图神经网络研究现状及其在反欺诈领域的应用

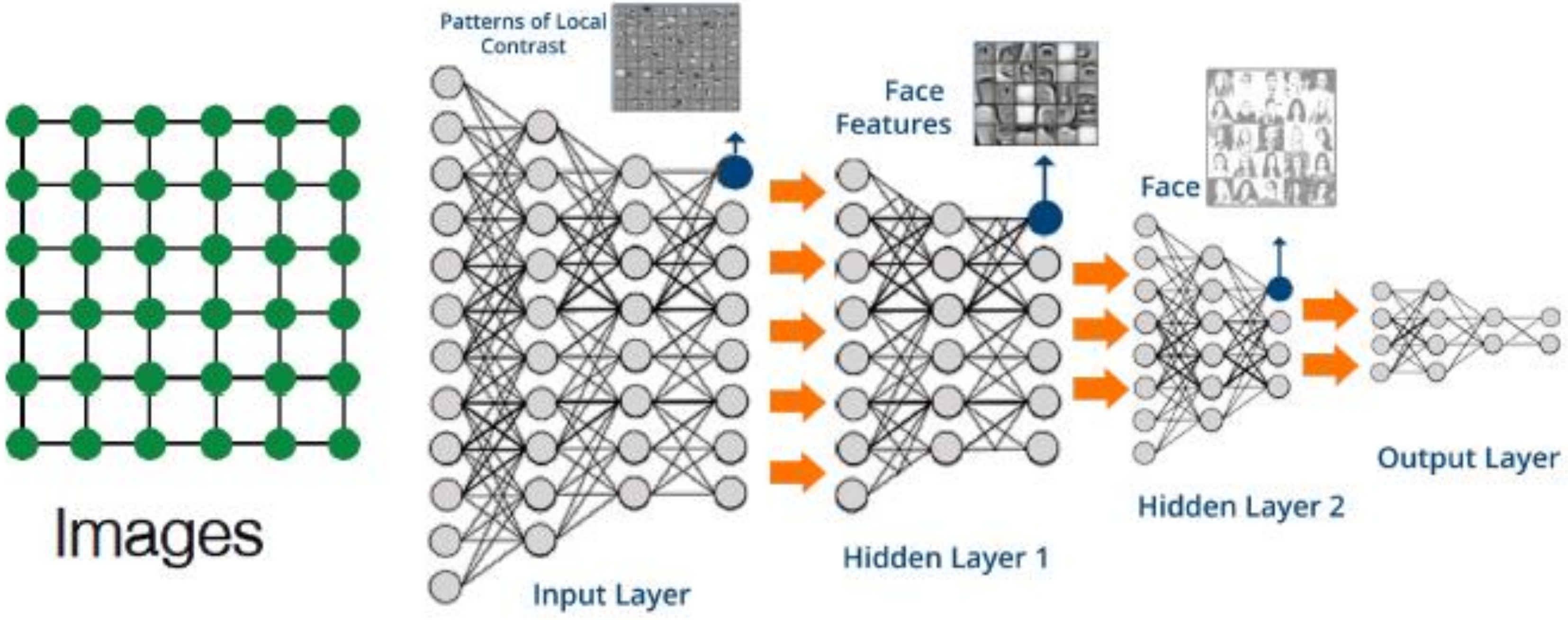
董邦宜
bydong2019@163.com

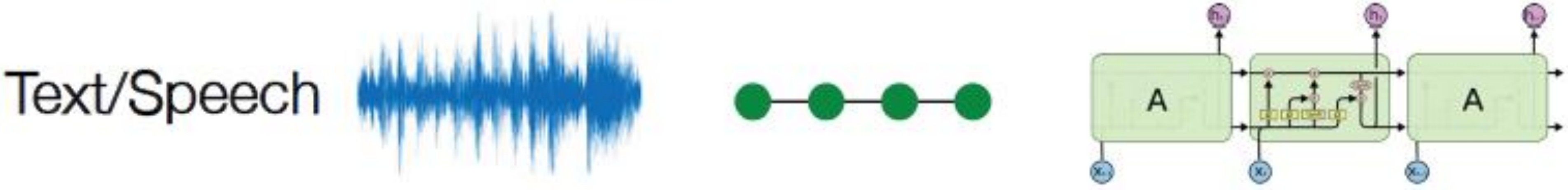
目录

1. 图神经网络的起源

为什么？

- 传统的卷积神经网络、RNN等深度模型只能处理欧式空间的数据（如图像、文本、语音），这些领域的数据具有平移不变性。

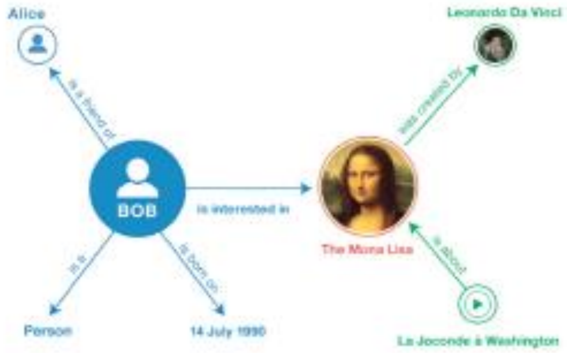




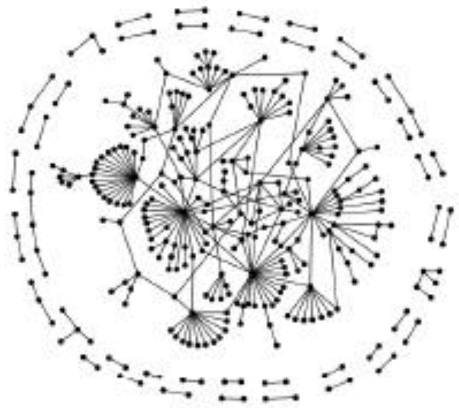
图数据

- 图数据可以自然的表达实际生活中的数据结构，如交通网络、交易关系、生物医学网络、社交网络等。

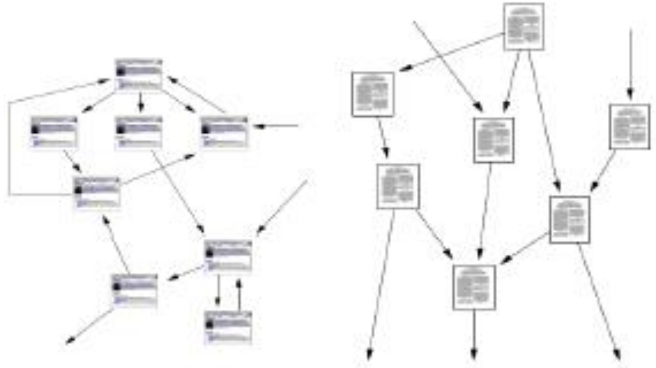
Social networks



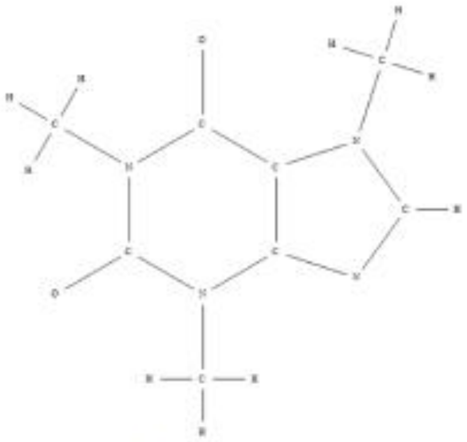
Knowledge graphs



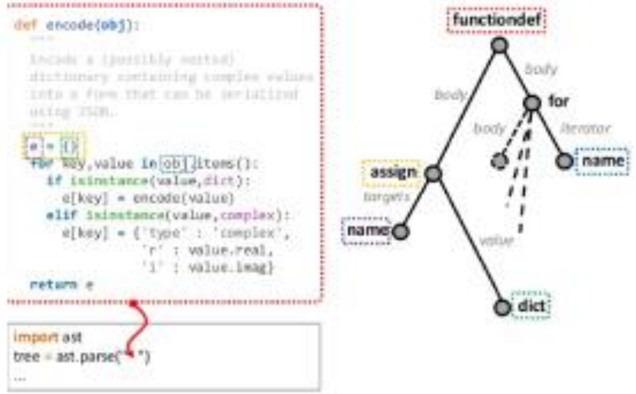
Biological networks



Complex Systems



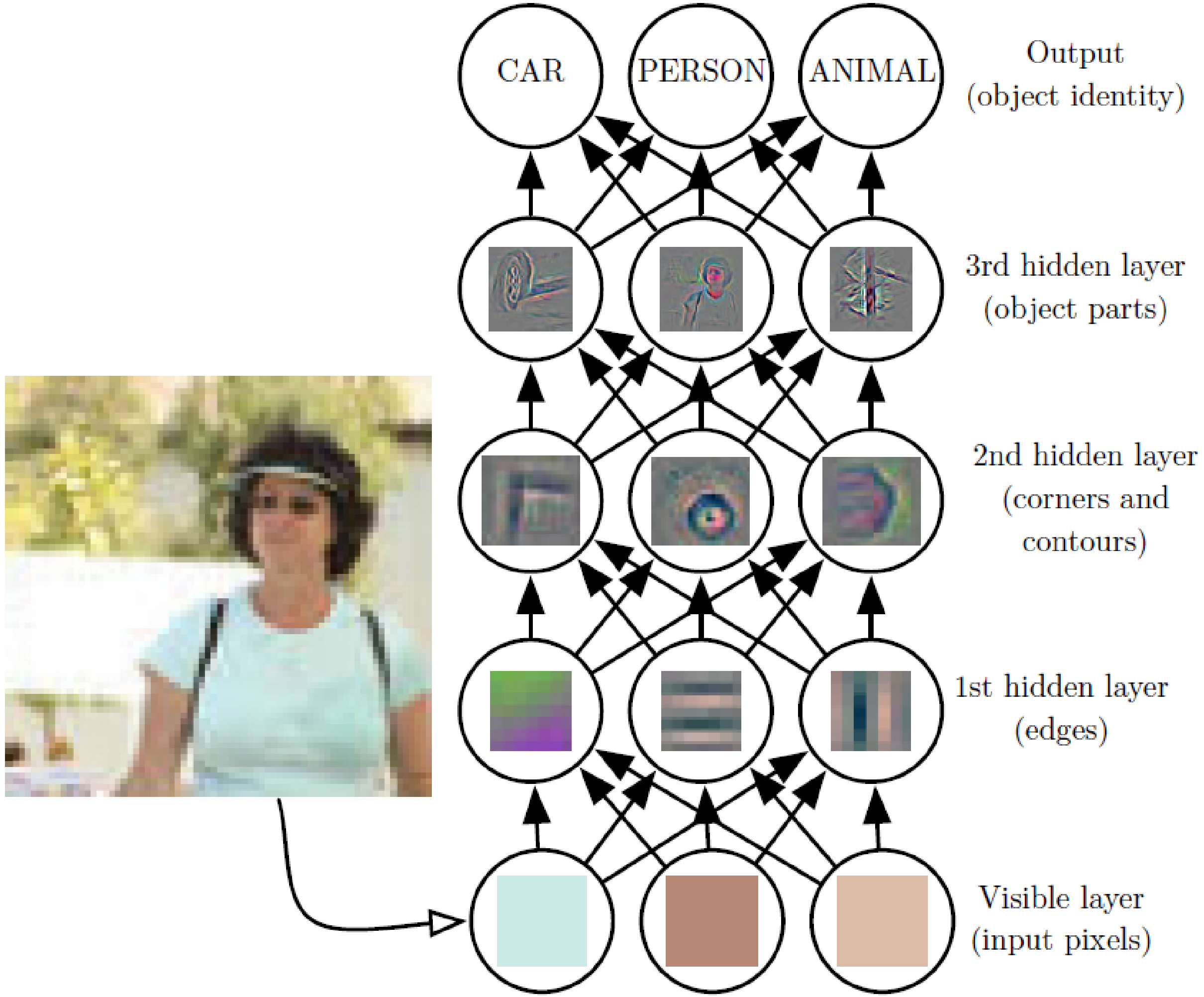
Molecules



Code

为什么要聚合邻居信息？

- 深度学习（deep learning）通过其他较简单的表示来表达复杂表示，解决表示学习中的核心问题
- 在现实中，一个物体不仅和自身的属性有关，还和与其交互的其他物体有紧密的联系，其表现的行为往往是自身和外界交互的结果所导致的
- 图结构数据能够很好地刻画单个物体与外界交互的情况
- 图神经网络用在图结构数据上，能够同时考虑到物体自身的属以及与之交互的信息



2. 图神经网络的基础方法

图神经网络方法分类

图神经网络

基于谱的方法

在谱域上定义图卷积

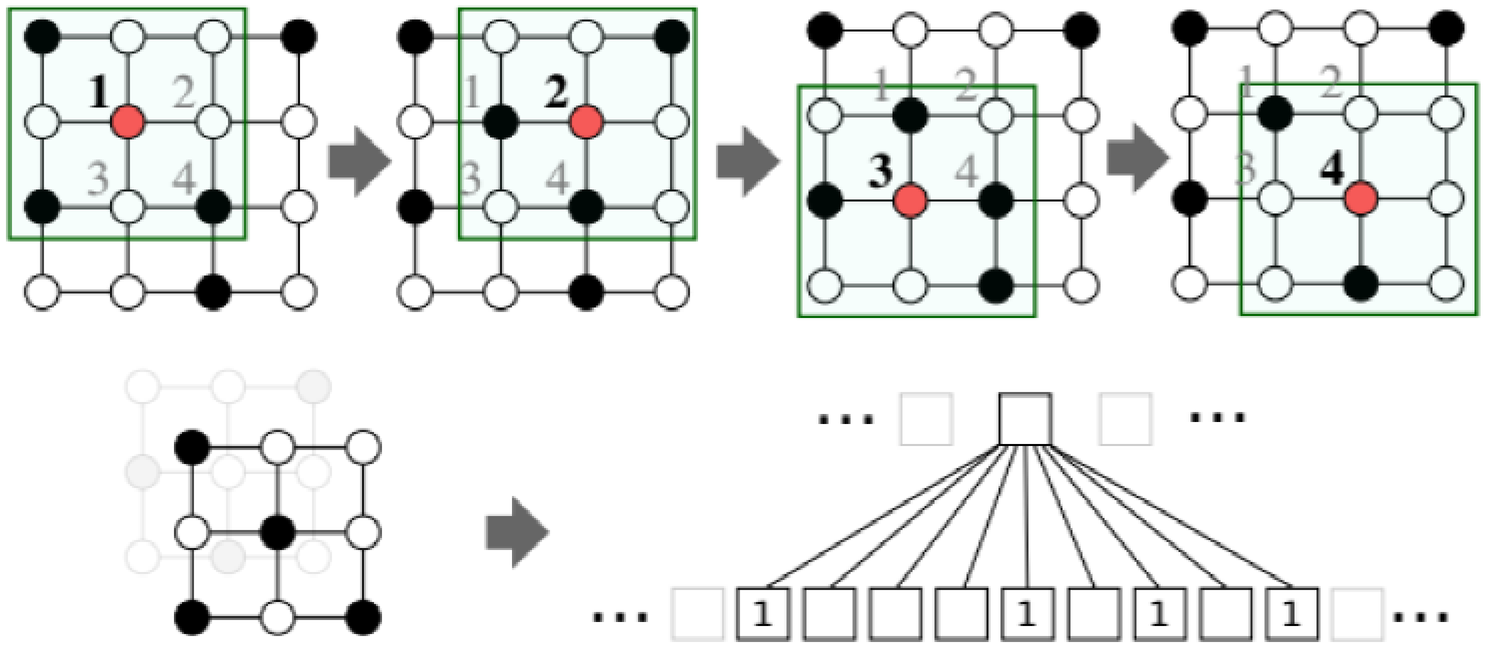
谱方法能够考虑到全局图的信息，但是很难并行化、不能扩展到大规模图数据上

基于空间的方法

在空间域上定义图卷积

空间的方法主要是用于聚合局部的邻居信息，能够通过批量化、采样的方式来扩展到大规模的图数据上

类比卷积神经网络



1. 选取指定数量的邻居

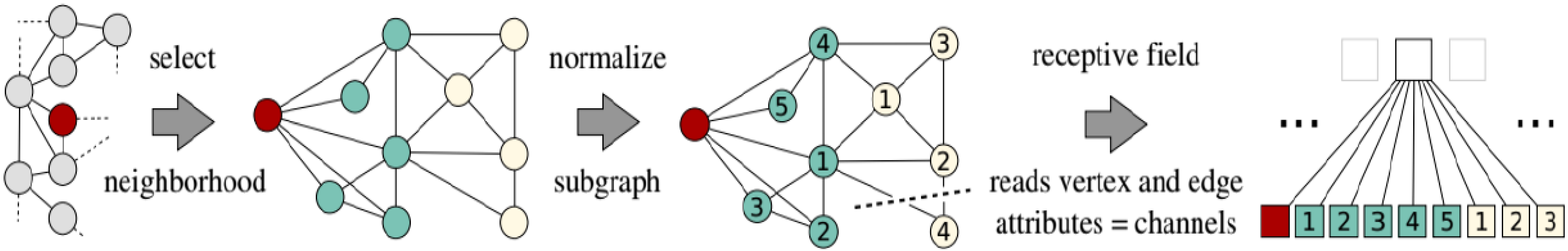
2. 给邻居编号

3. 参数共享

M. Niepert, M. Ahmed, K. Kutzkov. Learning Convolutional Neural Networks for Graphs. ICML, 2016.

卷积神经网络→图神经网络

- 根据节点之间的跳数(proximity metric)，为每个节点选取固定的邻居节点
- 根据节点之间的跳数（proximity metric），给邻居节点进行编号
- 参数共享



1. 选取固定数量邻居节点

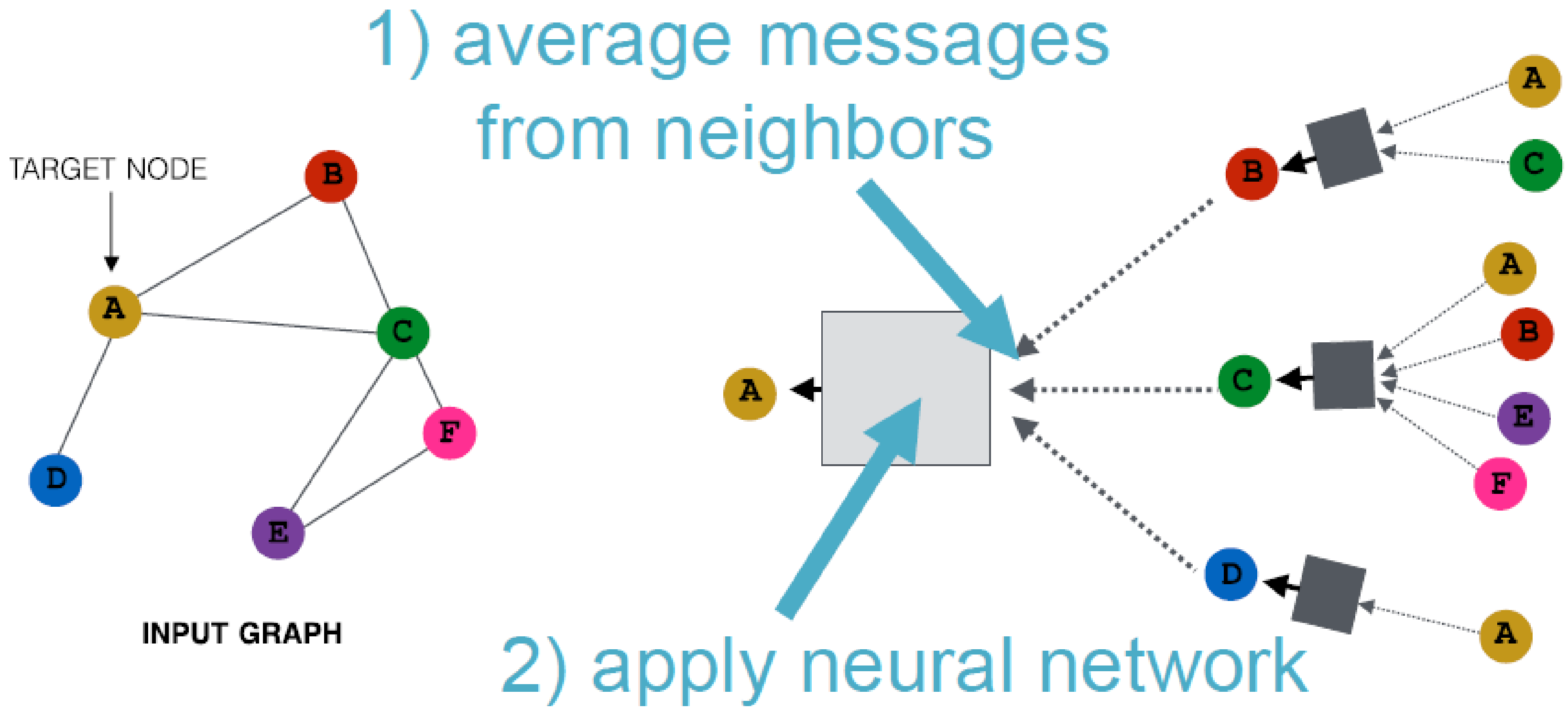
2. 邻居节点编号

3. 参数共享

M. Niepert, M. Ahmed, K. Kutzkov. Learning Convolutional Neural Networks for Graphs. ICML, 2016.

邻居信息聚合

- **Basic approach:** Average neighbor information and apply a neural network.



邻居信息聚合

- **Basic approach:** Average neighbor information and apply a neural network.

$$\mathbf{h}_v^0 = \mathbf{x}_v$$

Initial "layer 0" embeddings are equal to node features

$$\mathbf{h}_v^k = \sigma\left(\mathbf{W}_k \sum_{u \in N(v)} \frac{\mathbf{h}_u^{k-1}}{|N(v)|} + \mathbf{B}_k \mathbf{h}_v^{k-1}\right), \quad \forall k > 0$$

previous layer embedding of v

kth layer embedding of v

non-linearity (e.g., ReLU or tanh)

average of neighbor's previous layer embeddings

Graph neural network（GCN）

Basic Neighborhood Aggregation

$$\mathbf{h}_v^k = \sigma\left(\mathbf{W}_k \sum_{u \in N(v)} \frac{\mathbf{h}_u^{k-1}}{|N(v)|} + \mathbf{B}_k \mathbf{h}_v^{k-1}\right)$$

V.S.

$$H^{(k)} = \sigma\left(\tilde{D}^{-\frac{1}{2}}\tilde{A}\tilde{D}^{-\frac{1}{2}}H^{(k-1)}W^{(k)}\right)$$

GCN Neighborhood Aggregation

$$\mathbf{h}_v^k = \sigma\left(\mathbf{W}_k \sum_{u \in N(v) \cup v} \frac{\mathbf{h}_u^{k-1}}{\sqrt{|N(u)||N(v)|}}\right)$$

$$\tilde{A} = A + I_N, \tilde{D}_{ii} = \sum_j \tilde{A}_{ij}$$

use the same transformation matrix for self and neighbor embeddings

instead of simple average, normalization varies across neighbors

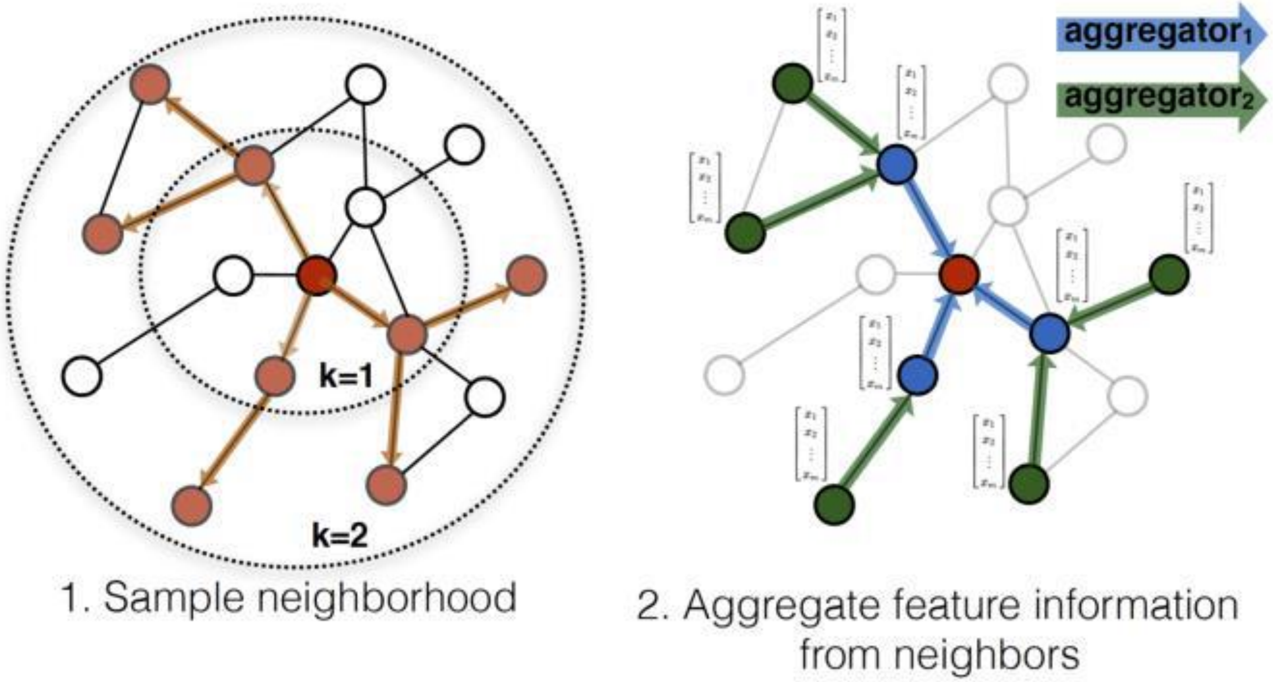
Kipf T N, Welling M. Semi-supervised classification with graph convolutional networks[J]. arXiv preprint arXiv:1609 .02907, 2016.

GraphSAGE

- Aggregating neighbors
- Sampling neighbors

$$a_v^{(k)} = \mathrm{AGGREGATE}^{(k)}\left(\left\{h_u^{(k-1)} : u \in \mathcal{N}(v)\right\}\right)$$

$$h_v^{(k)} = \mathrm{COMBINE}^{(k)}\left(h_v^{(k-1)}, a_v^{(k)}\right)$$



GraphSAGE: Inductive Learning

General framework of graph neural networks: Aggregate the information of neighboring nodes to update the representation of center node

Hamilton W, Ying Z, Leskovec J. Inductive representation learning on large graphs[C]//Advances in neural information processing systems. 2017: 1024-1034.

GraphSAGE v.s. GCN

- Simple neighborhood aggregation:

$$\mathbf{h}_v^k = \sigma\left(\mathbf{W}_k \sum_{u \in N(v)} \frac{\mathbf{h}_u^{k-1}}{|N(v)|} + \mathbf{B}_k \mathbf{h}_v^{k-1}\right)$$

- GraphSAGE:

concatenate self embedding and neighbor embedding

$$\mathbf{h}_v^k = \sigma\left(\left[\mathbf{W}_k \cdot \boxed{\mathrm{AGG}\left(\{\mathbf{h}_u^{k-1}, \forall u \in N(v)\}\right)}, \mathbf{B}_k \mathbf{h}_v^{k-1}\right]\right)$$

generalized aggregation

Hamilton W, Ying Z, Leskovec J. Inductive representation learning on large graphs[C]//Advances in neural information processing systems. 2017: 1024-1034.

Aggregation

- **Mean:**

$$\mathrm{AGG} = \sum_{u \in N(v)} \frac{\mathbf{h}_u^{k-1}}{|N(v)|}$$

- **Pool**
  - Transform neighbor vectors and apply symmetric vector function.

element-wise mean/max

$$\mathrm{AGG} = \boxed{\gamma}\left(\{\mathbf{Q}\mathbf{h}_u^{k-1}, \forall u \in N(v)\}\right)$$

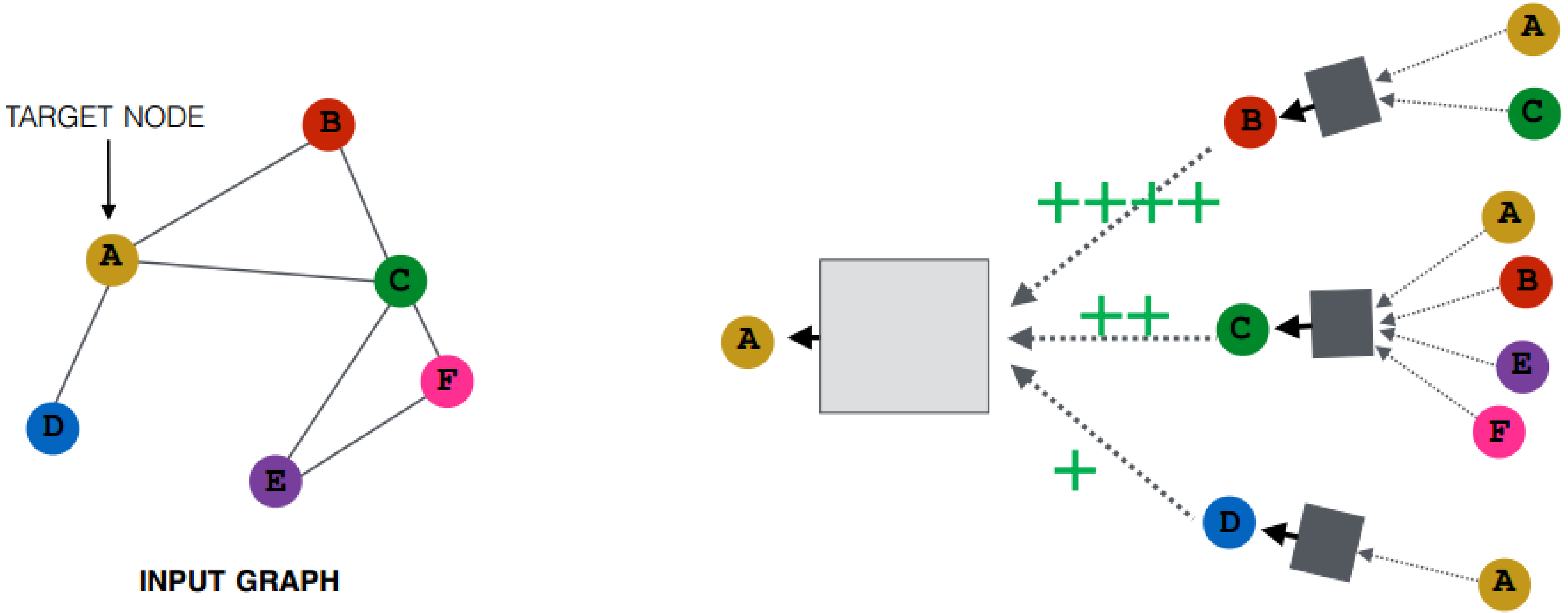
- **LSTM:**
  - Apply LSTM to random permutation of neighbors.

$$\mathrm{AGG} = \mathrm{LSTM}\left([\mathbf{h}_u^{k-1}, \forall u \in \pi(N(v))]\right)$$

Hamilton W, Ying Z, Leskovec J. Inductive representation learning on large graphs[C]//Advances in neural information processing systems. 2017: 1024-1034.

Graph Attention Networks

- 在邻居节点中，节点的重要性相同吗？



- 采用注意力机制来对不同的邻居节点采取不同的重要性

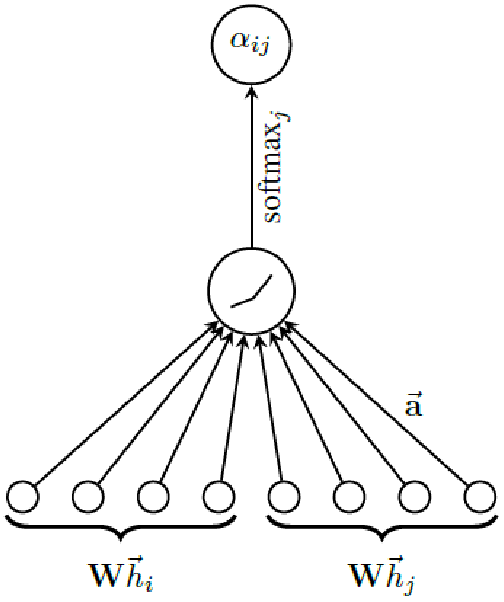
Graph Attention Network（GAT）

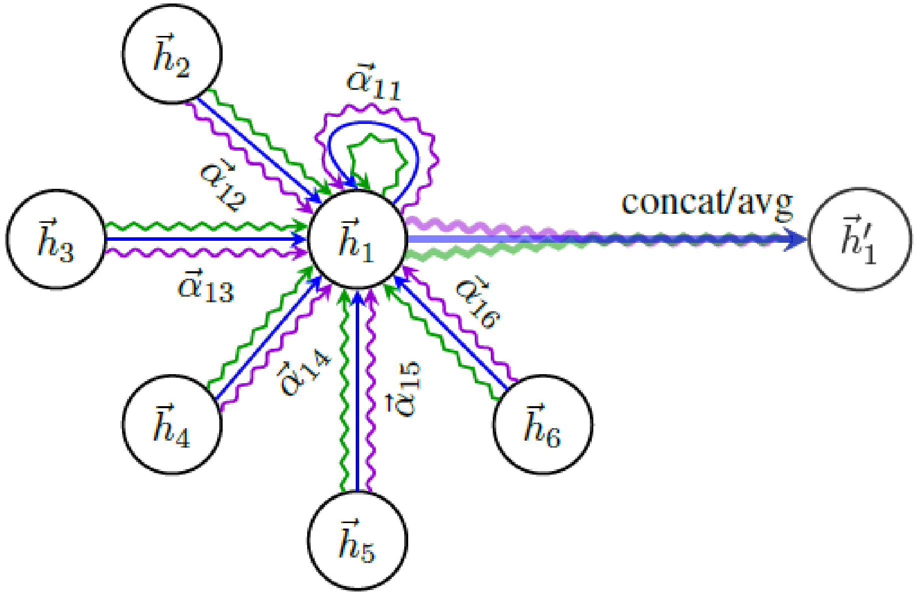
GAT Neighborhood Aggregation

$$\mathbf{h}_v^k = \sigma(\mathbf{W}_k \sum_{u \in N(v) \cup \{v\}} \alpha_{v,u} \mathbf{h}_u^{k-1})$$

Learned attention weights!

$$\alpha_{v,u} = \frac{\exp\left(\text{LeakyReLU}\left(\mathbf{a}^\top[\mathbf{Q}\mathbf{h}_v, \mathbf{Q}\mathbf{h}_u]\right)\right)}{\sum_{u' \in N(v) \cup \{v\}} \exp\left(\text{LeakyReLU}\left(\mathbf{a}^\top[\mathbf{Q}\mathbf{h}_v, \mathbf{Q}\mathbf{h}_{u'}]\right)\right)}$$





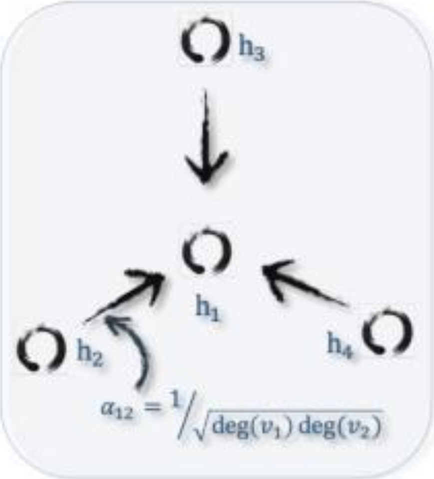
Veličković P, Cucurull G, Casanova A, et al. Graph attention networks[J]. arXiv preprint arXiv:1710.10903, 2017.

GAT v.s. GCN

GAT Neighborhood Aggregation

$$\mathbf{h}_v^k = \sigma(\mathbf{W}_k \sum_{u \in N(v) \cup \{v\}} \alpha_{v,u} \mathbf{h}_u^{k-1})$$

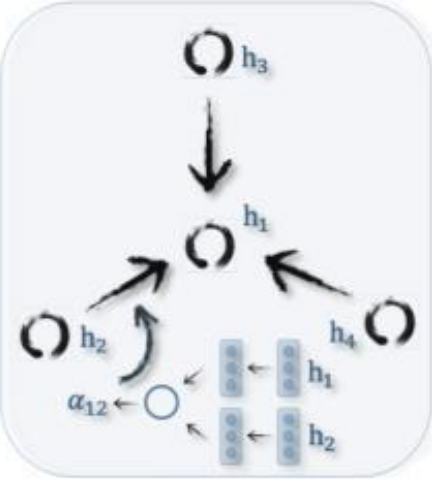
learned attention weights



GCN Neighborhood Aggregation

$$\mathbf{h}_v^k = \sigma\left(\mathbf{W}_k \sum_{u \in N(v) \cup v} \frac{\mathbf{h}_u^{k-1}}{\sqrt{|N(u)||N(v)|}}\right)$$

attention weights defined by degree

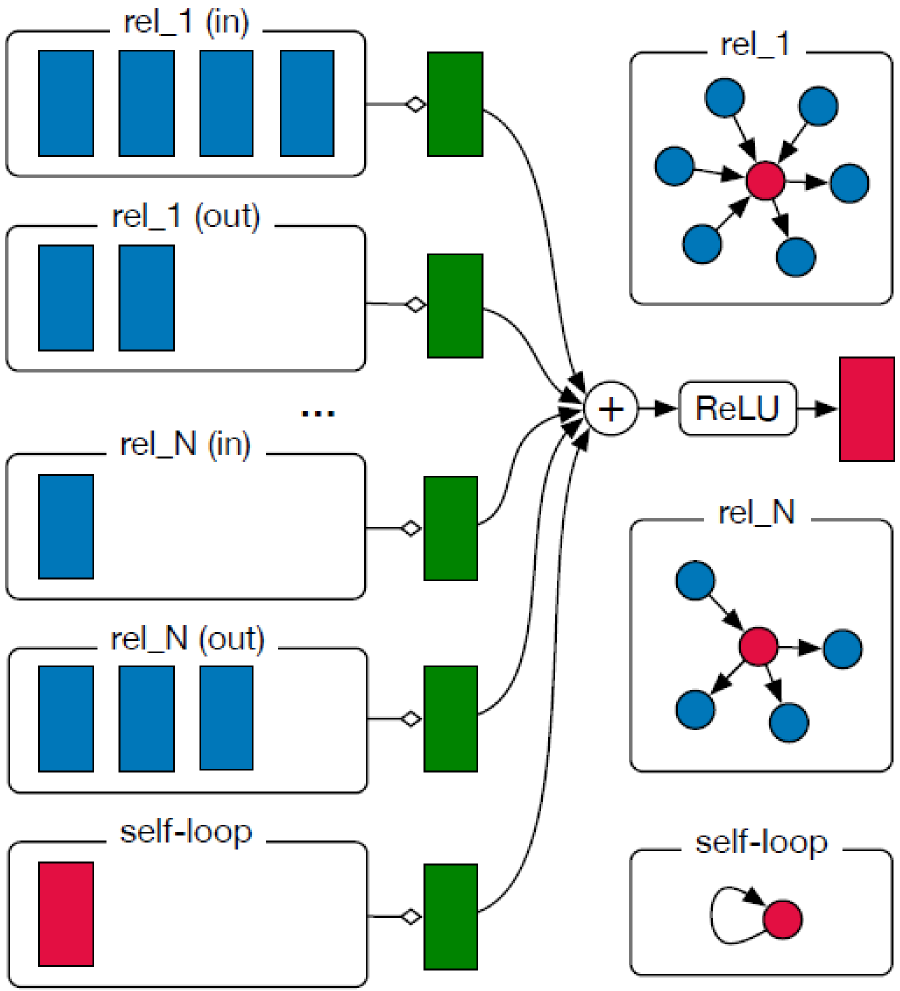


Veličković P, Cucurull G, Casanova A, et al. Graph attention networks[J]. arXiv preprint arXiv:1710.10903, 2017.

R-GCNs

- 问题：在异质信息网络中，不同的关系之间的影响不一样
- 解决方案：将异质信信息网络拆分成多个子网络

$$h_i^{(l+1)} = \sigma\left(\sum_{r\in\mathcal{R}}\sum_{j\in\mathcal{N}_i^r}\frac{1}{c_{i,r}}W_r^{(l)}h_j^{(l)} + W_0^{(l)}h_i^{(l)}\right)$$

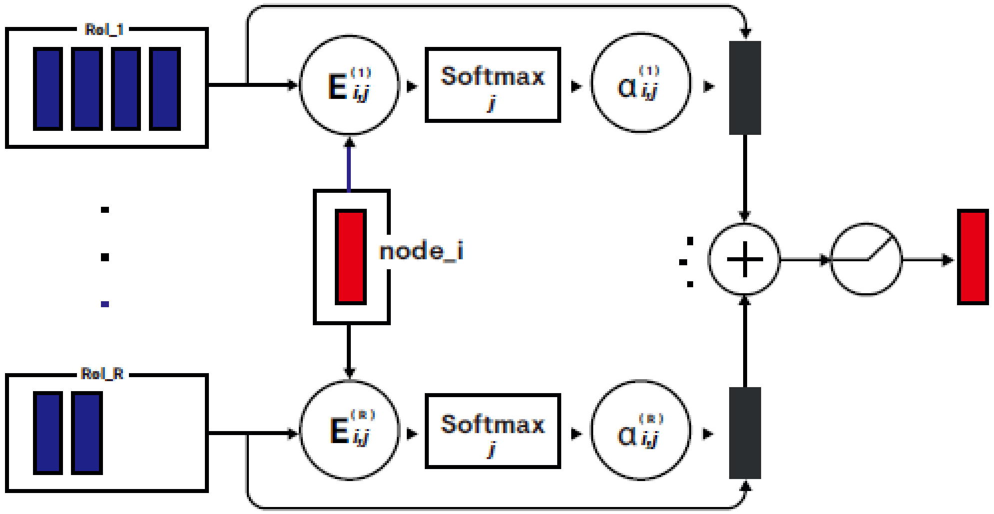


Schlichtkrull M, Kipf T N, Bloem P, et al. Modeling relational data with graph convolutional networks[C]//European Semantic Web Conference. Springer, Cham, 2018: 593-607.

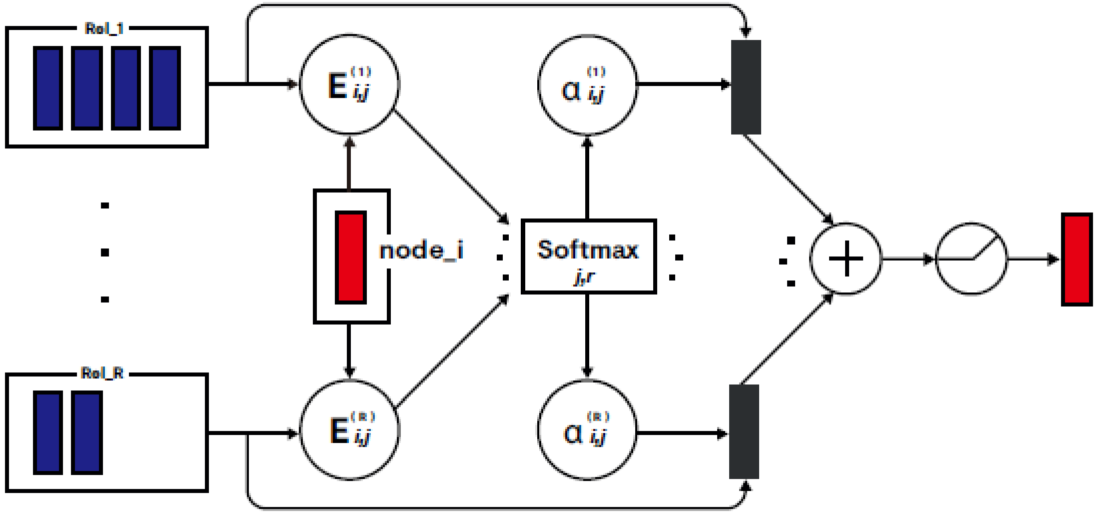
R-GAT

$$\alpha_{i,j}^{(r)} = \underset{j}{\mathrm{softmax}}\left(E_{i,j}^{(r)}\right) = \frac{\exp\left(E_{i,j}^{(r)}\right)}{\sum_{k\in\mathcal{N}_i^{(r)}}\exp\left(E_{i,k}^{(r)}\right)}, \qquad \forall i,r: \sum_{j\in\mathcal{N}_i^{(r)}} \alpha_{i,j}^{(r)} = 1. \qquad (6)$$

$$\alpha_{i,j}^{(r)} = \underset{j,r}{\mathrm{softmax}}\left(E_{i,j}^{(r)}\right) = \frac{\exp\left(E_{i,j}^{(r)}\right)}{\sum_{r'\in\mathcal{R}}\sum_{k\in\mathcal{N}_i^{(r')}}\exp\left(E_{i,k}^{(r')}\right)}, \qquad \forall i: \sum_{r\in\mathcal{R}}\sum_{j\in\mathcal{N}_i^{(r)}} \alpha_{i,j}^{(r)} = 1.$$



(1） 关系内的图注意力机制

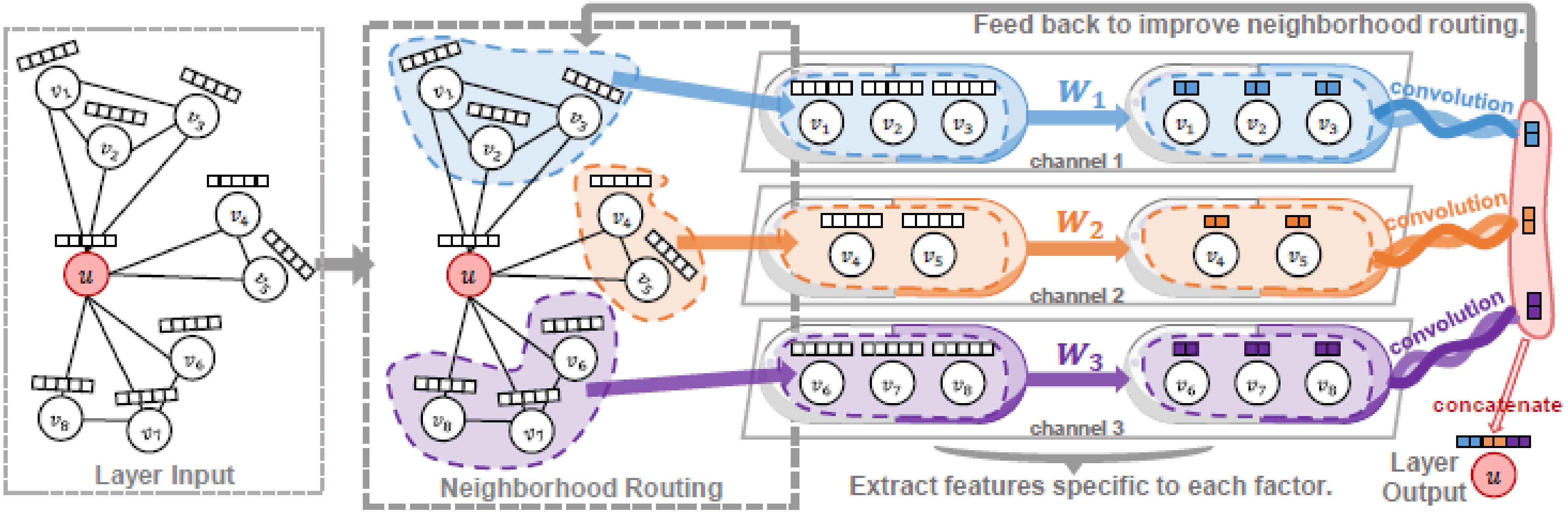


(2)跨关系的图注意力机制

Busbridge D, Sherburn D, Cavallo P, et al. Relational graph attention networks[J]. arXiv preprint arXiv:1904.05811, 2019.

DisenGNN

- 邻居中的个体可以聚集成不同的类
- 同一类别对节点影响相似，不同类别内的对节点的重要性不同



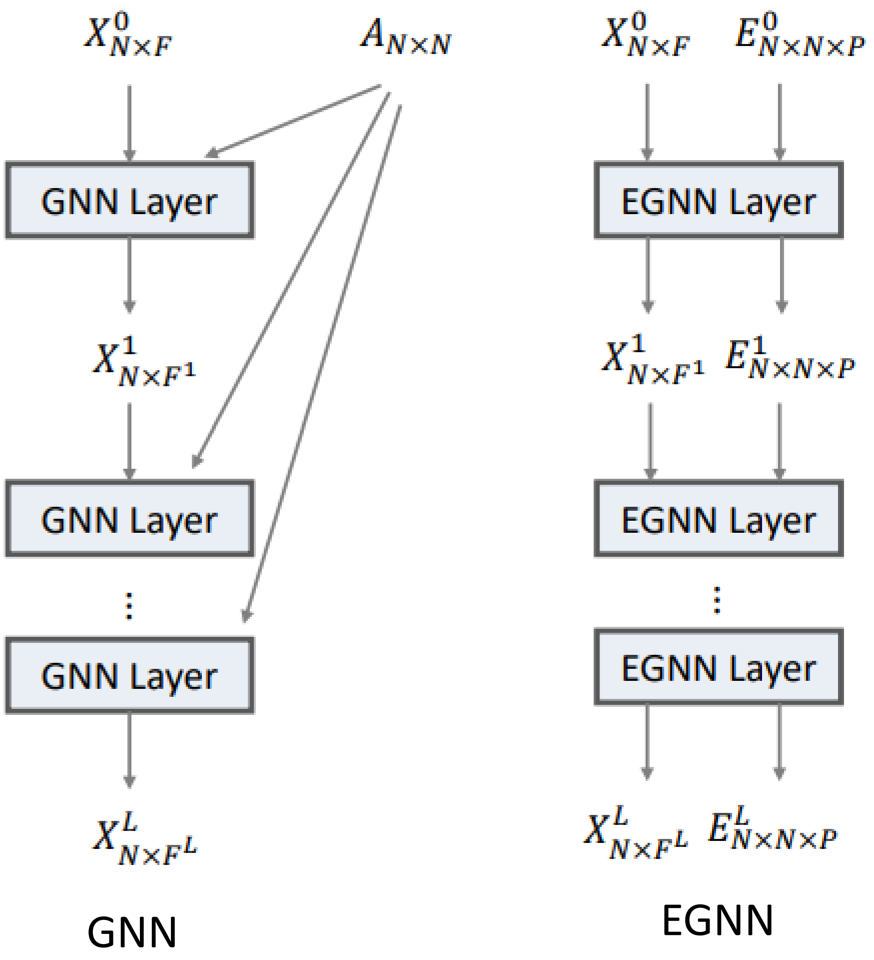
Ma J, Cui P, Kuang K, et al. Disentangled graph convolutional networks[C]//International Conference on Machine Learning. 2019: 4212-4221.

EGNN

- 相邻节点的边上，不再是单一的值，而是一个向量，包含多个属性信息

$$X^l = \sigma\left[\Big\Vert_{p=1}^{P}\left(\boxed{\alpha^l_{\cdot\cdot p}(X^{l-1}, E^{l-1}_{\cdot\cdot p})}\, g^l(X^{l-1})\right)\right].$$

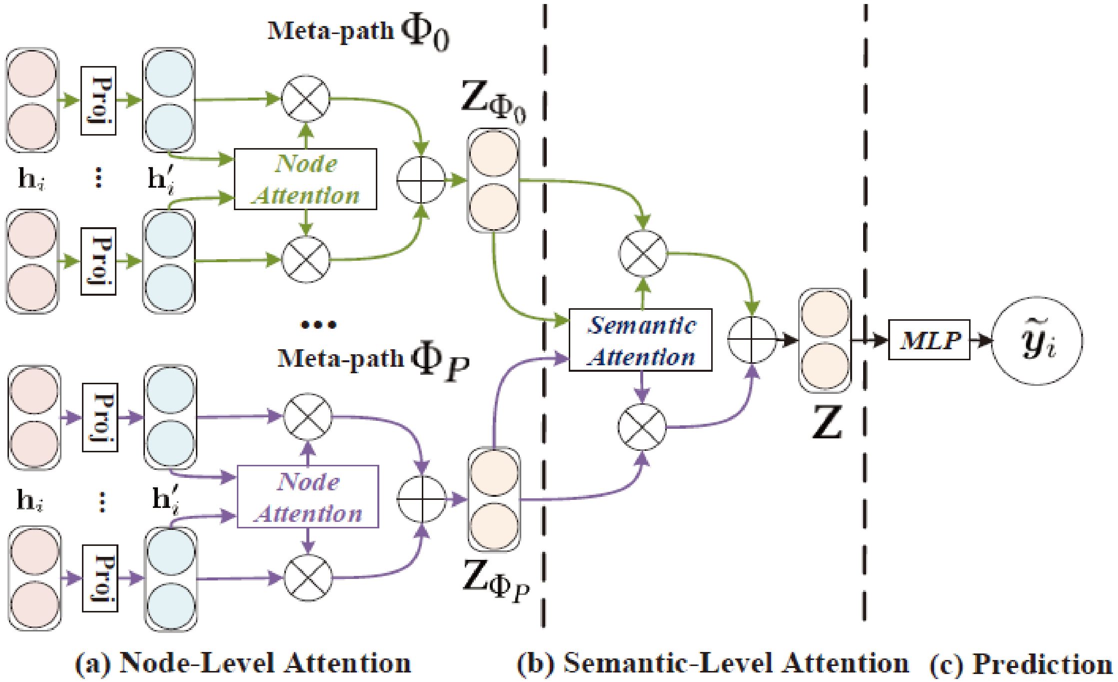
Consider the information of edges and nodes at the same time

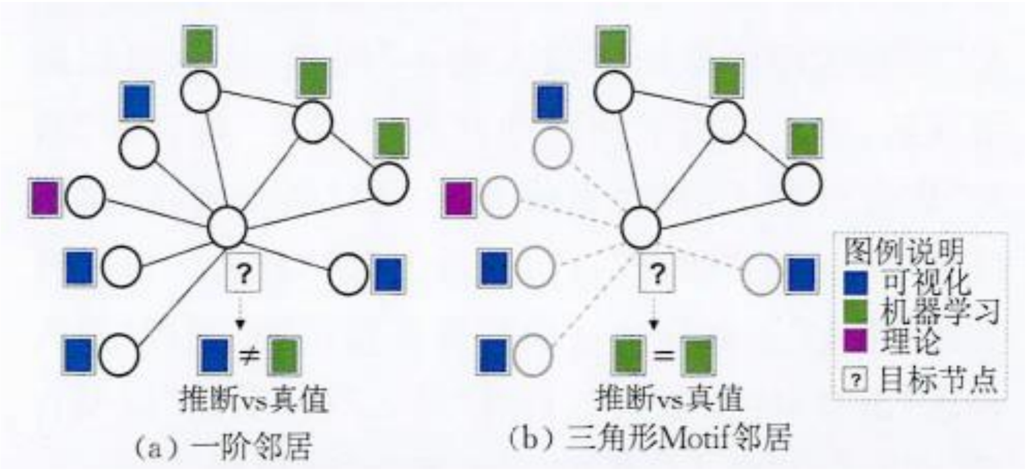


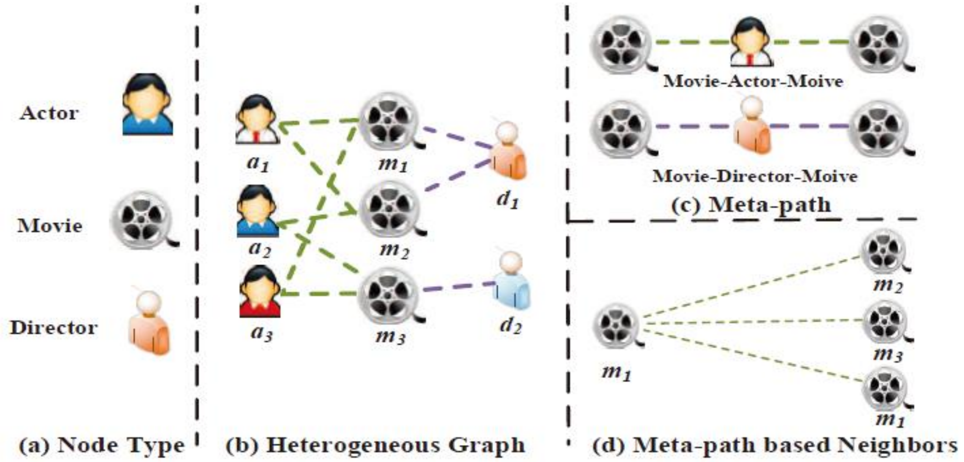
Gong L, Cheng Q. Exploiting edge features for graph neural networks[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2019: 9211-9219.

HAN

- 高阶信息的重要性
- 通过meta-path来显示定义不同阶信息



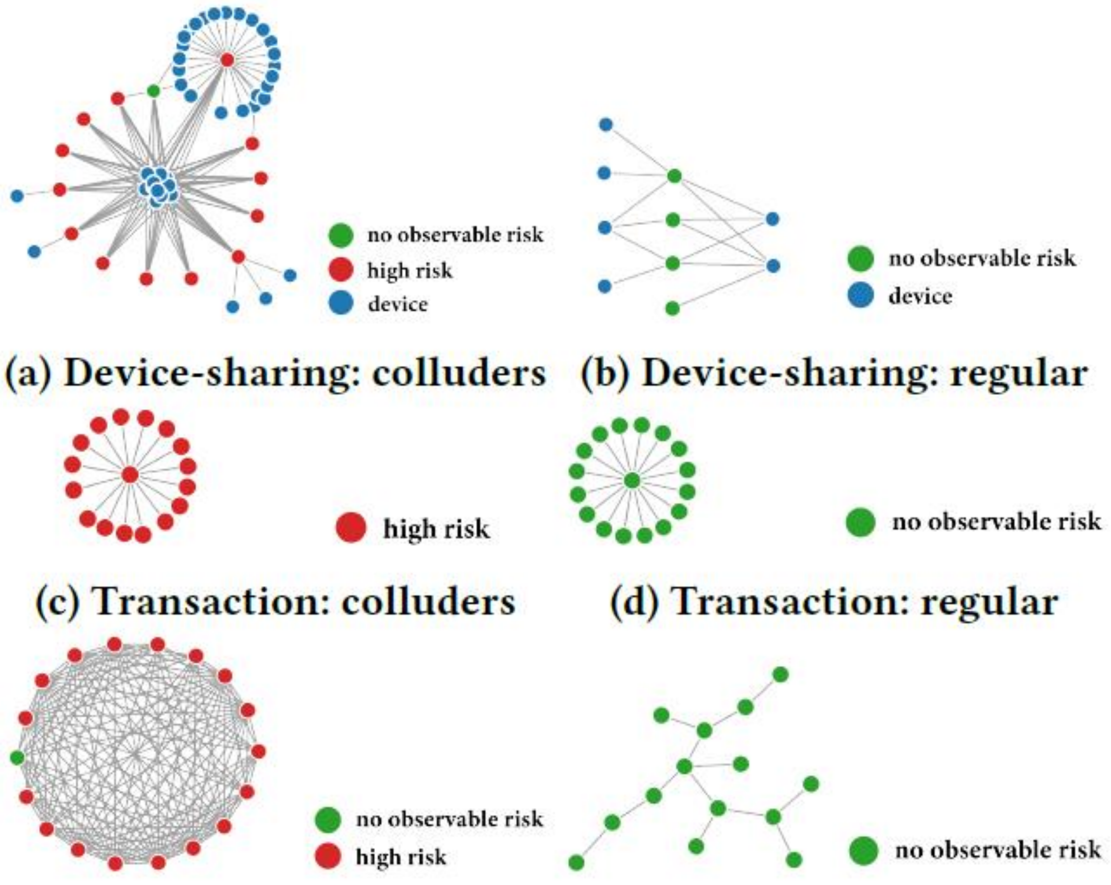




Wang X, Ji H, Shi C, et al. Heterogeneous graph attention network[C]//The World Wide Web Conference. 2019: 2022-2032.

3. 图神经网络在反欺诈领域的应用

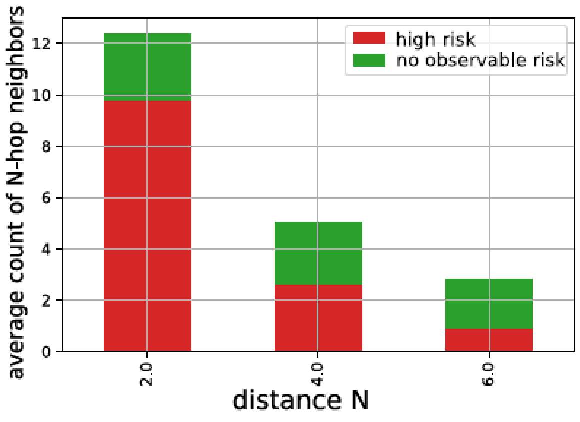
Who Stole the Postage? Fraud Detection in Return-Freight



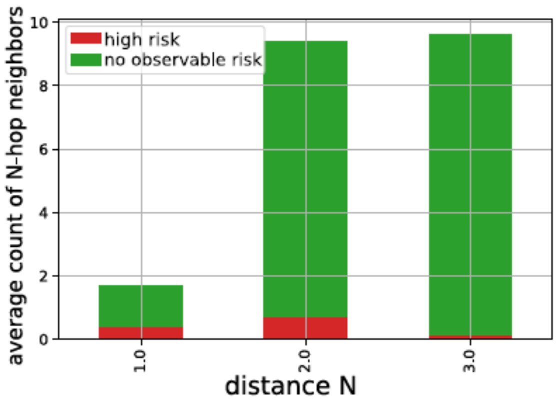
(a) Device-sharing: colluders (b) Device-sharing: regular

(c) Transaction: colluders (d) Transaction: regular

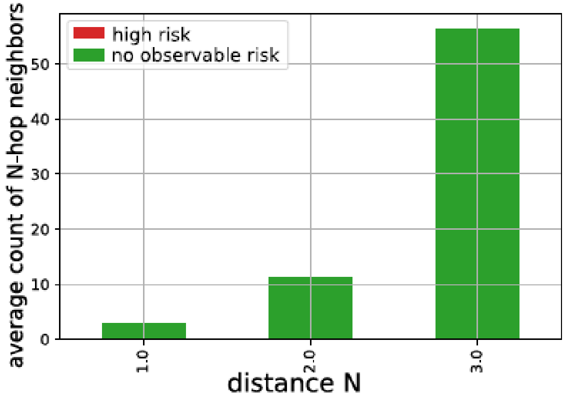
(e) Friendship: colluders (f) Friendship: regular



(a) Device-sharing



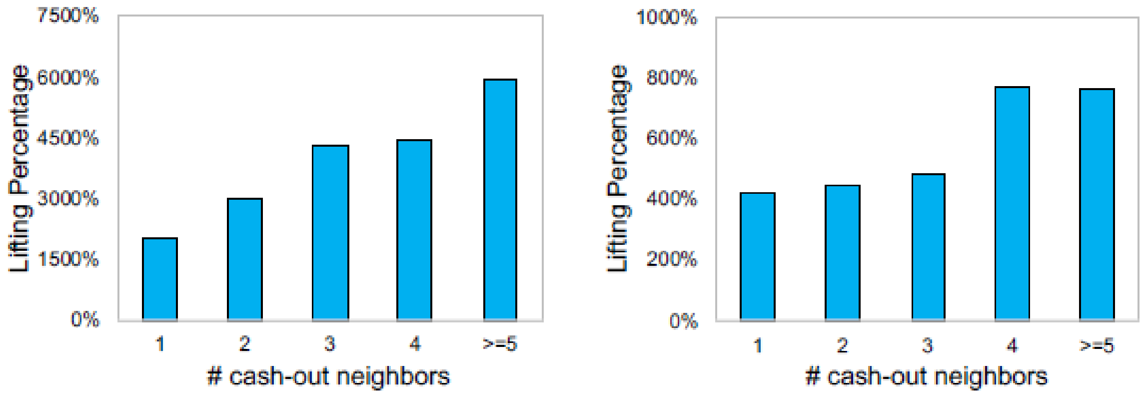
(b) Transaction



(c) Friendship

Liang C, Liu Z, Liu B, et al. Who Stole the Postage? Fraud Detection in Return-Freight Insurance Claims[J].

HACUD



(a) UMU (b) UU

Figure 3: The lifting percentages of cash-out rate in users with different amount of cash-out neighbors against users without any cash-out neighbor in two meta-paths.

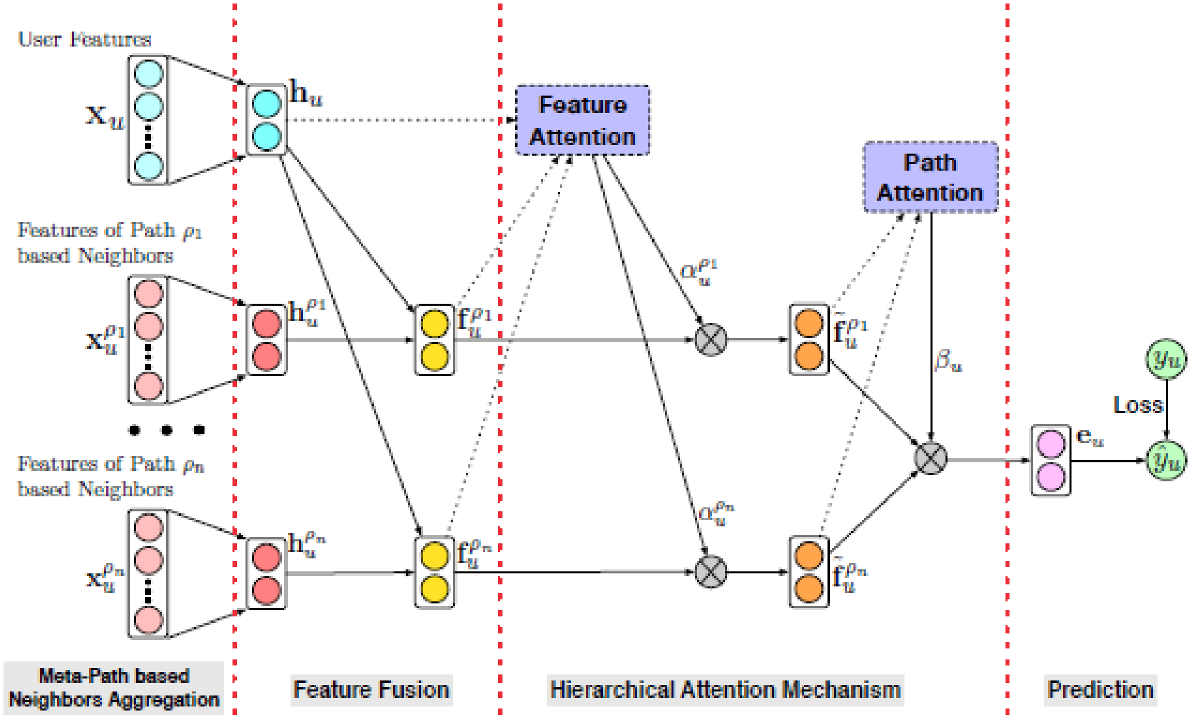


Figure 2: The architecture of the proposed model

- 邻居信息的重要性
- 不同邻居信息的重要性不同

Hu B, Zhang Z, Shi C, et al. Cash-out user detection based on attributed heterogeneous information network with a hierarchical attention mechanism[C]//Proceedings of the AAAI Conference on Artificial Intelligence. 2019, 33: 946-953.

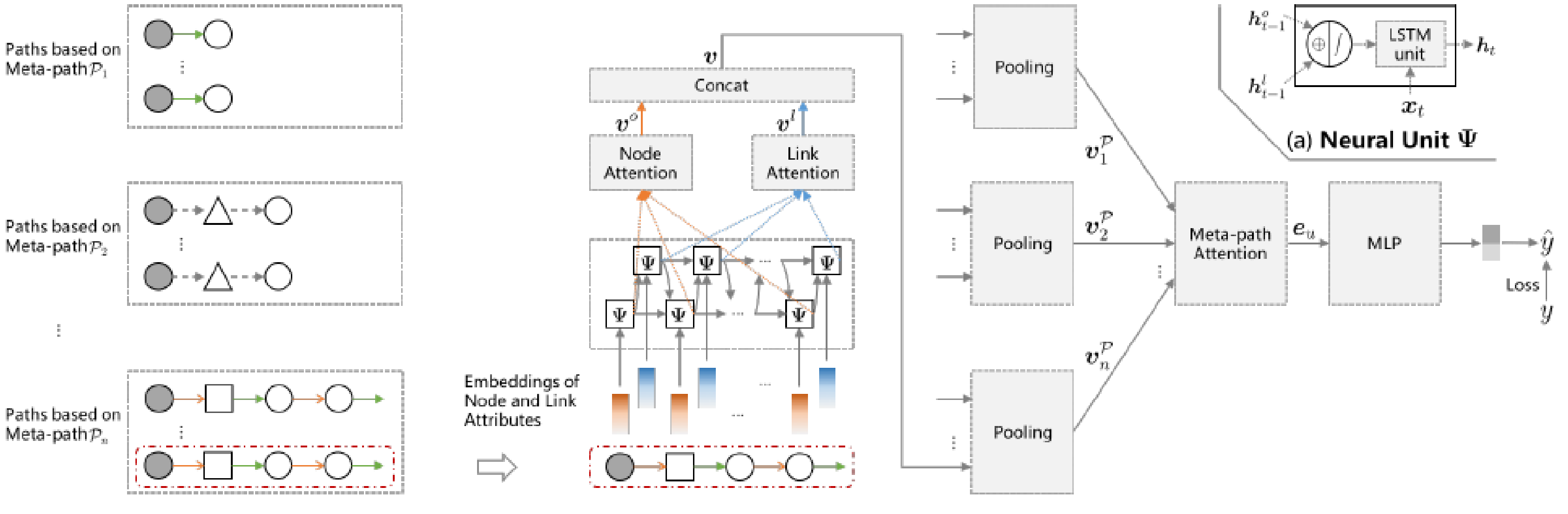
MAHINDER

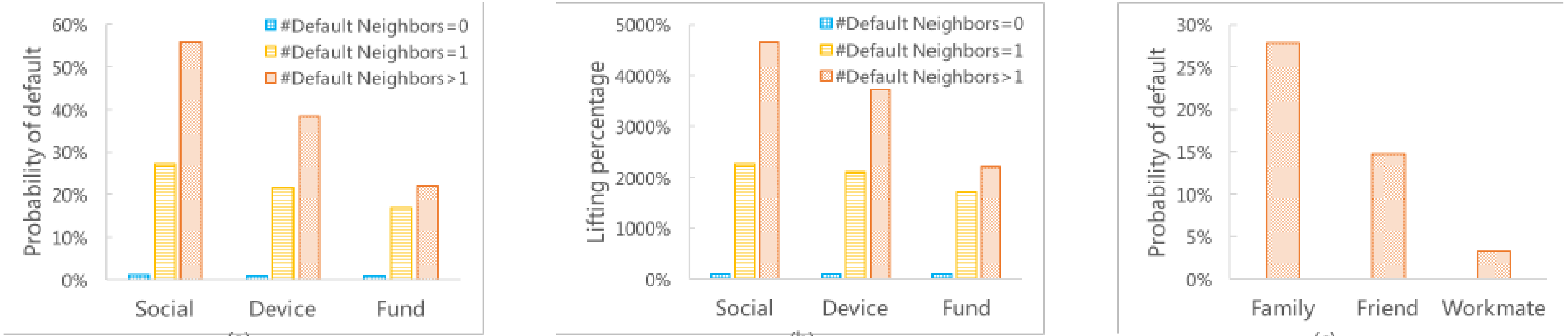
(1) intra-view meta-paths:

- UsU: $User \xrightarrow{social} User$
- UdU: $User \xrightarrow{device} User$
- UfU: $User \xrightarrow{fund} User$
- UsUsU: $User \xrightarrow{social} User \xrightarrow{social} User$
- UfUfU: $User \xrightarrow{fund} User \xrightarrow{fund} User$

(2) cross-view meta-paths:

- UdUsU: $User \xrightarrow{device} User \xrightarrow{social} User$
- UfUsU: $User \xrightarrow{fund} User \xrightarrow{social} User$
- UfUsUfU: $User \xrightarrow{fund} User \xrightarrow{social} User \xrightarrow{fund} User$





(a) (b) (c)

Zhong Q, Liu Y, Ao X, et al. Financial Defaulter Detection on Online Credit Payment via Multi-view Attributed Heterogeneous Information Network[C]//Proceedings of The Web Conference 2020. 2020: 785-795.

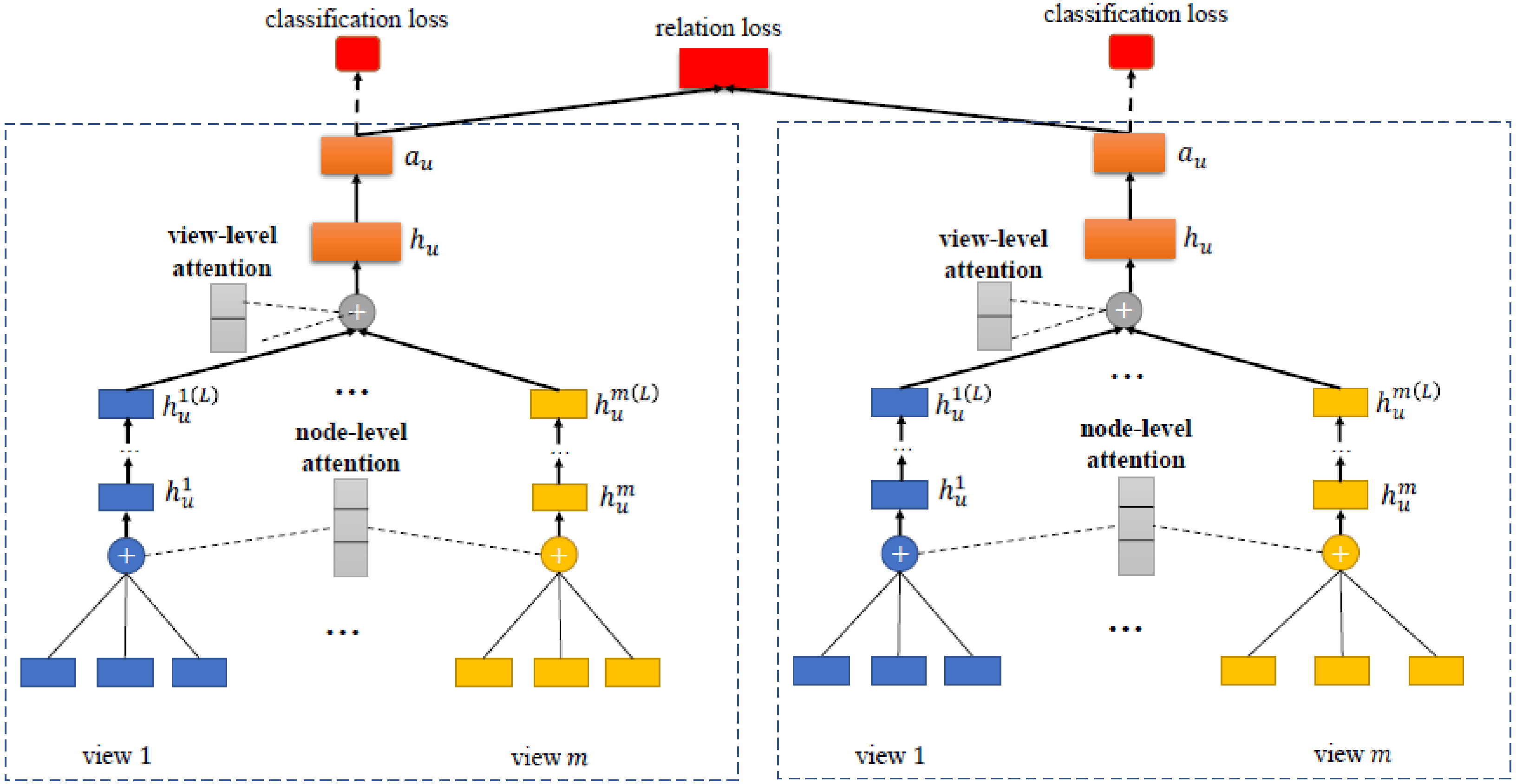
SemiGNN

□半监督欺诈检测

$$\mathcal{L}_{sup} = -\frac{1}{|U_L|}\sum_{u \in U_l}\sum_{i=1}^{k} I(y_u = i)\ log\frac{exp(a_u \cdot \theta_i)}{\sum_{j=1}^{k} exp(a_u \cdot \theta_j)},$$

$$\mathcal{L}_{graph} = \sum_{u \in U}\sum_{v \in \mathcal{N}_u \cup Neg_u} -log(\sigma(a_u^T a_v)) - Q \cdot E_{q \sim P_{neg}(u)} log(\sigma(a_u^T a_q)),$$

$$\mathcal{L}_{SemiGNN} = \alpha \cdot \mathcal{L}_{sup} + (1-\alpha) \cdot \mathcal{L}_{graph} + \lambda \mathcal{L}_{reg},$$



Wang D, Lin J, Cui P, et al. A Semi-supervised Graph Attentive Network for Financial Fraud Detection[C]//2019 IEEE International Conference on Data Mining (ICDM). IEEE, 2019: 598-607.

GraphConsis

- Context inconsistency

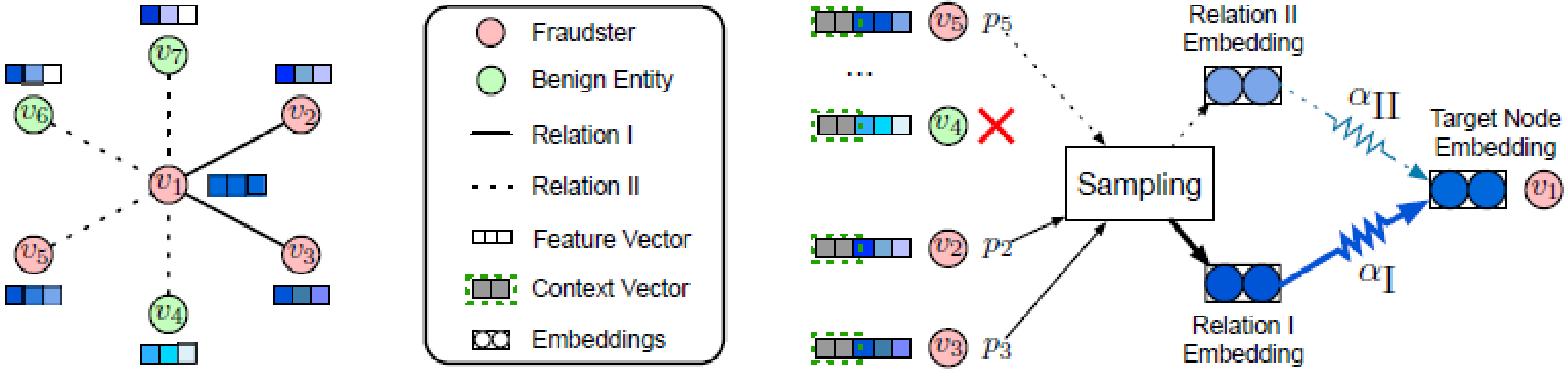
$$\mathbf{h}_v^{(1)} = \{\mathbf{x}_v \| \mathbf{c}_v\} \oplus \mathrm{AGG}^{(1)}\left(\{\mathbf{x}_{v'} \| \mathbf{c}_{v'} : v' \in \mathcal{N}_v\}\right)$$

- Feature inconsistency

$$s^{(l)}(u,v) = \exp\left(-\|\mathbf{h}_u^{(l)} - \mathbf{h}_v^{(l)}\|_2^2\right), \qquad p^{(l)}(u;v) = s^{(l)}(u,v) / \sum_{u \in \tilde{\mathcal{N}}_v} s^{(l)}(u,v).$$

- Relation inconsistency

$$\alpha_q^{(l)} = \exp\left(\sigma\left(\{\mathbf{h}_q^{(l)} \| \mathbf{t}_{r_q}\}\mathbf{a}^\top\right)\right) / \sum_{q=1}^{Q} \exp\left(\sigma\left(\{\mathbf{h}_q^{(l)} \| \mathbf{t}_{r_q}\}\mathbf{a}^\top\right)\right), \quad \mathrm{AGG}^{(l)}\left(\{\mathbf{h}_q^{(l-1)}\}\Big|_{q=1}^{Q}\right) = \sum_{q=1}^{Q} \alpha_q^{(l)} \mathbf{h}_q^{(l)},$$

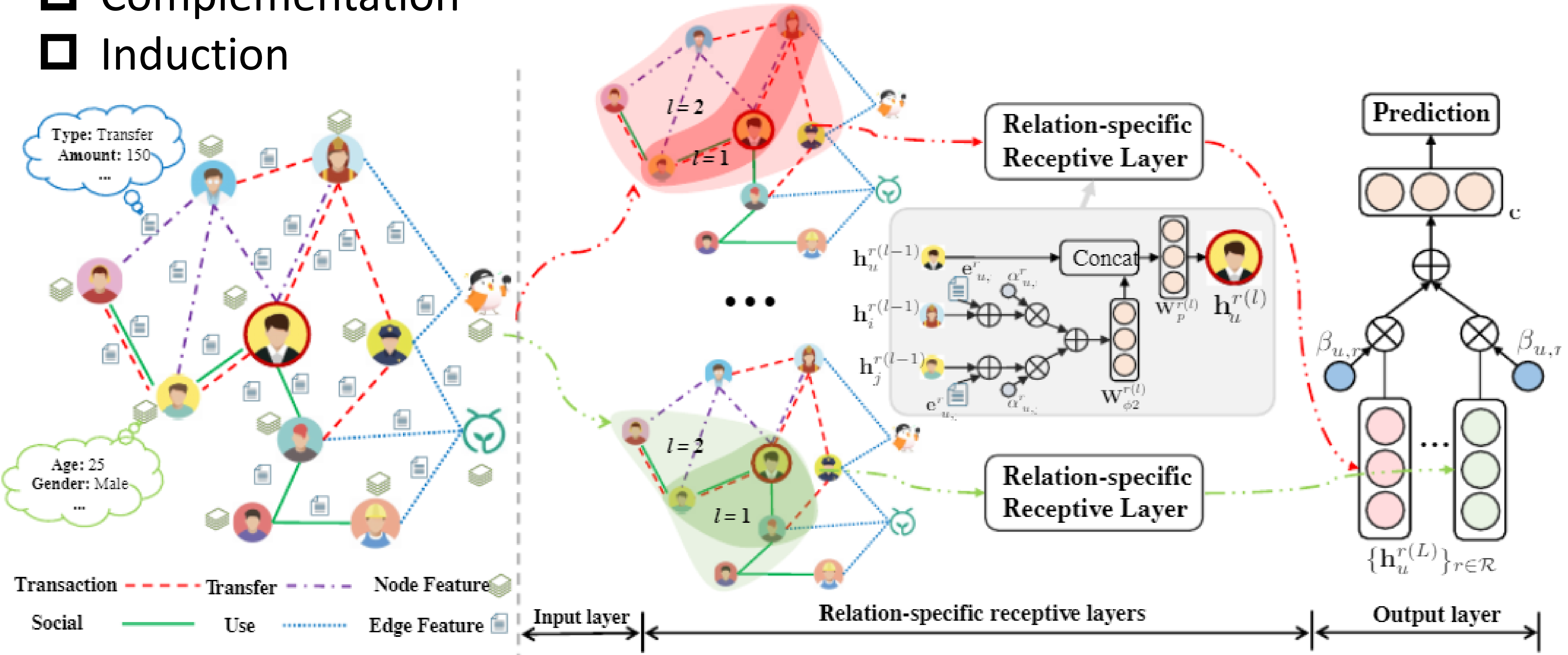


***Left*:** Inconsistency Problem ***Right*:** Proposed **GraphConsis** Model

Liu Z, Dou Y, Yu P S, et al. Alleviating the Inconsistency Problem of Applying Graph Neural Network to Fraud Detection[J]. arXiv preprint arXiv:2005.00625, 2020.

AMG

- ☐ Communicability
- ☐ Complementation
- ☐ Induction



(a)An example of AMG in financial scenario

(b)The architecture of the proposed model

Hu B, Zhang Z, Zhou J, et al. Loan Default Analysis with Multiplex Graph Learning[J]. institutions, 11: 25.