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.

Heterogeneity
How to learn deep models on biomedical networks?

Predictions, e.g., properties of cells, patient outcomes, new relationships like disease-gene associations, new functional modules
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.
Embedding Nodes

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

- **Input network**
- **d-dimensional embedding space**
- **Encode nodes**: $\text{ENC}(u)$ and $\text{ENC}(v)$
- **Nodes in d-dimensional space**: $Z_u$ and $Z_v$
Embedding Nodes

Goal: \( \text{similarity}(u, v) \approx z_v^T z_u \)

Need to define!

Input network

d-dimensional embedding space
Two Key Components

- **Encoder**: Map a node to a low-dimensional vector:
  \[
  \text{ENC}(v) = z_v
  \]
  node in the input graph

- **Similarity function** defines how relationships in the input network map to relationships in the embedding space:
  \[
  \text{similarity}(u, v) \approx z_v^\top z_u
  \]
  dot product between node embeddings

Similarity of \( u \) and \( v \) in the network
So Far: Shallow Encoders

Shallow encoders:

- One-layer of data transformation

- A single hidden layer maps node $u$ to embedding $z_u$ via function $f$:
  \[ z_u = f(z_v, v \in N_R(u)) \]
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:

\[ ENC(v) = \text{multiple layers of non-linear transformation of graph structure} \]
A Naïve Approach

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

We need to generalize convolutions beyond simple lattices to multimodal networks and leverage node features/attributes.
**Approach:** Deep Learning for Multimodal Networks

**Input:** Multimodal network

**Output:** Predictions, e.g., properties of drugs and proteins, new protein-protein and drug-drug associations
Multimodal Networks

Mode 1
- e.g., drugs
- E.g., Specific type of drug-drug interaction ($r_1$)

Mode 2
- e.g., proteins
- E.g., drug-target interaction ($r_4$)
- E.g., protein-protein interaction ($r_5$)

Edge type $i$  
Node types  
Feature vector
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
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 for edge type $r_3$:

- **1st order neighbor of $v$**
- **2nd order neighbor of $v$**
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:
  \[ h_v^{(0)} = x_v \]

- Per-layer update of node embeddings:
  \[ h_v^{(k)} = \phi \left( \sum_r \sum_{u \in N_r^v} c_{uv} W_r^{(k-1)} h_u^{(k-1)} + c_v^{(k-1)} h_v^{(k-1)} \right) \quad k = 1, \ldots, K \]

- Embeddings after \( K \) layers of neighborhood aggregation:
  \[ z_v = h_v^{(K)} \]
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

Parameter weight matrices

Probability that \( C \) and \( S \) are linked by an edge of type \( r_8 \)

Two input nodes, \( C \) and \( S \)

Tensor factorized model captures dependences between different edge types

Predicted edges

\[
p(C, r_1, S) = \sigma(z_c^T D_{r_1} R D_{r_1} z_s)
\]

\[
p(C, r_2, S) = \sigma(z_c^T D_{r_2} R D_{r_2} z_s)
\]

\[
p(C, r_3, S) = \sigma(z_c^T D_{r_3} R D_{r_3} z_s)
\]

\[
p(C, r_4, S) = \sigma(z_c^T D_{r_4} R D_{r_4} z_s)
\]

\[
p(C, r_n, S) = \sigma(z_c^T D_{r_n} R D_{r_n} z_s)
\]

\( R, D_{r_i} \) 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).
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.
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)

<table>
<thead>
<tr>
<th>Drugs taken</th>
<th>Unwanted side effects</th>
</tr>
</thead>
<tbody>
<tr>
<td>![Red Pill]</td>
<td>Peliosis hepatis (5%), Heart rate increased (10%), Aortic aneurysm (3%)</td>
</tr>
<tr>
<td>![Green Pill]</td>
<td>Joint stiffness (30%), Joint swelling (10%), Bone marrow fibrosis (3%)</td>
</tr>
<tr>
<td>![Green Pill]</td>
<td>Anaemia (15%), Bone marrow fibrosis (5%), Intestinal ulcer (5%)</td>
</tr>
<tr>
<td>![Green Pill]</td>
<td>Anaemia (15%), Bone marrow fibrosis (5%), Intestinal ulcer (5%), Joint stiffness (30%), Joint swelling (10%), Colon cancer (4%), Fatigue (40%)</td>
</tr>
<tr>
<td>![Red Pill]</td>
<td>Peliosis hepatis (5%), Heart rate increased (10%), Aortic aneurysm (3%), Joint stiffness (30%), Joint swelling (10%), Bone marrow fibrosis (3%)</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>
Reports on Unwanted Side Effects

The FDA Adverse Event Reporting System (FAERS)

<table>
<thead>
<tr>
<th>Drugs taken</th>
<th>Unwanted side effects</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Peliosis hepatis (5%), Heart rate increased (10%), Aortic</td>
</tr>
<tr>
<td></td>
<td>aneurysm (3%)</td>
</tr>
<tr>
<td></td>
<td>Joint stiffness (30%), Joint swelling (10%), Bone marrow</td>
</tr>
<tr>
<td></td>
<td>fibrosis (3%)</td>
</tr>
<tr>
<td></td>
<td>Anaemia (15%), Bone marrow fibrosis (5%), Intestinal</td>
</tr>
<tr>
<td></td>
<td>(5%)</td>
</tr>
<tr>
<td></td>
<td>Anaemia (15%), Bone marrow fibrosis (5%), Intestinal ulcer</td>
</tr>
<tr>
<td></td>
<td>(5%), Joint stiffness (30%), Joint swelling (10%), Colon</td>
</tr>
<tr>
<td></td>
<td>cancer (4%), Fatigue (40%)</td>
</tr>
<tr>
<td></td>
<td>Peliosis hepatis (5%), Heart rate increased (10%), Aortic</td>
</tr>
<tr>
<td></td>
<td>aneurysm (3%), Joint stiffness (30%), Joint swelling (10%),</td>
</tr>
<tr>
<td></td>
<td>Bone marrow fibrosis (3%)</td>
</tr>
<tr>
<td></td>
<td>...</td>
</tr>
</tbody>
</table>
Unexpected Drug Interactions

Co-prescribed drugs

Task: How likely will a particular combination of drugs lead to a particular side effect?

Side Effects

3% prob., 2% prob.
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

Objective:
Capture molecular, drug, and patient data for all drugs prescribed in the U.S.

We build a unique dataset:
- 4,651,131 drug-drug edges:
  - Patient data from an adverse event system, tested for confounders [FDA]
- 18,596 drug-protein edges
- 719,402 protein-protein edges:
  - Physical, metabolic enzyme-coupled, and signaling interactions

Drug and protein features:
- Drugs' chemical structure
- Proteins' membership in pathways

Gastrointestinal bleed side effect
Bradycardia side effect
Nausea side effect
Mumps side effect

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?

Simvastatin

Ciprofloxacin

$r_2$ (breakdown of muscle tissue)
Results: Side Effect Prediction

<table>
<thead>
<tr>
<th>Method</th>
<th>AUROC</th>
<th>AP@50</th>
</tr>
</thead>
<tbody>
<tr>
<td>Our method (Decagon)</td>
<td>0.834</td>
<td></td>
</tr>
<tr>
<td>RESCAL Tensor Factorization</td>
<td>0.693</td>
<td>0.476</td>
</tr>
<tr>
<td>Multi-relational Factorization</td>
<td>0.705</td>
<td>0.567</td>
</tr>
<tr>
<td>Shallow Network Embedding</td>
<td>0.725</td>
<td>0.643</td>
</tr>
</tbody>
</table>

[Nickel et al., ICML'11]
[Perros, Papalexakis et al., KDD'17]
[Zong et al., Bioinformatics'17]
Novel Predictions

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

<table>
<thead>
<tr>
<th>Rank</th>
<th>Drug</th>
<th>Drug</th>
<th>Side effect</th>
<th>Evidence found</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Pyrimethamine</td>
<td>Aliskiren</td>
<td>Sarcoma</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Tigecycline</td>
<td>Bimatoprost</td>
<td>Autonomic</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Telangiectases</td>
<td>Omeprazole</td>
<td>Dacarbazine</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Tolcapone</td>
<td>Pyrimethamine</td>
<td>Blood brain</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>Minocycline</td>
<td>Pyrimethamine</td>
<td>Blood brain</td>
<td></td>
</tr>
</tbody>
</table>

Case Report

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

- Ache<br>- Azelastine acid<br>- Cerebral thrombosis<br>- Muscle inflammation<br>- Breast inflammation<br>- Endometriosis
Utility of Predictions in the Clinic

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

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

Patient 1

Patient 2

Patient 3 put on an anti-nausea med

Time

No anti-nausea med

No anti-nausea med

With David Berkowitz, Adam Wright, Tina Hernandez-Boussard Labs
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.

Drug discovery 3–6 years
Preclinical testing 3 years
Phase I Phase II 3 years
Phase III 2 years
FDA approval 1–2 years

12–16 years, ~$1 billion to $2 billion

A shorter timescale
Because most repositioned drugs have already passed the early phases of development and clinical testing, they can potentially win approval in less than half the time and at one-quarter of the cost.

Drug repositioning
~6 years, ~$300 million
Key Insight: Subgraphs

**Disease:** Subgraph of rich protein network defined on disease proteins

**Drug:** Subgraph of rich protein network defined on drug’s target proteins

A drug likely treats a disease if it is close to the disease in **pharmacological space** [Paolini et al., Nature Biotech.’06; Menche et al., Science’15]

**Idea:** Use the paradigm of embeddings to operationalize the concept of closeness in pharmacological space
**Task:** Given drug $C$ and disease $D$, predict if $C$ treats $D$

**Task:** 1) Learn embeddings for $C$’s and $D$’s subgraphs 2) Use embeddings to predict link between $C$ and $D$
SUGAR: Neural Message Passing

Aggregate information from subgraphs

Aggregate information from neighbors

Message

$p(C, D)$

Edge decoder

Subgraph encoder
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

<table>
<thead>
<tr>
<th>Approach</th>
<th>AUPRC</th>
<th>AUROC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Our method (SUGAR)</td>
<td>0.851</td>
<td>0.888</td>
</tr>
</tbody>
</table>

- Graphlets [Bioinformatics’13]
- Bi-directional random walks [Bioinformatics’16]
- Heterogeneous graph inference [Bioinformatics’14]
- Drug-disease closeness [Nat. Commun.’17]
- Gene-based network overlap [Nat. Commun.’17]

**Up to 49% improvement**

**Up to 172% improvement**
Side Information further improves performance

<table>
<thead>
<tr>
<th>Metabolic pathways</th>
<th>Molecular functions</th>
<th>Biological processes</th>
<th>Cellular components</th>
<th>AUPRC</th>
<th>AUROC</th>
</tr>
</thead>
<tbody>
<tr>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>0.851</td>
<td>0.888</td>
</tr>
<tr>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td>0.869</td>
<td>0.893</td>
</tr>
<tr>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>0.874</td>
<td>0.912</td>
</tr>
<tr>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>0.893</td>
<td>0.912</td>
</tr>
<tr>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>0.901</td>
<td>0.928</td>
</tr>
</tbody>
</table>
Drug Repurposing at Stanford

<table>
<thead>
<tr>
<th>Drug</th>
<th>Disease</th>
<th>Rank:</th>
</tr>
</thead>
<tbody>
<tr>
<td>N-acetyl-cysteine</td>
<td>cystic fibrosis</td>
<td>36/5000</td>
</tr>
<tr>
<td>Xamoterol</td>
<td>neurodegeneration</td>
<td>10/5000</td>
</tr>
<tr>
<td>Plerixafor</td>
<td>cancer</td>
<td>26/5000</td>
</tr>
<tr>
<td>Sodium selenite</td>
<td>cancer</td>
<td>26/5000</td>
</tr>
<tr>
<td>Ebselen</td>
<td>C difficile</td>
<td>11/5000</td>
</tr>
<tr>
<td>Itraconazole</td>
<td>cancer</td>
<td>16/5000</td>
</tr>
<tr>
<td>Bestatin</td>
<td>lymphedema</td>
<td>28/5000</td>
</tr>
<tr>
<td>Ketoprofen</td>
<td>lymphedema</td>
<td>26/5000</td>
</tr>
<tr>
<td>Sildenafil</td>
<td>lymphatic malformation</td>
<td>46/5000</td>
</tr>
<tr>
<td>Tacrolimus</td>
<td>pulmonary arterial hypertension</td>
<td>114/5000</td>
</tr>
<tr>
<td>Benzamil</td>
<td>psoriasis</td>
<td>9/5000</td>
</tr>
<tr>
<td>Carvedilol</td>
<td>Chagas’ disease</td>
<td></td>
</tr>
</tbody>
</table>

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

Led to follow-up research on prostate cancer and schizophrenia at Stanford Medical School
Introducing Feedbacks for AI Loop

Domain expert

Will Benzamil treat psoriasis?

a. Heterogeneous biomedical network

b. Deep graph convolutional model

c. Predictions

Psoriasis
Ebselen $p = 0.96$
Bestatin $p = 0.84$
Benzamil $p = 0.76$
Sirolimus $p = 0.54$
...

What data can explain these predictions?

Drug
Disease
Protein
Molecular pathway
Drug side effect

“Olfactory signaling” pathways
“Innate immune response” pathways

Expert panel

Stanford Medicine
SPARK Translational Research Program
From Bench to Bedside
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:
• 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 ‘Eureka!’ but also ‘That’s weird!’

**Initial results**: Introduce feedback for the AI loop

<table>
<thead>
<tr>
<th>Networks</th>
<th>Networks</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- Exciting phrase is not only ‘Eureka!’ but also ‘That’s weird!’

**Initial results**: Introduce feedback for the AI loop

- Exciting phrase is not only ‘Eureka!’ but also ‘That’s weird!’

**Initial results**: Introduce feedback for the AI loop

- Exciting phrase is not only ‘Eureka!’ but also ‘That’s weird!’

**Initial results**: Introduce feedback for the AI loop

- Exciting phrase is not only ‘Eureka!’ but also ‘That’s weird!’

**Initial results**: Introduce feedback for the AI loop

- Exciting phrase is not only ‘Eureka!’ but also ‘That’s weird!’

**Initial results**: Introduce feedback for the AI loop

- Exciting phrase is not only ‘Eureka!’ but also ‘That’s weird!’
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

Current machine learning models:
- Only image at input
- Little or no knowledge about the disease or patient
3rd New Direction: Rich Interactions

Initial results: Deep framework for logical queries on knowledge graphs

Query:
Predict drugs $\mathcal{C}$ that might Target proteins, which are in turn Associated with diseases $d_1$ and $d_2$

Initial results: Biomedical network data and tools

<table>
<thead>
<tr>
<th>Networks and re</th>
<th>snap.stanford.edu/biodata</th>
<th>snap.stanford.edu/mambo</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Edges</td>
<td>Entities</td>
</tr>
<tr>
<td>CC-Neuron</td>
<td>49,471,006</td>
<td>cell, cell</td>
</tr>
<tr>
<td>ChCh-Miner</td>
<td>96,137</td>
<td>drug, drug</td>
</tr>
<tr>
<td>ChChSe-Decagon</td>
<td>4,649,441</td>
<td>drug, drug, side-effect</td>
</tr>
<tr>
<td>ChG-InterDecagon</td>
<td>131,034</td>
<td>drug, gene</td>
</tr>
<tr>
<td>ChG-Miner</td>
<td>15,424</td>
<td>drug, gene</td>
</tr>
<tr>
<td>ChG-TargetDecagon</td>
<td>18,690</td>
<td>drug, gene</td>
</tr>
</tbody>
</table>

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](http://snap.stanford.edu/mambo)

- **Network data**:
  - [snap.stanford.edu/projects.html](http://snap.stanford.edu/projects.html):
    - CRank, Decagon, MAMBO, NE, OhmNet, Pathways, Tree of Life, and many others
  - [snap.stanford.edu/biodata](http://snap.stanford.edu/biodata)
  - Algorithm benchmarking, method development
  - Easy to link entities across datasets

Networks and relationships

<table>
<thead>
<tr>
<th>Name</th>
<th>Edges</th>
<th>Entities</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CC-Neuron</td>
<td>49,471,006</td>
<td>cell, cell</td>
<td>Similarity network between cells in embryonic mouse brain</td>
</tr>
<tr>
<td>ChCh-Miner</td>
<td>96,137</td>
<td>drug, drug</td>
<td>Interactions between FDA-approved drugs</td>
</tr>
<tr>
<td>ChChSe-Decagon</td>
<td>4,649,441</td>
<td>drug, drug, side-effect</td>
<td>Side effects of drug combinations</td>
</tr>
<tr>
<td>ChG-InterDecagon</td>
<td>131,034</td>
<td>drug, gene</td>
<td>Chemical-gene interaction network</td>
</tr>
</tbody>
</table>
Easy Deep Learning on Graphs

- **Node2vec:**
  - [https://github.com/aditya-grover/node2vec](https://github.com/aditya-grover/node2vec) (Python)
  - [https://github.com/snap-stanford/snap/tree/master/examples/node2vec](https://github.com/snap-stanford/snap/tree/master/examples/node2vec) (C++)

- **Graph Convolutional Networks (GCNs):**
  - [https://github.com/tkipf/gcn](https://github.com/tkipf/gcn) (Tensorflow)
  - [https://github.com/tkipf/pygcn](https://github.com/tkipf/pygcn) (PyTorch)
  - [https://github.com/tkipf/keras-gcn](https://github.com/tkipf/keras-gcn) (Keras)

- **GraphSAGE:**
  - [https://github.com/williamleif/GraphSAGE](https://github.com/williamleif/GraphSAGE) (Tensorflow)

- **Metapath2vec and metapath2vec++ (Python):**
  - [https://ericdongyx.github.io/metapath2vec/m2v.html](https://ericdongyx.github.io/metapath2vec/m2v.html)

- **OhmNet (Python):**
  - [https://github.com/marinkaz/ohmnet](https://github.com/marinkaz/ohmnet)

- **Decagon (Tensorflow):**
  - [https://github.com/marinkaz/decagon](https://github.com/marinkaz/decagon)

- **GraphNets (Tensorflow):**
  - [https://github.com/deepmind/graph_nets](https://github.com/deepmind/graph_nets)

- **Deep Graph Library (PyTorch):**
  - [https://github.com/deepmind/graph_nets](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:
   - Your datasets (e.g., images, sequences) are not graphs
   - Are you sure?
   - Are examples in the data independent of each other?
   - Introduce graph context in your CNN/RNN models

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
Embedding the Human Disease Network

(This demo is a part of Deep Learning for Network Biology tutorial.)

Human disease network is a network, in which nodes represent diseases and two diseases are connected to each other if they share at least one gene in which mutations are associated with both diseases.

The network is described in Goh et al., The Human Disease Network, PNAS 2007.

The figure below show the human disease network.

Although the layout of the network was generated independently of any knowledge of disease classes, the resulting network is naturally and visibly clustered according to major disease classes (e.g., bone, cancer, cardiovascular, skeletal, or metabolic diseases; each disease class is represented by a different color). The size of a node is proportional to the number of genes participating in the corresponding disease.
Demo: Protein Interactions

Graph Convolutional Prediction of Protein Interactions in Yeast

(This demo is a part of Deep Learning for Network Biology tutorial.)

In this example, we demonstrate the utility of deep learning methods for an important prediction problem on biological graphs. In particular, we consider the problem of predicting protein-protein interactions (PPIs).

Protein-protein interactions (PPIs) are essential to almost every process in a cell. Understanding PPIs is crucial for understanding cell physiology in normal and disease states. Furthermore, knowledge of PPIs can be used:

- for drug development, since drugs can affect PPIs,
- to assign roles (i.e., protein functions) to uncharacterized proteins,
- to characterize the relationships between proteins that form multi-molecular complexes, such as the proteasome.

We represent the totality of PPIs that happen in a cell, an organism or a specific biological context with a protein-protein interaction network. These networks are mathematical representations of all physical contacts between proteins in the cell.

The development of large-scale PPI screening techniques, especially high-throughput affinity purification combined with mass-spectrometry and the yeast two-hybrid assay, has caused an explosion in the amount of PPI data and the construction of ever more complex and complete interaction networks. For example, the figure below is a graphical representation of three different types of protein-protein interaction networks in yeast S. cerevisiae. The structure of the binary interaction network is obviously different from the structure of the co-complex interaction network. The network structure of the literature-curated dataset resembles that of the co-complex dataset, even though the literature-curated datasets are reported to contain mostly binary interactions.

However, current knowledge of protein-protein interaction networks is both incomplete and noisy, as PPI screening techniques are limited in how many true interactions they can detect. Furthermore, PPI screening techniques often have high false positive and negative rates. These limitations present a great opportunity for computational methods to predict protein-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