CS F425: Deep Learning

34 Graph Neural Networks (GNN)

Dr. Kamlesh Tiwari
Assistant Professor, Department of CSIS,
BITS Pilani, Pilani Campus, Rajasthan-333031 INDIA

April 27, 2023 ON-CAMPUS Campus @ BITS-Pilani [Jan-May 2023]

http://ktiwari.in/dl

GNN Introduction: learning in non-Euclidean space [1]

Neural Networks that operate over graph structure data $G = (E, V)$

- Graphs can be irregular, may have a variable size of unordered nodes, also nodes may have a different number of neighbors
- Instances are no longer independent of each other

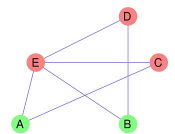
$G, V, E, v, e_{i,j}, N(v), A, A^T, A^n$
$[A, B], D, n = |V|, m = |E|, d, b, c$
$X, x_v \in R^d, H, h_v, W, \Theta, w, \theta$

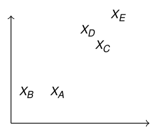
$$h_v^{(t)} = \sum_{u \in N(v)} f(x_v, x^e_{(v,u)}, x_u, h_u^{(t-1)})$$

[1] Wu, Zonghan, et al. **A comprehensive survey on graph neural networks.** IEEE transactions on neural networks and learning systems 32.1 (2020): 4-24.
A. Sperduti and A. Starita, **Supervised neural networks for the classification of structures,** IEEE Transactions on Neural Networks, vol. 8, no. 3, pp. 714–735, 1997.

Node representation with GNN

- A: Adjacency representation ($N \times N$)
- Embedding of similar nodes should be near ($H_0 = X$ is $N \times F_0$)



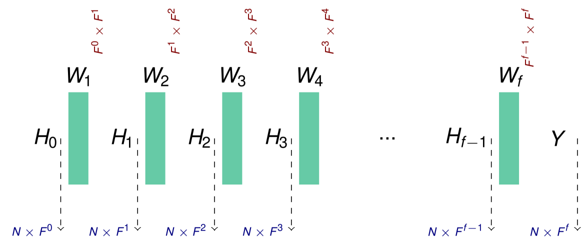


$$X_A = [f_A^1, f_A^2, ..., f_A^{F_0}]$$

GNN takes A and X as input and transforms $H_0 \to H_1 \to H_2 \to ...$

$$H_i = f_{W_i}(H_{i-1}, A)$$

Features with GNN



$$H_i = f_{W_i}(H_{i-1}, A)$$

Features with GNN

$$f_{W_i}(H_{i-1}, A) = \sigma(A \times H_{i-1} \times W_i)$$

1. For example consider a simple: $W = 1$ and ReLU activation
   $$f = A \times X$$
   - Neighborhood is aggregated
   - However misses own feature
2. Add self loops in A by using $\hat{A} = A + I$
   - Influential nodes [2] dominate the feature
3. Multiply with reciprocal of nodes degree D^{-1}
   $$f = D^{-1} \times \hat{A} \times X$$

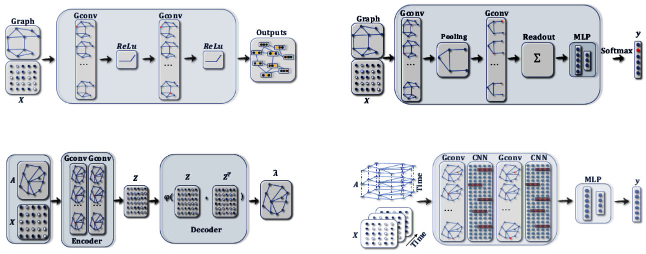
[2] Nodes having higher degree

Features with GNN

$$f_W(X, A) = \sigma((D^{-1} \times \hat{A} \times X) \times W)$$

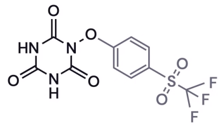
1. Depth of GNN corresponds to hops of feature aggregation
2. **Spectral Rule:**
   $$X_i = \sum_{j=1}^{N} \frac{A_{ij}}{\sqrt{D_{ii} D_{jj}}} X_j$$

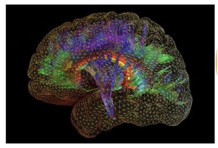
Different GNN Architectures



- Recurrent graph neural networks
- Convolutional graph neural networks
- Graph autoencoders
- Spatial-temporal graph neural networks

GNN Applications





- Social Network Analysis
- Drug discovery

Thank You!

Thank you very much for your attention!