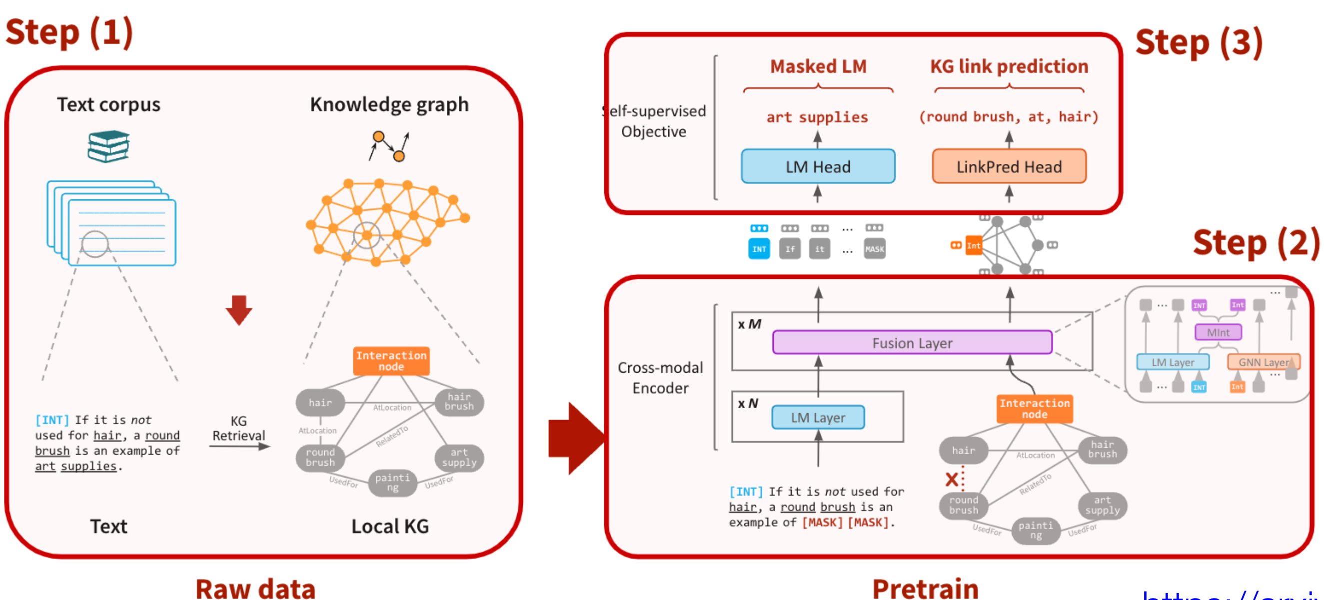


https://github.com/google-research/smore

Graphs & Language Models

Knowledge graphs help train more effective large language models.





https://arxiv.org/abs/2203.15827

Conclusion

- Today we announce new advances in research, new tools and platforms and exciting partnerships and updates to the PyG ecosystem
- We have an exciting program for you with talks from brightest minds in science and industry

Workshop Program

09:30 - 10:00 Matthias Fey, PyG – What's New in PyG
10:00 - 10:20 Ivaylo Bahtchevanov, PyG – Building PyG Open Source Community
10:20 - 10:40 Manan Shah & Dong Wang, Kumo.ai – Scaling-up PyG

11:00 - 11:20 Rishi Puri, Nvidia – Accelerating PyG with Nvidia GPUs
11:20 - 11:40 Ke Ding, Intel – Accelerating PyG with Intel CPUs
11:40 - 12:00 Andreas Damianou, Spotify – Podcast Recommendations Spotify
12:00 - 12:20 Hema Raghavan, Kumo.ai – Query the Future using PyG
12:20 - 12:30 Joseph Huang, Stanford – Stanford CS LINXS Summer Program

13:30 - 13:50 Marinka Zitnik, Harvard – Graph AI to Enable Precision Medicine
13:50 - 14:10 Bryan Perozzi, Google – Challenges and Solutions for GNNs at Google
14:10 - 14:30 Srijan Kumar, GaTech –GNNs for Web Safety and Integrity
14:30 - 14:50 Luna Dong, Meta – Graph Mining for Intelligent Assistants
14:50 - 15:10 Michi Yasunaga, Stanford – Graph Learning in NLP Applications

15:30 - 15:50 Weihua Hu, Stanford – Learning Backward Compatible Embeddings
15:50 - 16:10 Hongyu Ren, Stanford – Multi-hop Reasoning in Knowledge Graphs
16:10 - 17:00 Panel – Challenges and Opportunities for Graph Learning

- Naren Chittar, JPMorgan Chase (moderator)
- Evan Feinberg, Genesis Therapeutics
- Yunyao Li, Apple
- Neil Shah, Snap
- Karthik Subbian, Amazon

Logistics

- Q&A: Subscribe to PyG Slack: https://data.pyg.org/slack.html and join #workshop-2022
- Live stream: https://youtu.be/GYW286H3SKw

Thank you!!

Thank you for attending!

Thank you Stanford Data Science Initiative for organizing:

Joseph Huang

Malwana Adalat

Rok Sosic

Ivaylo Buhtchevanov (PyG)