THE UNIVERSITY OF QUEENSLAND
AUSTRALIA

Week 9's Tutorial

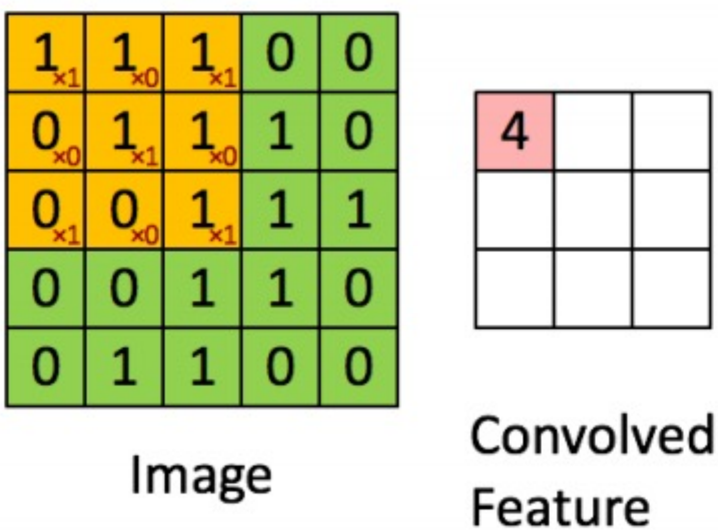
Graph Convolutional Networks and Social Recommender System

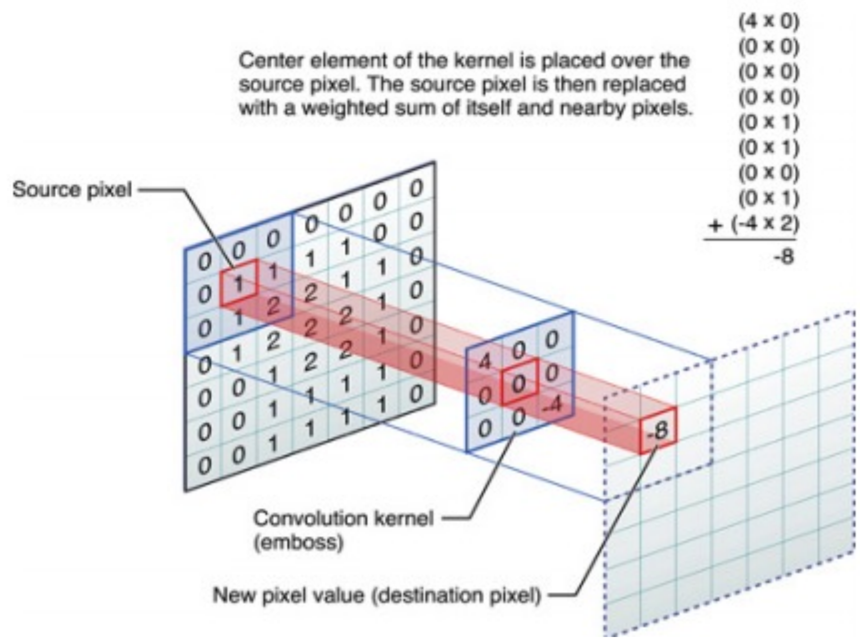
Speaker: Xin Xia
Course: INFS7450

1 Graph Convolutional Networks

CNN

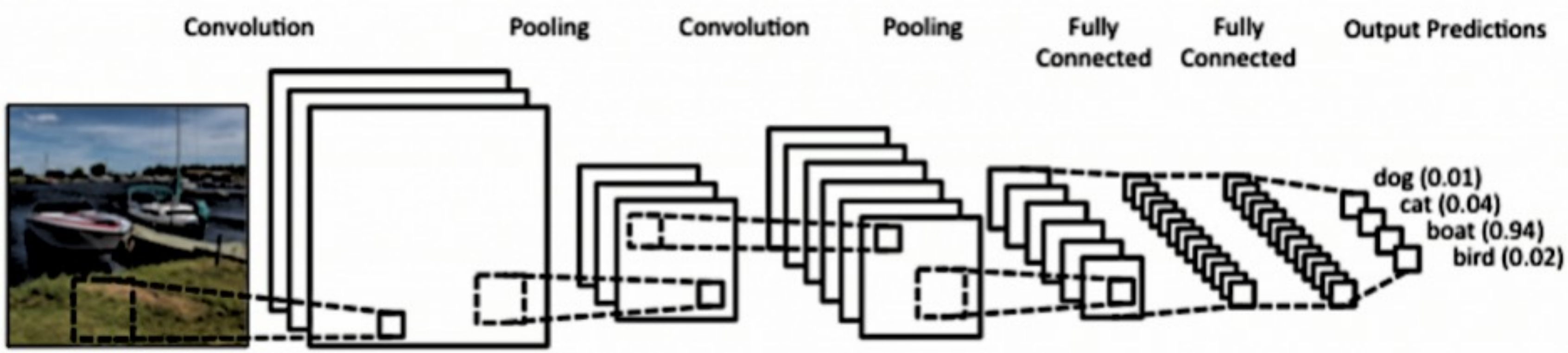
- Convolutional Neural Networks (CNN)
  - Operates on images - captures the spatial structure
  - Consists of learnable set of filters which perform 2D convolution on the image to get activation map





CNN

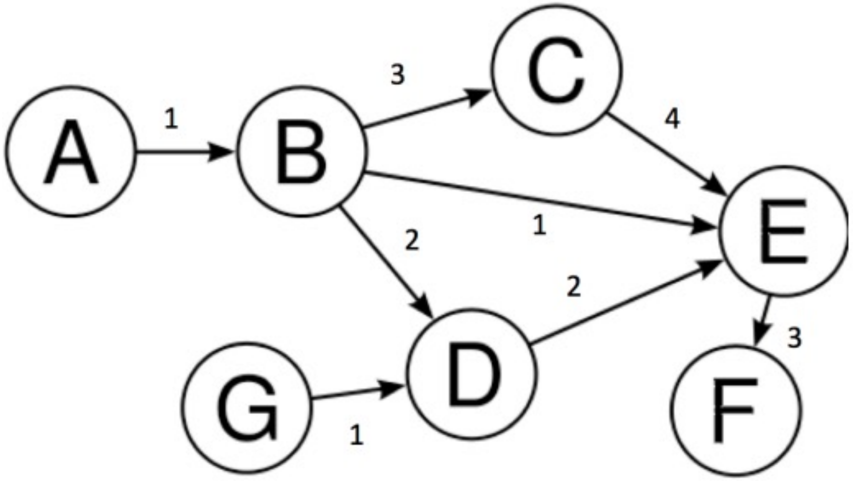
- CNN is very powerful for image classification.



An example architecture of a CNN being used for classification

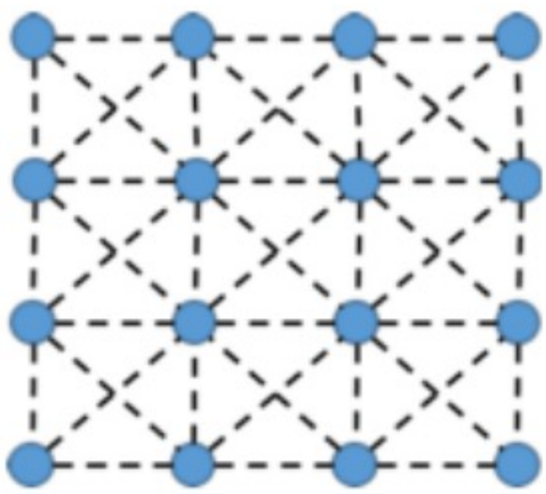
Drawbacks of CNN

- However, when it comes to graph data, CNN struggles.
- CNN is designed for data with Euclidean structures. But the structure of graph is arbitrary.



Drawbacks of CNN

- Image as a Graph



- Each pixel has 8 neighbors
- The node attributes are scalar values for grayscale image and 3-dimensional for RGB images
- The edge weights are binary (0 or 1), either present or absent

Why GCN

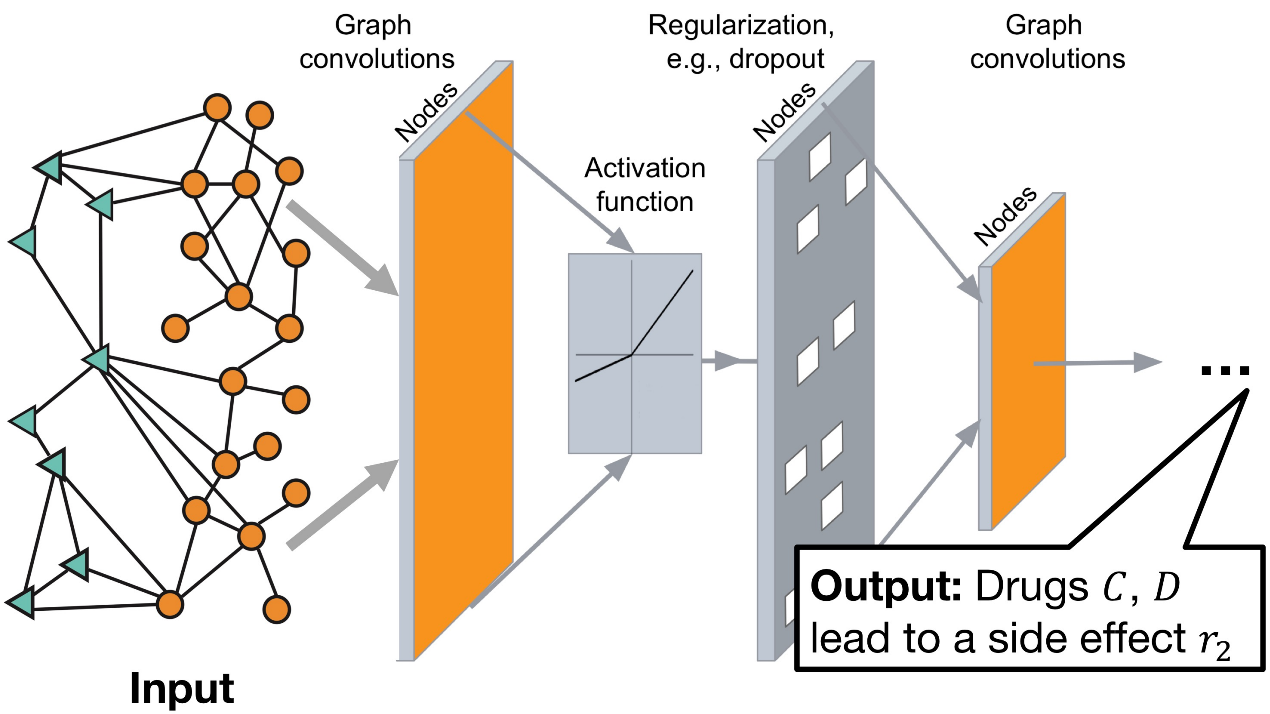
- Their arises many scenarios where the inherent structure of the data is a graph (e.g. social networks) and one has to learn from it. One can employ GCN for classification/clustering tasks!

What is GCN

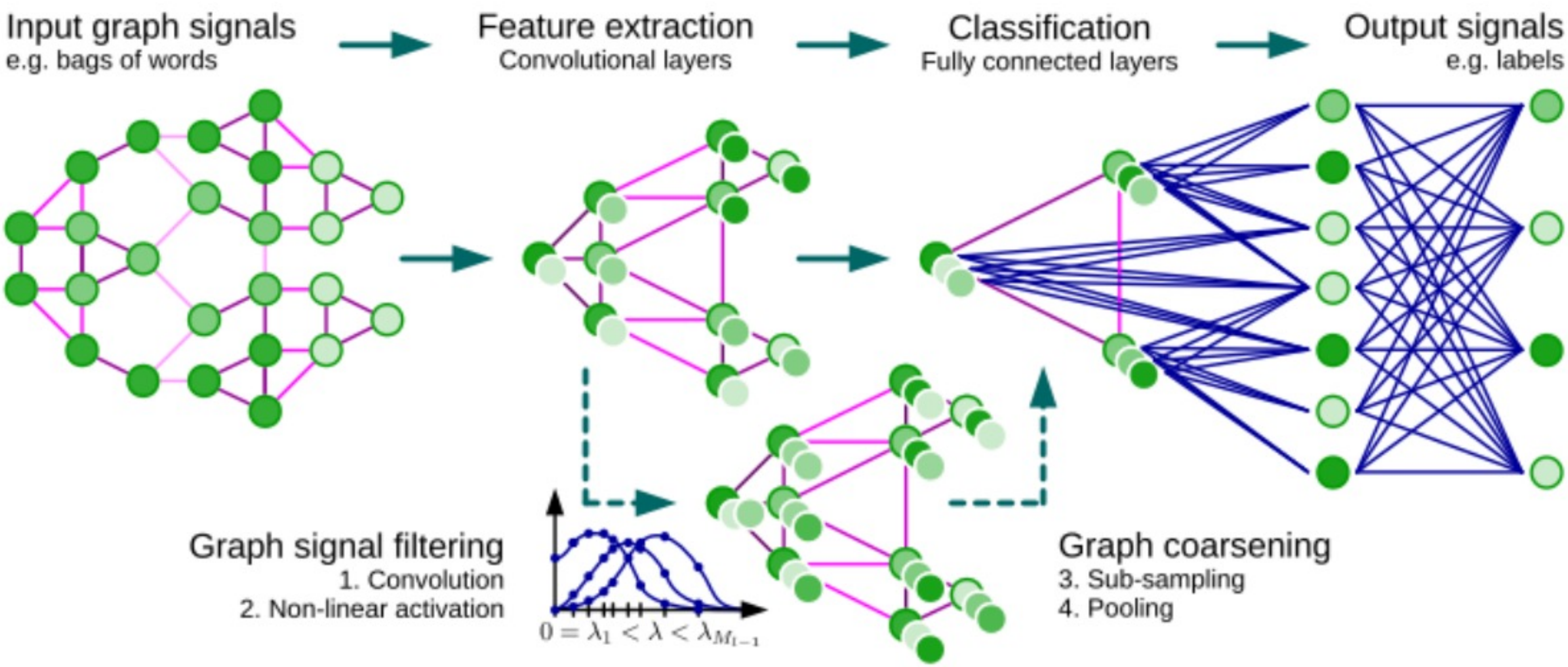
- Graph Convolutional Network is a type of Neural Network which directly operates on the graph structure. A typical application of GNN is node classification. Essentially, every node in the graph is associated with a label, and we want to predict the label of the nodes without ground-truth.

What is GCN

- The architecture is similar to a traditional CNN but it takes graphs as input, also the convolution and pooling operations are different in principle.

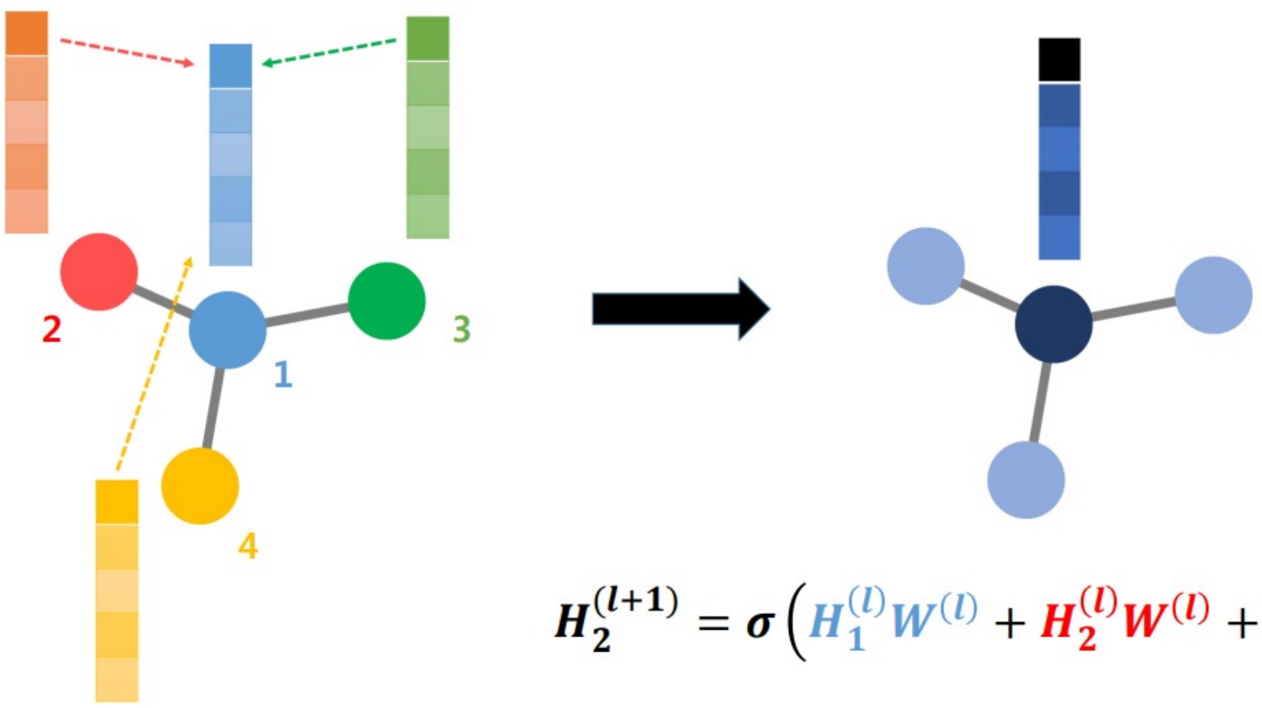


CNN vs GCN


Input graph signals
e.g. bags of words
Feature extraction
Convolutional layers
Classification
Fully connected layers
Output signals
e.g. labels
Graph signal filtering
1. Convolution
2. Non-linear activation
$0 = \lambda_1 < \lambda < \lambda_{M_{l-1}}$
Graph coarsening
3. Sub-sampling
4. Pooling

How GCN works

$W^{(l)}$: The filter to learn
$H^{(l)}$: The representations of nodes in layer *l*

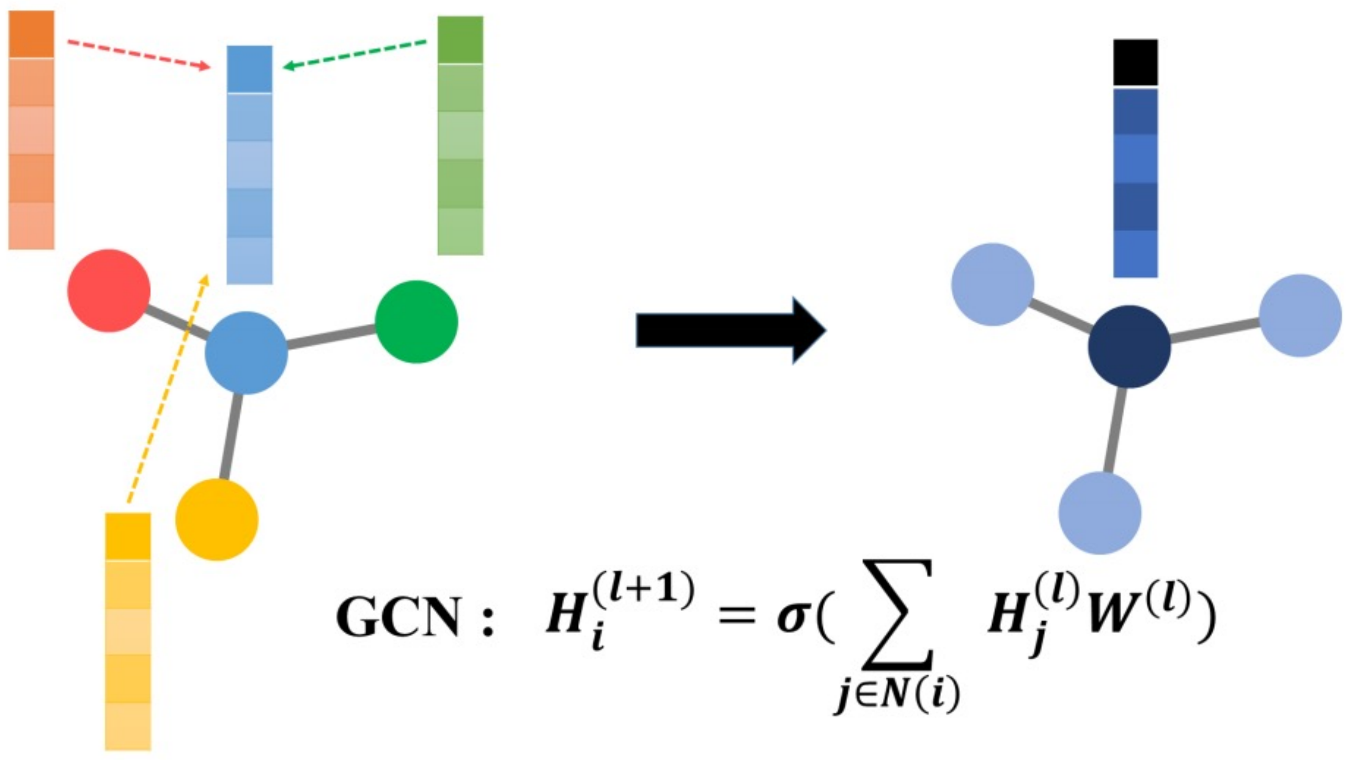


$$H_2^{(l+1)} = \sigma\left(H_1^{(l)}W^{(l)} + H_2^{(l)}W^{(l)} + H_3^{(l)}W^{(l)} + H_4^{(l)}W^{(l)}\right)$$

Aggregate information from the neighbors

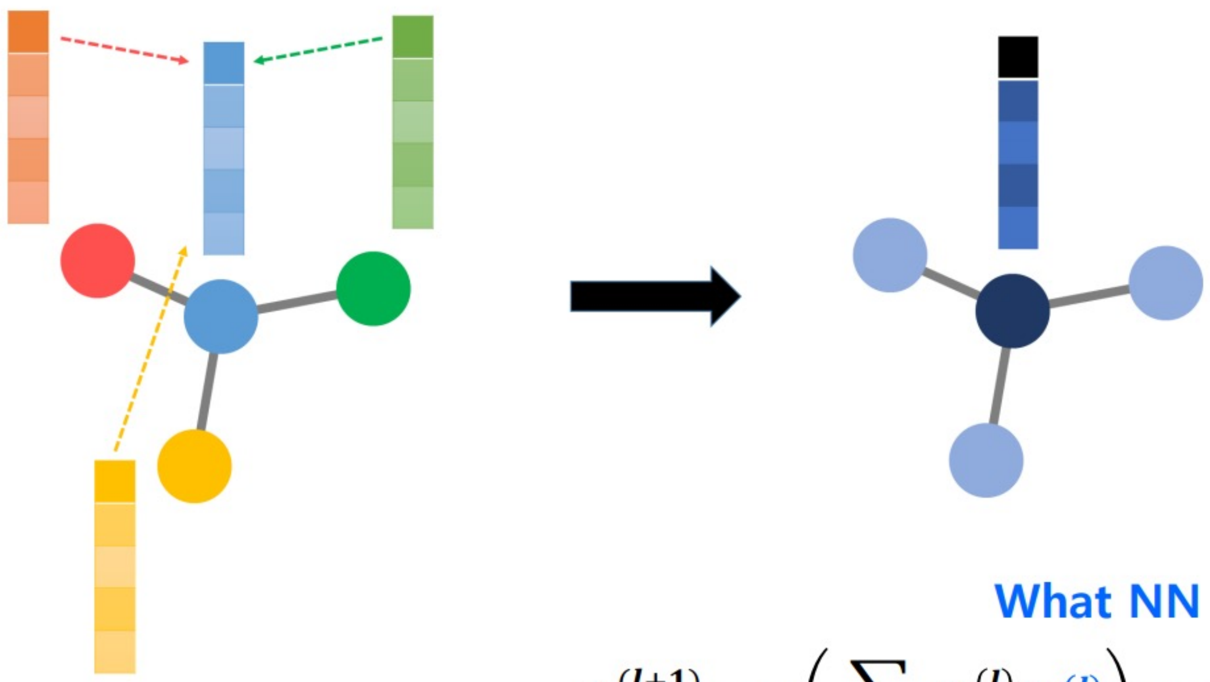
How GCN works

$W^{(l)}$: The filter to learn
$H^{(l)}$: The representations of nodes in layer *l*



$$\mathbf{GCN:} \quad H_i^{(l+1)} = \sigma\Big(\sum_{j \in N(i)} H_j^{(l)} W^{(l)}\Big)$$

How GCN works



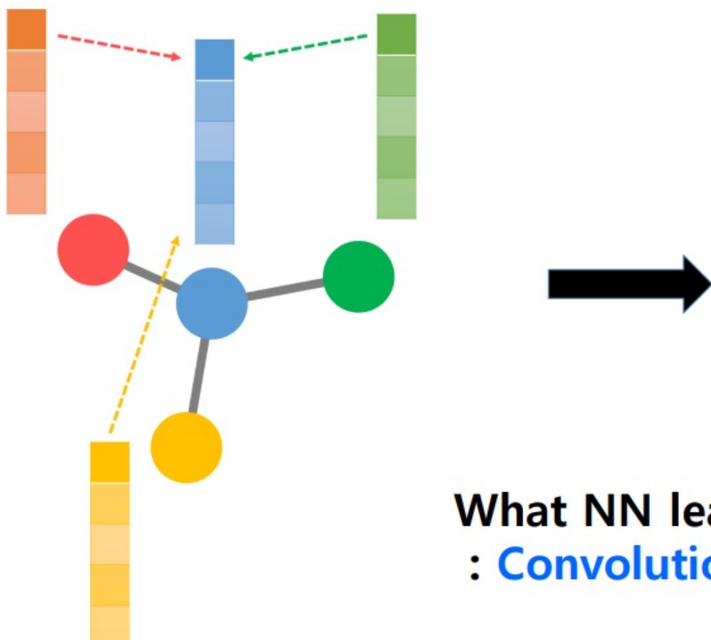
What NN learns

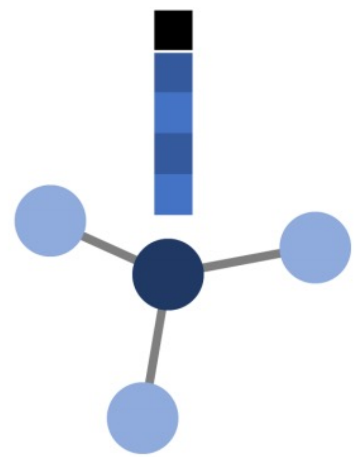
$$H_i^{(l+1)} = \sigma\left(\sum_{j \in N(i)} H_j^{(l)} W^{(l)}\right) = \sigma\left(A H_j^{(l)} W^{(l)}\right)$$

$$H^{(l+1)} = \sigma\left(\widetilde{D}^{-1/2}\tilde{A}\widetilde{D}^{-1/2}H^{(l)}W^{(l)}\right) \qquad \tilde{A} = A + I_N, \ \ \widetilde{D}_{ii} = \sum_j \tilde{A}_{ij}$$

How GCN works

A common variant of GCN: Graph Attention Networks (GAT)





What NN learns

: Convolution weight and attention coefficient

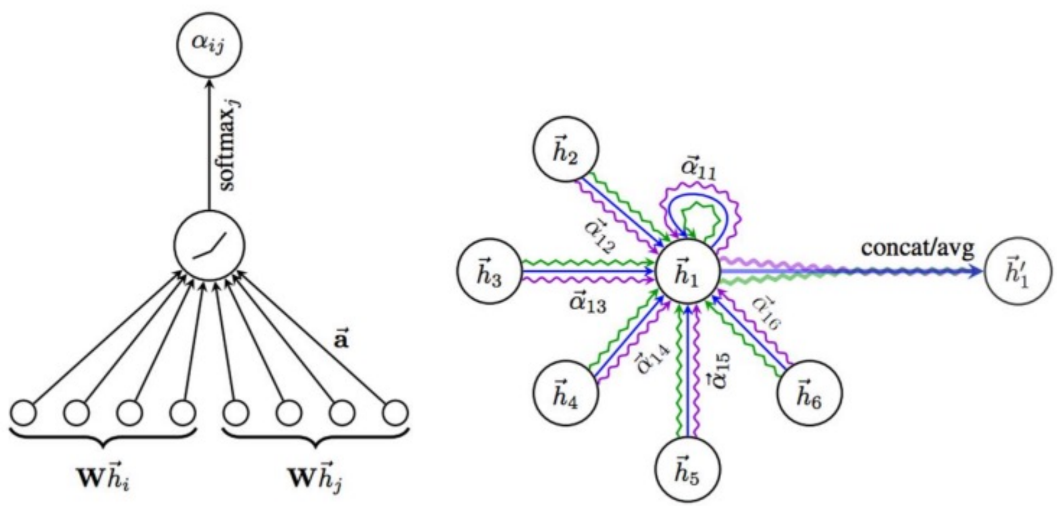
GAT : $$H_i^{(l+1)} = \sigma\left(\sum_{j \in N(i)} \alpha_{ij}^{(l)} H_j^{(l)} W^{(l)}\right)$$

How GCN works

A common variant of GCN: Graph Attention Networks (GAT)

What NN learns
: Convolution weight and attention coefficient

$$H_i^{(l+1)} = \sigma\left(\sum_{j\in N(i)} \alpha_{ij}^{(l)} H_j^{(l)} W^{(l)}\right) \qquad \alpha_{ij} = f(H_i W, H_j W)$$



How GCN works

A common variant of GCN: Graph Attention Networks (GAT)

What NN learns
: Convolution weight and attention coefficient

$$H_i^{(l+1)} = \sigma\left(\sum_{j \in N(i)} \alpha_{ij}^{(l)} H_j^{(l)} W^{(l)}\right) \qquad \alpha_{ij} = f(H_i W, H_j W)$$

- Velickovic, Petar, et al. – network analysis

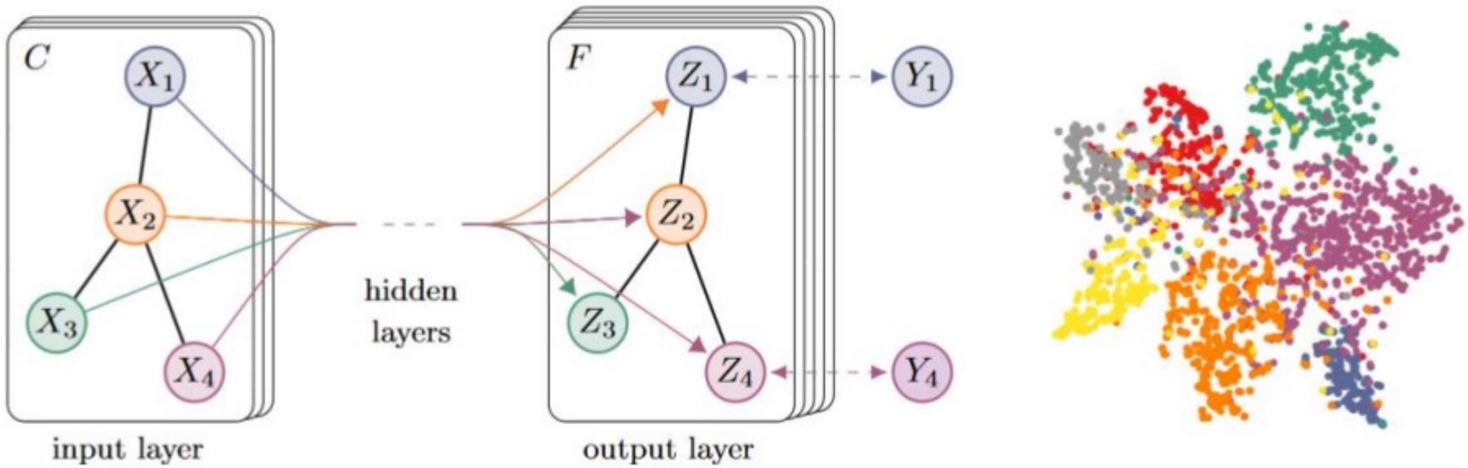
$$\alpha_{ij} = softmax(e_{ij}) = \frac{e_{ij}}{exp(\sum_{k \in N(i)} e_{ik})} \qquad e_{ij} = LearkyReLU(a^T[H_i W, H_j W])$$

- Seongok Ryu, et al. – molecular applications

$$\alpha_{ij} = \tanh\left((H_i W)^T C(H_j W)\right)$$

Application – Node classification

- Good node features → Good node classification results



(a) Graph Convolutional Network (b) Hidden layer activations

Figure 1: *Left*: Schematic depiction of multi-layer Graph Convolutional Network (GCN) for semi-supervised learning with C input channels and F feature maps in the output layer. The graph structure (edges shown as black lines) is shared over layers, labels are denoted by Y_i. *Right*: t-SNE (Maaten & Hinton, 2008) visualization of hidden layer activations of a two-layer GCN trained on the Cora dataset (Sen et al., 2008) using 5% of labels. Colors denote document class.

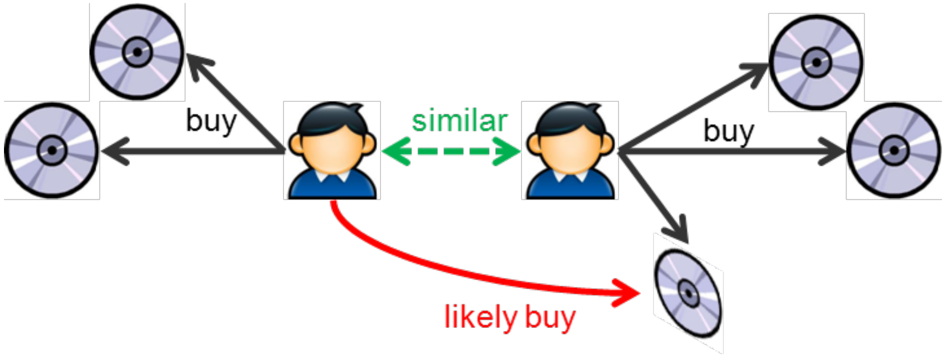
2 Social Recommender System

Introduction - RS

- Recommender systems (RS) play an important role in our lives.
- Recommender systems are algorithms aimed at suggesting relevant items to users.

Introduction - RS

- Collaborative filtering is the mainstream paradigm of recommender systems.
- The main idea that rules collaborative methods is that **past user-item interactions are sufficient to detect similar users and/or similar items and make predictions based on these estimated proximities**.



Introduction - RS

- However, recommender systems are often compromised by the problem of data sparsity.
- For the users with few interactions, it is difficult to generate good recommendations.

	Item 1	Item 2	...	Item 99	Item 100
Customer 1	5	NA	...	NA	3
Customer 2	NA	2	...	3	NA
...	...	...	...	...	...
Customer 49	2	3	...	NA	4
Customer 50	1	NA	...	NA	NA

Introduction - RS

- To alleviate data sparsity, side information should be incorporated into recommender systems to help infer user preference.
- With the popularity of social platforms, social networks may be the antidote to data sparsity.

Introduction - Social Networks

- Homophily
  - The tendency for people to have ties with people who are similar to themselves in socially significant ways.
- Social Influence
  - The change in an individual's thoughts, feelings, attitudes, or behaviors that results from interaction with another individual or a group.

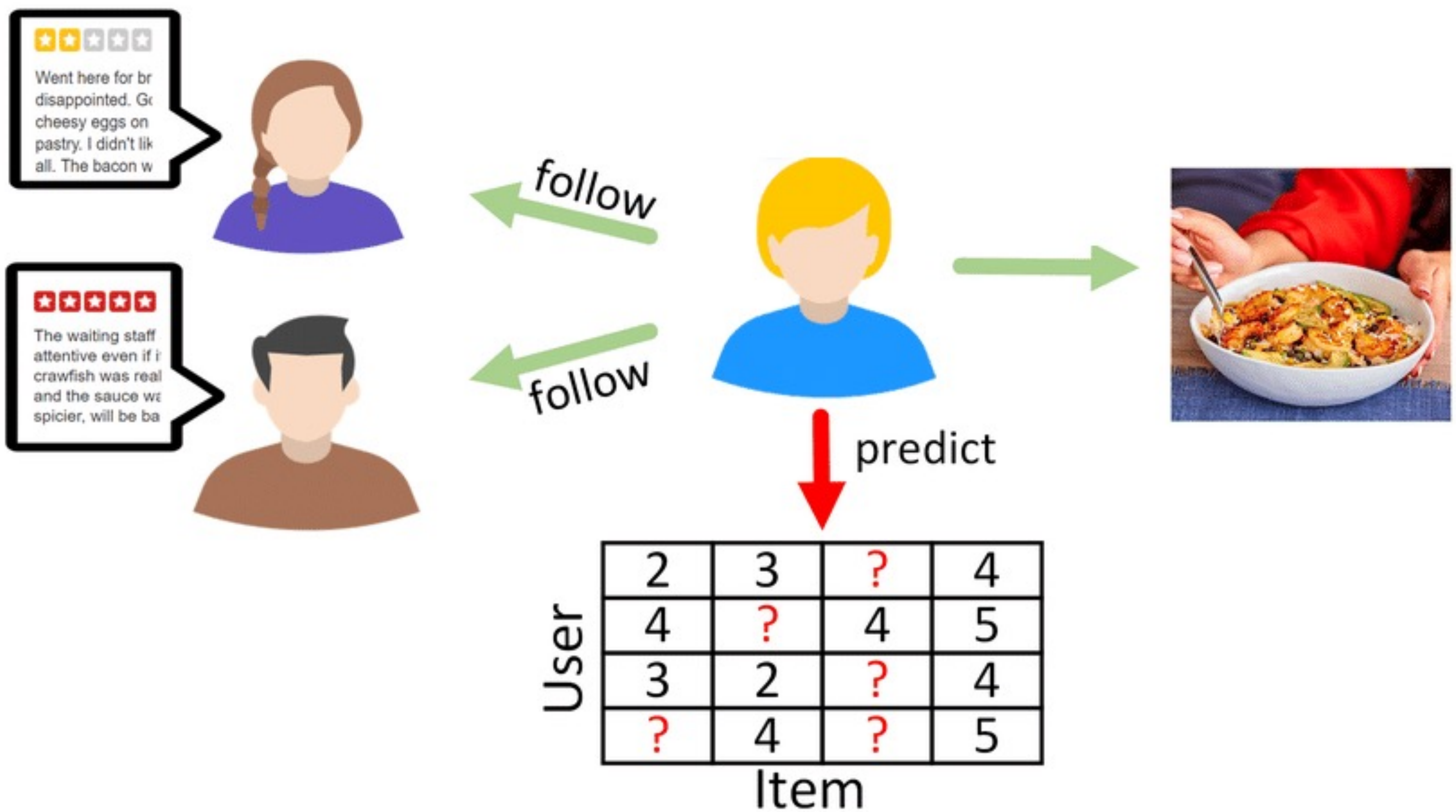
Introduction - Social Networks

- Because of homophily and social influence, we can infer users' preferences according to their friends' behaviors, which is the basis of social recommender systems (SRS).

- **Definition of social recommendation**
  - is any recommendation with online social relations as an additional input, i.e., augmenting an existing recommendation engine with additional social signals. Social relations can be trust relations, friendships, memberships or following relations.

Introduction - SRS

- How social recommender systems work?



SRS Models

- Two main categories:

SRS

Model based

Define a model for user-item interactions where users and items representations have to be learned from interactions matrix.

Memory based

Define no model for user-item interactions and rely on similarities between users or items in terms of observed interactions.

SRS Models – Memory Based

- Memory based approaches
  - Explore the social network for raters.
  - Aggregate the ratings to compute prediction.
  - Store the social rating network.
  - No learning phase.
  - Slow in prediction.

First generation of recommenders in SN were memory based approaches.

SRS Models – Model Based

- Model based approaches
  - Learn a model.
  - Store the model parameters only.
  - Substantial time for learning.
  - Fast in prediction.

SRS Models – MF Based

- Early studies are most based on matrix factorization.
  - Observed ratings $R_{u,i}$
  - Latent factors for users

$$U \in \mathbb{R}^{K \times N}$$

  - Latent factors for items

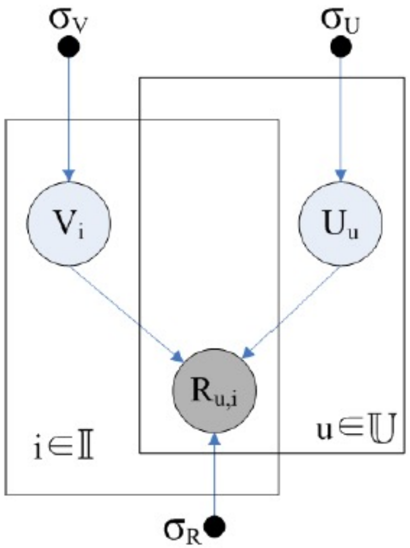
$$V \in \mathbb{R}^{K \times M}$$

$$p(R|U,V,\sigma_R^2) = \prod_{u=1}^{N} \prod_{i=1}^{M} \left[\mathcal{N}\left(R_{u,i}|U_u^T V_i, \sigma_r^2\right)\right]^{I_{u,i}^R}$$

Mnih, Andriy, and Russ R. Salakhutdinov. "Probabilistic matrix factorization." *Advances in neural information processing systems.* 2008.

SRS Models – MF Based

Graphical model



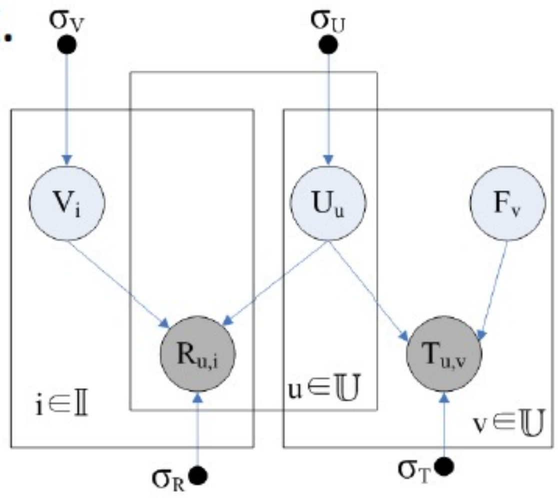
- Learn *U*, *V* that minimize

$$\sum_{all\ observed\ (u,i)} (R_{ui} - \hat{R}_{ui})^2 + \lambda(\|U\|^2 + \|V\|^2)$$

Mnih, Andriy, and Russ R. Salakhutdinov. "Probabilistic matrix factorization." *Advances in neural information processing systems.* 2008.

SRS Models – SoRec

- Matrix factorization model
  – Factorize the ratings and links together.
  – Social network as a binary matrix.
- One latent factor for items.
- Two latent factors for users:
  – One for the initiator,
  – One for the receiver.
- Same user factor for both contexts (rating actions and social actions).



. Ma, Hao, et al. "Sorec: social recommendation using probabilistic matrix factorization." CIKM 2008.

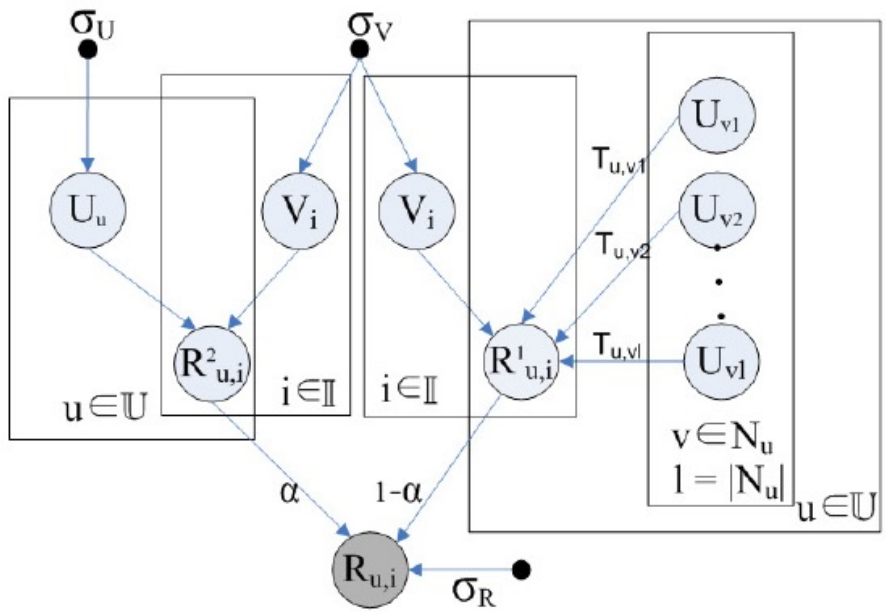
SRS Models – STE

- Social Trust Ensemble (STE)
- Linear combination of
  - Basic matrix factorization and
    Latent factors of the user and the item determine the observed rating.
  - Social network based approach
    Latent factors of the neighbors and the latent factor of the item determine the observed rating.

Ma, Hao, Irwin King, and Michael R. Lyu. "Learning to recommend with social trust ensemble." SIGIR. 2009.

SRS Models – STE

- Graphical model



$$\hat{R}_{u,i} = \alpha U_u^T V_i + (1-\alpha) \sum_{v \in N_u} T_{u,v} U_v^T V_i$$

Ma, Hao, Irwin King, and Michael R. Lyu. "Learning to recommend with social trust ensemble." SIGIR. 2009.

SRS Models – SocialMF

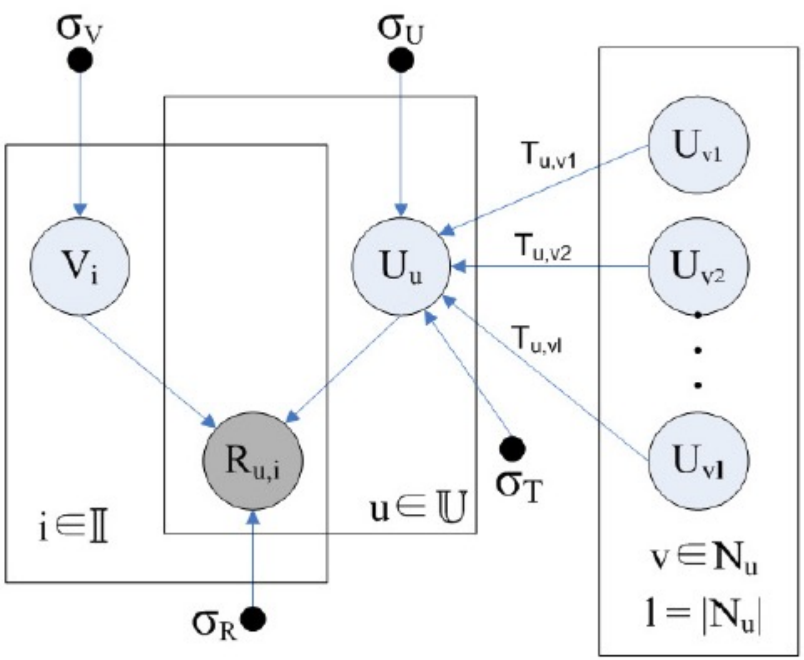
- Social Regularization Based
  - Social influence: behavior of a user u *is* affected by his direct neighbors N_u.
  - Latent factors of a user depend on those of his neighbors.

$$\widehat{U}_u = \sum_{v \in N_u} T_{u,v} U_v$$

  - $T_{u,v}$ is the normalized trust value.

Jamali, Mohsen et al. "A matrix factorization technique with trust propagation for recommendation in social networks." RecSys 2010

SRS Models – SocialMF



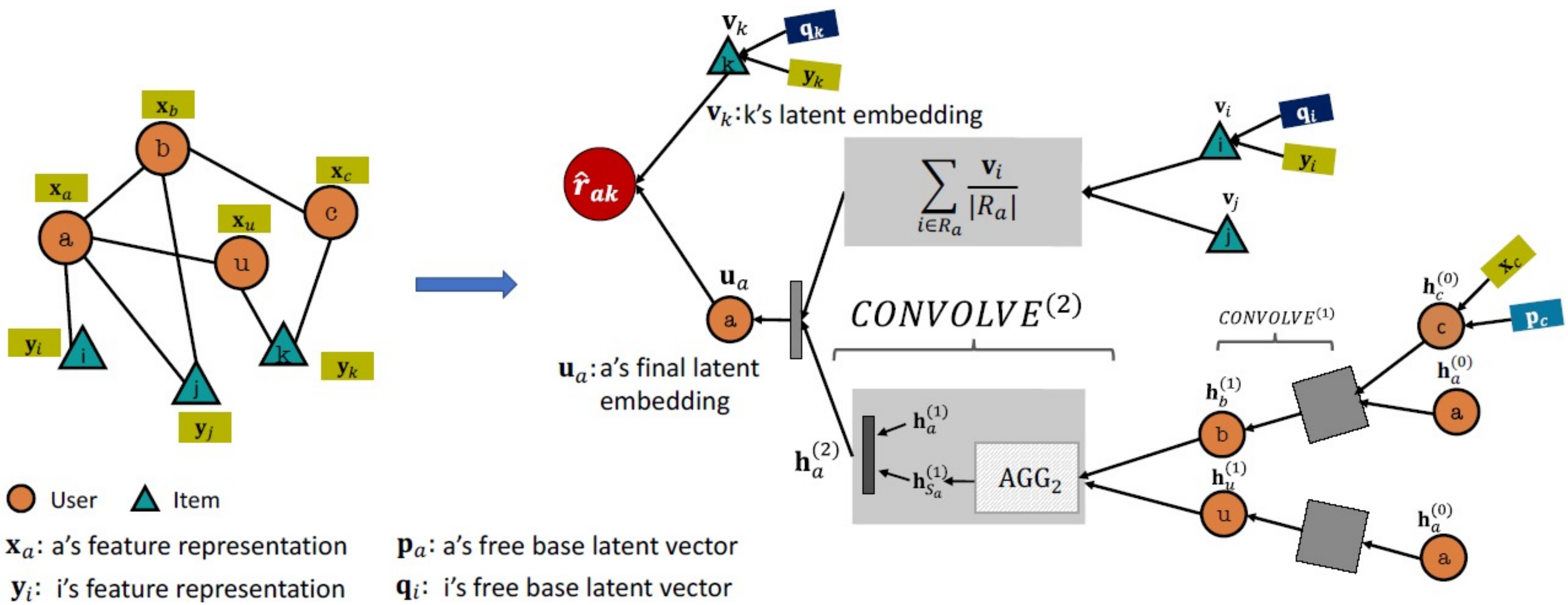
$$\sum_{all\ observed(u,i)} (R_{ui} - \widehat{R}_{ui})^2 + \lambda(\|U\|^2 + \|V\|^2)$$

$$+ \beta(\sum_{u}((U_u - \sum_{v} T_{u,v} U_v)(U_u - \sum_{v} T_{u,v} U_v)^T)$$

Jamali, Mohsen et al. "A matrix factorization technique with trust propagation for recommendation in social networks." RecSys 2010

SRS Models – Graph Based

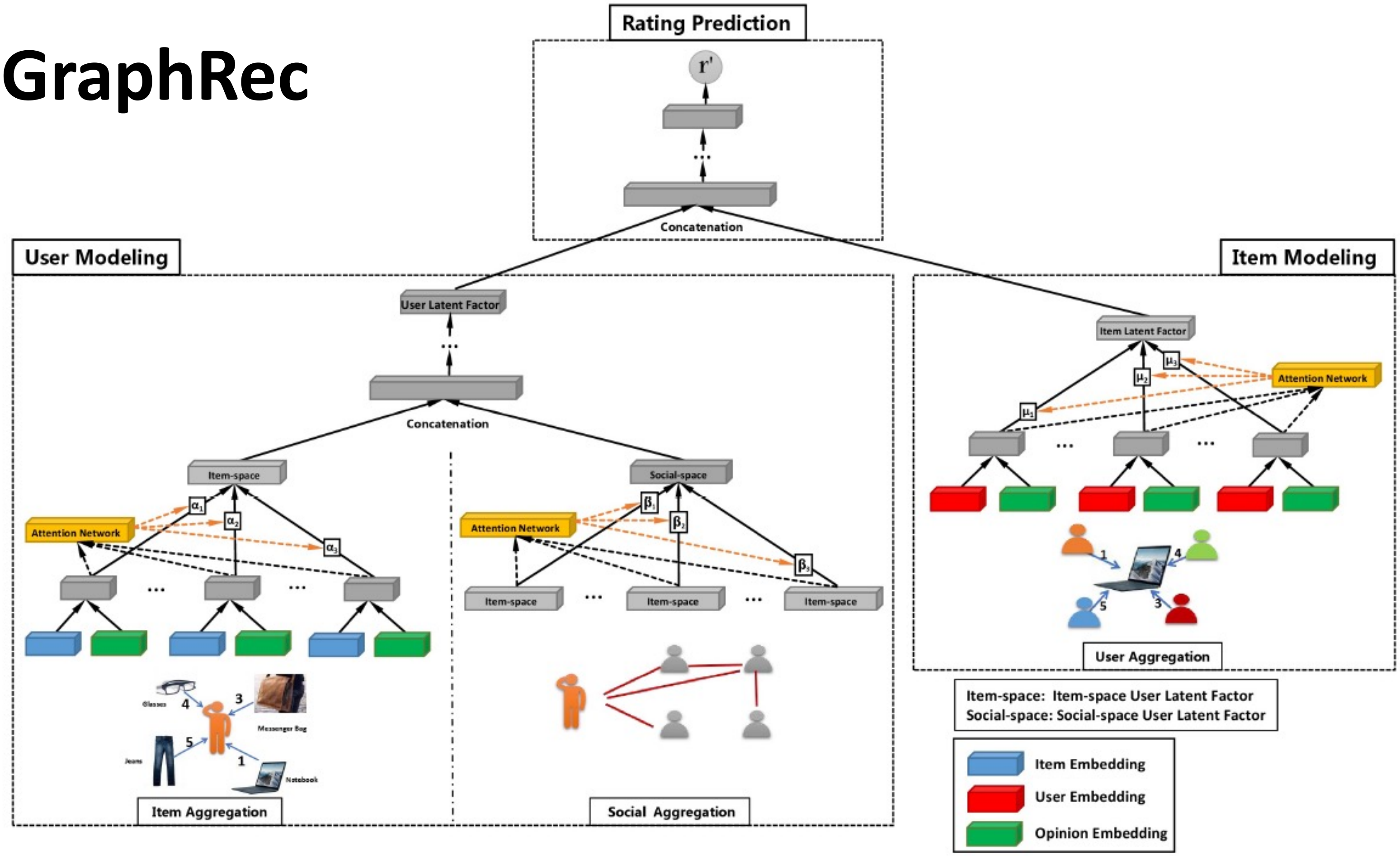
SocialGCN



Wu, Le, et al. "Socialgcn: An efficient graph convolutional network based model for social recommendation." arXiv preprint

SRS Models – Graph Based

GraphRec



Fan, et al. "Graph Neural Networks for Social Recommendation." WWW 2019

3 Problems in SRS

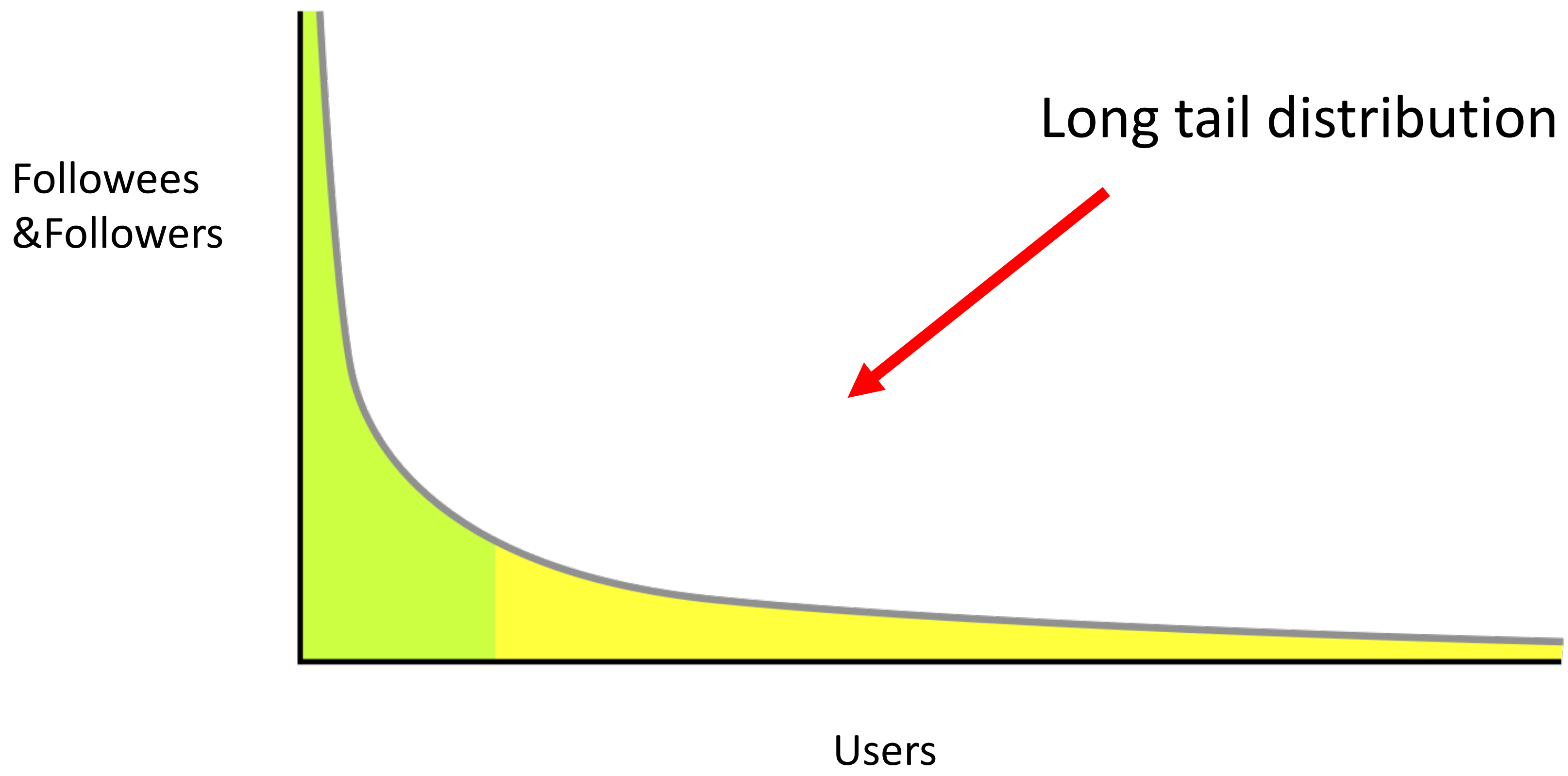
Problems in SRS

- Social Recommendation is not as successful as expected.

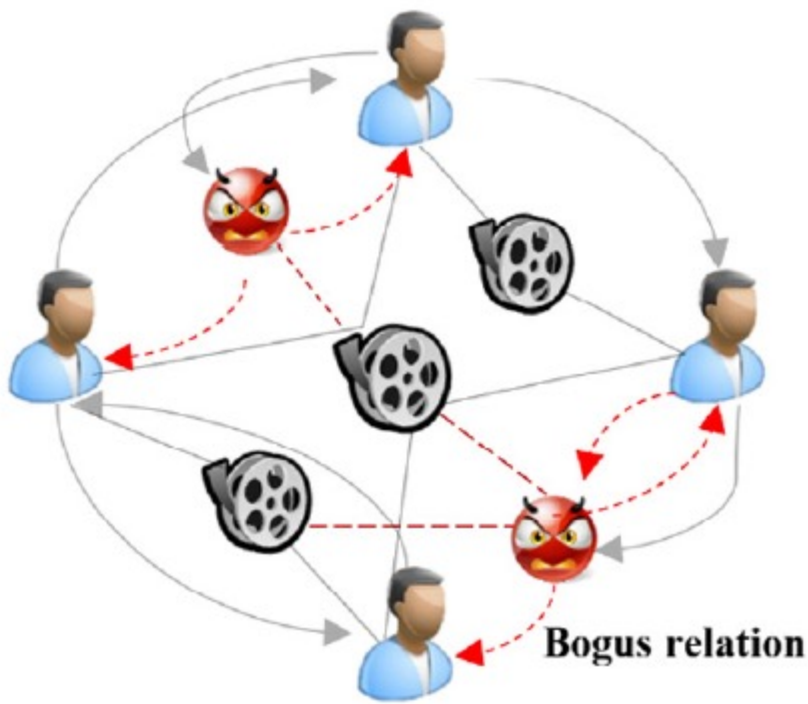
- Three possible causes:
  - 1. Social networks are as **sparse** as user-item interactions.
  - 2. Social networks are **noisy**.
  - 3. Social relationships have **multi-facets**.

Problems in SRS

Social networks are **sparse**

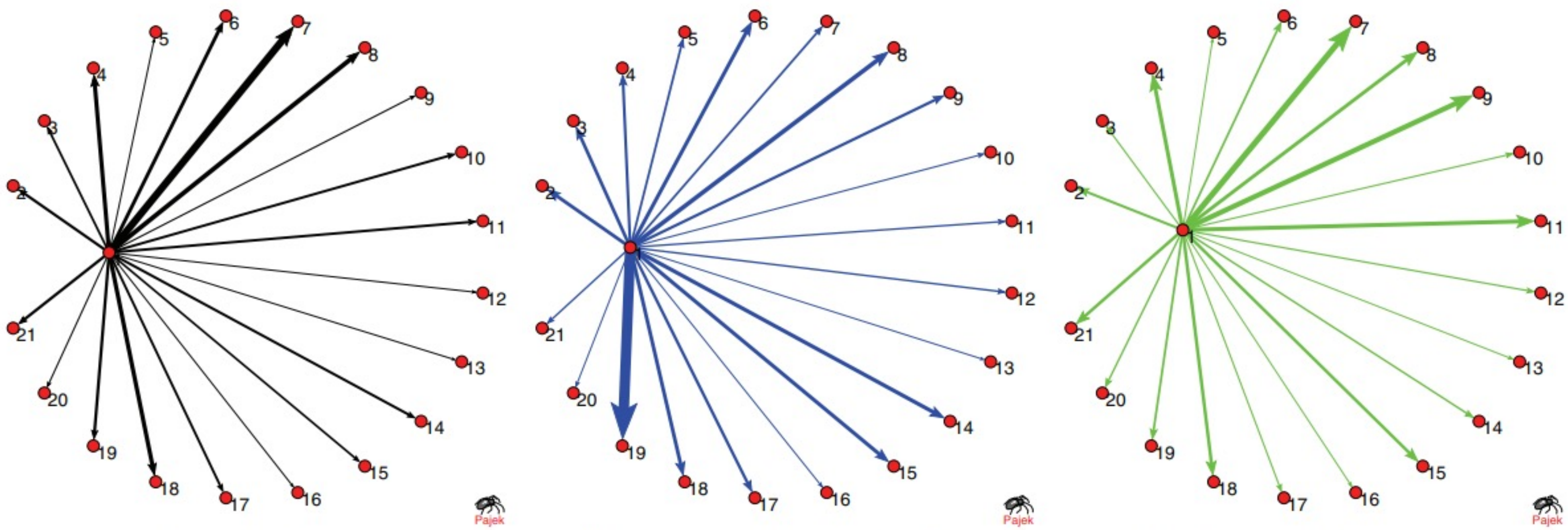


Problems in SRS



Social networks are **noisy.**

Problems in SRS



(a) Single Trust (b) Trust in Home & Garden (c) Trust in Restaurants

Social relations are **multi-faceted**

Tool for RS

Opensource Project

RecQ: A Python Framework for Recommender Systems (TensorFlow Based)

https://github.com/Coder-Yu/RecQ

Thank you!