KDD2021

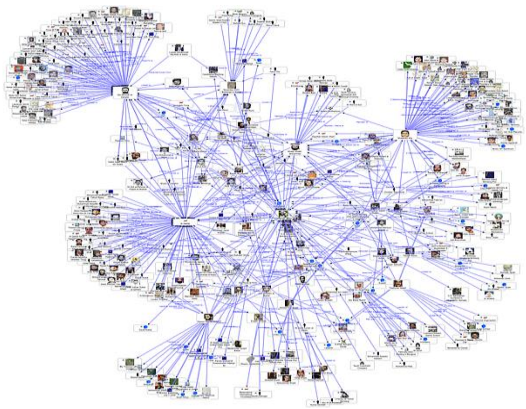
New Frontiers of Multi-Network Mining: Recent Developments and Future Trend



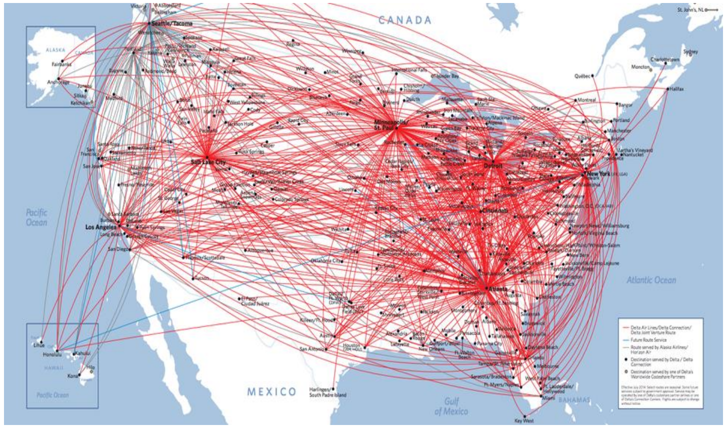

Boxin Du Si Zhang Yuchen Yan Hanghang Tong
University of Illinois at Urbana-Champaign

ILLINOIS

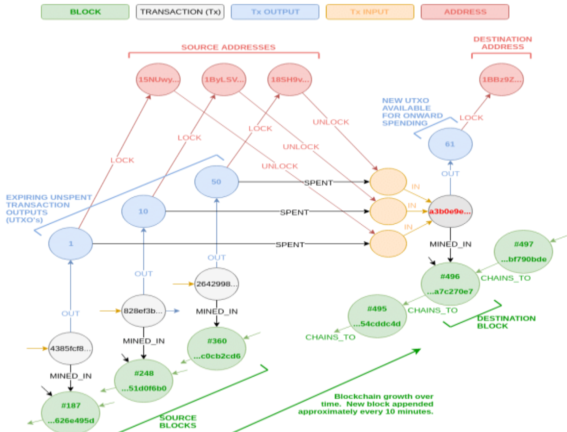
Networks and Graphs Are Everywhere



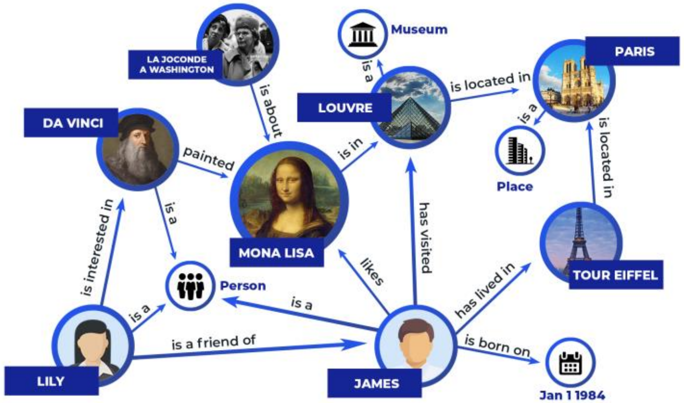
Social network



Transportation network



Bitcoin transaction network



Knowledge graph

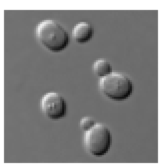
Network = graph in this tutorial

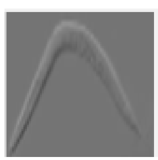
Networks Are Multi-sourced

Social media:

Transaction network:

PPI network:


yeast


elegan

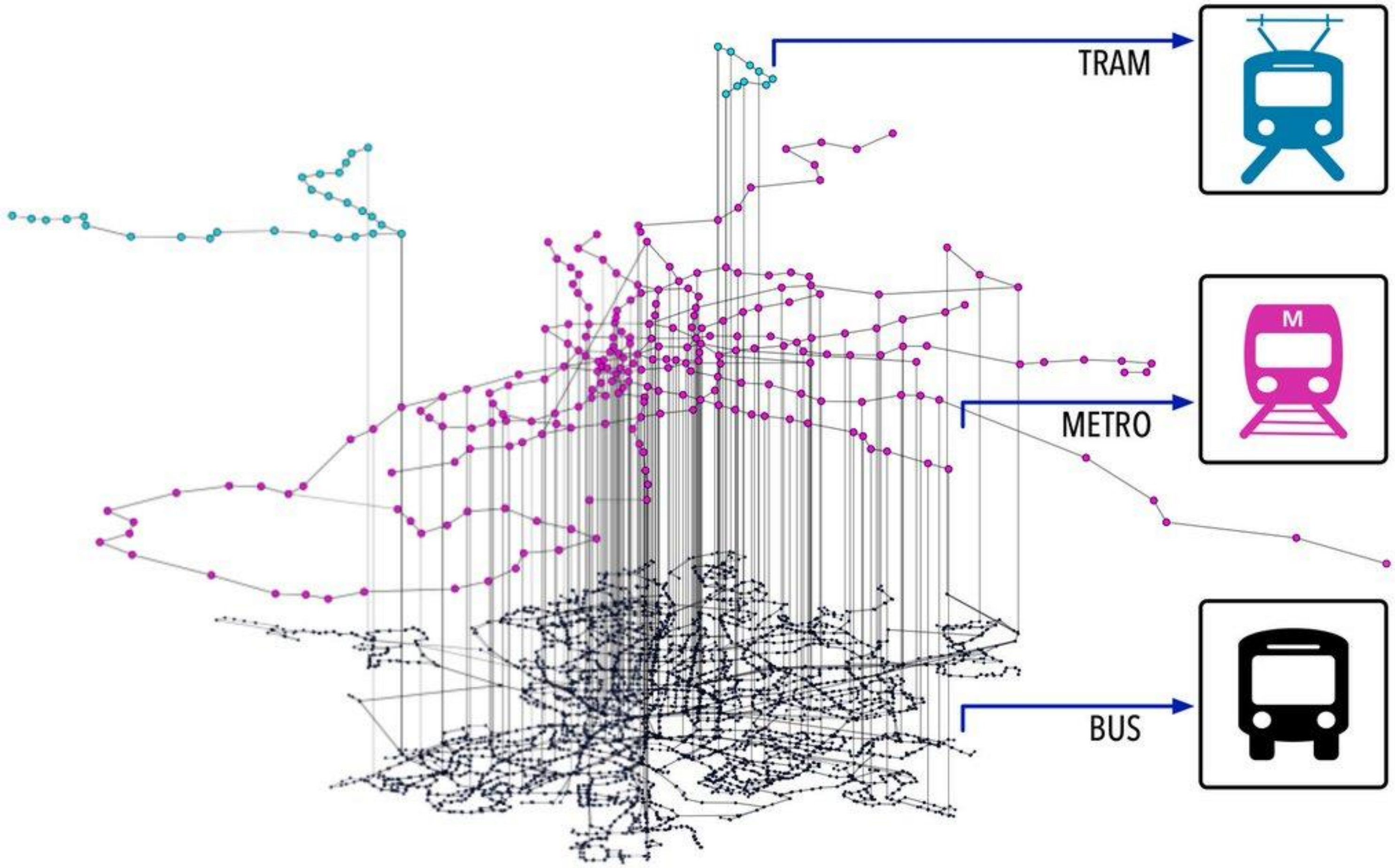

fly


mouse

Knowledge graph:

Example of Multi-networks

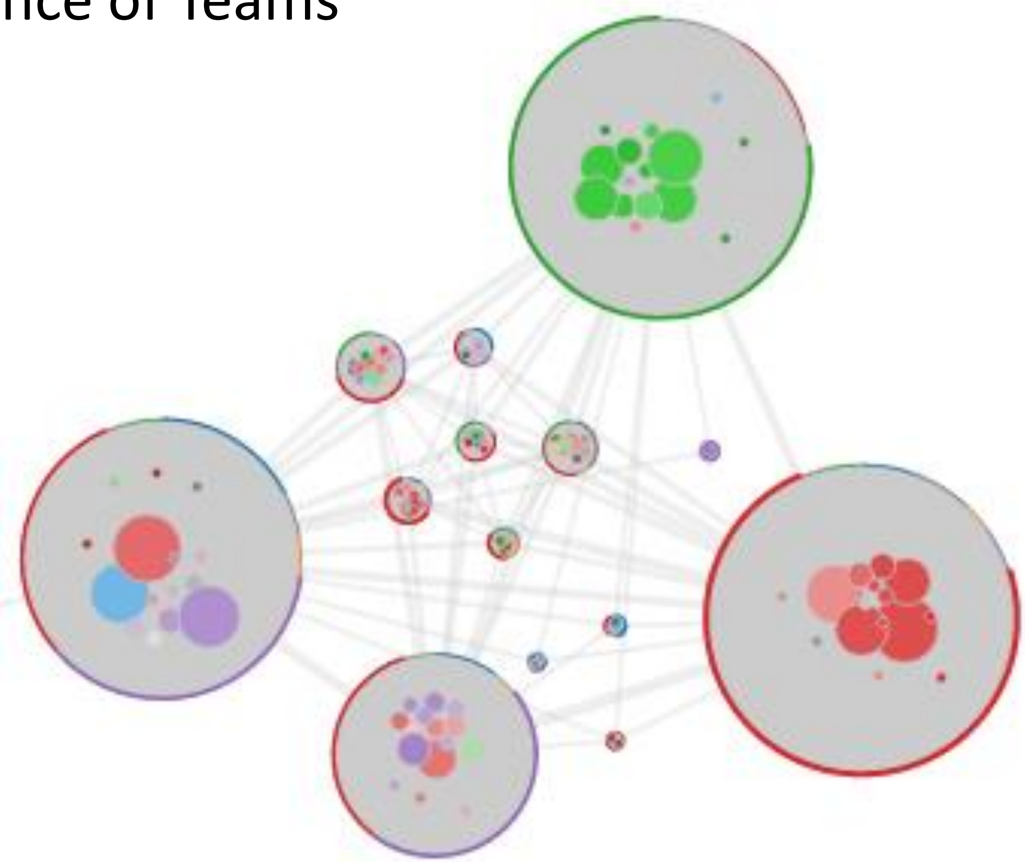
- Inter-dependent traffic networks



[1] Aleta, Alberto, and Yamir Moreno. "Multilayer networks in a nutshell." *Annual Review of Condensed Matter Physics* 10 (2019): 45-62.

Example of Multi-networks

- Network Science of Teams

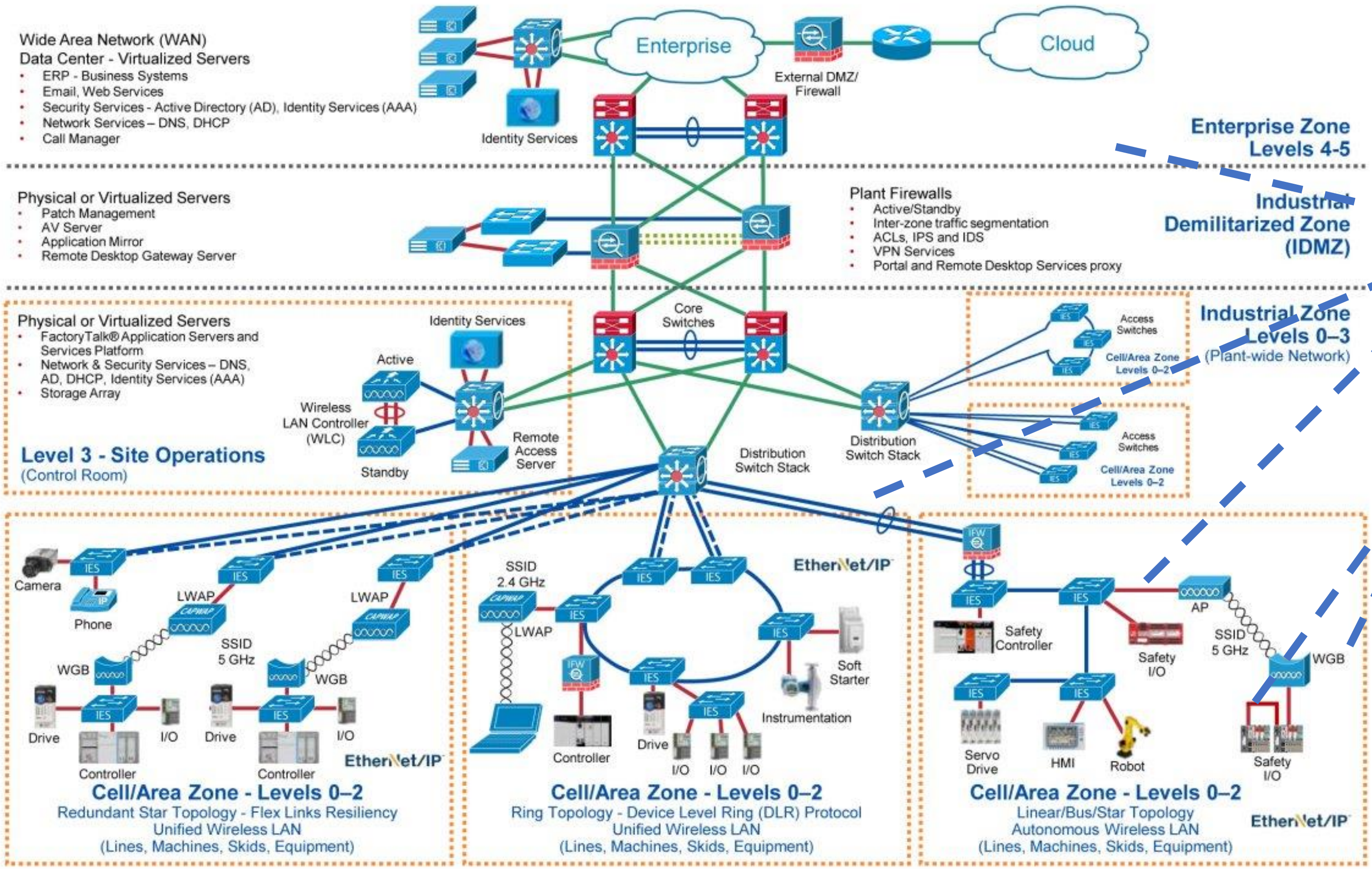


- Set of **team networks**, connected by [a project dependency network](#).

[1] G.S. McChrystal, C. Tantum, S. David, and F. Chris:. Team of teams: New rules of engagement for a complex world. Penguin, 2015.
[2] N. Contractor, L.A. DeChurch, A. Sawant, and X. Li: My Dream Team Assembler, 2013.
[3] W. Stefan, B. Jones, and B. Uzzi: The Increasing Dominance of Teams in the Production of Knowledge. Science, May 2007, 316:1036-1039.
[4] Network Science of Teams Project Website: http://team-net-work.org
[5] Li, L, and H. Tong: Network Science of Teams: Characterization, Prediction, and Optimization. WSDM 2018 tutorial

Example of Multi-networks

- Complex industrial networks:

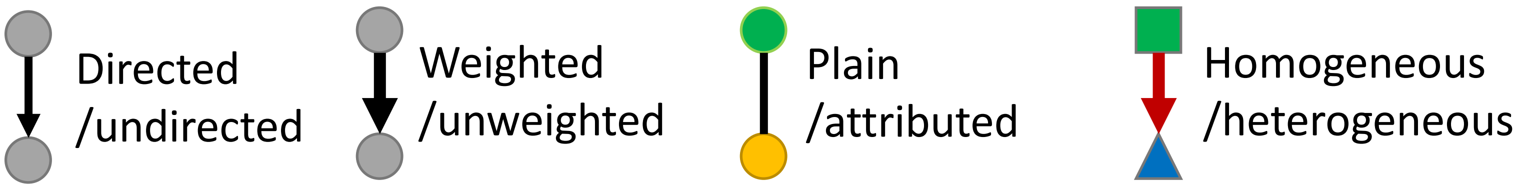


Key characteristics:
- Multiple layers
- Inter-dependent
- Attributed
- Heterogeneous
- …

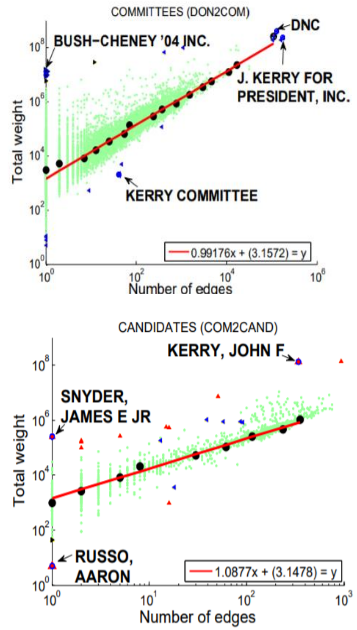
[1] https://www.rockwellautomation.com/en-us/capabilities/industrial-networks/design-guides.html

Single (Simple) Networks

- Basic definition: $\mathcal{G} = (V, E, \mathbf{A})$.
- Optional:



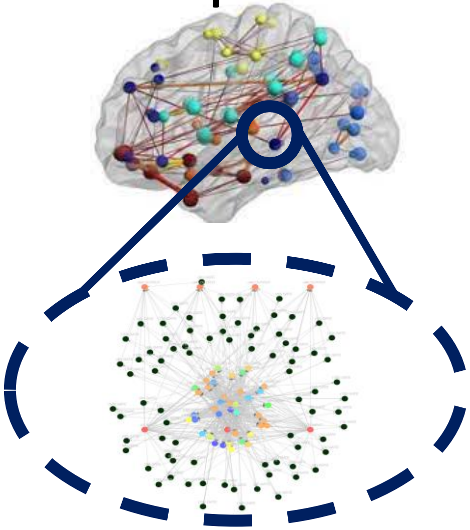
- Local characteristics:
  - Node: indecomposable, single entity
  - Edge: single, pairwise, node-node relationship
- Global characteristics:
  - Statistical properties: degree distribution, centrality, etc.
- Limitations:
  - Node-network, network-network relationship
  - High-order node relationship
  - Node relationship across multiple networks



Different structural distribution

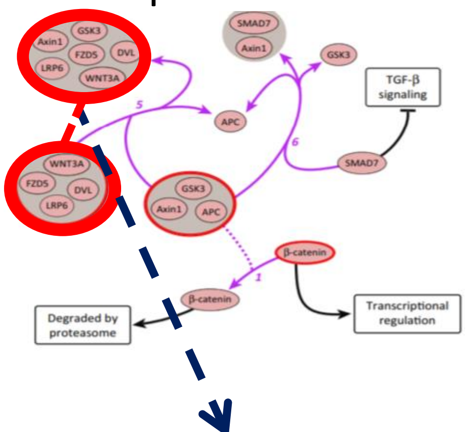
Multi-networks

- Motivation:
  - How to represent complex real-world data as network models?
  - How to handle the limitations of single & regular networks?
- Examples:

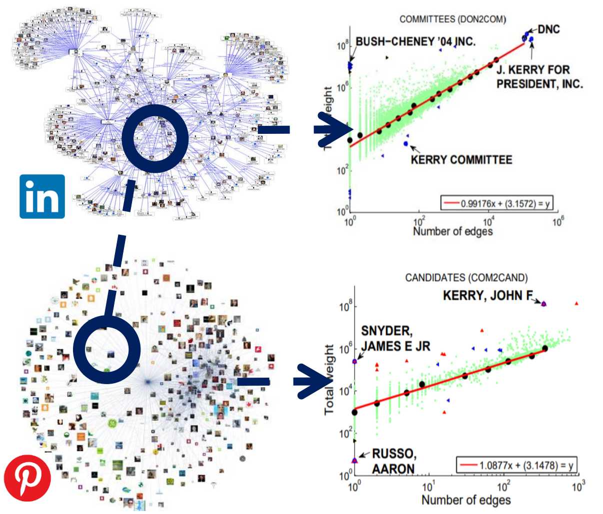


Node-network hierarchical relationship

High-order node relationship



Node set-node set relationship



Relationship of multi-network of different distributions

  - And many more…

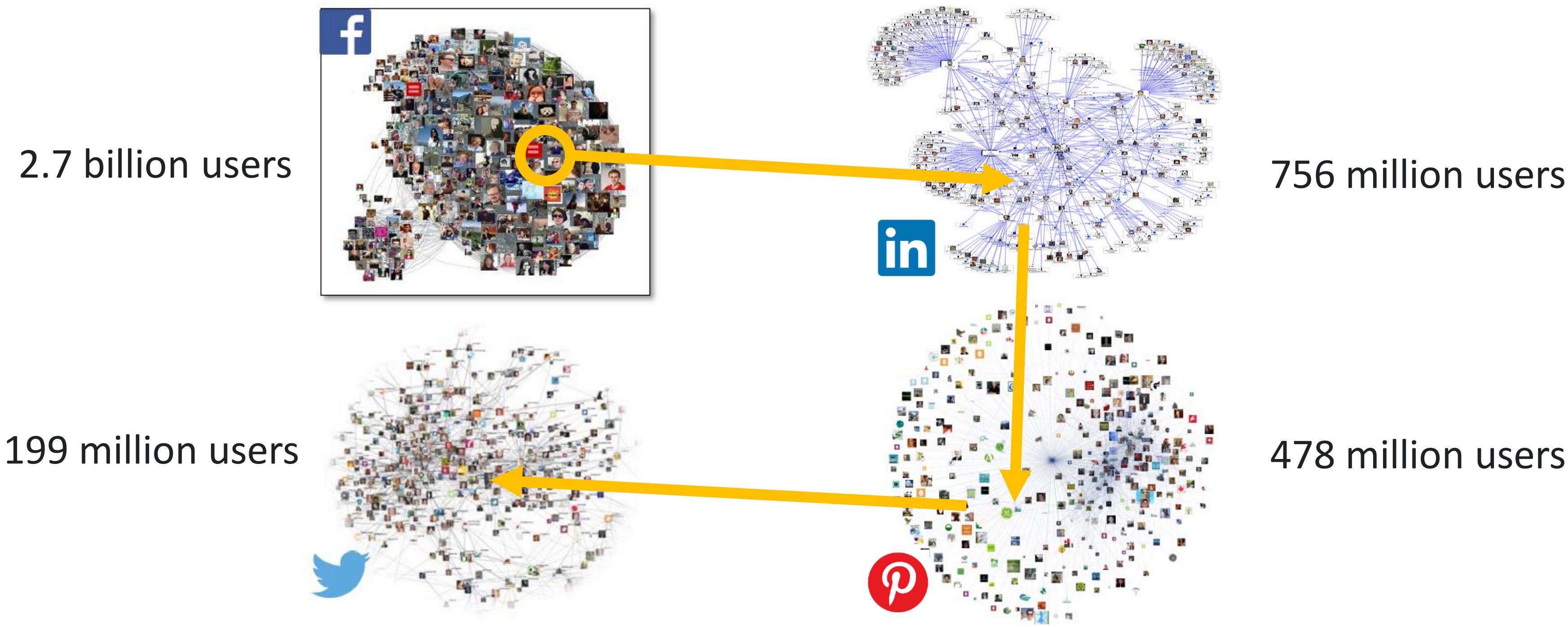
[1] Zhang, Xi, et al. "Multi-view graph convolutional network and its applications on neuroimage analysis for parkinson's disease." *AMIA Annual Symposium Proceedings*. Vol. 2018. American Medical Informatics Association, 2018.
[2] Yadati, Naganand, et al. "HyperGCN: A new method of training graph convolutional networks on hypergraphs." *arXiv preprint arXiv:1809.02589* (2018).

Multi-network Mining Challenges

- **C1. Data challenge**
  - Multi-network data models are more complex
- **C2. Algorithmic challenge**
  - How to solve the multi-network mining problems?
- **C3. Application challenge**
  - How to empower or enable multi-network applications?

C1. Multi-networks Are Complex

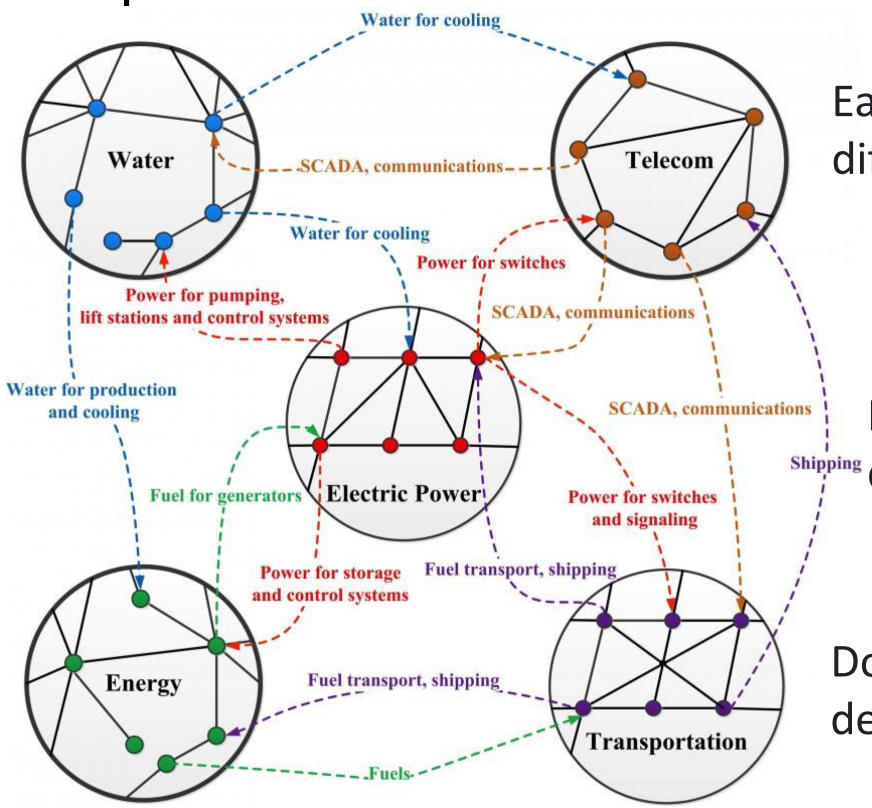
- Volume: the number of nodes/edges is large
- Example:
  - User identity alignment/matching: search several social networks



[1]Zhang, Si, and Hanghang Tong. "Network Alignment: Recent Advances and Future Directions." Proceedings of the 29th ACM International Conference on Information & Knowledge Management. 2020.
[2] Du, Boxin, Lihui Liu, and Hanghang Tong. "Sylvester Tensor Equation for Multi-way Association". SIGKDD (2021)

C1. Multi-networks Are Complex (cont'd)

- Variety: the structure is complicated
- Example:
  - Interdependent infrastructure network



Each domain has a different network

Nodes from different domains are dependent

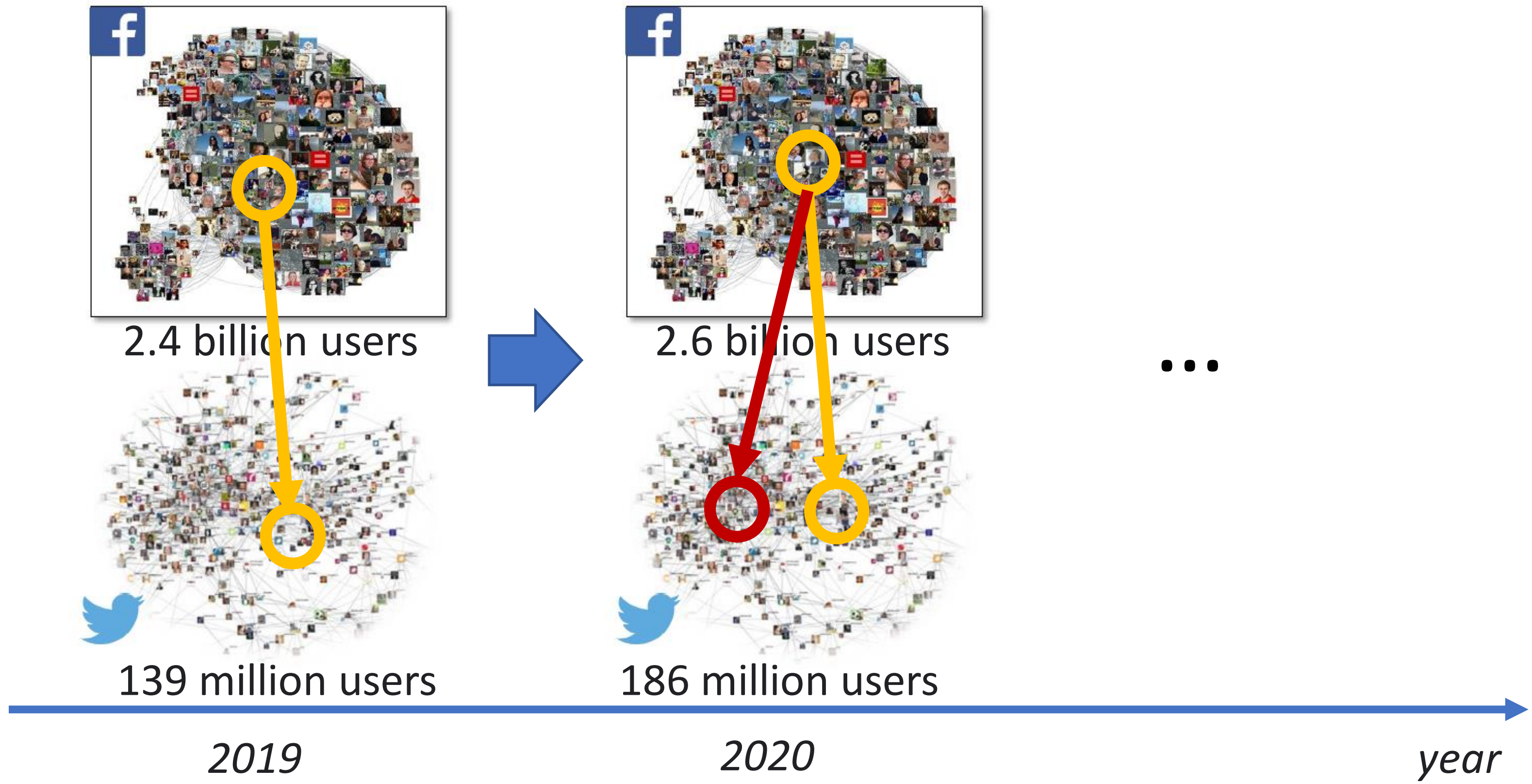
Domain network and inter-network dependence are attributed

[1] C. Chen, J. He, N. Bliss and H. Tong: "Towards Optimal Connectivity on Multi-layered Networks". IEEE Trans. Knowl. Data Eng., 29(10): 2332-2346 (2017)

[2] Gao, Jianxi, Daqing Li, and Shlomo Havlin. "From a single network to a network of networks." *National Science Review* 1.3 (2014): 346-356.

C1. Multi-networks Are Complex (cont'd)

- Velocity: multi-networks are changing dynamically
- Example:

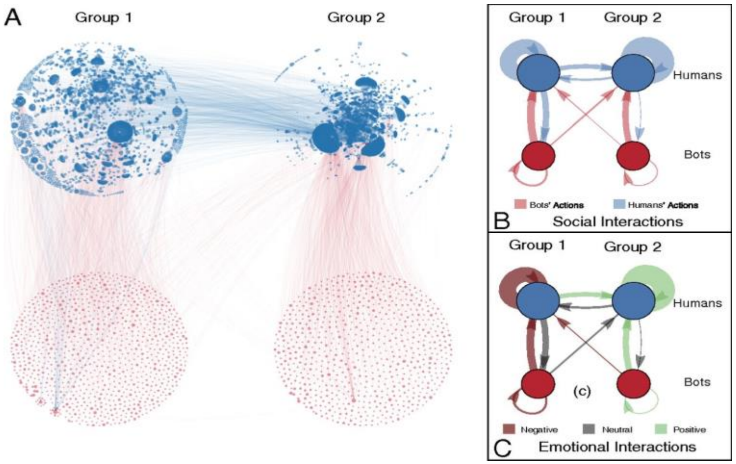


[1] Sun, Li, et al. "Dna: Dynamic social network alignment." *2019 IEEE International Conference on Big Data (Big Data)*. IEEE, 2019.
[2] Vijayan, Vipin, Dominic Critchlow, and Tijana Milenković. "Alignment of dynamic networks." *Bioinformatics* 33.14 (2017): i180-i189.

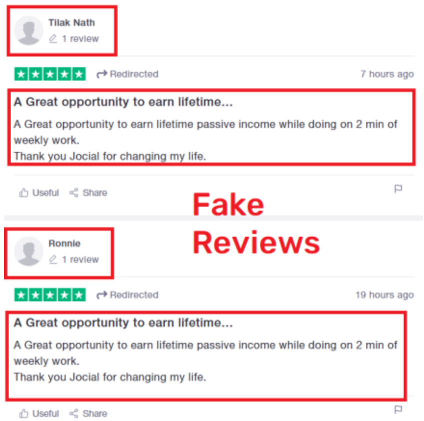
C1. Multi-networks Are Complex (cont'd)

- Veracity: multi-networks are noisy and incomplete
- Example:

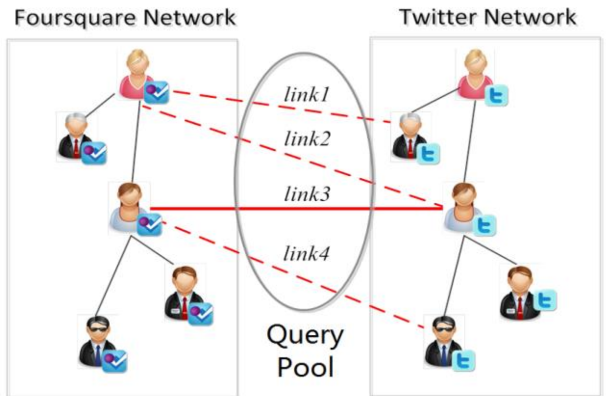
Bot users



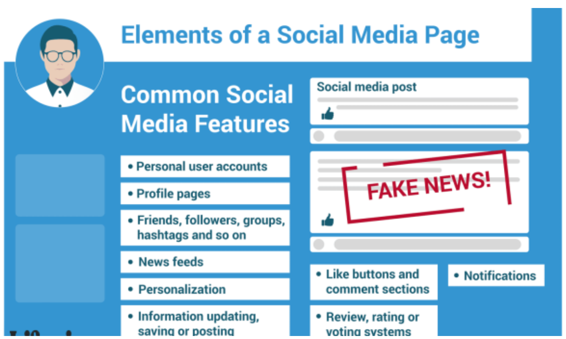
Fake reviews



Missing links



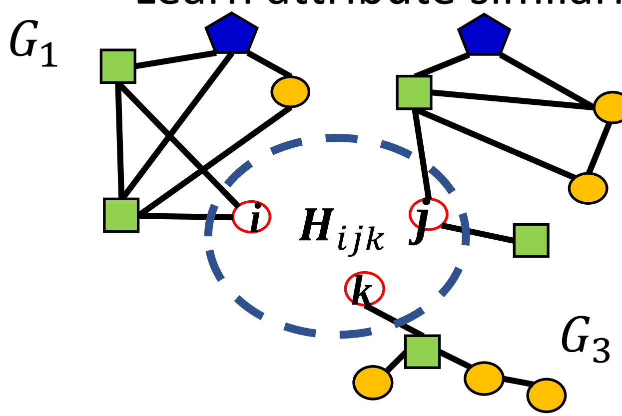
Fake features

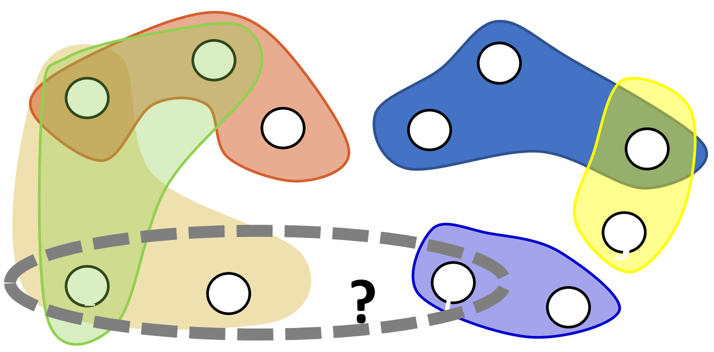


[1] Zhang, Jiawei, Bowen Dong, and S. Yu Philip. "Fakedetector: Effective fake news detection with deep diffusive neural network." *2020 IEEE 36th International Conference on Data Engineering (ICDE)*. IEEE, 2020.
[2] Al Hasan, Mohammad, and Mohammed J. Zaki. "A survey of link prediction in social networks." *Social network data analytics*. Springer, Boston, MA, 2011. 243-275.

C2: Algorithmic Challenge - Prob. Formulation

- How to encode multi-network structure/features?
- Examples:
  - Multi-network alignment (cross-network node mapping):
    - Encode topological similarities
    - Encode node/edge attribute similarities
  - Hypergraph link prediction (high-order node relationship):
    - Learn topological similarities of high-order objects
    - Learn attribute similarities of high-order objects

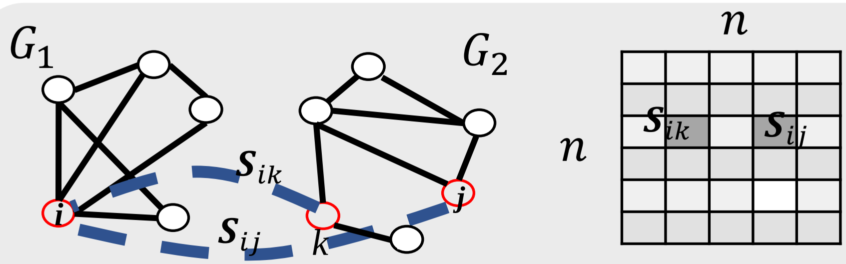




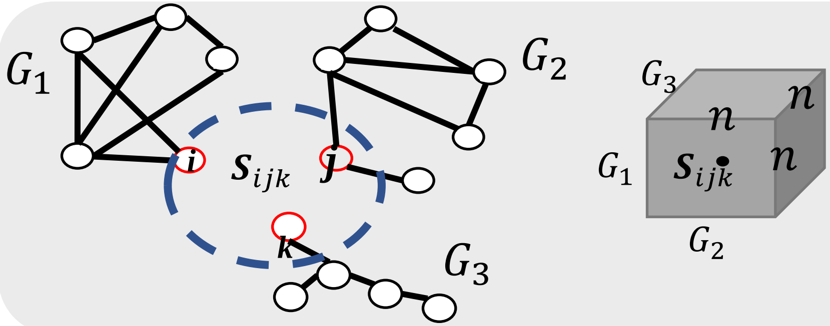
[1] Du, Boxin, Lihui Liu, and Hanghang Tong. "Sylvester Tensor Equation for Multi-way Association". SIGKDD (2021)
[2] Zhang, Muhan, Zhicheng Cui, Shali Jiang, and Yixin Chen. "Beyond link prediction: Predicting hyperlinks in adjacency space." In Thirty-Second AAAI Conference on Artificial Intelligence. 2018.

C2: Algorithmic Challenge - Complexity

- How to handle high time/space complexity?
- Example:
  - Multi-network association (cross-network node proximity):



IsoRank:
Time: $O(\#iter * (m^2+n^2))$
Space: $O(n^2)$



Fixed-point iteration:
Time: $O(\#iter * (m^3+n^3))$
Space: $O(n^3)$

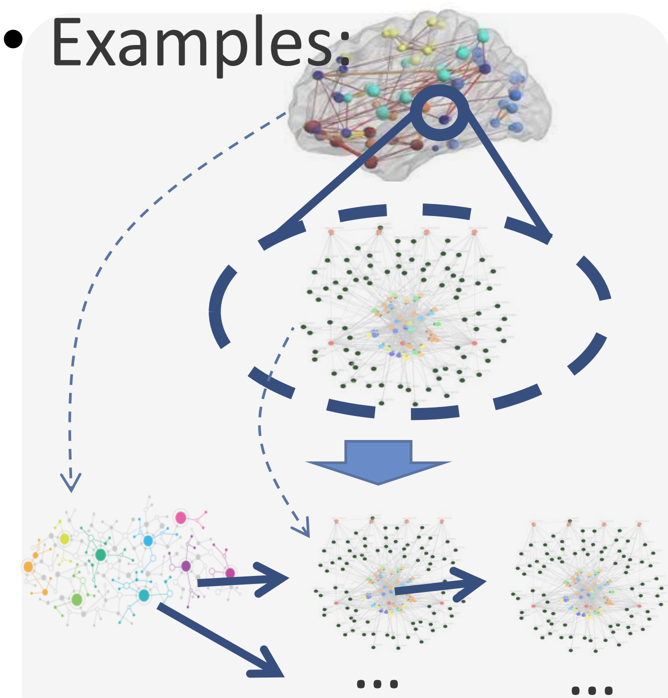
[1] Bahmani, Bahman, Abdur Chowdhury, and Ashish Goel. "Fast incremental and personalized pagerank." *arXiv preprint arXiv:1006.2880* (2010).
[2] Singh, Rohit, Jinbo Xu, and Bonnie Berger. "Pairwise global alignment of protein interaction networks by matching neighborhood topology." *Annual International Conference on Research in Computational Molecular Biology*. Springer, Berlin, Heidelberg, 2007.

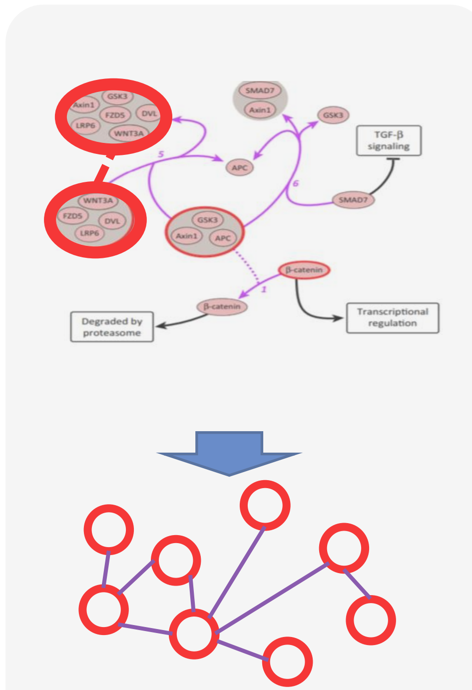
C2: Algorithmic Challenge - Computation

- Multi-network -> single (simple) network?
  - Transformation itself is non-trivial
  - Information loss
  - Potentially increase the complexity

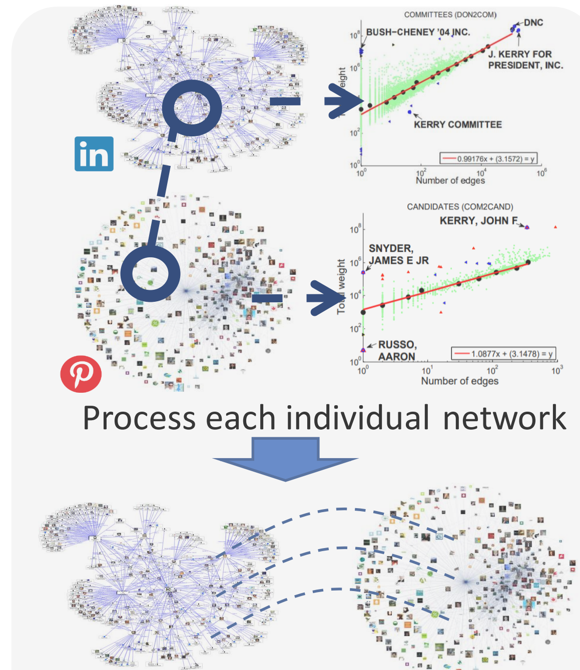
May not work!

- Examples:


Non-trivial transformation



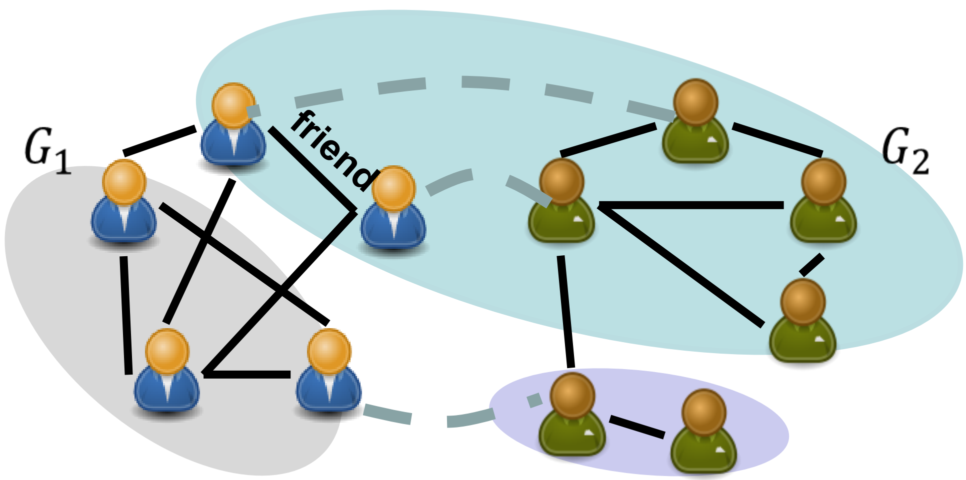
Node collection info loss



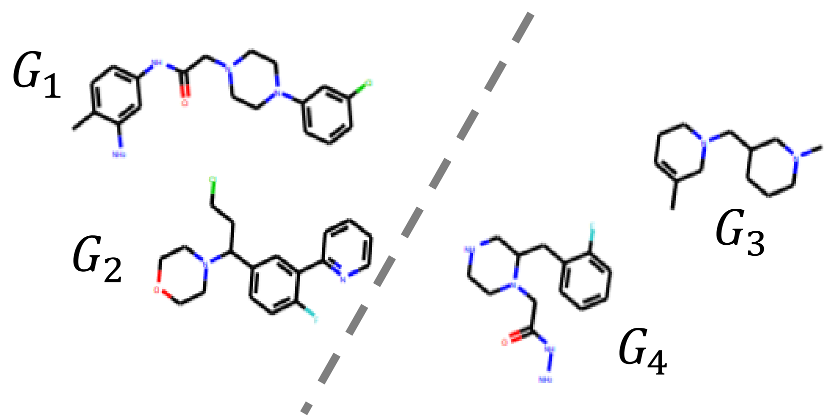
Large time/space complexity

C3. Application Challenge

- How to empower or enable multi-network applications?
- Example:
  - Ranking/clustering on multiple regular networks
  - Classification on complex multi-network models



Multi-network clustering

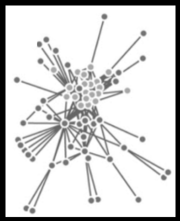
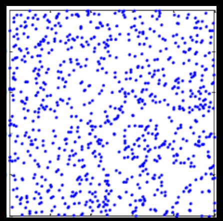


Multi-network classification

Multi-Networks: Where Are We?

Single (Simple) Networks

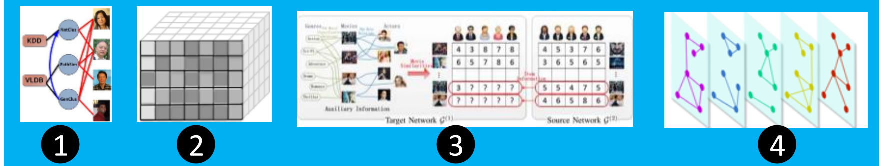
- **At Macro-level:** Nodes Linked by Edges (e.g., an Adjacency Matrix)

- **At Micro-level:** Node/Link as Atom

Beyond a Single Adj. Matrix

1. HIN [Sun & Han 2012-2020]
2. Tensor [Faloutsos+ 2008-2019]
3. HSN [Yu+ 2013-2019]
4. Multiplex [Kanawati 2015, Porter 2014]

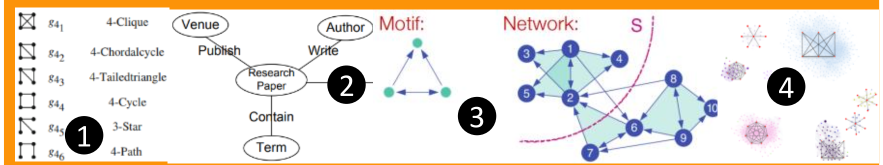


At Macro-level: Set of Networks Connected by Another Network

Key Advantages: Hierarchical Modeling, Towards Multi-Network Model Unification.

High-order Structure

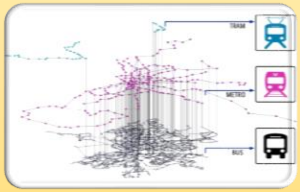
1. Graphlet [Neville+ 2016]
2. Meta-Path/Structure [Sun & Han 2012]
3. Motif/High-order Structure [Benson+ 2016]
4. Graph Vocabulary [Koutra, Faloutsos+ 2014]



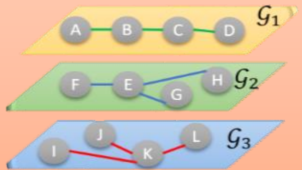
This Tutorial: Multi-Networks

- **At Micro-level:** Hidden Networks Deep Inside a Node/Link
- **Key Advantages**: Collective Mining with Focused Knowledge Transfer

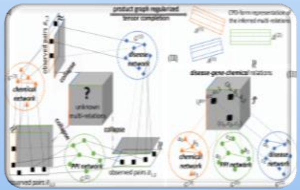
Roadmap



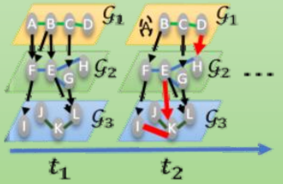
Introduction



Part I: Multi-network Data Models



Part II: Multi-network Mining Algorithms



Part III: Multi-network Future Directions

Overview of Part I

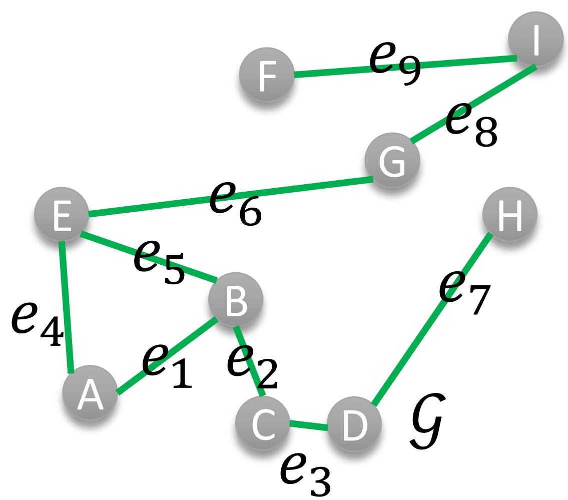
- Multi-network Data Models
  - Multi-view Network
  - Multiplex Network
  - Inter-dependent Network
  - Hypergraph
  - Network of Networks

Network: Definition

- Definition of network
  - Basic: $\mathcal{G} = (V, E, \mathbf{A})$.
  - V: node set, E: edge set, $\mathbf{A}$: adjacency matrix of the network.
  - Optional: node attribute matrix $\mathbf{X}$, edge attribute matrix $\mathbf{Y}$.

$\mathbf{X}$

A	1	1	1
B	1	0	0
C	1	1	0
D	0	1	0
E	1	0	1
F	0	1	1
G	1	0	1
H	0	0	1
I	1	0	0

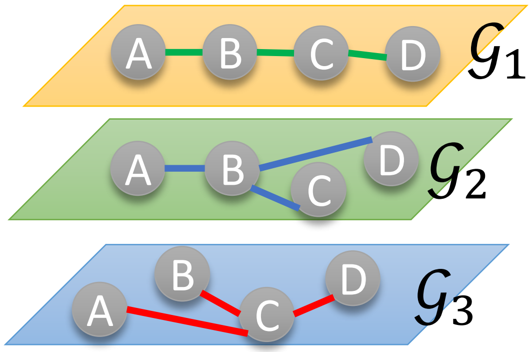


$\mathbf{Y}$

e_1	0	0	1	1
e_2	1	0	1	1
e_3	0	1	1	0
e_4	1	0	0	1
e_5	0	1	1	0
e_6	1	1	0	1
e_7	0	1	1	1
e_8	1	1	0	0
e_9	1	1	1	0

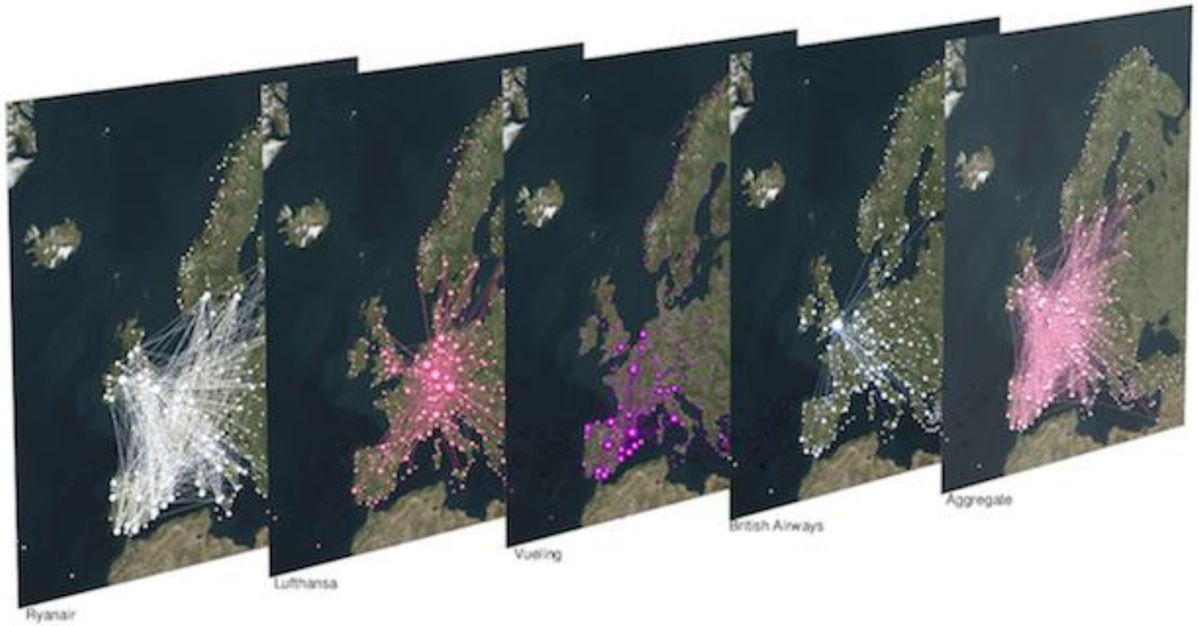
Multi-view Network: Definition

- A.k.a. multi-relation network, or multi-dimension network
- Definition of multi-view network
  - $\mathcal{G}_i = (V, E_i, \mathbf{A}_i)$.
  - For the same set of nodes, their relations can be formed from different views/aspects.
  - V: node set, E_i and $\mathbf{A}_i$: edge set and adjacency matrix of the i-th view network.
  - It can be represented as a tensor with a size $|V| \times |V| \times I$.



[1] Kanawati, Rushed. "Multiplex Network Mining: A Brief Survey." IEEE Intell. Informatics Bull. 16.1 (2015): 24-27.

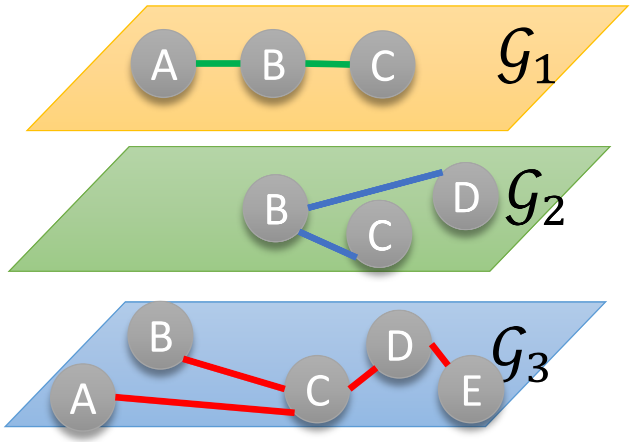
Multi-view Network: Application



Aircraft flow monitoring from different airlines in Europe.

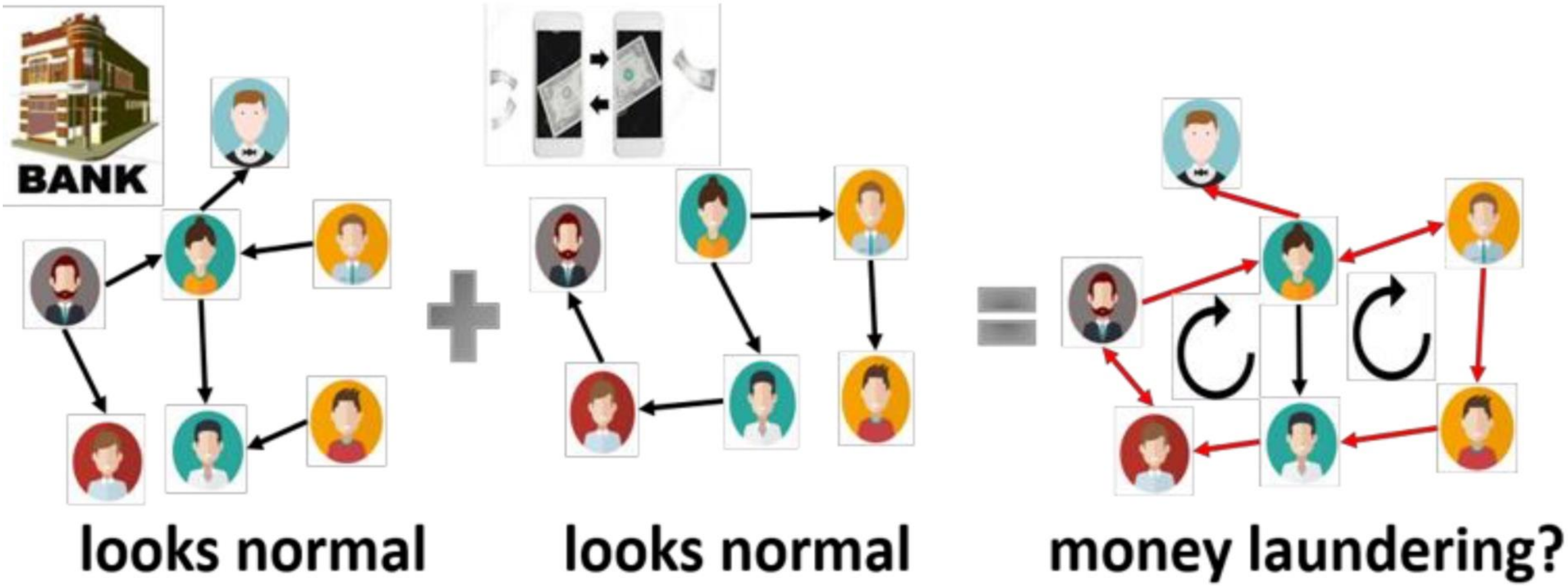
Multiplex Network: Definition

- Definition of multiplex network
  - $\mathcal{G}_i = (V_i, E_i, \mathbf{A}_i)$.
  - V_i: nodes for the i-th network, E_i: edges for the i-th network.
  - $\mathbf{A}_i$: the i-th network's adjacency matrix.
  - $V_i \cap V_j \neq \emptyset$. V_i and V_j have some common nodes.
- Multi-view network is a special case of multiplex network.



[1] Kivelä, Mikko, et al. "Multilayer networks." Journal of complex networks 2.3 (2014): 203-271.

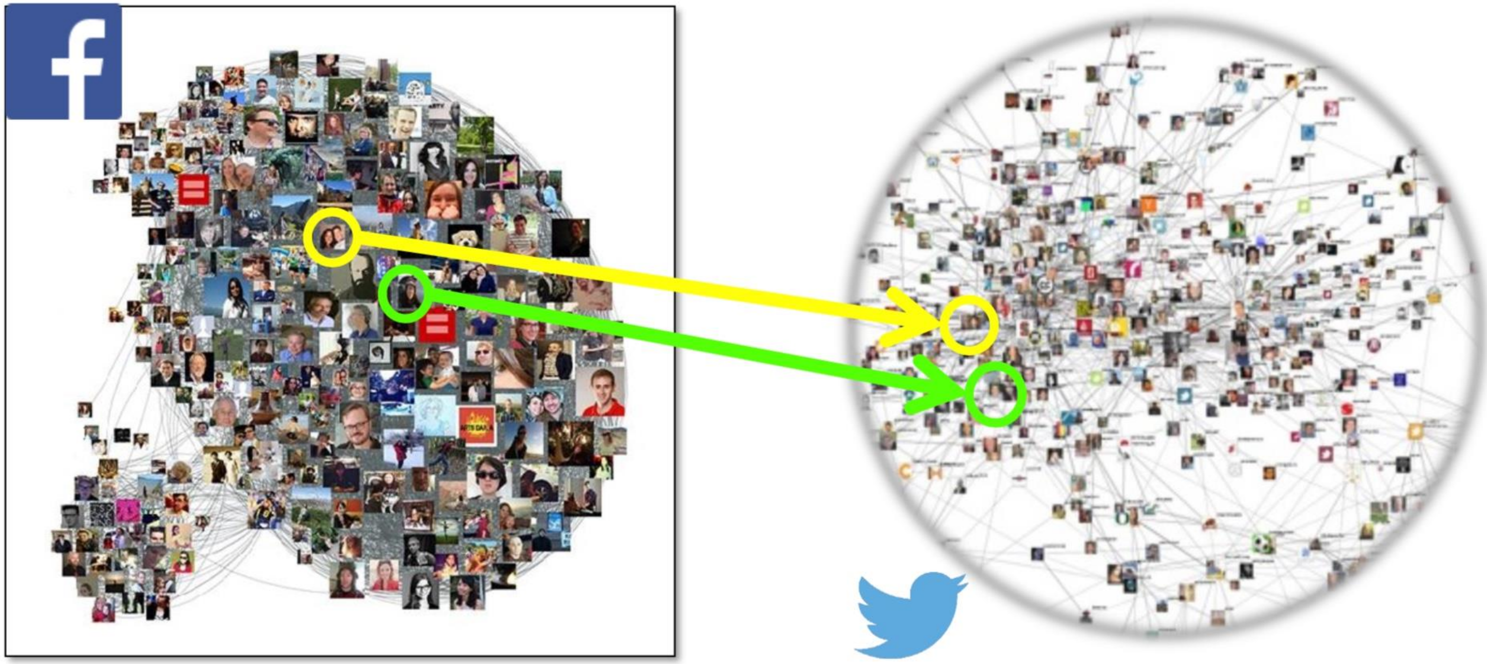
Multiplex Network: Applications



Fraud detection in the economy domain.

[1] Zhang, Si, and Hanghang Tong. "Network Alignment: Recent Advances and Future Directions." *Proceedings of the 29th ACM International Conference on Information & Knowledge Management.* 2020.

Multiplex Network: Applications

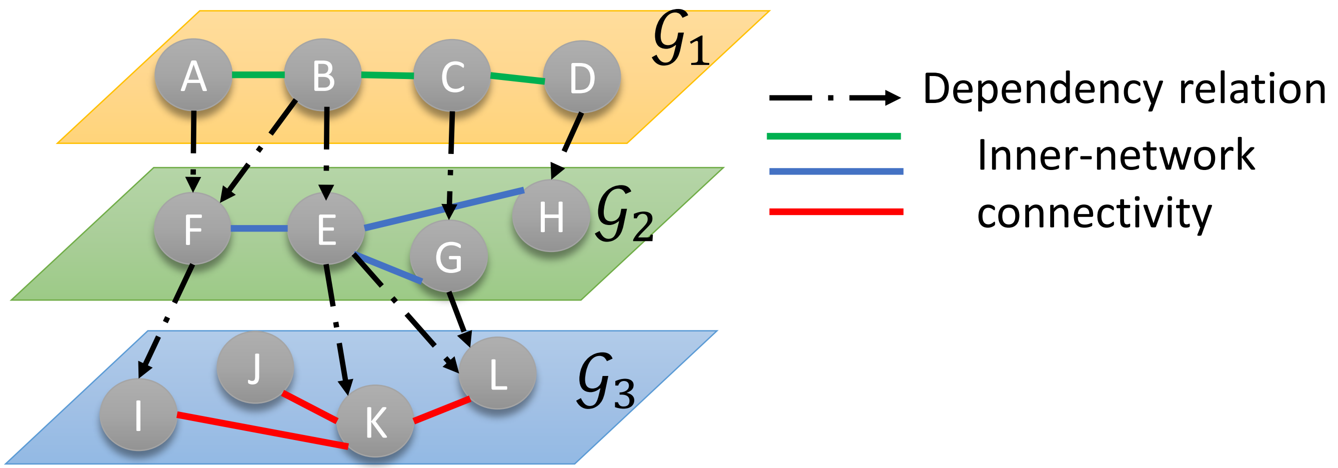


Friend recommendation.

[1] Zhang, Si, and Hanghang Tong. "Network Alignment: Recent Advances and Future Directions." *Proceedings of the 29th ACM International Conference on Information & Knowledge Management.* 2020.

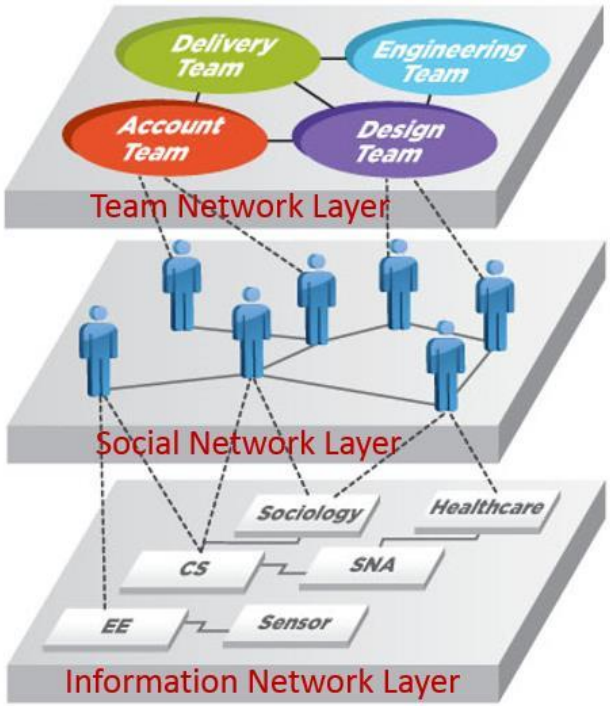
Inter-Dependent Network: Definition

- Definition of Inter-dependent network
  - $\mathcal{G}_i = (V_i, E_i, \mathbf{A}_i)$.
  - $\mathrm{G}^{(d)}$ is the graph level dependency matrix.
  - $\mathrm{G}_{ij}^{(d)} = 1$: we have a node-level dependency matrix $\mathbf{D}^{(ij)}$.
  - $\mathbf{D}_{n_1 n_2}^{(ij)} = 1$: the n_1-th node in $\mathcal{G}_i$ depends on the n_2-th node in $\mathcal{G}_j$.

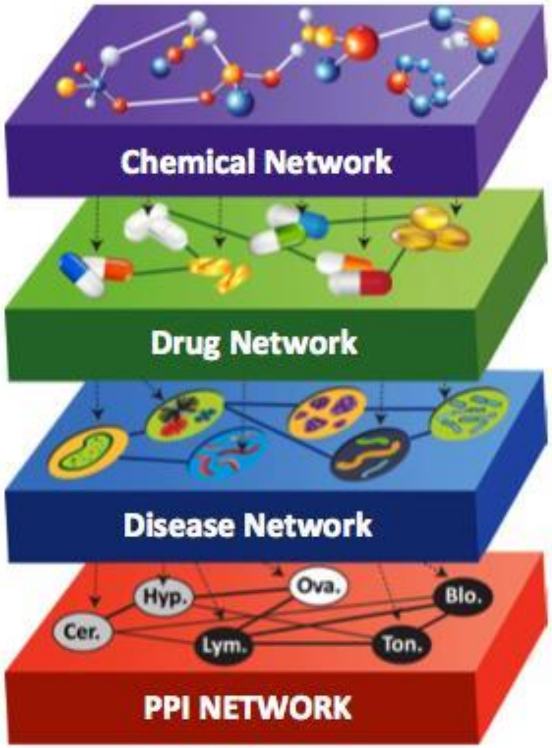


[1] Chen Chen, Hanghang Tong, Lei Xie, Lei Ying, and Qing He. 2017. Cross-Dependency Inference in Multi-Layered Networks: A Collaborative Filtering Perspective. ACM Trans. Knowl. Discov. Data 11, 4, Article 42 (August 2017), 26 pages. DOI:https://doi.org/10.1145/3056562

Inter-Dependent Network: Applications



Team recommendation in collaboration platforms.

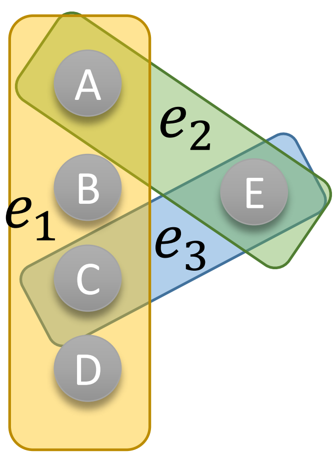


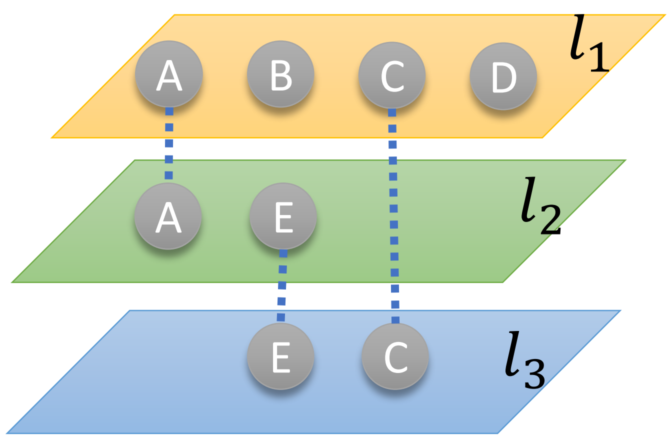
Drug discovery in bio-system.

[1] Chen Chen, Hanghang Tong, Lei Xie, Lei Ying, and Qing He. 2017. Cross-Dependency Inference in Multi-Layered Networks: A Collaborative Filtering Perspective. ACM Trans. Knowl. Discov. Data 11, 4, Article 42 (August 2017), 26 pages. DOI:https://doi.org/10.1145/3056562

Hypergraphs: Definition

- Undirected hypergraphs
  - $\mathcal{G}_u = (V_u, E_u, \mathbf{H}_u)$.
  - V_u: node set, E_u: hyperedge set, $\mathbf{H}_u$: incidence matrix.
  - Simple undirected network: 1-to-1 relation.
  - Undirected hypergraph: n-to-m relation.
  - Multi-layered network degenerates to undirected hypergraph:
    - Do not have within domain networks



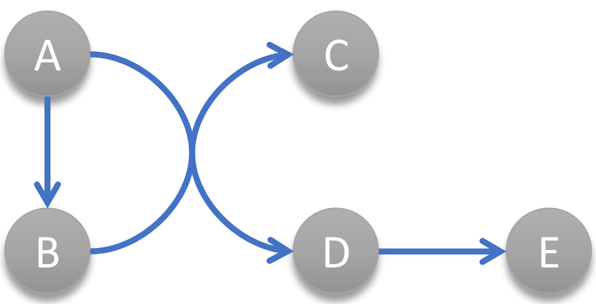


	e_1	e_2	e_3
A	1	1	0
B	1	0	0
C	1	0	1
D	1	0	0
E	0	1	1

[1] Klamt, Steffen, Utz-Uwe Haus, and Fabian Theis. "Hypergraphs and cellular networks." *PLoS computational biology* 5.5 (2009): e1000385.

Hypergraphs: Definition

- Directed hypergraphs
  - $\mathcal{G}_d = (V_d, E_d, \mathbf{H}_d)$.
  - V_d: node set, E_d: hyperedge set, $\mathbf{H}_d$: incidence matrix.
  - Direction between every pair of hyperedge.
  - Simple directed network: 1-to-1 directed relation.
  - Directed hypergraph: n-to-m directed relation.



Directed hyperedges:

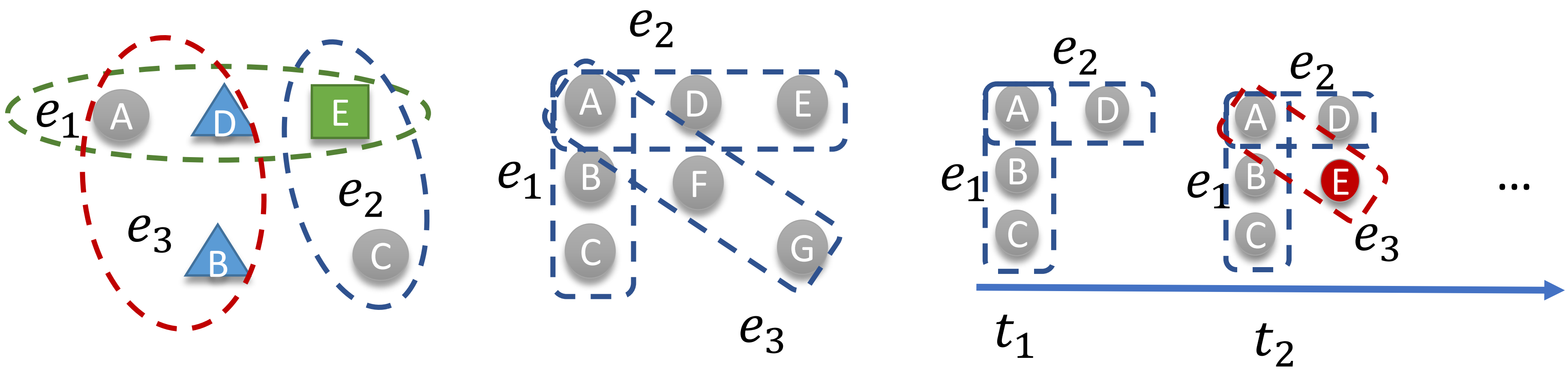
- e_1: $A \to B$
- e_2: $A + B \to C + D$
- e_3: $D \to E$

	e_1	e_2	e_3
A	-1	-1	0
B	1	-1	0
C	0	1	0
D	0	1	-1
E	0	0	1

[1] Klamt, Steffen, Utz-Uwe Haus, and Fabian Theis. "Hypergraphs and cellular networks." *PLoS computational biology* 5.5 (2009): e1000385.

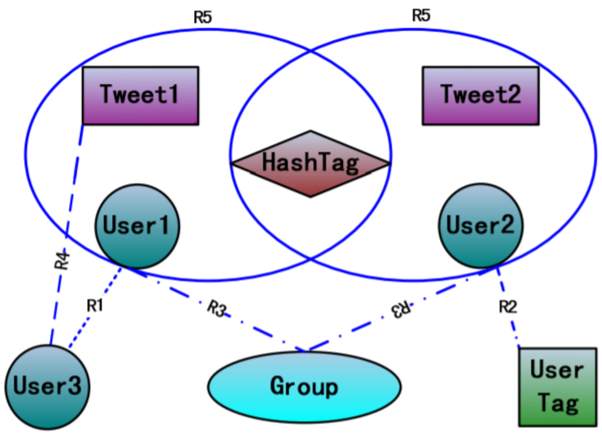
Hypergraphs: Definition

- Heterogeneous hypergraphs:
  - Nodes/hyperedges of different types.
- K-uniform hypergraphs:
  - Every hyperedge contains K nodes.
- Dynamic hypergraphs
  - Time-evolving hypergraphs with changing nodes/hyperedges.

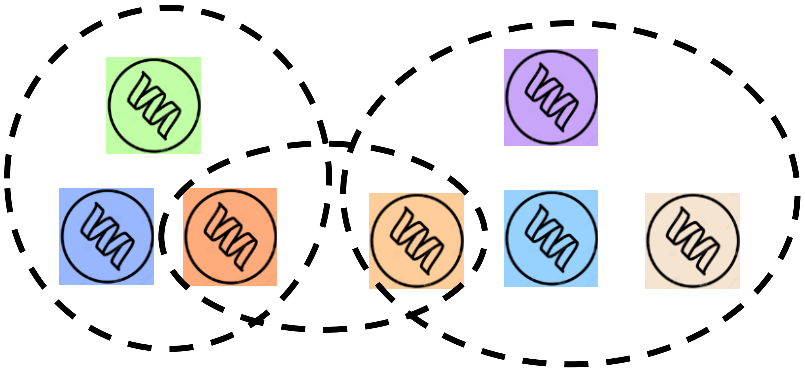


[1] Klamt, Steffen, Utz-Uwe Haus, and Fabian Theis. "Hypergraphs and cellular networks." *PLoS computational biology* 5.5 (2009): e1000385.
[2] Tu, Ke, et al. "Structural deep embedding for hyper-networks." *Thirty-Second AAAI Conference on Artificial Intelligence*. 2018.
[3] Jiang, Jianwen, et al. "Dynamic Hypergraph Neural Networks." *IJCAI*. 2019.

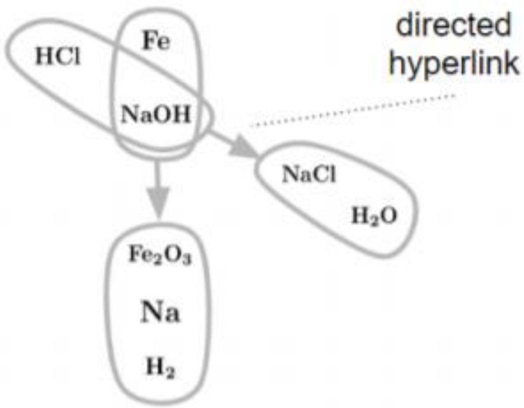
Hypergraph: Applications



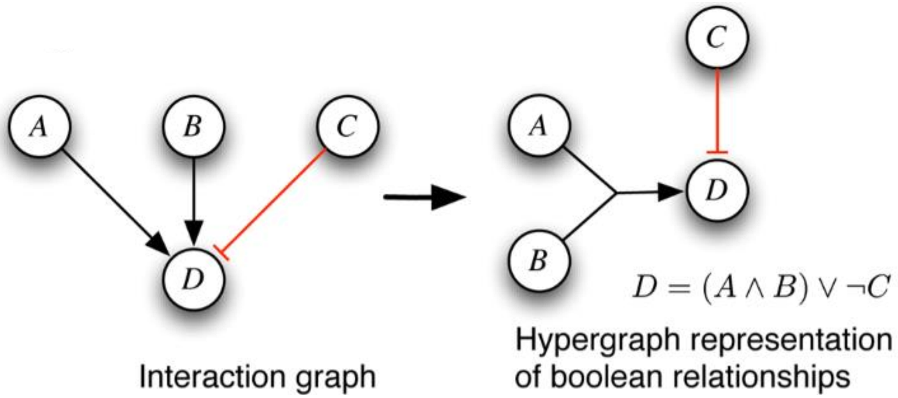
Heterogeneous hypergraph in social networks



Protein complexes by tandem affinity purification (TAP) on PPI network



Directed chemical reaction

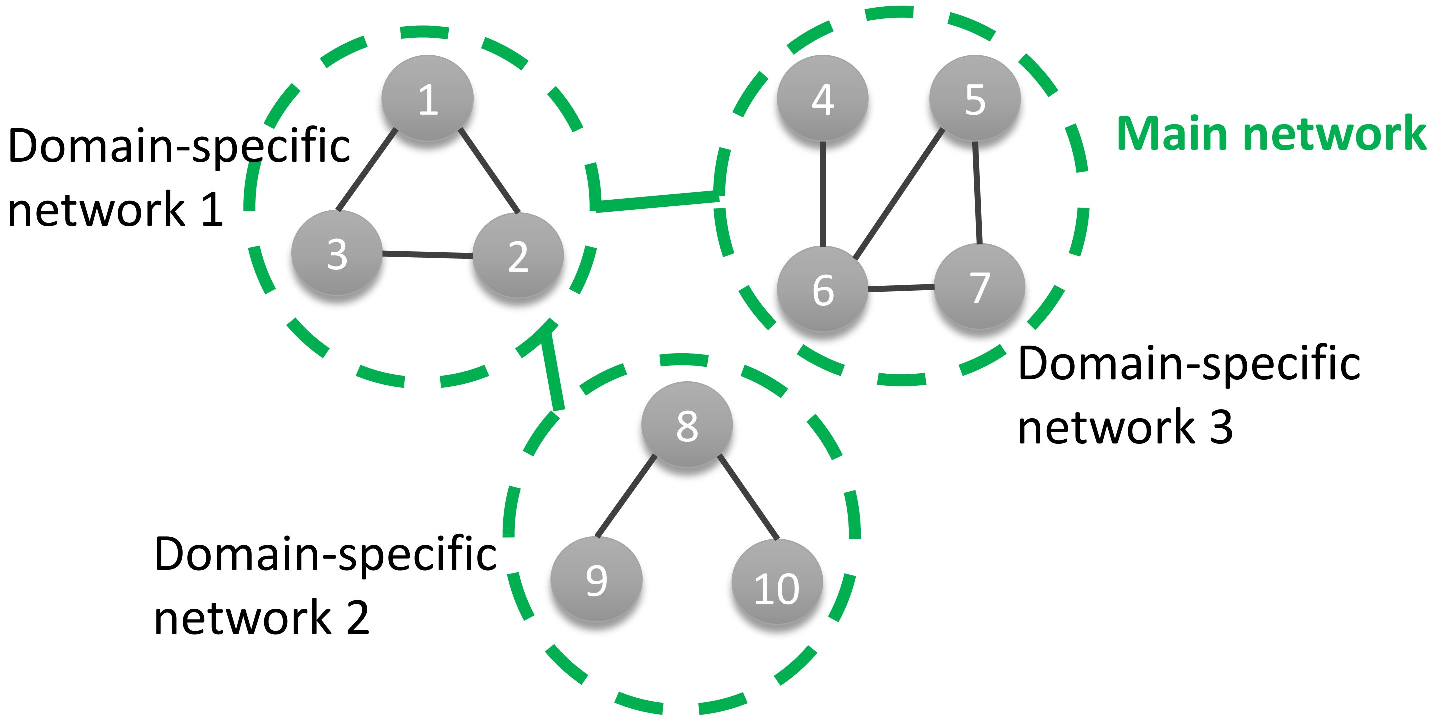


Logical networks

[1] Klamt, Steffen, Utz-Uwe Haus, and Fabian Theis. "Hypergraphs and cellular networks." *PLoS computational biology* 5.5 (2009): e1000385.
[2] Tu, Ke, et al. "Structural deep embedding for hyper-networks." *Thirty-Second AAAI Conference on Artificial Intelligence*. 2018.
[3] Li, Dong, et al. "Link prediction in social networks based on hypergraph." *Proceedings of the 22nd International Conference on World Wide Web*. 2013.

Network of Networks (NoN): Definition

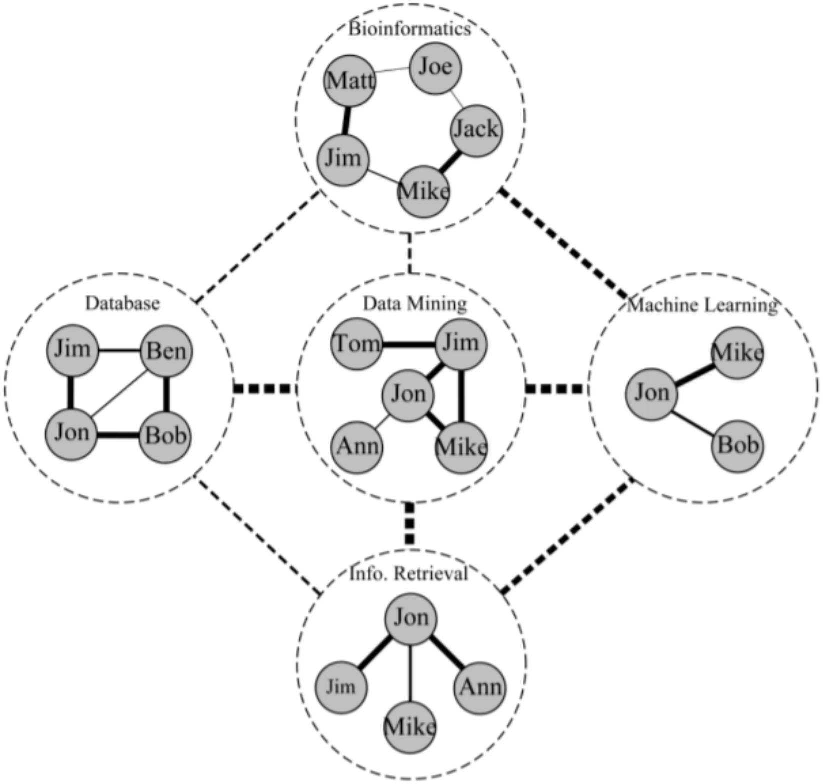
- Definition of NoN:
  - Main network: $G^{(m)} = \{V^{(m)}, E^{(m)}, \mathbf{A}^{(m)}\}$.
  - Domain-specific network: $\left\{G_i^{(d)} = \{V_i^{(d)}, E_i^{(d)}, \mathbf{A}_i^{(d)}\}\right\}, i = 1, \ldots, g$.



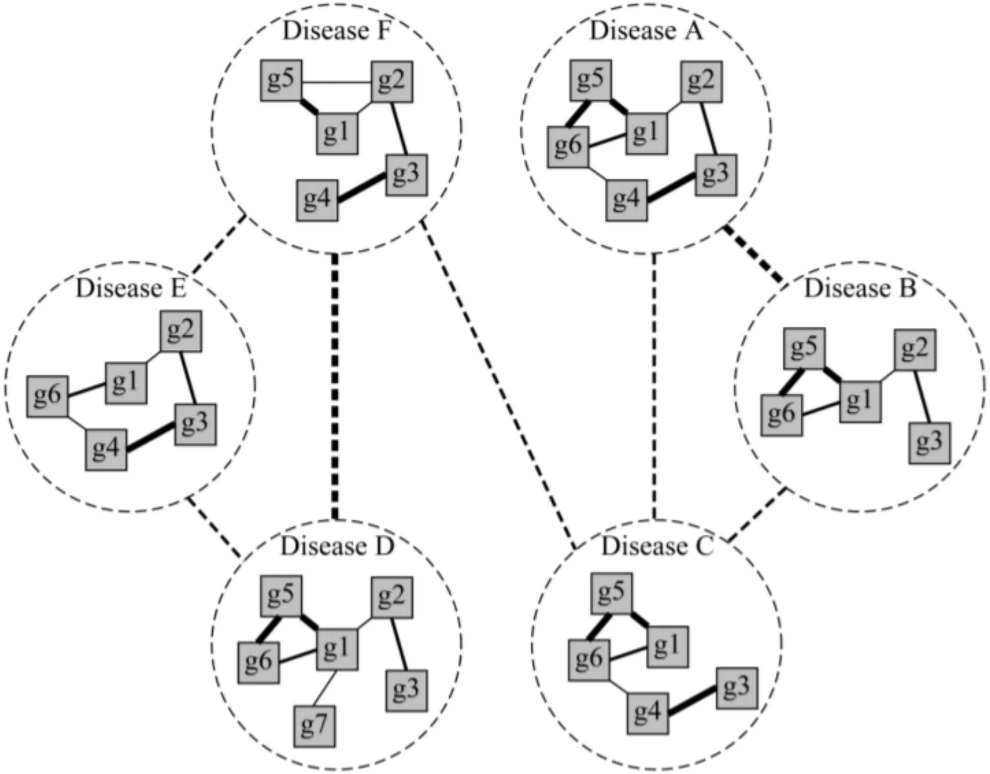
[1] Ni, Jingchao, et al. "Inside the atoms: ranking on a network of networks." *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*. 2014.

NoN: Applications

Research Area Network of Co-author Networks



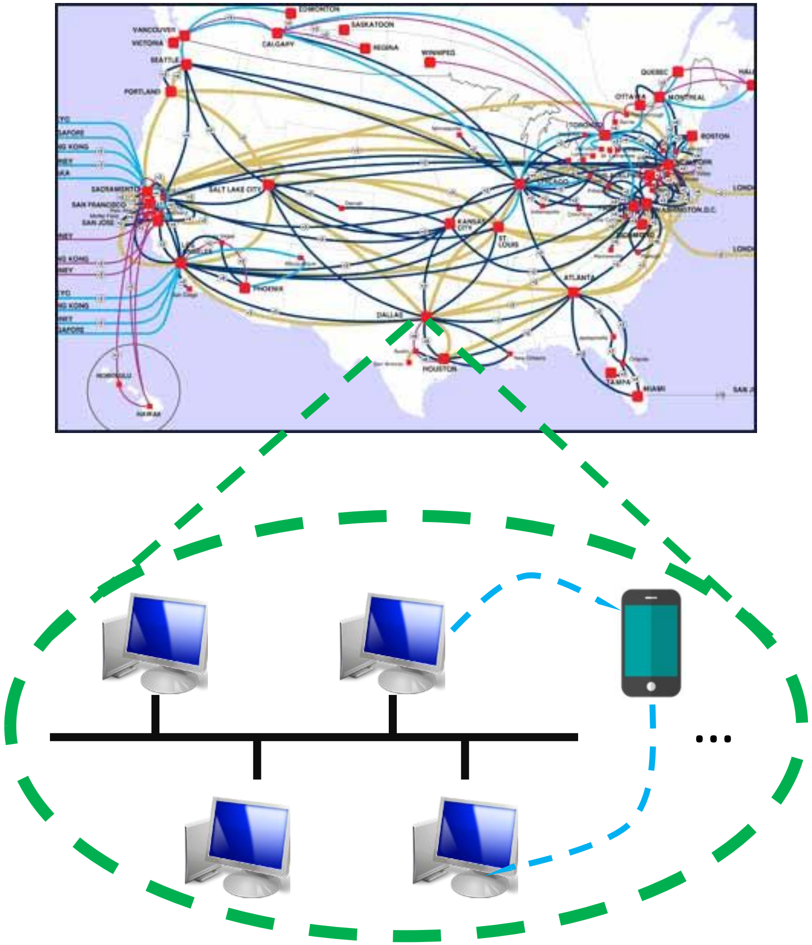
Disease Network of Protein Interaction Networks



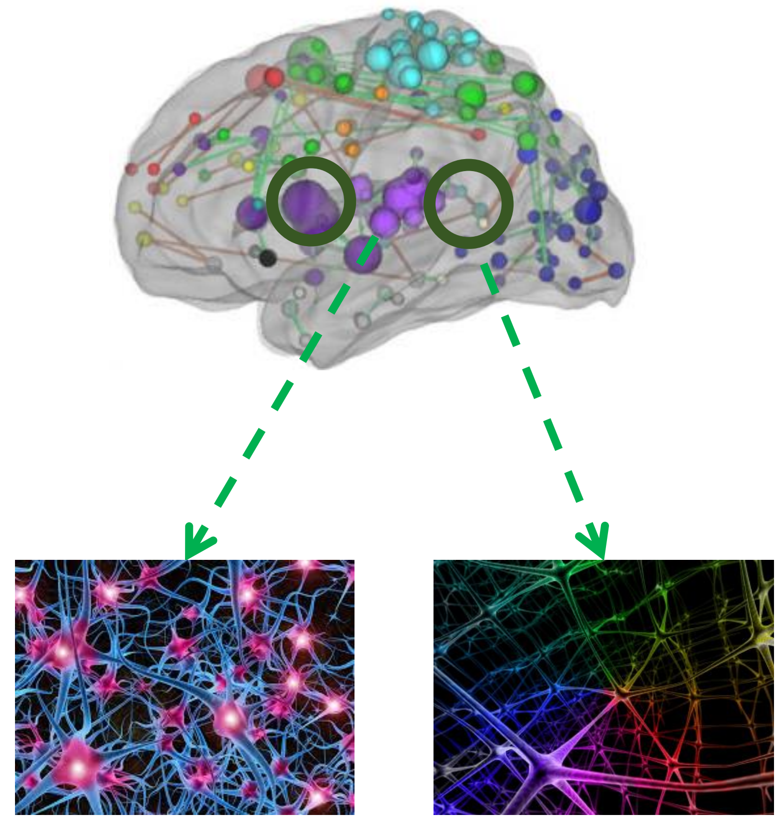
[1] Ni, Jingchao, et al. "Inside the atoms: ranking on a network of networks." *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*. 2014.

NoN: Applications

Telecommunication network



Brain network

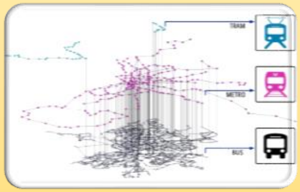


[1] https://phys.org/news/2014-12-scientists-worldwide-network-networks.html
[2] https://www.chrisharrison.net/index.php/Visualizations/InternetMap

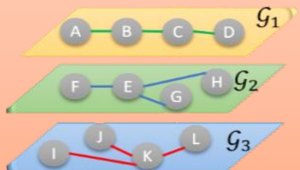
Summary: Multi-layered Networks

- General definition
  - Basic definition: $\mathcal{G}_i = (V_i, E_i, \mathbf{A}_i)$ is the i-th layer network.
  - Optional designs:
  - Node set: V_i, V_j can have overlap nodes.
    - Multi-view network: $V_i = V_j$, for any i, j.
    - Multiplex network: $V_i \cap V_j \neq \emptyset$.
    - Inter-dependent network: $V_i \cap V_j = \emptyset$.
  - Graph level cross-layer relation.
    - Multi-view network: no relation.
    - Inter-dependent network: dependency relation.
    - NoN: association relation.
  - Node level cross-layer relation.
    - Multi-view network, Hypergraph: alignment.
    - Inter-dependent network: dependency relation.

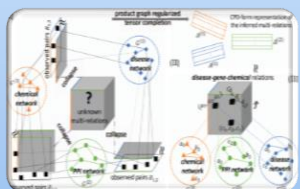
Roadmap



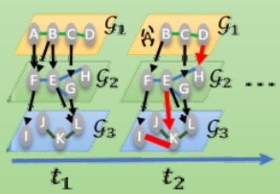
Introduction



Part I: Multi-network Data Models



Part II: Multi-network Mining Algorithms



Part III: Multi-network Future Directions

Overview of Part II

Multi-network Mining Algorithms

Classification

- **Label propagation-based multi-view/domain classification**
- GNN-based embedding
- Contrastive learning for multi-view

Hyperlink prediction

- NMF-based method
- Autoencoder -based embedding
- GNN-based embedding

Multi-network association

- Label propagation-based method
- w/o attribute
- w/ attribute
- Dependency inference
- Network alignment

Ranking

- Consistency based homogeneous
- Consistency based heterogeneous

Clustering

- Multi-view network clustering
- NoN clustering

Multi-network embedding

- MF-based embedding
- Random walk-based embedding
- GNN-based embedding

Preliminaries: Label Propagation for Graph-based Semi-supervised Learning (GSSL)

- Given:
  - An input data set with *N* instances $\{x_1, \ldots, x_l, x_{l+1}, \ldots, x_N\}$
    - $\{x_1, \ldots, x_l\}$: labeled as $\{y_1, \ldots, y_l\}$; $\{x_{l+1}, \ldots, x_N\}$: unlabeled $(y_{l+1}, \ldots, y_N = 0)$
- Output:
  - The predicted labels for $\{x_{l+1}, \ldots, x_N\}$
- General method of GSSL:
  - Undirected, connected, and weighted graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathbf{A})$

- Similar formulation: *SimRank*
- Intuition: how soon two random surfers (*i, j*) are expected to meet

$$\min_{\{f_i\}_{i=1}^{N}} \sum_{i,j=1}^{N} \mathbf{A}(i,j)\left(f_i - f_j\right)^2 + \lambda \sum_{i=1}^{N} (f_i - y_i)^2$$

$$\min_{\mathbf{F}} \mathrm{tr}\left(\mathbf{F}^{\mathrm{T}}\mathbf{L}\mathbf{F}\right) + \lambda||\mathbf{F} - \mathbf{Y}||_F^2$$

Matrix form w/ multi-class

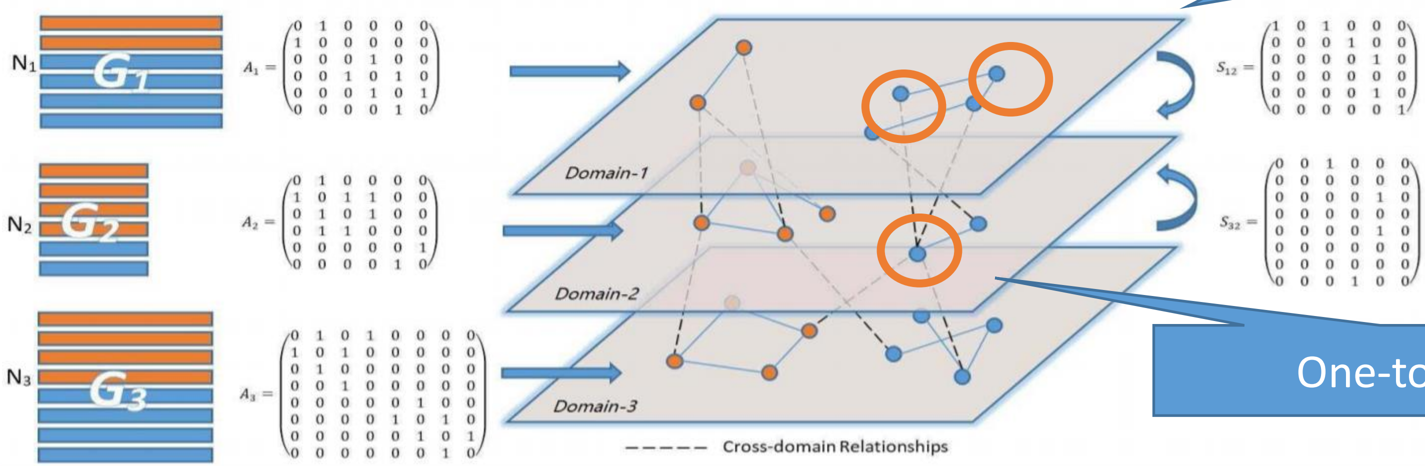
  - Notation:
    - $\mathbf{F}$: predicted label matrix; $\mathbf{Y}$: groundtruth label matrix
    - $\mathbf{L}$: graph Laplacian;

[1] Wu, Jia, et al. "Bag constrained structure pattern mining for multi-graph classification." IEEE transactions on knowledge and data engineering 26.10 (2014): 2382-2396.

MCS: Multidomain Classification With Domain Selection

- Goal: improve multidomain classification; select relevant domains
- Given:
  - $\mathcal{G}_m = (\mathcal{V}_m, \mathcal{E}_m, \mathbf{A}_m)$ with $m = 1, \ldots, M, \mathbf{A}_m \in \mathbb{R}^{N_m \times N_m}$
  - Target domain m, and labeled nodes from all domains
  - Cross-domain relations: $\{\mathbf{S}_{m,m'}\}_{m'=1, m \neq m}^{M}$ $(\mathbf{S}_{m,m'} \in \mathbb{R}^{N_m \times N_{m'}})$
- Output:
  - The unlabeled nodes from target domain m

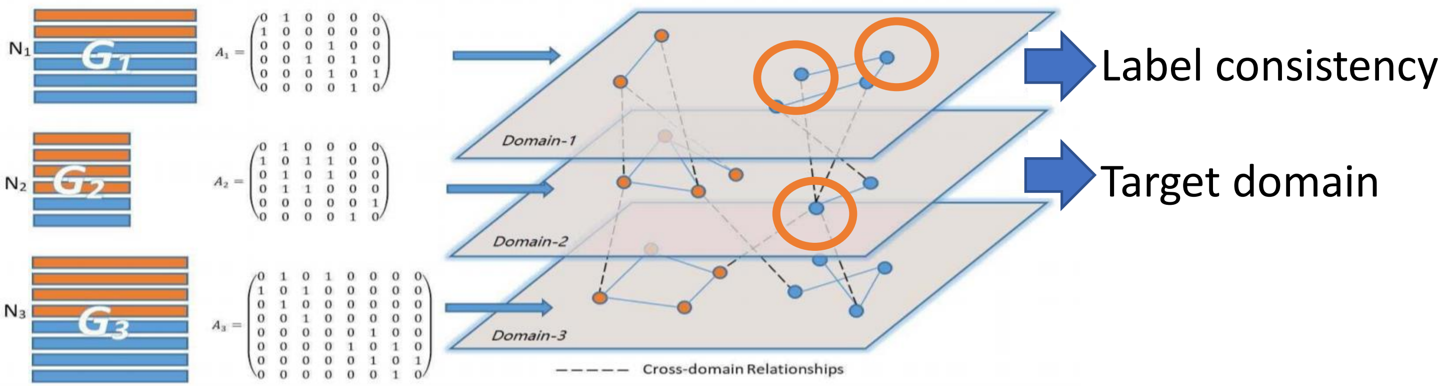
Multi-view networks can have different node set



One-to-many

MCS: Key Ideas

- Intuition:
  - $\mathbf{S}_{m,m'}$ enables cross-domain label propagation
  - Label consistency in target domain m
  - Label consistency in transferred labels from target domain
- Mathematically:
  - $\sum_{i,j} \mathbf{A}_{m'}(i,j)\left(\left[\mathbf{S}_{m',m}\,\mathbf{f}\right]_i - \left[\mathbf{S}_{m',m}\,\mathbf{f}\right]_j\right)^2$ should be small
  - $\mathbf{f}$: a node label vector of m-th domain
  - $\mathbf{S}_{m',m}\,\mathbf{f}$: label estimation of the corresponding nodes in the m'-th domain



[1] Chen, Chuan, et al. "A semisupervised classification approach for multidomain networks with domain selection." *IEEE transactions on neural networks and learning systems* 30.1 (2018): 269-283.

MCS: Formulation

- Objective function:

$$\min_{\mathbf{f},\mathbf{w}} \underbrace{\boxed{\mathbf{f}^T \mathbf{L}_m \mathbf{f}}}_{\text{Consistency of target domain}} + \sum_{m'=1, m'\neq m}^{M} \underbrace{\boxed{\omega_{m'} \mathbf{f}^T \mathbf{L}_{m',m} \mathbf{f}}}_{\text{Label consistency of other domains}} + \underbrace{\boxed{\lambda \|\mathbf{f}-\mathbf{y}\|_2^2}}_{\text{Label regularizer}} + \gamma \|\mathbf{w}\|_2^2$$

$$\sum_{m'=1, m'\neq m}^{M} \omega_{m'} = 1, \quad \omega_{m'} \geq 0.$$

Transformed adjacency matrix from domain *m'*

- $\mathbf{L}_{m',m} = \frac{\mathbf{D}_{\mathbf{S}_{m,m'}\mathbf{A}_{m'}\mathbf{S}_{m',m}} - \boxed{\mathbf{S}_{m,m'}\mathbf{A}_{m'}\mathbf{S}_{m',m}}}{\|\mathbf{D}_{\mathbf{S}_{m,m'}\mathbf{A}_{m'}\mathbf{S}_{m',m}} - \mathbf{S}_{m,m'}\mathbf{A}_{m'}\mathbf{S}_{m',m}\|_F}$: scaled graph Laplacian
- $\omega_{m'}$: other domains' contributes to the label estimation in the *m*-th domain
- λ, γ: weights for regularizers

[1] Chen, Chuan, et al. "A semisupervised classification approach for multidomain networks with domain selection." *IEEE transactions on neural networks and learning systems* 30.1 (2018): 269-283.

MCS: Algorithm

- Decompose into two subproblems
  - S1. Instance-prediction subproblem for instance label vector

$$\mathbf{f} = \lambda \left(\boxed{\lambda \mathbf{I} + \mathbf{L}_m + \sum_{m'=1, m' \neq m}^{M} w_{m'} \mathbf{L}_{m',m}} \right)^{-1} \mathbf{y}.$$

  - S2. Domain-weighting subproblem for domain weights

$$\min_{\mathbf{w}} \mathbf{v}^T \mathbf{w} + \gamma \, ||\mathbf{w}||_2^2$$
$$\text{s.t.} \quad \mathbf{w}^T \mathbf{1} = 1, \mathbf{w} \geq 0$$

Where $\mathbf{v} = [v_1, \ldots, v_M]$ w/o v_m, $v_1 \leq v_2 \leq \cdots \leq v_M$, $v_{m'} = \mathbf{f}^{\mathbf{T}} \mathbf{L}_{m',m} \mathbf{f}$

  - Solution:

$$w_{m'} = \begin{cases} \dfrac{\theta - v_{m'}}{2\gamma} & m' \leq P \\ 0 & m' > P \end{cases} \qquad \theta = \frac{2\gamma + \sum_{i=1}^{P} v_i}{\min\{P, M-1\}}$$

$$P = \arg\max_{m'} (\theta - v_{m'} > 0).$$

Time complexity:
$O(iter * (E + M^2))$

[1] Chen, Chuan, et al. "A semisupervised classification approach for multidomain networks with domain selection." *IEEE transactions on neural networks and learning systems* 30.1 (2018): 269-283.

MCS: Experiments

- Dataset: cancer subtype classification problem

Domain	Platform	Sample Type	Size
$\mathcal{G}_1$	UNC_IlluminaHiSeq_RNASeq	Normal/LUAD/LUSC	403
$\mathcal{G}_2$	BCGSC_IlluminaGA_miRNASeq	LUAD/LUSC	199
$\mathcal{G}_3$	BCGSC__IlluminaHiSeq_miRNASeq	Normal/LUAD/LUSC	891
$\mathcal{G}_4$	JHU_USC__HumanMethylation27	Normal/LUAD/LUSC	311
$\mathcal{G}_5$	JHU_USC__HumanMethylation450	Normal/LUAD/LUSC	919

- Target domain: G_1

$$\mathrm{RI} = \frac{\mathrm{TP}+\mathrm{TN}}{\mathrm{TP}+\mathrm{TN}+\mathrm{FP}+\mathrm{FN}}$$

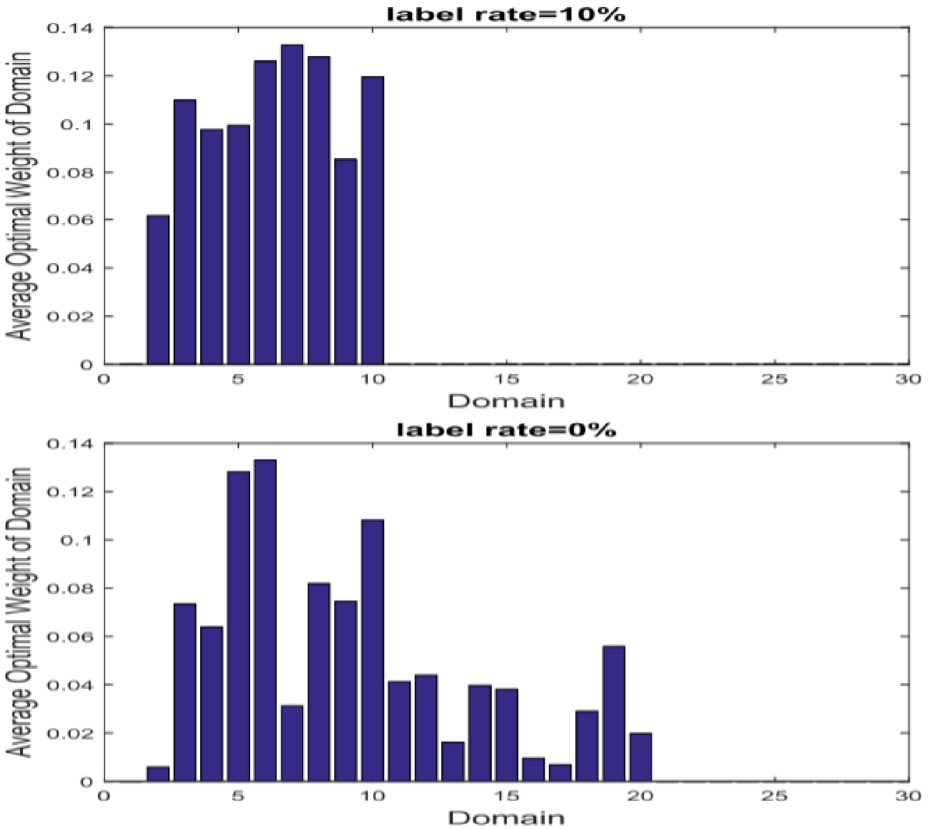
Focused Domain	Label Rate	Metrics	SSC	CGC	MCA	PSC	**MCS**
$\mathcal{G}_1$	10%	RI	0.6311	0.6521	0.7139	0.6829	**0.7298**
		Acc	0.3671	0.4479	0.6164	0.5426	**0.6482**
	15%	RI	0.6369	0.6421	0.7139	0.7014	**0.7694**
		Acc	0.3909	0.4113	0.6164	0.5888	**0.7155**
	20%	RI	0.6512	0.6721	0.7139	0.7288	**0.7872**
		Acc	0.4447	0.5122	0.6164	0.6463	**0.7421**
	25%	RI	0.6802	0.7115	0.7139	0.7437	**0.7959**
		Acc	0.5352	0.6113	0.6164	0.6734	**0.7546**

- Outperforms all baselines in other domains as well

[1] Chen, Chuan, et al. "A semisupervised classification approach for multidomain networks with domain selection." *IEEE transactions on neural networks and learning systems* 30.1 (2018): 269-283.

MCS: Experiments

- Synthetic data, domain selection evaluation



- Able to select relevant domains with the proposed formulation

[1] Chen, Chuan, et al. "A semisupervised classification approach for multidomain networks with domain selection." *IEEE transactions on neural networks and learning systems* 30.1 (2018): 269-283.

MCS: Relation with Other Methods

- SMGI: Multi-graph label propagation by sparse integration:

$$\min_{\mathbf{f},\mu} \boxed{\sum_{m=1}^{M} \mu_m \mathbf{f}^T \mathbf{L}_m \mathbf{f}} + \lambda_1 ||\mathbf{f} - \mathbf{y}||_2^2 + \lambda_2 ||\mu||_2^2$$
$$\text{s.t.} \quad \mu \mathbf{1} = 1, \mu \geq 0$$

μ: selecting domains

- Co-regularized Multidomain Graph Clustering (CGC) With Focused Domain:

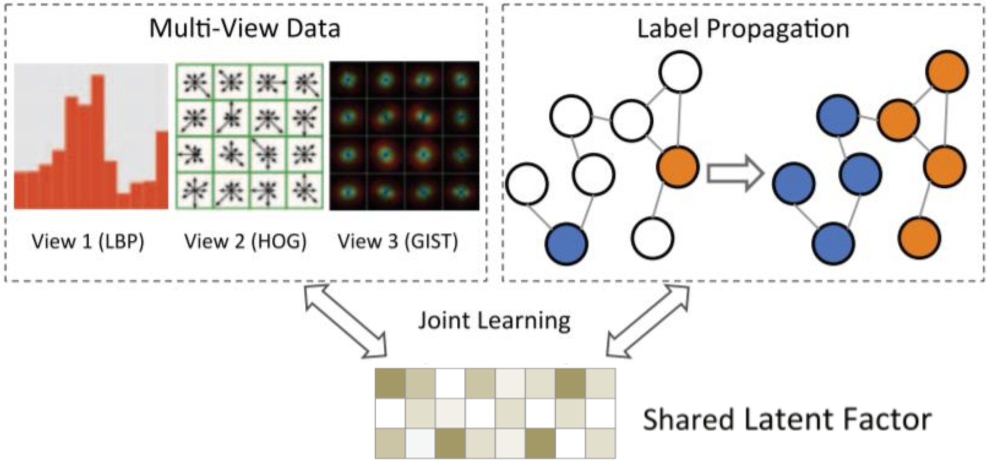
$$\min_{\mathbf{H}_m,\mu} ||\mathbf{A}_m - \mathbf{H}_m \mathbf{H}_m^T||_F^2 + \lambda_1 ||\mathbf{H}_m - \mathbf{y}||_2^2 + \lambda_2 ||\mu||_2^2$$
$$+ \boxed{\sum_{m'=1, m' \neq m}^{M} \mu_{m'} ||\mathbf{S}_{m,m'} \mathbf{H}_{m'} (\mathbf{S}_{m,m'} \mathbf{H}_{m'})^T - \mathbf{H}_m \mathbf{H}_m^T|||_F^2}$$
$$\text{s.t.} \quad \mathbf{H}_m \geq 0, \mu \geq 0, \mu \mathbf{1} = 1$$

NMF approach

[1] Chen, Chuan, et al. "A semisupervised classification approach for multidomain networks with domain selection." *IEEE transactions on neural networks and learning systems* 30.1 (2018): 269-283.
[2] M. Karasuyama and H. Mamitsuka, "Multiple graph label propagation by sparse integration," IEEE Trans. Neural Netw. Learn. Syst., 2013
[3] W. Cheng, Z. Guo, X. Zhang, and W. Wang,"CGC: A flexible and robust approach to integrating co-regularized multi-domain graph for clustering," Trans. Knowl. Discovery Data, vol. 10, no. 4, 2015, Art. no. 46.

MVGL: Multi-View Graph Learning

- Motivation:
  - There exists a unified latent graph for all views
  - Jointly learn the latent graph with classification
- Challenges:
  - C1. How to construct a robust graph from multiple views?
  - C2. How to ensure the sparsity of graph construction?
  - C3. How to integrate label propagation with graph construction?



[1] Li, Sheng, et al. "Multi-view graph learning with adaptive label propagation." *2017 IEEE International Conference on Big Data (Big Data)*. IEEE, 2017. Li, Sheng, et al. "Multi-view graph learning with adaptive label propagation." 2017 IEEE International Conference on Big Data (Big Data). IEEE, 2017..

MVGL: Problem Definition

- Given:
  - $\mathcal{X} = \{\mathbf{X}^{(v)}\}, v = 1, \cdots, V$: a multi-view dataset
- Remarks:
  - N samples in each view, $\mathbf{X}^{(v)} = \left\{x_1^{(v)}, \cdots, x_l^{(v)}, x_{l+1}^{(v)}, \cdots, x_N^{(v)}\right\} \in \mathbb{R}^{d(v) \times N}$
  - The first l samples in each view are labeled
  - $d(v)$ is the dimension of samples in the v-th view
- Output:
  - The labels for the unlabeled samples

[1] Li, Sheng, et al. "Multi-view graph learning with adaptive label propagation." *2017 IEEE International Conference on Big Data (Big Data)*. IEEE, 2017. Li, Sheng, et al. "Multi-view graph learning with adaptive label propagation." 2017 IEEE International Conference on Big Data (Big Data). IEEE, 2017..

MVGL: Key Ideas

- C1. How to construct a robust graph from multiple views?
  - Learn shared latent factors from all the views
  - Build a common graph based on the shared factors
  - These factors are view-independent features
- C2. How to ensure the sparsity of the graph?
  - Sparse constraint based on k-NN is incorporated to the model
- C3. How to integrate label propagation with graph construction process?
  - Joint learning framework to integrate graph construction and label propagation.

[1] Li, Sheng, et al. "Multi-view graph learning with adaptive label propagation." *2017 IEEE International Conference on Big Data (Big Data)*. IEEE, 2017. Li, Sheng, et al. "Multi-view graph learning with adaptive label propagation." 2017 IEEE International Conference on Big Data (Big Data). IEEE, 2017..

MVGL: Formulation

$$\min_{\mathbf{U}_v,\mathbf{R},\mathbf{W},\mathbf{S},\mathbf{F}} \boxed{\sum_{v=1}^{V} \left|\left|\mathbf{X}^{(v)} - \mathbf{U}_v\mathbf{R}\right|\right|_F^2} \boxed{- \lambda_1 ||\mathbf{W}||_F^2}$$

Feature decomposition

$$\boxed{+\lambda_2 tr\left(\mathbf{F}^{\mathrm{T}}\mathbf{L}\mathbf{F}\right) + \gamma\left|\left|\mathbf{Y} - \mathbf{F}\right|\right|_F^2}$$

Adaptive label propagation $\mathbf{L} = \mathbf{D} - \mathbf{W}$

$$+\boxed{\lambda_3 (\sum_{v=1}^{V} ||\mathbf{U}_v||_F^2 + ||\mathbf{R}||_F^2)}$$

Regularizer

$$\boxed{\text{s.t. } \mathbf{W} = \mathbf{S} \odot \left(\mathbf{R}^{\mathrm{T}}\mathbf{R}\right), \textstyle\sum_j^N \mathbf{S}_{ij} = k\,, \mathbf{S}_{ii} = 0}$$

Graph construction and sparsification

Sparsification constraint; **R**: latent representation shared by *all* views

K-NN selection

[1] Li, Sheng, et al. "Multi-view graph learning with adaptive label propagation." *2017 IEEE International Conference on Big Data (Big Data)*. IEEE, 2017. Li, Sheng, et al. "Multi-view graph learning with adaptive label propagation." 2017 IEEE International Conference on Big Data (Big Data). IEEE, 2017..

MVGL: Experiments

- Node classification on online news dataset (BBC, Reuters, The Guardian):
- Nodes: articles, labels: topic classes

# of labeled samples randomly chosen from each class

	Method	$N_{tr}=1$	$N_{tr}=2$	$N_{tr}=3$	$N_{tr}=4$	$N_{tr}=5$	$N_{tr}=6$
SV	LGC-V1 [9]	74.62±5.77	76.98±4.78	79.38±3.64	82.48±3.34	84.64±3.53	86.41±2.71
	LGC-V2 [9]	69.68±4.78	74.21±3.01	75.63±4.91	78.72±5.25	80.58±5.01	81.00±5.11
	LGC-V3 [9]	74.79±5.19	75.31±2.87	81.01±3.79	82.73±2.93	83.82±3.60	84.77±3.57
MV	FeaFusion	79.60±5.82	79.10±4.44	82.45±3.79	86.05±2.96	88.72±2.72	90.17±1.68
	GraphFusion	78.60±5.42	81.32±5.20	83.84±3.37	86.95±3.39	88.56±3.66	90.90±3.19
	simpleMKL [21]	59.51±5.67	62.61±9.20	70.07±8.46	77.31±3.58	79.64±1.44	81.95±2.81
	MvDA [22]	35.34±4.27	57.07±7.85	67.55±7.11	74.62±6.64	80.86±3.51	82.63±4.03
	MUDA [23]	35.77±4.62	59.21±7.23	68.65±5.93	77.32±6.10	82.16±3.27	84.55±3.92
	MVGL (Ours)	**84.27**±3.77	**85.65**±4.64	**87.74**±3.90	**89.31**±2.77	**90.14**±4.39	**91.86**±1.69

- Multi-view methods like FeaFusion and GraphFusion usually perform better than single-view methods
- The proposed MVGL approach outperforms the single-view and multi-view baselines when there are very limited information

[1] Li, Sheng, et al. "Multi-view graph learning with adaptive label propagation." *2017 IEEE International Conference on Big Data (Big Data)*. IEEE, 2017. Li, Sheng, et al. "Multi-view graph learning with adaptive label propagation." 2017 IEEE International Conference on Big Data (Big Data). IEEE, 2017..

Overview of Part II

Multi-network Mining Algorithms

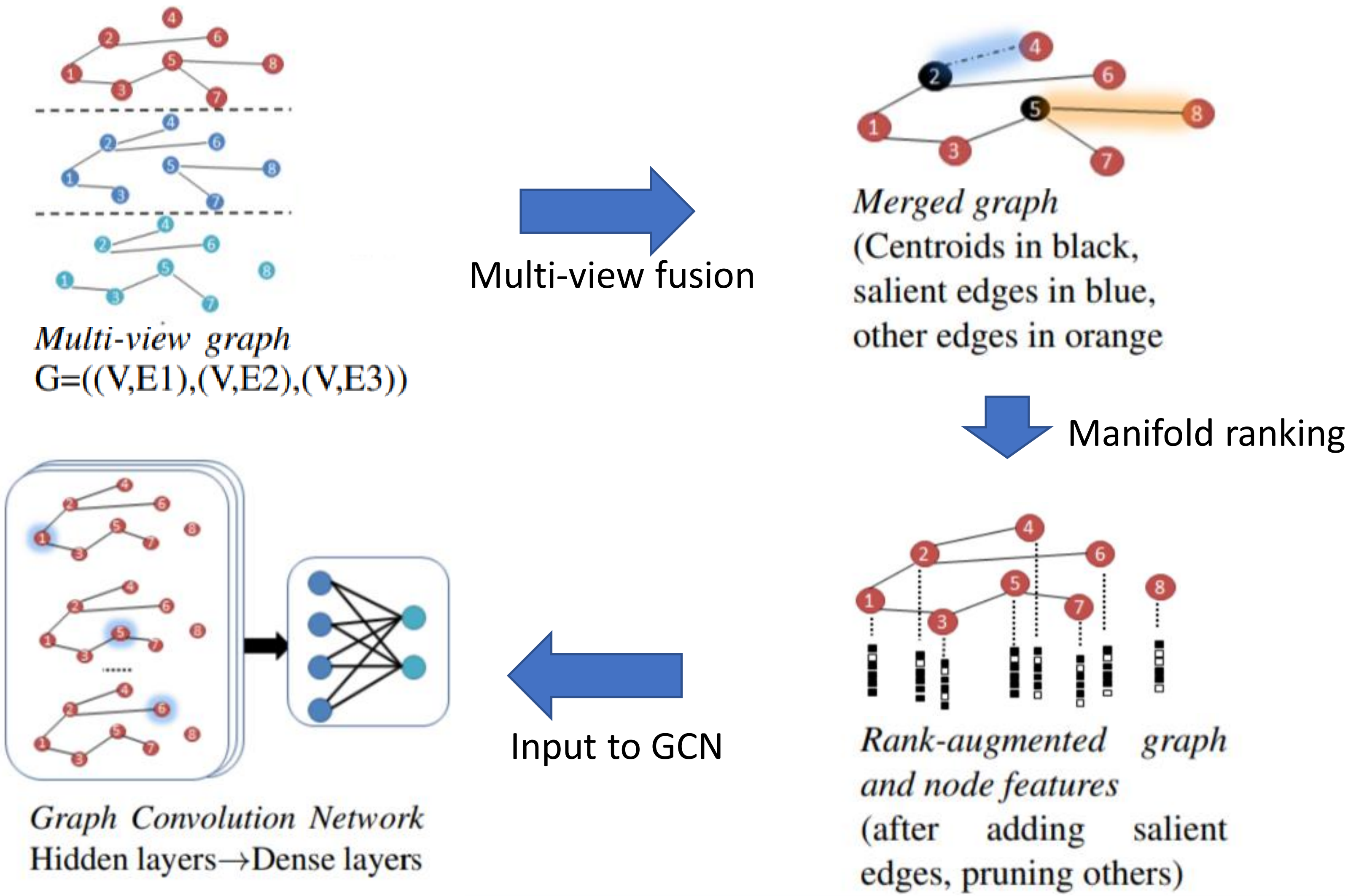
Classification	Hyperlink prediction	Multi-network association	Ranking	Clustering	Multi-network embedding
• Label propagation-based multi-view/domain classification • **GNN-based embedding** • Contrastive learning for multi-view	• NMF-based method • Autoencoder-based embedding • GNN-based embedding	• Label propagation-based method • w/o attribute • w/ attribute • Dependency inference • Network alignment	• Consistency based homogeneous • Consistency based heterogeneous	• Multi-view network clustering • NoN clustering	• MF-based embedding • Random walk-based embedding • GNN-based embedding

Multi-GCN: Graph Convolutional Networks for Multi-View Networks

- Motivation:
  - Graph Neural Networks prevail in many graph learning problems
  - Latent subspace-based method performs well in multi-view tasks
- Key questions:
  - How to apply GNN techniques to multi-view networks?
  - How to construct a latent subspace shared by multiple views?
- Key ideas:
  - Merge subspace representations of multiple views
  - Graph-based manifold ranking for latent network generation
  - Learn classification task by GNNs on latent network

[1] Khan, Muhammad Raza, and Joshua E. Blumenstock. "Multi-gcn: Graph convolutional networks for multi-view networks, with applications to global poverty." *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 33. No. 01. 2019.

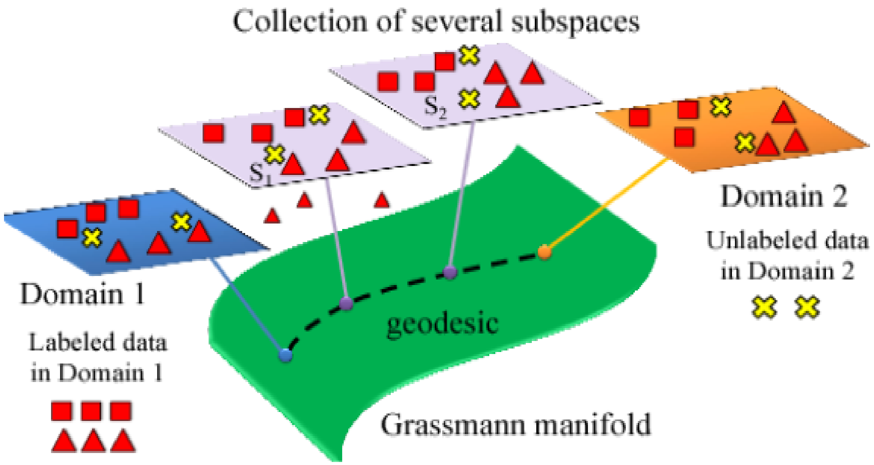
Multi-GCN: Overview



[1] Khan, Muhammad Raza, and Joshua E. Blumenstock. "Multi-gcn: Graph convolutional networks for multi-view networks, with applications to global poverty." *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 33. No. 01. 2019.

Multi-GCN: Preliminaries

- Grassman manifold:
  - A set of k-dimensional linear subspaces in $\mathbb{R}^{n\times k}$
  - Each unique subspace is mapped to a unique point on manifold
  - Points on the manifold are represented by orthonormal matrix
- Projection distance between two subspaces Y_1, Y_2[2]:
  - $d_{proj}^2(Y_1, Y_2) = \sum_{i=1}^{k} \sin^2\theta_i = k - tr(Y_1 Y_1^T Y_2 Y_2^T)$



[1] Khan, Muhammad Raza, and Joshua E. Blumenstock. "Multi-gcn: Graph convolutional networks for multi-view networks, with applications to global poverty." *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 33. No. 01. 2019.
[2] Dong, Xiaowen, et al. "Clustering on multi-layer graphs via subspace analysis on Grassmann manifolds." IEEE Transactions on signal processing 62.4 (2013): 905-918.

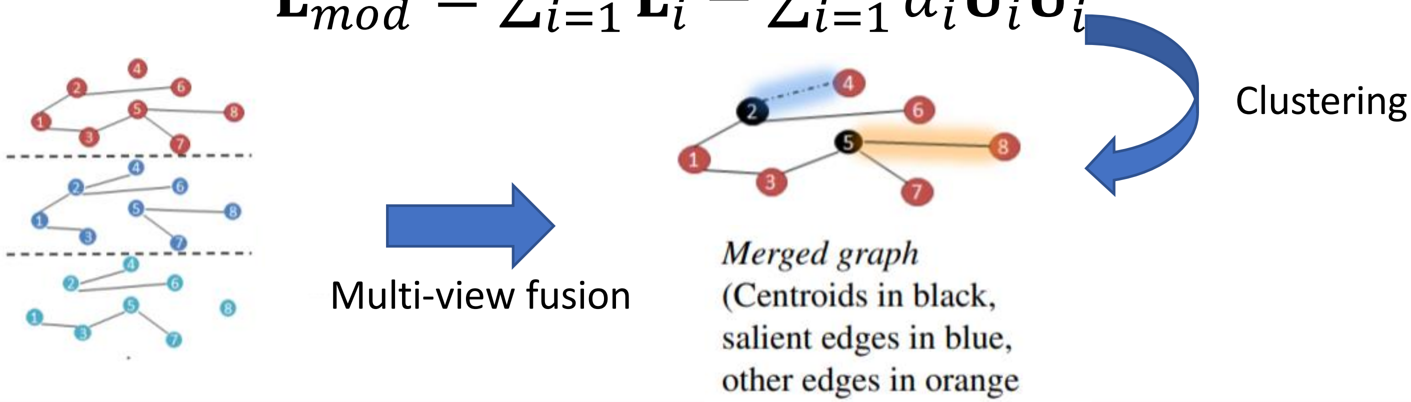
Multi-GCN: Model

- Merge subspace representations:
  - Step 1: $\min_{\mathbf{U}_i \in \mathbb{R}^{n \times k}} tr(\mathbf{U}_i^T \mathbf{L}_i \mathbf{U}_i)$ s.t. $\mathbf{U}_i^T \mathbf{U}_i = \mathbf{I}$

    Where $\mathbf{L}_i = \mathbf{D}_i^{-1/2}(\mathbf{D}_i - \mathbf{W}_i)\mathbf{D}_i^{-1/2}$
  - Step 2: $\min_{\mathbf{U}_i \in \mathbb{R}^{n \times k}} \sum_{i=1}^{M} tr(\mathbf{U}^T \mathbf{L}_i \mathbf{U}) + \boxed{\alpha_i [kM - tr(\mathbf{U}\mathbf{U}^{\mathrm{T}} \mathbf{U}_i \mathbf{U}_i^T)]}$

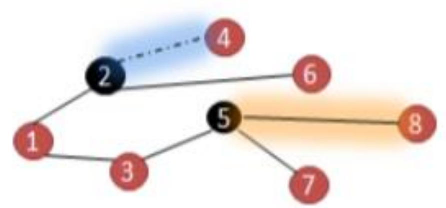
    $$\text{s.t. } \mathbf{U}_i^T \mathbf{U}_i = \mathbf{I}$$
  - Solution: the first k eigenvectors of modified Laplacian;

$$\mathbf{L}_{mod} = \sum_{i=1}^{M} \mathbf{L}_i - \sum_{i=1}^{M} \alpha_i \mathbf{U}_i \mathbf{U}_i^T$$



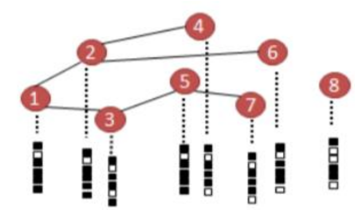
Multi-GCN: Model (cont'd)

- Graph-Based Manifold Ranking
  - Closed form solution: $\mathbf{f}^* = (\mathbf{I} - \beta \mathbf{L}_{mod})^{-1}\mathbf{q}$
  - Adding salient edges
  - Pruning non-salient edges
  - Input the modified and augmented graph to GNN
- Complexity: $O(MN^3 + MN^2K + N^2C^2 + tN)$
  - M: # of views; N: # of nodes per view; K,C: small constants



Merged graph
(Centroids in black, salient edges in blue, other edges in orange

Manifold ranking



Rank-augmented graph and node features
(after adding salient edges, pruning others)

Multi-GCN: Experiments

- Classification accuracy on mobile phone data:

Dataset	Data Type	Nodes	Edges (view 1)	Edges (view 2)	Classes	Features	Label Rate
Product Adoption	Phone logs (West Africa)	17,000	23,032	18,371	2	132	0.002
Poverty Prediction	Phone logs (East Africa)	422	544	1,799	2	1,709	0.094
Gender Prediction	Phone logs (South Asia)	958	992	978	2	821	0.042

Method	Product Adoption	Poverty Prediction	Gender Prediction
DeepWalk (first view)	56.43±0.187	51.91±0.62	53.18± 0.55
DeepWalk (second view)	51.97±0.112	50.34±0.36	50.84±0.64
DeepWalk (view union)	56.81± 0.114	50.87±0.95	52.34±0.50
Node2vec (first view)	53.87±0.20	52.26±0.58	50.12± 0.40
Node2vec (second view)	50.50±0.11	49.70±0.23	51.68±0.40
Node2vec (view union)	54.50±0.11	50.52±0.63	51.64±0.53
LINE (first view)	51.11±0.01	50.15±0.02	51.56± 0.001
LINE (second view)	50.83±0.01	52.29±0.001	50.00±0.001
LINE (view union)	56.26±0.003	50.18±0.001	51.33±0.002
GCN (first view)	70.74±2.2	55.19±2.33	63.97± 1.29
GCN (second view)	71.40±1.81	50.06±0.81	63.01±0.013
GCN (view union)	71.90±0.9	50.22±0.56	63.90±1.32
Multi-GCN (this paper)	**73.47±0.91**	**59.23±0.20**	**66.34± 1.03**

- Multi-GCN outperforms existing state-of-the-art benchmarks

[1] Khan, Muhammad Raza, and Joshua E. Blumenstock. "Multi-gcn: Graph convolutional networks for multi-view networks, with applications to global poverty." *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 33. No. 01. 2019.

Multi-GCN: Experiments

- Classification accuracy on citation networks:

Method	Citeseer	Cora
Predefined train-test splits		
ManiReg (first view) - Yang, Cohen, and Salakhutdinov (2016)	60.1	59.5
DeepWalk (first view) - Perozzi, Al-Rfou, and Skiena (2014)	43.2	67.2
Planetoid (first view) - Yang, Cohen, and Salakhutdinov (2016)	64.7	75.7
GCN (first view)	70.3	81.5
GCN (second view)	50.7	53.6
GCN (view union)	70.7	80.4
Multi-GCN (this paper)	**71.3**	**82.5**
Randomized train-test splits		
GCN (first view)	67.9 ± 0.5	80.1 ± 0.5
GCN (second view)	53.6 ± 0.1	56.9 ± 0.3
GCN (view union)	67.9 ± 0.3	78.5 ± 0.1
Multi-GCN (this paper)	$\mathbf{70.5 \pm 0.2}$	$\mathbf{81.1 \pm 0.2}$

- Multi-GCN outperforms existing representative benchmarks

Overview of Part II

Multi-network Mining Algorithms

Classification	Hyperlink prediction	Multi-network association	Ranking	Clustering	Multi-network embedding
• Label propagation-based multi-view/domain classification • GNN-based embedding • **Contrastive learning for multi-view**	• NMF-based method • Autoencoder -based embedding • GNN-based embedding	• Label propagation-based method • w/o attribute • w/ attribute • Dependency inference • Network alignment	• Consistency based homogeneous • Consistency based heterogeneous	• Multi-view network clustering • NoN clustering	• MF-based embedding • Random walk-based embedding • GNN-based embedding

Contrastive Multi-View Representation Learning on Graphs

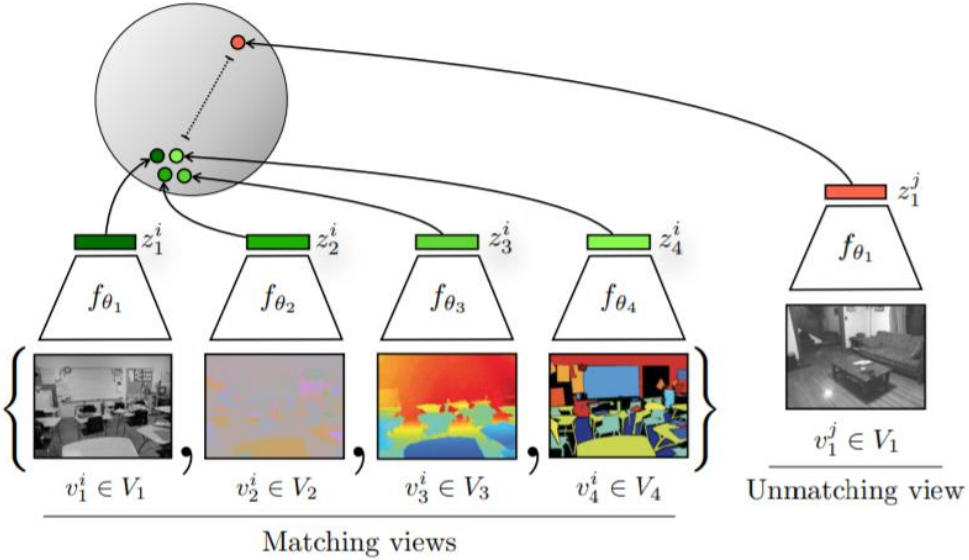
- Motivation
  - Multi-view visual representation learning: image classification [1]
  - Data augmentations for multiple views: for contrastive learning [2]
  - E.g.,

Original

Crop and resize

Rotate

Cutout



Multi-view representation learning for image

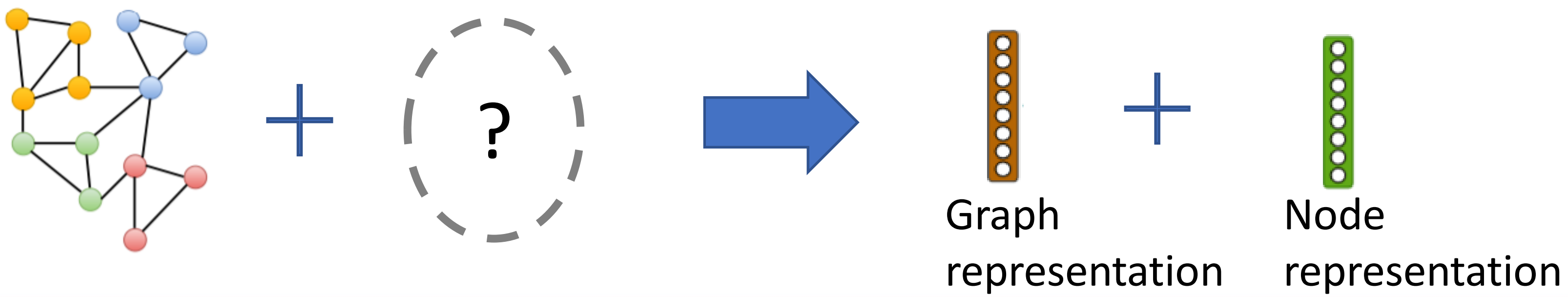
- Q: How to apply these techniques to graph representation?

[1] Chen, T., Kornblith, S., Norouzi, M., and Hinton, G. A simple framework for contrastive learning of visual representations. arXiv preprint arXiv:2002.05709, 2020

[2] Tian, Yonglong, Dilip Krishnan, and Phillip Isola. "Contrastive multiview coding." Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XI 16. Springer International Publishing, 2020.

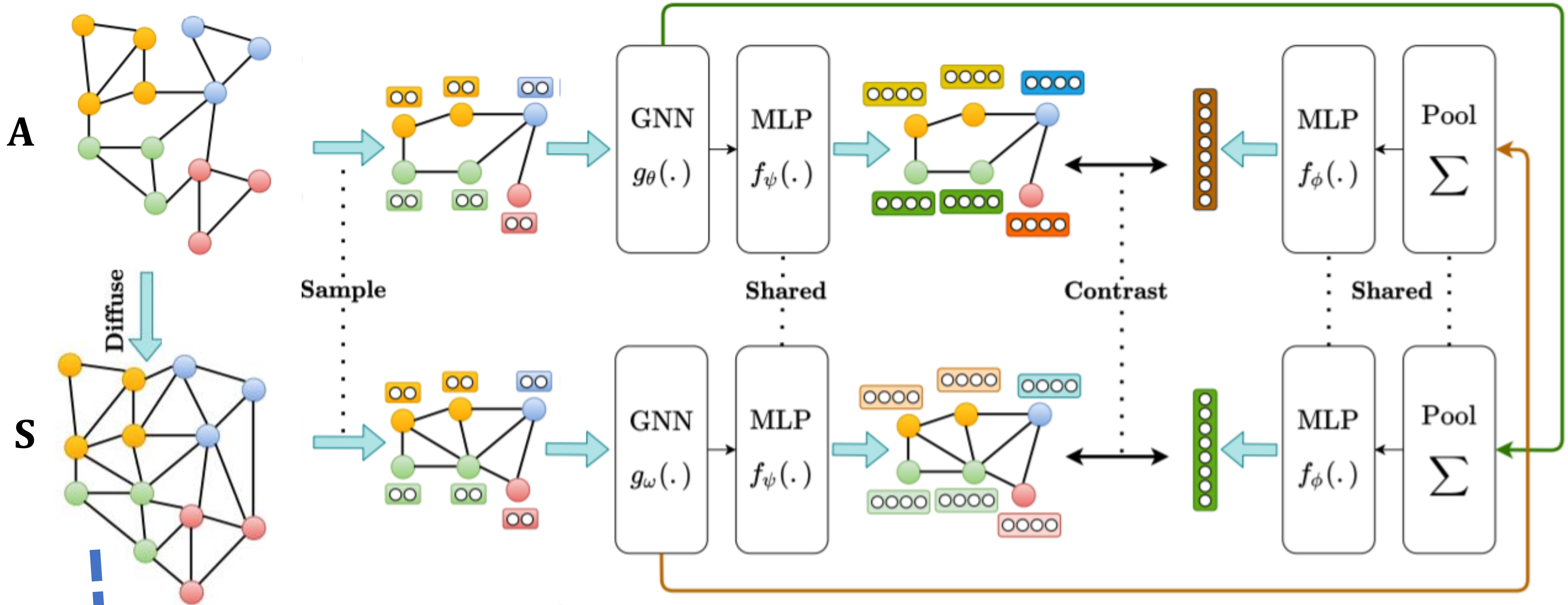
Key Ideas

- Structural augmentation mechanism:
  - Transform a sample graph into a correlated view
  - Sub-sample from all views
- Node and graph representation:
  - One GNN model for each view for node representation
  - Shared MLP layer for graph representation
- A discriminator to contrastive learning
  - Contrast node representation of one view w/ graph representation of another view



[1] Hassani, Kaveh, and Amir Hosein Khasahmadi. "Contrastive multi-view representation learning on graphs." International Conference on Machine Learning. PMLR, 2020.

Model Overview



Graph augmentation

GNN for node representation

Contrastive learning

GNN for graph representation

Diffusion:

$$\mathbf{S}^{\text{heat}} = \exp\left(t\mathbf{A}\mathbf{D}^{-1} - t\right)$$

$$\mathbf{S}^{\text{PPR}} = \alpha\left(\mathbf{I}_n - (1-\alpha)\mathbf{D}^{-1/2}\mathbf{A}\mathbf{D}^{-1/2}\right)^{-1}$$

Maximize the mutual information of node and graph representation of different views

[1] Hassani, Kaveh, and Amir Hosein Khasahmadi. "Contrastive multi-view representation learning on graphs." International Conference on Machine Learning. PMLR, 2020.

Training Details

- Ideas：
  - Maximize the MI between two views using deep InfoMax [2]
  - Simultaneously encode local (adjacency matrix) & global info. (diffusion matrix)
- Objective function:

$$\max_{\theta,\omega,\phi,\psi} \frac{1}{|\mathcal{G}|}\sum_{g\in\mathcal{G}}\left[\frac{1}{|g|}\sum_{i=1}^{|g|}\left[\boxed{\mathbf{MI}\left(\vec{h}_i^{\alpha},\vec{h}_g^{\beta}\right)}+\mathbf{MI}\left(\vec{h}_i^{\beta},\vec{h}_g^{\alpha}\right)\right]\right]$$

Number of views

Number of nodes in each graph view

Implemented by $D\left(h_i^{\alpha},h_g^{\beta}\right) =< h_i^{\alpha},h_g^{\beta} >$

- Representation inference:
  - Graph: $h = h_g^{\alpha} + h_g^{\beta}$, node: $\mathbf{H} = \mathbf{H}^{\alpha} + \mathbf{H}^{\beta}$
- Negative sampling:
  - Random feature permutation
  - Adjacency matrix corruption

[1] Hassani, Kaveh, and Amir Hosein Khasahmadi. "Contrastive multi-view representation learning on graphs." International Conference on Machine Learning. PMLR, 2020.
[2] Velickovic, Petar, et al. "Deep Graph Infomax." ICLR (Poster) 2.3 (2019): 4.

Experiments

- Node classification accuracy for supervised and unsupervised models:

	Method	Input	Cora	Citeseer	Pubmed
Supervised	MLP (Veličković et al., 2018)	X, Y	55.1	46.5	71.4
	ICA (Lu & Getoor, 2003)	A, Y	75.1	69.1	73.9
	LP (Zhu et al., 2003)	A, Y	68.0	45.3	63.0
	ManiReg (Belkin et al., 2006)	X, A, Y	59.5	60.1	70.7
	SemiEmb (Weston et al., 2012)	X, Y	59.0	59.6	71.7
	Planetoid (Yang et al., 2016)	X, Y	75.7	64.7	77.2
	Chebyshev (Defferrard et al., 2016)	X, A, Y	81.2	69.8	74.4
	GCN (Kipf & Welling, 2017)	X, A, Y	81.5	70.3	79.0
	MoNet (Monti et al., 2017)	X, A, Y	81.7 ± 0.5	–	78.8 ± 0.3
	JKNet (Xu et al., 2018)	X, A, Y	82.7 ± 0.4	$\mathbf{73.0 \pm 0.5}$	77.9 ± 0.4
	GAT (Veličković et al., 2018)	X, A, Y	$\mathbf{83.0 \pm 0.7}$	72.5 ± 0.7	$\mathbf{79.0 \pm 0.3}$
Unsupervised	Linear (Veličković et al., 2019)	X	47.9 ± 0.4	49.3 ± 0.2	69.1 ± 0.3
	DeepWalk (Perozzi et al., 2014)	X, A	70.7 ± 0.6	[illegible]	[illegible]
	GAE (Kipf & Welling, 2016)	X, A	71.5 ± 0.4	[illegible]	[illegible]
	VERSE (Tsitsulin et al., 2018)	X, S, A	72.5 ± 0.3	[illegible]	[illegible]
	DGI (Veličković et al., 2019)	X, A	82.3 ± 0.6	[illegible]	[illegible]
	DGI (Veličković et al., 2019)	X, S	83.8 ± 0.5	[illegible]	77.9 ± 0.3
	Ours	X, S, A	$\mathbf{86.8 \pm 0.5}$	$\mathbf{73.3 \pm 0.5}$	$\mathbf{80.1 \pm 0.7}$

Better than supervised methods!

[1] Hassani, Kaveh, and Amir Hosein Khasahmadi. "Contrastive multi-view representation learning on graphs." International Conference on Machine Learning. PMLR, 2020.

Experiments

- Node/graph classification accuracy:

		Node			Graph				
		CORA	CITESEER	PUBMED	MUTAG	PTC-MR	IMDB-BIN	IMDB-MULTI	REDDIT-BIN
MI EST.	NCE	85.8 ± 0.7	**73.3 ± 0.5**	**80.1 ± 0.7**	82.2 ± 3.2	54.6 ± 2.5	73.7 ± 0.5	50.8 ± 0.8	79.7 ± 2.2
	JSD	86.7 ± 0.6	72.9 ± 0.6	79.4 ± 1.0	**89.7 ± 1.1**	**62.5 ± 1.7**	**74.2 ± 0.7**	**51.1 ± 0.5**	**84.5 ± 0.6**
	NT-XENT	**86.8 ± 0.5**	72.9 ± 0.6	79.3 ± 0.8	75.4 ± 7.8	51.2 ± 3.3	63.6 ± 4.2	50.4 ± 0.6	82.0 ± 1.1
	DV	85.4 ± 0.6	**73.3 ± 0.5**	78.9 ± 0.8	83.4 ± 1.9	56.7 ± 2.5	72.5 ± 0.8	**51.1 ± 0.5**	76.3 ± 5.6
MODE	LOCAL-GLOBAL	**86.8 ± 0.5**	**73.3 ± 0.5**	**80.1 ± 0.7**	**89.7 ± 1.1**	**62.5 ± 1.7**	**74.2 ± 0.7**	**51.1 ± 0.5**	**84.5 ± 0.6**
	GLOBAL	–	–	–	85.4 ± 2.8	56.0 ± 2.1	72.4 ± 0.4	49.7 ± 0.8	80.8 ± 1.8
	MULTI-SCALE	83.2 ± 0.9	63.5 ± 1.5	75.7 ± 1.1	88.0 ± 0.8	56.6 ± 1.8	72.7 ± 0.4	50.6 ± 0.5	82.8 ± 0.6
	HYBRID	–	–	–	86.1 ± 1.7	56.1 ± 1.4	73.3 ± 1.2	49.6 ± 0.6	78.2 ± 4.2
	ENSEMBLE	86.2 ± 0.6	**73.3 ± 0.5**	79.7 ± 0.9	82.5 ± 1.9	54.0 ± 3.0	73.0 ± 0.4	49.9 ± 0.9	81.4 ± 1.8
VIEWS	ADJ-PPR	**86.8 ± 0.5**	**73.3 ± 0.5**	**80.1 ± 0.7**	**89.7 ± 1.1**	**62.5 ± 1.7**	**74.2 ± 0.7**	51.1 ± 0.5	**84.5 ± 0.6**
	ADJ-HEAT	86.4 ± 0.5	71.8 ± 0.5	77.2 ± 1.2	85.0 ± 1.9	55.8 ± 1.1	72.8 ± 0.5	50.0 ± 0.6	81.6 ± 0.9
	ADJ-DIST	84.5 ± 0.6	72.7 ± 0.7	74.6 ± 1.4	87.1 ± 1.0	58.7 ± 2.2	72.0 ± 0.7	50.7 ± 0.6	81.8 ± 0.7
	PPR-HEAT	85.8 ± 0.5	72.9 ± 0.5	78.1 ± 0.9	87.7 ± 1.2	57.6 ± 1.6	72.2 ± 0.6	**51.2 ± 0.8**	82.3 ± 1.0
	PPR-DIST	85.9 ± 0.5	73.2 ± 0.4	74.7 ± 1.2	87.1 ± 1.0	60.0 ± 2.5	72.4 ± 1.4	51.1 ± 0.8	82.5 ± 1.1
	HEAT-DIST	85.2 ± 0.4	70.4 ± 0.7	72.8 ± 0.7	87.4 ± 1.2	58.6 ± 1.7	72.2 ± 0.6	50.5 ± 0.5	80.3 ± 0.6

- Contrasting local-global outperforms other methods
- Contrasting encodings from adjacency and PPR views performs better

[1] Hassani, Kaveh, and Amir Hosein Khasahmadi. "Contrastive multi-view representation learning on graphs." International Conference on Machine Learning. PMLR, 2020.

Overview of Part II

Multi-network Mining Algorithms

Classification
- Label propagation-based multi-view/domain classification
- GNN-based embedding
- Contrastive learning for multi-view

Hyperlink prediction
- **NMF-based method**
- Autoencoder-based embedding
- GNN-based embedding

Multi-network association
- Label propagation-based method
- w/o attribute
- w/ attribute
- Dependency inference
- Network alignment

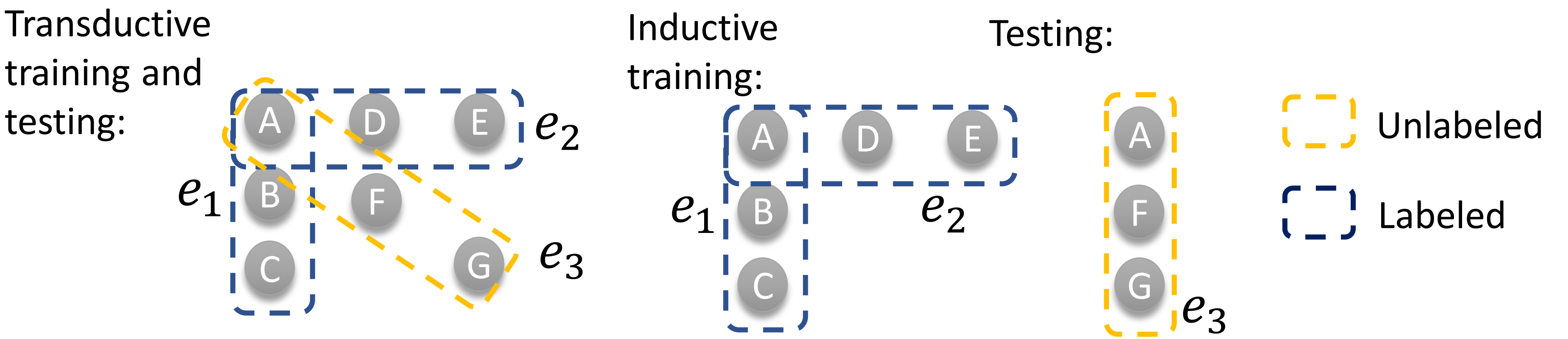
Ranking
- Consistency based homogeneous
- Consistency based heterogeneous

Clustering
- Multi-view network clustering
- NoN clustering

Multi-network embedding
- MF-based embedding
- Random walk-based embedding
- GNN-based embedding

Background: Hyperlink Prediction

- Problem settings:
  - Transductive setting:
    - Given: a tuple (H, D), where H = (V,E) is an incomplete hypernetwork
    - D is a set of candidate hyperlinks
    - Find: the most likely hyperlinks that are missing from H from D
  - Inductive setting:
    - Given: H = (V,E) is a given incomplete hypernetwork
    - Find: the the most likely hyperlinks that are missing from H from D
    - D is only seen when testing



CMM: Coordinated Matrix Minimization

- Problem definition (transductive setting):
  - Given $H = (V, E)$, incidence matrix: **S**
  - Find true hyperlinks in candidate set D (incidence matrix: $\mathbf{U}$)
- Observation:
  - Given incidence matrix $\boldsymbol{S} = \{0,1\} \in \mathbb{R}^{n\times m}$:
  - Hyperlink **s** (a column vector $\boldsymbol{S}$) -> $\boldsymbol{s}\boldsymbol{s}^{\mathbf{T}}$ vertex adjacency space
  - Abundant existing link prediction techniques for pairwise relation
- Key Ideas:
  - Infer the pairwise relationships in the adjacency space
  - Find the missing hyperlinks through constrained optimization
  - Two-step EM style optimization method

[1] Zhang, Muhan, Zhicheng Cui, Shali Jiang, and Yixin Chen. "Beyond link prediction: Predicting hyperlinks in adjacency space." In Thirty-Second AAAI Conference on Artificial Intelligence. 2018.

CMM: Formulation

- Objective function:

$$\min_{\boldsymbol{\Lambda},\mathbf{W}} ||\mathbf{A} + \mathbf{U}\boldsymbol{\Lambda}\mathbf{U}^{\mathrm{T}} - \mathbf{W}\mathbf{W}^{\mathrm{T}}||_F^2$$

Subject to $\lambda_i \in \{0,1\}, i = 1, \dots, m'$

$$\mathbf{W} \geq \mathbf{0}$$

Complete incidence matrix $[\mathbf{S}, \Delta\mathbf{S}][\mathbf{S}, \Delta\mathbf{S}]^{\mathrm{T}} = \mathbf{A} + \Delta\mathbf{A}$

$\Delta\mathbf{A} = \mathbf{U}\boldsymbol{\Lambda}\mathbf{U}^{\mathrm{T}}$

$\boldsymbol{\Lambda} = diag([\lambda_1, \dots, \lambda_{m'}])$: indicator matrix for columns of $\mathbf{U}$

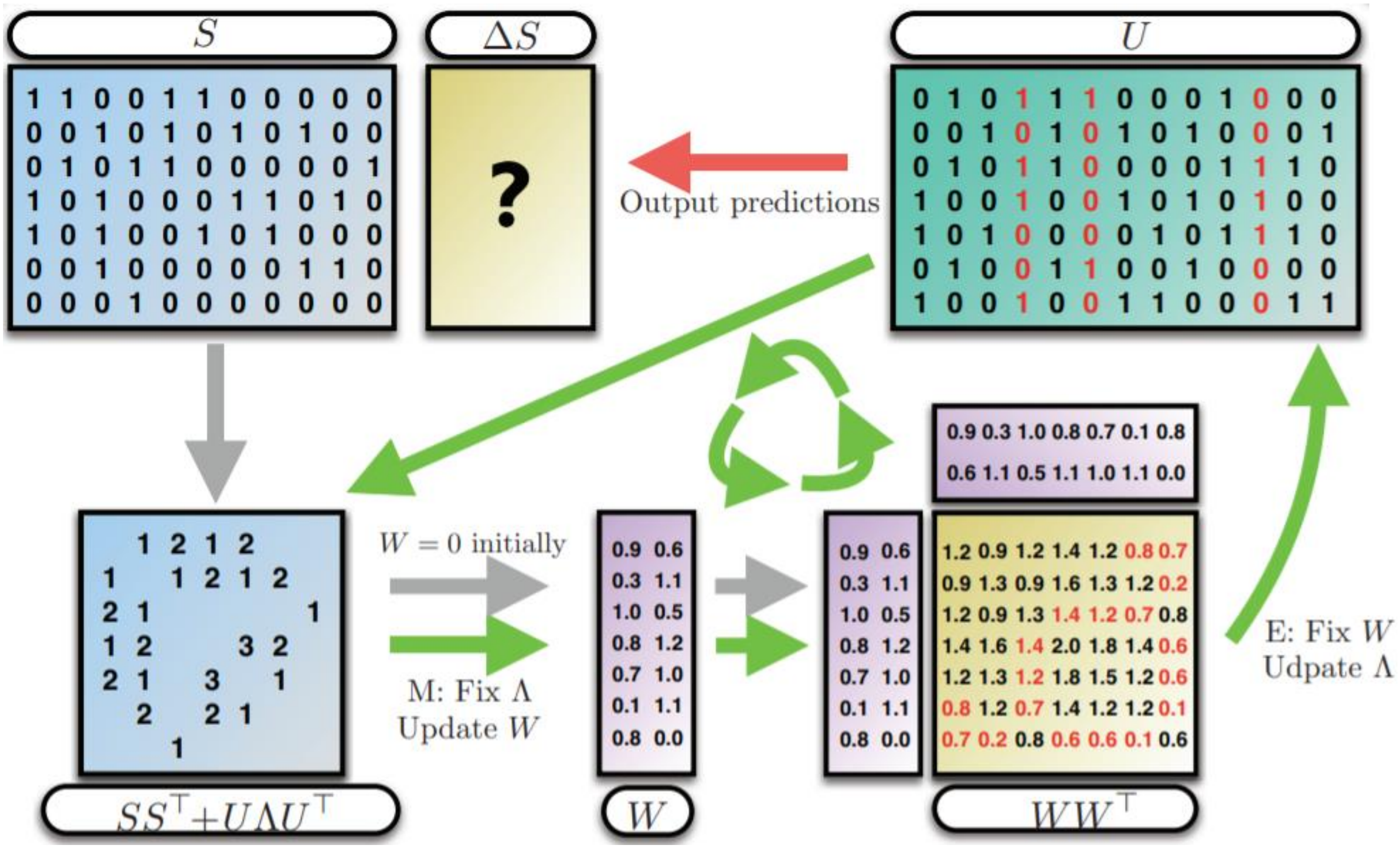
$\lambda_i = 1$: hyperlink u_i is a column in $\Delta\mathbf{S}$

$\lambda_i = 0$: otherwise

- E step (fix $\mathbf{W}$):
  - $\min_{\boldsymbol{\Lambda}} ||\mathbf{A} + \mathbf{U}\boldsymbol{\Lambda}\mathbf{U}^{\mathrm{T}} - \mathbf{W}\mathbf{W}^{\mathrm{T}}||_F^2$, Subject to $\lambda_i \in \{0,1\}, i = 1, \dots, m'$
- M step (fix $\boldsymbol{\Lambda}$):
  - $\min_{\mathbf{W}} ||\mathbf{A} + \mathbf{U}\boldsymbol{\Lambda}\mathbf{U}^{\mathrm{T}} - \mathbf{W}\mathbf{W}^{\mathrm{T}}||_F^2$, Subject to $\mathbf{W} \geq \mathbf{0}$

[1] Zhang, Muhan, Zhicheng Cui, Shali Jiang, and Yixin Chen. "Beyond link prediction: Predicting hyperlinks in adjacency space." In Thirty-Second AAAI Conference on Artificial Intelligence. 2018.

CMM: Algorithm



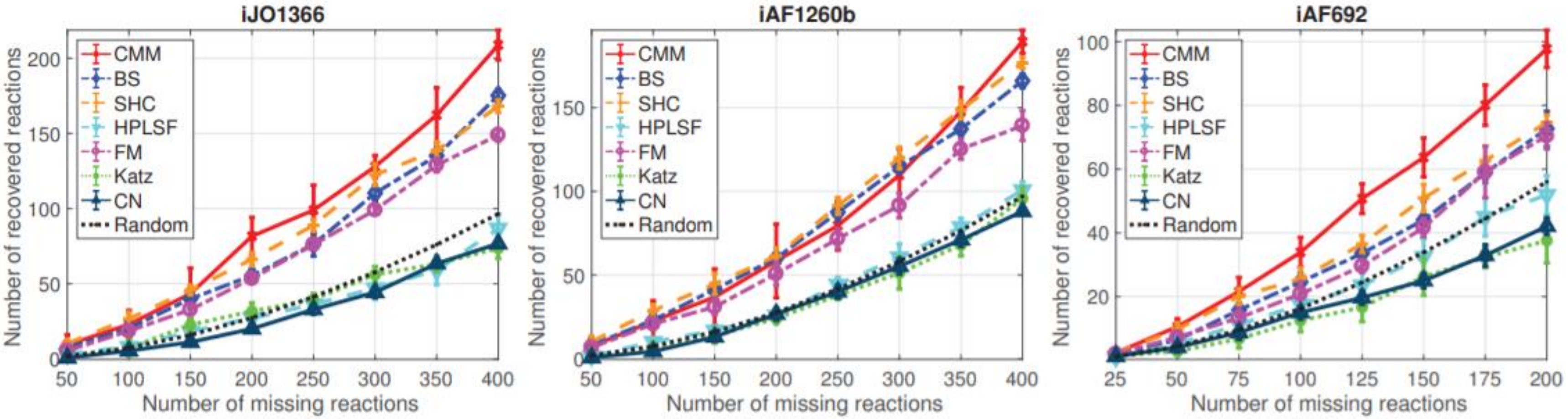
Updating details:

- E step: $\min_{\mathbf{x}} ||\mathbf{Cx}-\mathbf{d}||_2^2, s.t. \mathbf{x} \in \{0,1\}^{m'}$ $(\mathbf{C} = [\mathbf{c}_1, \ldots, \mathbf{c}_{m'}]^{\mathrm{T}}, \mathbf{c}_i = \mathrm{vec}(\mathbf{u}_i\mathbf{u}_i^{\mathrm{T}}))$
- M step: $\mathbf{x}_{\mathrm{new}} = [\mathbf{x} - \alpha\mathbf{H}^{-1}\nabla f(\mathbf{x})]^{+}, \mathbf{x} = \mathrm{vec}(\mathbf{W}), \mathbf{H}$: Hessian matrix of $f(\mathbf{x})$

[1] Zhang, Muhan, Zhicheng Cui, Shali Jiang, and Yixin Chen. "Beyond link prediction: Predicting hyperlinks in adjacency space." In Thirty-Second AAAI Conference on Artificial Intelligence. 2018.

CMM: Experiments

- Task: Predicting metabolic reactions
- Number of recovered reactions under different numbers of missing reactions:



(a) iJO1366 dataset. (b) iAF1260b dataset. (c) iAF692 dataset.

- CMM generally achieves the best performance

[1] Zhang, Muhan, Zhicheng Cui, Shali Jiang, and Yixin Chen. "Beyond link prediction: Predicting hyperlinks in adjacency space." In Thirty-Second AAAI Conference on Artificial Intelligence. 2018.

Overview of Part II

Multi-network Mining Algorithms

Classification
- Label propagation-based multi-view/domain classification
- GNN-based embedding
- Contrastive learning for multi-view

Hyperlink prediction
- NMF-based method
- **Autoencoder-based embedding**
- GNN-based embedding

Multi-network association
- Label propagation-based method
- w/o attribute
- w/ attribute
- Dependency inference
- Network alignment

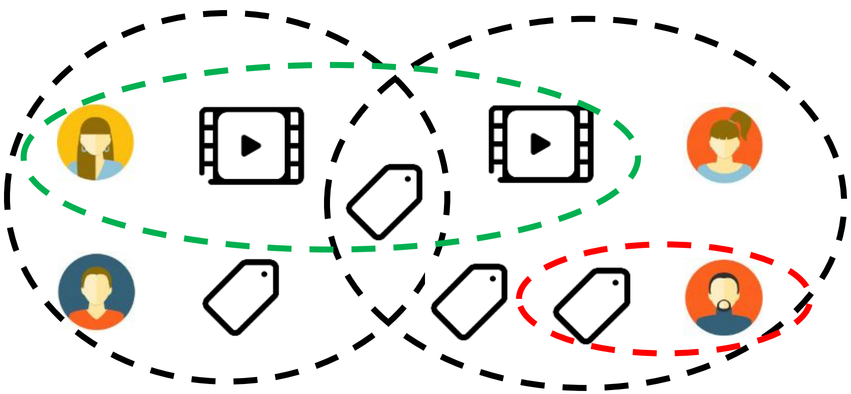
Ranking
- Consistency based homogeneous
- Consistency based heterogeneous

Clustering
- Multi-view network clustering
- NoN clustering

Multi-network embedding
- MF-based embedding
- Random walk-based embedding
- GNN-based embedding

DHNE: Deep Hyper-Network Embedding

- Observations:
  - Indecomposablity: the hyperedges are usually indecomposable.
    - Nodes in hyperedge have strong relationship ≠ nodes in subset have a strong relationship.
  - Structure Preserving: local and global structure
    - Local structures: not sufficient because of network sparsity
    - Global structures: use the neighborhood structure



User-movie-tag: strong relation

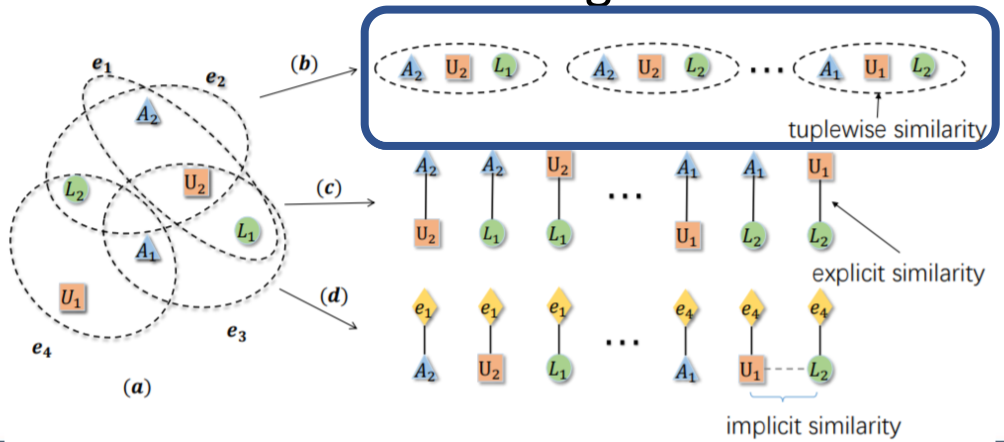
User-tag: not strong

Many missing links

[1] Tu, Ke, et al. "Structural deep embedding for hyper-networks." *Thirty-Second AAAI Conference on Artificial Intelligence*. 2018.

DHNE: Key Ideas

- Indecomposablity issue:
    - Indecomposable tuple-wise similarity function
    - Defined over all the nodes in a hyperedge
    - Tuple-wise similarity function as a deep neural network
- Structure preserving issue:
    - Deep autoencoder to learn node representations
    - Reconstruct neighborhood structures
    - Nodes with similar neighborhood structures -> similar embeddings

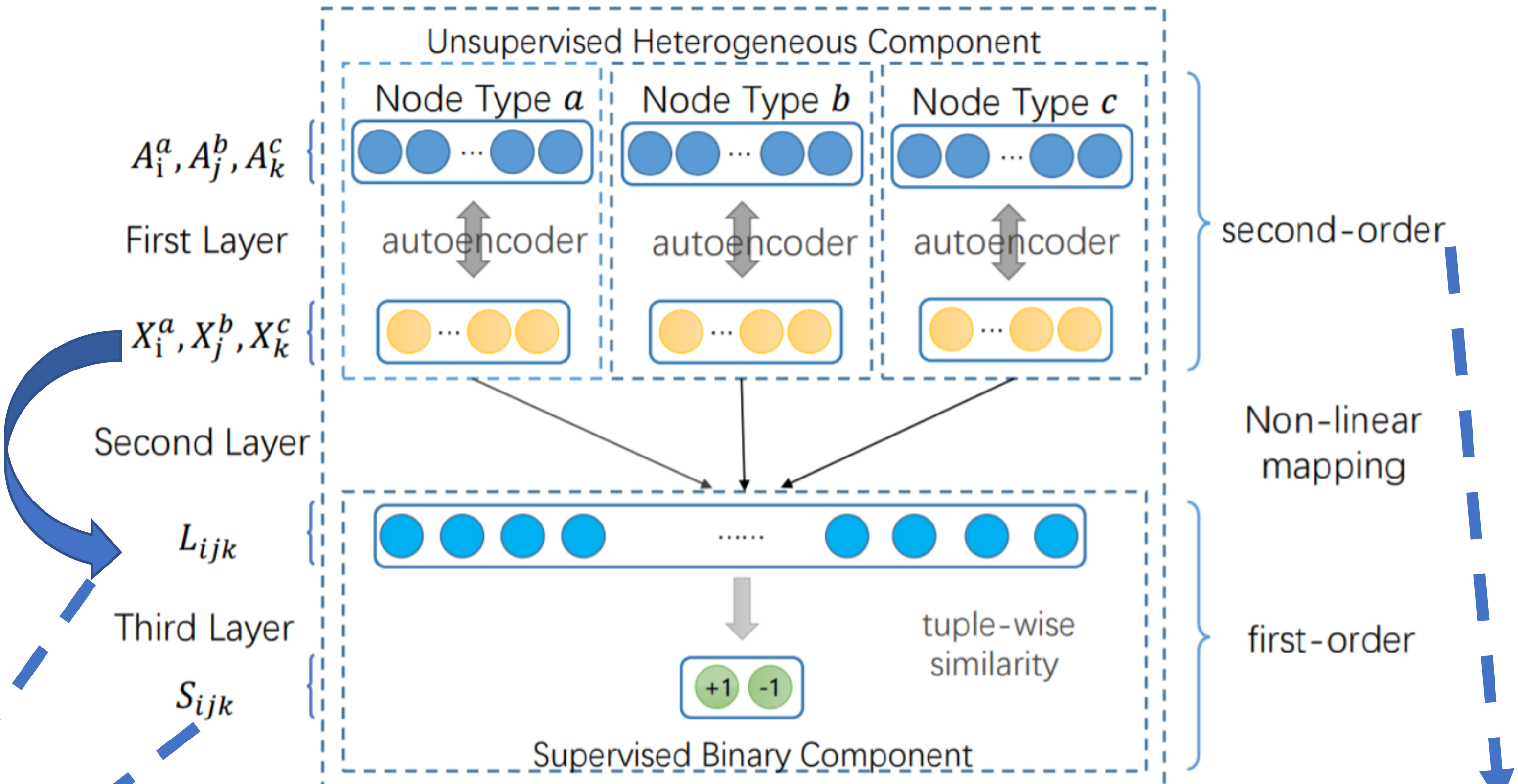


Proposed tuple-wise similarity

Comparison with other expansion method

[1] Tu, Ke, et al. "Structural deep embedding for hyper-networks." *Thirty-Second AAAI Conference on Artificial Intelligence*. 2018.

DHNE: Model Overview



$$\mathbf{L}_{ijk} = \sigma(\mathbf{W}_a^{(2)} * \mathbf{X}_i^a + \mathbf{W}_b^{(2)} * \mathbf{X}_j^b + \mathbf{W}_c^{(2)} * \mathbf{X}_k^c + \mathbf{b}^{(2)})$$

$$\mathcal{L}_2 = \sum ||sign(\mathbf{A}_i^t) \odot (\mathbf{A}_i^t - \hat{\mathbf{A}}_i^t)||_F^2$$

$$\mathbf{S}_{ijk} \equiv \mathcal{S}(\mathbf{X}_i^a, \mathbf{X}_j^b, \mathbf{X}_k^c) = \sigma(\mathbf{W}^{(3)} * \mathbf{L}_{ijk} + \mathbf{b}^{(3)})$$

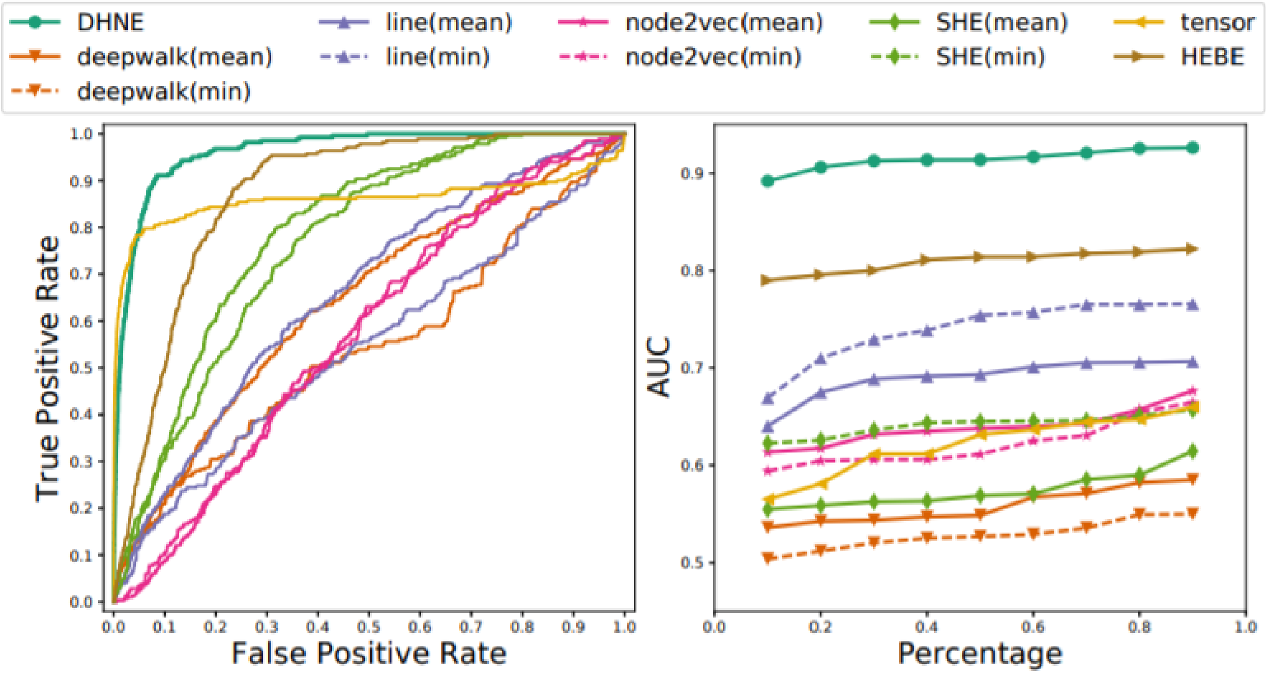
$$\mathcal{L}_1 = -(\mathbf{R}_{ijk} \log \mathbf{S}_{ijk} + (1 - \mathbf{R}_{ijk}) \log(1 - \mathbf{S}_{ijk}))$$

Total loss: $\mathcal{L} = \mathcal{L}_1 + \alpha \mathcal{L}_2$

[1] Tu, Ke, et al. "Structural deep embedding for hyper-networks." *Thirty-Second AAAI Conference on Artificial Intelligence*. 2018.

DHNE: Experiments

- Task: link prediction on GPS data ((user, location, activity) relations are used for building the hypernetwork)
  - left: ROC curve on GPS; right: Performance for link prediction on networks of different sparsity



- Achieves significant improvements over the baselines

[1] Tu, Ke, et al. "Structural deep embedding for hyper-networks." *Thirty-Second AAAI Conference on Artificial Intelligence*. 2018.

DHNE: Experiments

- Hyperlink prediction AUC value:

methods		GPS	MovieLens	drug	wordnet
DHNE		**0.9166**	**0.8676**	**0.9254**	**0.8268**
mean	deepwalk	0.6593	0.7151	0.5822	0.5952
	line	0.7795	0.7170	0.7057	0.6819
	node2vec	0.5835	0.8211	0.6573	0.8003
	SHE	0.8687	0.7459	0.5899	0.5426
min	deepwalk	0.5715	0.6307	0.5493	0.5542
	line	0.7219	0.6265	0.7651	0.6225
	node2vec	0.5869	0.7675	0.6546	0.7985
	SHE	0.8078	0.8012	0.6508	0.5507
tensor		0.8646	0.7201	0.6470	0.6516
HEBE		0.8355	0.7740	0.8191	0.6364

- DHNE outperforms all baselines

[1] Tu, Ke, et al. "Structural deep embedding for hyper-networks." *Thirty-Second AAAI Conference on Artificial Intelligence*. 2018.

Overview of Part II

Multi-network Mining Algorithms

Classification
- Label propagation-based multi-view/domain classification
- GNN-based embedding
- Contrastive learning for multi-view

Hyperlink prediction
- NMF-based method
- Autoencoder -based embedding
- **GNN-based embedding**

Multi-network association
- Label propagation-based method
- w/o attribute
- w/ attribute
- Dependency inference
- Network alignment

Ranking
- Consistency based homogeneous
- Consistency based heterogeneous

Clustering
- Multi-view network clustering
- NoN clustering

Multi-network embedding
- MF-based embedding
- Random walk-based embedding
- GNN-based embedding

HyperGCN: Graph Convolutional Networks for Hypergraphs

- Motivation:
  - How to apply Graph Convolutional Networks on hypergraphs?
- First trial:
  - Apply clique expansion on each hyperedge -> regular graph
  - GNNs for the transformed regular graph
  - Disadvantage: $O(n^2)$ edges in each hyperedge of regular graph
- Question:
  - How to transform hyperedges to have *linear* regular edges?
- The proposed method (densest k-subhypergraph):

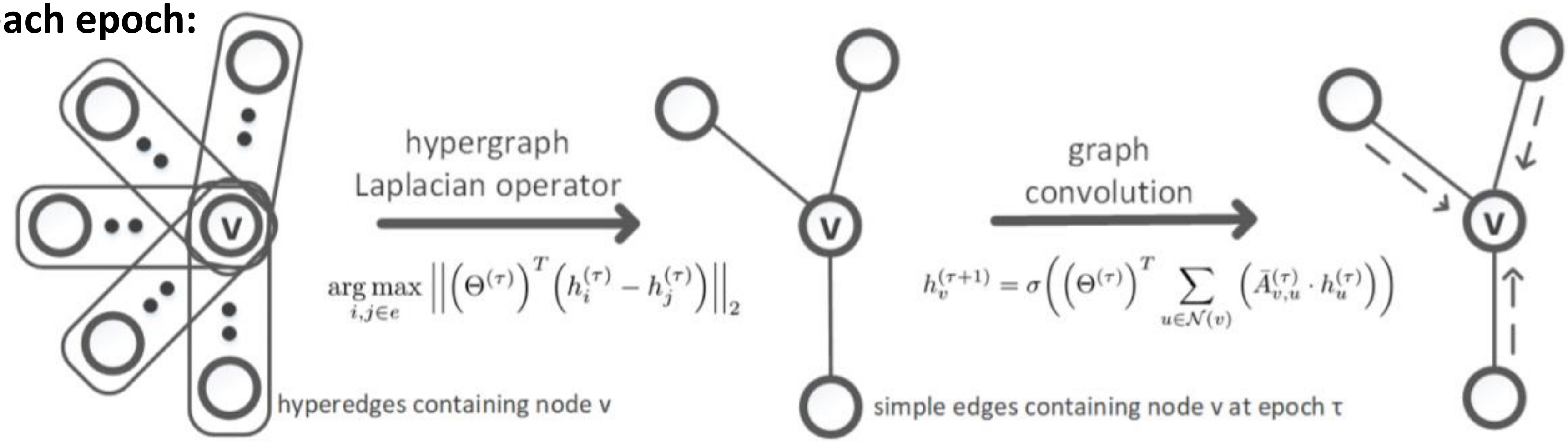
Model↓ Metric →	Training time	Density	DBLP	Pubmed
HGNN	170s	337	0.115s	0.019s
FastHyperGCN	**143s**	**352**	**0.035s**	**0.016s**

[1] Yadati, Naganand, et al. "HyperGCN: A new method of training graph convolutional networks on hypergraphs." *arXiv preprint arXiv:1809.02589* (2018).

HyperGCN: Model Details

- 1-HyperGCN:
  - Ideas:
    - Node representations within one hyperedge should be close
      $\Rightarrow$ $\sum_{e \in E} \max_{i,j \in e} ||\mathbf{h}_i - \mathbf{h}_j||_2^2$ should be small $\Rightarrow$ regularizer?
    - Select one representative edge for each hyperedge
  - Step 1: Find the hypergraph Laplacian with max node difference
  - Step 2: Apply GCN on the reduced regular graph

At each epoch:

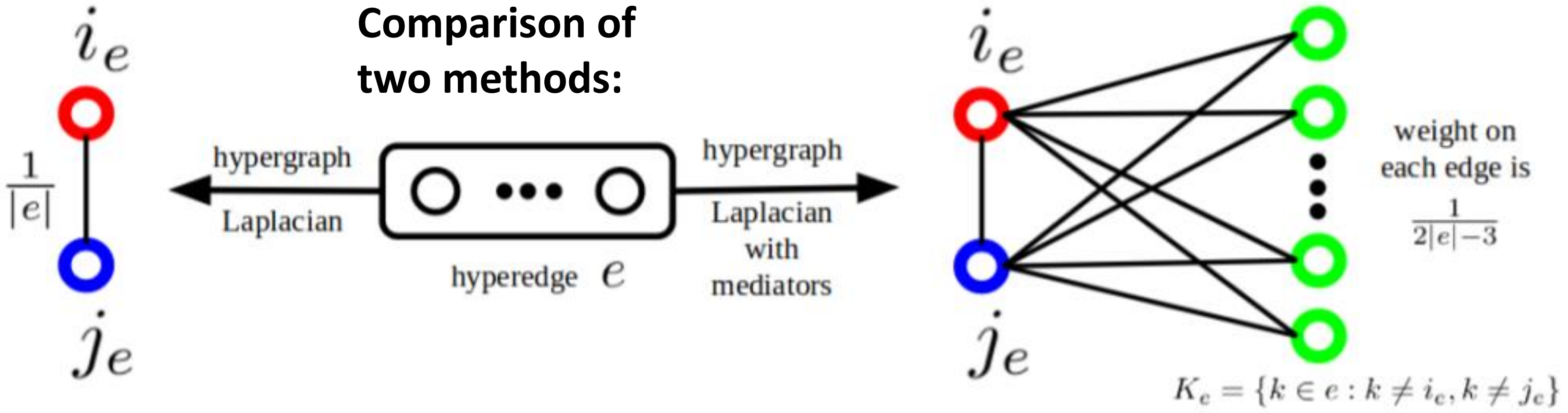


[1] Yadati, Naganand, et al. "HyperGCN: A new method of training graph convolutional networks on hypergraphs." *arXiv preprint arXiv:1809.02589* (2018).

HyperGCN: Model Details (cont'd)

- HyperGCN: enhanced 1-HyperGCN with mediators:
  - 1 edge might be insufficient to represent the whole hyperedge
  - Generalized hypergraph Laplacian: the rest of nodes as "mediators"
  - Number of edges: $2|e| - 3$
- FastHyperGCN:
  - Apply the enhanced 1-HyperGCN with initial features
  - Use fixed transformed hyperedge in every epoch

Comparison of two methods:



[1] Yadati, Naganand, et al. "HyperGCN: A new method of training graph convolutional networks on hypergraphs." *arXiv preprint arXiv:1809.02589* (2018).

HyperGCN: Experiments

- Results of SSL experiments
  - Mean test error ± standard deviation (lower is better)

Data	Method	DBLP co-authorship	Pubmed co-citation	Cora co-authorship	Cora co-citation	Citeseer co-citation
$\mathcal{H}$	CI	54.81 ± 0.9	52.96 ± 0.8	55.45 ± 0.6	64.40 ± 0.8	70.37 ± 0.3
$\mathbf{X}$	MLP	37.77 ± 2.0	30.70 ± 1.6	41.25 ± 1.9	42.14 ± 1.8	41.12 ± 1.7
$\mathcal{H}, \mathbf{X}$	MLP + HLR	30.42 ± 2.1	30.18 ± 1.5	34.87 ± 1.8	36.98 ± 1.8	37.75 ± 1.6
$\mathcal{H}, \mathbf{X}$	HGNN	25.65 ± 2.1	29.41 ± 1.5	31.90 ± 1.9	$\mathbf{32.41 \pm 1.8}$	$\mathbf{37.40 \pm 1.6}$
$\mathcal{H}, \mathbf{X}$	1-HyperGCN	33.87 ± 2.4	30.08 ± 1.5	36.22 ± 2.2	34.45 ± 2.1	38.87 ± 1.9
$\mathcal{H}, \mathbf{X}$	FastHyperGCN	27.34 ± 2.1	29.48 ± 1.6	32.54 ± 1.8	$\mathbf{32.43 \pm 1.8}$	$\mathbf{37.42 \pm 1.7}$
$\mathcal{H}, \mathbf{X}$	HyperGCN	$\mathbf{24.09 \pm 2.0}$	$\mathbf{25.56 \pm 1.6}$	$\mathbf{30.08 \pm 1.8}$	$\mathbf{32.37 \pm 1.7}$	$\mathbf{37.35 \pm 1.6}$

Uses the clique expansion to approximate the hypergraph

The graph of HGNN and all methods of this work: normalized clique expansion when maximum size of a hyperedge is 3

[1] Yadati, Naganand, et al. "HyperGCN: A new method of training graph convolutional networks on hypergraphs." *arXiv preprint arXiv:1809.02589* (2018).

HyperGCN: Experiments

- HyperGCN for combinatorial optimization
  - Results on the densest k-subhypergraph problem:
  - Given a hypergraph (V, E), to find a subset W ⊆ V of k nodes so as to maximize the number of hyperedges contained in V
  - Density (higher is better) of the set of vertices: obtained by each of the proposed approaches for k = 3|V|/4

Dataset→ Approach↓	Synthetic test set	DBLP co-authorship	Pubmed co-citation	Cora co-authorship	Cora co-citation	Citeseer co-citation
MaxDegree	174 ± 50	4840	1306	194	544	507
RemoveMinDegree	147 ± 48	**7714**	**7963**	450	1369	843
MLP	174 ± 56	5580	1206	238	550	534
MLP + HLR	231 ± 46	5821	3462	297	952	764
HGNN	337 ± 49	6274	7865	437	**1408**	**969**
1-HyperGCN	207 ± 52	5624	1761	251	563	509
FastHyperGCN	352 ± 45	7342	7893	452	**1419**	**969**
HyperGCN	$\mathbf{359 \pm 49}$	**7720**	**7928**	**504**	**1431**	**971**
# hyperedges, $\lvert E \rvert$	500	22535	7963	1072	1579	1079

[1] Yadati, Naganand, et al. "HyperGCN: A new method of training graph convolutional networks on hypergraphs." *arXiv preprint arXiv:1809.02589* (2018).

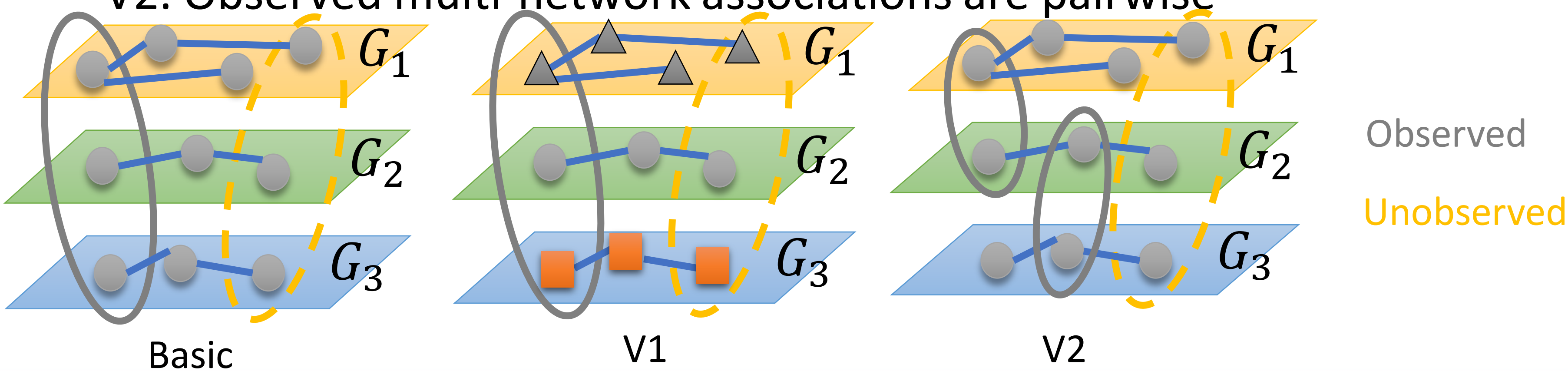
Overview of Part II

Multi-network Mining Algorithms

Classification	Hyperlink prediction	Multi-network association	Ranking	Clustering	Multi-network embedding
• Label propagation-based multi-view/domain classification • GNN-based embedding • Contrastive learning for multi-view	• NMF-based method • Autoencoder-based embedding • GNN-based embedding	• **Label propagation-based method** • w/o attribute • w/ attribute • Dependency inference • Network alignment	• Consistency based homogeneous • Consistency based heterogeneous	• Multi-view network clustering • NoN clustering	• MF-based embedding • Random walk-based embedding • GNN-based embedding

Multi-network Association

- Basic problem definition:
  - Given:
    - K networks $\{G_1, \dots, G_K\}$
    - Observed multi-network associations $\{(i_1, \dots, i_K), \dots\}, i_1 \in G_1, \dots, i_K \in G_K$
  - Output:
    - The values of the rest of unobserved multi-network associations
- Problem variants:
  - V1. K networks are heterogeneous (multi-relational associations)
  - V2. Observed multi-network associations are pairwise



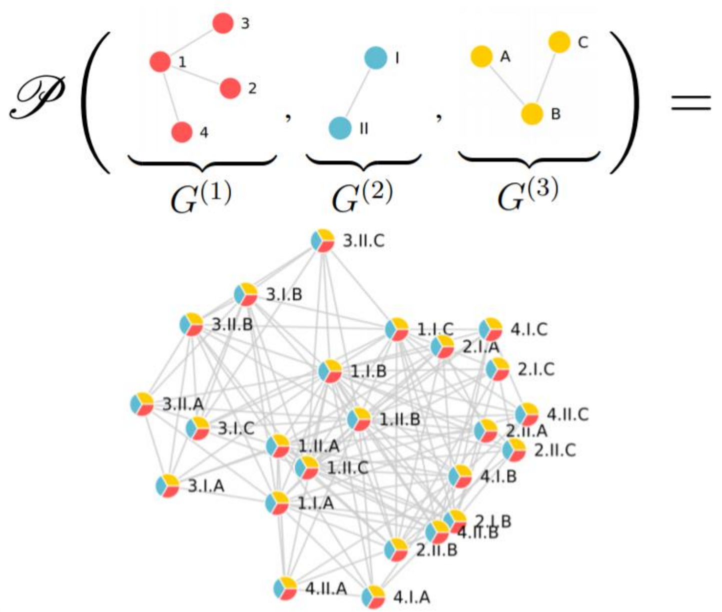
TOP: Transductive Learning over Product Graph

- Problem setting:
  - Transductive, cross-graph multi-relational learning (CGRL)
- Key ideas:
  - Heterogeneous graph sources -> single homogeneous graph
    - Via product graph
    - Adv.: Simplify problem formulation
  - Adopt a convex formulation and approximation of the CGRL
    - Adv.: Ensure robust optimization and efficient computation
  - Label propagation over the induced homogeneous graph
    - Adv. 1: Enables transductive learning
    - Adv. 2: Address label-sparsity by massively available non-observed tuples

[1] Liu, Hanxiao, and Yiming Yang. "Cross-graph learning of multi-relational associations." *International Conference on Machine Learning*. PMLR, 2016.

TOP: Product Graph

- Task 1: cross-network multi-relation learning
  - Given J graphs $G^{(1)}, \ldots, G^{(J)}$ with labeled multi-relations $O = \{(i_1, \ldots, i_J)\}$
  - Predict labels of the unlabeled multi-relations
- Task 2: label prediction on product graph
  - Given product graph P with labeled vertices $O = \{(i_1, \ldots, i_J)\}$
  - Predict labels of its unlabeled vertices.



Heterogeneous graphs

Single homogeneous graph

[1] Liu, Hanxiao, and Yiming Yang. "Cross-graph learning of multi-relational associations." *International Conference on Machine Learning*. PMLR, 2016.

TOP: Product Graph (cont'd)

- Spectral Graph Product (SGP) operator:
  - Given $G^{(1)}, \dots, G^{(J)}$
  - $\mathcal{P}(G^{(1)}, \dots, G^{(J)})$ is defined by the eigen system:

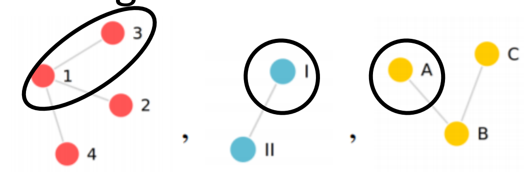
$$\left\{\kappa\left(\lambda_{i_1}^{(1)}, \dots, \lambda_{i_2}^{(J)}\right), \otimes_j v_{i_j}^{(j)}\right\}_{i_1, \dots, i_J}$$

where κ is a pre-specified nonnegative nondecreasing function over $\lambda_{i_1}^{(1)}, \dots, \lambda_{i_2}^{(J)}$ (eigenvalues for $G^{(1)}, \dots, G^{(J)}$)

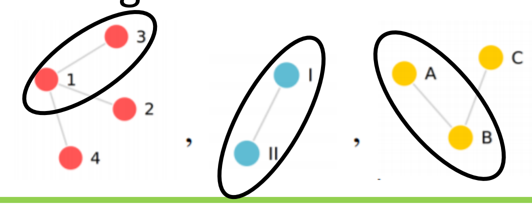
- E.g.,

SGP Type	$\kappa(\lambda_{i_1}^{(1)}, \cdots, \lambda_{i_J}^{(J)})$	$[\mathscr{P}_\kappa]_{(i_1,\cdots i_J),(i'_1,\cdots i'_J)}$
Tensor	$\prod_j \lambda_{i_j}^{(j)}$	$\prod_j G_{i_j, i'_j}^{(j)}$
Cartesian	$\sum_j \lambda_{i_j}^{(j)}$	$\sum_j G_{i_j, i'_j}^{(j)} \prod_{j' \neq j} \delta_{i_{j'} = i'_{j'}}$

- Cartesian product graph G^c:
- an edge exist iff



- Tensor product graph G^t:
- an edge exist iff



[1] Liu, Hanxiao, and Yiming Yang. "Cross-graph learning of multi-relational associations." *International Conference on Machine Learning*. PMLR, 2016.

TOP: Formulation

- Objective function:

$$\min_{f \in \mathbb{R}^{n_1 \times \cdots \times n_J}} \boxed{\ell_{\mathcal{O}}(f)} + \boxed{\frac{\gamma}{2}\|f\|^2_{\mathscr{P}_\kappa}}$$

Ranking l_2-hinge loss

$$\ell_{\mathcal{O}}(f) = \frac{\sum_{\substack{(i_1,\ldots,i_J) \in \mathcal{O} \\ (i'_1,\ldots,i'_J) \in \bar{\mathcal{O}}}} \left(f_{i_1 \ldots i_J} - f_{i'_1 \ldots i'_J}\right)_+^2}{|\mathcal{O} \times \bar{\mathcal{O}}|}$$

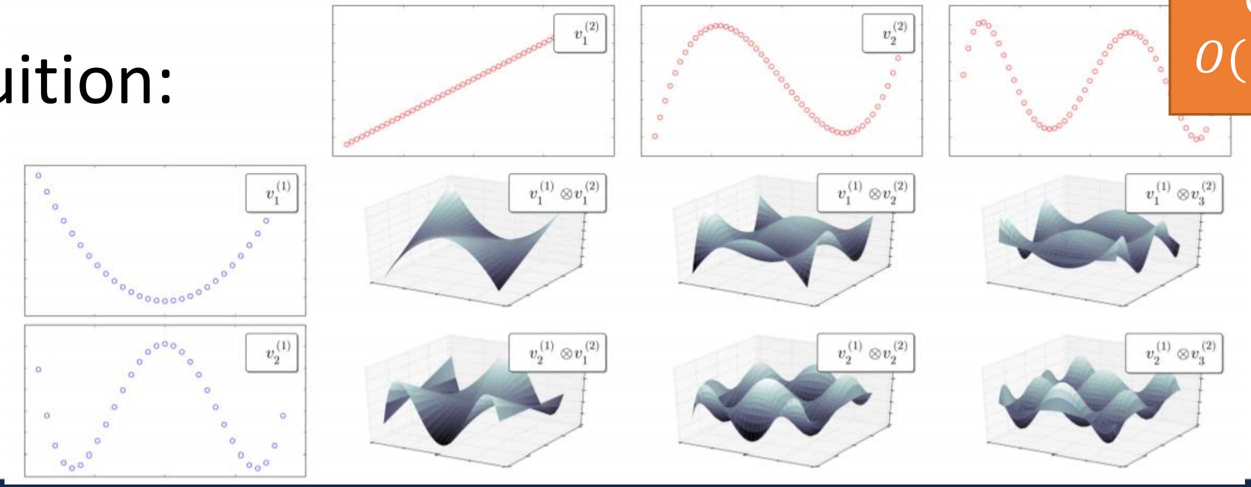
O: training set
$\bar{O}$: complement of O w.r.t. all possible multi-relations

Smoothness consistency on product graph

$$\|f\|^2_{\mathscr{P}_\kappa} = vec(f)^\top \mathscr{P}_\kappa^{-1} vec(f) = \sum_{i_1, i_2, \ldots, i_J} \frac{f\left(v_{i_1}^{(1)}, \ldots, v_{i_J}^{(J)}\right)^2}{\kappa\left(\lambda_{i_1}^{(1)}, \ldots, \lambda_{i_J}^{(J)}\right)}$$

Complexity: $O((\sum_j n_j)\prod_j n_j)$

- Intuition:



TOP: Approximation

- Include only the top-d_j eigenvectors in $V^{(j)}$ for each graph $G^{(j)}$
- Tensor f within the linear span of top $\prod_{j=1}^{J} d_j$ eigenvectors of the product graph

$$f = \sum_{k_1,\cdots,k_J=1}^{d_1,\cdots,d_J} \alpha_{k_1,\cdots,k_J} \bigotimes_j v_{k_j}^{(j)}$$
$$= \alpha \times_1 V^{(1)} \times_2 V^{(2)} \times_3 \cdots \times_J V^{(J)}$$

Tucker decomposition form

Original objective function:

$$\min_{f \in \mathbb{R}^{n_1 \times \cdots \times n_J}} \ell_{\mathcal{O}}(f) + \frac{\gamma}{2} \|f\|_{\mathscr{P}_\kappa}^2$$

Approximated objective function:

$$\min_{\alpha \in \mathbb{R}^{d_1 \times \cdots \times d_J}} \ell_{\mathcal{O}}(f) + \frac{\gamma}{2} \|\alpha\|_{\mathscr{P}_\kappa}^2$$
$$\text{s.t.} \quad f = \alpha \times_1 V^{(1)} \times_2 \cdots \times_J V^{(J)}$$

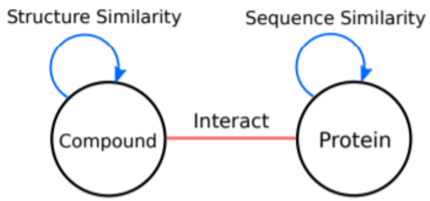
$$\|f\|_{\mathscr{P}_\kappa}^2 = \|\alpha\|_{\mathscr{P}_\kappa}^2 = \sum_{k_1,\ldots,k_J=1}^{d_1,\cdots,d_J} \frac{\alpha_{k_1,\cdots,k_J}^2}{\kappa(\lambda_{k_1}^{(1)}, \ldots, \lambda_{k_J}^{(J)})}$$

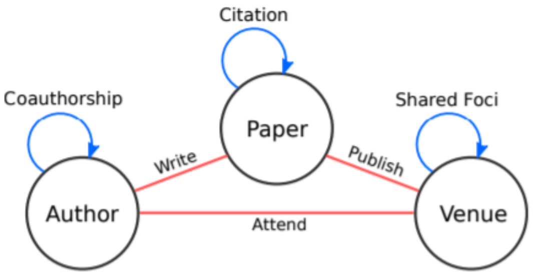
Complexity:
$O\left(\left(\sum_j n_j\right) \prod_j d_j\right)$
$d_j \ll n_j$

[1] Liu, Hanxiao, and Yiming Yang. "Cross-graph learning of multi-relational associations." *International Conference on Machine Learning*. PMLR, 2016.

TOP: Experiments

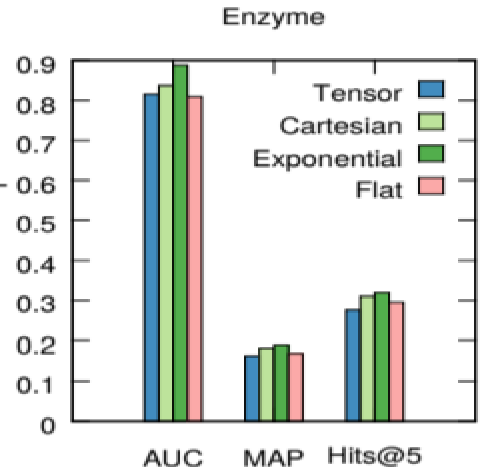
- Dataset:
  - Enzyme dataset for compound-protein interaction and the DBLP dataset of scientific publication record
  - The heterogeneous types of objects (the circles) and the relational structures

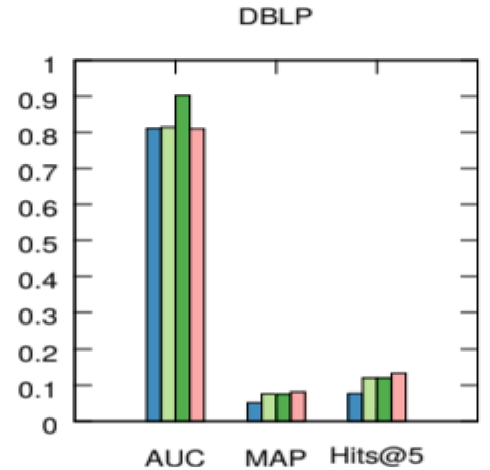




  - Performance of TOP with different SGPs:

Name	$\kappa(x, y)$ $(J = 2)$	$\kappa(x, y, z)$ $(J = 3)$
Tensor	xy	xyz
Cartesian	$x + y$	$x + y + z$
Exponential	e^{x+y}	$e^{xy+yz+xz}$
Flat	1	1

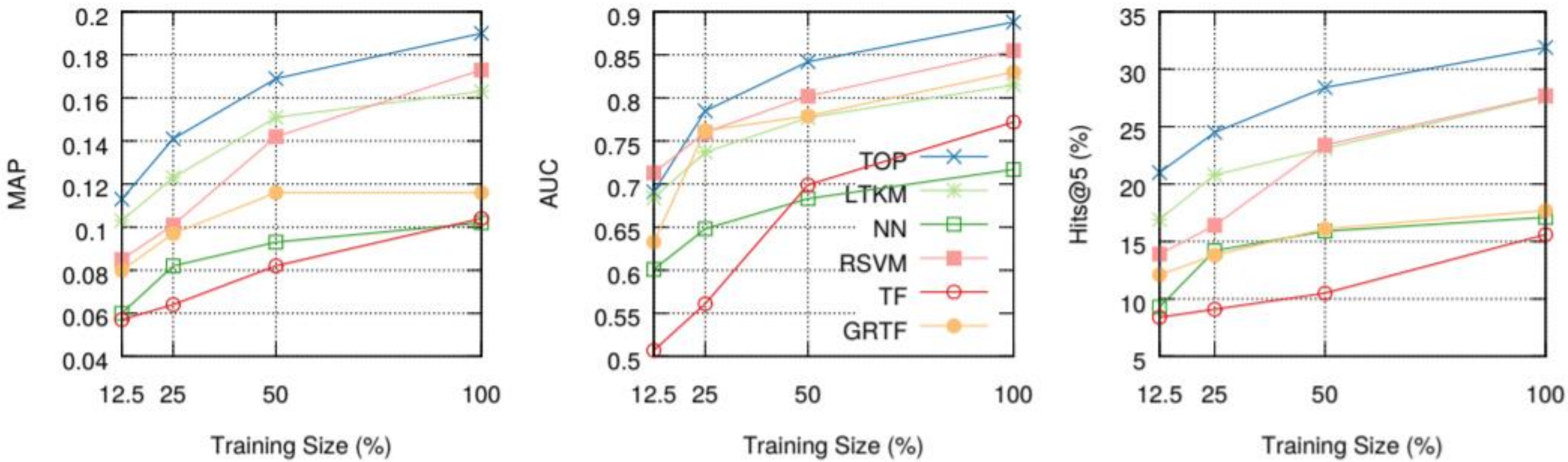




[1] Liu, Hanxiao, and Yiming Yang. "Cross-graph learning of multi-relational associations." *International Conference on Machine Learning*. PMLR, 2016.

TOP: Experiments

- Performance of different methods on Enzyme
  - Based on the quality of inferred target proteins given each compound



- Observation: Outperforms all baselines on all metrics

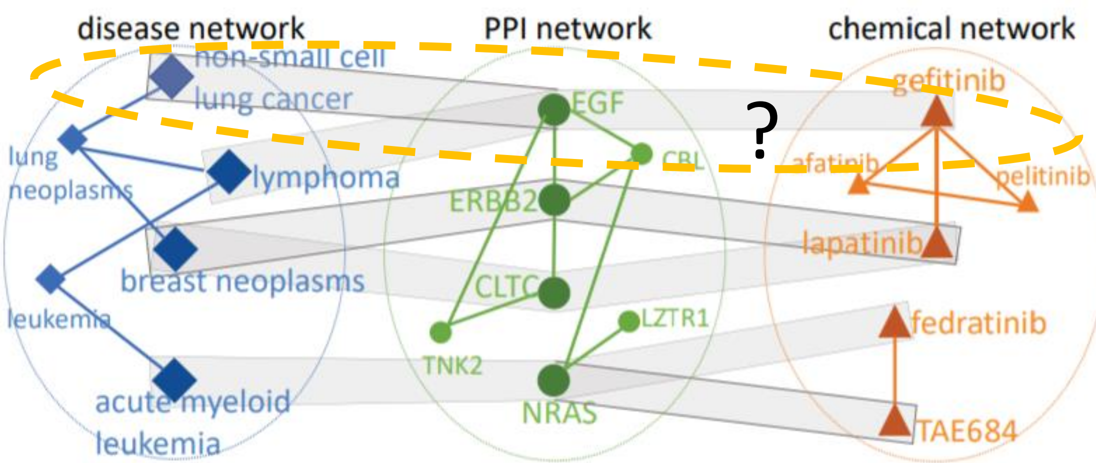
Overview of Part II

Multi-network Mining Algorithms

Classification	Hyperlink prediction	Multi-network association	Ranking	Clustering	Multi-network embedding
• Label propagation-based multi-view/domain classification • GNN-based embedding • Contrastive learning for multi-view	• NMF-based method • Autoencoder-based embedding • GNN-based embedding	• Label propagation-based method • **w/o attribute** • w/ attribute • Dependency inference • Network alignment	• Consistency based homogeneous • Consistency based heterogeneous	• Multi-view network clustering • NoN clustering	• MF-based embedding • Random walk-based embedding • GNN-based embedding

GT-COPR: Graph-Regularized Tensor Completion from Observed Pairwise Relations

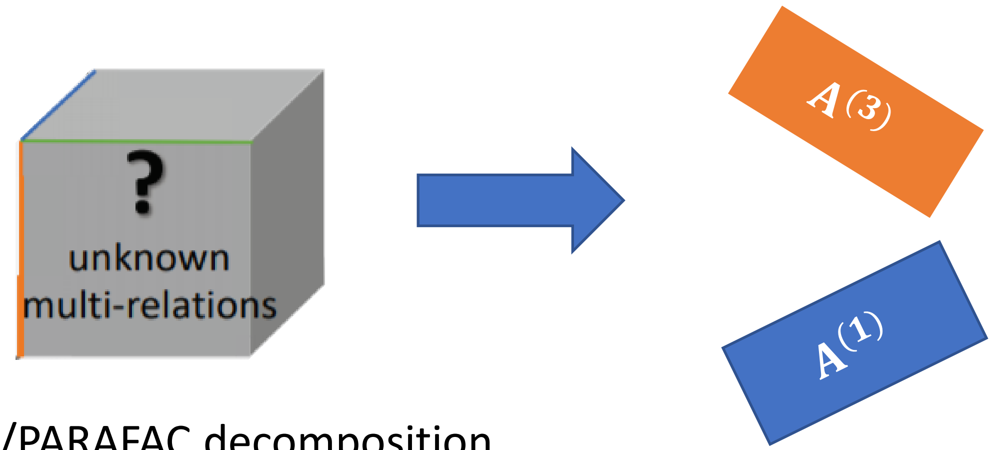
- Problem setting:
  - Observed multi-network associations are pairwise
  - Predict multi-network high-order multi-relational associations
- Limitation of existing work:
  - Observed high-order multi-relational associations are sparse
  - Do not utilize the observed pairwise multi-network associations



[1] Li, Zhuliu, et al. "Learning a Low-Rank Tensor of Pharmacogenomic Multi-relations from Biomedical Networks." *2019 IEEE International Conference on Data Mining (ICDM)*. IEEE, 2019.
[2] D. S. Wishart, Y. D. Feunang, A. C. Guo, E. J. Lo, A. Marcu, J. R. Grant, T. Sajed, D. Johnson, C. Li, Z. Sayeeda et al., "Drugbank 5.0: a major update to the drugbank database for 2018," Nucleic acids research, 2017

GT-COPR: Key Ideas

- Learn a compressed tensor in CPD-form*
  - Adv.: Space and time efficiencies for learning high-order multi-relations.
- Co-regularize tensor elements with the Laplacian of product graph
  - Adv.: Introduce local consistencies among n-way relations
- Tensor collapsing for capturing the cross-mode dependencies
  - Adv.: Preserve global consistencies with the observed bipartite relations



*CANDECOMP/PARAFAC decomposition

[1] Li, Zhuliu, et al. "Learning a Low-Rank Tensor of Pharmacogenomic Multi-relations from Biomedical Networks." *2019 IEEE International Conference on Data Mining (ICDM)*. IEEE, 2019.

GT-COPR: Formulation

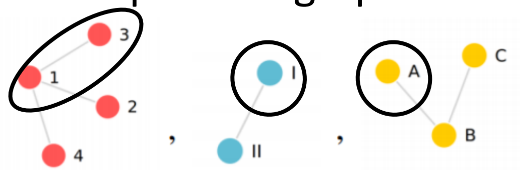
- Objective function:

$$\underset{\{A^{(i)}:\ i=1,\ldots,n\}}{\textit{minimize}} \quad \mathcal{J} = \sum_{i,j:\ i<j} \frac{1}{2} \| R_{i,j} - \frac{1}{\prod_{l \neq i,j} I_l} \boxed{\mathbf{collapse}(\hat{\mathcal{T}}, i, j)} \|_F^2$$

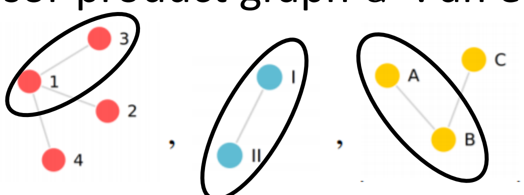
$$+ \frac{\lambda}{2} \boxed{\mathbf{vec}(\hat{\mathcal{T}})^T L \mathbf{vec}(\hat{\mathcal{T}})} + \frac{\beta}{2} \sum_{i=1}^{n} \| A^{(i)} \|_F^2$$

$$\textit{subject to} \quad A^{(i)} \geq 0, \forall i = 1, \ldots, n,$$

- Cartesian product graph G^c: an edge exist iff



- Tensor product graph G^t: an edge exist iff



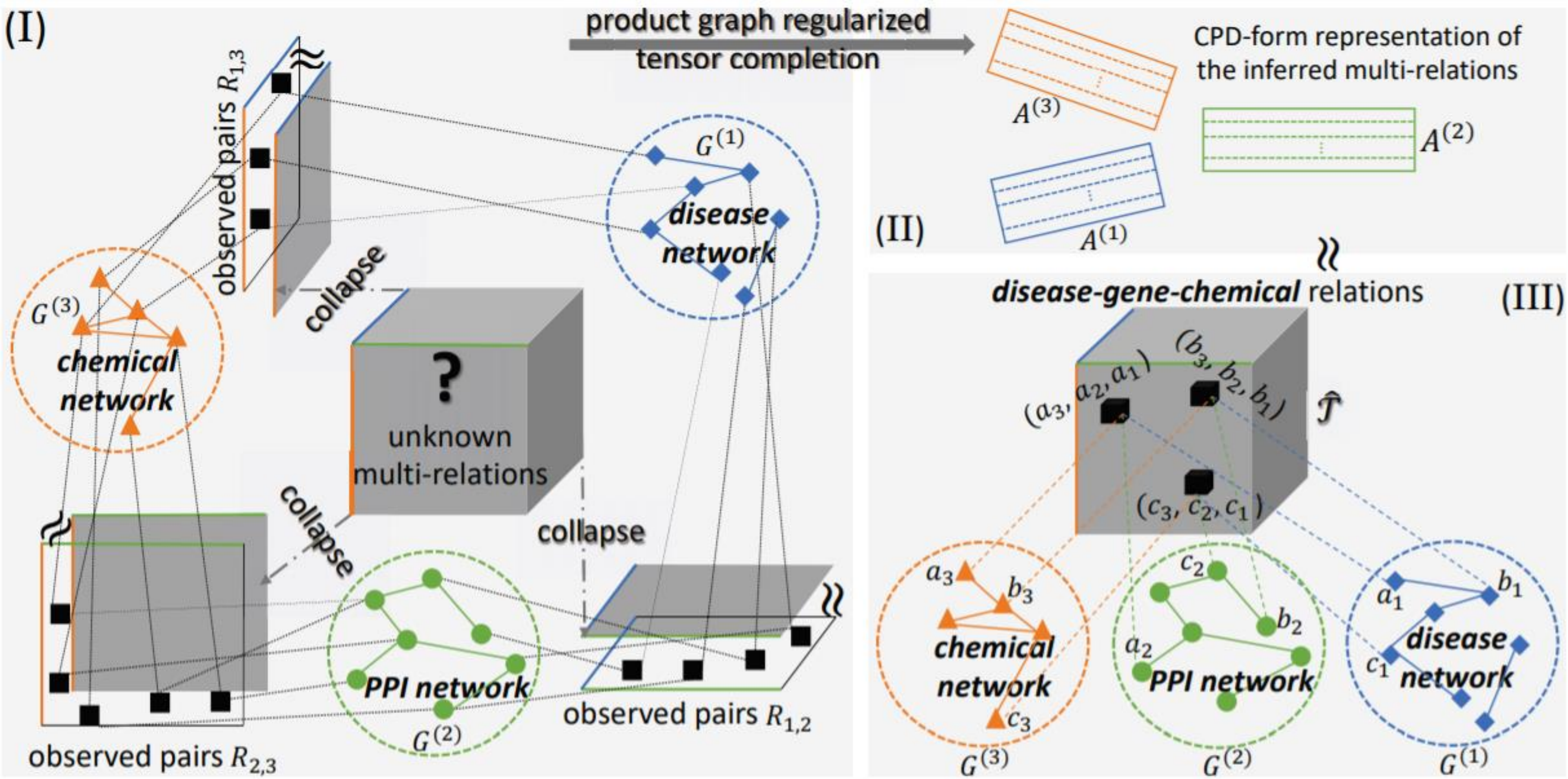
- Strong product graph G^s: the edge exists iff it is in either G^c or G^t

- Tensor $\mathcal{T} \in \mathbb{R}^{I_n \times I_{n-1} \times \cdots \times I_1}$ of inferred n-way relations is approximated by rank-K CPD form $\hat{\mathcal{T}} = [\![A^{(n)}, A^{(n-1)}, \ldots, A^{(1)}]\!]$
- **collapse**$(\hat{\mathcal{T}}, i, j)$: collapse $\hat{\mathcal{T}}$ into $I_i \times I_j$ matrix by summing over the slides along corresponding modes

L: Laplacian of a product graph

[1] Li, Zhuliu, et al. "Learning a Low-Rank Tensor of Pharmacogenomic Multi-relations from Biomedical Networks." *2019 IEEE International Conference on Data Mining (ICDM)*. IEEE, 2019.

GT-COPR: Algorithm Overview



[1] Li, Zhuliu, et al. "Learning a Low-Rank Tensor of Pharmacogenomic Multi-relations from Biomedical Networks." *2019 IEEE International Conference on Data Mining (ICDM)*. IEEE, 2019.

GT-COPR: Algorithm Details

- Derivatives:

$$\frac{\partial \mathcal{J}_1}{\partial A^{(i)}} = -(\sum_{j\neq i}\Theta_j^{(-i)} + \sum_{j,k\neq i:\ j<k}\Theta_{j,k}^{(-i)}) + \mathbf{1}_i\mathbf{1}_i^T A^{(i)}(\sum_{j,k\neq i:\ j<k}\Phi_{j,k}^{(-i)}) + A^{(i)}(\sum_{j\neq i}\Phi_j^{(-i)})$$

$$\frac{\partial \mathcal{J}_2}{\partial A^{(i)}} = L^{(i)}A^{(i)}(\circledast_{j\neq i}(A^{(j)T}A^{(j)})) + A^{(i)}(\sum_{j\neq i}\Psi_j^{(-i)}) + D^{(i)}A^{(i)}(\circledast_{j\neq i}(A^{(j)T}D^{(j)}A^{(j)})) - W^{(i)}A^{(i)}(\circledast_{j\neq i}(A^{(j)T}W^{(j)}A^{(j)}))$$

$$\frac{\partial \mathcal{J}_3}{\partial A^{(i)}} = A^{(i)}$$

- Time complexity: $O(\sum_{j,k:\ j<k} K|R_{j,k}| + \sum_j (K^2 I_j + K|W^{(j)}|))$
- Space complexity: $O(\sum_{j,k:\ j<k} |R_{j,k}| + \sum_j (|W^{(j)}| + KI_j))$

$|W^{(j)}|$: # of edges
I_j: # of nodes
K: # of ranks in approximation
$|R_{j,k}|$: # of pairwise relations

[1] Li, Zhuliu, et al. "Learning a Low-Rank Tensor of Pharmacogenomic Multi-relations from Biomedical Networks." *2019 IEEE International Conference on Data Mining (ICDM)*. IEEE, 2019.

GT-COPR: Experiments

- Task: learning disease-gene-chemical relations
  - Fiber-wise evaluation

	Evaluation by gene fibers				Evaluation by disease fibers				Evaluation by chemical fibers			
Methods	AUROC	MAP	Hits@10	Hits@5	AUROC	MAP	Hits@10	Hits@5	AUROC	MAP	Hits@10	Hits@5
GT-COPR (Cartesian)	0.9129	0.1878	0.3440	0.4161	**0.8741**	**0.3006**	0.4337	**0.5630**	0.9749	0.2765	0.3651	**0.4478**
GT-COPR (Tensor)	**0.9132**	**0.1928**	**0.3605**	0.4027	0.8692	0.2842	0.4329	0.5460	**0.9759**	**0.2797**	**0.3827**	0.4399
GT-COPR (Strong)	0.9130	0.1894	0.3468	**0.4192**	0.8697	0.2829	**0.4344**	0.5489	0.9750	0.2759	0.3562	0.4463
SNMF	0.8660	0.1604	0.2827	0.2864	0.7467	0.1463	0.2102	0.2481	0.9236	0.1097	0.1563	0.1877
FASCINATE	0.8978	0.1414	0.2579	0.2522	0.8378	0.2209	0.3310	0.3483	0.9704	0.1489	0.1535	0.1453
wiZAN-Dual	0.7832	0.1287	0.2845	0.2806	0.8678	0.2899	0.4109	0.5495	0.9060	0.1579	0.2651	0.3116
GWNMTF	0.8749	0.0524	0.0185	0.0148	0.7373	0.0886	0.1909	0.1948	0.7076	0.0185	0.0388	0.0519
GWNMF	0.8924	0.0604	0.0753	0.0321	0.7359	0.0597	0.0876	0.0072	0.7241	0.0131	0.0081	0.0028

  - Slice-wise evaluation

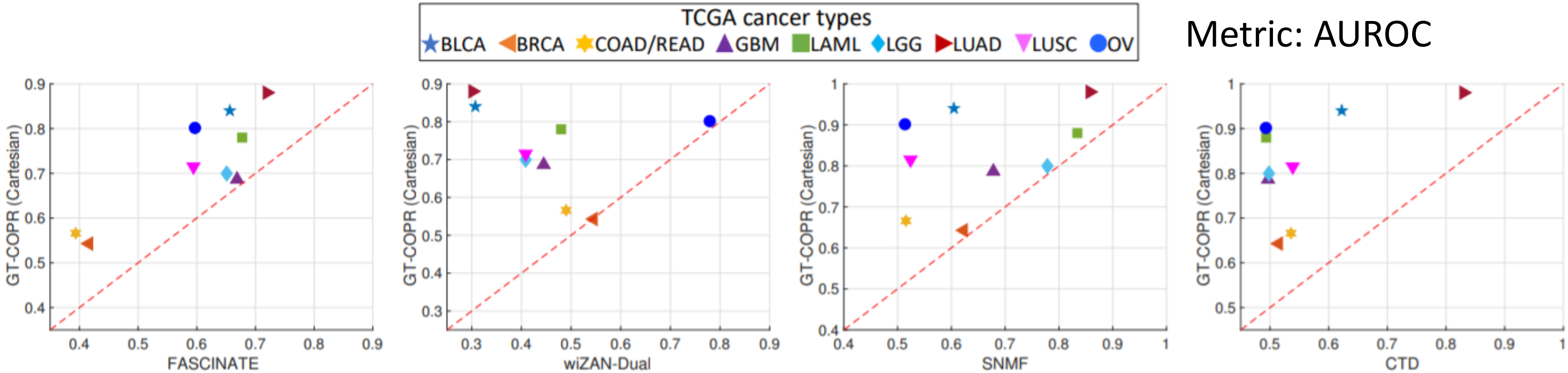
	Evaluation by gene slices				Evaluation by disease slices				Evaluation by chemical slices			
Methods	AUROC	MAP	Hits@10	Hits@5	AUROC	MAP	Hits@10	Hits@5	AUROC	MAP	Hits@10	Hits@5
GT-COPR (Cartesian)	0.9934	0.0755	0.4776	0.6324	0.9843	0.0302	0.0694	0.0710	0.9825	**0.0463**	**0.2123**	**0.1890**
GT-COPR (Tensor)	**0.9945**	0.0687	**0.5223**	**0.6905**	**0.9853**	**0.0385**	0.0935	0.0903	0.9708	0.0337	**0.2123**	**0.1890**
GT-COPR (Strong)	0.9919	**0.0802**	0.4375	**0.6905**	0.9840	0.0303	0.0694	0.0710	**0.9874**	0.0392	**0.2123**	**0.1890**
SNMF	0.9032	0.0159	0.0759	0.0993	0.8568	0.0152	**0.1403**	**0.1387**	0.9181	0.0241	0.1156	0.1240
FASCINATE	0.9861	0.0223	0.0558	0.0182	0.8698	0.0159	0.0710	0.0613	0.9687	0.0155	0.0532	0.0474
wiZAN-Dual	0.9642	0.0369	0.0339	0.0616	0.9198	0.0111	0.0903	0.1000	0.9435	0.0096	0.1146	0.0994
GWNMTF	0.7624	0.0003	0	0	0.6646	0.0003	0.0032	0.0065	0.8645	0.0005	0	0
GWNMF	0.7583	0.0002	0.0002	0.0005	0.7589	0.0004	0	0	0.8550	0.0003	0	0

[1] Li, Zhuliu, et al. "Learning a Low-Rank Tensor of Pharmacogenomic Multi-relations from Biomedical Networks." *2019 IEEE International Conference on Data Mining (ICDM)*. IEEE, 2019.

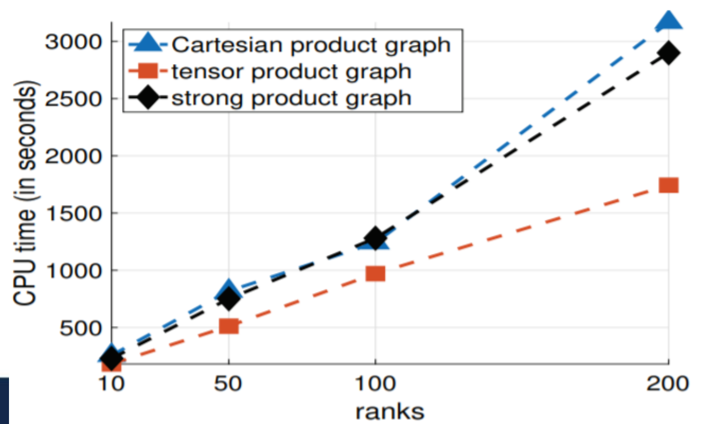
GT-COPR: Experiments

- Prediction of significant cancer-specific pharmacogenomic interactions:

Metric: AUROC



- GT-COPR generally has the best performance
- Similar observations can be found with tensor/strong PG



Overview of Part II

Multi-network Mining Algorithms

Classification
- Label propagation-based multi-view/domain classification
- GNN-based embedding
- Contrastive learning for multi-view

Hyperlink prediction
- NMF-based method
- Autoencoder-based embedding
- GNN-based embedding

Multi-network association
- Label propagation-based method
- w/o attribute
- **w/ attribute**
- Dependency inference
- Network alignment

Ranking
- Consistency based homogeneous
- Consistency based heterogeneous

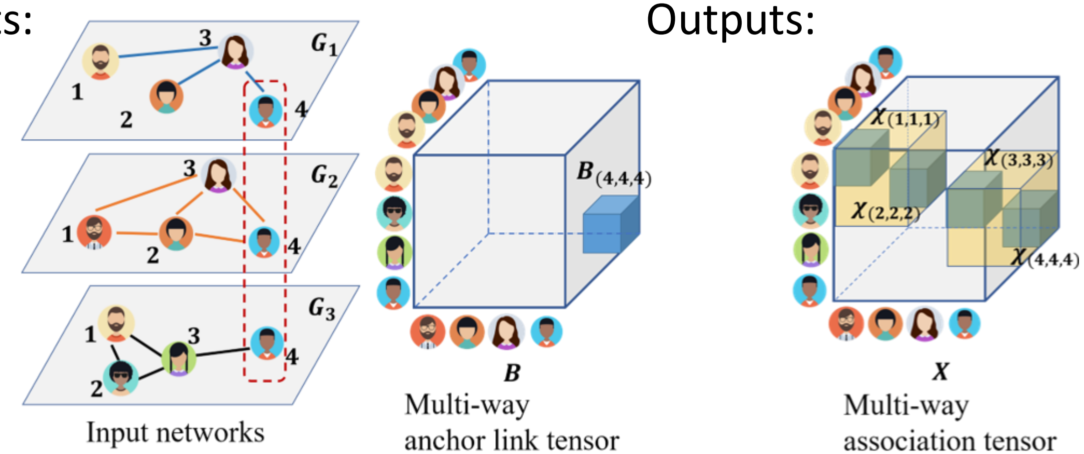
Clustering
- Multi-view network clustering
- NoN clustering

Multi-network embedding
- MF-based embedding
- Random walk-based embedding
- GNN-based embedding

SyTE: Sylvester Tensor Equation for Multi-Way Association

- Given:
  - A set of K networks $\{G_k\ (k = 1, \ldots, K)\}$ (with node number n_k).
  - A multi-way anchor association tensor $\boldsymbol{\mathcal{B}}$.
- Output: Multi-way association tensor $\boldsymbol{\mathcal{X}}$
  - Entries of $\boldsymbol{\mathcal{X}}$: the strength of multi-way association.
  - Multi-way association: collective association of a node set.

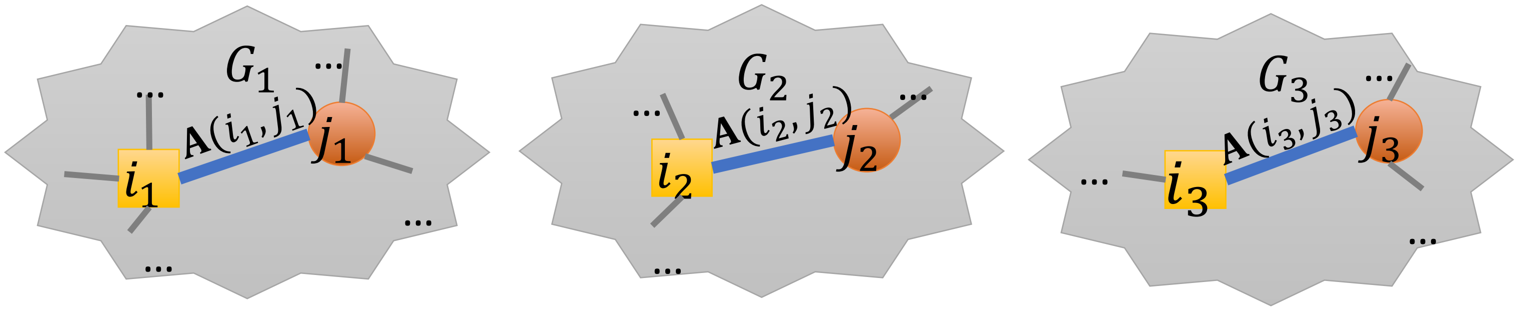
Inputs: Outputs:



[1] Du, Boxin, Lihui Liu, and Hanghang Tong. "Sylvester Tensor Equation for Multi-way Association". SIGKDD (2021)
[2] **More details**: Session Time: 16-Aug 01:00PM-02:30PM SGT (https://virtual.2021.kdd.org/paper_Research_Track-124.html)

SyTE: Intuition

- Intuition: If multi-way association $\boldsymbol{\mathcal{X}}(i_K, \dots, i_1)$ is close to $\boldsymbol{\mathcal{X}}(j_K, \dots, j_1)$:
  - Two sets of nodes are strongly connected
  - Nodes in each of the two sets share the same attribute respectively
  - Nodes from node sets are connected by the same edge attribute



- Large $\boldsymbol{\mathcal{X}}(i_3, i_2, i_1)$ and Large $\boldsymbol{\mathcal{X}}(j_3, j_2, j_1)$ indicate:
  - Large $\mathbf{A}_1(i_1, j_1)$, $\mathbf{A_2}(i_2, j_2)$ and $\mathbf{A_3}(i_3, j_3)$
  - $\{i_1, i_2, i_3\}$, $\{j_1, j_2, j_3\}$: same node attribute
  - $\{i_1, j_1\}$, $\{i_2, j_2\}$, $\{i_3, j_3\}$: same edge attribute

[1] Si Zhang and Hanghang Tong. 2016. Final: Fast attributed network alignment. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM, 1345–1354.
[2] Rohit Singh, Jinbo Xu, and Bonnie Berger. 2008. Global alignment of multiple protein interaction networks with application to functional orthology detection. Proceedings of the National Academy of Sciences 105, 35 (2008), 12763–12768

SyTE: Formulation

- Objective function:

$$J(\mathcal{X}) = \sum_{\substack{i_1,\ldots,i_K \\ j_1,\ldots,j_K}} \Big[\beta \boxed{\left(\frac{\mathcal{X}(i_K,\ldots,i_1)}{\sqrt{d(i_1,\ldots,i_K)}} - \frac{\mathcal{X}(j_K,\ldots,j_1)}{\sqrt{d(j_1,\ldots,j_K)}}\right)^2} * \boxed{t(A_1,\ldots,A_K)} * \boxed{f(i_k)f(j_k)} * \boxed{g(i_k,j_k)} + \boxed{\gamma\big(\mathcal{X}(i_K,\ldots,i_1) - \mathcal{B}(i_K,\ldots,i_1)\big)^2 \Big]}$$

Normalized association smoothness preserver

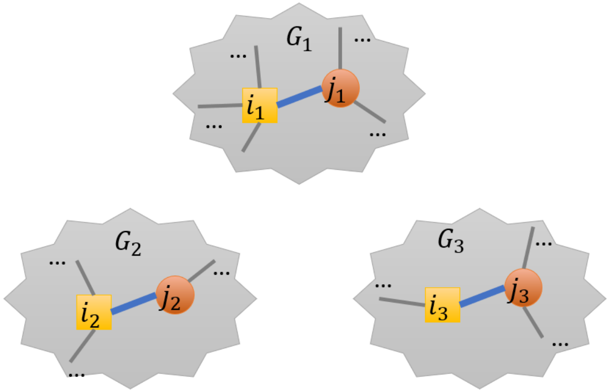
Topology consistency

Edge attribute consistency

Node attribute consistency

Anchor association regularizer

- Details:
  - $t(\mathbf{A}_1,\ldots,\mathbf{A}_K) = \mathbf{A}_1(i_1,j_1)\cdots\mathbf{A}_K(i_K,j_K)$
  - $f(i_k) = \mathbb{1}(\mathbf{N}_1(i_1,i_1) = \cdots = \mathbf{N}_K(i_K,i_K))$
  - $g(i_k,j_k) = \mathbb{1}(\mathbf{E}_1(i_1,j_1) = \cdots = \mathbf{E}_K(i_K,j_K))$
  - $d(i_1,\ldots,i_K) = \sum_{j_1,\ldots,j_K} \mathbf{A}_1(i_1,j_1)\cdots\mathbf{A}_K(i_K,j_K)$
  - β, γ: weighting parameters



[1] Du, Boxin, Lihui Liu, and Hanghang Tong. "Sylvester Tensor Equation for Multi-way Association". SIGKDD (2021)

[2] **More details**: Session Time: 16-Aug 01:00PM-02:30PM SGT (https://virtual.2021.kdd.org/paper_Research_Track-124.html)

SyTE: Sylvester Tensor Equation

- On plain networks:

$$\mathcal{X} - \alpha \mathcal{X} \times_1 \tilde{\mathbf{A}}_K \times_2 \cdots \times_K \tilde{\mathbf{A}}_1 - (1-\alpha)\mathcal{B} = \mathbf{0}$$

  - where $\tilde{\mathbf{A}}_i = \left(\mathbf{D}_i^{-1/2}\right)\mathbf{A}_i\left(\mathbf{D}_i^{-1/2}\right)$. $\dashrightarrow$ Normalization

- Corresponding linear system:
  - $\left(\mathbf{I} - \tilde{\mathbf{A}}_1 \otimes \cdots \otimes \tilde{\mathbf{A}}_K\right)\mathbf{x} = \mathbf{b}$ $\dashrightarrow$ $\mathbf{x} = \mathrm{vec}(\mathcal{X}), \mathbf{b} = \mathrm{vec}(\mathcal{B})$

- On attributed networks:

$$\mathcal{X} - \alpha \sum_{o,p,q} \mathcal{X} \times_1 \tilde{\mathbf{A}}_K^{(o,p,q)} \times_2 \cdots \times_K \tilde{\mathbf{A}}_1^{(o,p,q)} - (1-\alpha)\mathcal{B} = \mathbf{0}$$

  - where $\tilde{\mathbf{A}}_i^{(o,p,q)} = (\mathbf{D}_i^{-\frac{1}{2}}\mathbf{N}_i^p)(\mathbf{E}_i^o \odot \mathbf{A}_i)(\mathbf{D}_i^{-\frac{1}{2}}\mathbf{N}_i^q)$.
  - $\mathbf{N}_i^p$: diagonal node attribute matrix for attribute *p*
  - $\mathbf{E}_i^o$: edge attribute matrix for attribute *o*

- Corresponding linear system:
  - $\left(\mathbf{I} - \sum_{o,p,q} \tilde{\mathbf{A}}_1^{(o,p,q)} \otimes \cdots \otimes \tilde{\mathbf{A}}_K^{(o,p,q)}\right)\mathbf{x} = \mathbf{b}$

[1] Du, Boxin, Lihui Liu, and Hanghang Tong. "Sylvester Tensor Equation for Multi-way Association". SIGKDD (2021)
[2] **More details**: Session Time: 16-Aug 01:00PM-02:30PM SGT (https://virtual.2021.kdd.org/paper_Research_Track-124.html)

SyTE: Key Ideas on Plain Networks

- Decompose the equation into a series of subsystems
  - Utilize the sparsity of $\boldsymbol{\mathcal{B}}$ for decomposition
  - Each subsystem is relatively easier to solve
- Subsystem by a Tensorized Krylov subspace method
  - Tensorized Krylov subspace vs. traditional Krylov subspace: $O(m^K) \rightarrow O(sKlm)$
  - Solve each subsystem by generalized minimal residual method

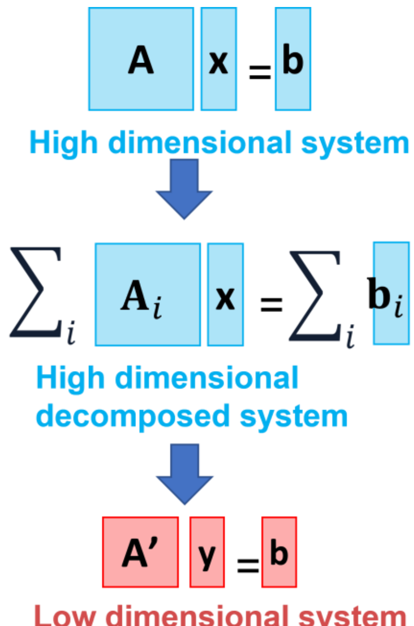
$$(\mathbf{I} - \tilde{\mathbf{A}}_1 \otimes \cdots \otimes \tilde{\mathbf{A}}_K)\mathbf{x} = \boxed{\mathbf{b}}$$

Decompose

$$\sum_i \boxed{(\mathbf{I} - \tilde{\mathbf{A}}_1 \otimes \cdots \otimes \tilde{\mathbf{A}}_K)}\,\mathbf{x}_i = \sum_i \otimes_{j=1}^K \mathbf{b}_j^{(i)}$$

Tensorized Krylov subspace

$$(\otimes_{i=1}^K \mathbf{I}_{l_i+1,l_i} - \otimes_{i=1}^K \tilde{\mathbf{H}}_i)\mathbf{y} = \otimes_{i=1}^K \mathbf{U}_{l_i+1}^T \mathbf{r}_0$$



[1] Du, Boxin, Lihui Liu, and Hanghang Tong. "Sylvester Tensor Equation for Multi-way Association". SIGKDD (2021)
[2] **More details**: Session Time: 16-Aug 01:00PM-02:30PM SGT (https://virtual.2021.kdd.org/paper_Research_Track-124.html)

SyTE: Key Ideas on Attributed Networks

- Decompose the equation by node attributes
  - The solution tensor has a block-diagonal structure
- Diagonal tensors by block coordinate descent (BCD)
  - For diagonal block variables
- Adopt approximation in BCD for faster computation
  - Faster computation

$$\mathcal{X} - \alpha \sum_{o,p,q} \mathcal{X} \times_1 \tilde{\mathbf{A}}_K^{(o,p,q)} \times_2 \cdots \times_K \tilde{\mathbf{A}}_1^{(o,p,q)} = (1-\alpha)\mathcal{B}$$

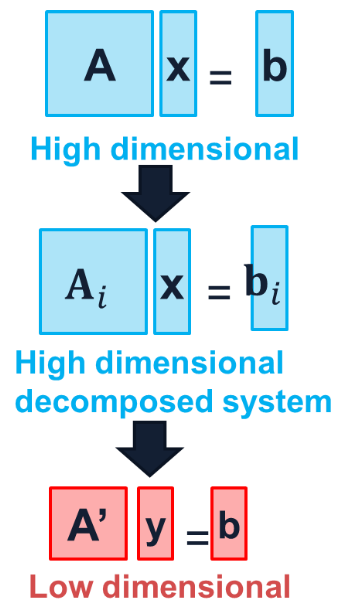
Decompose

$$\mathcal{X}^{i\ldots i} - \sum_{j=1} \mathcal{X}^{j\ldots j} \times_1 \tilde{\mathbf{A}}_1^{ij} \ldots \times_K \tilde{\mathbf{A}}_K^{ij} = \mathcal{B}^{i\ldots i}$$

Approximate

$$\mathcal{X}^{i\ldots i} - \mathcal{X}^{i\ldots i} \times_1 \tilde{\mathbf{A}}_1^{ii} \ldots \times_K \tilde{\mathbf{A}}_K^{ii} = \mathcal{B}^{i\ldots i}_{updated}$$

SyTE-P-Fast

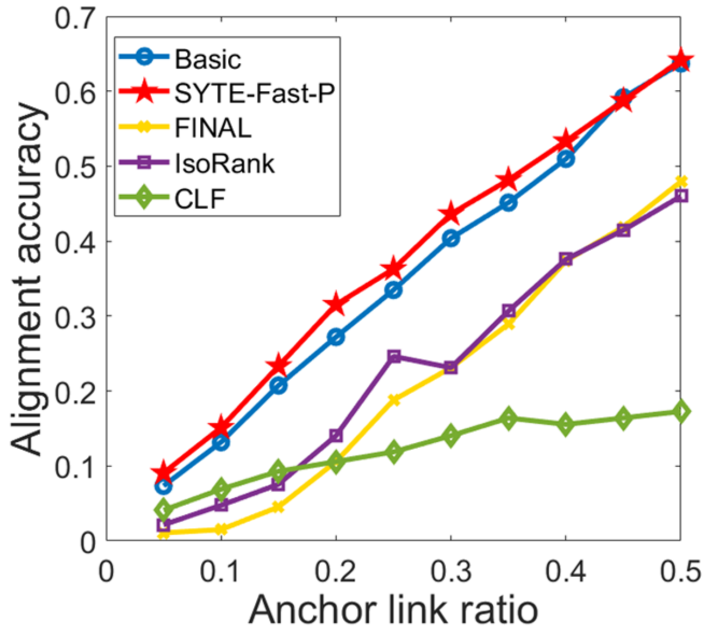


[1] Du, Boxin, Lihui Liu, and Hanghang Tong. "Sylvester Tensor Equation for Multi-way Association". SIGKDD (2021)
[2] **More details**: Session Time: 16-Aug 01:00PM-02:30PM SGT (https://virtual.2021.kdd.org/paper_Research_Track-124.html)

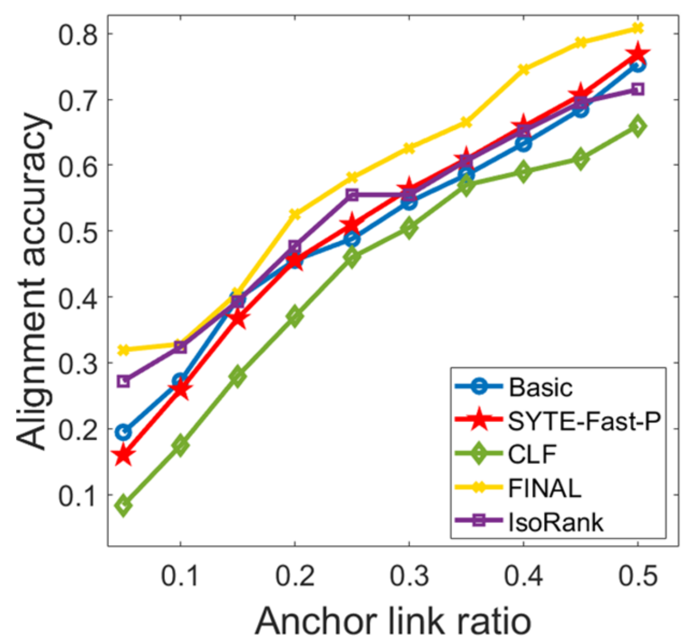
SyTE: Experiments

- Multi-network alignment:
  - High-order metric: successful alignment if all nodes from input networks are aligned correctly.
  - Pairwise metric: successful alignment if any pair of nodes from input networks are aligned correctly.

By high-order metric



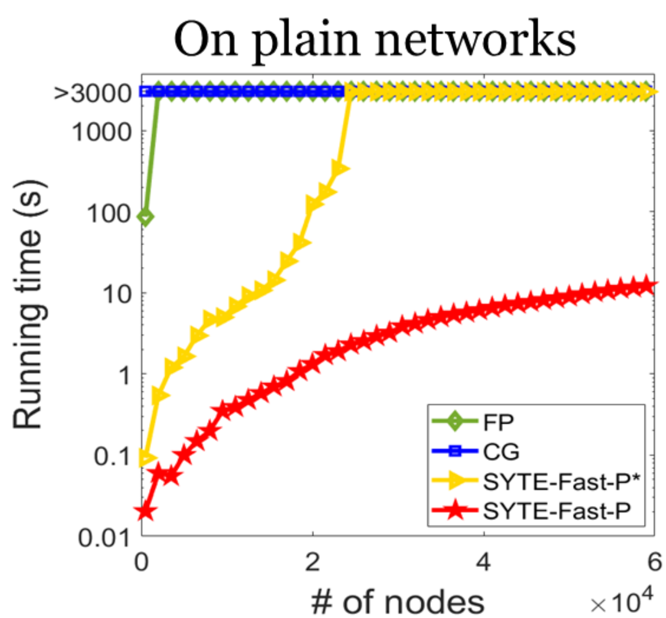
By pairwise metric

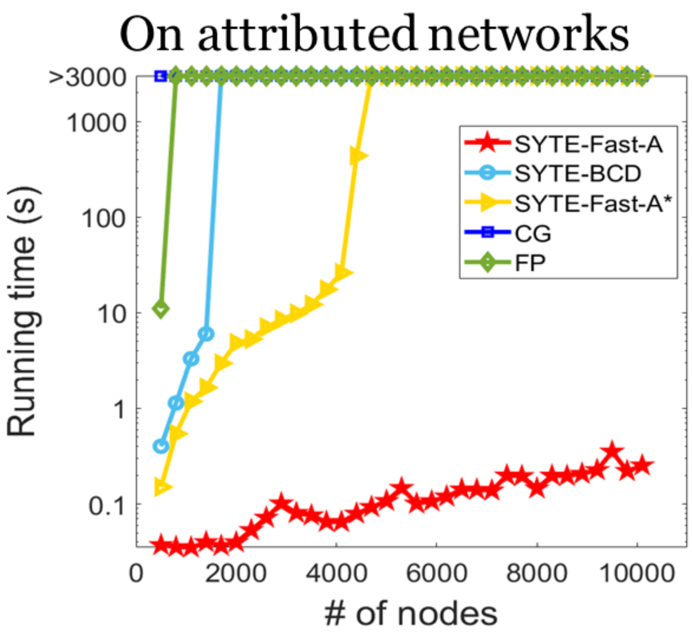


[1] Du, Boxin, Lihui Liu, and Hanghang Tong. "Sylvester Tensor Equation for Multi-way Association". SIGKDD (2021)
[2] **More details**: Session Time: 16-Aug 01:00PM-02:30PM SGT (https://virtual.2021.kdd.org/paper_Research_Track-124.html)

SyTE: Experiments

- Scalability:





- SyTE-Fast-P/A exhibits a linear scalability w.r.t. the # of nodes of the input networks

[1] Du, Boxin, Lihui Liu, and Hanghang Tong. "Sylvester Tensor Equation for Multi-way Association". SIGKDD (2021)
[2] **More details**: Session Time: 16-Aug 01:00PM-02:30PM SGT (https://virtual.2021.kdd.org/paper_Research_Track-124.html)

Overview of Part II

Multi-network Mining Algorithms

Classification
- Label propagation-based multi-view/domain classification
- GNN-based embedding
- Contrastive learning for multi-view

Hyperlink prediction
- NMF-based method
- Autoencoder-based embedding
- GNN-based embedding

Multi-network association
- Label propagation-based method
- w/o attribute
- w/ attribute
- **Dependency inference**
- Network alignment

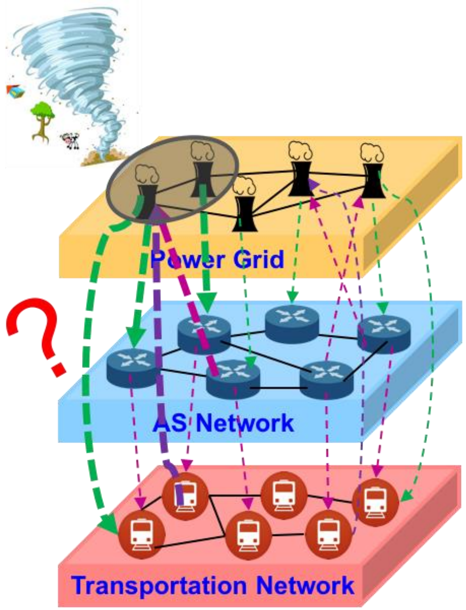
Ranking
- Consistency based homogeneous
- Consistency based heterogeneous

Clustering
- Multi-view network clustering
- NoN clustering

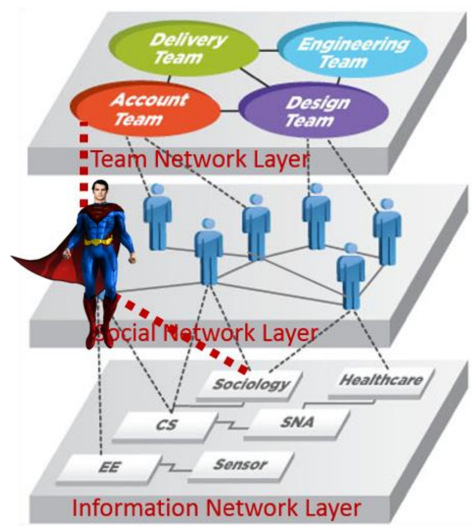
Multi-network embedding
- MF-based embedding
- Random walk-based embedding
- GNN-based embedding

Cross-Layer Dependency Inference

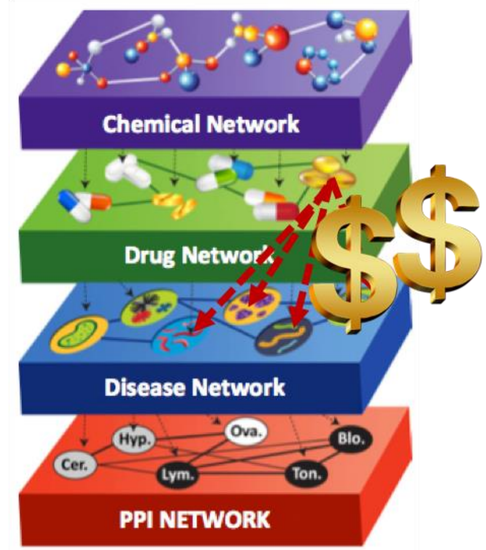
- Obs. 1: Cross-layer dependencies in multi-layered networks are often incomplete



Network Dynamics



Incomplete Record



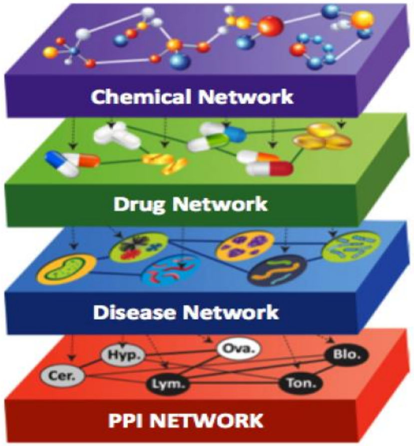
Limited Accessibility

- **Q1:** How to infer the hidden cross-layer dependencies?

[1] Chen, Chen, et al. "FASCINATE: fast cross-layer dependency inference on multi-layered networks." Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. 2016.

Problem Definition

- Given: a multi-layered network
  - Layer-layer dependency matrix $\boldsymbol{G}$
  - Within-layer connectivity matrices $\mathcal{A} = \{\boldsymbol{A}_1, \cdots, \boldsymbol{A}_g\}$
  - Observed cross-layer dependency matrices $\mathcal{D} = \{\boldsymbol{D}_{ij}\}$
- Find: true cross-layer dependency matrices $\{\widetilde{\boldsymbol{D}}_{ij}\}$



- $\boldsymbol{A}_1$ for chemical network, etc.
- $\boldsymbol{G}(1,2) = 1, \boldsymbol{G}(1,3) = 0$;
- $\boldsymbol{D}_{12}$ are represented by solid arrows between $\mathcal{G}_1$ and $\mathcal{G}_2$

[1] Chen, Chen, et al. "FASCINATE: fast cross-layer dependency inference on multi-layered networks." Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. 2016.

Fast Cross-Layer Association Inference (FASCINATE): Formulation

- Key idea: as a collective collaborative filtering problem
  - Within-layer networks as user-user network, item-item similarity network, etc.
  - Cross-layer dependency as user-item ratings
- Optimization problem:

$$\min_{F_i \geq 0 (i=1,\ldots,g)} J = \sum_{i,j:G(i,j)=1} \underbrace{\| \mathbf{W}_{i,j} \odot (\mathbf{D}_{i,j} - \mathbf{F}_i \mathbf{F}_j{'}) \|_F^2}_{\text{Matching observed cross-layer dependencies}} +$$

$$\alpha \sum_i tr \underbrace{(\mathbf{F}_i{'}(\mathbf{T}_i - \mathbf{A}_i)\mathbf{F}_i)}_{\text{Node homophily}} + \beta \sum_i \underbrace{\| \mathbf{F}_i \|_F^2}_{\text{Regularization}}$$

[1] Chen, Chen, et al. "FASCINATE: fast cross-layer dependency inference on multi-layered networks." Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. 2016.

FASCINATE: Optimization Algorithm

- Block coordinate descent method
- For each $\boldsymbol{F}_i$, use multiplicative update method

$$
\begin{aligned}
\frac{\partial J_i}{\partial \mathbf{F}_i} = 2\Big(& \sum_{j:\ \mathbf{G}(i,j)=1} [-(\mathbf{W}_{i,j} \odot \mathbf{W}_{i,j} \odot \mathbf{D}_{i,j})\mathbf{F}_j \\
& + (\mathbf{W}_{i,j} \odot \mathbf{W}_{i,j} \odot (\mathbf{F}_i \mathbf{F}_j'))\mathbf{F}_j] \\
& + \alpha \mathbf{T}_i \mathbf{F}_i - \alpha \mathbf{A}_i \mathbf{F}_i + \beta \mathbf{F}_i \Big)
\end{aligned}
$$

$$
\mathbf{F}_i(u,v) \leftarrow \mathbf{F}_i(u,v) \sqrt{\frac{\mathbf{X}(u,v)}{\mathbf{Y}(u,v)}}
$$

where

$$
\mathbf{X} = \sum_{j:\ \mathbf{G}(i,j)=1} (\mathbf{W}_{i,j} \odot \mathbf{W}_{i,j} \odot \mathbf{D}_{i,j})\mathbf{F}_j + \alpha \mathbf{A}_i \mathbf{F}_i
$$

$$
\mathbf{Y} = \sum_{j:\ \mathbf{G}(i,j)=1} (\mathbf{W}_{i,j} \odot \mathbf{W}_{i,j} \odot (\mathbf{F}_i \mathbf{F}_j'))\mathbf{F}_j + \alpha \mathbf{T}_i \mathbf{F}_i + \beta \mathbf{F}_i
$$

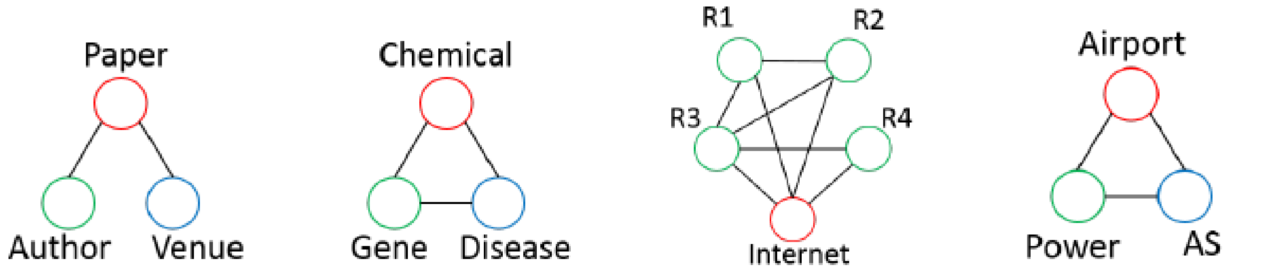
[1] Chen, Chen, et al. "FASCINATE: fast cross-layer dependency inference on multi-layered networks." Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. 2016.

FASCINATE: Experimental Setups

- Datasets:

Dataset	# of Layers	# of Nodes	# of Links	# of CrossLinks
SOCIAL	3	125,344	214,181	188,844
BIO	3	35,631	253,827	75,456
INFRA-5	5	349	379	565
INFRA-3	3	15,126	29,861	28,023,500

- Abstract dependency structure



(a) SOCIAL (b) BIO (c) INFRA-5 (d) INFRA-3

[1] Chen, Chen, et al. "FASCINATE: fast cross-layer dependency inference on multi-layered networks." Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. 2016.

FASCINATE: Experimental Results

- Effectiveness of dependency inference on BIO dataset

Methods	MAP	R-MPR	HLU	AUC	Prec@10
FASCINATE	0.0660	**0.2651**	**8.4556**	**0.7529**	**0.0118**
FASCINATE-CLUST	**0.0667**	0.2462	8.2160	0.7351	0.0108
MulCol	0.0465	0.2450	6.0024	0.7336	0.0087
PairSid	0.0308	0.1729	3.8950	0.6520	0.0062
PairCol	0.0303	0.1586	3.7857	0.6406	0.0056
PairNMF	0.0053	0.0290	0.5541	0.4998	0.0007
PairRec	0.0056	0.0435	0.5775	0.5179	0.0007
FlatNMF	0.0050	0.0125	0.4807	0.5007	0.0007
FlatRec	0.0063	0.1009	0.6276	0.5829	0.0009

- Fascinate outperforms all baselines

[1] Chen, Chen, et al. "FASCINATE: fast cross-layer dependency inference on multi-layered networks." Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. 2016.

FASCINATE – Experimental Results

- Effectiveness of dependency inference on INFRA-5

Methods	MAP	R-MPR	HLU	AUC	Prec@10
FASCINATE	**0.5040**	**0.3777**	**67.2231**	**0.8916**	**0.2500**
FASCINATE-CLUST	0.4297	0.3220	56.8215	0.8159	0.2340
MulCol	0.4523	0.3239	59.8115	0.8329	0.2413
PairSid	0.3948	0.2392	49.5484	0.7413	0.2225
PairCol	0.3682	0.2489	48.5966	0.7406	0.2309
PairNMF	0.1315	0.0464	15.7148	0.5385	0.0711
PairRec	0.0970	0.0099	9.4853	0.5184	0.0399
FlatNMF	0.3212	0.2697	44.4654	0.7622	0.1999
FlatRec	0.1020	0.0778	11.5598	0.5740	0.0488

- Fascinate outperforms all baselines

[1] Chen, Chen, et al. "FASCINATE: fast cross-layer dependency inference on multi-layered networks." Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. 2016.

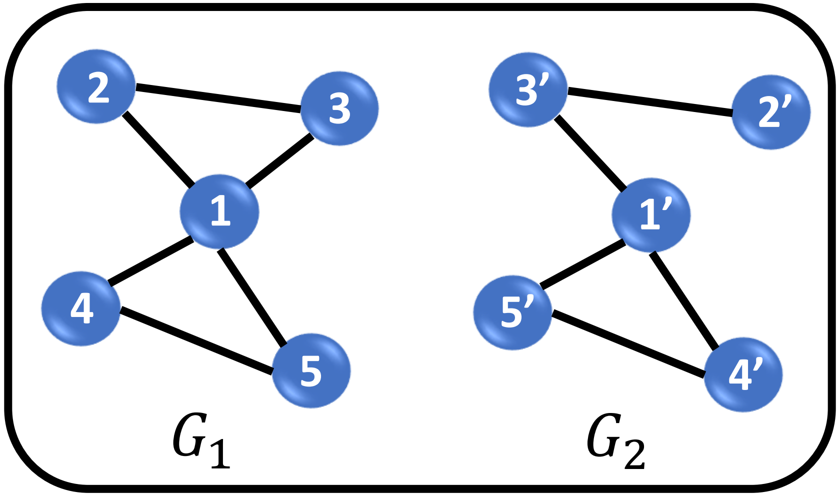
Overview of Part II

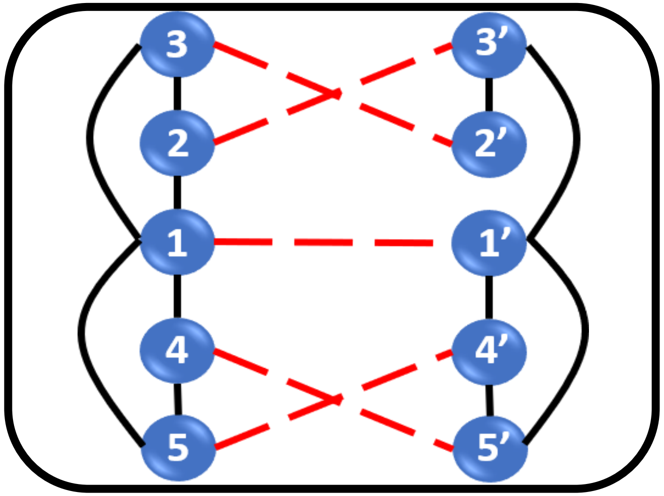
Multi-network Mining Algorithms

Classification	Hyperlink prediction	Multi-network association	Ranking	Clustering	Multi-network embedding
• Label propagation-based multi-view/domain classification • GNN-based embedding • Contrastive learning for multi-view	• NMF-based method • Autoencoder-based embedding • GNN-based embedding	• Label propagation-based method • w/o attribute • w/ attribute • Dependency inference • **Network alignment**	• Consistency based homogeneous • Consistency based heterogeneous	• Multi-view network clustering • NoN clustering	• MF-based embedding • Random walk-based embedding • GNN-based embedding

Network Alignment: Problem Definition

- Given:
  - a set of networks $\{G_l\}$ $(l \geq 2)$ where $G_l = \{V_l, E_l, \mathbf{A}_l\}$;
    - $V_l, E_l, \mathbf{A}_l$ are the nodes, edges and adjacency matrix of G_l;
  - prior alignment matrices $\{\boldsymbol{H}_{l_1,l_2}\}$ between G_{l_1} and G_{l_2}.
- Find: the alignment matrices $\{\boldsymbol{S}_{l_1,l_2}\}$ between G_{l_1} and G_{l_2}.



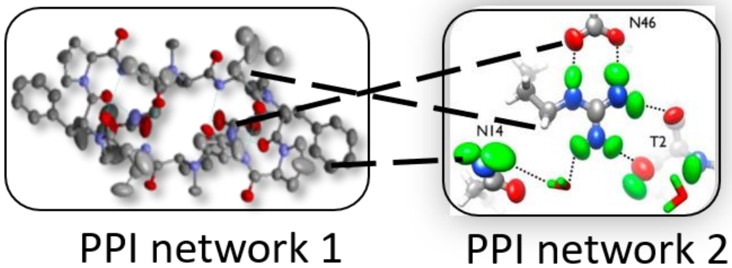


[1] Si Zhang and Hanghang Tong. 2020. Network Alignment: Recent Advances and Future Directions. Proceedings of the 29th ACM International Conference on Information & Knowledge Management. Association for Computing Machinery, New York, NY, USA.

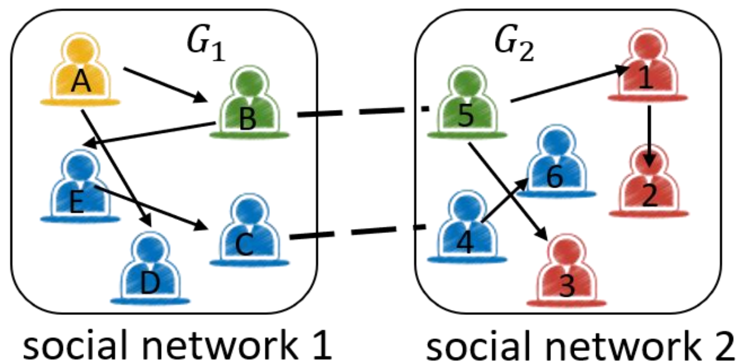
Why Do We Care?

Identify Species-Specific Pathways

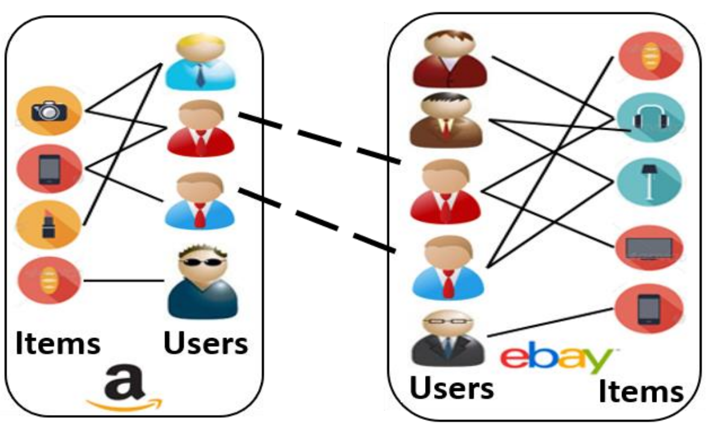
Protein-Protein Interaction (PPI) networks



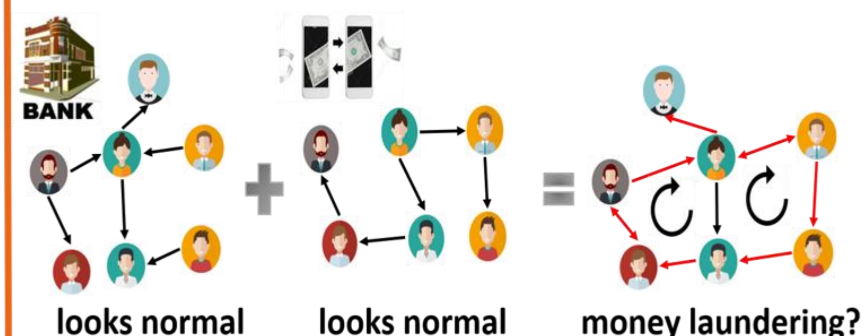
Cross Network Information Diffusion



Cross-Site Recommendation



Fraud Detection



[1] Si Zhang and Hanghang Tong. 2020. Network Alignment: Recent Advances and Future Directions. Proceedings of the 29th ACM International Conference on Information & Knowledge Management. Association for Computing Machinery, New York, NY, USA.

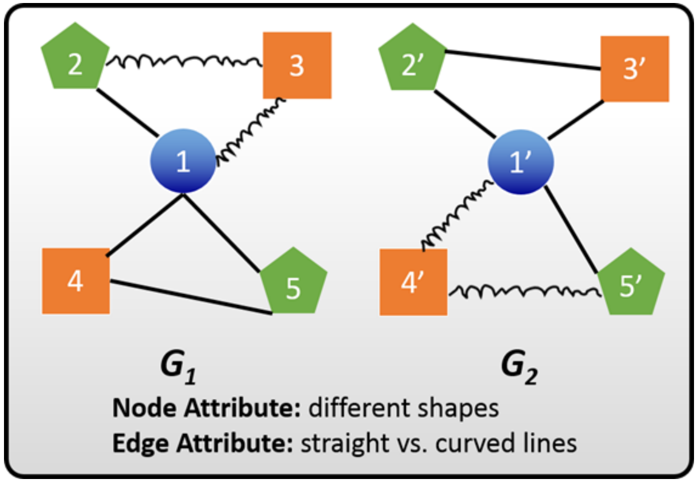
Key Challenge #1: Complexity

- Time complexity:
  - Most of existing works have an at least $O(n^2)$ time complexity
  - Inefficient computations for large-scale networks
- Space complexity:
  - At least $O(n^2)$ to store the alignment matrix
  - Costly memory consumptions
- **Q:** How to efficiently solve network alignment?

[1] Si Zhang and Hanghang Tong. 2020. Network Alignment: Recent Advances and Future Directions. Proceedings of the 29th ACM International Conference on Information & Knowledge Management. Association for Computing Machinery, New York, NY, USA.

Key Challenge #2: Variety

- Networks have rich contextual information
  - Node attributes, e.g., gender, age, etc.
  - Edge attributes, e.g., relation types, etc.



- **Q:** How to encode contextual information to enhance the alignment performance?

[1] Si Zhang and Hanghang Tong. 2020. Network Alignment: Recent Advances and Future Directions. Proceedings of the 29th ACM International Conference on Information & Knowledge Management. Association for Computing Machinery, New York, NY, USA.

Key Challenge #3: Disparity

- Networks appear in various sources
  - Networks may capture distinct information
    - Facebook: to connect friend, family, etc.
    - LinkedIn: to connect professionals
  - Same nodes have different behavior patterns
    - E.g., a user is very active in Facebook but quiet in Twitter

- **Q:** How to handle the disparity behind multi-sourced networks?

[1] Si Zhang and Hanghang Tong. 2020. Network Alignment: Recent Advances and Future Directions. Proceedings of the 29th ACM International Conference on Information & Knowledge Management. Association for Computing Machinery, New York, NY, USA.

Overview of Network Alignment Methods

Recent Network Alignment (NA) Algorithms

Pairwise NA

- Consistency-based
  - w/o attributes
  - w/ attributes
- Embedding-based
  - w/o attributes
  - w/ attributes
- Optimal transport-based
  - w/o attributes

Collective NA

- Consistency-based
  - w/o attributes
  - w/ attributes
- Embedding-based
  - w/o attributes

Higher-Order NA

- Consistency-based
  - Single-level
  - Multilevel

Related Tasks

- Entity alignment
  - Non-GNN based
  - GNN-based
- Cross-layer inference
- Cross-network transformation

[1] Si Zhang and Hanghang Tong. 2020. Network Alignment: Recent Advances and Future Directions. Proceedings of the 29th ACM International Conference on Information & Knowledge Management. Association for Computing Machinery, New York, NY, USA.

Coffee Break Time

 *We will resume the tutorial 15 minutes later.*

Overview of Part II

Multi-network Mining Algorithms

Classification
- Label propagation-based multi-view/domain classification
- GNN-based embedding
- Contrastive learning for multi-view

Hyperlink prediction
- NMF-based method
- Autoencoder -based embedding
- GNN-based embedding

Multi-network association
- Label propagation-based method
- w/o attribute
- w/ attribute
- Dependency inference
- Network alignment

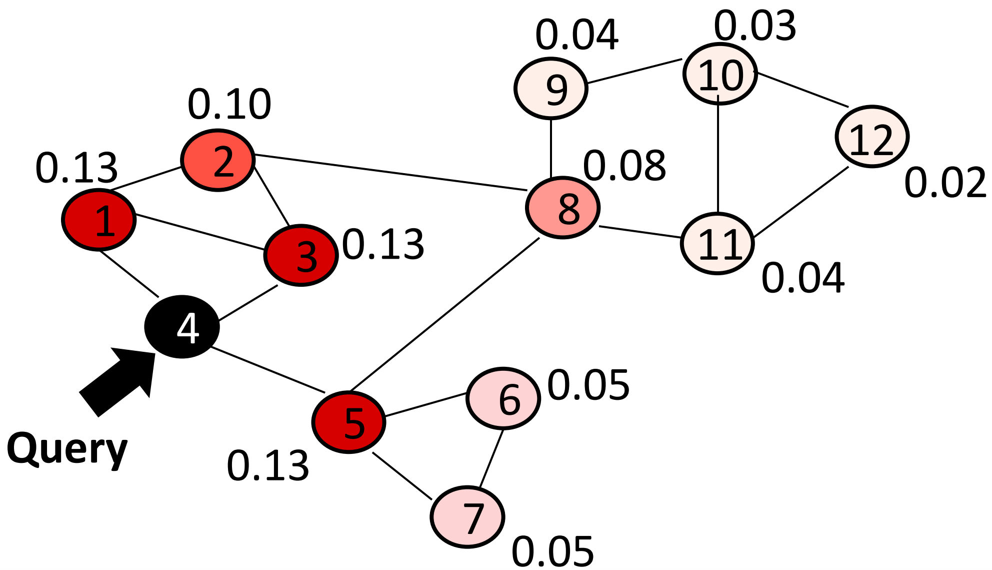
Ranking
- **Consistency based homogeneous**
- Consistency based heterogeneous

Clustering
- Multi-view network clustering
- NoN clustering

Multi-network embedding
- MF-based embedding
- Random walk-based embedding
- GNN-based embedding

Ranking on Single Network: Intuition

- Assumption: Homophily (guilt-by-association)
- Example: Two researchers are close if they
  - Share many common co-authors
  - Work on similar topics
  - Publish at same venue(s).



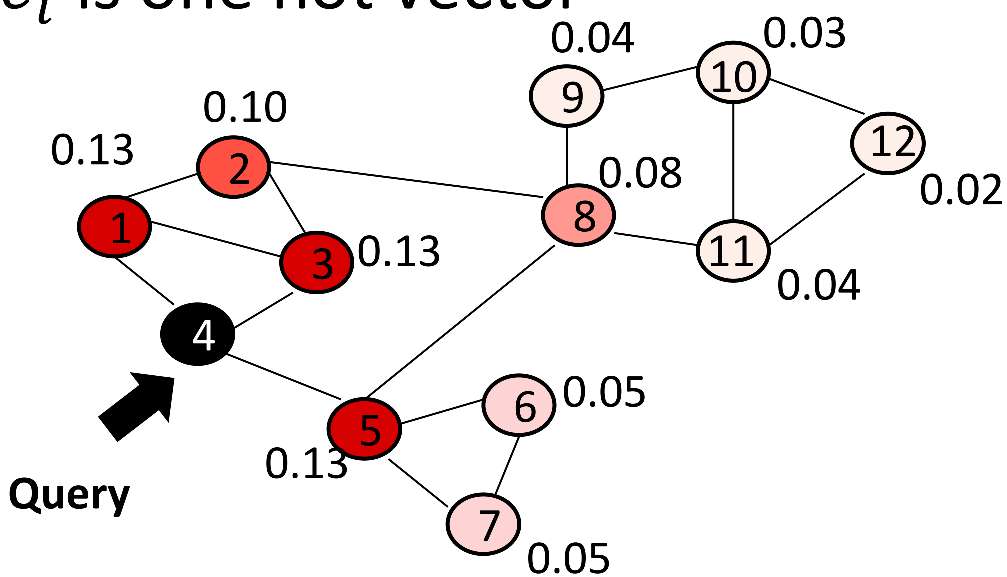
	Node 4
Node 1	0.13
Node 2	0.10
Node 3	0.13
Node 4	0.22
Node 5	0.13
Node 6	0.05
Node 7	0.05
Node 8	0.08
Node 9	0.04
Node 10	0.03
Node 11	0.04
Node 12	0.02

Ranking vector

[1] H. Tong, C. Faloutsos, J.-Y. Pan: Fast Random Walk with Restart and Its Applications. ICDM 2006
[2] Ni, Jingchao, et al. "Inside the atoms: ranking on a network of networks." Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining. 2014.

Ranking on Single Network: Formulation

- Nearby nodes, higher scores
- $r_i = c\mathbf{A}r_i + (1-c)e_i$
- e_i is one hot vector



More red, more relevant

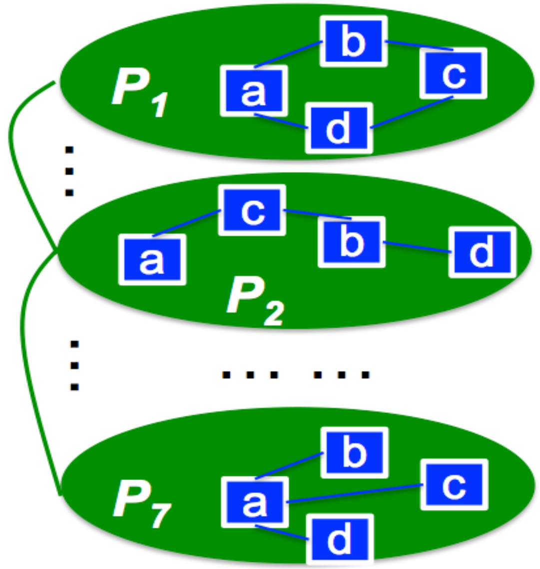
	Node 4
Node 1	0.13
Node 2	0.10
Node 3	0.13
Node 4	0.22
Node 5	0.13
Node 6	0.05
Node 7	0.05
Node 8	0.08
Node 9	0.04
Node 10	0.03
Node 11	0.04
Node 12	0.02

Ranking vector

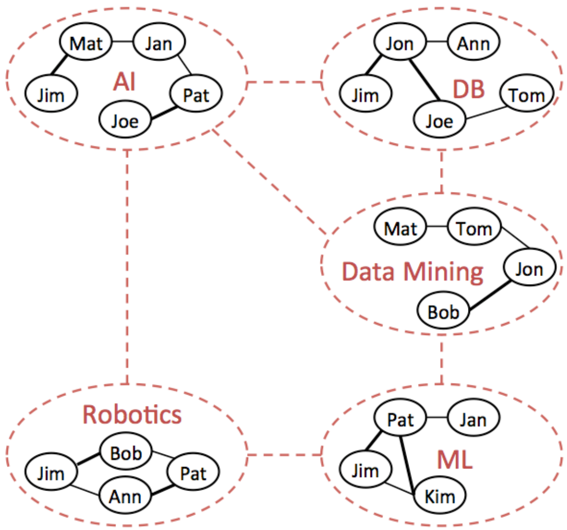
[1] H. Tong, C. Faloutsos, J.-Y. Pan: Fast Random Walk with Restart and Its Applications. ICDM 2006
[2] Ni, Jingchao, et al. "Inside the atoms: ranking on a network of networks." Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining. 2014.

CrossRank: Motivation

- A1: Given a disease (e.g., P_1), what are the most relevant proteins (blue nodes)?
- A2: Who is most influential considering both the within and cross-area influence?



Tissue-Specific PPI Networks

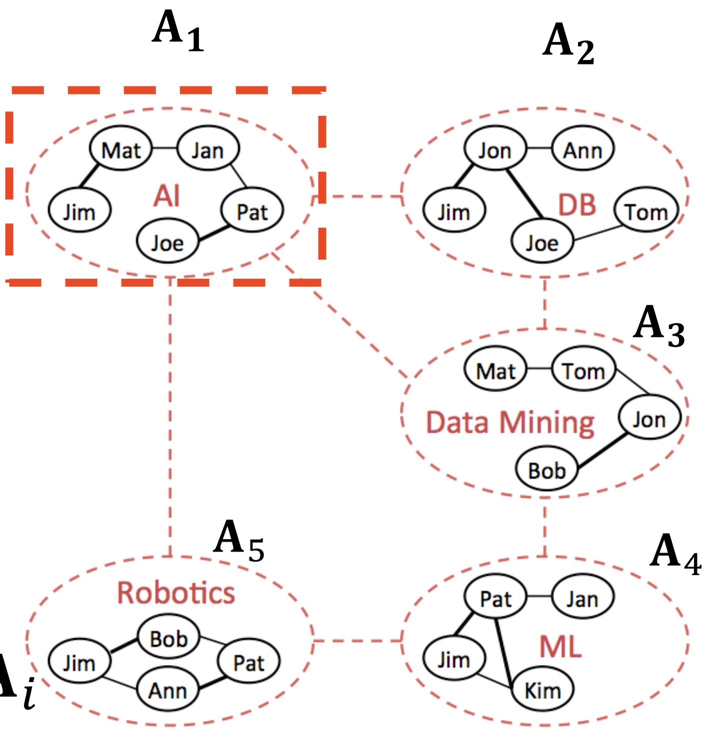


Collaboration Networks

[1] Ni, Jingchao, et al. "Inside the atoms: ranking on a network of networks." Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining. 2014.

CrossRank: Problem Definition

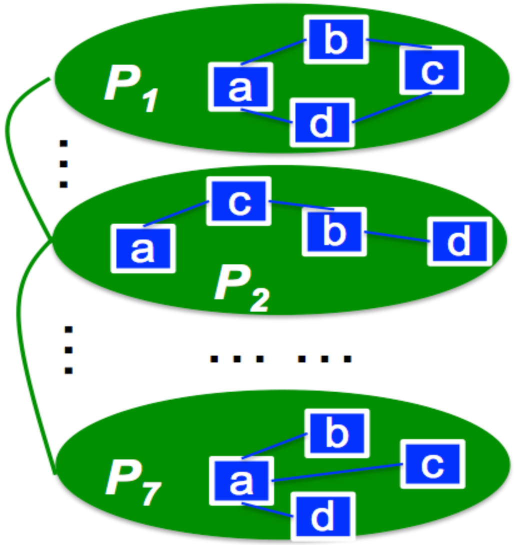
- Given:
  - A NoN
  - the query vectors $\mathbf{e}_i$ (i = 1, …, g)
  - Example: $\mathbf{e}_1 = (0, 1, 0, 0, 1)$
  - $\mathbf{e}_1$ refers to Mat and Joe in $\mathbf{A_1}$



- Find:
  - ranking vectors $\mathbf{r}_i$ for the nodes in $\mathbf{A}_i$

[1] Ni, Jingchao, et al. "Inside the atoms: ranking on a network of networks." Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining. 2014.

CrossRank: Intuition



- Within-network smoothness
  - Similar rankings for close nodes
  - Example: PageRank
- Query preference
  - High scores for the queried nodes
- Cross-network consistency
  - Similar ranking scores for an overlapped domain node if the domains this node belongs to are similar with each other.
  - Example: If
    - a protein is highly relevant to disease-i.
    - disease-i is very similar to disease-j.
    - then it is likely that the same protein is also highly relevant to disease-j.

[1] Ni, Jingchao, et al. "Inside the atoms: ranking on a network of networks." Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining. 2014.

CrossRank: Formulation

$$J(\mathbf{r}_1, \ldots, \mathbf{r}_g) = \underbrace{c \sum_{i=1}^{g} \mathbf{r}_i'(\mathbf{I}_{n_i} - \widetilde{\mathbf{A}}_i)\mathbf{r}_i}_{\text{within-network smoothness}} + \underbrace{(1-c)\sum_{i=1}^{g} \|\mathbf{r}_i - \mathbf{e}_i\|^2}_{\text{query preference}} + \boxed{\underbrace{a \sum_{i,j=1}^{g} \left\| \frac{\mathbf{r}_i(I_{ij})}{\sqrt{d_m(i)}} - \frac{\mathbf{r}_j(I_{ij})}{\sqrt{d_m(j)}} \right\|^2 \mathbf{G}(i,j)}_{\text{cross-network consistency}}}$$

- $\mathbf{r}_i$ is the ranking vector of the domain-specific network $\mathbf{A}_i$.
- $d_m(i)$ is the degree of main node i in the main network $\mathbf{G}$.
- I_{ij} is the set of common nodes between $\mathbf{A}_i$ and $\mathbf{A}_j$.
- $\boldsymbol{G}(i,j)$ is the similarity between $\mathbf{A}_i$ and $\mathbf{A}_j$.

Similar ranking scores for an overlapped domain node if the two domains it belongs to are similar.

[1] Ni, Jingchao, et al. "Inside the atoms: ranking on a network of networks." Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining. 2014.

CrossRank: Optimization

- Matrix of objective function:

$$J(\mathbf{r}) = c\mathbf{r}'(\mathbf{I_n} - \widetilde{\mathbf{A}})\mathbf{r} + (1-c)\|\mathbf{r}-\mathbf{e}\|^2 + 2a\mathbf{r}'\mathbf{X}\mathbf{r}$$

$$\widetilde{\mathbf{A}} = \begin{bmatrix} \widetilde{\mathbf{A}}_1 & \cdots & \mathbf{0} \\ \vdots & \ddots & \vdots \\ \mathbf{0} & \cdots & \widetilde{\mathbf{A}}_g \end{bmatrix} \qquad \mathbf{r} = \begin{bmatrix} \mathbf{r}_1 \\ \vdots \\ \mathbf{r}_g \end{bmatrix} \qquad \mathbf{e} = \begin{bmatrix} \mathbf{e}_1 \\ \vdots \\ \mathbf{e}_g \end{bmatrix}$$

X encodes the cross-network consistency

RWR-like update rule

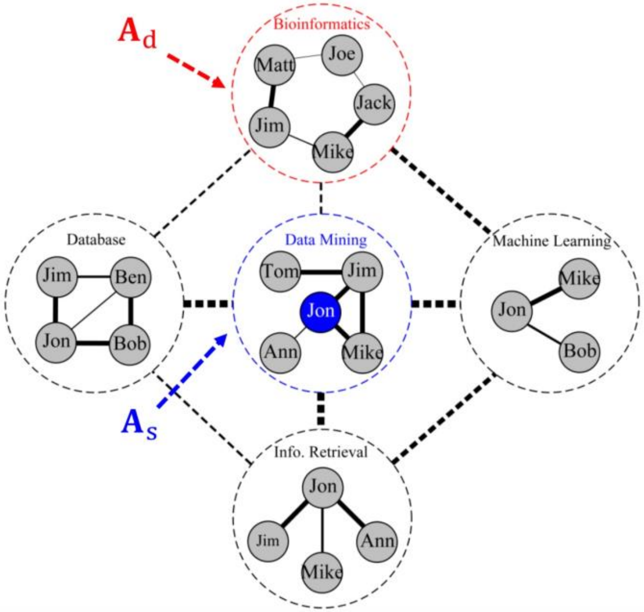
$$\mathbf{r} = \Big(\underbrace{\frac{c}{1+2a}}_{\text{within-network walk}}\widetilde{\mathbf{A}} + \underbrace{\frac{2a}{1+2a}}_{\text{cross-network walk}}\widetilde{\mathbf{Y}}\Big)\mathbf{r} + \frac{1-c}{1+2a}\mathbf{e}$$

X is also the Laplacian matrix of **Y**.

[1] Ni, Jingchao, et al. "Inside the atoms: ranking on a network of networks." Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining. 2014.

CrossQuery: Problem Definition

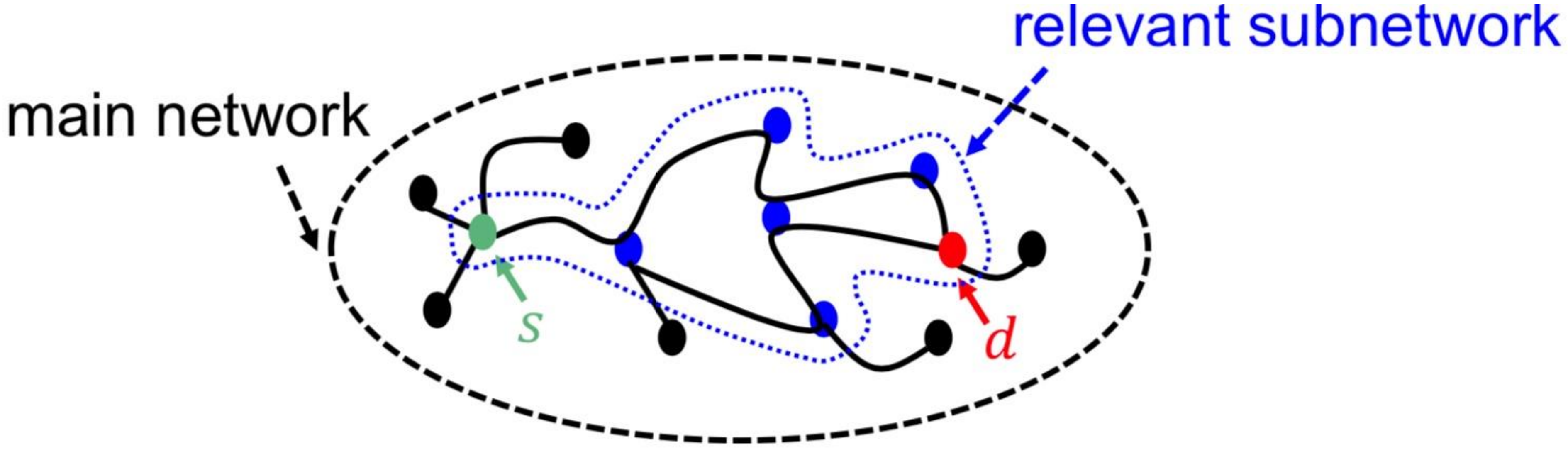
- Given:
  - a NoN
  - a query node from a source domain-specific network $\mathbf{A}_\mathrm{s}$
  - a target domain-specific network $\mathbf{A}_\mathrm{d}$
  - an integer *k*
- Find:
  - the top-*k* most relevant nodes from $\mathbf{A}_\mathrm{d}$



[1] Ni, Jingchao, et al. "Inside the atoms: ranking on a network of networks." Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining. 2014.

CrossQuery: Method

- CrossQuery-Basic:
  - Restricting the candidate nodes in the target domain-specific network.
- CrossQuery-Fast:
  - Prune less relevant main nodes in the main network.



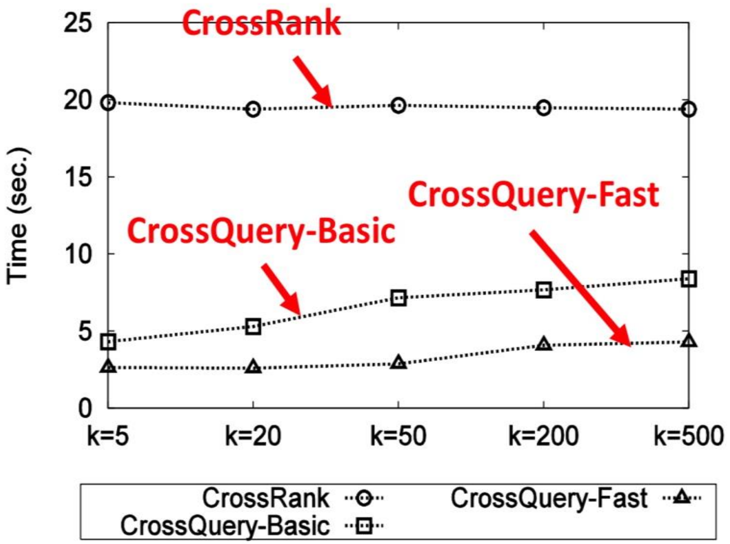
[1] Ni, Jingchao, et al. "Inside the atoms: ranking on a network of networks." Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining. 2014.

CrossQuery: Experimental Results

- Collaboration Prediction on DBLP NoN

#Papers	Hops	#Pairs	Methods	AUC	Accuracy
≥ 3	$[3, 4]$	45	PC	0.7196	0.4444
			Katz	0.7439	0.5556
			PropFlow	0.7558	0.6222
			PathSim	0.5636	0.2444
			PageRank	0.7417	0.5333
			CrossQuery	**0.7685**	**0.6444**
≥ 3	$[3, 6]$	70	PC	0.6009	0.3000
			Katz	0.6243	0.3714
			PropFlow	0.6268	0.4429
			PathSim	0.5278	0.2143
			PageRank	0.6378	0.3714
			CrossQuery	**0.6632**	**0.4571**
≥ 5	$[3, 4]$	23	PC	0.6521	0.2609
			Katz	0.6717	**0.3478**
			PropFlow	0.6850	**0.3478**
			PathSim	0.4279	0.1304
			PageRank	0.6743	**0.3478**
			CrossQuery	**0.7099**	**0.3478**
≥ 5	$[3, 6]$	38	PC	0.5692	0.2105
			Katz	0.5786	0.2368
			PropFlow	0.5950	**0.2895**
			PathSim	0.4362	0.1053
			PageRank	0.5880	0.2368
			CrossQuery	**0.6308**	**0.2895**

Area	Conference included
DM	KDD, ICDM, SDM, PKDD, PAKDD
ML	ICML, NIPS, AAAI, IJCAI, UAI, ECML
DB	VLDB, SIGMOD, ICDE, ICDT, EDBT, PODS
IR	SIGIR, WWW, ACL, ECIR, CIKM
BIO	ISMB, RECOMB, ECCB, BIBE, BIBM, WABI



- Observation: CrossQuery is effective and fast for DBLP NoN.

[1] Ni, Jingchao, et al. "Inside the atoms: ranking on a network of networks." Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining. 2014.

Overview of Part II

Multi-network Mining Algorithms

Classification
- Label propagation-based multi-view/domain classification
- GNN-based embedding
- Contrastive learning for multi-view

Hyperlink prediction
- NMF-based method
- Autoencoder -based embedding
- GNN-based embedding

Multi-network association
- Label propagation-based method
- w/o attribute
- w/ attribute
- Dependency inference
- Network alignment

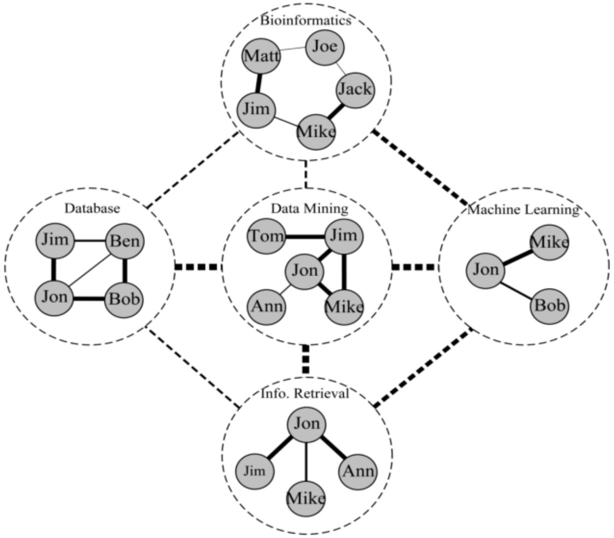
Ranking
- Consistency based homogeneous
- **Consistency based heterogeneous**

Clustering
- Multi-view network clustering
- NoN clustering

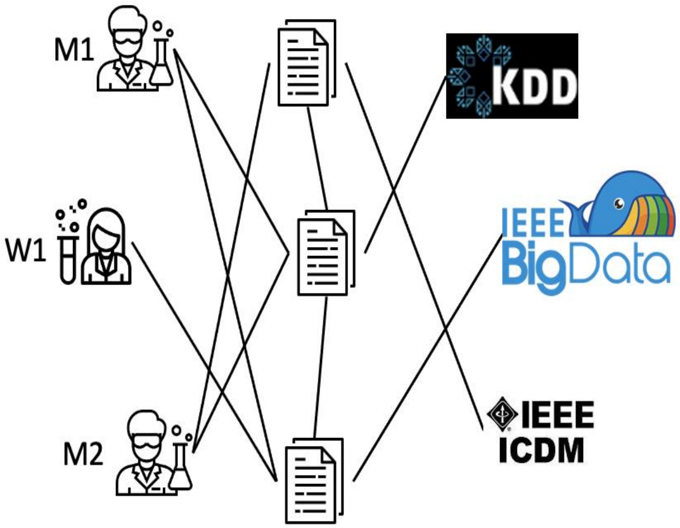
Multi-network embedding
- MF-based embedding
- Random walk-based embedding
- GNN-based embedding

Network of Heterogeneous Information Networks (NeoHIN): Intuition

- Strong representation power of HIN
- Domain-specific and cross-domain ranking in NoN
- Enjoy the best of both kinds of models



NoN: area network of co-author networks



Scholar HIN

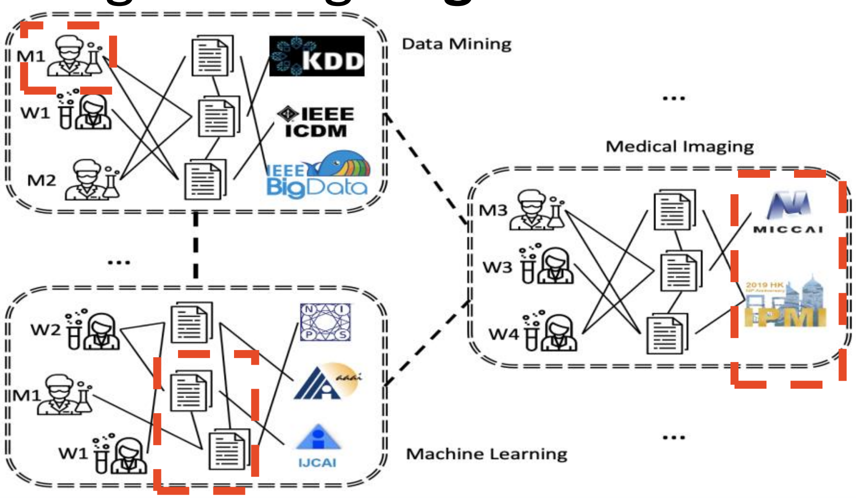
[2] Z. Xu, S. Zhang, Y. Xia, L. Xiong and H. Tong, "Ranking on Network of Heterogeneous Information Networks," 2020 IEEE International Conference on Big Data (Big Data), 2020, pp. 848-857, doi: 10.1109/BigData50022.2020.9378121.

NeoHIN: Problem Definition

- Given:
  - A network of heterogeneous network (NeoHIN).
  - A set of target nodes for ranking.
  - A set of query nodes of interests (optional).
  - A set of meta-path of interests (optional).
- Find: the rankings among **target nodes** w.r.t. **query node(s)**

query node example 1

query nodes example 2



target nodes

[1] Z. Xu, S. Zhang, Y. Xia, L. Xiong and H. Tong, "Ranking on Network of Heterogeneous Information Networks," 2020 IEEE International Conference on Big Data (Big Data), 2020, pp. 848-857, doi: 10.1109/BigData50022.2020.9378121.

HITS-NeoHIN: Formulation

- Integrate the cross-domain consistency into HITS.

$$\min J_i(\mathbf{u}_i, \mathbf{v}_i) = \frac{c}{2}||\mathbf{A}_i - \mathbf{u}_i\mathbf{v}_i'||_F^2 + (1-c)\left(||\mathbf{u}_i - \mathbf{e}_{ui}||_F^2 + \left||\mathbf{v}_i - \mathbf{e}_{vi}||_F^2\right)\right),$$
$$s.t.\ \forall x, \mathbf{u}_i(\mathrm{x}) \geq 0, \mathbf{v}_i(\mathrm{x}) \geq 0$$

Objective of Hits from the i-th domain

Query preference

- HITS-NeoHIN:

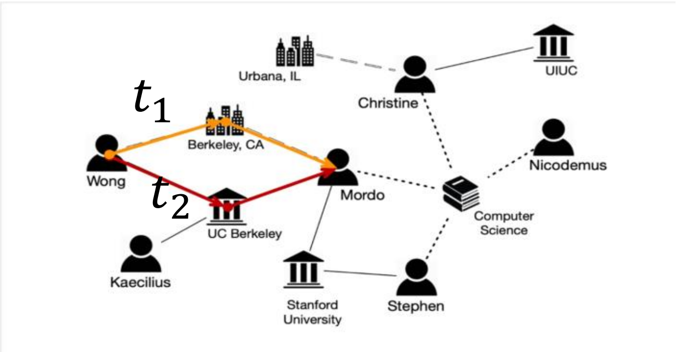
For two similar domains *i*, *j*, similar ranking scores for an overlapped node if the two domains it belongs to are similar

$$\min J(\mathbf{u}, \mathbf{v}) = \sum_{i=1}^{g} J_i(\mathbf{u}_i, \mathbf{v}_i) +$$
$$\underbrace{a\sum_{i=1}^{g}\sum_{i=1}^{g}\left|\left|\frac{\mathbf{u}_i(I_{ij})}{\sqrt{\mathbf{d}(i)}} - \frac{\mathbf{u}_j(I_{ij})}{\sqrt{\mathbf{d}(j)}}\right|\right|_2^2 \mathbf{G}(i,j) + a\sum_{i=1}^{g}\sum_{i=1}^{g}\left|\left|\frac{\mathbf{v}_i(I_{ij})}{\sqrt{\mathbf{d}(i)}} - \frac{\mathbf{v}_j(I_{ij})}{\sqrt{\mathbf{d}(j)}}\right|\right|_2^2 \mathbf{G}(i,j)}_{\text{Cross-domain consistency}},$$

[1] Kleinberg, J. M. (1999). Authoritative sources in a hyperlinked environment. Journal of the ACM (JACM), 46(5), 604-632.

PrePeoHIN: Formulation

- Integrate PreP algorithm into NeoHIN
- PreP: $\min L_{O,i}(\eta_i, \rho_i, \phi_i, \theta_i)$
  $= -\log(p(pc_i, \eta_i, \rho_i, \phi_i, \theta_i | \alpha_i, \beta_i))$



Objective of PreP from the i-th domain

Contribution of meta-paths

For two similar domains i, j, the importance of a meta path t in these two domains (η_{it}, η_{jt}) should be similar.

- PreP-NeoHIN:

$$\min L = \sum_{i=1}^{g} L_{O,i}(\eta_i, \rho_i, \phi_i, \theta_i) + \gamma \sum_{i=1}^{g} \sum_{j=1}^{g} \sum_{t=1}^{|T|} \left(\frac{\eta_{it}}{\sqrt{\mathbf{d}(i)}} - \frac{\eta_{jt}}{\sqrt{\mathbf{d}(j)}}\right)^2 \mathbf{G}(i,j),$$

Objective of PreP from every domain

Cross-domain consistency

[1] Shi, Y., Chan, P. W., Zhuang, H., Gui, H., & Han, J.. Prep: Path-based relevance from a probabilistic perspective in heterogeneous information networks. KDD 2017.

NeoHIN: Experimental Results

- Two synthetic networks with 2,000 nodes
- Cross-domain link prediction on synthetic dataset

Algorithm	Accuracy					
	K=5	K=10	K=15	K=20	K=25	K=30
CrossRank	**0.045**	**0.090**	0.135	0.180	0.202	0.225
HITS-NoN	0.034	0.056	0.112	0.124	0.135	0.180
HITS-NeoHIN	**0.045**	**0.090**	**0.146**	**0.191**	**0.213**	**0.236**

- Metapath-based link prediction

Algorithm	ROC-AUC	AUPRC
PReP	0.553	0.307
PReP-NeoHIN	**0.566**	**0.404**

Observation: NeoHIN has achieved the best performance.

[1] Z. Xu, S. Zhang, Y. Xia, L. Xiong and H. Tong, "Ranking on Network of Heterogeneous Information Networks," 2020 IEEE International Conference on Big Data (Big Data), 2020, pp. 848-857, doi: 10.1109/BigData50022.2020.9378121.

NeoHIN: Experimental Results

- Five domain networks from AMiner
- Cross-domain link prediction

Algorithm	Accuracy				
	K=100	K=200	K=300	K=400	K=500
PageRank	0.016	0.092	0.131	0.150	0.198
CrossRank	0.063	0.120	0.162	0.223	0.258
HITS	0.042	0.087	0.128	0.154	0.172
HITS-NoN	0.064	0.130	0.172	0.233	0.273
HITS-HIN	0.016	0.082	0.126	0.143	0.167
HITS-NeoHIN	**0.109**	**0.160**	**0.203**	**0.246**	**0.291**

- Metapath-based link prediction

Algorithm	ROC-AUC	AUPRC
PathCount	0.414	0.464
PathSim	0.491	0.513
JoinSim	0.574	0.579
PReP	0.542	0.524
PReP-NeoHIN	**0.584**	**0.607**

[1] Z. Xu, S. Zhang, Y. Xia, L. Xiong and H. Tong, "Ranking on Network of Heterogeneous Information Networks," 2020 IEEE International Conference on Big Data (Big Data), 2020, pp. 848-857, doi: 10.1109/BigData50022.2020.9378121.
[2] J. Tang, J. Zhang, L. Yao, J. Li, L. Zhang, and Z. Su, "Arnetminer: extraction and mining of academic social networks," in *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2008, pp. 990–998.

Overview of Part II

Multi-network Mining Algorithms

Classification
- Label propagation-based multi-view/domain classification
- GNN-based embedding
- Contrastive learning for multi-view

Hyperlink prediction
- NMF-based method
- Autoencoder -based embedding
- GNN-based embedding

Multi-network association
- Label propagation-based method
- w/o attribute
- w/ attribute
- Dependency inference
- Network alignment

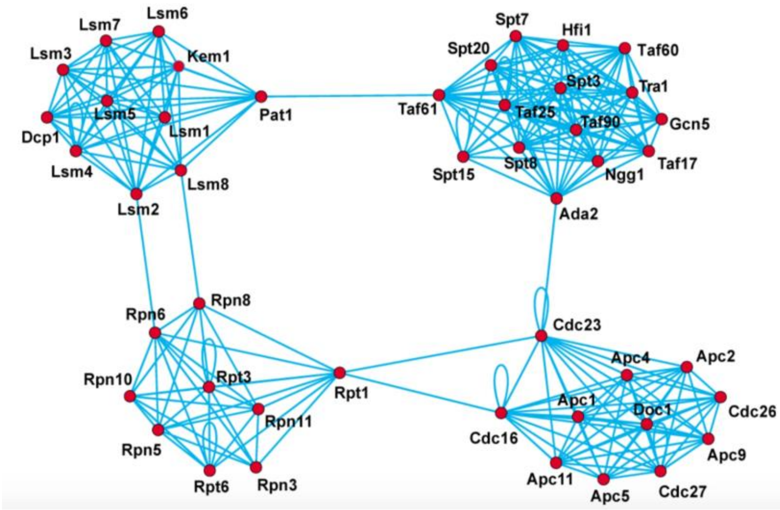
Ranking
- Consistency based homogeneous
- Consistency based heterogeneous

Clustering
- **Multi-view network clustering**
- NoN clustering

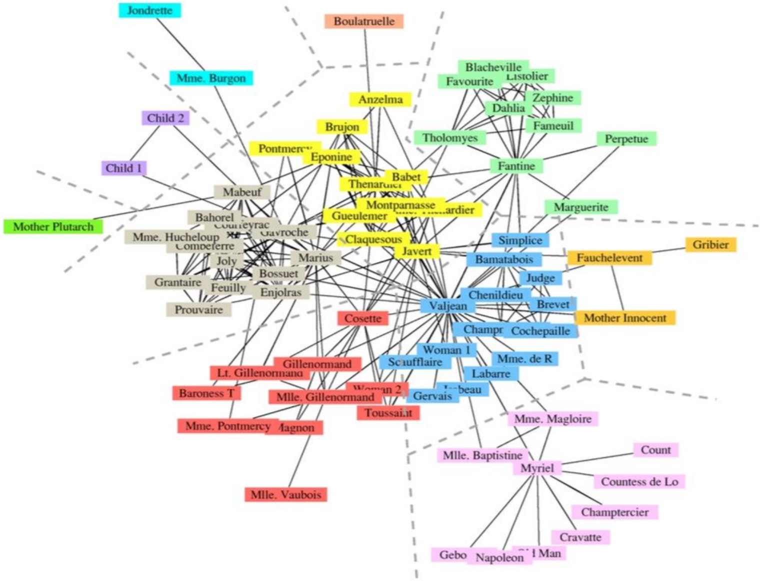
Multi-network embedding
- MF-based embedding
- Random walk-based embedding
- GNN-based embedding

Single Network Clustering: Motivations

- Detect sub-networks that satisfy certain properties
- Many connections within clusters
- Few connections across clusters



Protein complexes in a PPI network

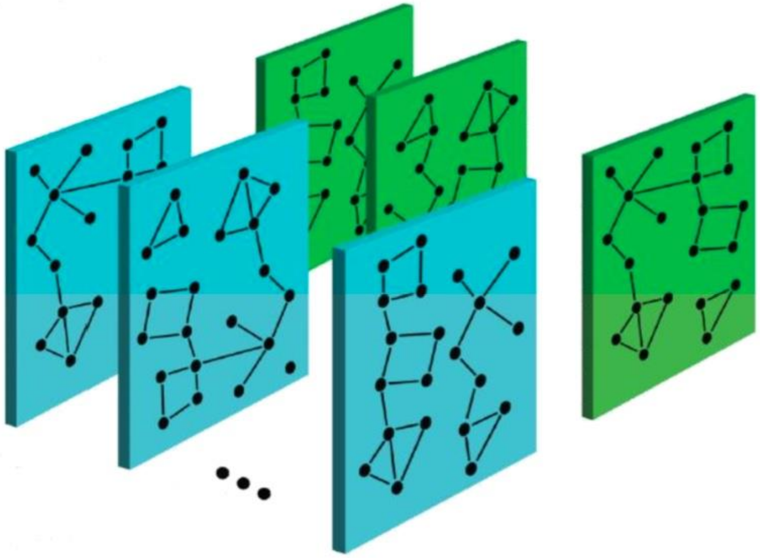


A novel character interaction network

[1] Credit to 592-ST-NSB-Clustering.pdf
[2] Jingchao Ni, Hanghang Tong, Wei Fan, and Xiang Zhang. 2015. Flexible and Robust Multi-Network Clustering. Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining.
[3] Mark EJ Newman and Michelle Girvan. Finding and evaluating community structure in networks. Physical review E 69.2 (2004): 026113.

Multi-Network Clustering: Motivations

- Networks are often collected from multiple sources.
- Single network is noisy and provides partial knowledge.
- Multi-network can provide complementary information.



An example of multi-network

[1] Kivelä, Mikko, et al. "Multilayer networks." Journal of complex networks 2.3 (2014): 203-271.

Co-regularized Multi-view Spectral Clustering (CMSC): Single View

- let $\mathrm{X} = \{\mathrm{x}_1^{(v)}, \mathrm{x}_2^{(v)}, \ldots, \mathrm{x}_n^{(v)}\}$ denote the examples in view v and K^{v} denote the similarity matrix.
- The single view spectral clustering is:

$$\max_{\mathbf{U}^{(v)} \in \mathbb{R}^{n \times k}} tr\left(\mathbf{U}^{(v)^T} \mathcal{L}^{(v)} \mathbf{U}^{(v)}\right), \quad \text{s.t.} \quad \mathbf{U}^{(v)^T} \mathbf{U}^{(v)} = I$$

- where $\mathbf{L}^{(v)}$ is the normalized graph Laplacian.

[1] Figure credit to https://developer.nvidia.com/discover/cluster-analysis
[2]Abhishek Kumar, Piyush Rai, and Hal Daumé. 2011. Co-regularized multi-view spectral clustering. In Proceedings of the 24th International Conference on Neural Information Processing Systems (NIPS'11). Curran Associates Inc., Red Hook, NY, USA, 1413–1421.

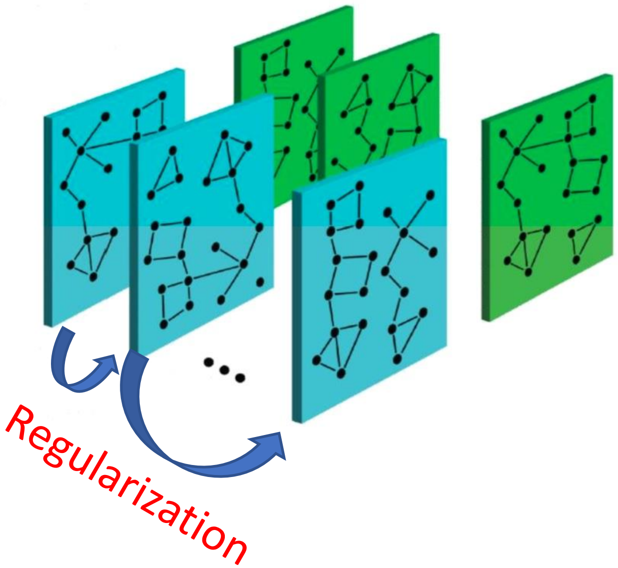
CMSC-Pairwise: Intuitions

- Different views provide compatible information.
- Regularize the disagreement between views v and w.
  - Two-view case:

$$D(\mathbf{U}^{(v)}, \mathbf{U}^{(w)}) = \left\| \frac{\boxed{\mathbf{K}_{\mathbf{U}^{(v)}}}}{\|\mathbf{K}_{\mathbf{U}^{(v)}}\|_F^2} - \frac{\mathbf{K}_{\mathbf{U}^{(w)}}}{\|\mathbf{K}_{\mathbf{U}^{(w)}}\|_F^2} \right\|_F^2$$

  - Reformulation:

$$D(\mathbf{U}^{(v)}, \mathbf{U}^{(w)}) = -tr\left(\mathbf{U}^{(v)}\mathbf{U}^{(v)^T}\mathbf{U}^{(w)}\mathbf{U}^{(w)^T}\right)$$



CMSC-Pairwise: Formulation

- Multi-view case:
  - The objective function:

$$\max_{\mathbf{U}^{(1)},\mathbf{U}^{(2)},\ldots,\mathbf{U}^{(m)}\in\mathbb{R}^{n\times k}} \boxed{\sum_{v=1}^{m} tr\left(\mathbf{U}^{(v)^T}\mathcal{L}^{(v)}\mathbf{U}^{(v)}\right)} + \lambda \sum_{\substack{1\le v,w\le m\\ v\neq w}} \boxed{tr\left(\mathbf{U}^{(v)}\mathbf{U}^{(v)^T}\mathbf{U}^{(w)}\mathbf{U}^{(w)^T}\right)}$$

$$\text{s.t.}\quad \mathbf{U}^{(v)^T}\mathbf{U}^{(v)} = I,\ \forall\, 1\le v\le V$$

Within-layer spectral clustering

Cross-layer pairwise regularization

  - Transformed Laplacian form:

$$\max_{\mathbf{U}^{(v)}} tr\left\{\mathbf{U}^{(v)^T}\left(\mathcal{L}^{(v)} + \lambda \sum_{\substack{1\le w\le m,\\ w\neq v}} \mathbf{U}^{(w)}\mathbf{U}^{(w)^T}\right)\mathbf{U}^{(v)}\right\},\quad \text{s.t.}\quad \mathbf{U}^{(v)^T}\mathbf{U}^{(v)} = I$$

CMSC-Centroid: Formulation

- Key idea: Set an underlying centroid matrix $\mathbf{U}^*$

$$\max_{\mathbf{U}^{(1)},\mathbf{U}^{(2)},\ldots,\mathbf{U}^{(m)},\mathbf{U}^*\in\mathbb{R}^{n\times k}} \sum_{v=1}^{m} tr\left(\mathbf{U}^{(v)^T}\mathcal{L}^{(v)}\mathbf{U}^{(v)}\right) + \boxed{\sum_{v}\lambda_v tr\left(\mathbf{U}^{(v)}\mathbf{U}^{(v)^T}\mathbf{U}^*\mathbf{U}^{*^T}\right)},$$
$$\text{s.t.}\quad \mathbf{U}^{(v)^T}\mathbf{U}^{(v)} = I,\ \forall\, 1\leq v\leq V,\quad \mathbf{U}^{*^T}\mathbf{U}^* = I$$

Centroid matrix regularization

- Solving for the $\mathbf{U}^*$ requires:

$$\max_{\mathbf{U}^*\in\mathbb{R}^{n\times k}} \sum_{v}\lambda_v tr\left(\mathbf{U}^{(v)}\mathbf{U}^{(v)^T}\mathbf{U}^*\mathbf{U}^{*^T}\right),\quad \text{s.t.}\quad \mathbf{U}^{*^T}\mathbf{U}^* = I$$

CMSC: Experimental Results

- Two synthetic datasets and three real datasets
- Clustering on five datasets with NMI metric

Method	Synth data 1	Synth data 2	Reuters	Handwritten	Caltech
Best Single View	0.267 (0.0)	0.898 (0.0)	0.287 (0.019)	0.641 (0.008)	0.510 (0.008)
Feature Concat	0.294 (0.0)	0.923 (0.0)	0.298 (0.020)	0.619 (0.015)	–
Kernel Addition	0.339 (0.0)	0.973 (0.0)	0.323 (0.021)	0.744 (0.030)	0.383 (0.008)
Kernel Product	0.277 (0.0)	0.959 (0.0)	0.123 (0.010)	0.754 (0.026)	0.429 (0.007)
CCA	0.330 (0.0)	0.932 (0.0)	0.147 (0.003)	0.682 (0.019)	0.466 (0.007)
Min Disagreement	0.313 (0.0)	0.936 (0.0)	0.342 (0.024)	0.745 (0.024)	0.389 (0.008)
Co-regularized (P) (2)	0.378 (0.0)	0.981 (0.0)	0.375 (0.002)	0.759 (0.031)	0.527 (0.007)
Co-regularized (P) (3)	–	0.989 (0.0)	–	–	0.533 (0.008)
Co-regularized (P) (4)	–	–	–	–	0.564 (0.007)
Co-regularized (C) (2)	0.367 (0.0)	0.955 (0.0)	0.360 (0.025)	0.768 (0.025)	0.522 (0.004)
Co-regularized (C) (3)	–	0.989 (0.0)	–	–	0.512 (0.007)
Co-regularized (C) (4)	–	–	–	–	0.561 (0.005)

- (1), (2), (3) indicate the number of views used.
- Letters (P) and (C) indicate pairwise and centroid based methods.

Overview of Part II

Multi-network Mining Algorithms

Classification
- Label propagation-based multi-view/domain classification
- GNN-based embedding
- Contrastive learning for multi-view

Hyperlink prediction
- NMF-based method
- Autoencoder-based embedding
- GNN-based embedding

Multi-network association
- Label propagation-based method
- w/o attribute
- w/ attribute
- Dependency inference
- Network alignment

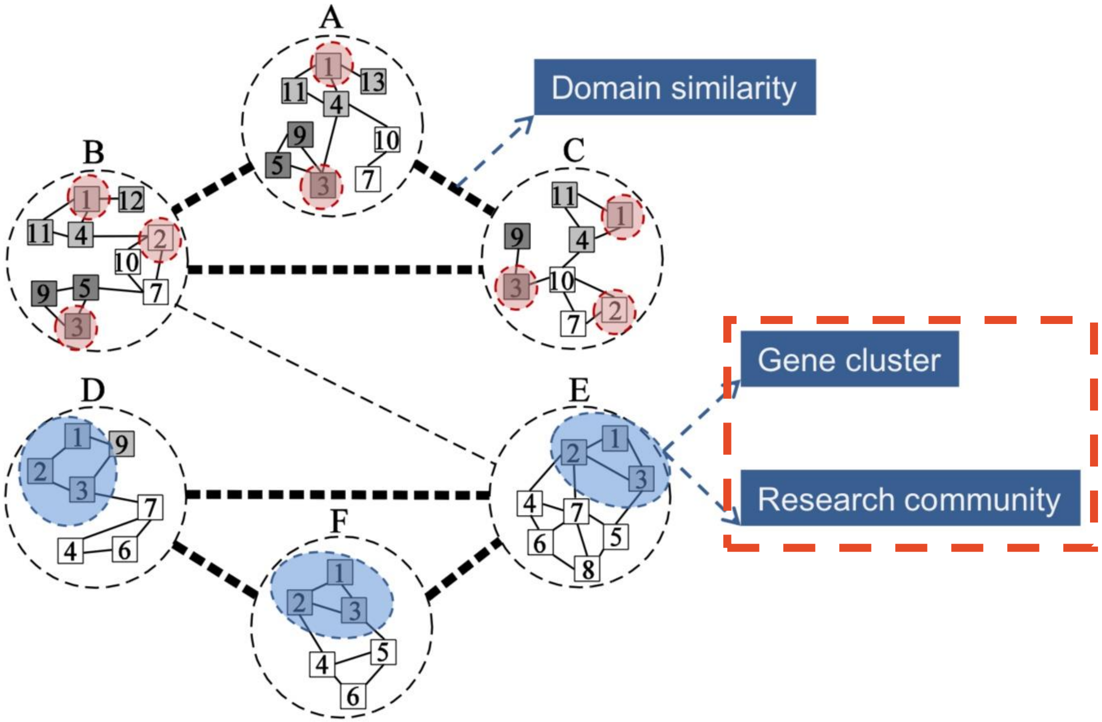
Ranking
- Consistency based homogeneous
- Consistency based heterogeneous

Clustering
- Multi-view network clustering
- **NoN clustering**

Multi-network embedding
- MF-based embedding
- Random walk-based embedding
- GNN-based embedding

NoN Clustering (NoNClus): Intuitions

- Different networks have different meanings for clusters.
- Domain similarity is important for the clustering task.



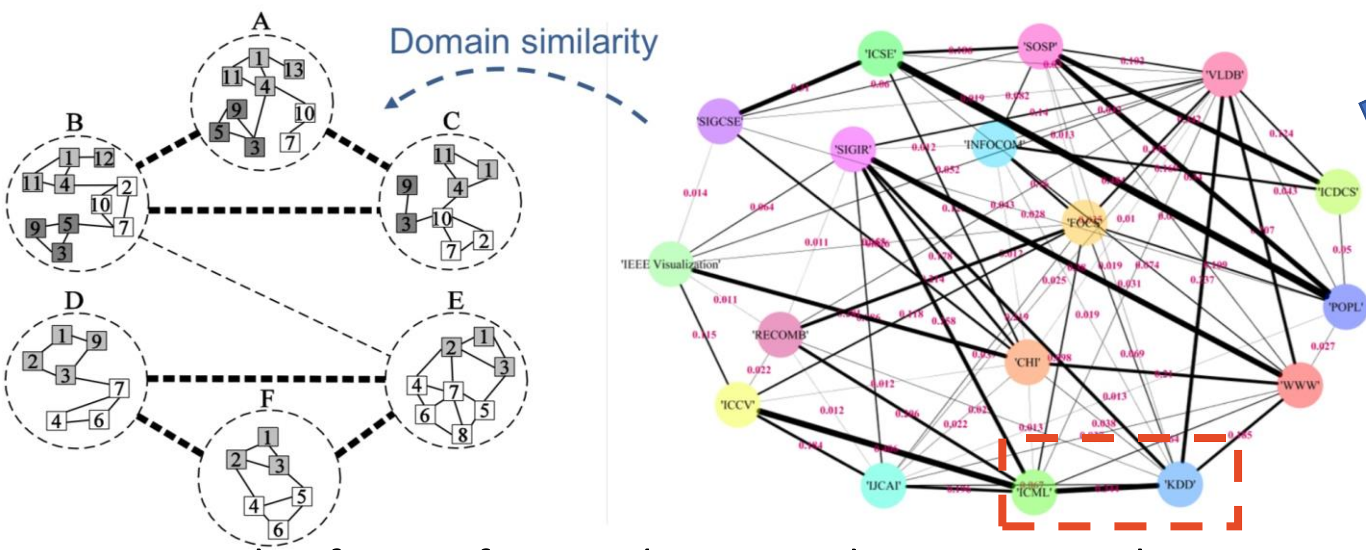
Example:
Dependent on the meaning of domain network E, the cluster in E can represent Gene or research community.

[1]Slides credit to https://nijingchao.github.io/slide/kdd15_nonclus_slides.pdf

[2]Jingchao Ni, Hanghang Tong, Wei Fan, and Xiang Zhang. 2015. Flexible and Robust Multi-Network Clustering. Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. Association for Computing Machinery, New York, NY, USA, 835–844. DOI:https://doi.org/10.1145/2783258.2783262

NoNClus: Formulation

- Phase 1: main network clustering.
  - Symmetric Non-negative Matrix Factorization (SNMF).
  - Minimizing $$\boxed{J_M = \left\| \mathbf{G} - \mathbf{H}\mathbf{H}^T \right\|_F^2 \quad s.t. \quad \mathbf{H} \geq 0}$$



Cluster the main venue network.

An example of NoN of research papers, domain network represents venue.

NoNClus: Formulation (cont.)

- Phase 2: domain specific network clustering (Simplified).
  - Domain-specific networks in the same cluster have same underlying clustering structure.
  - All domains have n nodes and t clusters.
  - Let the domain cluster assignment vector for node x in $\mathbf{A}^{(i)}$ be $u_{x*}^{(i)}$ (i=1, …, g).
  - Define k hidden domain cluster assignment vectors $v_{x*}^{(j)}$ (j=1, …, k)

$$J_x = \sum_{i=1}^{g}\sum_{j=1}^{k} h_{ij} \left\| u_{x*}^{(i)} - v_{x*}^{(j)} \right\|_F^2 \Rightarrow J_D = \underbrace{\sum_{i=1}^{g} \left\| \mathbf{A}^{(i)} - \mathbf{U}^{(i)}(\mathbf{U}^{(i)})^T \right\|_F^2}_{\text{Domain-specific network clustering}} + \underbrace{a \sum_{i=1}^{g}\sum_{j=1}^{k} h_{ij} \left\| \mathbf{U}^{(i)} - \mathbf{V}^{(j)} \right\|_F^2}_{\text{Main cluster guided regularization}}$$

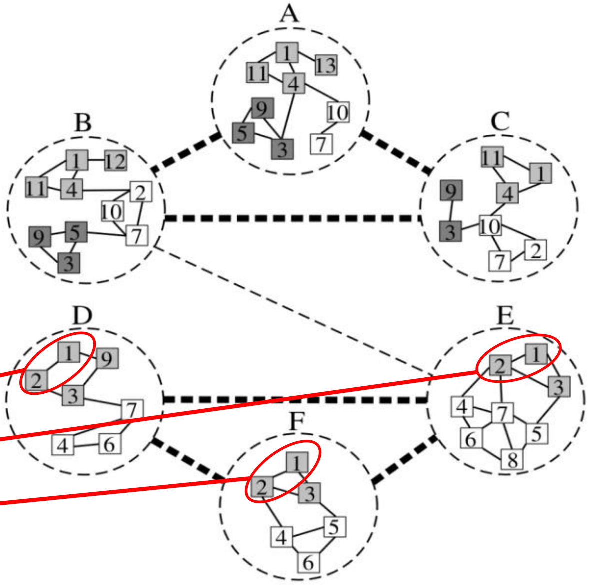
Recall h_{ij} represents main cluster membership

NoNClus: Formulation (cont.)

- Phase 2: domain specific network clustering (General)
  - Different domains can have different sets of nodes.
  - Different domains can have different number of clusters.
  - Indirect regularization
  - Minimize

$$\boxed{h_{ij}\left(\hat{\mathbf{u}}_{x*}^{(ij)}(\hat{\mathbf{u}}_{y*}^{(ij)})^T - \hat{\mathbf{v}}_{x*}^{(ij)}(\hat{\mathbf{v}}_{y*}^{(ij)})^T\right)^2}$$

Nodes in same domain networks   Regularized vector



Example: since D, E, F are in the same cluster, if nodes 1 and 2 have similar cluster assignments in D, their cluster assignments in E and F should also be similar.

NoNClus: Experimental Results

- Clustering accuracy on two synthetic datasets
- In *view* dataset, all $\mathbf{A}^{(i)}$ have the same size.
- In *dom* dataset, different $\mathbf{A}^{(i)}$ have different sizes.

Dataset	Method	Main cluster 1			Main cluster 2			Main Cluster 3				
		Net 1	Net 2	Net 3	Net 4	Net 5	Net 6	Net 7	Net 8	Net 9	Net 10	Overall
view	SNMF	0.8751	0.8716	0.8735	0.8796	0.8732	0.8754	0.8722	0.8690	0.8682	0.8746	0.8732
	Spectral	0.8587	0.8586	0.8675	0.8619	0.8571	0.8624	0.8626	0.8582	0.8583	0.8622	0.8607
	CTSC	0.6249	0.6258	0.6279	0.6221	0.6236	0.6196	0.9157	0.9118	0.9106	0.9181	0.7400
	PairCRSC	0.9166	0.9174	0.9227	0.9186	0.9176	0.9173	0.9355	0.9335	0.9378	0.9353	0.9252
	CentCRSC	0.9050	0.9031	0.9090	0.9021	0.9090	0.9077	0.9391	0.9408	0.9342	0.9378	0.9188
	TF	–	–	–	–	–	–	–	–	–	–	0.6505
	CGC	0.6364	0.6337	0.6407	0.6385	0.6273	0.6316	0.7332	0.7365	0.7251	0.7210	0.6724
	NoNClus	**0.9444**	**0.9403**	**0.9463**	**0.9447**	**0.9435**	**0.9418**	**0.9617**	**0.9621**	**0.9643**	**0.9629**	**0.9512**
dom	SNMF	0.6584	0.6687	0.6583	0.7123	0.7063	0.7129	0.6558	0.6596	0.6620	0.6630	0.6787
	Spectral	0.5554	0.5618	0.5556	0.5799	0.5768	0.5811	0.5167	0.5188	0.5241	0.5242	0.5490
	CGC	0.7303	0.7297	0.7229	0.7992	0.7962	0.7965	0.7859	0.7840	0.7837	0.7876	0.7797
	NoNClus	**0.7882**	**0.7960**	**0.7914**	**0.8649**	**0.8650**	**0.8654**	**0.8409**	**0.8363**	**0.8367**	**0.8389**	**0.8388**

- Observation: *dom* dataset is difficult than *view* dataset.

Overview of Part II

Multi-network Mining Algorithms

Classification

- Label propagation-based multi-view/domain classification
- GNN-based embedding
- Contrastive learning for multi-view

Hyperlink prediction

- NMF-based method
- Autoencoder-based embedding
- GNN-based embedding

Multi-network association

- Label propagation-based method
- w/o attribute
- w/ attribute
- Dependency inference
- Network alignment

Ranking

- Consistency based homogeneous
- Consistency based heterogeneous

Clustering

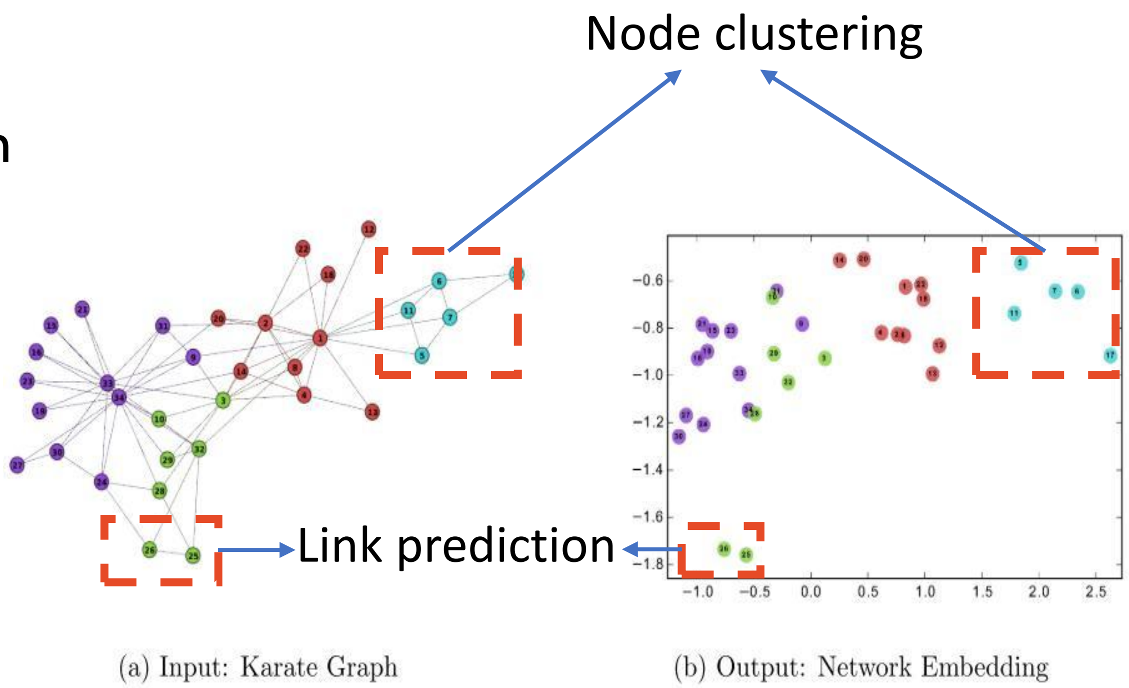
- Multi-view network clustering
- NoN clustering

Multi-network embedding

- **MF-based embedding**
- Random walk-based embedding
- GNN-based embedding

Embedding on Single Network: Motivation

- Represent each node with a vector
- Applications:
  - Node classification
  - Link prediction
  - Node visualization

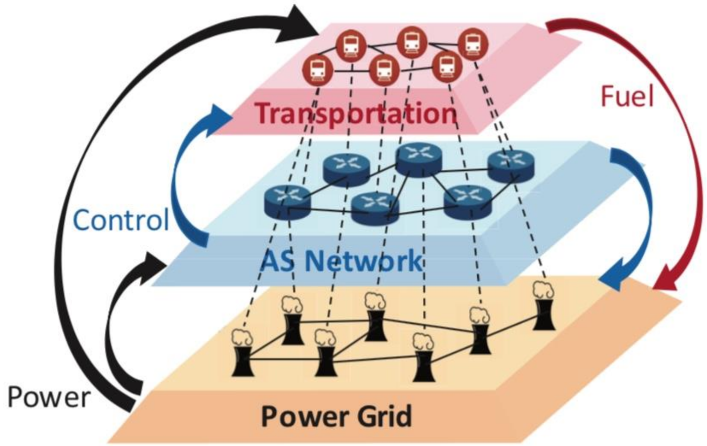


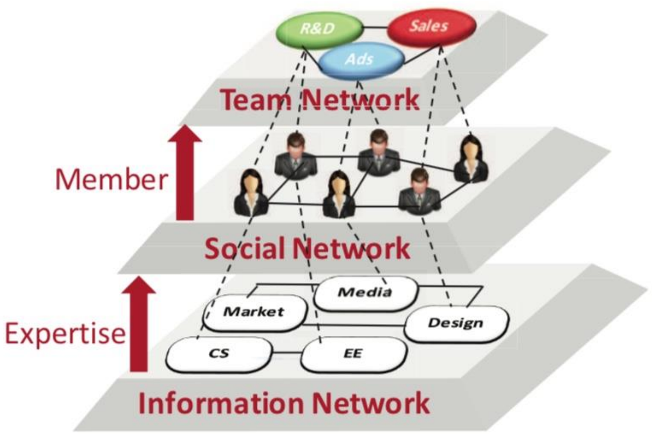
Visualization of network embedding.

[1] Bryan Perozzi, Rami Al-Rfou, and Steven Skiena. 2014. DeepWalk: online learning of social representations. In Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining.
[2] Tang, Jian, et al. "Line: Large-scale information network embedding." *Proceedings of the 24th international conference on world wide web*. 2015.
[3] Tang, Jian, et al. "Visualizing large-scale and high-dimensional data." *Proceedings of the 25th international conference on world wide web*.

Multi-layered Network Embedding: Motivations

- Current works focus on single network embedding.
- Networks are complicated with cross-domain interactions.
  - Examples: critical infrastructure systems and organization-level collaboration platform.



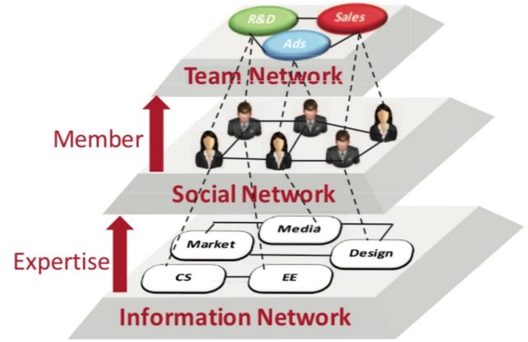


[1] Li, Jundong, C. Chen, Hanghang Tong and H. Liu. “Multi-Layered Network Embedding.” *SDM* (2018).

Multi-layered Network Embedding (MANE) : Problem Definition

- Given:
  - The embedding dimension $d_1, d_2, \ldots, d_g$ for different layers;
  - A set of g within-layer adjacency matrices A = $\{\mathbf{A}_1, \ldots, \mathbf{A}_g\}$;
  - Observed cross-layer dependency matrix D = $\{\mathbf{D}_{i,j}, (i, j = 1, \ldots, g)(i \neq j)\}$ where $\mathbf{D}_{i,j} \in \{0,1\}^{n_i \times n_j}$ denotes the cross-layer network dependency between $\mathbf{A}_i$ and $\mathbf{A}_j$;
- Output: the embedding representation $\mathbf{F}_i \in R^{n_i \times d_i}$

Input:
Team Network: $\mathbf{A}_1$
Member Dependency: $\mathbf{D}_{2,1}$
Social Network: $\mathbf{A}_2$
Expertise Dependency: $\mathbf{D}_{3,2}$
Information Network: $\mathbf{A}_3$

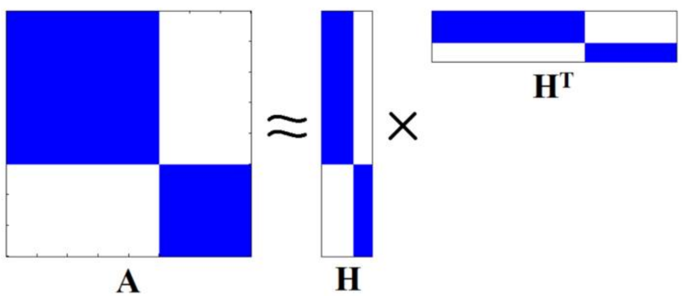


Output:
Team Network: $\mathbf{F}_1$
Social Network: $\mathbf{F}_2$
Information Network: $\mathbf{F}_3$

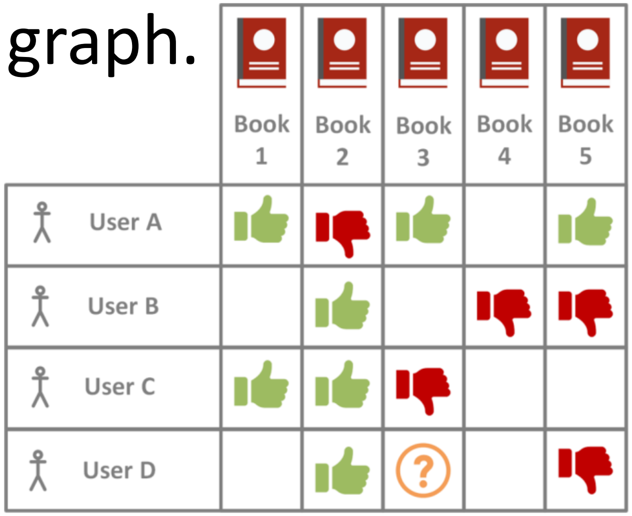
[1] Li, Jundong, C. Chen, Hanghang Tong and H. Liu. "Multi-Layered Network Embedding." *SDM* (2018).

Multi-layered Network Embedding (MANE): Intuition

- Symmetric non-negative matrix factorization (SNMF).
- Collaborative filtering for bipartite graph.



SNMF for single network



Collaborative filtering

- Key idea:
  - MF for cross-layer edge
  - Smoothness regularization for within-layer edge

[1] Figure credit to https://www.advancinganalytics.co.uk/blog/2020/5/13/recommendation-systems.
[2] Kuang, Da, Chris Ding, and Haesun Park. "Symmetric nonnegative matrix factorization for graph clustering." *Proceedings of the 2012 SIAM international conference on data mining*. Society for Industrial and Applied Mathematics, 2012.

MANE: Within-layer Connection

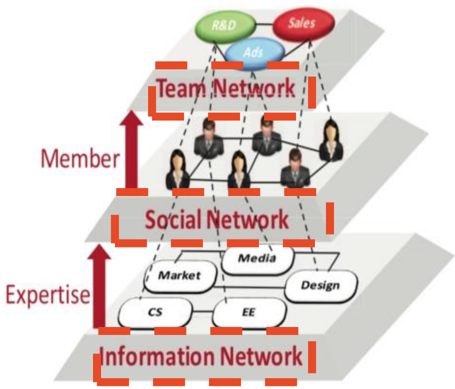
- Smoothness requirement:

$$\min_{\mathbf{F}_k} \frac{1}{2} \sum_{i,j} \mathbf{A}_k(i,j) \| \frac{\mathbf{F}_k(i,:)}{\sqrt{Deg_i}} - \frac{\mathbf{F}_k(j,:)}{\sqrt{Deg_j}} \|_2^2,$$

- Reformulation:

Laplacian matrix for the k-th layer network

$$\max_{\mathbf{F}_k} tr(\mathbf{F}_k' \boxed{\mathbf{L}_k} \mathbf{F}_k) \text{ s.t. } \mathbf{F}_k'\mathbf{F}_k = \mathbf{I},$$



- Objective function for all layers:

$$\max_{\mathbf{F}_1,\mathbf{F}_2,\ldots,\mathbf{F}_g} \sum_{i=1}^{g} tr(\mathbf{F}_i'\mathbf{L}_i\mathbf{F}_i) \text{ s.t. } \mathbf{F}_i'\mathbf{F}_i = \mathbf{I} \; (\forall i = 1,,,.,g)$$

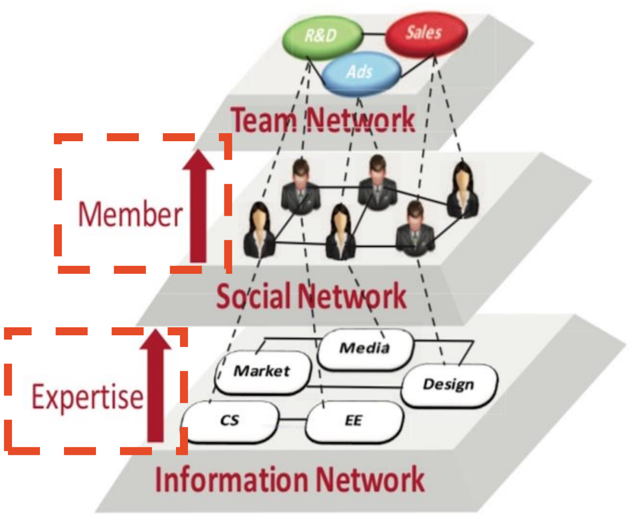
[1] Li, Jundong, C. Chen, Hanghang Tong and H. Liu. "Multi-Layered Network Embedding." *SDM* (2018).

MANE: Cross-Layer Dependency

- Interaction requirement:

$$\min_{\mathbf{F}_i,\mathbf{F}_j,\mathbf{K}_{ij}} \|\mathbf{D}_{ij} - \mathbf{F}_i \boxed{\mathbf{K}_{ij}} \mathbf{F}_j'\|_F^2$$

$$\text{s.t. } \mathbf{F}_i'\mathbf{F}_i = \mathbf{I},\ \mathbf{F}_j'\mathbf{F}_j = \mathbf{I}.$$



$\mathbf{K}_{ij}$ is the interaction matrix.

- Objective function:

$$\max_{\mathbf{F}_i,\mathbf{K}_{ij}} \sum_{i=1}^{g} tr(\mathbf{F}_i'\mathbf{L}_i\mathbf{F}_i) - \alpha \sum_{i,j=1}^{g} \|\mathbf{D}_{ij} - \mathbf{F}_i\mathbf{K}_{ij}\mathbf{F}_j'\|_F^2$$

$$\text{s.t. } \mathbf{F}_i'\mathbf{F}_i = \mathbf{I}\ (\forall i = 1, ..., g),$$

[1] Li, Jundong, C. Chen, Hanghang Tong and H. Liu. "Multi-Layered Network Embedding." *SDM* (2018).

MANE: Experimental Result

- Node classification on AMiner datasets (3 layers)
- Metrics: Macro-F1 and Micro-F1

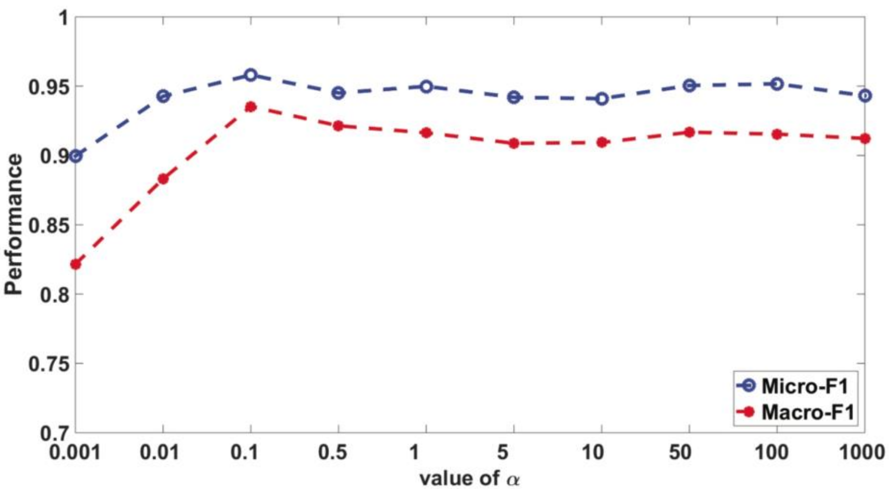
Training Ratio		10%	20%	30%	40%	50%	60%	70%	80%	90%
Macro-F1	CCF	85.34	88.37	89.70	90.91	91.07	91.24	91.31	91.44	91.62
	CMF	84.33	87.65	88.84	89.39	90.38	90.43	90.47	90.50	91.20
	NMF	74.40	75.52	75.97	76.49	80.91	84.88	85.20	86.00	86.79
	Deepwalk	82.67	85.05	86.02	86.81	87.32	87.37	87.47	88.01	88.14
	Deepwalk-within	63.92	67.33	68.48	68.87	69.89	70.50	71.70	71.87	72.27
	LINE	83.59	84.85	86.90	87.53	88.09	88.15	88.34	88.37	88.50
	LINE-within	44.72	51.80	56.58	61.58	63.27	66.55	67.69	68.21	70.85
	Metapath2vec	85.46	86.53	87.23	87.71	88.06	89.45	89.42	89.99	90.80
	MANE	**88.80**	**90.46**	**91.15**	**91.78**	**92.31**	**92.37**	**92.38**	**92.43**	**92.72**
Micro-F1	CCF	92.44	92.89	92.87	93.35	93.76	93.94	94.40	94.38	94.48
	CMF	92.07	92.88	92.62	93.10	93.25	93.57	94.18	94.30	92.64
	NMF	88.06	88.28	88.48	88.73	89.42	89.55	89.80	90.07	90.36
	Deepwalk	89.99	90.54	90.82	91.08	91.33	91.59	91.72	91.84	92.03
	Deepwalk-within	83.23	84.21	84.70	84.75	85.08	85.11	85.52	85.69	86.32
	LINE	89.16	90.83	91.40	91.74	91.90	91.93	91.96	92.05	92.20
	LINE-within	66.51	72.27	73.97	75.74	76.61	77.34	78.13	78.45	79.36
	Metapath2vec	92.16	93.51	93.77	93.89	93.74	93.93	93.94	94.37	94.96
	MANE	**93.15**	**94.44**	**94.73**	**94.75**	**95.19**	**95.28**	**95.31**	**95.45**	**95.59**

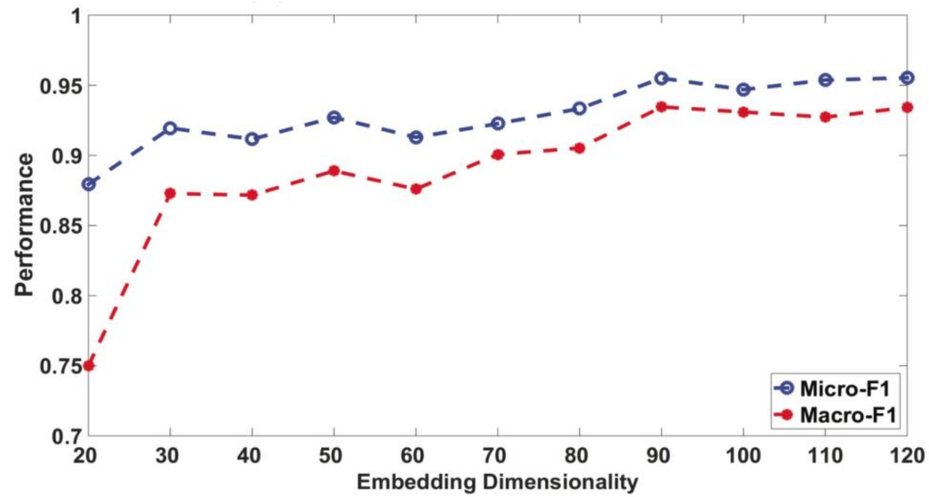
- Observation: MANE is better than single network embedding methods.

[1] Li, Jundong, C. Chen, Hanghang Tong and H. Liu. "Multi-Layered Network Embedding." *SDM* (2018).

MANE: Parameter Study

- Performance w.r.t. the cross-domain parameter α
- Performance w.r.t. to embedding dimensionality





- Observations:
  - Integrate the cross-layer part boosts the performance.
  - Large dimension captures more information.

[1] Li, Jundong, C. Chen, Hanghang Tong and H. Liu. "Multi-Layered Network Embedding." *SDM* (2018).

Overview of Part II

Multi-network Mining Algorithms

Classification
- Label propagation-based multi-view/domain classification
- GNN-based embedding
- Contrastive learning for multi-view

Hyperlink prediction
- NMF-based method
- Autoencoder -based embedding
- GNN-based embedding

Multi-network association
- Label propagation-based method
- w/o attribute
- w/ attribute
- Dependency inference
- Network alignment

Ranking
- Consistency based homogeneous
- Consistency based heterogeneous

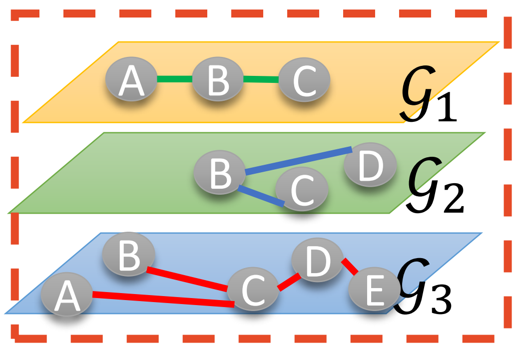
Clustering
- Multi-view network clustering
- NoN clustering

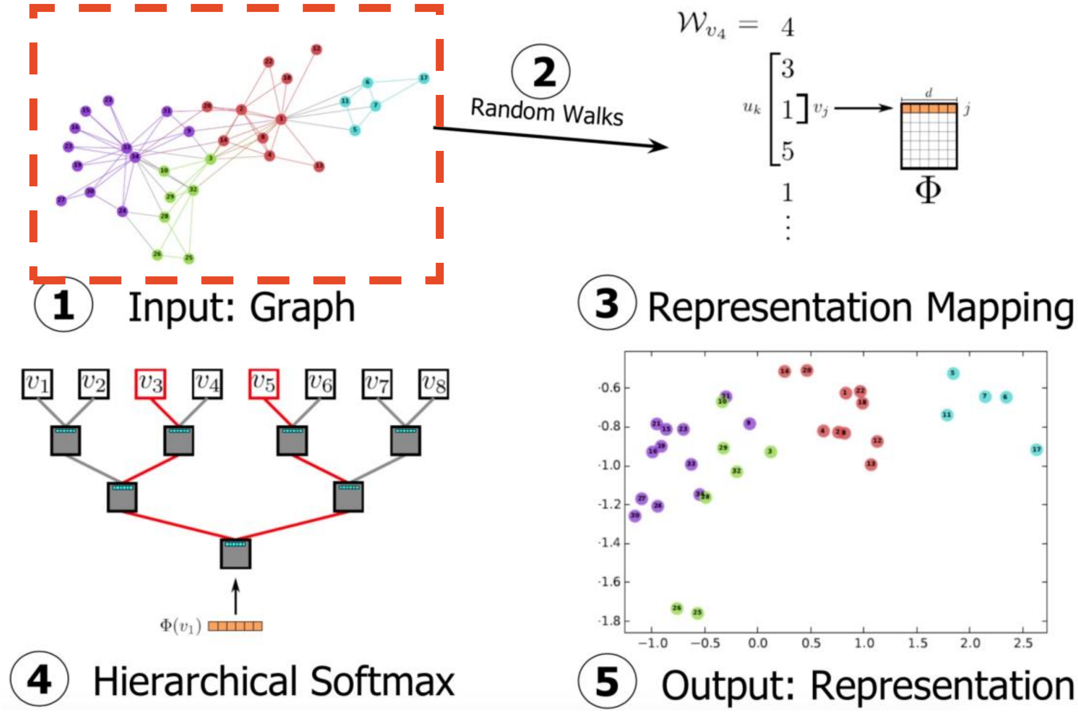
Multi-network embedding
- MF-based embedding
- **Random walk-based embedding**
- GNN-based embedding

Multiplex Network Embedding (MNE): Intuitions

- DeepWalk shows advantages in single network embedding
- Leverage common nodes across layers for regularization

(1) Nodes in single network have unique connectivity;
(2) Nodes in multiplex network demonstrates different connectivities across layers.



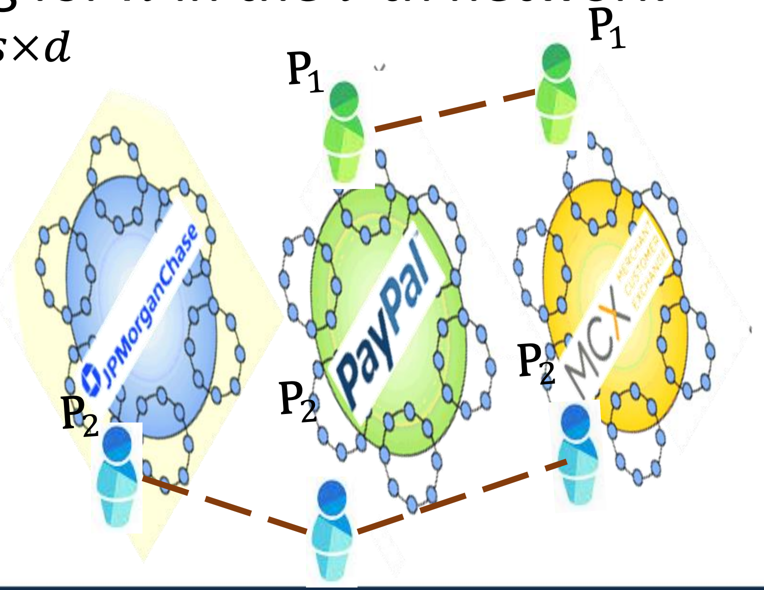


[1] Zhang, Hongming, et al. "Scalable Multiplex Network Embedding." *IJCAI*. Vol. 18. 2018.
[2] Bryan Perozzi, Rami Al-Rfou, and Steven Skiena. 2014. DeepWalk: online learning of social representations. In Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining.

MNE: Problem Definition

- Given:
  - $\{G_i = (N_i, E_i)\}$,where N_i is the node set and E_i is the edge set.
- Find:
  - $\mathbf{b}_\mathrm{n} \in R^d$, the common-shared embedding for node n
  - $\mathbf{u}_\mathrm{n}^i \in R^s$, the specific embedding for n in the i-th network
  - A transformation matrix $\mathbf{X}^i \in R^{s \times d}$
  - Final embedding

$$\mathbf{v}_n^i = \mathbf{b}_n + w^i \cdot \mathbf{X}^{i^T} \mathbf{u}_n^i$$



P_1 appears in two layers while P_2 appears in three layers.

[1] Zhang, Hongming, et al. "Scalable Multiplex Network Embedding." *IJCAI*. Vol. 18. 2018.
[2] Figure credit to Si Zhang and Hanghang Tong. 2020. Network Alignment: Recent Advances and Future Directions. Proceedings of the 29th ACM International Conference on Information & Knowledge Management.

MNE: Algorithm

- Skipgram:

$$-\log P_{\theta^i}(n_{j-c}, ..., n_{j-1}, n_{j+1}, ..., n_{j+c} \mid n_j)$$

- Probability:

$$P_{\theta^i}(n_k \mid n_j) = \frac{\exp({\mathbf{v}'_{n_k}}^T \cdot \mathbf{v}^i_{n_j})}{\sum_n \exp({\mathbf{v}'_n}^T \cdot \mathbf{v}^i_{n_j})}$$

where $\mathbf{v}'_n$ represents the parameters of context vectors shared by all relation types.

- Word2vec: use negative sampling

$$E = -\log \sigma({\mathbf{v}'_{n_k}}^T \cdot \mathbf{v}^i_{n_j}) - \sum_{\boxed{n \in \mathcal{N}^i_{n_j}}} \log \sigma(-{\mathbf{v}'_n}^T \cdot \mathbf{v}^i_{n_j})$$

Negative sampling set

[1] Zhang, Hongming, et al. "Scalable Multiplex Network Embedding." *IJCAI*. Vol. 18. 2018.
[2] Mikolov, Tomas, et al. "Efficient estimation of word representations in vector space." *arXiv preprint arXiv:1301.3781* (2013).

MNE: Experimental Results

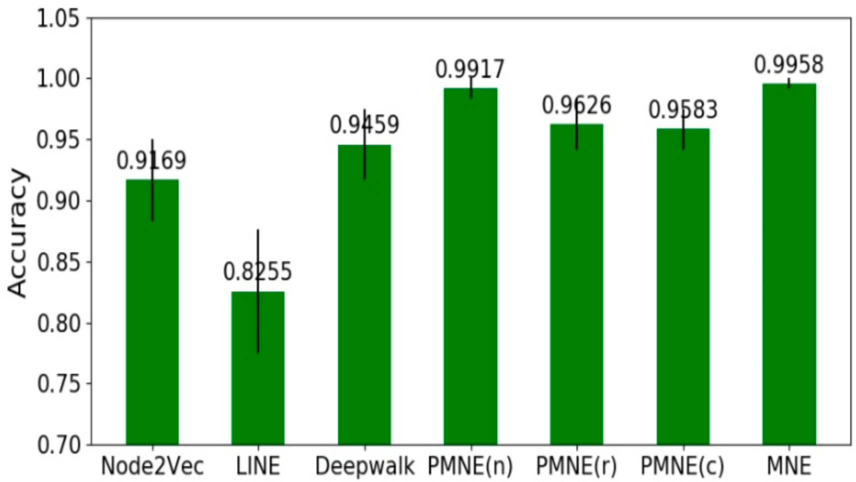
- Link prediction for six datasets with AUC metric

Model	Vickers	CKM	LAZEGA	C.ELEGANS	Twitter	Private
DeepWalk	0.821 (0.030)	0.781 (0.008)	0.780 (0.007)	0.821 (0.006)	0.502 (0.002)	0.621 (0.007)
LINE	0.676 (0.011)	0.637 (0.012)	0.695 (0.006)	0.732 (0.006)	0.519 (0.003)	0.512 (0.010)
Node2Vec	0.821 (0.030)	0.781 (0.008)	0.780 (0.007)	0.820 (0.006)	0.504 (0.003)	0.644 (0.010)
PMNE (n)	0.810 (0.032)	**0.917** (0.008)	0.792 (0.009)	0.843 (0.003)	0.446 (0.004)	0.629 (0.006)
PMNE (r)	0.844 (0.025)	0.904 (0.008)	0.813 (0.007)	0.835 (0.007)	0.446 (0.001)	0.659 (0.005)
PMNE (c)	0.837 (0.029)	0.847 (0.016)	0.797 (0.011)	0.824 (0.009)	0.449 (0.002)	0.506 (0.004)
Common Neighbor (CN)	0.799 (0.011)	0.877 (0.006)	0.809 (0.007)	0.869 (0.002)	0.592 (0.002)	0.691 (0.002)
Jaccard Coeficient (JC)	0.778 (0.007)	0.873 (0.006)	0.826 (0.007)	0.833 (0.001)	0.520 (0.002)	0.573 (0.004)
Adamic/Adar (AA)	0.803 (0.019)	0.875 (0.013)	0.814 (0.008)	0.881 (0.001)	0.592 (0.002)	0.691 (0.003)
MNE	**0.871** (0.014)	0.900 (0.010)	**0.839** (0.013)	**0.910** (0.006)	**0.622** (0.003)	**0.723** (0.002)

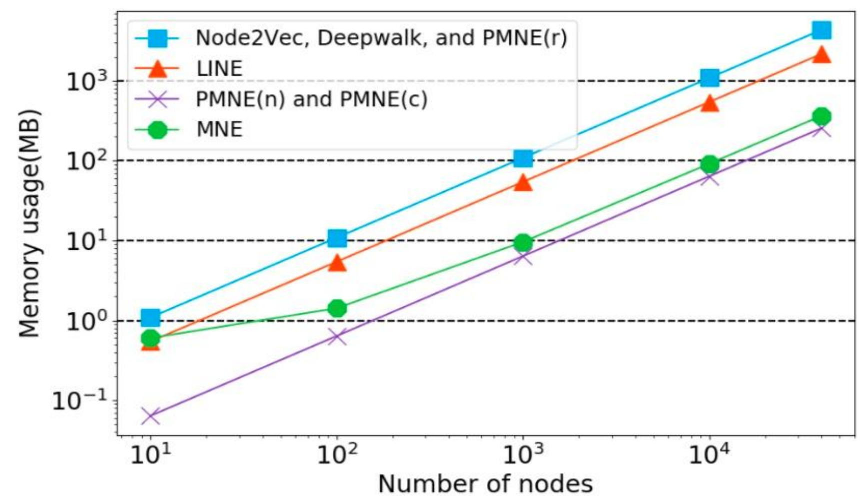
- Observation: MNE has a 2%-3% advantage over other baselines.

MNE: Experimental Results (cont.)

- Node classification and scalability



Node classification performance.



Scalability.

- Observation: MNE performs well and has a linear memory usage.

Overview of Part II

Multi-network Mining Algorithms

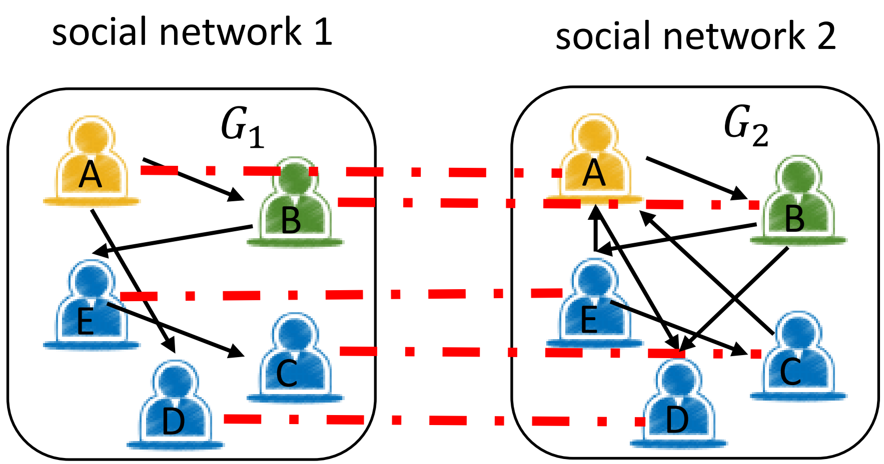
- **Classification**
  - Label propagation-based multi-view/domain classification
  - GNN-based embedding
  - Contrastive learning for multi-view
- **Hyperlink prediction**
  - NMF-based method
  - Autoencoder-based embedding
  - GNN-based embedding
- **Multi-network association**
  - Label propagation-based method
  - w/o attribute
  - w/ attribute
  - Dependency inference
  - Network alignment
- **Ranking**
  - Consistency based homogeneous
  - Consistency based heterogeneous
- **Clustering**
  - Multi-view network clustering
  - NoN clustering
- **Multi-network embedding**
  - MF-based embedding
  - Random walk-based embedding
  - **GNN-based embedding**

Deep Multiplex Graph Infomax (DMGI): Problem Definition

- Given:
  - $\{G_r = (N, E_r, \mathbf{X})\}, r = 1, \ldots, M$
  - N: node set, $\mathbf{X}$: the attribute matrix.
  - E_r: edge set for the relation r
- Find: node embedding for each node $\mathbf{z}_i \in \mathbf{Z} \in R^{n \times d}$.

$\mathbf{X}$

A	1	0	0
B	0	1	0
C	0	0	1
D	0	0	1
E	0	0	1



Note that in DMGI, attribute matrix is shared in all layers. Colors represent attributes.

[1]Park, Chanyoung, et al. "Unsupervised attributed multiplex network embedding." *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 34. No. 04. 2020.

DMGI: Intuition

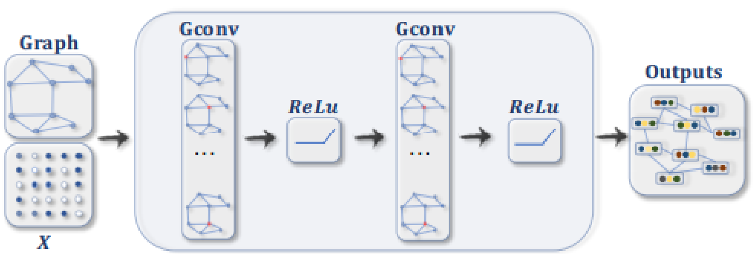
- Graph Convolutional Network (GCN) for attributed network

$$\mathbf{H}^{(r)} = g_r(\mathbf{X}, \mathbf{A}^{(r)} | \mathbf{W}^{(r)}) = \sigma\left(\hat{\mathbf{D}}_r^{-\frac{1}{2}} \hat{\mathbf{A}}^{(r)} \hat{\mathbf{D}}_r^{-\frac{1}{2}} \mathbf{X} \mathbf{W}^{(r)}\right)$$

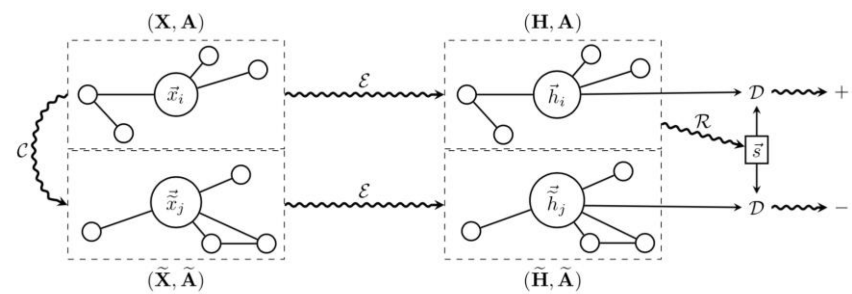
- Deep Graph Infomax (DGI) for modeling global properties of graph

$$\mathcal{L} = \sum_{v_i \in \mathcal{V}}^{n} \log \mathcal{D}\left(\mathbf{h}_i, \mathbf{s}\right) + \sum_{j=1}^{n} \log\left(1 - \mathcal{D}\left(\tilde{\mathbf{h}}_j, \mathbf{s}\right)\right)$$

$\mathbf{h}_i$: node embedding, $\mathbf{s}$: graph embedding and $\tilde{\mathbf{h}}_j$: negative node embedding.



GCN



DGI

- Combine GCN and DGI for attributed multi-network embedding

[1]Park, Chanyoung, et al. "Unsupervised attributed multiplex network embedding." *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 34. No. 04. 2020.
[2] Kipf, Thomas N., and Max Welling. "Semi-supervised classification with graph convolutional networks." *arXiv preprint arXiv:1609.02907* (2016).
[3] Veličković, Petar, et al. "Deep graph infomax." *arXiv preprint arXiv:1809.10341* (2018).

DMGI: Framework

- Single layer level embedding:
  - GCN: $\mathbf{H}^{(r)} = g_r(\mathbf{X}, \mathbf{A}^{(r)}|\mathbf{W}^{(r)}) = \sigma\left(\hat{\mathbf{D}}_r^{-\frac{1}{2}}\hat{\mathbf{A}}^{(r)}\hat{\mathbf{D}}_r^{-\frac{1}{2}}\mathbf{X}\mathbf{W}^{(r)}\right)$
  - Global structure embedding:

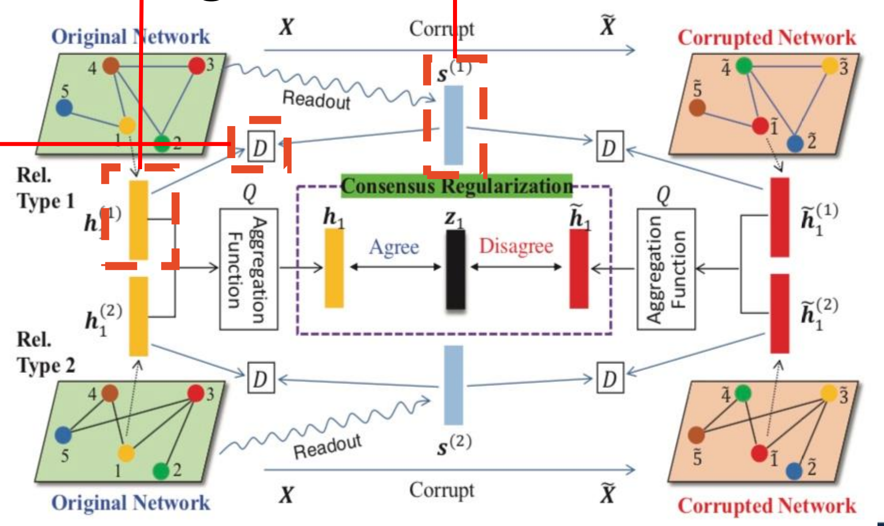
$$\mathbf{s}^{(r)} = \text{Readout}(\mathbf{H}^{(r)}) = \sigma\left(\frac{1}{n}\sum_{i=1}^{n}\mathbf{h}_i^{(r)}\right)$$

  - Relation-type specific node embedding:

$$\mathcal{L}^{(r)} = \sum_{v_i \in \mathcal{V}} \log \mathcal{D}\left(\mathbf{h}_i^{(r)}, \mathbf{s}^{(r)}\right) + \sum_{j=1} \log\left(1 - \mathcal{D}\left(\tilde{\mathbf{h}}_j^{(r)}, \mathbf{s}^{(r)}\right)\right)$$

  - Score function:

$$\mathcal{D}\left(\mathbf{h}_i^{(r)}, \mathbf{s}^{(r)}\right) = \sigma(\mathbf{h}_i^{(r)T}\mathbf{M}^{(r)}\mathbf{s}^{(r)})$$



[1]Park, Chanyoung, et al. "Unsupervised attributed multiplex network embedding." *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 34. No. 04. 2020.

DMGI: Framework (cont.)

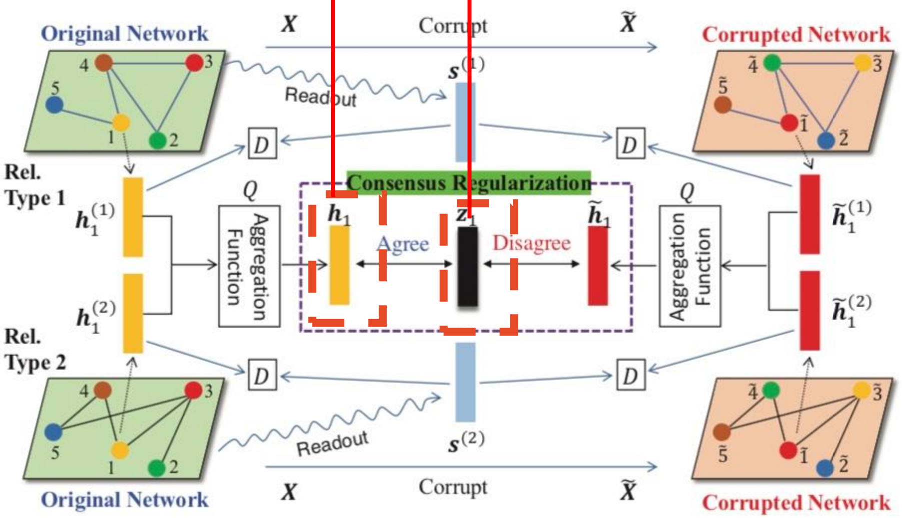
- Joint modeling and consensus regularization:
  - Consensus embedding Z:

$$\ell_{cs} = \left[\mathbf{Z} - \mathcal{Q}\left(\{\mathbf{H}^{(r)} \mid r \in \mathcal{R}\}\right)\right]^2 - \left[\mathbf{Z} - \mathcal{Q}\left(\{\tilde{\mathbf{H}}^{(r)} \mid r \in \mathcal{R}\}\right)\right]^2$$

$$\mathbf{H} = \mathcal{Q}\left(\{\mathbf{H}^{(r)} \mid r \in \mathcal{R}\}\right) = \frac{1}{|\mathcal{R}|} \sum_{r \in \mathcal{R}} \mathbf{H}^{(r)}$$

  - Unsupervised loss:

$$\mathcal{J} = \sum_{r \in \mathcal{R}} \mathcal{L}^{(r)} + \alpha \ell_{cs} + \beta ||\Theta||^2$$

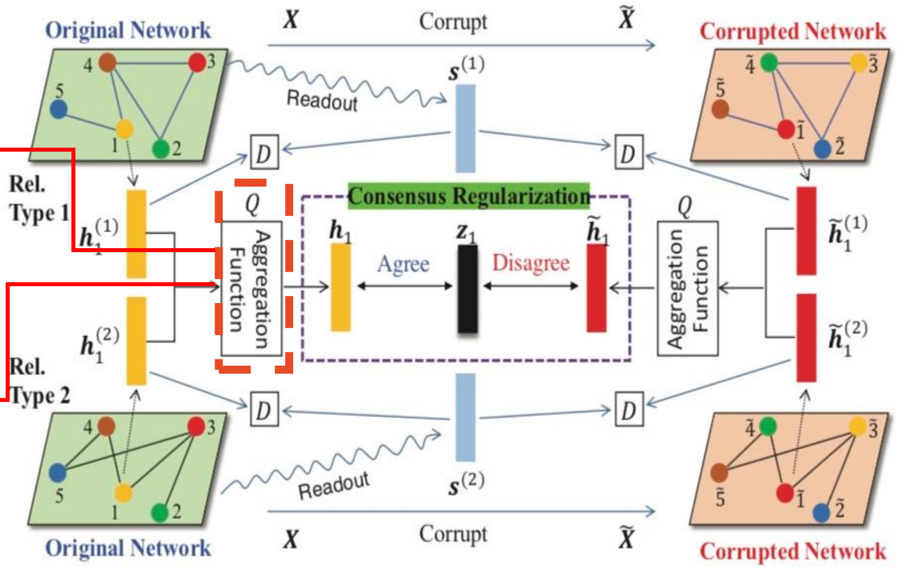


DMGI: Framework (cont.)

- Joint modeling and consensus regularization:
- Attention mechanism:

$$\mathbf{h}_i = \mathcal{Q}\left(\{\mathbf{h}^{(r)} \mid r \in \mathcal{R}\}\right) = \sum_{r\in\mathcal{R}} a_i^{(r)} \mathbf{h}^{(r)}$$

$$a_i^{(r)} = \frac{\exp\left(\mathbf{q}^{(r)} \cdot \mathbf{h}_i^{(r)}\right)}{\sum_{r'\in\mathcal{R}} \exp\left(\mathbf{q}^{(r')} \cdot \mathbf{h}_i^{r'}\right)}$$



where $\mathbf{q}^{(r)}$ is the feature vector of relation r.

- Semi-supervised learning:

$$\mathcal{J}_{\text{semi}} = \sum_{r\in\mathcal{R}} \mathcal{L}^{(r)} + \alpha\ell_{\text{cs}} + \beta||\Theta|| + \gamma\ell_{\text{sup}}$$

[1]Park, Chanyoung, et al. "Unsupervised attributed multiplex network embedding." *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 34. No. 04. 2020.

DMGI: Experimental Results

- Node clustering and similarity search
- Metrics: NMI for clustering and Sim@5 for similarity search

Method	ACM		IMDB		DBLP		Amazon	
	NMI	Sim@5	NMI	Sim@5	NMI	Sim@5	NMI	Sim@5
Deepwalk	0.310	0.710	0.117	0.490	0.348	0.629	0.083	0.726
node2vec	0.309	0.710	0.123	0.487	0.382	0.629	0.074	0.738
GCN/GAT	0.671	0.867	0.176	0.565	0.465	0.724	0.287	0.624
DGI	0.640	0.889	0.182	0.578	0.551	0.786	0.007	0.558
ANRL	0.515	0.814	0.163	0.527	0.332	0.720	0.166	0.763
CAN	0.504	0.836	0.074	0.544	0.323	0.792	0.001	0.537
DGCN	0.691	0.690	0.143	0.179	0.462	0.491	0.143	0.194
CMNA	0.498	0.363	0.152	0.069	0.420	0.511	0.070	0.435
MNE	0.545	0.791	0.013	0.482	0.136	0.711	0.001	0.395
mGCN	0.668	0.873	0.183	0.550	0.468	0.726	0.301	0.630
HAN	0.658	0.872	0.164	0.561	0.472	0.779	0.029	0.495
DMGI	0.687	0.898	**0.196**	**0.605**	0.409	0.766	**0.425**	0.816
$\text{DMGI}_{\text{attn}}$	**0.702**	**0.901**	0.185	0.586	**0.554**	**0.798**	0.412	**0.825**

- Observations:
  - DMGI outperforms all baselines;
  - the attention mechanism is useful.

[1]Park, Chanyoung, et al. "Unsupervised attributed multiplex network embedding." *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 34. No. 04. 2020.

DMGI: Experimental Results (cont.)

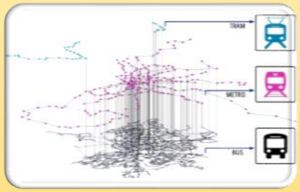
- Node classification on four real world datasets.
- Metrics: Macro-F1 and Micro-F1

	ACM		IMDB		DBLP		Amazon	
	MaF1	MiF1	MaF1	MiF1	MaF1	MiF1	MaF1	MiF1
Deepwalk	0.739	0.748	0.532	0.550	0.533	0.537	0.663	0.671
node2vec	0.741	0.749	0.533	0.550	0.543	0.547	0.662	0.669
GCN/GAT	0.869	0.870	0.603	0.611	0.734	0.717	0.646	0.649
DGI	0.881	0.881	0.598	0.606	0.723	0.720	0.403	0.418
ANRL	0.819	0.820	0.573	0.576	0.770	0.699	0.692	0.690
CAN	0.590	0.636	0.577	0.588	0.702	0.694	0.498	0.499
DGCN	0.888	0.888	0.582	0.592	0.707	0.698	0.478	0.509
CMNA	0.782	0.788	0.549	0.566	0.566	0.561	0.657	0.665
MNE	0.792	0.797	0.552	0.574	0.566	0.562	0.556	0.567
mGCN	0.858	0.860	0.623	0.630	0.725	0.713	0.660	0.661
HAN	0.878	0.879	0.599	0.607	0.716	0.708	0.501	0.509
DMGI	**0.898**	**0.898**	**0.648**	**0.648**	0.771	0.766	0.746	0.748
$\text{DMGI}_{\text{attn}}$	0.887	0.887	0.602	0.606	**0.778**	**0.770**	**0.758**	**0.758**

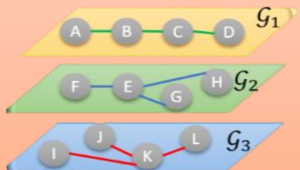
- DMGI improves classification performance.

[1]Park, Chanyoung, et al. "Unsupervised attributed multiplex network embedding." *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 34. No. 04. 2020.

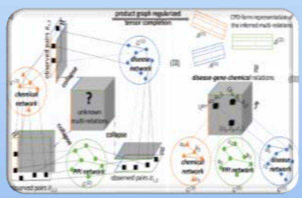
Roadmap



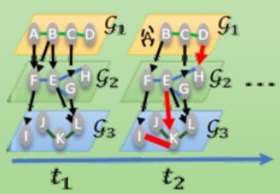
Introduction



Part I: Multi-network Data Models



Part II: Multi-network Mining Algorithms



Part III: Multi-network Future Directions

Overview of Part III

Future Directions

- Novel Multi-network Models
- Advanced Mining Algorithms
- Diverse Multi-network Applications

Novel Multi-network Models

Future Directions: Novel Multi-network Models

Temporal multi-network models

- Temporal multiplex network
- Temporal hypergraph
- Temporal inter-dependent network
- Temporal NoN

Combination of knowledge graph with multi-networks

- Multi-view knowledge graph
- High-arity knowledge graph

Combination of traditional multi-network models

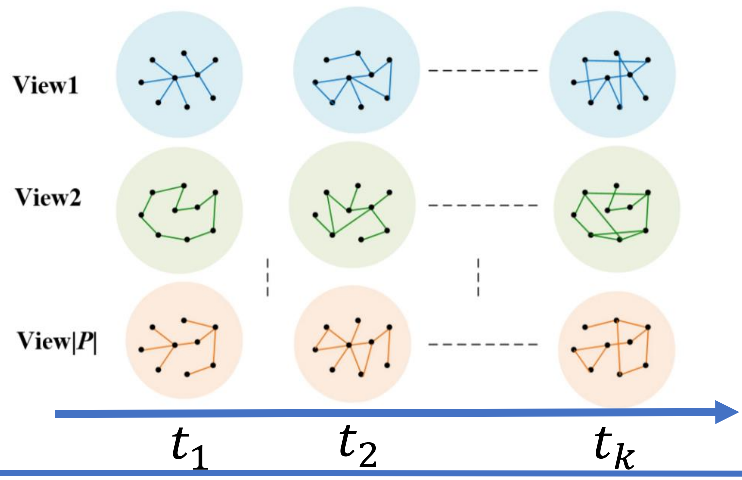
- Network of ‘X’
- Multi-layered hypergraph

Generalized multi-network models

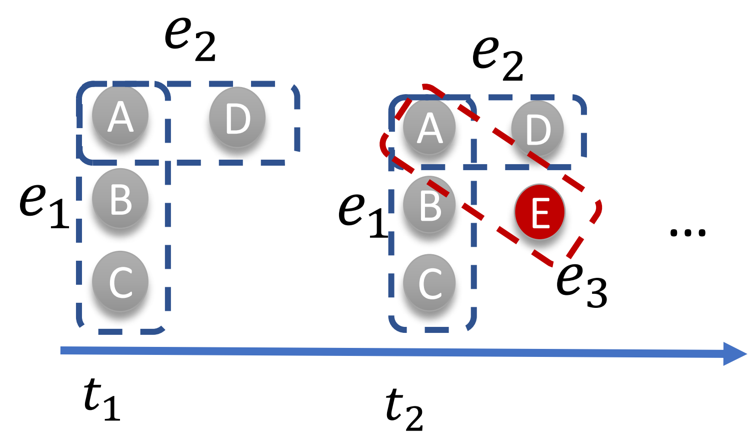
- Generalized NoN
- Attributed multi-layered network

Temporal Multi-network Models

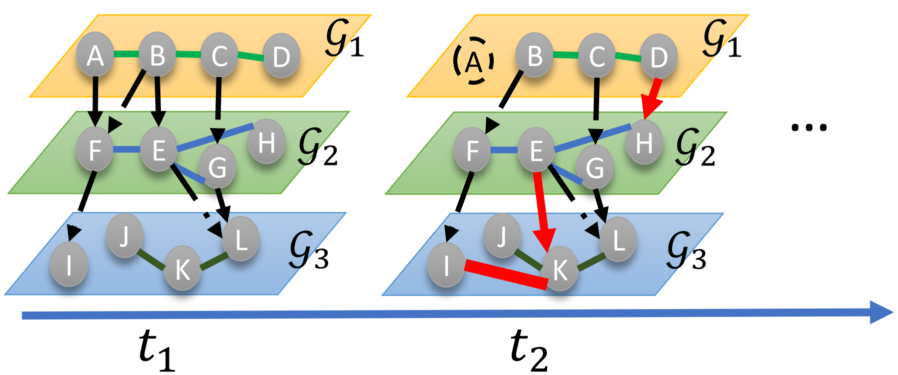
Temporal multi-view networks:



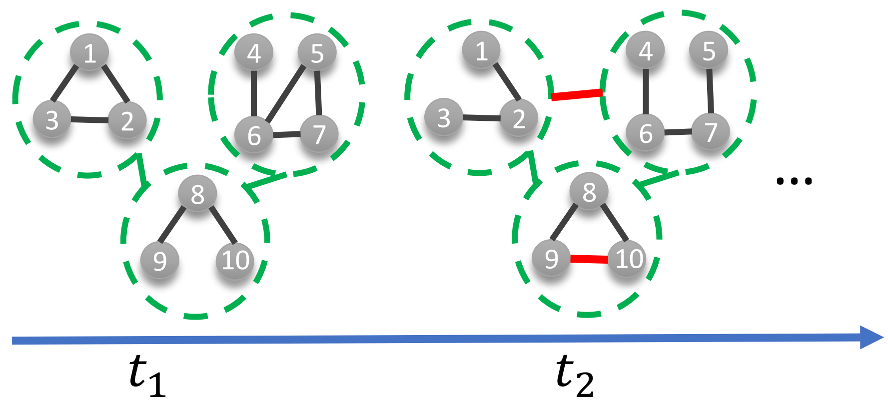
Temporal hypergraphs:



Temporal inter-dependent networks:

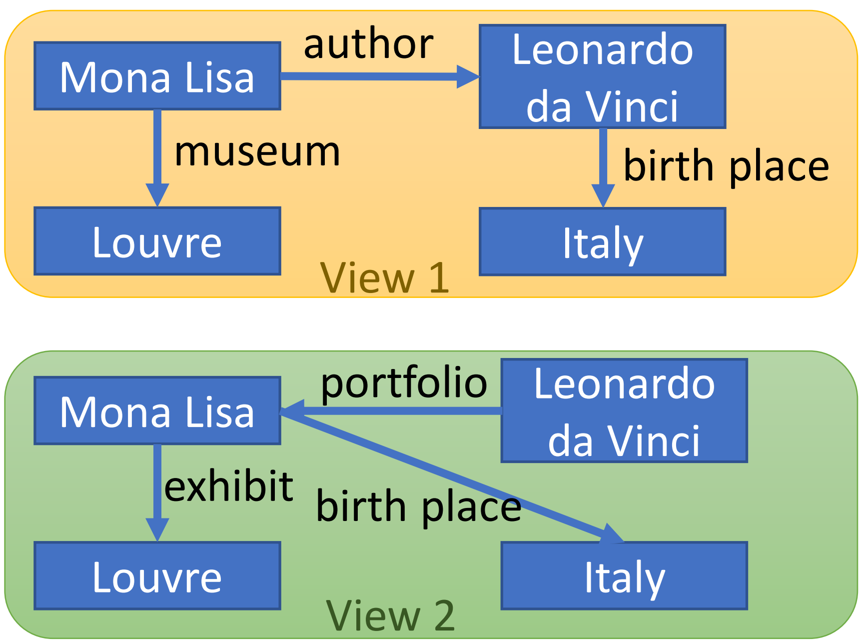


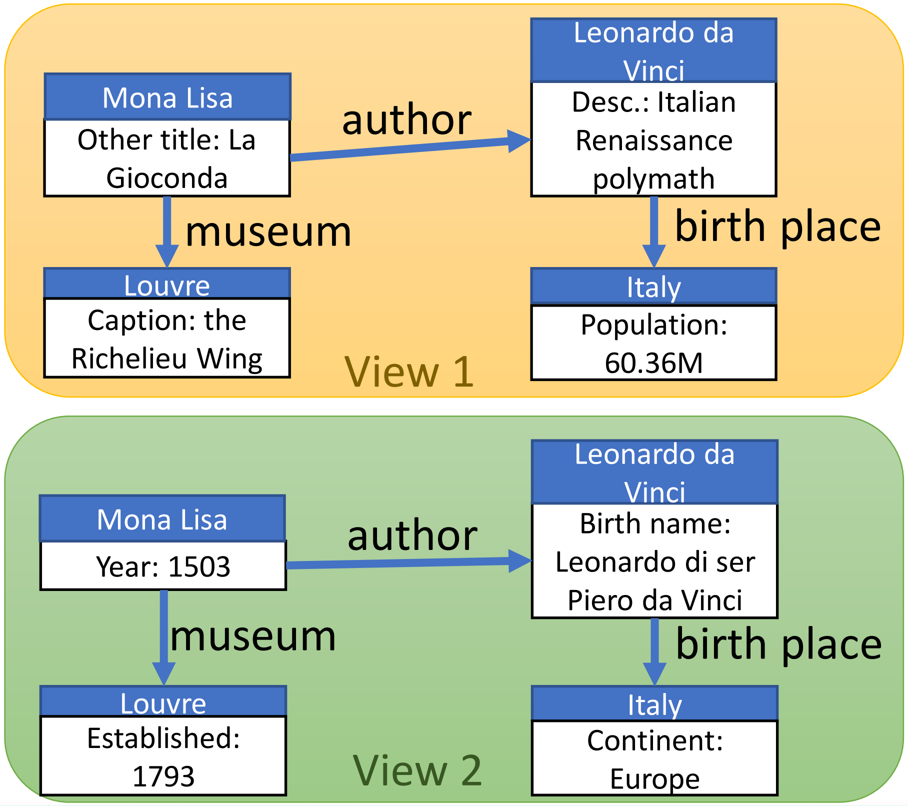
Temporal NoNs:



Multi-view Knowledge Graph

- Definition: $\mathcal{G}_i = (\mathcal{V}_i, \mathcal{E}_i), i = 1, \ldots, K$
  - Optional: features F_i
  - $\mathcal{V}_i$: entity set of the i-th view
  - $\mathcal{E}_i$: (head, relation, tail) triple set of the i-th view
  - E.g.,

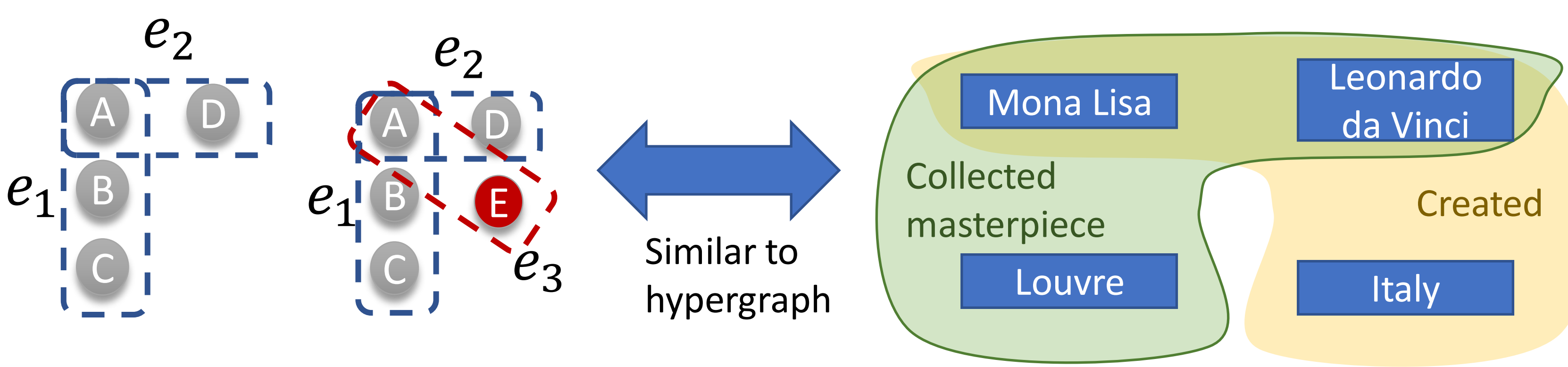




[1] Zhang, Qingheng, et al. "Multi-view knowledge graph embedding for entity alignment." *arXiv preprint arXiv:1906.02390* (2019).

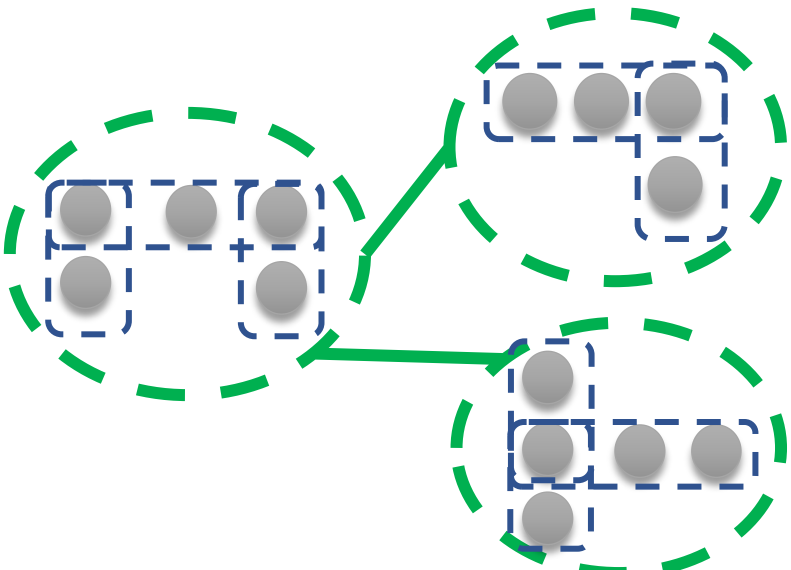
High-arity Knowledge Graph

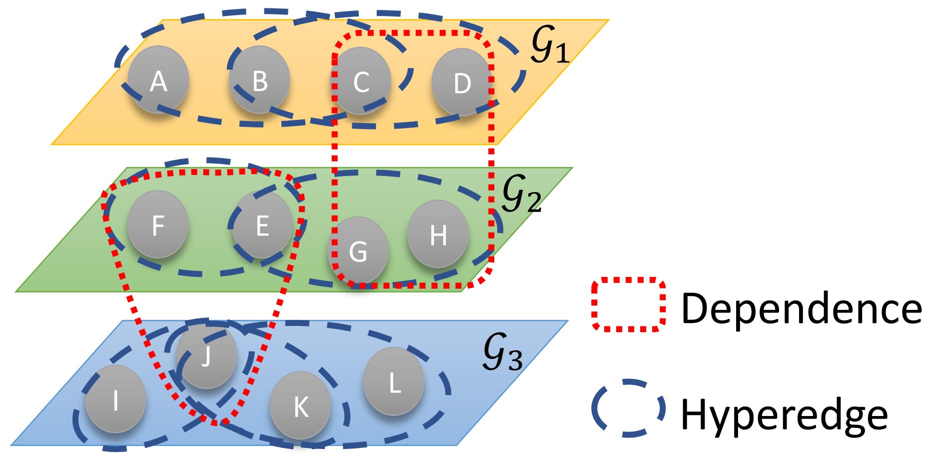
- **Motivations**:
  - Existing knowledge graph: 2-arity
  - E.g., author(Leonardo da Vinci, Mona Lisa)
  - High-arity knowledge graph: higher-arity (high-order relation)
  - E.g., studied(Hawking, PhD, Princeton)
- **Future directions**:
  - How to construct high-arity knowledge graph?
  - How to mine on high-arity knowledge graph?



Combination of Traditional Multi-network Models

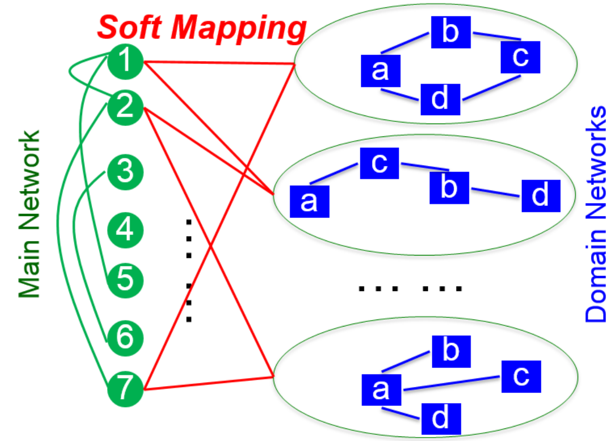
- Network of X:
  - X: regular networks (covered), hypergraphs, multi-view networks, etc.
- Multi-layered hypergraph:
  - Generalization of multi-layered graph





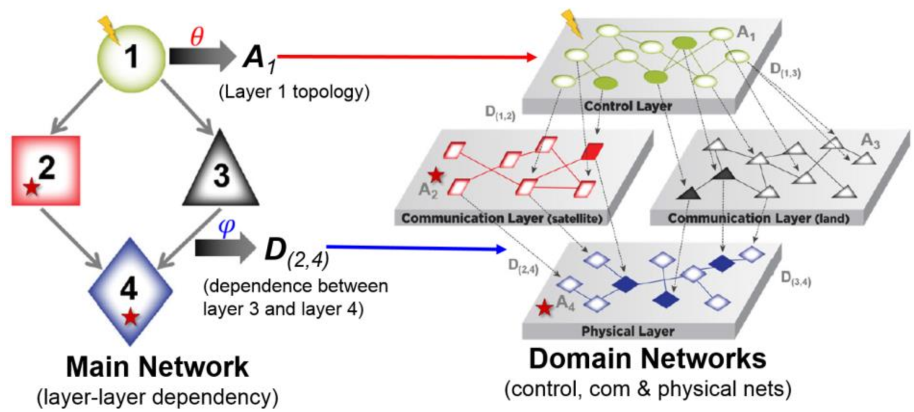
Generalized Network of Networks

- G1: Multi-layered Hierarchical NoN



- G2: Soft Mapping Function θ
  - *1-to-many or many-to-many*
- G3: Map Edges to Networks φ

  - *G*: Main Network
  - ***A***: Domain Networks
  - ***D***: Cross-Layer Dep'
  - θ: Function $V_G \rightarrow \boldsymbol{A}$
  - φ: Function $E_G \rightarrow \boldsymbol{D}$

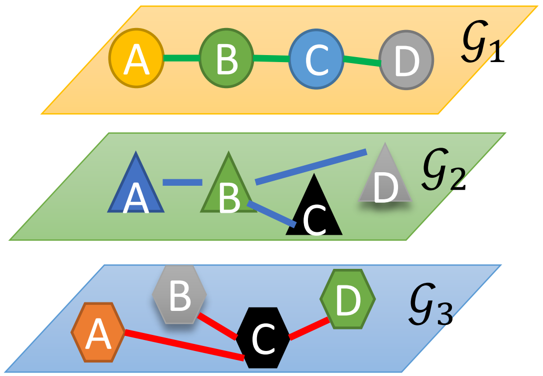


[1] C. Chen, J. He, N. Bliss and H. Tong: "On the Connectivity of Multi-layered Networks: Models, Measures and Optimal Control" ICDM 2015.
[2] C. Chen, J. He, N. Bliss and H. Tong: "Towards Optimal Connectivity on Multi-layered Networks". IEEE Trans. Knowl. Data Eng., 29(10): 2332-2346 (2017)

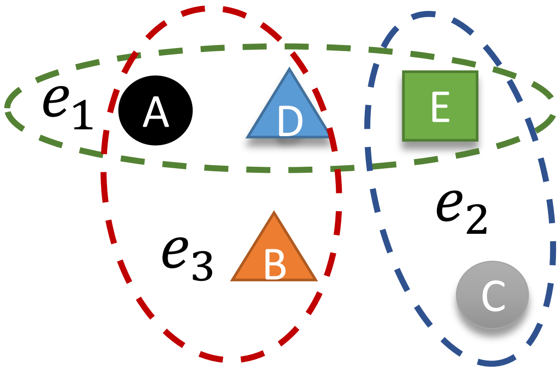
Attributed Multi-layered Network

- Shapes: node types; colors: attributes

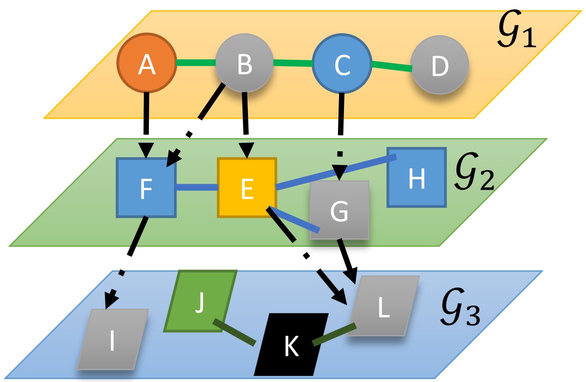
Attributed multi-view networks:



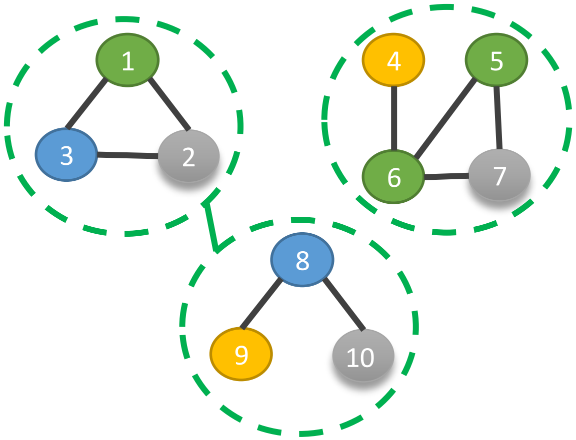
Attributed hypergraphs:



Attributed inter-dependent networks:



Attributed NoNs:



Advanced Mining Algorithms

Future Directions: Advanced Mining Algorithms

Active multi-network learning	Adversarial multi-network learning	Temporal multi-network learning	Traditional mining tasks on novel multi-network models
• Active multi-network alignment/association • Active multi-network embedding	• Adversarial multi-network embedding • Adversarial multi-network alignment/association	• Temporal multi-network embedding	• Clustering/ranking on generalized NoN

Active Multi-network Alignment/Association

- **Motivation**: human interaction with multi-network models
  - Find the most informative node (set) for groundtruth query
  - Maximize alignment/association accuracy on the rest of nodes
- **Challenges**:
  - How to define and quantify node (set) information for query?
  - How to identify informative node (set)?
- **Future directions**:
  - Matching distribution-based certainty measurement
  - Network derivative/influence function-based measurement

[1] Malmi, Eric, Aristides Gionis, and Evimaria Terzi. "Active network alignment: a matching-based approach." *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*. 2017.
[2] Qinghai Zhou , Liangyue Li , Xintao Wu, Nan Cao, Lei Ying, Hanghang Tong. “Attent: Active Attributed Network Alignment.” In Proceedings of the Web Conference 2021 (WWW ’21)

Active Multi-network Embedding

- **Motivation**:
  - Select nodes for query to optimize the embedding model
- **Challenges**:
  - How to select the most informative nodes for specific tasks?
  - How to handle multi-network structure (attributes)?
- **Future directions**:
  - Combine active learning with multi-network GNN methods
  - Multi-armed bandit for active node selection strategies

[1] Madhawa, Kaushalya, and Tsuyoshi Murata. "Active Learning for Node Classification: An Evaluation." *Entropy* 22.10 (2020): 1164.
[2] Madhawa, Kaushalya, and Tsuyoshi Murata. "A multi-armed bandit approach for exploring partially observed networks." Applied Network Science 4.1 (2019): 1-18.

Adversarial Multi-network Alignment/Association

- **Motivations**:
  - Existing adversarial attacks on network alignment are based on derivative-based importance score
  - But no work exits on adversarial defense
- **Challenge**:
  - Compared to adversarial attack/defense in single network, multiple networks may further complicate the defense process.
- **Future direction**:
  - Adversarial training for multi-network alignment/association (w/ GNN)

Adversarial Multi-network Embedding

- **Motivations**:
  - Improve the robustness of embedding on multi-networks
  - Generalized adversarial network embedding to multi-networks
- **Challenges**:
  - Multi-network structure complicates the embedding generation and discrimination
- **Future directions**:
  - Combine multi-network GNN model w/ adversarial training

[1] Dai, Quanyu, et al. "Adversarial network embedding." *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 32. No. 1. 2018.

Temporal Multi-network Embedding

- **Motivations**:
  - Real-world data is often dynamic
  - Direct application of static method is costly
- **Challenges**:
  - How to leverage dynamics (e.g., representation smoothness)
  - How to improve efficiency w/o using static method
- **Future directions**:
  - Matrix approximation to avoid unnecessary re-computations
  - Dynamic multi-network embedding-based methods

Clustering/Ranking on Generalized NoN

- **Motivations**:
  - Generalized NoN is more complex than NoN model
  - Ranking/clustering problems are also more complex
- **Challenges**:
  - How to construct real-world data as generalized NoN
  - How to generalized existing ranking/clustering methods
- **Future directions**:
  - Hierarchical label propagation-based optimization method
  - Novel random walk-based strategy for personalized ranking

Diverse Multi-network Applications

Future Directions: Diverse Multi-network Applications

Multi-network for knowledge graphs

- Multiplex networks for knowledge graph completion

Multi-network for bio-informatics

- Multi-view network for neuroimage analysis

Multi-network for recommender system

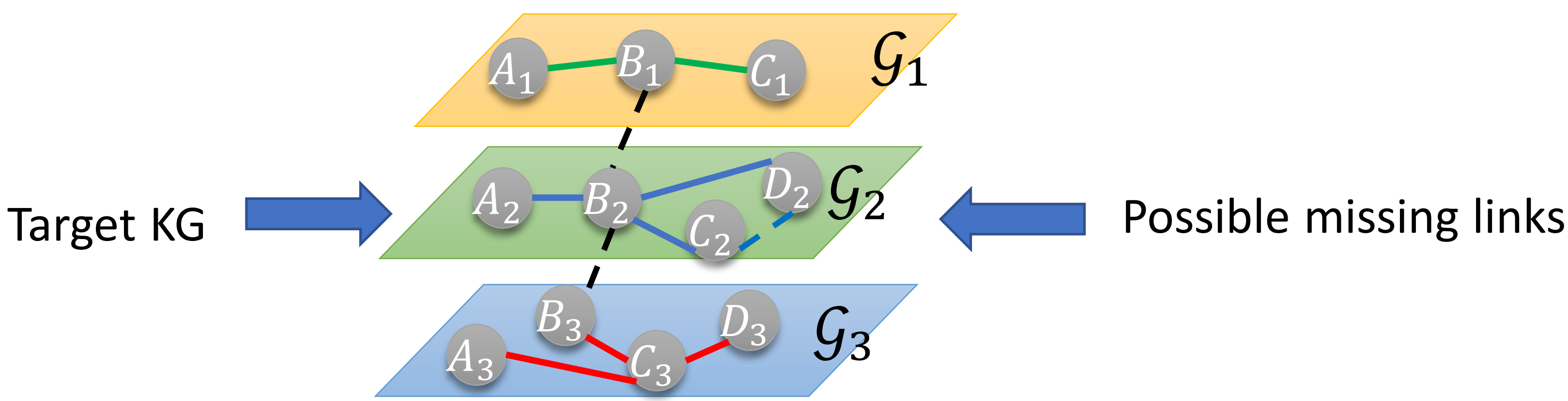
- Hypergraphs for recommender system
- Multi-layered network for cross-domain recommendation

Multi-network for graph classification

- Multiple structure-view learning for graph classification

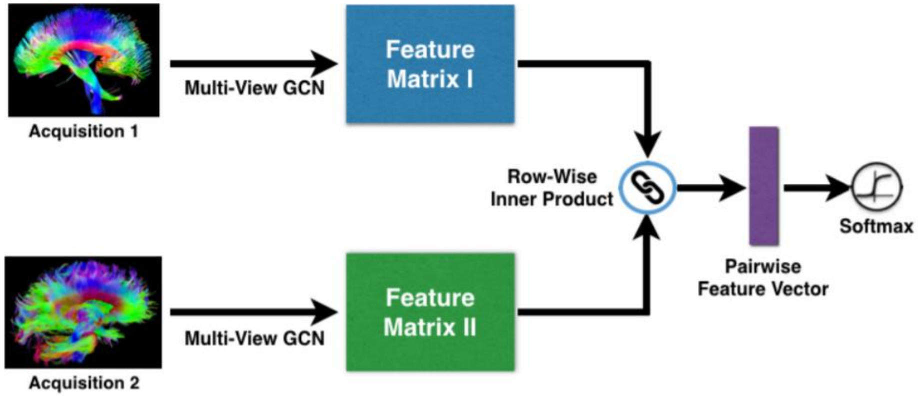
Multiplex Networks for Knowledge Graph Completion (KGC)

- **Motivation**: Mining missing triples from knowledge graph
- **Challenges**:
  - KGs are often sparse (many missing links)
  - How to leverage complementary knowledge from different sources
- **Future Directions**:
  - Construct multiplex knowledge graph for KGs of different sources
  - Multi-task learning: jointly learn the KGC with KG entity alignment



Multi-view network for neuroimage analysis

- **Motivations**:
  - Neuroimaging: important information source for neurodegenerative disease
  - Assist clinical diagnose with multi-network mining methods
- **Challenges**:
  - Neuroimage is often multi-view and heterogeneous
- **Future directions**:
  - Apply multi-view GNN-based model on the neuroimage classification
  - E.g., On Parkinson's Progression Markers Initiative (PPMI) data:

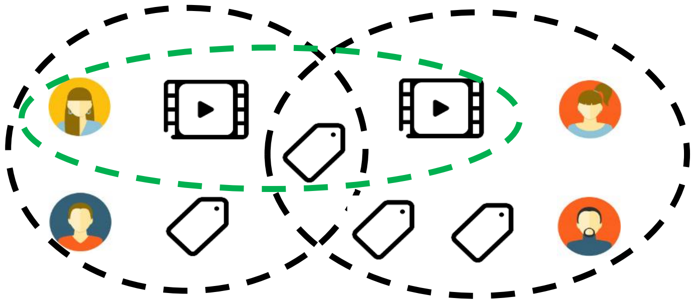


[1] Zhang, Xi, et al. "Multi-view graph convolutional network and its applications on neuroimage analysis for parkinson's disease." *AMIA Annual Symposium Proceedings*. Vol. 2018. American Medical Informatics Association, 2018.

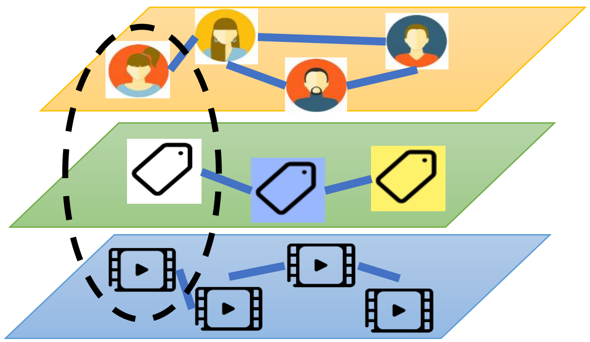
Hypergraphs for Recommender System

- **Motivations**:
  - Hypergraphs for bundle/high-order recommendation
  - Focus: recommend a set of items
- **Challenges**:
  - How to construct the hypergraphs for bundles
  - How to incorporate the high-order relation of hypergraphs
- **Future directions**:

Bundles as heterogeneous hyperedges

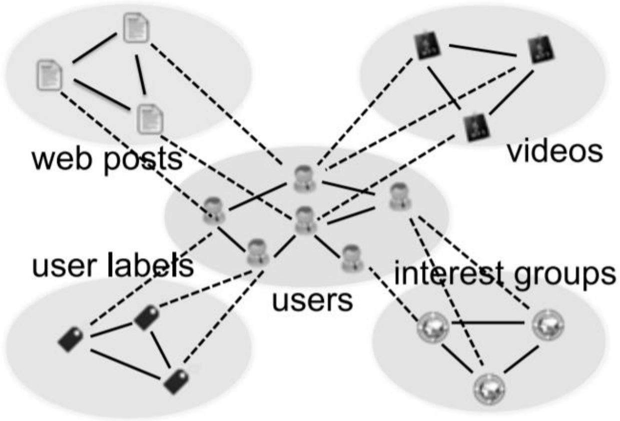


Bundles as multi-network node set



Multi-layered Network For Cross-domain Recommendation

- **Motivations**:
  - How to transfer knowledge across domains in recommendation
  - How to recommend items from different domains to users
- **Challenges**:
  - How to handle cold start issue from certain domains of items?
  - How to recommend bundles of items from different domains?
- **Future directions**:
  - Multi-domain data -> multi-layered networks
  - Random walk-based embedding
  - GNN-based embedding



[1] Jiang, Meng, et al. "Social recommendation with cross-domain transferable knowledge." *IEEE transactions on knowledge and data engineering* 27.11 (2015): 3084-3097.

Multi-networks For Graph Classification

- **Motivations**:
  - Traditional graph classification: single graph
  - Can not handle complicated objects containing complex structures
- **Challenge**:
  - How to classify complex objects from multiple structure views?
- **Future directions**:
  - Complex objects as multi-networks (bag of networks)
  - Collectively leverage substructures and features from multi-networks

[1] Wu, Jia, et al. "Multiple structure-view learning for graph classification." *IEEE transactions on neural networks and learning systems* 29.7 (2017): 3236-3251.

Summary

- Background and Motivation:
  - Multi-networks: multi-sourced, complex network data models
  - Multi-network mining: challenging, important graph mining tools
- Multi-network models:
  - Five types of representative multi-network data models
  - A unified view of all the introduced multi-network models
- Multi-network mining algorithms:
  - Algorithms for traditional/novel mining tasks and applications
- Future directions
  - Novel multi-network models
  - Advanced multi-network mining algorithms
  - Diverse multi-network applications

References: Related Tutorials

- Hypergraph Learning: Methods, Tools and Applications in Medical Image Analysis (MICCAI 2019). http://gaoyue.org/en/more/index.htm
- Si Zhang and Hanghang Tong. 2020. Network Alignment: Recent Advances and Future Directions. Proceedings of the 29th ACM International Conference on Information & Knowledge Management. Association for Computing Machinery, New York, NY, USA.
- Jiawei Han, Yizhou Sun, Xifeng Yan, and PhilipS Yu.2010.Mining heterogeneous information networks. In Tutorial at the 2010 ACM SIGKDD Conf. on Knowledge Discovery and Data Mining (KDD'10), Washington, DC.
- Haochen Chen, Bryan Perozzi, Rami Al-Rfou, and Steven Skiena. "A tutorial on network embeddings." arXiv preprint arXiv:1808.02590 (2018).

References: Data and Code

- FINAL: https://sizhang2.web.illinois.edu/resources/FINAL-KDD16.zip
- FASCINATE: https://github.com/chenannie45/FASCINATE
- MANE: http://www.ece.virginia.edu/~jl6qk/code/MANE.zip
- DMGI: https://github.com/pcy1302/DMGI
- CMM: https://github.com/muhanzhang/HyperLinkPrediction
- DHNE: https://github.com/tadpole/DHNE
- CrossRank: https://github.com/nijingchao/NoNCrossRank
- NoNClus: https://github.com/nijingchao/NoNClus
- SyTE: https://github.com/boxindu/SYTE

References

- Zhang, Si, and Hanghang Tong. "Network Alignment: Recent Advances and Future Directions." Proceedings of the 29th ACM International Conference on Information & Knowledge Management. 2020.
- Chen Chen, Hanghang Tong, Lei Xie, Lei Ying, and Qing He. 2017. Cross-Dependency Inference in Multi-Layered Networks: A Collaborative Filtering Perspective. ACM Trans. Knowl. Discov. Data 11, 4, Article 42 (August 2017), 26 pages. DOI:https://doi.org/10.1145/3056562
- Klamt, Steffen, Utz-Uwe Haus, and Fabian Theis. "Hypergraphs and cellular networks." *PLoS computational biology* 5.5 (2009): e1000385.
- Tu, Ke, et al. "Structural deep embedding for hyper-networks." *Thirty-Second AAAI Conference on Artificial Intelligence*. 2018.
- Jiang, Jianwen, et al. "Dynamic Hypergraph Neural Networks." *IJCAI*. 2019.
- Li, Dong, et al. "Link prediction in social networks based on hypergraph." *Proceedings of the 22nd International Conference on World Wide Web*. 2013.
- H. Tong, C. Faloutsos, J.-Y. Pan: Fast Random Walk with Restart and Its Applications. ICDM 2006
- Jingchao Ni, Hanghang Tong, Wei Fan, and Xiang Zhang. 2015. Flexible and Robust Multi-Network Clustering. Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. Association for Computing Machinery, New York, NY, USA, 835–844. DOI:https://doi.org/10.1145/2783258.2783262
- Z. Xu, S. Zhang, Y. Xia, L. Xiong and H. Tong, "Ranking on Network of Heterogeneous Information Networks," 2020 IEEE International Conference on Big Data (Big Data), 2020, pp. 848-857, doi: 10.1109/BigData50022.2020.9378121.
- Dai, Quanyu, et al. "Adversarial network embedding." Proceedings of the AAAI Conference on Artificial Intelligence. Vol. 32. No. 1. 2018.

References

- Kleinberg, J. M. (1999). Authoritative sources in a hyperlinked environment. Journal of the ACM (JACM), 46(5), 604-632.
- Abhishek Kumar, Piyush Rai, and Hal Daumé. 2011. Co-regularized multi-view spectral clustering. In Proceedings of the 24th International Conference on Neural Information Processing Systems (NIPS'11). Curran Associates Inc., Red Hook, NY, USA, 1413–1421.
- Shi, Y., Chan, P. W., Zhuang, H., Gui, H., & Han, J.. Prep: Path-based relevance from a probabilistic perspective in heterogeneous information networks. KDD 2017.
- Bryan Perozzi, Rami Al-Rfou, and Steven Skiena. 2014. DeepWalk: online learning of social representations. In Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining.
- Tang, Jian, et al. "Line: Large-scale information network embedding." *Proceedings of the 24th international conference on world wide web*. 2015.
- Tang, Jian, et al. "Visualizing large-scale and high-dimensional data." *Proceedings of the 25th international conference on world wide web*. Jiang, Jianwen, et al. "Dynamic Hypergraph Neural Networks." *IJCAI*. 2019.
- Mark EJ Newman and Michelle Girvan. Finding and evaluating community structure in networks. Physical review E 69.2 (2004): 026113.
- Li, Jundong, C. Chen, Hanghang Tong and H. Liu. “Multi-Layered Network Embedding.” *SDM* (2018).
- Zhang, Hongming, et al. "Scalable Multiplex Network Embedding." *IJCAI*. Vol. 18. 2018.
- Ni, Jingchao, et al. "Inside the atoms: ranking on a network of networks." Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining. 2014.

References

- Park, Chanyoung, et al. "Unsupervised attributed multiplex network embedding." *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 34. No. 04. 2020.
- Kipf, Thomas N., and Max Welling. "Semi-supervised classification with graph convolutional networks." *arXiv preprint arXiv:1609.02907* (2016).
- Veličković, Petar, et al. "Deep graph infomax." *arXiv preprint arXiv:1809.10341* (2018).
- Chen, Chuan, et al. "A semisupervised classification approach for multidomain networks with domain selection." *IEEE transactions on neural networks and learning systems* 30.1 (2018): 269-283.
- M. Karasuyama and H. Mamitsuka, "Multiple graph label propagation by sparse integration," IEEE Trans. Neural Netw. Learn. Syst., 2013.
- W. Cheng, Z. Guo, X. Zhang, and W. Wang,"CGC: A flexible and robust approach to integrating co-regularized multi-domain graph for clustering," Trans. Knowl. Discovery Data, vol. 10, no. 4, 2015, Art. no. 46.
- Li, Sheng, et al. "Multi-view graph learning with adaptive label propagation." *2017 IEEE International Conference on Big Data (Big Data)*. IEEE, 2017. Li, Sheng, et al. "Multi-view graph learning with adaptive label propagation." 2017 IEEE International Conference on Big Data (Big Data). IEEE, 2017.
- Khan, Muhammad Raza, and Joshua E. Blumenstock. "Multi-gcn: Graph convolutional networks for multi-view networks, with applications to global poverty." *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 33. No. 01. 2019.
- Yadati, Naganand, et al. "HyperGCN: A new method of training graph convolutional networks on hypergraphs." *arXiv preprint arXiv:1809.02589* (2018).
- Zhang, Muhan, Zhicheng Cui, Shali Jiang, and Yixin Chen. "Beyond link prediction: Predicting hyperlinks in adjacency space." In Thirty-Second AAAI Conference on Artificial Intelligence. 2018.
- Kivelä, Mikko, et al. "Multilayer networks." Journal of complex networks 2.3 (2014): 203-271.

References

- Zhang, Si, and Hanghang Tong. "Final: Fast attributed network alignment." *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 2016.
- Liu, Hanxiao, and Yiming Yang. "Cross-graph learning of multi-relational associations." *International Conference on Machine Learning*. PMLR, 2016.
- Li, Zhuliu, et al. "Learning a Low-Rank Tensor of Pharmacogenomic Multi-relations from Biomedical Networks." *2019 IEEE International Conference on Data Mining (ICDM)*. IEEE, 2019
- Rohit Singh, Jinbo Xu, and Bonnie Berger. 2008. Global alignment of multiple protein interaction networks with application to functional orthology detection. Proceedings of the National Academy of Sciences 105, 35 (2008), 12763–12768
- Du, Boxin, Lihui Liu, and Hanghang Tong. "Sylvester Tensor Equation for Multi-way Association". SIGKDD (2021)
- Zhang, Zhenghao, Jianbin Huang, and Qinglin Tan. "Multi-view Dynamic Heterogeneous Information Network Embedding." arXiv preprint arXiv:2011.06346 (2020).
- C. Chen, J. He, N. Bliss and H. Tong: "On the Connectivity of Multi-layered Networks: Models, Measures and Optimal Control" ICDM 2015.
- C. Chen, J. He, N. Bliss and H. Tong: "Towards Optimal Connectivity on Multi-layered Networks". IEEE Trans. Knowl. Data Eng., 29(10): 2332-2346 (2017)
- Malmi, Eric, Aristides Gionis, and Evimaria Terzi. "Active network alignment: a matching-based approach." Proceedings of the 2017 ACM on Conference on Information and Knowledge Management. 2017.
- Qinghai Zhou , Liangyue Li , Xintao Wu, Nan Cao, Lei Ying, Hanghang Tong. "Attent: Active Attributed Network Alignment." In Proceedings of the Web Conference 2021 (WWW '21)

References

- Zhang, Xi, et al. "Multi-view graph convolutional network and its applications on neuroimage analysis for parkinson's disease." AMIA Annual Symposium Proceedings. Vol. 2018. American Medical Informatics Association, 2018.
- Wu, Jia, et al. "Multiple structure-view learning for graph classification." IEEE transactions on neural networks and learning systems 29.7 (2017): 3236-3251.
- Mikolov, Tomas, et al. "Efficient estimation of word representations in vector space." arXiv preprint arXiv:1301.3781 (2013).