HiDDen: Hierarchical Dense Subgraph Detection with Application to Financial Fraud Detection

Presented by Si Zhang (ASU)


Si Zhang


Dawei Zhou


Mehmet Yigit Yildirim


Scott Alcorn

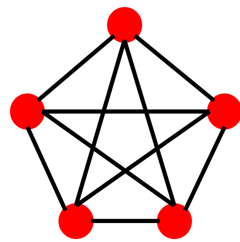

Jingrui He


Hasan Davulcu


Hanghang Tong

Dense Subgraph: What?

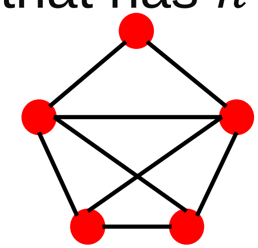
- Def: A subgraph of a high density
- Examples:
  - Clique: each node connects to every other node in the graph



density=1

K_5

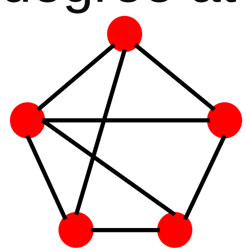
  - α-Quasi-Clique: a graph that has n nodes and at least $\frac{\alpha n(n-1)}{2}$ edges



density=0.8

0.8-Quasi-Clique

  - K-core: each node has a degree at least k

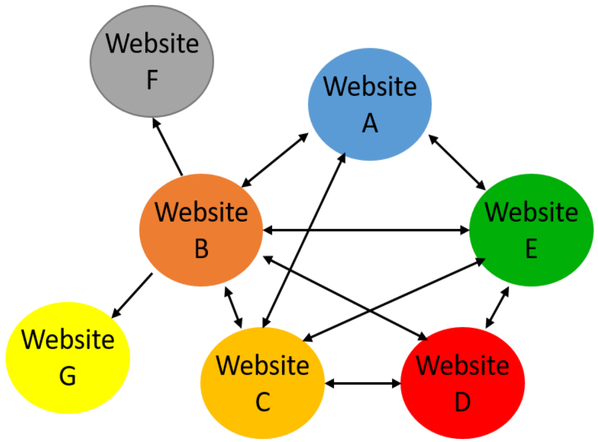


density=0.8

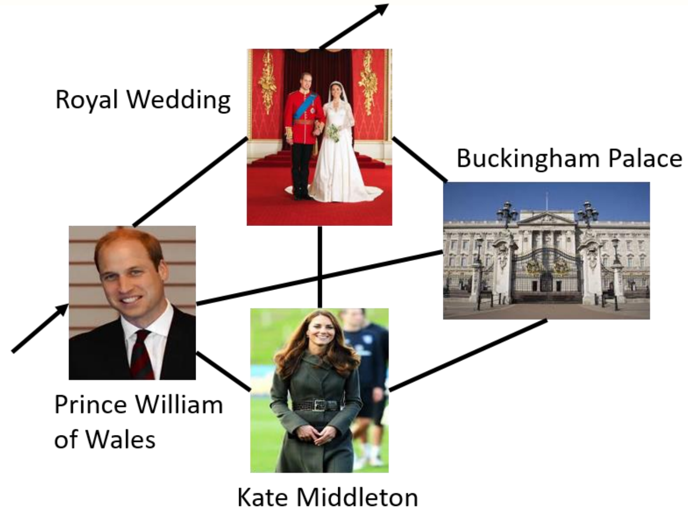
3-core

Dense Subgraph: Why?

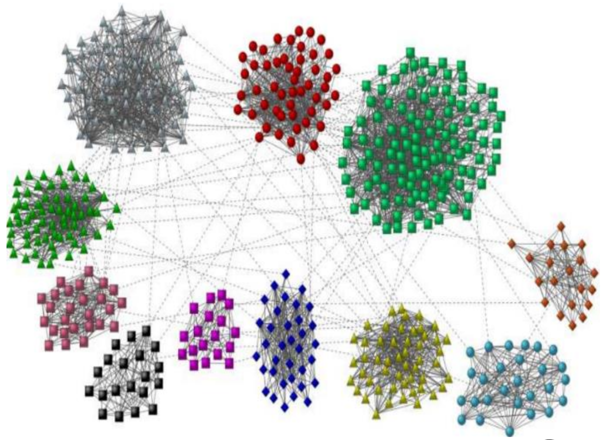
- Applications



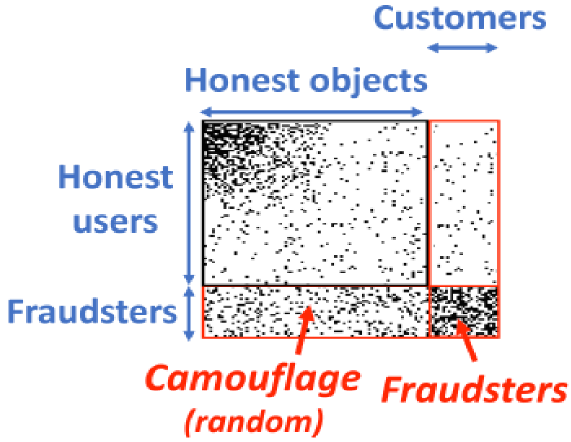
Spam Link Farms [Gibson'05]



Story Identification [Angel'13]



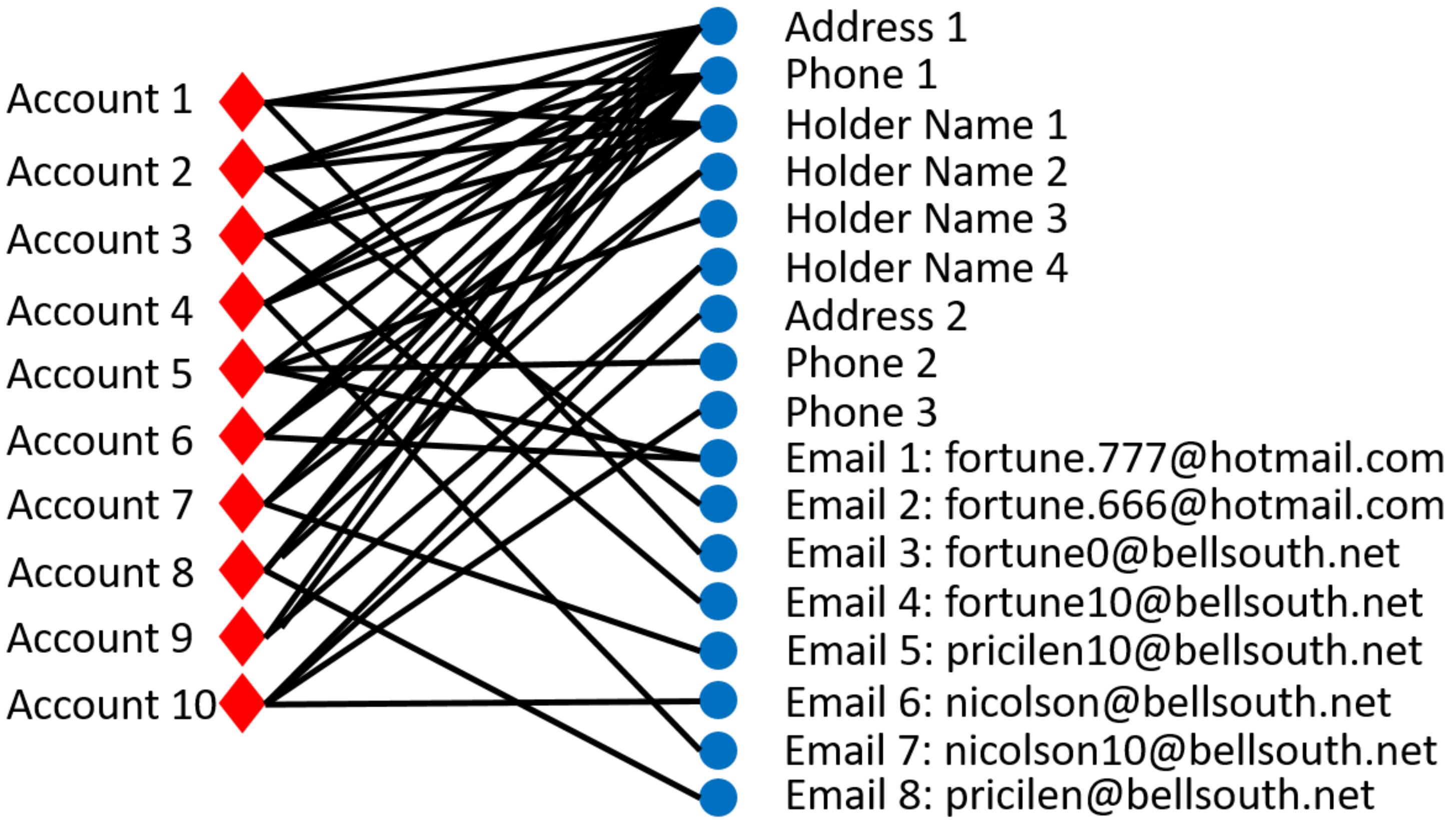
Community Detection [Sozio'10]



Fraud Detection [Hooi'16]

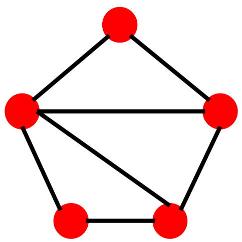
Dense Subgraph: Why?

- Synthetic Identity Detection



Dense Subgraph: How?

- Density Measures
  - Edge density: $d = \frac{2m}{n(n-1)}$
  - Average degree: $d = \frac{2m}{n}$
  - Triangle density: $d = \frac{\#\ of\ triangles}{n(n-1)(n-2)/6}$



edge density=0.8

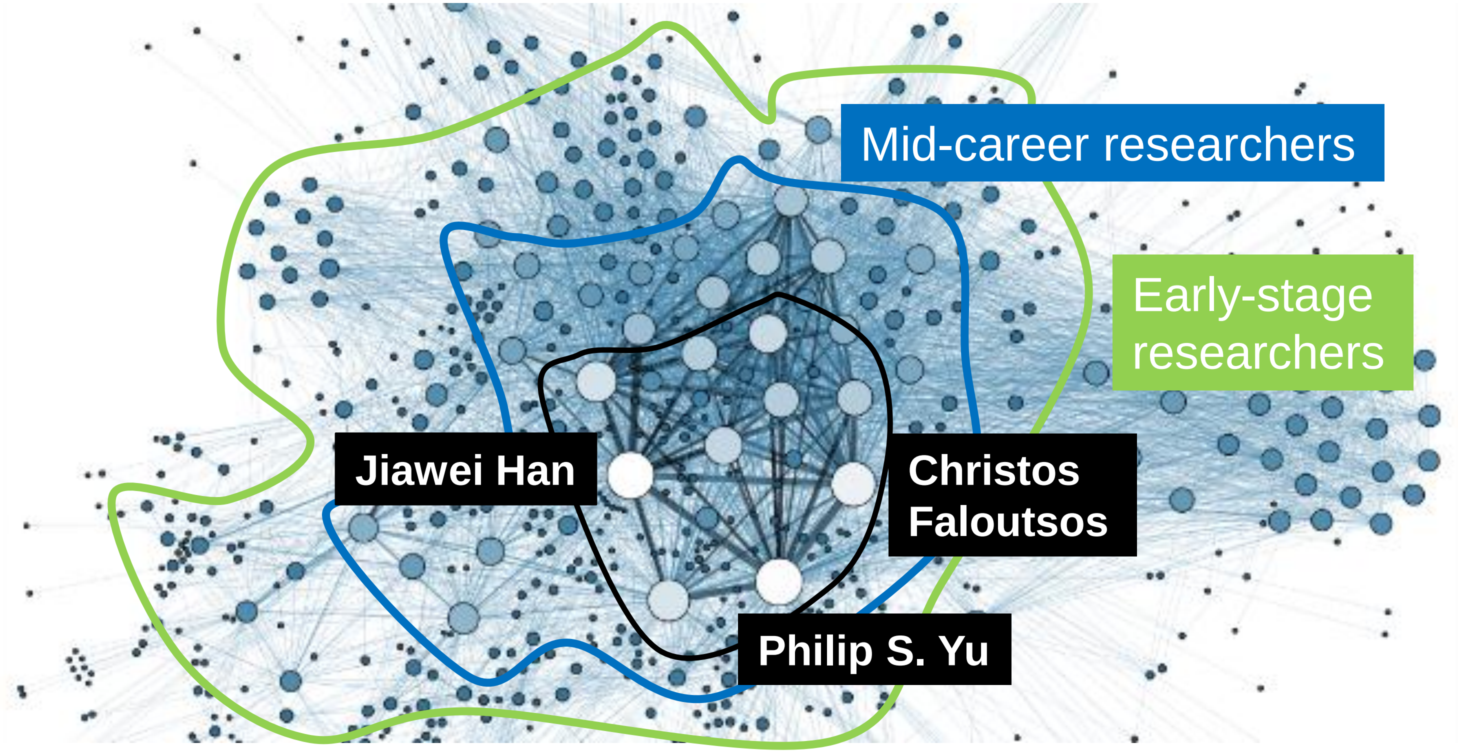
average degree=3.2

triangle density=0.3

- Existing Methods
  - Densest subgraph: greedy method [Charikar'00]
  - k-clique [Tsourakakis'15], k-core, k-plex
  - Denser than the densest [Tsourakakis'13]
- Key Idea: to flatly extract one or more partitions in a graph

Why Hierarchical Dense Subgraphs?

- A more comprehensive view of dense subgraph structures
- Example:



Challenges: Hierarchical Dense Subgraphs

- C1. Optimization Formulation
  - Flat detection: quadratic optimization constrained on simplex

$$\max_{\boldsymbol{x}} \ \boxed{\boldsymbol{x}^T \boldsymbol{A} \boldsymbol{x}}$$

To maximize the number of edges in the subgraph

$$s.t \quad \sum_{i=1}^{n} \boldsymbol{x}_i^{\beta} = 1, \qquad \boldsymbol{x}_i \geq 0$$

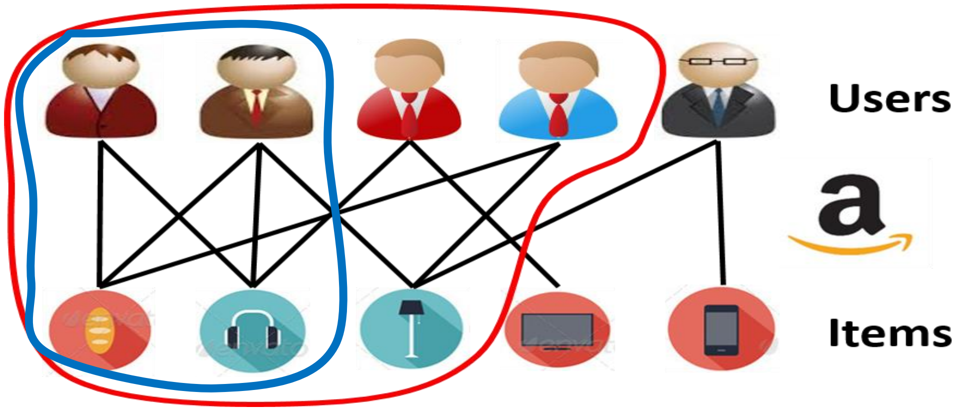
  - **Question**: how to formulate multiple hierarchies together?

- C2. Optimization Algorithm
  - Flat detection: non-convex or polynomial approximation
  - **Question**: how to develop an effective and scalable algorithm?

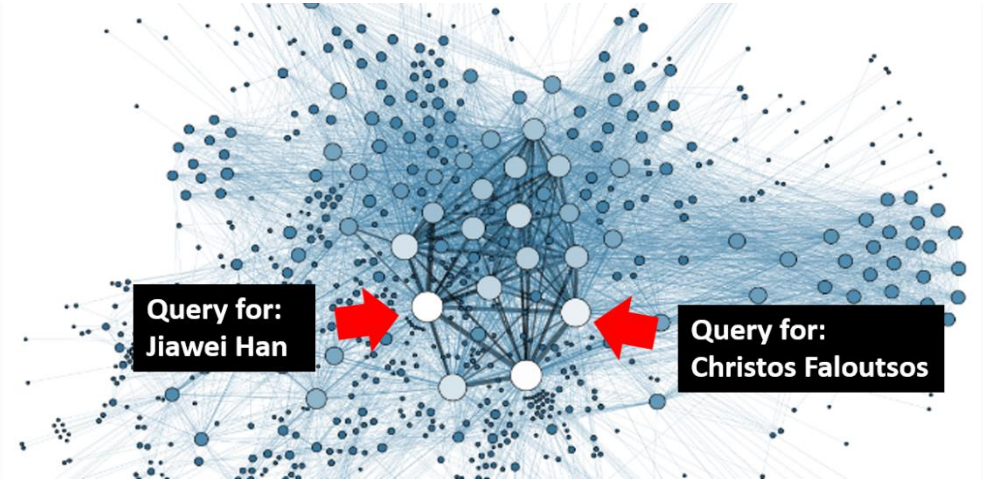
DATA Lab

Challenges: Hierarchical Dense Subgraphs

- C3. Generalizations
  - **Question**: How to generalize to bipartite graphs?



  - **Question**: How to detect for a set of certain query nodes?

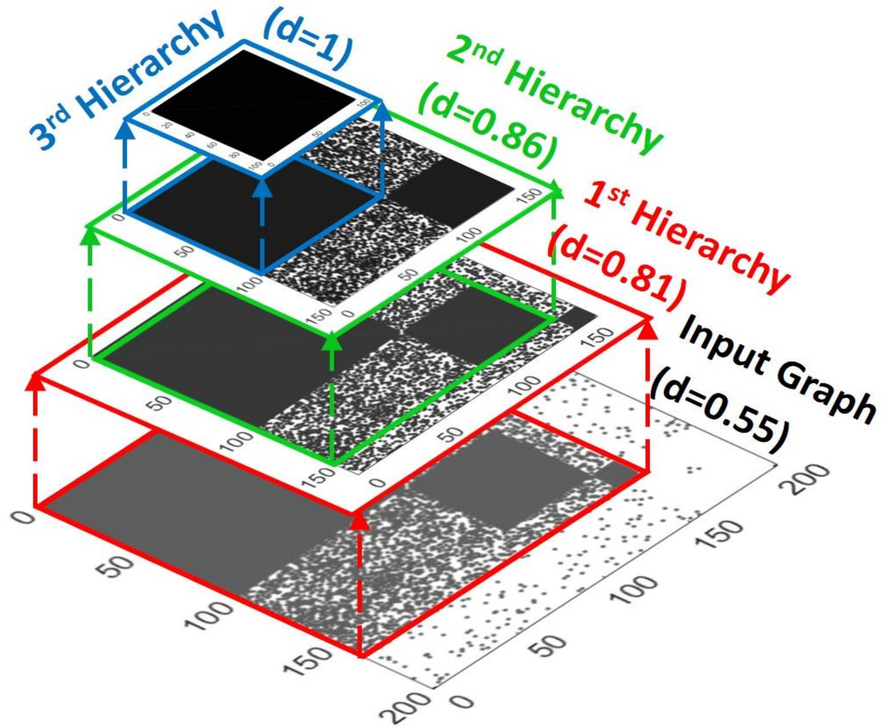


Outline

- Motivations ✓
- Q1: HiDDen Formulation
- Q2: HiDDen Algorithm
- Q3: HiDDen Generalizations
- Experimental Results
- Conclusions

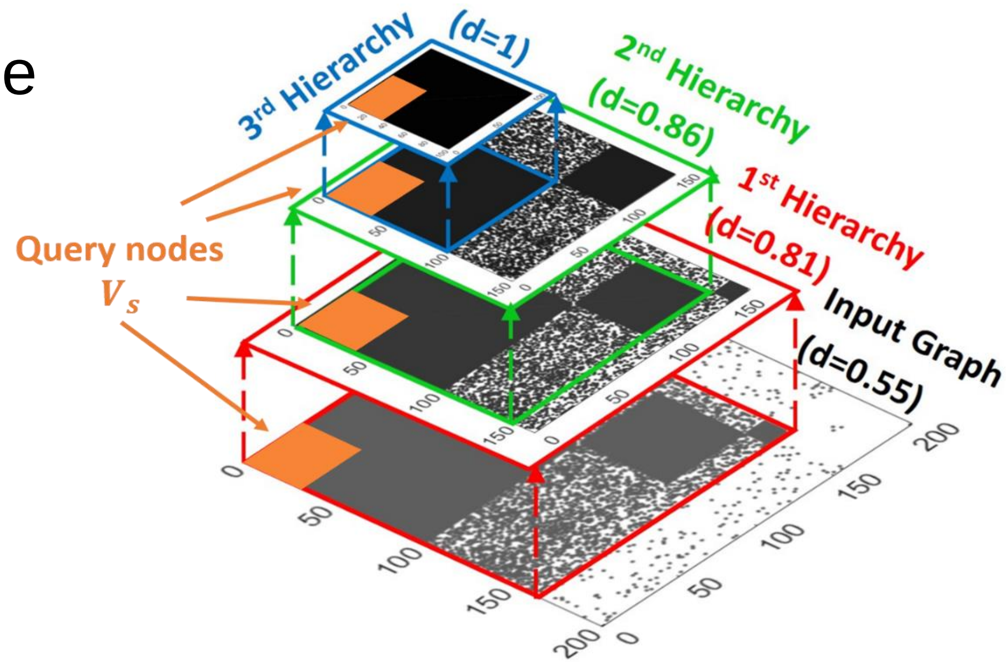
Prob. Def: Hierarchical Dense Subgraph Detection

- **Given**:
  - (1) adjacency matrix $\boldsymbol{A}$; (2) missing edge penalty p
  - (3) number of hierarchies K; (4) density increase ratio η.
- **Output**: subgraph node indicator vectors $\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_K$.
- An Illustrative Example



Prob. Def: Query-Specific Hierarchical Dense Subgraph Detection

- **Given**:
  - (1) adjacency matrix $\boldsymbol{A}$; (2) missing edge penalty p
  - (3) number of hierarchies K; (4) density increase ratio η;
  - (5) query node set V_s.
- **Output**: subgraph node indicator vectors $\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_K$.
- An Illustrative Example



HiDDen Formulation: Density Measure

- Intuition:
  - #1: Maximize the number of existing edges
  - #2: Minimize the penalty of the missing edges
- Mathematical Details:

$$\max_{\boldsymbol{x}} \quad J(\boldsymbol{x}) = \boldsymbol{x}^T \boldsymbol{A} \boldsymbol{x} - p\boldsymbol{x}^T(\mathbf{1}_{n\times n} - \boldsymbol{I} - \boldsymbol{A})\boldsymbol{x}$$

$$s.t \quad \boldsymbol{x} \in \{0,1\}^n$$

Intuition #1 ($\boldsymbol{x}^T \boldsymbol{A} \boldsymbol{x}$)

Intuition #2 ($\boldsymbol{x}^T(\mathbf{1}_{n\times n} - \boldsymbol{I} - \boldsymbol{A})\boldsymbol{x}$)

- Correctness:
  - Equivalent to edge surplus density w.r.t quasi-clique
- Relaxation:

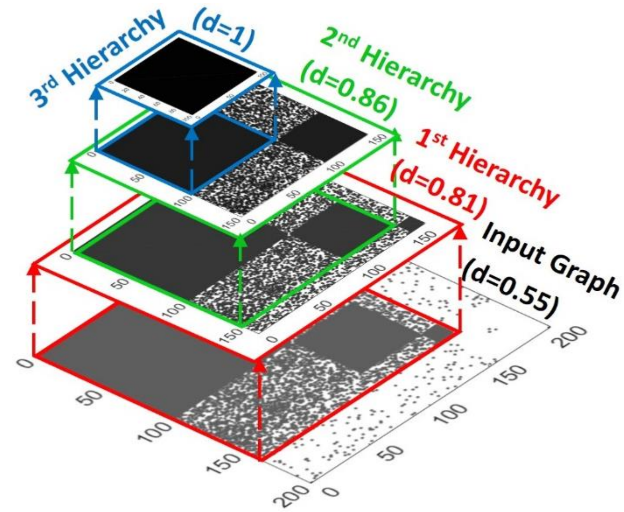
$$\boldsymbol{x} \in \{0,1\}^n \longrightarrow \mathbf{0} \le \boldsymbol{x} \le \mathbf{1}$$

HiDDen Formulation: Constraints for Hierarchies

- Constraints:
  - #1 – Density variety: densities in two hierarchies exhibit a difference
  - #2 – Nested node set: larger subgraphs contain smaller subgraphs
- Mathematical Details:
  - Density variety:

$$\frac{\left(\boldsymbol{x}^k\right)^T \boldsymbol{A}\boldsymbol{x}^k}{(\boldsymbol{x}^k)^T(\mathbf{1}_{n\times n} - \boldsymbol{I})\boldsymbol{x}^k} \geq \eta \frac{\left(\boldsymbol{x}^{k-1}\right)^T \boldsymbol{A}\boldsymbol{x}^{k-1}}{(\boldsymbol{x}^{k-1})^T(\mathbf{1}_{n\times n} - \boldsymbol{I})\boldsymbol{x}^{k-1}}$$

Example: $d_3 \geq 1.1 \times d_2$



  - Nested node set:

$$V^{k+1} \subseteq V^k \subseteq V^{k-1} \longrightarrow \boldsymbol{x}^{k+1} \leq \boldsymbol{x}^k \leq \boldsymbol{x}^{k-1}$$

Example: $V^3 \subseteq V^2 \subseteq V^1 \subseteq V$

HiDDen Formulation: Objective Function

- Objective function:

$$\max_{\boldsymbol{x}_1,\boldsymbol{x}_2,\ldots,\boldsymbol{x}_K} \sum_{k=1}^{K} \left(\boldsymbol{x}^k\right)^T [(1+p)\boldsymbol{A} - p(\boldsymbol{1}_{n\times n} - \boldsymbol{I})]\boldsymbol{x}^k$$

edge surplus in k-th hierarchy

$$s.t \quad \frac{\left(\boldsymbol{x}^j\right)^T \boldsymbol{A}\boldsymbol{x}^j}{(\boldsymbol{x}^j)^T(\boldsymbol{1}_{n\times n} - \boldsymbol{I})\boldsymbol{x}^j} \geq \eta \frac{\left(\boldsymbol{x}^{j-1}\right)^T \boldsymbol{A}\boldsymbol{x}^{j-1}}{(\boldsymbol{x}^{j-1})^T(\boldsymbol{1}_{n\times n} - \boldsymbol{I})\boldsymbol{x}^{j-1}}$$

density variety

$$\boldsymbol{x}^{j+1} \leq \boldsymbol{x}^j \leq \boldsymbol{x}^{j-1}$$

nested node set

$$\forall\, j = 1, 2, \ldots, K$$

- **Observation**: a non-convex quadratic constrained quadratic programming problem (QCQP)
- **Question**: can we simplify the problem?

HiDDen Formulation: QCQP Relaxation

- Constraint #1 Relaxation:
  - Relax it to a regularization, i.e.,

$$\frac{(\boldsymbol{x}^j)^T \boldsymbol{A}\boldsymbol{x}^j}{(\boldsymbol{x}^j)^T(\mathbf{1}_{n\times n} - \boldsymbol{I})\boldsymbol{x}^j} \geq \eta \frac{(\boldsymbol{x}^{j-1})^T \boldsymbol{A}\boldsymbol{x}^{j-1}}{(\boldsymbol{x}^{j-1})^T(\mathbf{1}_{n\times n} - \boldsymbol{I})\boldsymbol{x}^{j-1}}$$

relax

$$\max_{\boldsymbol{x}^j} \quad (\boldsymbol{x}^j)^T \boldsymbol{A}\boldsymbol{x}^j - C^{j-1}(\boldsymbol{x}^j)^T(\mathbf{1}_{n\times n} - \boldsymbol{I})\boldsymbol{x}^j$$

where $C^{j-1} = \eta \frac{(\boldsymbol{x}^{j-1})^T \boldsymbol{A}\boldsymbol{x}^{j-1}}{(\boldsymbol{x}^{j-1})^T(\mathbf{1}_{n\times n} - \boldsymbol{I})\boldsymbol{x}^{j-1}}$ is a constant w.r.t $\boldsymbol{x}^j$

  - Relax to a quadratic optimization
  - Intrinsically increase the missing edge penalties in each hierarchy

HiDDen Formulation: Overall Objective Function

- Overall objective function

for 1st hierarchy

for k-th hierarchy

$$\min_{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_K} -(1+p)(\boldsymbol{x}^1)^T \boldsymbol{A}\boldsymbol{x}^1 + p(\|\boldsymbol{x}^1\|_1^2 - \|\boldsymbol{x}^1\|_2^2)$$

$$-(1+p+\beta)\sum_{k=2}^{K}(\boldsymbol{x}^k)^T \boldsymbol{A}\boldsymbol{x}^k + \sum_{k=2}^{K}(p+\beta C^{k-1})\left(\|\boldsymbol{x}^k\|_1^2 - \|\boldsymbol{x}^k\|_2^2\right)$$

$$s.t. \quad \boldsymbol{x}^{j+1} \le \boldsymbol{x}^j \le \boldsymbol{x}^{j-1}, \qquad \forall\, j = 1, 2, \ldots, K$$

  - p is the parameter of missing edge penalty
  - β controls the importance of the constraint relaxation
  - $p + \beta C^{j-1}$ is the increased penalty for the k-th hierarchy

Outline

- Motivations ✓
- Q1: HiDDen Formulation ✓
- Q2: HiDDen Algorithm
- Q3: HiDDen Generalizations
- Experimental Results
- Conclusions

HiDDen Algorithm

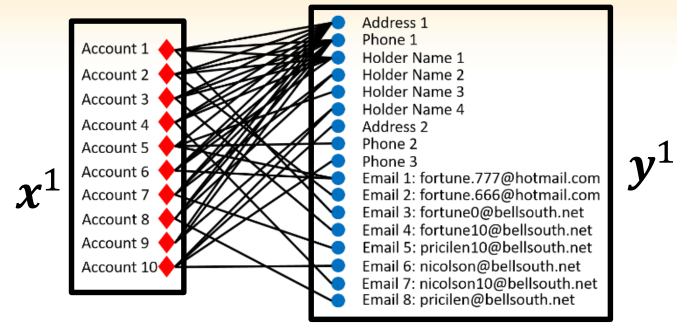
- Observation: a non-convex quadratic optimization problem
- Solution: alternative projected gradient descent method
  - $\nabla_{\boldsymbol{x}^1} f = -2(1+p)\boldsymbol{A}\boldsymbol{x}^1 + 2p\|\boldsymbol{x}^1\|_1 \mathbf{1} - 2p\boldsymbol{x}^1$
  - $\nabla_{\boldsymbol{x}^k} f = -2(1+p+\beta)\boldsymbol{A}\boldsymbol{x}^k + 2\left(p+\beta C^{k-1}\right)\left(\left\|\boldsymbol{x}^k\right\|_1 \mathbf{1} - \boldsymbol{x}^k\right)$
  - Armijo's rule line search
  - Stopping criterion: adopted from [Lin 2007]
- Benefits:
  - Converge to a stationary point
  - Time complexity: $O(mK)$
- **Question**: how to generalize to bipartite graph & query-specific

Outline

- Motivations ✓
- Q1: HiDDen Formulation ✓
- Q2: HiDDen Algorithm ✓
- Q3: HiDDen Generalizations
- Experimental Results
- Conclusions

HiDDen Generalizations: Bipartite Graph

- Key idea: indicator vectors for two types of nodes, $\boldsymbol{x}^k$ & $\boldsymbol{y}^k$ $(k = 1, \cdots, K)$



- Objective function:

for 1st hierarchy

for k-th hierarchy

$$\min_{\boldsymbol{x}^1,\dots,\boldsymbol{x}^K,\boldsymbol{y}^1,\dots,\boldsymbol{y}^K} -(1+p)(\boldsymbol{x}^1)^T\boldsymbol{A}\boldsymbol{y}^1 + p\|\boldsymbol{x}^1\|_1\|\boldsymbol{y}^1\|_1 - (1+p+\beta)\sum_{k=2}^{K}(\boldsymbol{x}^k)^T\boldsymbol{A}\boldsymbol{y}^k + \sum_{k=2}^{K}\left(p+\beta C^{k-1}\right)\left\|\boldsymbol{x}^k\right\|_1\left\|\boldsymbol{y}^k\right\|_1$$

$$s.t. \quad \boldsymbol{x}^{j+1} \le \boldsymbol{x}^j \le \boldsymbol{x}^{j-1}, \boldsymbol{y}^{j+1} \le \boldsymbol{y}^j \le \boldsymbol{y}^{j-1}, \forall\, j = 1, 2, \dots, K$$

- Solution: alternative projected gradient descent method
  - Alternate between $\boldsymbol{x}^1, \dots, \boldsymbol{x}^K$ and $\boldsymbol{y}^1, \dots, \boldsymbol{y}^K$
  - Stopping criterion: similar to previous

HiDDen Generalizations: Query–Specific

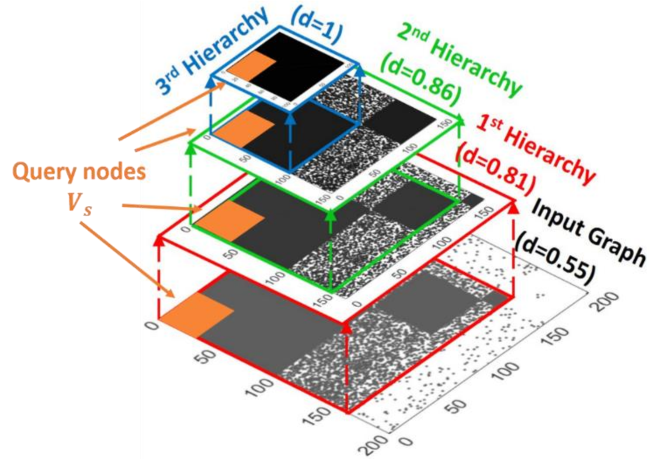
- Intuition: constrain $x_i^k = 1, \quad \text{for } i \in V_s$
- Challenges: could lead to a mixed integer problem
- Key Idea: relax to $x_i^k \geq \delta$, where $\delta \in (0, 1)$ is relatively large
- Objective function:

$$\min_{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_K} -(1+p)(\boldsymbol{x}^1)^T \boldsymbol{A}\boldsymbol{x}^1 + p(\|\boldsymbol{x}^1\|_1^2 - \|\boldsymbol{x}^1\|_2^2) - (1+p+\beta)\sum_{k=2}^{K}\left(\boldsymbol{x}^k\right)^T \boldsymbol{A}\boldsymbol{x}^k$$

$$+\sum_{k=2}^{K}(p+\beta C^{j-1})\left(\|\boldsymbol{x}^k\|_1^2 - \|\boldsymbol{x}^k\|_2^2\right)$$

$$s.t. \quad \boldsymbol{x}^{j+1} \leq \boldsymbol{x}^j \leq \boldsymbol{x}^{j-1}, \qquad \forall\, j = 1, 2, \ldots, K$$

$$\boldsymbol{x}_i^{K+1} = \delta, \text{ if } i \in V_s; \text{ otherwise, } \boldsymbol{x}_i^{K+1} = 0$$



– Example: query for node-1 and node-2

$$x_1^k \geq 0.9, x_2^k \geq 0.9, \qquad \text{for } k = 1, \cdots, K$$

Outline

- Motivations ✓
- Q1: HiDDen Formulation ✓
- Q2: HiDDen Algorithm ✓
- Q3: HiDDen Generalizations ✓
- Experimental Results
- Conclusions

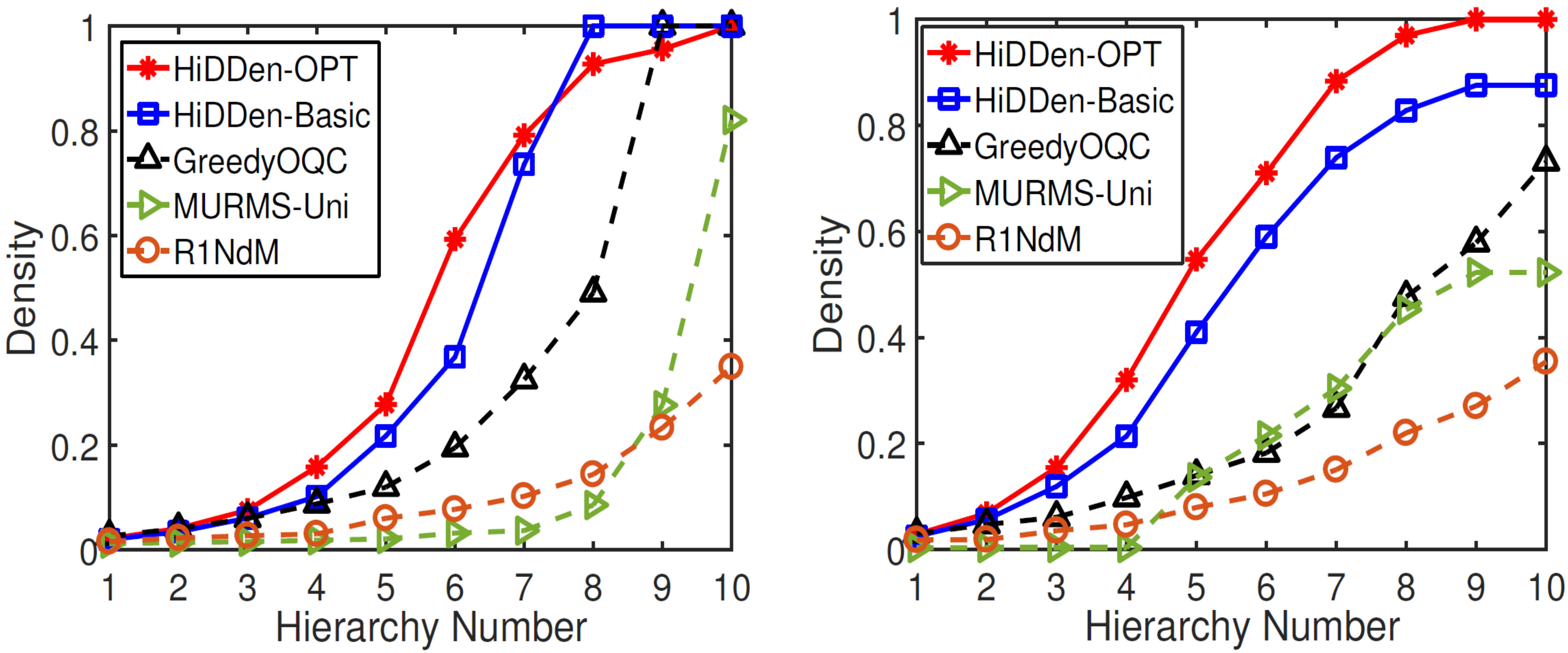
Experimental Setup

- **Datasets**:
  - DBLP co-author network (nodes: 38,624, edges: 200,332)
  - Autonomous system network (nodes: 6,474, edges: 25,142)
  - Financial network (account nodes: 29,851, PII nodes: 61,159)
  - Trafficking network (traffickers: 1416, word nodes: 4225)
- **Evaluation Objectives**:
  - Effectiveness: density of each hierarchy and density variety
  - Efficiency: running time and scalability
- **Comparison Methods:**

HiDDen (Our Methods)	Baseline Methods
☐ HiDDen-Basic (quadratic programming separately) ☐ HiDDen-OPT (alternative gradient descent)	☐ GreedyOQC [Tsourakakis'13] ☐ MURMS-Uni [Ding'08] ☐ R1NdM [Belachew'15]

DATA Lab

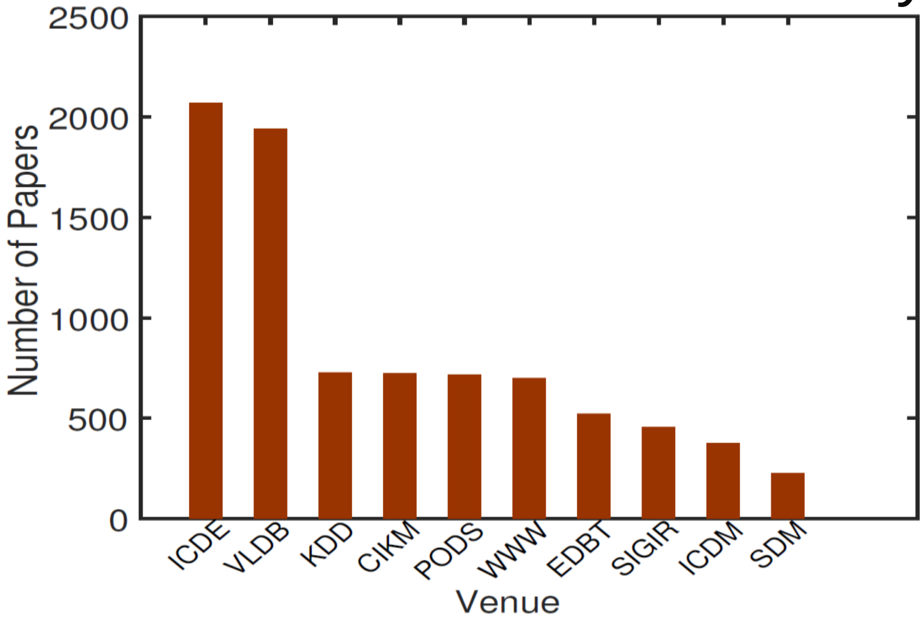
R1. Effectiveness Results – Unipartite Network



Observation: densities are higher and increase up to 1

R2. Case Study on Co-Authorship Network

- Differences between 1st and 5th hierarchy:

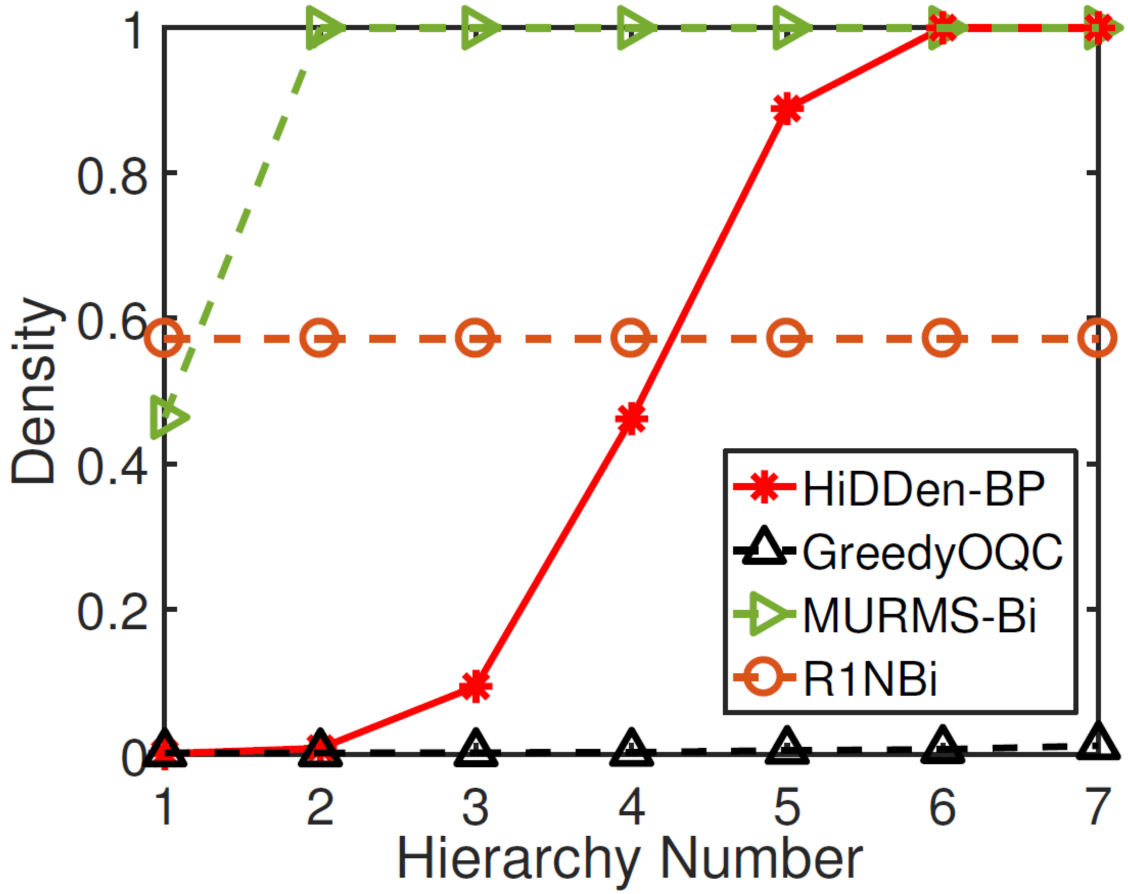


Observation: (1) difference in research area; (2) most of people in 5th hierarchy are in mid-career

- Differences between 5th and 10th hierarchy:

Observation: 10th hierarchy contain only flagship researchers

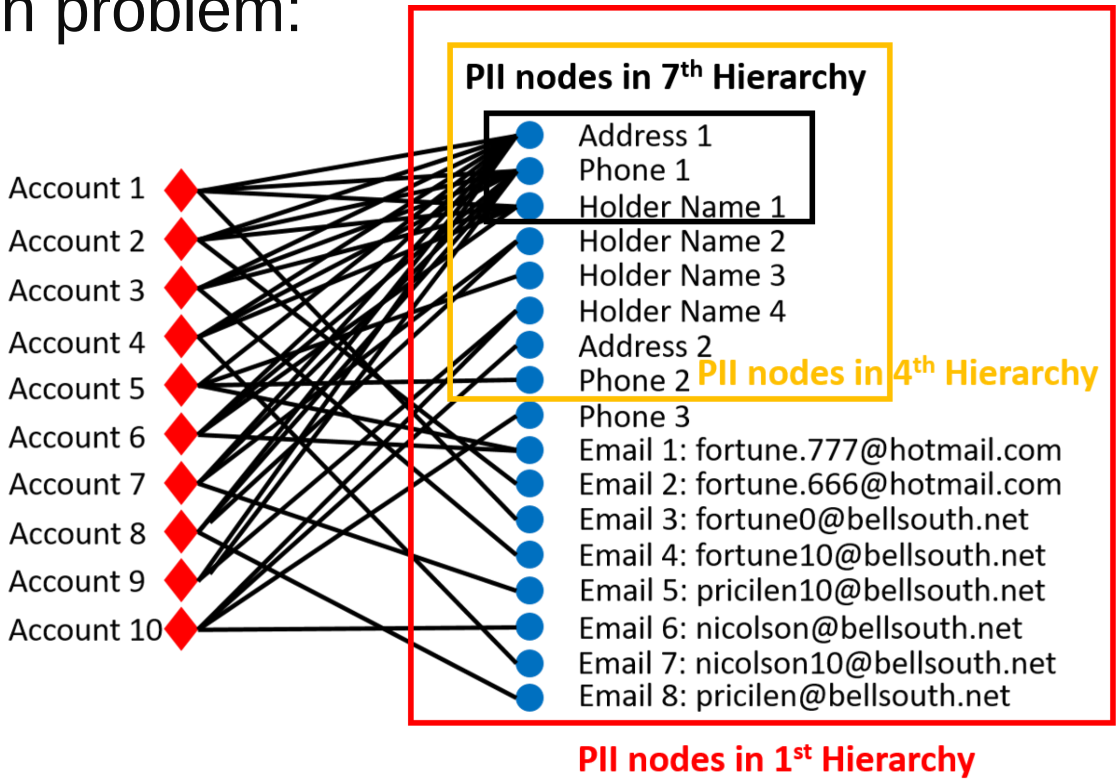
R3. Effectiveness Results – Bipartite Network



Observation: densities exhibit a good variety and are up to 1

R4. Case Study on Financial Network

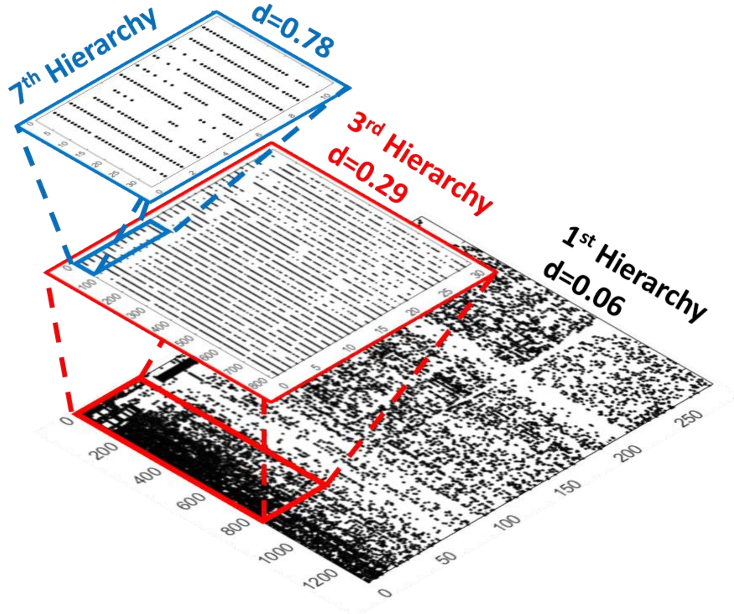
- Differences among hierarchies for synthetic identity fraud detection problem:



Observation: multiple hierarchies of dense subgraph can more accurately detect the synthetic identity fraud

R5. Case Study on Trafficking Network

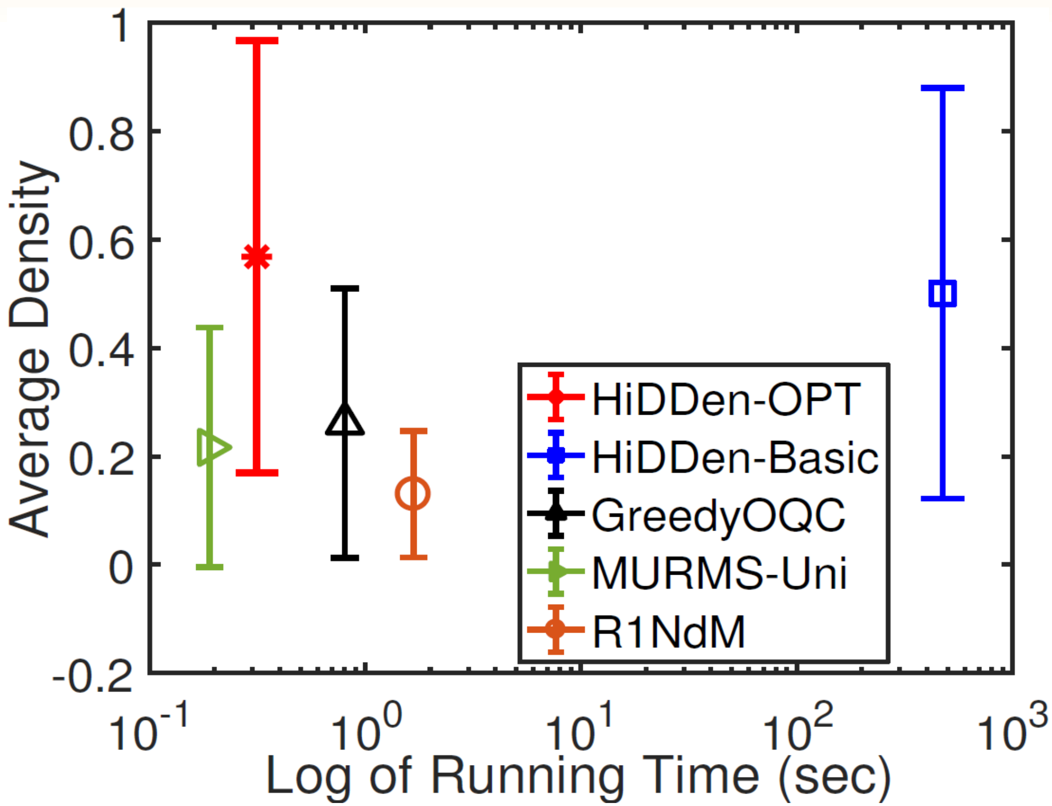
- Differences among hierarchies for trafficking problem:



7th Hierarchy	33 traffickers: some of them are from same family; 8 words
3rd Hierarchy	815 traffickers (nearly half); words (30 in total): prostitution, girls, victims, police, underage, sex, trafficked, recruited, minor, adult, drugs, arrested, money, women, hotel
1st Hierarchy	1326 traffickers and 284 words

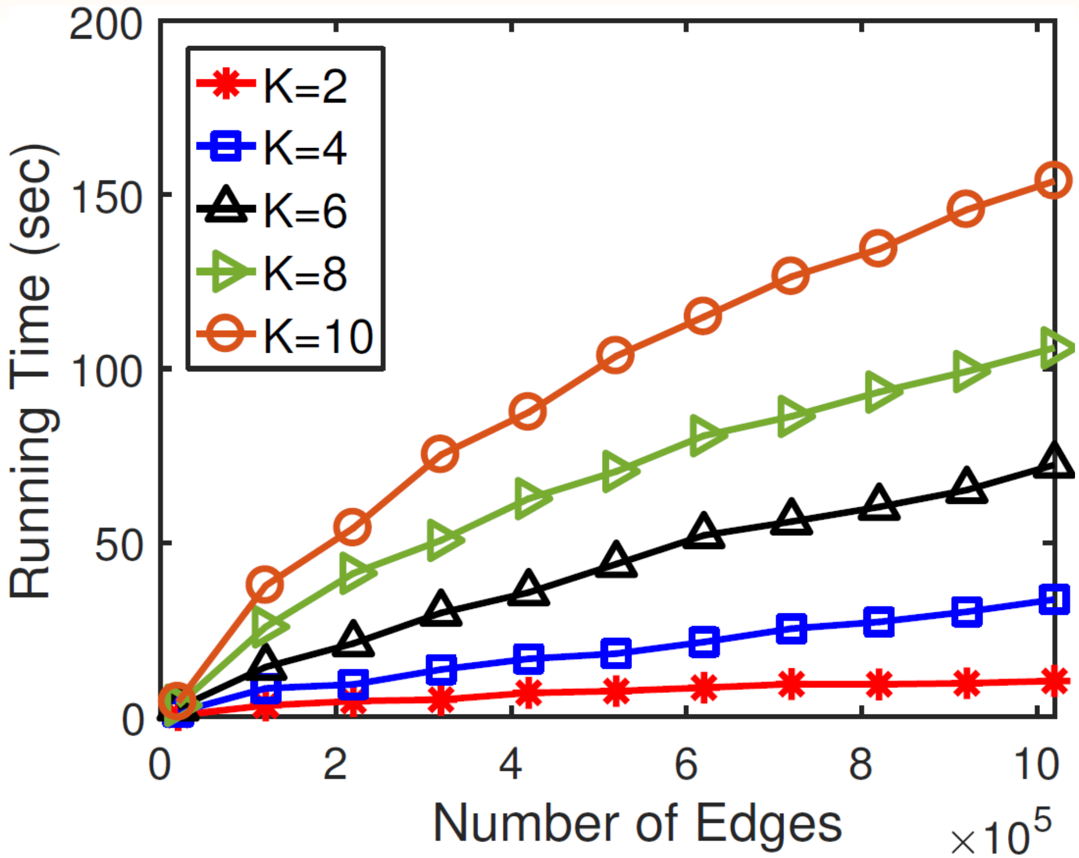
Observation: (1) most of the traffickers are **forcing the underage girls for prostitution in hotels in exchange for cash, drugs, and other items**; (2) some are from same family

R6. Quality-Speed Balance



Observation: HiDDen gains a better balance between running time and avg. density, as well as density variety

R7. Scalability of HiDDen



Observation: HiDDen has a linear time complexity w.r.t # of edges

Outline

- Motivations ✓
- Q1: HiDDen Formulation ✓
- Q2: HiDDen Algorithm ✓
- Q3: HiDDen Generalizations ✓
- Experimental Results ✓
- Conclusions

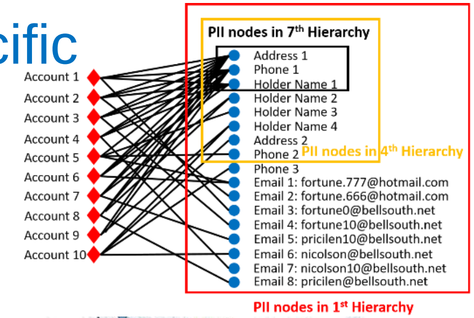
Conclusions

- Hierarchical Dense Subgraph Detection
  - **Q1:** Formulation
  - **A1:** HiDDen →

$$\min_{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_K} -(1+p)(\boldsymbol{x}^1)^T \boldsymbol{A}\boldsymbol{x}^1 + p(\|\boldsymbol{x}^1\|_1^2 - \|\boldsymbol{x}^1\|_2^2) - (1+p+\beta) \sum_{k=2}^{K} \left(\boldsymbol{x}^k\right)^T \boldsymbol{A}\boldsymbol{x}^k + \sum_{k=2}^{K} (p + \beta C^{j-1}) \left(\left\|\boldsymbol{x}^k\right\|_1^2 - \left\|\boldsymbol{x}^k\right\|_2^2\right)$$

$$s.t. \quad \boldsymbol{x}^{j+1} \leq \boldsymbol{x}^j \leq \boldsymbol{x}^{j-1}, \qquad \forall\, j = 1, 2, \ldots, K$$

  - **Q2:** Algorithm
  - **A2:** Alternative Projected Gradient Descent Method
  - **Q3:** Generalizations
  - **A3:** Algorithms for bipartite graphs & query-specific



- Results
  - HiDDen outperform other baseline methods in density and variety
  - HiDDen has a linear time complexity