IDC410
A course on Image Processing and Machine Learning
(Lecture 21)

Shashikant Dugad,

IISER Mohali

Graphical Neural Network

GNN Lectures adapted from following References

1. Youtube Lectures given by Petar Velickovic on YouTube:

https://www.youtube.com/watch?v=uF53xsT7mjc&t=1350s

https://www.youtube.com/watch?v=8owQBFAHw7E&list=PPSV

https://www.youtube.com/watch?v=uF53xsT7mjc&list=PPSV&t=728s

2. Other Youtube Videos

https://www.youtube.com/watch?v=fOctJB4kVIM&list=PPSV

https://www.youtube.com/watch?v=ABCGCf8cJOE&list=PPSV

https://www.youtube.com/watch?v=0YLZXjMHA-8&list=PPSV

https://www.youtube.com/watch?v=2KRAOZIULzw&list=PPSV

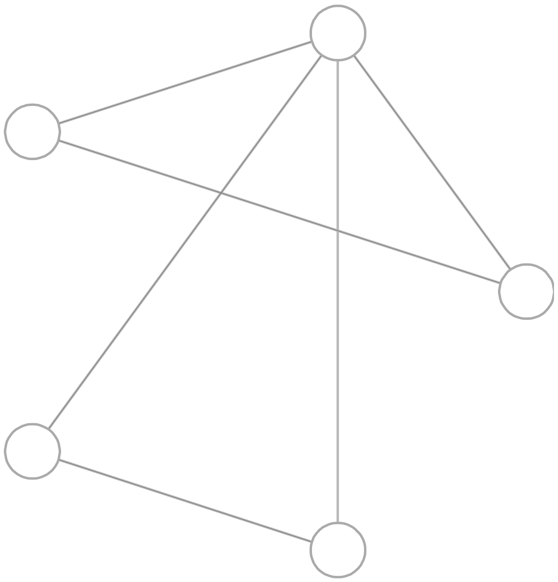
https://www.youtube.com/watch?v=wJQQFUcHO5U&list=PPSV

3. Notes:

https://distill.pub/2021/gnn-intro/

https://distill.pub/2021/understanding-gnns/

Schematic of Graph



- V: Represents Node Vertex (or node) level attributes
- E: Represents Edge (or link) level attributes with directions
- G: Represents Global (or master node) level attributes

For a Social Group Interaction Graph:

- Attributes (features) of *V* can be Name, Age, Gender, Hobbies etc 
- Attributes (features) of *E* can be friend, colleague, boss etc. 
- Attributes (features) of *G* can be behavior of the group, such as polite, hostile etc. 

Feature Matrix

- The feature matrix contains all the feature data *(columns)* of each node *(rows)*. Table below contains 4 features and 5 nodes

X	Y	Layer	Eff_ADC
73.0367	20.1063	1	152.1
54.8556	57.2024	2	174.07
73.0367	19.3055	3	179.14
82.7457	11.2976	4	518.83
74.4237	19.3055	5	136.89

Example: Image as a Graph

- **Image is special case of Graph with N^2 nodes for an image with size NxN**
- **Each pixel is a node and edges are strictly formed with neighboring pixel.**
- **Each pixels has 8 edges (except those on the 1st or last row/column of the image**
- **Side-pixels and corner-pixels in image has 5 and 3 edges respectively**

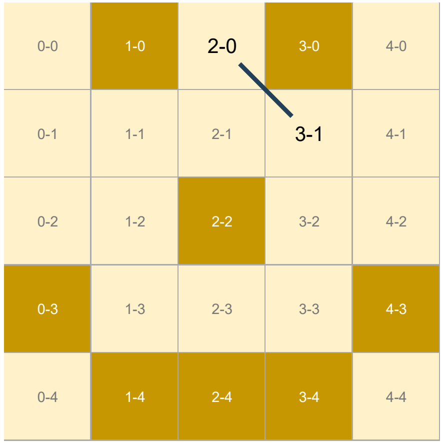
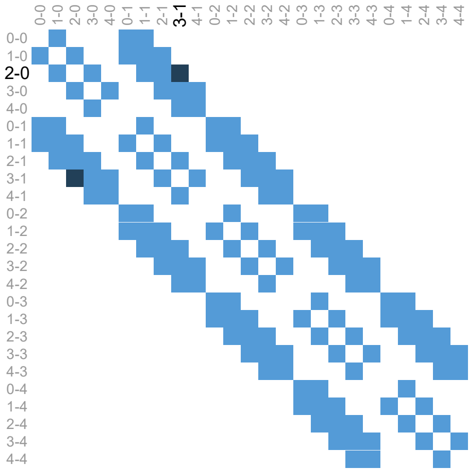
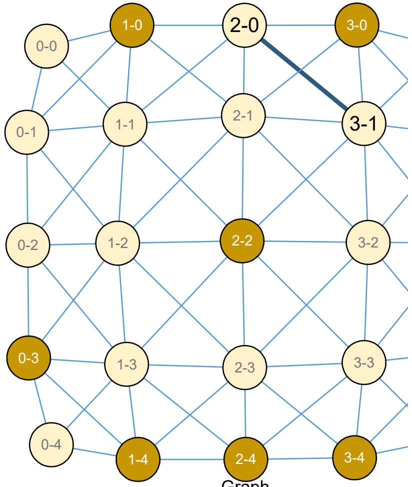


Image Pixels



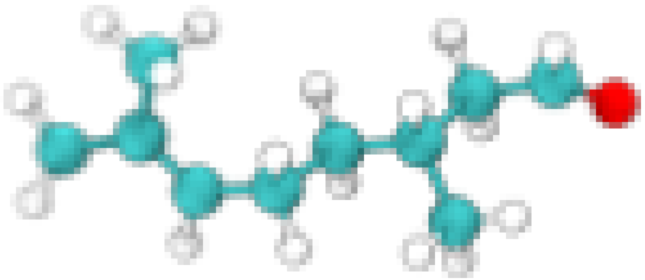
Adjacency Matrix

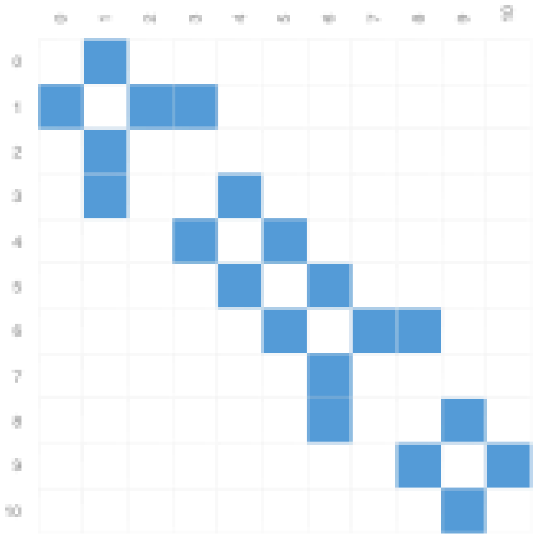


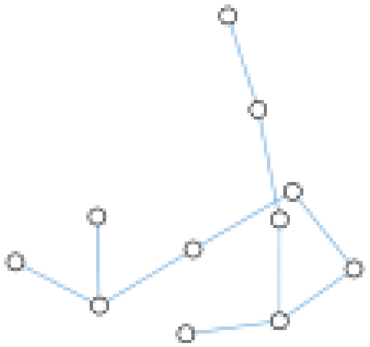
Graph

Example: Molecule as a Graph

- **Molecules are construct of of atoms and electrons in 3D space. They interact and form a stable bond. Different pairs of atoms may have different bonds bonds at different distances. Such construct can be visualised through a formation of graph, where nodes are atoms and edges are bonds. Graph representation for molecule with 11 atoms is shown below:**

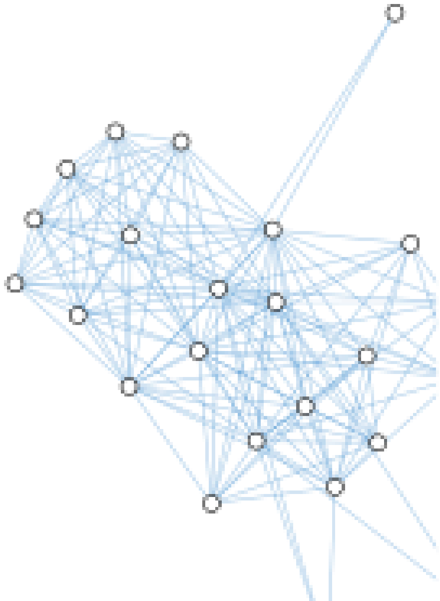






Example: Social Network as a Graph

- **Social networks are tools to study patterns in collective behaviour of people, institutions and organizations. We can build a graph representing groups of people by modelling individual as nodes and their relationships as edges**



Expected Predictions on Graphs using GNN

- There are three general types of expected prediction tasks on graphs:
    - a) Graph-level, b) Node-level, and c) Edge-level
- In a ***graph-level task***, we predict a single property for a whole graph. For a ***node-level task***, we predict some properties for each node in a graph. For an ***edge-level task***, we want to predict the property or presence of edges in a graph.
- Any of the three levels prediction tasks can be solved with a single model class approach, the ***Graphical Convolutional Network (GCN)***.

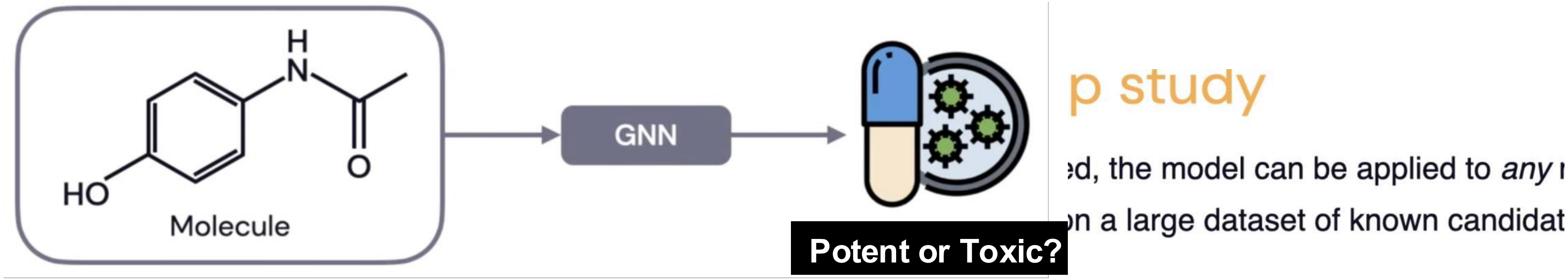
Example: Graph-Level Predictions

- **Goal is to predict property of entire graph, let us say response of a molecule on human body**

- **Task: Identify whether the given molecule is a potential *potent drug (antibiotic) or a toxic material* (binary classification of graph-level prediction)**
  - **Atoms are visualized as nodes and bonds as edges**
  - **Features of nodes (atoms) can be atomic number, atomic weight, valency, electron affinity etc. and Features of edges could be bond type, bond energy etc.**
  - **Have graphs for large number of *known* potent and toxic molecules**
  - **Do training using the established dataset**
  - **Translate new molecules to their respective graphs and then apply GNN on each graph to classify their behavior *(potent or toxic)***

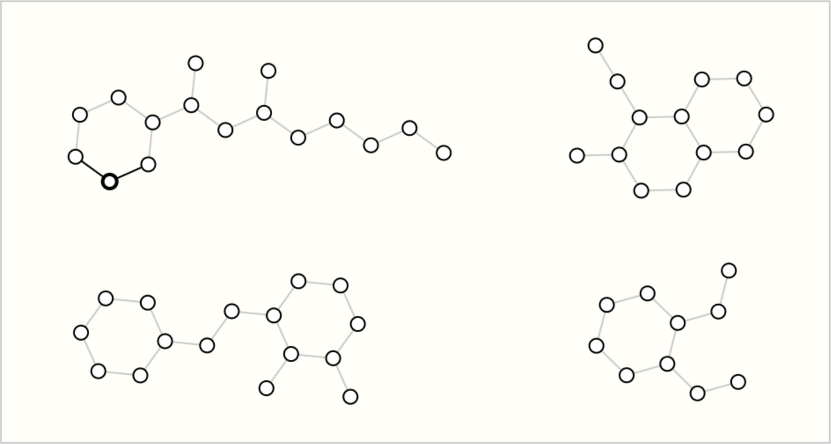
Example: Graph-Level Predictions

- **Select say top 100 new molecules identified as potent drug**
- **Do some additional filtering using laboratory tests**
- **Carryout detailed laboratory tests followed by trials to establish its efficacy**
- **New antibiotic drug called *Halicin* was discovered in this manner !!!**

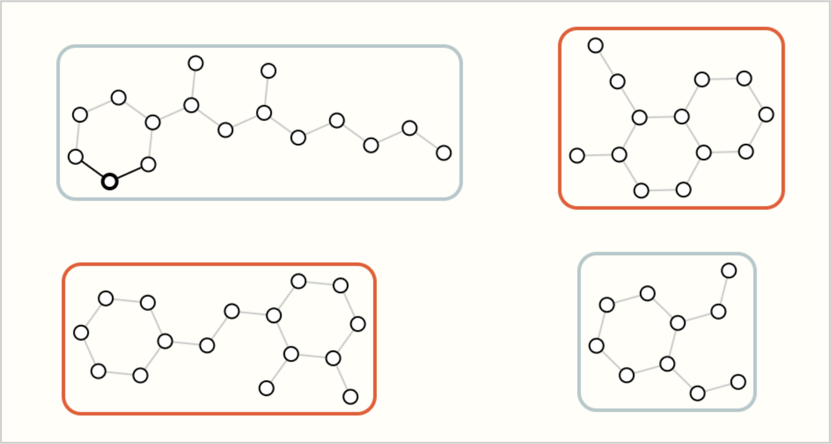


Example: Graph-Level Predictions

Task: To find number of Rings in the Graph



Input: graphs



Output: labels for each graph, (e.g., "does the graph contain two rings?")

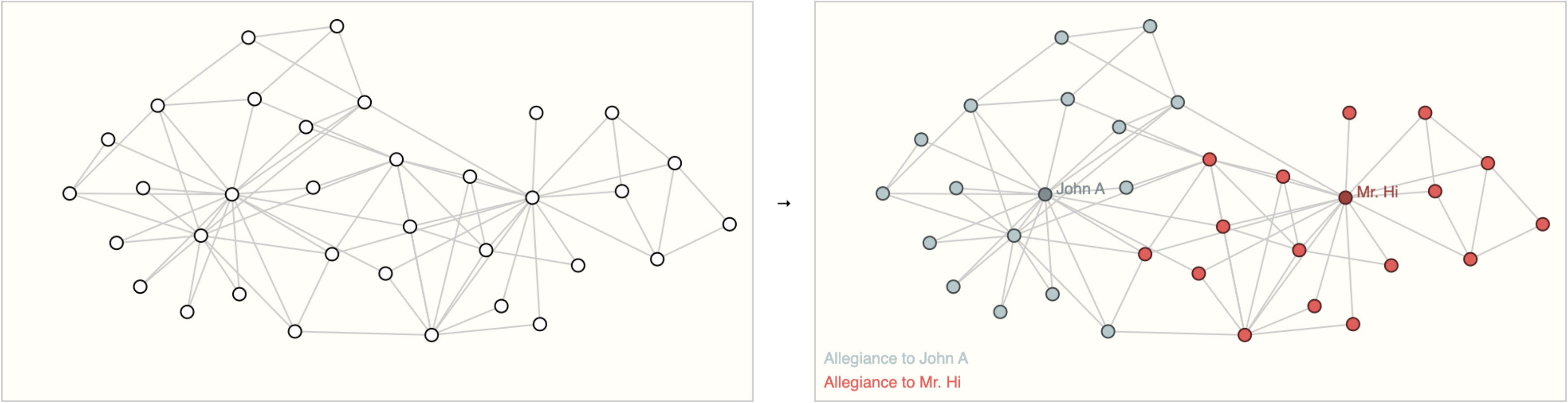
Example: Node-Level Predictions

- **Node-level tasks are concerned with predicting the identity or role of a each node within the graph**

- **A classic example of a node-level prediction is a problem of karate club. The dataset is a single social network graph made up of individuals As the story goes, a feud between Mr. Hi (Instructor) and John H (Administrator) creates an unrest in the karate club. The nodes represent individuals and the edges represent interactions between these members outside of karate. The prediction problem is to classify whether a given member becomes loyal to either Mr. Hi or John H, after the feud. In this case, distance between a node to either the Instructor or Administrator is highly correlated to this label**

Example: Node-Level Predictions

- Node-level prediction problems are analogous to image segmentation, where we are trying to label the role of each pixel in an image. For example, each pixel is assigned to some class in YOLO



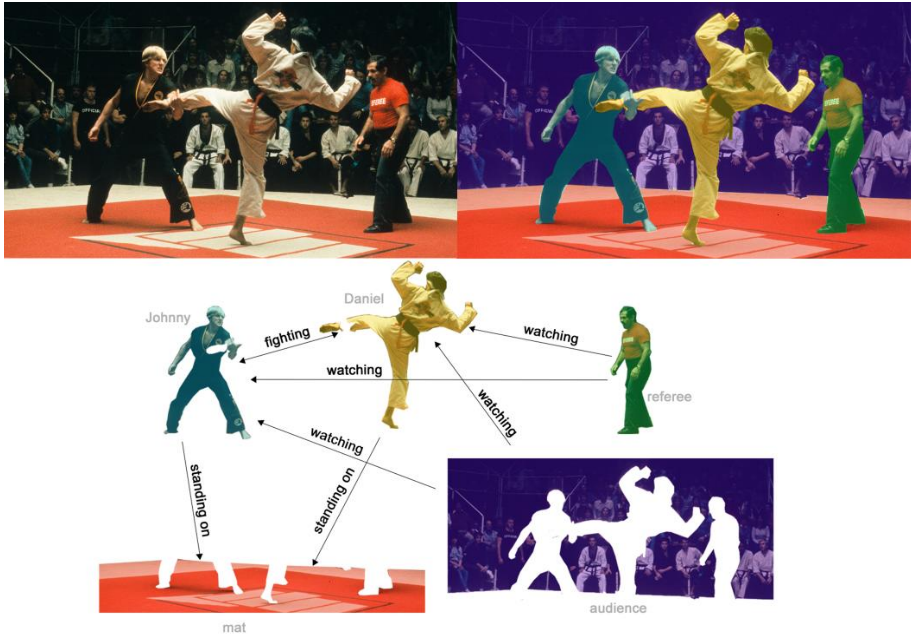
Input: graph with unlabled nodes

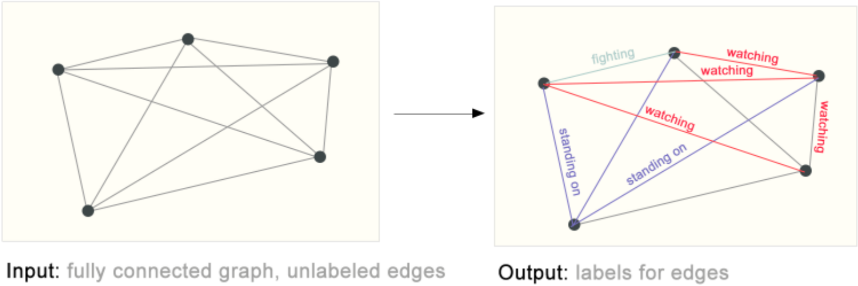
Output: graph node labels

Example: Edge-Level Predictions

- **Beyond identifying objects in an image, deep learning models can also be used to predict (or establish) the relationship between the objects i.e. nodes**
- **Edge-level inference is about prediction of relationship between objects (nodes) seen in an image**
- **Given the set of nodes (representing objects in the image), we wish to predict which of these nodes share an edge or what the value of that edge is.**
- **If we wish to discover connections between entities, we could consider the graph fully connected and based on their predicted value of each edge, we prune edges to arrive at a sparse graph**

Example: Edge-Level Predictions





The initial graph built from the previous visual scene. On the right is a possible edge-labeling of this graph when some connections were pruned based on the model's output.

The image is segmented into five entities: each of the fighters, the referee, the audience and the mat.