Random Walk with Restart on Hypergraphs: Fast Computation and an Application to Anomaly Detection

Jaewan Chun — KAIST — jjwpalace@kaist.ac.kr

Geon Lee — KAIST — geonlee0325@kaist.ac.kr

Kijung Shin — KAIST — kijungs@kaist.ac.kr

Jinhong Jung — Songsil University — jinhong@ssu.ac.kr

ECML PKDD 2024 VILNIUS 9-13 SEPTEMBER

KAIST · Soongsil University

Summary

Motivation

- Measuring node proximity is important with many practical applications including clustering, ranking, and anomaly detection
- Random walk with restart (RWR) is a widely-used measure for node proximity in graphs
- However, for hypergraphs, fast computation of RWR has been unexplored

Proposed Algorithm: ARCHER

- We propose two computation methods for RWR on hypergraphs that are complementary (i.e., offering relative advantages on different hypergraphs)
- We propose ARCHER, which adaptively and efficiently selects a computation method
- We propose an application of RWR on hypergraphs to anomaly detection

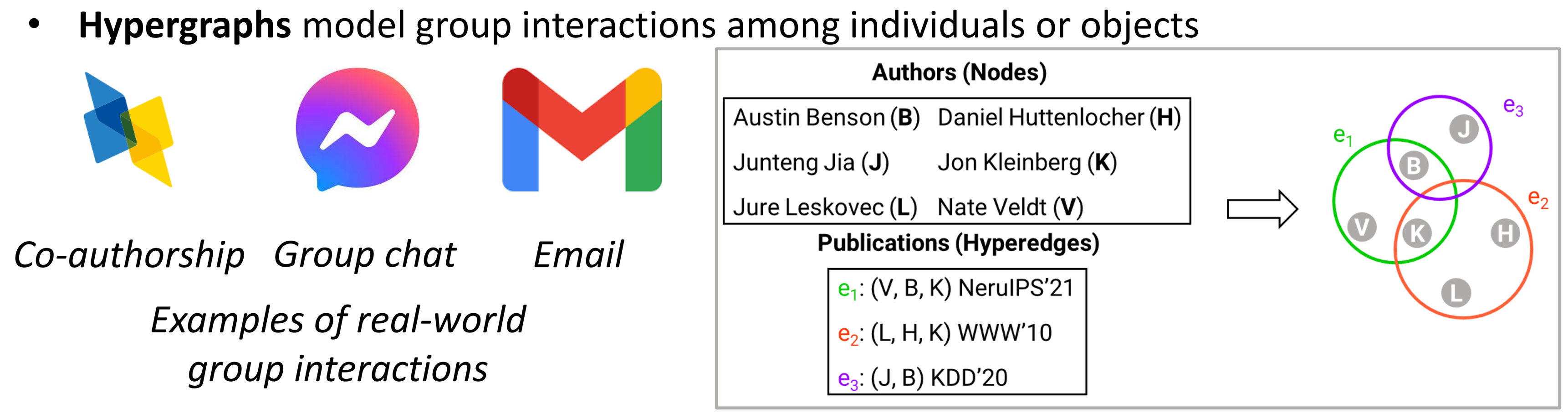
Contribution

- **Efficient**: ARCHER is fast and space-efficient
- **Complementary Computation Algorithms**: Two computation methods are complementary depending on datasets
- **Automatic Selection**: ARCHER accurately selects between two computation methods
- **Anomaly Detection**: RWR on hypergraphs are useful for anomaly detection tasks

Introduction

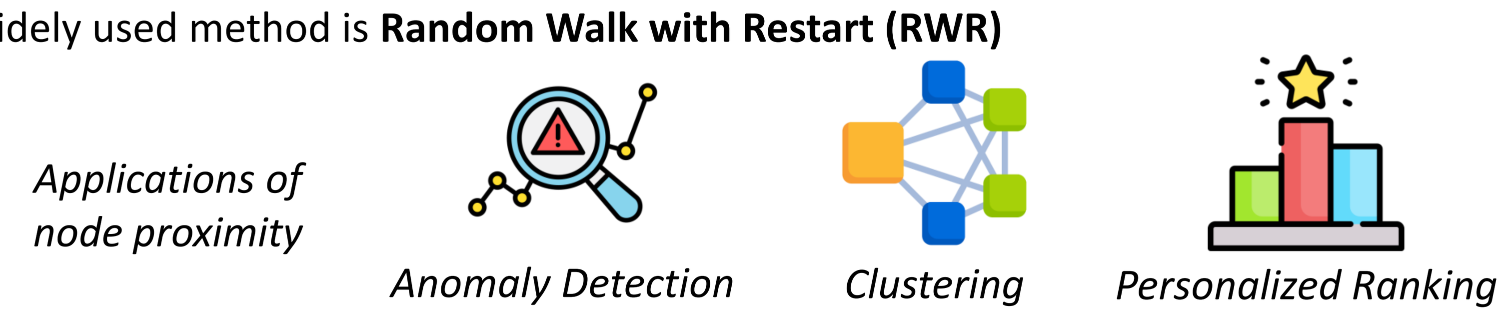
Hypergraph

- **Hypergraphs** model group interactions among individuals or objects



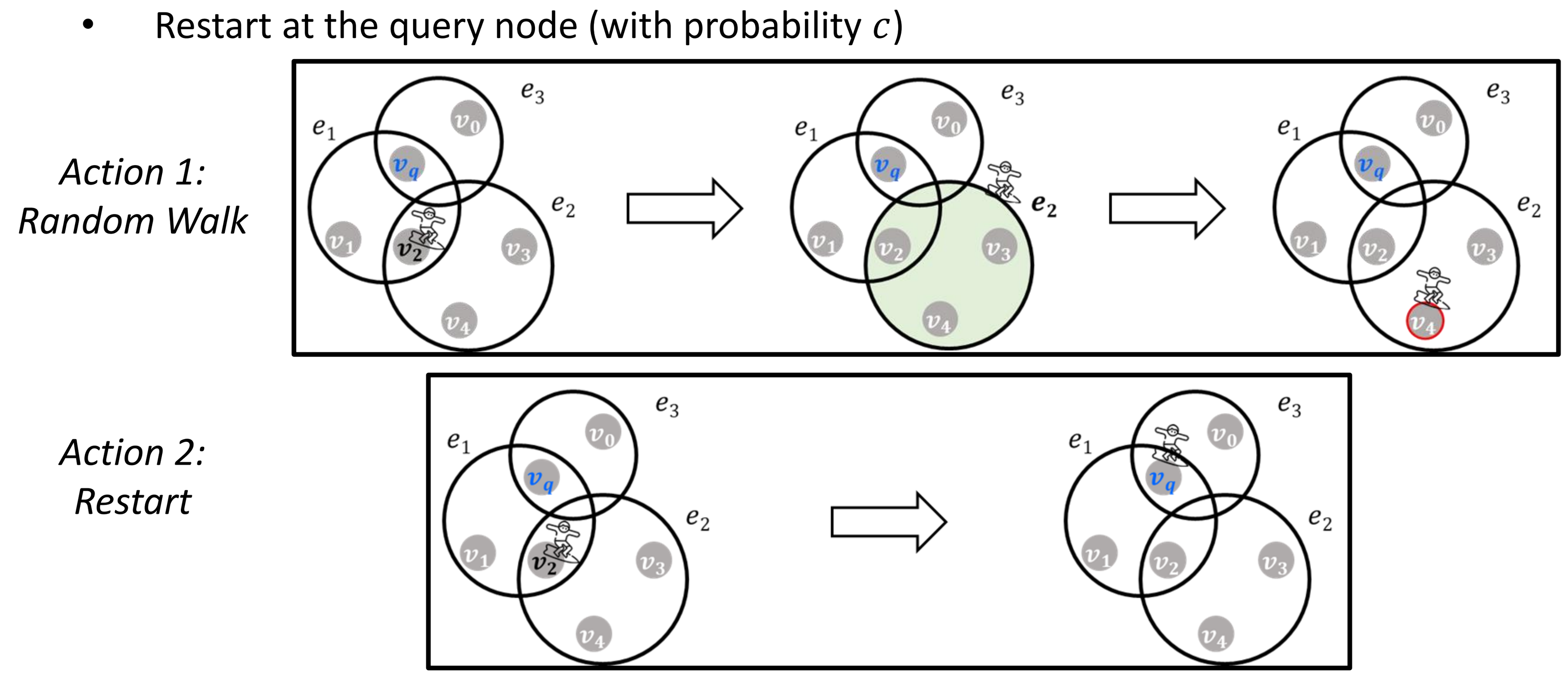
Measuring Proximity between Nodes

- Measuring proximity between nodes on hypergraphs has many practical applications
- A widely used method is **Random Walk with Restart (RWR)**



Random Walk with Restart (RWR) on Hypergraph

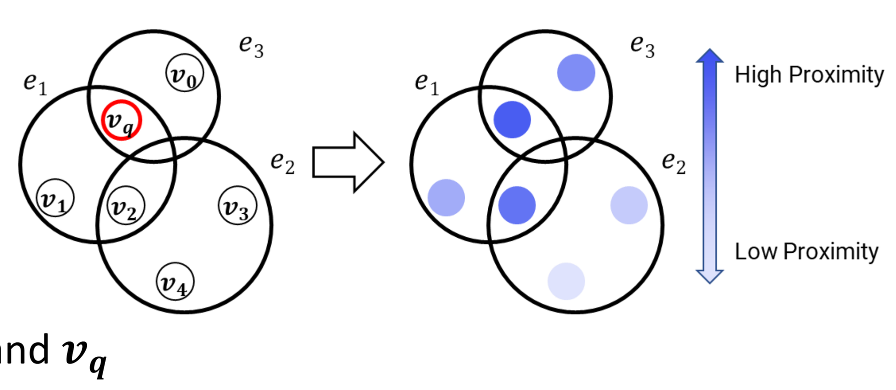
- Stationary probability of a "random surfer" over the nodes
- Random walk (with probability $1-c$)
  1. Select a hyperedge e containing current node
  2. Move to a node v selected from hyperedge e
- Restart at the query node (with probability c)



Proposed Algorithm: ARCHER

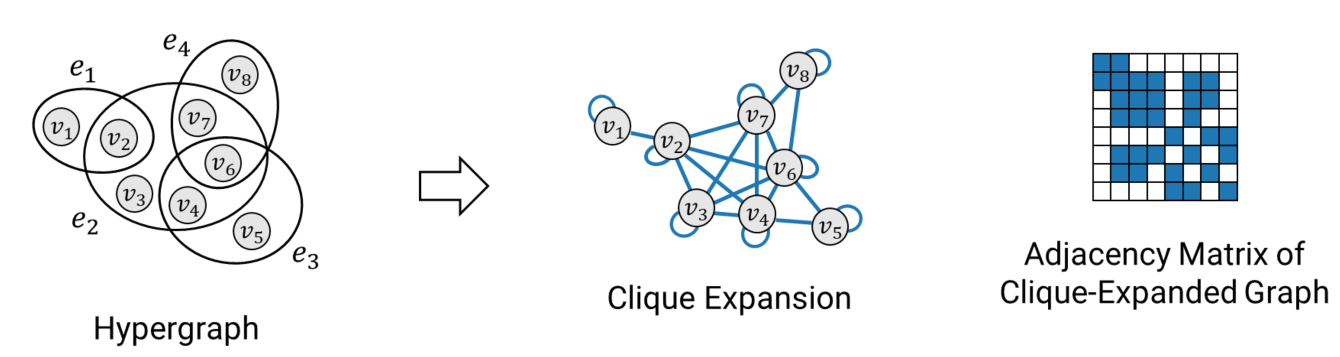
Formal Task Description

- **Given**
  - Hypergraph $G_H = (V, E)$
    - node set V and hyperedge set E
  - Query node v_q
  - Restart probability c
- **Output:** RWR scores between each node and v_q



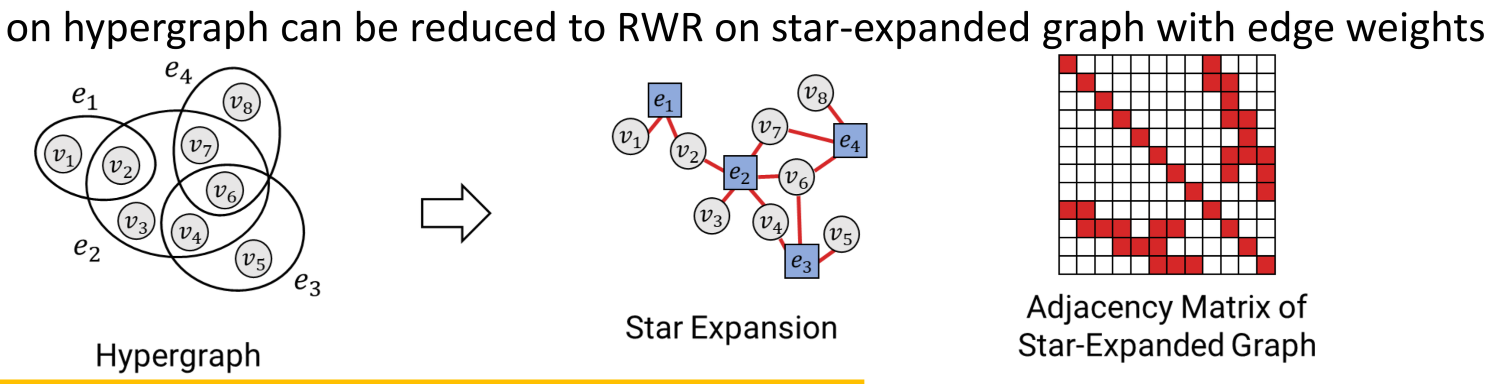
Component 1: Clique-expansion-based Method

- **Clique-expansion:** a graph constructed from hypergraph by replacing each original hyperedge with a clique
- RWR on hypergraph can be reduced to RWR on clique-expanded graph with edge weights



Component 2: Star-expansion-based Method

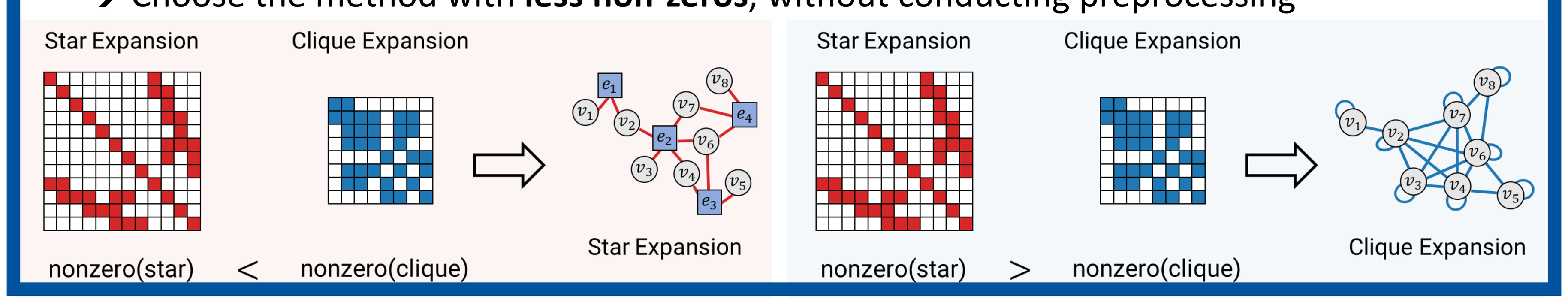
- **Star-expansion:** a graph constructed from hypergraph by
- (1) aggregating nodes and hyperedges into new set of nodes
- (2) adding edges between each pair of incident node and hyperedge
- RWR on hypergraph can be reduced to RWR on star-expanded graph with edge weights



Component 3: Automatic Selection Method

- For RWR computation, we preprocess the expanded graph into parameters
- Cost of preprocessing vary based on dataset
  - Case 1: star-expansion is efficient (up to 137.6x less time, 16.2x less space)
  - Case 2: clique-expansion is efficient (up to 6.4x less time, 9.6x less space)
- Hint: empirically cost of RWR depends on **non-zeros of adjacency matrix**

→ Choose the method with **less non-zeros**, without conducting preprocessing



Application: Anomaly Detection

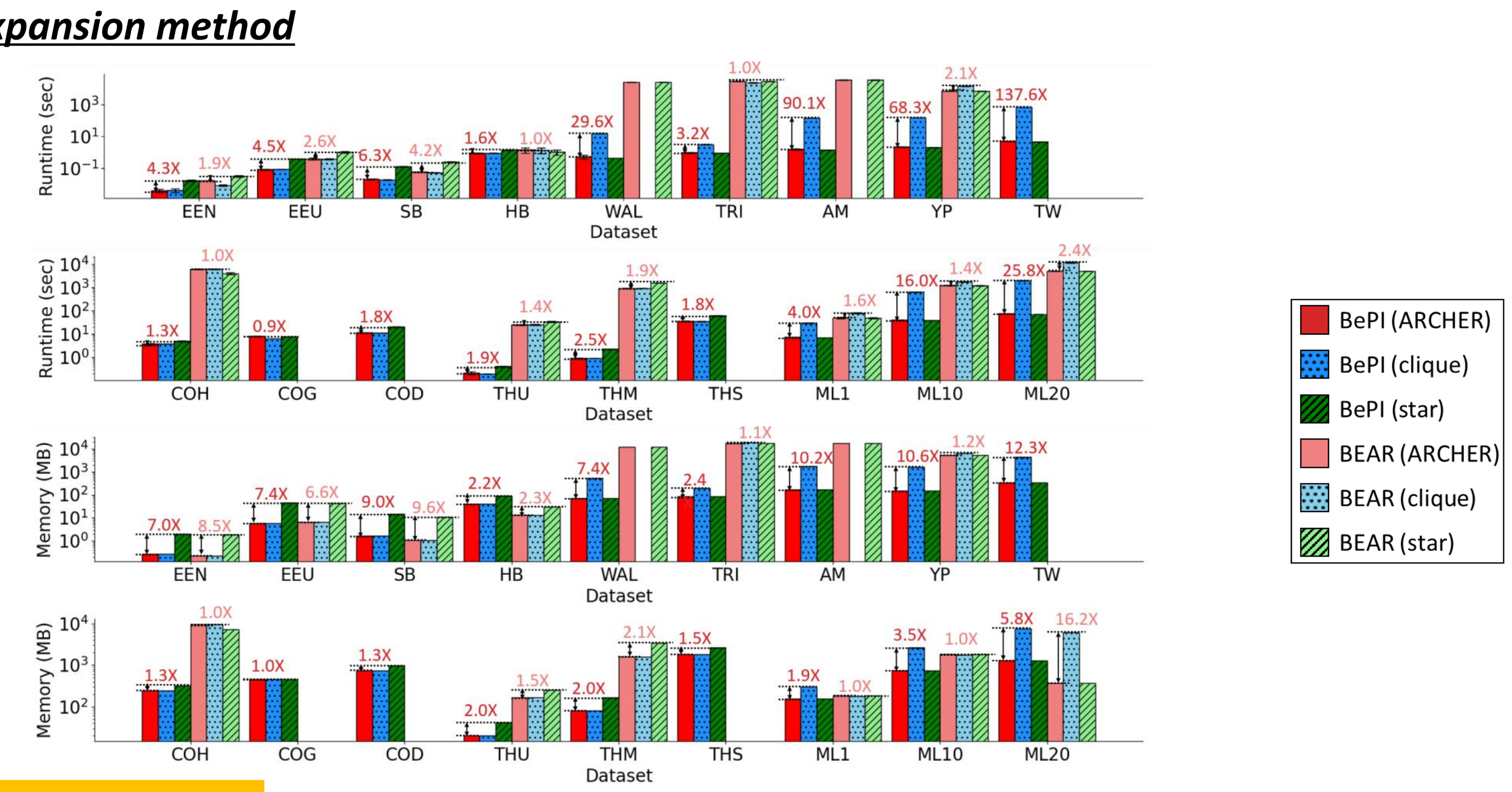
Anomaly Detection on Hypergraph

- **Task**: Given a hypergraph, detect anomalous hyperedges
- **Proposed Method**: Define the normality score of a hyperedge as the average of RWR scores between all pairs of nodes within the hyperedge
- Normality score $ns(e) = \frac{1}{|e|(|e|-1)} \sum_{u \in e} \sum_{v \in e \setminus \{u\}} r_{u \to v}$
- **ARCHER** accelerates the computation of the normality score

Experimental Results

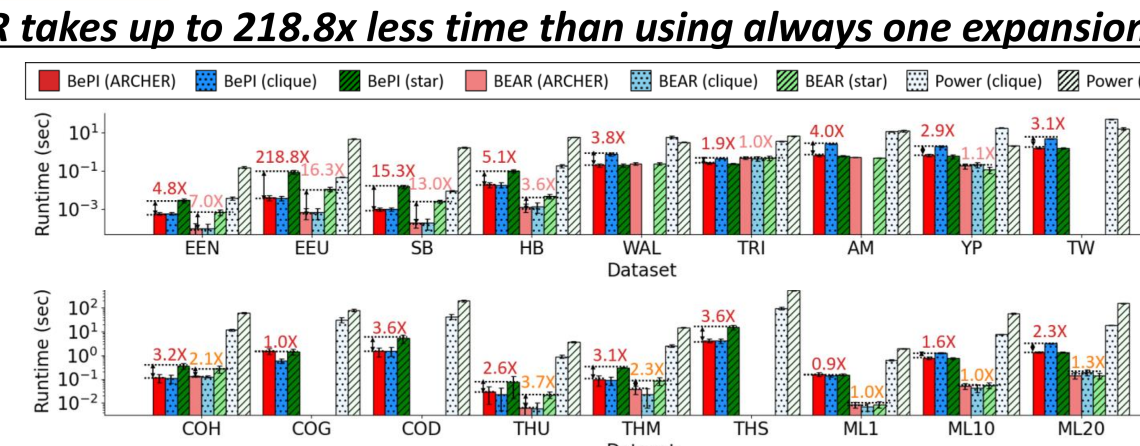
Q1) Preprocessing Cost

- ***ARCHER takes up to 137.6x less time, and 16.2x less space than using always one expansion method***



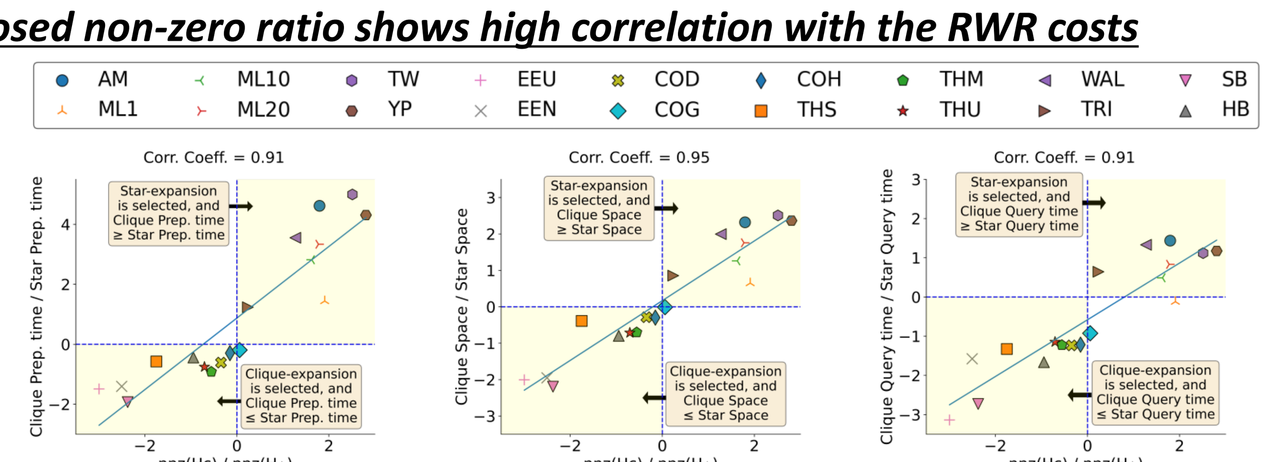
Q2) Query Time

- ***ARCHER takes up to 218.8x less time than using always one expansion method***



Q3) Automatic Selection Method

- ***Proposed non-zero ratio shows high correlation with the RWR costs***



Q4) Application to Anomaly Detection

- ***ARCHER outperforms baselines in 3 real-world datasets***