Network Representation Learning

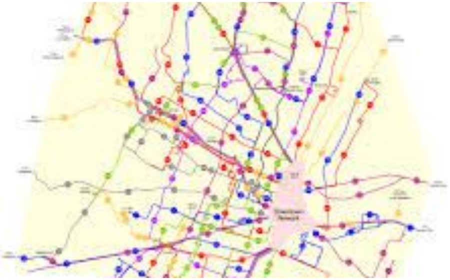
Hsi-Wen Chen

Outline

- Network Embedding
- Graph Neural Network

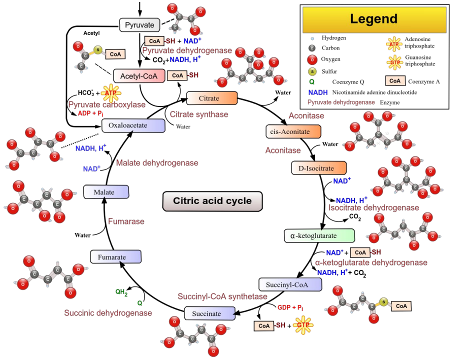
Why Graph?

Social

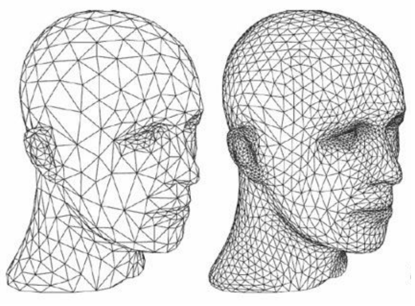


Transportaion

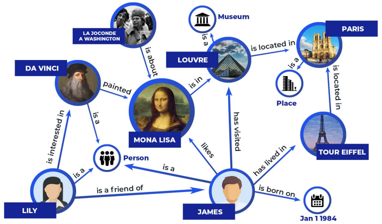
Molucular



Biology



Comouter Graphing



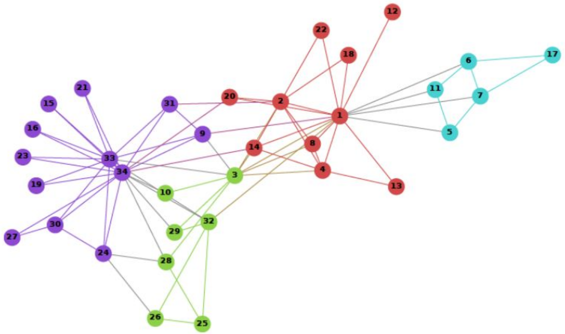
Knowledge

Tasks

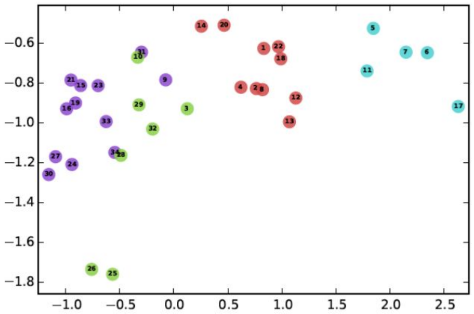
- Node-level
  - node classification
- Edge-level
  - link prediction, edge classification
- Graph-level (global-level)
  - graph classification

Network Embedding

Find embedding of nodes to d dimensions so that “similar” nodes in the graph have embeddings that are close together.

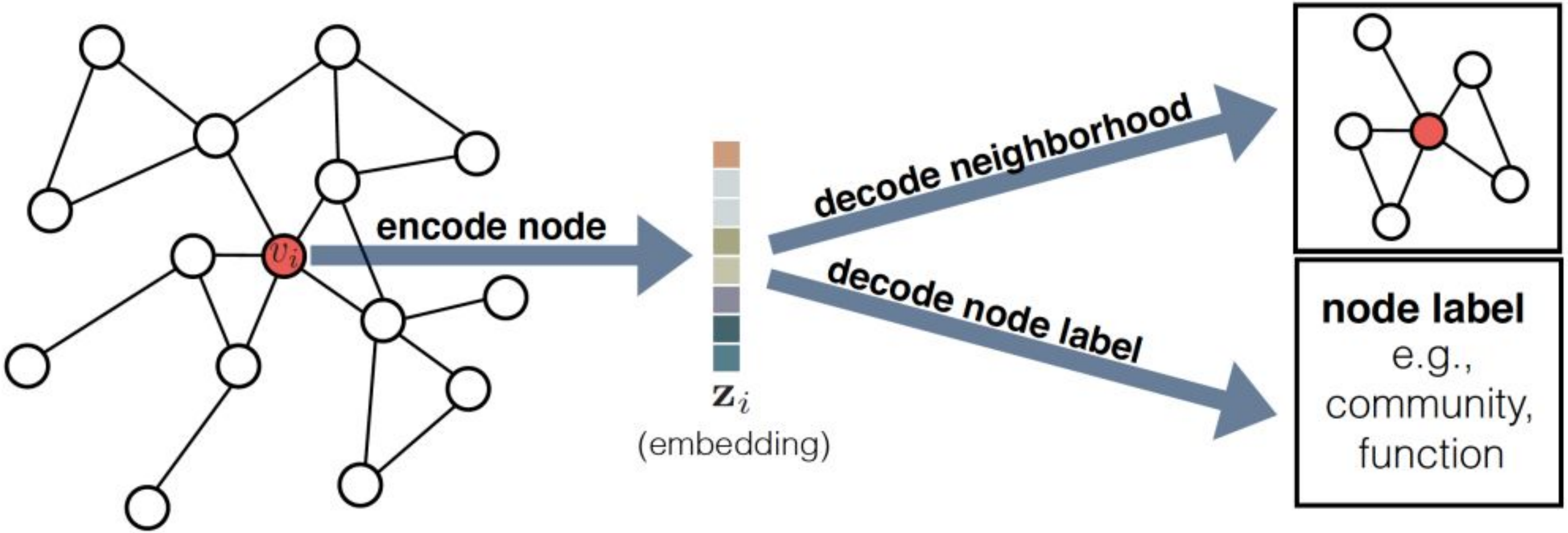


Input



Output

Network Embedding



Network Embedding

Goal is to encode nodes so that similarity in the embedding space approximates similarity in the original network.

Encoder: maps each node to a low dimensional vector.

$$z_i = \phi(v_i)$$

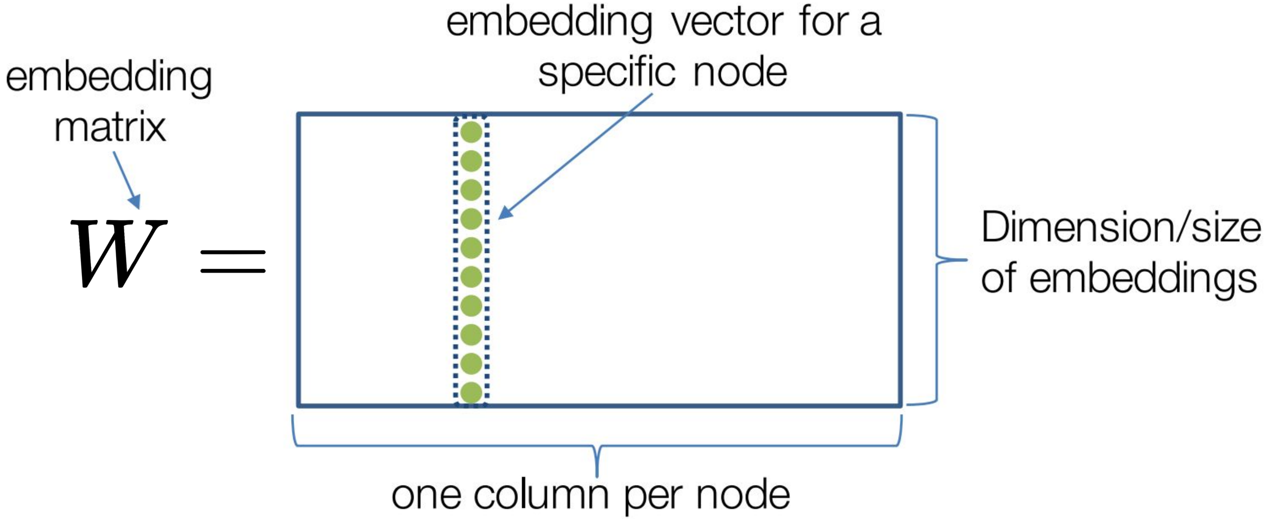
Similarity function: specifies how relationships in vector space map to relationships in the original network.

$$similarity(v_i, v_j) = z_i^T z_j$$

Shallow Embedding

Simplest encoding approach: encoder is just an embedding-lookup

$$z_i = W v_i$$



Matrix Factorization

Dot products between node embeddings approximate edge existence.

$$L = \sum_{(v_i, v_j) \in V^2} ||z_i^T z_j - A_{i,j}||$$

Sol 1. Solve matrix decomposition solvers (e.g., SVD or QR decomposition routines). Very slow. O(V^3)

Sol 2. Use stochastic gradient descent (SGD) as a general optimization method. Highly scalable, general approach. O(V^2)

Matrix Factorization

One-hop

Distributed Natural Large Scale Graph Factorization. In WWW, 2013

Multi-hop

GraRep: Learning Graph Representations with Global Structural Information. In CIKM 2015.

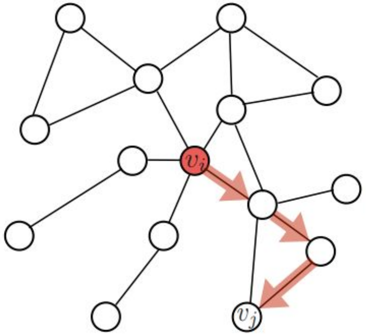
Different Similairty (e.g., Jaccard similarity and Adamic-Adar score)

Asymmetric Transitivity Preserving Graph Embedding. In KDD 2016

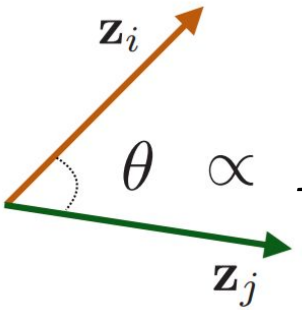
Random Walk

Estimate probability of visiting node v_j on a random walk starting from node v_i using some random walk strategy

$$L = -\sum_{v_i \in V} \sum_{v_j \in N(v_i)} log(P(v_j|v_i))$$



$$\rightarrow P(v_j|v_i)$$



$$\theta \propto P(v_j|v_i)$$

Negative Sampling

Instead of normalizing w.r.t. all nodes, just normalize against randomly unlink pair

$$P(v_j|v_i) = \frac{exp(z_i^T z_j)}{\sum_{v_n \in V} exp(z_i^T z_n)}$$

$$log(\frac{exp(z_i^T z_j)}{\sum_{v_n \in V} exp(z_i^T z_n)}) = log(\sigma(z_i^T z_j)) - \sum_{v_n \sim P(v_i)} log(\sigma(z_i^T z_n))$$

Different Walk Strategy

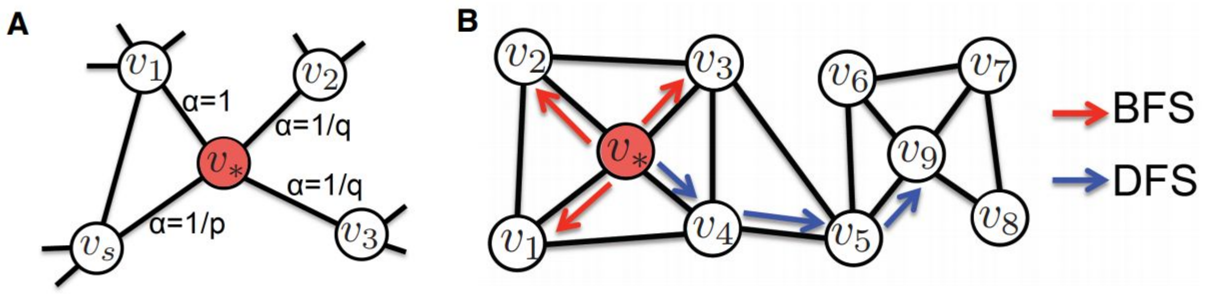
Basic

DeepWalk: Online Learning of Social Representations. In KDD, 2014.

LINE: Large-scale Information Network Embedding. In WWW, 2015.

Bias

node2vec: Scalable Feature Learning for Networks. In KDD,2016.



Random Walk = Matrix Factorization

Network Embedding as Matrix Factorization: Unifying DeepWalk, LINE, PTE, and node2vec. In WSDM. 2018.

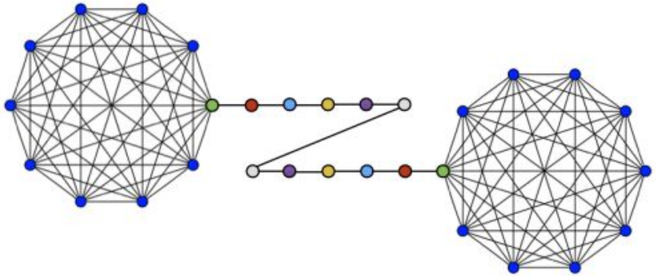
Algorithm	Matrix
DeepWalk	$\log\left(\operatorname{vol}(G)\left(\frac{1}{T}\sum_{r=1}^{T}(\boldsymbol{D}^{-1}\boldsymbol{A})^r\right)\boldsymbol{D}^{-1}\right) - \log b$
LINE	$\log\left(\operatorname{vol}(G)\boldsymbol{D}^{-1}\boldsymbol{A}\boldsymbol{D}^{-1}\right) - \log b$
PTE	$\log\left(\begin{bmatrix} \alpha \operatorname{vol}(G_{\text{ww}})(\boldsymbol{D}_{\text{row}}^{\text{ww}})^{-1}\boldsymbol{A}_{\text{ww}}(\boldsymbol{D}_{\text{col}}^{\text{ww}})^{-1} \\ \beta \operatorname{vol}(G_{\text{dw}})(\boldsymbol{D}_{\text{row}}^{\text{dw}})^{-1}\boldsymbol{A}_{\text{dw}}(\boldsymbol{D}_{\text{col}}^{\text{dw}})^{-1} \\ \gamma \operatorname{vol}(G_{\text{lw}})(\boldsymbol{D}_{\text{row}}^{\text{lw}})^{-1}\boldsymbol{A}_{\text{lw}}(\boldsymbol{D}_{\text{col}}^{\text{lw}})^{-1} \end{bmatrix}\right) - \log b$
node2vec	$\log\left(\frac{\frac{1}{2T}\sum_{r=1}^{T}\left(\sum_u \boldsymbol{X}_{w,u}\underline{\boldsymbol{P}}_{c,w,u}^{r} + \sum_u \boldsymbol{X}_{c,u}\underline{\boldsymbol{P}}_{w,c,u}^{r}\right)}{(\sum_u \boldsymbol{X}_{w,u})(\sum_u \boldsymbol{X}_{c,u})}\right) - \log b$

Rethinking Similarity

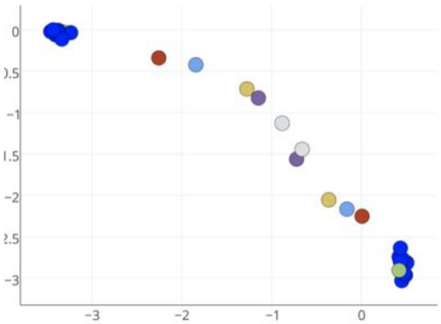
Affinity is not the only way to describe the nodes similarity

struc2vec: Learning Node Representations from Structural Identity. In KDD. 2016.

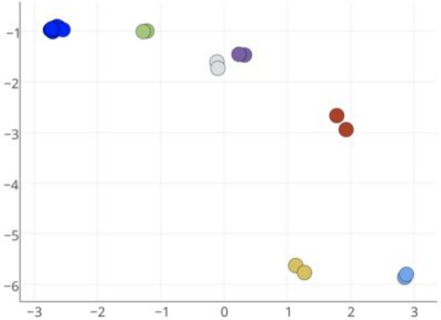
Learning structural node embeddings via diffusion wavelets. In KDD, 2018.



(a) Barbell Graph B(10, 10)



(d) *node2vec*

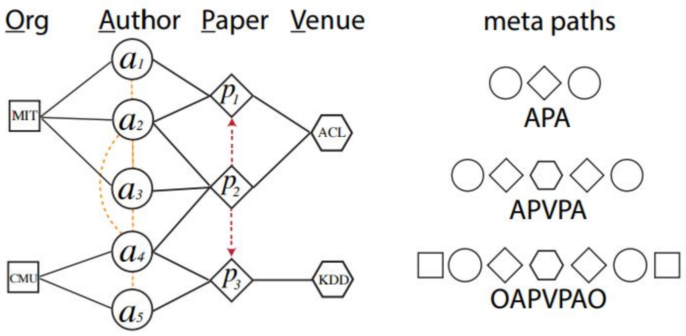


(e) *struc2vec*

Hetergeneous Information Network

Overemphasize of the types

metapath2vec: Scalable Representation Learning for Heterogeneous Networks. In KDD, 2017.

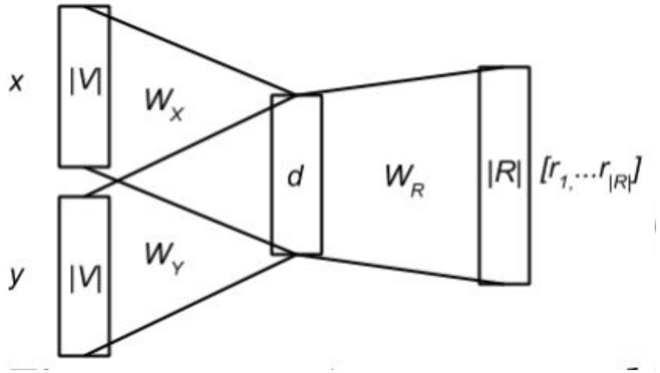


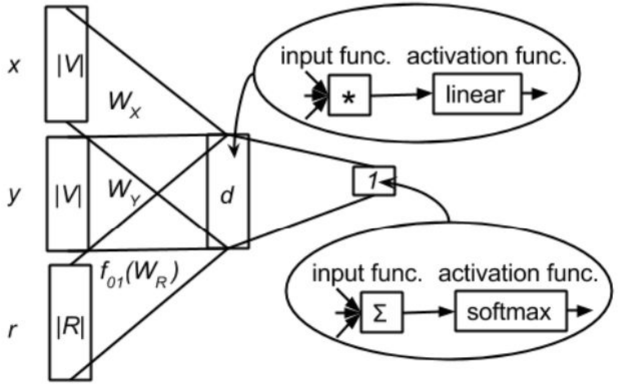
$$\mathcal{P}: V_1 \xrightarrow{R_1} V_2 \xrightarrow{R_2} \cdots V_t \xrightarrow{R_t} V_{t+1} \cdots \xrightarrow{R_{l-1}} V_l$$

Hetergeneous Information Network

Costly to scan the whole network to find all the relationships

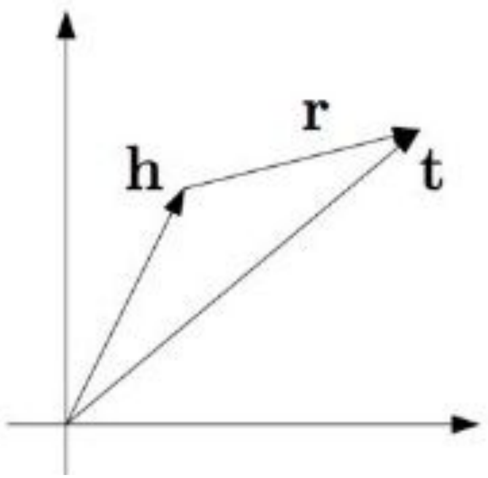
HIN2Vec: Explore Meta-paths in Heterogeneous Information Networks for Representation Learning. In CIKM, 2017.





Knowledge Graph

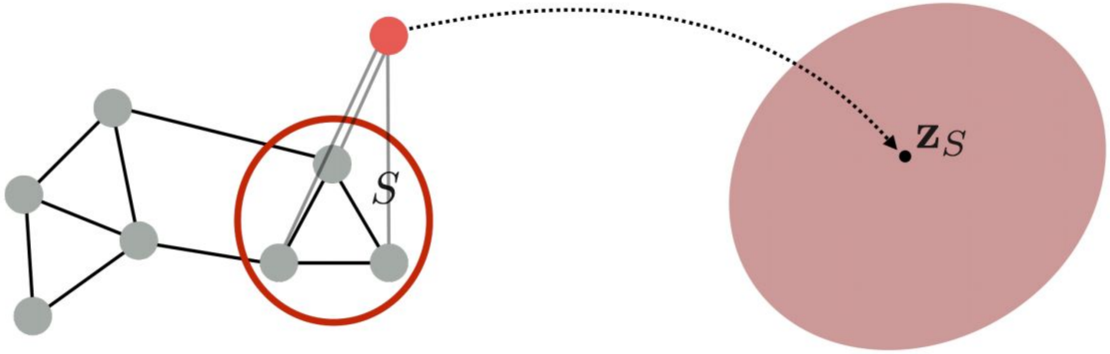
TransE: Translating Embeddings for Modeling Multi-relational Data. In NIPS, 2013.
TransH: Knowledge Graph Embedding by Translating on Hyperplanes. In AAAI, 2014.
TransR & CTransR: Learning Entity and Relation Embeddings for Knowledge Graph Completion. In AAAi, 2015.



$$z_h + z_r = z_t$$

Subgraph:Virtual Node

Gated Graph Sequence Neural Networks. In ICLR, 2015.



Graph Neural Network

Aforementioned methods only consider pairwise correlation.

$$z_i = z_j$$

Generate node embeddings based on local neighborhoods.

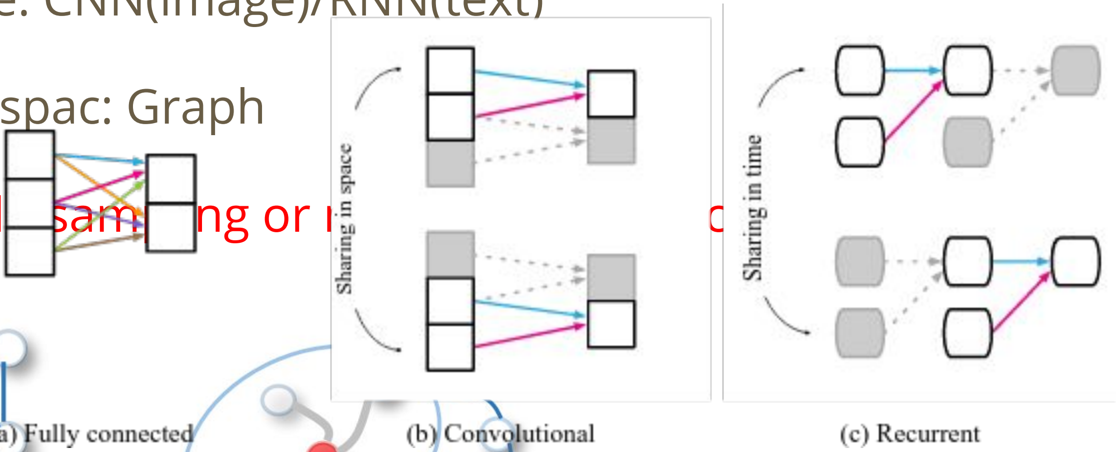
$$z_i = AGG(\sum_{v_j \in N(v_i)} z_j)$$

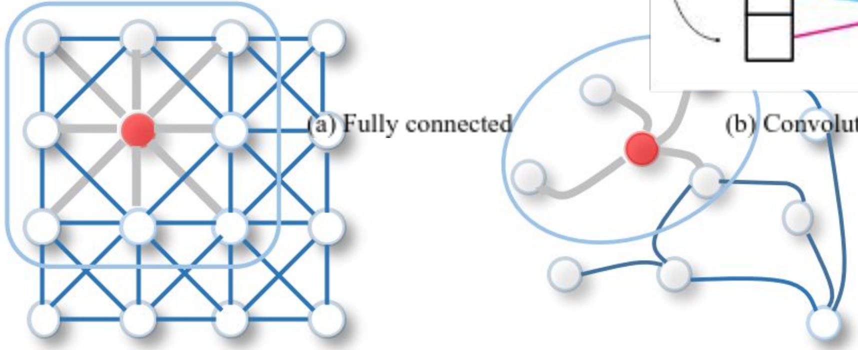
Graph Neural Network

Euclidien space: CNN(image)/RNN(text)

Non-Euclidien spac: Graph

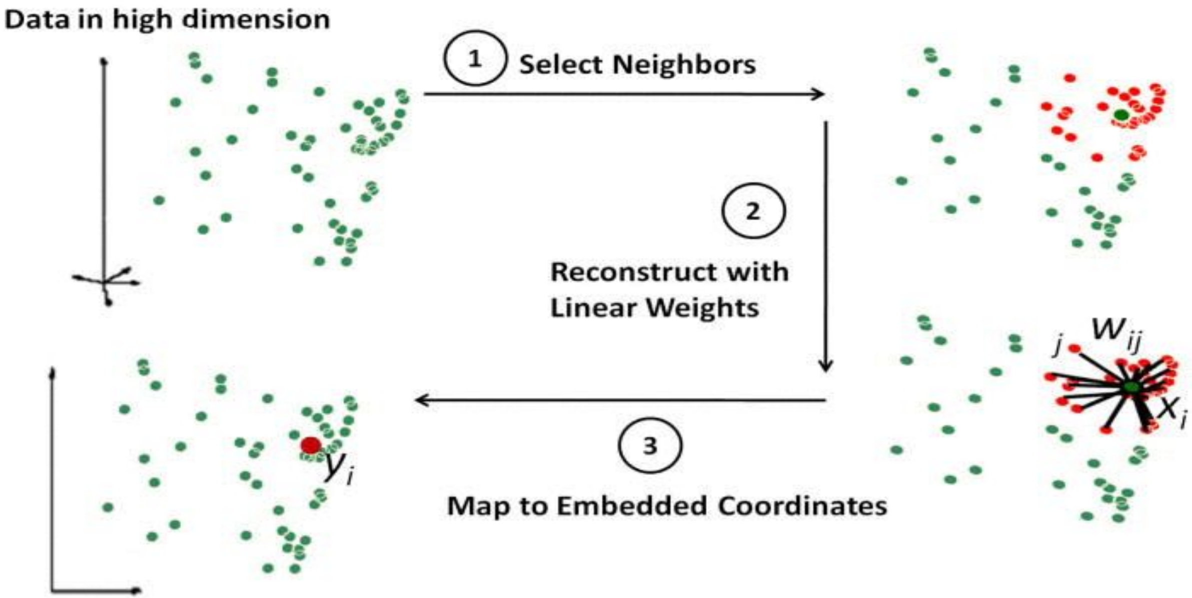
However, we d sam ng or





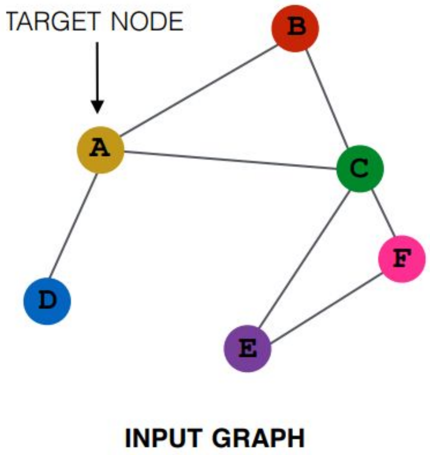
Mainfold learning

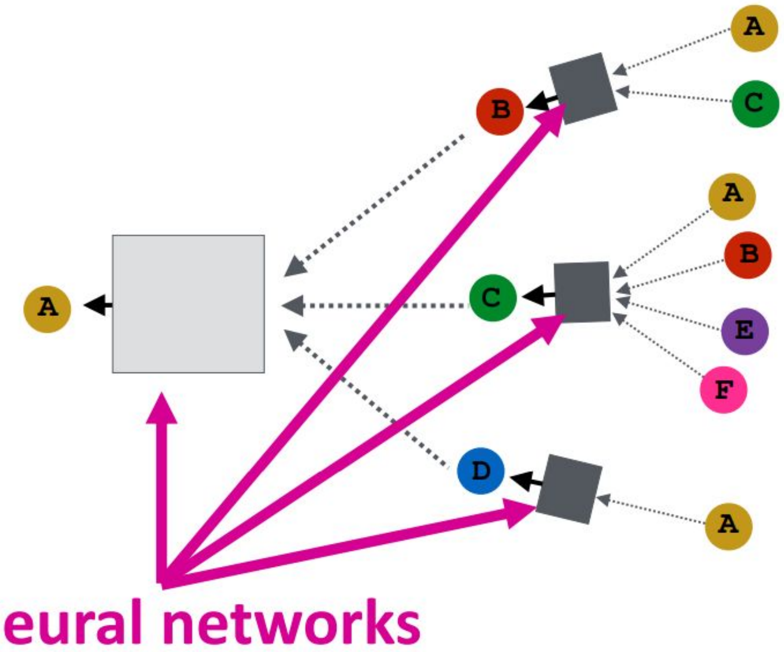
Nonlinear dimensionality reduction by locally linear embedding. In Science. 2000.



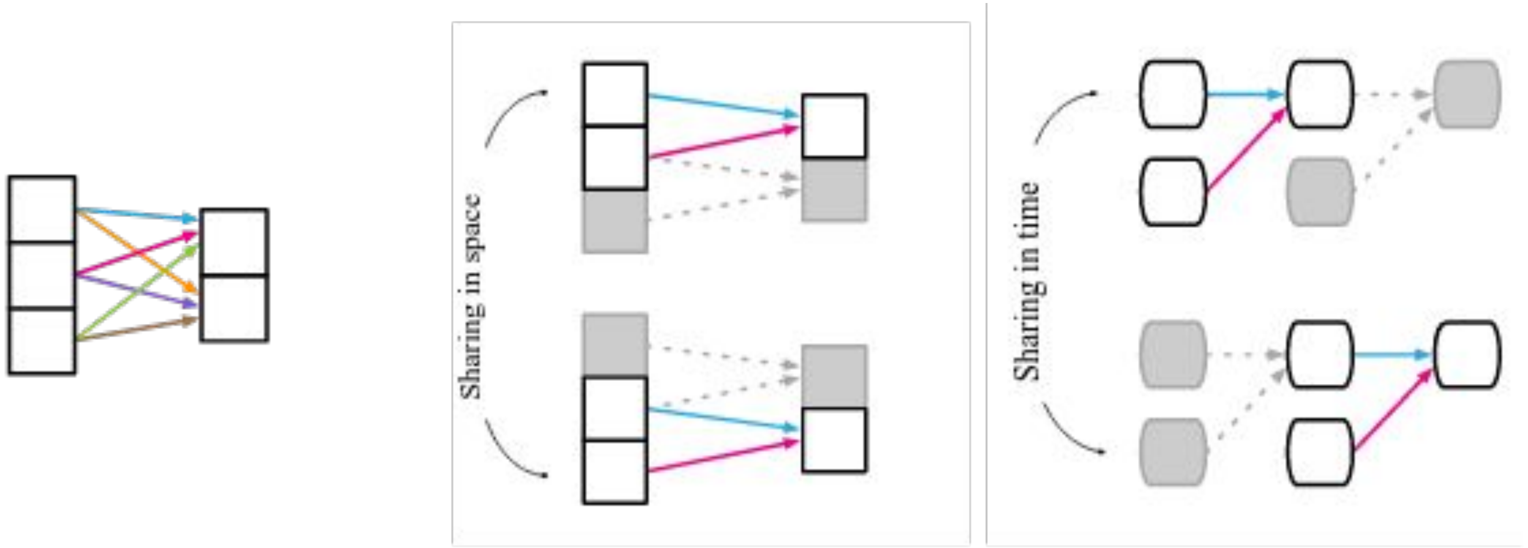
$$z_i = \sum_{v_j \in N(v_i)} W z_j$$

Graph Neural Network





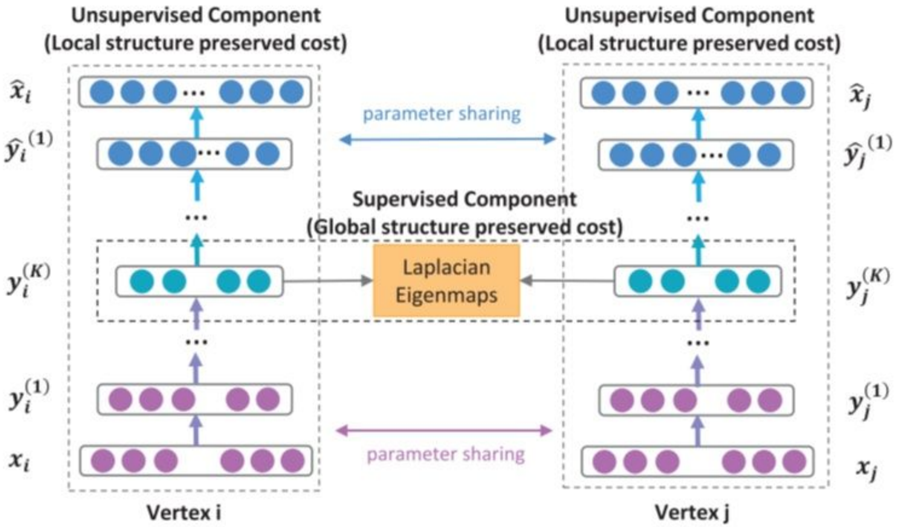
Is All about Aggregator



(a) Fully connected (b) Convolutional (c) Recurrent

Autoencoder

Structral Deep Network Embedding. In KDD, 2016.

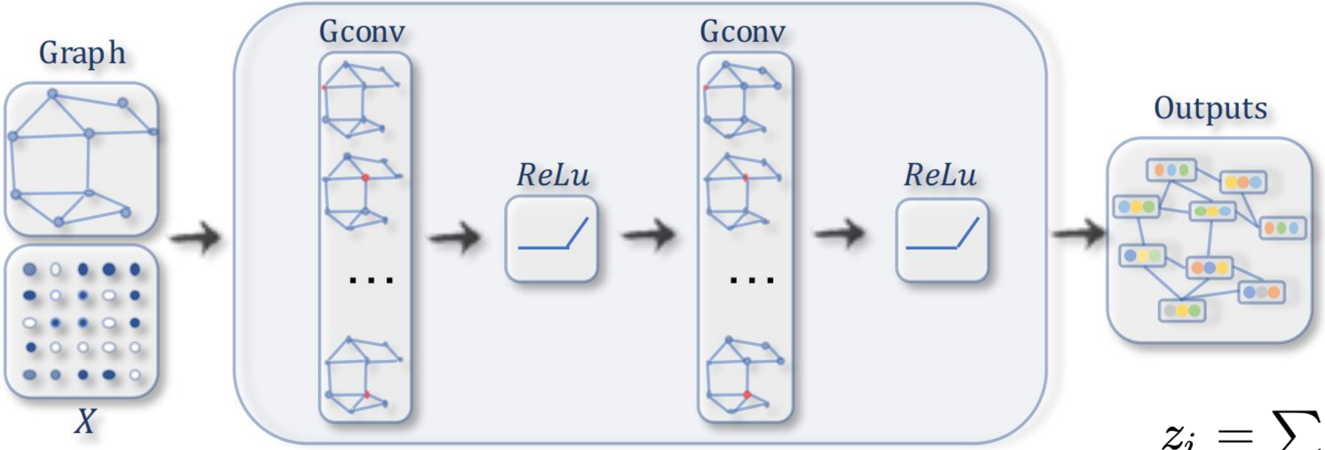


$$z_i = \sum_{v_j \in N(v_i)} W v_j$$

$$z_i = \sum_{v_j \in N(v_i)} z_j$$

Convolutional Neural Network

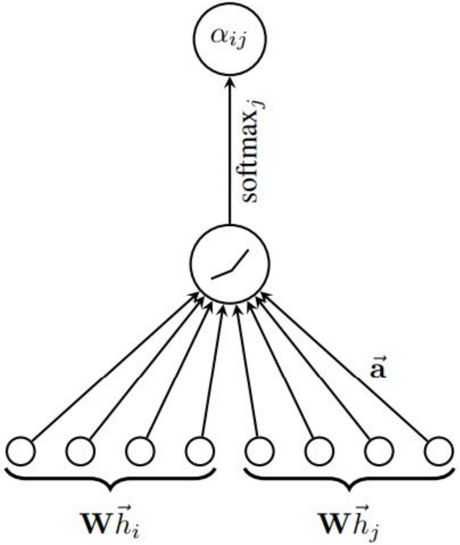
Semi-supervised classification with graph convolutional networks. In NIPS, 2016.

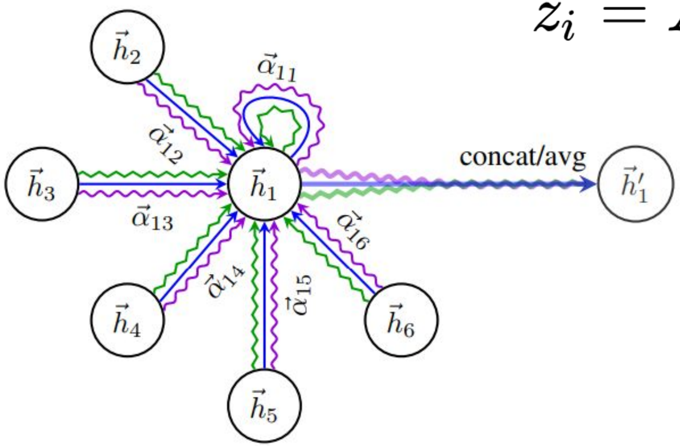


$$z_i = \sum_{v_j \in N(v_i)} \frac{1}{\sqrt{|N(v_i)||N(v_j)|}} z_j$$

Attention

Graph Attention Networks. In ICLR, 2018.

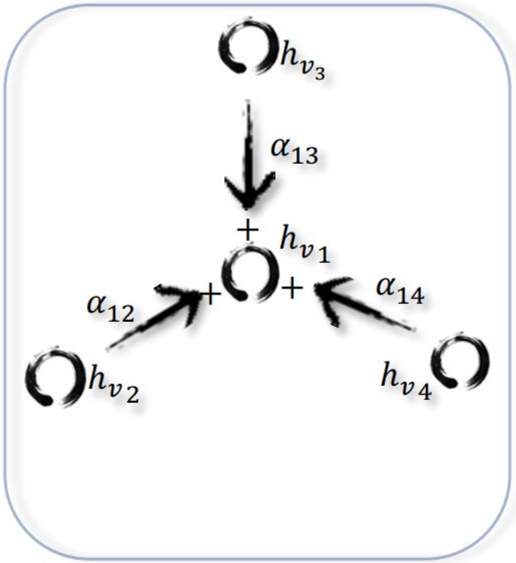




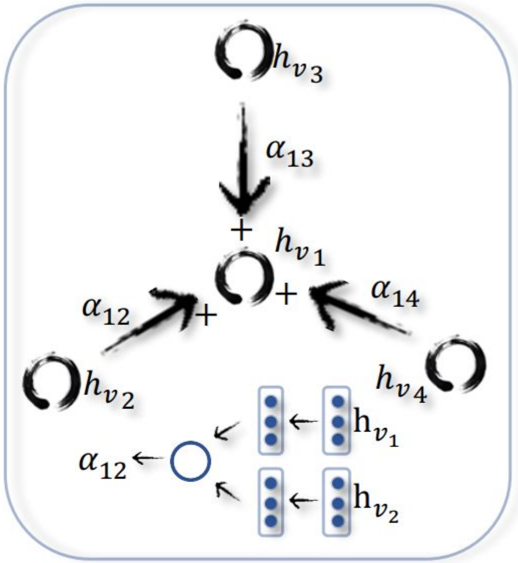
$$z_i = Attn(z_j | v_j \in N(v_i))$$

Attention

GCN

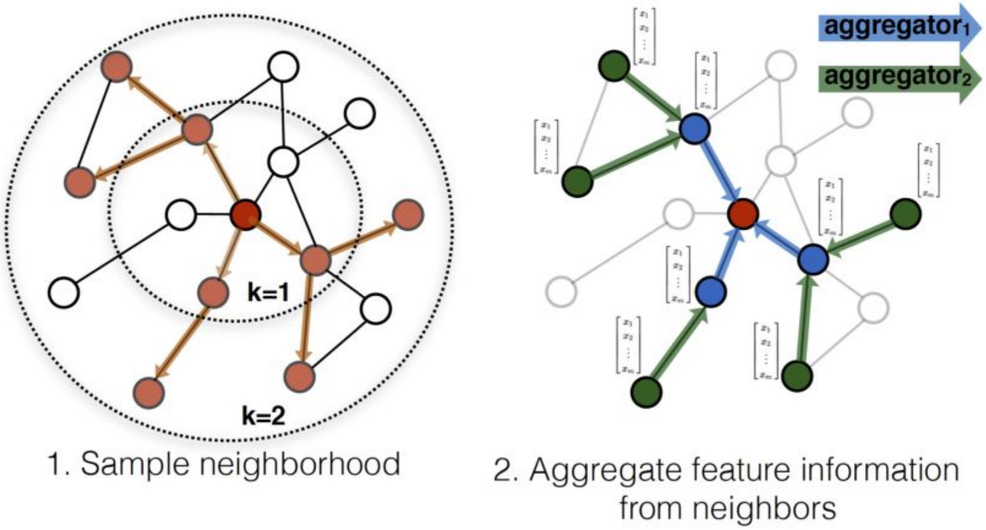


GAT



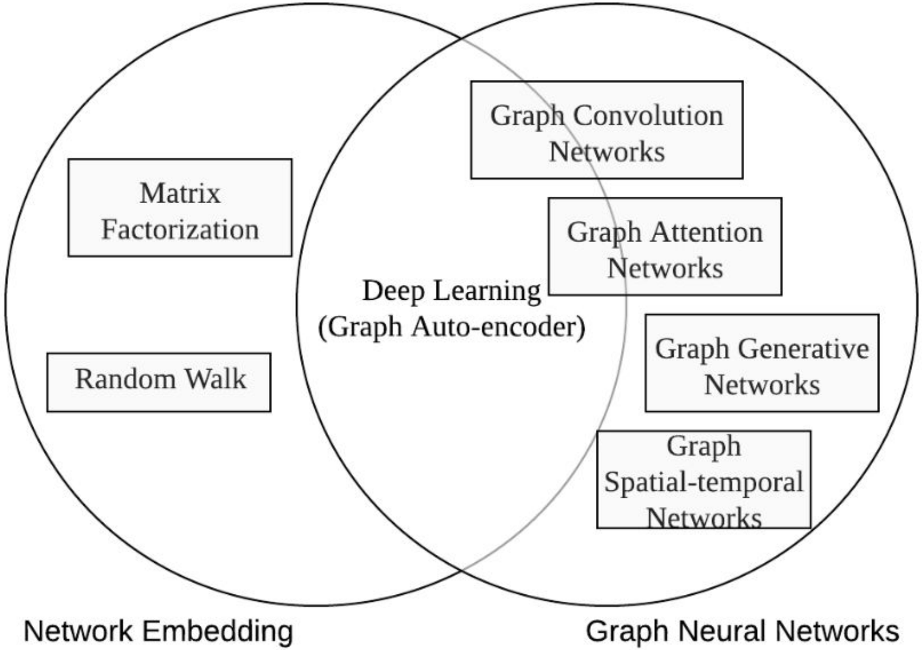
GraphSAGE

Inductive representation learning on large graphs. In NIPS, 2017.



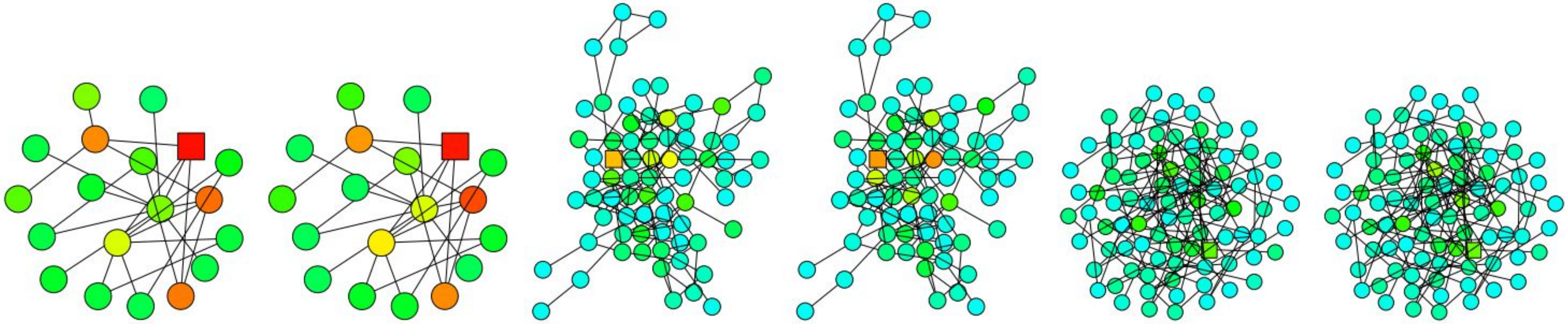
$$z_i = Avg(z_j | v_j \in N(v_i))$$
$$z_i = Pool(z_j | v_j \in N(v_i))$$
$$z_i = LSTM(z_j | v_j \in N(v_i))$$

Nework Embedding vs Graph Neural Network



Deep Net”work”?

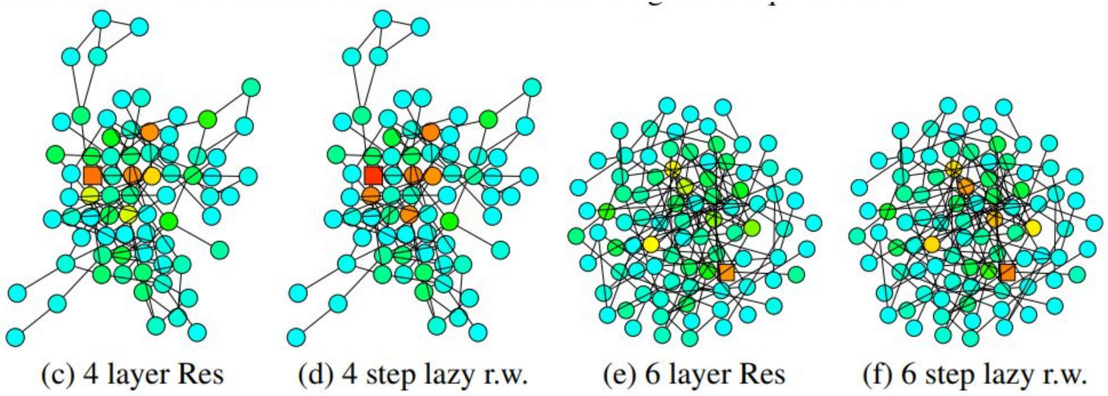
Oversmooth! No more than 2-3 layer.

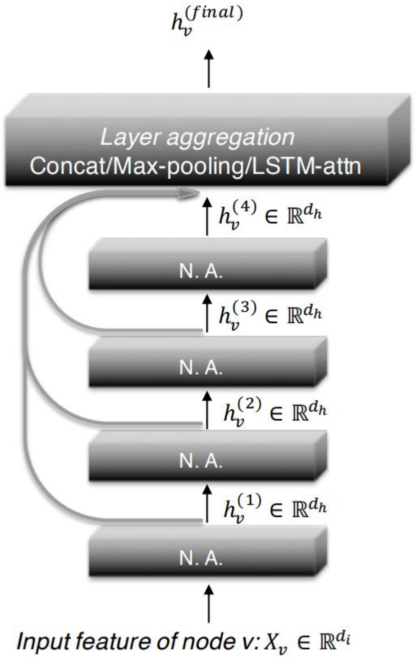


(a) 2 layer GCN (b) 2 step r.w. (c) 4 layer GCN (d) 4 step r.w. (e) 6 layer GCN (f) 6 step r.w.

Highway

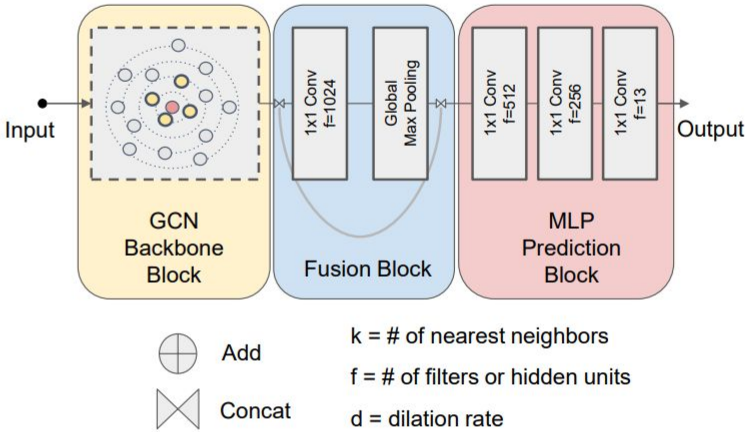
Representation Learning on Graphs with Jumping Knowledge Networks. In ICML, 2018.
DeepGCNs: Can GCNs Go as Deep as CNNs?. In ICCV, 2019.

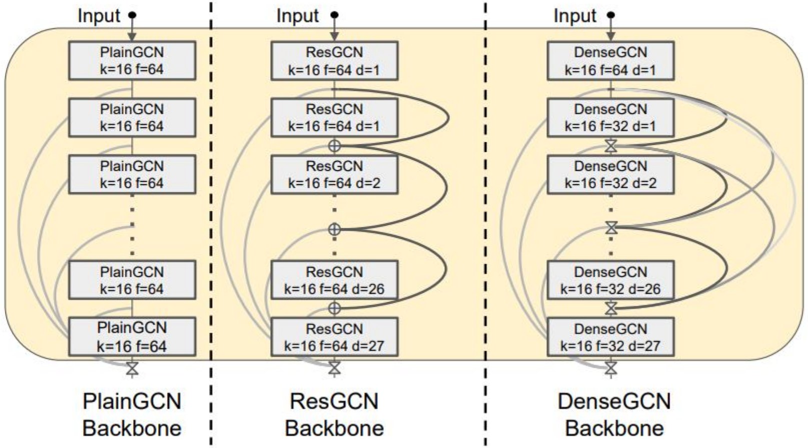




Highway

DeepGCNs: Can GCNs Go as Deep as CNNs?. In ICCV, 2019.

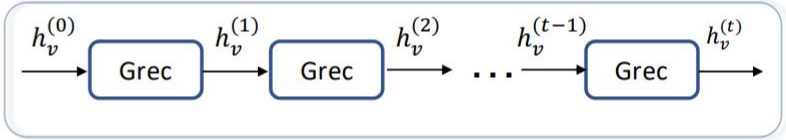




Recurrent Neural Network (Gate)

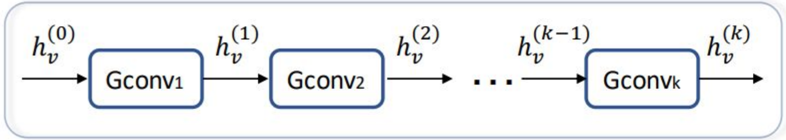
Learning Steady-States of Iterative Algorithms over Graphs. in ICML, 2017.

GGNN



Useful for logical formulas and program.

GCN

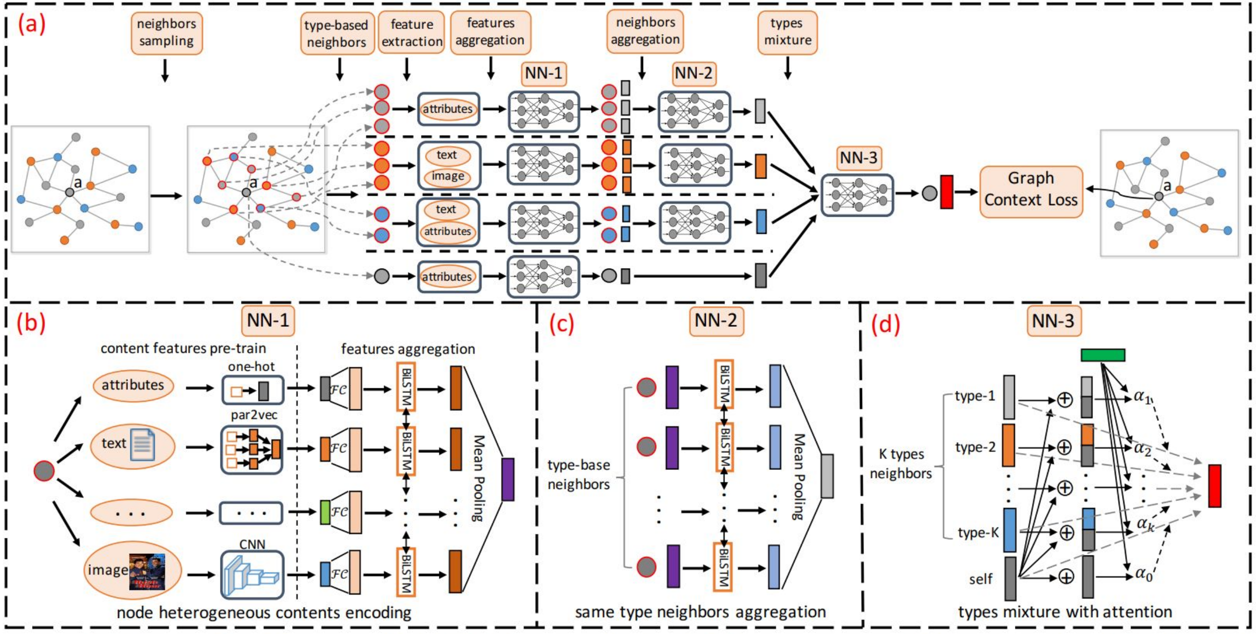


$$z_i = GRU(z_i, \sum_{v_j \in N(v_i)} z_j)$$

$$z_i = \alpha z_i + (1-\alpha) \sum_{v_j \in N(v_i)} z_j$$

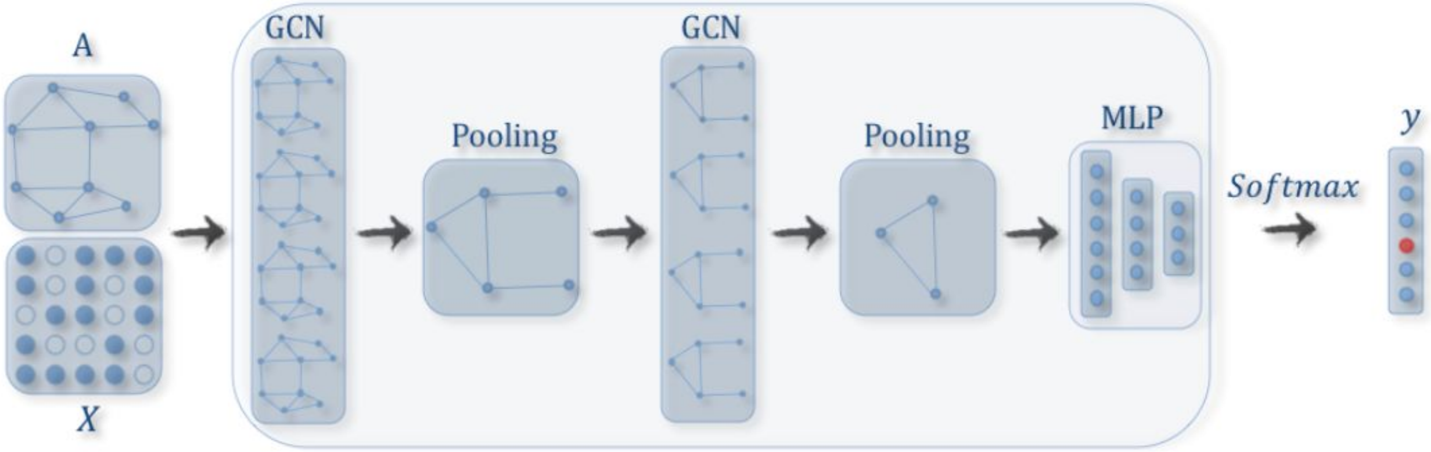
Hetergeneous Information Network

Heterogeneous Graph Neural Network. In KDD, 2019.



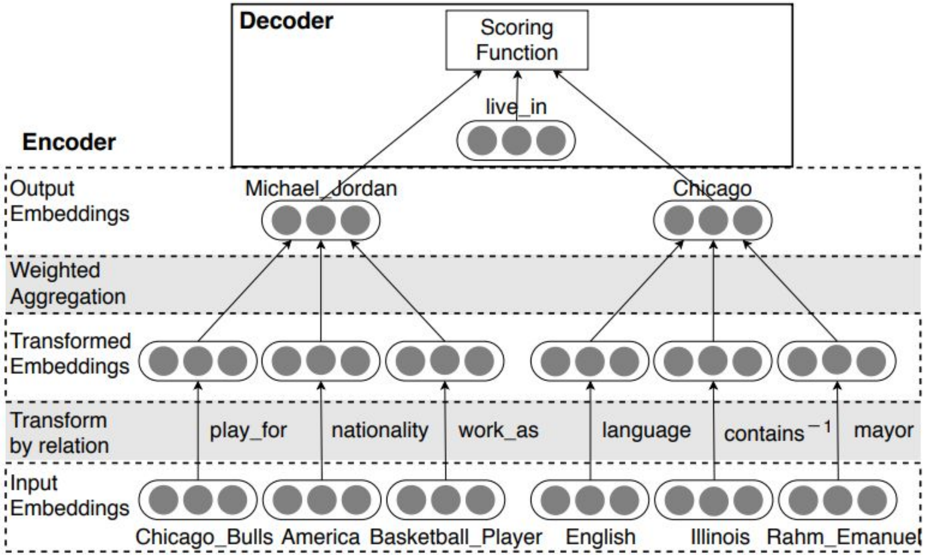
Subgraph: Graph Pooling

Hierarchical Graph Representation Learning with Differentiable Pooling. In NIPS, 2018.
Self-Attention Graph Pooling. In ICML, 2019.
Graph Convolutional Networks with EigenPooling. In KDD, 2019.



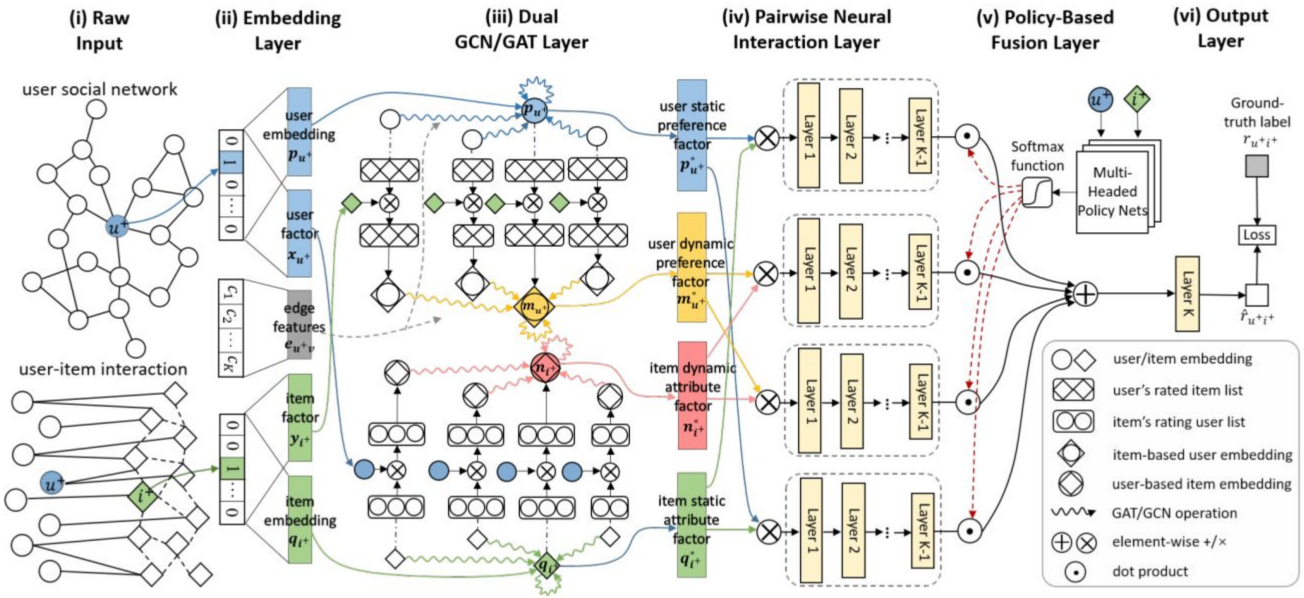
Knowledge Graph

Logic Attention Based Neighborhood Aggregation for Inductive Knowledge Graph Embedding. In AAAI, 2019.



Recommendation

Dual Graph Attention Networks for Deep Latent Representation of Multifaceted Social Effects in Recommender Systems. In WWW, 2019.



Graph Generation

MolGAN: An implicit generative model for small molecular graphs. In arxiv, 2018.