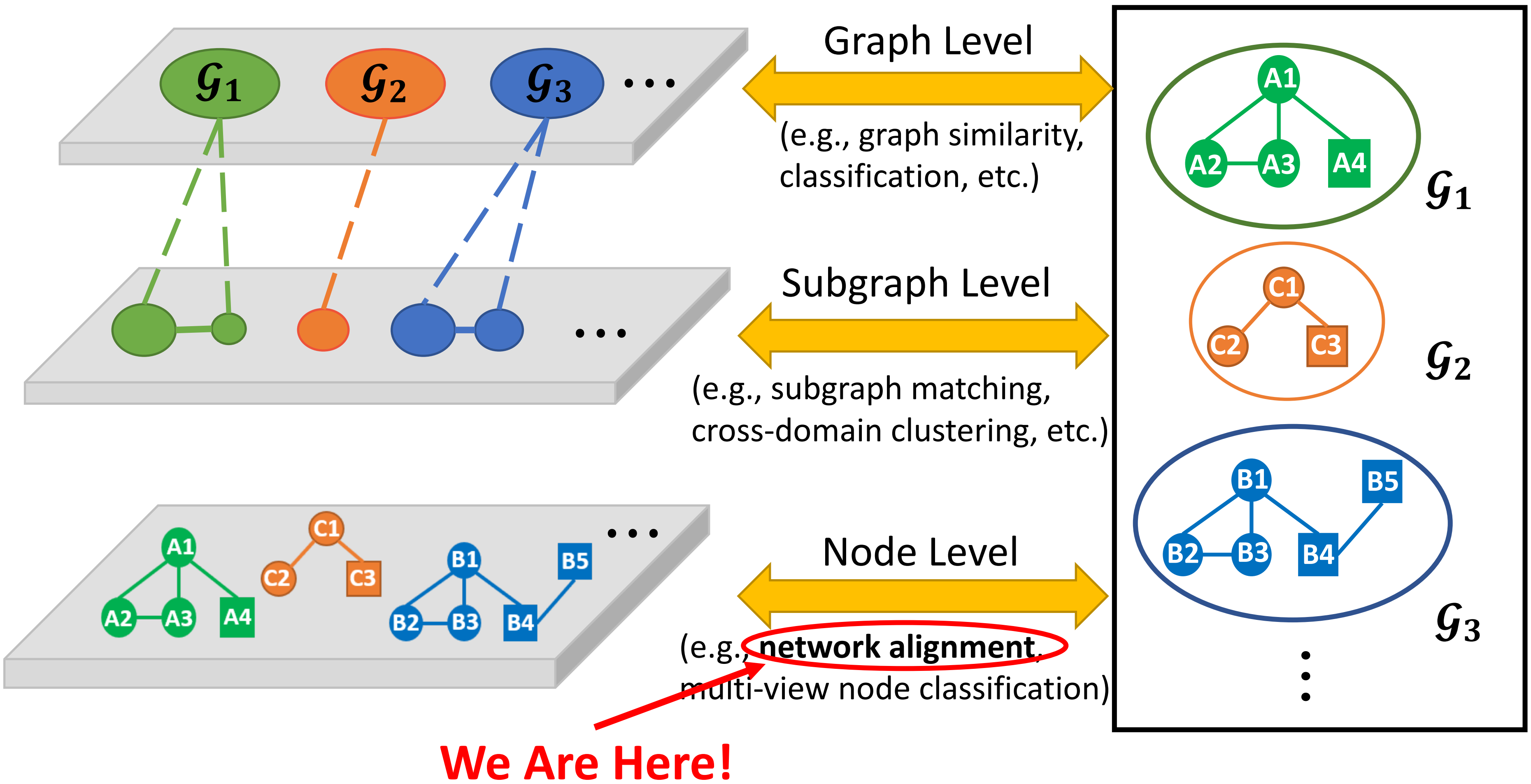
Network Alignment: Recent Advances and Future Directions

Si Zhang
(UIUC)

Hanghang Tong
(UIUC)

ILLINOIS

Mining on Multiple Networks



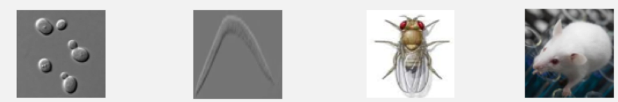
Multiple Networks Are Prevalent

Online Social Networks

Transaction Networks

CHASE
Bank of America
PayPal
venmo
Alipay

PPI Networks



yeast elegans fly mouse

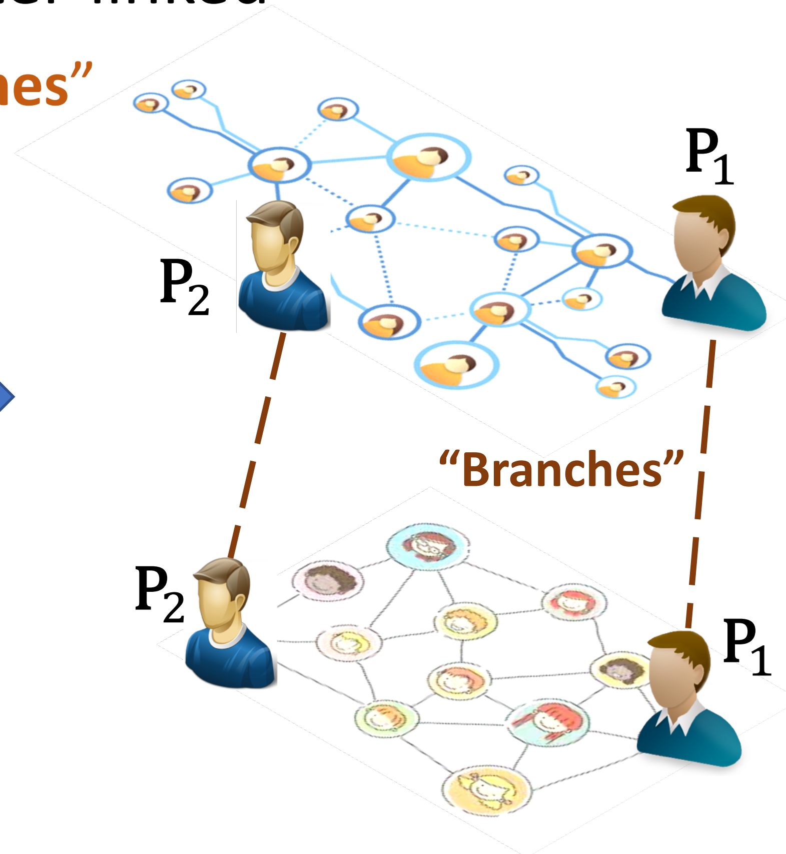
Knowledge Graphs

DBpedia
Freebase
yago select knowledge

Multiple Networks: Examples

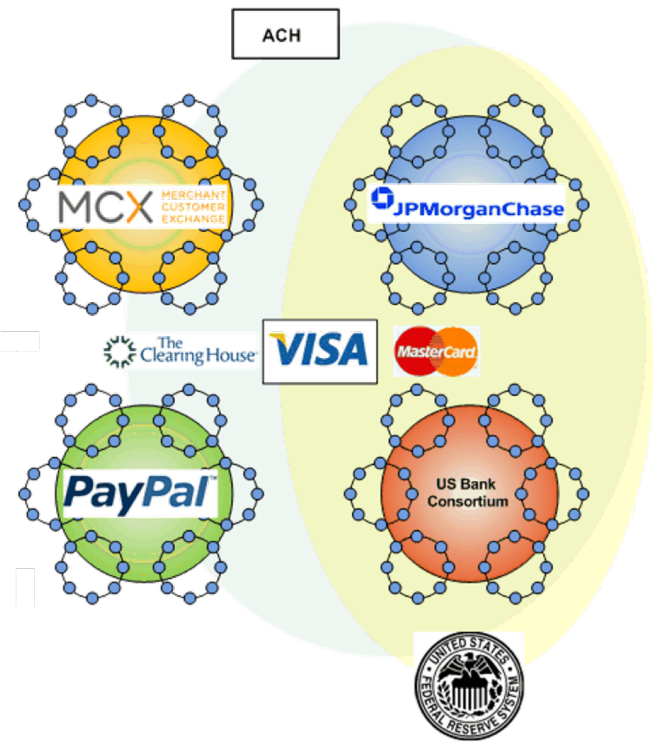
- Multiple social networks are inter-linked

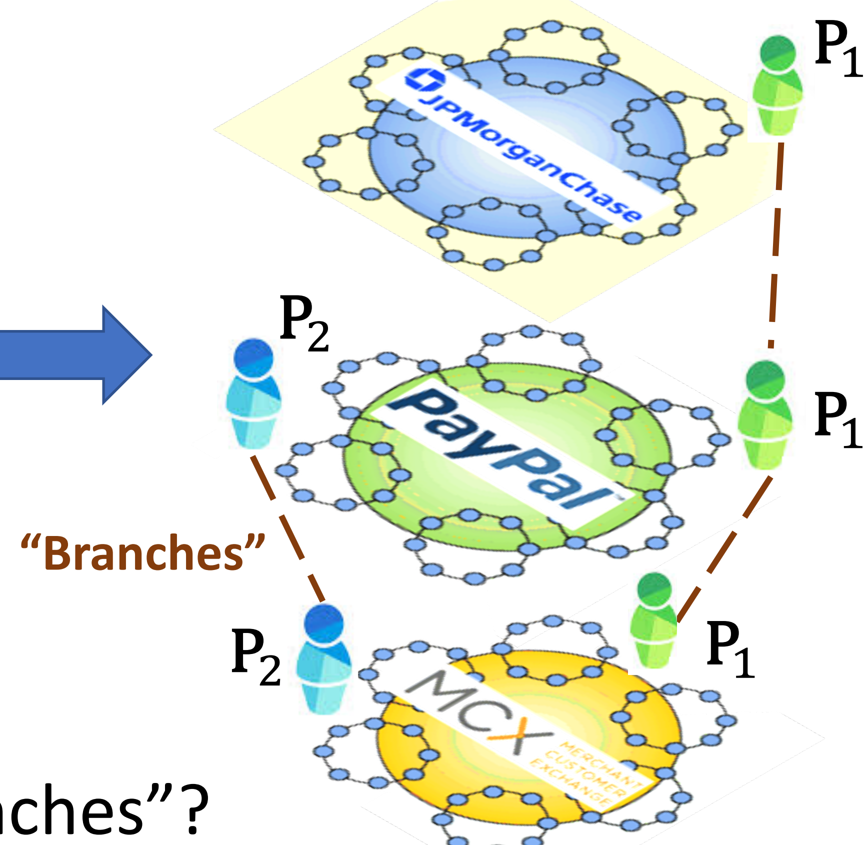
Linked by **"branches"**



Multiple Networks: Examples

- Multiple transaction networks are inter-linked

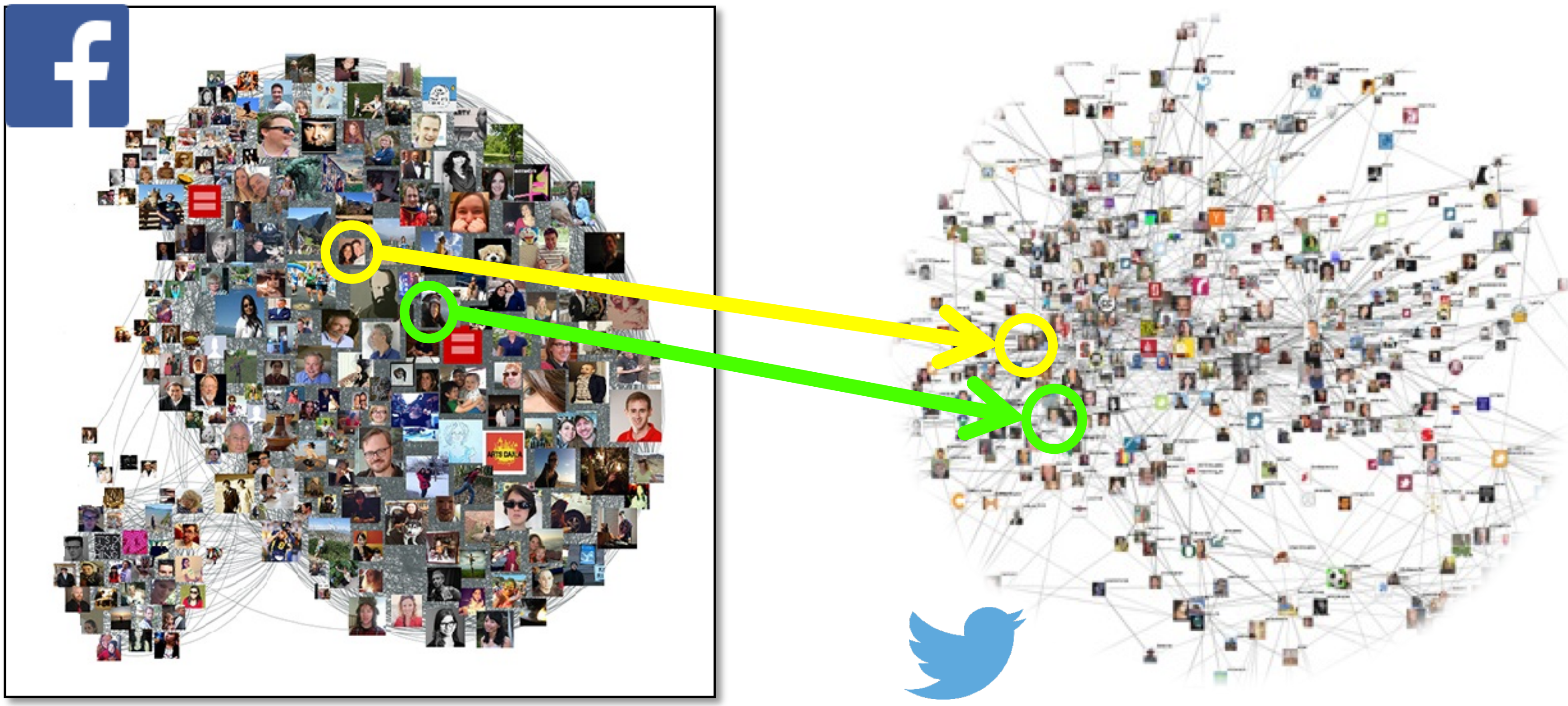




- **Q:** How to find those "branches"?

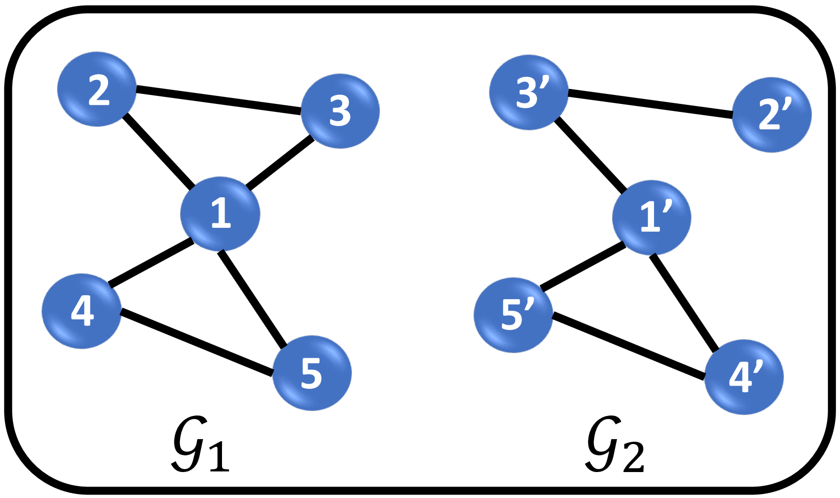
What Is Network Alignment?

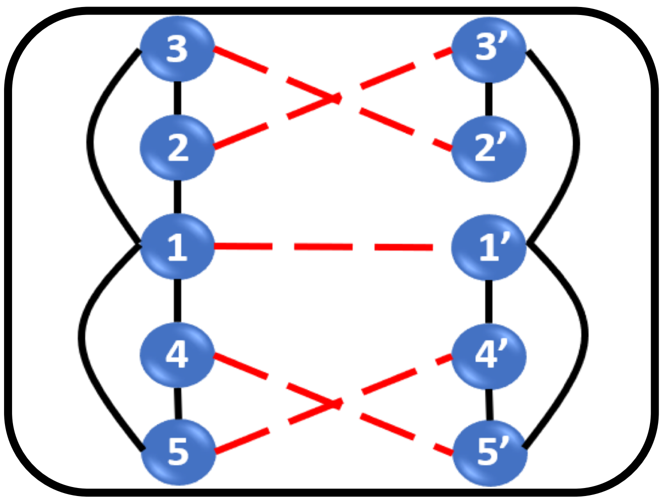
- Find node correspondence across multiple networks



Network Alignment: Prob. Def.

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

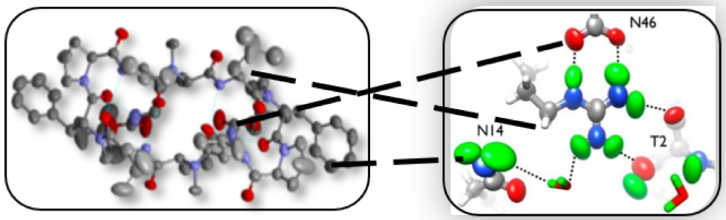




Why Do We Care?

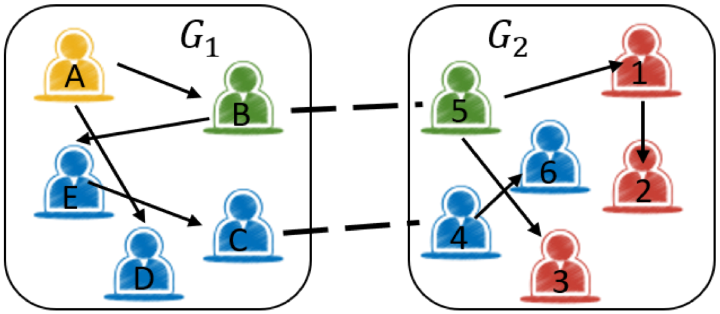
Identify Species-Specific Pathways

Protein-Protein Interaction (PPI) networks



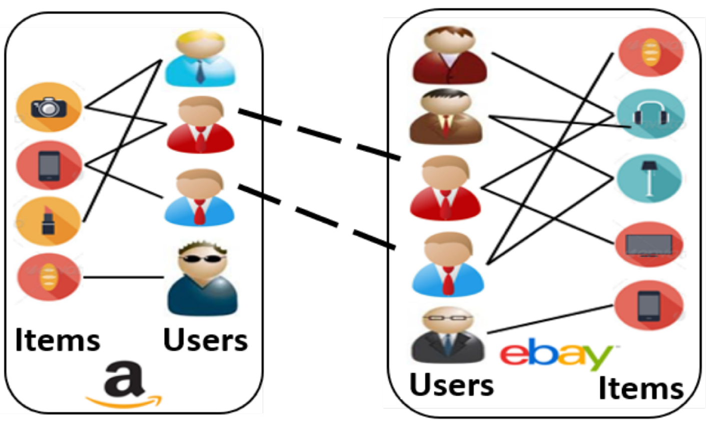
PPI network 1 PPI network 2

Cross Network Information Diffusion

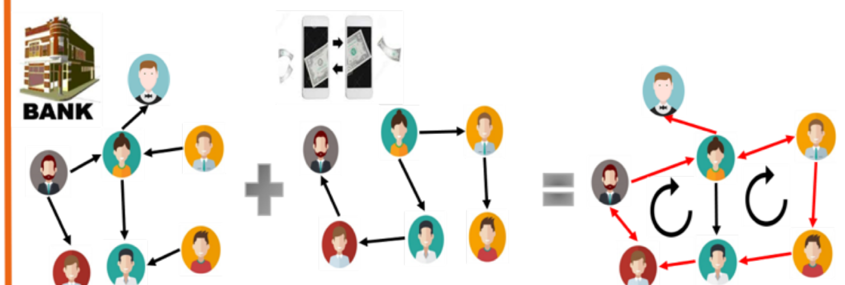


social network 1 social network 2

Cross-Site Recommendation



Fraud Detection



looks normal looks normal money laundering?

Related Setting: Graph Matching

- It solves for the permutation matrix $\boldsymbol{P}$ that minimizes

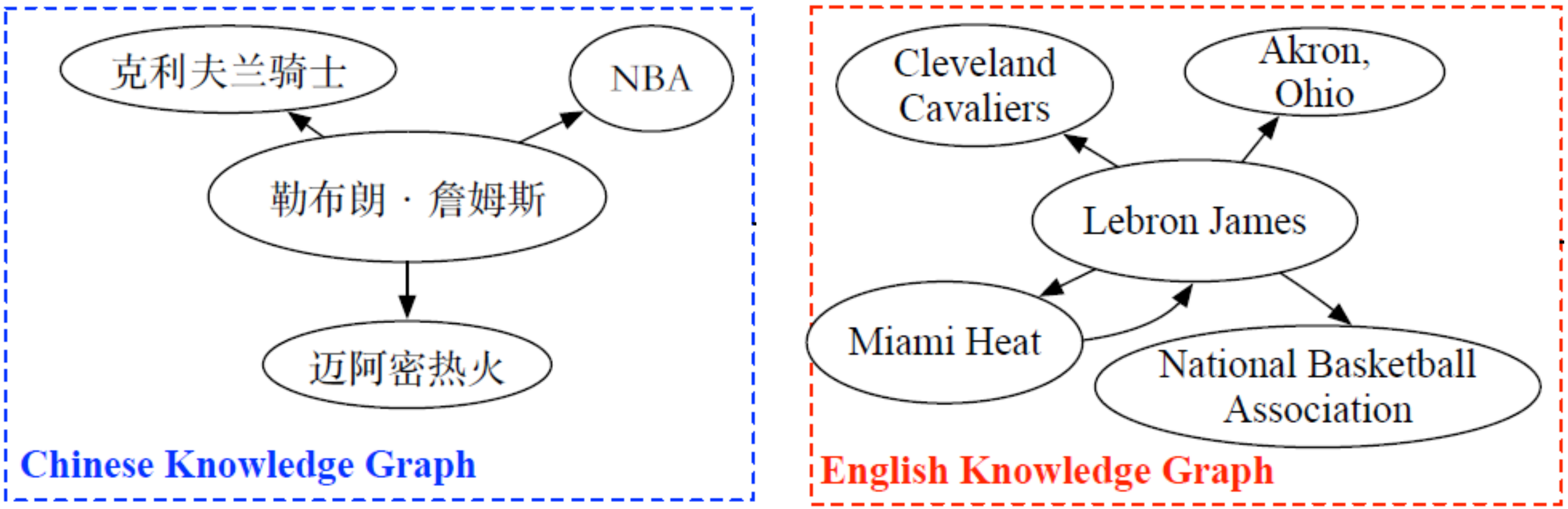
$$\|\boldsymbol{A}_2 - \boldsymbol{P}^T \boldsymbol{A}_1 \boldsymbol{P}\|_F^2 + \mathrm{Tr}(\boldsymbol{H}^T \boldsymbol{P})$$

  - Can be interpreted as a quadratic assignment problem
  - $\boldsymbol{P} \in \{0,1\}^{n\times n}, \boldsymbol{P}\mathbf{1} = \mathbf{1}, \mathbf{1}^T\boldsymbol{P} = \mathbf{1}^T$

- Need relaxations on the constraints
  - Doubly stochastic relaxation
  - Spectral relaxation
- Optional external information $\boldsymbol{H}$

Related Setting: Entity Alignment

- To align entities across knowledge graphs



Xu, Kun, et al. "Cross-lingual knowledge graph alignment via graph matching neural network." *arXiv preprint arXiv:1905.11605* (2019).

Traditional Methods

- Graph matching-based methods [Koutra'13, Zhang'15]

$$\min_{\boldsymbol{S}} \ \|\boldsymbol{A}_2 - \boldsymbol{S}^T \boldsymbol{A}_1 \boldsymbol{S}\|_F^2$$

  - Assumption: networks are noisy permutations of each other
  - Sparse probabilistic relaxation, i.e., $0 \leq \boldsymbol{S}_{ij} \leq 1$, $\|\boldsymbol{S}\|_0 \leq t$
  - For bipartite graphs, $\min_{\boldsymbol{P},\boldsymbol{Q}} \ \|\boldsymbol{B}_2 - \boldsymbol{P}\boldsymbol{B}_1\boldsymbol{Q}\|_F^2$ [Koutra'13]

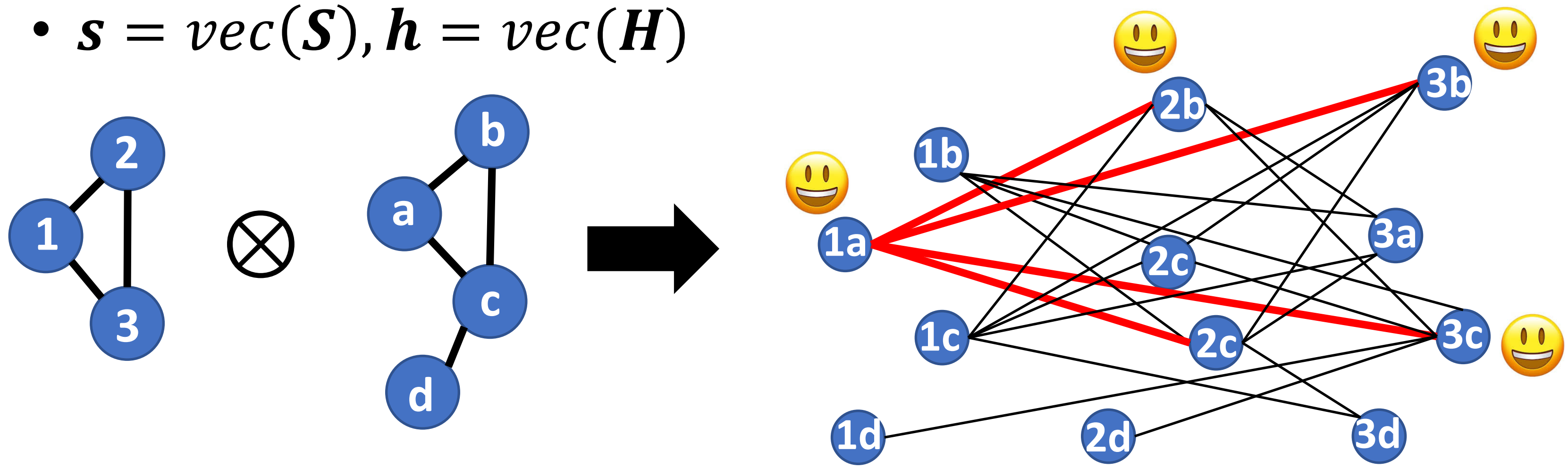
Koutra, Danai, Hanghang Tong, and David Lubensky. "Big-align: Fast bipartite graph alignment." *2013 IEEE 13th International Conference on Data Mining*. IEEE, 2013.
Zhang, Jiawei, and S. Yu Philip. "Multiple anonymized social networks alignment." *2015 IEEE International Conference on Data Mining*. IEEE, 2015.

Traditional Methods

- Random walk-based methods (e.g., IsoRank) [Singh'08, Liao'09]
  - Intuition: random walks on Kronecker product graph

$$\boldsymbol{s} = \alpha(\boldsymbol{A}_1 \otimes \boldsymbol{A}_2)\boldsymbol{s} + (1-\alpha)\boldsymbol{h}$$

  - $\boldsymbol{s} = vec(\boldsymbol{S}), \boldsymbol{h} = vec(\boldsymbol{H})$



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.
Liao, Chung-Shou, et al. "IsoRankN: spectral methods for global alignment of multiple protein networks." *Bioinformatics* 25.12 (2009): i253-i258.

Key Challenge #1: Complexity

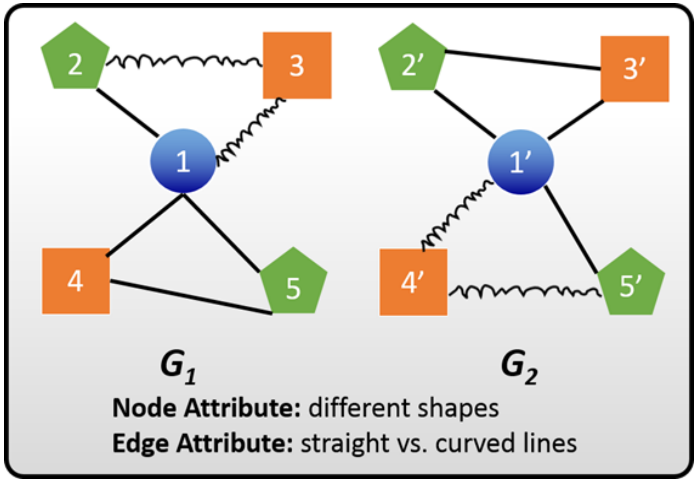
- Time complexity:
  - Most of existing works have an at least $O(n^2)$ time complexity
  - Inefficient computations for large-scale networks

- Space complexity:
  - At least $O(n^2)$ to store the alignment matrix
  - Costly memory consumptions

- **Q:** How to efficiently solve network alignment?

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?

Key Challenge #3: Heterogeneity

- 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 heterogeneity behind multi-sourced networks?

RoadMap

- Motivations and Background ✓
- Part I: Recent Network Alignment Algorithms
- Part II: Network Alignment Applications
- Part III: Future Research Directions

Overview of Part I

Part I: 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

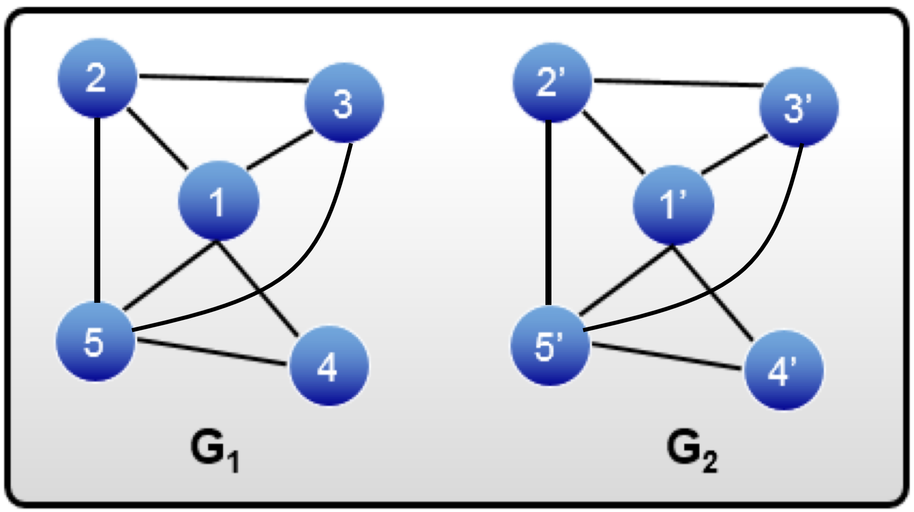
- Consistency-based
  - Single-level
  - Multilevel

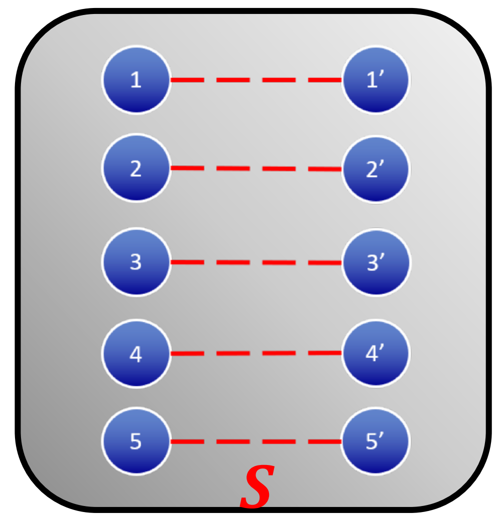
Related Tasks

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

Pairwise Network Alignment

- **Given:** **two** networks $\mathcal{G}_1, \mathcal{G}_2$ with/without attributes
- **Find:** the node correspondence across networks

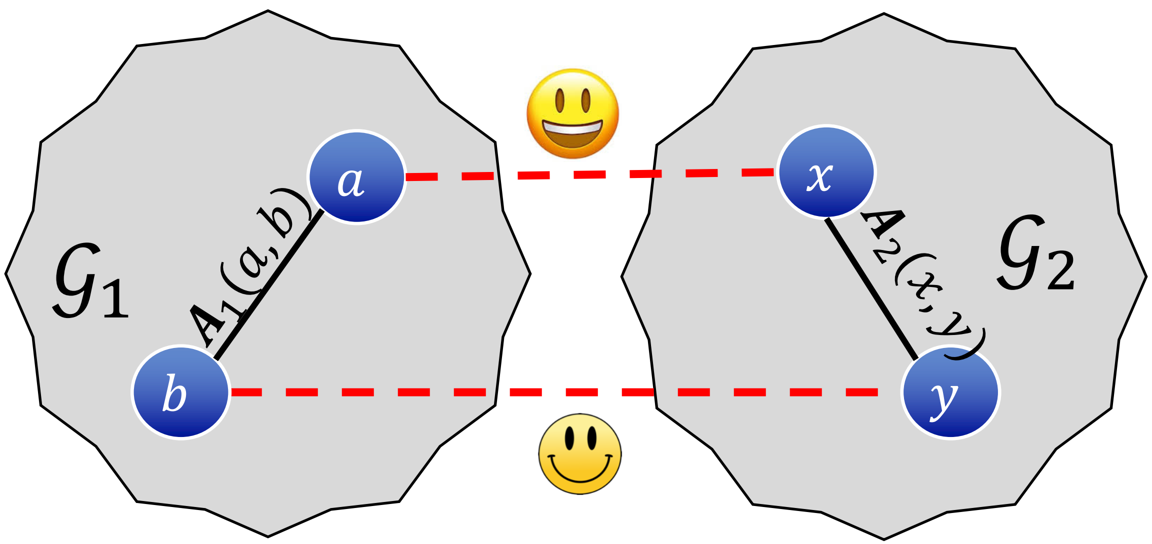


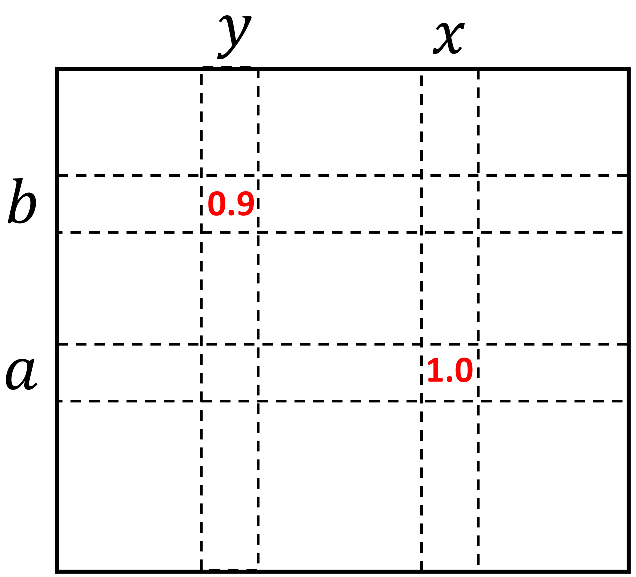


Illustrative example of pairwise network alignment w/o attributes

Consistency-Based Methods

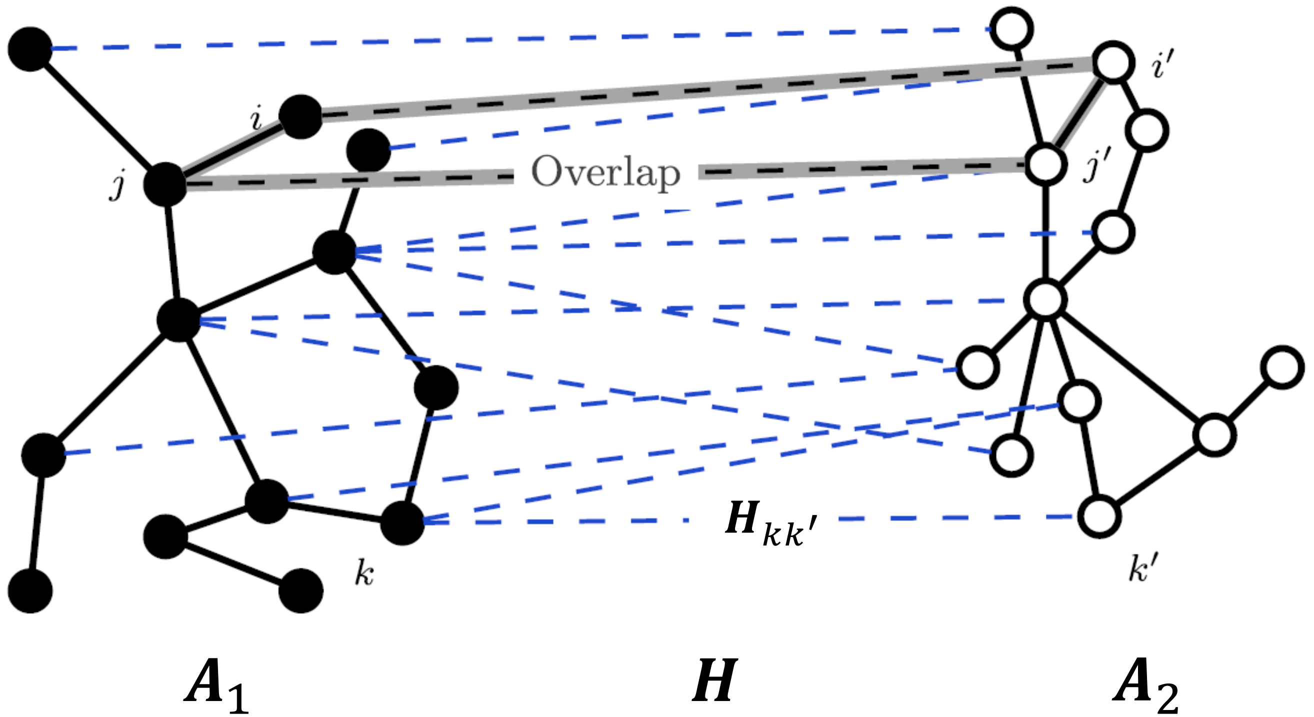
- Intuition:
  - If two nodes are aligned, e.g., node-a in $\mathcal{G}_1$ and node-x in $\mathcal{G}_2$
  - Then their neighbors are likely to be aligned





NetAlign: A Message Passing Method

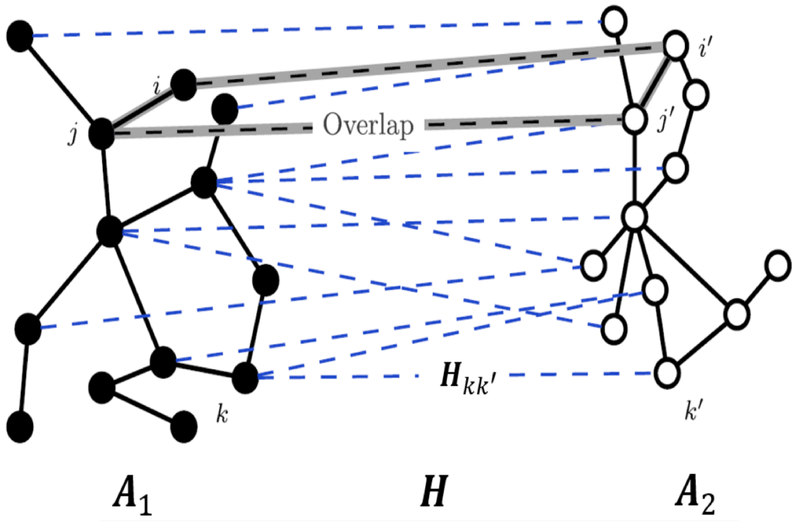
- Key idea: to maximize the number of overlaps



Bayati, Mohsen, et al. "Algorithms for large, sparse network alignment problems." *2009 Ninth IEEE International Conference on Data Mining*. IEEE, 2009.

NetAlign – Formulation #1

- To maximize the # of overlaps
  - Equivalent to maximizing the # of nonzeros in $\boldsymbol{A}$
  - $\frac{\beta}{2}\boldsymbol{s}^T\boldsymbol{A}\boldsymbol{s}$



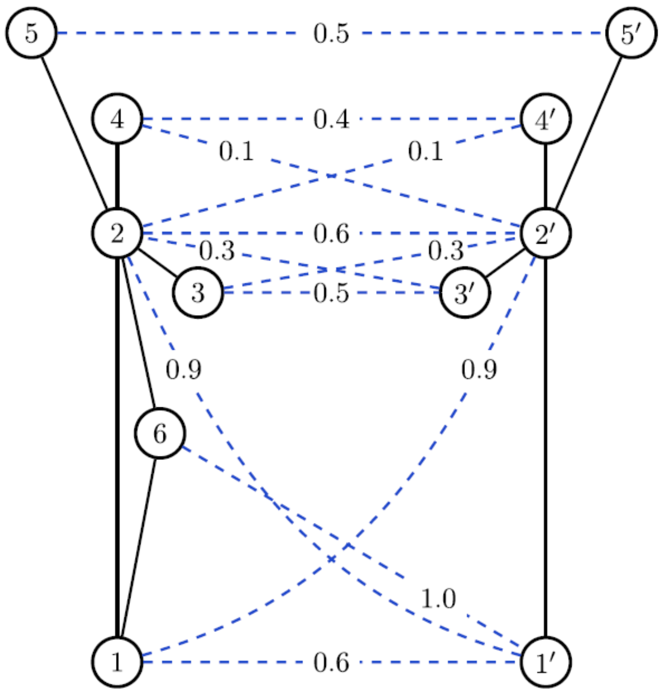
- $\boldsymbol{A}(ii', jj') = 1$ if
  - $\boldsymbol{A}_1(i,j) = 1$
  - $\boldsymbol{A}_2(i',j') = 1$
  - $\boldsymbol{H}(i,i') > 0, \boldsymbol{H}(j,j') > 0$

- $\boldsymbol{s}_{ii'}\boldsymbol{A}(ii', jj')\boldsymbol{s}_{jj'}$ is high if
  - i, i' are likely to be aligned
  - j, j' are likely to be aligned

Bayati, Mohsen, et al. "Algorithms for large, sparse network alignment problems." *2009 Ninth IEEE International Conference on Data Mining*. IEEE, 2009.

NetAlign – Formulation #2

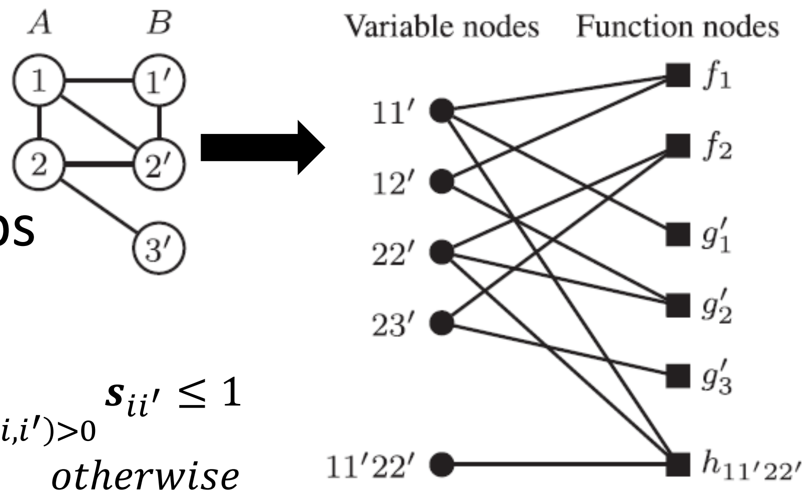
- Encode the prior knowledge
  - $\boldsymbol{s}^T \text{vec}(\boldsymbol{H}) = \sum_{ii'} \boldsymbol{S}(i,i')\boldsymbol{H}(i,i')$ → score from prior knowledge
- Valid matching constraints
  - $\sum_{i',\, s.t.\, \boldsymbol{H}(i,i')>0} \boldsymbol{S}(i,i') \leq 1$
  - $\sum_{i,\, s.t.\, \boldsymbol{H}(i,i')>0} \boldsymbol{S}(i,i') \leq 1$
  - $\boldsymbol{S}(i,i') \in \{0,1\}$



Bayati, Mohsen, et al. "Algorithms for large, sparse network alignment problems." *2009 Ninth IEEE International Conference on Data Mining*. IEEE, 2009.

NetAlign – Factor Graph

- Nodes:
  - Variable nodes: e.g.,
    - Node pairs that form overlaps
  - Function nodes: constraints



$$f_i = \begin{cases} 1 & \sum_{\boldsymbol{H}(i,i')>0} \boldsymbol{s}_{ii'} \leq 1 \\ 0 & otherwise \end{cases} \qquad g_{i'} = \begin{cases} 1 & \sum_{\boldsymbol{H}(i,i')>0} \boldsymbol{s}_{ii'} \leq 1 \\ 0 & otherwise \end{cases}$$

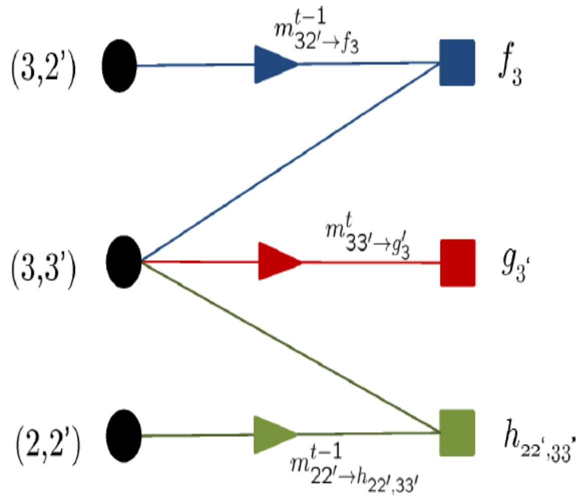
$$h_{ii'jj'} = \begin{cases} 1 & \boldsymbol{s}_{ii'jj'} = \boldsymbol{s}_{ii'} s_{jj'} \\ 0 & otherwise \end{cases}$$

- Edges: connecting each function node to the variable nodes it acts on

Bayati, Mohsen, et al. "Algorithms for large, sparse network alignment problems." *2009 Ninth IEEE International Conference on Data Mining*. IEEE, 2009.

NetAlign – Algorithm

- Belief propagation
  - Iteratively makes local and greedy decisions
  - Updated by passing messages between nodes in factor graph
- Messages $m^t_{ii' \to f_i}, m^t_{ii' \to g_{i'}}$
  - Control matching constraints
  - Also contain info about term $\alpha \boldsymbol{s}^T \mathrm{vec}(\boldsymbol{H})$
- Messages $m^t_{ii' \to h_{ii'jj'}}$
  - Agents in a square should communicate
  - Term $\frac{\beta}{2} \boldsymbol{s}^T \boldsymbol{A} \boldsymbol{s}$



Bayati, Mohsen, et al. "Algorithms for large, sparse network alignment problems." *2009 Ninth IEEE International Conference on Data Mining*. IEEE, 2009.

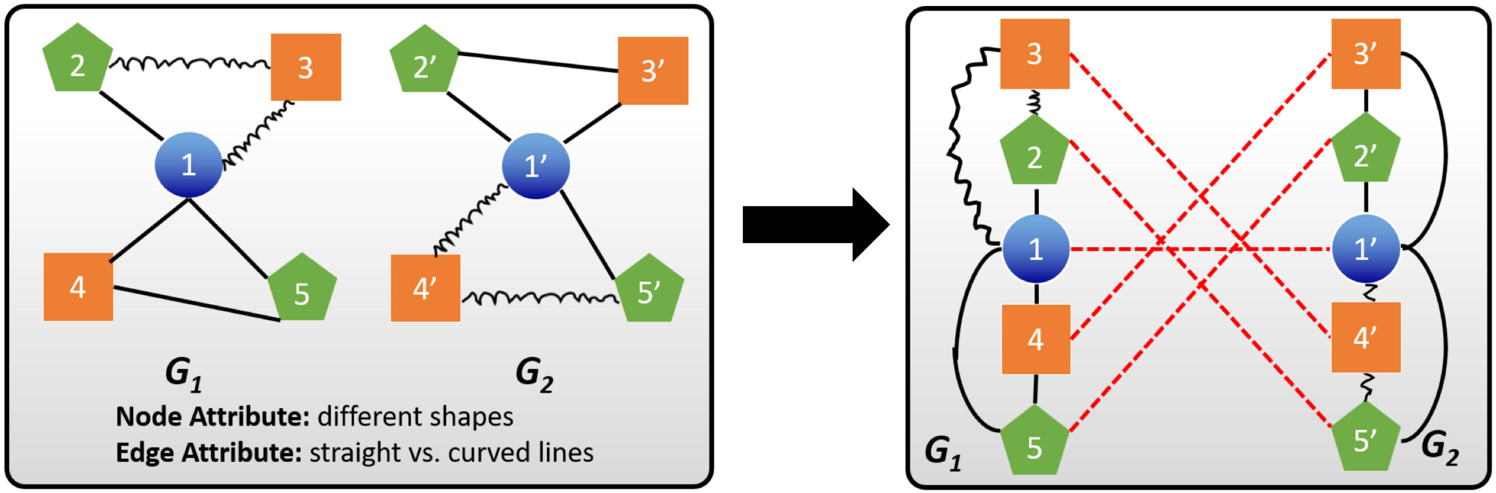
Experimental Results

Alg.	Data		Overlap	Sol. Time	Total Time
MWM	musm-homo	393	36.2%	—	—
	dmela-scere	135	35.4%	—	—
	lcsh-small	119	36.8%	—	—
	lcsh2wiki	2346	13.3%	—	—
Iso	musm-homo	1027	94.5%	0.0	0.4
	dmela-scere	301	79.0%	3.7	10.7
	lcsh-small	257	79.6%	0.0	0.7
	lcsh2wiki	11732	66.6%	11.7	587.3
MP	musm-homo	1076	99.0%	2.6	13.2
	dmela-scere	369	96.9%	26.7	34.9
	lcsh-small	316	97.8%	7.6	12.6
	lcsh2wiki	15974	90.7%	3795.3	4198.4
MP++	musm-homo	1062	97.7%	14.4	17.3
	dmela-scere	376	98.7%	28.7	33.3
	lcsh-small	318	98.5%	11.8	15.2
	lcsh2wiki	15771	89.6%	4103.8	4990.2
MR	musm-homo	1070	98.4%	12.5	12.6
	dmela-scere	375	98.4%	22.7	79.4
	lcsh-small	318	98.5%	4.1	16.8
	lcsh2wiki	16836	95.6%	4878.2	4988.0

Bayati, Mohsen, et al. "Algorithms for large, sparse network alignment problems." *2009 Ninth IEEE International Conference on Data Mining*. IEEE, 2009.

Final: Attributed Network Alignment

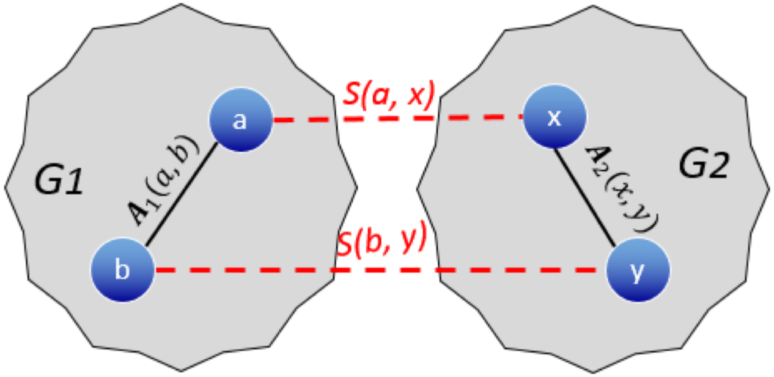
- **Given:**
  - two networks $\{\mathcal{G}_l\}$ $(l = 1, 2)$ where $\mathcal{G}_l = \{\mathcal{V}_l, \mathcal{E}_l, \boldsymbol{A}_l, \boldsymbol{N}_l, \boldsymbol{E}_l\}$ and $\boldsymbol{N}_l, \boldsymbol{E}_l$ denote the node attributes and edge attributes;
  - prior alignment matrices $\boldsymbol{H}$ between $\mathcal{G}_1$ and $\mathcal{G}_2$.
- **Find:** the alignment matrix $\boldsymbol{S}$ between $\mathcal{G}_1$ and $\mathcal{G}_2$.



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.

Final – Formulation #1

- Topological consistency
  - **Intuition:** similar node-pairs tend to have similar neighboring node-pairs



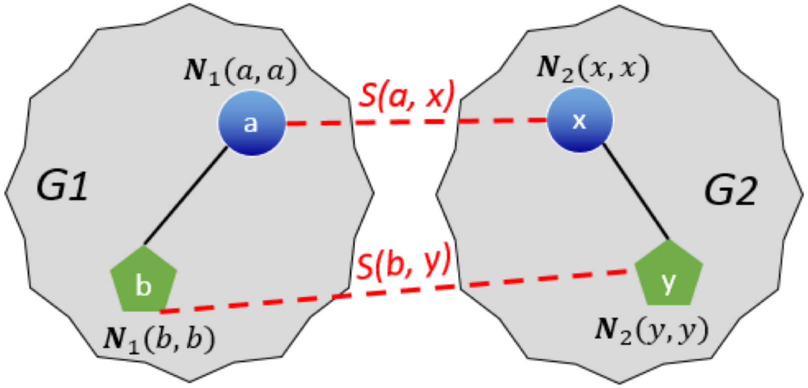
  - Example:
    - Large $\boldsymbol{S}(a,x)$
    - Large $\boldsymbol{A}_1(a,b)$ and $\boldsymbol{A}_2(x,y)$

$\longrightarrow$ large $\boldsymbol{S}(b,y)$

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.

Final – Formulation #2

- Node attribute consistency
  - **Intuition:** similar node-pairs share similar node attributes

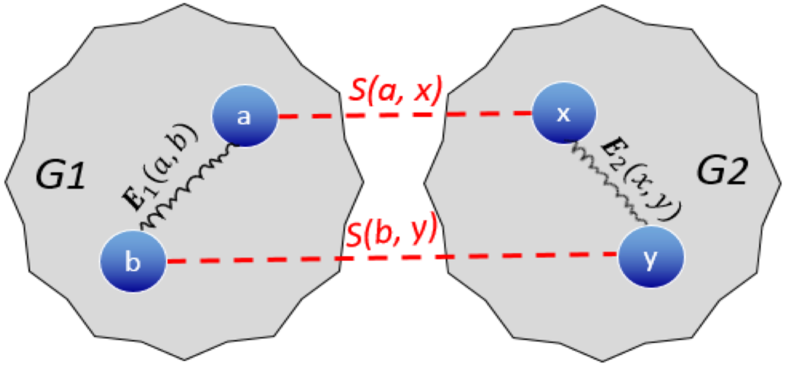


  - Large $\boldsymbol{S}(a,x) \longrightarrow$ node-a and node-x share similar attributes

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.

Final – Formulation #3

- Edge attribute consistency
  - **Intuition:** similar node-pairs connect to their neighbor-pairs via similar edge attributes



- Example:
  - Large $\boldsymbol{S}(a, x)$
  - Large $\boldsymbol{S}(b, y)$

→ Edge (a, b) & (x, y) share similar attributes

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.

Final – Overall Formulation

- Objective function

$$\min_{\mathbf{S}} \; J(\mathbf{S}) = \sum_{a,b,x,y} \left[\frac{\mathbf{S}(x,a)}{\sqrt{f(x,a)}} - \frac{\mathbf{S}(y,b)}{\sqrt{f(y,b)}} \right]^2 \underbrace{\mathbf{A}_1(a,b)\mathbf{A}_2(x,y)}_{\text{#1. Topology Consistency}}$$

$$\times \underbrace{\Phi(x,a)\Phi(y,b)}_{\text{#2. Node Attribute Consistency}} \times \underbrace{\Psi\big((x,y),(a,b)\big)}_{\text{#3. Edge Attribute Consistency}}$$

- Matrix-form objective function

$$\min_{\mathbf{S}} J(\mathbf{S}) = \min_{\mathbf{s}} \sum_{v,w} \left[\frac{\mathbf{s}(v)}{\sqrt{\mathbf{D}(v,v)}} - \frac{\mathbf{s}(w)}{\sqrt{\mathbf{D}(w,w)}} \right]^2 \mathbf{W}(v,w)$$

$$= \min_{\mathbf{s}} \mathbf{s}^T \big(\mathbf{I} - \widetilde{\mathbf{W}}\big)\mathbf{s}$$

$\boxed{\mathbf{s} = \mathrm{vec}(\mathbf{S})}$

attributed Kronecker product

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.

Final – Algorithm

- Fixed-point solution: by setting derivative to 0
  - Converges to the global optimal solution

$$\boldsymbol{s} = a\widetilde{\boldsymbol{W}}\boldsymbol{s} + (1-\alpha)\boldsymbol{h} \Rightarrow \boldsymbol{s} = (1-\alpha)\left(\boldsymbol{I} - \alpha\widetilde{\boldsymbol{W}}\right)^{-1}\boldsymbol{h}$$

- **Intuition:** a similarity propagation to neighboring node-pairs, which is additionally calibrated by node/edge attributes
- Speed-up variants:
  - Low-rank approximation for full alignment
  - Low-rank approximation for on-query alignment

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.

Final – Low-Rank Approximation Algorithm

- If we only consider node attributes

$$\boldsymbol{s} = (1-\alpha)\left(\boldsymbol{I} - \alpha \boldsymbol{D}_N^{-\frac{1}{2}} \boldsymbol{N}(\boldsymbol{A}_1 \otimes \boldsymbol{A}_2)\boldsymbol{N}\boldsymbol{D}_N^{-\frac{1}{2}}\right)^{-1} \boldsymbol{h}$$

- **Key Idea:** Low rank approximation of $\boldsymbol{A}_1$ and $\boldsymbol{A}_2$

$$\boldsymbol{A}_1 \approx \boldsymbol{U}_1 \boldsymbol{\Lambda}_1 \boldsymbol{U}_1^T$$
$$\boldsymbol{A}_2 \approx \boldsymbol{U}_2 \boldsymbol{\Lambda}_2 \boldsymbol{U}_2^T$$

Sherman-Morrison Lemma

$$\boldsymbol{s} \approx (1-\alpha)\left(\boldsymbol{I} + \alpha \boldsymbol{D}_N^{-\frac{1}{2}} \boldsymbol{N}\boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{U}^T\boldsymbol{N}\boldsymbol{D}_N^{-\frac{1}{2}}\right)\boldsymbol{h}$$

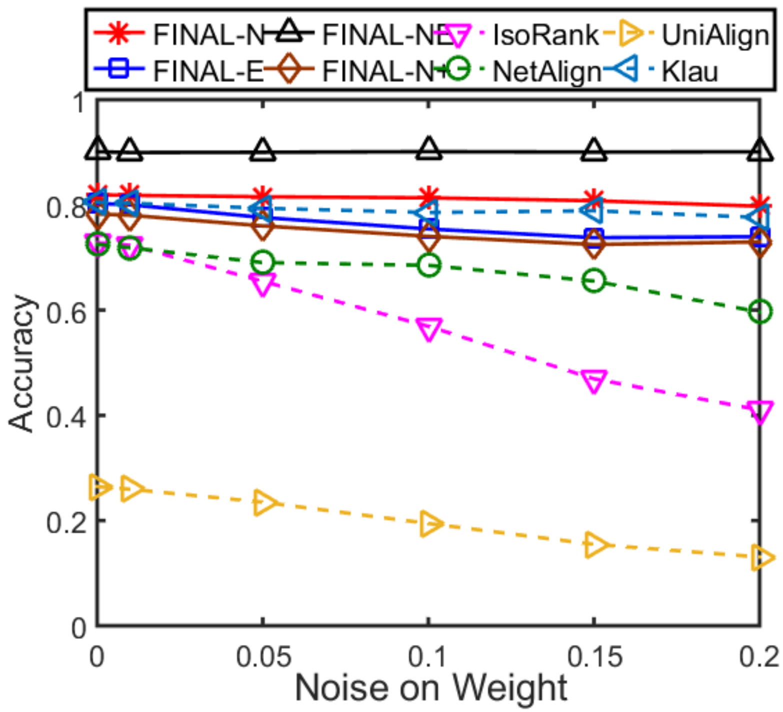
where $\boldsymbol{U} = \boldsymbol{U}_1 \otimes \boldsymbol{U}_2$

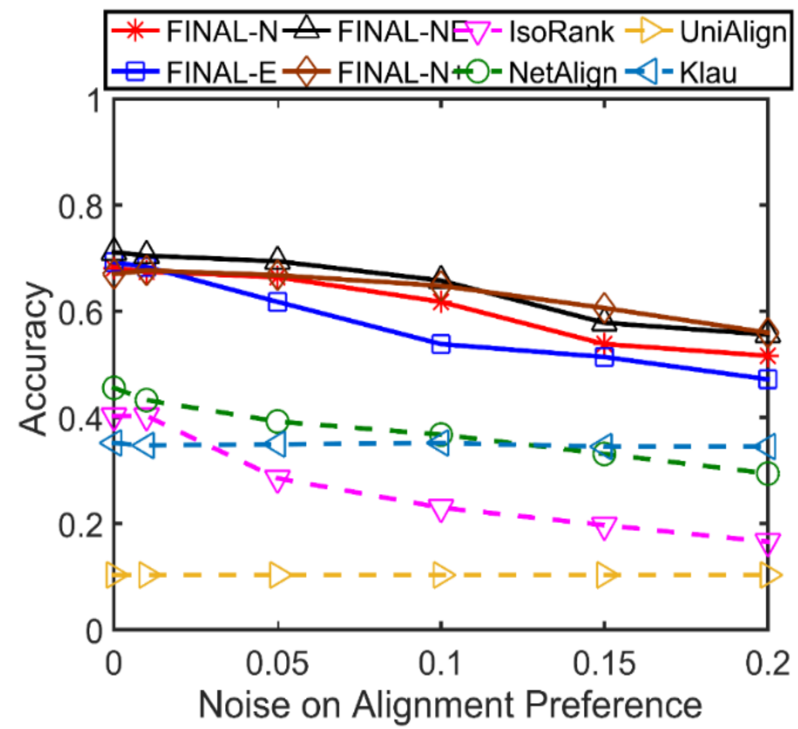
$$\boldsymbol{\Lambda} = [(\boldsymbol{\Lambda}_1 \otimes \boldsymbol{\Lambda}_2)^{-1} - \alpha \boldsymbol{U}^T\boldsymbol{N}\boldsymbol{D}_N^{-1}\boldsymbol{N}\boldsymbol{U}]^{-1}$$

- **Complexity**: $O(n^6)$ or $O(mnt_{\max})$ → $O(n^2r^4)$

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.

Final – Experimental Results

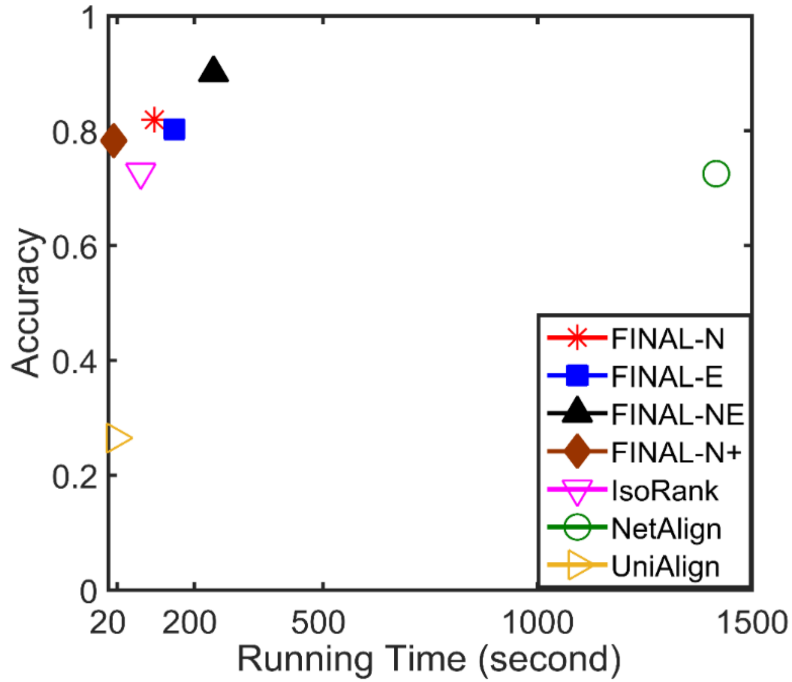


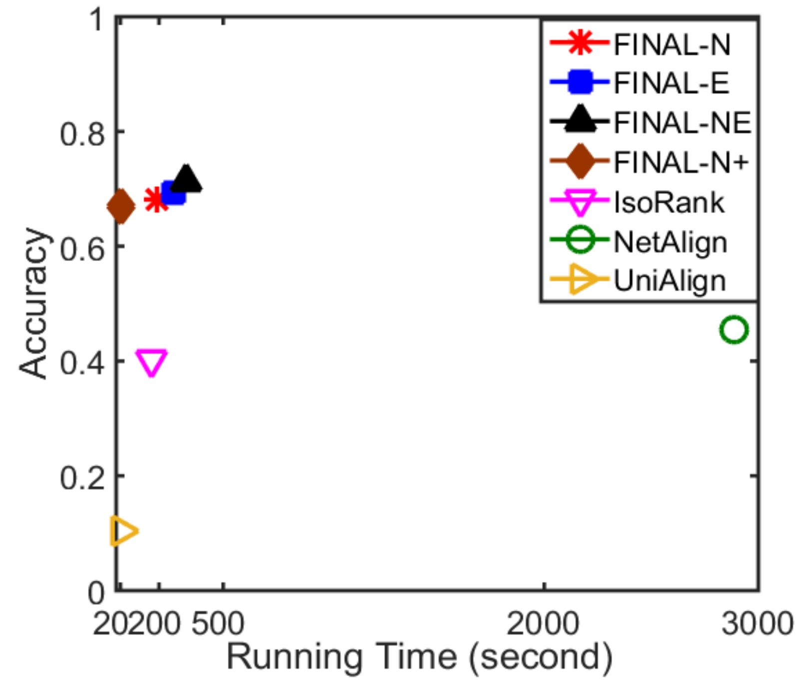


Observation: attributes help improve network alignment.

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.

Final – Experimental Results

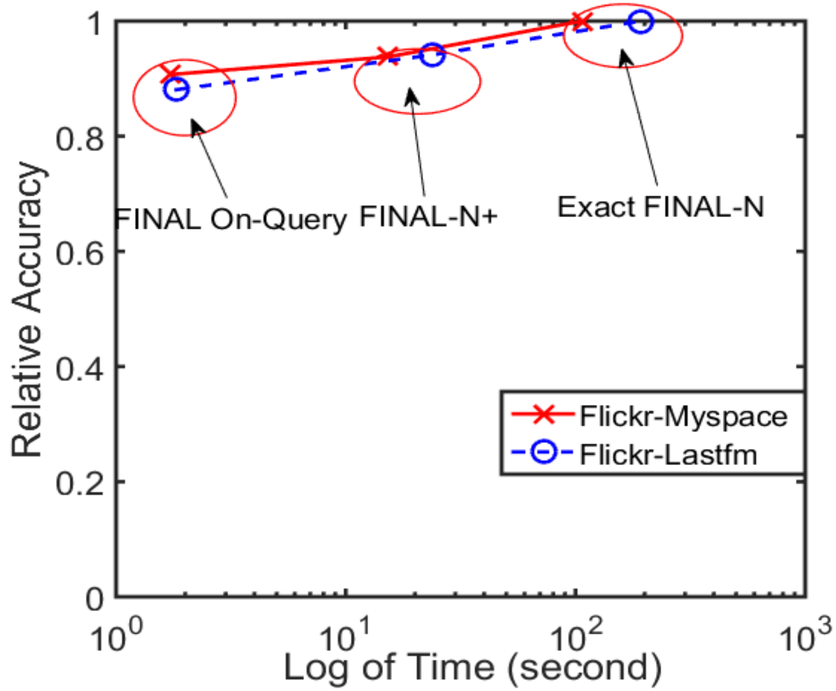




Observation: FINAL gains a better quality-speed balance.

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.

Final – Experimental Results



Observation: FINAL On-Query gains around 90% accuracy relative to exact FINAL-N, but more than 100 times faster.

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.

Final – More on Computations

- Further speed-up: from $O(n^2)$ to $O(m)$
  - Key idea: indirect representation of $\boldsymbol{S}$ [1]
  - Theorem: Low-rank of $\boldsymbol{A}_1$ and $\boldsymbol{A}_2$ → low-rank of $\boldsymbol{S}$

$$\boldsymbol{S} \Leftarrow \boldsymbol{U}_2 \times \underbrace{\boldsymbol{M}}_{\boldsymbol{r}_2 \times \boldsymbol{r}_1} \times \boldsymbol{U}_1^T$$

- Alignment quality: linear complexity w/o approximation
  - Multilevel alignment (perfect interpolation theorem) [2]
  - Implicit Krylov subspace methods [3]

[1] Zhang, Si, et al. "ineat: Incomplete network alignment." 2017 IEEE International Conference on Data Mining (ICDM). IEEE, 2017.
[2] Zhang, Si, et al. "Multilevel network alignment." The World Wide Web Conference. 2019.
[3] Du, Boxin, and Hanghang Tong. "Fasten: Fast sylvester equation solver for graph mining." Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. 2018.

Overview of Part I

Part I: 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

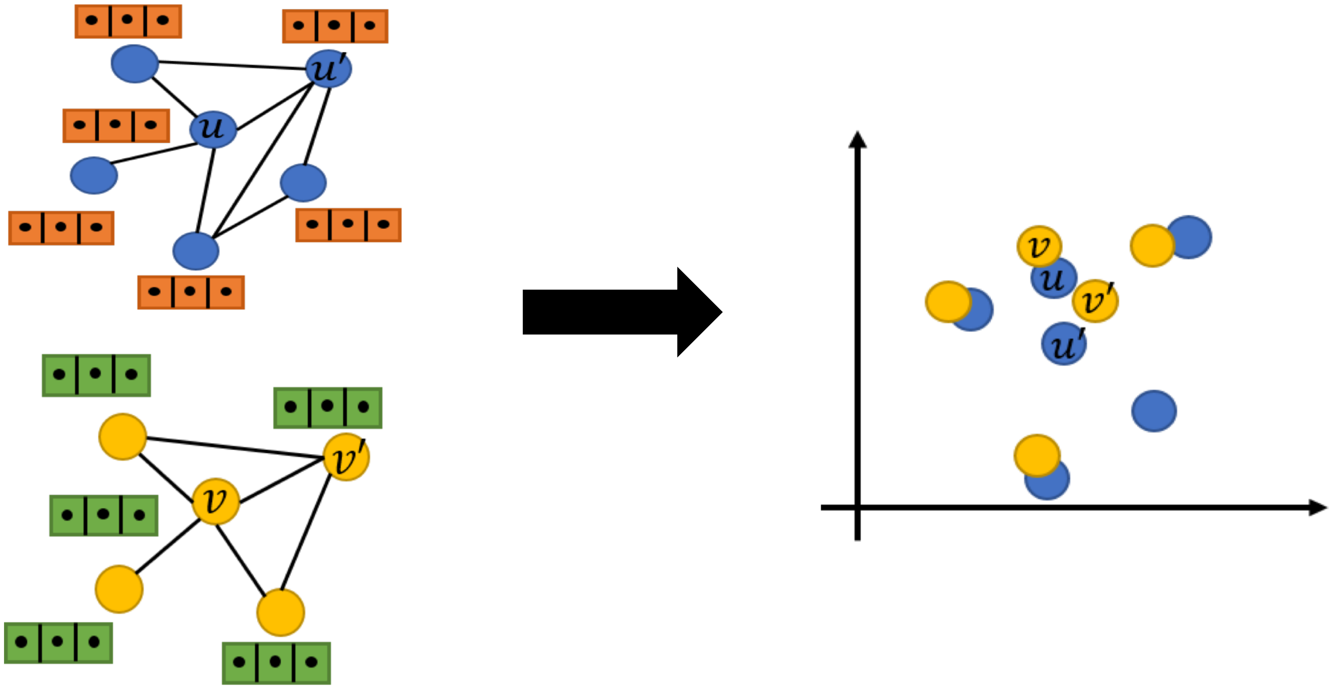
- Consistency-based
  - Single-level
  - Multilevel

Related Tasks

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

Embedding-Based Methods

- Intuition: to learn node representations that
  - Preserve structural/attribute proximity within networks
  - Preserve proximity across aligned nodes



IONE: Aligning Users by Network Embedding

- Background: network embedding by LINE (2nd order)
  - Compute two distributions:

Empirical distribution of neighborhood structure:

$$\hat{p}_2(v_j \mid v_i) = \frac{w_{ij}}{\sum_{k \in V} w_{ik}}$$

Model distribution of neighborhood structure:

$$p_2(v_j \mid v_i) = \frac{\exp(\vec{u}'^T_i \vec{u}_j)}{\sum_{k \in V} \exp(\vec{u}'^T_k \vec{u}_i)}$$

  - Minimize the KL divergence

by omitting constant terms

$$O_2 = \sum_i KL(\hat{p}_2(\cdot \mid v_i), p_2(\cdot \mid v_i)) = -\sum_{(i,j) \in E} w_{ij} \log p_2(v_j \mid v_i)$$

Tang, Jian, et al. "Line: Large-scale information network embedding." Proceedings of the 24th international conference on world wide web. 2015.
Liu, Li, et al. "Aligning Users across Social Networks Using Network Embedding." *Ijcai*. 2016.

IONE – Within-Network Embedding

- **Intuition:** to preserve structure proximity
- Embedding vectors for node-i
  - A node vector $\boldsymbol{u}_i$
  - Context vectors: (1) input context $\boldsymbol{u}_i'$, (2) output context $\boldsymbol{u}_i''$

Input context model distribution:
$$p_1(v_j|v_i) = \frac{\exp\left(\overrightarrow{u_j}'^T \cdot \overrightarrow{u_i}\right)}{\Sigma_{k=1}^{|V|} \exp\left(\overrightarrow{u_k}'^T \cdot \overrightarrow{u_i}\right)}$$

Output context model distribution:
$$p_2(v_i|v_j) = \frac{\exp\left(\overrightarrow{u_i}''^T \cdot \overrightarrow{u_j}\right)}{\Sigma_{k=1}^{|V|} \exp\left(\overrightarrow{u_k}''^T \cdot \overrightarrow{u_j}\right)}$$

Empirical distributions: $\widehat{p}_1(i,j) = w_{ij}/d_i^{out}$ $\quad \widehat{p}_2(i,j) = w_{ij}/d_j^{in}$

- Objective: minimize KL divergences

Liu, Li, et al. "Aligning Users across Social Networks Using Network Embedding." *Ijcai*. 2016.

IONE – Cross-Network Embedding

- **Intuition:** aligned nodes coincide in embedding space

Model distribution: $$p_1\left(v_j^Y \middle| v_k^X\right) = \frac{\exp(\boldsymbol{u}_j'^{Y\,T}\boldsymbol{u}_k^X)}{\sum_{k\in V_X}\exp(\boldsymbol{u}_j'^{Y\,T}\boldsymbol{u}_k^X)}$$

Empirical distribution: $$\hat{p}_1\left(v_j^Y \middle| v_k^X\right) = \sum_{v_i\in V_Y} p_a\left(v_i^Y \middle| v_k^X\right)\times\frac{w_{ij}}{d_i^{out}}$$

  - $p_a\left(v_i^Y \middle| v_k^X\right)$: probability that v_k^X and v_i^Y are aligned

- Objective: minimize KL divergences
  - e.g., $p_1(v_j^Y|v_i^X)$ vs. $\hat{p}_1\left(v_j^Y \middle| v_i^X\right)$

Liu, Li, et al. "Aligning Users across Social Networks Using Network Embedding." *Ijcai*. 2016.

IONE – Model Inference

- SGD with negative sampling

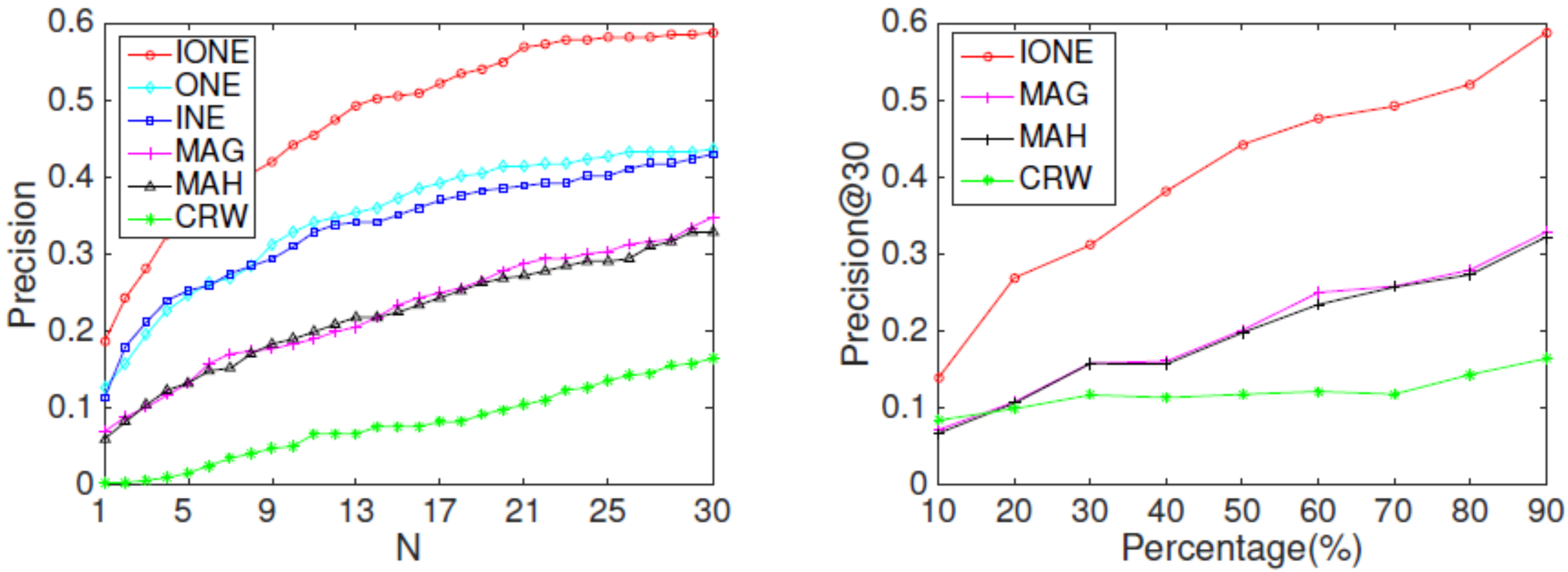
$$\log p_1(v_j^X | v_i^X) \propto \log \sigma(\overrightarrow{u_j}'^{X^T} \cdot \overrightarrow{u_i}^X) + \Sigma_{m=1}^K E_{v_n \sim p_n(v)} \log \sigma(-\overrightarrow{u_n}'^{X^T} \cdot \overrightarrow{u_i}^X)$$

$$\log p_1(v_j^Y | v_k^X) \propto \log \sigma(\overrightarrow{u_j}'^{Y^T} \cdot \overrightarrow{u_k}^X) + \Sigma_{m=1}^K E_{v_n \sim p_n(v)} \log \sigma(-\overrightarrow{u_n}'^{Y^T} \cdot \overrightarrow{u_k}^X)$$

Liu, Li, et al. "Aligning Users across Social Networks Using Network Embedding." *Ijcai*. 2016.

IONE – Experimental Results

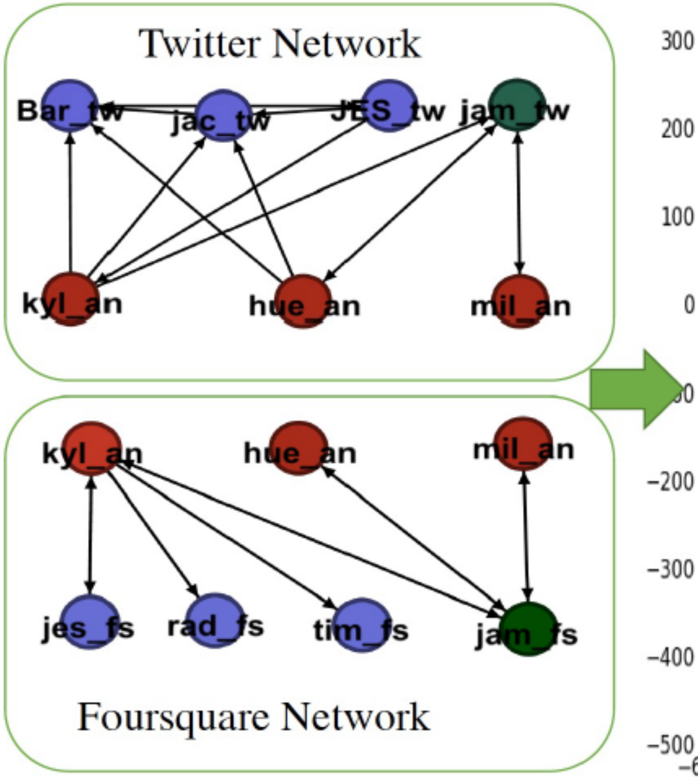
- Dataset: Foursquare-Twitter

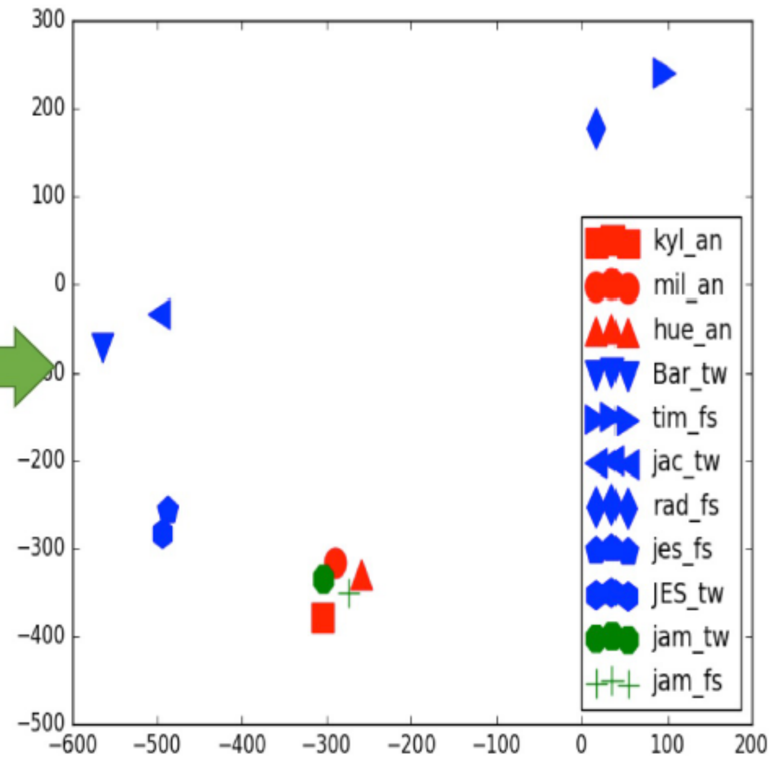


(a) $P@N$ (b) $P@30$ vs. Training ratio

Liu, Li, et al. "Aligning Users across Social Networks Using Network Embedding." *Ijcai*. 2016.

IONE – Case Study





Liu, Li, et al. "Aligning Users across Social Networks Using Network Embedding." *Ijcai*. 2016.

DeepLink: Deep Learning for User Identity Linkage

- Motivations:
  - Heterogeneity across networks → Complex alignment
  - Scarcity of labeled alignment → Supervised training is not easy

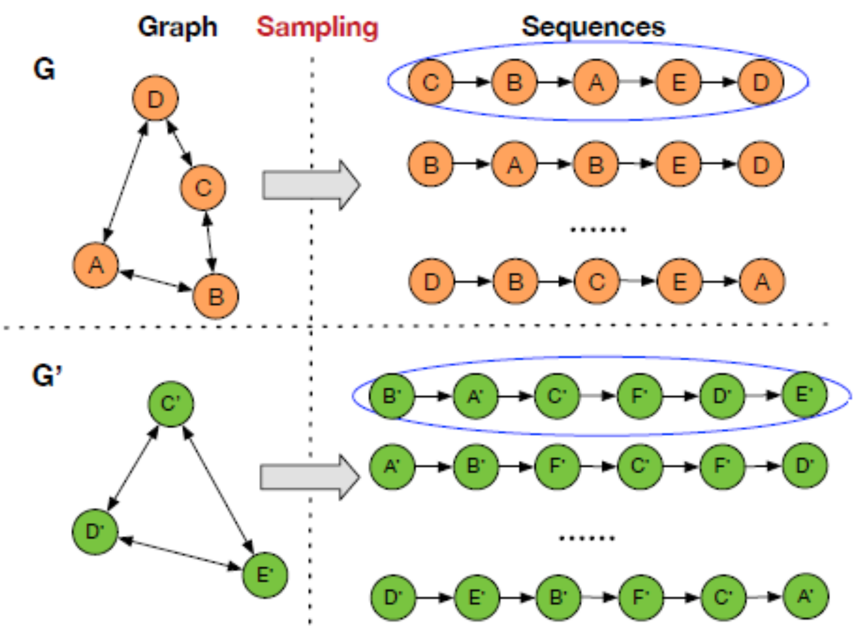
- Key questions:
  - How to learn non-linear transformation for alignment?
  - How to boost supervised training algorithm?

- Key idea: use deep neural network with dual-learning

Zhou, Fan, et al. "Deeplink: A deep learning approach for user identity linkage." IEEE INFOCOM 2018-IEEE Conference on Computer Communications. IEEE, 2018.

DeepLink – Network Embedding

- Key idea: pre-trained Skip-gram based embedding
  - To predict the context of a center node



- Context sampling:
  - Random walks from center nodes
- Objective function:
  - Original: to maximize

$$p(u_{t+j} \mid u_t) = \frac{\exp(\mathbf{v}_{u_{t+j}}^{\mathrm{T}} \mathbf{v}'_{u_t})}{\sum_{i=1}^{m} \exp(\mathbf{v}_{u_i}^{\mathrm{T}} \mathbf{v}'_{u_t})}$$

  - With negative sampling:

$$\log[\sigma(\mathbf{v}_{u_{t+j}}^{\mathrm{T}} \mathbf{v}'_{u_t})] + \sum_{i=1}^{K} \mathbb{E}_{u_i \sim p_n(u)}[\log(1 - \sigma(\mathbf{v}_{u_i}^{\mathrm{T}} \mathbf{v}'_{u_t}))]$$

Zhou, Fan, et al. "Deeplink: A deep learning approach for user identity linkage." IEEE INFOCOM 2018-IEEE Conference on Computer Communications. IEEE, 2018.

DeepLink – Neural Mapping Learning

- Goal: to learn non-linear alignment across networks
- Intuition: neural networks capture complex nonlinearity
- Key idea: use two multilayer perceptrons as mappings
  - One MLP (denoted by Φ) to map from network $\mathcal{G}^s$ to $\mathcal{G}^t$
  - Another MLP (denoted by Φ^{-1}) for $\mathcal{G}^t$ to $\mathcal{G}^s$

Zhou, Fan, et al. "Deeplink: A deep learning approach for user identity linkage." IEEE INFOCOM 2018-IEEE Conference on Computer Communications. IEEE, 2018.

DeepLink – Dual Learning

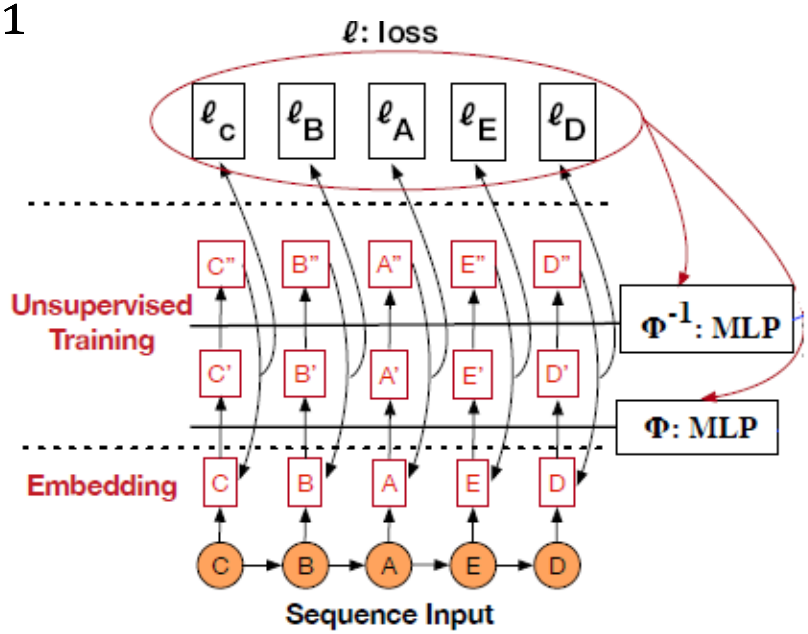
- Goal: to address the lack of labeled alignment

- Components:
  - **Unsupervised alignment pre-training** uses node embedding to learning two weak mapping functions Φ and Φ^{-1}
  - **Supervised alignment learning** uses labeled alignment to improve weak mapping functions

Zhou, Fan, et al. "Deeplink: A deep learning approach for user identity linkage." IEEE INFOCOM 2018-IEEE Conference on Computer Communications. IEEE, 2018.

DeepLink – Unsupervised Pre-training

- Goal: to learn self-consistent mappings
- Method: autoencoder type of architecture
  - Encoder: mapping function Φ
  - Decoder: mapping function Φ^{-1}
- Objective function:
  - Minimize difference between $\Phi^{-1}\left(\Phi(\boldsymbol{v}_u)\right)$ and $\boldsymbol{v}_u$



Zhou, Fan, et al. "Deeplink: A deep learning approach for user identity linkage." IEEE INFOCOM 2018-IEEE Conference on Computer Communications. IEEE, 2018.

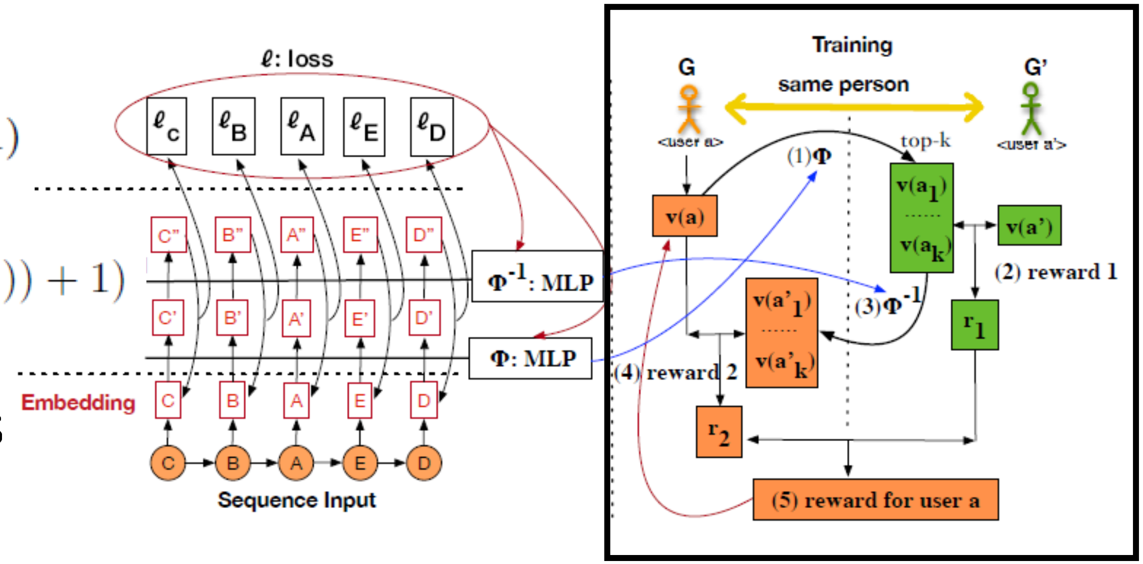
DeepLink – Supervised Learning

- Key idea: align according to some reward functions
- Method:
  - Find k-similar embeddings $\boldsymbol{v}'(u_i)$ in $\mathcal{G}^t$ for mapped embeddings of node-a in $\mathcal{G}^s$, i.e., $u_i \in \mathrm{Top}(\Phi(v(u_a)))$
  - Rewards:

$$r_{s,t}^{a} = \frac{1}{k}\sum_{i=1}^{k}\log(\cos(\mathbf{v}(u_i), \mathbf{v}'(u_a)) + 1)$$

$$r_{t,s}^{a} = \frac{1}{k}\sum_{i=1}^{k}\log(\cos(\Phi^{-1}(\mathbf{v}'(u_i)), \mathbf{v}(u_a)) + 1)$$

  - To maximize rewards



Zhou, Fan, et al. "Deeplink: A deep learning approach for user identity linkage." IEEE INFOCOM 2018-IEEE Conference on Computer Communications. IEEE, 2018.

DeepLink – Experimental Results

- Dataset: Foursquare-Twitter

Comparisons of alignment precision.

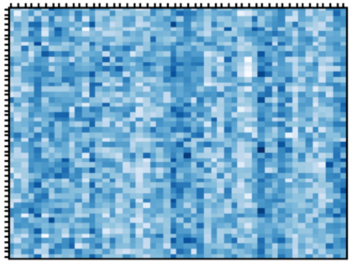
	Precision					
	P@1	*P@5*	*P@9*	*P@13*	*P@21*	*P@30*
CRW	0.0000	0.0219	0.0476	0.0538	0.0909	0.1603
MAG	0.0638	0.1362	0.1705	0.2081	0.2708	0.3229
MAH	0.0500	0.1219	0.1886	0.2148	0.2513	0.0003
ONE	0.1229	0.2533	0.3038	0.3510	0.4071	0.4270
IONE	0.2238	0.4033	0.4638	0.5010	0.5571	0.5970
DeepLink	**0.3447**	**0.5942**	**0.6609**	**0.6866**	**0.7000**	**0.7048**

Observation: DeepLink achieves highest accuracy in top-k identity matching.

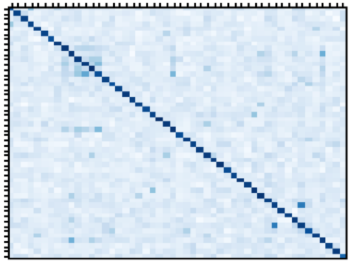
Zhou, Fan, et al. "Deeplink: A deep learning approach for user identity linkage." IEEE INFOCOM 2018-IEEE Conference on Computer Communications. IEEE, 2018.

DeepLink – Experimental Results

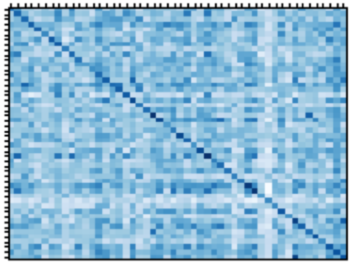
- Visualization of cosine similarities of randomly sampled anchor nodes (the more diagonalized, the better).



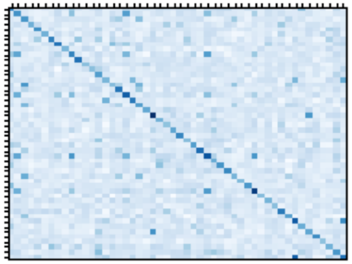
(a) IONE Anchor Nodes



(b) DeepLink Anchor Nodes



(c) IONE Testing Nodes



(d) DeepLink Testing Nodes

Observations:

- IONE disrupts the embedding similarities of labeled alignment pairs after training.
- In contrast, DeepLink still preserves the anchor linkage.
- Similarly for testing anchor nodes.

Zhou, Fan, et al. "Deeplink: A deep learning approach for user identity linkage." IEEE INFOCOM 2018-IEEE Conference on Computer Communications. IEEE, 2018.

Regal: Representation Learning-Based Graph Alignment

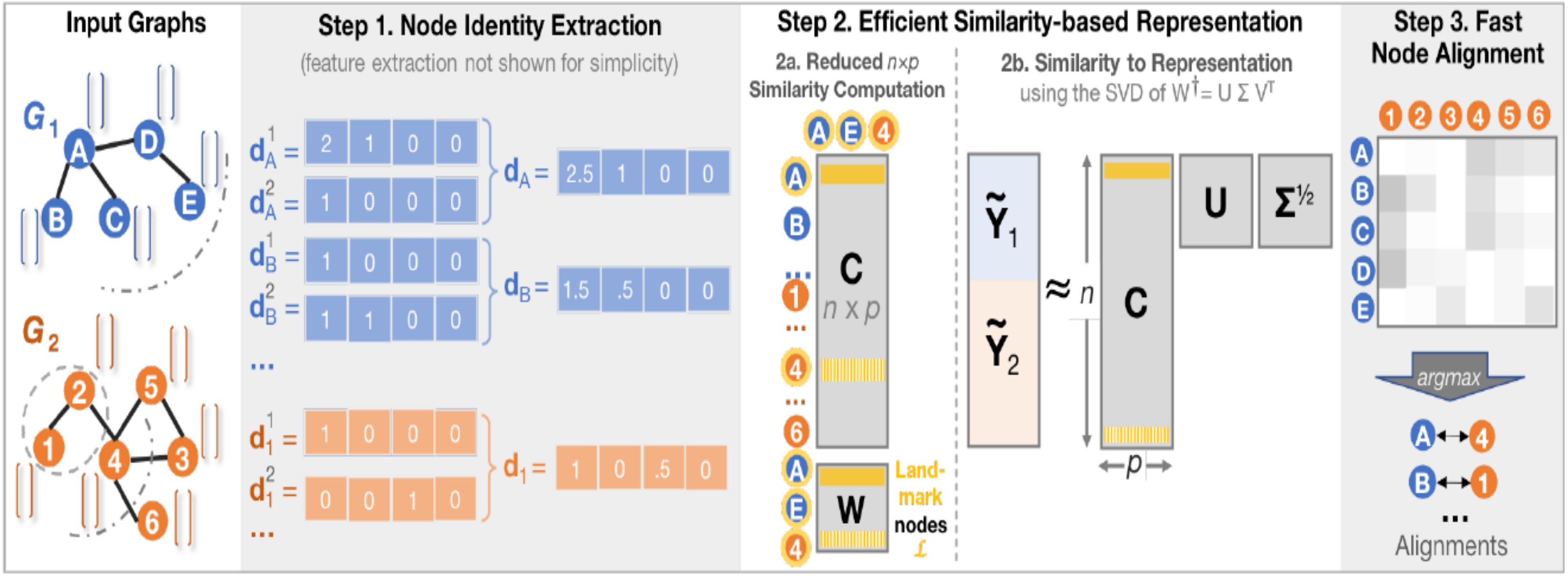
- Goal: unsupervised embedding for network alignment

	Regal	IONE
Key idea	matrix factorization	skip-gram w/ negative sampling
Attributes	w/ node attributes	w/o attributes
Supervision	unsupervised	semi-supervised
Complexity	sub-quadratic	sub-quadratic

Heimann, Mark, et al. "Regal: Representation learning-based graph alignment." *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*. 2018.

Regal – Overview

- Node feature extraction
- Node embedding learning by matrix factorization
- Network alignment



Heimann, Mark, et al. "Regal: Representation learning-based graph alignment." *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*. 2018.

Regal – Node Feature Extraction

- Structural identity
  - $\mathcal{R}_u^k$: the set of nodes exactly k steps away from u
  - $\boldsymbol{d}_u^k(i)$: the number of nodes in $\mathcal{R}_u^k$ with degree of i
  - $\boldsymbol{d}_u = \sum_{i=1}^{K} \delta^{k-1} \boldsymbol{d}_u^k$ (δ is the discount factor)
  - Logarithmic binning: $\boldsymbol{d}_u^k(i)$ is the number of nodes $u \in \mathcal{R}_u^k$ such that $\lfloor \log_2 \deg(u) \rfloor = i$

- Attribute-based identity
  - Node input feature vector $\boldsymbol{f}_u$

Heimann, Mark, et al. "Regal: Representation learning-based graph alignment." *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*. 2018.

Regal – Cross-Network Node Similarity

- Direct computation

$$sim(u,v) = \exp[-\gamma_s \|\boldsymbol{d}_u - \boldsymbol{d}_v\|_2^2 - \gamma_a \times dist(\boldsymbol{f}_u, \boldsymbol{f}_v)]$$

  - Limitation: costly computation $O(n^2)$ where $n = n_1 + n_2$
- Efficient computation
  - Reduce to node-landmark similarity
  - $\mathcal{L}$: a set of p landmark nodes chosen randomly
  - Node-landmark similarity matrix: $\boldsymbol{C}(u,v)$, $v \in \mathcal{L}$
  - Landmark-landmark similarity

$$\boldsymbol{W}(v_1, v_2) = \boldsymbol{C}(v_1, v_2), v_1 \in \mathcal{L}$$

Heimann, Mark, et al. "Regal: Representation learning-based graph alignment." *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*. 2018.

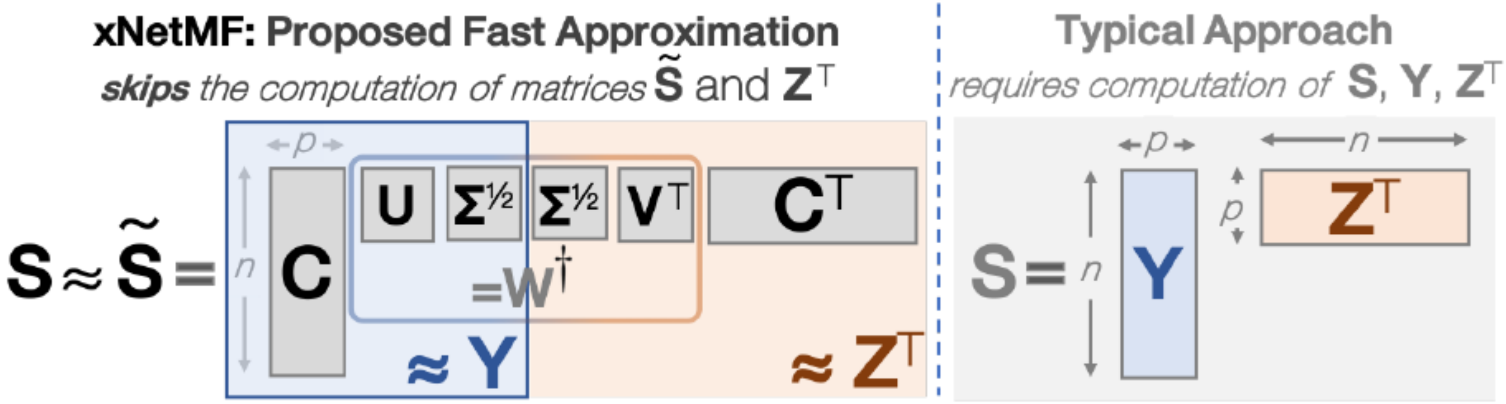
Regal – Node Embedding Learning

- Nystrom-based approximation

$$S \approx \tilde{S} = CW^{+}C^{T}$$

  - W^{+}: pseudo-inverse of W

- Embedding: $Y = CU\Sigma^{\frac{1}{2}}$ where $[U, \Sigma, V] = \mathrm{SVD}(W^{+})$

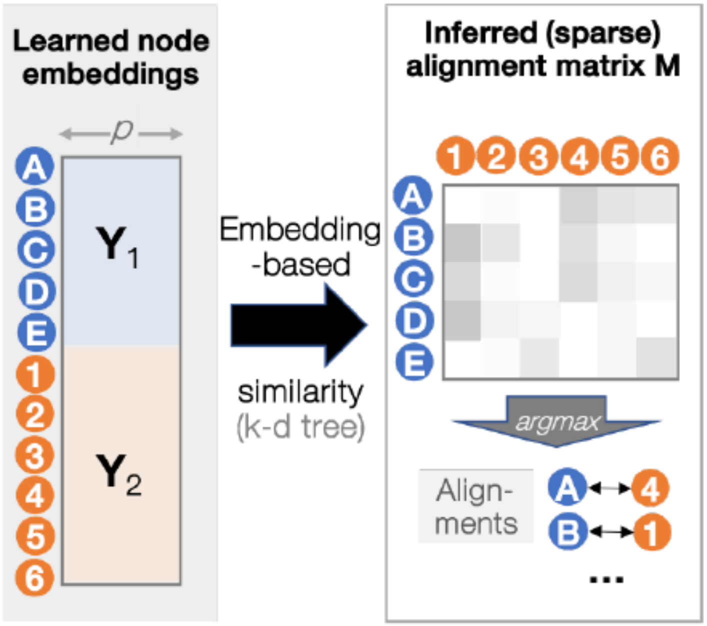


Regal – Alignment Inference

- K-D tree for fast similarity search
- Similarity scores:

$$sim(u,v) = e^{-\|\widetilde{Y}_1[u]-\widetilde{Y}_2[v]\|_2^2}$$

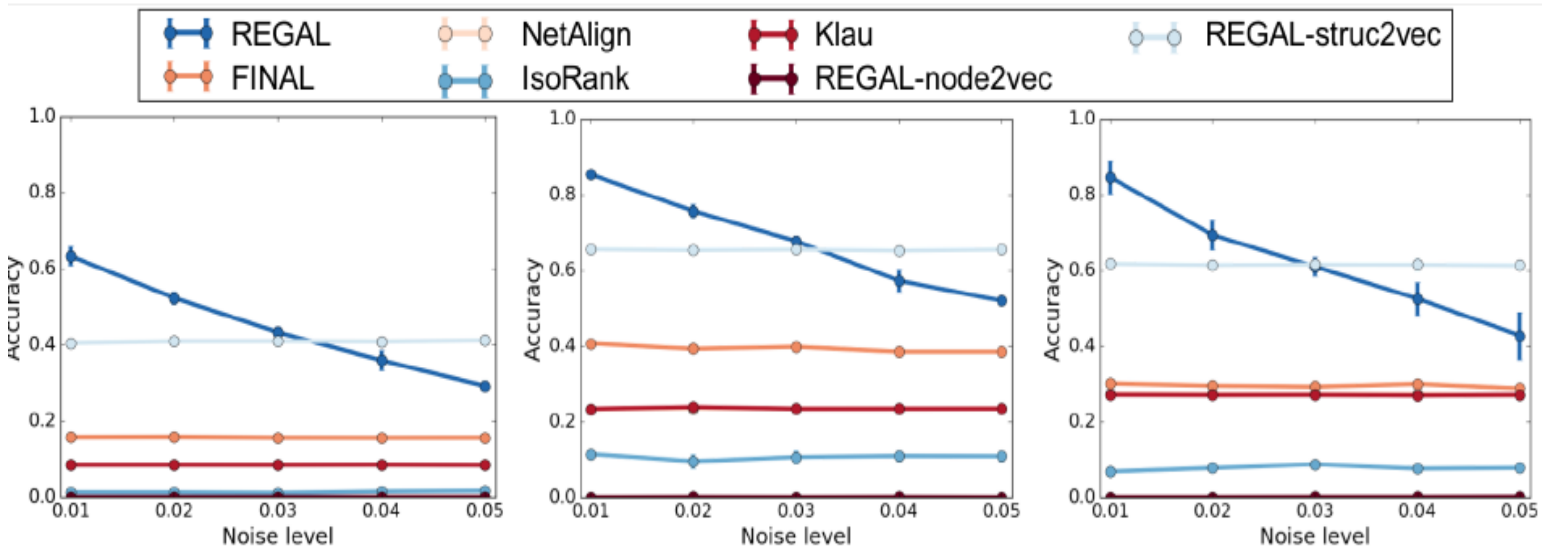
- Complexity:
  - Feature extraction: $O(nKd_{avg}^2)$
  - Node similarity: $O(npb)$
  - Node embedding: $O(np^2)$
  - Alignment: $O(n\log n)$



Heimann, Mark, et al. "Regal: Representation learning-based graph alignment." *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*. 2018.

Regal – Experimental Results

- Data constructions: (1) noisy permutations of one network, (2) synthetic node attributes



(a) **Arxiv** (198 110 **edges**) (b) **PPI** (76 584 **edges**) (c) **Arenas Email** (5 451 **edges**)

Heimann, Mark, et al. "Regal: Representation learning-based graph alignment." *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*. 2018.

Regal – Experimental Results

- Running time:

Dataset	Arxiv	PPI	Arenas
FINAL	4182 (180)	62.88 (32.20)	3.82 (1.41)
NetAlign	149.62 (282.03)	22.44 (0.61)	**1.89 (0.07)**
IsoRank	**17.04 (6.22)**	**6.14 (1.33)**	**0.73 (0.05)**
Klau	1291.00 (373)	476.54 (8.98)	43.04 (0.80)
REGAL-node2vec	709.04 (20.98)	139.56 (1.54)	15.05 (0.23)
REGAL-struc2vec	1975.37 (223.22)	441.35 (13.21)	74.07 (0.95)
REGAL	**86.80 (11.23)**	**18.27 (2.12)**	2.32 (0.31)

Faster computations due to landmark strategy and K-D tree search.

Heimann, Mark, et al. "Regal: Representation learning-based graph alignment." *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*. 2018.

Overview of Part I

Part I: 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

Gromov-Wasserstein Learning for Graph Matching and Node Embedding

- Backgrounds:
  - Networks are often noisy.
  - Many methods learn specific transformations across embeddings of different networks.
- Key question:
  - How to jointly learn node embeddings and infer alignment?
- Benefits of joint problem:
  - Distance between learned node embeddings as auxiliary information of edges → help reduce noise
  - Learn in same manifold → lower risk of model misspecification

Xu, Hongteng, et al. "Gromov-wasserstein learning for graph matching and node embedding." arXiv preprint arXiv:1901.06003 (2019).

GWL - Preliminaries

- Gromov-Wasserstein distance
  - An optimal transport-like distance for metric spaces
  - Calculates distances between pairs of samples of each domain
  - Measures how these distances compare to those in other domains
- Gromov-Wasserstein discrepancy
  - A relaxation by using dissimilarity measurement instead of strict distance metrics
- Metric-measure space of a graph
  - Corresponds to a pair $(\boldsymbol{C}, \boldsymbol{\mu}) \in R^{|V|\times|V|}\times\Sigma^{|V|}$ of a graph $\mathcal{G}$.
  - $\boldsymbol{C} = [c_{ij}]$ represents a node distance/dissimilarity matrix.
  - $\boldsymbol{\mu} = [\mu_i]$ is the empirical distribution of nodes.

Xu, Hongteng, et al. "Gromov-wasserstein learning for graph matching and node embedding." arXiv preprint arXiv:1901.06003 (2019).

GWL - Gromov-Wasserstein Learning Framework

- Gromov-Wasserstein discrepancy between graphs
  - Given $\mathcal{G}_s$ and $\mathcal{G}_t$, the discrepancy between $(\boldsymbol{C}_s, \boldsymbol{\mu}_s)$ and $(\boldsymbol{C}_t, \boldsymbol{\mu}_t)$

$$\begin{aligned} d_{GW}(\boldsymbol{\mu}_s, \boldsymbol{\mu}_t) &:= \min_{\boldsymbol{T}\in\Pi(\boldsymbol{\mu}_s,\boldsymbol{\mu}_t)} \sum\nolimits_{i,j,i',j'} L(c^s_{ij}, c^t_{i'j'})T_{ii'}T_{jj'} \\ &= \min_{\boldsymbol{T}\in\Pi(\boldsymbol{\mu}_s,\boldsymbol{\mu}_t)} \langle \boldsymbol{L}(\boldsymbol{C}_s, \boldsymbol{C}_t, \boldsymbol{T}), \boldsymbol{T}\rangle. \end{aligned}$$

  - $L(\cdot,\cdot)$: element-wise loss, e.g., mean square or KL-divergence
  - $\boldsymbol{T}$: optimal transport between nodes of two networks, indicating probabilities of alignment
  - $\boldsymbol{L}_{jj'} = \sum_{i,i'} L\left(c^s_{ij}, c^t_{i'j'}\right)\boldsymbol{T}_{ii'}$
  - $\boldsymbol{L}(\boldsymbol{C}_s, \boldsymbol{C}_t, \boldsymbol{T}) = \left[\boldsymbol{L}_{jj'}\right] \in R^{|V_s|\times|V_t|}$

Xu, Hongteng, et al. "Gromov-wasserstein learning for graph matching and node embedding." arXiv preprint arXiv:1901.06003 (2019).

GWL - Gromov-Wasserstein Learning Framework

- Proposed model
  - Use node embeddings $\boldsymbol{X}_s, \boldsymbol{X}_t$ for dissimilarity matrices

$$\min_{\boldsymbol{X}_s, \boldsymbol{X}_t} \min_{\boldsymbol{T} \in \Pi(\boldsymbol{\mu}_s, \boldsymbol{\mu}_t)} \underbrace{\langle \boldsymbol{L}(\boldsymbol{C}_s(\boldsymbol{X}_s), \boldsymbol{C}_t(\boldsymbol{X}_t), \boldsymbol{T}),\ \boldsymbol{T} \rangle}_{\text{Gromov-Wasserstein discrepancy}} + \underbrace{\alpha \langle \boldsymbol{K}(\boldsymbol{X}_s, \boldsymbol{X}_t),\ \boldsymbol{T} \rangle}_{\text{Wasserstein discrepancy}} + \underbrace{\beta R(\boldsymbol{X}_s, \boldsymbol{X}_t)}_{\text{prior information}}.$$

  - $\boldsymbol{C}_s(\boldsymbol{X}_s) = (1-\alpha)\boldsymbol{C}_s + \alpha \boldsymbol{K}(\boldsymbol{X}_s, \boldsymbol{X}_s)$ where $\boldsymbol{C}_s$ is computed by edge weights and $\boldsymbol{K}(\boldsymbol{X}_s, \boldsymbol{X}_s)$ measures distance within same network based on node embedding.
  - $\boldsymbol{R}(\boldsymbol{X}_s, \boldsymbol{X}_t) = \sum_{k=s,t} L(\boldsymbol{K}(\boldsymbol{X}_k, \boldsymbol{X}_k), \boldsymbol{C}_k) \boxed{+ L(\boldsymbol{K}(\boldsymbol{X}_s, \boldsymbol{X}_t), \boldsymbol{C}_{st})}$

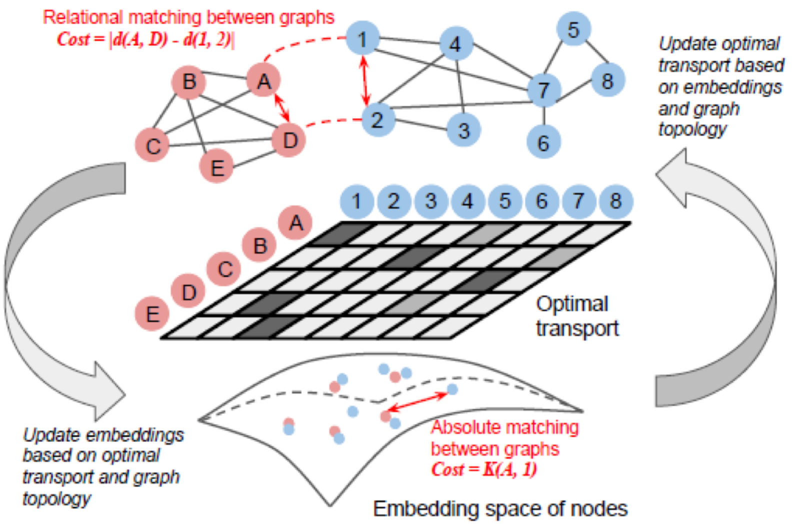
Optional when given labeled alignment

GWL – Learning Algorithm

- Alternatively learn optimal transport and embedding
- Learning optimal transport
  - Proximal point method

$$\min_{\boldsymbol{T}\in\Pi(\boldsymbol{\mu}_s,\boldsymbol{\mu}_t)} \langle L(C_s(\boldsymbol{X}_s^{(m)}), C_t(\boldsymbol{X}_t^{(m)}), \boldsymbol{T}), \boldsymbol{T}\rangle + \alpha\langle K(\boldsymbol{X}_s^{(m)}, \boldsymbol{X}_t^{(m)}),\ \boldsymbol{T}\rangle + \boxed{\gamma \mathrm{KL}(\boldsymbol{T}\|\boldsymbol{T}^{(n)})}$$

A proximal term based on KL-divergence



- Updating embeddings
  - Given optimal transport $\widehat{\boldsymbol{T}}^{(m)}$, solve by gradient descent

$$\min_{\boldsymbol{X}_s,\boldsymbol{X}_t} \alpha_m\langle K(\boldsymbol{X}_s, \boldsymbol{X}_t),\ \widehat{\boldsymbol{T}}^{(m)}\rangle + \beta R(\boldsymbol{X}_s, \boldsymbol{X}_t)$$

Xu, Hongteng, et al. "Gromov-wasserstein learning for graph matching and node embedding." arXiv preprint arXiv:1901.06003 (2019).

GWL – Experimental Results

- Communication network alignment
  - Dataset: MC3 used in the Mini-Challenge 3 of VAST Challenge 2018
- Model Variants:
  - GWL-C and GWL-R: use cosine and RBF distance on embeddings
  - GWD: no embedding -based distance

Method	Call→Email (Sparse) Node Correctness (%)	Call→Email (Dense) Node Correctness (%)
GAA	34.22	0.53
LRSA	38.20	2.93
TAME	37.39	2.67
GRAAL	39.67	0.48
MI-GRAAL	35.53	0.64
MAGNA++	7.88	0.09
HugAlign	36.21	3.86
NETAL	36.87	1.77
GWD	23.16±0.46	1.77±0.22
GWL-R	39.64±0.57	3.80±0.23
GWL-C	**40.45**±0.53	**4.23**±0.27

Xu, Hongteng, et al. "Gromov-wasserstein learning for graph matching and node embedding." arXiv preprint arXiv:1901.06003 (2019).

GWL – Experimental Results

- Procedure recommendation
  - Dataset: MIMIC-III dataset
  - Goal: recommend suitable procedures for patients, according to their disease characteristics.

Method	Top-1 (%)			Top-5 (%)		
	P	R	F1	P	R	F1
Word2Vec	39.95	13.27	18.25	28.89	46.98	32.59
GloVe	32.66	13.01	17.22	27.93	44.79	31.47
DWL (Scratch)	37.89	12.42	17.16	27.39	43.81	30.81
DWL (Finetune)	40.00	13.76	18.71	30.59	**48.56**	34.28
GWD-R	46.29	17.01	22.32	31.82	43.81	33.77
GWD-C	43.16	15.79	20.77	31.42	42.99	33.25
GWL-R	46.20	16.93	22.22	32.03	44.75	34.18
GWL-C	**47.46**	**17.25**	**22.71**	**32.09**	45.64	**34.31**

Xu, Hongteng, et al. "Gromov-wasserstein learning for graph matching and node embedding." arXiv preprint arXiv:1901.06003 (2019).

Overview of Part I

Part I: 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

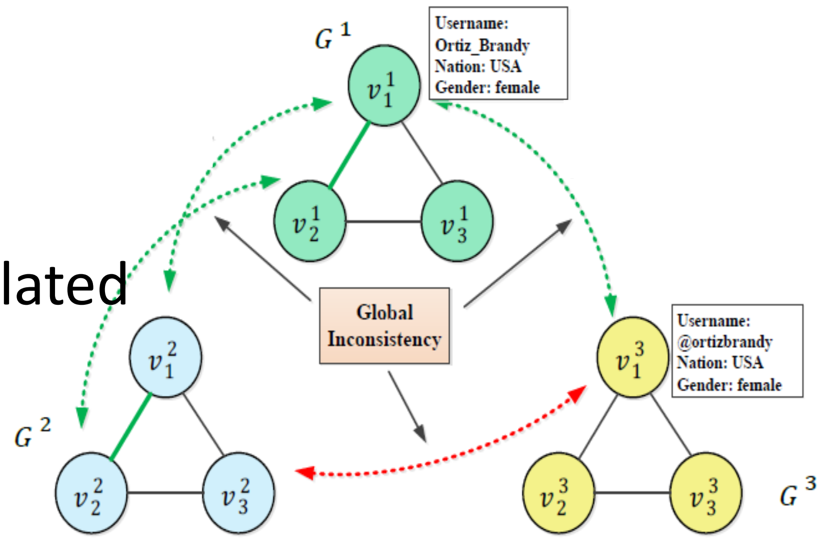
Collective NA
- **Consistency-based**
  - **w/o attributes**
  - **w/ attributes**
- Embedding-based
  - w/o attributes

Higher-Order NA
- Consistency-based
  - Single-level
  - Multilevel

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

Collective Network Alignment

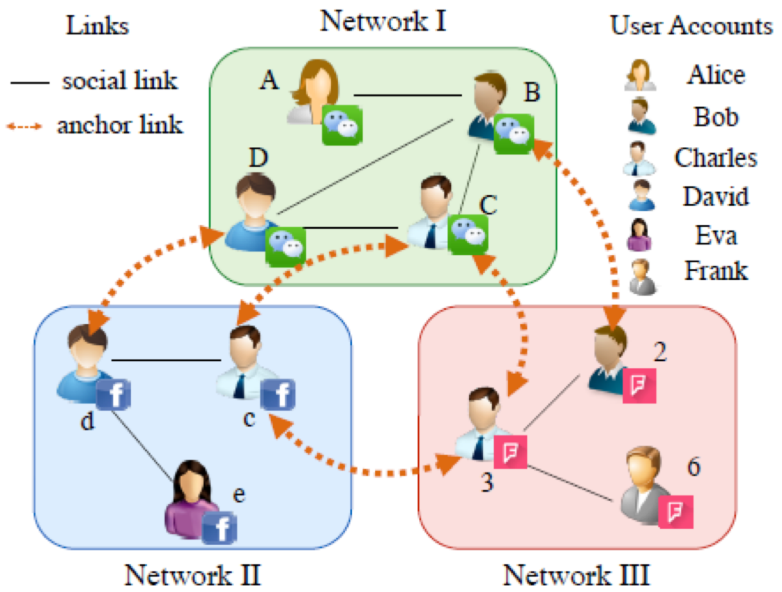
- Goal: to find alignment across **multiple** networks
- Possible solution
  - Find pairwise alignment
  - Then combine
  - Transitivity constraint may be violated



- Problem setting:
  - **Given:** **more than two** networks $\mathcal{G} = \{\mathcal{G}_1, \cdots, \mathcal{G}_m\}$
  - **Find:** alignment across $\mathcal{G}_i, \mathcal{G}_j$ $(i, j = 1, \cdots, m)$ jointly

Multiple Anonymized Social Networks Alignment

- Goal: to find anchor links/alignment across multiple networks without attributes
- Key challenge: how to preserve transitivity property



Zhang, Jiawei, and S. Yu Philip. "Multiple anonymized social networks alignment." 2015 IEEE International Conference on Data Mining. IEEE, 2015.

UMA – Unsupervised Pairwise Alignment

- Key idea: to minimize the alignment inconsistency
  - I.e., the number of non-shared edges between those mapped from $\mathcal{G}^{(i)}$ and those in $\mathcal{G}^{(j)}$
- Mathematical formulation

$$\bar{\mathbf{T}}^{(i,j)} = \arg\min_{\mathbf{T}^{(i,j)}} \left\| (\mathbf{T}^{(i,j)})^\top \mathbf{S}^{(i)} \mathbf{T}^{(i,j)} - \mathbf{S}^{(j)} \right\|_F^2$$

$$s.t. \quad \mathbf{T}^{(i,j)} \in \{0,1\}^{|\mathcal{U}^{(i)}| \times |\mathcal{U}^{(j)}|},$$

$$\boxed{\begin{aligned} &\mathbf{T}^{(i,j)} \mathbf{1}^{|\mathcal{U}^{(j)}| \times 1} \preccurlyeq \mathbf{1}^{|\mathcal{U}^{(i)}| \times 1}, \\ &(\mathbf{T}^{(i,j)})^\top \mathbf{1}^{|\mathcal{U}^{(i)}| \times 1} \preccurlyeq \mathbf{1}^{|\mathcal{U}^{(j)}| \times 1}, \end{aligned}}$$

one-to-one mapping constraints

  - $\boldsymbol{S}^{(i)}, \boldsymbol{S}^{(j)}$: adjacency matrices of networks $\mathcal{G}^{(i)}$ and $\mathcal{G}^{(j)}$
  - $\boldsymbol{T}^{(i,j)}$: alignment matrix from $\mathcal{G}^{(i)}$ to $\mathcal{G}^{(j)}$

Zhang, Jiawei, and S. Yu Philip. "Multiple anonymized social networks alignment." 2015 IEEE International Conference on Data Mining. IEEE, 2015.

UMA – Transitivity Penalties

- Measure the number of inconsistent edges between the mapped from $\mathcal{G}^{(i)} \rightarrow \mathcal{G}^{(j)} \rightarrow \mathcal{G}^{(k)}$ and $\mathcal{G}^{(i)} \rightarrow \mathcal{G}^{(k)}$

- Mathematical formulation

$$C(\{G^{(i)}, G^{(j)}, G^{(k)}\})$$
$$= \left\| (\mathbf{T}^{(j,k)})^\top (\mathbf{T}^{(i,j)})^\top \mathbf{S}^{(i)} \mathbf{T}^{(i,j)} \mathbf{T}^{(j,k)} - (\mathbf{T}^{(i,k)})^\top \mathbf{S}^{(i)} \mathbf{T}^{(i,k)} \right\|_F^2$$

- Extension to n $(n \geq 3)$ networks

$$C(\{G^{(1)}, G^{(2)}, \cdots, G^{(n)}\})$$
$$= \sum_{\forall \{G^{(i)}, G^{(j)}, G^{(k)}\} \subset \{G^{(1)}, G^{(2)}, \cdots, G^{(n)}\}} C(\{G^{(i)}, G^{(j)}, G^{(k)}\})$$

Zhang, Jiawei, and S. Yu Philip. "Multiple anonymized social networks alignment." 2015 IEEE International Conference on Data Mining. IEEE, 2015.

UMA – Optimization Problem

- **Objective:** to minimize the alignment inconsistency and transitivity penalties simultaneously
- Mathematical formulation

$$
\begin{aligned}
&\bar{\mathbf{T}}^{(i,j)}, \bar{\mathbf{T}}^{(j,k)}, \bar{\mathbf{T}}^{(k,i)} \\
&= \arg\min_{\mathbf{T}^{(i,j)}, \mathbf{T}^{(j,k)}, \mathbf{T}^{(k,i)}} \left\| (\mathbf{T}^{(i,j)})^\top \mathbf{S}^{(i)} \mathbf{T}^{(i,j)} - \mathbf{S}^{(j)} \right\|_F^2 \\
&+ \left\| (\mathbf{T}^{(j,k)})^\top \mathbf{S}^{(j)} \mathbf{T}^{(j,k)} - \mathbf{S}^{(k)} \right\|_F^2 + \left\| (\mathbf{T}^{(k,i)})^\top \mathbf{S}^{(k)} \mathbf{T}^{(k,i)} - \mathbf{S}^{(i)} \right\|_F^2 \\
&+ \alpha \cdot \left\| (\mathbf{T}^{(j,k)})^\top (\mathbf{T}^{(i,j)})^\top \mathbf{S}^{(i)} \mathbf{T}^{(i,j)} \mathbf{T}^{(j,k)} - \mathbf{T}^{(k,i)} \mathbf{S}^{(i)} (\mathbf{T}^{(k,i)})^\top \right\|_F^2 \\
&s.t.\ \mathbf{T}^{(i,j)} \in \{0,1\}^{|\mathcal{U}^{(i)}| \times |\mathcal{U}^{(j)}|}, \mathbf{T}^{(j,k)} \in \{0,1\}^{|\mathcal{U}^{(j)}| \times |\mathcal{U}^{(k)}|} \\
&\quad \mathbf{T}^{(k,i)} \in \{0,1\}^{|\mathcal{U}^{(k)}| \times |\mathcal{U}^{(i)}|} \\
&\quad \mathbf{T}^{(i,j)} \mathbf{1}^{|\mathcal{U}^{(j)}| \times 1} \preccurlyeq \mathbf{1}^{|\mathcal{U}^{(i)}| \times 1}, (\mathbf{T}^{(i,j)})^\top \mathbf{1}^{|\mathcal{U}^{(i)}| \times 1} \preccurlyeq \mathbf{1}^{|\mathcal{U}^{(j)}| \times 1}, \\
&\quad \mathbf{T}^{(j,k)} \mathbf{1}^{|\mathcal{U}^{(k)}| \times 1} \preccurlyeq \mathbf{1}^{|\mathcal{U}^{(j)}| \times 1}, (\mathbf{T}^{(j,k)})^\top \mathbf{1}^{|\mathcal{U}^{(j)}| \times 1} \preccurlyeq \mathbf{1}^{|\mathcal{U}^{(k)}| \times 1}, \\
&\quad \mathbf{T}^{(k,i)} \mathbf{1}^{|\mathcal{U}^{(i)}| \times 1} \preccurlyeq \mathbf{1}^{|\mathcal{U}^{(k)}| \times 1}, (\mathbf{T}^{(k,i)})^\top \mathbf{1}^{|\mathcal{U}^{(k)}| \times 1} \preccurlyeq \mathbf{1}^{|\mathcal{U}^{(i)}| \times 1},
\end{aligned}
$$

Alignment inconsistency

Transitivity penalties

One-to-one constraints

Relaxations

Linear constraint + L1 norm

Zhang, Jiawei, and S. Yu Philip. "Multiple anonymized social networks alignment." 2015 IEEE International Conference on Data Mining. IEEE, 2015.

UMA – Transitive Network Matching

- Goal: to solve for binary variable $x_{l,m}^{(i,j)}$ indicating whether node u_l in $\mathcal{G}^{(i)}$ is aligned with node u_m in $\mathcal{G}^{(j)}$
- Optimization problem
  - Select high scores in alignment
  - One-to-one constraint
  - Transitivity constraint

$$\max_{\mathbf{x}^{(i,j)},\mathbf{x}^{(j,k)},\mathbf{x}^{(k,i)}} \sum_{l,m} x_{l,m}^{(i,j)} \mathbf{T}^{(i,j)}(l,m) + \sum_{l,m} x_{l,m}^{(i,j)} \mathbf{T}^{(i,j)}(l,m) + \sum_{l,m} x_{l,m}^{(i,j)} \mathbf{T}^{(i,j)}(l,m),$$

$$s.t. \sum_{u_m^{(j)} \in \mathcal{U}^{(j)}} x_{l,m}^{(i,j)} \leq 1, \sum_{u_o^{(k)} \in \mathcal{U}^{(k)}} x_{l,o}^{(i,k)} \leq 1, \forall u_l^{(i)} \in \mathcal{U}^{(i)},$$

$$\sum_{u_l^{(i)} \in \mathcal{U}^{(i)}} x_{m,l}^{(j,i)} \leq 1, \sum_{u_o^{(k)} \in \mathcal{U}^{(k)}} x_{m,o}^{(j,k)} \leq 1, \forall u_m^{(j)} \in \mathcal{U}^{(j)},$$

$$\sum_{u_l^{(i)} \in \mathcal{U}^{(i)}} x_{o,l}^{(k,i)} \leq 1, \sum_{u_m^{(j)} \in \mathcal{U}^{(j)}} x_{o,m}^{(k,j)} \leq 1, \forall u_o^{(k)} \in \mathcal{U}^{(k)}$$

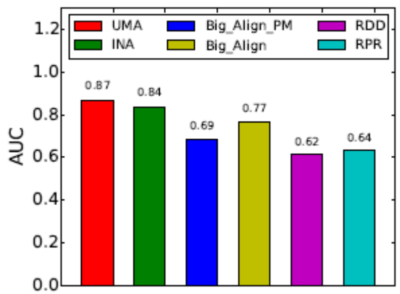
$$x_{l,m}^{(i,j)} + x_{m,o}^{(j,k)} + x_{o,l}^{(k,i)} \neq 2, \forall l \in \{1, 2, \cdots, |\mathcal{U}^{(i)}|\},$$
$$\forall m \in \{1, 2, \cdots, |\mathcal{U}^{(j)}|\}, \forall o \in \{1, 2, \cdots, \mathcal{U}^{(k)}|,$$

$$x_{l,m}^{(i,j)} \in \{0,1\}, \forall u_l^{(i)} \in \mathcal{U}^{(i)}, u_m^{(j)} \in \mathcal{U}^{(j)}.$$
$$x_{m,o}^{(j,k)} \in \{0,1\}, \forall u_m^{(j)} \in \mathcal{U}^{(j)}, u_o^{(k)} \in \mathcal{U}^{(k)}.$$
$$x_{o,l}^{(k,i)} \in \{0,1\}, \forall u_o^{(k)} \in \mathcal{U}^{(k)}, u_l^{(i)} \in \mathcal{U}^{(i)}.$$

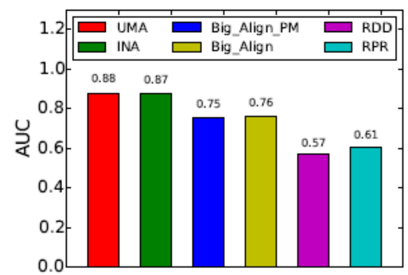
Zhang, Jiawei, and S. Yu Philip. "Multiple anonymized social networks alignment." 2015 IEEE International Conference on Data Mining. IEEE, 2015.

UMA – Experimental Results

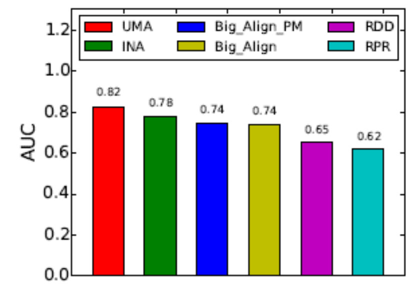
- Dataset: Stack Overflow, Super User and Programmers
- Alignment performance



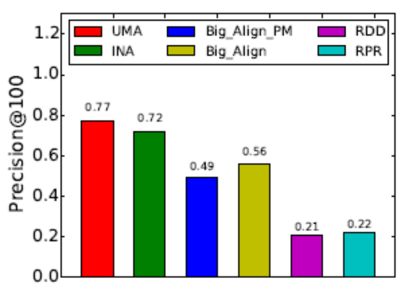
(a) AUC ($G^{(i)}$, $G^{(j)}$)



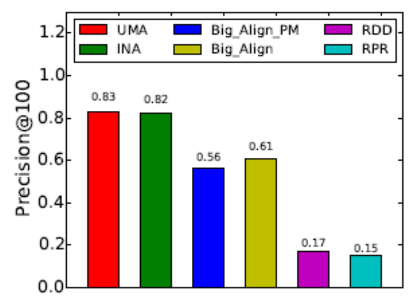
(b) AUC ($G^{(j)}$, $G^{(k)}$)



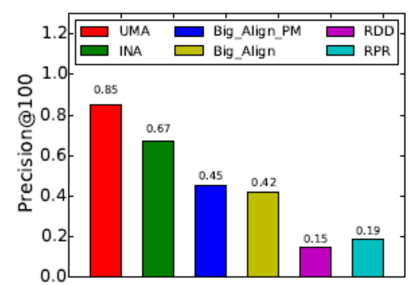
(c) AUC ($G^{(k)}$, $G^{(i)}$)



(d) Precision@100 ($G^{(i)}$, $G^{(j)}$)



(e) Precision@100 ($G^{(j)}$, $G^{(k)}$)



(f) Precision@100 ($G^{(k)}$, $G^{(i)}$)

Zhang, Jiawei, and S. Yu Philip. "Multiple anonymized social networks alignment." 2015 IEEE International Conference on Data Mining. IEEE, 2015.

COSNET: Connecting Social Networks with Local and Global Consistency

- Intuitions: binary classification over node pairs
  - Instances: node pairs $X = \{\boldsymbol{x}_i\}$
  - Labels: $Y = \{y_i\}$, $y_i = 1$ if $\boldsymbol{x}_i$ refers to same node, otherwise 0

- Factors considered:
  - Node feature consistency (e.g., user profiles)
  - Structural consistency
  - Global consistency (i.e., transitivity constraints)

Zhang, Yutao, et al. "Cosnet: Connecting heterogeneous social networks with local and global consistency." *Proceedings of the 21st ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 2015.

COSNET – Node Feature Consistency

- Intuition: to encode the feature similarity for $\boldsymbol{x}_i$
- Formulation:

$$E_l(Y, X) = \sum_i \mathbf{w}_l^\intercal \mathbf{g}_l(\mathbf{x}_i, y_i)$$

  - $\boldsymbol{g}_l(\boldsymbol{x}_i, y_i)$ is a vector-valued feature function
    - Encodes the user profile similarity for node pair x_i
  - w_l is the model parameter

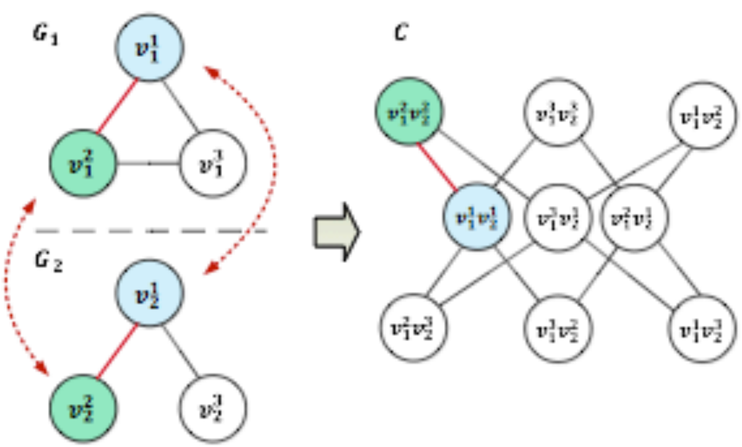
Zhang, Yutao, et al. "Cosnet: Connecting heterogeneous social networks with local and global consistency." *Proceedings of the 21st ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 2015.

COSNET – Structural Consistency

- Intuition:
  - If two nodes are aligned, their neighbors are likely to be aligned
- Matching graph $MG = (V_{MG}, E_{MG})$
  - Same as Kronecker product graph
- Pairwise formulation:

$$E_e(Y, X) = \sum_{\langle \mathbf{x}_i, \mathbf{x}_j \rangle \in E_{MG}} \mathbf{w}_e^{\intercal} \mathbf{f}_e(y_i, y_j)$$

$$\mathbf{f}_e(y_i, y_j) = \begin{cases} (1,0,0)^{\intercal} & \text{if } y_i = y_j = 0 \\ (0,1,0)^{\intercal} & \text{if } y_i + y_j = 1 \\ (0,0,1)^{\intercal} & \text{if } y_i = y_j = 1 \end{cases}$$



COSNET – Global Consistency Violation

DEFINITION 2 (GLOBAL INCONSISTENCY). *Given a set of social networks* $\mathbf{G}$, *a set of user pairs* X *and the corresponding labels* Y, *if there exists a sequence of user pairs* $\langle \mathbf{x}_{i_1}, \mathbf{x}_{i_2}, \cdots, \mathbf{x}_{i_n} \rangle$, *such that*

$$\forall i = i_1, i_2, \cdots, i_n, y_i = 1$$

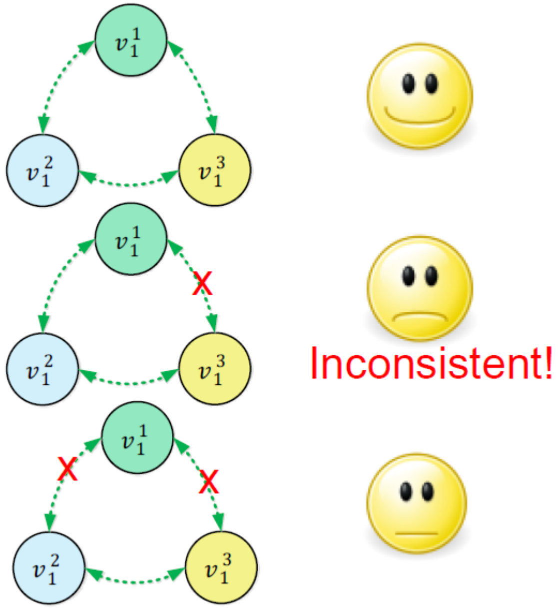
and

$$\forall k = 1, 2, \cdots, n-1, \mathcal{V}_{i_k}^2 = \mathcal{V}_{i_{k+1}}^1$$

and

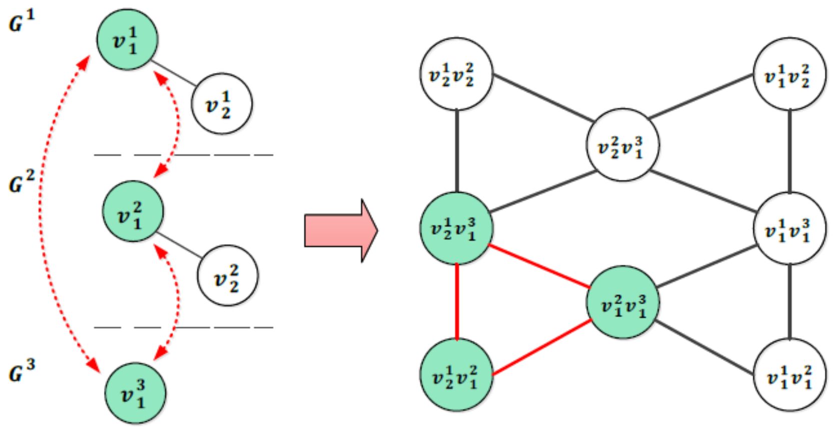
For the pair $\langle \mathcal{V}_{i_n}^2, \mathcal{V}_1^1 \rangle$, *the corresponding label* $y_j = 0$

then we say that the assigned labels Y *causes global inconsistency given* $\mathbf{G}$ *and* X.



COSNET – Global Consistency

- Triadic closure in the matching graph



- Formulation:

$$E_t(Y, X) = \sum_{c \in T_{MG}} \mathbf{w}_t^\intercal \mathbf{f}_t(Y_c)$$

$$\mathbf{f}_t(y_i, y_j) = \begin{cases} (1,0,0,0)^\intercal & \text{if } |Y_c| = 0 \\ (0,1,0,0)^\intercal & \text{if } |Y_c| = 1 \\ (0,0,1,0)^\intercal & \text{if } |Y_c| = 2 \\ (0,0,0,1)^\intercal & \text{if } |Y_c| = 3 \end{cases}$$

Zhang, Yutao, et al. "Cosnet: Connecting heterogeneous social networks with local and global consistency." *Proceedings of the 21st ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 2015.

COSNET – Model Learning

- Objective function:

$$
\begin{aligned}
E(Y,X) &= \sum_{\mathbf{x}_i \in V_{MG}} \mathbf{w}_l^\intercal \mathbf{g}_l(\mathbf{x}_i, y_i) + \sum_{\langle \mathbf{x}_i, \mathbf{x}_j \rangle \in E_{MG}} \mathbf{w}_e^\intercal \mathbf{f}_e(y_i, y_j) \\
&+ \sum_{c \in T_{MG}} \mathbf{w}_t^\intercal \mathbf{f}_t(Y_c)
\end{aligned}
$$

- Define distance of two matching configurations Y and Y'

$$
\begin{aligned}
\Delta(Y,Y') &= \sum_{\mathbf{x}_i \in V_{MG}} \delta_l(y_i, y_i') + \sum_{c \in T_{MG}} \delta_c(Y_c, Y_c') + \\
& \sum_{\langle \mathbf{x}_i, \mathbf{x}_j \rangle \in E_{MG}} \delta_e(\langle y_i, y_j \rangle, \langle y_i', y_j' \rangle)
\end{aligned}
$$

Hamming distance

Zhang, Yutao, et al. "Cosnet: Connecting heterogeneous social networks with local and global consistency." *Proceedings of the 21st ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 2015.

COSNET – Model Learning

- By max-margin theory:

$$\min_{W} \frac{1}{2}||W||^2 + \mu\xi$$
$$\text{s.t.} \quad E(\hat{Y}, X; W) \leq E(Y, X; W) - \Delta(Y, \hat{Y}) + \xi$$

  - $\hat{Y}, Y$: input labeled configuration and learned configuration
  - $W = (\boldsymbol{w}_l, \boldsymbol{w}_e, \boldsymbol{w}_t)$: model parameters
  - ξ: slack variable to handle non-separable data
  - μ: trade-off between the maximum margin & error penalty
  - Constraint: distance between the energy of $\hat{Y}, Y$ at least $\Delta(\hat{Y}, Y)$

Zhang, Yutao, et al. "Cosnet: Connecting heterogeneous social networks with local and global consistency." *Proceedings of the 21st ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 2015.

COSNET – Model Learning

- The original problem is intractable.
- Use Lagrangian relaxation for dual decomposition

$$\min_{W,\boldsymbol{\lambda}} \frac{1}{2}||W||^2 + \mu(E(\hat{Y}, X; W) - \max_{\boldsymbol{\lambda}} L(Y, X, \boldsymbol{\lambda}; W))$$

$$\text{s.t.} \quad \sum_{y_i \in Y_i} \lambda_i^f = 0, \ \ \forall f \in \mathcal{F}$$

  - $f \in \mathcal{F}$: factor functions
  - $\boldsymbol{\lambda}$: Lagrange multipliers
- Convex and non-differentiable
- Solution: projected sub-gradient method

Zhang, Yutao, et al. "Cosnet: Connecting heterogeneous social networks with local and global consistency." *Proceedings of the 21st ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 2015.

COSNET – Public Dataset

- Data statistics

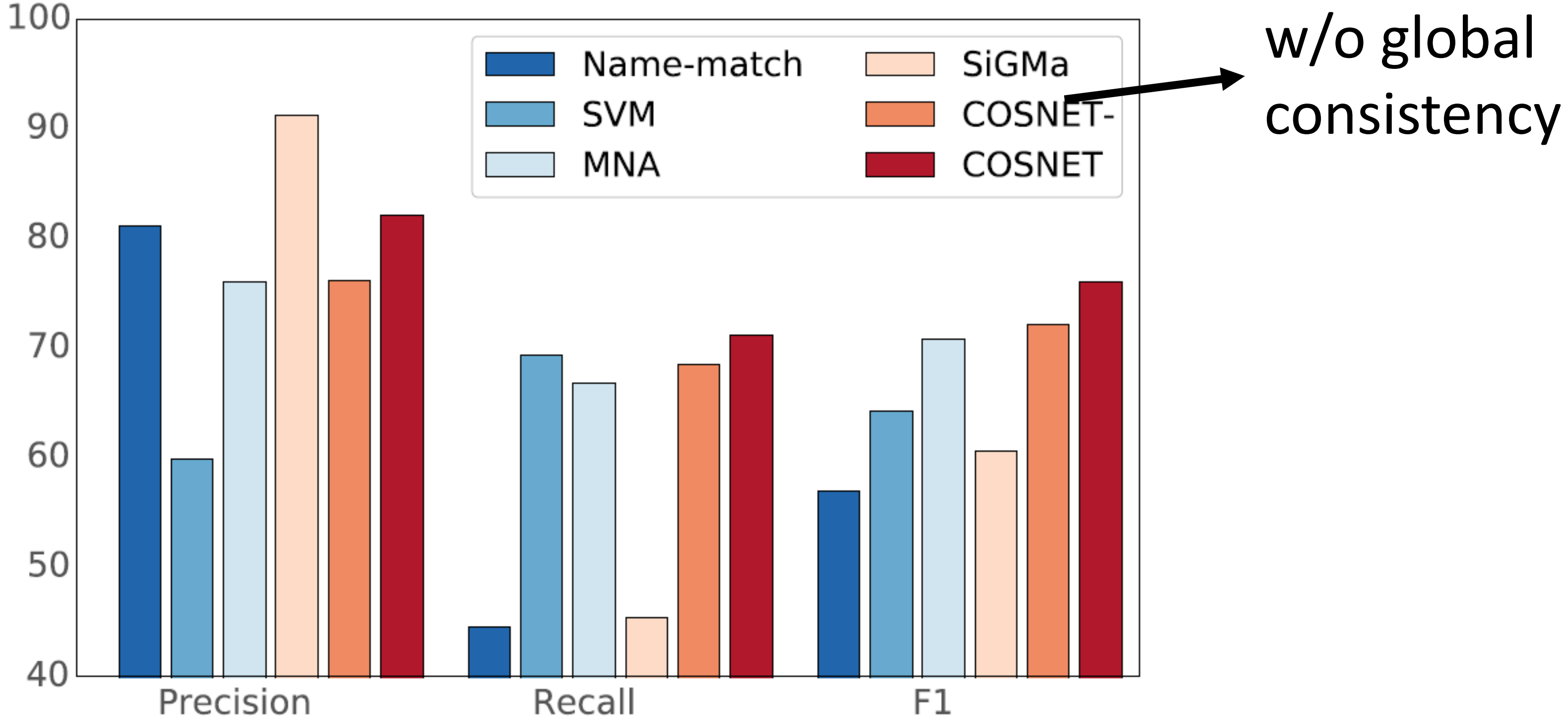
Dataset	Network	#Users	#Relationships
SNS	Twitter	40,171,624	1,468,365,182
	LiveJournal	3,017,286	87,037,567
	Flickr	215,495	9,114,557
	Last.fm	136,420	1,685,524
	MySpace	854,498	6,489,736
Academia	LinkedIn	2,985,414	25,965,384
	ArnetMiner	1,053,188	3,916,907
	VideoLectures	11,178	786,353

- Link: https://www.aminer.cn/cosnet

Zhang, Yutao, et al. "Cosnet: Connecting heterogeneous social networks with local and global consistency." *Proceedings of the 21st ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 2015.

COSNET – Experimental Results

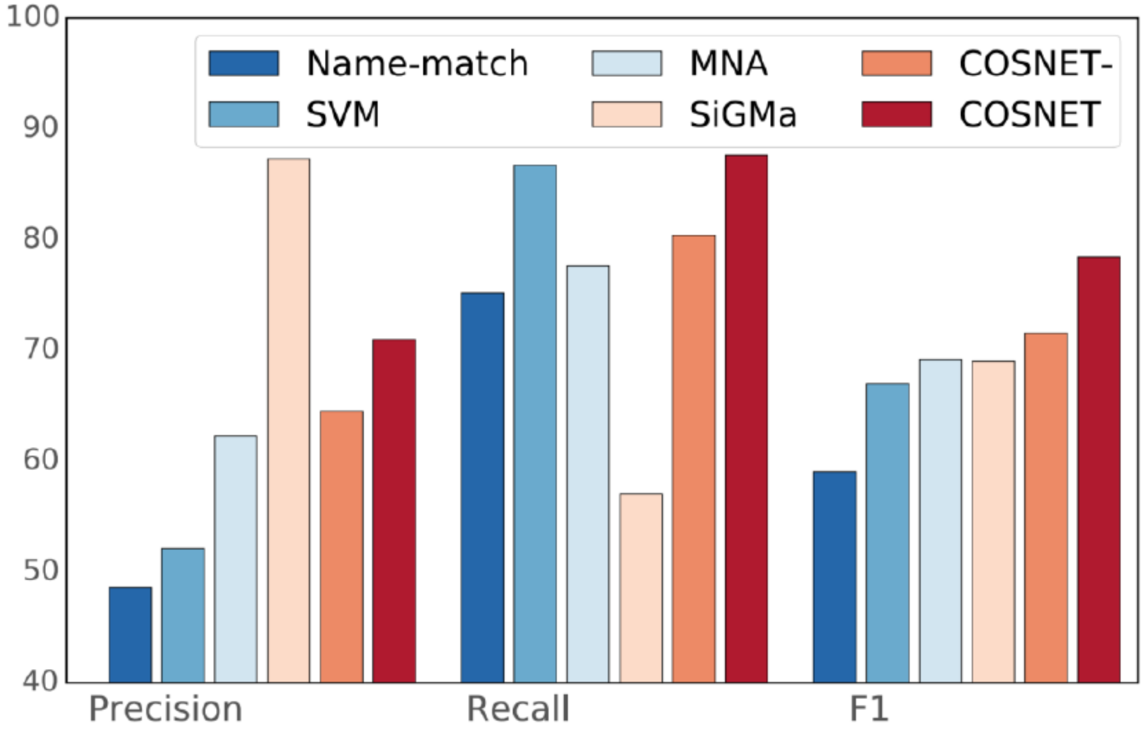
- Connecting social media sites
  - Twitter, LiveJournal, Last.fm, Flickr, MySpace



Zhang, Yutao, et al. "Cosnet: Connecting heterogeneous social networks with local and global consistency." *Proceedings of the 21st ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 2015.

COSNET – Experimental Results

- Connecting Aminer with LinkedIn and VideoLectures



Zhang, Yutao, et al. "Cosnet: Connecting heterogeneous social networks with local and global consistency." *Proceedings of the 21st ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 2015.

Overview of Part I

Part I: 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

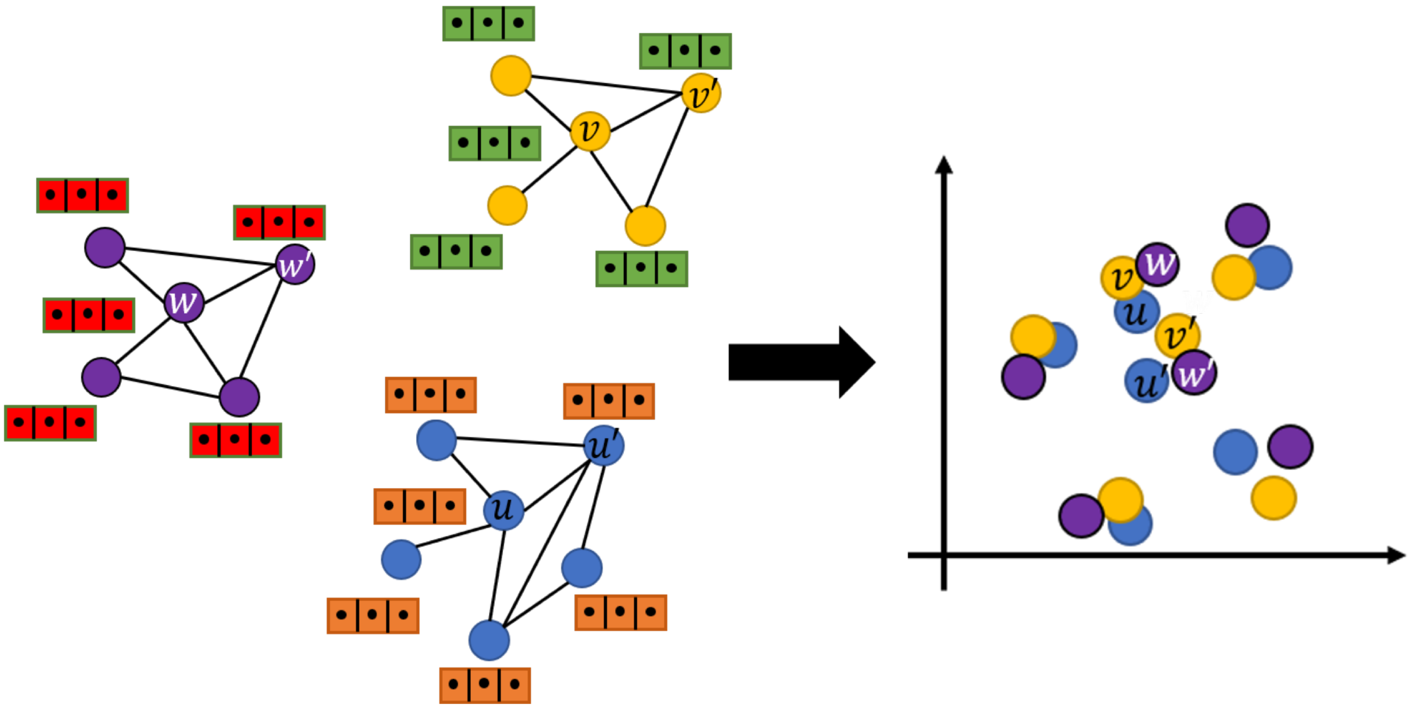
- Consistency-based
  - Single-level
  - Multilevel

Related Tasks

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

Embedding-Based Collective Network Alignment

- Goal: to learn node embeddings that can infer alignment in the embedding space



Cross-Network Embedding for Multi-Network Alignment

- Motivations: networks heterogeneity
  - Different networks may own different semantic meanings;
  - Same node may have distinct embeddings in different networks

- Goal: to learn node embeddings for multiple network alignment

- Key question: how to capture the commonness among anchor node counterparts and specific semantics in different networks?

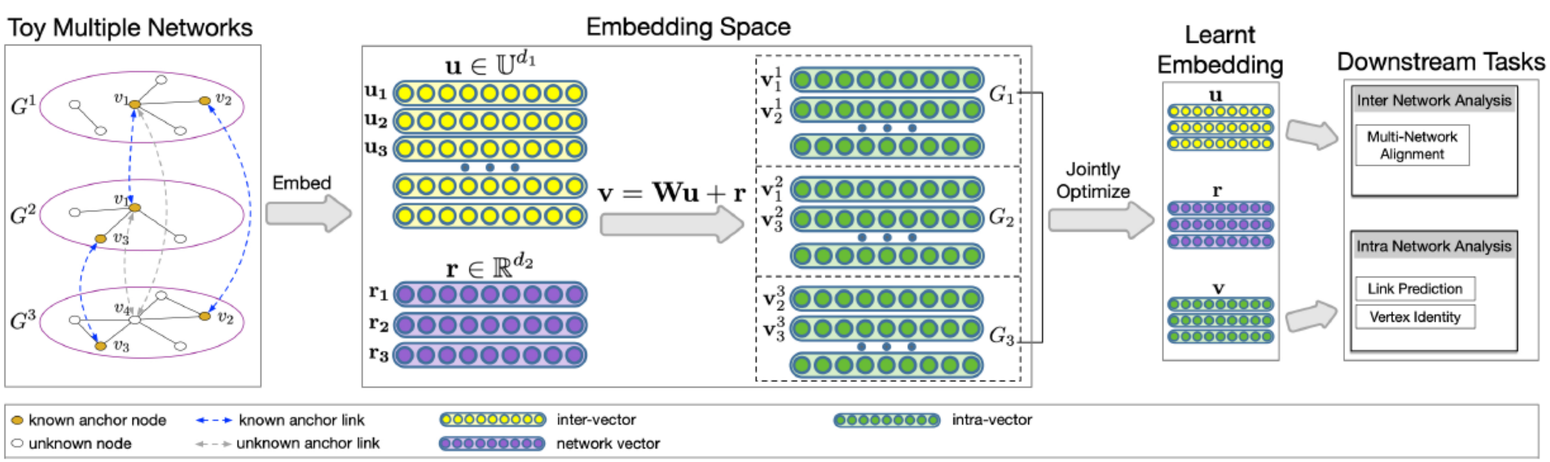
Chu, Xiaokai, et al. "Cross-network embedding for multi-network alignment." The World Wide Web Conference. 2019.

CrossMNA – Cross Network Embedding

- Key idea: split node embedding into two components

$$\boldsymbol{v}_i^k = \boldsymbol{W}\boldsymbol{u}_i + \boldsymbol{r}^k$$

  - Intra-vector $\boldsymbol{v}_i^k$: captures structural information in a network
  - Inter-vector $\boldsymbol{u}_i$: captures the commonness of anchor node
  - Network vector $\boldsymbol{r}^k$: captures network-specific semantics



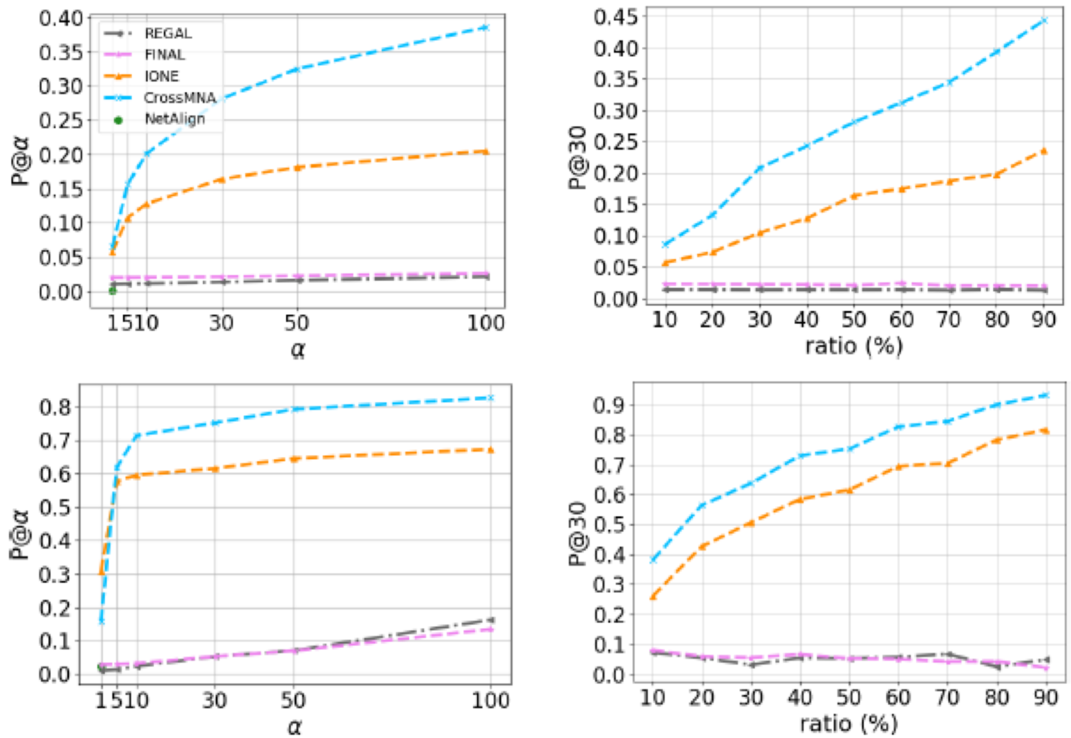
Chu, Xiaokai, et al. "Cross-network embedding for multi-network alignment." The World Wide Web Conference. 2019.

CrossMNA – Experimental Results

- Multiple network alignment

Twitter dataset

Arxiv dataset



Precision@α vs. α Precision@30 vs. training ratio

Chu, Xiaokai, et al. "Cross-network embedding for multi-network alignment." The World Wide Web Conference. 2019.

CrossMNA – Experimental Results

- Multiple network link prediction

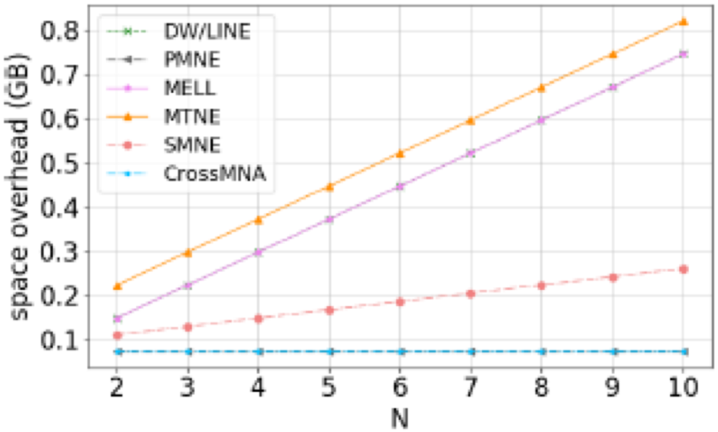
Dataset	arXiv			SacchCere			Twitter		
	30%	50%	80%	30%	50%	80%	30%	50%	80%
DeepWalk	87.86	94.41	98.12	69.20	73.96	78.29	57.09	59.96	63.74
LINE	75.05	85.73	94.75	60.54	65.05	68.87	53.12	52.97	53.22
node2vec	88.06	94.57	97.11	71.15	76.33	80.16	56.84	61.22	65.93
PMNE	90.12	94.47	95.24	77.61	79.85	81.35	61.12	70.72	75.91
MELL	93.51	96.30	98.84	76.18	79.92	81.21	70.64	75.89	79.84
CrossMNA	**96.46**	**97.53**	**99.19**	76.88	**81.12**	**82.59**	**75.85**	**80.48**	**85.29**

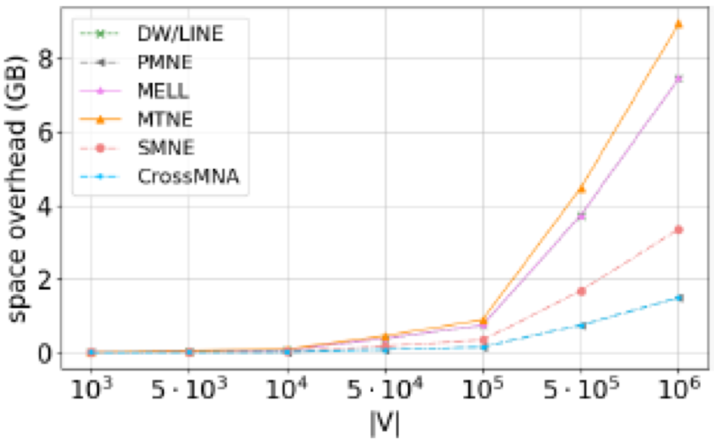
Observation: CrossMNA performs better due to transmitting complementary information across networks.

Chu, Xiaokai, et al. "Cross-network embedding for multi-network alignment." The World Wide Web Conference. 2019.

CrossMNA – Experimental Results

- Scalability: memory usage





Observation: CrossMNA has less memory usage than other baseline methods.

Chu, Xiaokai, et al. "Cross-network embedding for multi-network alignment." The World Wide Web Conference. 2019.

Overview of Part I

Part I: 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**

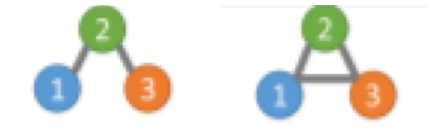
Related Tasks
- Entity alignment
  - Non-GNN based
  - GNN-based
- Cross-layer inference
- Cross-network transformation

Higher-order Network Alignment

- Higher-order network mining:
  - Involves higher-order structures, instead of edges
- Motivations:
  - Traditional approaches (e.g., NetAlign) aim to maximize # of conserved edges (overlaps/squares).
  - Leverage higher-order structures exist in networks (e.g., motifs, clusters, etc.).
- Single-level: use higher-order structures to align nodes
- Multilevel: to align both nodes and clusters at multi-level

Triangular Alignment (TAME)

- Network motifs: connected subgraphs that occurs with significantly higher frequency
  - 3rd-order: 3-node line, triangle



  - k^{th}-order: k-node star, etc.

- Objective: to maximize # of aligned substructures

Mohammadi, Shahin, et al. "Triangular alignment (TAME): A tensor-based approach for higher-order network alignment." IEEE/ACM transactions on computational biology and bioinformatics 14.6 (2016): 1446-1458.

TAME – Formulation #1

- Binary quadratic program in NetAlign

$$\begin{array}{ll} \underset{\boldsymbol{x}}{\text{maximize}} & (1-\alpha)\boldsymbol{w}^T\boldsymbol{x} + \frac{\alpha}{2}\boxed{\boldsymbol{x}^T\mathbf{S}\boldsymbol{x}} \\ \text{subject to} & \mathbf{C}\boldsymbol{x} \le \mathbf{1}_{|V_G|+|V_H|} \\ & \boldsymbol{x}(ii') \in \{0,1\}. \end{array}$$

To maximize # of conserved edges

- Higher-order extension

$$\boldsymbol{x}^T(\boldsymbol{\mathcal{T}}_H \otimes \boldsymbol{\mathcal{T}}_G)\boldsymbol{x}^{m-1} = (\boldsymbol{\mathcal{T}}_H \otimes \boldsymbol{\mathcal{T}}_G)\boldsymbol{x}^m$$

$$\begin{array}{ll} \underset{\boldsymbol{x}}{\text{maximize}} & (1-\alpha)\boldsymbol{w}^T\boldsymbol{x} + \frac{\alpha}{m!}\boxed{(\mathcal{T}_H \otimes \mathcal{T}_G)\boldsymbol{x}^m} \\ \text{subject to} & \mathbf{C}\boldsymbol{x} \le \mathbf{1}_{|V_G|+|V_H|} \\ & \boldsymbol{x}(ii') \in \{0,1\}. \end{array}$$

$$\begin{array}{ll} \underset{\boldsymbol{x}}{\text{maximize}} & (1-\alpha)\boldsymbol{w}^T\boldsymbol{x} + \frac{\alpha}{6}\boxed{(\triangle_{H\times G})\boldsymbol{x}^3} \\ \text{subject to} & \mathbf{C}\boldsymbol{x} \le \mathbf{1}_{|V_G|+|V_H|} \\ & \boldsymbol{x}(ii') \in \{0,1\}. \end{array}$$

$\mathcal{T}_H$ and $\mathcal{T}_G$: the motif-tensors associated with a m-node motif in both graphs G and H

- $\Delta_{H\times G} = \Delta_H \otimes \Delta_G$: Kronecker product of triangle tensors
- Counts # of conserved triangles

TAME – Formulation #2

- Relaxed formulation
  - Remove one-to-one constraint and relax $\boldsymbol{x}$ to be any reals
  - Add a 2-norm constraint on $\boldsymbol{x}$ to make it bounded

$$\begin{array}{ll} \underset{\boldsymbol{x}}{\text{maximize}} & (\triangle_{H\times G})\boldsymbol{x}^3 \\ \text{subject to} & \|\boldsymbol{x}\| = 1. \end{array} \longrightarrow \text{Tensor eigenvector problem}$$

  - The classic SS-HOPM is costly to solve it.
- Implicit kernel for computing tensor-vector products

$$\begin{aligned}
&(\triangle_{H\times G}\boldsymbol{x}^2)_{ii'} \\
&\quad = \sum_{jj',kk'} \triangle_{H\times G}(ii',jj',kk')\boldsymbol{x}(jj')\boldsymbol{x}(kk') \\
&\quad = \sum_{j,j',k,k'} \triangle_G(i,j,k)\triangle_H(i',j',k')\mathbf{X}(j,j')\mathbf{X}(k,k') \\
&\quad = \sum_{j,k} \triangle_G(i,j,k)\sum_{j'}\mathbf{X}(j,j')\sum_{k'}\triangle_H(i',j',k')\mathbf{X}(k,k')
\end{aligned}$$

$$\Longrightarrow$$

$$\begin{aligned}
&\triangle_{H\times G}\boldsymbol{x}^2(ii') = \\
&\quad 2\sum_{(j,k)\in\mathcal{N}_{\triangle_G}(i)}\sum\nolimits_{(j',k')\in\mathcal{N}_{\triangle_H}(i')} \mathbf{X}(j,j')\mathbf{X}(k,k') + \mathbf{X}(j,k')\mathbf{X}(k,j').
\end{aligned}$$

Mohammadi, Shahin, et al. "Triangular alignment (TAME): A tensor-based approach for higher-order network alignment." IEEE/ACM transactions on computational biology and bioinformatics 14.6 (2016): 1446-1458.

TAME – Algorithm

- Key ideas:
  - To use implicit tensor-kernel product $\widetilde{\boldsymbol{x}} = \Delta_{H\times G}\boldsymbol{x}^2$ for $\Delta_{H\times G}\boldsymbol{x}^3 = \boldsymbol{x}^T\widetilde{\boldsymbol{x}}$
  - SS-HOPM main loop computes topological similarity matrices
  - A score function to solve a bipartite max-weight matching

⇓

To encode integer constraint of X and one-to-one mapping constraint

Algorithm 2 The Triangular AlignMEnt (TAME) algorithm

Input: Triangle tensors $\triangle_G, \triangle_H$; Sequence similarities $\boldsymbol{w}$; Shift parameter β

Output: The best topological scores $\mathbf{X}$ from any iteration

$k = 0$ {Iteration number}

$\boldsymbol{w} \leftarrow \boldsymbol{w}/\|\boldsymbol{w}\|$

$\boldsymbol{x}_0 = \boldsymbol{w}$

$t_0 = 0$

repeat

$\quad \boxed{\tilde{\boldsymbol{x}}_{k+1} = \mathbf{impTTV}(\triangle_G, \triangle_H, \boldsymbol{x}_k)}$

$\quad \lambda_{k+1} = \boldsymbol{x}_k^T \tilde{\boldsymbol{x}}_{k+1}$

$\quad \hat{\boldsymbol{x}}_{k+1} = \tilde{\boldsymbol{x}}_{k+1} + \beta \boldsymbol{x}_k$

$\quad \boldsymbol{x}_{k+1} = \frac{\hat{\boldsymbol{x}}_{k+1}}{\|\hat{\boldsymbol{x}}_{k+1}\|}$

$\quad \mathbf{X}_{k+1} = \mathbf{unvec}(\boldsymbol{x}_{\mathbf{k+1}})$

$\quad \boxed{t_{k+1} = \mathbf{score}(\mathbf{X}_{\mathbf{k+1}})}$

$\quad$ Update $(\mathbf{X}, t)_{\text{best}}$ to $(\mathbf{X}, t)_{k+1}$ if $t_{k+1} > t_{\text{best}}$

$\quad k = k + 1$

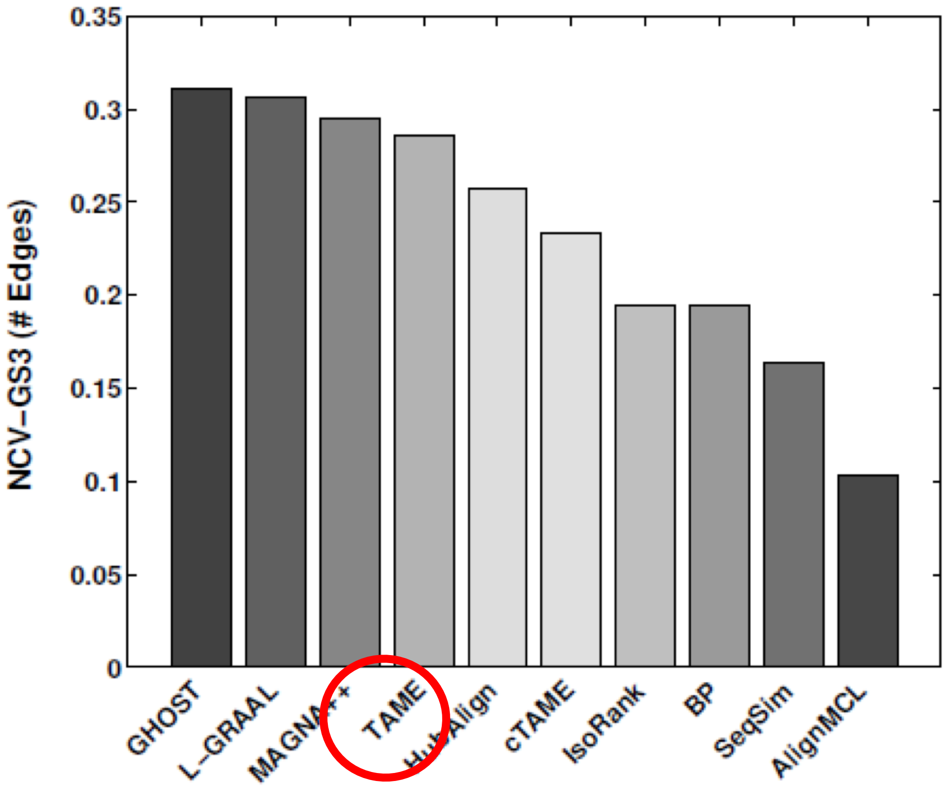
until $\lambda_k - \lambda_{k-1}$ is small or the max iteration is hit

return $\mathbf{X}_{\text{best}}$

Mohammadi, Shahin, et al. "Triangular alignment (TAME): A tensor-based approach for higher-order network alignment." IEEE/ACM transactions on computational biology and bioinformatics 14.6 (2016): 1446-1458.

TAME – Experimental Results

- Alignment quality on yeast vs. human dataset

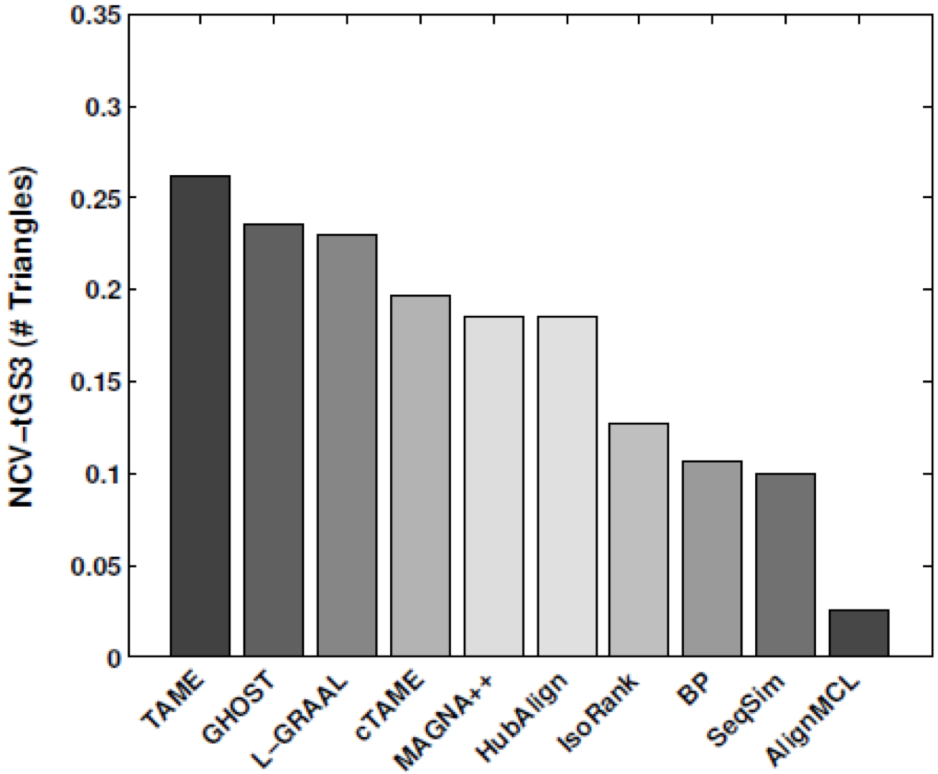


Observation: TAME performs closely to the best method in preserving the # of conserved edges

Mohammadi, Shahin, et al. "Triangular alignment (TAME): A tensor-based approach for higher-order network alignment." IEEE/ACM transactions on computational biology and bioinformatics 14.6 (2016): 1446-1458.

TAME – Experimental Results

- Metric: # of conserved triangles

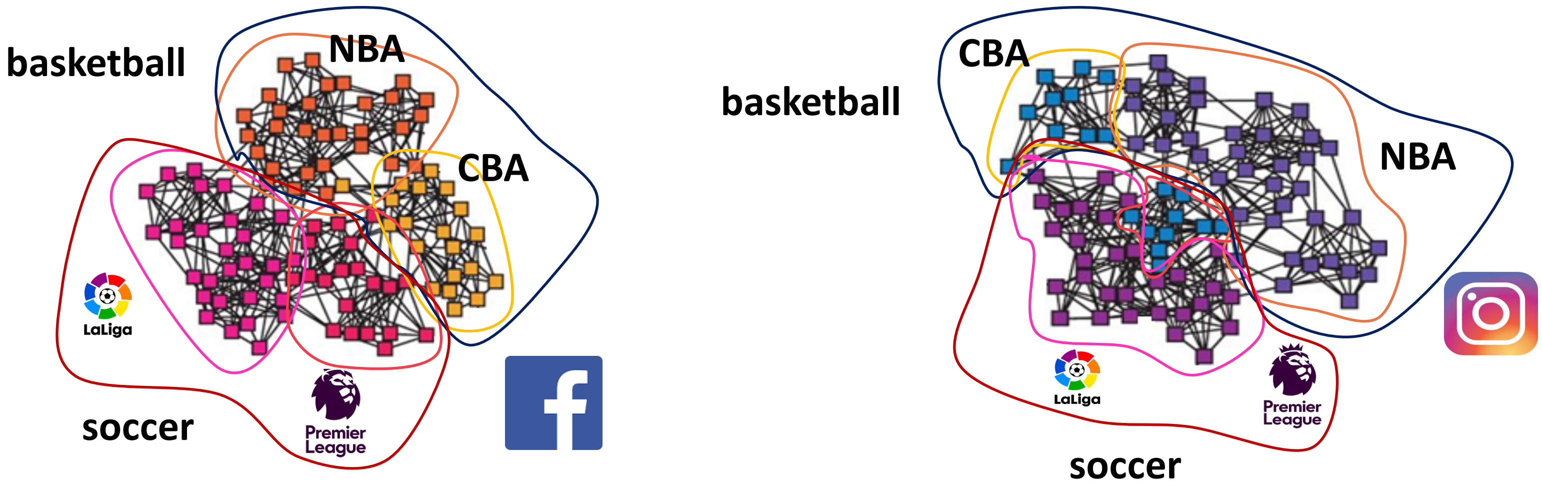


Observation: TAME ranks the highest in terms of the number of conserved triangles

Mohammadi, Shahin, et al. "Triangular alignment (TAME): A tensor-based approach for higher-order network alignment." IEEE/ACM transactions on computational biology and bioinformatics 14.6 (2016): 1446-1458.

Multilevel Network Alignment

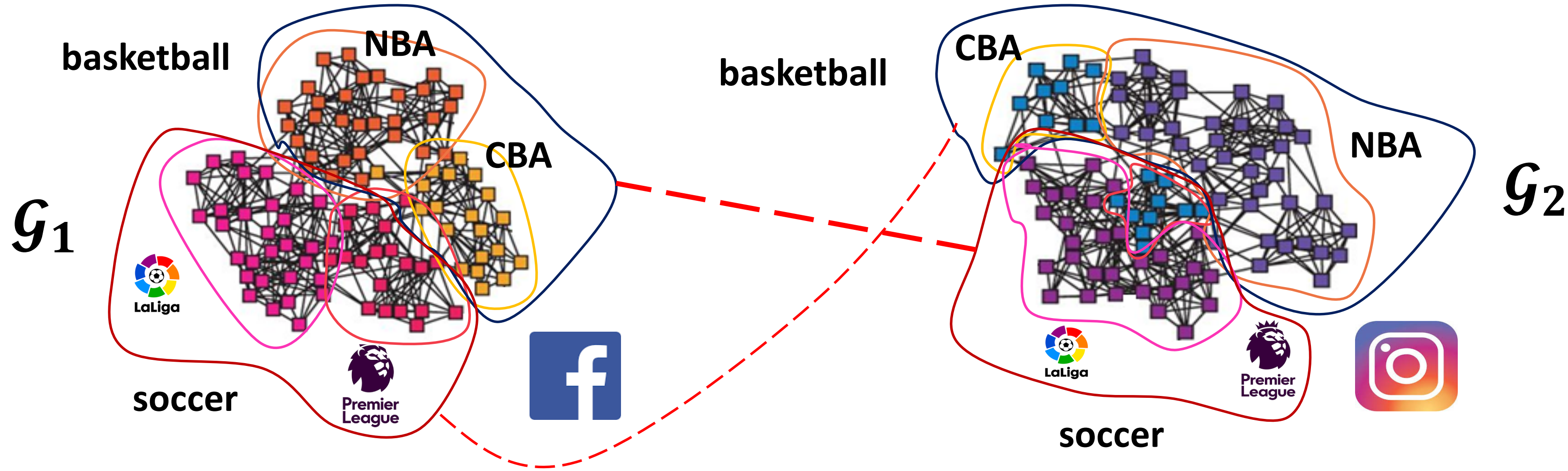
- **Goals:** to find node correspondence as well as the correspondence among clusters at different levels
- Motivation:
  - Networks exhibit hierarchical cluster-within-clusters structure



Zhang, Si, et al. "Multilevel network alignment." The World Wide Web Conference. 2019.

Moana – Challenges

- C1: Alignment accuracy

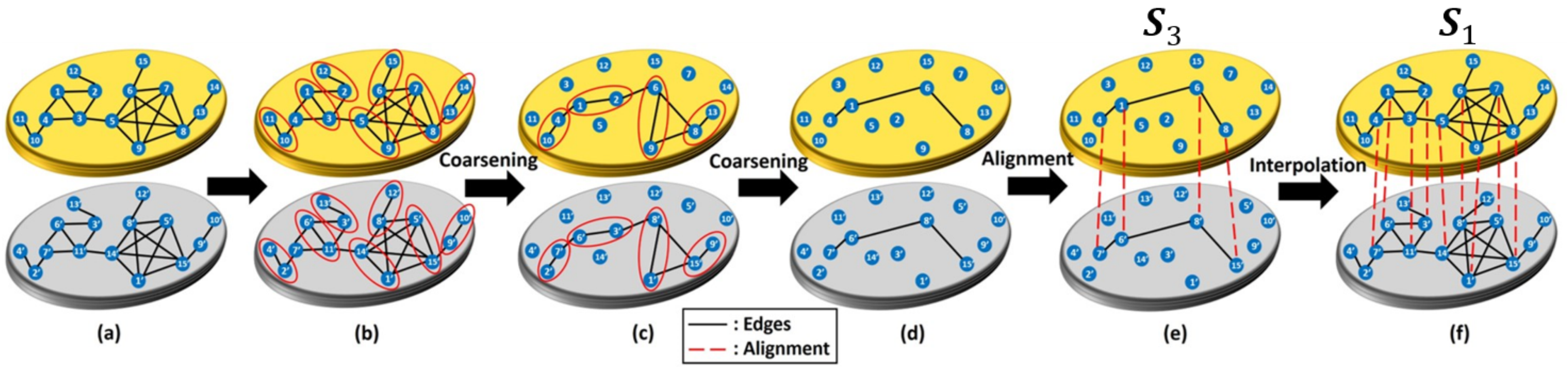


  - **Errors propagate through levels**
- C2: Scalability   Better than quadratic?

Zhang, Si, et al. "Multilevel network alignment." The World Wide Web Conference. 2019.

Moana – Problem Definition

- Given:
  - (1) adjacency matrices $\bar{\boldsymbol{A}}_1, \bar{\boldsymbol{B}}_1$ of two undirected networks;
  - (2) a sparse prior alignment preference $\boldsymbol{H}_1$;
  - (3) the number of levels $L \geq 2$ of interests.
- Find: a set of alignment matrices $\boldsymbol{S}_l$ at level-l, $l = 1, \cdots, L$
  - where $\boldsymbol{S}_1$ indicates the alignment at the node level



Zhang, Si, et al. "Multilevel network alignment." The World Wide Web Conference. 2019.

Moana Formulation: Multilevel Optimization

- Generic strategy
  - coarsening → alignment → interpolation
- Alignment interpolations
  - Bilinear interpolations by $\boldsymbol{P}_l \in R^{p_l \times n_1}, \boldsymbol{Q}_l \in R^{q_l \times n_2}$ $(p_l \leq n_1, q_l \leq n_2)$
  - w.l.o.g., $\boldsymbol{S}_1 = \boldsymbol{Q}_1^T \boldsymbol{S}_2 \boldsymbol{P}_1$ between level-1 & level-2

Moana Formulation: Multilevel Optimization

- Multilevel alignment formulation

FINAL-P at node level

Level-1: $$\min_{\boldsymbol{s}_1} \alpha \boldsymbol{s}_1^T (\boldsymbol{I} - \boldsymbol{A}_1 \otimes \boldsymbol{B}_1) \boldsymbol{s}_1 + (1-\alpha) \|\boldsymbol{s}_1 - \boldsymbol{h}_1\|_2^2$$

If $\boldsymbol{P}_1 \boldsymbol{P}_1^T = \boldsymbol{I}$ and $\boldsymbol{Q}_1 \boldsymbol{Q}_1^T = \boldsymbol{I}$

Level-2: $$\min_{\boldsymbol{s}_2} \alpha \boldsymbol{s}_2^T (\boldsymbol{I} - \boldsymbol{A}_2 \otimes \boldsymbol{B}_2) \boldsymbol{s}_2 + (1-\alpha) \|\boldsymbol{s}_2 - \boldsymbol{h}_2\|_2^2$$

- $\boldsymbol{A}_2 = \boldsymbol{P}_1 \boldsymbol{A}_1 \boldsymbol{P}_1^T, \boldsymbol{B}_2 = \boldsymbol{Q}_1 \boldsymbol{B}_1 \boldsymbol{Q}_1^T$ and $\boldsymbol{H}_2 = \boldsymbol{Q}_1 \boldsymbol{H}_1 \boldsymbol{P}_1^T$
- same properties (e.g., convexity) and algorithm as FINAL-P
- 'good' (semi-) orthogonal $\boldsymbol{P}_1, \boldsymbol{Q}_1$ make $\boldsymbol{A}_2, \boldsymbol{B}_2$ well-represented

Zhang, Si, et al. "Multilevel network alignment." The World Wide Web Conference. 2019.

Moana Formulation: Perfect Interpolation

- Denote $\boldsymbol{S}_l^*, \boldsymbol{S}_{l+1}^*$ are optimal solutions at level-l and level-$(l+1)$
- Perfect interpolation (to address error propagation):

  Interpolation from the optimal alignment matrix at level-$(l+1)$ is equal to that at level-l

  - If $\boldsymbol{P}_l, \boldsymbol{Q}_l$ $(l = 1, \cdots, L-1)$ are orthogonal
  - Then $\boldsymbol{S}_l^* = \boldsymbol{Q}_l^T \boldsymbol{S}_{l+1}^* \boldsymbol{P}_l$

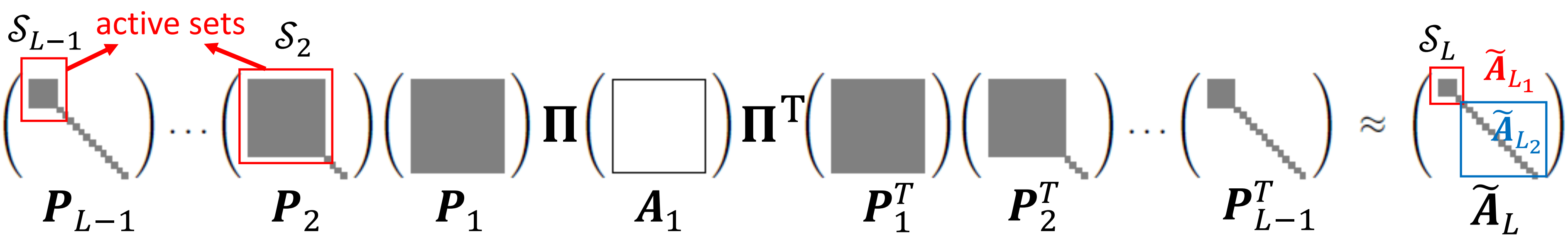
Moana – Coarsening Algorithm

- Generic strategy
  - Coarsening → alignment → interpolation
- Network coarsening by $\boldsymbol{P}_l, \boldsymbol{Q}_l$
  - $\boldsymbol{A}_{l+1} = \boldsymbol{P}_l \boldsymbol{A}_l \boldsymbol{P}_l^T, \boldsymbol{B}_{l+1} = \boldsymbol{Q}_l \boldsymbol{B}_l \boldsymbol{Q}_l^T$
- Requirements on $\boldsymbol{P}_l, \boldsymbol{Q}_l$
  - **Perfect interpolation:** they are orthogonal matrix
  - **Efficient computation:** they are sparse matrix
  - **Informative coarsening:** they can uncover hierarchical cluster-within-clusters structures

Zhang, Si, et al. "Multilevel network alignment." The World Wide Web Conference. 2019.

Moana – Coarsening Algorithm

- Multiresolution matrix factorization



- Coarsening procedure
  - $\boldsymbol{P}_{L-1} \cdots \boldsymbol{P}_2 \boldsymbol{P}_1 \boldsymbol{A}_1 \boldsymbol{P}_1^T \boldsymbol{P}_2^T \cdots \boldsymbol{P}_{L-1}^T = \boldsymbol{A}_L \rightarrow \widetilde{\boldsymbol{A}}_L$
  - $\boldsymbol{Q}_{L-1} \cdots \boldsymbol{Q}_2 \boldsymbol{Q}_1 \boldsymbol{B}_1 \boldsymbol{Q}_1^T \boldsymbol{Q}_2^T \cdots \boldsymbol{Q}_{L-1}^T = \boldsymbol{B}_L \rightarrow \widetilde{\boldsymbol{B}}_L$
- $\boldsymbol{S}(\mathcal{S}_{B_l}, \mathcal{S}_{A_l})$ indicates the alignment among clusters at the l-th granularity

Kondor, Risi, Nedelina Teneva, and Vikas Garg. "Multiresolution matrix factorization." *International Conference on Machine Learning*. 2014.

Moana – Alignment Algorithm

- Generic strategy
  - coarsening → alignment → interpolation
- Alignment across the coarsest networks

$$\tilde{\mathbf{A}}_L = \mathbf{\Pi}_A \begin{bmatrix} \tilde{\mathbf{A}}_{L_1} & \mathbf{0} \\ \mathbf{0} & \tilde{\mathbf{A}}_{L_2} \end{bmatrix} \mathbf{\Pi}_A^T$$
$$\tilde{\mathbf{B}}_L = \mathbf{\Pi}_B \begin{bmatrix} \tilde{\mathbf{B}}_{L_1} & \mathbf{0} \\ \mathbf{0} & \tilde{\mathbf{B}}_{L_2} \end{bmatrix} \mathbf{\Pi}_B^T$$

$$\tilde{\boldsymbol{S}}_L = \alpha \begin{bmatrix} \tilde{\boldsymbol{B}}_{L_1} & \mathbf{0} \\ \mathbf{0} & \tilde{\boldsymbol{B}}_{L_2} \end{bmatrix} \begin{bmatrix} \tilde{\boldsymbol{S}}_{L_1} & \tilde{\boldsymbol{S}}_{L_2} \\ \tilde{\boldsymbol{S}}_{L_3} & \tilde{\boldsymbol{S}}_{L_4} \end{bmatrix} \begin{bmatrix} \tilde{\boldsymbol{A}}_{L_1} & \mathbf{0} \\ \mathbf{0} & \tilde{\boldsymbol{A}}_{L_2} \end{bmatrix} + (1-\alpha) \begin{bmatrix} \tilde{\boldsymbol{H}}_{L_1} & \tilde{\boldsymbol{H}}_{L_2} \\ \tilde{\boldsymbol{H}}_{L_3} & \tilde{\boldsymbol{H}}_{L_4} \end{bmatrix}$$

block-wise computation

$$\tilde{\boldsymbol{S}}_{L_1} = \alpha \tilde{\boldsymbol{B}}_{L_1} \tilde{\boldsymbol{S}}_{L_1} \tilde{\boldsymbol{A}}_{L_1} + (1-\alpha) \tilde{\boldsymbol{H}}_{L_1}$$
$$\tilde{\boldsymbol{S}}_{L_2} = \alpha \tilde{\boldsymbol{B}}_{L_1} \tilde{\boldsymbol{S}}_{L_2} \tilde{\boldsymbol{A}}_{L_2} + (1-\alpha) \tilde{\boldsymbol{H}}_{L_2}$$
$$\tilde{\boldsymbol{S}}_{L_3} = \alpha \tilde{\boldsymbol{B}}_{L_2} \tilde{\boldsymbol{S}}_{L_3} \tilde{\boldsymbol{A}}_{L_1} + (1-\alpha) \tilde{\boldsymbol{H}}_{L_3}$$
$$\tilde{\boldsymbol{s}}_{L_4} = (1-\alpha)\left(\boldsymbol{I} - \alpha \tilde{\boldsymbol{A}}_{L_2} \otimes \tilde{\boldsymbol{B}}_{L_3}\right)^{-1} \tilde{\boldsymbol{h}}_{L_4}$$

- Alignment at finer levels
  - perfect interpolations: $\boldsymbol{S}_l = \boldsymbol{Q}_l^T \boldsymbol{S}_{l+1} \boldsymbol{P}_l$

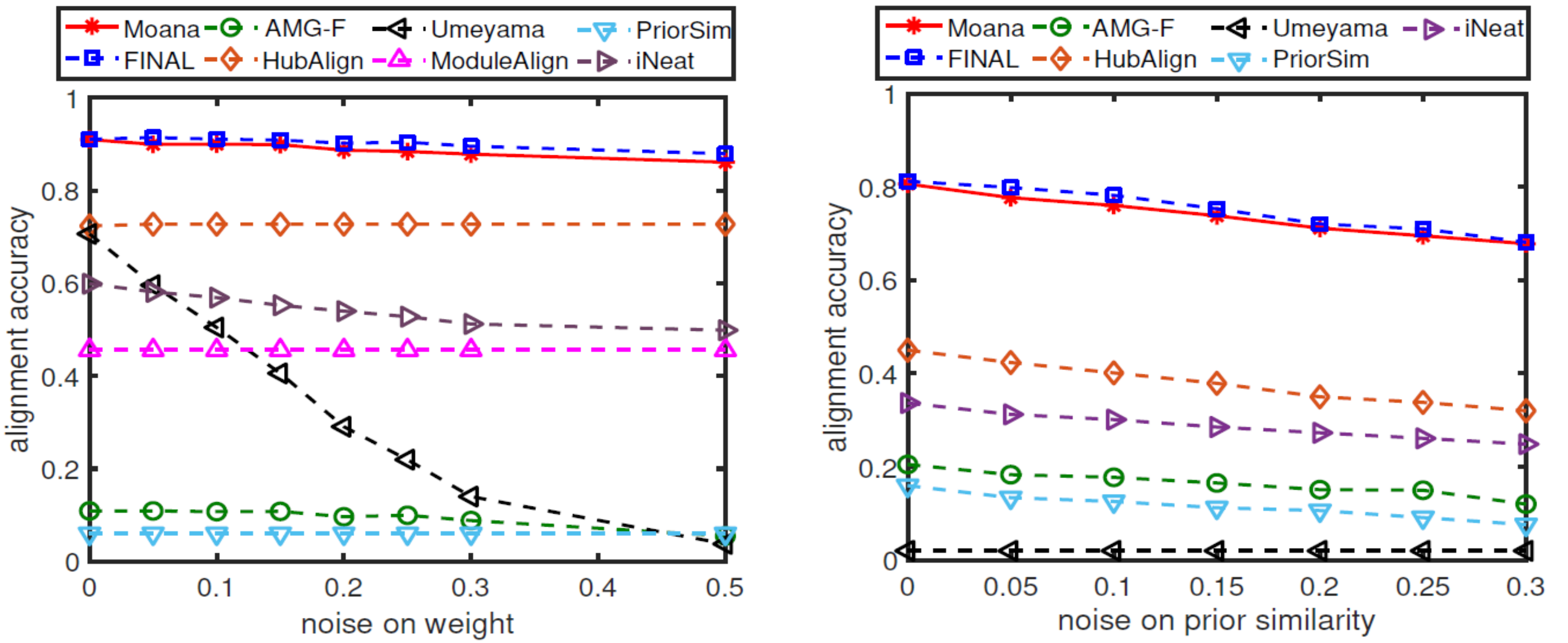
Zhang, Si, et al. "Multilevel network alignment." The World Wide Web Conference. 2019.

Moana – Experimental Setups

- Datasets
  - Gr-Qc network vs. its permutation (nodes: 5,241 vs. 5,241)
  - Google+ vs. its permutation (nodes: 23,628 vs. 23,628)
  - Amazon co-purchasing networks (nodes: 74,596 vs. 66,951)
  - ACM vs DBLP coauthor networks (nodes: 9,872 vs. 9,916)

Zhang, Si, et al. "Multilevel network alignment." The World Wide Web Conference. 2019.

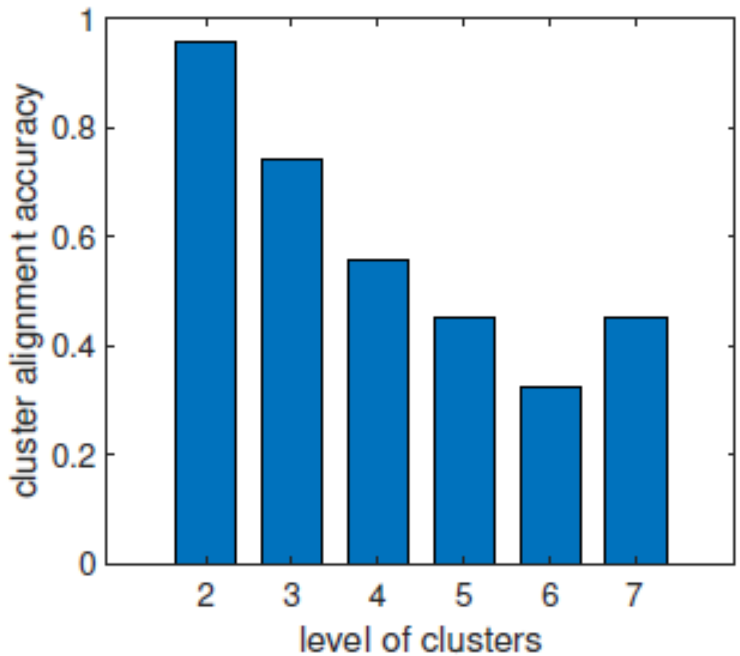
Moana – Experimental Results



Observations: (1) the performance of Moana is close to FINAL-P;
(2) Moana outperforms all other methods.

Zhang, Si, et al. "Multilevel network alignment." The World Wide Web Conference. 2019.

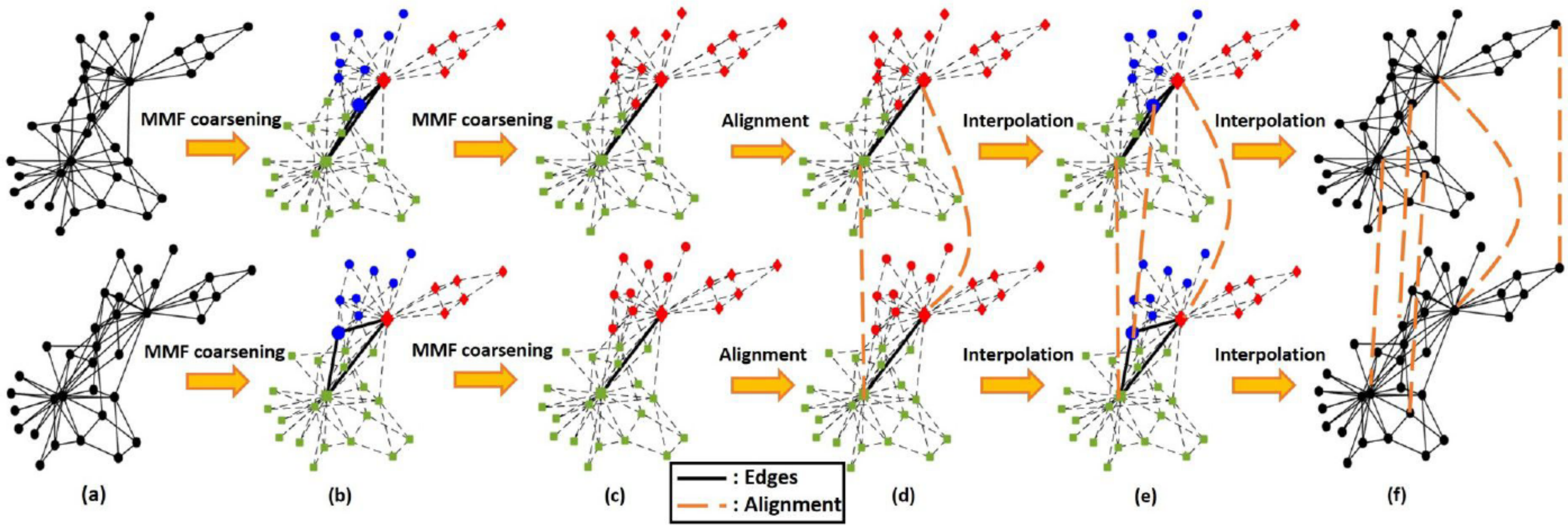
Moana – Experimental Results



Observation: Moana achieves a good performance in cluster alignment at different levels.

Zhang, Si, et al. "Multilevel network alignment." The World Wide Web Conference. 2019.

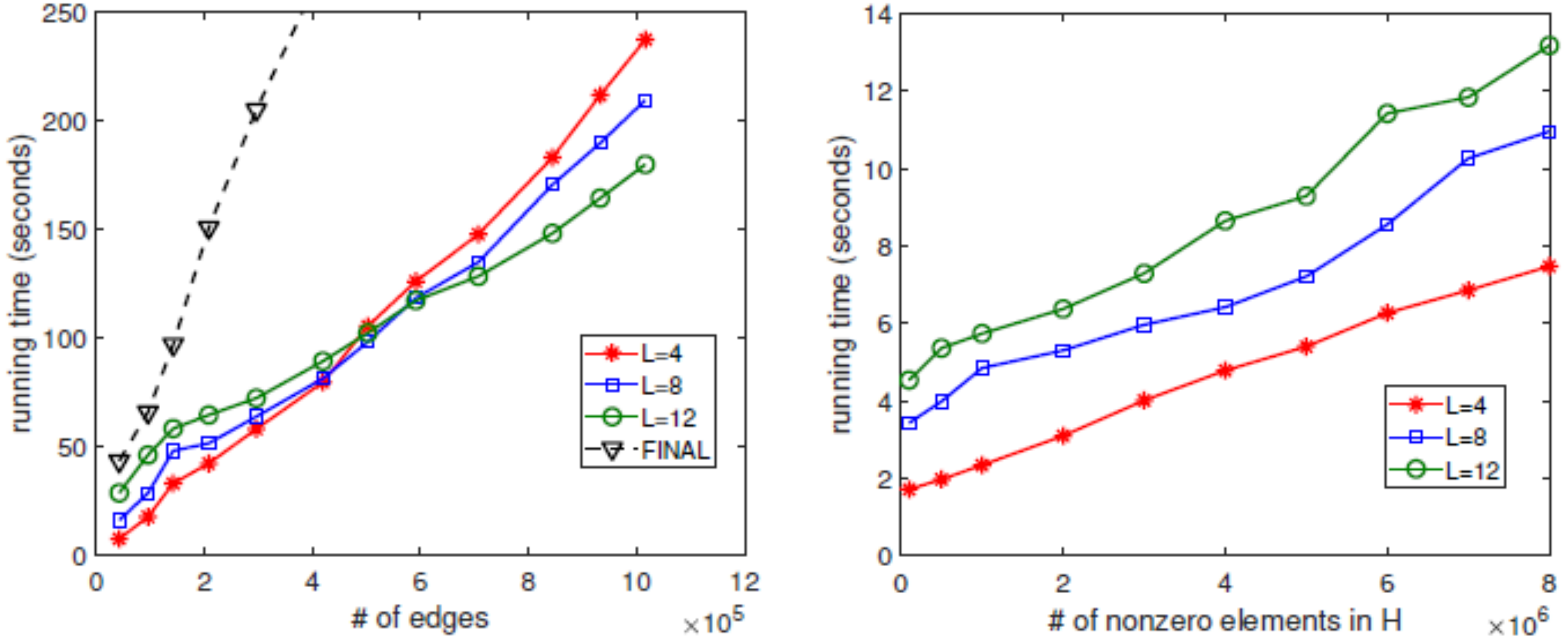
Moana – Experimental Results



Observation: Moana can unveil meaningful alignment of clusters at different granularities.

Zhang, Si, et al. "Multilevel network alignment." The World Wide Web Conference. 2019.

Moana – Experimental Results



Observation: (1)Moana scales linearly w.r.t. the number of edges;
(2)Moana scales linearly w.r.t. the number of nonzero elements in $\boldsymbol{H}_1$.

Overview of Part I

Part I: 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

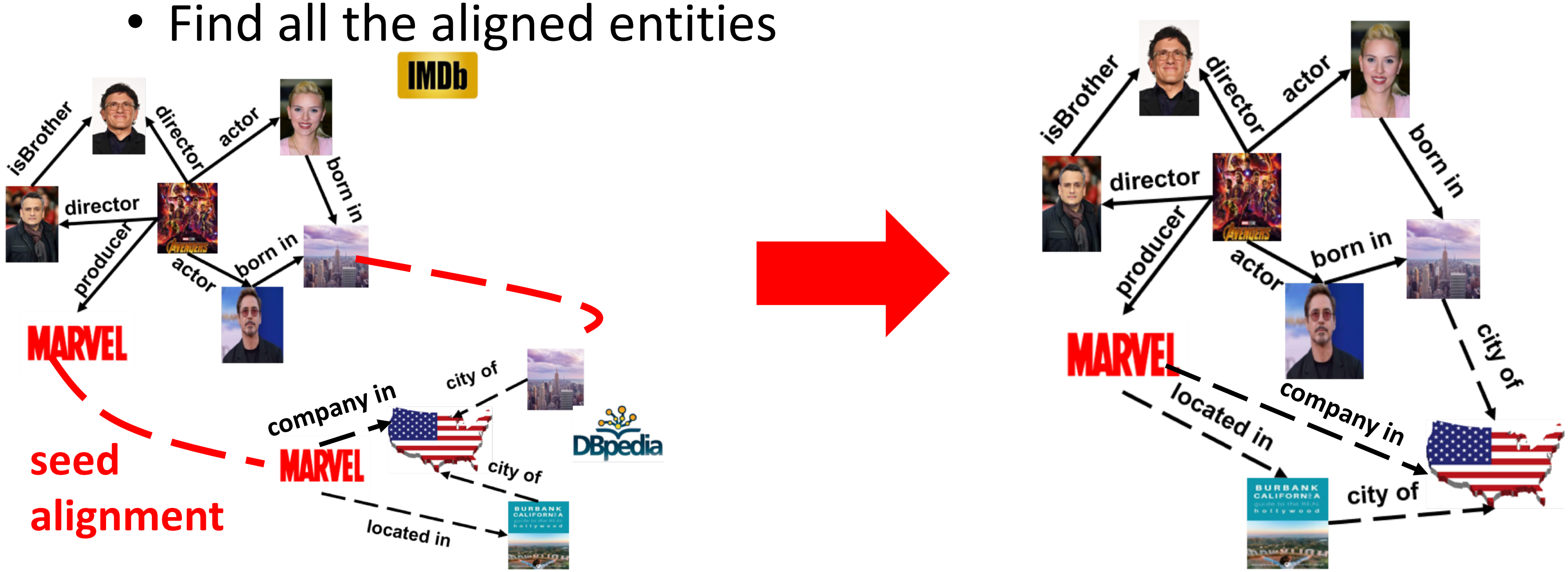
Collective NA
- Consistency-based
  - w/o attributes
  - w/ attributes
- Embedding-based
  - w/o attributes

Higher-Order NA
- Consistency-based
  - Single-level
  - Multilevel

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

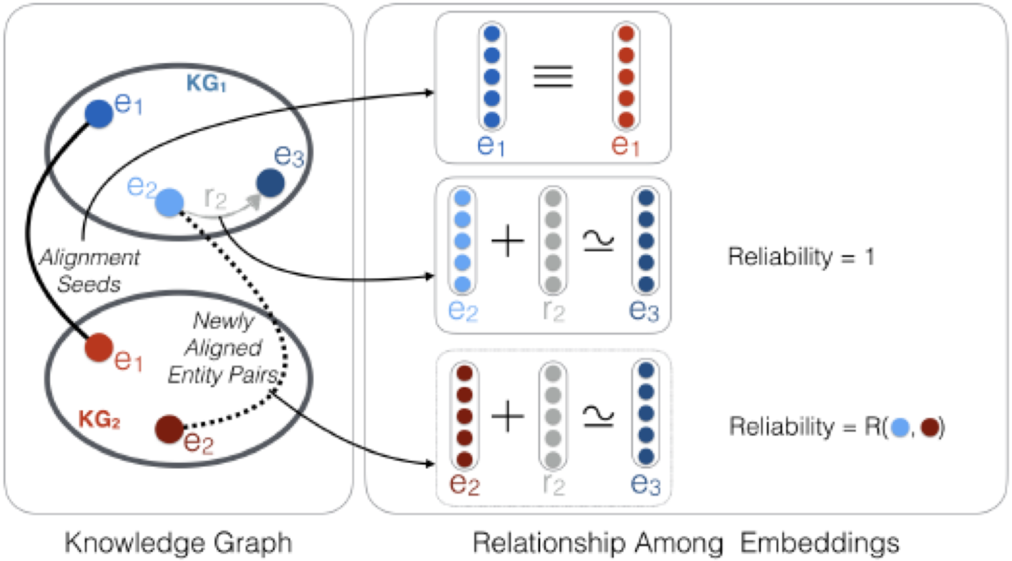
Entity Alignment

- Goal: to link entities among multiple knowledge graphs
- Problem Definition:
  - Given KGs $\{KG_i | KG_i = (E_i, R_i, T_i)\}$ and seed alignment $\mathcal{L}$;
  - Find all the aligned entities



Iterative Entity Alignment via Joint Knowledge Embeddings

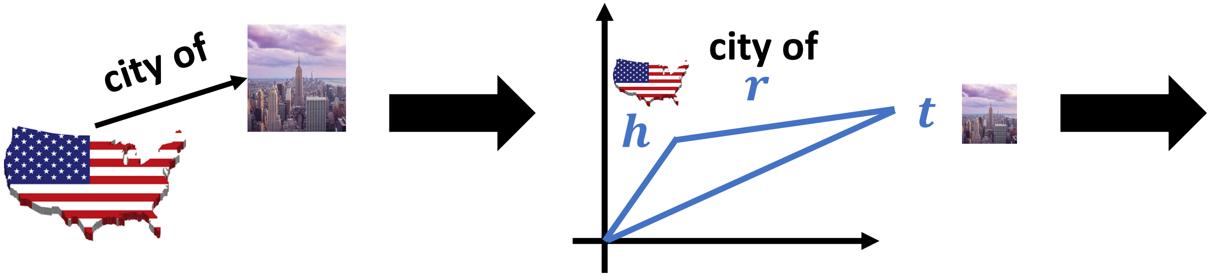
- Key components:
  - Knowledge embedding: TransE, PTransE
  - Joint embedding: translation-based, linear transformation
  - Iterative alignment: adding newly aligned entities



Zhu, Hao, et al. "Iterative Entity Alignment via Joint Knowledge Embeddings." IJCAI. Vol. 17. 2017.

ITransE – Knowledge Embeddings

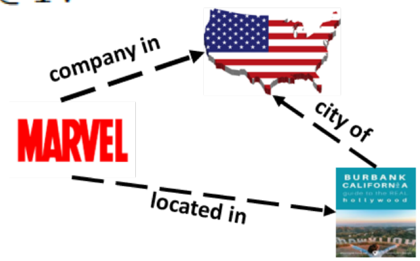
- TransE: relations as translating vectors



$$E(h,r,t) = \|\boldsymbol{h}+\boldsymbol{r}-\boldsymbol{t}\|$$

  - Loss function: $L(h,r,t) = \sum_{(h',r',t') \in T^-} [\gamma + E(h,r,t) - E(h',r',t')]_+$
  - Negative samples: $T^- = \{(h',r,t)|h' \in E\} \cup \{(h,r,t')|t' \in E\} \cup \{(h,r',t)|r' \in R\}, \quad (h,r,t) \in T.$

- PTransE: to encode multi-step relation path



$$E(p,r) = ||\mathbf{p}-\mathbf{r}|| = ||\mathbf{p}-(\mathbf{t}-\mathbf{h})|| = E(h,p,t)$$

Zhu, Hao, et al. "Iterative Entity Alignment via Joint Knowledge Embeddings." IJCAI. Vol. 17. 2017.

ITransE – Joint Embeddings

- Key idea: to join embeddings in a unified space
- Translation-based model:
  - Key idea: view alignment as a special relation
  - Formulation: given $e_1 \in E_1, e_2 \in E_2 \rightarrow \boldsymbol{e}_1 + \boldsymbol{r}^{(E_1 \rightarrow E_2)} \approx \boldsymbol{e}_2$

$$E(e_1, e_2) = ||\mathbf{e}_1 + \mathbf{r}^{(E_1 \rightarrow E_2)} - \mathbf{e}_2||$$

- Linear transformation model:
  - Key idea: embedding space can be transformed linearly
  - Formulation: transformation matrix $\boldsymbol{M}^{(E_1 \rightarrow E_2)}$

$$E(e_1, e_2) = ||\mathrm{M}^{(E_1 \rightarrow E_2)}\mathbf{e}_1 - \mathbf{e}_2||$$

Zhu, Hao, et al. "Iterative Entity Alignment via Joint Knowledge Embeddings." IJCAI. Vol. 17. 2017.

ITransE – Iterative Alignment

- Key idea: iteratively adding newly aligned entities
- Soft alignment:
  - Reliability scores of newly aligned entities

$$R(e_1, e_2) = \sigma\Big(k(\theta - E(e_1, e_2))\Big)$$

  - Score function for soft alignment

$$\mathcal{I}_S = \sum_{(e_1,e_2)\in\mathbb{M}} R(e_1, e_2)(\mathcal{H}_{(e_1,e_2)} + \mathcal{H}_{(e_2,e_1)}),$$

$$\mathcal{H}_{(e_1,e_2)} = \sum_{(e_1,r,t)} U(e_2, r, t) + \sum_{(h,r,e_1)} U(h, r, e_2),$$

  - Limit # of newly aligned entities to a threshold in each alignment procedure

ITransE – Experimental Results

- Dataset: DFB1, DFB2, DFB3 from FB15K
- Entity alignment performance
  - ITransE: iterative alignment w/ TransE
  - IPTransE: iterative alignment w/ PTransE

Metric	DFB-1			DFB-2			DFB-3		
	Hits@1	Hits@10	Mean Rank	Hits@1	Hits@10	Mean Rank	Hits@1	Hits@10	Mean Rank
MTransE (LT)	38.9	61.0	237.7	12.3	33.8	419.2	6.5	22.0	699.8
MTransE (TB)	13.6	35.1	547.7	13.9	35.4	675.7	4.5	16.1	1255.5
TransE + PS	61.9	79.2	105.2	41.1	67.0	154.9	12.2	34.6	431.9
ITransE (HA)	62.6	78.9	100.0	41.2	66.9	151.9	12.3	33.7	432.3
ITransE (SA)	**67.1**	**83.1**	**80.1**	**57.7**	**77.7**	**109.3**	**16.2**	**40.9**	**367.2**
PTransE + PS	65.8	83.4	62.9	46.3	72.1	96.8	15.8	40.2	346.9
IPTransE (HA)	66.1	83.3	59.1	46.2	72.6	94.2	15.1	39.7	337.6
IPTransE (SA)	**71.7**	**86.5**	**49.0**	**63.5**	**82.2**	**67.5**	**20.4**	**47.4**	**281.0**

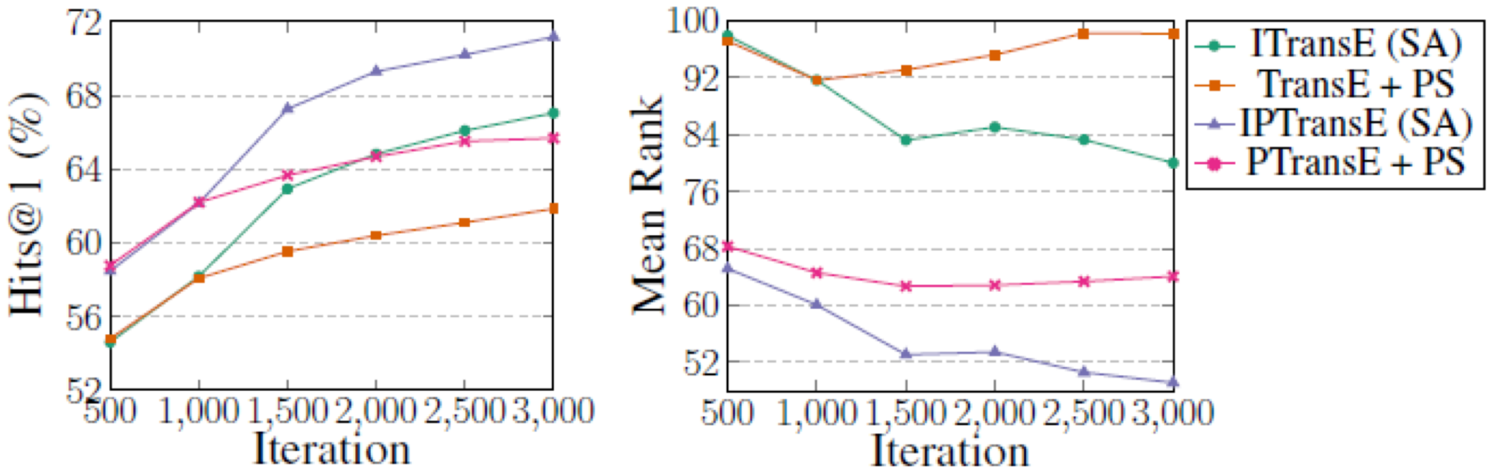
Observations:

- IPTransE performs better than ITransE
- Soft alignment performs better than hard alignment

Zhu, Hao, et al. "Iterative Entity Alignment via Joint Knowledge Embeddings." IJCAI. Vol. 17. 2017.

ITransE – Experimental Results

- Effectiveness of soft alignment strategy



Observation: the performance of all methods increase with iterations.

Zhu, Hao, et al. "Iterative Entity Alignment via Joint Knowledge Embeddings." IJCAI. Vol. 17. 2017.

Knowledge Graph Alignment via Graph Convolutional Networks

- Key idea: use GCNs to embed entities where aligned entities are expected to be as close as possible.
- Assumptions:
  - Equivalent entities tend to have similar attributes
  - Equivalent entities are neighbored by other equivalent entities
- Embedding framework:

$$[H_s^{(l+1)}; H_a^{(l+1)}] = \sigma\left(\hat{D}^{-\frac{1}{2}}\hat{A}\hat{D}^{-\frac{1}{2}}[H_s^{(l)}W_s^{(l)}; H_a^{(l)}W_a^{(l)}]\right)$$

Wang, Zhichun, et al. "Cross-lingual knowledge graph alignment via graph convolutional networks." Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing. 2018.

GCN-Align – Construct Adjacency Matrix

- KGs are relational multi-graphs (i.e., typed relations)
- Key idea: two measures on relations

Relation functionality: $fun(r) = \frac{\#Head\_Entities\_of\_r}{\#Triples\_of\_r}$

Inverse functionality: $ifun(r) = \frac{\#Tail\_Entities\_of\_r}{\#Triples\_of\_r}$

- Edge weight: influence of i-th entity over j-th entity

$$a_{ij} = \sum_{\langle e_i, r, e_j \rangle \in G} ifun(r) + \sum_{\langle e_j, r, e_i \rangle \in G} fun(r)$$

Wang, Zhichun, et al. "Cross-lingual knowledge graph alignment via graph convolutional networks." Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing. 2018.

GCN-Align – Alignment Prediction

- Model training:
  - Margin-based rank loss for both $\boldsymbol{h}_s$ and $\boldsymbol{h}_a$
  - $\boldsymbol{h}_s$: structure embedding
  - $\boldsymbol{h}_a$: attribute embedding
- Small distance for aligned entities for prediction

$$D(e_i, v_j) = \beta \frac{\|\boldsymbol{h}_s(e_i) - \boldsymbol{h}_s(v_j)\|_1}{d_s} + (1-\beta) \frac{\|\boldsymbol{h}_a(e_i) - \boldsymbol{h}_a(v_j)\|_1}{d_a}$$

  - d_s, d_a: dimensions of structure and attribute embedding
  - β: hyperparameter balancing importance of two embeddings
- For each entity e_i, return a list of candidate entities in KG_2

Wang, Zhichun, et al. "Cross-lingual knowledge graph alignment via graph convolutional networks." Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing. 2018.

GCN-Align – Experimental Results

- Datasets: DBP15K from DBpedia with different languages

$DBP15K_{JA-EN}$		$JA \rightarrow EN$			$EN \rightarrow JA$		
		Hits@1	Hits@10	Hits@50	Hits@1	Hits@10	Hits@50
*JE		18.92	39.97	54.24	17.80	38.44	52.48
*MTransE		27.86	57.45	75.94	23.72	49.92	67.93
*JAPE	SE w/o neg.	33.10	63.90	80.80	29.71	56.28	73.84
	SE	34.27	66.39	83.61	31.40	60.80	78.51
	SE + AE	36.25	68.50	85.35	38.37	67.27	82.65
JAPE′	SE w/o neg.	28.90	60.61	80.03	25.34	53.36	71.94
	SE	29.35	63.31	82.76	26.37	57.35	76.87
	SE + AE	31.06	64.11	81.57	32.45	62.21	79.08
GCN	SE	38.21	72.49	82.69	36.90	68.50	79.51
	SE + AE	**39.91**	**74.46**	**86.10**	**38.42**	**71.81**	**83.72**
$DBP15K_{FR-EN}$		$FR \rightarrow EN$			$EN \rightarrow FR$		
		Hits@1	Hits@10	Hits@50	Hits@1	Hits@10	Hits@50
*JE		15.38	38.84	56.50	14.61	37.25	54.01
*MTransE		24.41	55.55	74.41	21.26	50.60	69.93
*JAPE	SE w/o neg.	29.55	62.18	79.36	25.40	56.55	74.96
	SE	29.63	64.55	81.90	26.55	60.30	78.71
	SE + AE	32.39	66.68	83.19	32.97	65.91	82.38
JAPE′	SE w/o neg.	28.23	60.99	78.47	24.68	55.25	74.19
	SE	27.58	62.03	79.98	24.93	58.95	77.79
	SE + AE	30.21	65.81	82.57	31.42	63.86	80.95
GCN	SE	36.51	73.42	85.93	36.08	72.37	85.44
	SE + AE	**37.29**	**74.49**	**86.73**	**36.77**	**73.06**	**86.39**

Wang, Zhichun, et al. "Cross-lingual knowledge graph alignment via graph convolutional networks." Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing. 2018.

Overview of Part I

Part I: 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

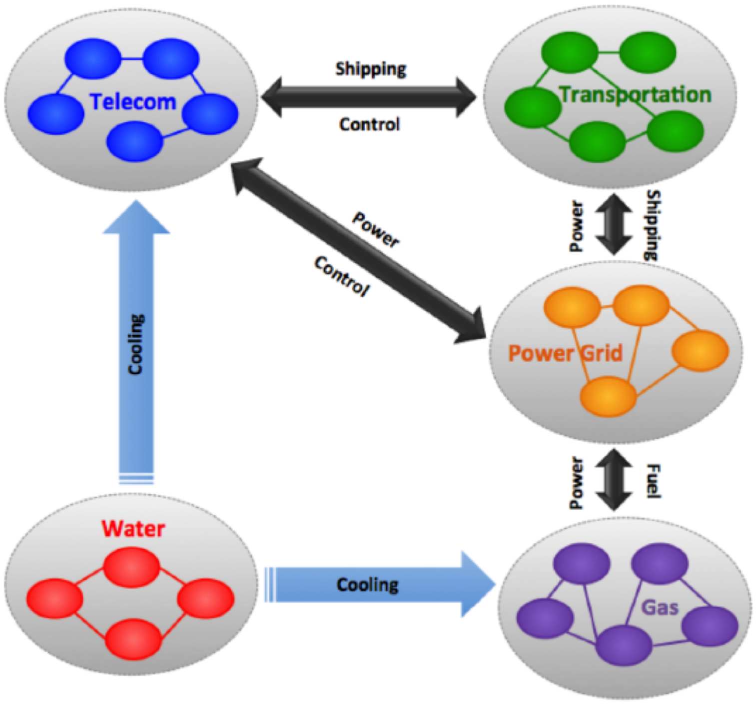
- Consistency-based
  - Single-level
  - Multilevel

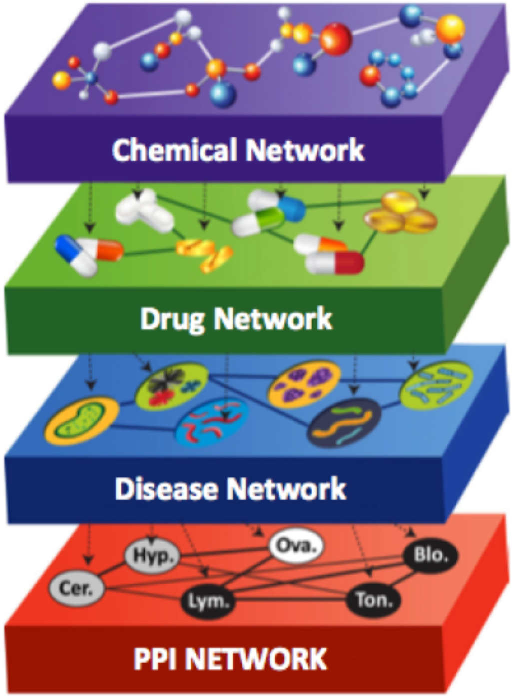
Related Tasks

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

Multi-layered Networks

- An example of multi-layered networks





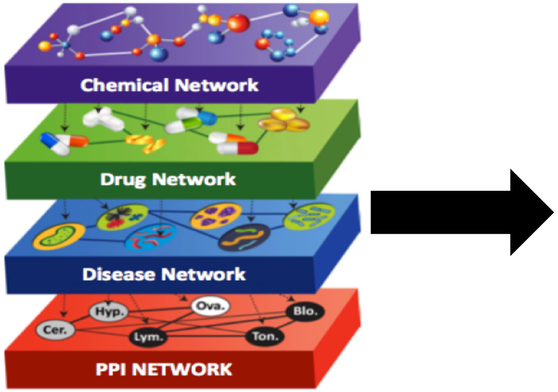
Infrastructure networks

Biological system networks

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.

Cross-Layer Dependency Inference

- 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 $\{\widetilde{\boldsymbol{D}}_{ij}\}$
- To link different types of nodes (alignment links same)



- $\boldsymbol{A}_1$ 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 $\mathcal{G}_2$

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:

$$\min_{\mathbf{F}_i \geq \mathbf{0}(i=1,\ldots,g)} J = \underbrace{\sum_{i,j:\ \mathbf{G}(i,j)=1} \|\mathbf{W}_{i,j} \odot (\mathbf{D}_{i,j} - \mathbf{F}_i \mathbf{F}_j{}')\|_F^2}_{\text{C1: Matching Observed Cross-Layer Dependencies}}$$

$$+ \underbrace{\alpha \sum_{i=1}^{g} \mathrm{tr}(\mathbf{F}_i{}'(\mathbf{T}_i - \mathbf{A}_i)\mathbf{F}_i)}_{\text{C2: Node Homophily}} + \underbrace{\beta \sum_{i=1}^{g} \|\mathbf{F}_i\|_F^2}_{\text{C3: Regularization}}$$

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

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

$$\Downarrow$$

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

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

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

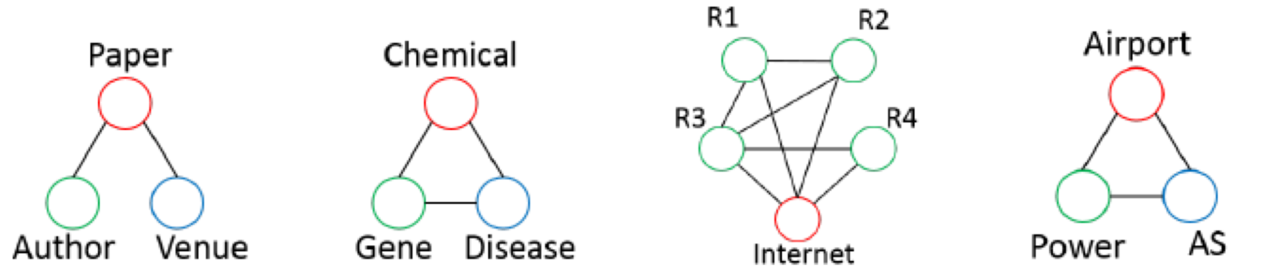
Chen, Chen, et al. "FASCINATE: fast cross-layer dependency inference on multi-layered networks." Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. 2016.

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

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

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

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

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 I

Part I: 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

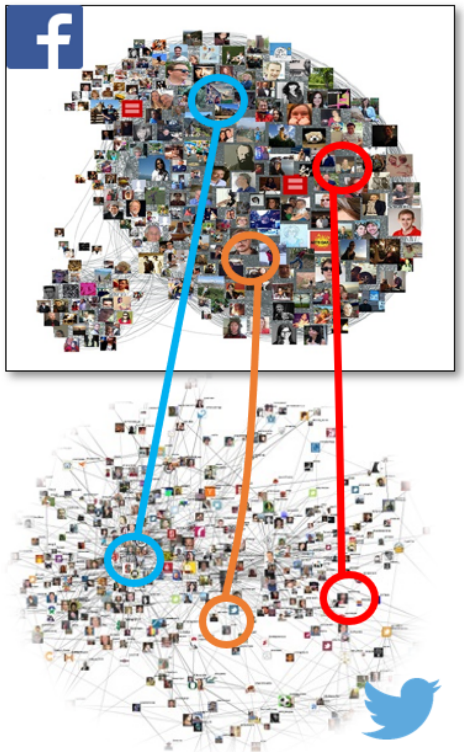
- Consistency-based
  - Single-level
  - Multilevel

Related Tasks

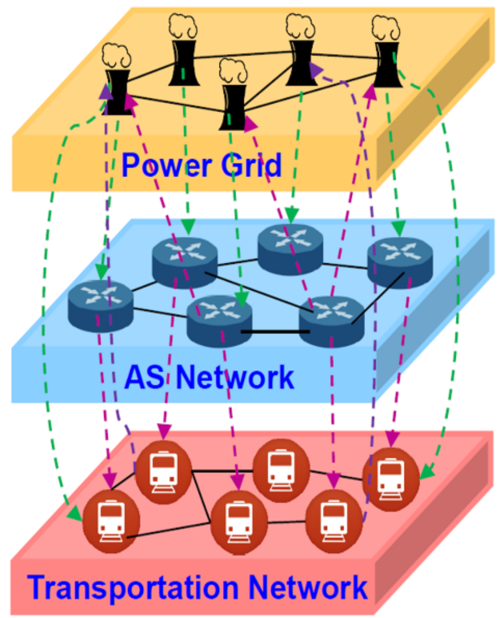
- Entity alignment
  - Non-GNN based
  - GNN-based
- Cross-layer inference
- **Cross-network transformation**

Cross-Network Node Associations

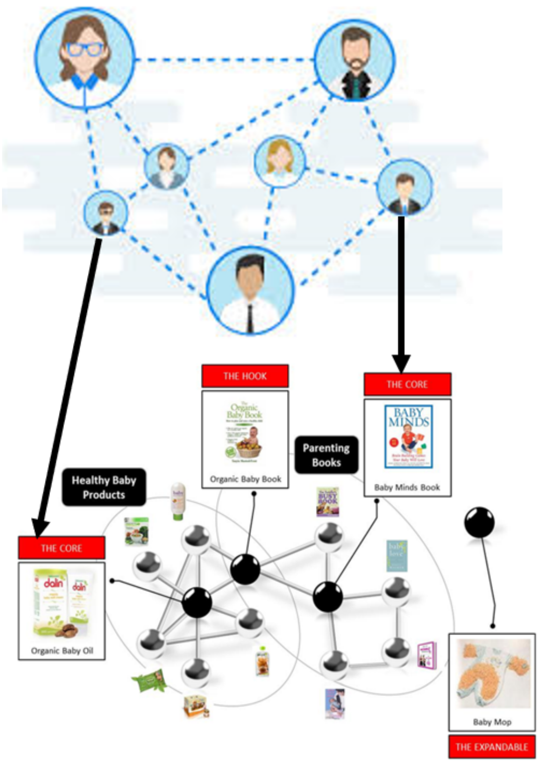
- Goal: to find node associations across different networks



Node Correspondence



Cross-Layer Node Dependency



Users Purchase Products

Zhang, Si, et al. "NetTrans: Neural Cross-Network Transformation." Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. 2020.

Limitations of Traditional Methods

- Linear and/or consistency assumptions

$$\min \ \|\boldsymbol{B}_0 - \boldsymbol{P}\boldsymbol{A}_0\boldsymbol{P}^T\|_F^2$$

$$\Downarrow$$

$$\min \ \left\|\mathrm{vec}(\boldsymbol{B}_0) - \boxed{\widetilde{\boldsymbol{P}}\mathrm{vec}(\boldsymbol{A}_0)}\right\|_2^2$$

Linear transformation

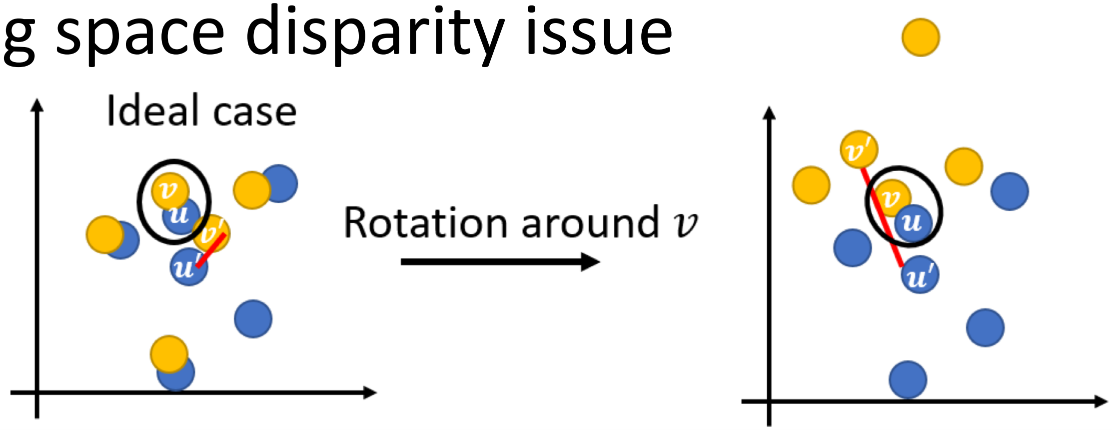
Graph matching-based network alignment

$$\min \ \left\|\boldsymbol{R} - \boldsymbol{U}_1^T\boldsymbol{U}_2\right\|_F^2 + \alpha \sum_i \boxed{\mathrm{Tr}\left(\boldsymbol{U}_i^T(\boldsymbol{D}_i - \boldsymbol{A}_i)\boldsymbol{U}_i\right)}$$

Network-based regularization

Factorization-based recommendation and cross-layer dependency inference

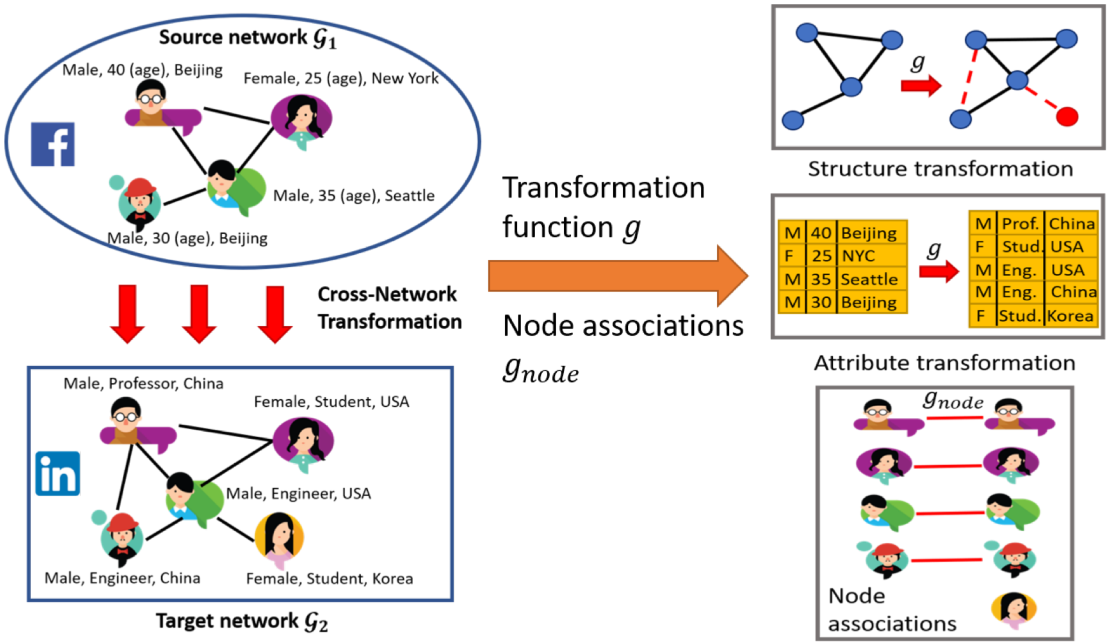
- Embedding space disparity issue



Zhang, Si, et al. "NetTrans: Neural Cross-Network Transformation." Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. 2020.

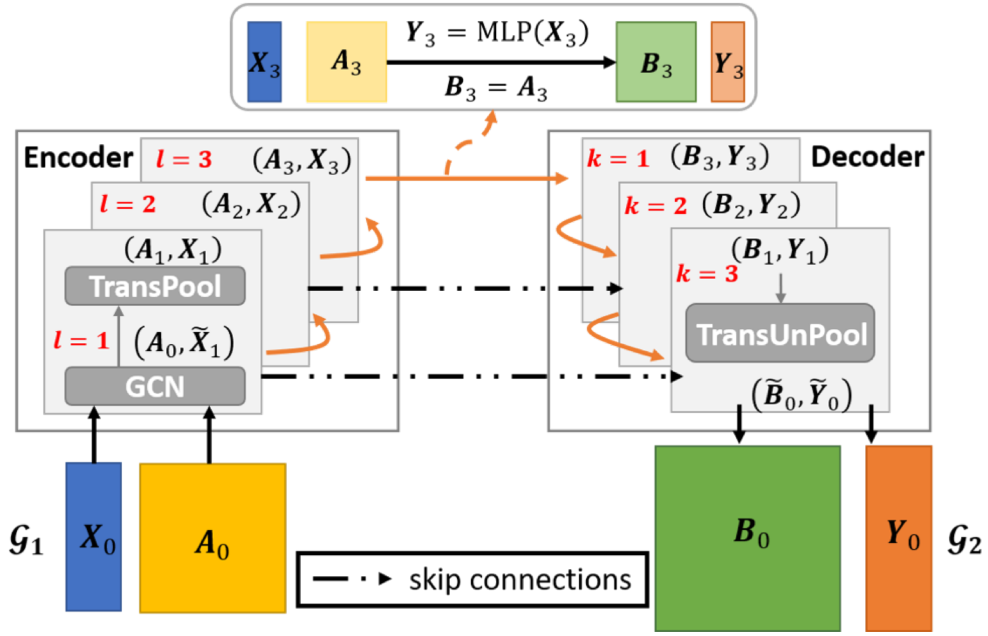
Cross-Network Transformation

- **Given:** (1) Source and target networks $\mathcal{G}_1 = \{\mathcal{V}_1, \boldsymbol{A}_0, \boldsymbol{X}_0\}$, $\mathcal{G}_2 = \{\mathcal{V}_2, \boldsymbol{B}_0, \boldsymbol{Y}_0\}$; Observed cross-network node associations $\boldsymbol{L}$
- **Output:** (1) Cross-network transformation function g, s.t. $g(\mathcal{G}_1) \approx \mathcal{G}_2$; (2) Node association function g_{node}



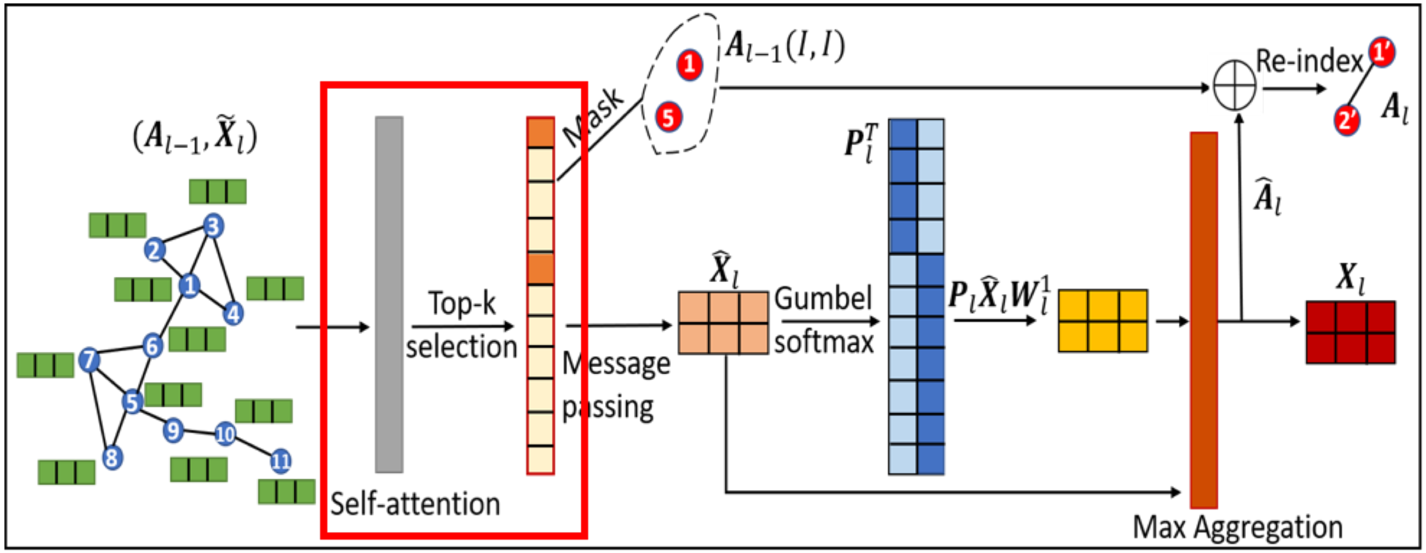
NetTrans – Model Overview

- Key idea: encoder-decoder architecture
  - Encoder: to coarsen source network at different resolutions
  - Decoder: to reconstruct target network at different resolutions



NetTrans – Encoder

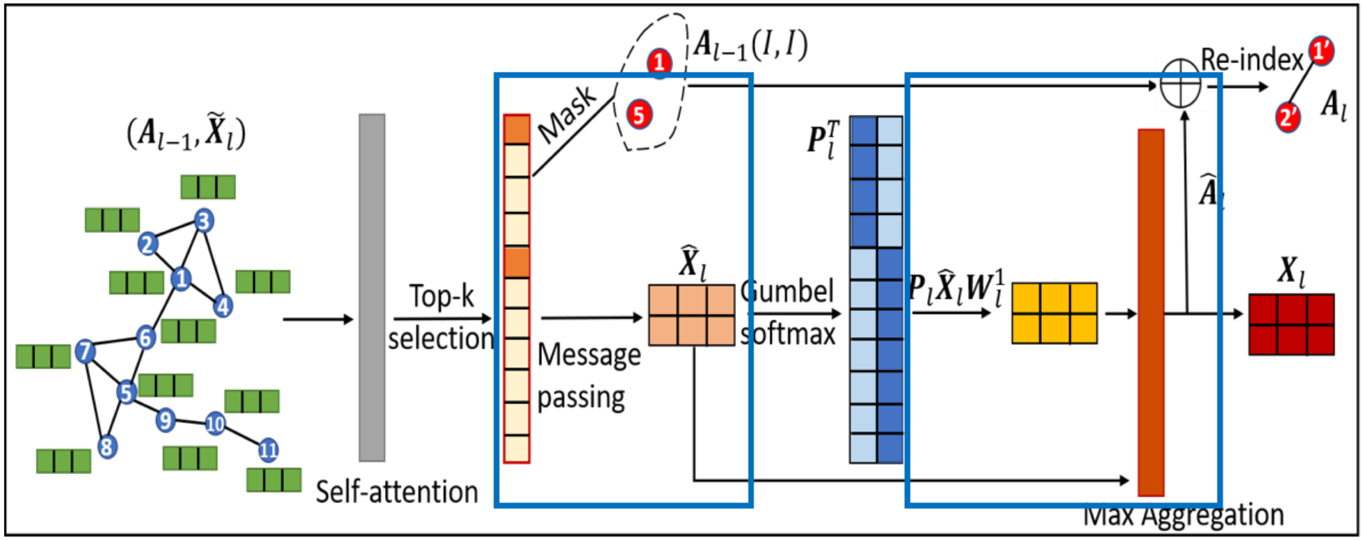
- Key component: TransPool as a pooling layer
- Supernode selection
  - Self-attention-based pooling



Zhang, Si, et al. "NetTrans: Neural Cross-Network Transformation." Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. 2020.

NetTrans – Encoder

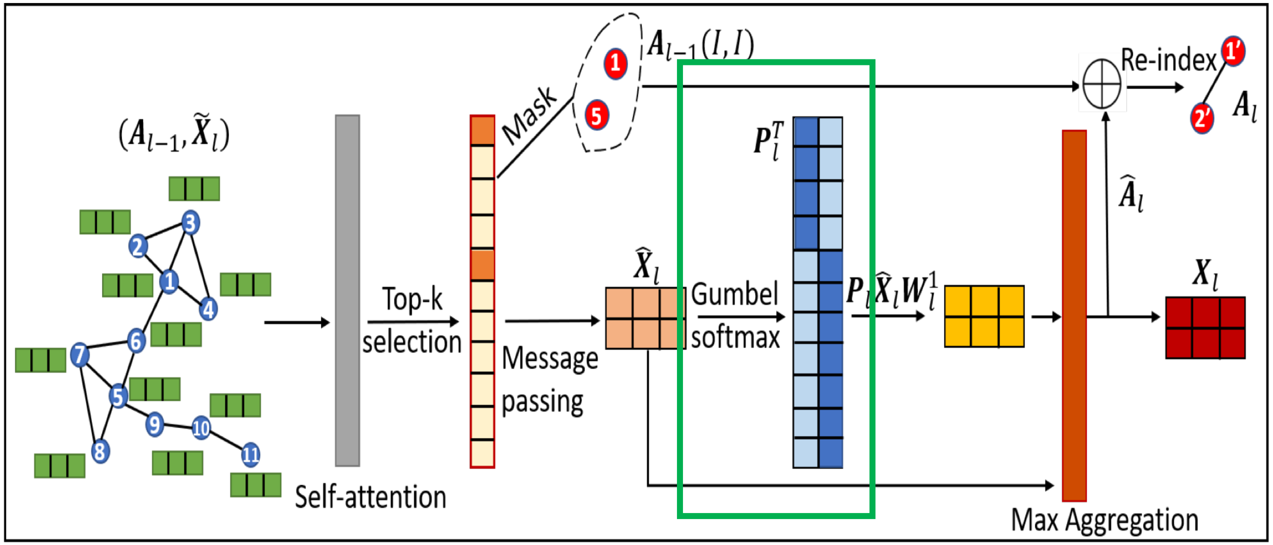
- Supernode representation learning
  - Attention-based message passing
  - Aggregation by node-to-supernode assignment



Zhang, Si, et al. "NetTrans: Neural Cross-Network Transformation." Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. 2020.

NetTrans – Encoder

- Node-to-supernode assignment
  - Gumbel softmax to approximate $\boldsymbol{P}$
  - Supernode candidate pruning

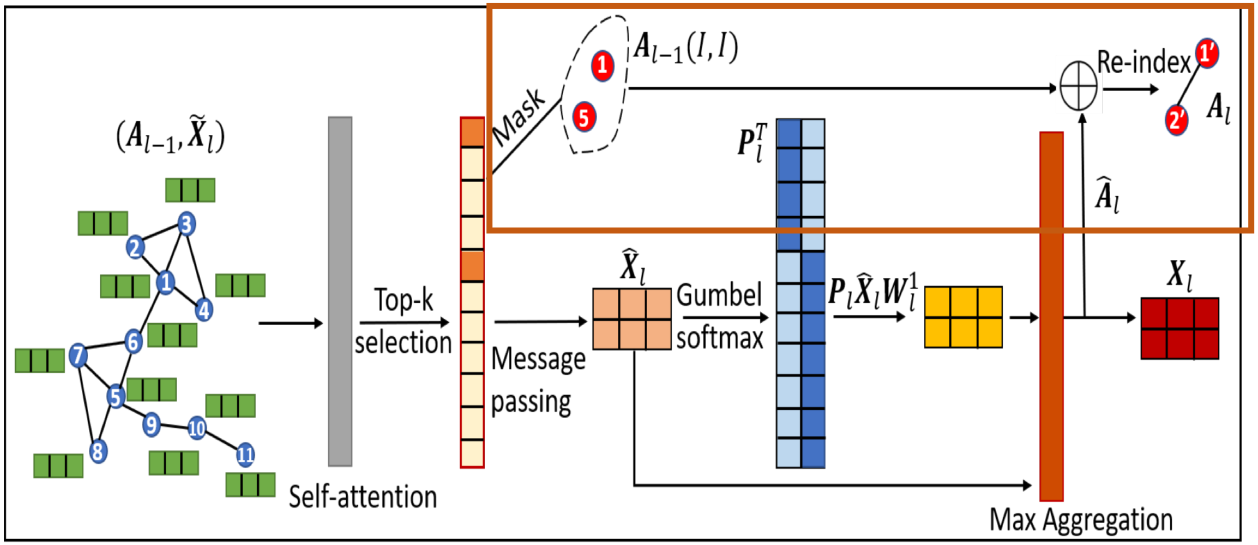


Zhang, Si, et al. "NetTrans: Neural Cross-Network Transformation." Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. 2020.

NetTrans – Encoder

- Supernode connections
  - Use auxiliary connections $\widehat{\boldsymbol{A}}_l$

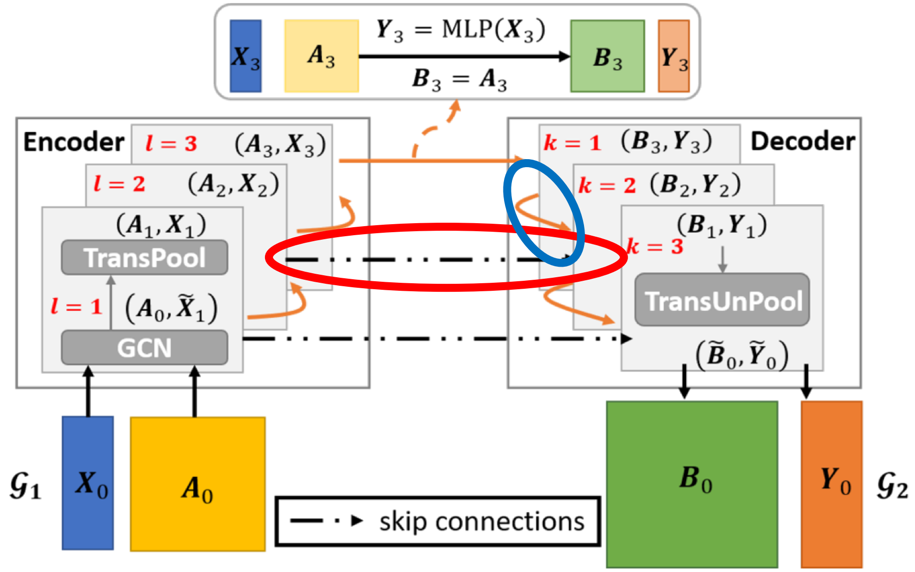
$$\boldsymbol{A}_l = \frac{1}{2}\left(\boldsymbol{A}_{l-1}(I, I) + \widehat{\boldsymbol{A}}_l\right)$$



Zhang, Si, et al. "NetTrans: Neural Cross-Network Transformation." Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. 2020.

NetTrans – Decoder

- Goal: to reconstruct target network
- Key idea: same latent meanings of supernodes
  - Part #1: leverage $\mathcal{G}_1$ by skip connections
  - Part #2: calibrate part #1 from supernodes to nodes



Zhang, Si, et al. "NetTrans: Neural Cross-Network Transformation." Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. 2020.

NetTrans – Experimental Results

- Effectiveness of NetTrans for network alignment

	Cora1-Cora2			ACM-DBLP			Foursquare-Twitter		
	Hits@10	Hits@30	Accuracy	Hits@10	Hits@30	Accuracy	Hits@10	Hits@30	Accuracy
NetTrans	**90.98%**	**97.51%**	**89.89%**	**84.09%**	**94.52%**	**58.21%**	**24.68%**	**34.58%**	**9.17%**
FINAL-N	88.73%	90.77%	87.58%	82.91%	90.71%	54.39%	24.09%	33.80%	8.47%
FINAL-P	62.28%	80.01%	54.34%	69.70%	83.12%	36.34%	24.09%	33.80%	8.47%
REGAL	60.90%	69.20%	46.26%	63.68%	71.80%	41.78%	0.15%	2.20%	0.11%
IONE	73.03%	79.92%	42.29%	58.93%	84.19%	33.00%	13.44%	28.17%	4.13%
CrossMNA	59.06%	68.62%	33.26%	42.54%	49.69%	21.04%	3.37%	14.79%	2.48%

Observation: NetTrans outperforms all other baselines for network alignment task

Zhang, Si, et al. "NetTrans: Neural Cross-Network Transformation." Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. 2020.

NetTrans – Experimental Results

- Effectiveness of NetTrans for social recommendation

	Ciao-0.2			Ciao-0.3			Ciao-0.5		
	Prec@10	Rec@10	Rec@50	Prec@10	Rec@10	Rec@50	Prec@10	Rec@10	Rec@50
NetTrans	**13.87%**	**11.08%**	**29.90%**	**11.01%**	**13.23%**	**28.15%**	**10.87%**	**12.43%**	**39.02%**
BPR	1.37%	0.6%	20.25%	1.38%	0.62%	20.18%	1.00%	0.37%	14.97%
wpZAN	11.99%	9.19%	20.77%	9.88%	10.33%	23.22%	9.85%	11.64%	26.04%
GraphRec	8.65%	6.62%	17.56%	8.42%	6.60%	18.07%	6.94%	6.63%	18.08%
SamWalker	4.94%	1.97%	5.98%	4.39%	2.07%	5.67%	2.48%	1.58%	4.05%
NGCF	2.77%	1.21%	3.26%	2.77%	1.48%	3.61%	3.17%	1.99%	4.77%

Observation: NetTrans outperforms all other baselines for recommendation task

Zhang, Si, et al. "NetTrans: Neural Cross-Network Transformation." Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. 2020.

RoadMap

- Motivations and Background
- Part I: Recent Network Alignment Algorithms
- Part II: Network Alignment Applications
- Part III: Future Research Directions

Zhang, Si, et al. "NetTrans: Neural Cross-Network Transformation." Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. 2020.

Overview of Part II

Part II: Network Alignment Applications

Social Analysis

- User identity linkage
- Recommendation
  - Friends
  - Products
- Information diffusion

Bioinformatics

- Identify functional orthologs and knowledge transfer
  - Evolutionary relationships
  - Human aging
- Connectome Analysis

Knowledge Base

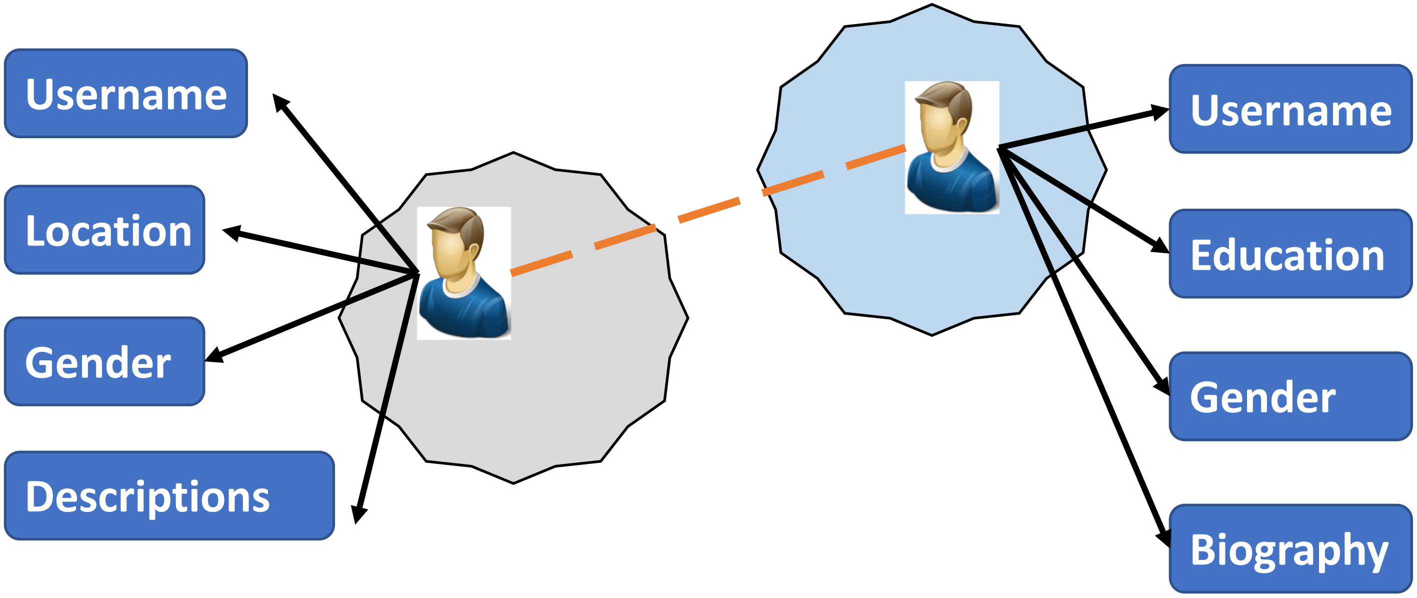
- Knowledge completion

Security

- Modeling adversarial activities

Social Analysis – User Identity Linkage

- User Identity Linkage
  - To identify the same physical user across social platforms



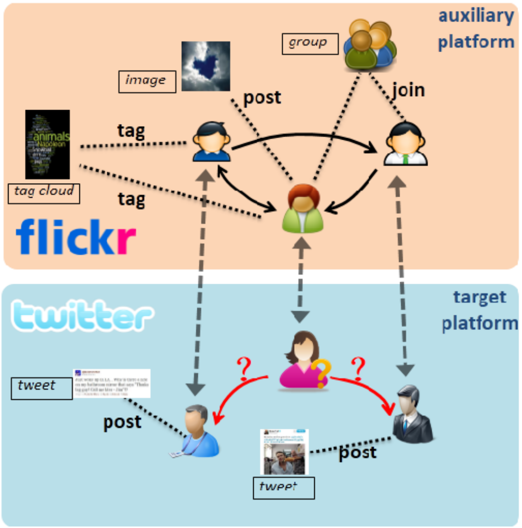
  - Can be used for de-anonymization, information integration, etc.

User Identity Linkage

- Existing methods:
  - Profile based [Zafarani’13, Zhang’14, Perito’11, Vosecky’09]
  - Network based [Zhou’16, Zhang’15, Liu’16]
  - Profile + network based [Zhang’15, Shen’14, Zhang’’16]
- Network-based can be considered as network alignment w/o attributes.
- Profile + network-based methods can be viewed as network alignment w/ attributes.

Social Analysis - Recommendation

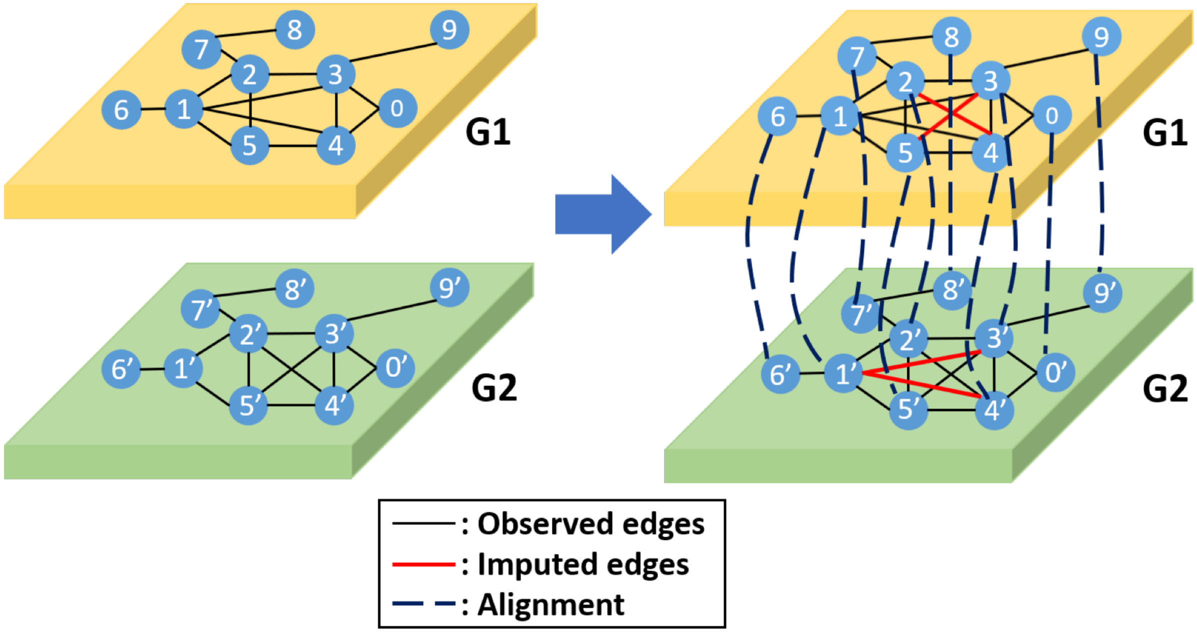
- Friend recommendation:
  - For two social networks, if we know
    - User u_1 is a friend of user u_2 in $\mathcal{G}_1$
    - User v_1 in $\mathcal{G}_2$ and user u_1 in $\mathcal{G}_1$ are same person
    - User v_2 in $\mathcal{G}_2$ and user u_2 in $\mathcal{G}_2$ are same person
    - But user v_1 and user v_2 are not friend in $\mathcal{G}_2$
  - Then, we can recommend v_1 to user v_2



Yan, Ming, et al. "Friend transfer: Cold-start friend recommendation with cross-platform transfer learning of social knowledge." *2013 IEEE International Conference on Multimedia and Expo (ICME)*. IEEE, 2013.
Nelakurthi, Arun Reddy, and Jingrui He. "Finding cut from the same cloth: Cross network link recommendation via joint matrix factorization." Thirty-First AAAI Conference on Artificial Intelligence. 2017.

Cross-Site Friend Recommendation

- Think of it as a cross-site link prediction problem
- Given two incomplete social networks, we jointly solve network alignment and link prediction problems



CENALP – Network Embedding

- DeepWalk-based network embedding
  - Key idea: build a world-view graph

$$\mathbf{W} = \begin{bmatrix} q \cdot \mathbf{P}_{\mathcal{G}} & (1-q) \cdot \mathbf{P}_{\mathcal{G}\mathcal{G}'} \\ (1-q) \cdot \mathbf{P}_{\mathcal{G}'\mathcal{G}} & q \cdot \mathbf{P}_{\mathcal{G}'} \end{bmatrix}$$

  - Within-network node sampling with a probability of q, and cross-network sampling with $(1-q)$
  - Allows for cross-network Skip-gram embedding
- Construction of $\boldsymbol{P}_{\mathcal{G}\mathcal{G}'}$ by structure and attribute

$$\text{dist} = \left| \min_{d \in s_k(u)} \log(d+1) - \min_{d \in s_k(u')} \log(d+1) \right| + \left| \max_{d \in s_k(u)} \log(d+1) - \max_{d \in s_k(u')} \log(d+1) \right|.$$

$$\text{sim}_{\text{attr}}(u, u') = \frac{\mathbf{y}_u^\top \cdot \mathbf{y}_{u'}}{||\mathbf{y}_u||_2 \cdot ||\mathbf{y}_{u'}||_2},$$

Du, Xingbo, et al. "Cross-network Skip-gram Embedding for Joint Network Alignment and Link Prediction." IEEE Transactions on Knowledge and Data Engineering (2020).

CENALP – Network Alignment and Link Prediction

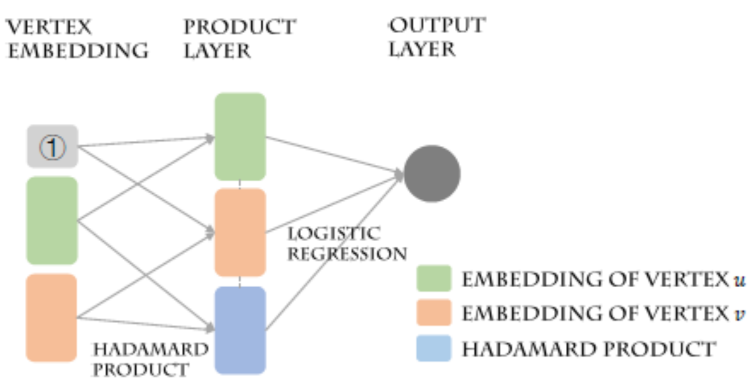
- Greedy alignment by embedding-based similarity
  - Given embeddings of u, u' in two networks

$$\text{sim}_{\text{emb}}(u, u') = \frac{\mathbf{x}_u^\top \cdot \mathbf{x}_{u'}}{||\mathbf{x}_u||_2 \cdot ||\mathbf{x}_{u'}||_2}$$

  - Greedy-based alignment objective

$$u^*, u'^* = \arg\max_{u,u'} \quad \text{sim}_{\text{emb}}(u, u')$$

- Embedding for link prediction



$$\text{layer}_0(\mathcal{I}) = \underset{\substack{\mathbf{x}_u, \mathbf{x}_v \in \mathcal{X}(\mathcal{I}) \\ \mathbf{x}_u \neq \mathbf{x}_v}}{\text{concatenate}} (\mathbf{x}_u \circ \mathbf{x}_v),$$

$$\text{layer}_1(\mathcal{I}; w, b) = w \cdot \text{layer}_0(\mathcal{I}) + b \triangleq \hat{w} \cdot \widehat{\text{layer}}_0(\mathcal{I}),$$

$$\text{layer}_2(\mathcal{I}; \hat{w}) = \sigma\left[\text{layer}_1(\mathcal{I}; \hat{w})\right] = \frac{1}{1 + e^{-\text{layer}_1(\mathcal{I}; \hat{w})}},$$

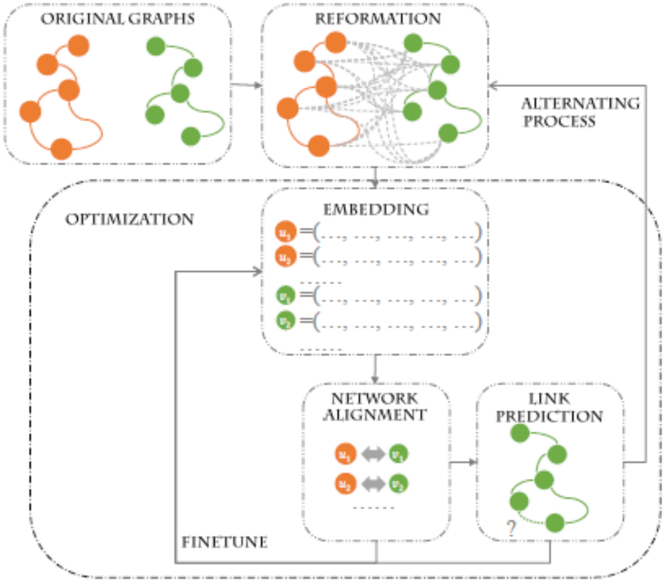
Du, Xingbo, et al. "Cross-network Skip-gram Embedding for Joint Network Alignment and Link Prediction." IEEE Transactions on Knowledge and Data Engineering (2020).

CENALP – Algorithm

- Objective function

$$\mathcal{L} = \sum_{\omega \in walks} \sum_{u_i \in \omega} \Bigg[\sum_{u_j \in C_{u_i}} \log \sigma(\mathrm{x}_{u_j}^{out\top} \cdot \mathrm{x}_{u_i}^{in}) + \sum_{k=1}^{K_{neg}} \mathbb{E}_{u_k \sim R_k(u)} \log \sigma(-\mathrm{x}_{u_k}^{out\top} \cdot \mathrm{x}_{u_i}^{in}) \Bigg].$$

- Overall procedure



CENALP – Experimental Results

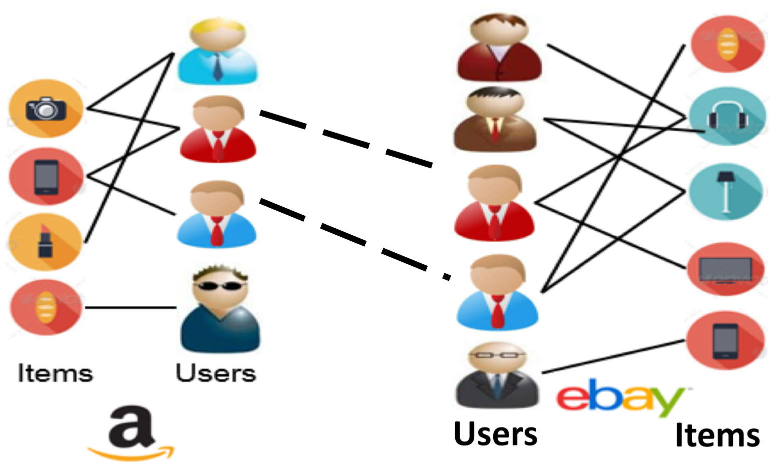
- AUC score of link prediction

Dataset	Method	Training rate.									
		95%	90%	85%	80%	75%	70%	65%	60%	55%	50%
DBLP ↕ disturbed copy	JC [11]	92.28%	90.93%	90.37%	88.44%	85.43%	84.12%	80.51%	77.99%	74.81%	72.16%
	AA [12]	92.33%	91.01%	90.30%	88.32%	85.44%	84.07%	80.48%	77.94%	74.82%	72.14%
	SC [13]	78.84%	78.89%	78.68%	78.54%	74.72%	77.21%	76.83%	75.87%	73.91%	74.40%
	n2v [22]	78.40%	81.34%	78.72%	78.32%	75.01%	74.30%	73.32%	71.60%	70.56%	68.61%
	n2v+LR	95.58%	95.44%	94.30%	94.30%	92.37%	93.07%	90.86%	90.80%	89.25%	88.08%
	CLF [8]	96.92%	96.55%	95.70%	95.69%	94.43%	94.28%	92.95%	92.93%	92.23%	90.68%
	MNN [43]	98.13%	97.76%	97.57%	97.28%	97.10%	96.78%	96.38%	96.44%	96.03%	95.86%
	CE-CLF	98.37%	98.29%	97.88%	97.80%	97.34%	96.74%	96.75%	96.36%	96.09%	95.86%
	CELP	98.48%	98.39%	98.36%	98.57%	98.28%	98.49%	98.58%	98.31%	98.38%	98.27%
	CENALP	**99.65%**	**99.27%**	**99.35%**	**99.38%**	**99.36%**	**99.08%**	**99.01%**	**99.29%**	**99.25%**	**99.06%**
Facebook ↕ Twitter	JC [11]	74.76%	77.89%	76.94%	75.64%	73.15%	72.63%	70.48%	68.90%	67.25%	65.12%
	AA [12]	74.77%	77.54%	76.57%	75.76%	73.36%	72.68%	71.09%	68.84%	67.19%	65.18%
	SC [13]	84.39%	83.39%	86.88%	84.53%	83.83%	83.79%	81.56%	80.80%	81.81%	77.61%
	n2v [22]	75.62%	78.94%	78.23%	79.36%	76.18%	75.25%	74.64%	74.86%	74.49%	71.16%
	n2v+LR	82.20%	85.58%	83.04%	84.51%	81.64%	82.07%	81.17%	81.02%	80.08%	78.04%
	CLF [8]	84.88%	85.02%	86.18%	86.70%	84.00%	83.99%	82.95%	82.43%	81.96%	80.75%
	MNN [43]	95.72%	96.44%	96.28%	96.30%	96.21%	96.25%	96.07%	95.88%	95.47%	95.23%
	CE-CLF	96.52%	96.84%	96.37%	96.34%	96.30%	95.69%	94.92%	94.31%	93.11%	92.00%
	CELP	97.29%	97.52%	97.46%	**97.85%**	**97.99%**	97.56%	97.23%	**97.15%**	96.66%	96.14%
	CENALP	**97.31%**	**97.77%**	**97.51%**	97.24%	97.60%	**97.86%**	**97.34%**	96.47%	**96.74%**	**96.16%**

Du, Xingbo, et al. "Cross-network Skip-gram Embedding for Joint Network Alignment and Link Prediction." IEEE Transactions on Knowledge and Data Engineering (2020).

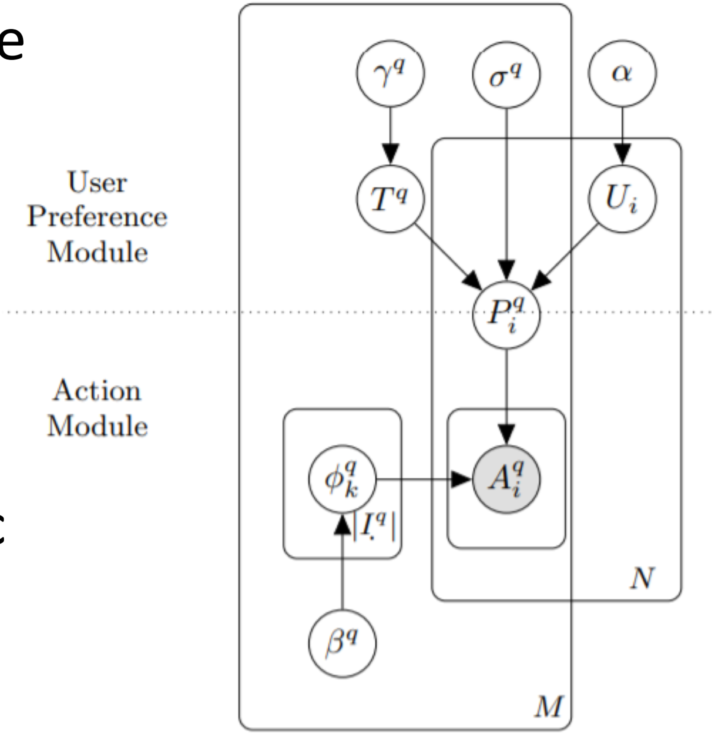
Social Analysis - Recommendation

- Cross-site product recommendation:
  - Intuition: if users are aligned, purchase histories can be combined for better recommendation
  - Key idea: leverage cross-site actions to improve user modeling
  - Benefits: may mitigate issues, e.g., cold start, etc.



JUMA – Approach

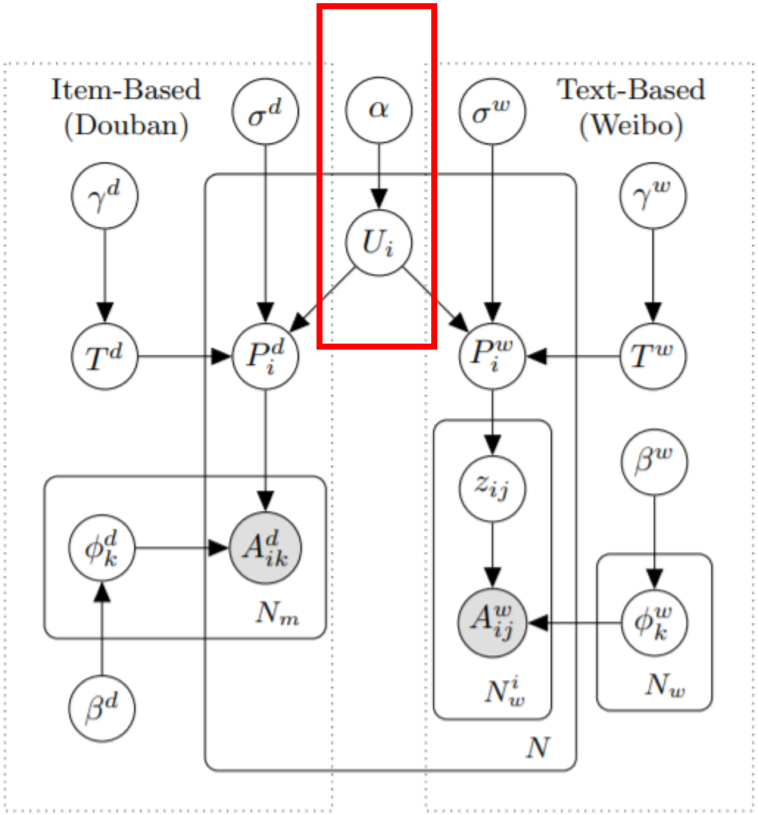
- Key idea: use a probabilistic graphical model for joint user modeling over aligned sites
  - User's site-specific preference P_i^q is transferred from universal preference U_i by transferring model T^q.
  - User conducts actions A_i^q based on P_i^q and site-specific item models $\{\phi_k^q\}$.



Cao, Xuezhi, and Yong Yu. "Joint user modeling across aligned heterogeneous sites." Proceedings of the 10th ACM Conference on Recommender Systems. 2016.

JUMA – Approach

- Joint user modeling over aligned sites
  - For item-based site (Douban), use matrix factorization method.
  - For text-based site (Weibo), use Latent Dirichlet Allocation (LDA) to model topic distributions for microblogs.



Cao, Xuezhi, and Yong Yu. "Joint user modeling across aligned heterogeneous sites." Proceedings of the 10th ACM Conference on Recommender Systems. 2016.

JUMA – Experimental Results

- Effectiveness of recommendation

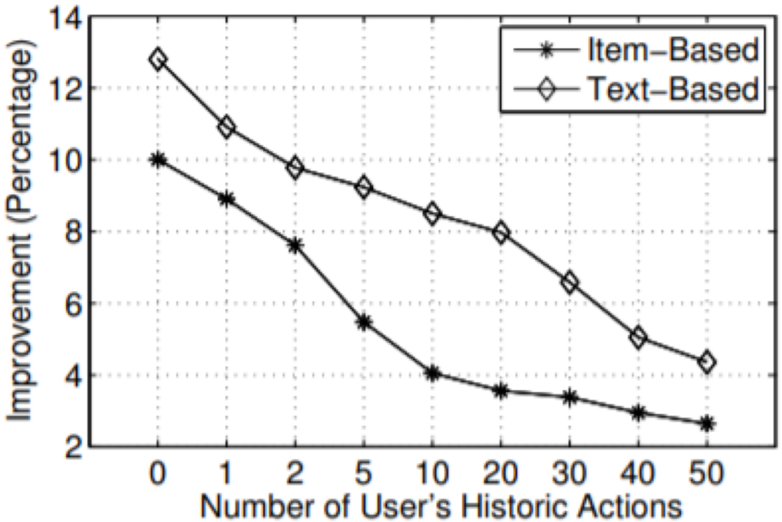
TARGET	ALGS	AUC SCORE, VARYING TRAINING INFORMATION RATIO					
		0.4	0.5	0.6	0.7	0.8	0.9
Text-Based (Weibo)	LDA	0.6514 ± 0.0017	0.6605 ± 0.0015	0.6694 ± 0.0016	0.6769 ± 0.0018	0.6839 ± 0.0015	0.6928 ± 0.0014
	JUMA	**0.6824 ± 0.0014**	**0.6892 ± 0.0016**	**0.6976 ± 0.0014**	**0.7058 ± 0.0017**	**0.7120 ± 0.0012**	**0.7194 ± 0.0013**
	CTR	0.7021 ± 0.0021	0.7133 ± 0.0017	0.7262 ± 0.0018	0.7352 ± 0.0017	0.7432 ± 0.0016	0.7532 ± 0.0015
	JUMA$^+$	**0.7338 ± 0.0017**	**0.7420 ± 0.0015**	**0.7502 ± 0.0018**	**0.7592 ± 0.0015**	**0.7670 ± 0.0015**	**0.7743 ± 0.0014**
Item-Based (Douban)	PMF	0.7275 ± 0.0016	0.7323 ± 0.0013	0.7384 ± 0.0015	0.7428 ± 0.0016	0.7485 ± 0.0014	0.7521 ± 0.0013
	SVD++	0.7856 ± 0.0012	0.7929 ± 0.0010	0.7986 ± 0.0016	0.8055 ± 0.0012	0.8089 ± 0.0013	0.8112 ± 0.0011
	TMF	0.7872 ± 0.0015	0.7946 ± 0.0012	0.8001 ± 0.0013	0.8071 ± 0.0019	0.8102 ± 0.0014	0.8132 ± 0.0012
	mmTM	0.6929 ± 0.0019	0.6940 ± 0.0011	0.6943 ± 0.0012	0.6963 ± 0.0015	0.7034 ± 0.0010	0.7064 ± 0.0018
	JUMA	**0.8127 ± 0.0017**	**0.8172 ± 0.0016**	**0.8219 ± 0.0013**	**0.8235 ± 0.0011**	**0.8243 ± 0.0015**	**0.8259 ± 0.0013**

Observation: JUMA performs best for both text-based site Weibo and item-based site Douban.

Cao, Xuezhi, and Yong Yu. "Joint user modeling across aligned heterogeneous sites." Proceedings of the 10th ACM Conference on Recommender Systems. 2016.

JUMA – Experimental Results

- Effectiveness of addressing cold-start

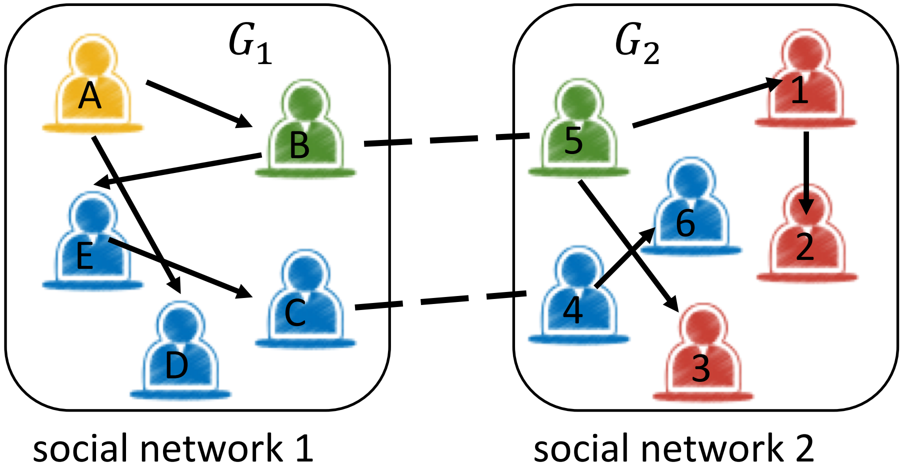


Observation: Improvements are higher when dealing with cold users than non-cold users.

Cao, Xuezhi, and Yong Yu. "Joint user modeling across aligned heterogeneous sites." Proceedings of the 10th ACM Conference on Recommender Systems. 2016.

Social Analysis – Information Diffusion

- Motivations
  - Users can post messages in multiple platforms;
  - Information thus propagates within-network and across networks.



M&M – Approach

- Goal: multi-aligned multi-relational network influence maximizer
- Key idea: to extends traditional linear threshold to depict diffusion across networks
- Activation probability functions:
  - For intra-network relation *i*

$$g_{v,i}^{(1)}(t+1) = \frac{\sum_{u \in \Gamma_{in}(v,i)} \phi^{i}_{(u,v)} \varphi(u,t)}{\sum_{u \in \Gamma_{in}(v,i)} \phi^{i}_{(u,v)}}$$

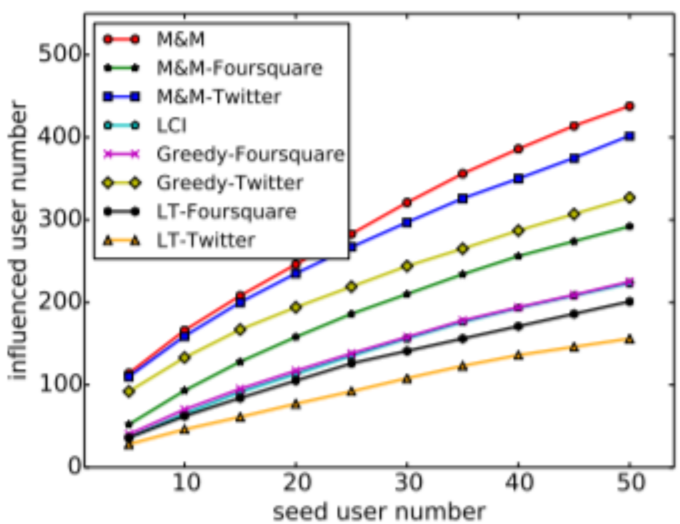
  - For inter-network relation *j*

$$h_{v,j}^{(1)}(t+1) = \frac{\sum_{u \in \Gamma_{in}(v,j)} \phi^{j}_{(u,v)} \varphi(u,t)}{\sum_{u \in \Gamma_{in}(v,j)} \phi^{j}_{(u,v)}}$$

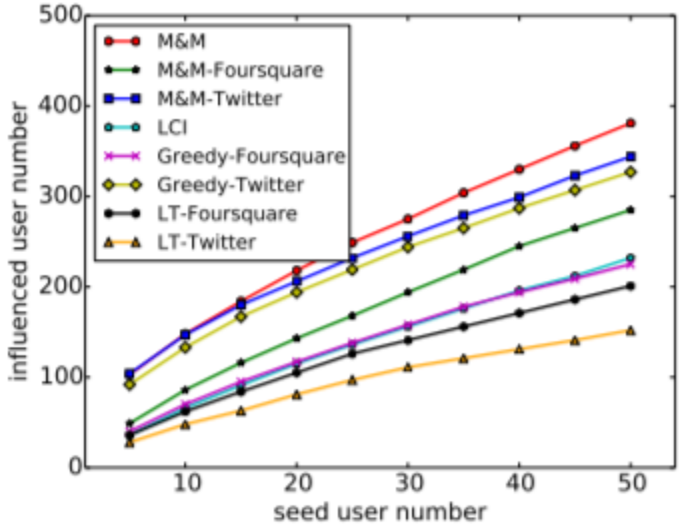
Zhan, Qianyi, et al. "Influence maximization across partially aligned heterogenous social networks." Pacific-Asia Conference on Knowledge Discovery and Data Mining. Springer, Cham, 2015.

M&M – Experimental Results

- Effectiveness of influence maximization
- Metric: # of activated users by the seed users



(a) 30% anchor user



(b) 60% anchor user

Zhan, Qianyi, et al. "Influence maximization across partially aligned heterogenous social networks." Pacific-Asia Conference on Knowledge Discovery and Data Mining. Springer, Cham, 2015.

Overview of Part II

Part II: Network Alignment Applications

Social Analysis

- User identity linkage
- Recommendation
  - Friends
  - Products
- Information diffusion

Bioinformatics

- Identify functional orthologs and knowledge transfer
  - Evolutionary relationships
  - Human aging

Knowledge Base

- Knowledge completion

Security

- Modeling adversarial activities

Bioinformatics – Knowledge Transfer

- Motivations:
  - Traditional methods are based on sequence alignment
  - Network data and sequence data provide complementary insights
  - Restricting to sequences may limit knowledge transfer
- Network alignment to identify functional orthologs
  - Benefits: insightful for knowledge of aging and other biological processes.

Faisal, Fazle E., et al. "The post-genomic era of biological network alignment." EURASIP Journal on Bioinformatics and Systems Biology 2015.1 (2015): 3.

Knowledge Transfer – Evolutionary Relationships Discovery

- Goal: using network alignment to guide biological knowledge transfer
  - From well-studied species to less well-studied species
- Methods:
  - GRAAL and H-GRAAL: focused on phylogenetic tree inference based on metabolic networks
  - MI-GRAAL:
    - Used these PPI network data to infer evolutionary relationships
    - Considered five herpesviruses based on their network similarities.

Faisal, Fazle E., et al. "The post-genomic era of biological network alignment." EURASIP Journal on Bioinformatics and Systems Biology 2015.1 (2015): 3.

Knowledge Transfer – Human Aging Discovery

- Motivations:
  - Susceptibility to diseases increases with age
  - Important to study molecular mechanisms behind aging and aging-associated diseases
- Traditional methods:
  - Transferring knowledge from well-studied species to human between conserved sequence regions
- Network alignment-based methods:
  - MI-GRAAL and IsoRankN: align well known aging-related network parts of one species to known aging-related network parts of other species

Faisal, Fazle E., et al. "The post-genomic era of biological network alignment." EURASIP Journal on Bioinformatics and Systems Biology 2015.1 (2015): 3.

Overview of Part II

Part II: Network Alignment Applications

Social Analysis

- User identity linkage
- Recommendation
  - Friends
  - Products
- Information diffusion

Bioinformatics

- Identify functional orthologs and knowledge transfer
  - Evolutionary relationships
  - Human aging

Knowledge Base

- Knowledge completion

Security

- Modeling adversarial activities

Knowledge Completion

- Goal: to complete a triple (h, r, t) when one of h, r, t is missing
- Application scenario by entity alignment:
  - Two sets of triplets (i.e., KGs) for training
  - One set of triplets for testing
  - Two training KGs can be aligned
- Methods:
  - Basically can be any KG alignment methods
  - ITransE/IPTransE for example

Zhu, Hao, et al. "Iterative Entity Alignment via Joint Knowledge Embeddings." IJCAI. Vol. 17. 2017.

ITransE – Experimental Results

- Effectiveness of ITransE for knowledge completion

Metric	Entity Prediction				Relation Prediction			
	Mean Rank		Hits@10		Mean Rank		Hits@1	
	Raw	Filter	Raw	Filter	Raw	Filter	Raw	Filter
MTransE (LT)	240.8	131.3	36.4	47.3	37.2	36.9	48.3	56.9
MTransE (TB)	851.3	759.7	9.4	10.8	293.7	293.4	27.4	27.7
TransE	246.1	131.6	42.5	54.3	55.9	55.6	44.2	50.7
TransE + Aux	232.8	121.5	43.3	54.9	50.1	49.8	44.4	50.9
ITransE (SA)	**209.2**	**101.0**	**44.2**	**55.1**	**19.8**	**19.6**	**54.2**	**60.7**
PTransE	213.0	97.2	50.9	72.1	2.33	1.96	67.4	86.9
PTransE + Aux	206.3	80.4	52.7	80.7	2.34	1.93	**68.8**	90.5
IPTransE (SA)	**197.5**	**70.6**	**53.0**	**80.8**	**2.03**	**1.62**	68.6	**90.8**

Observation: By successfully leveraging the auxiliary information (i.e., second KG by alignment), ITransE and IPTransE perform better than other baseline methods.

Zhu, Hao, et al. "Iterative Entity Alignment via Joint Knowledge Embeddings." IJCAI. Vol. 17. 2017.

Overview of Part II

Part II: Network Alignment Applications

Social Analysis
- User identity linkage
- Recommendation
  - Friends
  - Products
- Information diffusion

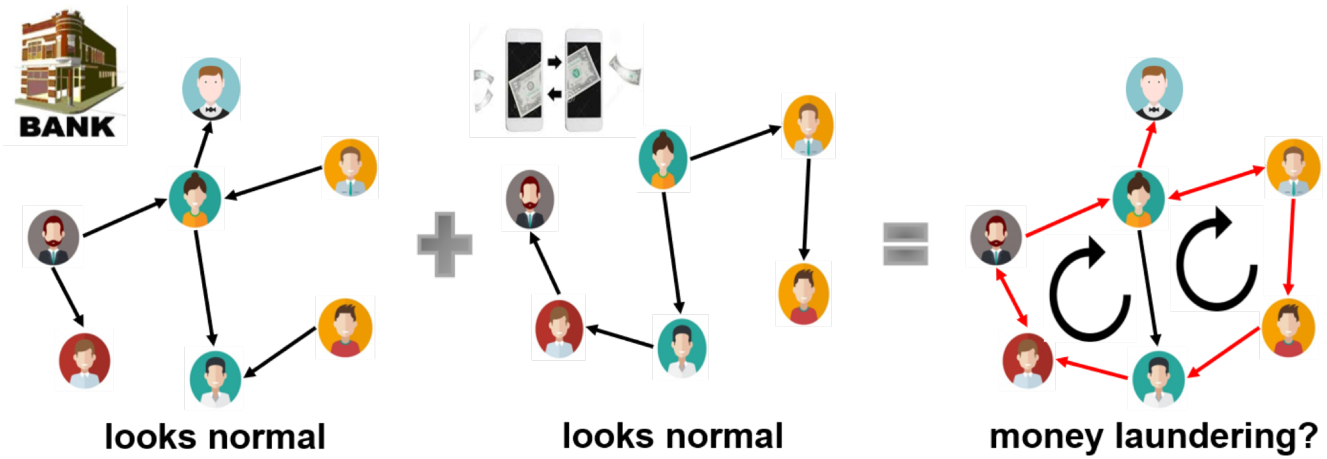
Bioinformatics
- Identify functional orthologs and knowledge transfer
  - Evolutionary relationships
  - Human aging

Knowledge Base
- Knowledge completion

Security
- Modeling adversarial activities

Security – Modeling Adversarial Activities

- Background:
  - Networks are natural structure to model adversarial activities
    - Smuggling
    - Illegal arm dealing
    - Illicit drug production
  - But such activities are often embedded in different domains



Xu, Jiejun, et al. "GTA3 2018: Workshop on Graph Techniques for Adversarial Activity Analytics." Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining. 2018.

MAA – Challenges

- Domain heterogeneity
  - Communication networks
    - Phone call, emails, text, etc.
  - People who call each other may unlikely text often
  - Similarly, email network is structurally distinct from phone call network
- Spatial-temporal challenge
  - Relations contain much spatial-temporal information
    - Who calls whom at which location and at what time
- Very large-scale networks

MAA – Approaches

- Any scalable network alignment methods
  - w/o attribute: only based on connections
  - w/ attribute: view spatial-temporal information as attributes
- Encode temporal information:
  - Count # of connections in certain time window
  - Values at all time windows form node attributes
  - Can be used as attribute-based prior similarity matrix
  - And/or as the attributes in attributed alignment methods (e.g., FINAL)

RoadMap

- Motivations and Background ✓
- Part I: Recent Network Alignment Algorithms ✓
- Part II: Network Alignment Applications ✓
- Part III: Future Research Directions

Big Network Alignment – 4Vs

- 4V characteristics also hold for networks

Big Network Alignment – Volume

- Real-world networks are very large-scale
  - Facebook, Instagram, Twitter have billions of users
- **Challenge:** most of existing methods have at least $O(n^2)$ complexity
  - Some recent consistency-based and embedding-based methods reduce the complexity to linear
  - Complexity may be even larger if we handle multiple networks collectively
- **Question:** how to efficiently do network alignment?
- **Possible directions:** (1) leverage approximation techniques, (2) parallelizable algorithm

Big Network Alignment – Variety

- Real-world networks have rich information
  - Node/edge attributes, text descriptions, temporal information
- Methods exist to handle attribute information
  - But few can handle temporal relation information
  - Who called whom at what time, etc.
- **Question:** how to better incorporate side-information into network alignment?
- **Possible directions:** heterogeneous network alignment, temporal network alignment, etc.

Big Network Alignment – Variety

- Network heterogeneity
  - Networks to be aligned carry different types of information
  - Even same user may behave differently in different networks
- Existing methods explicitly or implicitly build upon consistency assumptions
  - But network heterogeneity may easily violate this assumption
- **Questions:**
  - How to align different types of networks (e.g., LinkedIn vs. FB)?
  - How to adaptively control consistency assumption?
- **Possible directions:** Deep learning methods that are highly learnable.

Big Network Alignment – Velocity

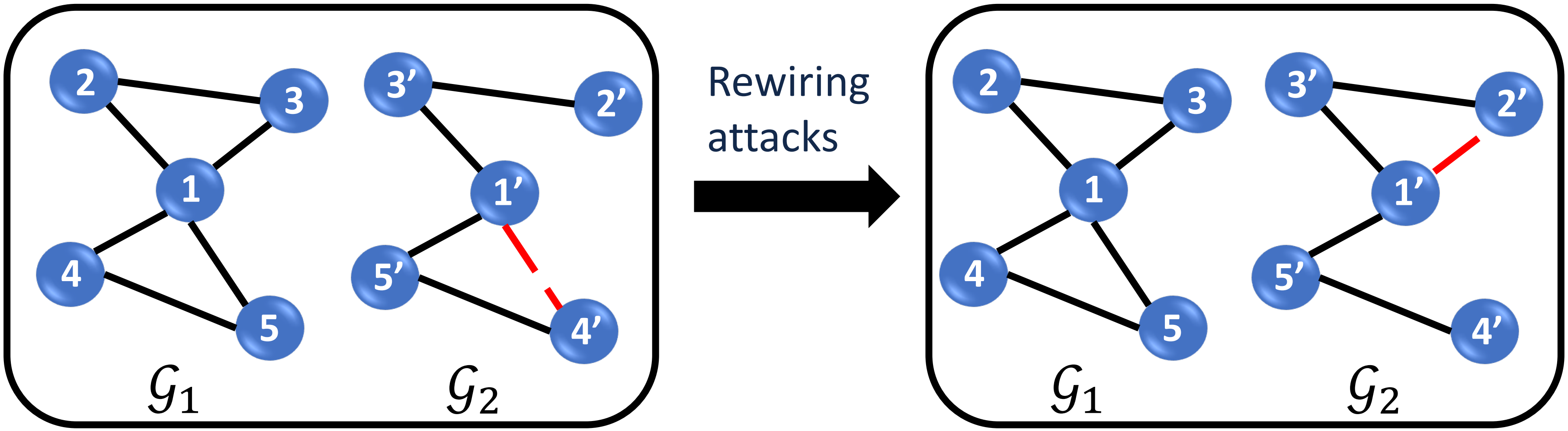
- Networks are dynamically changing over time.
- Dynamic network alignment
  - Simple solution: run from scratch at each timestamp
  - Limitation: time consuming; can't capture dynamics
- **Questions:**
  - How to efficiently handle alignment over dynamic networks?
  - How to leverage the dynamics (e.g., smoothness)?
- **Possible directions:**
  - Matrix approximation to avoid unnecessary re-computations.
  - Dynamic network embedding-based alignment methods.

Big Network Alignment – Veracity

- Real-world networks are often noisy and incomplete.
  - Missing connections
  - Missing nodes
  - Missing attribute information
- Existing methods:
  - Jointly solve network alignment and link prediction
  - Benefit: if handled properly, they mutually benefit each other
- **Challenge:** error propagation
  - If alignment or imputed edges are not correct, the performance will be hurt.

Adversarial Network Alignment

- Improve the alignment effectiveness and robustness
- Noise/adversarial attacks can mislead alignment



Adversarial Network Alignment

- **Background:**
  - 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.
- **Possible direction:**
  - Graph neural network-based adversarial learning on network alignment

Integrated Network Alignment

- Explainable network alignment
  - **Background:** there exist explainable network mining tasks
    - Network embedding
    - Graph neural networks
    - Ranking, clustering, etc.
  - **Problem goal:**
    - Explain why two nodes should be aligned or not
  - **Possible directions:**
    - Extend explainable network embedding to embedding-based network alignment

Integrated Network Alignment

- Fair network alignment
  - **Background:**
    - Fairness has been studied recently in many machine learning and data mining tasks
    - Fairness in graphs has attracted attentions very recently, but for single network
  - **Problem goal:**
    - To debias the network alignment
  - **Possible direction:**
    - Extend fairness in single network mining to multiple networks first, then combine the specific objective of network alignment

Summary

- Background and motivation
  - Network alignment aims to find node correspondence across networks
  - A key step to many mining tasks across multiple networks
- Recent network alignment algorithms
  - Pairwise network alignment
  - Collective network alignment
  - Higher-order network alignment
  - Other related tasks
- Network alignment applications
- Future directions

References

- 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.
- Liao, Chung-Shou, et al. "IsoRankN: spectral methods for global alignment of multiple protein networks." Bioinformatics 25.12 (2009): i253-i258.
- Bayati, Mohsen, et al. "Algorithms for large, sparse network alignment problems." 2009 Ninth IEEE International Conference on Data Mining. IEEE, 2009.
- 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, Li, et al. "Aligning Users across Social Networks Using Network Embedding." Ijcai. 2016.
- Zhou, Fan, et al. "Deeplink: A deep learning approach for user identity linkage." IEEE INFOCOM 2018-IEEE Conference on Computer Communications. IEEE, 2018.
- Heimann, Mark, et al. "Regal: Representation learning-based graph alignment." Proceedings of the 27th ACM International Conference on Information and Knowledge Management. 2018.
- Xu, Hongteng, et al. "Gromov-wasserstein learning for graph matching and node embedding." arXiv preprint arXiv:1901.06003 (2019).
- Zhang, Yutao, et al. "Cosnet: Connecting heterogeneous social networks with local and global consistency." Proceedings of the 21st ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. 2015.

References

- Chu, Xiaokai, et al. "Cross-network embedding for multi-network alignment." The World Wide Web Conference. 2019.
- Mohammadi, Shahin, et al. "Triangular alignment (TAME): A tensor-based approach for higher-order network alignment." IEEE/ACM transactions on computational biology and bioinformatics 14.6 (2016): 1446-1458.
- Zhang, Si, et al. "Multilevel network alignment." The World Wide Web Conference. 2019.
- Zhu, Hao, et al. "Iterative Entity Alignment via Joint Knowledge Embeddings." IJCAI. Vol. 17. 2017.
- Wang, Zhichun, et al. "Cross-lingual knowledge graph alignment via graph convolutional networks." Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing. 2018.
- 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.
- Zhang, Si, et al. "NetTrans: Neural Cross-Network Transformation." Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. 2020.
- Yan, Ming, et al. "Friend transfer: Cold-start friend recommendation with cross-platform transfer learning of social knowledge." 2013 IEEE International Conference on Multimedia and Expo (ICME). IEEE, 2013.
- Du, Xingbo, et al. "Cross-network Skip-gram Embedding for Joint Network Alignment and Link Prediction." IEEE Transactions on Knowledge and Data Engineering (2020).

References

- Cao, Xuezhi, and Yong Yu. "Joint user modeling across aligned heterogeneous sites." Proceedings of the 10th ACM Conference on Recommender Systems. 2016.
- Zhan, Qianyi, et al. "Influence maximization across partially aligned heterogenous social networks." Pacific-Asia Conference on Knowledge Discovery and Data Mining. Springer, Cham, 2015.
- Faisal, Fazle E., et al. "The post-genomic era of biological network alignment." EURASIP Journal on Bioinformatics and Systems Biology 2015.1 (2015): 3.
- Xu, Jiejun, et al. "GTA3 2018: Workshop on Graph Techniques for Adversarial Activity Analytics." Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining. 2018.