Improving QSAR Modeling for Predictive Toxicology using Publicly Aggregated Semantic Graph Data and Graph Neural Networks

Joseph D. Romano, Yun Hao, and Jason H. Moore

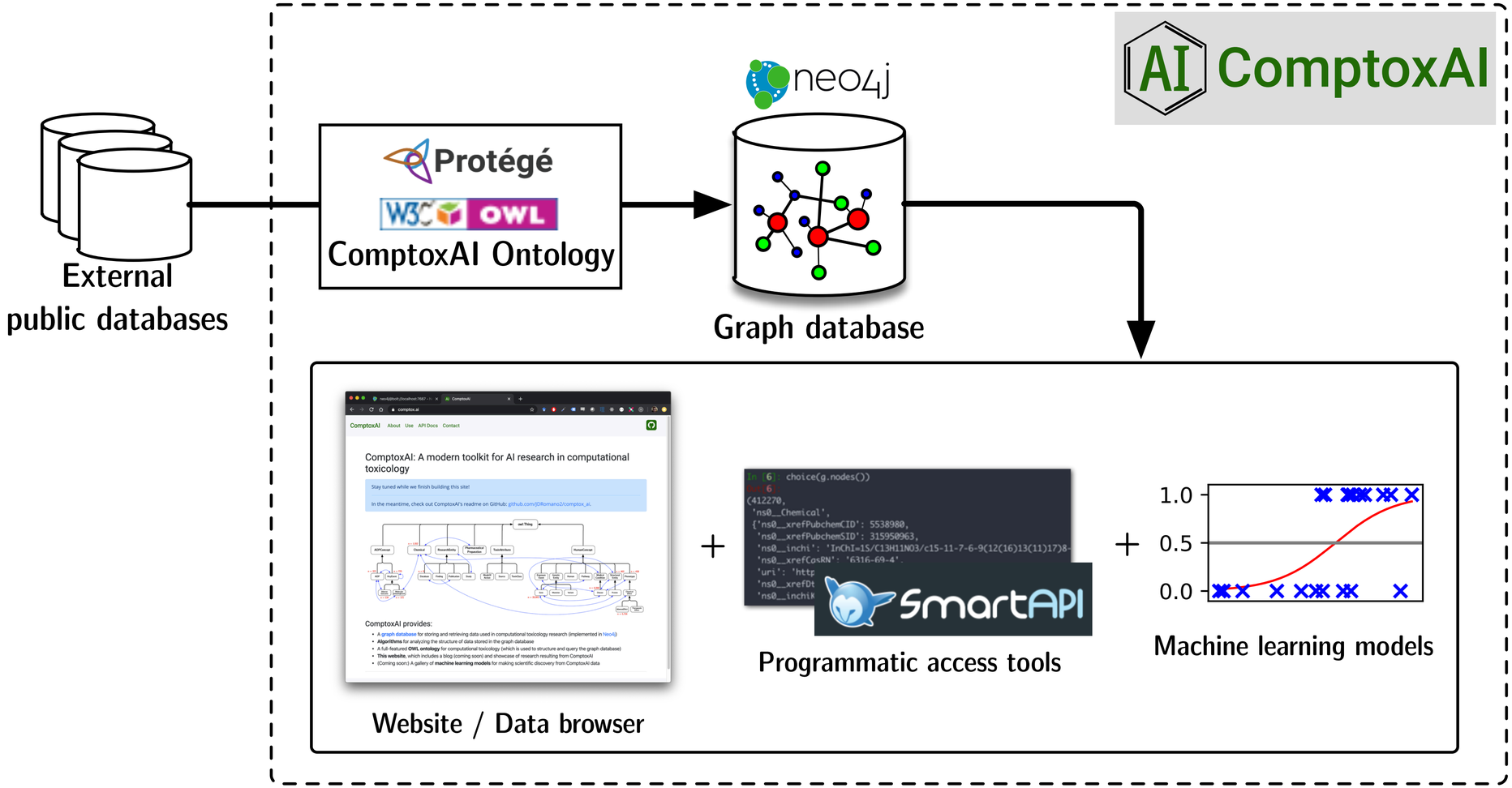
Pacific Symposium on Biocomputing 27:187-198 (2022)

Presented January 6, 2022

Penn
University of Pennsylvania

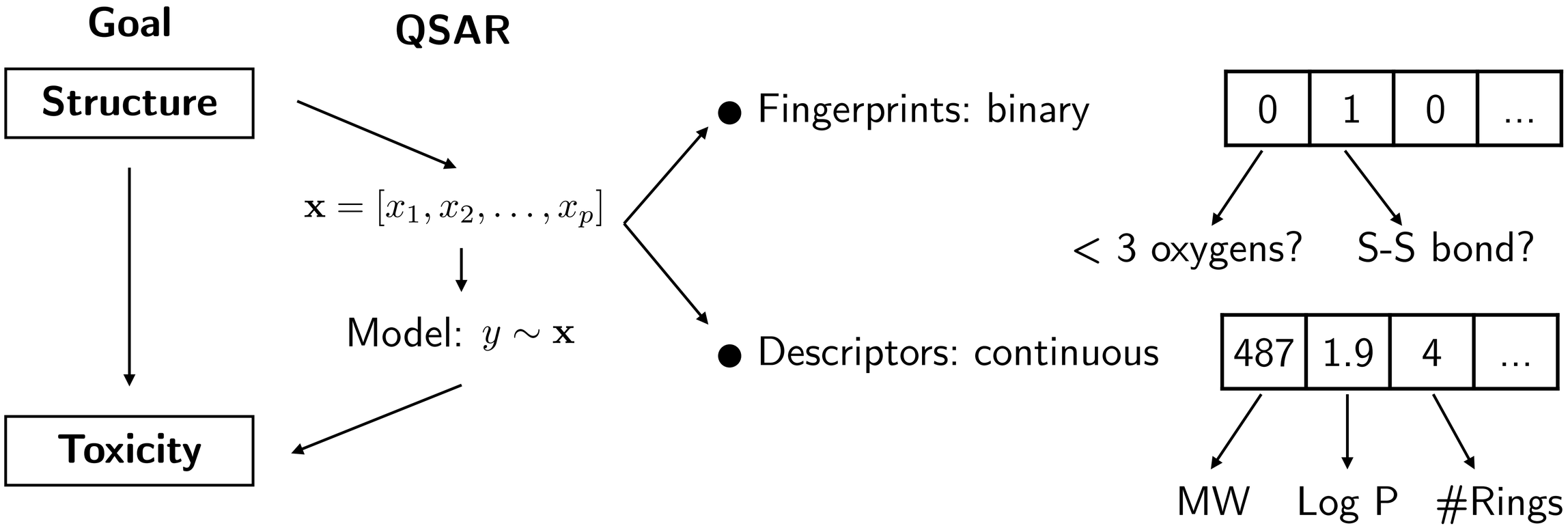
Penn Institute for Biomedical Informatics

ComptoxAI



https://comptox.ai

QSAR

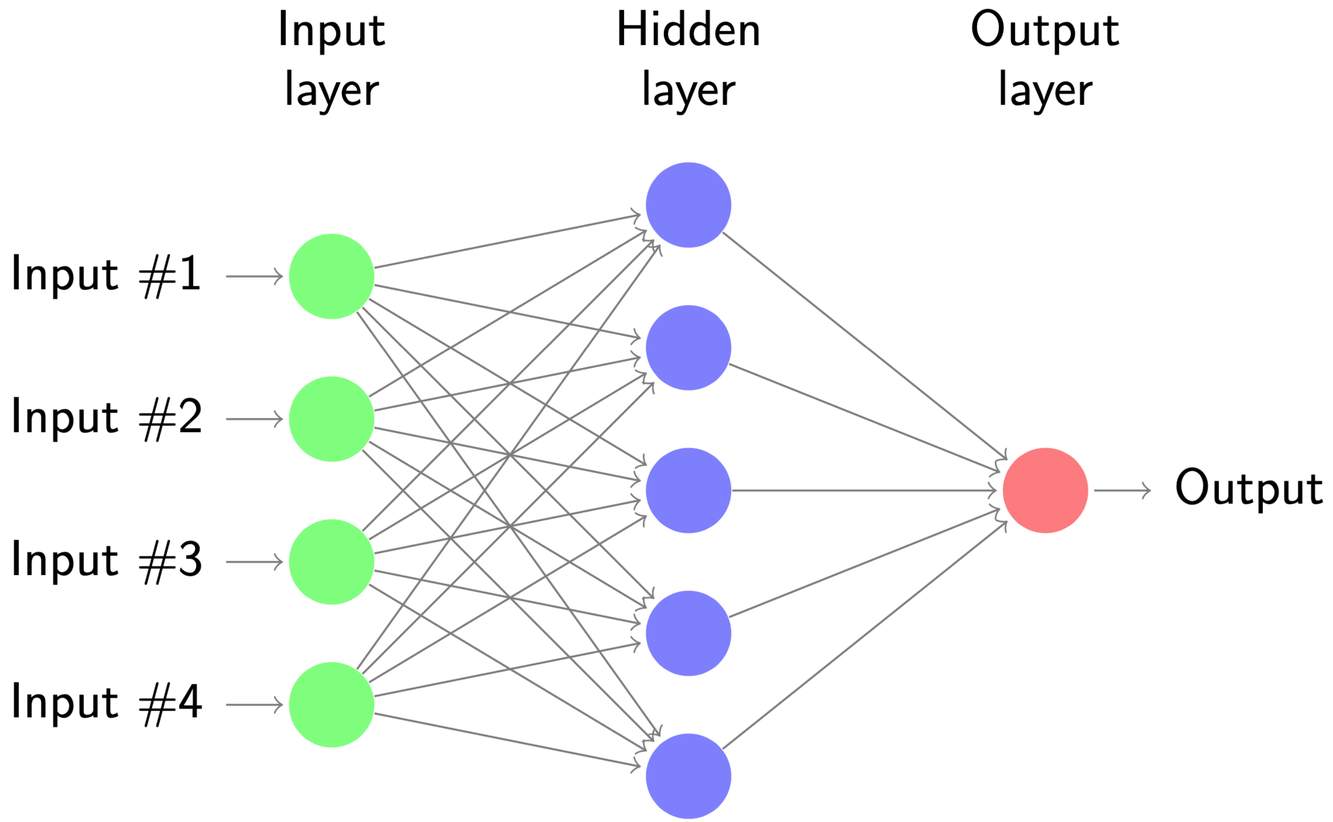


- QSAR: Quantitative Structure-Activity Relationship

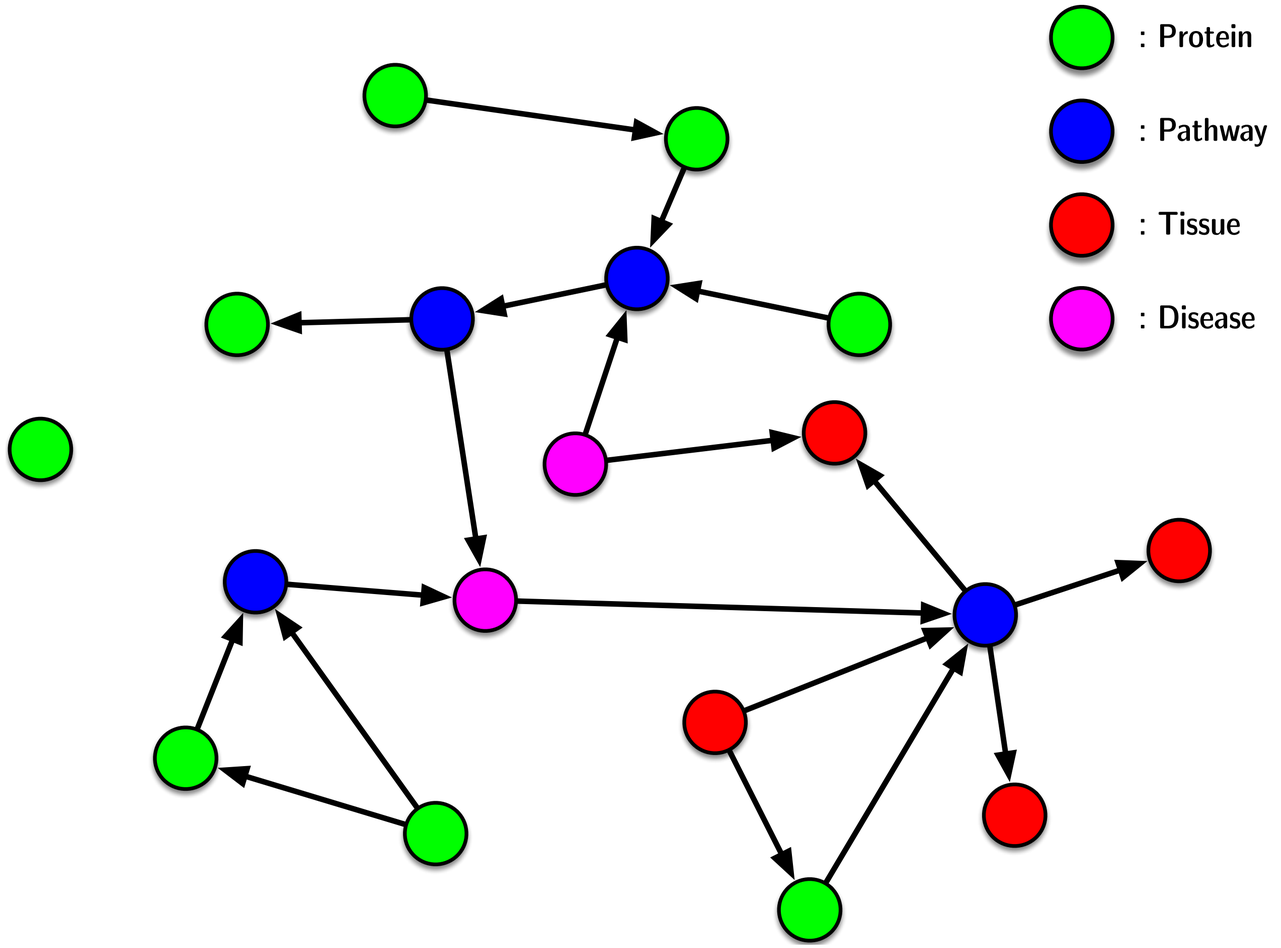
Slide credit Yun Hao

(Artificial) Neural Networks

- Consist of nodes organized into layers, which are usually stacked
- Deep learning —> NN with tens or hundreds of layers

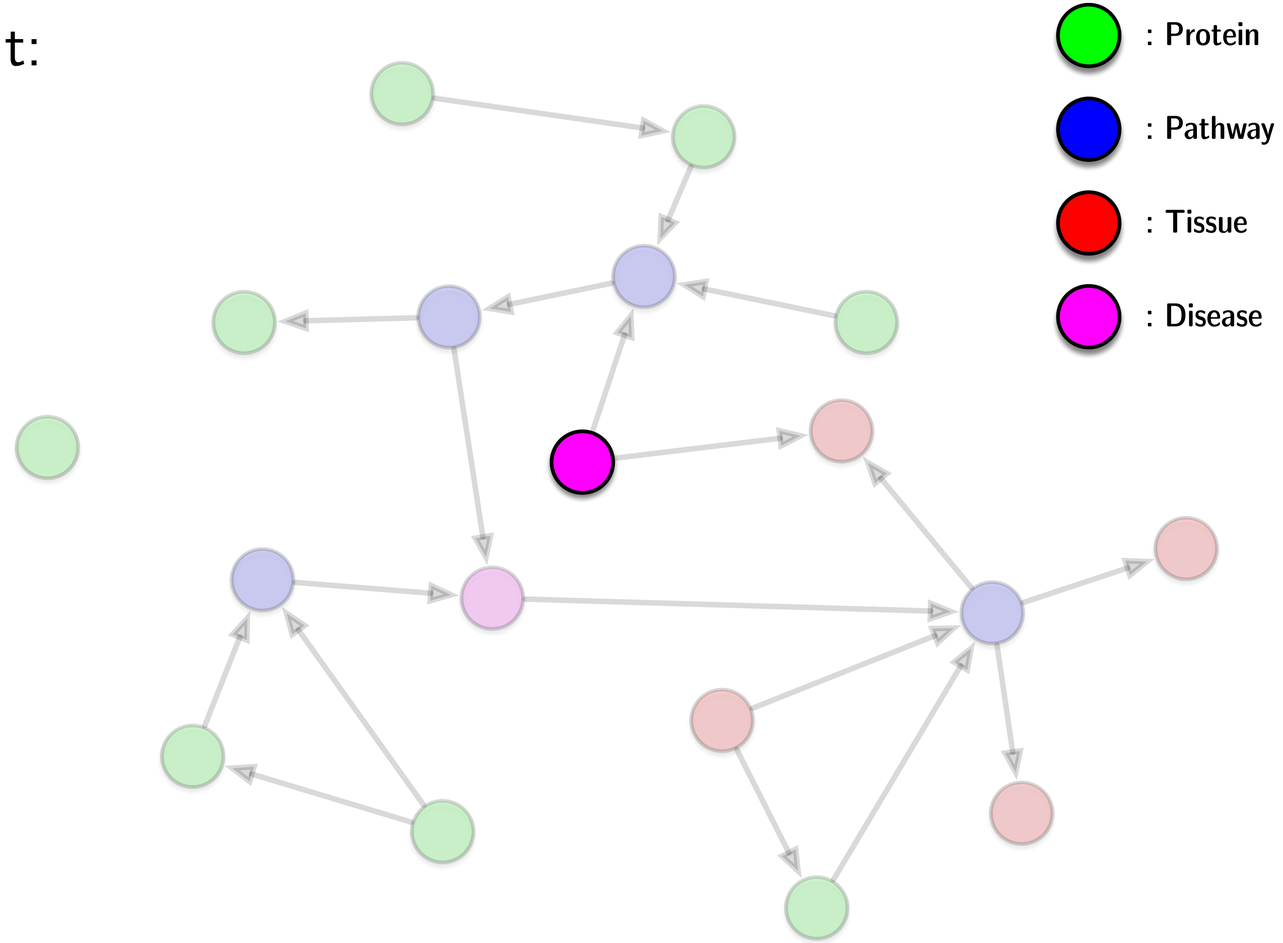


Graph Neural Networks



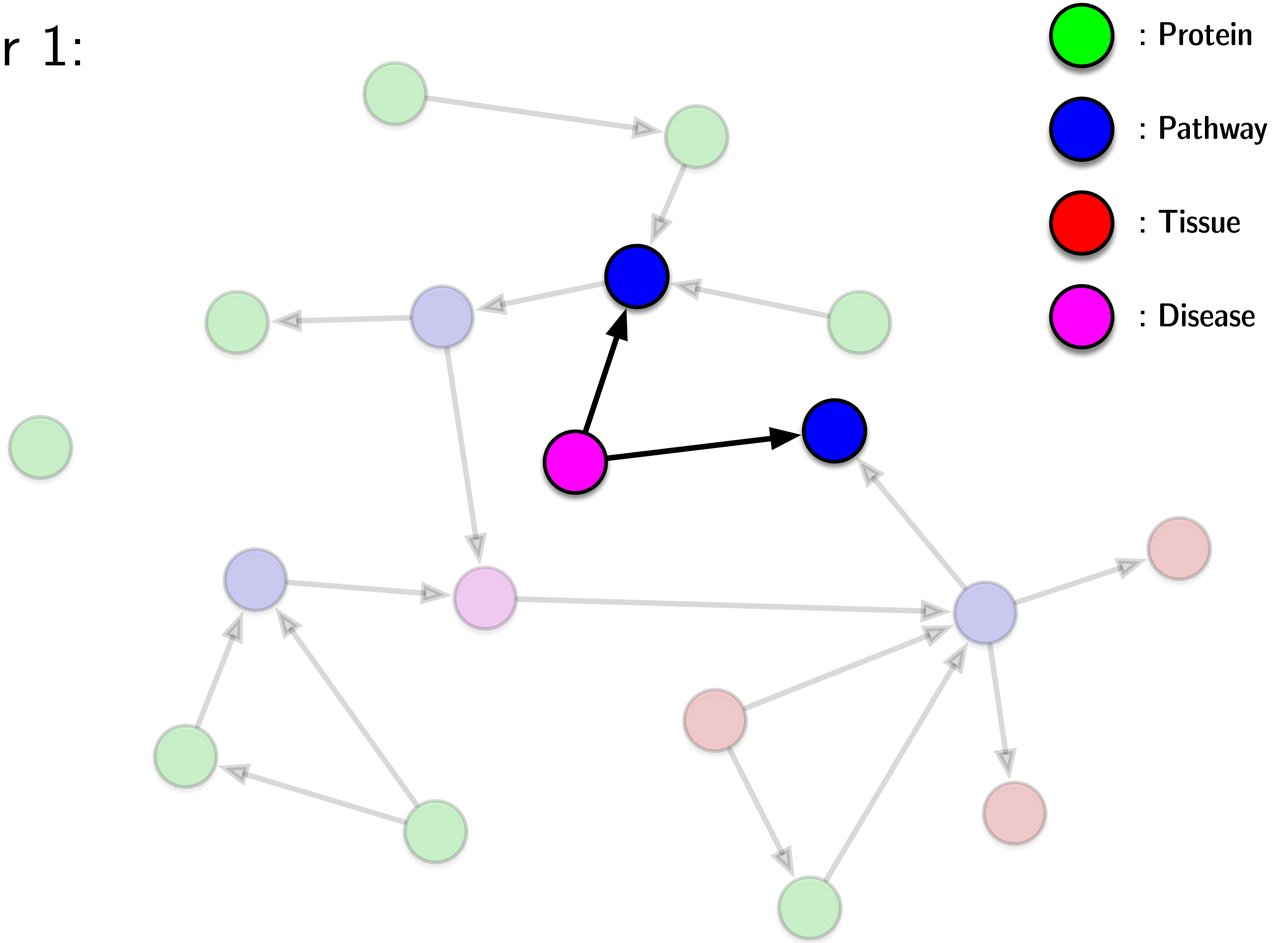
Graph Neural Networks

- Input:



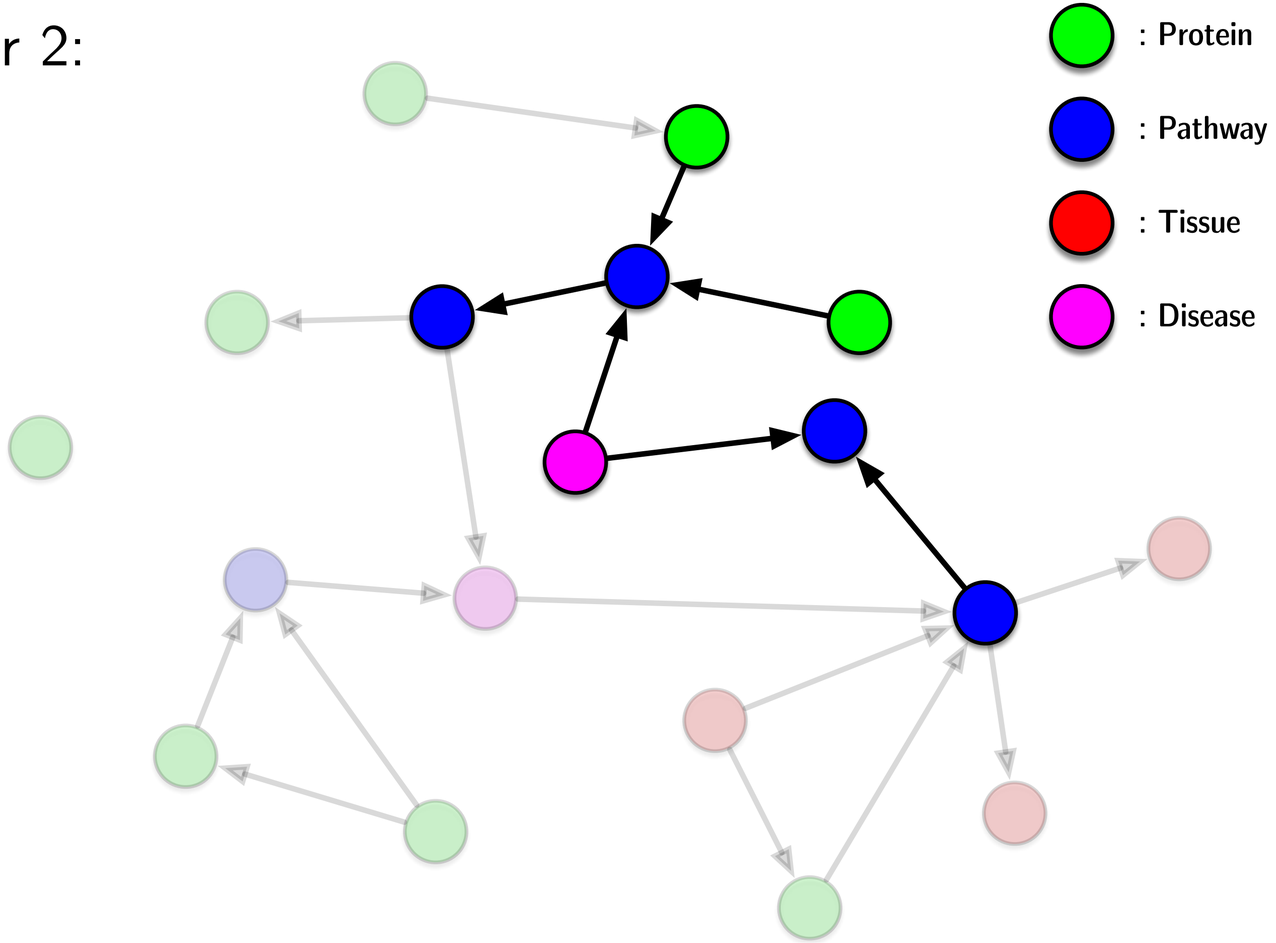
Graph Neural Networks

- Layer 1:



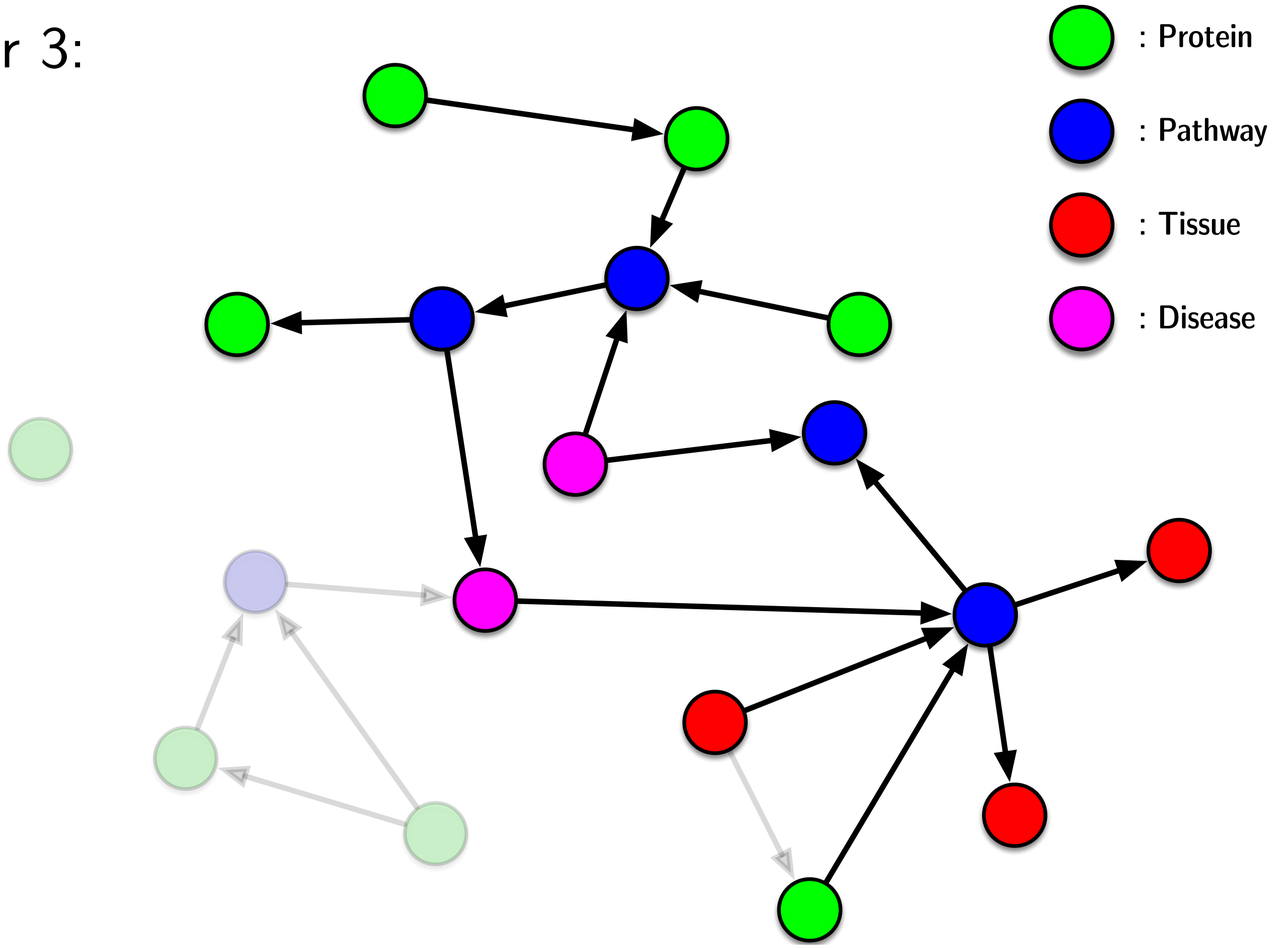
Graph Neural Networks

- Layer 2:

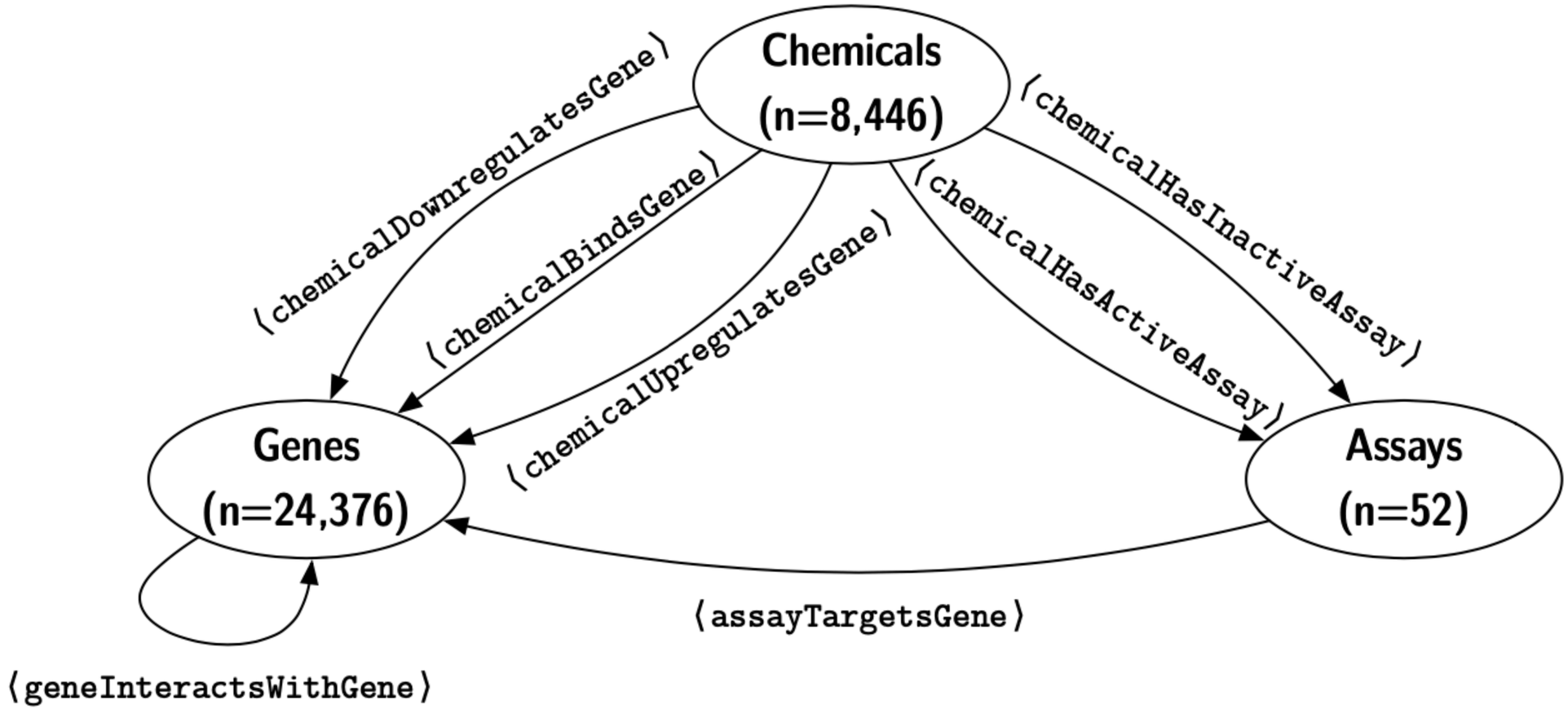


Graph Neural Networks

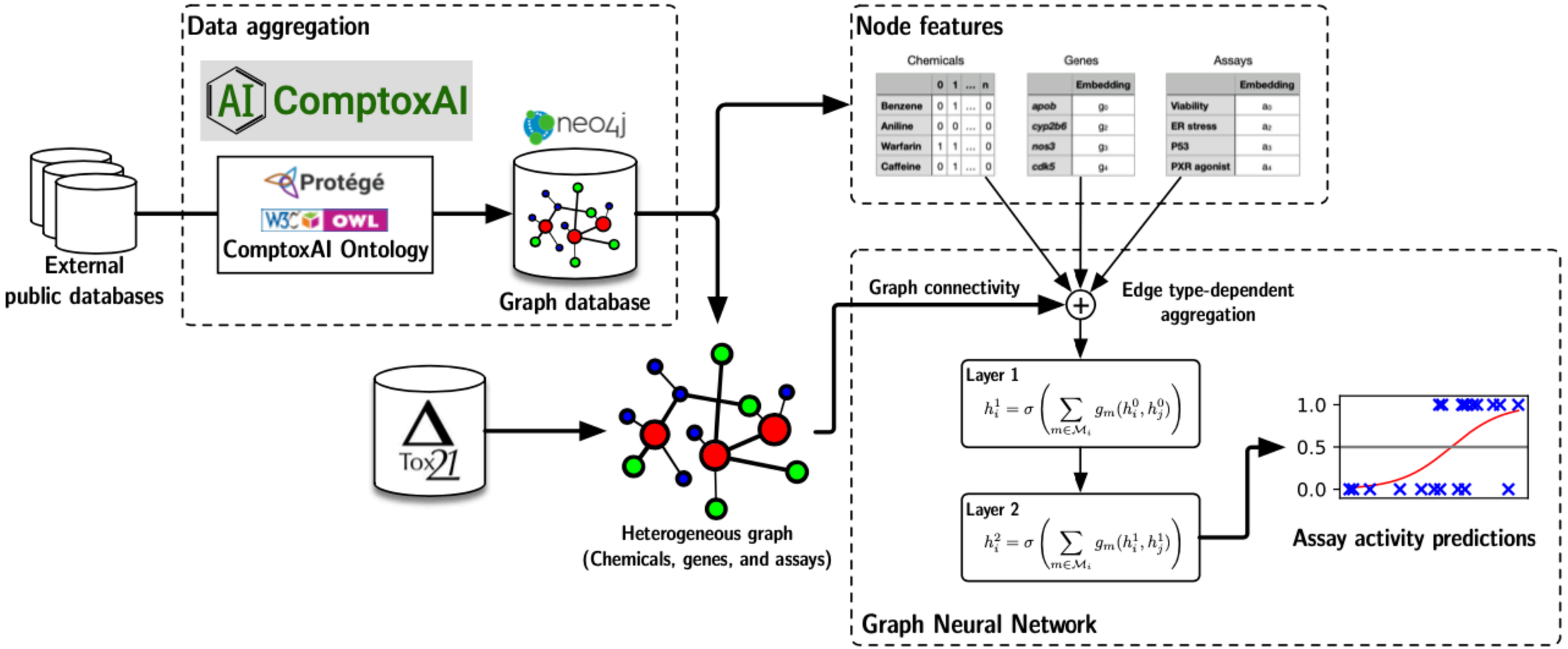
- Layer 3:



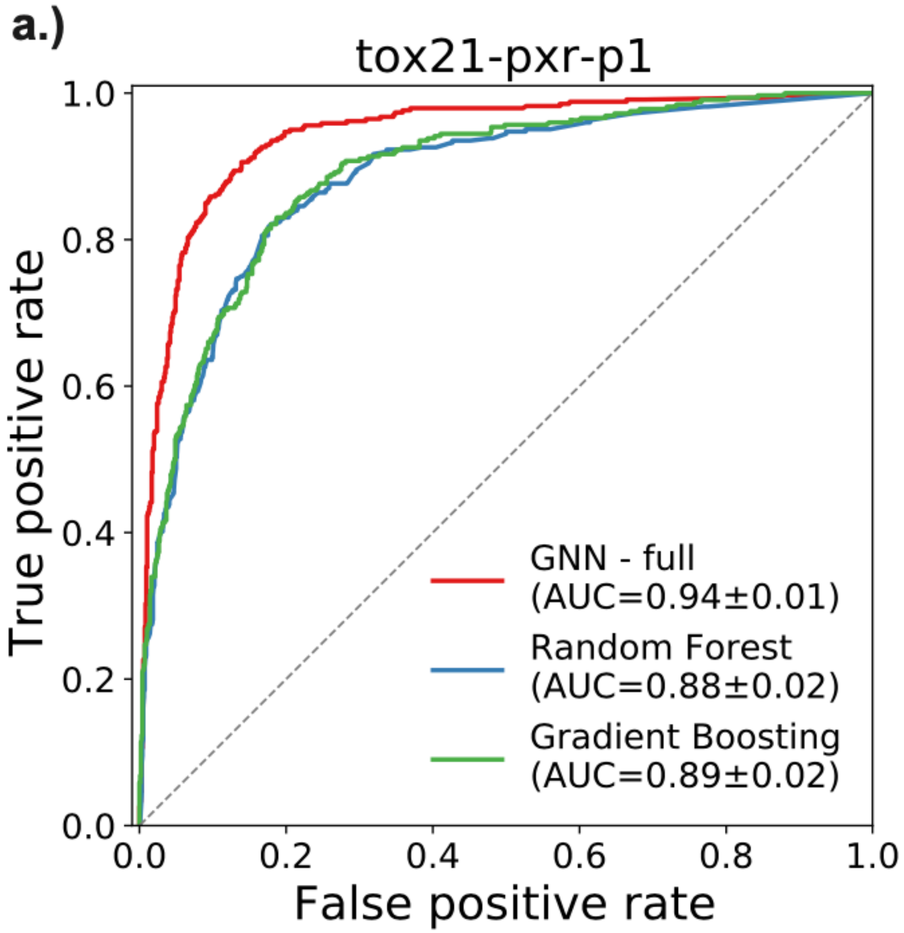
QSAR Subgraph

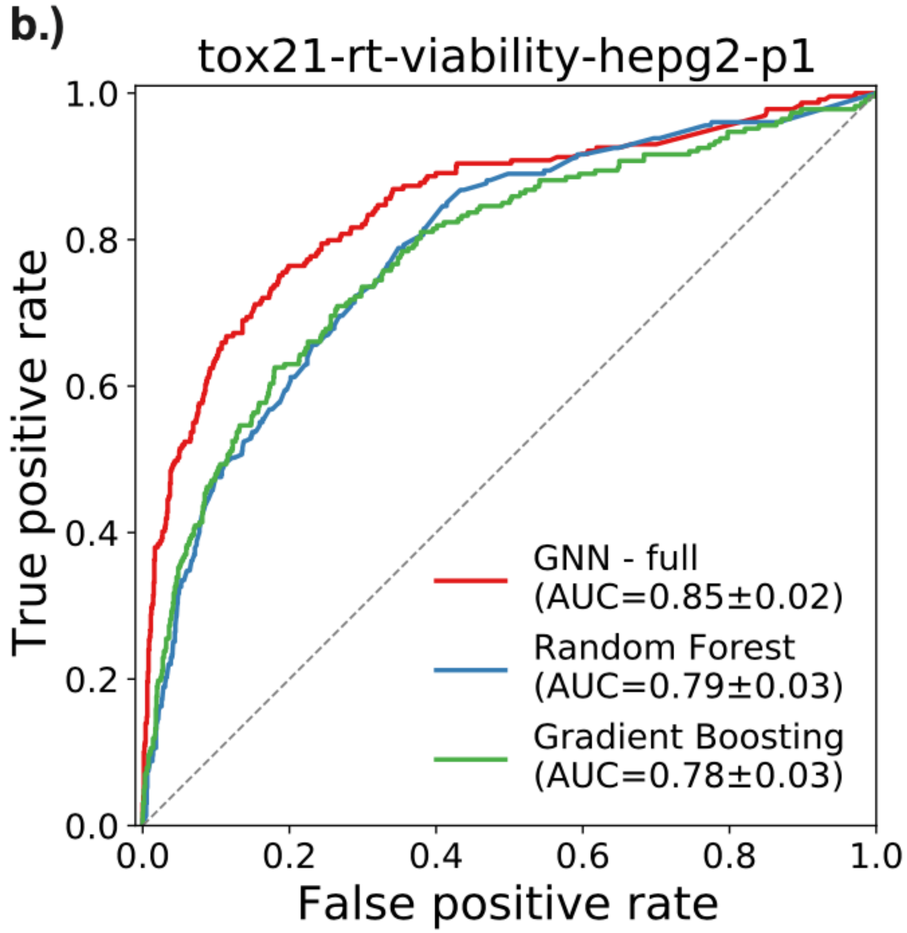


GNN Pipeline

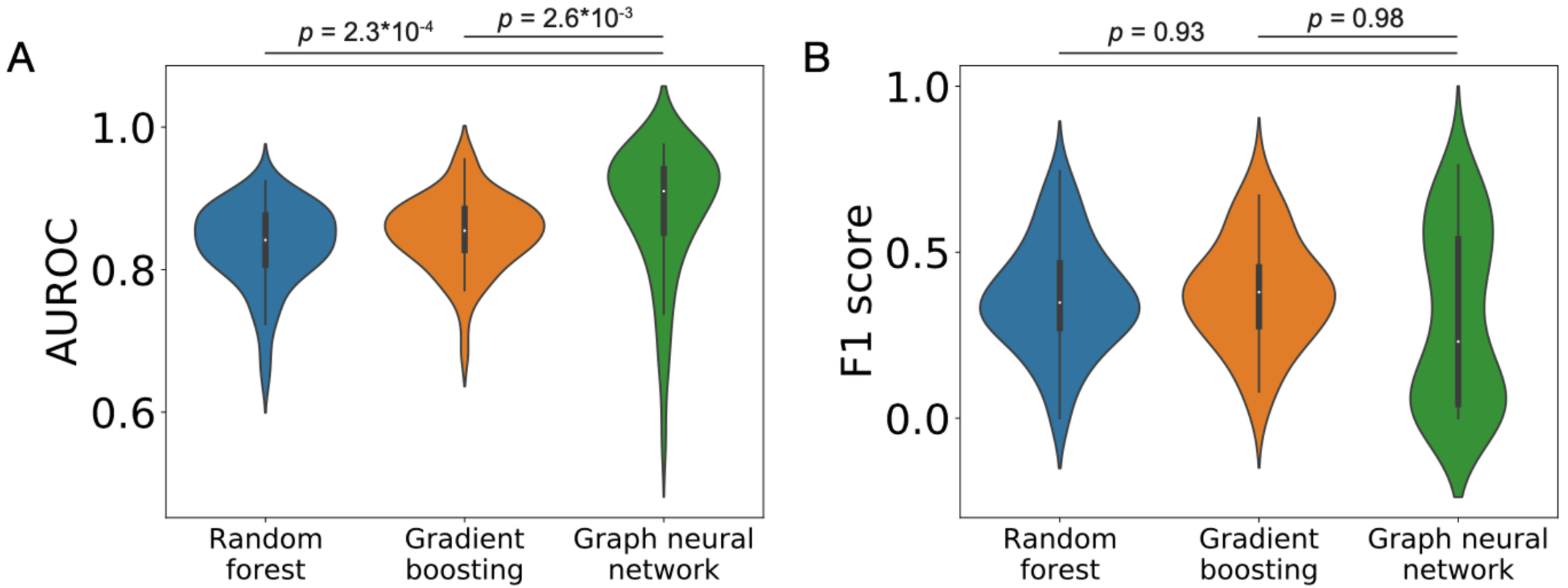


QSAR Performance

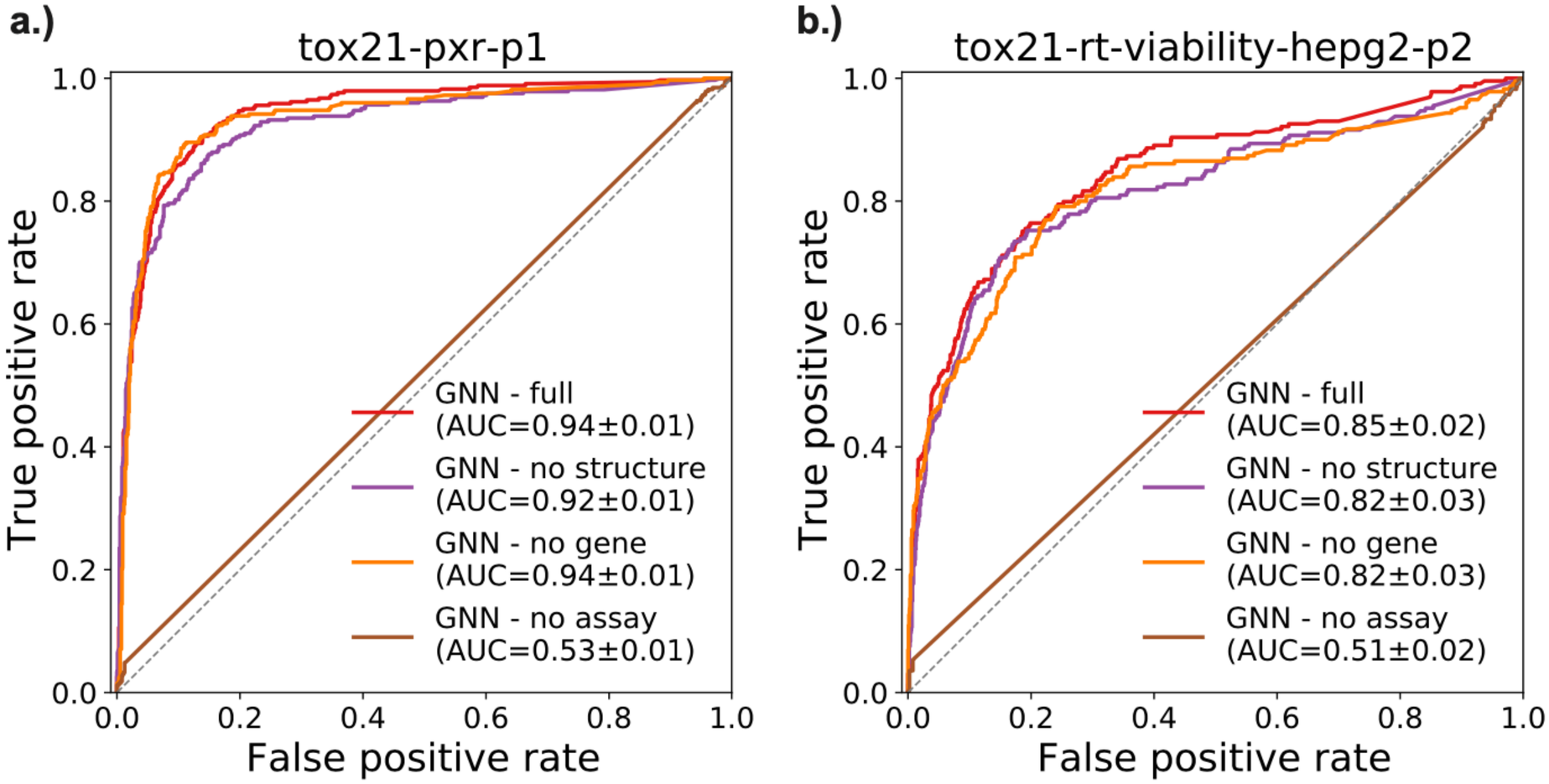




QSAR Performance



Why do the GNNs perform so much better?



Potential for model explainability

- Each relationship in the graph conveys semantic meaning based on node types and relationship types
- For any given assay's GNN, edge weights are proportional to their influence on the final prediction
- Example: *HepG2 cell viability* assay activity prediction
  - Top weighted "other" assays:
    - HepG2 Caspase-3/7 mediated cytotoxicity
    - NIH/3T3 Sonic hedgehog antagonism
  - The first makes obvious sense; is there a mechanistic explanation for the other?

Future work

- Expand on the concept types included in the subgraph (i.e., add diseases, pathways, cell types, etc.)
- Test continuous endpoints (IC_{50}, etc.)
- Evaluate more complex network architectures:
  - Link prediction models
  - Use regularization to better utilize information from non-Assay nodes (important for Graph ML in heterogeneous networks)
  - Deeper networks? May be useful as the network grows
- Develop easy-to-use graphical tools to lower the barrier for diverse user types
  - Use ontology reasoning to further improve explainability

Acknowledgements

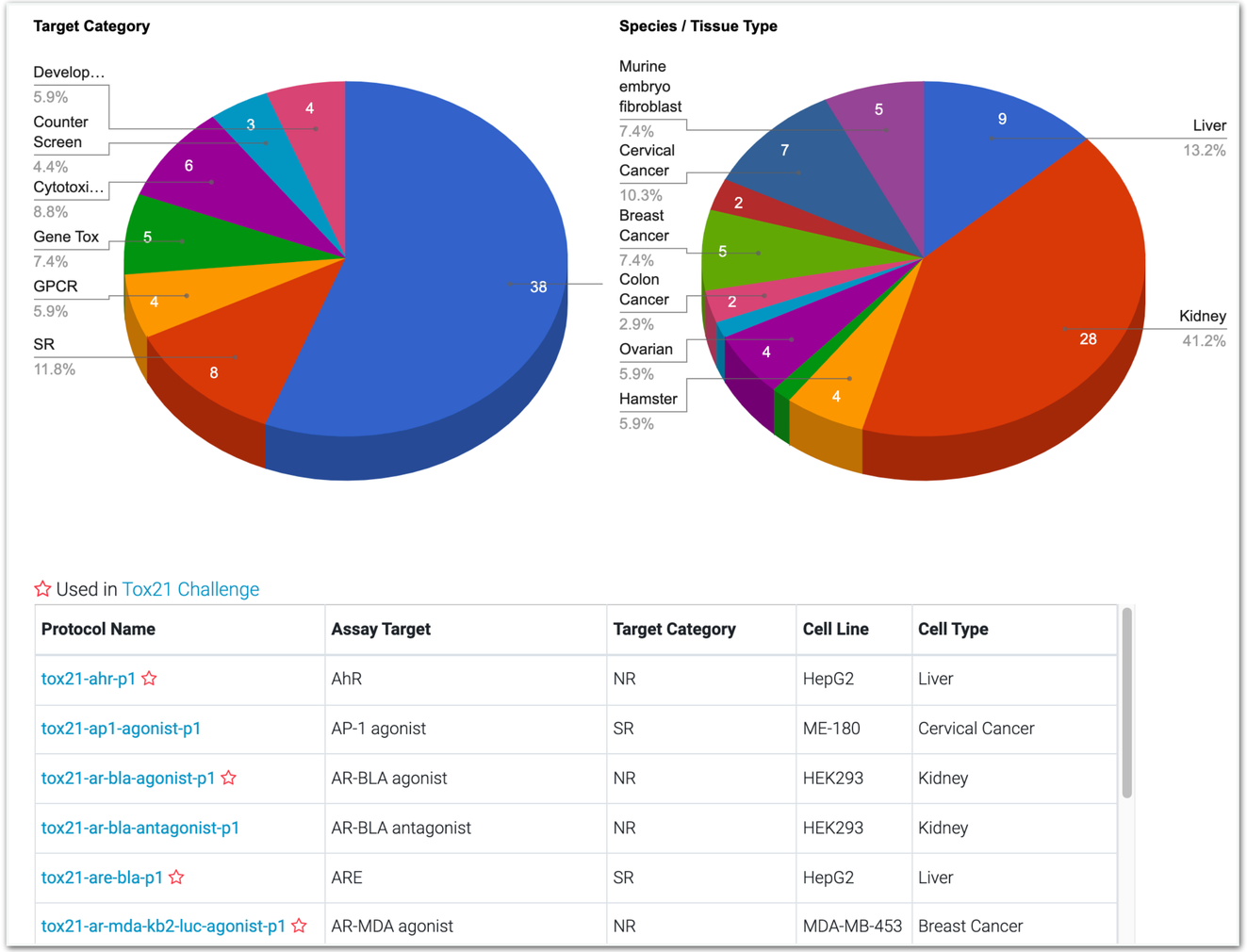
- Yun Hao (Penn)
- Jason Moore (Cedars-Sinai)
- Trevor Penning (Penn)
- Li Shen (Penn)

- Grant funding:
  - K99-LM013646 (Romano)
  - R01-LM010098, R01-LM012601, R01-AI116794, UL1-TR001878, UC4-DK112217 (Moore)
  - T32-ES019851, P30-ES013508 (Penning)

Additional slides:

Tox21 screening dataset

- Tox21: "Toxicology in the 21st Century" dataset for high-throughput chemical screening
  - ~60 specific toxicology-focused biochemical assays
  - ~8,000 chemicals evaluated on those assays



☆ Used in Tox21 Challenge

Protocol Name	Assay Target	Target Category	Cell Line	Cell Type
tox21-ahr-p1 ☆	AhR	NR	HepG2	Liver
tox21-ap1-agonist-p1	AP-1 agonist	SR	ME-180	Cervical Cancer
tox21-ar-bla-agonist-p1 ☆	AR-BLA agonist	NR	HEK293	Kidney
tox21-ar-bla-antagonist-p1	AR-BLA antagonist	NR	HEK293	Kidney
tox21-are-bla-p1 ☆	ARE	SR	HepG2	Liver
tox21-ar-mda-kb2-luc-agonist-p1 ☆	AR-MDA agonist	NR	MDA-MB-453	Breast Cancer

Node classification labeling algorithm

- To build a training dataset for a single assay:
  - Look at the edge linking each chemical to the assay of interest
    - If edge is “chemicalHasActiveAssay”, label the chemical “1”
    - If edge is “chemicalHasInactiveAssay”, label the chemical “0”
    - If there is no edge, don’t label the chemical
  - Remove the node (and incident edges) for the assay of interest to prevent information leakage

GCN Architecture details

Each layer of the network is defined as an edge-wise aggregation of adjacent nodes:

$$h_i^{(l)} = \sigma\left(\sum_{r\in\mathcal{R}} \rho_{j\in\mathcal{N}_i^r}\left(W_r^{(l-1)} h_j^{(l-1)} + W_0^{(l-1)} h_i^{(l-1)}\right)\right). \tag{A.1}$$

where h_i^l is the hidden representation of node i in layer l, $\mathcal{N}(i)$ is the set of immediate neighbors of node i, and σ is a nonlinear activation function (either softmax or leaky ReLU, as explained in **Appendix B**). ρ can be any differential 'reducer' function that combines messages passed from incident edges of a single type; in the case of this study we use summation. Since our graph contains relatively few edge types, regularization of the weight matrices W is not needed.

(See paper for more details)

Node Classification details

For classifying chemicals as active or inactive with regards to an assay of interest, we stack 2 GCN layers in the form given by (A.1), with a leaky ReLU activation between the two layers and softmax applied to the second layer's output. Since we only classify chemical nodes, we ignore outputs for all other node types (and for chemicals with undefined labels); labels are generated via **Algorithm 1** We train the network by minimizing binary cross-entropy between the network's softmax outputs and true activity values:

$$\mathcal{L} = -\sum_{i \in \mathcal{Y}} \ell(h_i^{(0)}) \cdot \ln h_i^{(2)} + (1 - \ell(h_i^{(0)})) \cdot \ln(1 - h_i^{(2)}). \tag{B.1}$$

where $\mathcal{Y}$ is the set of all labeled nodes, $\ell(h_i^{(0)})$ is the true label of node i, and $h_i^{(2)}$ is the final layer output of node i.

The relatively shallow architecture of the network allows us to optimize the model using the Adam algorithm applied to the entire training data set, but the model can be adapted to mini-batch training when appropriate or necessary.

(See paper for more details)