KDD 2019

Learning From Networks

——*Algorithms, Theory, & Applications*

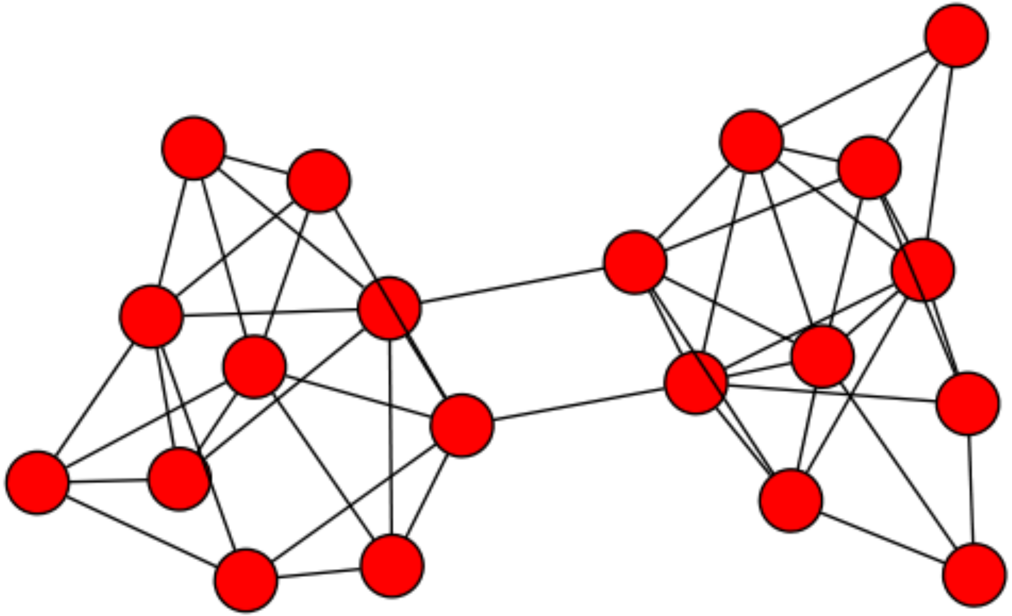
Xiao Huang*, Peng Cui*, Yuxiao Dong*, Jundong Li, Huan Liu, Jian Pei, Le Song, Jie Tang, Fei Wang, Hongxia Yang, Wenwu Zhu

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)

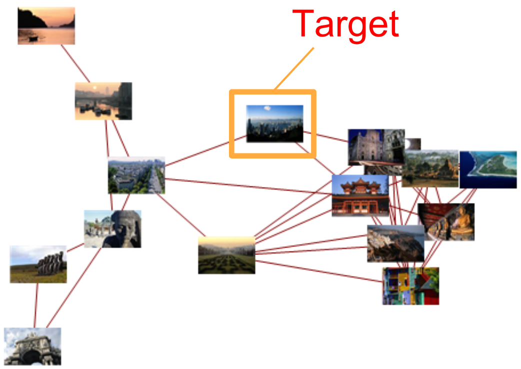
The general description of data and their relations



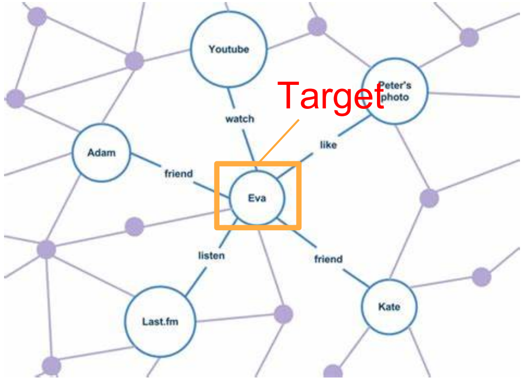
Why network is important?

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

Image Characterization



Social Capital

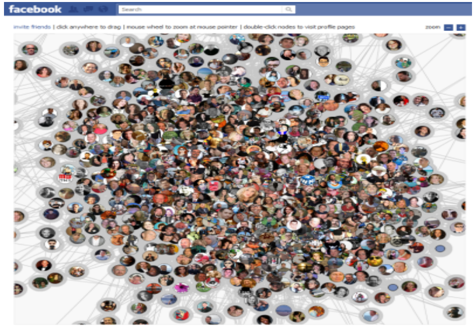


Reflected by relational subjects

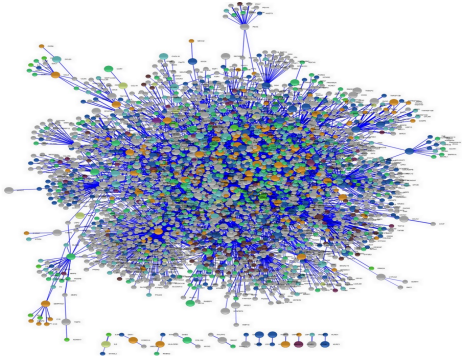
Decided by relational subjects

Graph/network data is everywhere

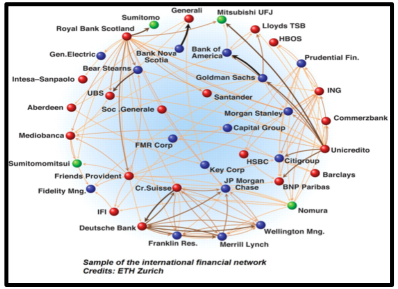
Social Networks



Biology Networks

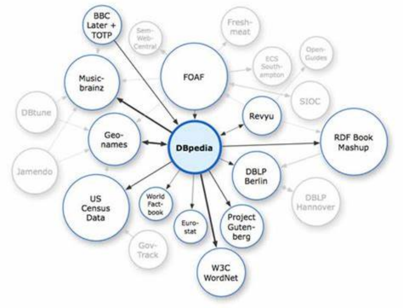


Finance Networks



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

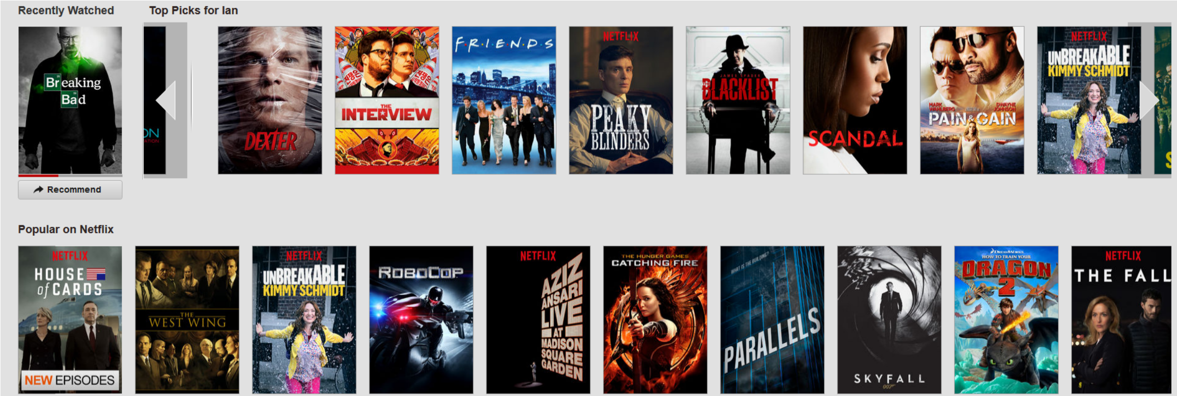
- Multiple possible views
  - Micro view: nodes and edges
  - Macro view: global structures and properties
  - Temporal view: changes over time
- Many important tasks
  - Patterns, classification, prediction, clustering, outlier detection, …

Why is graph and network mining challenging?

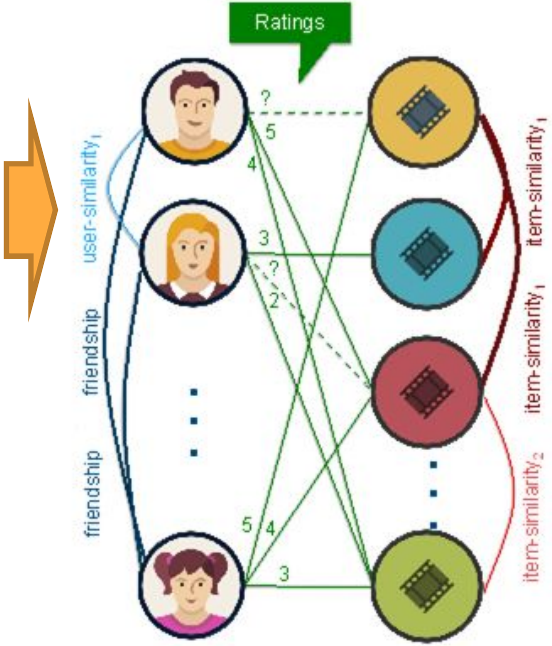
- Key assumptions in efficient data processing
  - Well structured data
  - Synchronized
- Challenges in graph and network data
  - Some basic structures, but many unstructured elements
  - Asynchronous

Many applications are intrinsically network problems

Recommendation Systems



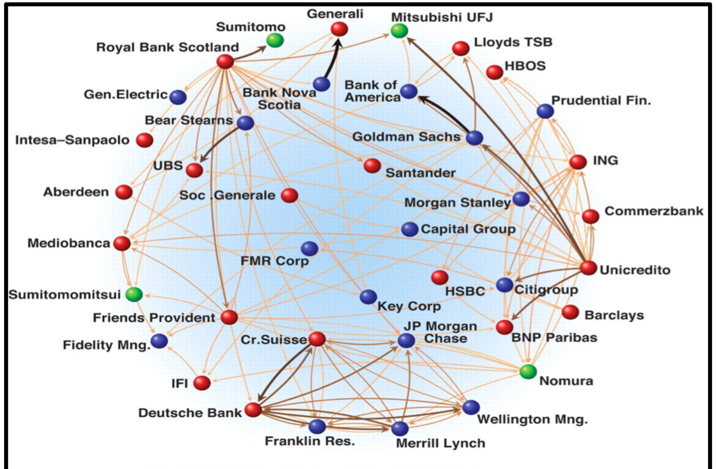
Link prediction in bipartite graphs



Many applications are intrinsically network problems

Financial credit & risk management

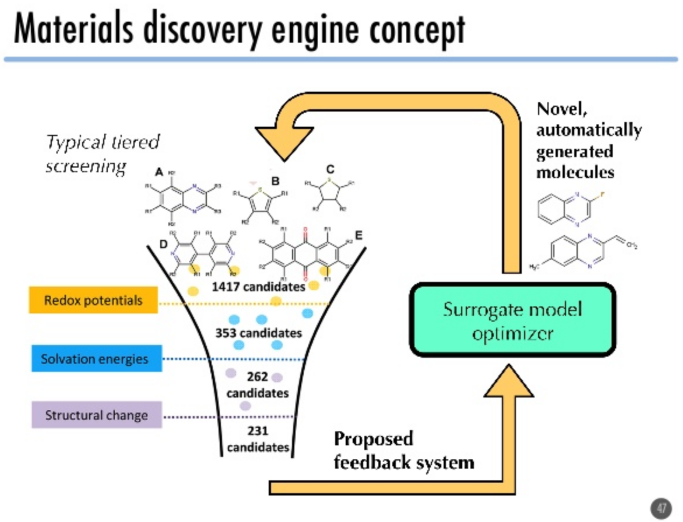
Node importance & classification



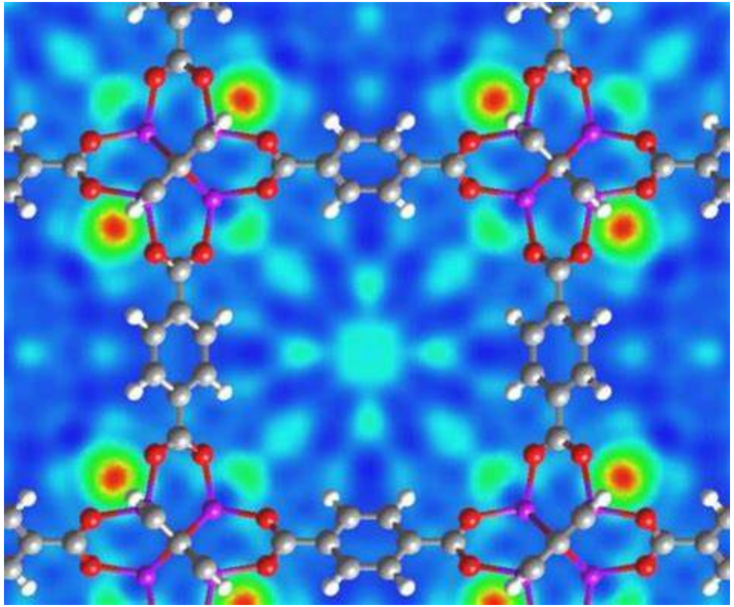
Sample of the international financial network
Credits: ETH Zurich

Many applications are intrinsically network problems

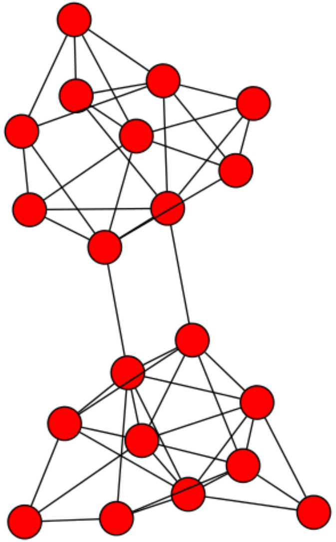
New material discovery



Subgraph pattern discovery



Traditional methods – graph theory



Graph Theory

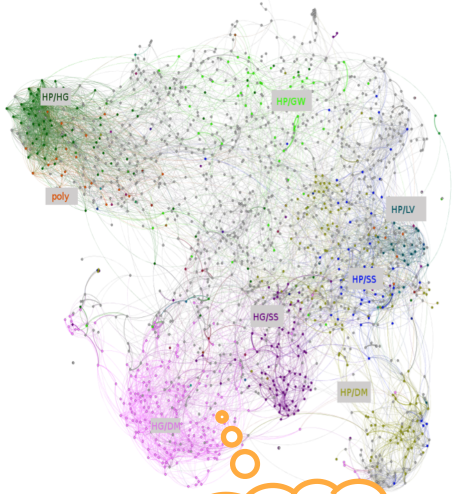
Centrality Problem

Isomorphism Problem

Routing Problem

…

Real networks

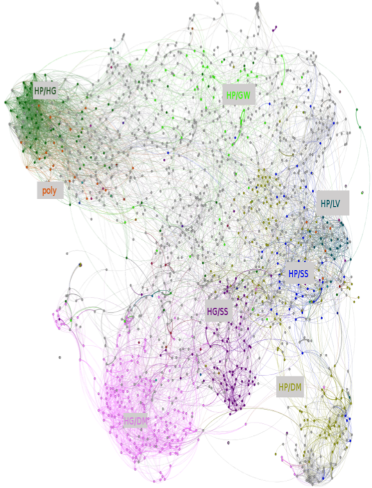


Billion nodes

Challenge 1: Problem Scale

Traditional methods – graph analysis

Real Networks

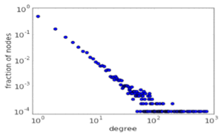


High Complexity

Graph Analysis

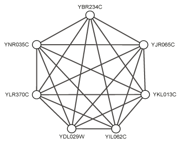
Graph Patterns

Power-law



Triadic Closure

Clustering effect



Applications

Link prediction

Community detection

Anomaly detection

…

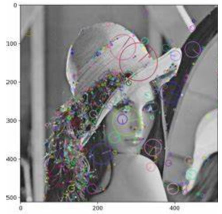
High Diversity

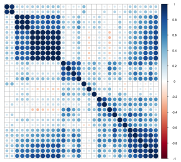
Challenge 2: Complexity and Diversity

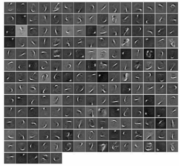
From graph theory and analysis to learning

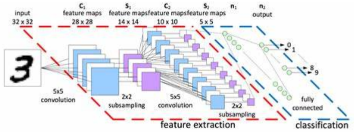
Progressive development of learning related fields

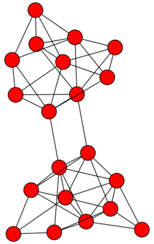
Raw Data → Feature Crafting → Feature Selection → Representation Learning → End-to-end Learning







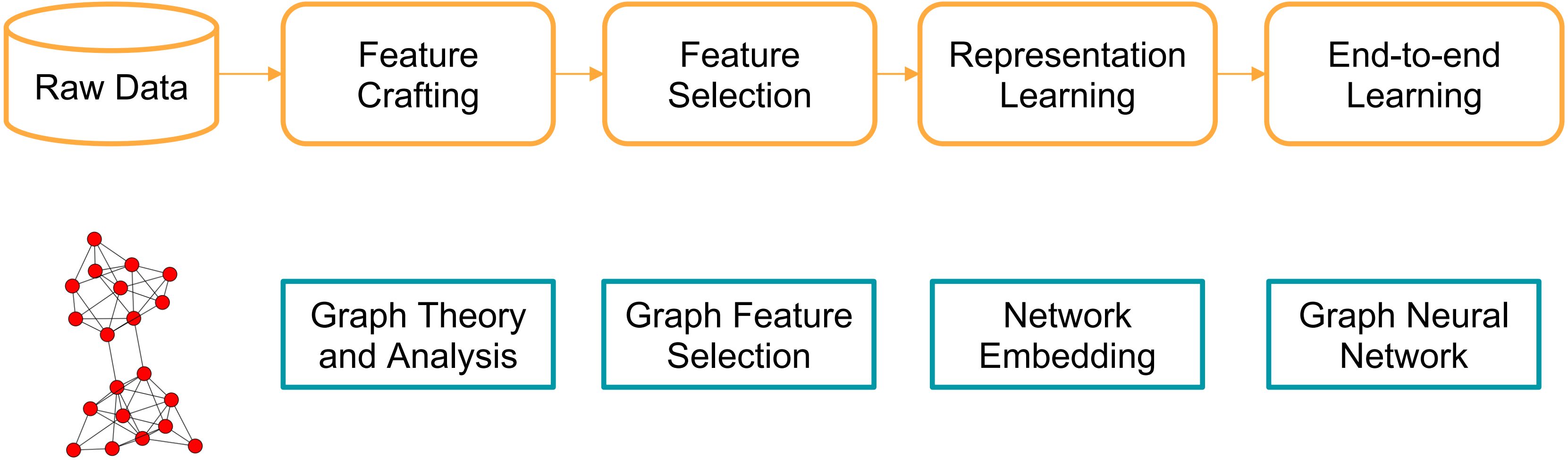




Go through a similar development path?

Learning from Networks

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



Key problems of learning from networks

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

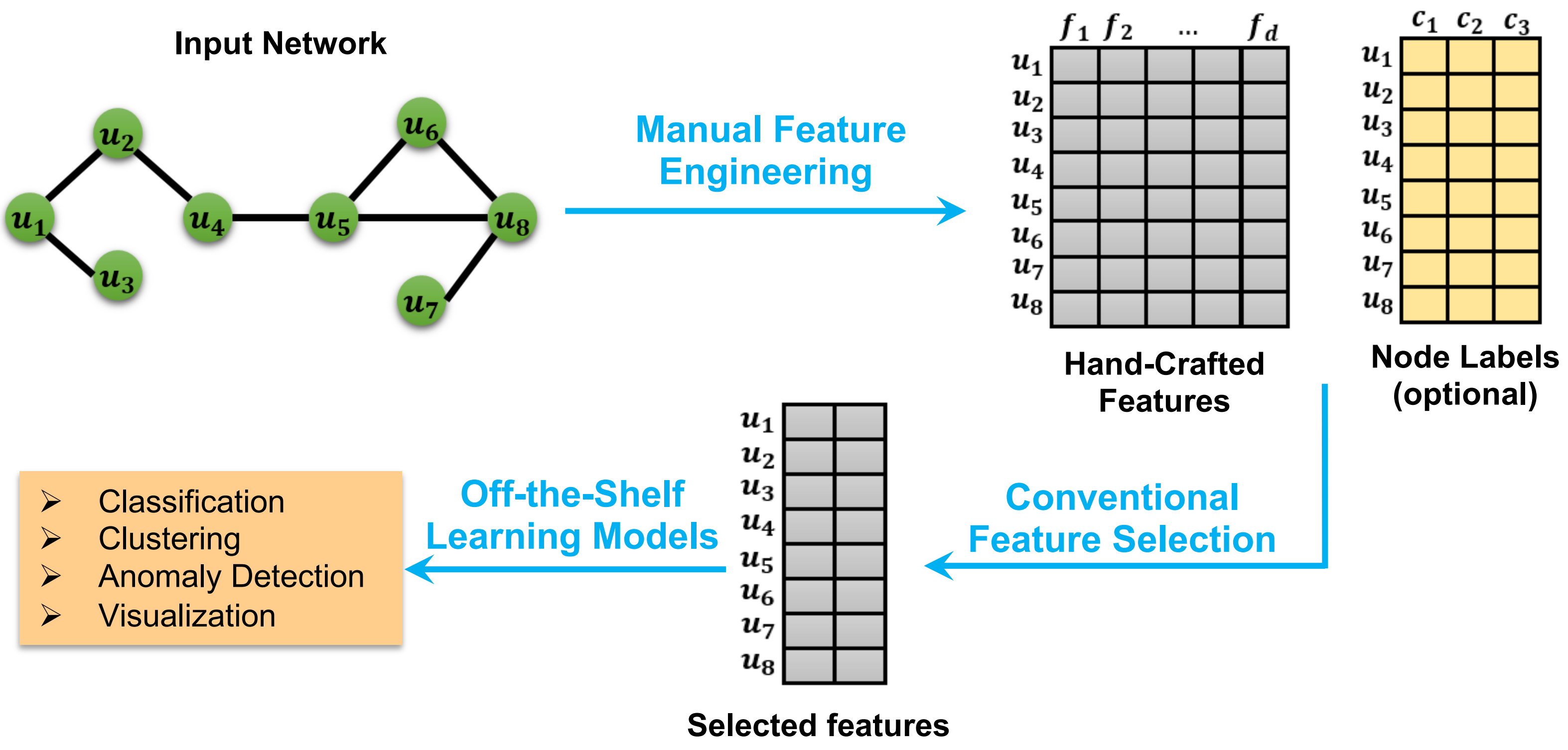
High-dimensional features

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

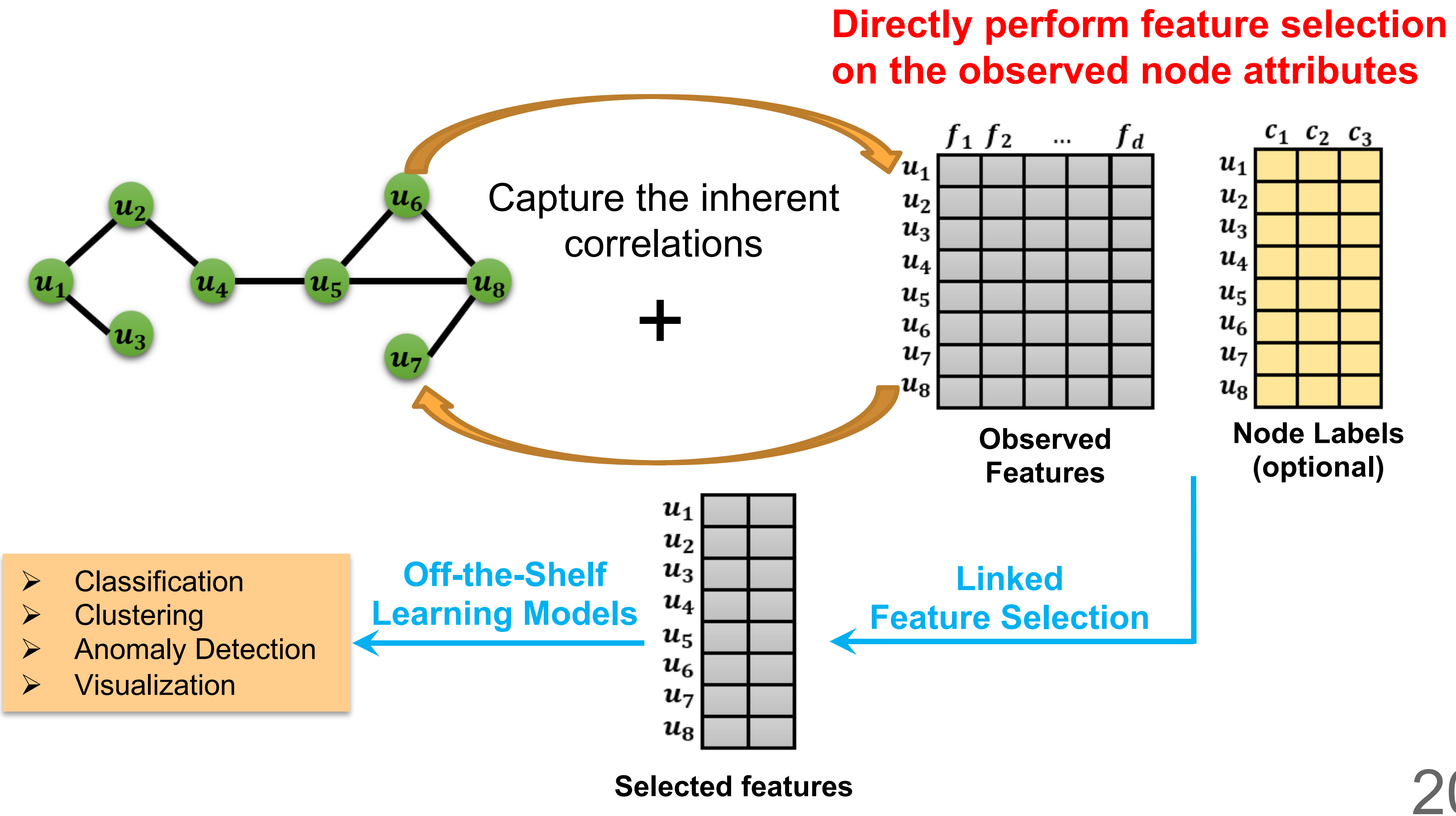
- Scenario 1: without explicit node features
  - Manual feature engineering methods can generate **a large number** of features
  - Not clear what features may be useful for learning on graphs
- Scenario 2: with explicit node features
  - Observed node features are very **high-dimensional, noisy, and 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

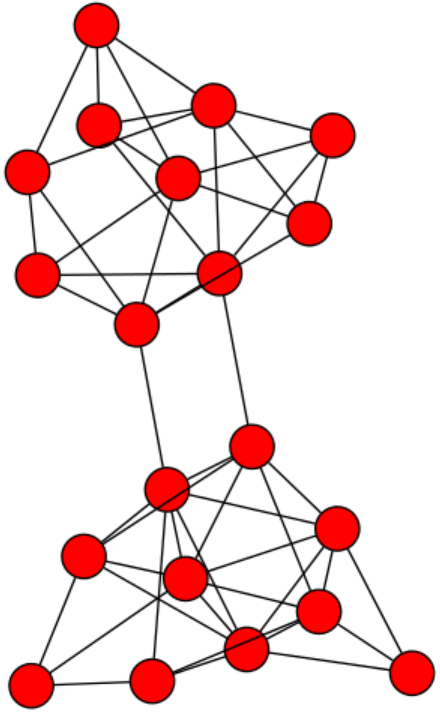
- ❑ **Topological feature representation**
  Netowork embedding

- ❑ **Fusion of topological and semantic information**
  Attribute network embedding

- ❑ **End-to-end framework**
  Graph neural network

Topology to vector space

G = (V, E)

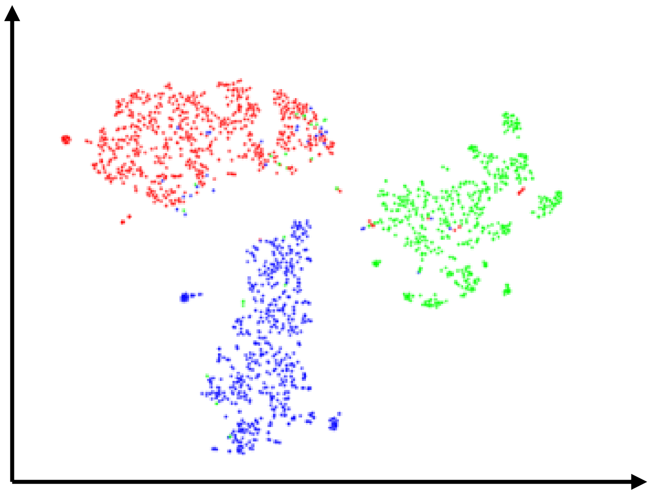


generate

embed

G = (V)

Vector Space



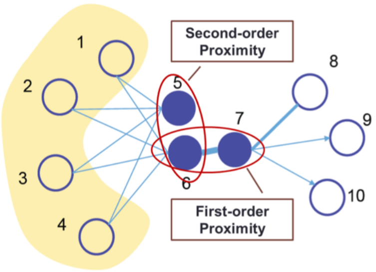
- Easy to parallelize
- Can apply classical ML methods

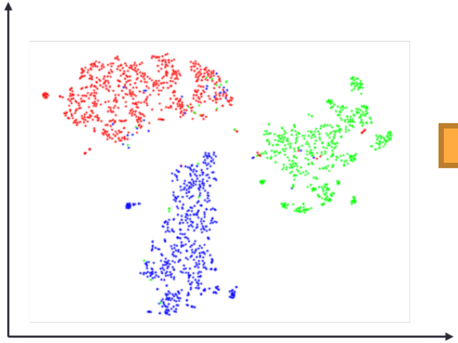
Basic requirements of network embedding

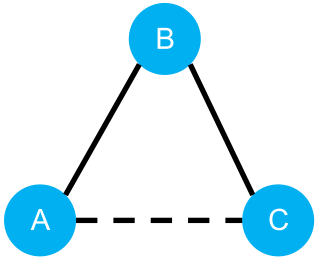
Goal **Support network inference in vector space**

Reflect network structure

Maintain network properties







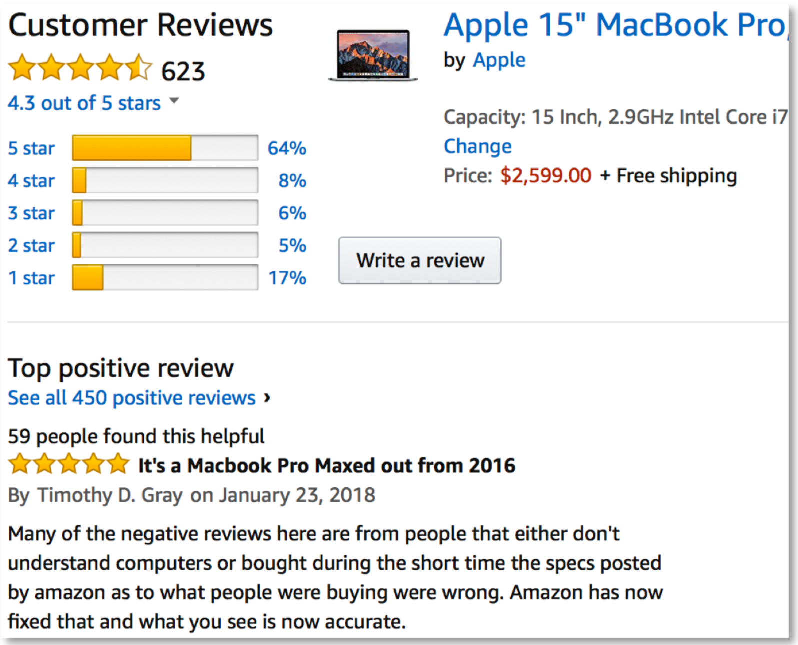
Key issues in network embedding

- **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**
  Netowork embedding
- ❑ **Fusion of topological and semantic information**
  Attribute network embedding
- ❑ **End-to-end framework**
  Graph neural network

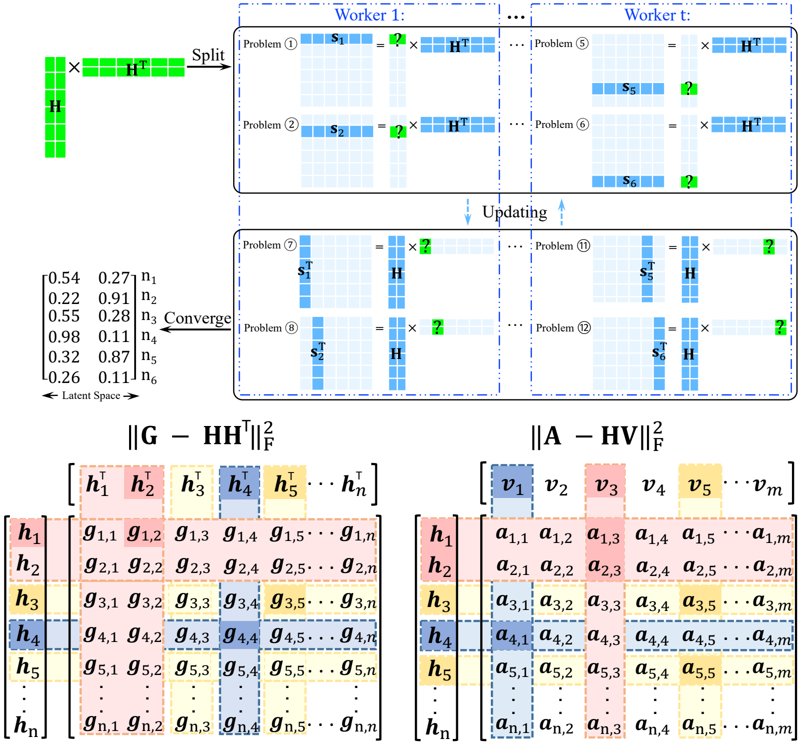
How to Jointly Embed Node Attributes & Network?



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

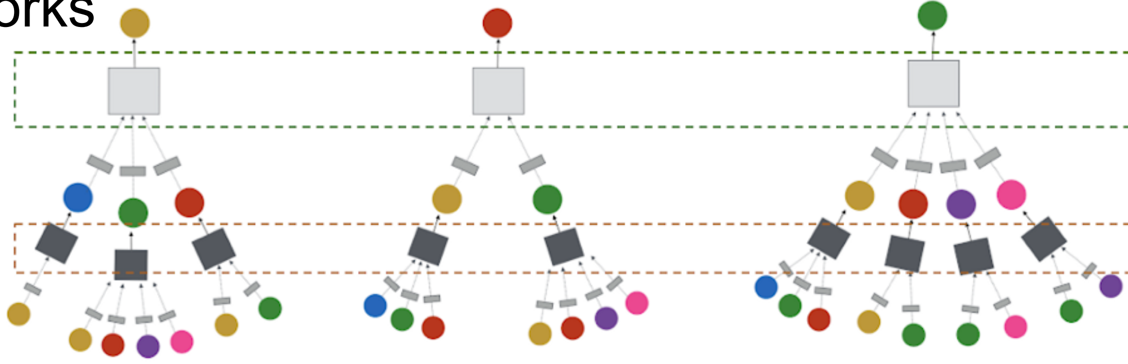
Mining attributed networks with shallow embedding

- Focuses:
  Joint learning, embedding networks, & accelerating optimization
- Methods:
  Coupled spectral embedding
  Coupled matrix & tri-factorization
  Random walk based embedding
- Techniques:
  Coupling, spectral graph theory, distributed optimization, random walks, etc.



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

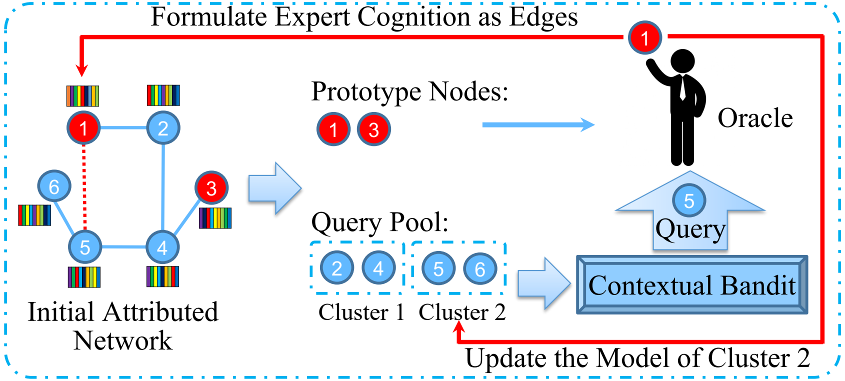


Human-centric Network Analysis

- Focuses:
  Interpretable embedding, & utilizing network embedding to incorporate human knowledge

- Methods:
  Interpretable node representation learning.
  Attributed network analysis with humans in the loop

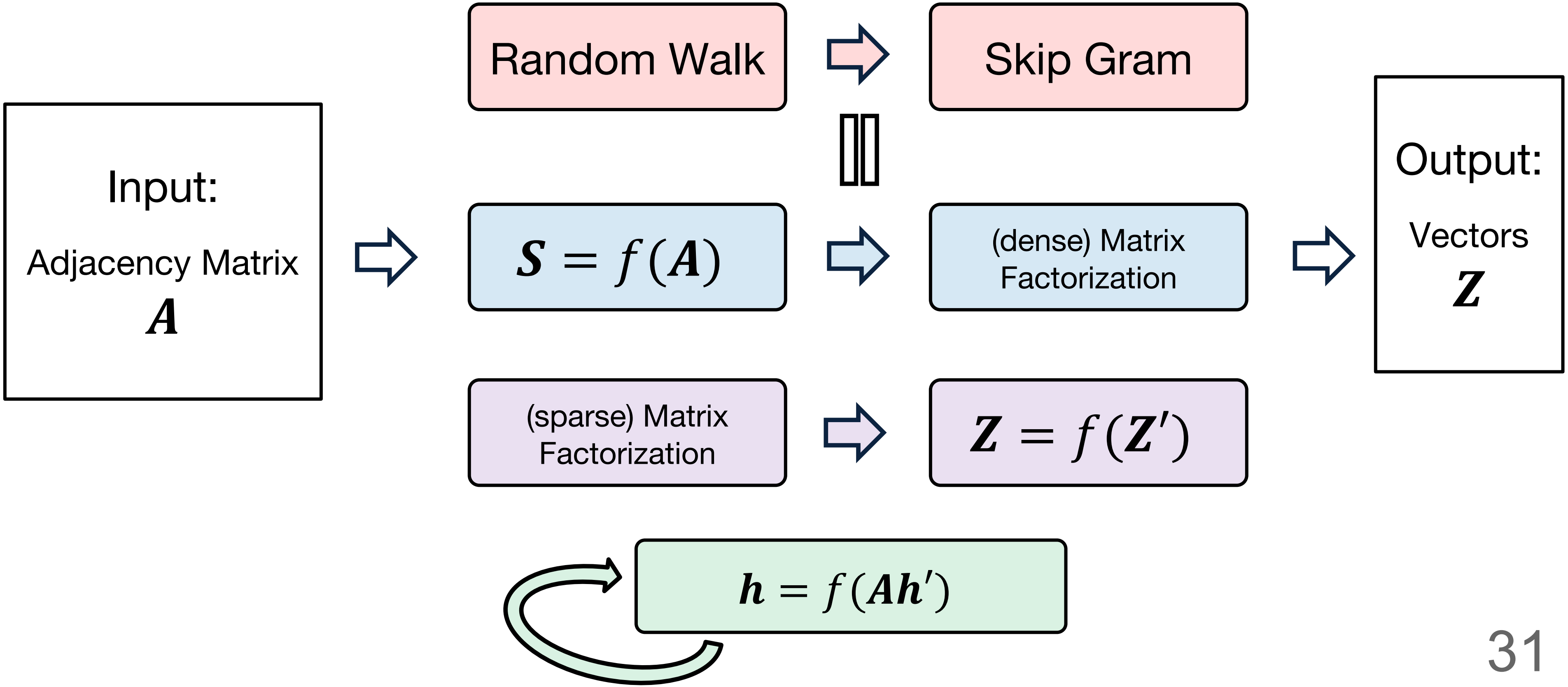
- Techniques:
  Linking embedding with interpretable node attributes, encode knowledge as links, etc.



Key problems of learning from networks

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

Connecting NE with Graph Neural Networks

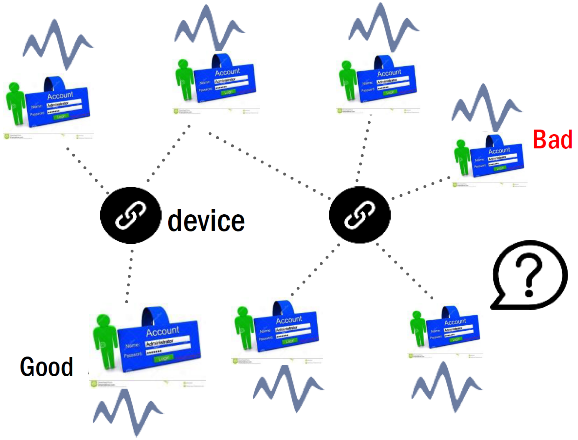


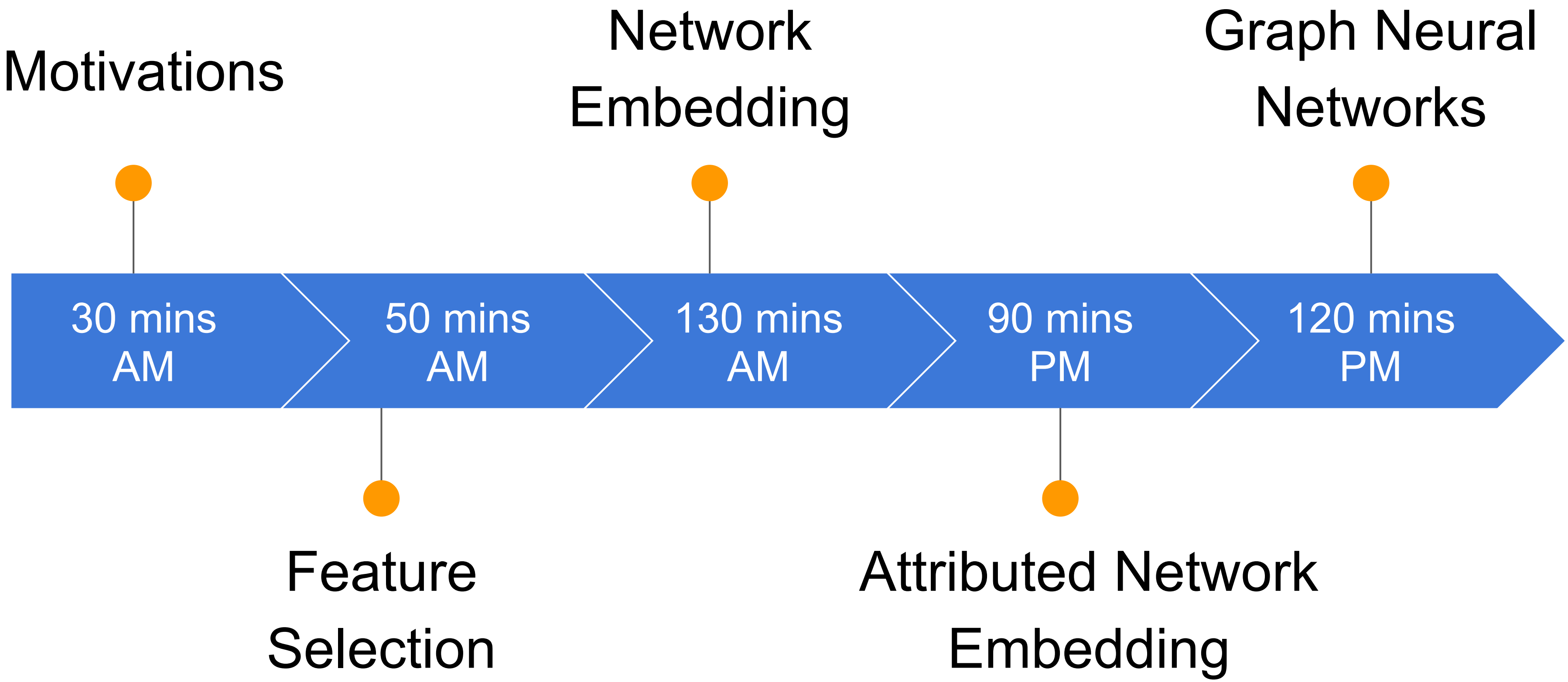
Graph Neural Networks

FastGCNs, Graph attention	2018: Velickovic et al., ICLR'18, Chen et al., ICLR 2018
Neural message passing, GraphSage	2017: Gilmer et al., ICML'17; Hamilton et al., NIPS'17
Gated graph neural network	2016: Li et al., ICLR'16
structure2vec	2016: Dai et al., ICML'16
Graph convolutional network	2015: Duvenaud et al., NIPS'15; Kipf & Welling ICLR'17
Spectral graph convolution	2014: Bruna et al., ICLR'14
Graph neural network	2005: Gori et al., IJCNN'05

Real world applications

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




Motivations
Network Embedding
Graph Neural Networks
30 mins AM
50 mins AM
130 mins AM
90 mins PM
120 mins PM
Feature Selection
Attributed Network Embedding