Part 2: Next-Gen Machine Learning for Network Biology

Marinka Zitnik
Stanford University

Two Lectures

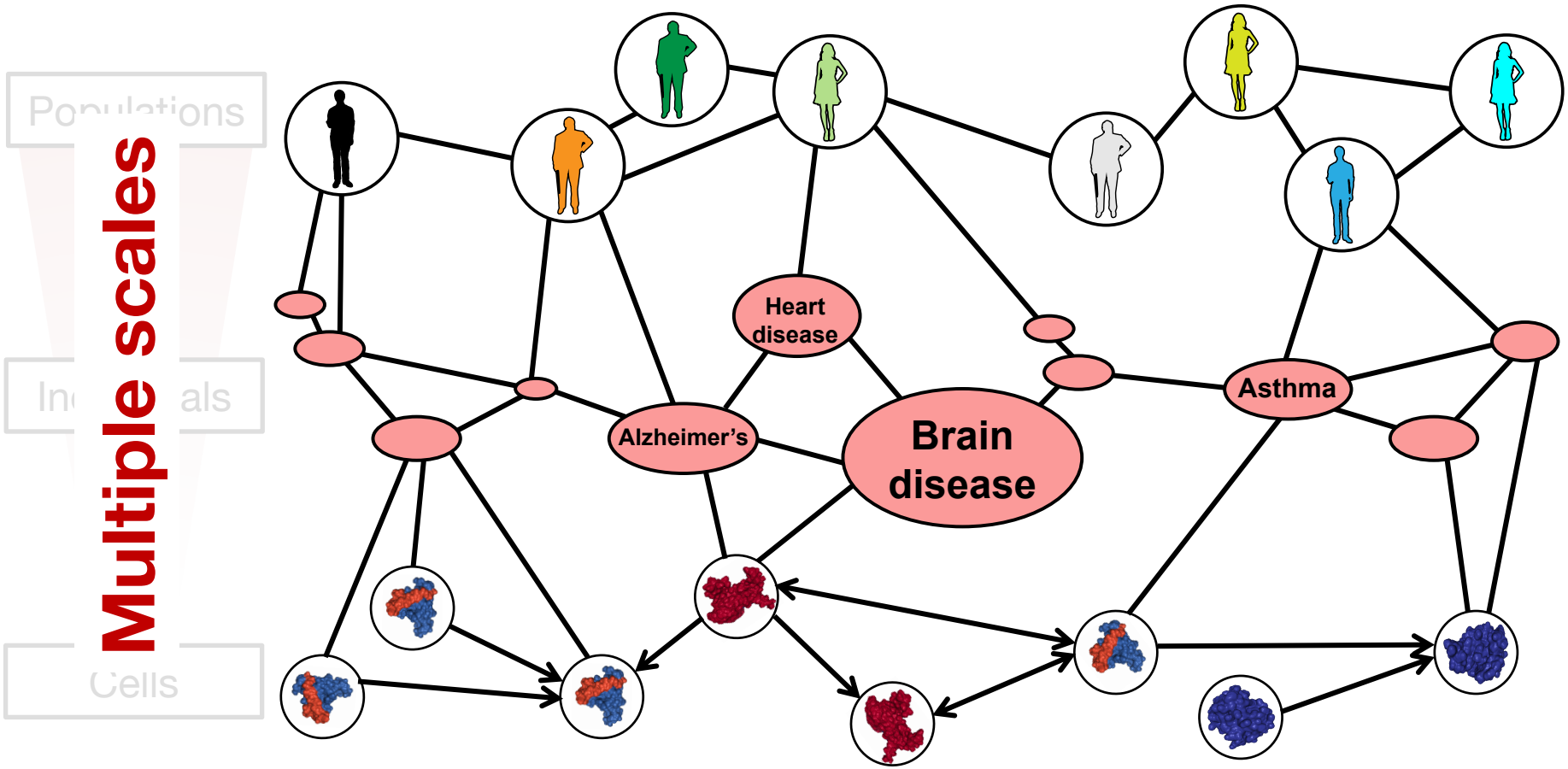
Part 1: May 15, 2019, 2:30 pm - 4:00 pm

- Methodology: Shallow network embeddings:
  - Map nodes to low-dimensional features
- Resources: Data, tools, codebases
- Applications: PPIs, Disease pathways, Tissues

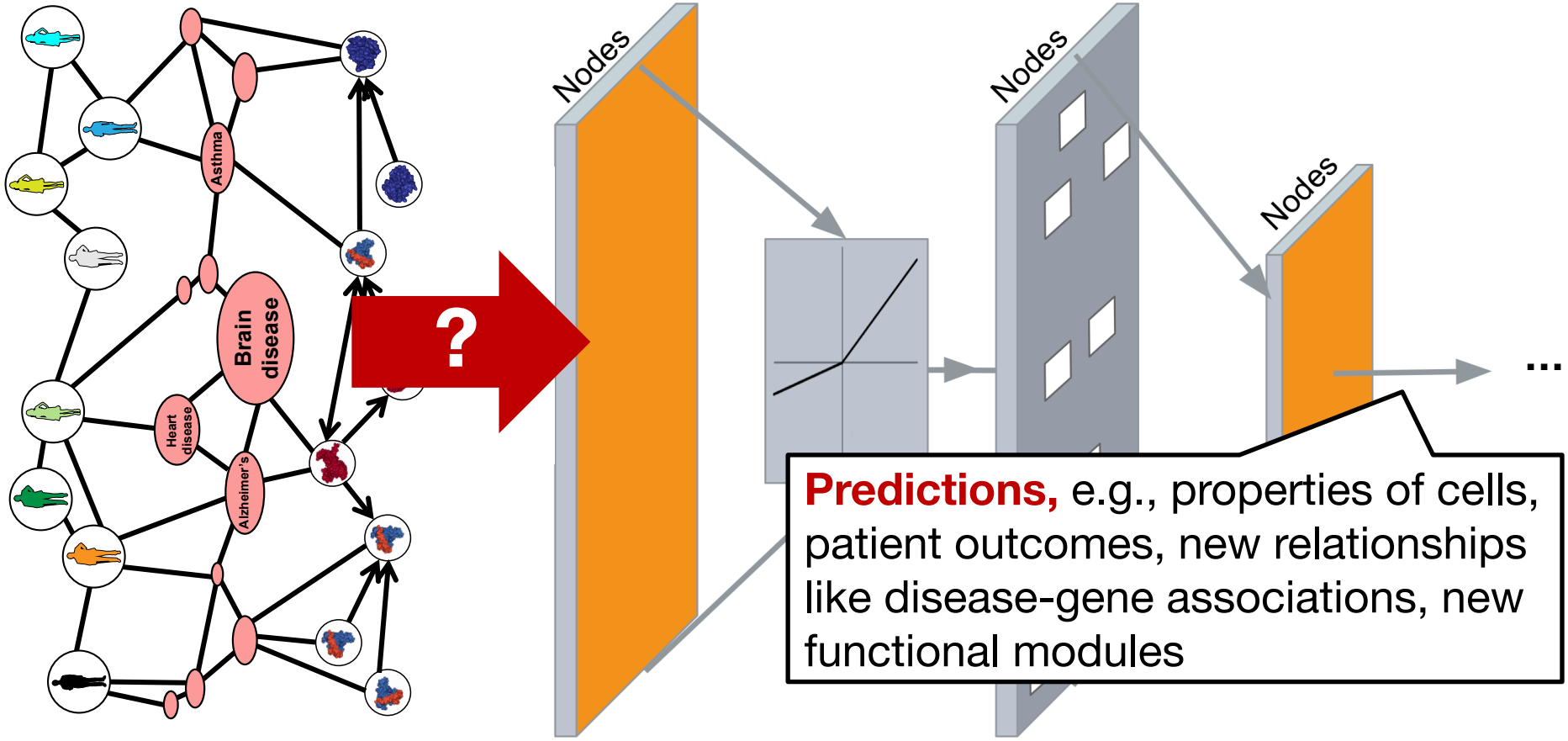
Part 2: May 16, 2019, 9:00 am – 10:30 am

- Methodology: Deep network embeddings:
  - Graph neural networks for rich biomedical graphs
- Resources: Data, practical advice and demos
- Applications: Polypharmacy, Drug repurposing

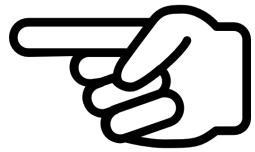
Networks allow for integration of biomedical data



How to learn deep models on biomedical networks?



Outline of this Lecture

1) Deep Graph Neural Networks 

2) Polypharmacy & Drug Interactions

3) Drug Repurposing

4) New Directions and Opportunities

5) Practical Advice and Demos

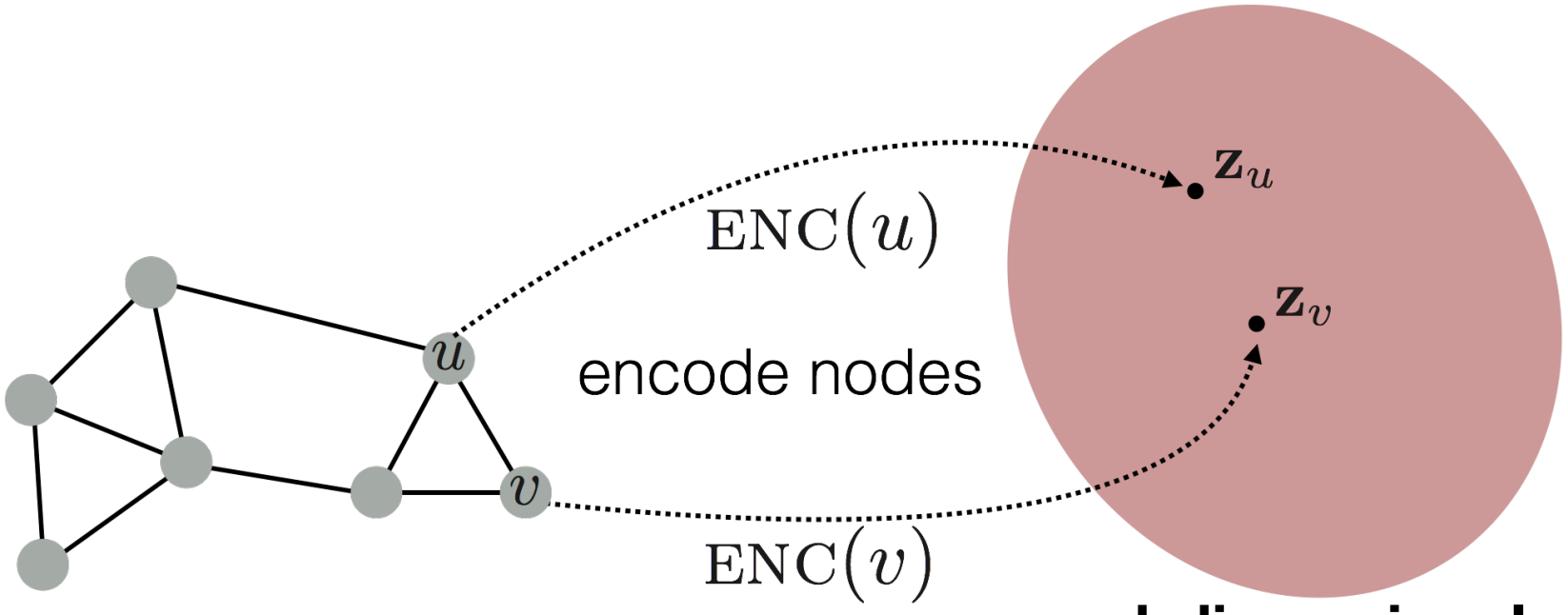
Deep Learning for Multimodal Networks

Based on material from:

- Zitnik et al. 2018. Deep Learning for Network Biology. *ISMB.*
- Zitnik et al. 2018. Modeling Polypharmacy Side Effects with Graph Convolutional Networks. *ISMB & Bioinformatics.*

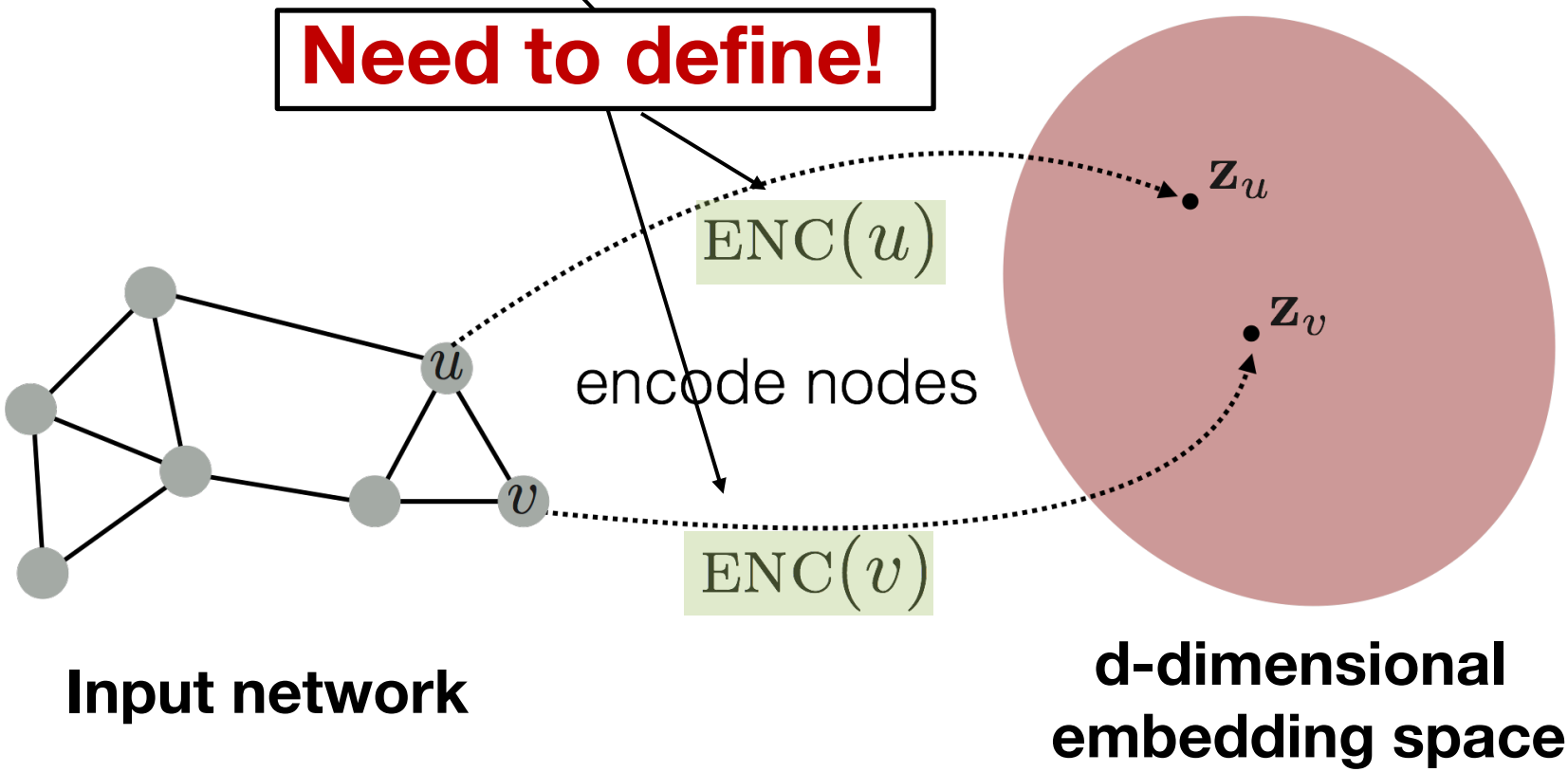
Embedding Nodes

Goal: Map nodes so that **similarity in the embedding space** (e.g., dot product) approximates **similarity in the network**



Embedding Nodes

Goal: $\text{similarity}(u, v) \approx \mathbf{z}_v^\top \mathbf{z}_u$



Two Key Components

- **Encoder:** Map a node to a low-dimensional vector:

$$\text{ENC}(v) = \mathbf{z}_v$$

$\mathbf{d}$-dimensional embedding

node in the input graph

- **Similarity function** defines how relationships in the input network map to relationships in the embedding space:

$$\text{similarity}(u, v) \approx \mathbf{z}_v^\top \mathbf{z}_u$$

Similarity of u and v in the network

dot product between node embeddings

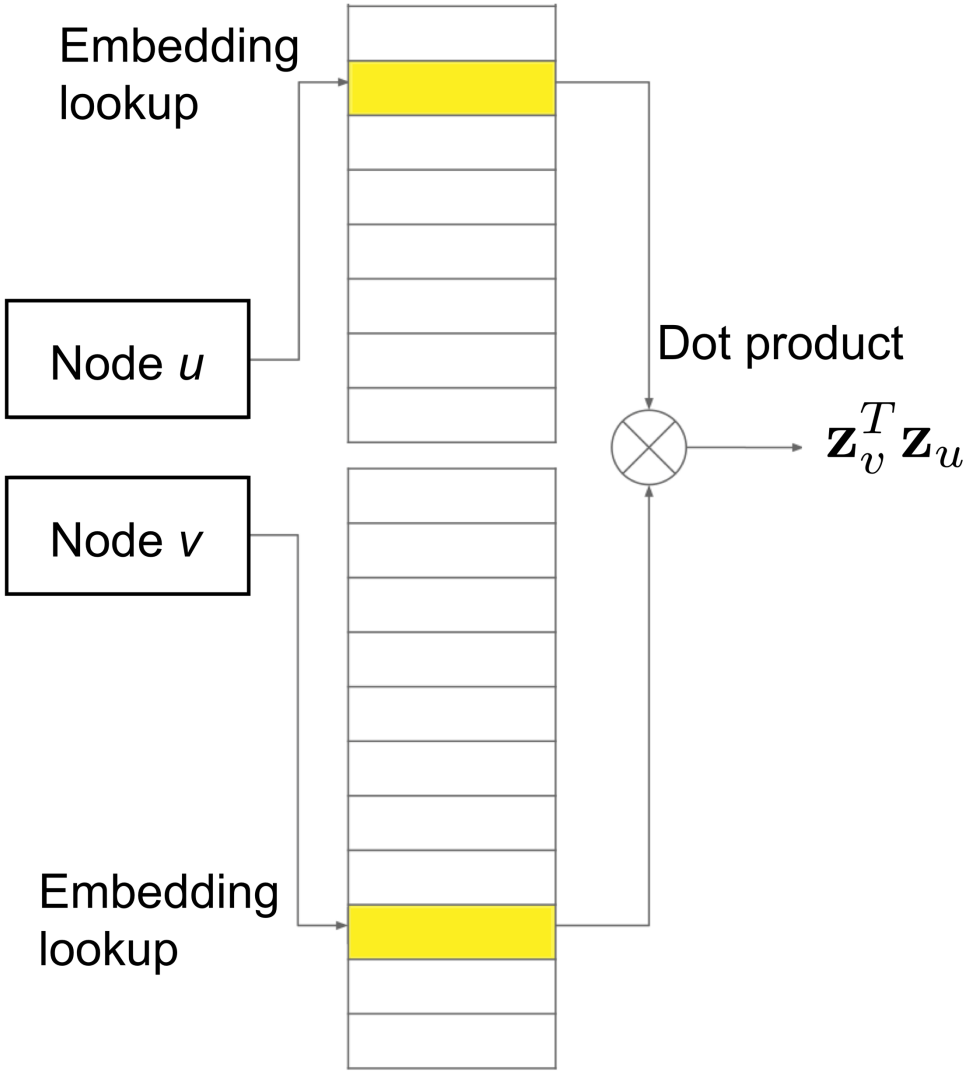
So Far: Shallow Encoders

Shallow encoders:

- One-layer of data transformation

- A single hidden layer maps node u to embedding $\boldsymbol{z}_u$ via function f:

$$\boldsymbol{z}_u = f\left(\boldsymbol{z}_v, v \in N_R(u)\right)$$



Shallow Encoders

Limitations of shallow encoding:

- **O(|V|) parameters are needed**:
  - No sharing of parameters between nodes
  - Every node has its own unique embedding
- **Inherently "transductive"**:
  - Cannot generate embeddings for nodes not seen during training
- **Do not incorporate node features, extra information**:
  - Many graphs are rich, have features for nodes/edges that we can and should leverage

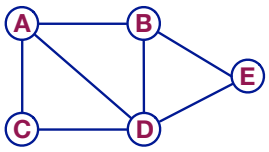
Deep Graph Encoders

Next: We discuss deep methods based on **graph neural networks:**

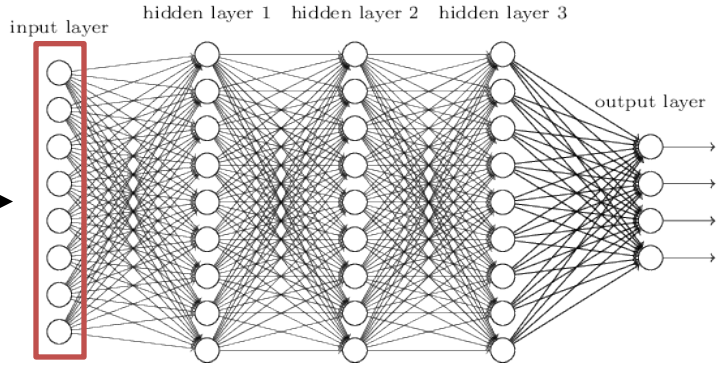
$$\text{ENC}(v) = \text{multiple layers of non-linear transformation of graph structure}$$

A Naïve Approach

- Join adjacency matrix and features
- Feed them into any classic neural net:



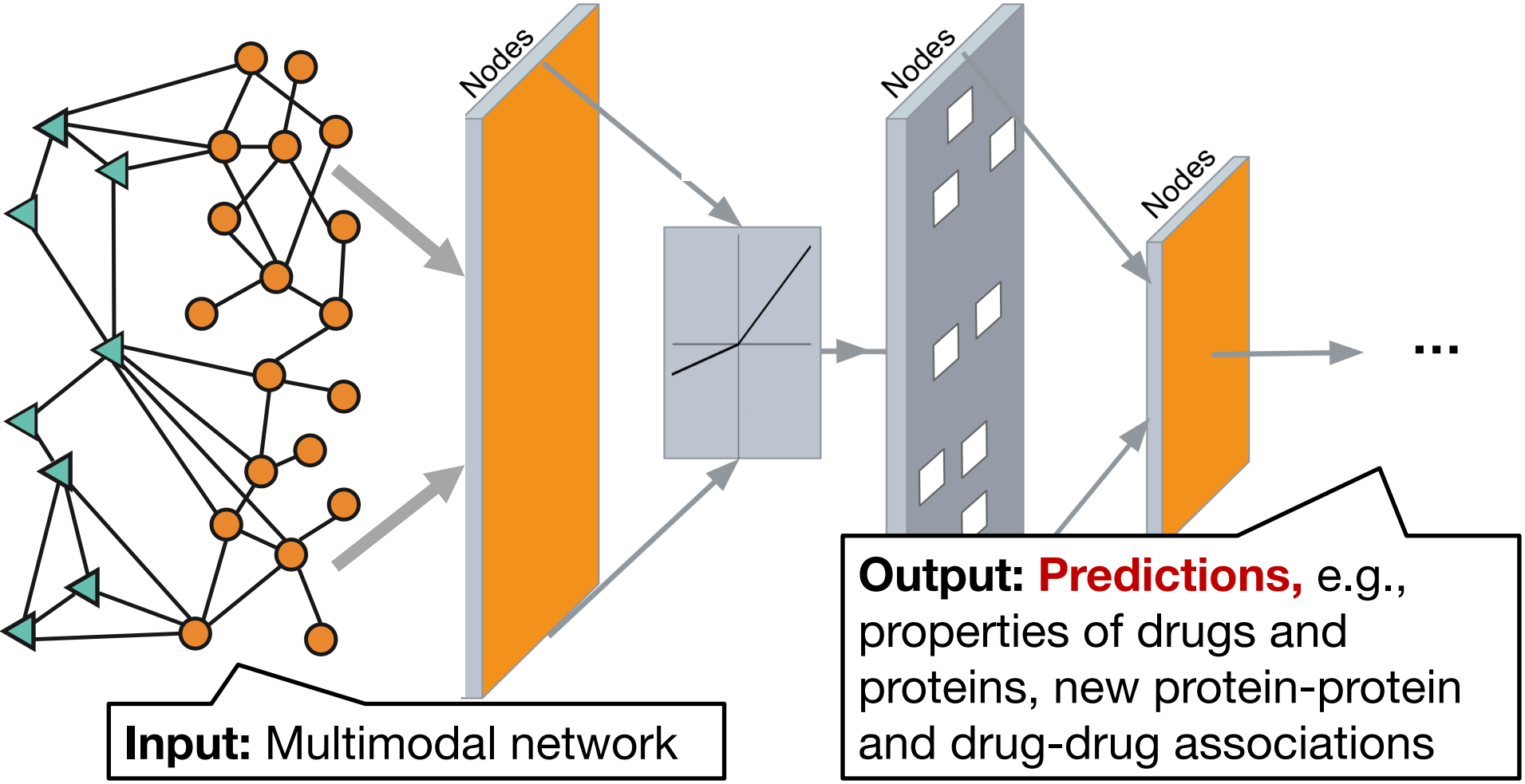
	A	B	C	D	E	Feat	
A	0	1	1	1	0	1	0
B	1	0	0	1	1	0	0
C	1	0	0	1	0	0	1
D	1	1	1	0	1	1	1
E	0	1	0	1	0	1	0



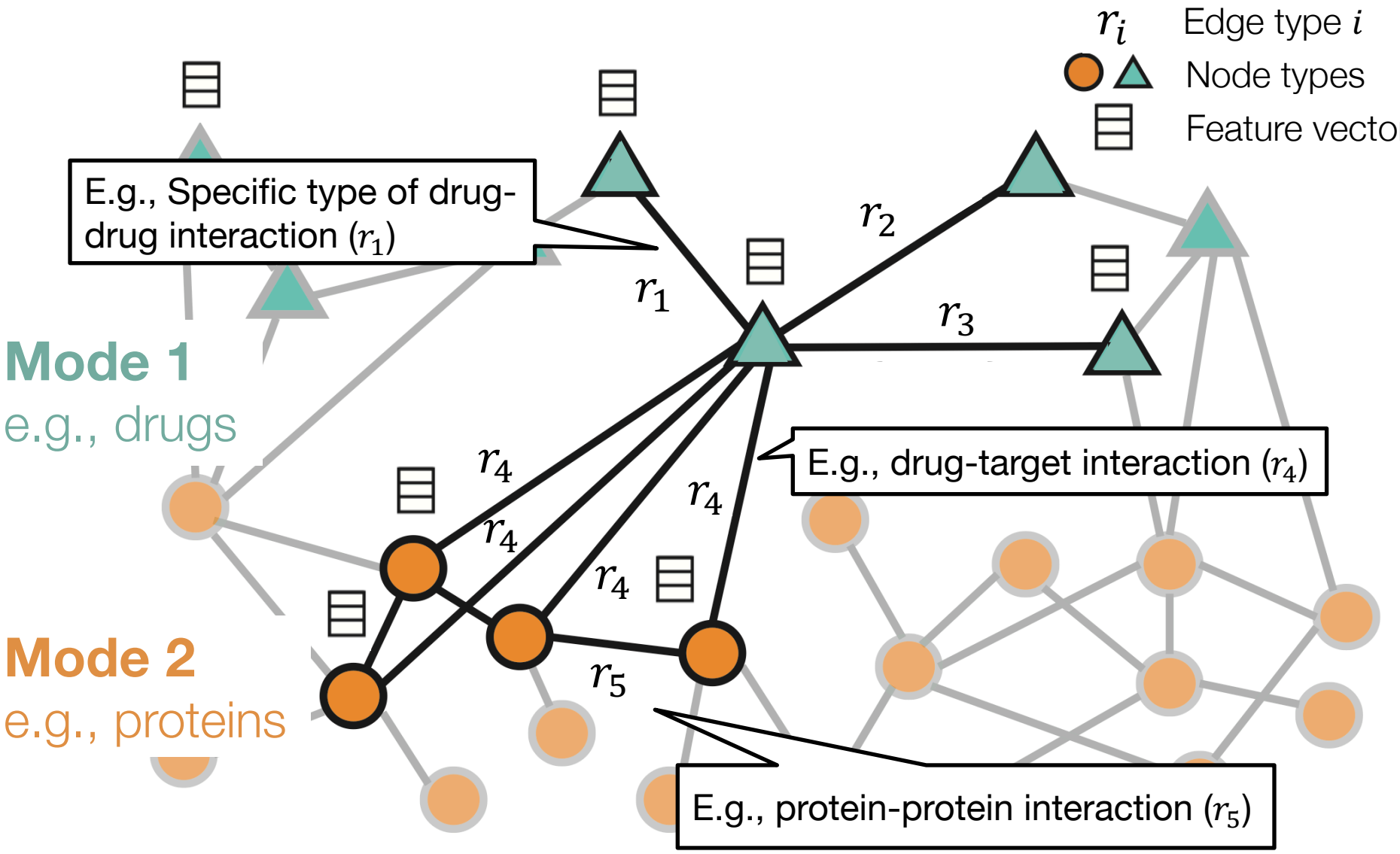
?

We need to generalize convolutions beyond simple lattices to multimodal networks and leverage node features/attributes

Approach: Deep Learning for Multimodal Networks



Multimodal Networks



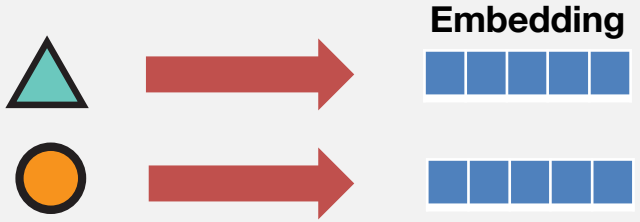
Preview of Results

1. Used new approach to **predict safety and side effects of drug combinations in real patients:**
   - First-ever systematic and predictive study of drug combinations
   - Follow-up research on prostate cancer and validations in the clinic

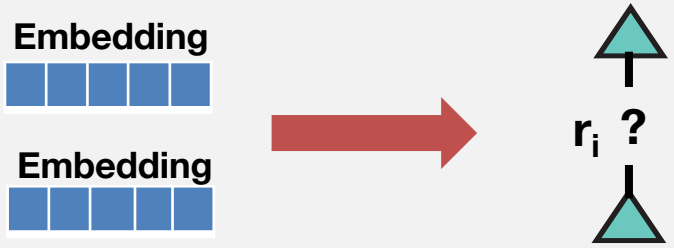
2. Used new approach to **repurpose old drugs for new diseases:**
   - Outperforms baselines by up to 172%
   - Correctly predicted drugs repurposed at Stanford

Overview of our deep learning approach for multimodal networks

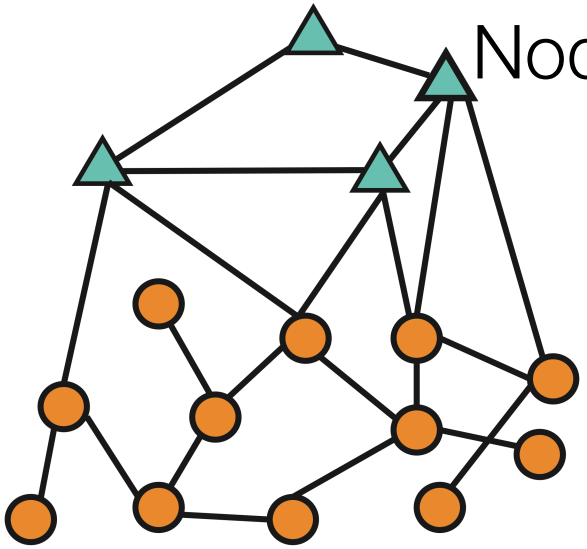
1. **Encoder:** Take a multimodal network and learn an *embedding* for every node



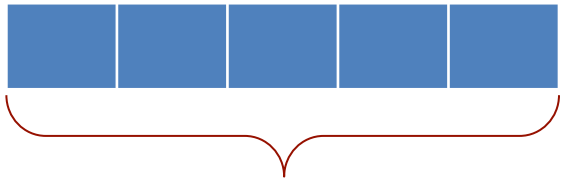
2. **Decoder:** Use the learned embeddings to predict labeled edges between nodes



Embedding Nodes



$$f: v \rightarrow \mathbb{R}^d$$



$\mathbb{R}^d$

Feature representation, embedding

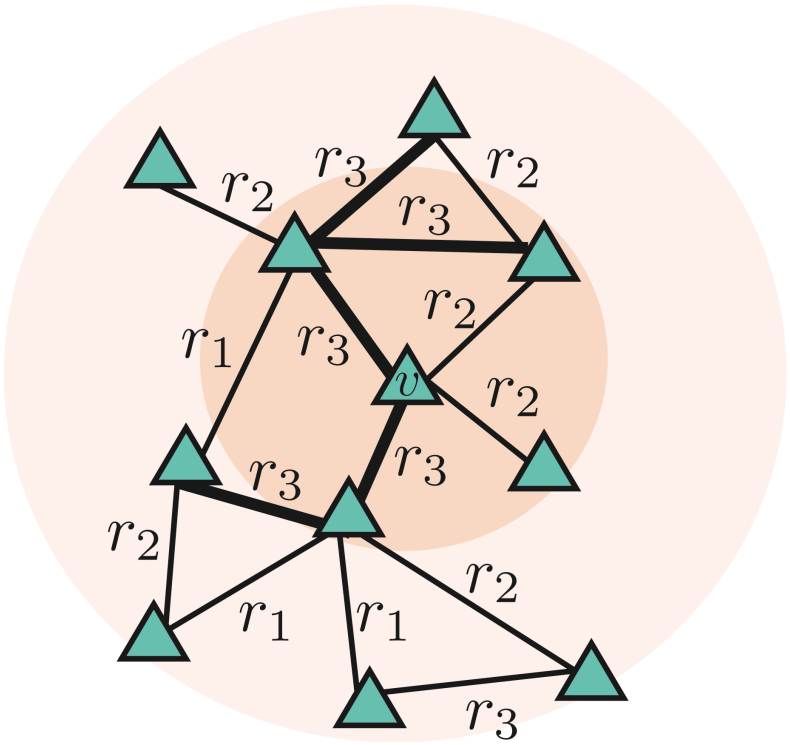
Objective: Map nodes to d-dimensional embeddings such that nodes with similar network neighborhoods are embedded close together

Next: How to learn mapping function f?

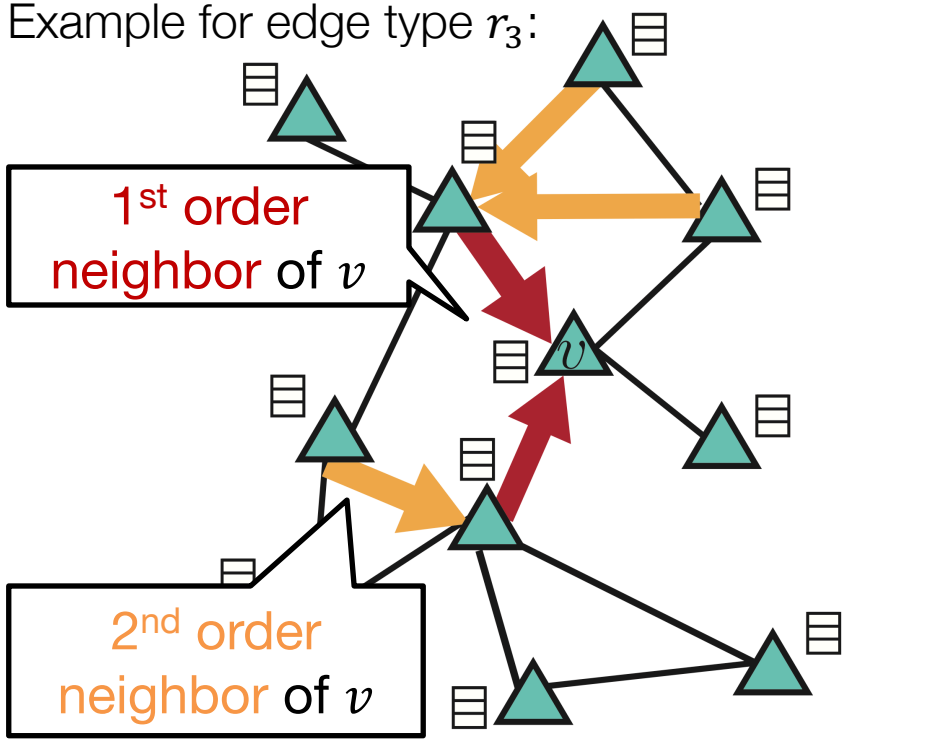
Key Idea: Aggregate Neighbors

Generate embeddings based on **local network neighborhoods separated by edge type**

1) Determine a node's computation graph for each edge type

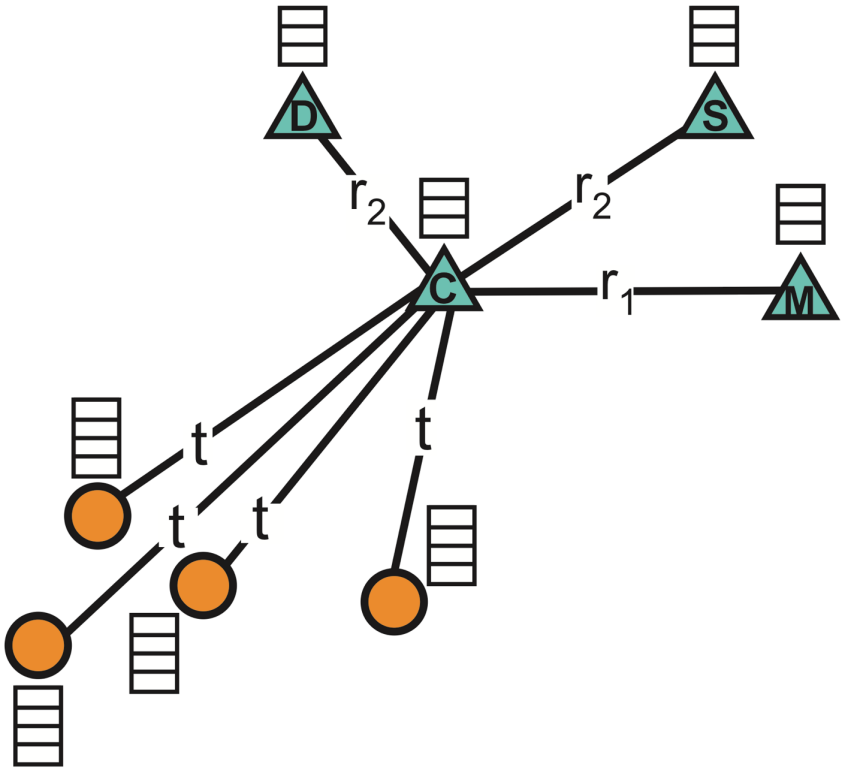


2) Learn how to transform and propagate information across computation graph

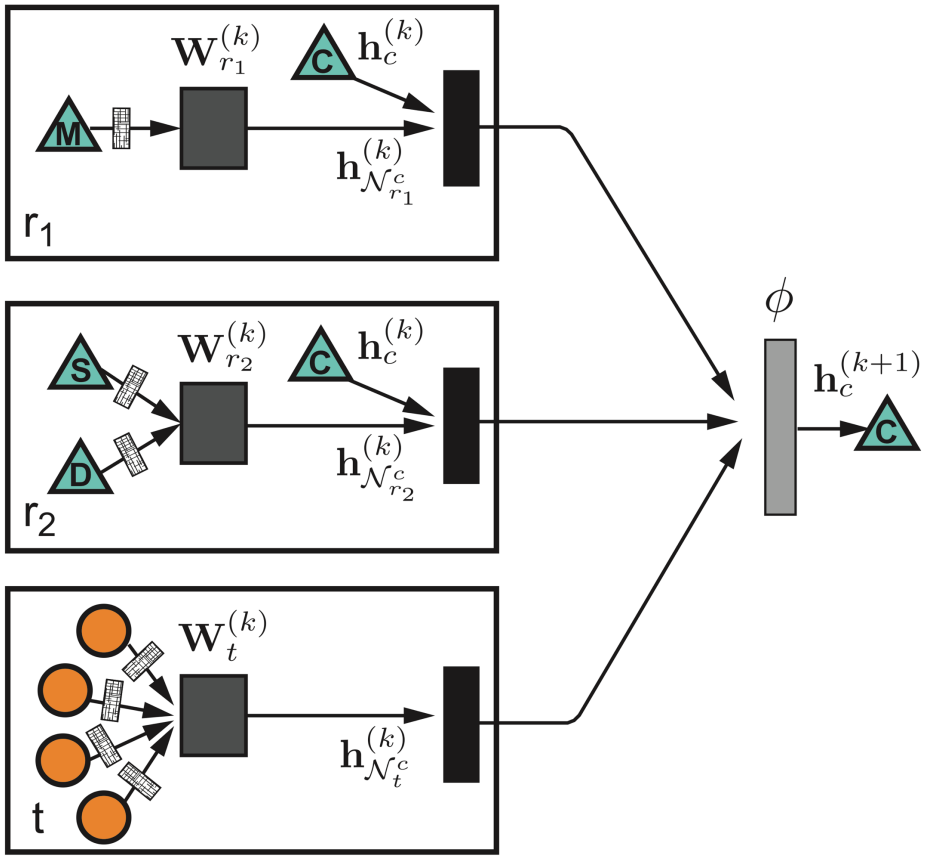


Example: Aggregate Neighbors

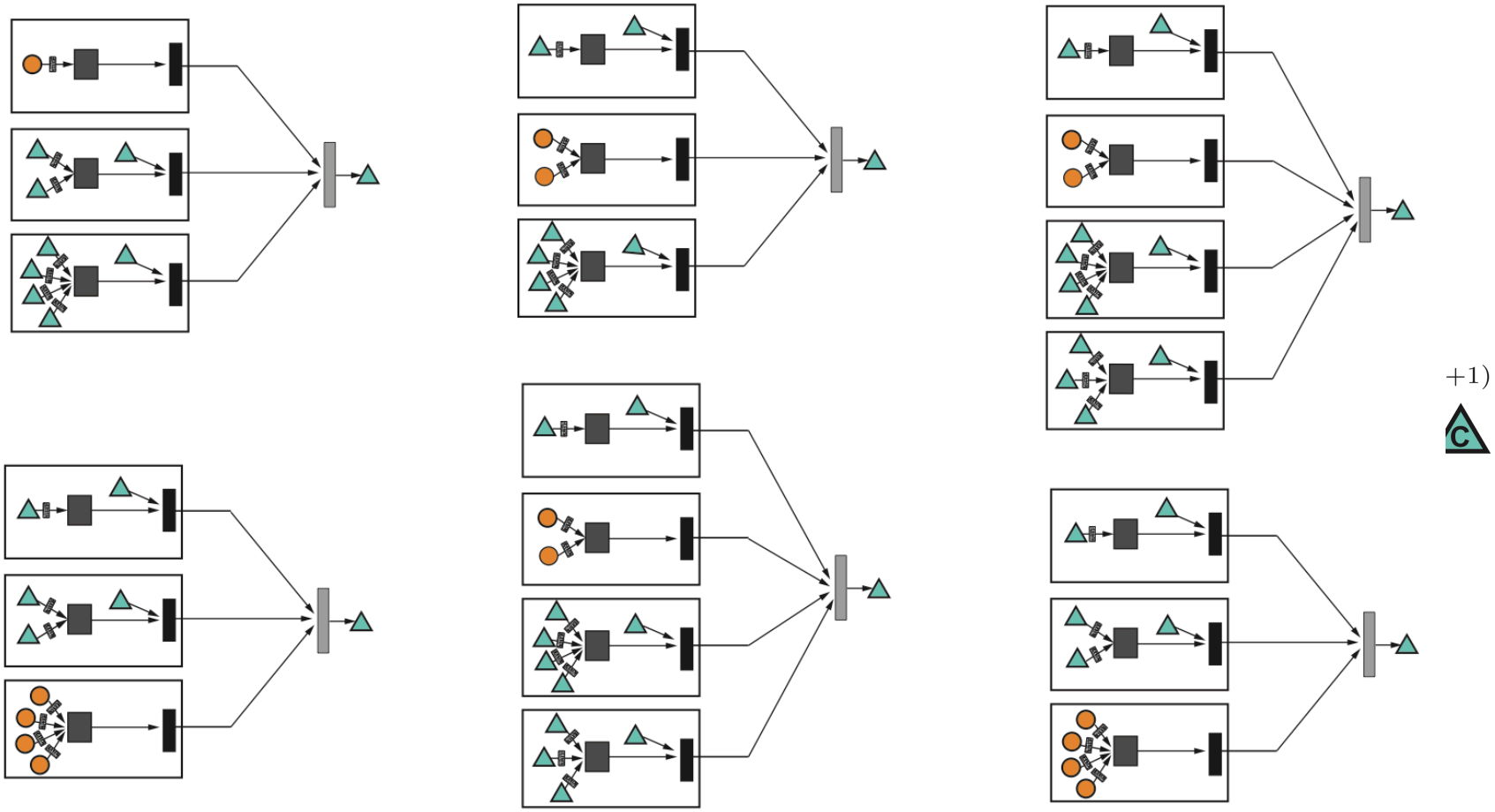
1st order network neighborhood of node C



1st order computation graph of node C

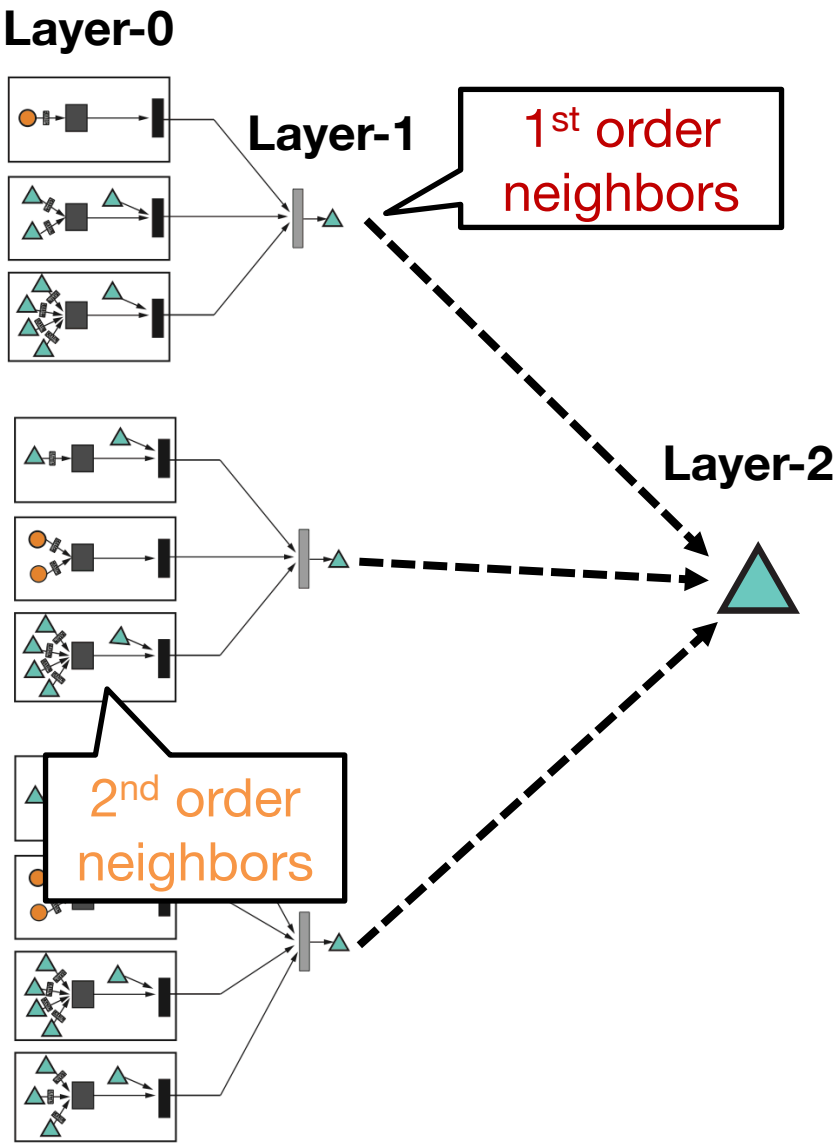


Every node learns how to aggregate its own neighbors



Every node defines a unique computation graph

Deep Model: Many Layers



Model can be of arbitrary depth:

- Nodes have embeddings at each layer
- Layer-0 embeddings are nodes' input features

Deep model with K layers:

- Convolves information across K^{th} order neighborhood
- Embedding of a node depends on nodes at most K hops away

Recap: Nodes with **similar network neighborhoods** are embedded **close together**

The Math: Deep Graph Encoder

Key element: Each node's computation graph defines a neural network with a different architecture

- **Initial 0-th layer embeddings are equal to node features:**

$$\mathbf{h}_v^{(0)} = \mathbf{x}_v$$

Aggregate neighbor's previous-layer embeddings, separated by edge type

Ability to integrate side information about nodes

- **Per-layer update of node embeddings:**

$$\mathbf{h}_v^{(k)} = \phi\left(\sum_r \sum_{u \in N_v^r} c_r^{uv} \mathbf{W}_r^{(k-1)} \mathbf{h}_u^{(k-1)} + c_r^v \mathbf{h}_v^{(k-1)}\right) \quad k = 1, \ldots, K$$

Previous-layer embedding of v

- **Embeddings after K layers of neighborhood aggregation:**

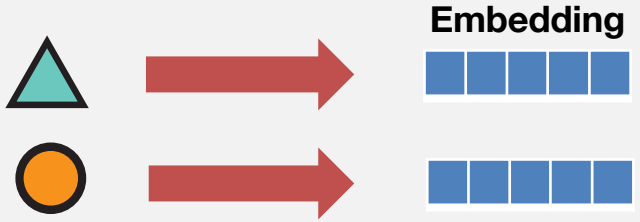
$$\mathbf{z}_v = \mathbf{h}_v^{(K)}$$

$\mathbf{W}_r^{(k)}$ Para

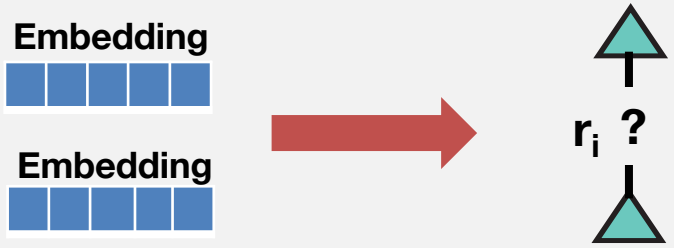
Normalization constant, fixed *e.g.*, $1/|N_v^r|$, or learned

Overview of our deep learning approach for multimodal networks

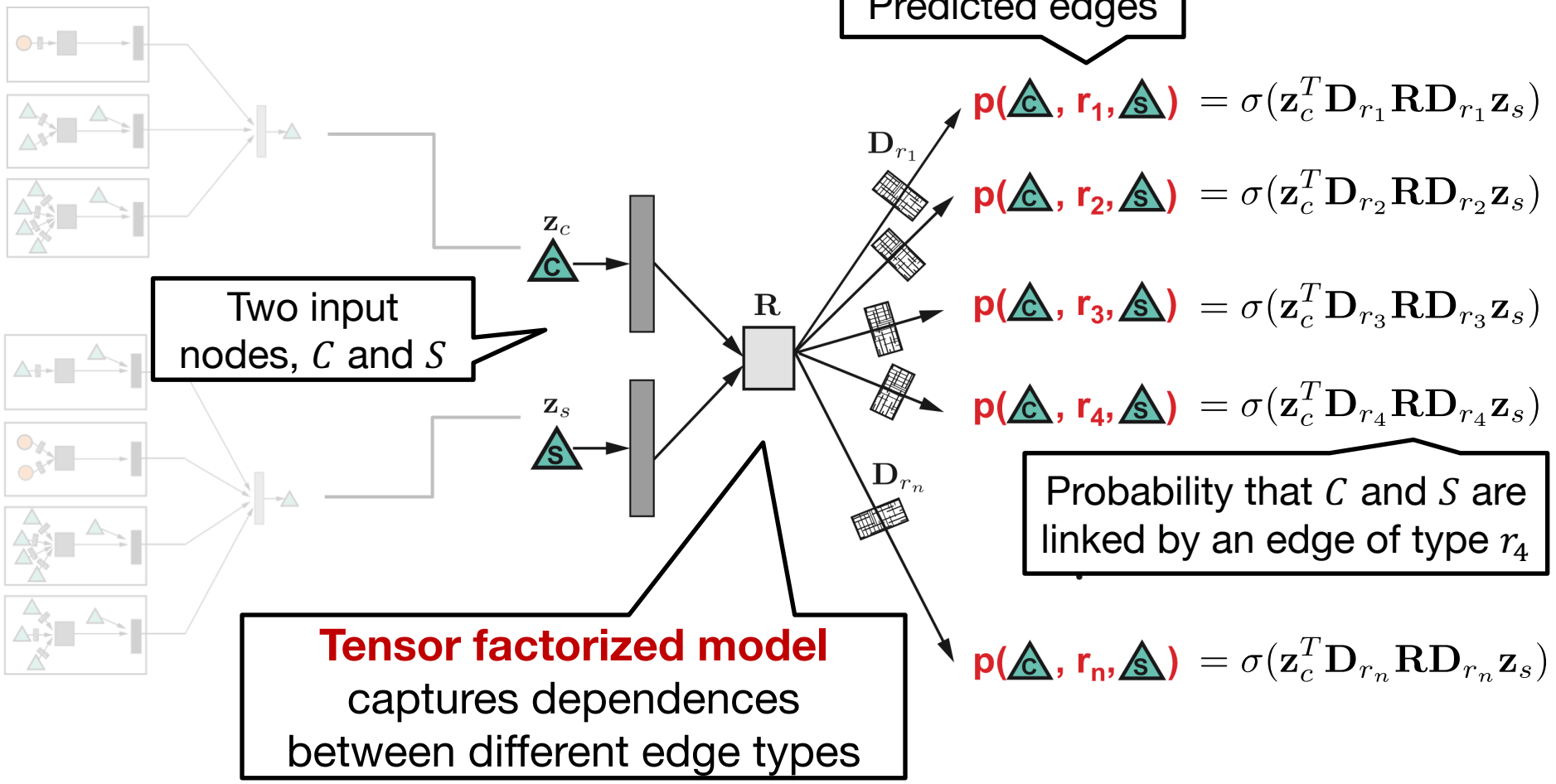
1. **Encoder:** Take a multimodal network and learn an *embedding* for every node



2. **Decoder:** Use the learned embeddings to predict labeled edges between nodes



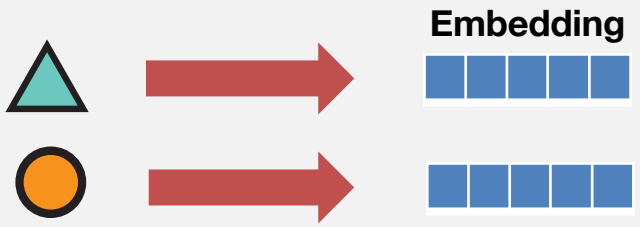
Heterogeneous Edge Decoder



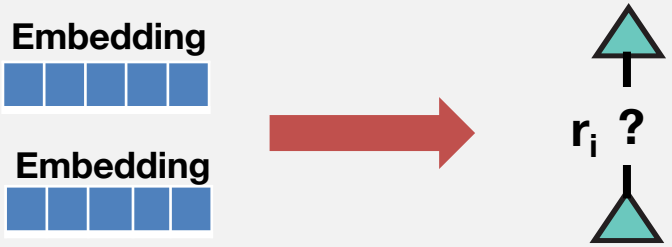
$\mathbf{R}, \mathbf{D}_{r_i}$ Parameter weight matrices

Overview of our deep learning approach for multimodal networks

1. **Encoder:** Take a multimodal network and learn an *embedding* for every node



2. **Decoder:** Use the learned embeddings to predict labeled edges between nodes



Training the model: Feed embeddings into any loss function and run stochastic gradient descent to train weight parameters:

- Use a loss based on e.g., random walks, node proximity in the graph
- Directly train the model for a supervised task (e.g., node classification)

Recap: Deep Learning for Multimodal Networks

Key new insights:

- Advances Graph Neural and Graph Convolutional models [e.g., Kipf et al., ICLR'17; Hamilton et al., NIPS'17] by treating **multimodal networks**
- Generates **powerful deep embeddings**, unlike shallow embedding methods [e.g., Matrix factorization, Node2vec, DeepWalk]
- Can define and tackle **new prediction problems** [e.g., graph-level classification and subgraph-level link prediction]

We can now apply deep learning much more broadly, to any multimodal network

New frontiers for applications in **biology** and **medicine**

Outline of this Lecture

1) Deep Graph Neural Networks ✓

2) Polypharmacy & Drug Interactions

3) Drug Repurposing

4) New Directions and Opportunities

5) Practical Advice and Demos

Polypharmacy and Drug-Drug Interactions

Based on material from:

- Zitnik et al. 2018. Deep Learning for Network Biology. *ISMB.*
- Zitnik et al. 2018. Modeling Polypharmacy Side Effects with Graph Convolutional Networks. *ISMB & Bioinformatics.*

Polypharmacy

Patients take multiple drugs to treat complex or co-existing diseases

46% of people over 65 years take more than 5 drugs

Many take more than **20** drugs to treat heart diseases, depression or cancer

15% of the U.S. population affected by unwanted side effects

Annual costs in treating side effects exceed **$177** billion in the US alone

[Ernst and Grizzle, JAPA'01; Kantor et al., JAMA'15]

Reports on Unwanted Side Effects

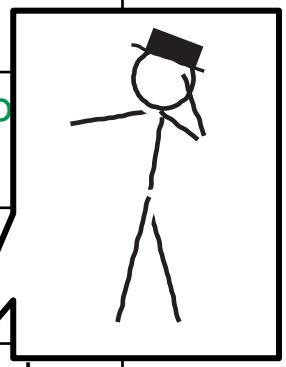
The FDA Adverse Event Reporting System (FAERS)

Drugs taken	**Unwanted side effects**
	Peliosis hepatis (5%), Heart rate increased (10%), Aortic aneurysm (3%)
	Joint stiffness (30%), Joint swelling (10%), Bone marrow fibrosis (3%)
	Anaemia (15%), Bone marrow fibrosis (5%), Intestinal ulcer (5%)
	Anaemia (15%), Bone marrow fibrosis (5%), Intestinal ulcer (5%), Joint stiffness (30%), Joint swelling (10%), Colon cancer (4%), Fatigue (40%)
	Peliosis hepatis (5%), Heart rate increased (10%), Aortic aneurysm (3%), Joint stiffness (30%), Joint swelling (10%), Bone marrow fibrosis (3%)
...	...

Reports on Unwanted Side Effects

The FDA Adverse Event Reporting System (FAERS)

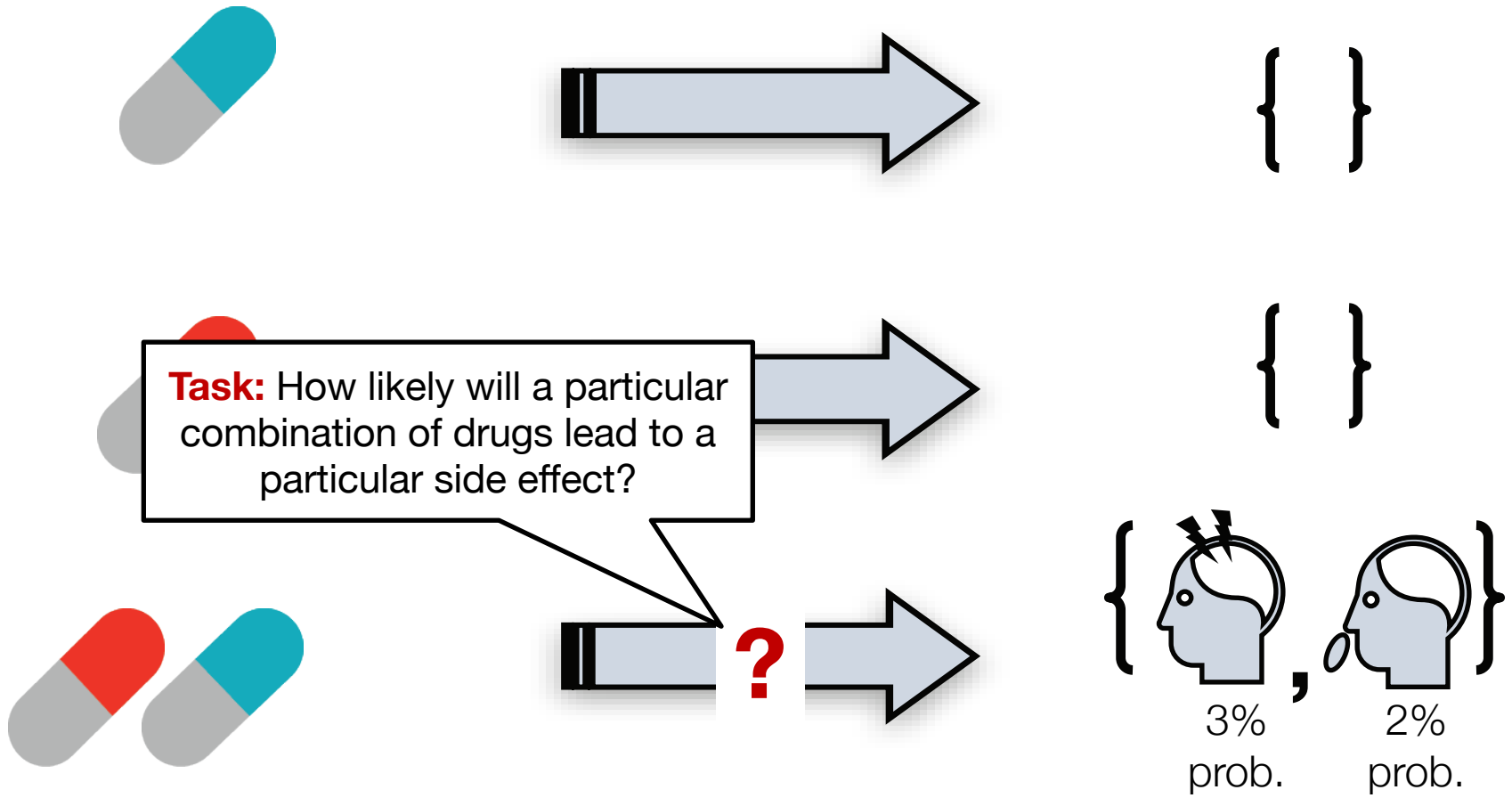
Drugs taken	Unwanted side effects
	Peliosis hepatis (5%), Heart rate increased (10%), Aortic aneurysm (3%)
	Joint stiffness (30%), Joint swelling (10%), Bone marro fibrosis (3%)
	Anaemia (15%), Bone marrow fibrosis (5%), Intestinal (5%)
	Anaemia (15%), Bone marrow fibrosis (5%), Intestinal ulcer (5%), Joint stiffness (30%), Joint swelling (10%), **Colon cancer (4%), Fatigue (40%)**
	Peliosis hepatis (5%), Heart rate increased (10%), Aortic aneurysm (3%), Joint stiffness (30%), Joint swelling (10%), Bone marrow fibrosis (3%)
...	...



Unexpected Drug Interactions

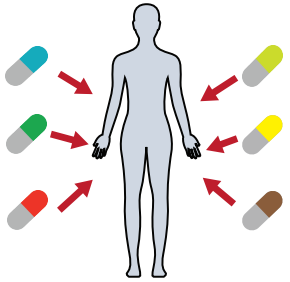
Co-prescribed drugs

Side Effects



Why is modeling polypharmacy hard?

Combinatorial explosion

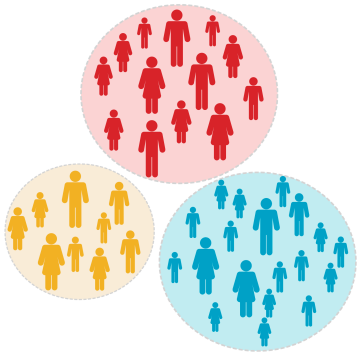


- >13 million possible combinations of 2 drugs
- >20 billion possible combinations of 3 drugs

Non-linear & non-additive interactions

- Different effect than the additive effect of individual drugs

Small subsets of patients



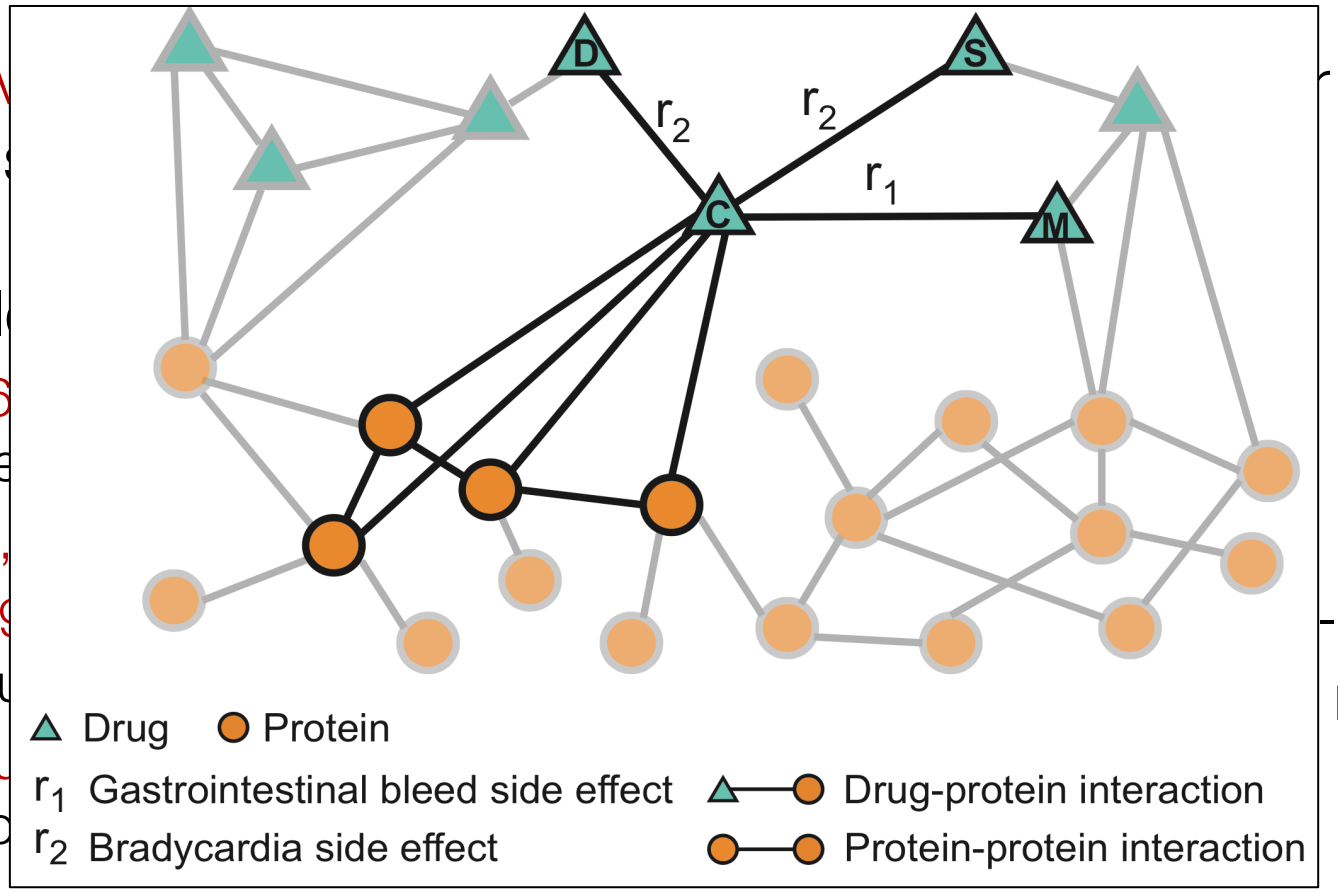
- Side effects are interdependent
- No info on drug combinations not yet used in patients

We need Polypharmacy Dataset

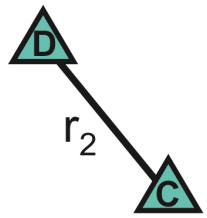
Objectiv
all drug

We buil

- 4,6
  eve
- 18,
- 719
  cou
- Dru
  pro



Drug-drug



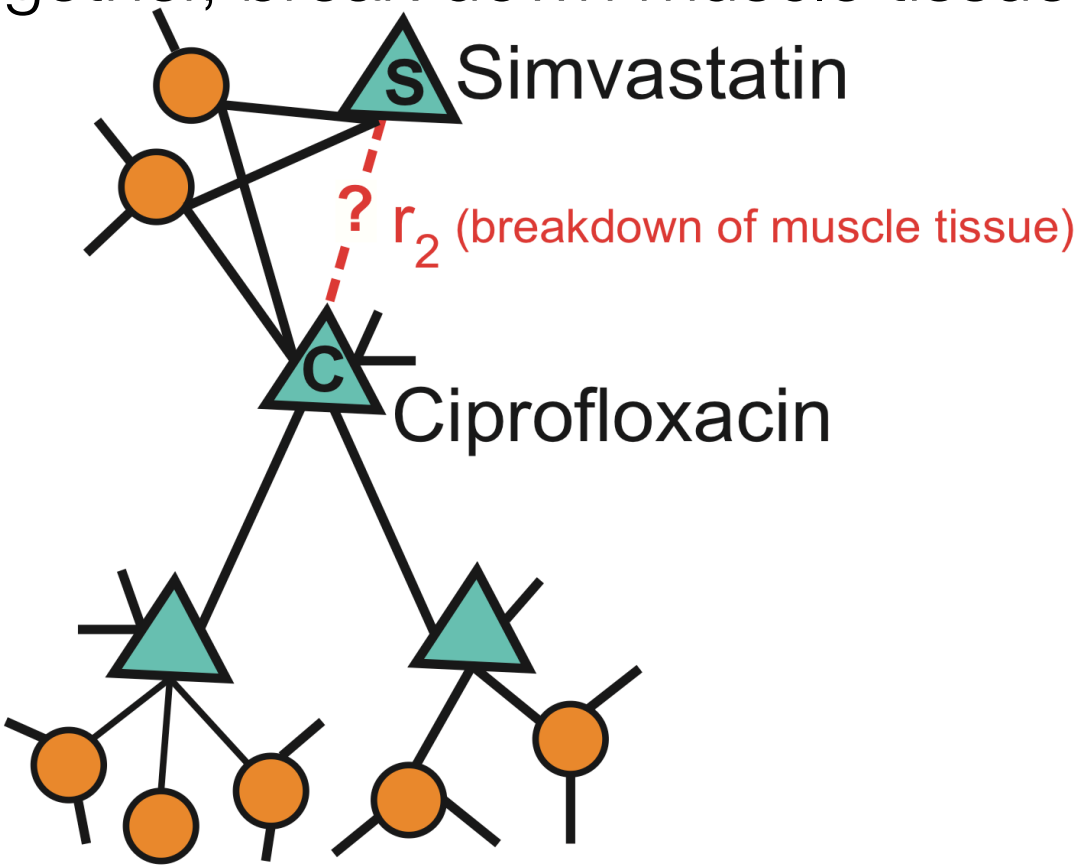
Drug-protein

Protein-protein

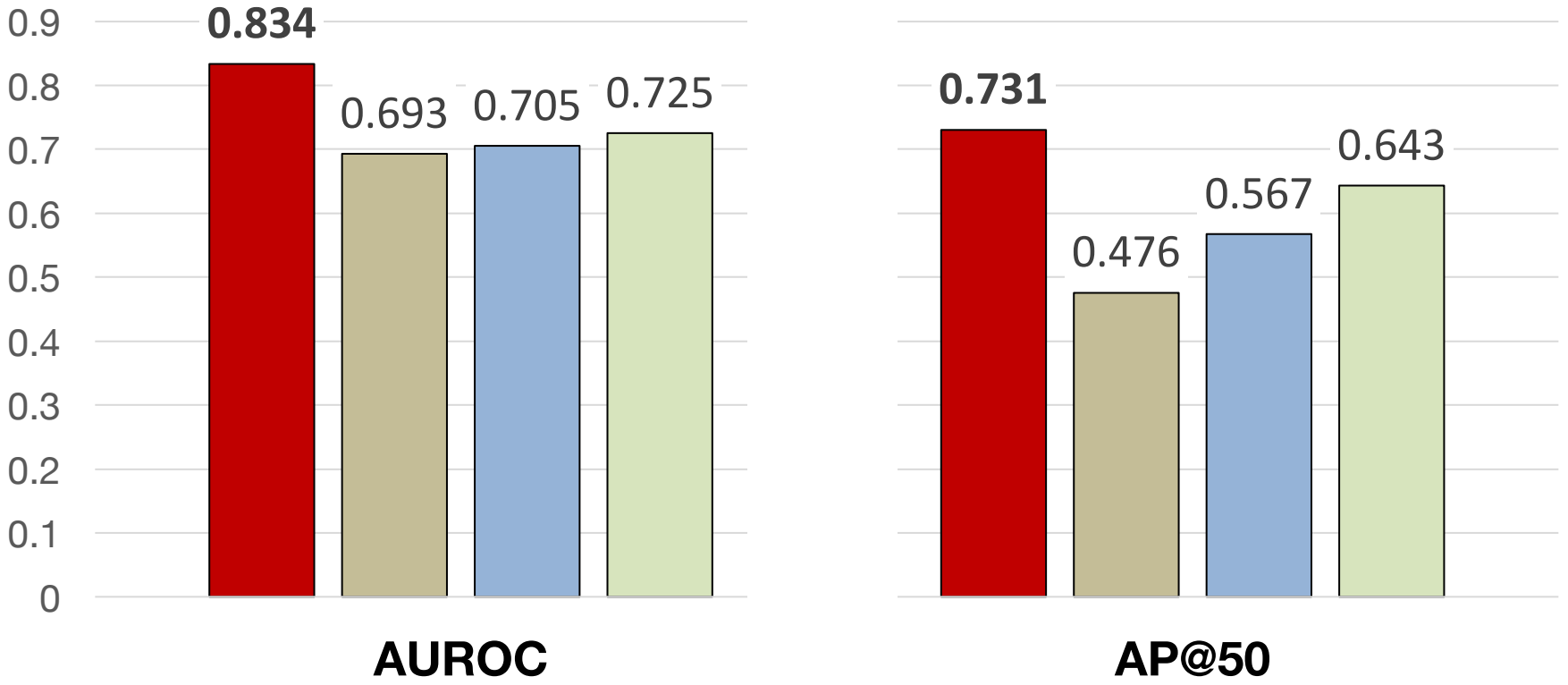
Gives multimodal network with over 5 million edges separated into 1,000 different edge types

We apply our deep approach to the polypharmacy network

E.g.: How likely will Simvastatin and Ciprofloxacin, when taken together, break down muscle tissue?



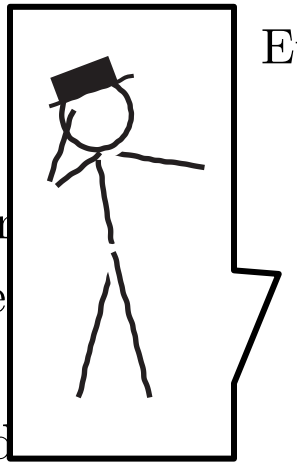
Results: Side Effect Prediction



Novel Predictions

- Train deep model on data generated **prior to 2012**
- How many **predictions** have been **confirmed after 2012**?

Rank	Drug	Drug	Side effect	Evidence found
1	Pyrimethamine	Aliskiren	Sarcoma	
2	Tigecycline	Bimatoprost	Autonomic r	
3	Telangiectases	Omeprazole	Dacarbazine	
4	Tolcapone	Pyrimethamine	Blood brain	
			ead	
			ular acidosis	
			Cerebral thrombosis	
8	Atorvastatin	Amlodipine	Muscle inflammation	
9	Aliskiren	Tioconazole	Breast inflammation	
10	Estradiol	Nadolol	Endometriosis	



Case Report

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

Utility of Predictions in the Clinic

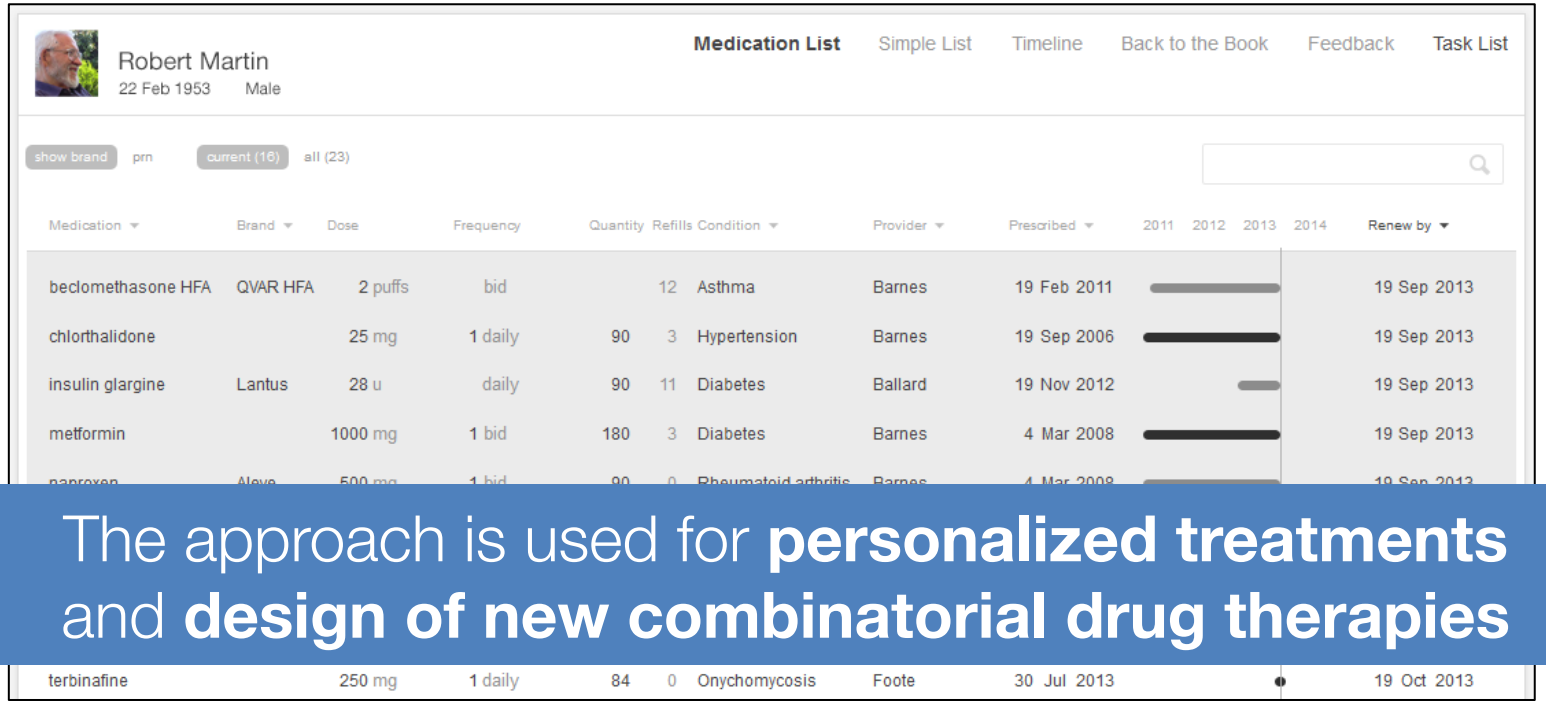
Clinical validation: Drug-drug interaction markers, lab values, and surrogates

HARVARD MEDICAL SCHOOL

Stanford MEDICINE

NEWTON-WELLESLEY HOSPITAL

MASSACHUSETTS GENERAL HOSPITAL

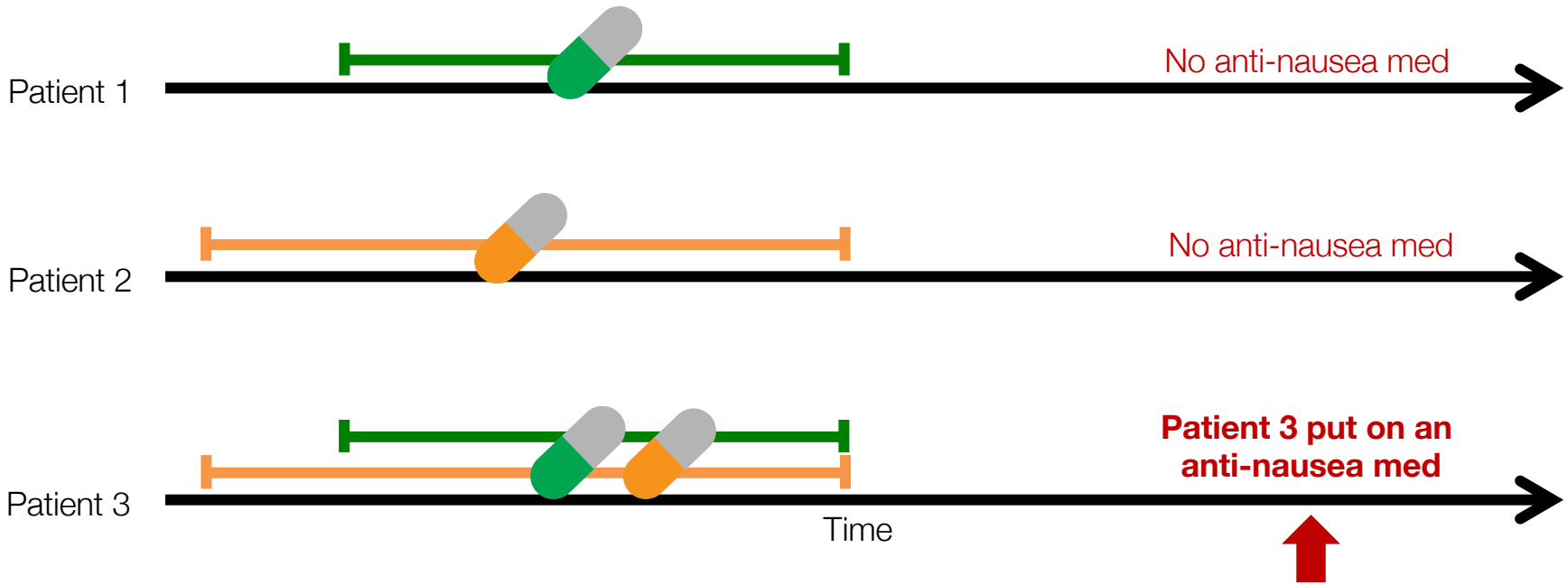


The approach is used for **personalized treatments** and **design of new combinatorial drug therapies**

Validation in the Clinic: Key Idea

Question: Is it a good idea to prescribe a particular combination of drugs to a particular patient?

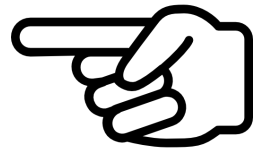
- E.g., Prediction: {💊,💊} cause nausea as a side effect



Outline of this Lecture

1) Deep Graph Neural Networks ✓

2) Polypharmacy & Drug Interactions ✓

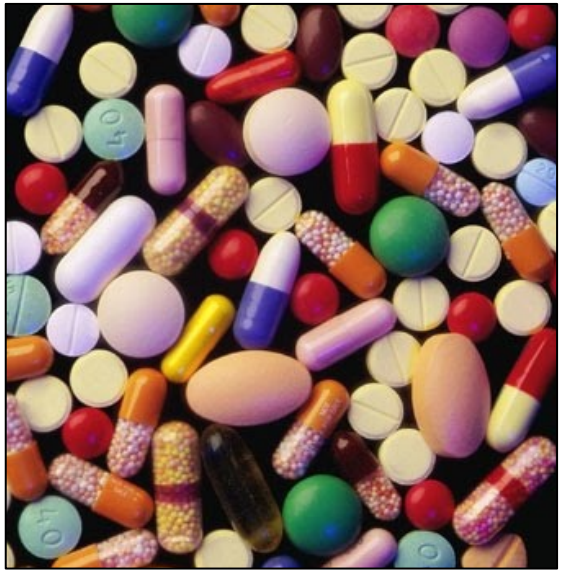
3) Drug Repurposing 

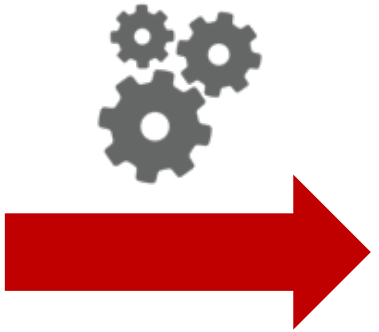
4) New Directions and Opportunities

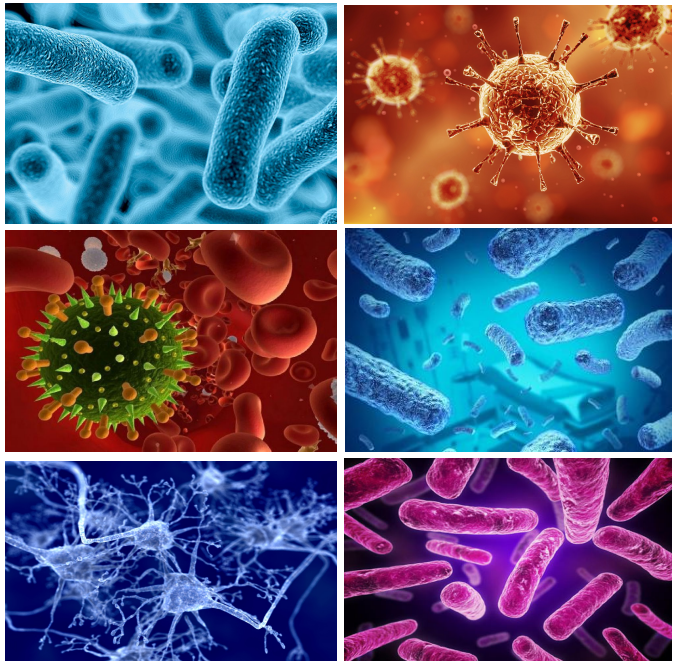
5) Practical Advice and Demos

Computational Drug Repurposing

Computational Drug Discovery







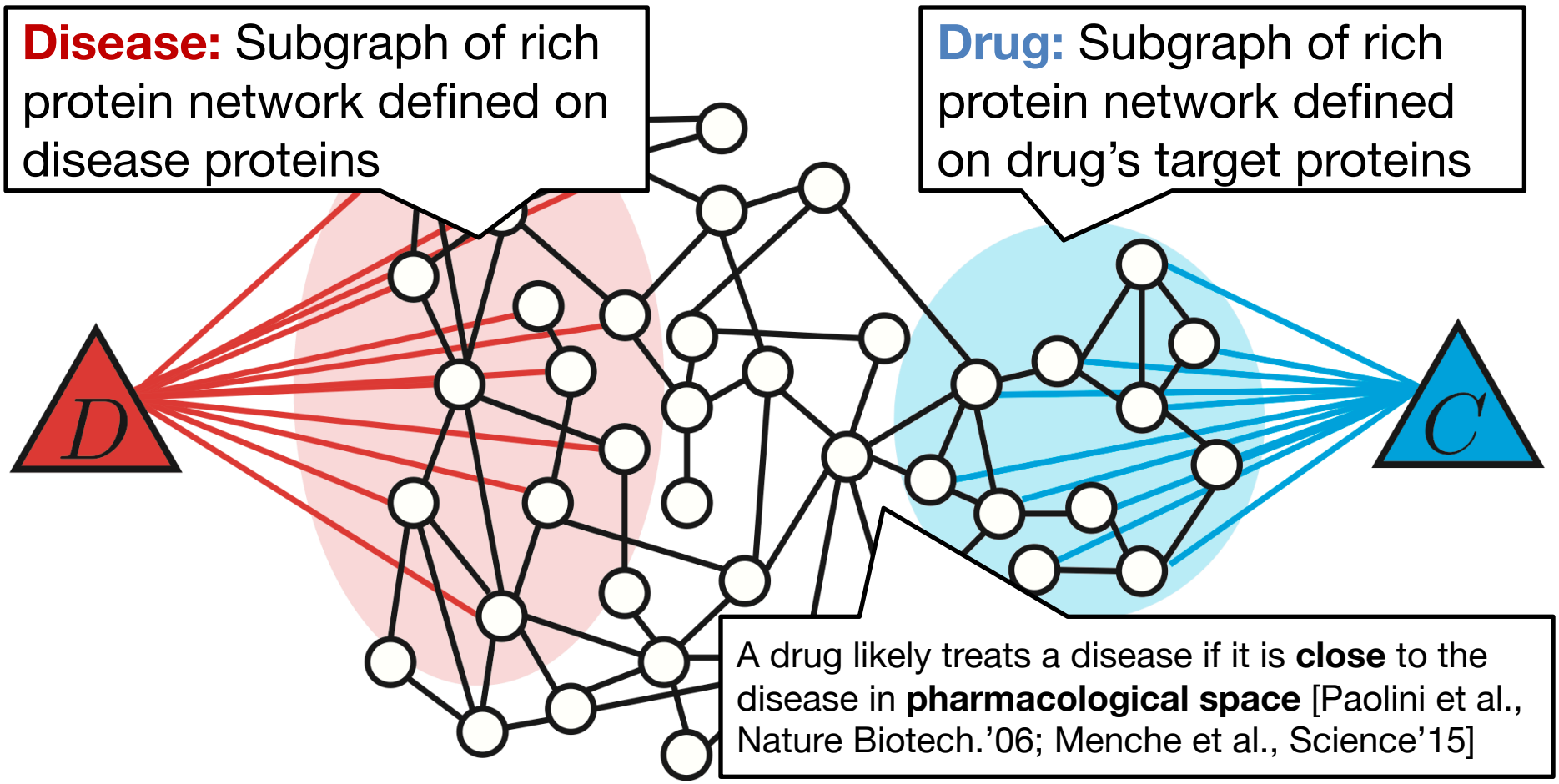
Goal: Find which diseases a new drug (molecule) could treat

New tricks for old drugs

Faced with skyrocketing costs for developing new drugs, researchers are looking at ways to repurpose older ones — and even some that failed in initial trials.

Nature 2016

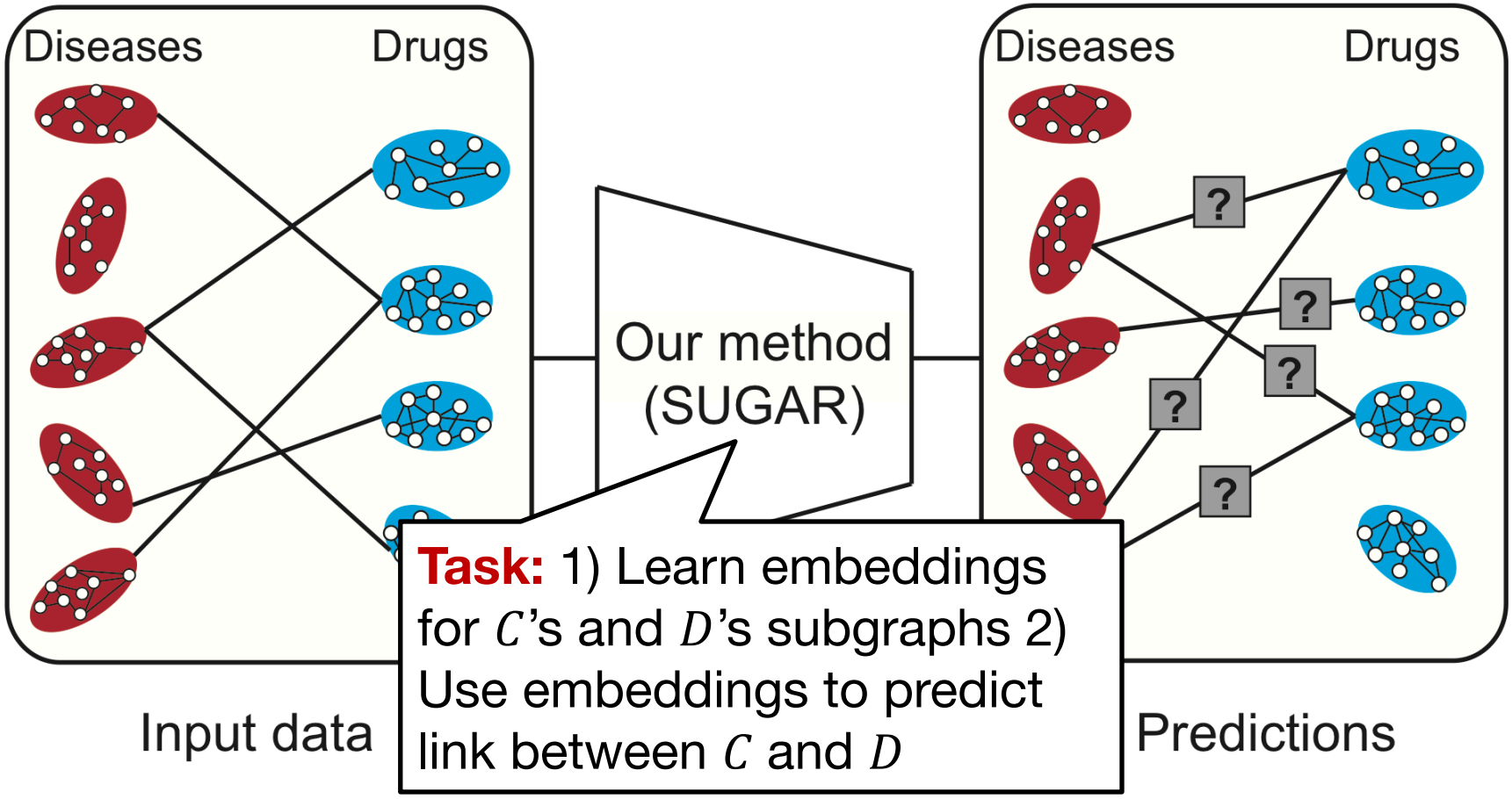
Key Insight: Subgraphs



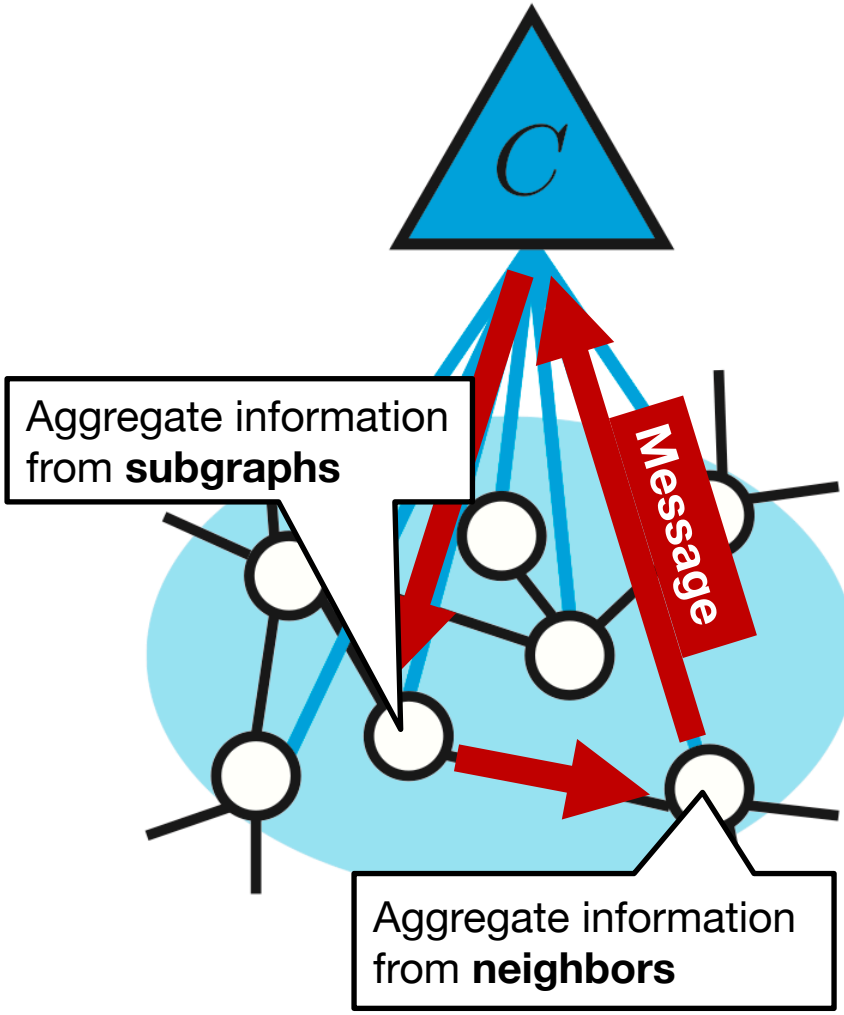
Idea: Use the paradigm of embeddings to operationalize the concept of closeness in pharmacological space

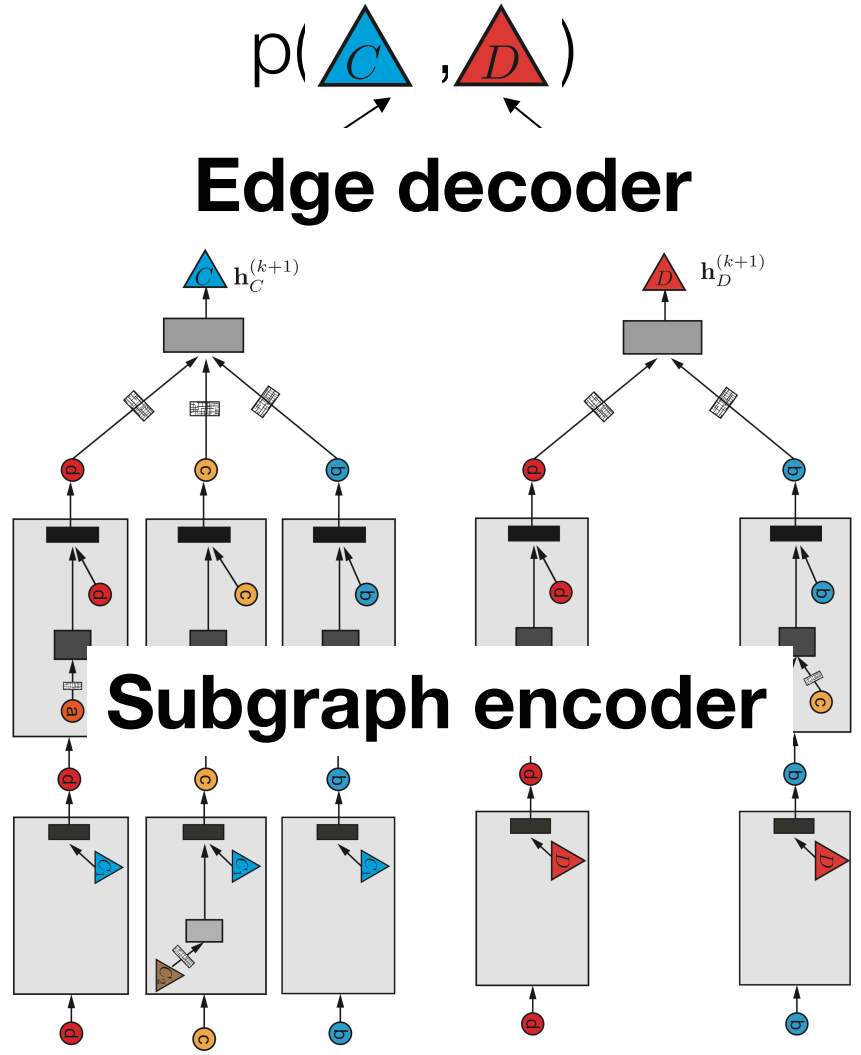
Link Prediction Between Subgraphs

Task: Given drug C and disease D, predict if C treats D



SUGAR: Neural Message Passing





We need Drug Repurposing Dataset

- Protein-protein interaction network culled from 15 knowledge databases [Menche et al. *Science* 15]
  - 19K nodes, 350K edges
- Drug-protein and disease-protein links:
  - DrugBank, OMIM, DisGeNET, STITCH DB and others
  - 5K drugs, 20K diseases
  - 20K drug-protein links, 560K disease-protein links
- Drug medical indications:
  - DrugBank, MEDI-HPS, DailyMed, RepoDB and others
  - 6K drug-disease indications
- Side information on drugs, diseases, proteins, etc.: Molecular pathways, disease symptoms, side effects

Predictive Performance

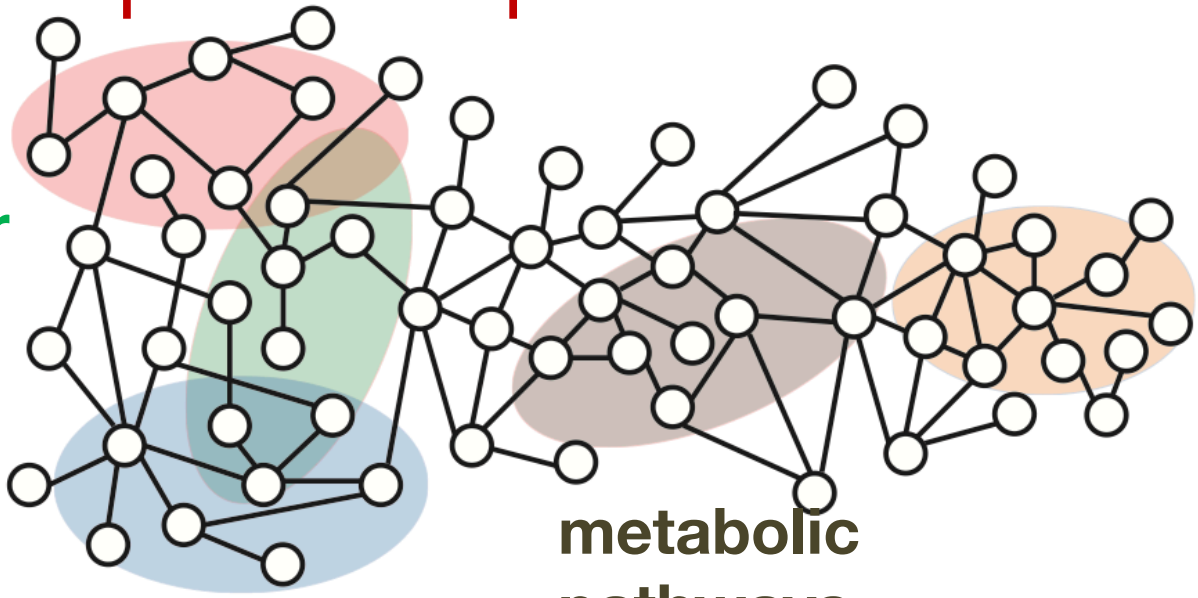
Task: Given a disease and a drug, predict if the drug could treat the disease

Approach	AUPRC	AUROC
Our method (SUGAR)	0.851	0.888
Graphlets [Bioinformatics'13]		
PREdicting Drug IndiCaTions [Mol. Sys. Biol.'11]		
Bi-directional random walks [Bioinformatics'16]		
Heterogeneous graph inference [Bioinformatics'14]		
Drug-disease closeness [Nat. Commun.'17]		
Drug-disease dispersion [Nat. Commun.'17]		
Gene-based network overlap [Nat. Commun.'17]		

Up to 49% improvement

Up to 172% improvement

Side Information further improves performance

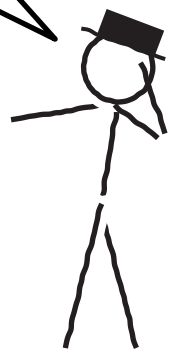


Metabolic pathways	Molecular functions	Biological processes	Cellular components	AUPRC	AUROC
				0.851	0.888
✓				0.869	0.893
✓	✓			0.874	0.912
✓	✓	✓		0.893	0.912
✓	✓	✓	✓	0.901	0.928

Drug Repurposing at Stanford

Stanford MEDICINE | SPARK Translational Research Program
From Bench to Bedside

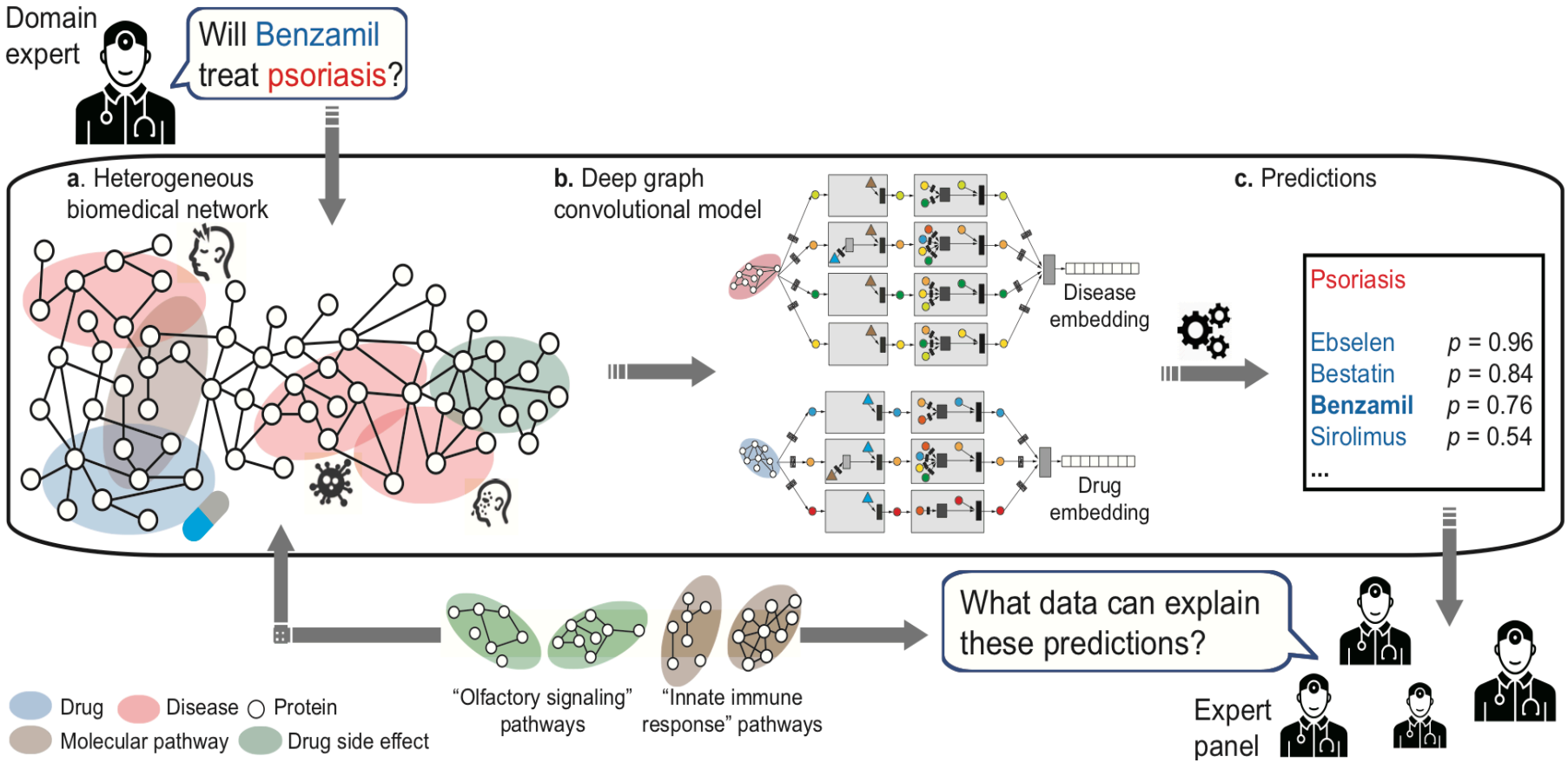
Task: Predict if an existing drug can be repurposed for a new disease



Drug	Disease	
N-acetyl-cysteine	cystic fibrosis	
Xamoterol	neurodegenerat	
Plerixafor	cancer	
Sodium selenite	cancer	Rank: 36/5000
Ebselen	C difficile	Rank: 10/5000
Itraconazole	cancer	Rank: 26/5000
Bestatin	lymphedema	Rank: 11/5000
Bestatin	pulmonary arterial hypertension	Rank: 16/5000
Ketaprofen	lymphedema	Rank: 28/5000
Sildenafil	lymphatic malformation	Rank: 26/5000
Tacrolimus	pulmonary arterial hypertension	Rank: 46/5000
Benzamil	psoriasis	Rank: 114/5000
Carvedilol	Chagas' disease	Rank: 9/5000

Led to follow-up research on prostate cancer and schizophrenia at Stanford Medical School

Introducing Feedbacks for AI Loop



Outline of this Lecture

1) Deep Graph Neural Networks ✓

2) Polypharmacy & Drug Interactions ✓

3) Drug Repurposing ✓

4) New Directions and Opportunities

5) Practical Advice and Demos

New Directions and Opportunities

Material based on:

- Zitnik et al. 2019. Machine Learning for Integrating Data in Biology and Medicine: Principles, Practice, and Opportunities. *Information Fusion.*
- Camacho et al. 2018 Next-Generation Machine Learning for Biological Networks. *Cell.*

New Directions

1. Construct contextual explanatory models and introduce feedbacks for the AI loop

2. Design models to train more with less data

3. Create deep learning models for rich interaction data and computations over graphs

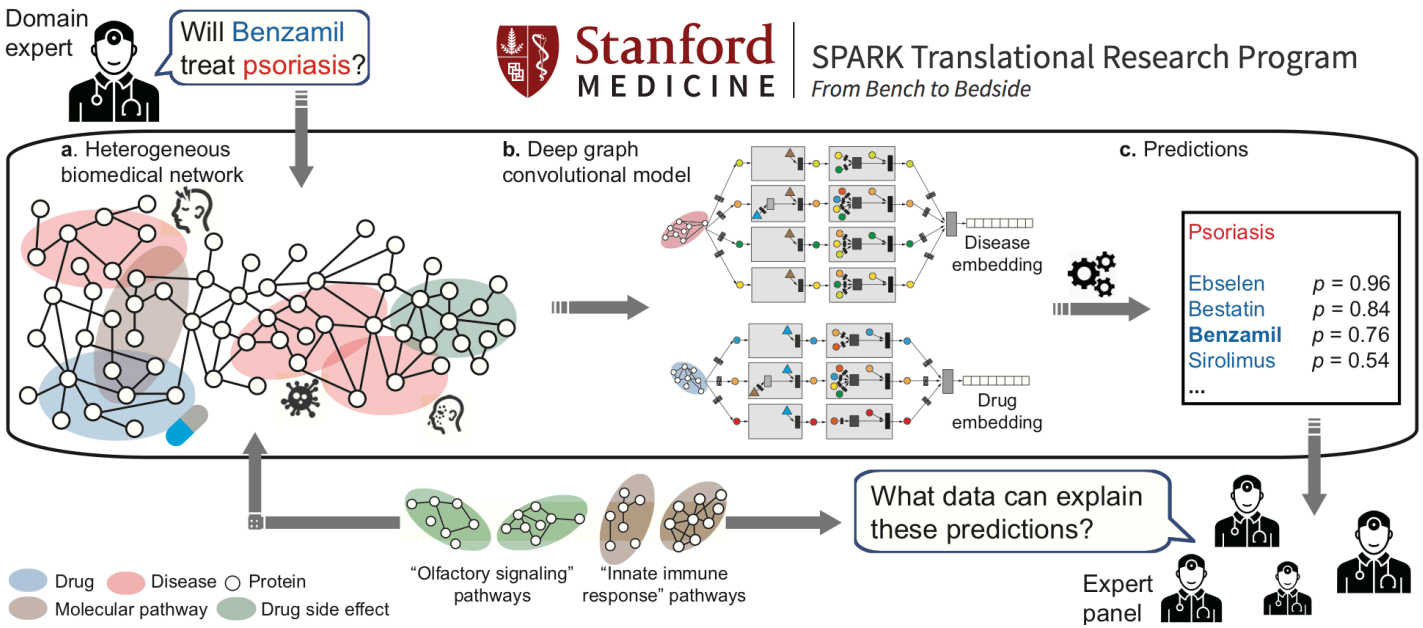
1st New Direction: Explanations

- Exciting phrase is not only
- meaningful interpretations

Networks

Networks

Initial results: Introduce feedback for the AI loop

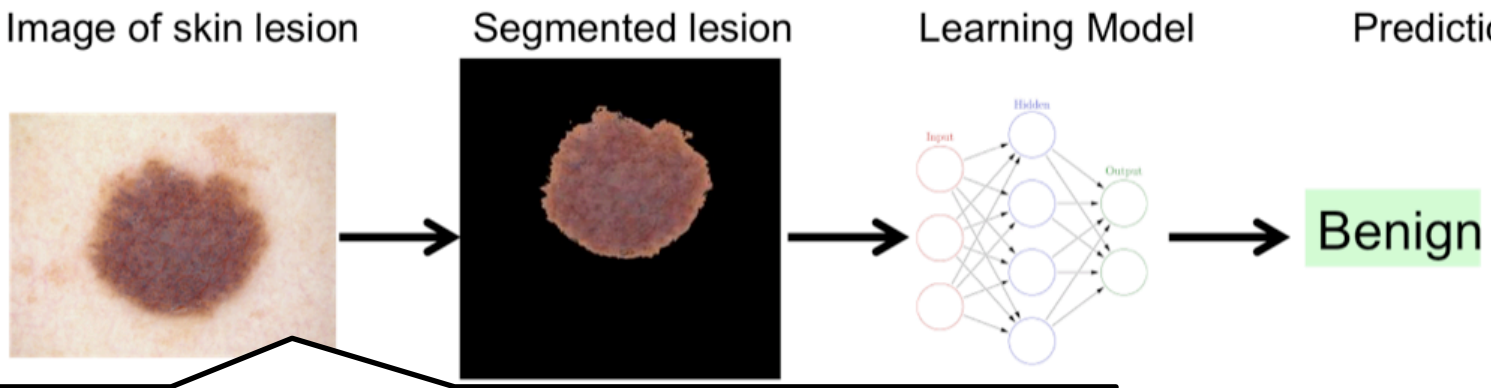


Prediction

Prediction

2nd New Direction: Train with Less Data

- Algorithms to train more with less network data
- Learn about never-before-seen systems, generalize across contexts, e.g., patients, diseases, environments
- Natural case studies:
  - Single-cell genomics, Polygenic analyses, Health informatics

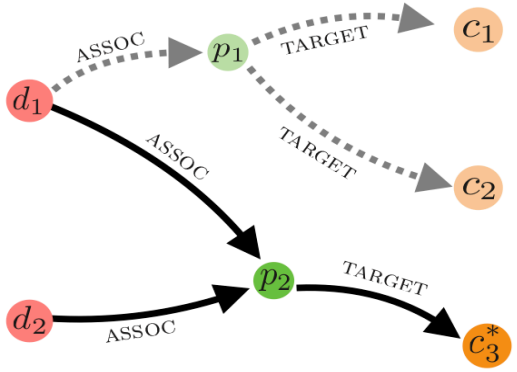


3rd New Direction: Rich Interactions

Initial results: Deep framework for logical queries on knowledge graphs [NeurIPS'18]

Query:

Predict drugs C that might TARGET proteins, which are in turn ASSOCiated with diseases d_1 and d_2



- Prediction beyond node classification and link prediction

Initial results: Biomedical network data and tools

Networks and re **snap.stanford.edu/biodata**

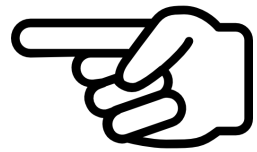
Name	Edges	Entities	Description
CC-Neuron	49,471,006	cell, cell	Similarity network between cells in embroyonic mouse brain
ChCh-Miner	96,137	drug, drug	Interactions between FDA-approved drugs
ChChSe-Decagon	4,649,441	drug, drug, side-effect	Side effects of drug combinations
ChG-InterDecagon	131,034	drug, gene	Chemical-gene interaction network
ChG-Miner	15,424	drug, gene	Drug-target interaction network
ChG-TargetDecagon	18,690	drug, gene	Drug-target interaction network

snap.stanford.edu/mambo

Scales to terabytes of data, e.g., networks with 2.3 billion edges and over 2,000 modes

Outline of this Lecture

1) Deep Graph Neural Networks ✓
2) Polypharmacy & Drug Interactions ✓
3) Drug Repurposing ✓
4) New Directions and Opportunities ✓
5) Practical Advice and Demos



Practical Advice and Demos

Deep Learning for Network Biology

How to Start?

Lecture Resources

- **MAMBO:** Multimodal biomedical networks
  - Scales to networks with 2.3 billion edges and over 2,000 modes
  - snap.stanford.edu/mambo
- **Network data:**
  - snap.stanford.edu/projects.html:
    - CRank, Decagon, MAMBO, NE, OhmNet, Pathways, Tree of Life, and many others
  - snap.stanford.edu/biodata
    - Algorithm benchmarking, method development
    - Easy to link entities across datasets

Networks and relationships

Name	Edges	Entities	Description
CC-Neuron	49,471,006	cell, cell	Similarity network between cells in embroyonic mouse brain
ChCh-Miner	96,137	drug, drug	Interactions between FDA-approved drugs
ChChSe-Decagon	4,649,441	drug, drug, side-effect	Side effects of drug combinations
ChG-InterDecagon	131,034	drug, gene	Chemical-gene interaction network

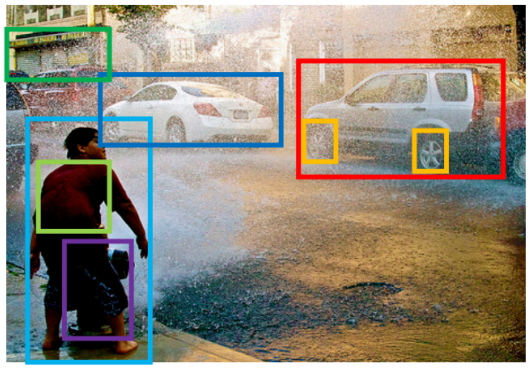
Easy Deep Learning on Graphs

- Node2vec:
  - https://github.com/aditya-grover/node2vec (Python)
  - https://github.com/snap-stanford/snap/tree/master/examples/node2vec (C++)
- Graph Convolutional Networks (GCNs):
  - https://github.com/tkipf/gcn (Tensorflow)
  - https://github.com/tkipf/pygcn (PyTorch)
  - https://github.com/tkipf/keras-gcn (Keras)
- GraphSAGE:
  - https://github.com/williamleif/GraphSAGE (Tensorflow)
  - https://github.com/williamleif/graphsage-simple (Pytorch)
- Metapath2vec and metapath2vec++ (Python):
  - https://ericdongyx.github.io/metapath2vec/m2v.html
- OhmNet (Python):
  - https://github.com/marinkaz/ohmnet
- Decagon (Tensorflow):
  - https://github.com/marinkaz/decagon
- GraphNets (Tensorflow):
  - https://github.com/deepmind/graph_nets
- Deep Graph Library (PyTorch):
  - https://github.com/deepmind/graph_nets

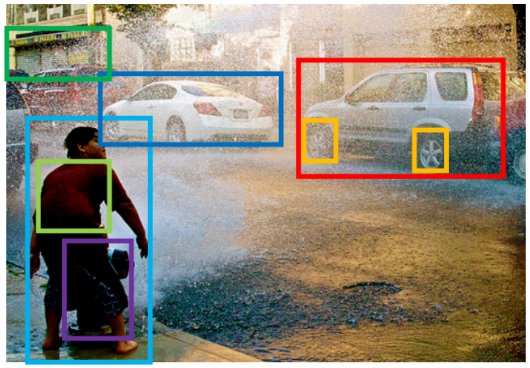
Recap

1. **You have been using graphs in your projects:**
   - Keep on using them!
   - No tedious feature engineering necessary anymore
   - Combine node/edge attributes with extra information
   - End-to-end training can achieve SotA performance

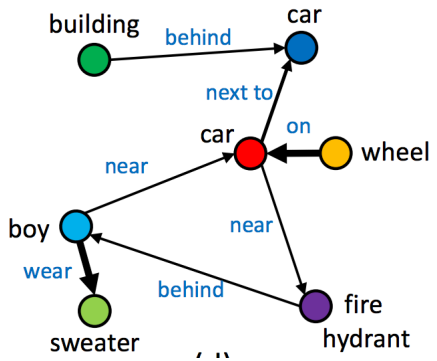
1. **You haven't used graphs yet:**



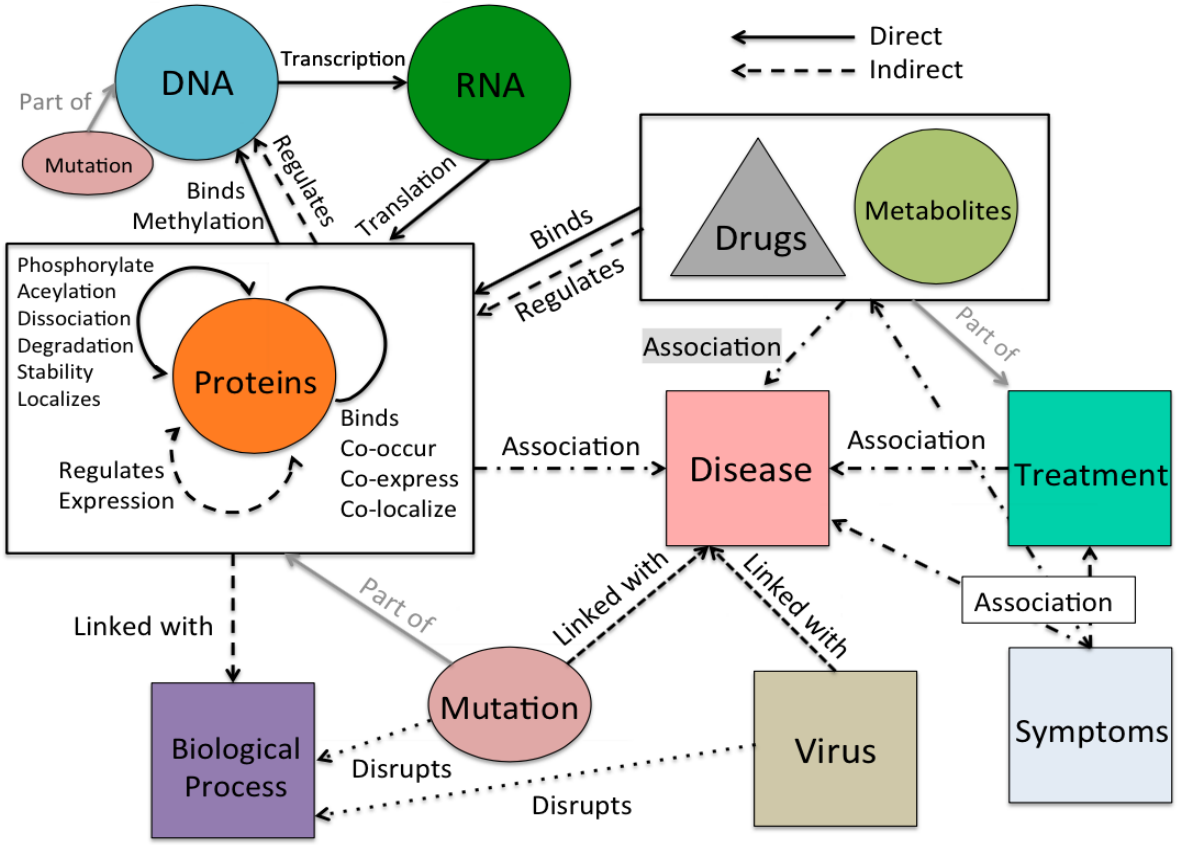
vs.



+



Network Biology and Medicine

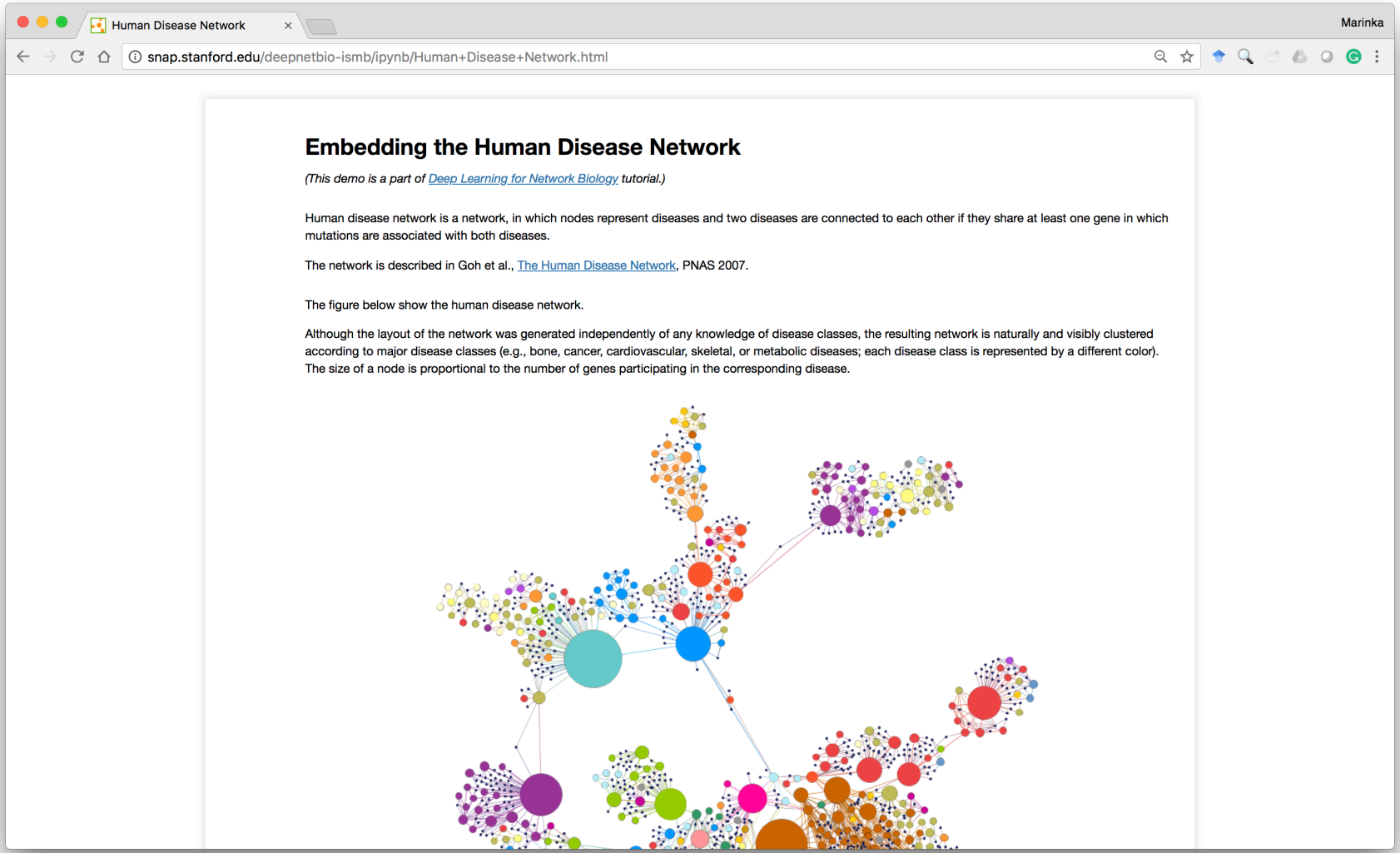


- Large networks of interactions
- Opportunity to integrate knowledge with diverse experimental readouts and perform discovery

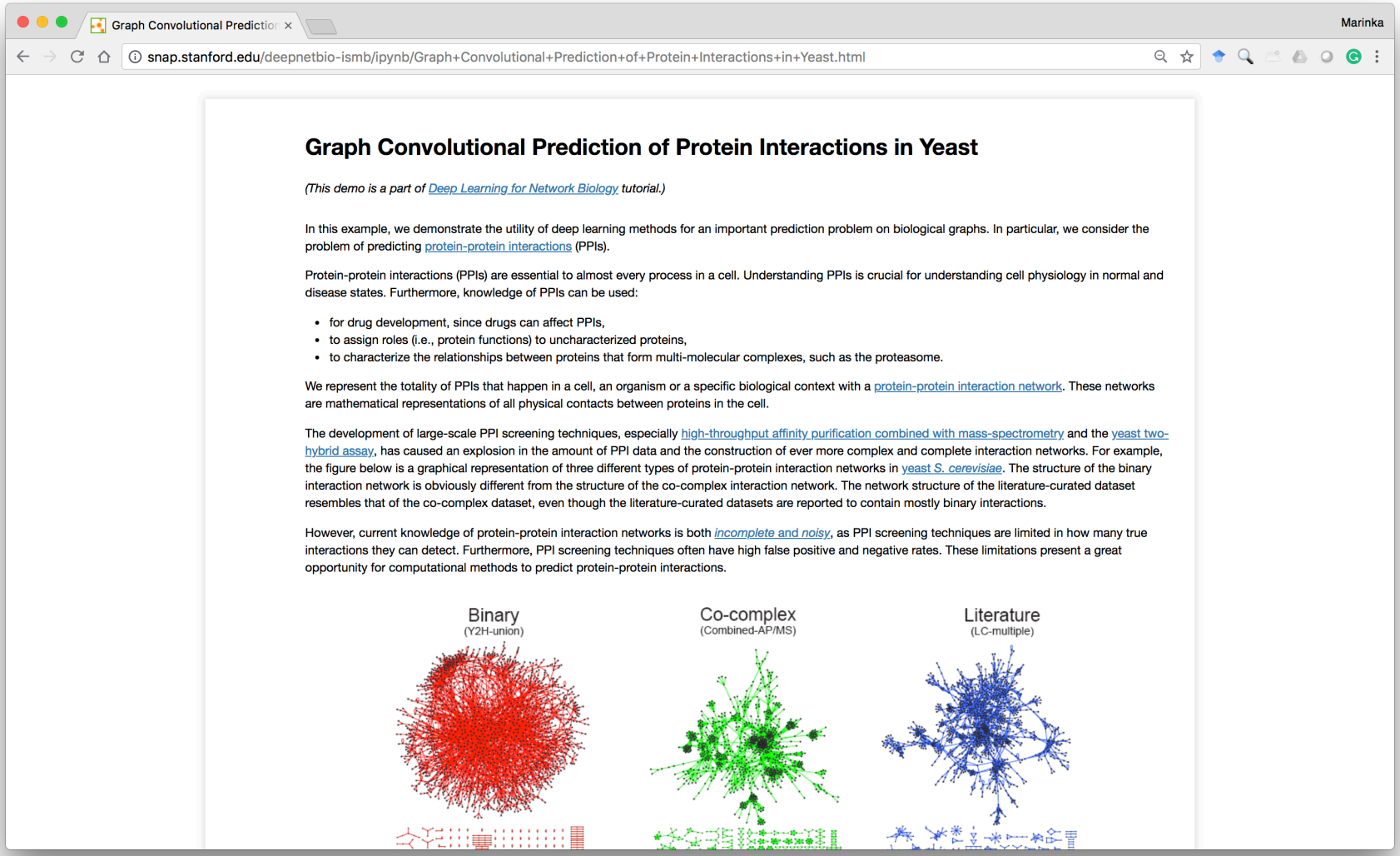
How can this technology be used for biomedical problems?

- **Node prediction:** E.g., Predicting tissue-specific protein functions
- **Pairs of nodes:** E.g., Predicting side-effects of drug combinations
- **Subgraph prediction:** E.g., Predicting which drug treats what disease
- **Graph prediction:** E.g., Predicting properties of molecular graphs

Demo: Diseases



Demo: Protein Interactions

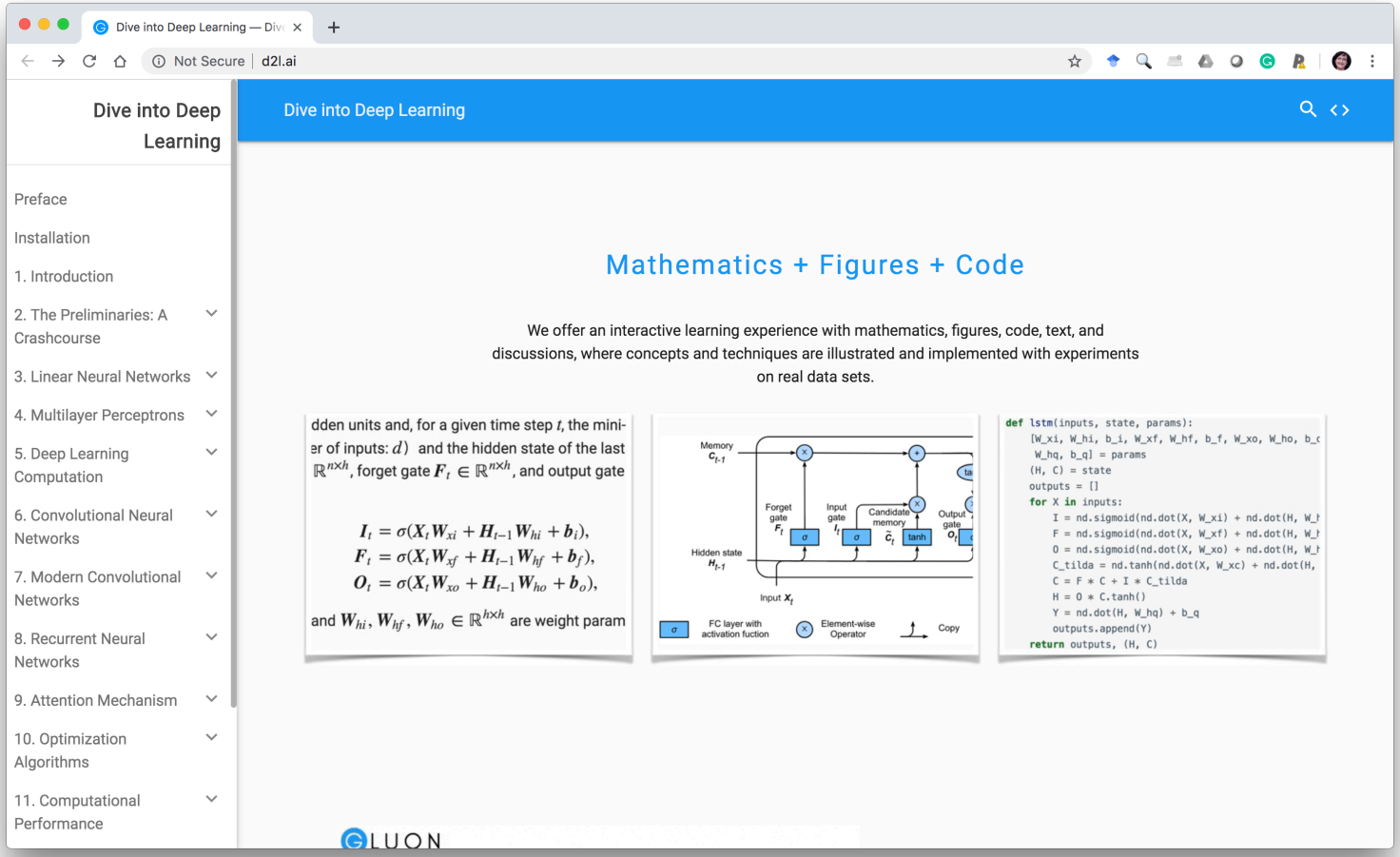


General Tips

1) Network preprocessing is important:
   - Renormalization tricks, variance-scaled initialization, data whitening
2) Use the ADAM optimizer, decay the learning rate
3) ReLU often works really well
4) No activation function at output layer:
   - Easy mistake if layers are built with a shared function
5) Include bias term in every layer
6) Graph convolution layer of size 64 or 128 is plenty
7) Large graphs that cannot fit on one GPU card:
   - Sampling and batching across samples for maximum parallelism

General Tips

https://d2l.ai



Debugging Deep Networks

- Debug?!:
  - Loss/accuracy not converging during training
- Important for model development:
  - **Overfit on training data:**
    - Accuracy should be essentially 100% or error close to 0
    - If neural network cannot overfit a single data point, something is wrong
  - **Scrutinize your loss function!**
  - **Scrutinize your visualizations!**

Outline of this Lecture

1) Deep Graph Neural Networks ✓
2) Polypharmacy & Drug Interactions ✓
3) Drug Repurposing ✓
4) New Directions and Opportunities ✓
5) Practical Advice and Demos ✓

Two Lectures

Part 1: May 15, 2019, 2:30 pm - 4:00 pm

- Methodology: Shallow network embeddings:
  - Map nodes to low-dimensional features
- Resources: Data, tools, codebases
- Applications: PPIs, Disease pathways, Tissues

Part 2: May 16, 2019, 9:00 am – 10:30 am

- Methodology: Deep network embeddings:
  - Graph neural networks for rich biomedical graphs
- Resources: Data, practical advice and demos
- Applications: Polypharmacy, Drug repurposing