

# Learning From Networks

—*Algorithms, Theory, & Applications*

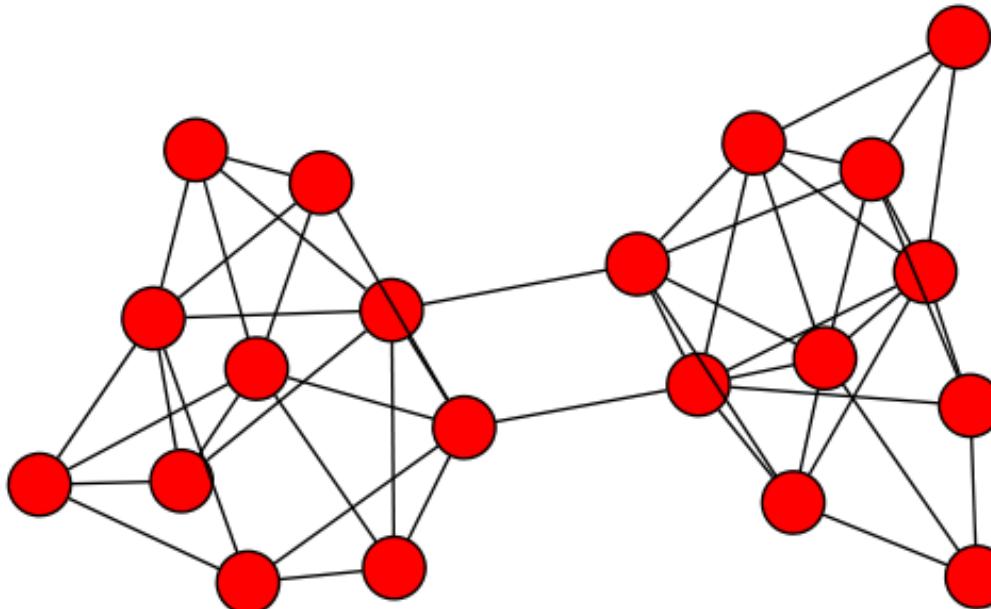
Xiao Huang\*, Peng Cui\*, Yuxiao Dong\*, Jundong Li, Huan Liu, Jian Pei, Le Song, Jie Tang, Fei Wang, Hongxia Yang, Wenwu Zhu

xhuang@tamu.edu; cuip@tsinghua.edu.cn; yuxdong@microsoft.com; jundongl@asu.edu;  
huan.liu@asu.edu; jpei@cs.sfu.ca; le.song@antfin.com; jietang@tsinghua.edu.cn;  
few2001@med.cornell.edu; yang.yhx@alibaba-inc.com; wwzhu@tsinghua.edu.cn

Joint first authors, the others are listed in alphabetical order

# Network (Graph)

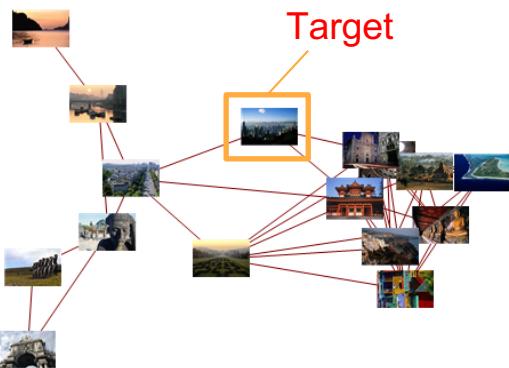
The general description of data and their relations



# Why network is important?

Can you name a case where only care about an object but not its relations with other subjects?

Image Characterization



Reflected by relational subjects

Social Capital



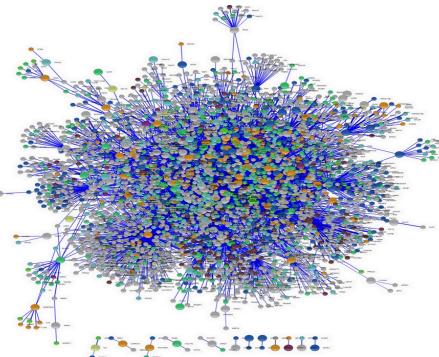
Decided by relational subjects

# Graph/network data is everywhere

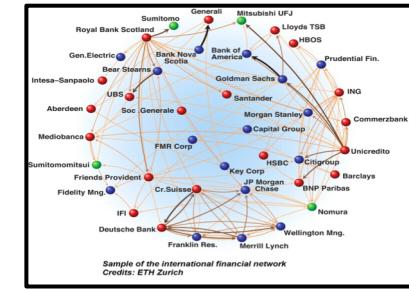
## Social Networks



## Biology Networks



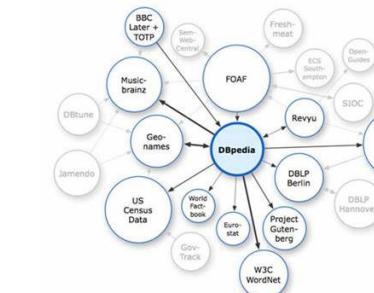
## Finance Networks



## Internet of Things



## Information Networks



## Logistic Networks



# Graph as a data model

- The last resort for the curse of complexity in real applications
  - Geographical networks, relationships, ...
- Divide-and-conquer in modeling
  - Individual nodes and edges are well structured
  - Global structures are weakly organized

# Mining graphs and networks

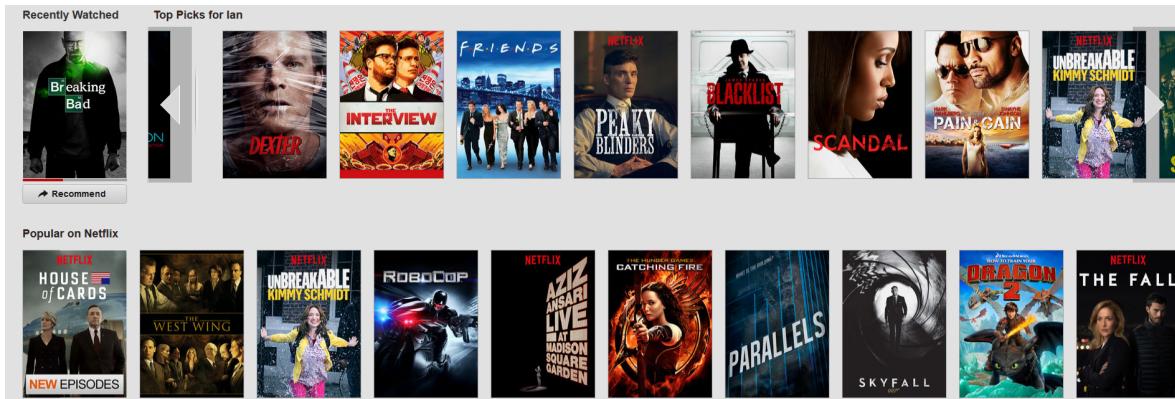
- Multiple possible views
  - Micro view: nodes and edges
  - Macro view: global structures and properties
  - Temporal view: changes over time
- Many important tasks
  - Patterns, classification, prediction, clustering, outlier detection, ...

# Why is graph and network mining challenging?

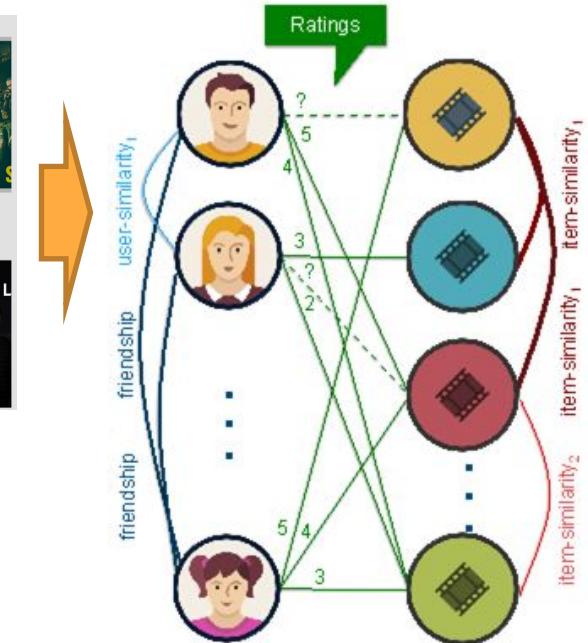
- Key assumptions in efficient data processing
  - Well structured data
  - Synchronized
- Challenges in graph and network data
  - Some basic structures, but many unstructured elements
  - Asynchronous

# Many applications are intrinsically network problems

## Recommendation Systems



Link prediction in bipartite graphs

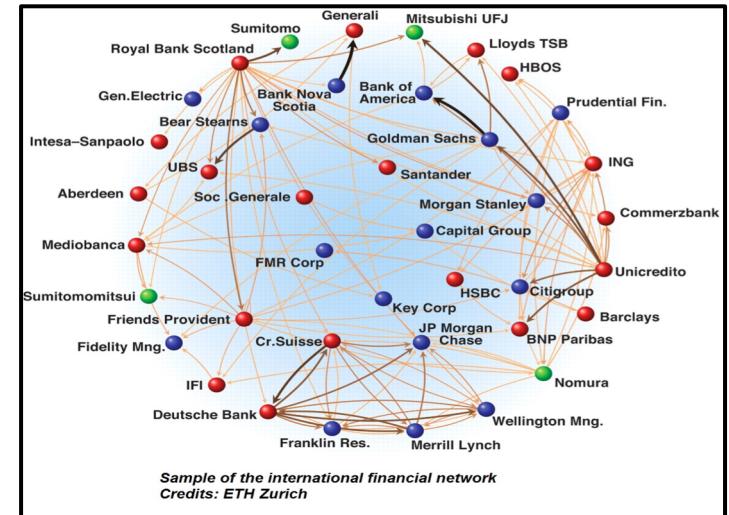


# Many applications are intrinsically network problems

## Financial credit & risk management



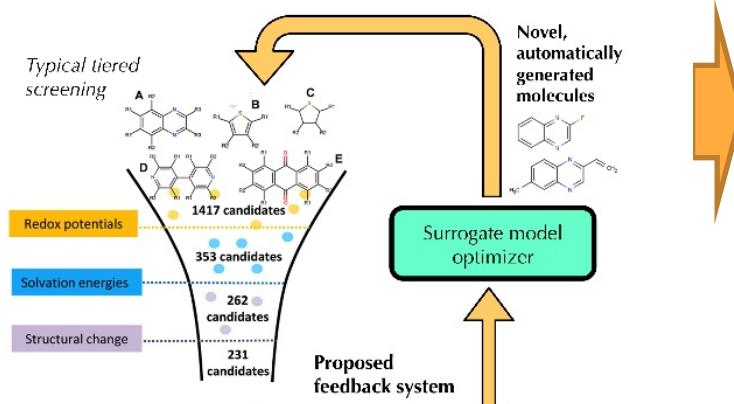
## Node importance & classification



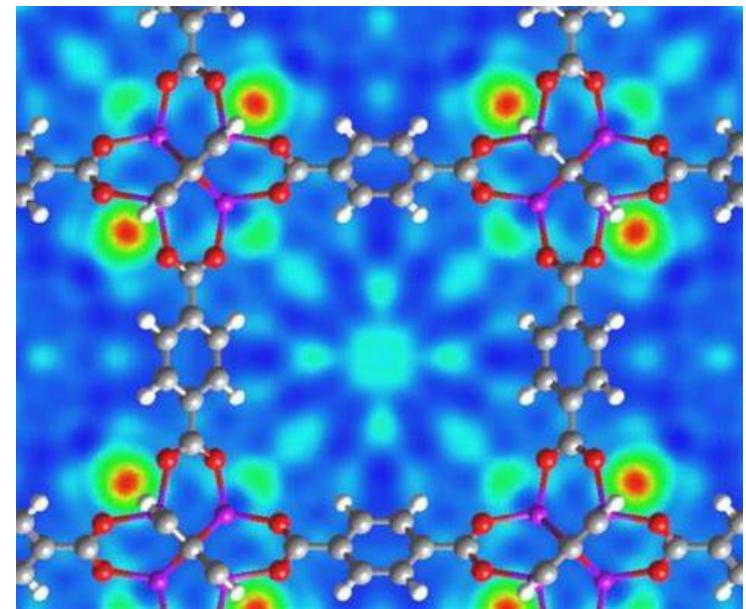
# Many applications are intrinsically network problems

## New material discovery

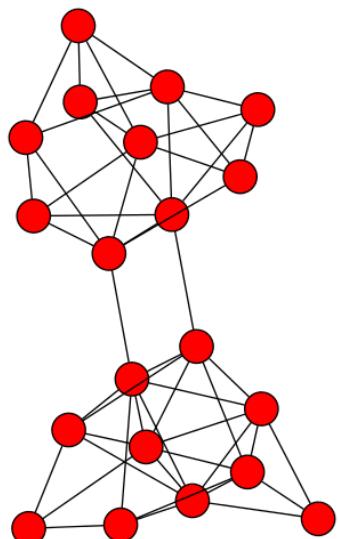
### Materials discovery engine concept



## Subgraph pattern discovery



# Traditional methods – graph theory



Graph  
Theory

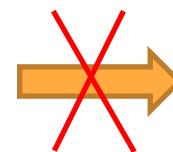


Centrality Problem

Isomorphism Problem

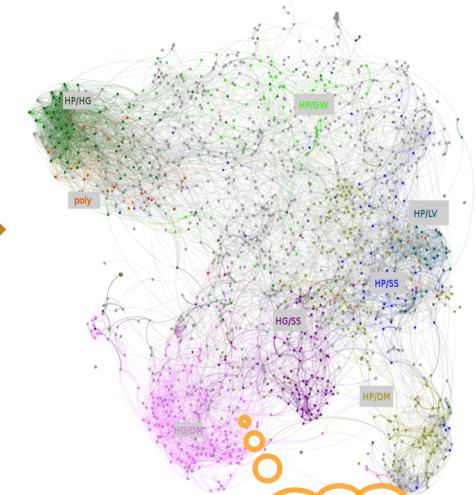
Routing Problem

...



Challenge 1: Problem Scale

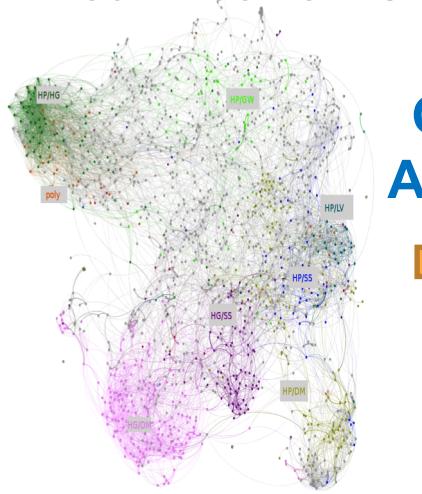
Real networks



Billion  
nodes

# Traditional methods – graph analysis

## Real Networks



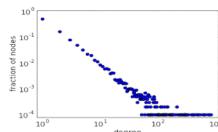
Graph Analysis



High Complexity

## Graph Patterns

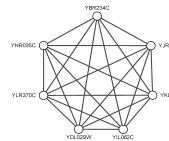
Power-law



Triadic Closure



Clustering effect



## Applications

Link prediction

Community detection

Anomaly detection

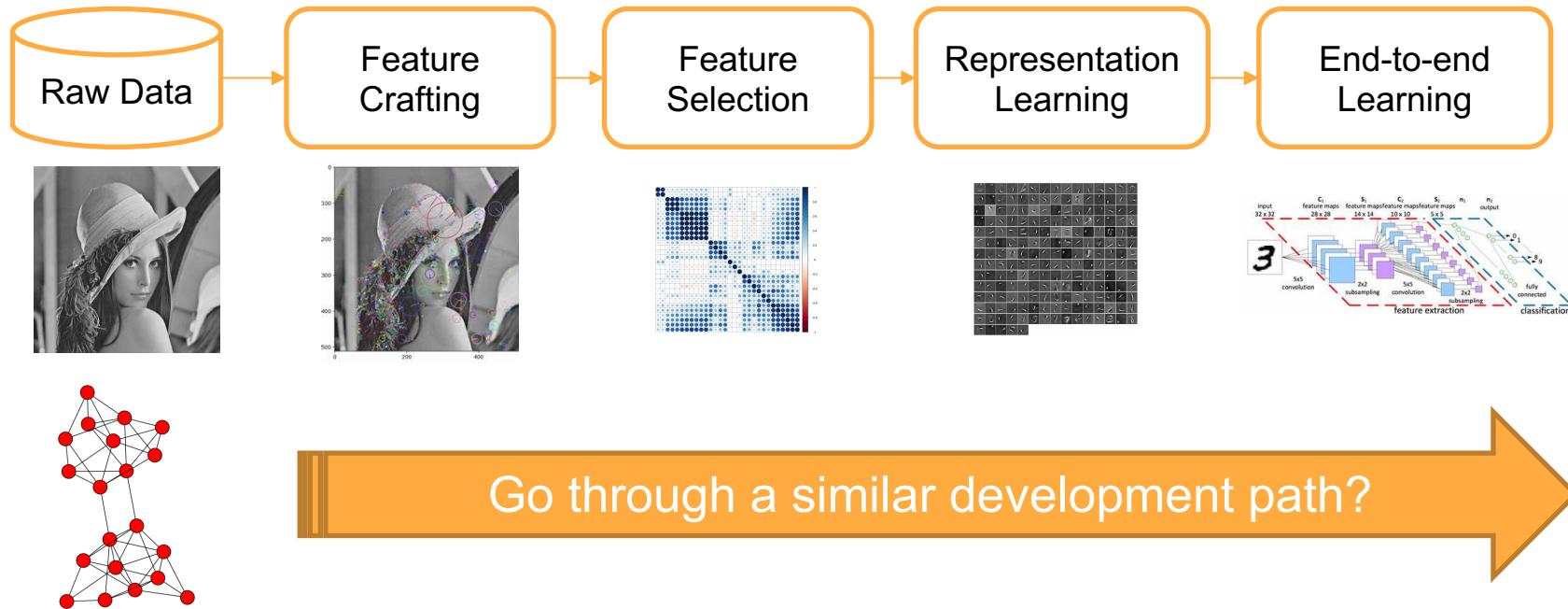
...

High Diversity

Challenge 2: Complexity and Diversity

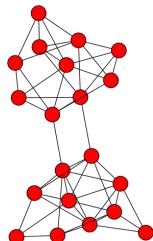
# From graph theory and analysis to learning

Progressive development of learning related fields



# Learning from Networks

Provide **general learning solutions** to **various tasks** over a diverse range of complex networks.



Graph Theory  
and Analysis

Graph Feature  
Selection

Network  
Embedding

Graph Neural  
Network

# Key problems of learning from networks

- **High-dimensional features**  
Feature selection
- **Topological feature representation**  
Network embedding
- **Fusion of topological and semantic information**  
Attribute network embedding
- **End-to-end framework**  
Graph neural network

# Key problems of learning from networks

- **High-dimensional features**

- Feature selection

- **Topological feature representation**

- Network embedding

- **Fusion of topological and semantic information**

- Attribute network embedding

- **End-to-end framework**

- Graph neural network

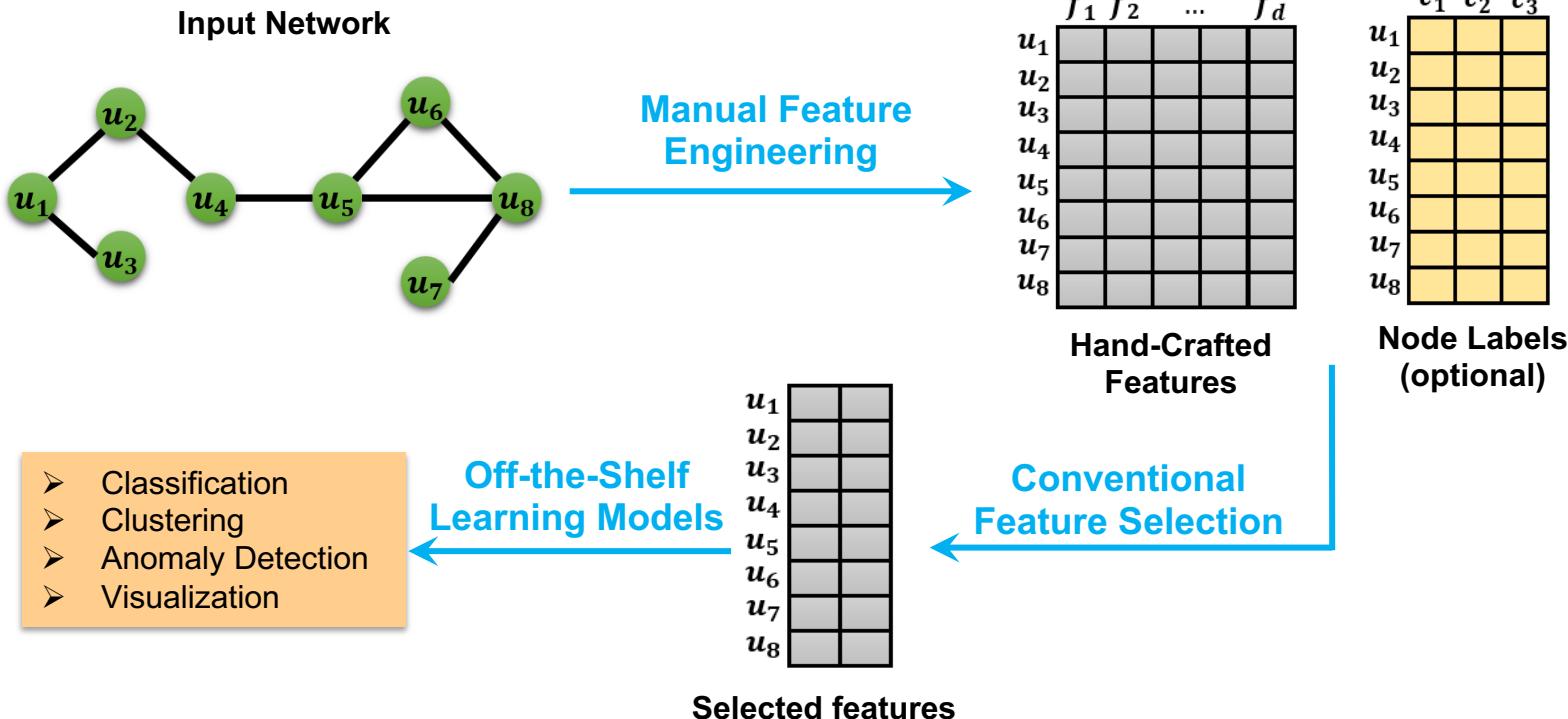
# High-dimensional features

Features of nodes are often in a high-dimensional feature space

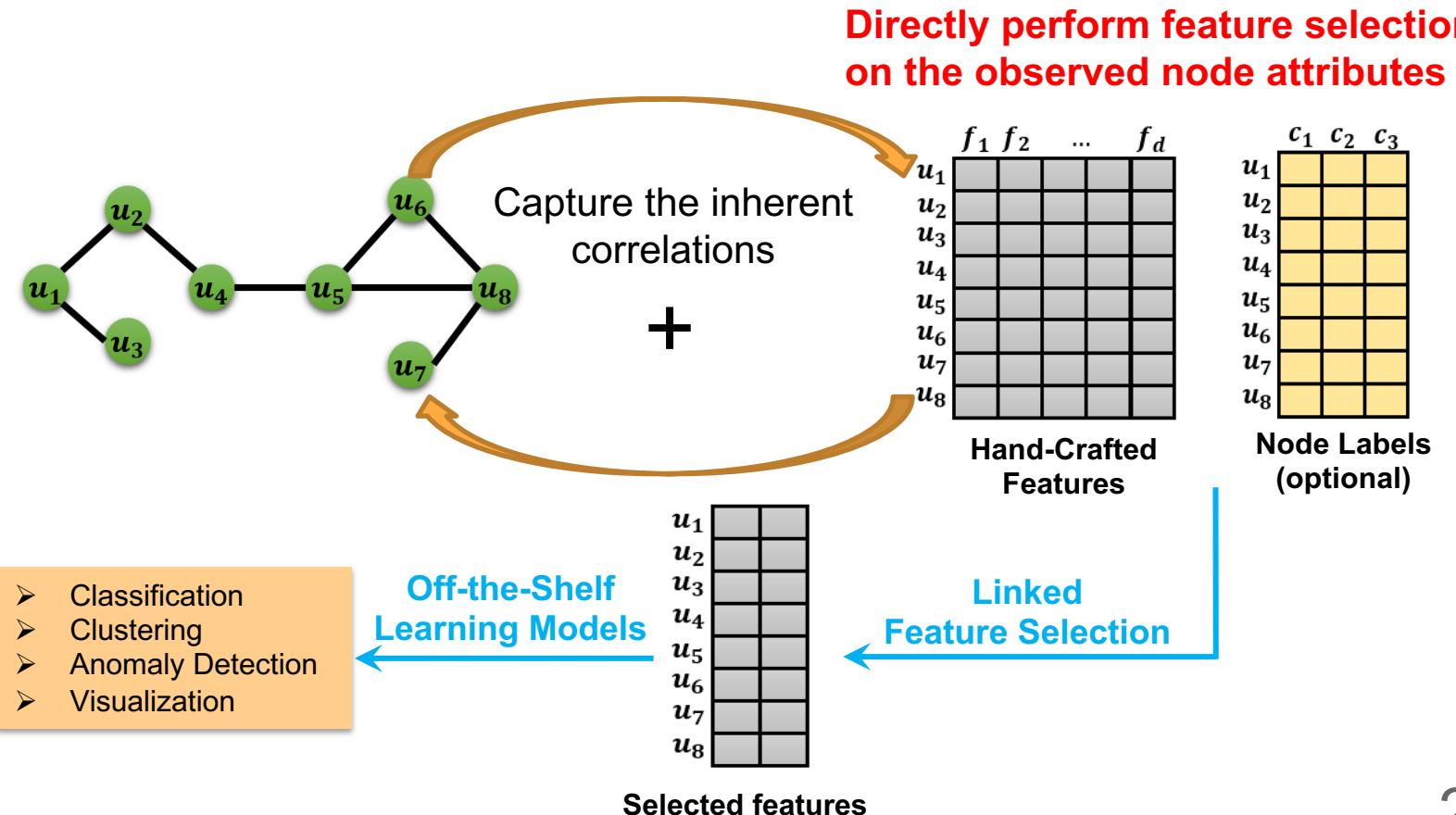
- Scenario 1: without explicit node features
  - Manual feature engineering methods can generate **a large number** of features
  - Not clear what features may be useful for learning on graphs
- Scenario 2: with explicit node features
  - Observed node features are very **high-dimensional, noisy, and sparse**
  - The intrinsic dimensionality of data may be small, e.g., the number of genes responsible for a certain disease

High-dimensional data is often notorious to tackle due to the ***curse of dimensionality***

# Feature selection without explicit node features



# Feature selection with explicit node features

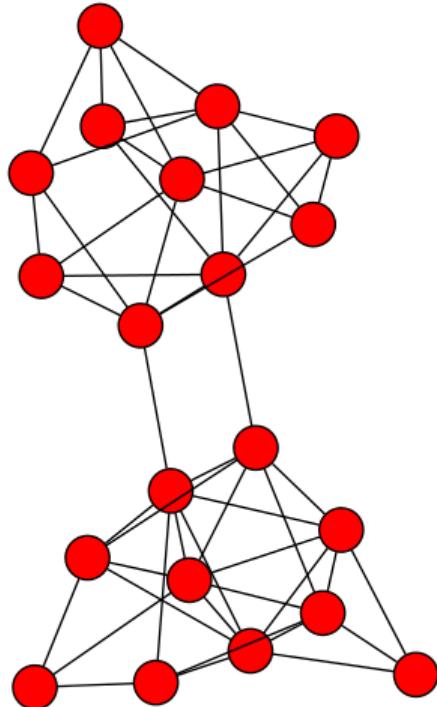


# Key problems of learning from networks

- High-dimensional features
  - Feature selection
- **Topological feature representation**
  - Network embedding
- Fusion of topological and semantic information
  - Attribute network embedding
- End-to-end framework
  - Graph neural network

# Topology to vector space

$$G = (V, E)$$

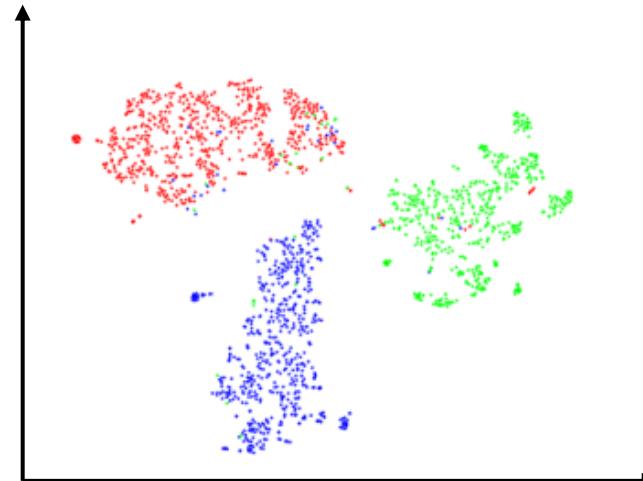


$$G = (V)$$

Vector Space

generate

embed



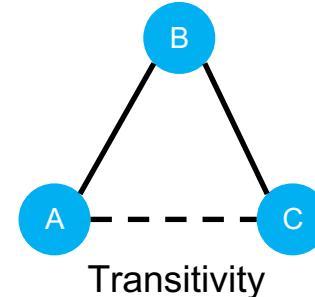
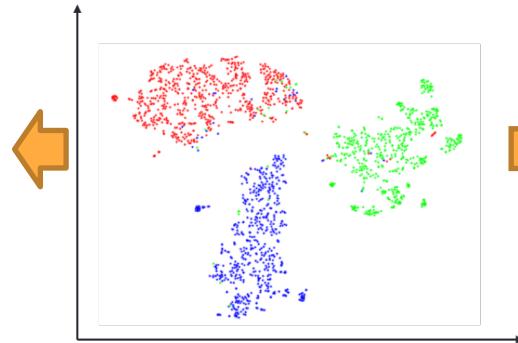
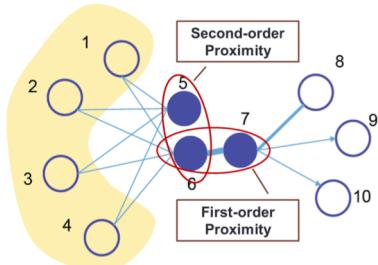
- Easy to parallelize
- Can apply classical ML methods

# Basic requirements of network embedding

**Goal** Support network inference in vector space

Reflect network structure

Maintain network properties



# **Key issues in network embedding**

- **Structure-preserved network embedding**
- **Property-preserved network embedding**
- **Dynamic network embedding**
- **Robustness, Interpretability and Applicability**

# Key problems of learning from networks

- High-dimensional features
  - Feature selection
- Topological feature representation
  - Network embedding
- **Fusion of topological and semantic information**
  - [Attribute network embedding](#)
- End-to-end framework
  - Graph neural network

# How to Jointly Embed Node Attributes & Network?

Texas A&M University  @TAMU · Jun 7  
A new \$1 million @ENERGY grant will help @TAMUEngineering explore the use of big data, A.I., & machine learning to bolster power grids! #tamu



Big Data Analytics Could Reduce Power Grid Outages - Texas A&M T...  
A Texas A&M team will use a \$1 million Department of Energy grant for research that could improve assessment of events that affect power sys...  
[today.tamu.edu](http://today.tamu.edu)

3 23

Texas A&M University  @TAMU · Jun 7  
Texas A&M is ranked No. 8 in the nation in this year's @schoolsEDU 'Best Colleges' survey! Whoop!  #tamu

Customer Reviews  623  
4.3 out of 5 stars ▾

Star Rating	Percentage
5 star	64%
4 star	8%
3 star	6%
2 star	5%
1 star	17%

[Write a review](#)

Apple 15" MacBook Pro by Apple

Capacity: 15 Inch, 2.9GHz Intel Core i7  
[Change](#)  
Price: **\$2,599.00** + Free shipping

Top positive review  
[See all 450 positive reviews](#) ▾  
59 people found this helpful  
 It's a Macbook Pro Maxed out from 2016  
By Timothy D. Gray on January 23, 2018  
Many of the negative reviews here are from people that either don't understand computers or bought during the short time the specs posted by amazon as to what people were buying were wrong. Amazon has now fixed that and what you see is now accurate.

- Node attributes are prevalent in real-world networks
- Examples: **[user content]** in social media, **[reviews]** in co-purchasing networks, & paper abstracts in citation networks

# Mining attributed networks with shallow embedding

- Focuses:

Joint learning, embedding networks, & accelerating optimization

- Methods:

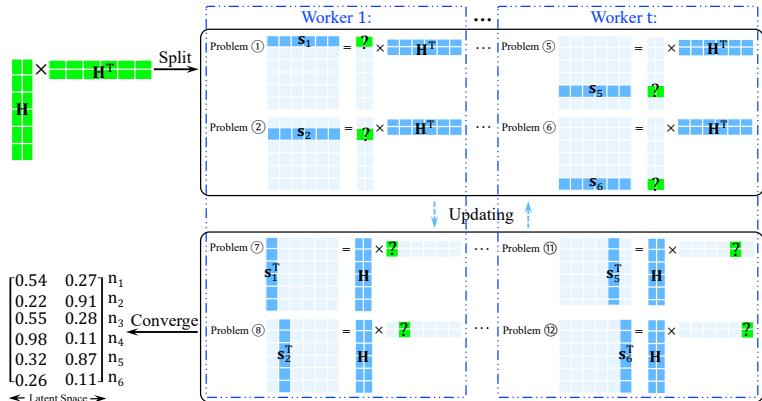
Coupled spectral embedding

Coupled matrix & tri-factorization

Random walk based embedding

- Techniques:

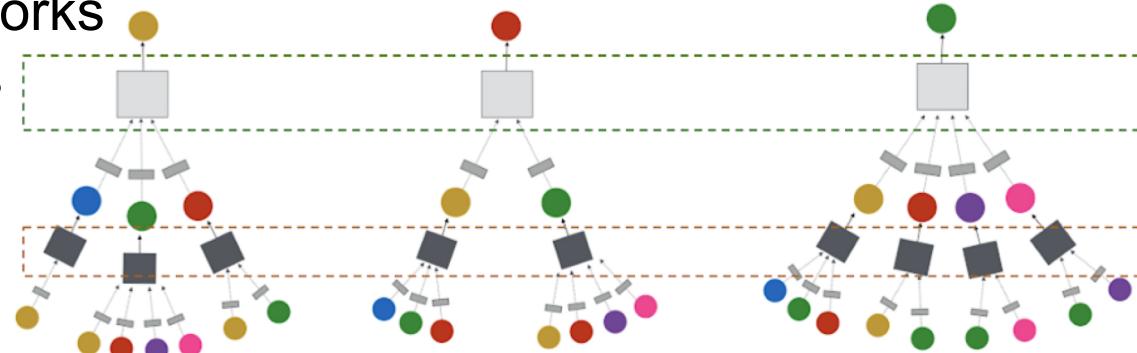
Spectral graph theory, Coupling,  
distributed optimization, joint  
random walks, etc.



$$\begin{aligned} & \|\mathbf{G} - \mathbf{H}\mathbf{H}^T\|_F^2 & & \|\mathbf{A} - \mathbf{H}\mathbf{V}\|_F^2 \\ & \left[ \begin{matrix} \mathbf{h}_1^T & \mathbf{h}_2^T & \mathbf{h}_3^T & \mathbf{h}_4^T & \mathbf{h}_5^T & \cdots & \mathbf{h}_n^T \end{matrix} \right] & \left[ \begin{matrix} \mathbf{v}_1 & \mathbf{v}_2 & \mathbf{v}_3 & \mathbf{v}_4 & \mathbf{v}_5 & \cdots & \mathbf{v}_m \end{matrix} \right] \\ & \left[ \begin{matrix} \mathbf{h}_1 \\ \mathbf{h}_2 \\ \mathbf{h}_3 \\ \mathbf{h}_4 \\ \mathbf{h}_5 \\ \vdots \\ \mathbf{h}_n \end{matrix} \right] \left[ \begin{matrix} \mathbf{g}_{1,1} & \mathbf{g}_{1,2} & \mathbf{g}_{1,3} & \mathbf{g}_{1,4} & \mathbf{g}_{1,5} & \cdots & \mathbf{g}_{1,n} \\ \mathbf{g}_{2,1} & \mathbf{g}_{2,2} & \mathbf{g}_{2,3} & \mathbf{g}_{2,4} & \mathbf{g}_{2,5} & \cdots & \mathbf{g}_{2,n} \\ \mathbf{g}_{3,1} & \mathbf{g}_{3,2} & \mathbf{g}_{3,3} & \mathbf{g}_{3,4} & \mathbf{g}_{3,5} & \cdots & \mathbf{g}_{3,n} \\ \mathbf{g}_{4,1} & \mathbf{g}_{4,2} & \mathbf{g}_{4,3} & \mathbf{g}_{4,4} & \mathbf{g}_{4,5} & \cdots & \mathbf{g}_{4,n} \\ \mathbf{g}_{5,1} & \mathbf{g}_{5,2} & \mathbf{g}_{5,3} & \mathbf{g}_{5,4} & \mathbf{g}_{5,5} & \cdots & \mathbf{g}_{5,n} \\ \vdots & & \vdots & & \vdots & & \vdots \\ \mathbf{g}_{n,1} & \mathbf{g}_{n,2} & \mathbf{g}_{n,3} & \mathbf{g}_{n,4} & \mathbf{g}_{n,5} & \cdots & \mathbf{g}_{n,n} \end{matrix} \right] & & \left[ \begin{matrix} \mathbf{h}_1 \\ \mathbf{h}_2 \\ \mathbf{h}_3 \\ \mathbf{h}_4 \\ \mathbf{h}_5 \\ \vdots \\ \mathbf{h}_n \end{matrix} \right] \left[ \begin{matrix} \mathbf{a}_{1,1} & \mathbf{a}_{1,2} & \mathbf{a}_{1,3} & \mathbf{a}_{1,4} & \mathbf{a}_{1,5} & \cdots & \mathbf{a}_{1,m} \\ \mathbf{a}_{2,1} & \mathbf{a}_{2,2} & \mathbf{a}_{2,3} & \mathbf{a}_{2,4} & \mathbf{a}_{2,5} & \cdots & \mathbf{a}_{2,m} \\ \mathbf{a}_{3,1} & \mathbf{a}_{3,2} & \mathbf{a}_{3,3} & \mathbf{a}_{3,4} & \mathbf{a}_{3,5} & \cdots & \mathbf{a}_{3,m} \\ \mathbf{a}_{4,1} & \mathbf{a}_{4,2} & \mathbf{a}_{4,3} & \mathbf{a}_{4,4} & \mathbf{a}_{4,5} & \cdots & \mathbf{a}_{4,m} \\ \mathbf{a}_{5,1} & \mathbf{a}_{5,2} & \mathbf{a}_{5,3} & \mathbf{a}_{5,4} & \mathbf{a}_{5,5} & \cdots & \mathbf{a}_{5,m} \\ \vdots & & \vdots & & \vdots & & \vdots \\ \mathbf{a}_{n,1} & \mathbf{a}_{n,2} & \mathbf{a}_{n,3} & \mathbf{a}_{n,4} & \mathbf{a}_{n,5} & \cdots & \mathbf{a}_{n,m} \end{matrix} \right] \end{aligned}$$

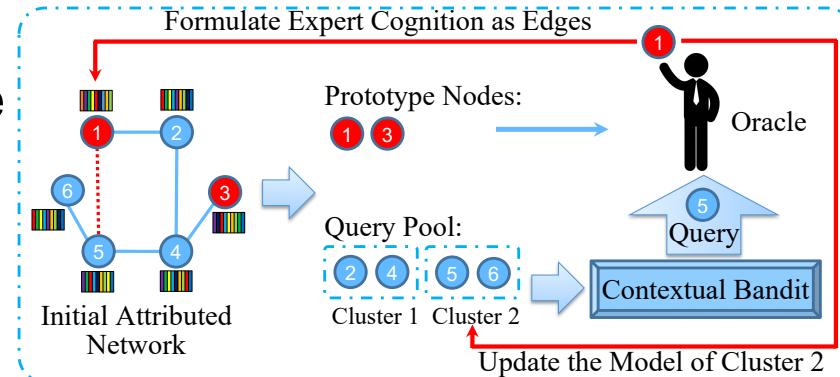
# Mining attributed networks with deep embedding

- Focuses:  
Deep architectures for networks & joint learning
- Methods:  
Objective function based deep embedding  
Graph neural networks
- Architectures:  
Graph convolutional networks  
Graph recurrent networks



# Human-centric Network Analysis

- **Focuses:**  
Interpretable embedding, & utilizing network embedding to incorporate human knowledge
- **Methods:**  
Interpretable node representation learning.  
Attributed network analysis with humans in the loop
- **Techniques:**  
Linking embedding with interpretable node attributes, encode knowledge as links, etc.



# Key problems of learning from networks

- High-dimensional features

- Feature selection

- Topological feature representation

- Network embedding

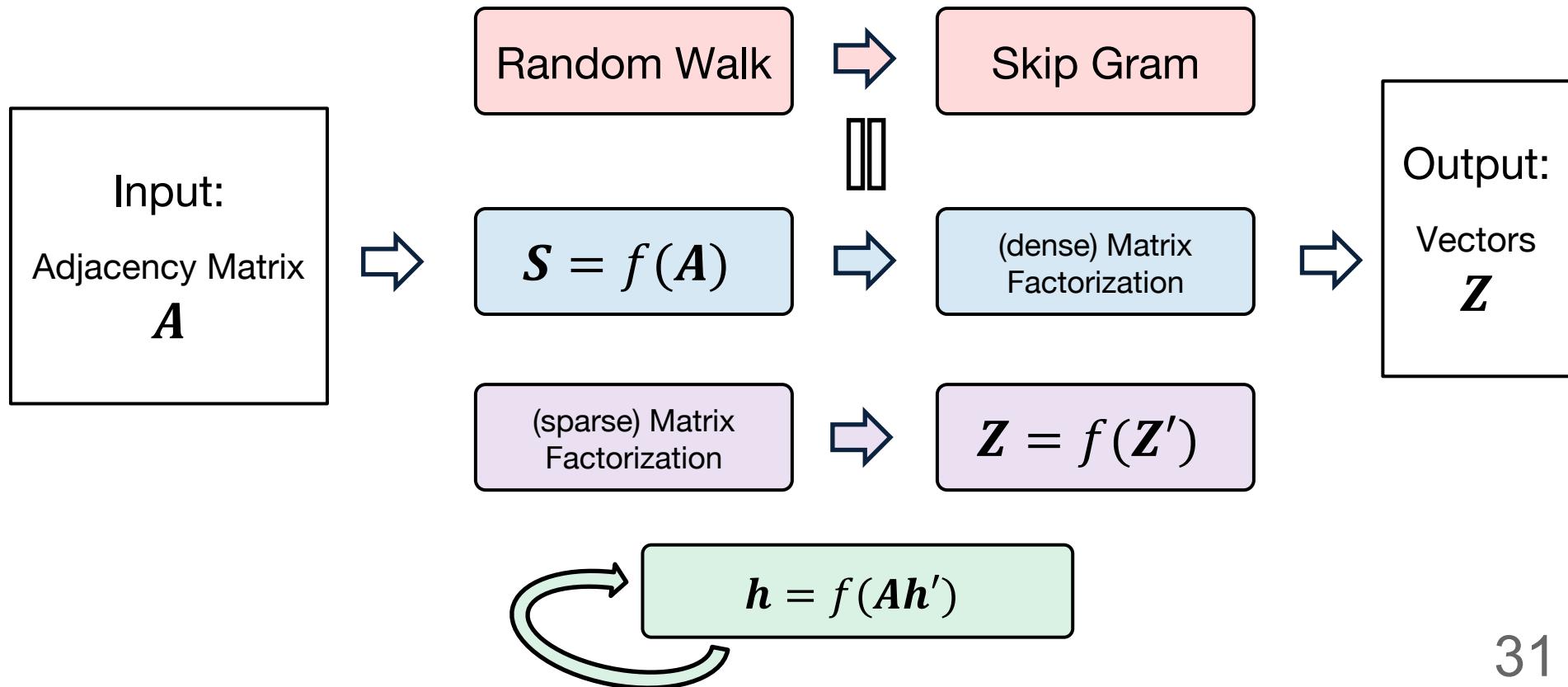
- Fusion of topological and semantic information

- Attribute network embedding

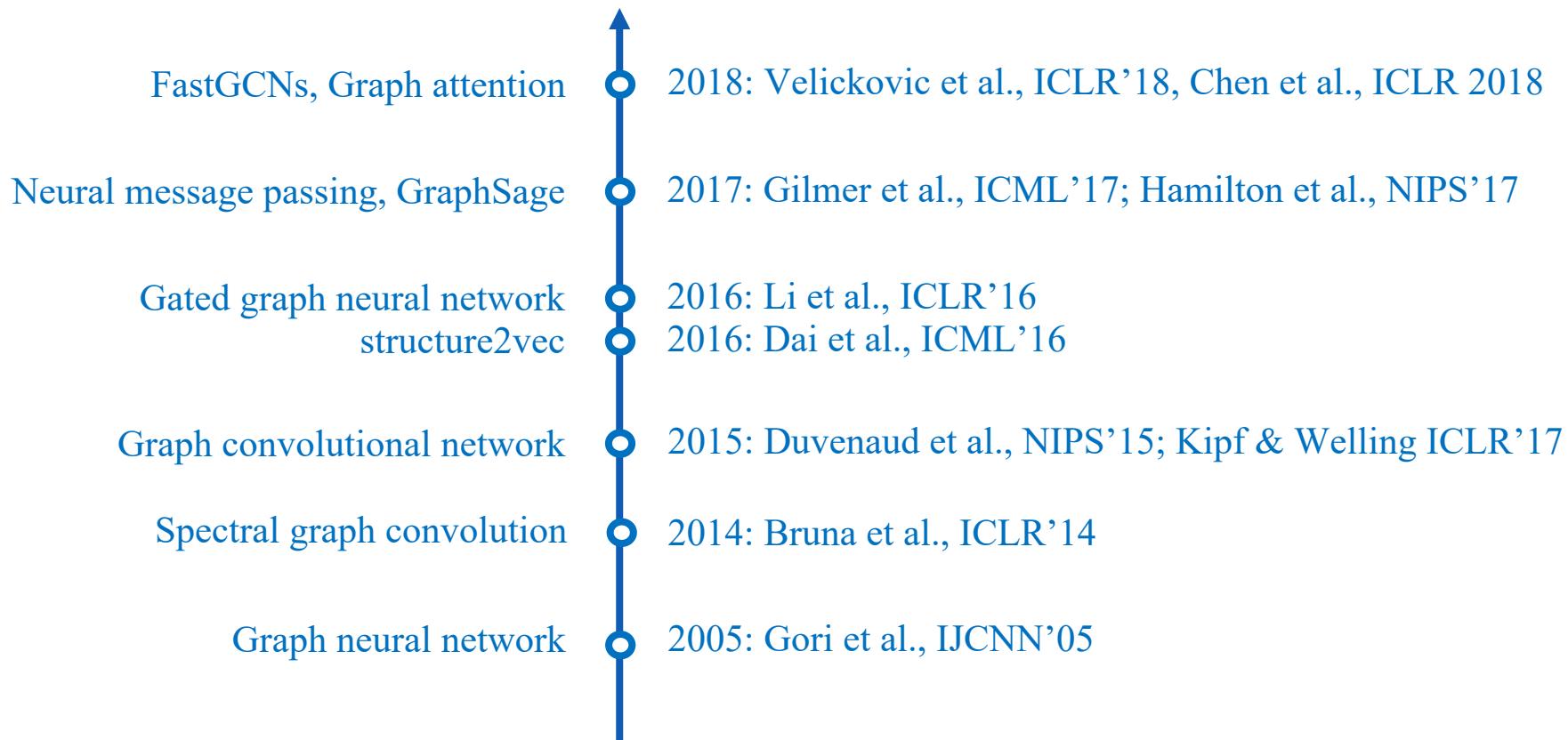
- End-to-end framework

- Graph neural network

# Connecting NE with Graph Neural Networks

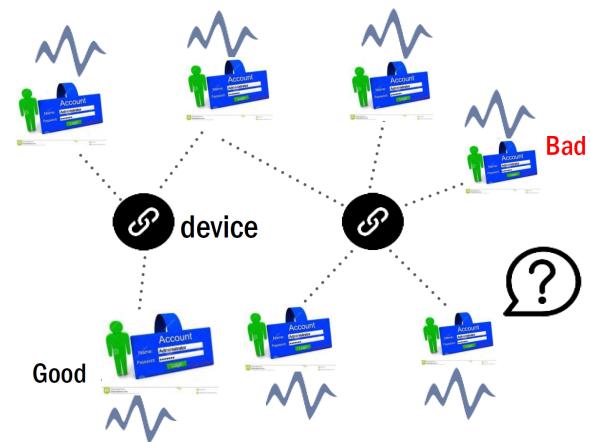


# Graph Neural Networks



# Real world applications

- Heterogeneous knowledge graphs
- Online recommendation
- Online to offline recommendation
- Anomaly detection in FinTech



## Motivations

## Network Embedding

## Graph Neural Networks

