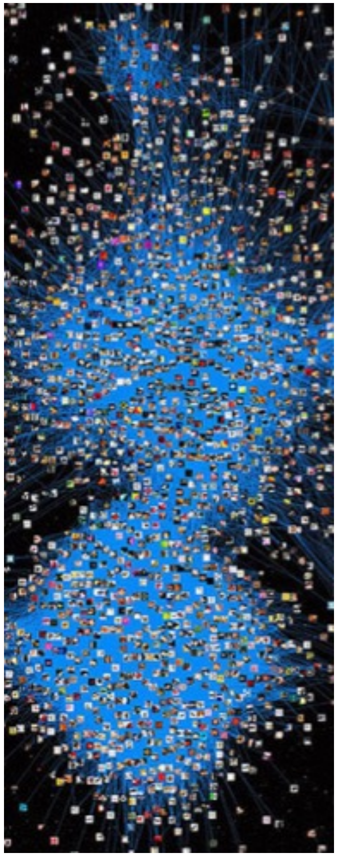
How Powerful are Graph Neural Networks

Joint work with R. Ying, J. You, M. Zitnik, W. Hamilton, W. Hu, et al.

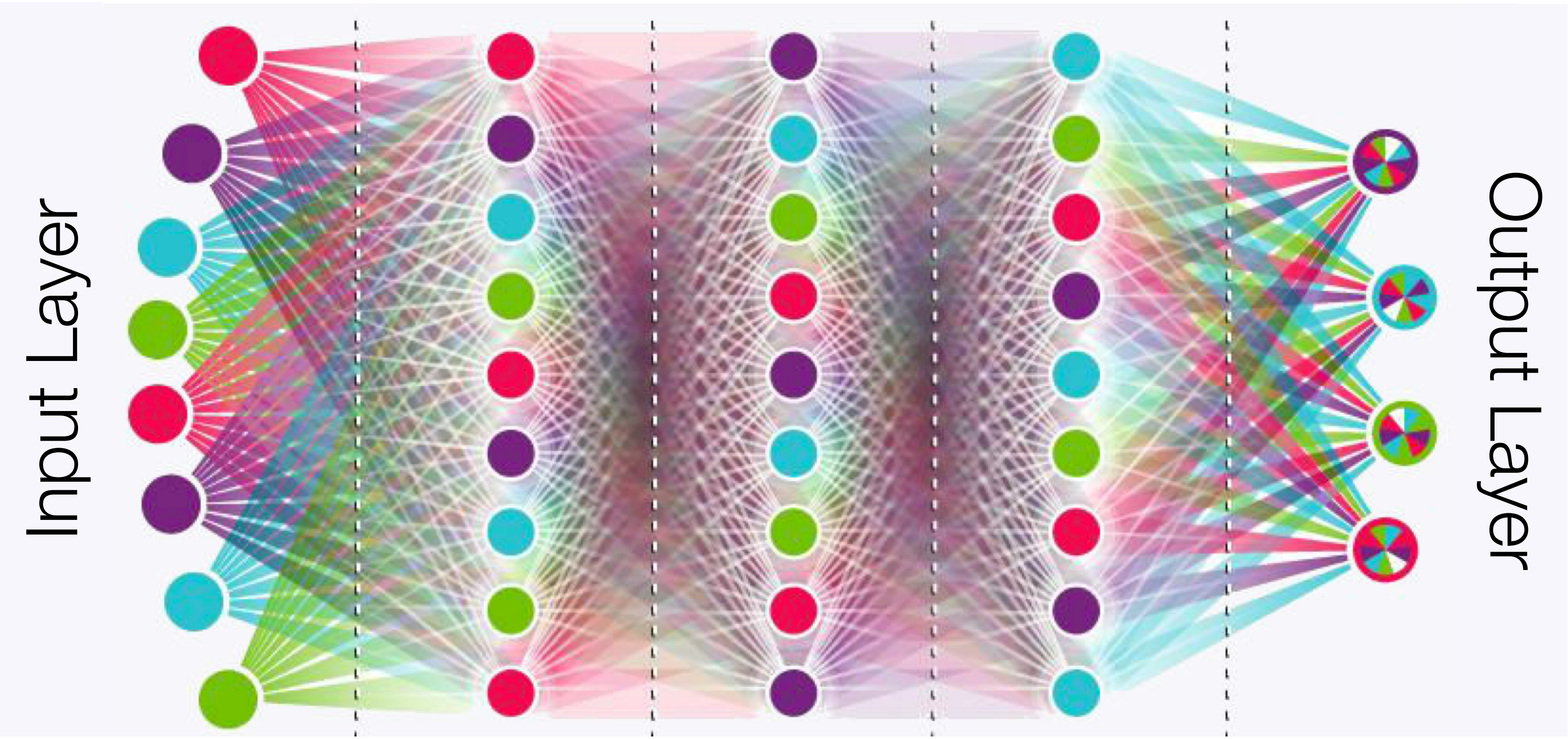
Jure Leskovec

BIG Data

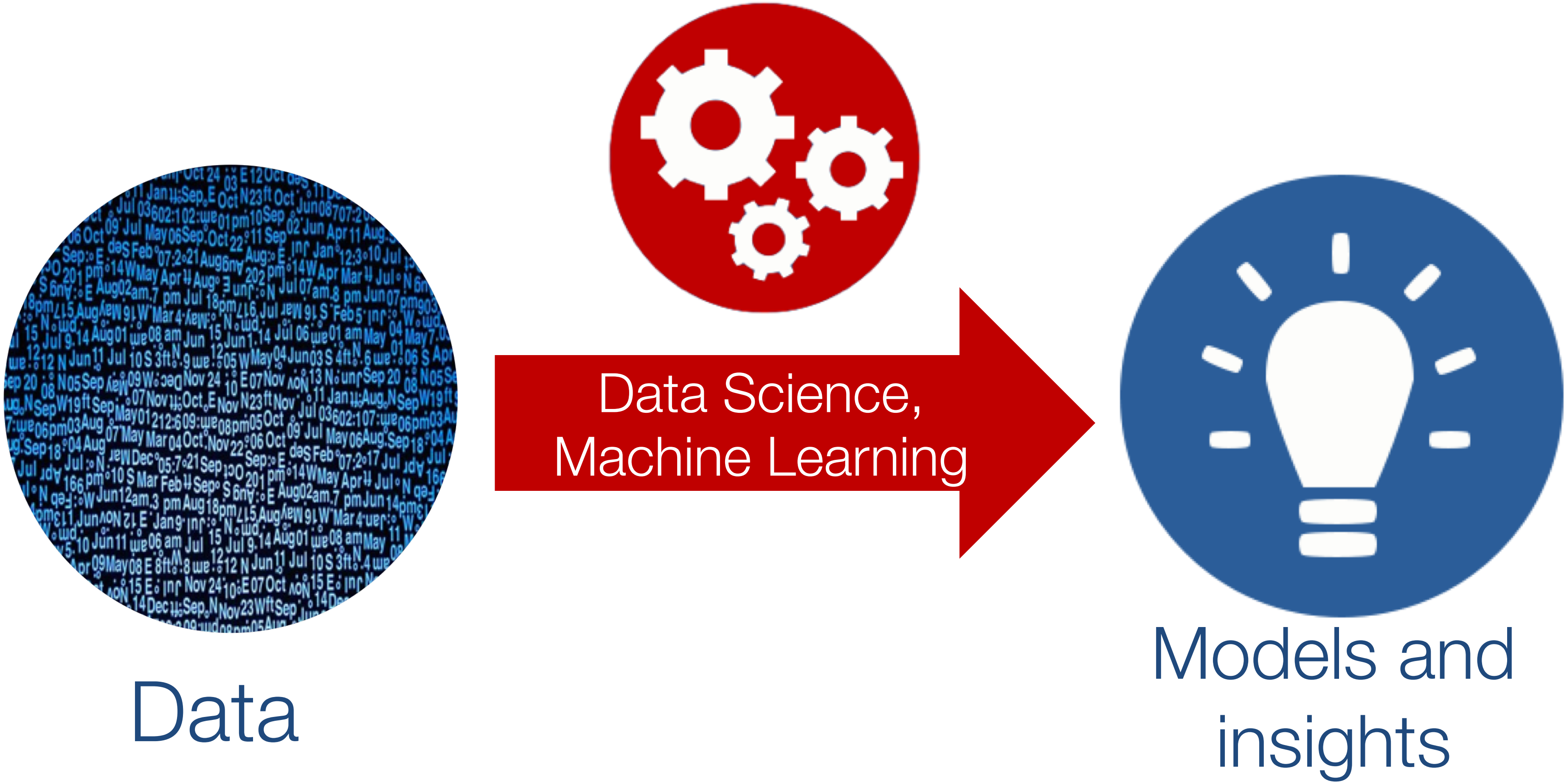
Digital transformation of science and society

Machine Learning



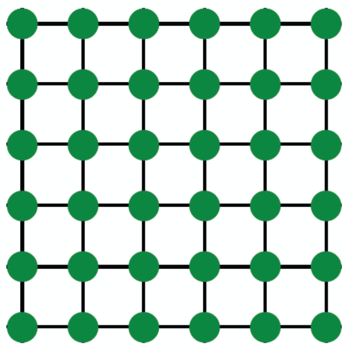
Advances in Machine Learning & Statistics

New Paradigm For Discovery

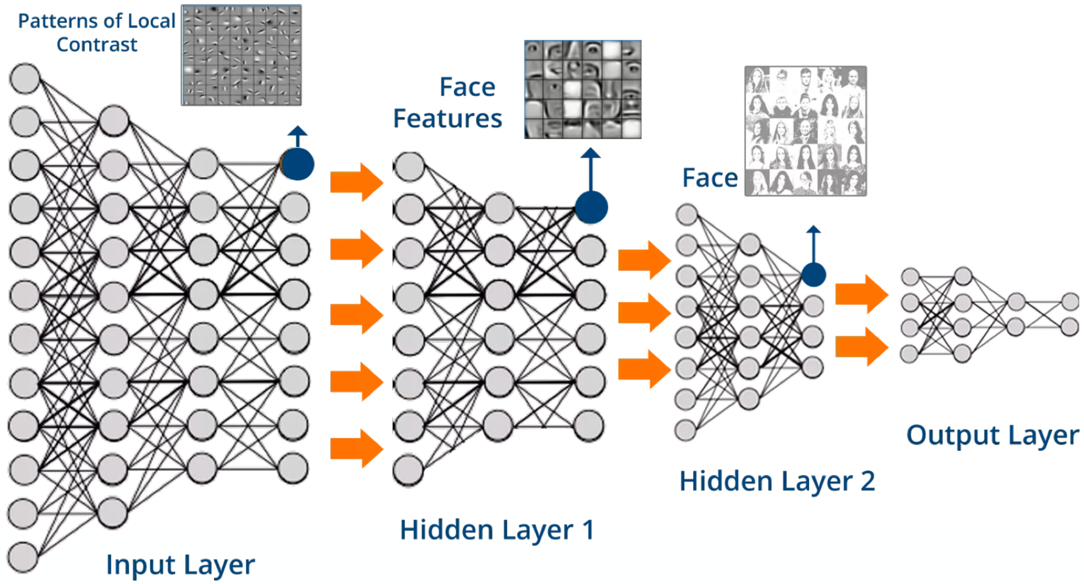


Massive data: Observe “invisible” patterns

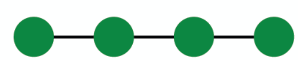
Modern ML Toolbox

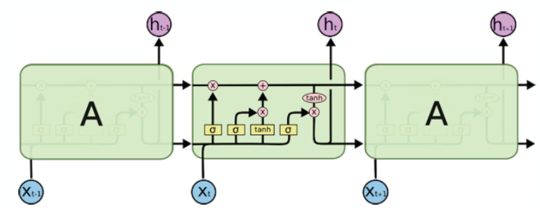


Images



Text/Speech





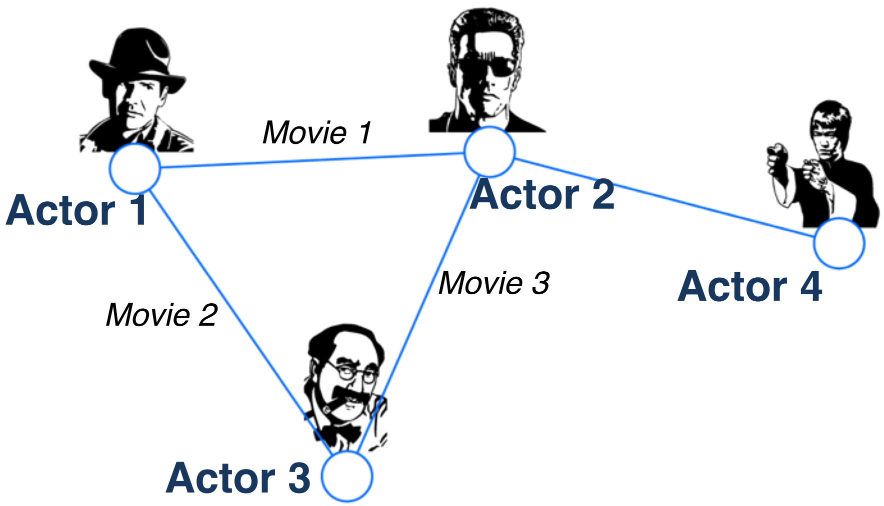
Modern deep learning toolbox is designed for simple sequences & grids

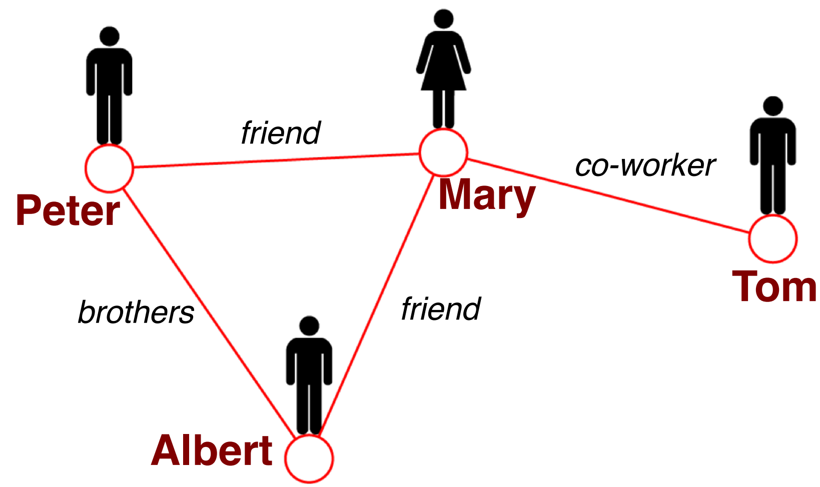
But not everything can be represented as a sequence or a grid

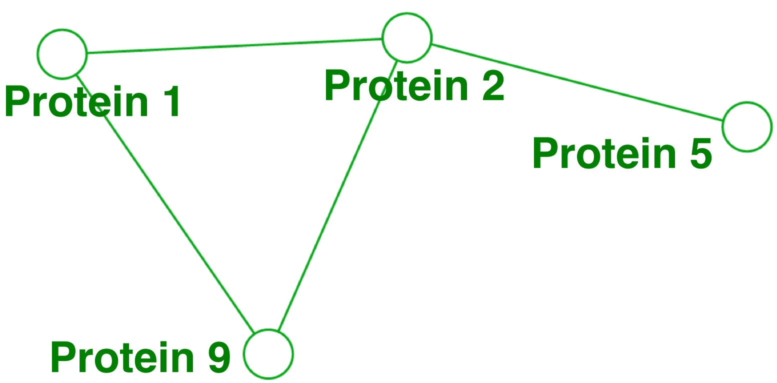
How can we develop neural networks that are much more broadly applicable?

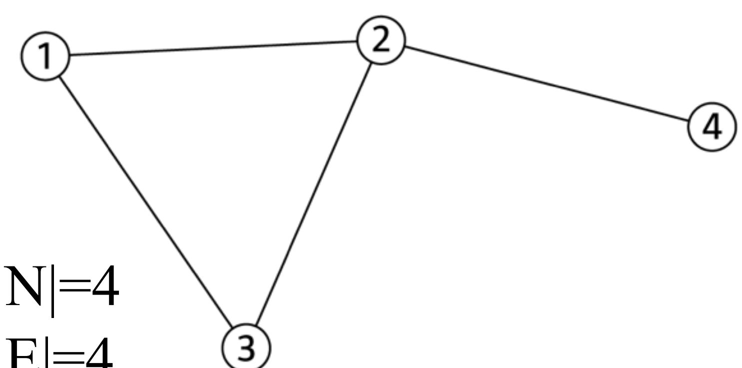
New frontiers beyond classic neural networks that learn on images and sequences

Networks: Common Language





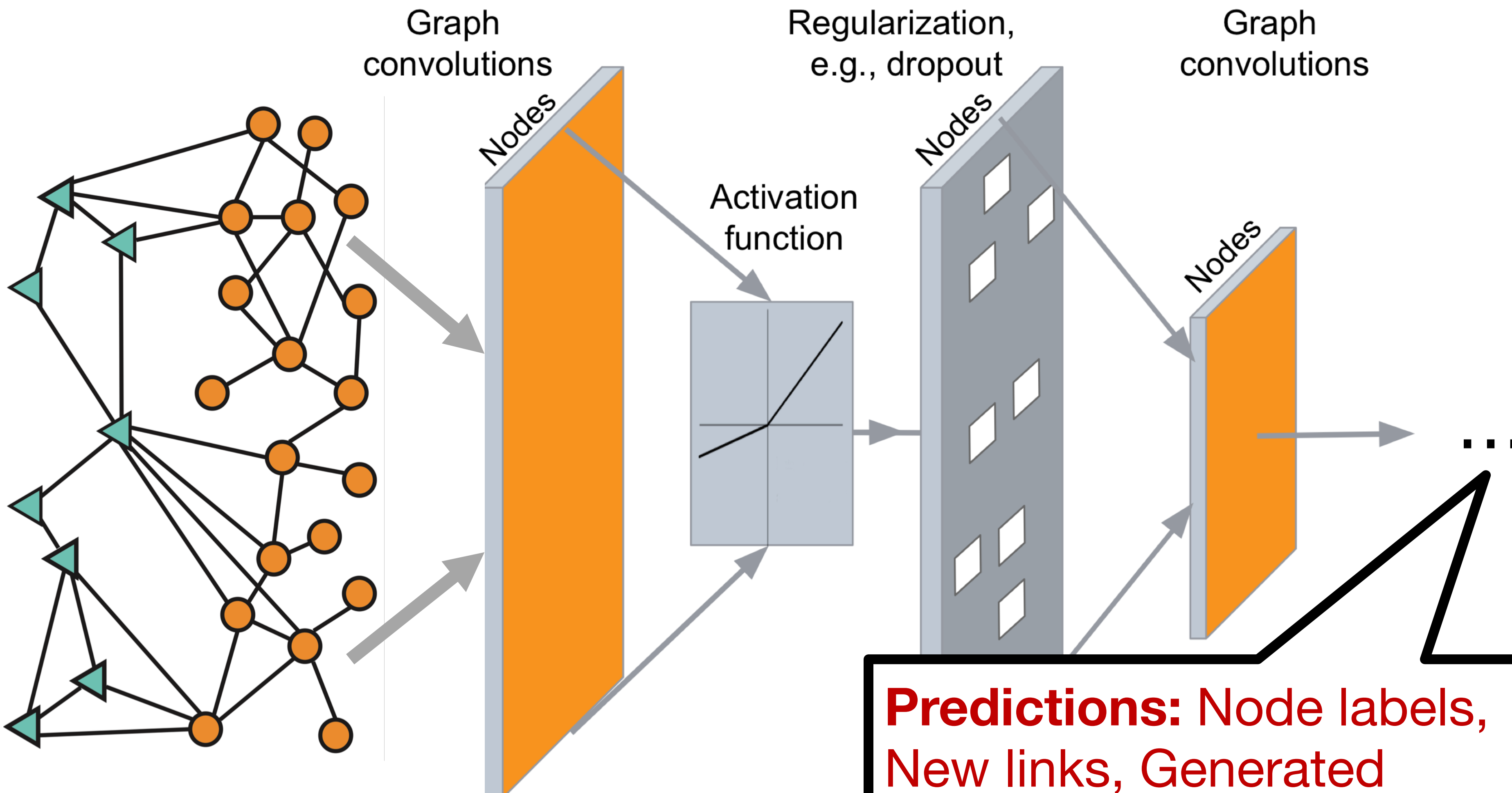




$|N|=4$

$|E|=4$

Deep Learning in Graphs



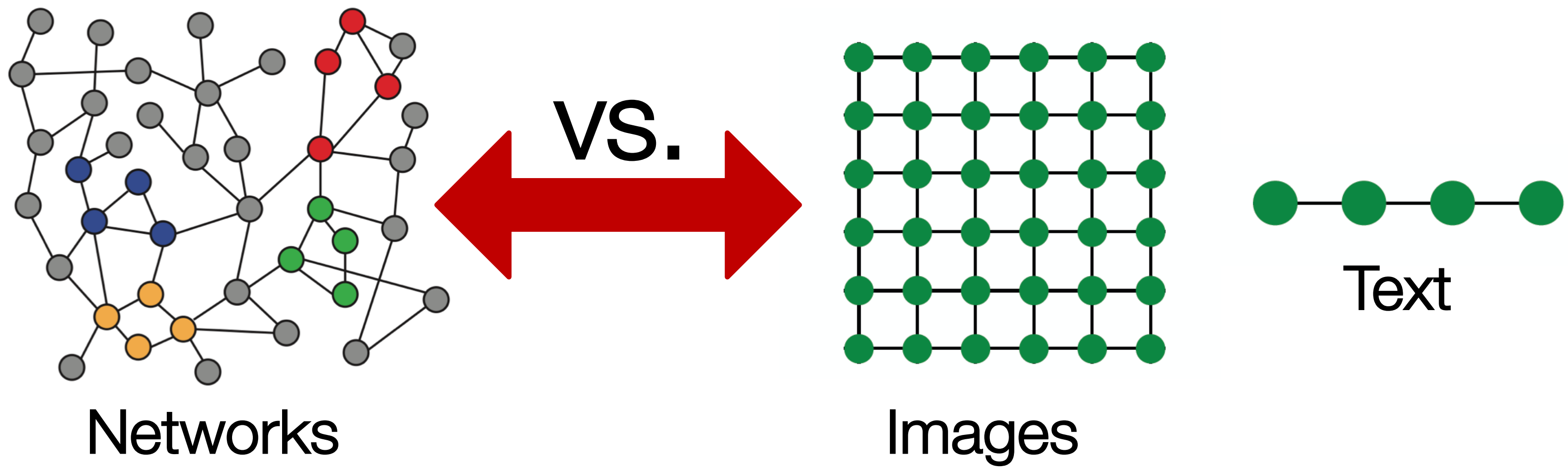
Predictions: Node labels, New links, Generated graphs and subgraphs

Input: Network

Why is it Hard?

But networks are far more complex!

- Arbitrary size and complex topological structure (i.e., no spatial locality like grids)



- No fixed node ordering or reference point
- Often dynamic and have multimodal features

GraphSAGE: Graph Neural Networks

Inductive Representation Learning on Large Graphs.
W. Hamilton, R. Ying, J. Leskovec. Neural Information Processing Systems (NIPS), 2017.
Representation Learning on Graphs: Methods and Applications.
W. Hamilton, R. Ying, J. Leskovec. IEEE Data Engineering Bulletin, 2017.

http://snap.stanford.edu/graphsage

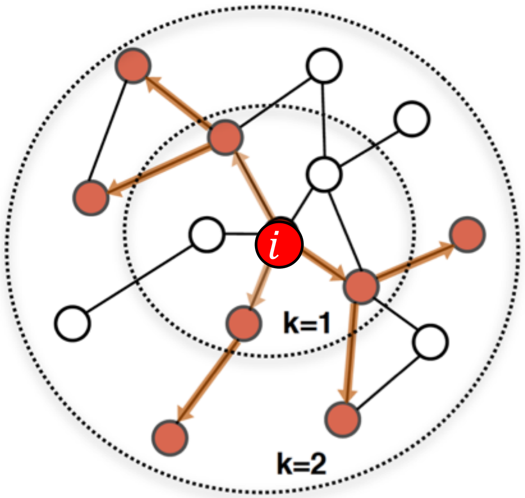
Setup

We have a graph G:

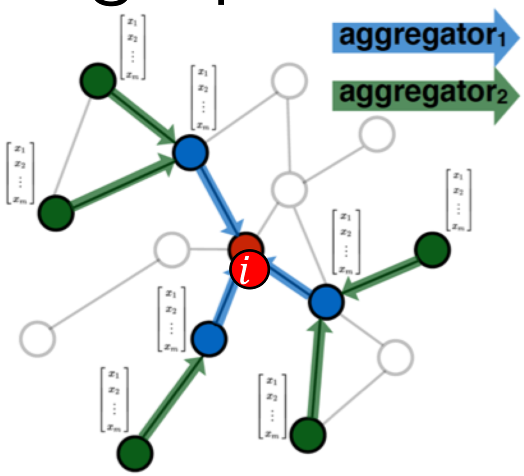
- V is the vertex set
- $\boldsymbol{A}$ is the (binary) adjacency matrix
- $\boldsymbol{X} \in \mathbb{R}^{m \times |V|}$ is a matrix of node features
  - Meaningful node features:
    - Social networks: User profile
    - Biological networks: Gene expression profiles, gene functional information

Graph Neural Networks

Idea: Node's neighborhood defines a computation graph



Determine node computation graph

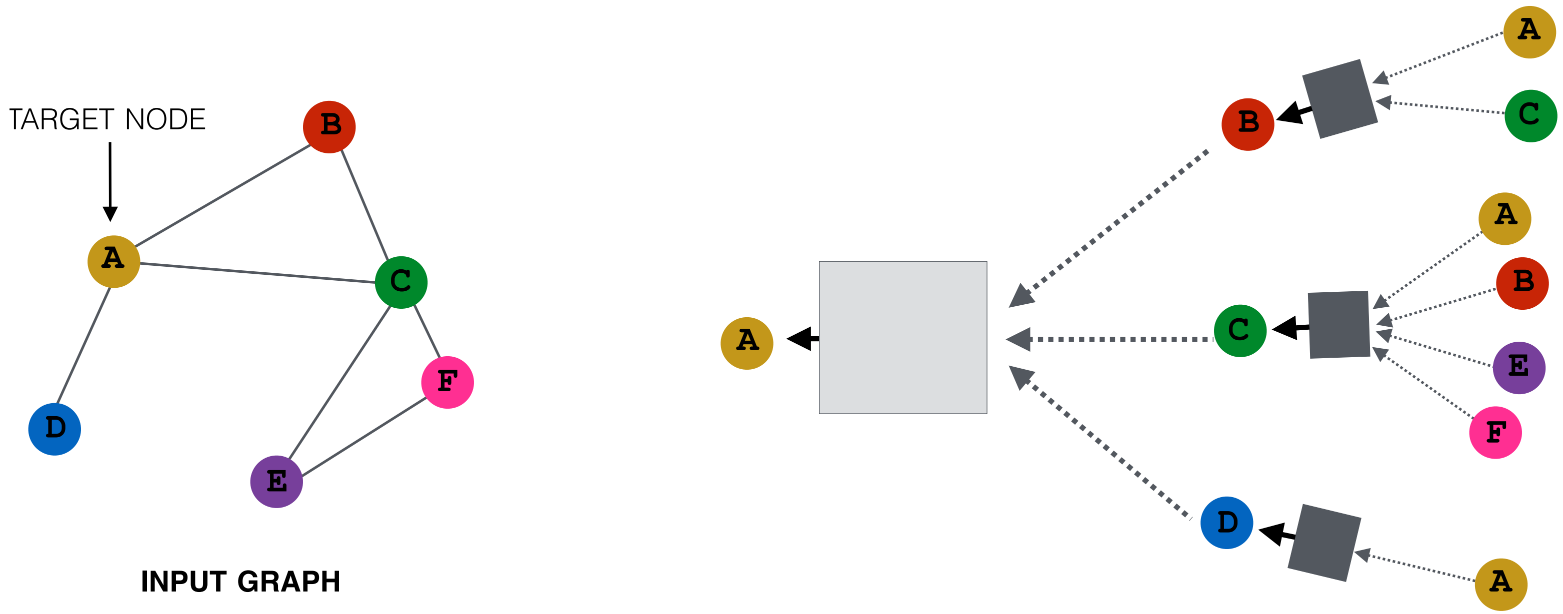


Propagate and transform information

Learn how to propagate information across the graph to compute node features

The Graph Neural Network Model. Scarselli et al. *IEEE Transactions on Neural Networks* 2005
Semi-Supervised Classification with Graph Convolutional Networks. T. N. Kipf, M. Welling, ICLR 2017

Graph Neural Networks

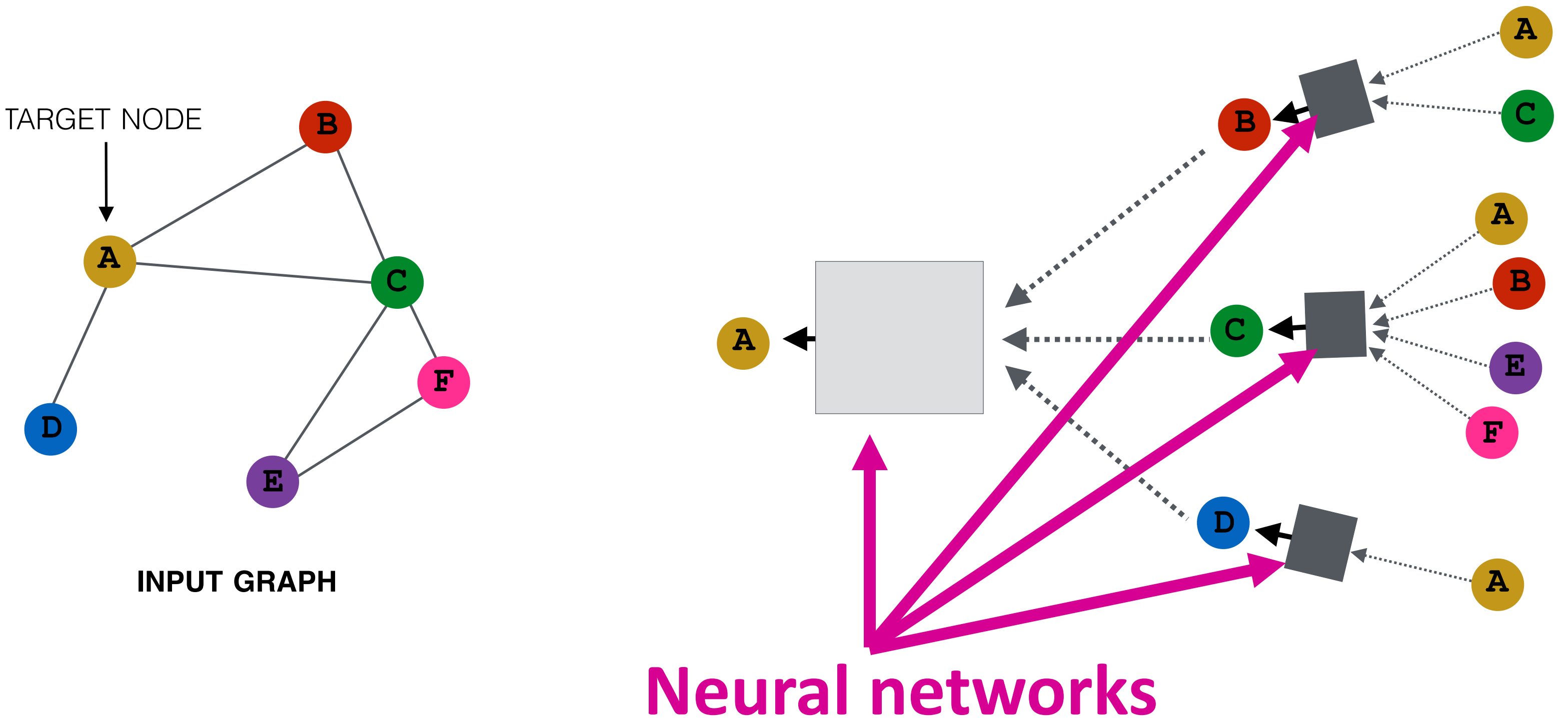


Each node defines a computation graph

- Each edge in this graph is a transformation/aggregation function

[Inductive Representation Learning on Large Graphs](). W. Hamilton, R. Ying, J. Leskovec. NIPS, 2017.

Graph Neural Networks

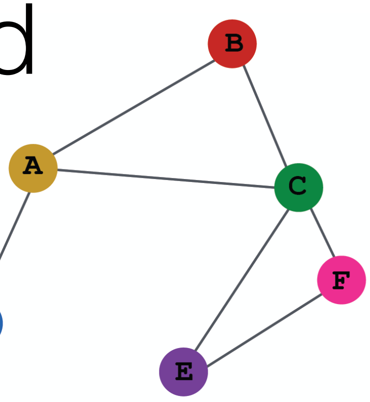


Intuition: Nodes aggregate information from their neighbors using neural networks

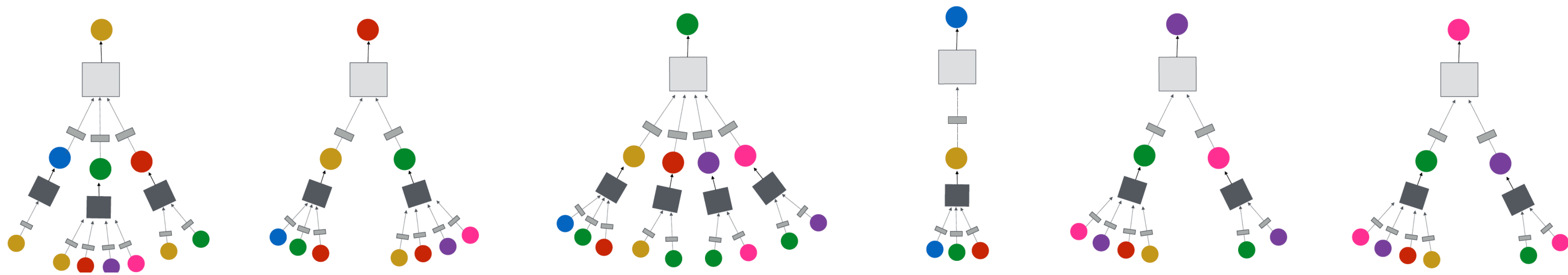
[Inductive Representation Learning on Large Graphs](). W. Hamilton, R. Ying, J. Leskovec. NIPS, 2017.

Idea: Aggregate Neighbors

Intuition: Network neighborhood defines a computation graph



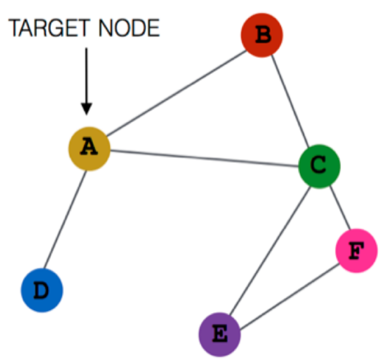
Every node defines a computation graph based on its neighborhood!

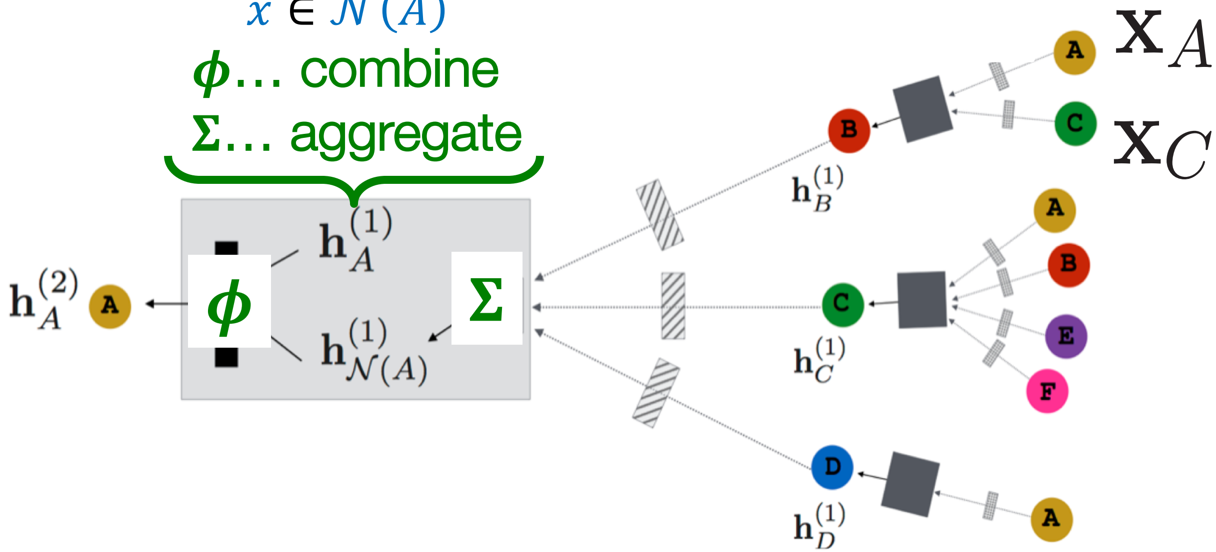


Can be viewed as learning a generic linear combination of graph low-pass and high-pass operators

Our Approach: GraphSAGE

$$\phi\Big(\sum_{x \in \mathcal{N}(A)} (h_x)\Big)$$





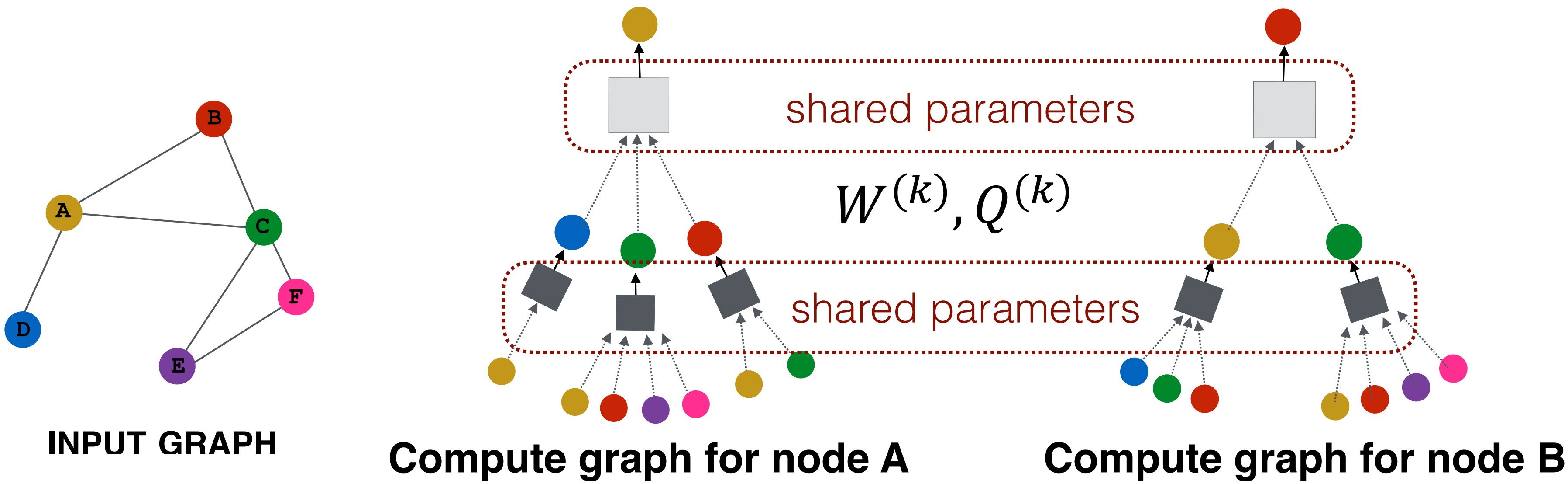
Update for node A:

$$\underbrace{h_A^{(k+1)}}_{\substack{k+1^{st}\text{ level}\\ \text{embedding of node } A}} = \sigma\Big(\underbrace{W^{(k)} h_A^{(k)}}_{\substack{\text{Transform } A\text{'s own}\\ \text{embedding from level } k}},\ \sum_{x\in\mathcal{N}(A)} \underbrace{\big(\sigma(Q^{(k)} h_x^{(k)})\big)}_{\substack{\text{Transform and aggregate}\\ \text{embeddings of neighbors } n}}\Big)$$

- $h_A^{(0)}$ = attributes X_A of node A, $\sigma(\cdot)$ is a sigmoid activation function

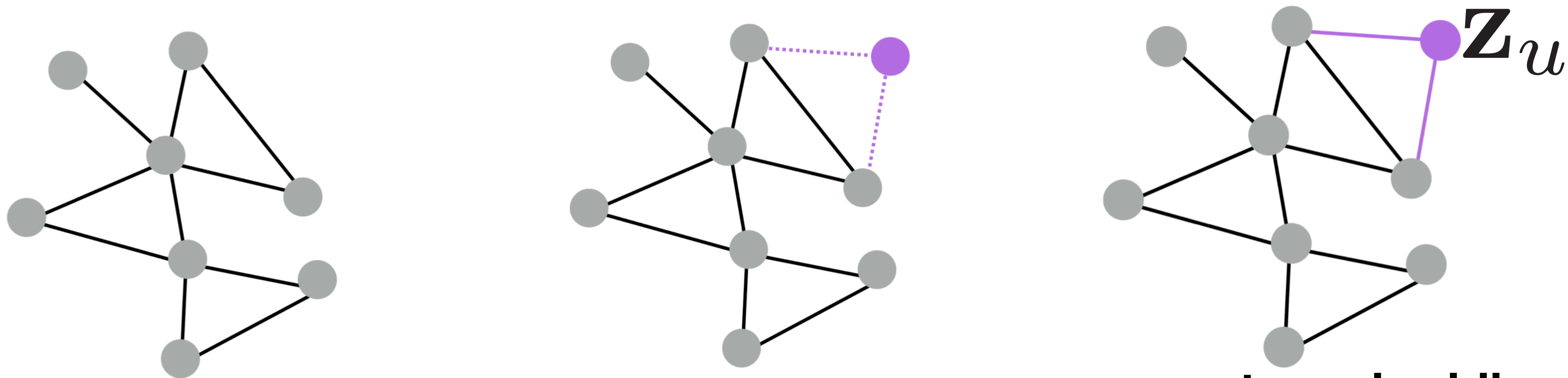
[NIPS '17]

GraphSAGE: Training

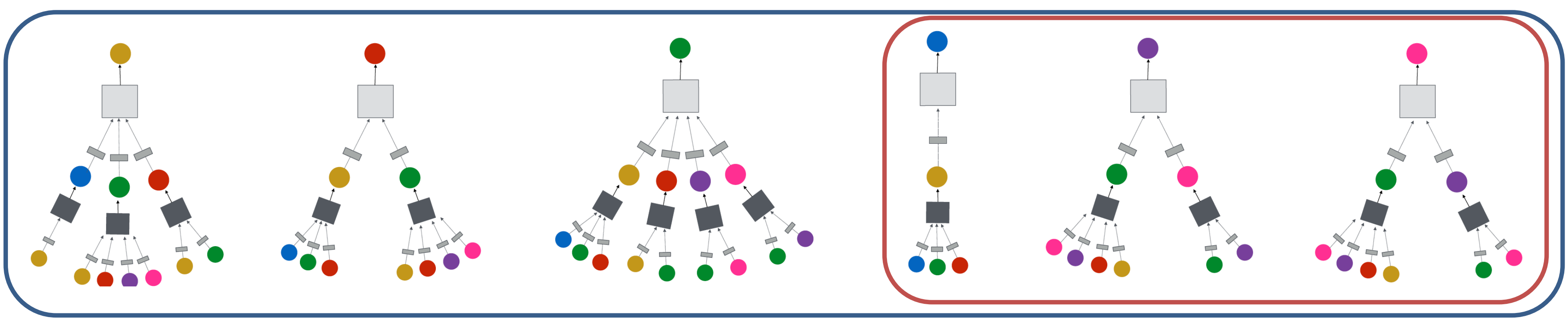


- Aggregation parameters are shared for all nodes
- Number of model parameters is independent of |V|
- Can use different loss functions:
  - Classification/Regression: $\mathcal{L}(h_A) = ||y_A - f(h_A)||^2$
  - Pairwise Loss: $\mathcal{L}(h_A, h_B) = \max(0, 1 - dist(h_A, h_B))$

Inductive Capability



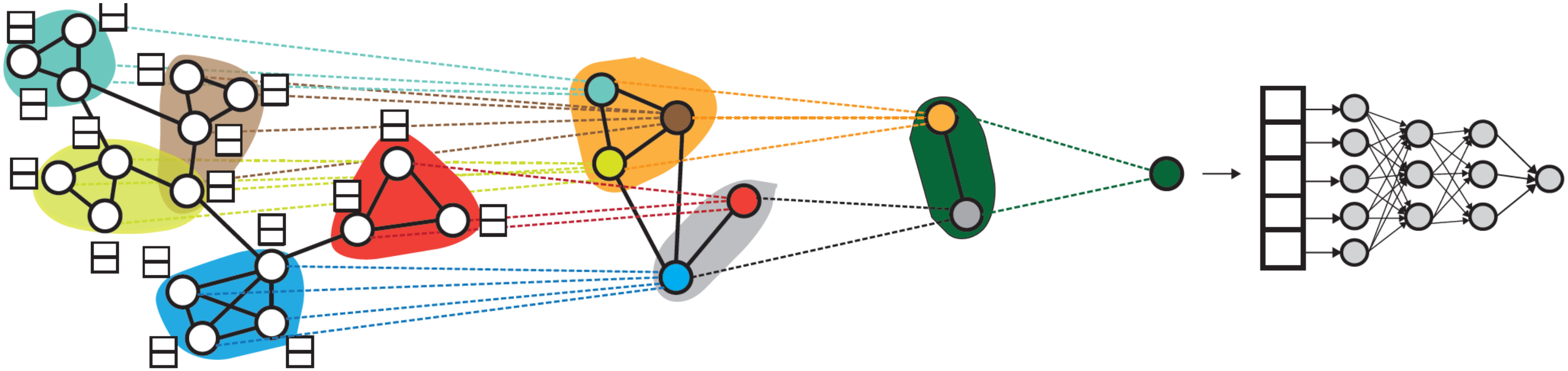
train with a snapshot **new node arrives** **generate embedding for new node**



Even for nodes we never trained on!

[NeurIPS '18]

Embedding Entire Graphs



Don't just embed individual nodes.
Embed the entire graph.

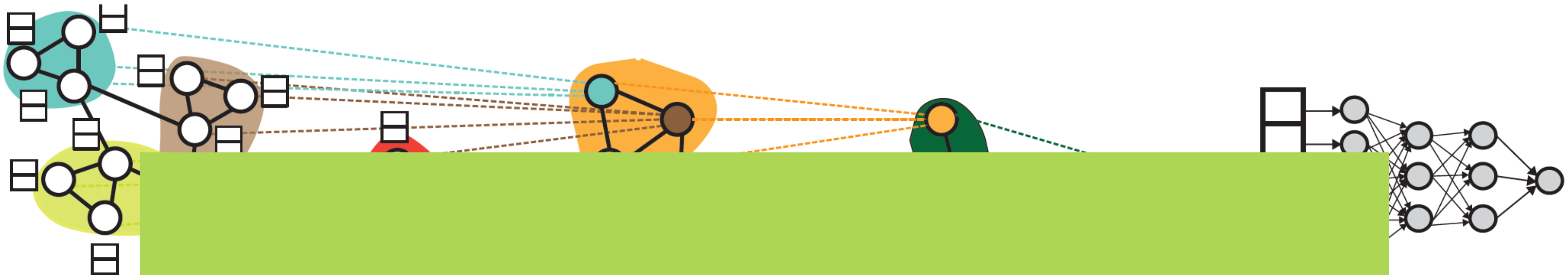
Problem: Learn how to hierarchical pool the nodes to embed the entire graph

Our solution: DIFFPOOL

- Learns hierarchical pooling strategy
- Sets of nodes are pooled hierarchically

[Hierarchical Graph Representation Learning with Differentiable Pooling](). R. Ying, et al. NeurIPS 2018.

Embedding Entire Graphs



How expressive are Graph Neural Networks?

Dor

Emb

Pro e

nod

Our solution: DIFFPOOL

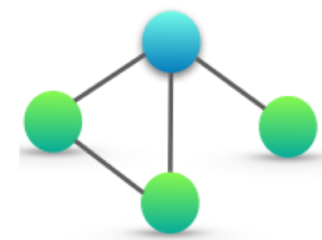
- Learns hierarchical pooling strategy
- Sets of nodes are pooled hierarchically

Hierarchical Graph Representation Learning with Differentiable Pooling. R. Ying, et al. NeurIPS 2018.

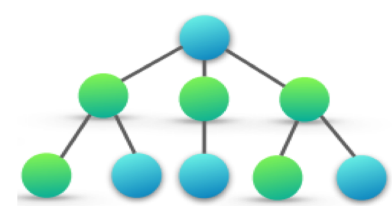
How expressive are GNNs?

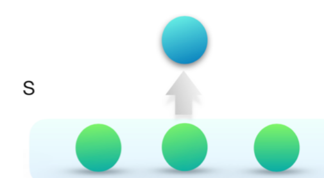
Theoretical framework: Characterize GNN's discriminative power:

- Characterize upper bound of the discriminative power of GNNs
- Propose a maximally powerful GNN
- Characterize discriminative power of popular GNNs



GNN tree:

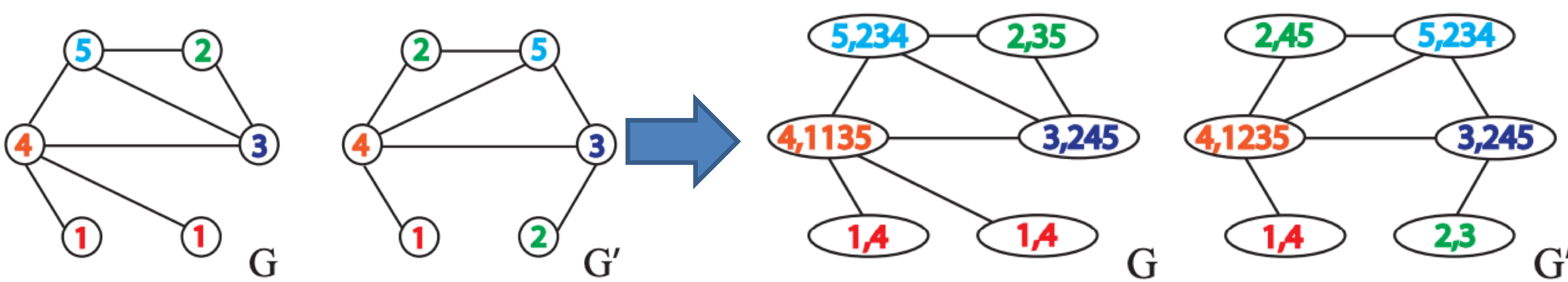


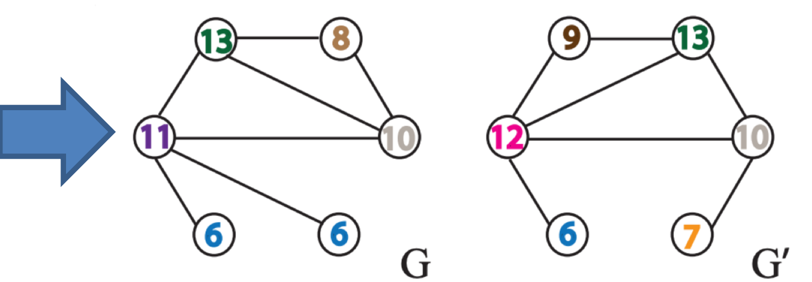


[How Powerful are Graph Neural Networks?](#) K. Xu, et al. ICLR 2019.

Discriminative Power of GNNs

Theorem: GNNs can be at most as powerful as the Weisfeiler-Lehman graph isomorphism test (a.k.a. canonical labeling or color refinement)



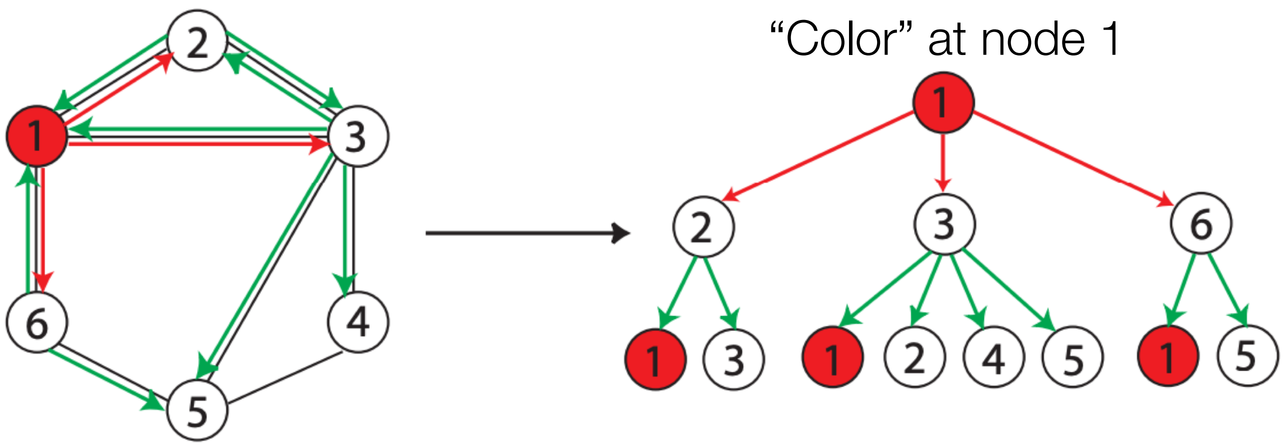


Color nodes by their degrees.
Aggregate colors of neighbors into a multiset.
Compress multisets into new colors.
Repeat n times or until colors in G and G' differ.

Discriminative Power of GNNs

Theorem: Power(GNNs) $\leq$ Power(WL)

Why?



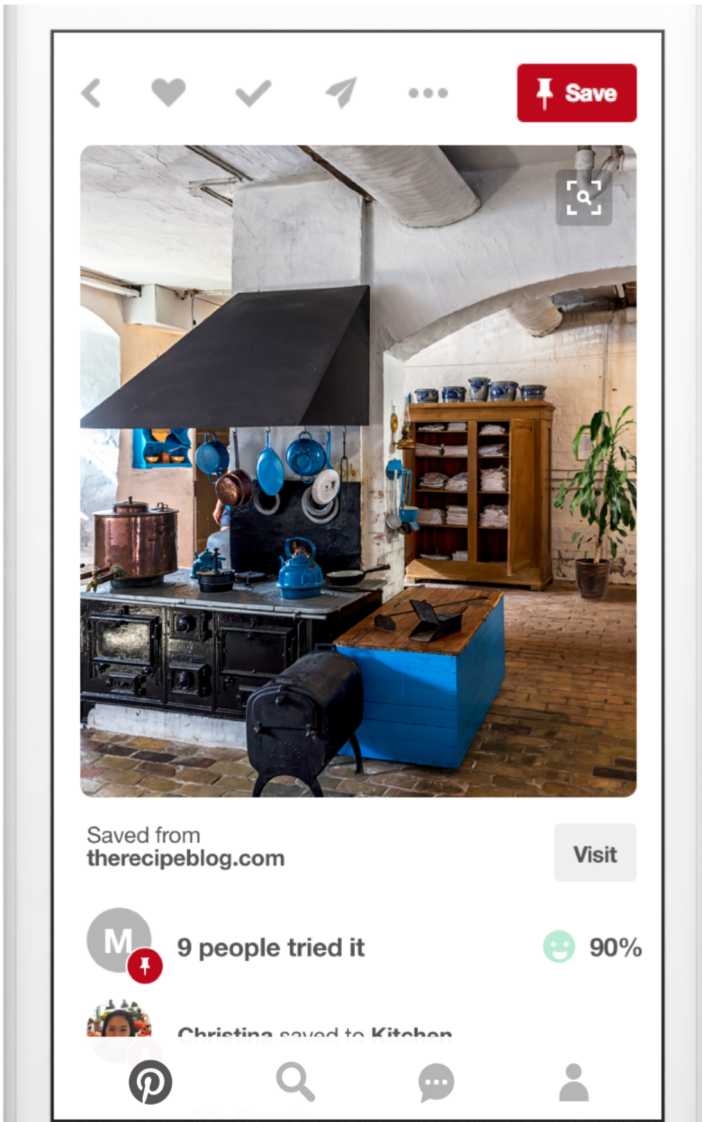
So, to distinguish 2 nodes, GNN needs to distinguish structure of their rooted subtrees

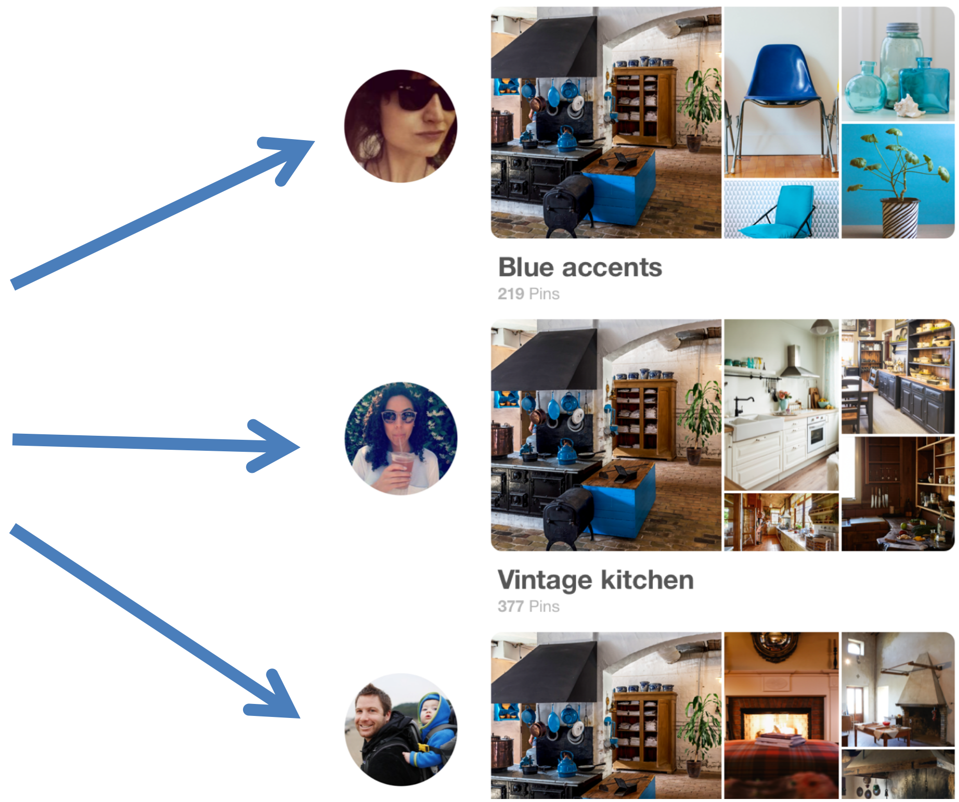
We develop GIN – provably most powerful GNN!

PinSAGE for Recommender Systems

[Graph Convolutional Neural Networks for Web-Scale Recommender Systems](). R. Ying, R. He, K. Chen, P. Eksombatchai, W. L. Hamilton, J. Leskovec. *KDD*, 2018.

Pinterest



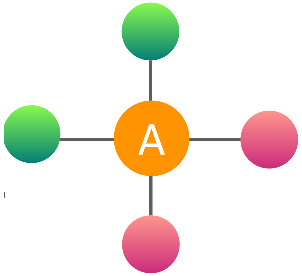


- 300M users
- 4+B pins, 2+B boards

Application: Pinterest

PinSage graph convolutional network:

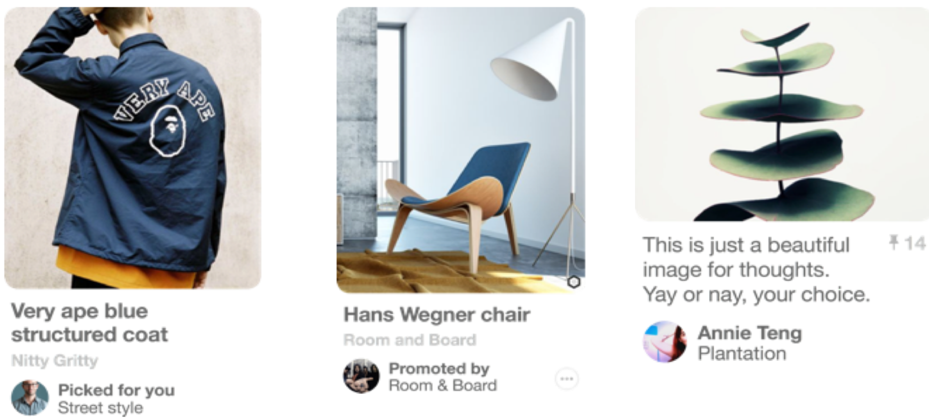
- **Goal:** Generate embeddings for nodes in a large-scale Pinterest graph containing billions of objects
- **Key Idea:** Borrow information from nearby nodes
  - E.g., bed rail Pin might look like a garden fence, but gates and beds are rarely adjacent in the graph



  - Pin embeddings are essential to various tasks like recommendation of Pins, classification, ranking
    - Services like “Related Pins”, “Search”, “Shopping”, “Ads”

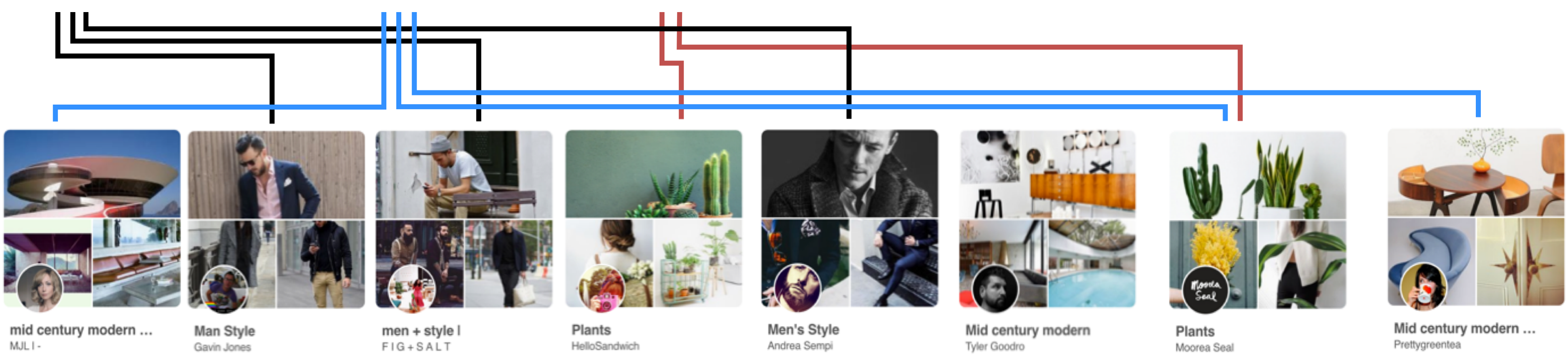
Pinterest Graph

Human curated collection of pins



Pins: Visual bookmarks someone has saved from the internet to a board they've created.

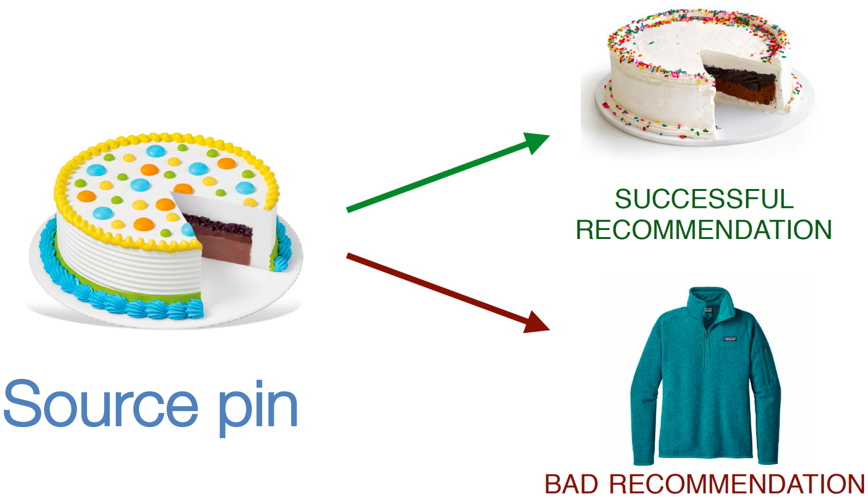
Pin features: Image, text, links



Boards

Pin Recommendation

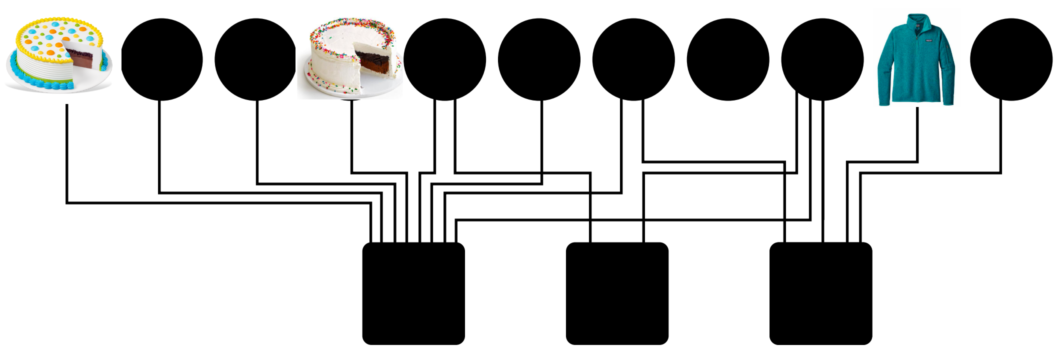
Task: Recommend related pins to users

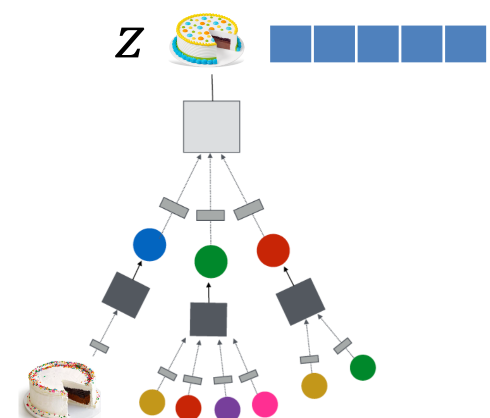


Task: Learn node embeddings z_i such that

$$d(z_{cake1}, z_{cake2}) < d(z_{cake1}, z_{sweater})$$

Predict whether two nodes in a graph are related





PinSAGE Training

Goal: Identify target pin among 3B pins

- Issue: Need to learn with resolution of 100 vs. 3B
- Massive size: 3 billion nodes, 20 billion edges
- Idea: Use harder and harder negative samples

Source pin

Positive

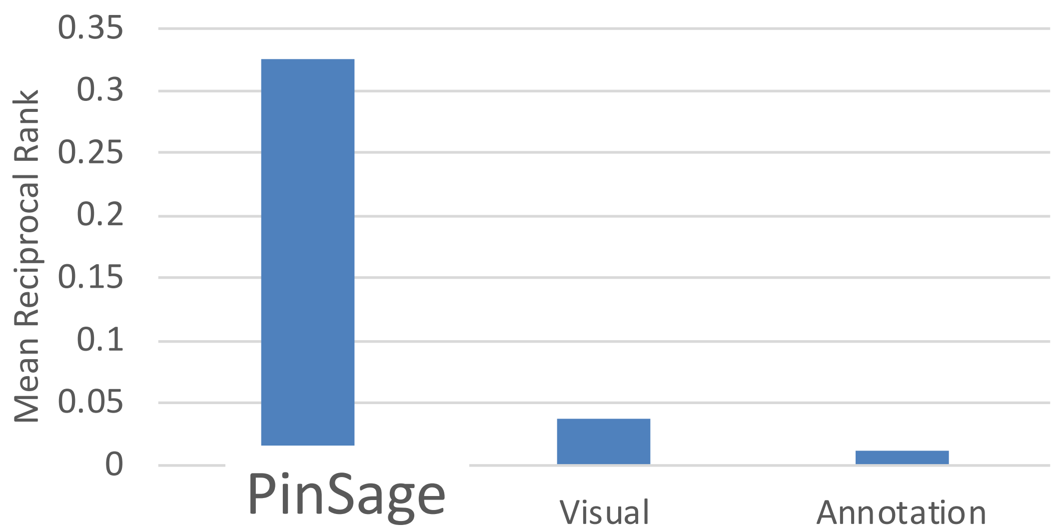

Easy negative

Hard negative

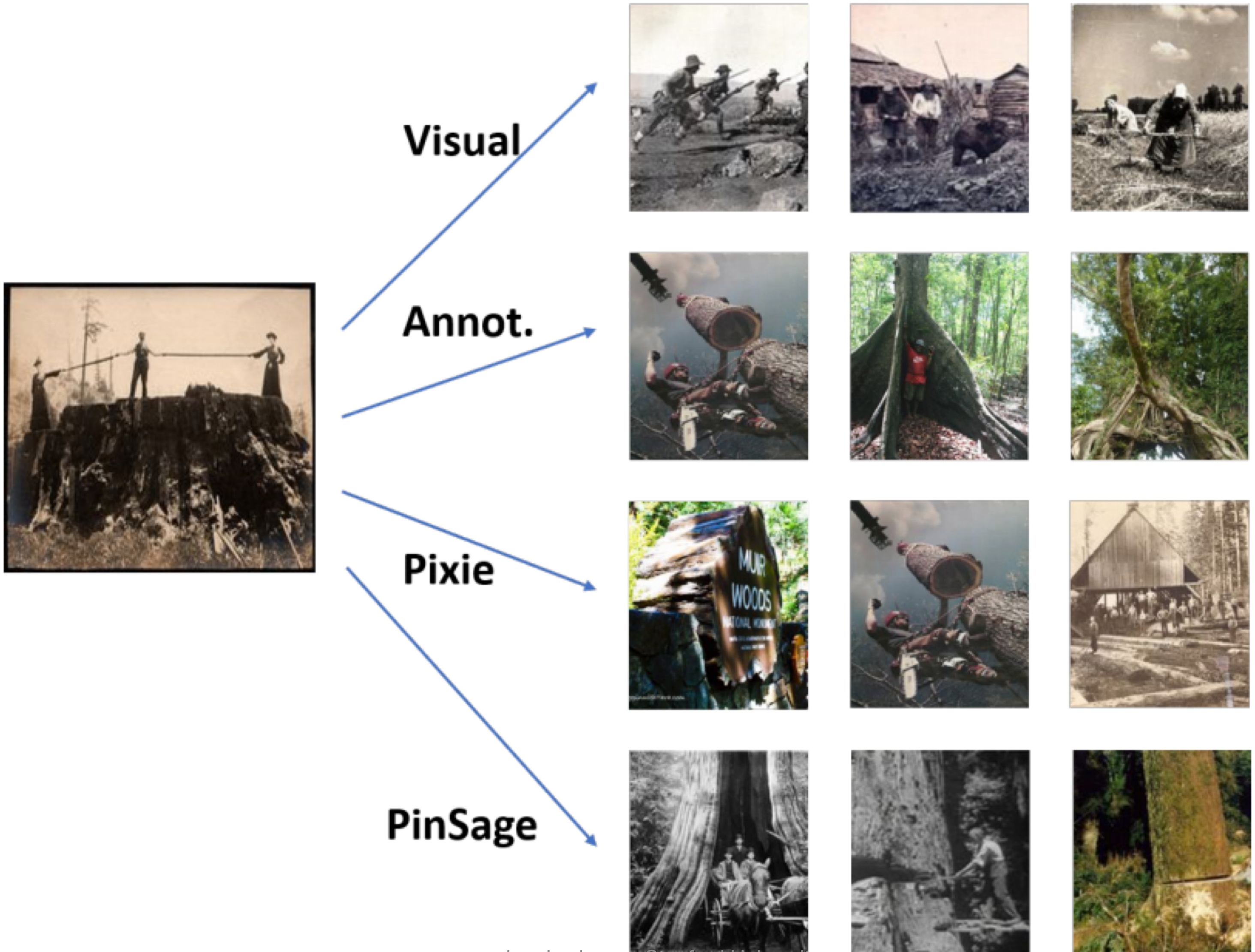
PinSAGE Performance

Related Pin recommendations

- Given a user is looking at pin **Q**, predict what pin **X** are they going to save next
- **Setup:** Embed 3B pins, perform nearest neighbor to generate recommendations



PinSAGE Example



Computational Drug Discovery: Drug Side Effect Prediction

Modeling Polypharmacy Side Effects with Graph Convolutional Networks. M. Zitnik, M. Agrawal, J. Leskovec. Bioinformatics, 2018.

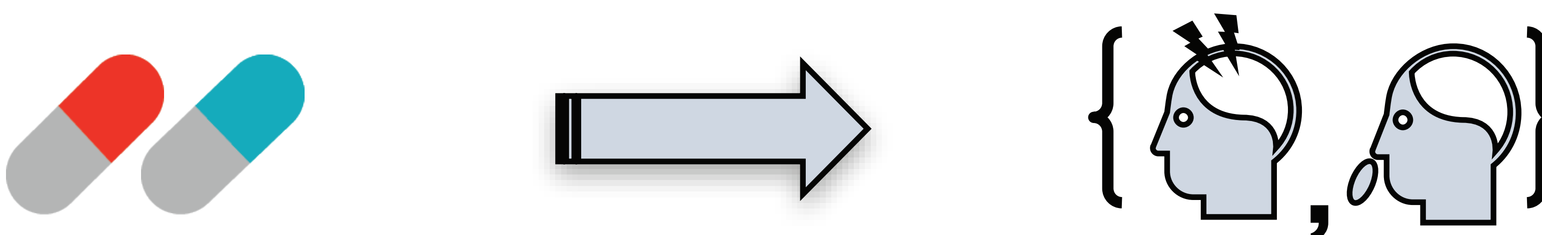
http://snap.stanford.edu/decagon/

Polypharmacy side effects

Many patients take multiple drugs to treat complex or co-existing diseases:

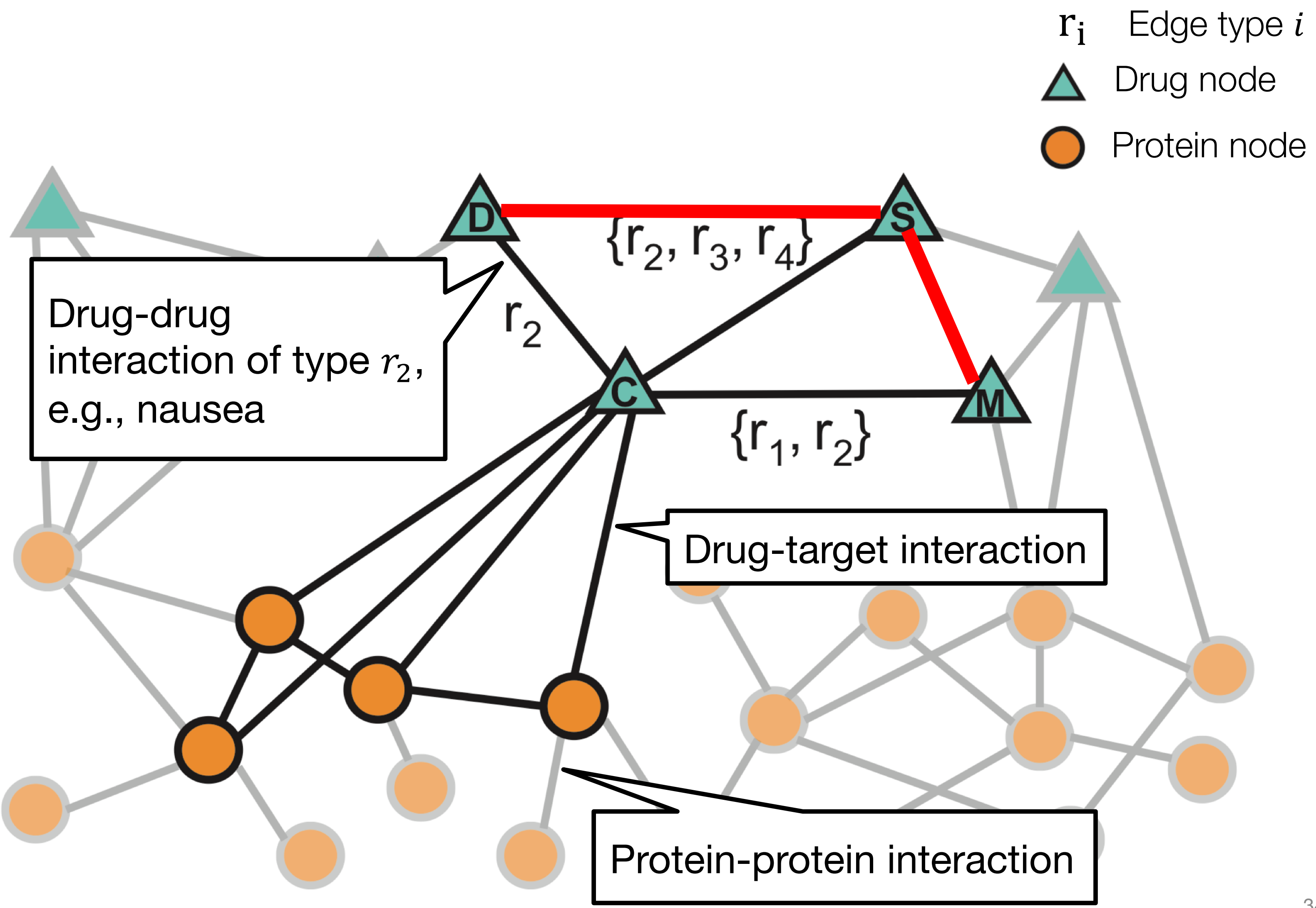
- 46% of people ages 70-79 take more than 5 drugs
- Many patients take more than 20 drugs to treat heart disease, depression, insomnia, etc.

Task: Given a pair of drugs predict adverse side effects



30% prob., 65% prob.

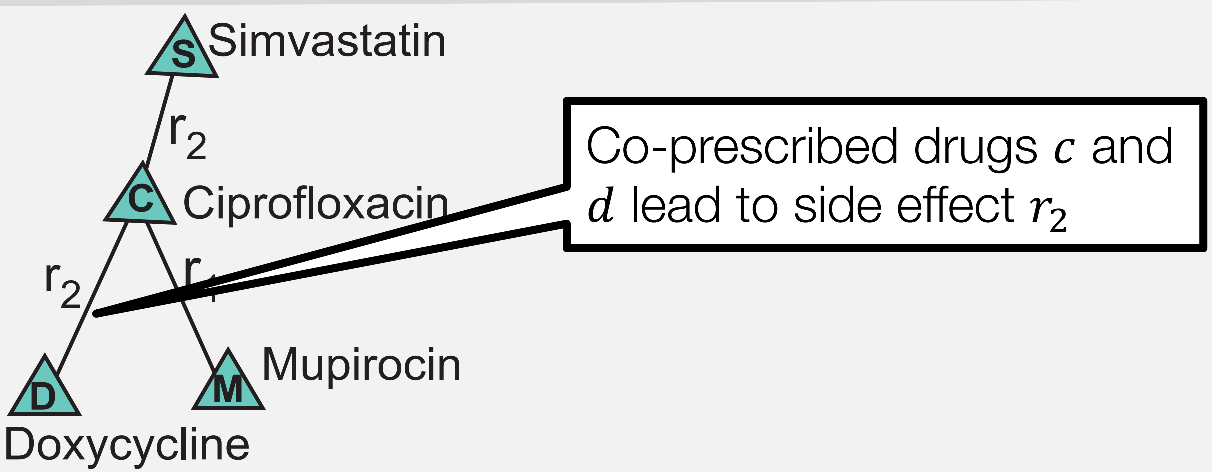
Approach: Build a Graph



Task: Link Prediction

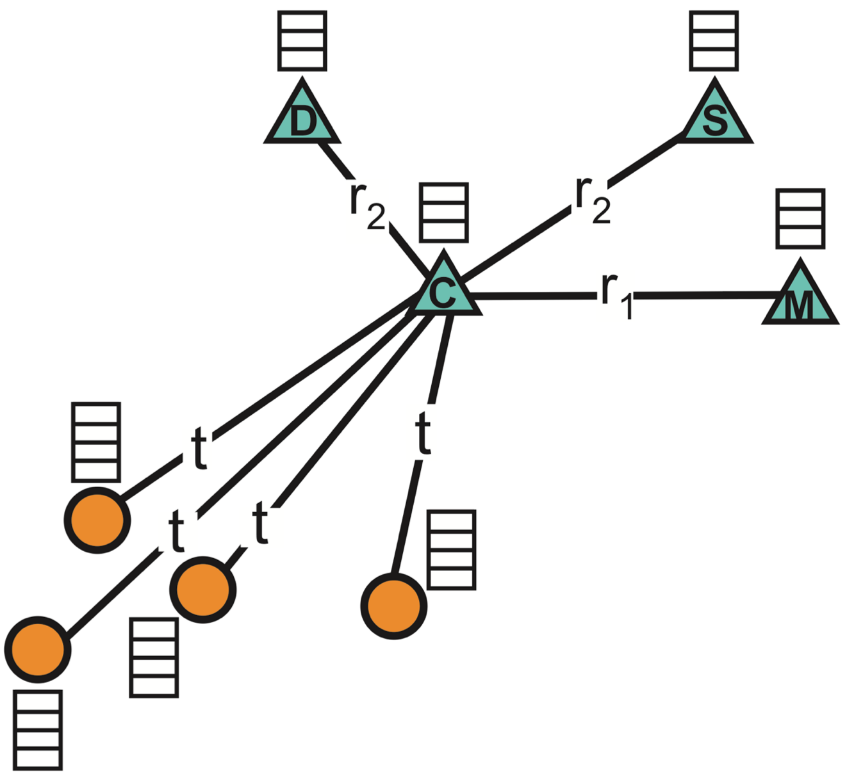
Task: Given a partially observed graph, predict labeled edges between drug nodes

Example query: Given drugs c, d, how likely is an edge (c, r_2, d)?

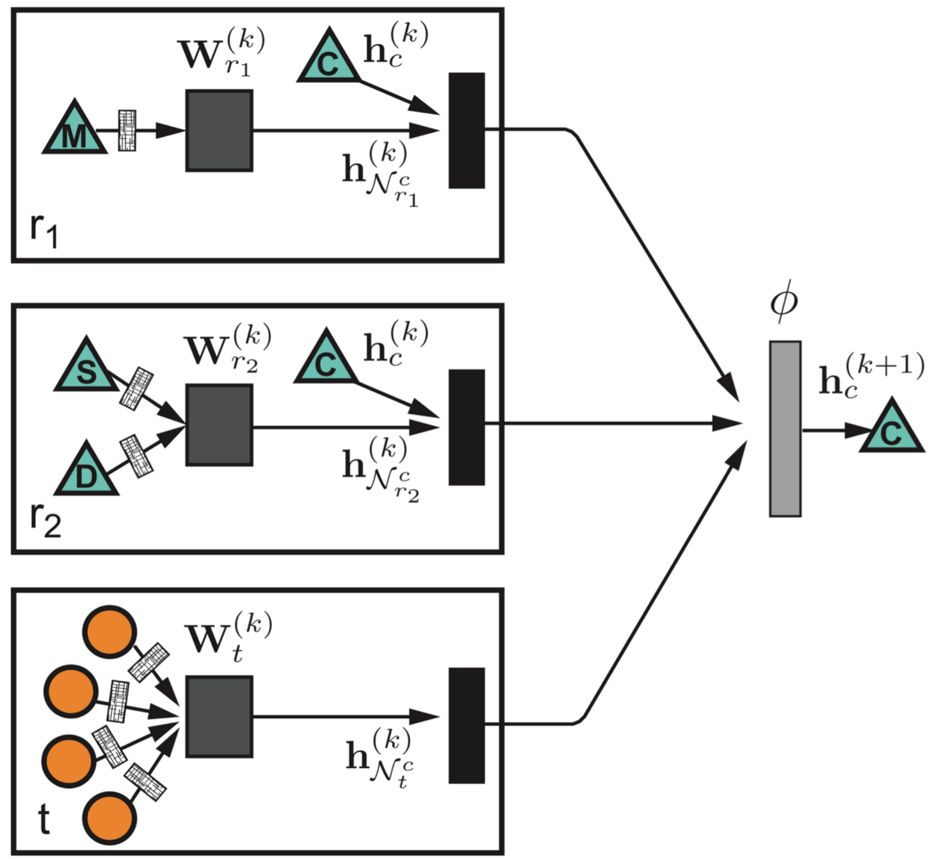


Decagon: Graph Neural Net

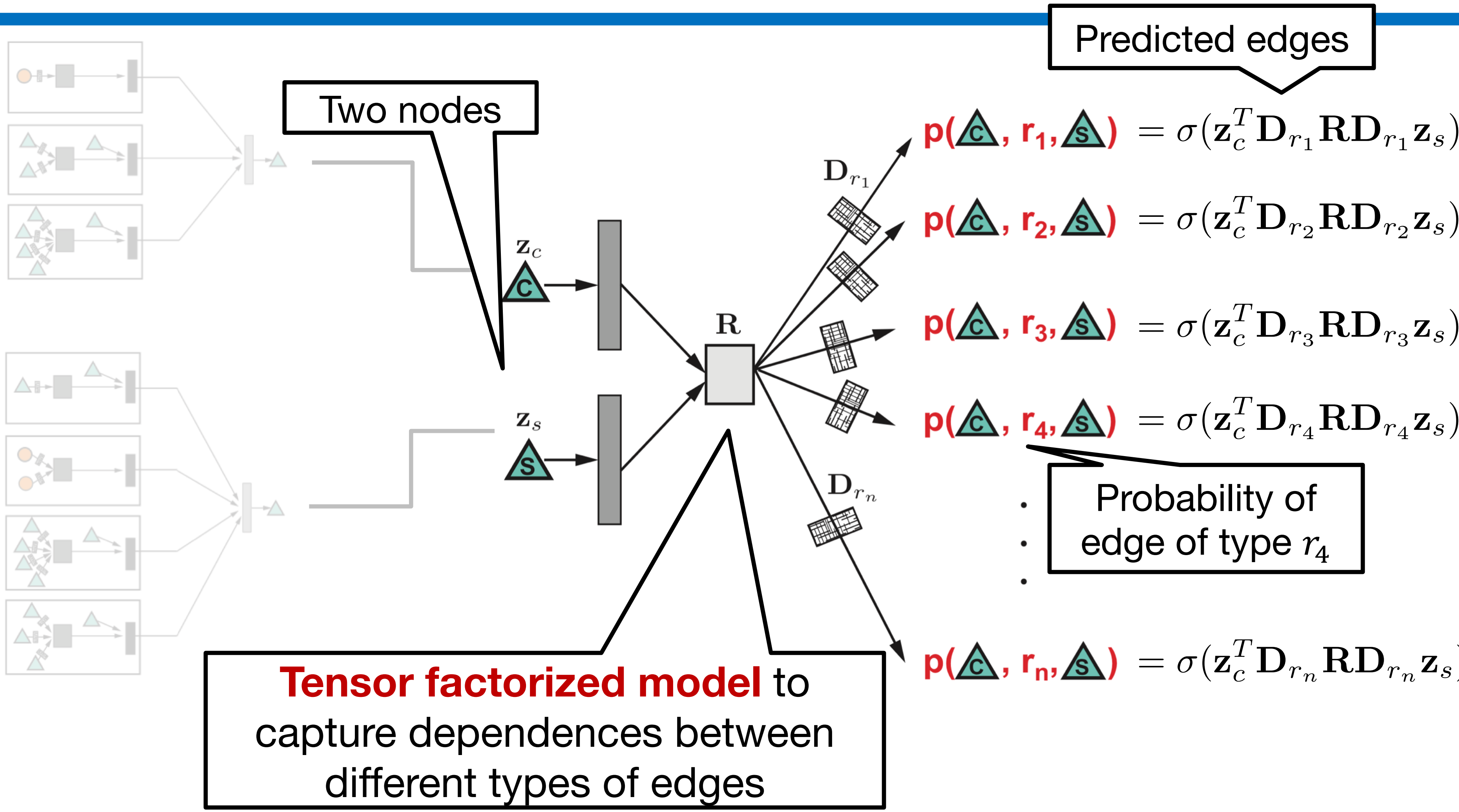
Network neighborhood of node C



Node C's computation graph

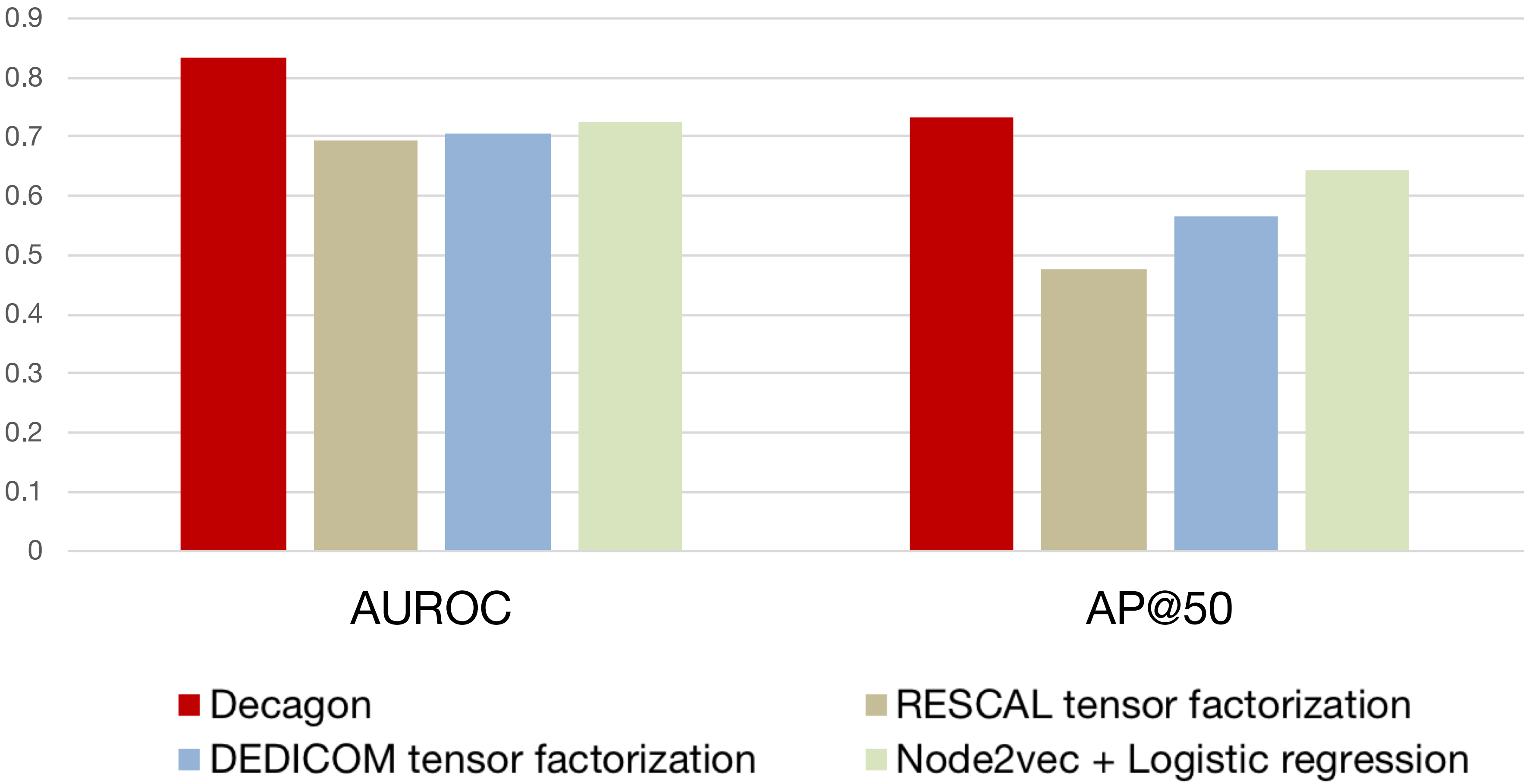


Decoder: Link Prediction



$\mathbf{R}, \mathbf{D}_{r_i}$ Parameter weight matrices

Results: Side Effect Prediction



36% average in AP@50 improvement over baselines

De novo Predictions

Rank	Drug c	Drug d	Side effect r
1	Pyrimethamine	Aliskiren	Sarcoma
2	Tigecycline	Bimatoprost	Autonomic neuropathy
3	Omeprazole	Dacarbazine	Telangiectases
4	Tolcapone	Pyrimethamine	Breast disorder
5	Minoxidil	Paricalcitol	Cluster headache
6	Omeprazole	Amoxicillin	Renal tubular acidosis
7	Anagrelide	Azelaic acid	Cerebral thrombosis
8	Atorvastatin	Amlodipine	Muscle inflammation
9	Aliskiren	Tioconazole	Breast inflammation
10	Estradiol	Nadolol	Endometriosis

De novo Predictions

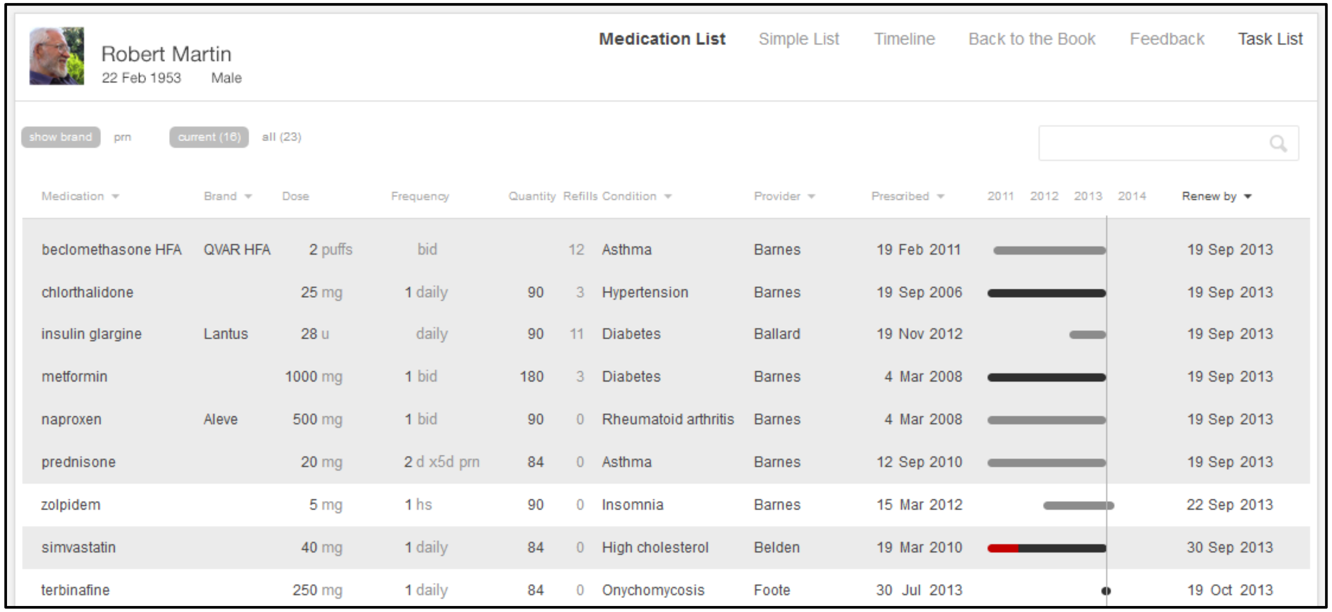
Rank	Drug c	Drug d	Side effect r	Evidence found
1	Pyrimethamine	Aliskiren	Sarcoma	Stage *et al.* 2015
2	Tigecycline	Bimatoprost	Autonomic neuropathy	
3	Omeprazole	Dacarbazine	Telangiectases	
4	Tolcapone	Pyrimethamine	Breast disorder	Bicker *et al.* 2017
5	Minoxidil	Paricalcitol	Cluster headache	
6	Omeprazole	Amoxicillin	Renal tubular acidosis	Russo *et al.* 2016
7	Anagrelide	Azelaic acid	Cerebral thrombosis	
8	Atorvastatin	Amlodipine	Muscle inflammation	Banakh *et al.* 2017
9	Aliskiren	Tioconazole	Breast inflammation	Parving *et al.* 2012
10	Estradiol	Nadolol	Endometriosis	

Case Report

Severe Rhabdomyolysis due to Presumed Drug Interactions between Atorvastatin with Amlodipine and Ticagrelor

Predictions in the Clinic

Clinical validation via drug-drug interaction markers, lab values, and



Medication	Brand	Dose	Frequency	Quantity	Refills	Condition	Provider	Prescribed	2011 2012 2013 2014	Renew by
beclomethasone HFA	QVAR HFA	2 puffs	bid		12	Asthma	Barnes	19 Feb 2011		19 Sep 2013
chlorthalidone		25 mg	1 daily	90	3	Hypertension	Barnes	19 Sep 2006		19 Sep 2013
insulin glargine	Lantus	28 u	daily	90	11	Diabetes	Ballard	19 Nov 2012		19 Sep 2013
metformin		1000 mg	1 bid	180	3	Diabetes	Barnes	4 Mar 2008		19 Sep 2013
naproxen	Aleve	500 mg	1 bid	90	0	Rheumatoid arthritis	Barnes	4 Mar 2008		19 Sep 2013
prednisone		20 mg	2 d x5d prn	84	0	Asthma	Barnes	12 Sep 2010		19 Sep 2013
zolpidem		5 mg	1 hs	90	0	Insomnia	Barnes	15 Mar 2012		22 Sep 2013
simvastatin		40 mg	1 daily	84	0	High cholesterol	Belden	19 Mar 2010		30 Sep 2013
terbinafine		250 mg	1 daily	84	0	Onychomycosis	Foote	30 Jul 2013		19 Oct 2013

NEWTON-WELLESLEY HOSPITAL

MASSACHUSETTS GENERAL HOSPITAL

Stanford MEDICINE

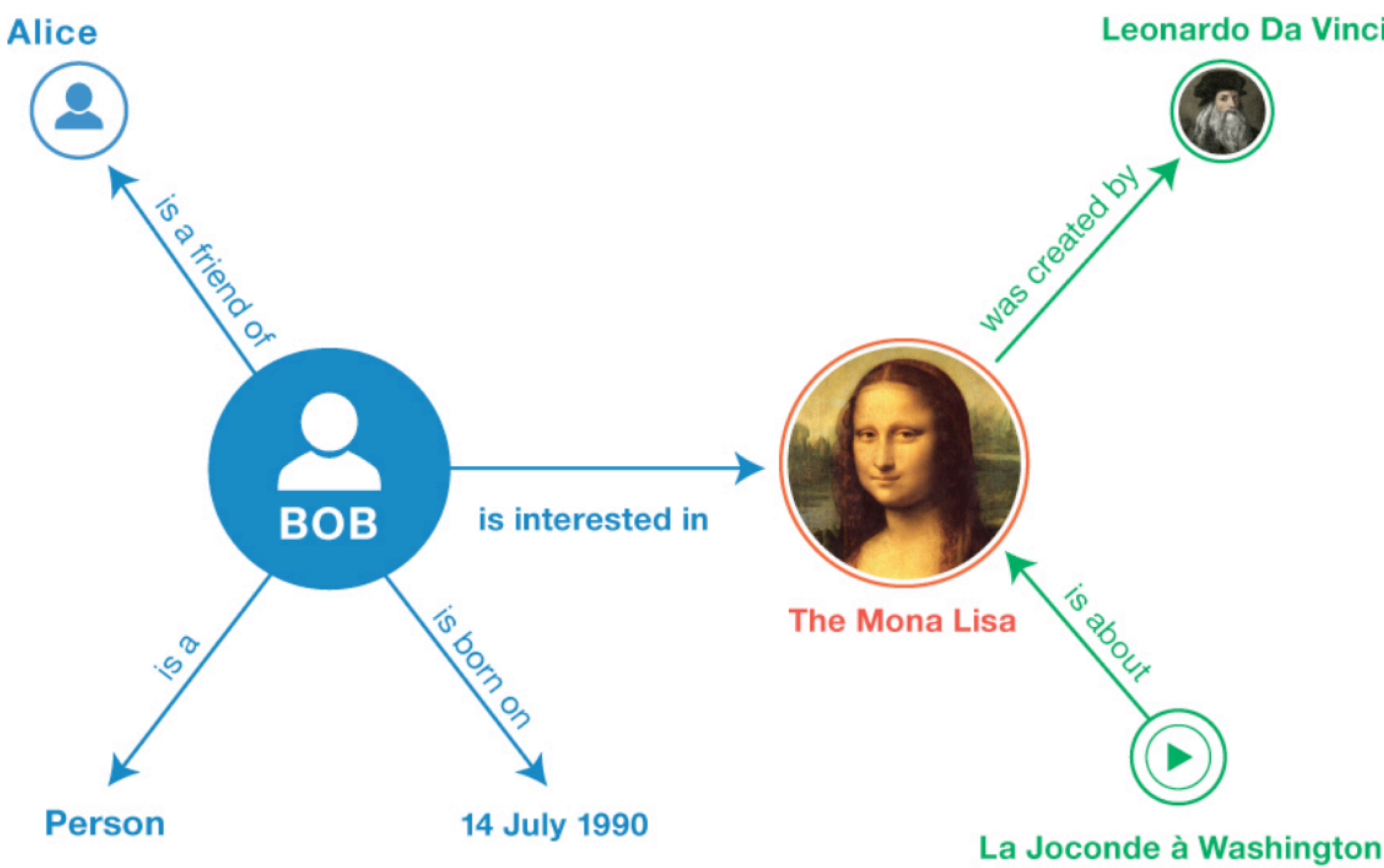
HARVARD MEDICAL SCHOOL

First method to **predict side effects of drug pairs,** even for drug combinations not yet used in patients

Reasoning in Knowledge Graphs

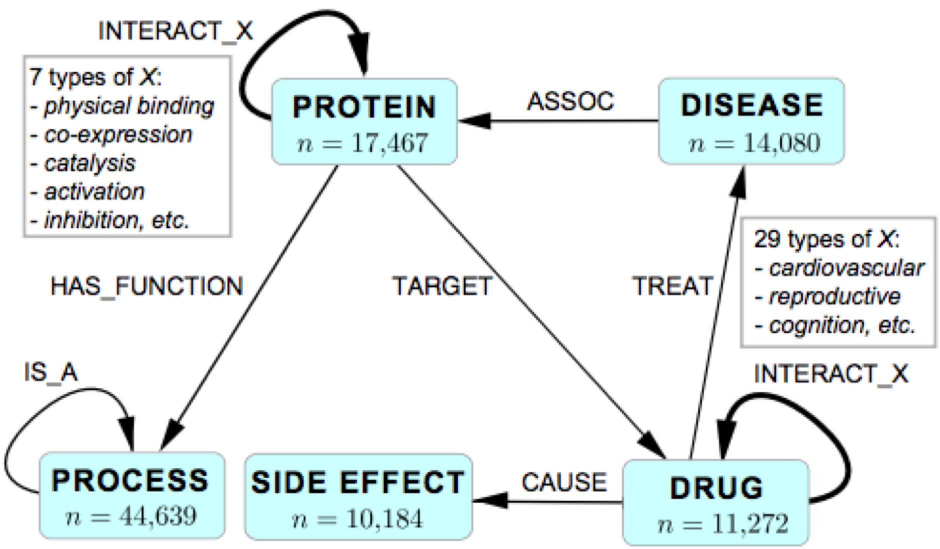
Embedding Logical Queries on Knowledge Graphs. W. Hamilton, P. Bajaj, M. Zitnik, D. Jurafsky, J. Leskovec. *Neural Information Processing Systems (NIPS)*, 2018.

Knowledge as a Graph

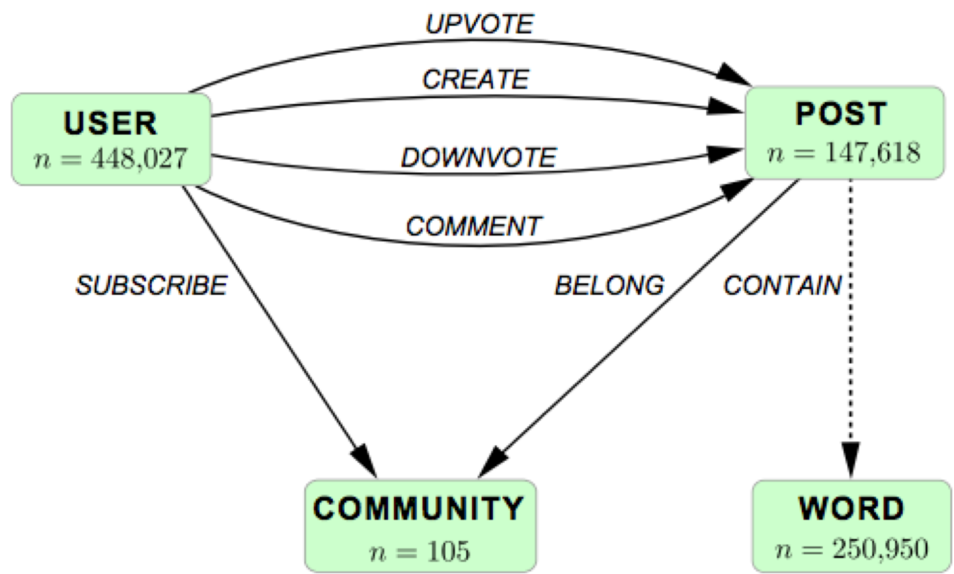


Knowledge Graph

Heterogeneous Knowledge Graphs



Biological interactions



Online communities

Conjunctive Graph Queries

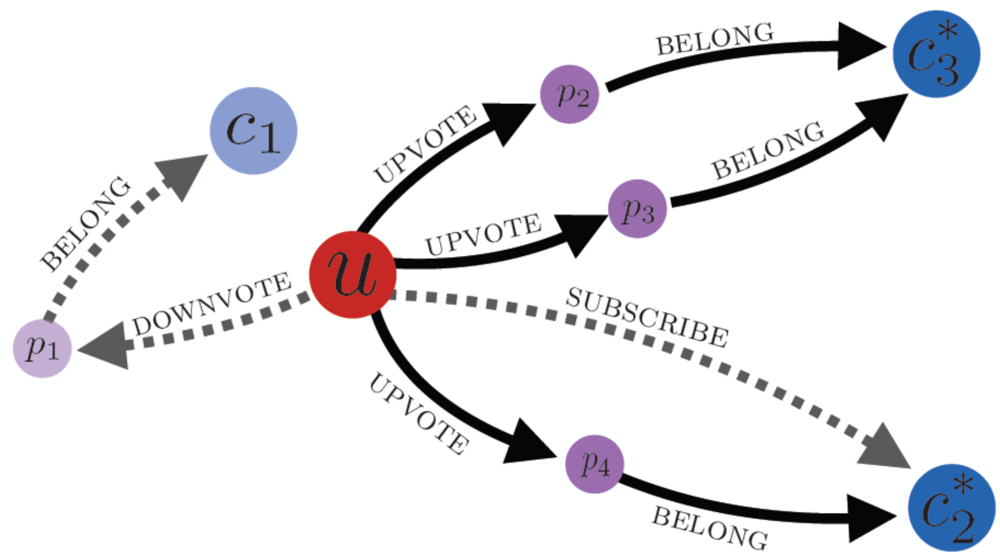
Query formula

$C_?.\exists P : \textsc{upvote}(u, P) \wedge \textsc{belong}(P, C_?)$

"Predict communities $C_?$ in which user u is likely to upvote a post"

Query DAG

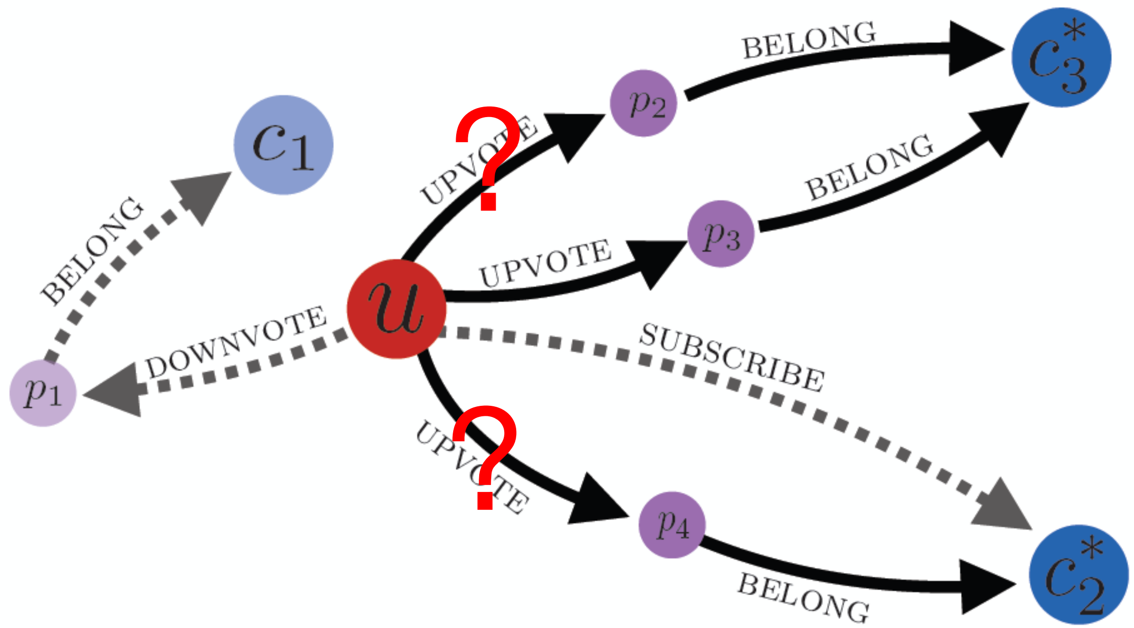
Example subgraphs that satisfy the query



Predictive Graph Queries

Key challenges: Big graphs and queries can involve noisy and unobserved data!

Some links might be noisy or unobserved or haven't occurred yet



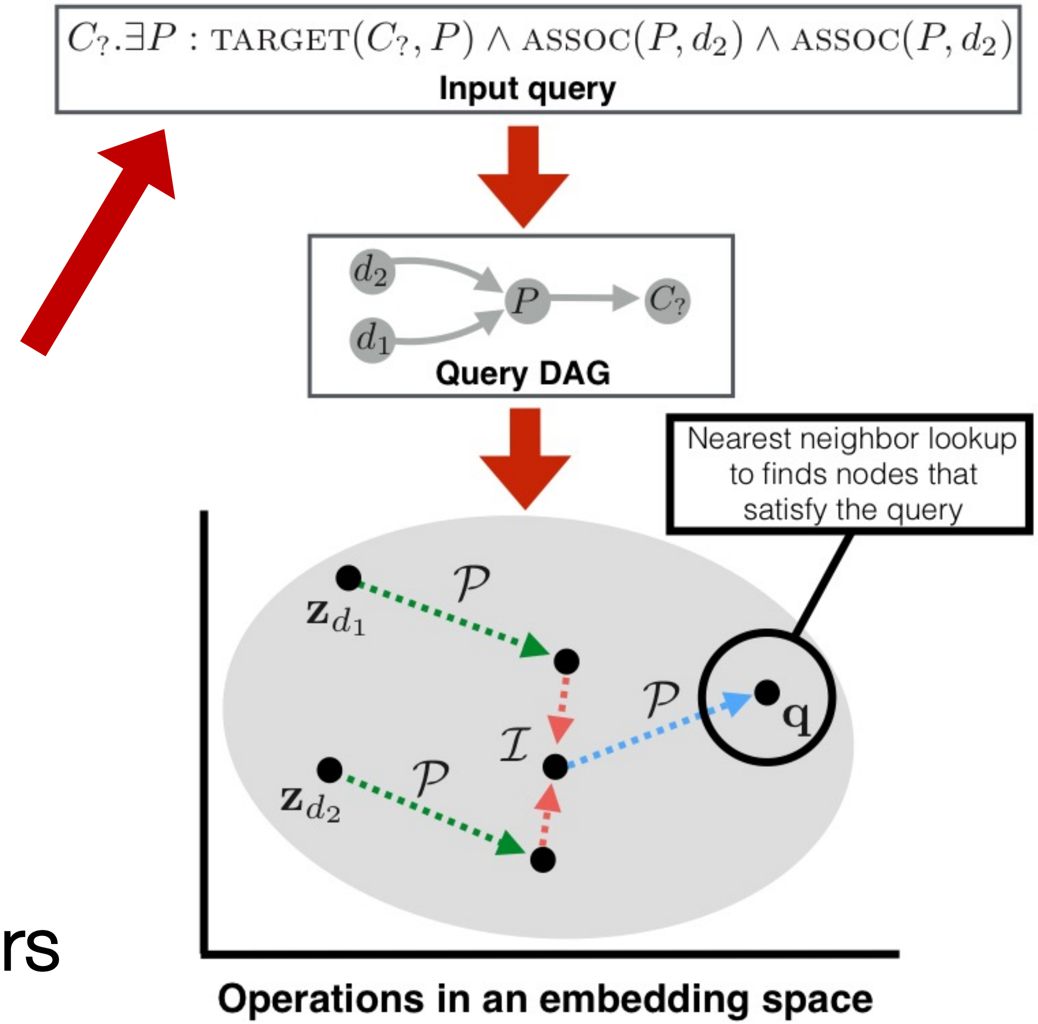
Problem: Naïve link prediction and graph template matching are too expensive

Overview of Our Framework

Goal: Answer complex logical queries

E.g.: "**Predict drugs C** likely *target* proteins P *associated* with diseases d_1 and d_2"

Idea: Logical operators become spatial operators



Model Specification

τ… edge type
γ… node type
R_τ… matrix
W_γ… matrix
Ψ… aggregator
NN… neural net

Given: Knowledge graph

Find:

- Node embeddings
- Projection operator P: $P(q, \tau) = R_\tau \cdot z_q$
  - Applies transition R_τ of relation τ to q
- Intersection operator I: $I(q_{1\ldots n}) = W_\gamma \cdot \mathrm{AGG}_{\mathrm{j=1\ldots n}}(\mathrm{NN}(\mathrm{q_i}))$
  - Set intersection in the embedding space

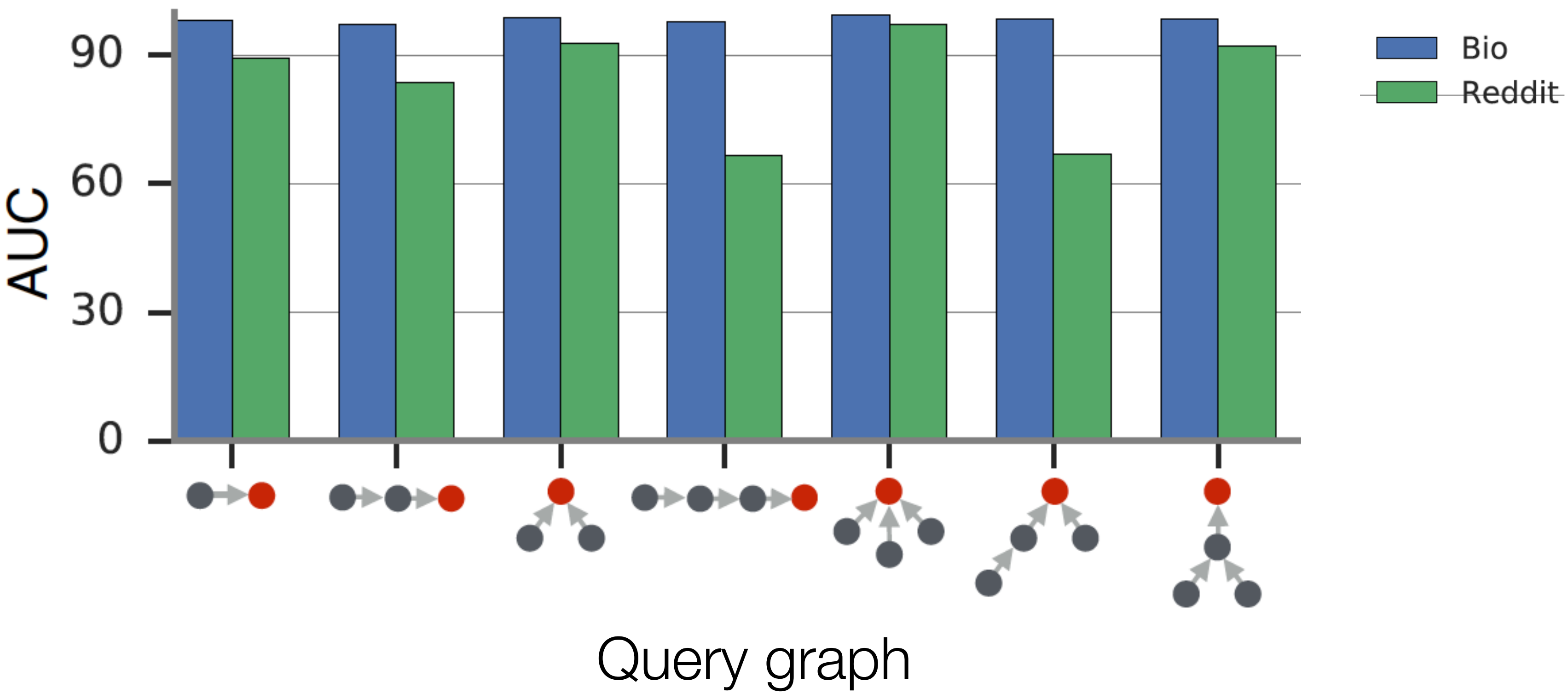
Model Training

Training examples: Queries on the graph

- Positives: Path with a known answer
- "Standard" negatives: Random nodes of the correct answer type
- "Hard" negatives: Correct answers if a logical conjunction is relaxed to a disjunction
- Loss: $\mathcal{L}(q) = \max\left(0, 1 - \mathtt{score}(\mathbf{q}, \mathbf{z}_{v^*}) + \mathtt{score}(\mathbf{q}, \mathbf{z}_{v_N})\right)$

Performance

- Performance on different query types:



How can this technology be used for other problems?

We can now apply neural networks much more broadly

New frontiers beyond classic neural networks that learn on images and sequences

Many other applications:

- **Nodes:** Predict tissue-specific protein functions
- **Subgraphs:** Predict which drug treats what disease
- **Graphs:** Predict properties of molecules/drugs

Summary

- Graph Convolutional Neural Networks
  - Generalize beyond simple convolutions
- Fuses node features & graph info
  - State-of-the-art accuracy for node classification and link prediction
- Model size independent of graph size; can scale to billions of nodes
  - Largest embedding to date (3B nodes, 20B edges)
- Leads to significant performance gains

Conclusion

Results from the past 2-3 years have shown:

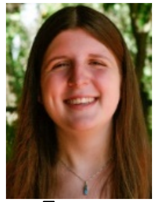
- Representation learning paradigm can be extended to graphs
- No feature engineering necessary
- Can effectively combine node attribute data with the network information
- State-of-the-art results in a number of domains/tasks
- Use end-to-end training instead of multi-stage approaches for better performance

PhD Students

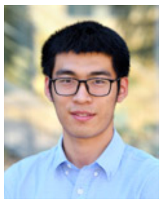

Alexandra Porter


Camilo Ruiz


Claire Donnat

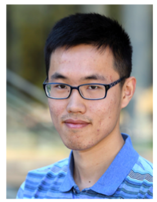

Emma Pierson


Weihua Hu


Jiaxuan You


Bowen Liu

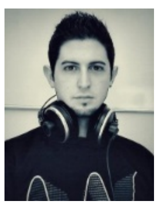

Mohit Tiwari

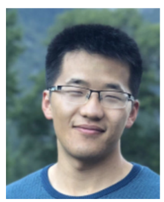

Rex Ying

Post-Doctoral Fellows


Baharan Mirzasoleiman


Marinka Zitnik


Michele Catasta

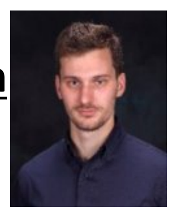

Pan Li


Shantao Li


Srijan Kumar


Hingwei Wang

Research Staff


Adrijan Bradaschia


Rok Sosic

Industry Partnerships

Funding

Collaborators

Dan Jurafsky, Linguistics, Stanford University
David Grusky, Sociology, Stanford University
Stephen Boyd, Electrical Engineering, Stanford University
David Gleich, Computer Science, Purdue University
VS Subrahmanian, Computer Science, University of Maryland
Sarah Kunz, Medicine, Harvard University
Russ Altman, Medicine, Stanford University
Jochen Profit, Medicine, Stanford University
Eric Horvitz, Microsoft Research
Jon Kleinberg, Computer Science, Cornell University
Sendhill Mullainathan, Economics, Harvard University
Scott Delp, Bioengineering, Stanford University
James Zou, Medicine, Stanford University

References

- Tutorial on Representation Learning on Networks at WWW 2018 http://snap.stanford.edu/proj/embeddings-www/
- Inductive Representation Learning on Large Graphs. W. Hamilton, R. Ying, J. Leskovec. NIPS 2017.
- Representation Learning on Graphs: Methods and Applications. W. Hamilton, R. Ying, J. Leskovec. IEEE Data Engineering Bulletin, 2017.
- Graph Convolutional Neural Networks for Web-Scale Recommender Systems. R. Ying, R. He, K. Chen, P. Eksombatchai, W. L. Hamilton, J. Leskovec. KDD, 2018.
- Modeling Polypharmacy Side Effects with Graph Convolutional Networks. M. Zitnik, M. Agrawal, J. Leskovec. Bioinformatics, 2018.
- Graph Convolutional Policy Network for Goal-Directed Molecular Graph Generation. J. You, B. Liu, R. Ying, V. Pande, J. Leskovec, NeurIPS 2018.
- Embedding Logical Queries on Knowledge Graphs. W. Hamilton, P. Bajaj, M. Zitnik, D. Jurafsky, J. Leskovec. NeuIPS, 2018.
- How Powerful are Graph Neural Networks? K. Xu, W. Hu, J. Leskovec, S. Jegelka. ICLR 2019.

- Code:
  - http://snap.stanford.edu/graphsage
  - http://snap.stanford.edu/decagon/
  - https://github.com/bowenliu16/rl_graph_generation
  - https://github.com/williamleif/graphqembed
  - https://github.com/snap-stanford/GraphRNN