Towards Optimal Teams in Big Networks

Hanghang Tong

hanghang.tong@asu.edu

http://tonghanghang.org

@ the 2nd International Workshop on Machine Learning Methods for Recommender Systems (**MLRec 2016**)

Teams Are Everywhere

1. Film Crew

2. Sports Team

3. Sales Team

4. Research Team

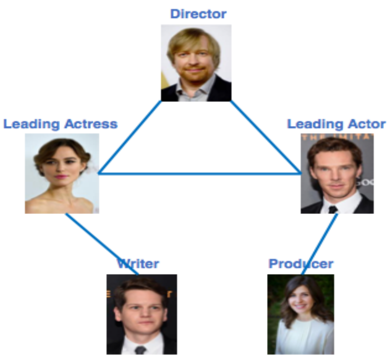
5. Military Team

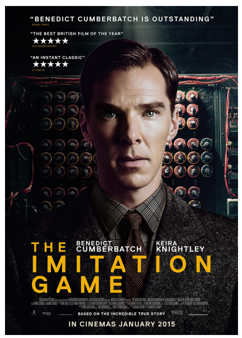
6. Development Team

- Wuchty, Stefan, Ben Jones, and Brian Uzzi. "The Increasing Dominance of Teams in the Production of Knowledge," Science, May 2007, 316:1036-1039.

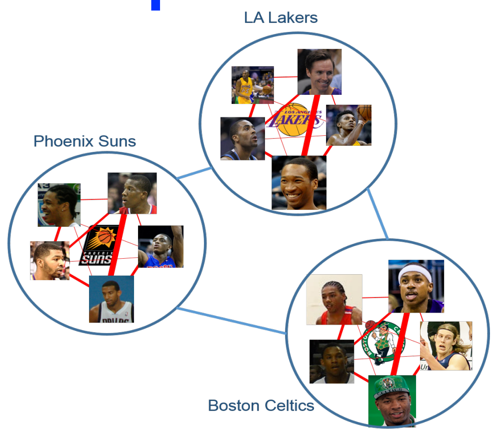
Networks Are Everywhere in Teams

1. Film Crew





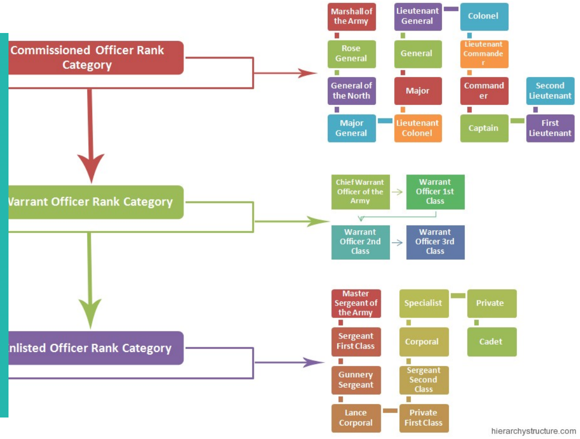
2. Sports Team



3. Sales Team

4. Research Team

5. Military Team

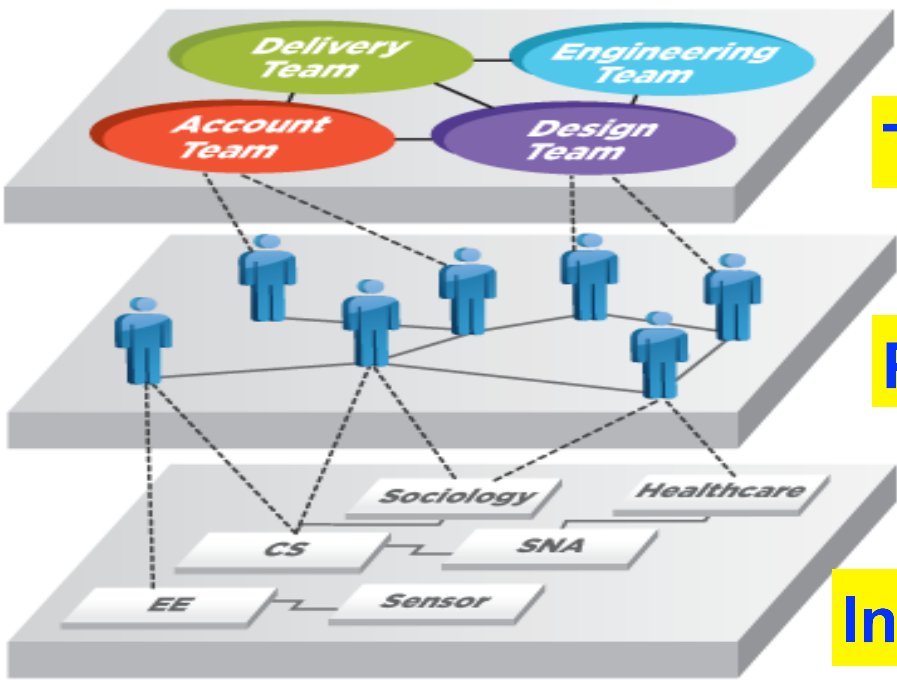


6. Development Team

- Wuchty, Stefan, Ben Jones, and Brian Uzzi. "The Increasing Dominance of Teams in the Production of Knowledge," Science, May 2007, 316:1036-1039.

Network Science of Teams

People collaborate as a team to collectively perform some complex tasks



Team Level Network

Person Level Network

Information Topic Level Network

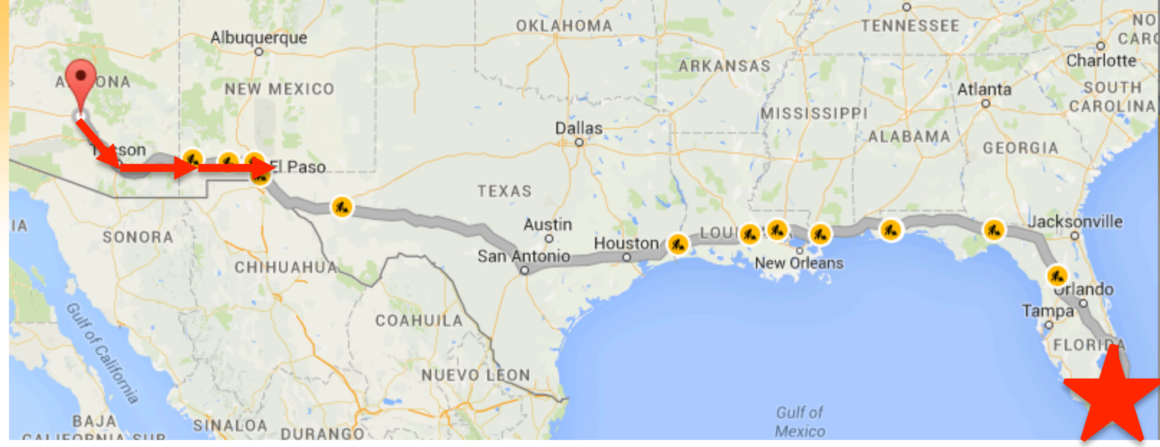
- Wuchty, Stefan, Ben Jones, and Brian Uzzi. "The Increasing Dominance of Teams in the Production of Knowledge," Science, May 2007, 316:1036-1039.

Research Questions

- Q1: What do high-performing teams share in common? [Uzzi+Science13]
- Q2: How to foresee the success at an early stage? [Wang+Science13]
- Q3: What's the optimal design for a team in the context of networks?[Lappas+KDD09, Rangapuram+WWW13]

- S. Wuchty, B. Jones, and B. Uzzi. The Increasing Dominance of Teams in the Production of Knowledge, Science, 2007
- D. Wang, C. Song, and A.-L. Barabasi. Quantifying long-term scientific impact. Science, 342(6154): 127-132, 2013.
- T. Lappas, K. Liu, and E. Terzi. Finding a team of experts in social networks. In KDD, pages 467–476, 2009.
- S. S. Rangapuram, T. Buhler, and M. Hein. Towards realistic team formation in social networks based on densest subgraphs. WWW 2013.

Roadmap



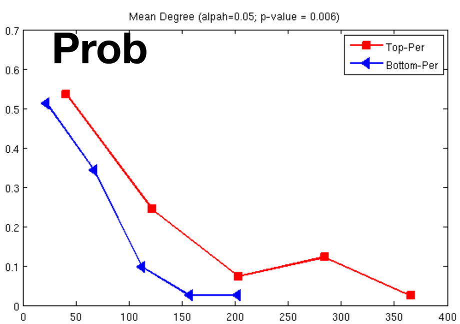
- Motivations
- Q1: Team Performance Characterization
- Q2: Team Performance Prediction
- Q3: Team Performance Optimization
- Open Challenges

Degrees, Forwarding, Tie Skewness and Sociability

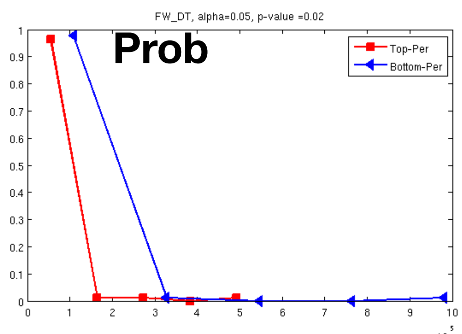
Empirical Findings:

- The average degree 
- The maximum tie skewness of team members 
- The sociability of team members 
- The information forwarding efficiency. 

Network metrics	T-Statistic	P-value
Mean degree	1.739	0.084
Min sociability	-1.907	0.058
Max sociability	-2.165	0.031
Max tie skewness	3.305	0.0001



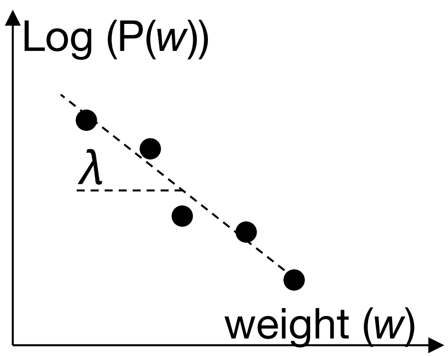
ave degree



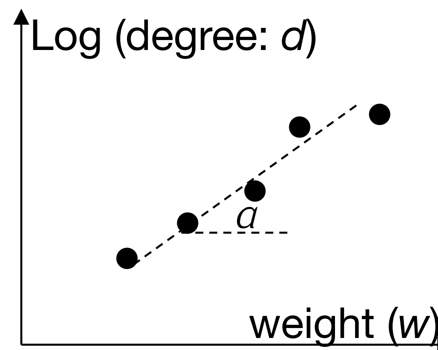
ave forwarding delay

Red: top-teams; Blue: bottom-teams

For a given person:



$P(w) \propto w^{-\lambda}$; λ: tie skewness

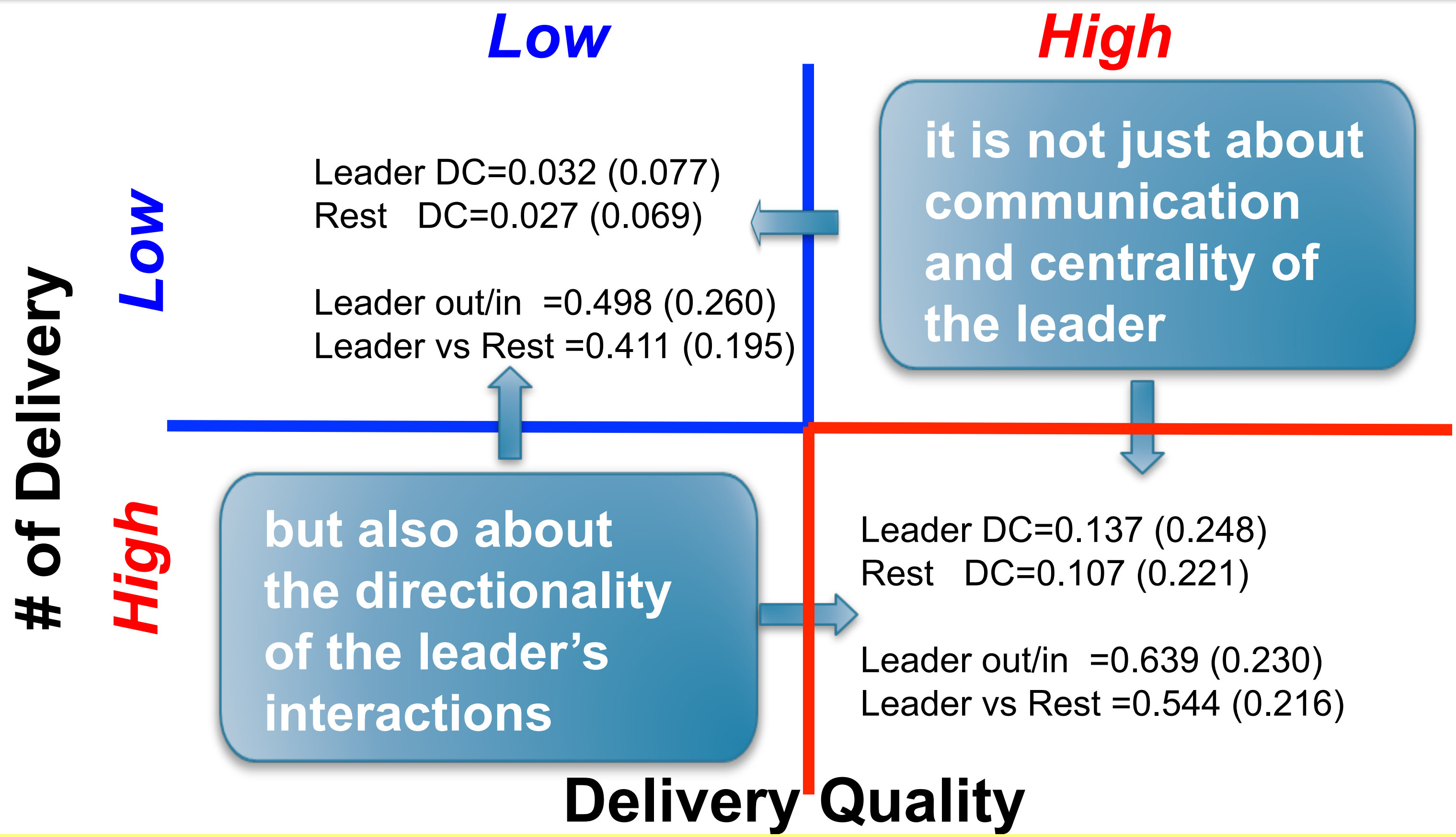


$d \propto w^{a}$; a: sociability

A **focused** team with **larger reachability** performs better

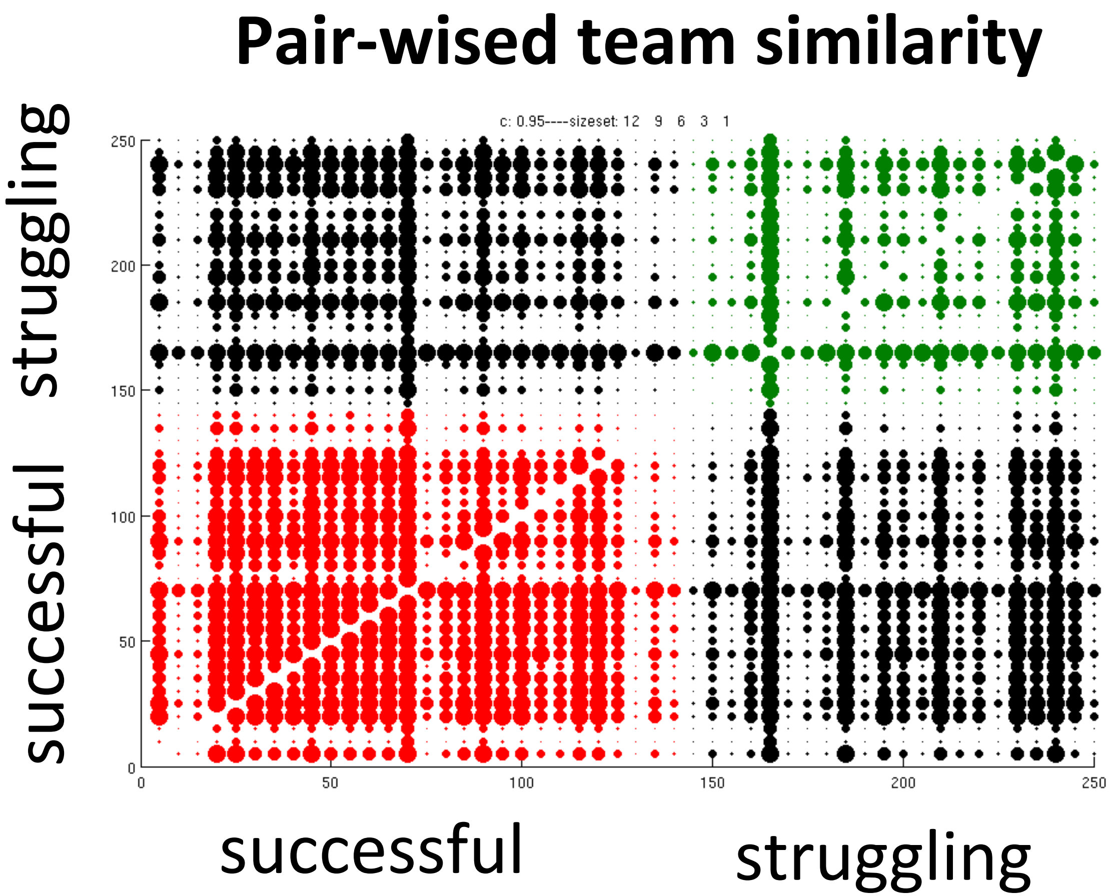
The Effect of Team Leaders

Result initially found with sales teams and replicated in 2 independent studies with software teams showing measurable effects on productivity and quality even after taking into account team-level communication structure. Accounts for > 55% variance



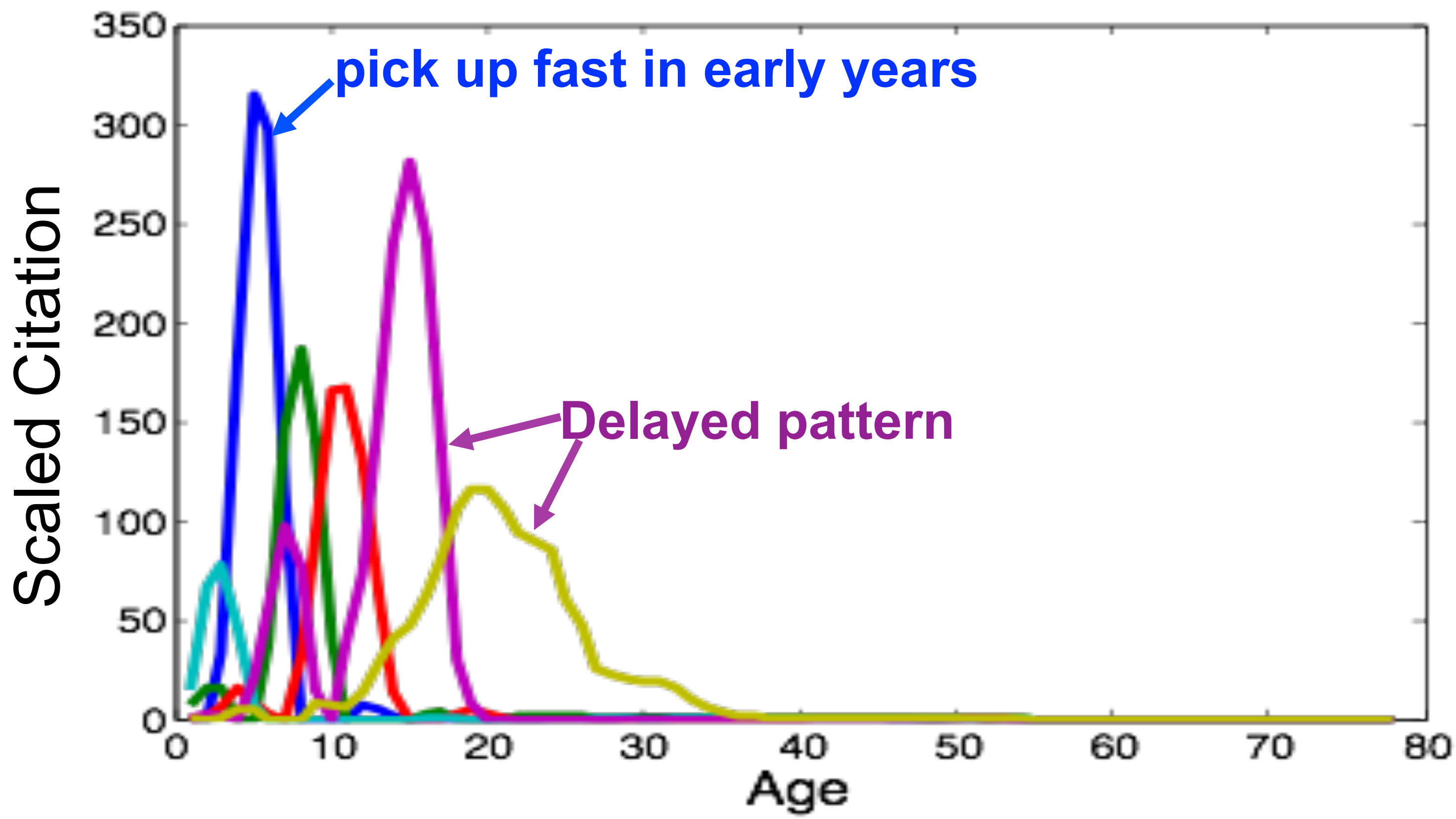
Teams perform better when (formal) leader is central in communication **out-flow** but not in-flow [Ehrlich & Tong WIDS12]

The Effect of Team Network Connectivity



"Happy families are all alike; every unhappy family is unhappy in its own way."
- Leo Tolstoy

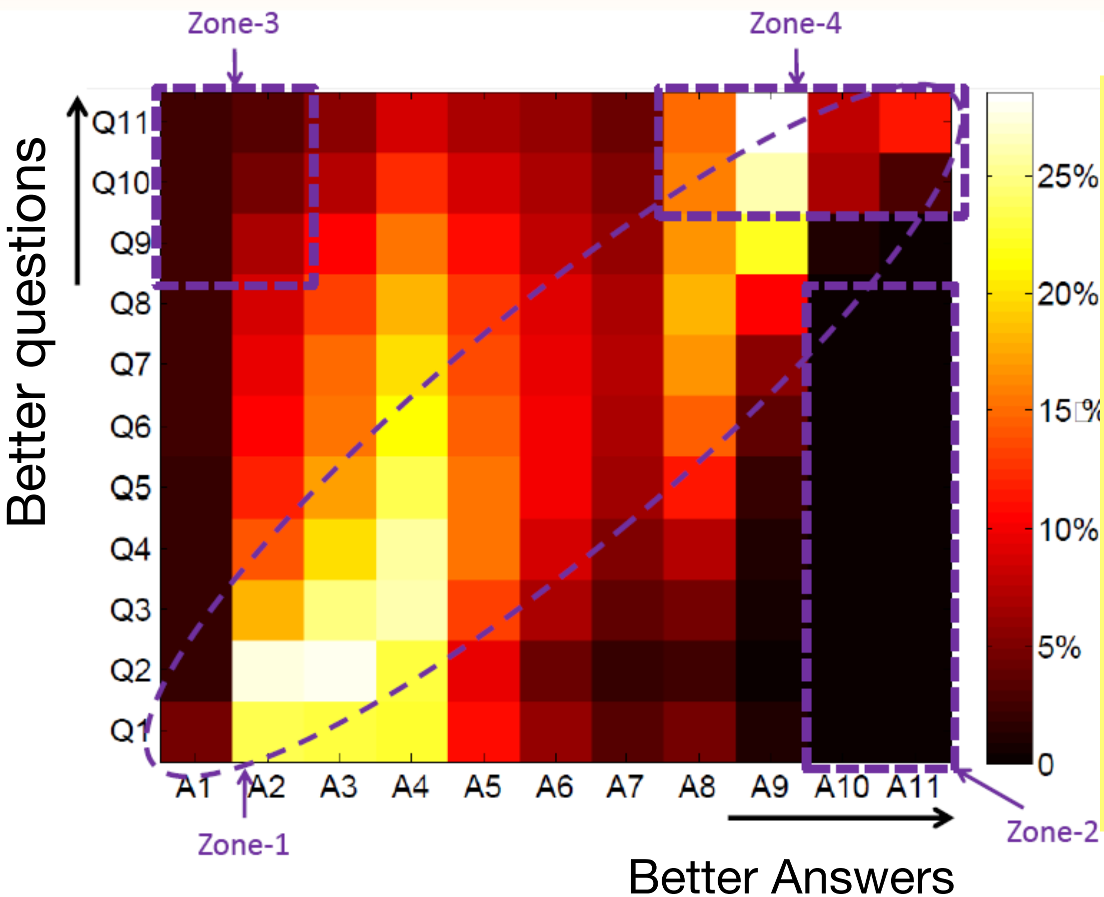
Performance Dynamics
(metric: long-term citation counts)



Impact of scientific work from different domains behaves differently

- L. Li, and H. Tong: The Child is Father of the Man: Foresee the Success at the Early Stage. KDD 2015: 655-664

Performance/Impact Coupling



Zone 1: strong positive correlation, *r=0.67, p-value < 0.0001*

Zone 2: want a good A? Ask a good Q first!

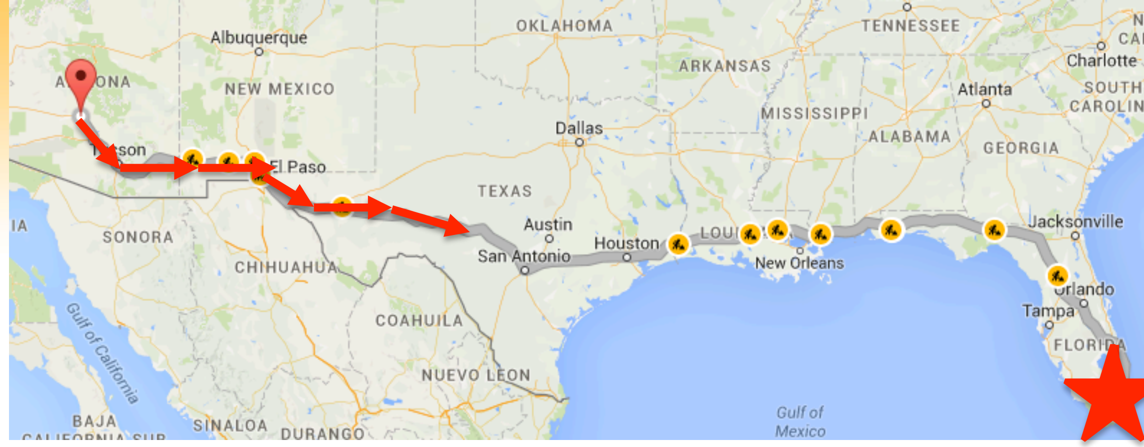
Zone 3: harder Qs

Zone 4: high Q ←→ high A

- Analysis conducted on stack overflow,
- independently verified on another CQA: math overflow

- Y. Yao, H. Tong, F. Xu, J. Lu: Predicting long-term impact of CQA posts: a comprehensive viewpoint. KDD 2014
- “Data Mining Reveals the Secret to Getting Good Answers”, MIT Technology Review, 2013

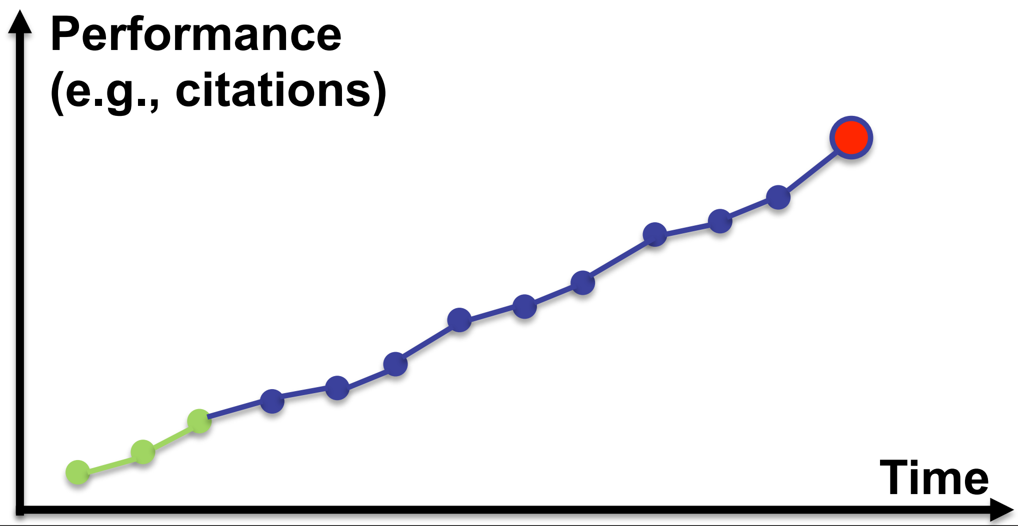
Roadmap



- Motivations
- Q1: Team Performance Characterization
- Q2: Team Performance Prediction
- Q3: Team Performance Optimization
- Open Challenges

Performance Prediction: Setup

- **Given:** Initial Performance of a team
- **Predict:**
  - (1) Long-Term Performance [KDD15]
  - (2) Performance Trajectory [SDM16]



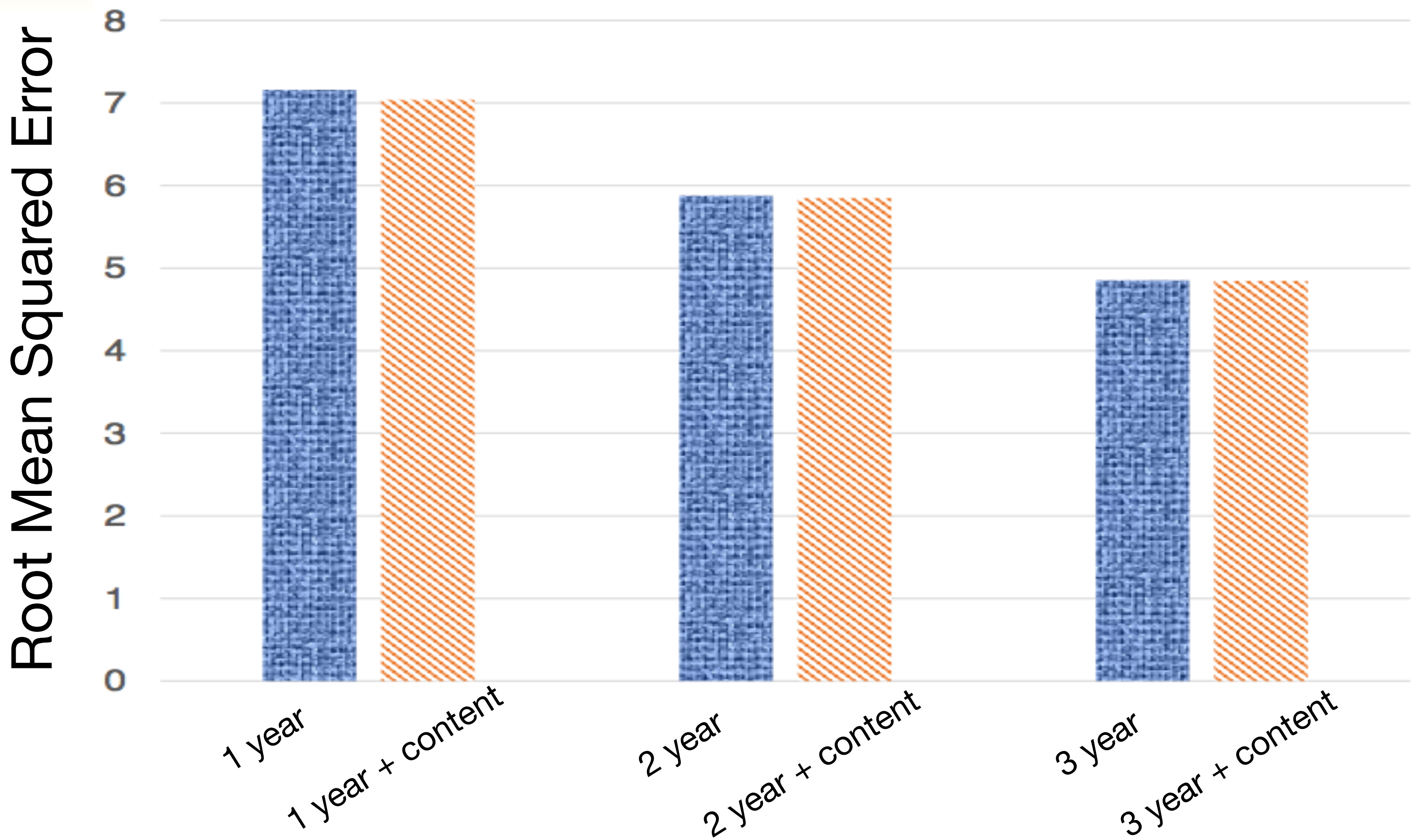
- L. Li, and H. Tong: The Child is Father of the Man: Foresee the Success at the Early Stage. KDD 2015: 655-664
- L. Li, *H. Tong*, J. Tang and W. Fan: "iPath: Forecasting the Pathway to Impact". SDM 2016

Performance Prediction: Challenges

- C1: Scholarly feature design
- C2: Non-linearity
- C3: Domain heterogeneity
- C4: Dynamics

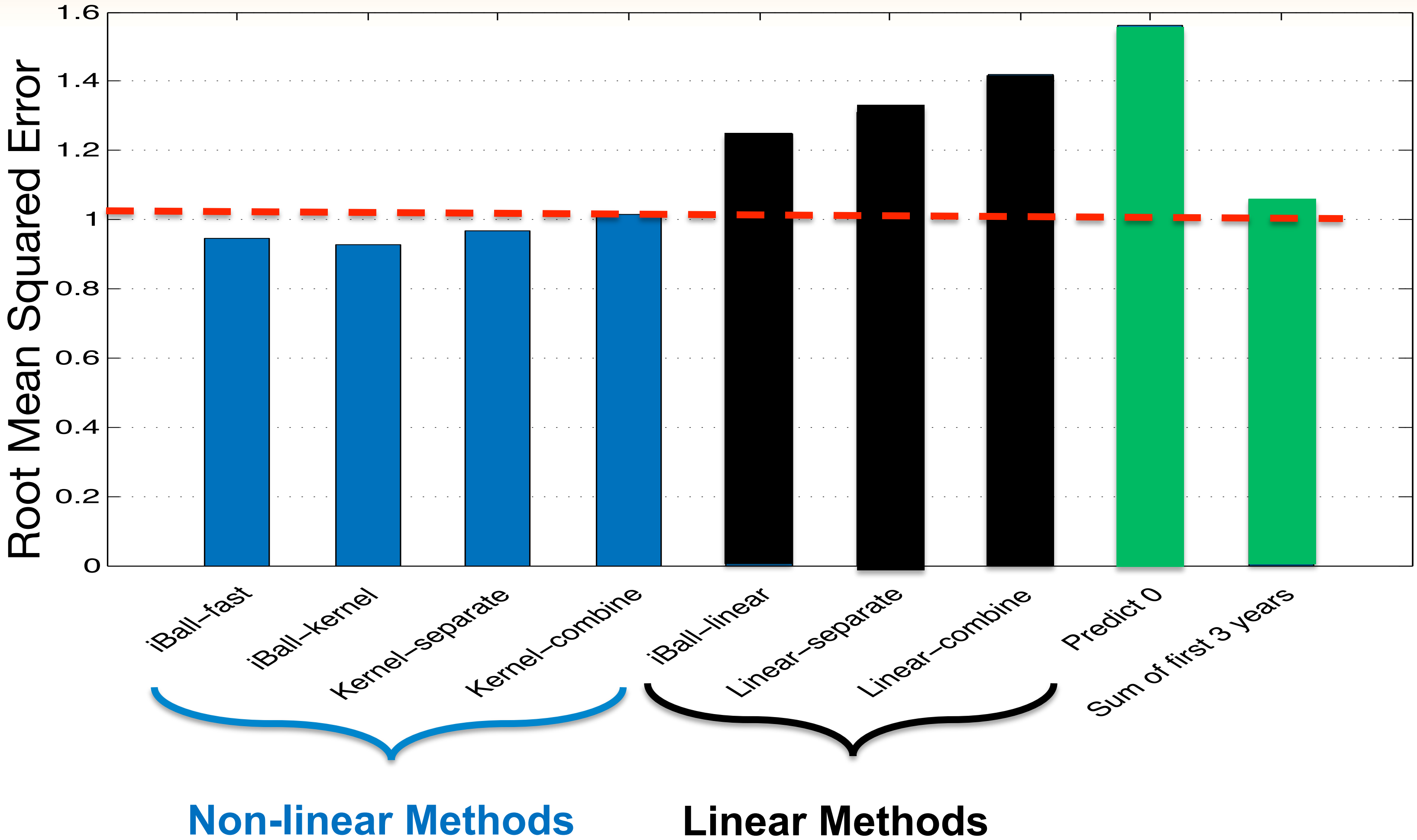
- L. Li, and H. Tong: The Child is Father of the Man: Foresee the Success at the Early Stage. KDD 2015: 655-664

C1: Scholarly Feature Design



Obs.: Adding content features brings little improvement

C2: Non-linearity

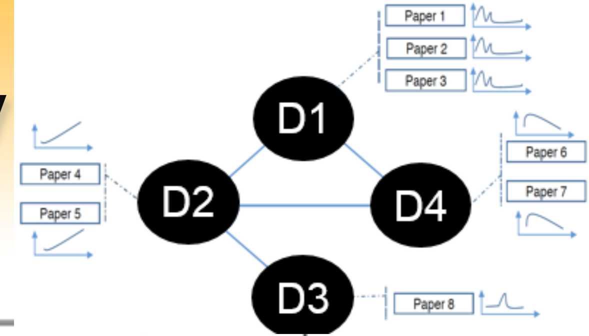


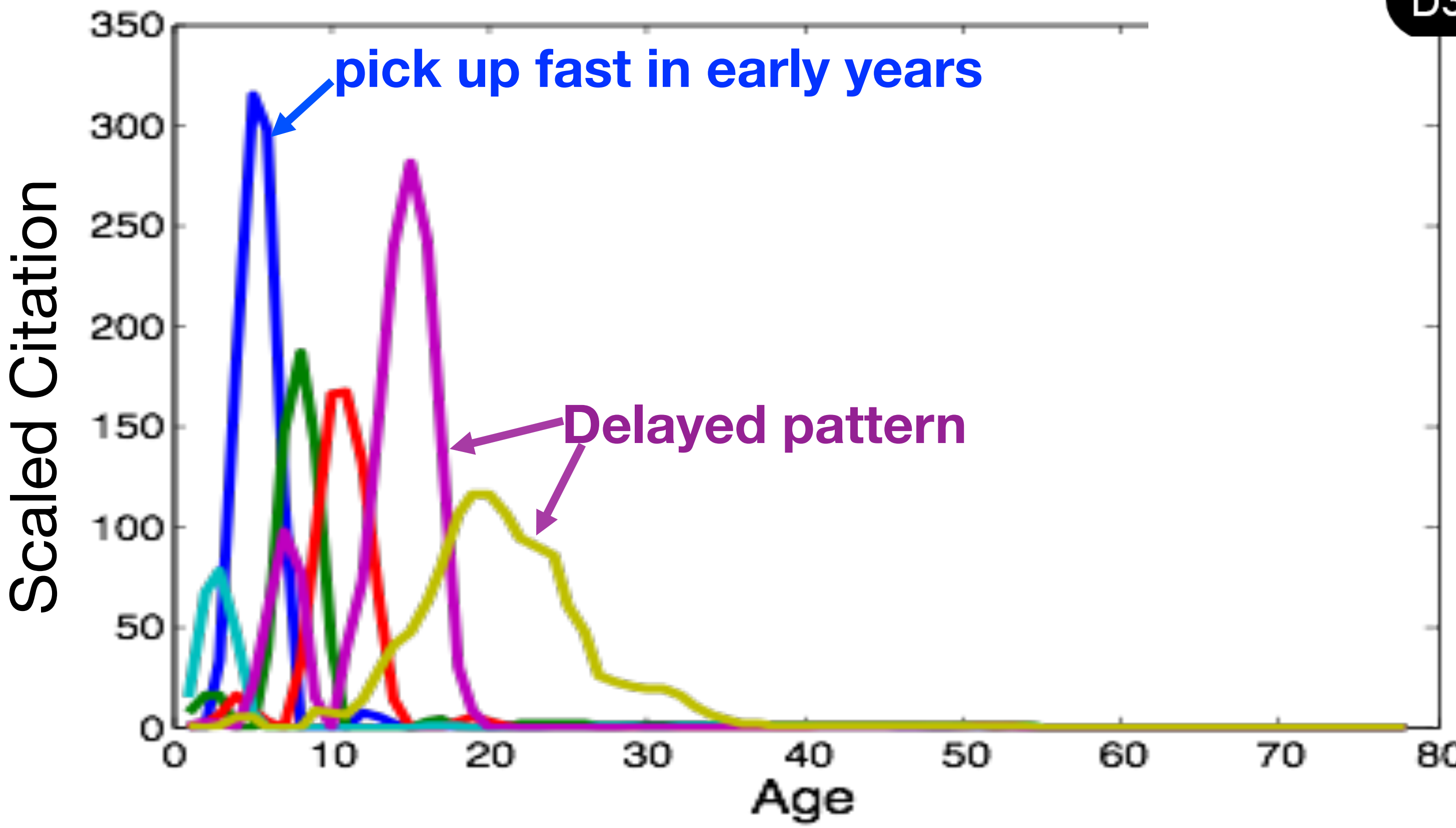
Obs.: Non-linear methods > linear ones

DATA Lab

Arizona State University

C3: Domain heterogeneity

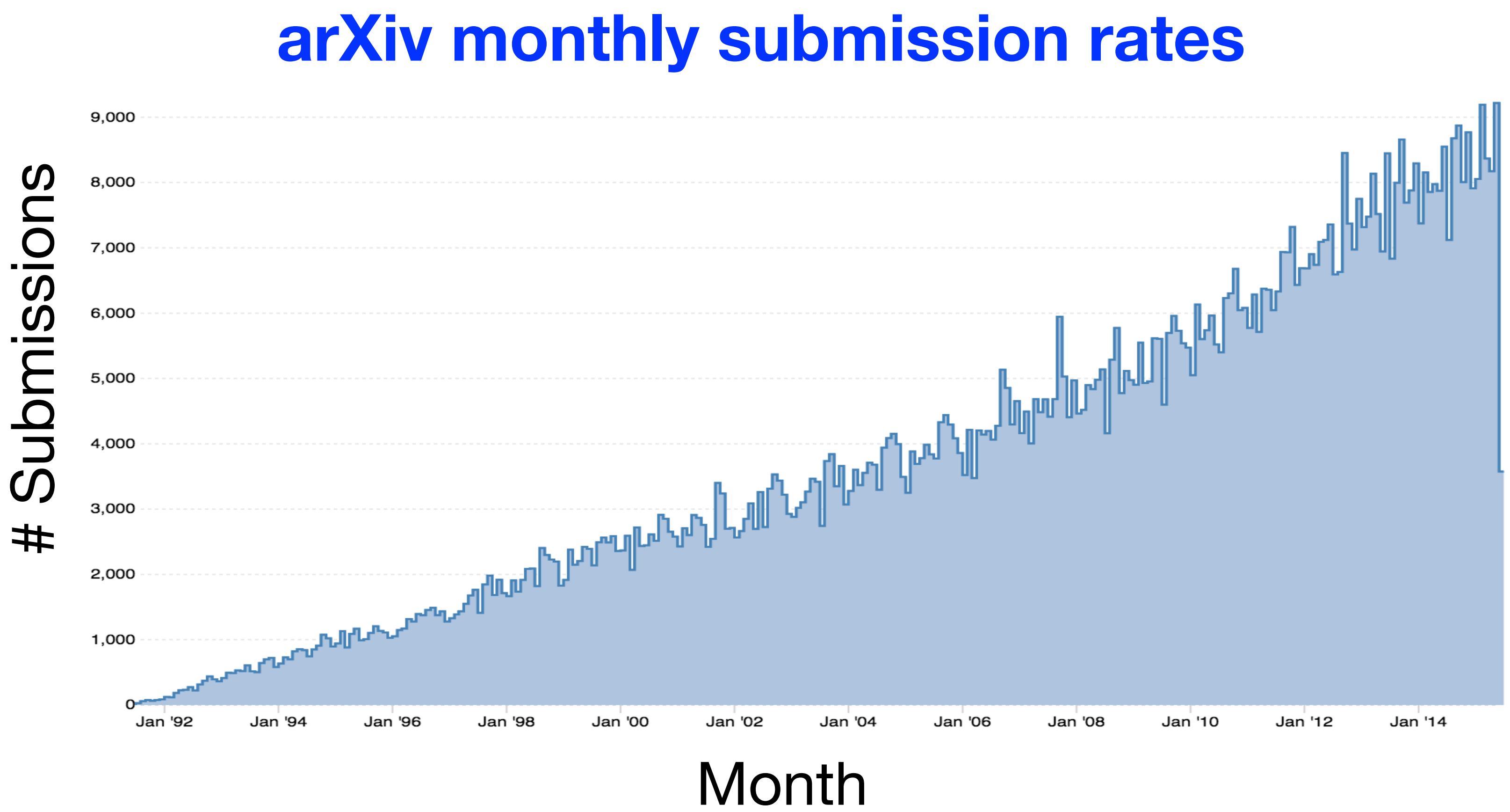




Obs.: Impact of scientific work from different domains behaves differently

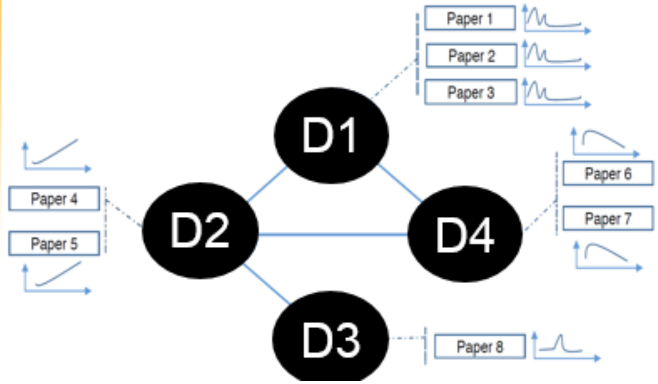
Arizona State University

C4: Dynamics



Q: How to quickly update the predictive model?

iBall — Formulations



- Optimization Formulation

$$\min_{\mathbf{w}^{(\mathrm{i})}, i=1,\ldots,n_d} \sum_{i=1}^{n_d} \overbrace{\mathcal{L}[f(\mathbf{X}^{(\mathrm{i})}, \mathbf{w}^{(\mathrm{i})}), \mathbf{Y}^{(\mathrm{i})}] + \lambda \sum_{i=1}^{n_d} \Omega(\mathbf{w}^{(\mathrm{i})})}^{\text{Within-Domain Model}}$$

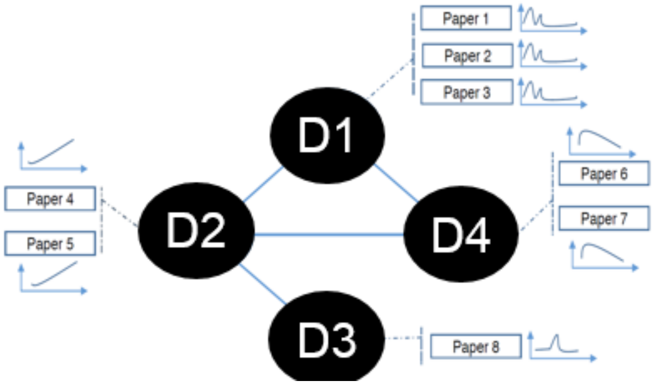
$$+\theta \sum_{i=1}^{n_d} \sum_{j=1}^{n_d} \mathbf{A}_{\mathrm{ij}} \underbrace{g(\mathbf{w}^{(\mathrm{i})}, \mathbf{w}^{(\mathrm{j})})}_{\text{Cross-Domain Consistency}}$$

- Remarks
  - **Within-Domain Model**: regression/classification, linear/non-linear
  - **Cross-Domain Consistency**: similar domains have similar models

Question: how to instantiate such consistency?

- L. Li, and H. Tong: The Child is Father of the Man: Foresee the Success at the Early Stage. KDD 2015: 655-664

iBall — linear formulation



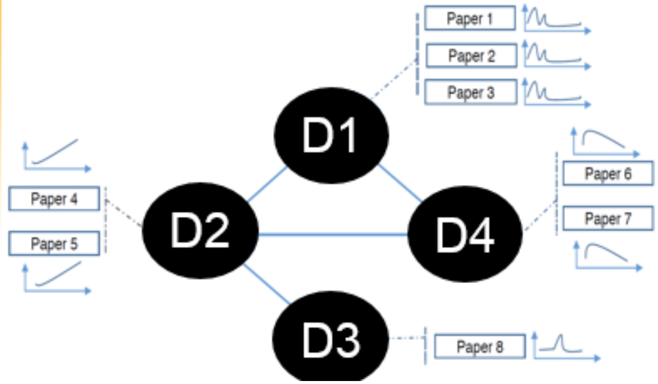
Details:

$$\min_{\mathbf{w}^{(\mathrm{i})}, i=1,\ldots,n_d} \sum_{i=1}^{n_d} \|\mathbf{X}^{(\mathrm{i})}\mathbf{w}^{(\mathrm{i})} - \mathbf{Y}^{(\mathrm{i})}\|_2^2 + \lambda \sum_{i=1}^{n_d} \|\mathbf{w}^{(\mathrm{i})}\|_2^2 + \theta \sum_{i=1}^{n_d} \sum_{j=1}^{n_d} \underbrace{\mathbf{A}_{\mathrm{ij}} \|\mathbf{w}^{(\mathrm{i})} - \mathbf{w}^{(\mathrm{j})}\|_2^2}$$

Intuitions: similar domains (large $\mathbf{A}_{ij}$)

→ same feature has similar effect (small $\|\mathbf{w}^{(\mathrm{i})} - \mathbf{w}^{(\mathrm{j})}\|_2^2$)

iBall — non-linear formulation



Details:

$$\min_{\mathbf{w}^{(\mathbf{i})}, i=1,\ldots,n_d} \sum_{i=1}^{n_d} \|\mathbf{K}^{(\mathbf{i})}\mathbf{w}^{(\mathbf{i})} - \mathbf{Y}^{(\mathbf{i})}\|_2^2 + \lambda \sum_{i=1}^{n_d} \mathbf{w}^{(\mathbf{i})'}\mathbf{K}^{(\mathbf{i})}\mathbf{w}^{(\mathbf{i})}$$
$$+\theta \sum_{i=1}^{n_d}\sum_{j=1}^{n_d} \mathbf{A_{ij}} \|\mathbf{K}^{(\mathbf{i})}\mathbf{w}^{(\mathbf{i})} - \mathbf{K}^{(\mathbf{ij})}\mathbf{w}^{(\mathbf{j})}\|_2^2$$

Predicted output (domain *i* → domain *i*)

Predicted output (domain *j* → domain *i*)

Intuitions: similar domains (large $\mathbf{A}_{ij}$)

→ similar predicted outputs (small $\|\mathbf{K}^{(\mathbf{i})}\mathbf{w}^{(\mathbf{i})} - \mathbf{K}^{(\mathbf{ij})}\mathbf{w}^{(\mathbf{j})}\|_2^2$)

iBall — Closed-form Solutions

- Closed-form Solution

$$\mathbf{w} = \mathbf{S}^{-1}\mathbf{Y}$$

  - iBall — linear:

$$\mathbf{w} = [\mathbf{w}^{(1)}; \ldots; \mathbf{w}^{(k)}] \quad \mathbf{Y} = [\mathbf{X}^{(1)\prime}\mathbf{Y}^{(1)}; \ldots; \mathbf{X}^{(k)\prime}\mathbf{Y}^{(k)}]$$

$$\mathbf{S} = \begin{array}{c} \\ \end{array}\begin{array}{cc} \begin{array}{ccc} & \text{i-th block column} & \text{j-th block column} \end{array} & \\ \begin{bmatrix} \cdots & \cdots & \cdots \\ \cdots & \mathbf{X}^{(\mathbf{i})\prime}\mathbf{X}^{(\mathbf{i})} + (\theta \sum_{j=1}^{k} \mathbf{A}_{\mathbf{ij}} + \lambda)\mathbf{I} & -\theta\mathbf{A}_{\mathbf{ij}}\mathbf{I} \\ \cdots & \cdots & \cdots \end{bmatrix} & \begin{array}{l} \text{i-th block} \\ \text{row} \end{array} \end{array}$$

Time Complexity: *$O(dk)^3$*

d: # of features; k: # of domains
(*dk:* in the order of 10 or 100)

iBall — Closed-form Solutions

- Closed-form Solution

$$\mathbf{w} = \mathbf{S}^{-1}\mathbf{Y}$$

  - iBall — non-linear:

$$\mathbf{w} = [\mathbf{w}^{(1)}; \ldots; \mathbf{w}^{(k)}] \qquad \mathbf{Y} = [\mathbf{Y}^{(1)}; \ldots; \mathbf{Y}^{(k)}]$$

$$\mathbf{S} = \begin{array}{c c} \begin{array}{ccc} & \text{i-th block column} & \text{j-th block column} \end{array} & \\ \begin{bmatrix} \cdots & \cdots & \cdots \\ \cdots & (1+\theta\sum_{j=1}^{k}\mathbf{A_{ij}})\mathbf{K}^{(\mathbf{i})} + \lambda\mathbf{I} & -\theta\mathbf{A_{ij}}\mathbf{K}^{(\mathbf{ij})} \\ \cdots & \cdots & \cdots \end{bmatrix} & \begin{array}{c} \\ \text{i-th block row} \\ \\ \end{array} \end{array}$$

Time Complexity: $O(n^3)$

n: total # of training examples
(in the order of millions)

iBall — Scale-up with Dynamic Update

- **Key idea #1**: Approx ***S*** by low-rank approx
- **Details:**

$$\mathbf{S}_{t+1} \approx \mathbf{U}_{t+1}\mathbf{\Lambda}_{t+1}\mathbf{U}'_{t+1} \rightarrow \mathbf{w}_{t+1} = \mathbf{S}_{t+1}^{-1}\mathbf{Y}_{t+1} = \mathbf{U}_{t+1}\mathbf{\Lambda}_{t+1}^{-1}\mathbf{U}'_{t+1}\mathbf{Y}_{t+1}$$

(Overall: $O(n^2 r)$) (Overall: $O(nr)$)

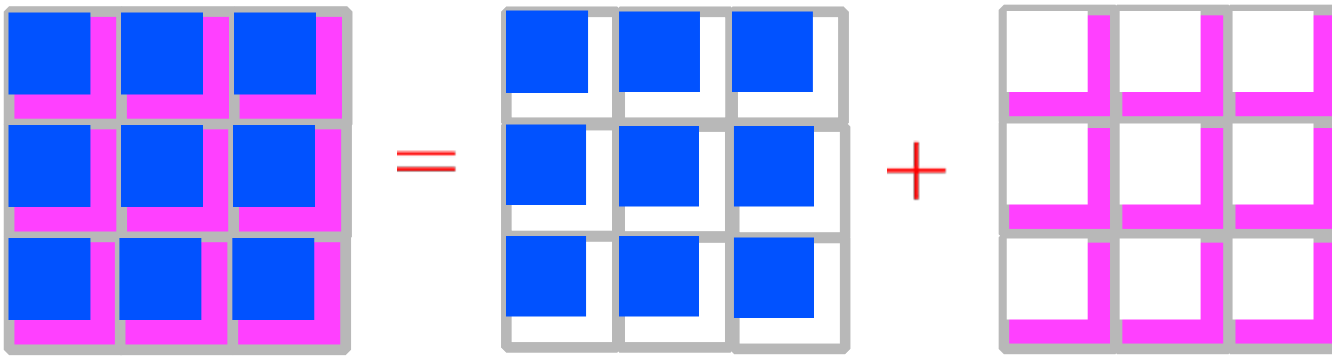
- **Complexity:** $O(n^3) \rightarrow O(n^2 r + nr)$
- **Benefit:** avoid matrix inverse

Question: how to avoid re-computing low-rank approx at each time step?

iBall — Scale-up with Dynamic Update

- **Key idea #2:** Incrementally update the low rank structure of **S**
- **Details:**

white: zeros
blue: old at *t*
pink: new at *t+1*



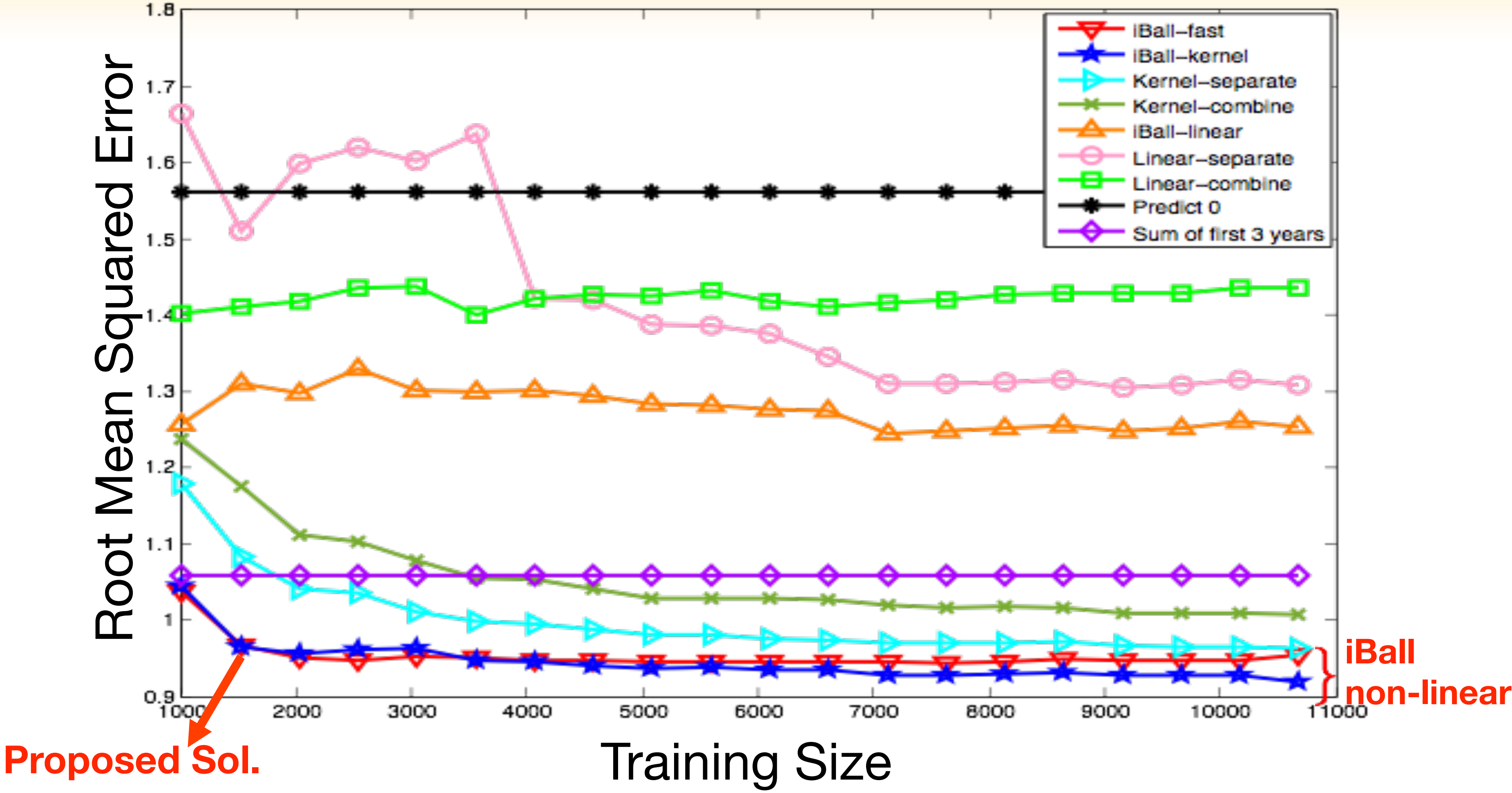
$\mathbf{S}_{t+1}$ $\tilde{\mathbf{S}}_t$ $\Delta\mathbf{S}$

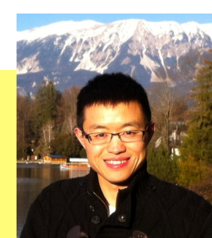
(low rank, sparse)

- **Complexity:** $O(n^2 r) \rightarrow O((n+m)(r^2 + r'^2)), r \ll n$
- **Benefit:** avoid re-computing low-rank approx

• L. Li, H. Tong, Y. Xiao, W. Fan. Cheetah: Fast Graph Kernel Tracking on Dynamic Graphs. SDM 2015.

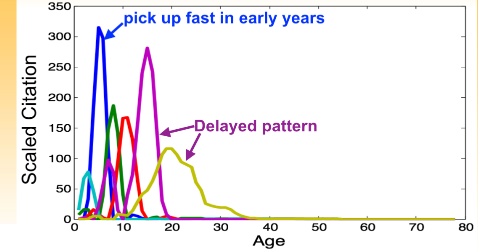
Paper Citation Prediction Performance

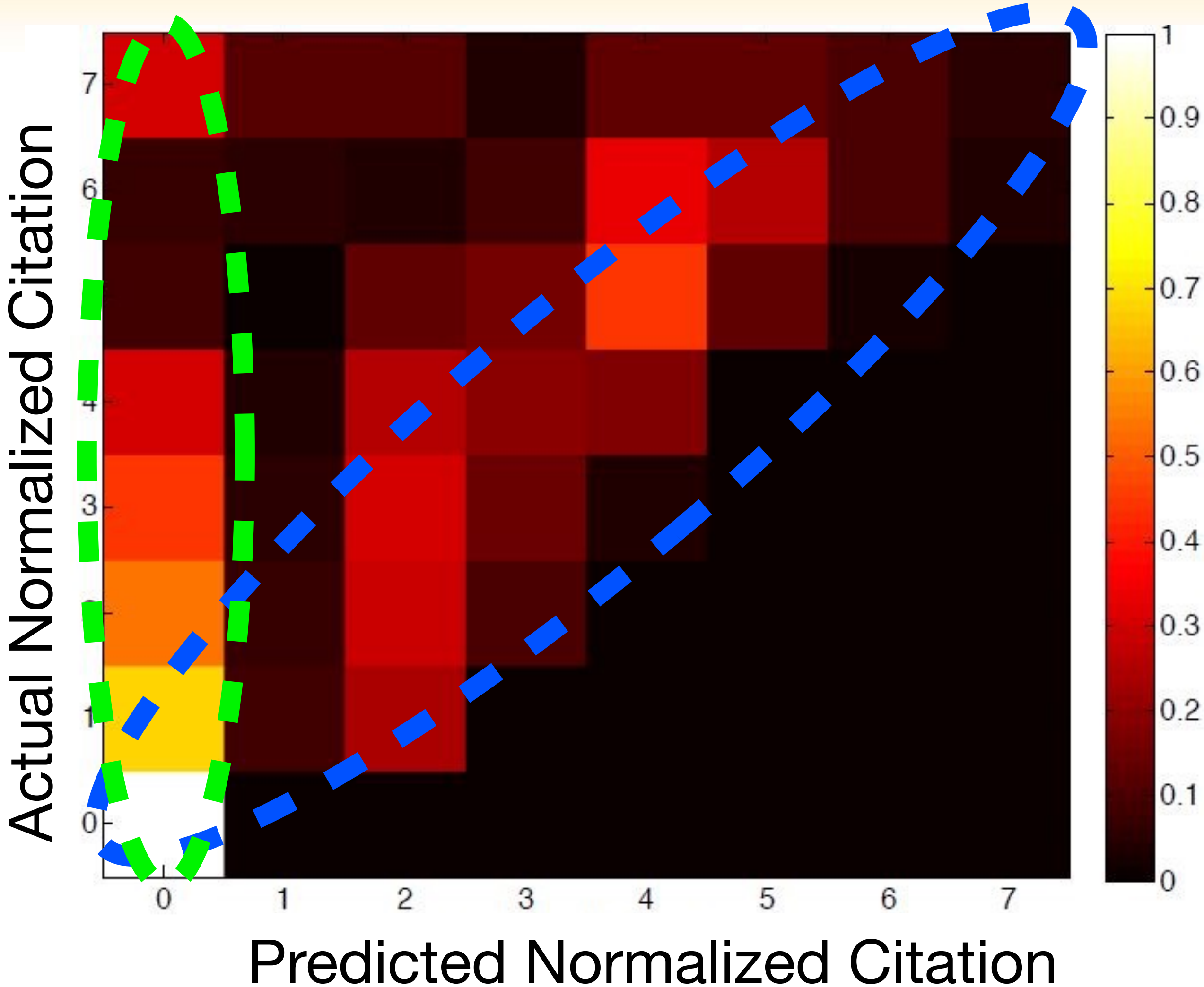




Datasets: AMiner (2,243,976 papers, 1,274,360 authors, 8,882 venues)

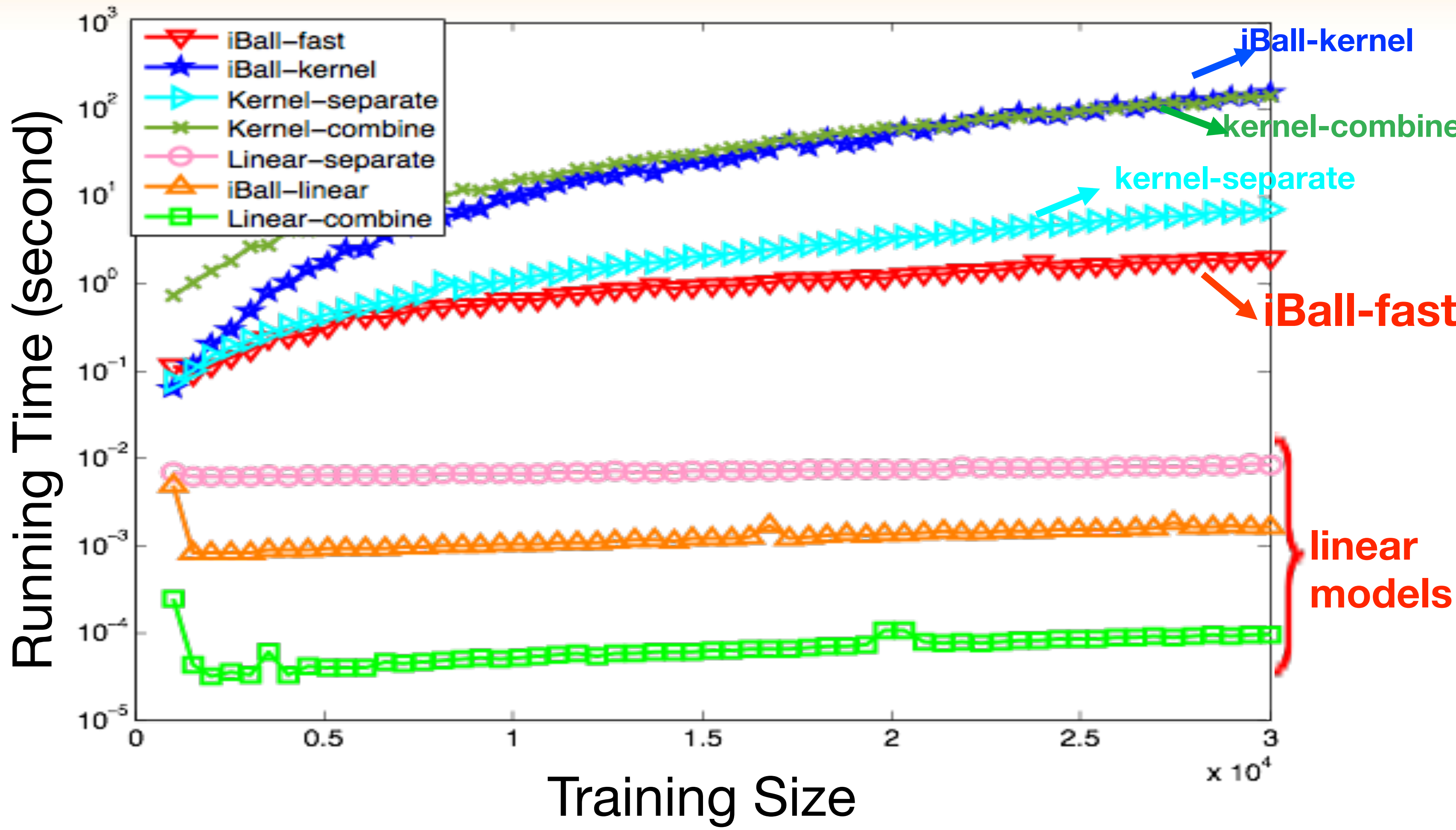
Error Analysis





Obs.: bright region at x = y

Running Time Comparison

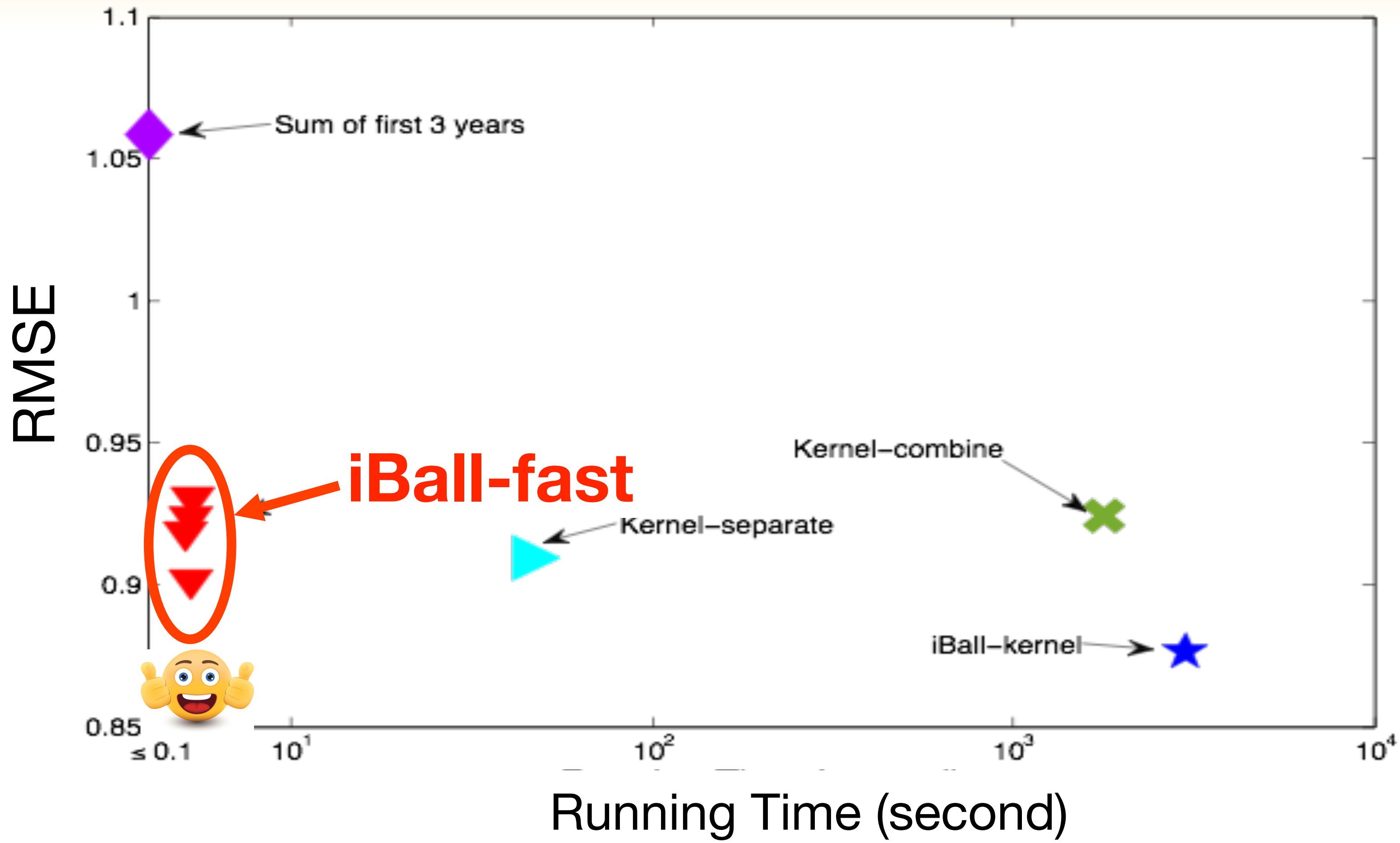


Obs.: iBall-fast outperforms other non-linear methods

DATA Lab

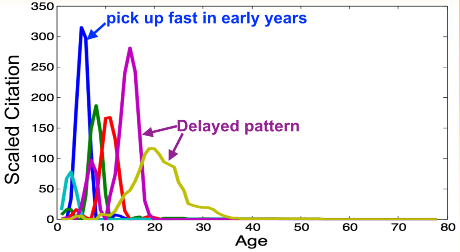
Arizona State University

Quality vs. Speed



Obs.: iBall-fast: good trade-off between quality and speed

iBall: Summary



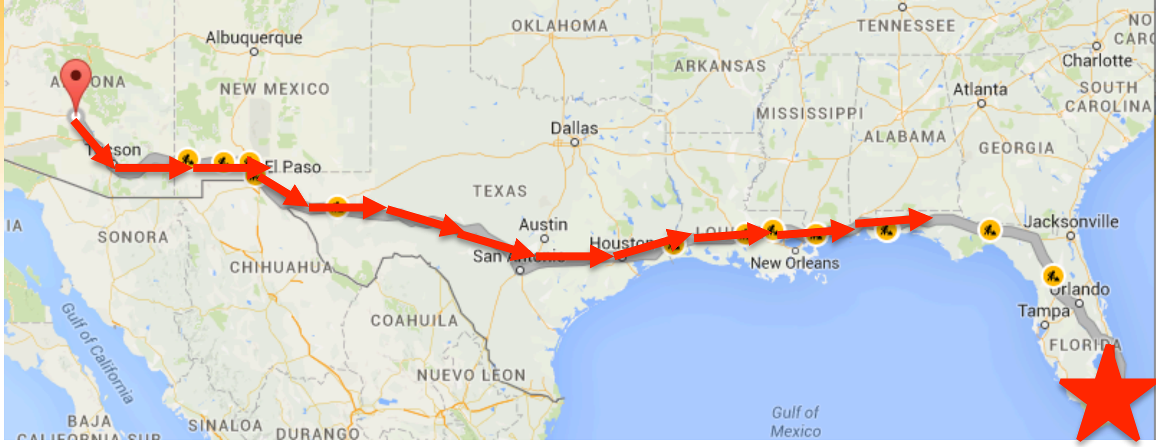
- **Goal**: predict long-term impact of scholarly entities
- **Solutions**: joint predictive model (**iBall**)

Challenges	C1 ***feature design***	C2 ***non-linearity***	C3 ***domain-heterogeneity***	C4 ***dynamics***
Tactics	***first 3 years' citation***	***kernel trick***	***domain consistency***	***low-rank approximation***

- **Results:**
  - iBall joint models better than separate versions
  - iBall-fast updates efficiently and accurately

- L. Li, and H. Tong: The Child is Father of the Man: Foresee the Success at the Early Stage. KDD 2015: 655-664

Roadmap



- Motivations
- Q1: Team Performance Characterization
- Q2: Team Performance Prediction
- Q3: Team Performance Optimization
  - Team Replacement
  - Team Enhancement
- Open Challenges

Churn of A Team Member

- **Case 1:** Employee resigns in a sales team
- **Case 2:** Task force down in a SWAT team
- **Case 3:** Rotation tactic between benches in NBA team

Q: How to find the best alternative when a team member leaves?

- L. Li, H. Tong, N. Cao, K. Ehrlich, Y.-R. Lin and N. Buchler: Replacing the Irreplaceable: Fast Algorithms for Team Member Recommendation, WWW 2015
- N. Cao, Y.-R. Lin, L. Li, H. Tong: g-Miner: Interactive Visual Group Mining on Multivariate Graphs, ACM CHI 2015
- System prototype & video demo: http://team-net-work.org

Team Member Replacement

Problem Definition:

Given: (1) A labelled social network $G := \{A, L\}$

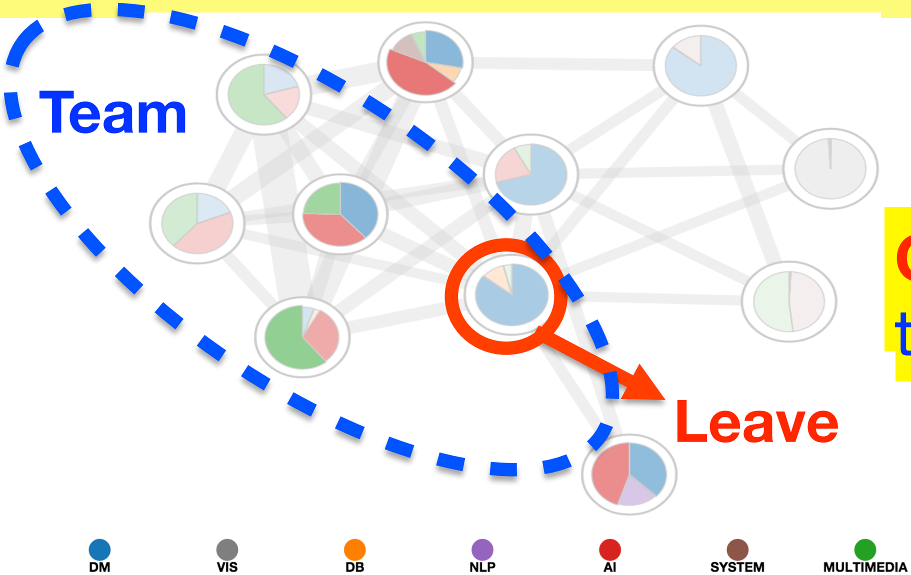
Adj. Matrix

Skill Indicator

(2) A team $G(\mathcal{T})$

(3) A team member $p \in \mathcal{T}$

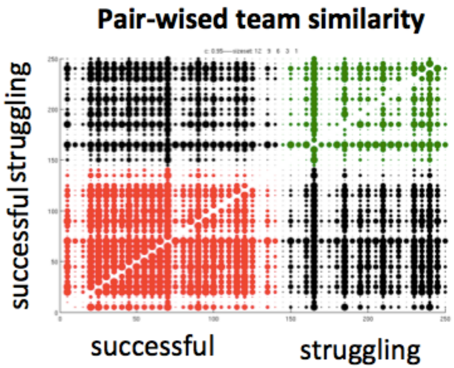
Recommend: A "best" alternative $q \notin \mathcal{T}$ to replace the person *p's* role in the team $G(\mathcal{T})$



Q: who is a good candidate to replace the person to leave

DATA Lab

Social Science Literature

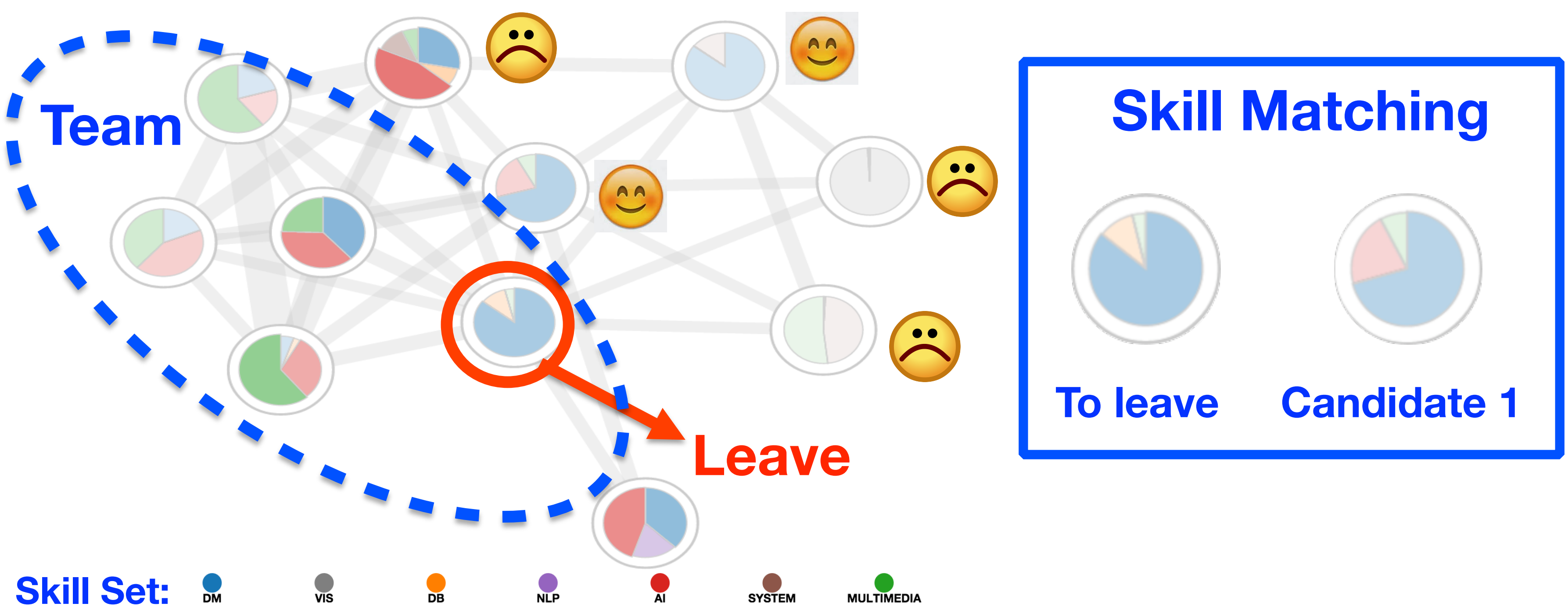


- Team members prefer to work with people they have worked before [Hinds+OBHDP00]
- Distributed teams perform better when members know each other [Cummings+CSCW08]
- Specific communication patterns amongst team members are critical for performance [Cataldo+CHI12]

Conjecture: The similarity should be measured in the **context of the team itself**

Design Objectives

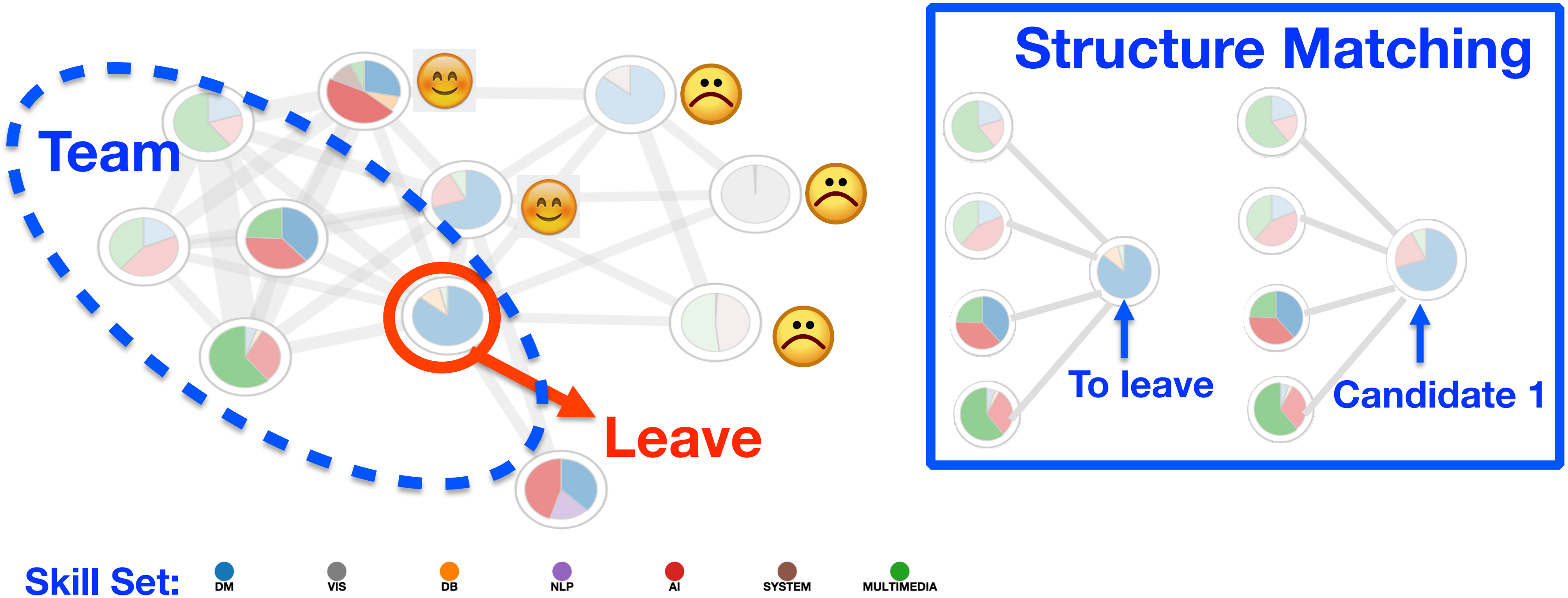
Objective 1: A good candidate should have a similar skill set



New team would have a similar skill set as the old team to continue to complete the task

Design Objectives

Objective 2: A good candidate should have a similar network structure



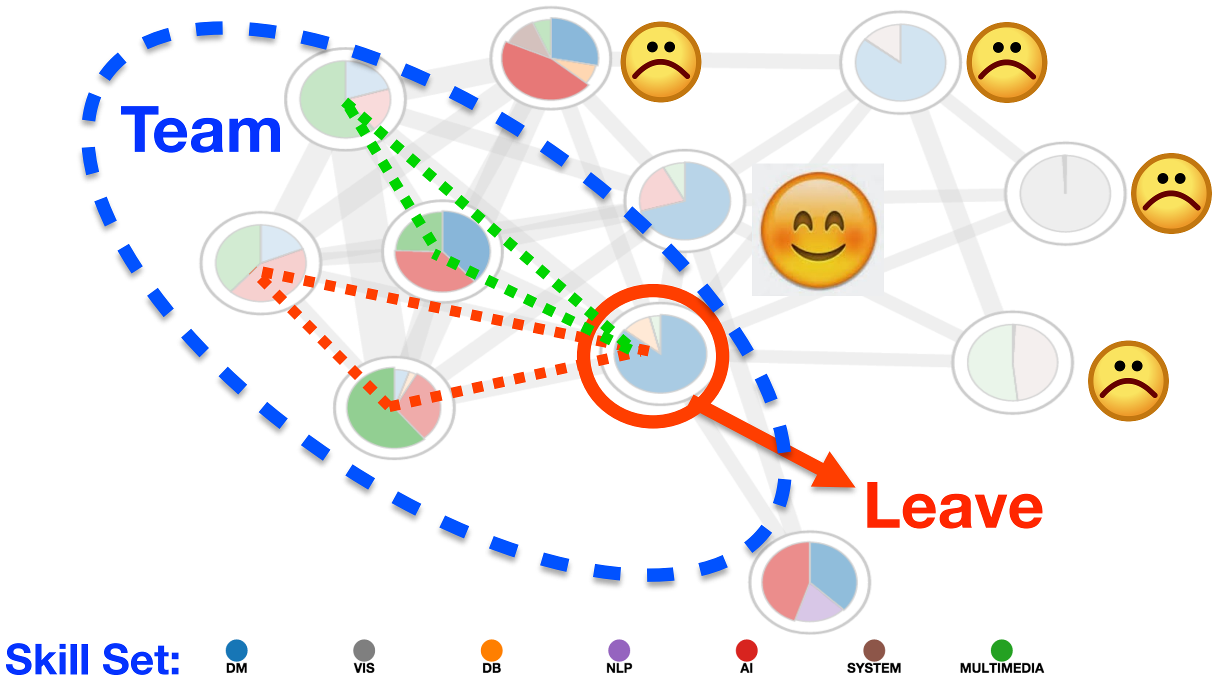
New team would have a similar network structure as the old team to collaborate effectively

DATA Lab

Arizona State University

Design Objectives

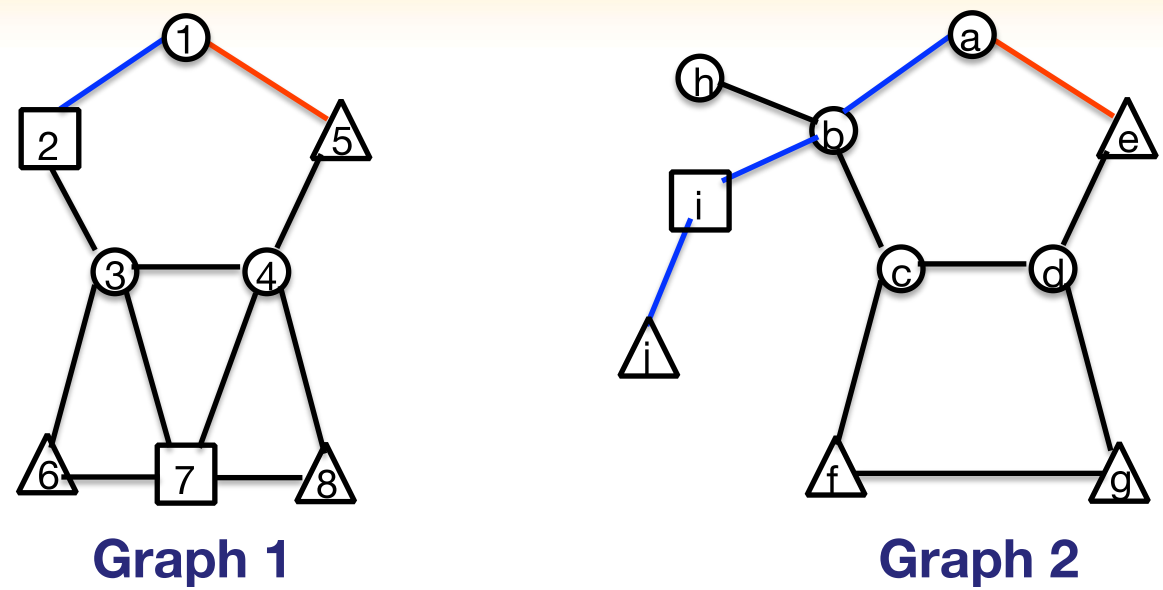
The skill and structure match should be fulfilled simultaneously!



New team would have similar skill and communication configuration for each sub-task

- L. Li, H. Tong, N. Cao, K. Ehrlich, Y.-R. Lin and N. Buchler: Replacing the Irreplaceable: Fast Algorithms for Team Member Recommendation, WWW 2015

Random Walk based Graph Kernel



Details:

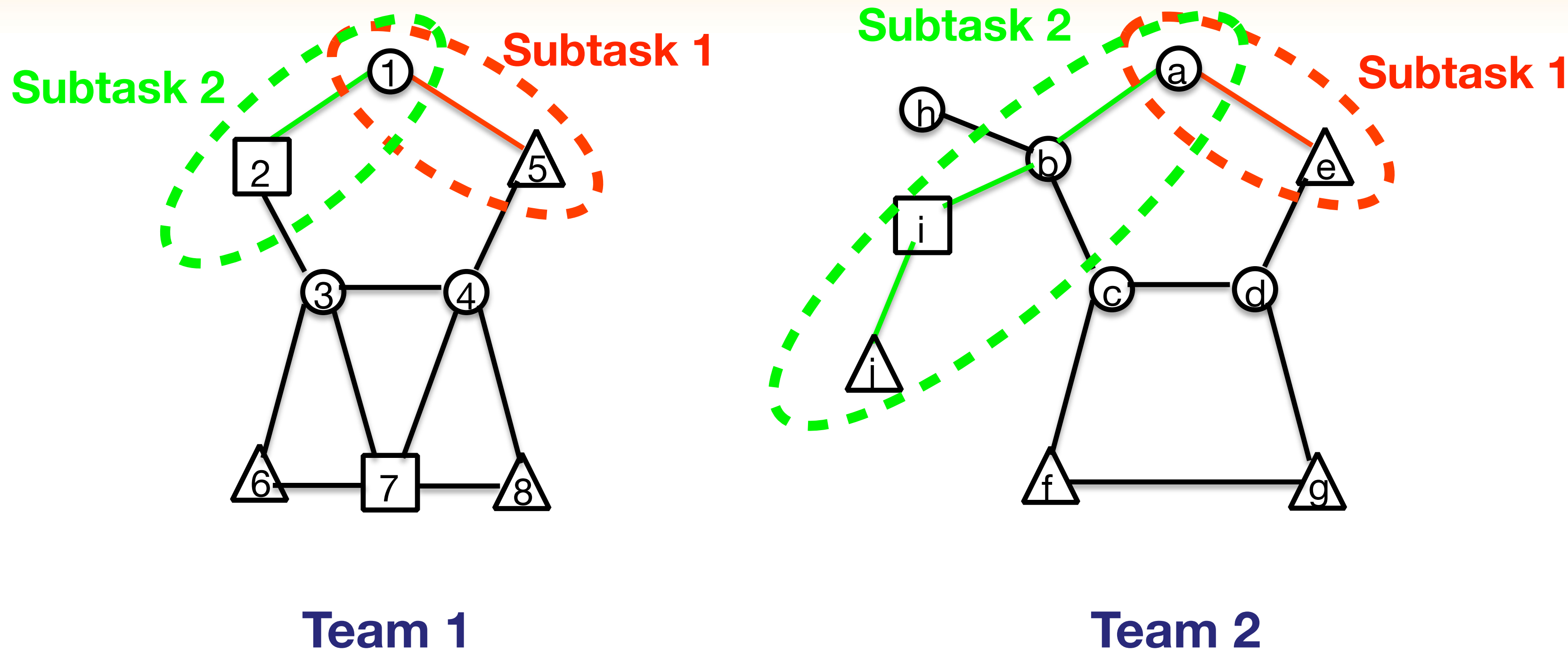
1. Compare similarity of every pair of nodes from each graph

— Eg: (1,2) vs (a, j) $\longrightarrow$ less similar

(1,5) vs (a,e) $\longrightarrow$ more similar

2. Node pair similarity is measured by random walks
3. Two graphs are similar if they share many similar node pairs

Random Walk based Graph Kernel

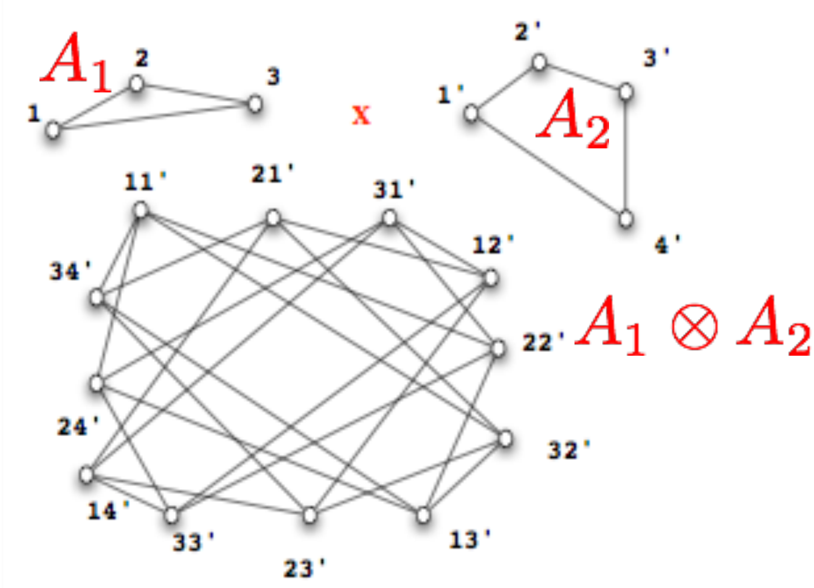


Remarks:

- Incorporates both **attributes** and **structures** similarity
- Ideal fit for our two design objectives **simultaneously**

Kronecker Product Graph w/o Attribute

Graph Illustration



Matrix Description

$$A_1 = \begin{bmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{bmatrix} \qquad A_2 = \begin{bmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{bmatrix}$$

$$A_1 \otimes A_2 = \begin{bmatrix} 0&0&0&0&0&1&0&1&0&1&0&1 \\ 0&0&0&0&1&0&1&0&1&0&1&0 \\ 0&0&0&0&0&1&0&1&0&1&0&1 \\ 0&0&0&0&1&0&1&0&1&0&1&0 \\ 0&1&0&1&0&0&0&0&0&1&0&1 \\ 1&0&1&0&0&0&0&0&1&0&1&0 \\ 0&1&0&1&0&0&0&0&0&1&0&1 \\ 1&0&1&0&0&0&0&0&1&0&1&0 \\ 0&1&0&1&0&1&0&1&0&0&0&0 \\ 1&0&1&0&1&0&1&0&0&0&0&0 \\ 0&1&0&1&0&1&0&1&0&0&0&0 \\ 1&0&1&0&1&0&1&0&0&0&0&0 \end{bmatrix}$$

Kronecker product

One Random Walk on A_1

$+$

One Random Walk on A_2

$=$ One Random Walk on $A_1 \otimes A_2 = A_\times$

- S. V. N. Vishwanathan, Nicol N. Schraudolph, Imre Risi Kondor, and Karsten M. Borgwardt. Graph Kernels. Journal of Machine Learning Research, 11:1201–1242, April 2010.

RW Graph Kernel — Formulation

Taking expectations instead of summing

$$\begin{aligned}\mathrm{Ker}(G_1, G_2) &= \sum_k c^k q'_\times (L_\times A_\times)^k L_\times p_\times \\ &= q'_\times (I - c\underset{\uparrow}{L_\times} A_\times)^{-1} L_\times p_\times\end{aligned}$$

Attribute Indicator

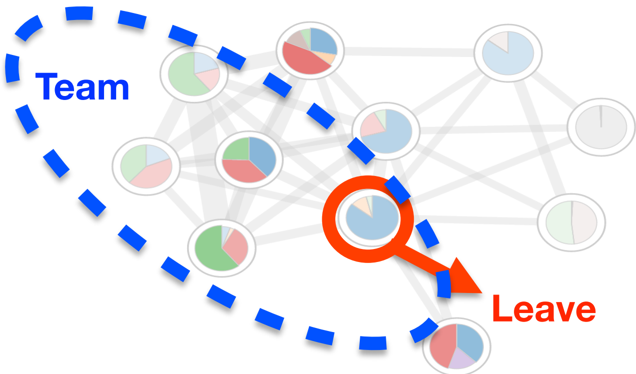
- **Computational cost** (A_x: t^2 x t^2)
  - *Exact methods:* [Vishwanathan+JMLR2010]
    - $O(t^6)$ - Direct computation
    - $O(t^3)$ - Sylvester equation
  - *Approx methods:* $O(t^2r^4+mr+r^6)$ [Kang+SDM12]

- U. Kang, Hanghang Tong, Jimeng Sun. Fast Random Walk Graph Kernel. SDM 2012
- S. V. N. Vishwanathan, N. N. Schraudolph, I. Kondor, and K. M. Borgwardt. Graph Kernels. JMLR 2010.

TEAMREP-BASIC

Find a new member *q* not in the current team that satisfies:

$$q = \arg\max_{j, j \notin \mathcal{T}} \mathrm{Ker}(G(\mathcal{T}), G(\mathcal{T}_{p \to j}))$$



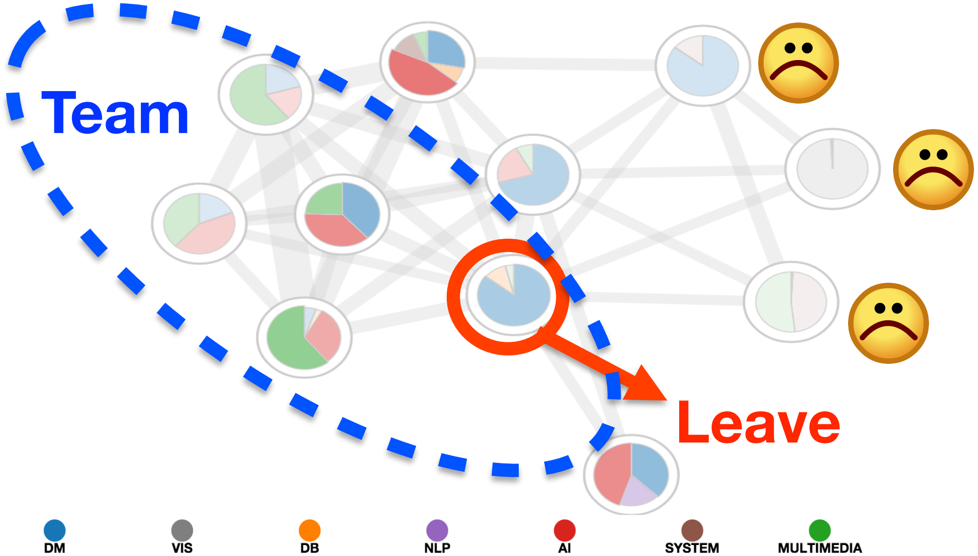
One graph kernel computation for every possible candidate

- **Challenge:** need to compute many graph kernel
  overall complexity: ***O(nt³)***
- **Questions:**
  - **Q1:** how to reduce the number of graph kernels
  - **Q2:** how to speed up the computation for each graph kernel

• L. Li, H. Tong, N. Cao, K. Ehrlich, Y.-R. Lin and N. Buchler: Replacing the Irreplaceable: Fast Algorithms for Team Member Recommendation, WWW 2015

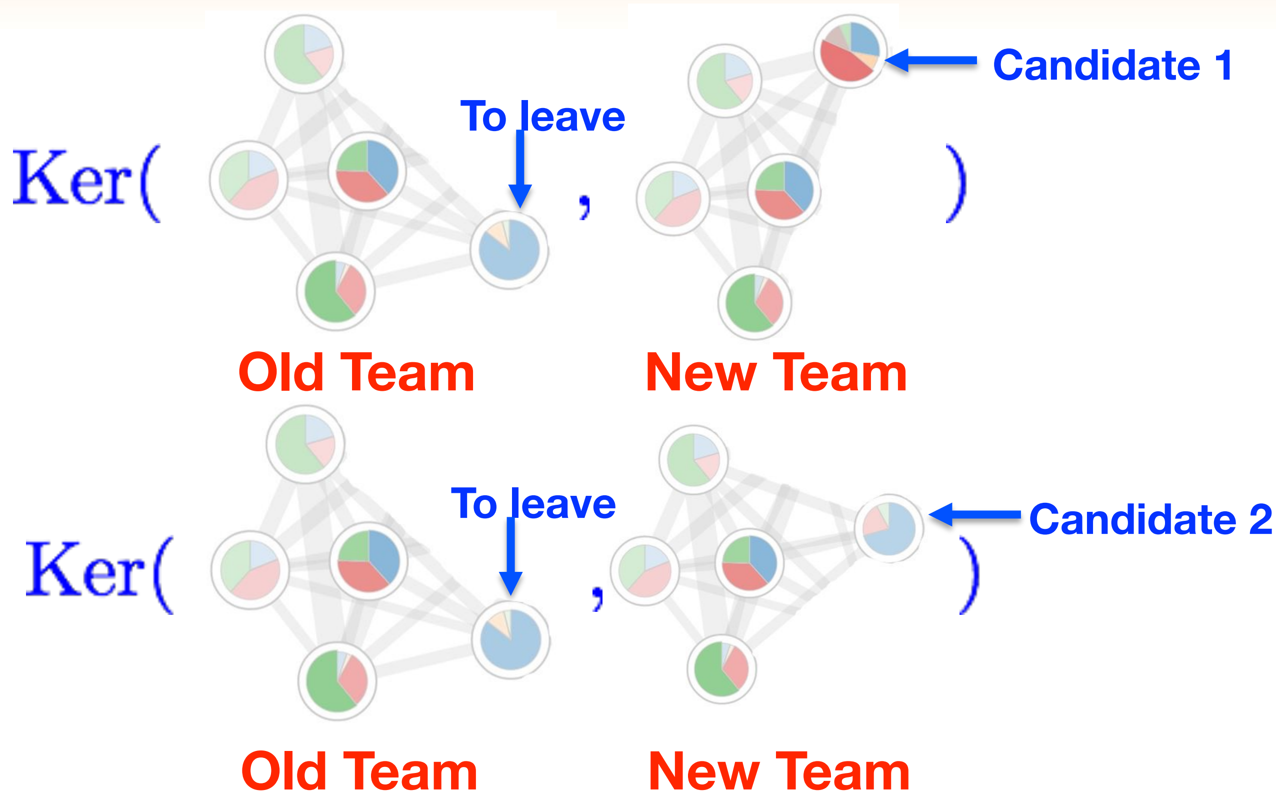
Scale-up: Candidate Filtering

***Pruning Strategy*:** Filter out all the candidates w/o any connections to any of the rest team members.



- **Theorem:** The pruning is safe: wont' miss any potentially good replacement
- **Benefit:** The number of graph kernel computations is reduced to O(size of the neighborhood of T) $O(\sum_{i \in \mathcal{T}/p} d_i)$

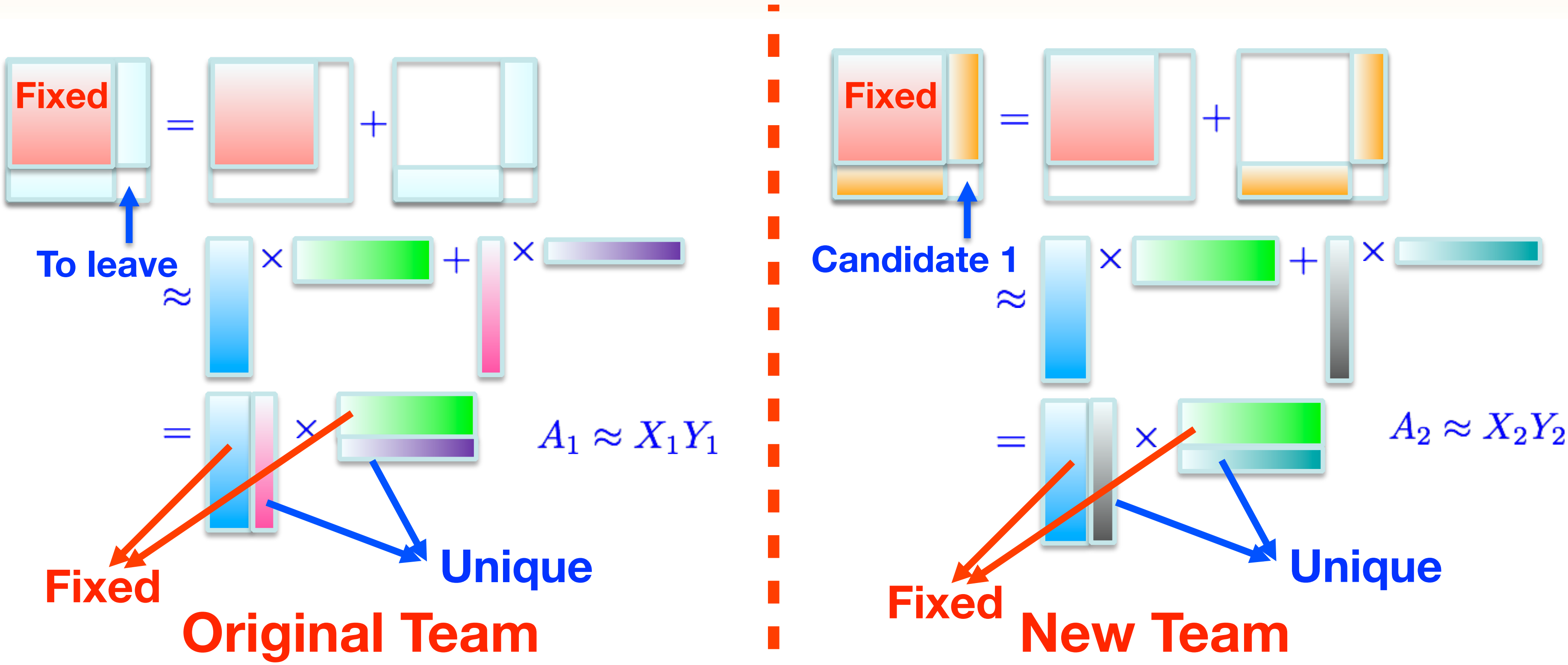
Speedup — Observation



Observation:

Many redundancies — *the nodes and edges within the rest team members remain the same*

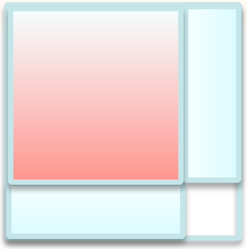
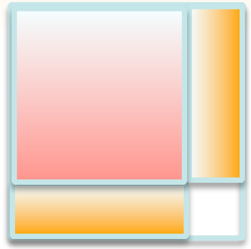
Speedup — Approx Approach



The common part is the adjacency matrix of the rest team members

Speedup — Approx Approach

Details

Ker( , )

$$\approx y'(1 - cL_\times (X_1Y_1)\otimes(X_2Y_2))^{-1}L_\times x$$

$$= y'L_\times x + cy'L_\times (X_1 \otimes X_2)\boxed{M}(Y_1 \otimes Y_2))L_\times x$$

$$M = (I - c(\sum_{j=1}^{l} Y_1 L_1^{(j)} X_1 \otimes Y_2 L_2^{(j)} X_2))^{-1}$$

M is of size $(r+2)^2 \times (r+2)^2$

Time Complexity: $O((\sum_{i\in\mathcal{T}/p} d_i)(lt^2r + r^6))$

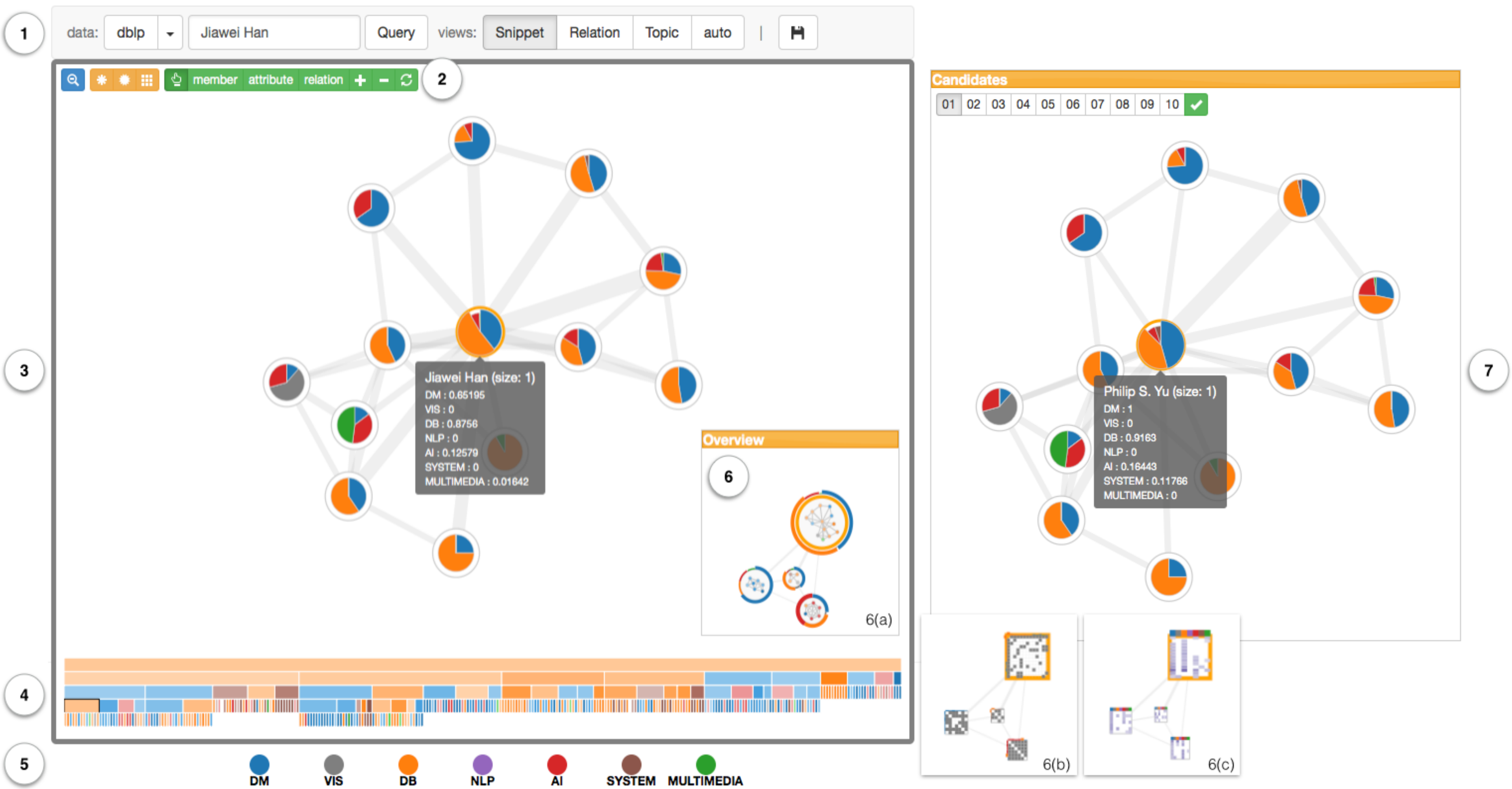
$[\sum_{i\in\mathcal{T}/p} d_i \ll n, r \ll t]$

Original Complexity: $O(nt^3)$

Prototype Systems

Questions

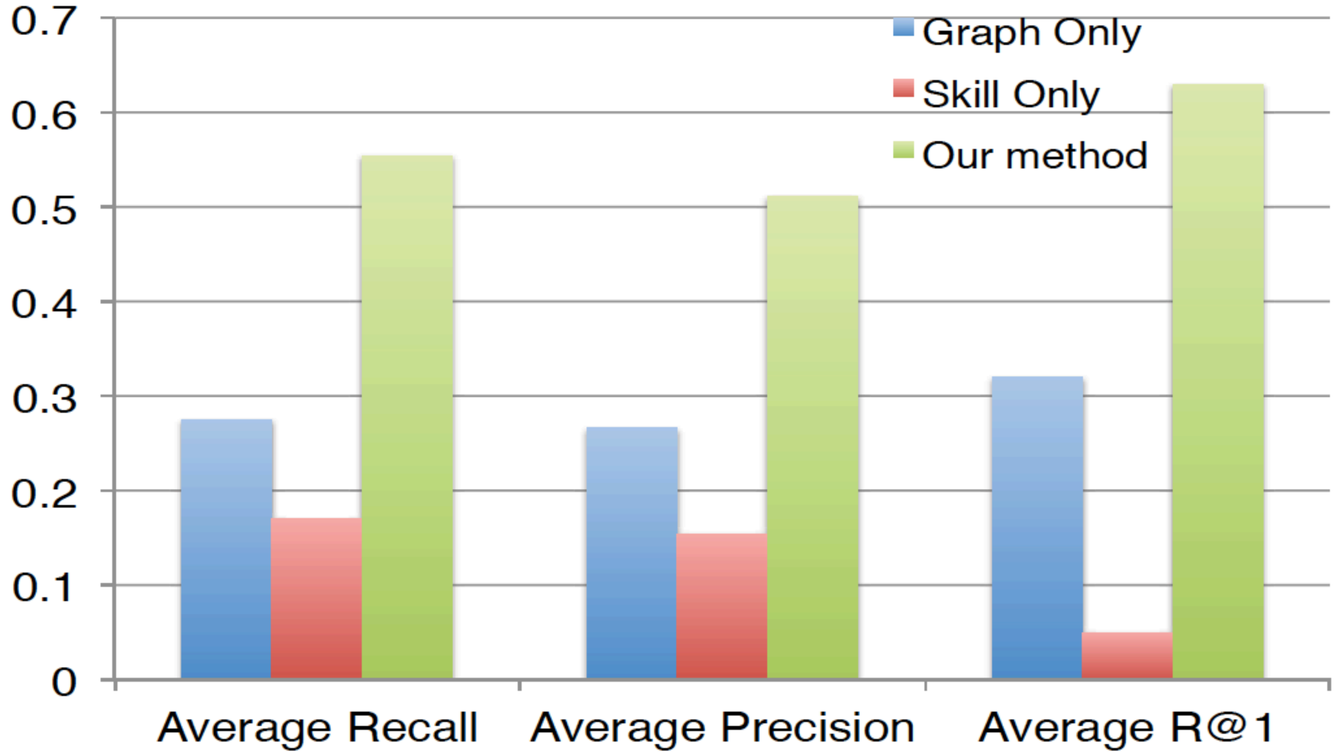
- **Q1: How effective is skill + structure?**
- **Q2**: How fast is pruning?
- **Q3:** How fast is proposed solution?
- **Q4**: How is the scalability?



prototype: http://team-net-work.org

- Nan Cao, Yu-Ru Lin, Liangyue Li, Hanghang Tong."g-Miner: Interactive Visual Group Mining on Multivariate Graphs", ACM **CHI** 2015.

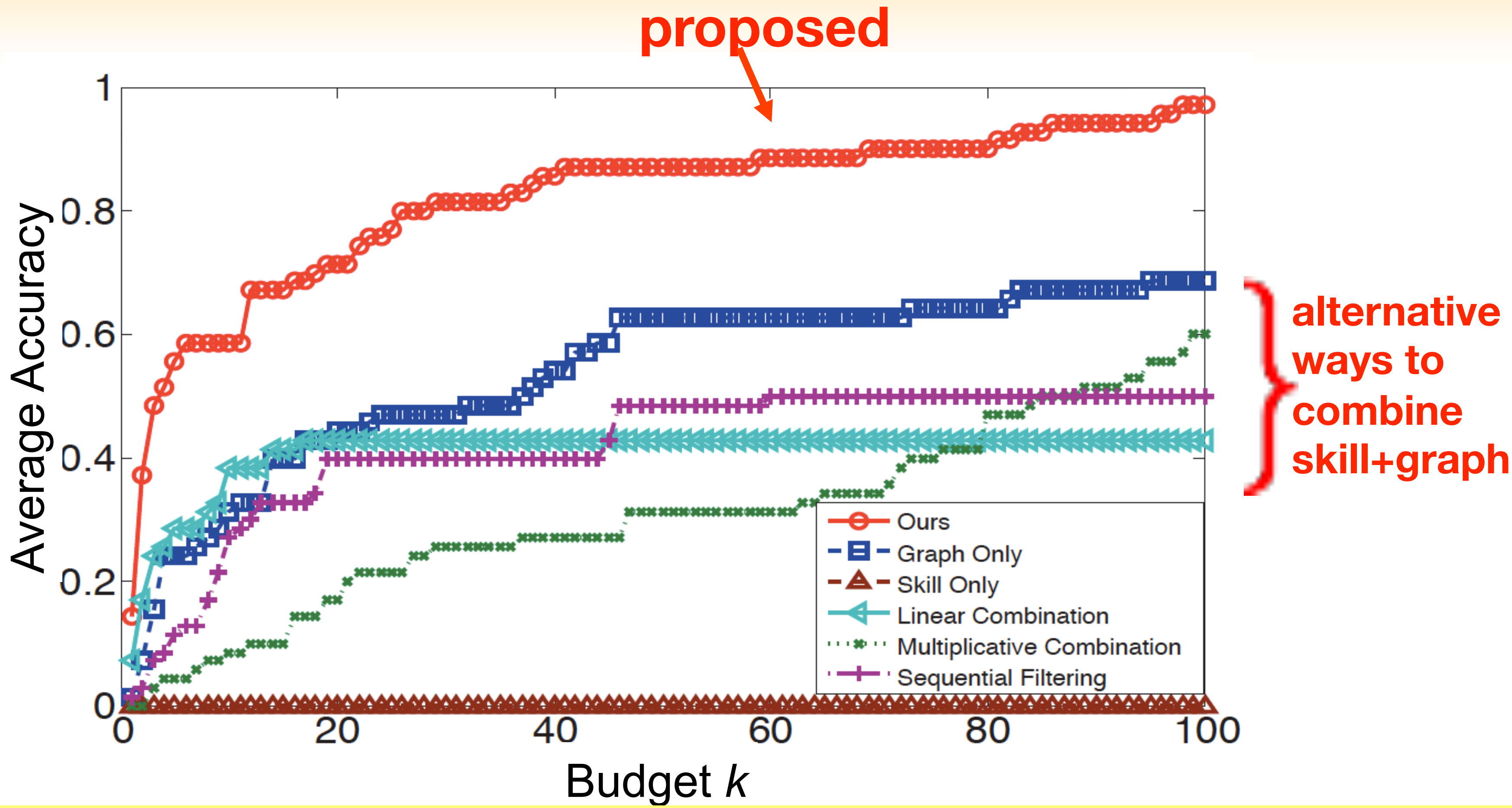
User Studies



Our method achieves the best average recall, precision and R@1

- L. Li, H. Tong, N. Cao, K. Ehrlich, Y.-R. Lin and N. Buchler: Replacing the Irreplaceable: Fast Algorithms for Team Member Recommendation, WWW 2015

Application in Author Alias Prediction

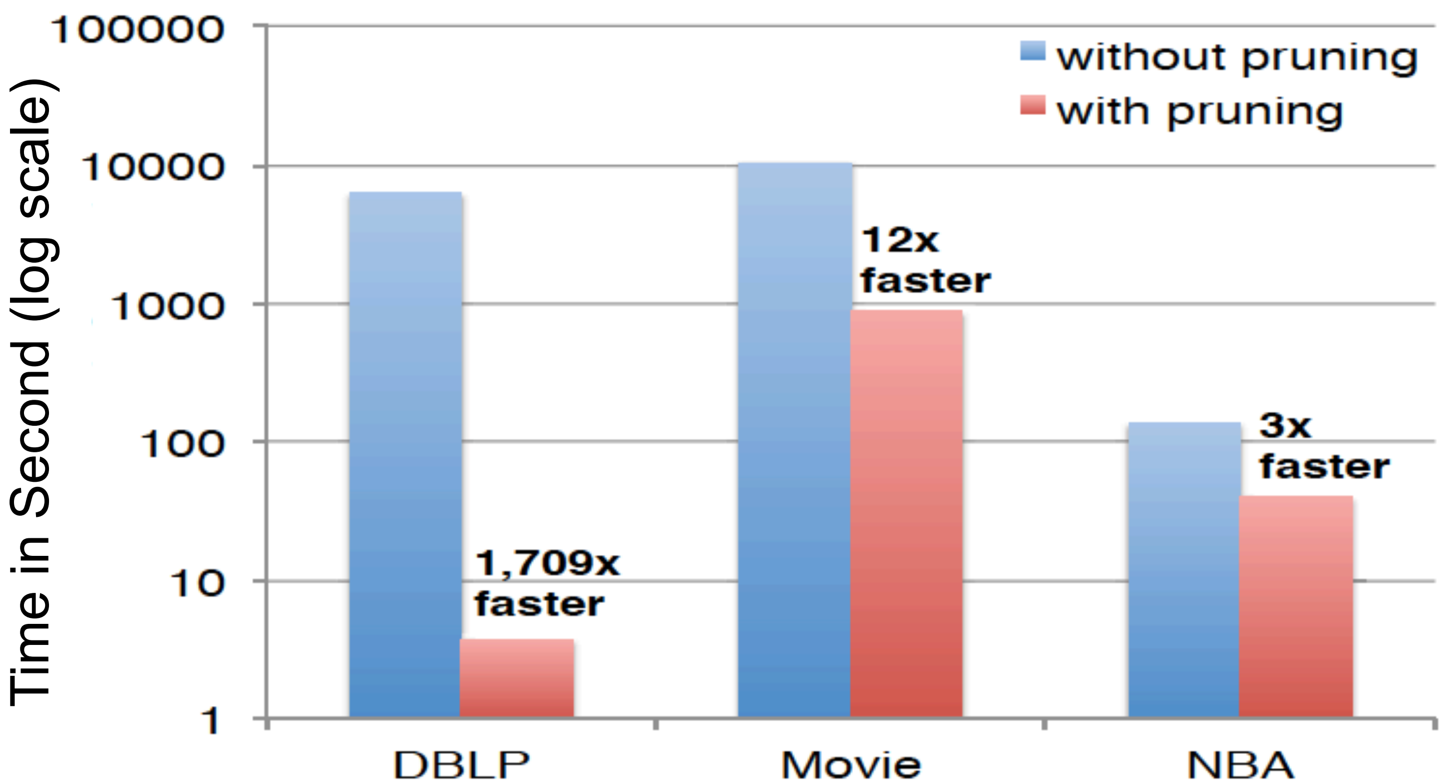


Our method achieves the highest accuracy

Author Alias: *Alexander J. Smola* vs. *Alex J. Smola*

Speed-up by Pruning

Questions
- **Q1**: How effective is skill + structure?
- **Q2: How fast is pruning?**
- **Q3:** How fast is proposed solution?
- **Q4**: How is the scalability?

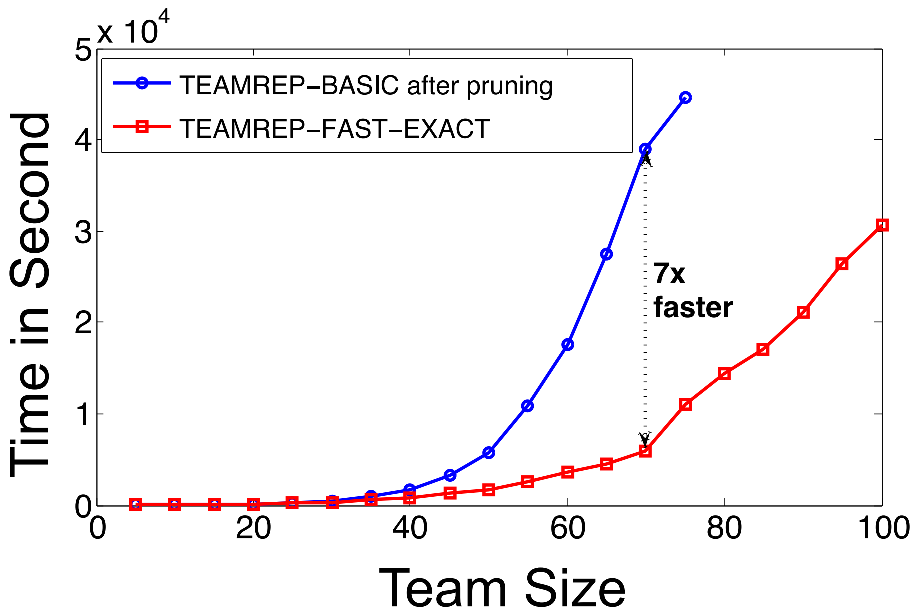


Pruning has dramatic speed improvement

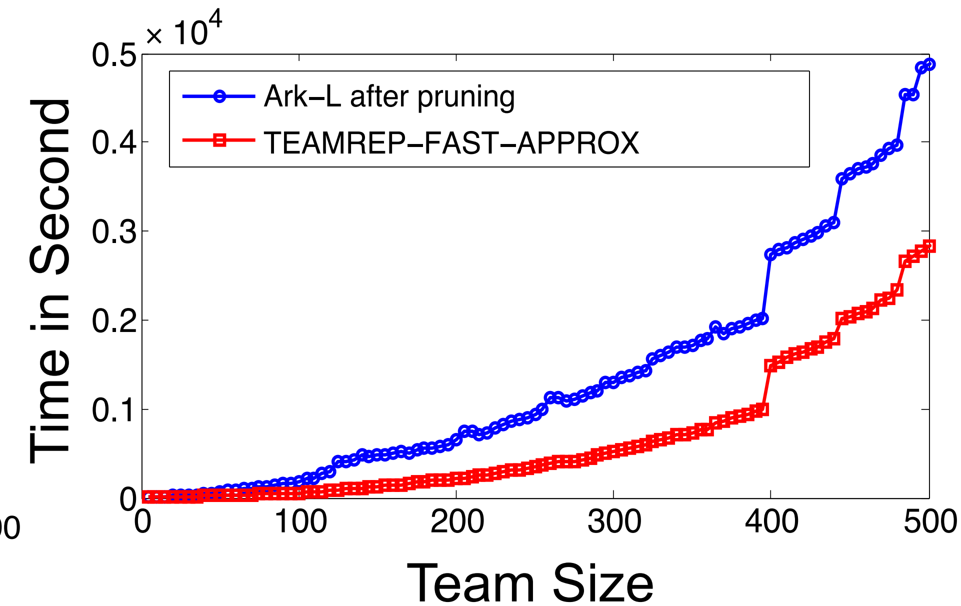
Further Speed-up

Questions

- **Q1**: How effective is skill + structure?
- **Q2**: How fast is pruning?
- **Q3: How fast is proposed solution?**
- **Q4**: How is the scalability?



Exact methods



Approximate methods

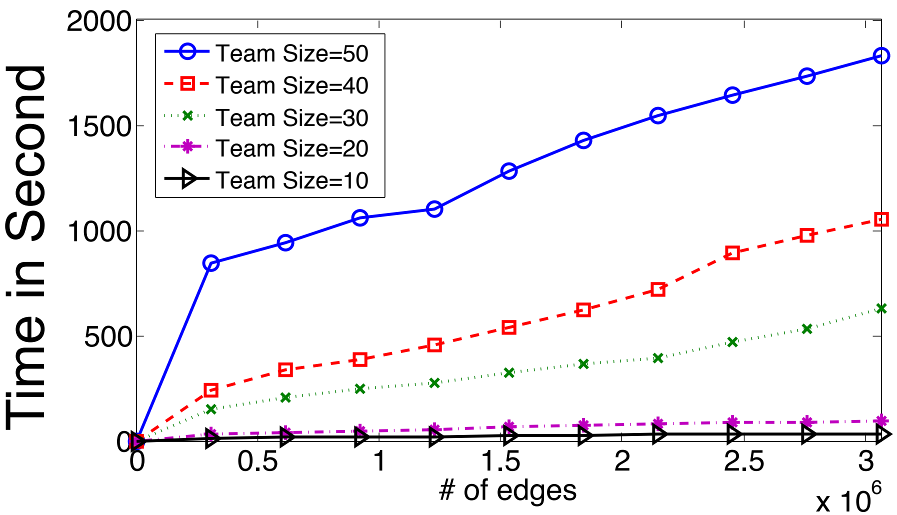
Exploiting redundancy leads to additional speed-up!

DATA Lab

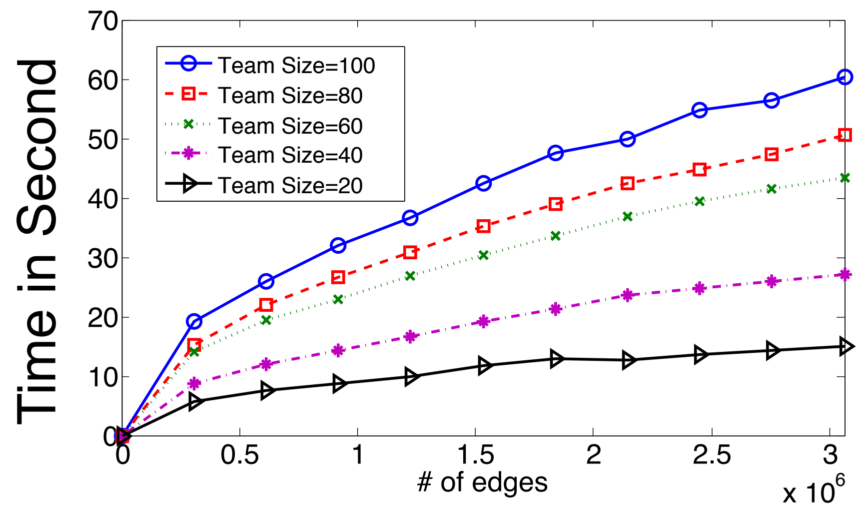
Scalability

Questions

- **Q1**: How effective is skill + structure?
- **Q2**: How fast is pruning?
- **Q3:** How fast is proposed solution?
- **Q4**: **How is the scalability?**



TEAMREP-FAST-EXACT

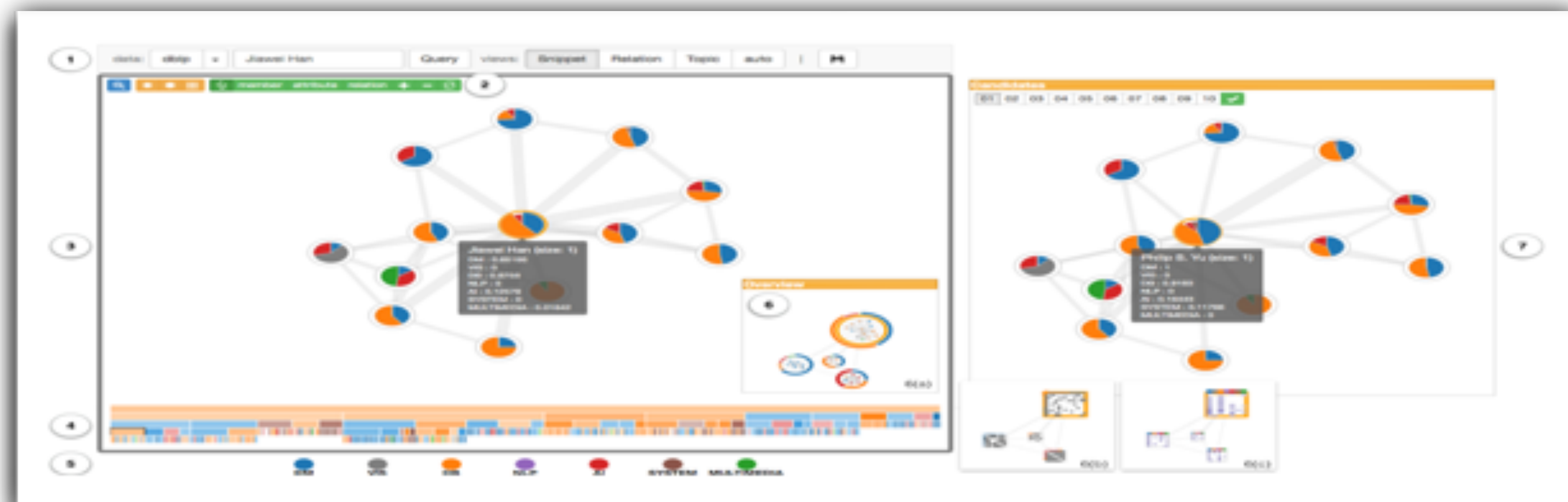


TEAMREP-FAST-APPROX

Our fast solutions scale sub-linearly

Team Member Replacement - Summary

- **Problem Def**: Team Member Replacement
- **Design Objectives**: skill + structural matching
- **Solutions**: graph kernel and fast algorithms
- **Prototype Systems:** http://team-net-work.org/



- L. Li, H. Tong, N. Cao, K. Ehrlich, Y.-R. Lin and N. Buchler: Replacing the Irreplaceable: Fast Algorithms for Team Member Recommendation, WWW 2015
- N. Cao, Y.-R. Lin, L. Li, H. Tong: g-Miner: Interactive Visual Group Mining on Multivariate Graphs, ACM CHI 2015

Beyond Team Member Replacement

- **Team Shrinkage**
  - If we need to reduce the size of an existing team (e.g., for the purpose of cost reduction), who shall leave the team?
- **Team Expansion**
  - If the team leader perceives the need to enhance certain expertise of the entire team, who shall we bring into the team?
- **Team Conflict Resolution**
  - If the team leader sees a conflict between certain team members, how shall we resolve it?

Key Idea: Solve all these team enhancement scenarios by team member replacement !

- L. Li, H. Tong, N. Cao, K. Ehrlich, Y.-R. Lin and N. Buchler: Enhancing Team Composition in Professional Networks: Problem Definitions and Fast Solutions, 2016

Open Challenges

- Team Performance Characterization
  - Correlation → Causality
  - When does “1+1 < 2” ?
- Team Performance Prediction
  - Joint Content-Individual-Team Prediction
  - Prediction→ Attribution
- Team Performance Optimization
  - Predictive Optimization
  - Team Optimization → Network Optimization

Acknowledgement

Norbou Buchler, Tina Eliassi-Rad, Nan Cao, Kate Erhlich, Wei Fan, Liangyue Li, Yuru Lin, Jie Tang, Brian Uzzi, Dashun Wang

Project website: http://team-net-work.org (for data, paper, slides and systems, etc)