IN5550: Neural Methods in Natural Language Processing
Lecture 12, part 3
Graph Neural Networks

Roxana Pop

University of Oslo

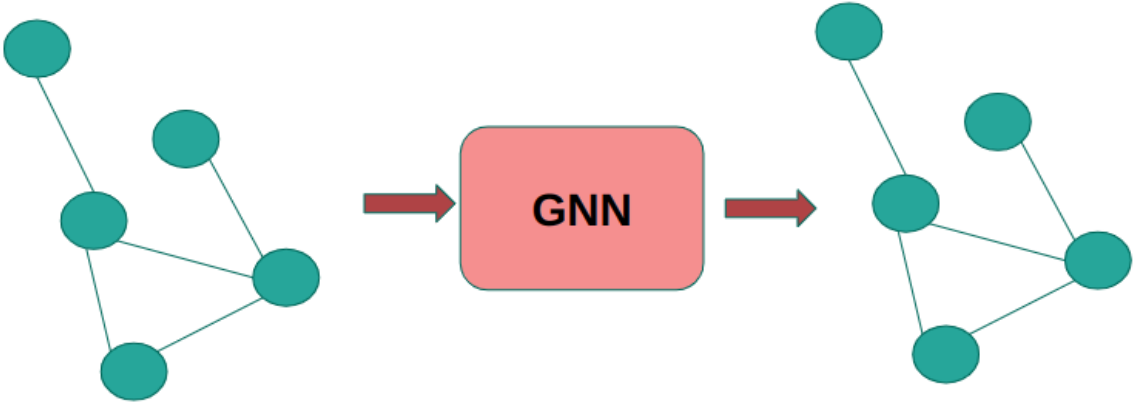


Contents

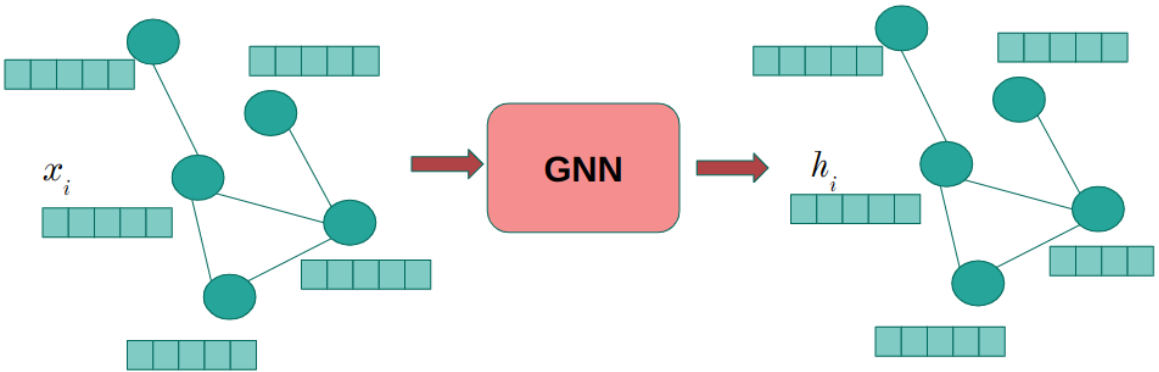


1. GNN abstraction
2. GNN flavors
3. GATs and Transformers

GNNs - Graph Neural Networks

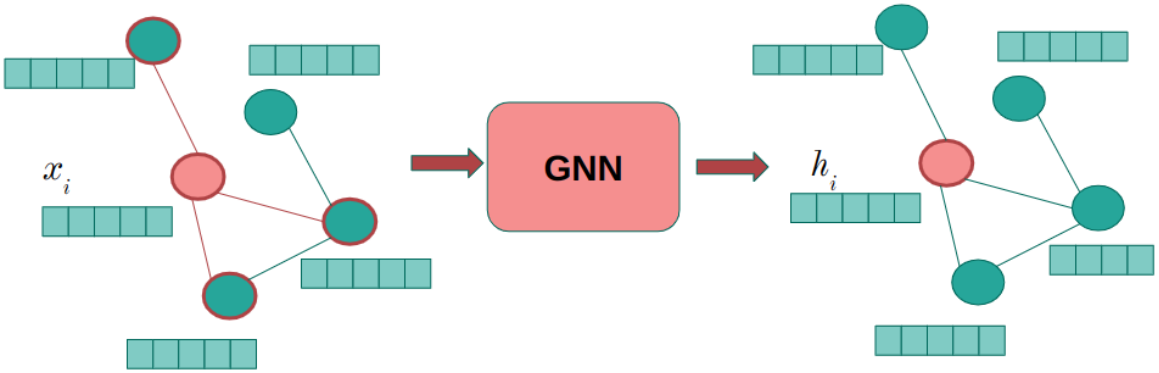


GNNs - Graph Neural Networks



$$h_i = g(\mathbf{x}_i, \mathcal{N}_i)$$

GNNs - Graph Neural Networks



$$h_i = g(\mathbf{x}_i, \mathcal{N}_i)$$

Contents

1 GNN abstraction

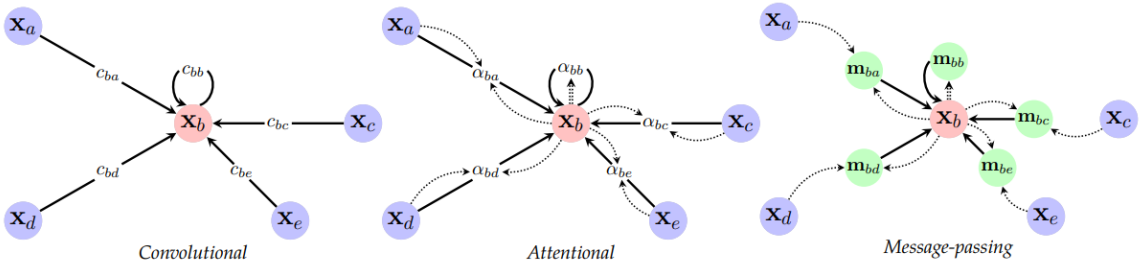
2 GNN flavors

3 GATs and Transformers

GNN flavors

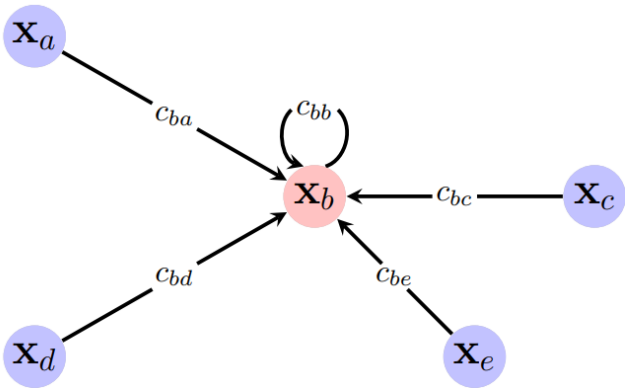
- $h_i = g(\mathbf{x}_i, \mathcal{N}_i)$
- But how is g defined?
- Depends on the flavor of the GNN

GNN flavors



From [Bronstein et al., 2021]

Convolutional GNNs



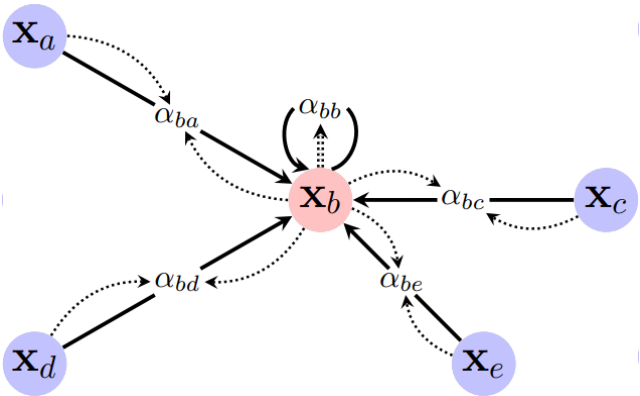
$$\mathbf{h}_u = \phi(\mathbf{x}_u, \bigoplus_{v \in \mathcal{N}_u} c_{uv} \psi(\mathbf{x}_v))$$

Note: $\bigoplus$ – summation

The latent representation $\mathbf{h}_u$ of node u is computed using:

- Input features $\mathbf{x}_u$ of u
- Aggregated neighbor features
-
  - Weighted sum
  - c_{uv}: constant; showing importance of neighbor v for u's representation

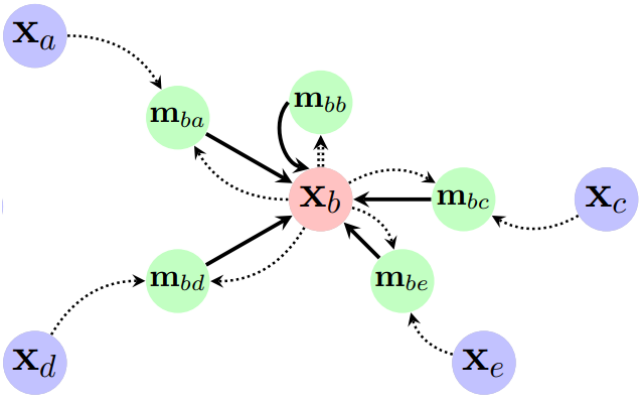
Attentional GNNs



$$\mathbf{h}_u = \phi(\mathbf{x}_u, \bigoplus_{v \in \mathcal{N}_u} a(\mathbf{x}_u, \mathbf{x}_v)\psi(\mathbf{x}_v))$$

The importance of a neighboring node v is computed using the attention $a(\mathbf{x}_u, \mathbf{x}_v)$ between the node of interest u and the neighbor v

Message-passing GNNs



$$\mathbf{h}_u = \phi(\mathbf{x}_u, \bigoplus_{v \in \mathcal{N}_u} \psi(\mathbf{x}_u, \mathbf{x}_v))$$

Average "messages" between the node u and its neighbors instead of averaging just the representations of the neighbors.

More general

convolutional $\subseteq$ attentional $\subseteq$ message-passing

Many architectures

Take a look at `https://distill.pub/2021/understanding-gnns/`

GCN | GAT | GraphSAGE | GIN

$$h_v^{(0)} = x_v \quad \text{for all } v \in V.$$

Node v's initial embedding.

... is just node v's original features.

and for $k = 1, 2, \ldots$ upto K:

$$h_v^{(k)} = f^{(k)}\left(W^{(k)} \cdot \frac{\sum_{u \in \mathcal{N}(v)} h_u^{(k-1)}}{|\mathcal{N}(v)|} + B^{(k)} \cdot h_v^{(k-1)} \right) \quad \text{for all } v \in V.$$

Node v's embedding at step k.

Mean of v's neighbour's embeddings at step $k-1$.

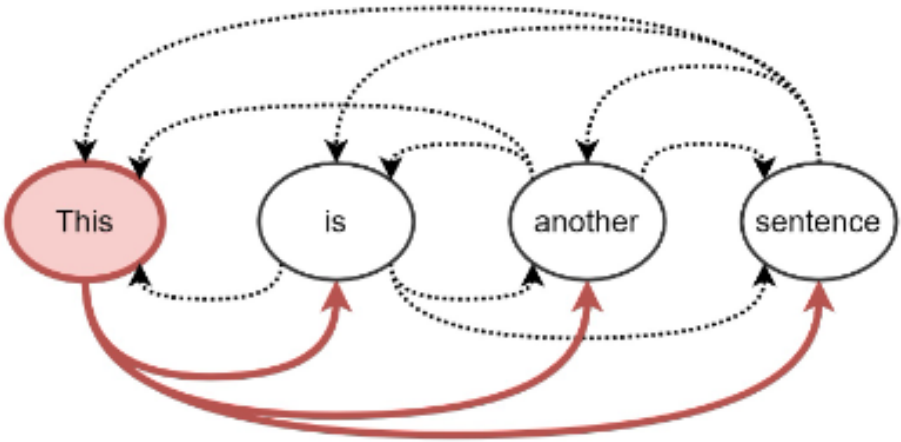
Node v's embedding at step $k-1$.

Contents



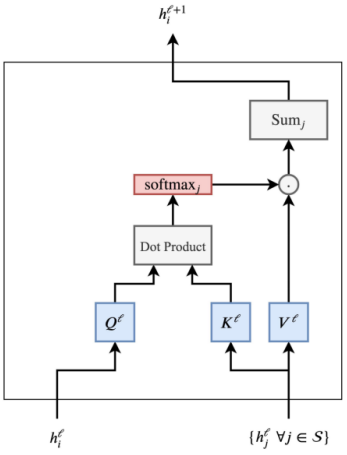
1. GNN abstraction
2. GNN flavors
3. GATs and Transformers

Sentences as graphs



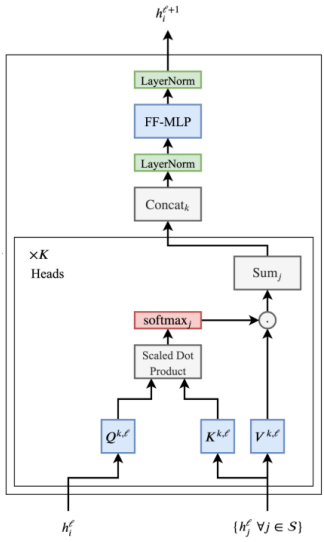
Taken from [Joshi, 2020]

Transformers and GATs [Joshi, 2020]



GAT

[Veličković et al., 2017]



Transformer [Vaswani et al., 2017]

References I

Bronstein, M. M., Bruna, J., Cohen, T., and Velickovic, P. (2021).
Geometric deep learning: Grids, groups, graphs, geodesics, and gauges.
CoRR, abs/2104.13478.

Joshi, C. (2020).
Transformers are graph neural networks.
https://graphdeeplearning.github.io/post/transformers-are-gnns/.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. (2017).
Attention is all you need.
In Advances in Neural Information Processing Systems, pages 5998–6008.

Veličković, P., Cucurull, G., Casanova, A., Romero, A., Liò, P., and Bengio, Y. (2017).
Graph attention networks.
6th International Conference on Learning Representations.