Preliminaries

Graph Convolutional Networks

- M defined in 1stChebNet (Kipf and Welling 2017) as follow:
- $\mathbf{H}_k = M\left(\mathbf{A}, \mathbf{H}_{k-1}; \boldsymbol{W}_{k-1}\right) = \sigma\left(\hat{\mathbf{A}}\mathbf{H}_{k-1}\boldsymbol{W}_{k-1}\right)$
  - $\hat{\mathbf{A}} = \tilde{\mathbf{D}}^{-\frac{1}{2}}\tilde{\mathbf{A}}\tilde{\mathbf{D}}^{-\frac{1}{2}}$: normalized adjacency matrix
  - $\tilde{\mathbf{A}} = \mathbf{A} + \mathbf{I}_N$: adding self-connection
  - $\tilde{\mathbf{D}}_{ii} = \Sigma_j \tilde{\mathbf{A}}_{ij}$: degree of the i-th node

Preliminaries

DropEdge

- Novel method to reduce over-fitting for GCN-based models (Rong et al. 2019).
- Randomly drops out edges from input graphs to generate different deformed copies with certain rate at each training epoch.
  - This method augments the randomness and the diversity of input data.
- Formally, suppose the total number of edges in the graph $\mathbf{A}$ is N_e, and the dropping rate is p
  - $\mathbf{A}' = \mathbf{A} - \mathbf{A}_{drop}$: adjacency matrix after DropEdge
  - $\mathbf{A}_{drop}$ is constructed using $N_e \times p$ edges randomly sampled from the original edge set.