Graph Analysis: Decomposition-Based Analysis Using Multilayer Networks

Sharma Chakravarthy

Information **T**echnology **Lab**oratory (IT Lab)

Computer Science and Engineering Department

The University of Texas at Arlington, Arlington, TX

Email: sharma@cse.uta.edu

URL: http://itlab.uta.edu/sharma

Presentation Outline

- ***What is big data analytics/science (BDA)***
- ***What we are doing at IT Lab towards BDA***
- ***Why Graph modeling and Analysis is important***
- ***Modeling and Analyzing complex data sets using Multilayer Networks***
    - ***How to decompose***
    - ***How to compute loss-less community and hubs***
- ***conclusions***

What is Big Data Analytics/Science?

- *It is not a single approach or a single solution to a single problem*
- *Different problems require different approaches and analysis techniques*

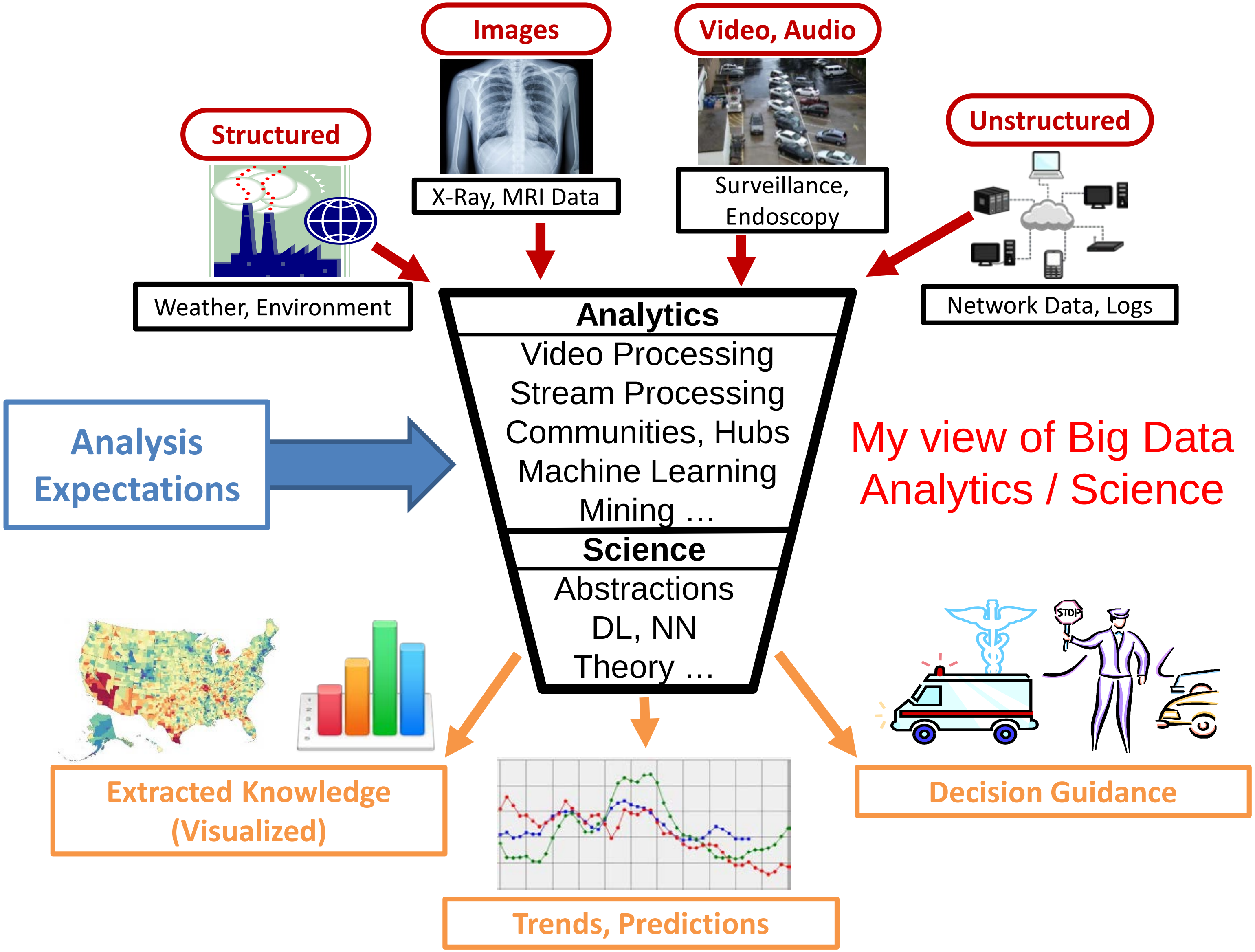
- *Hence, in my view, using/extending current approaches and developing new ones (a suite of approaches) for modeling and analyzing a given data set for a given analysis requirements*

Given: Complex data set + Analysis requirements

Determine/develop: Modeling and Analysis methods/approaches

What is Big Data

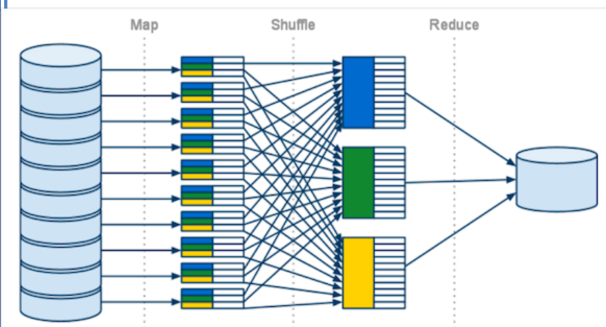
- Data set pertaining to the 4Vs, i.e. **Volume, Velocity, Variety** and **Veracity**
- **Big data analytics for analyzing given data sets**
- **What we want to do today**
- **Path up to this point or how we have arrived here!**

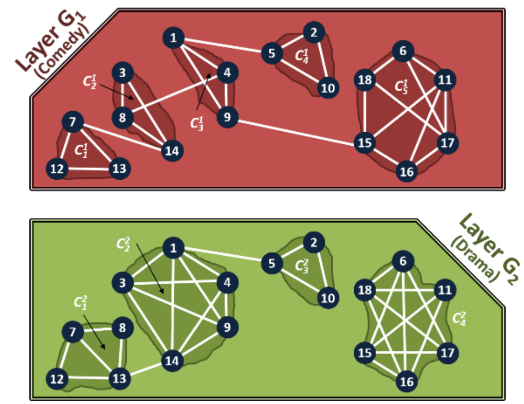


Transforming **Disparate Data** into **Knowledge and Decisions**

Sharma Chakravarthy
Professor
Information Technology Lab (IT Lab)
Ph.D. (University of Maryland, College Park, 1985)

http://itlab.uta.edu





Scalablity using Map/Reduce

Multilayer Network Analysis

Video Situation Analysis

Social Network Analysis

INFORMATION TECHNOLOGY LABORATORY @ UTA-CSE

http://itlab.uta.edu

ERB 632
sharma@cse.uta.edu

Information Technology Laboratory (ERB 514)

Prof. Sharma Chakravarthy (ERB 632)

Email: sharma@cse.uta.edu, *URL:* http://itlab.uta.edu/sharma

Funding Sources: NSF, Spawar, AFRL, Rome Lab, ONR, DARPA, TI, MCC

Select Projects

- ✓ **Multilayer Network Analysis**
- ✓ **Graph Mining scalability using Map/Reduce**
- ✓ **MavVStream: (video situation analysis Processing)**
- ✓ **Expertise identification in Q/A community**
- ✓ **Ranking in web databases**
- ✓ **WebVigiL:** (Change Monitoring for the web)
- ✓ **Mining:** Graph, Text, Assoc Rules
- ✓ **Information Search, Filtering, and classification**

Select Publications

1. Das, S., Chakravarthy, S. (2018). Duplicate Reduction in Graph Mining: Approaches, Analysis, and Evaluation. IEEE Transactions on Knowledge and Data Engineering
2. Santra, A., Bhowmick, S., Chakravarthy, S. (2017). Efficient Community Re-creation in Multilayer Networks Using Boolean Operations. *International Conference on Computational Science, ICCS 2017*
3. Bhatnagar, V., Kaur, S., Chakravarthy, S. (2014). Clustering data streams using grid-based synopsis. Knowledge and Information Systems
4. Telang, A., Chakravarthy, S., Li, C. (2013). Personalized ranking in web databases: establishing and utilizing an appropriate workload. Distributed and Parallel Databases
5. A. Telang, C. Li, and S. Chakravarthy, One size Does Not Fit All: User- and Similarity-based Ranking in Web Databases, in TKDE, April 2012
6. A. Venkatachalam, M. Aery, S. Chakravarthy, and A. Telang, m-InfoSift: Multi folder email classification based on Graph Mining, ICDM 2010, Sydney, Australia
7. S. Padmanabhan and S. Chakravarthy, HDB-Subdue: A Scalable Approach to Graph Mining, DAWAK 2009
8. Sharma Chakravarthy and Qingchun Jiang, Stream Data Processing: A Quality of Service Perspective, 2009, Book, by Springer Verlag.
9. M. Aery, S. Chakravarthy: **eMailSift: Email Classification Based on Structure and Content** in IEEE ICDM 2005

PhD Students –

Mr. Abhiskek Santra
Mr. Aleksander

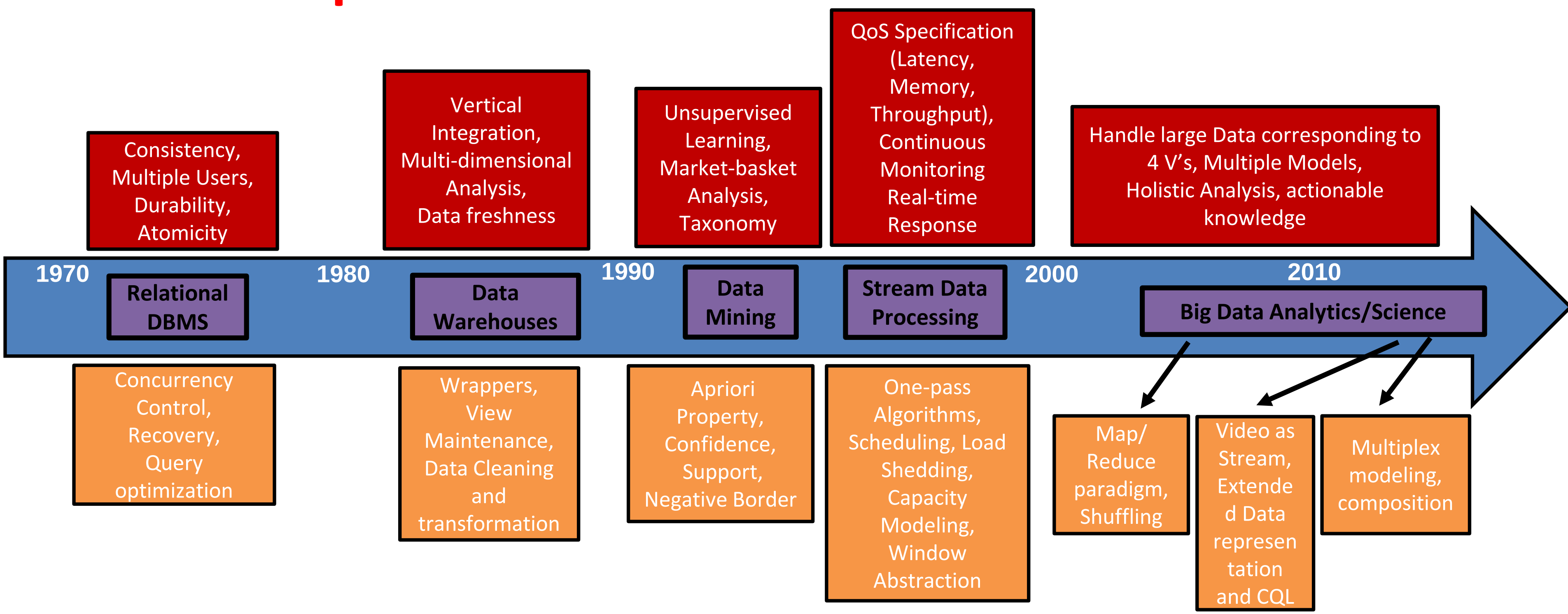
MS Thesis

Ms. Kanthi Komar
Mr. Jay Bodra
Mr. Mayur Arora

ALWAYS LOOKING FOR GOOD UNDERGRAD, MS, AND PHD STUDENTS

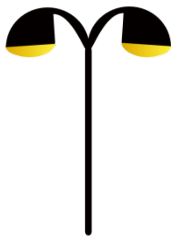
Where are we headed?

- **Without understanding the past, it is very difficult to appreciate the present and plan for the future!**
- **Technology provides solutions; it does mean it solves problems!**



Consider Data Set 1

- **UK accident data set: consists of accidents and a number of attributes associated with each accident**



Light Conditions

Weather Conditions

Road Surface Conditions

- Analysis requirements
  - **Accident Prone Regions? Based on weather conditions**
  - **Most dominant weather feature? cause of most accidents**
  - **Given budget, what aspect should be addressed in which region?**

- **Modeling:** Data consists of Multiple relationships among **same type**(s) **of entities**

Consider Data Set 2

- **DBLP data set: consists of** collaboration, research domains, conferences, cities

dblp

WikiCFP
A Wiki for Calls For Papers

Collaboration | **Direct Flights** | **Research Domains** | **Attendance** | **Residences** | **Conf Venues**

- Analysis requirements:
  - **Best city to hold a workshop?**
  - **Which group of co-authors publish more in which group of related conferences?**

- **Modeling:**
  - Multiple relationships existing among **different type(s) of entities. Also,** *Connectivity among scientists, cities and conferences*

Other similar Data Sets (1)

- **Data Characteristic**: Multiple relationships existing among **same type**(s) **of entities**

Interaction among a set of people

facebook Linked in twitter

Analysis:

- **Most popular or socially active group of people?**
  - **in twitter, LinkedIn; in facebook, twitter?**
- **Most influential set of people?**

Other similar data sets (2)

- **Data Characteristic**: Multiple relationships existing among **same type**(s) **of entities**

Airline connectivity among a set of US cities

Airline connectivity among a set of Indian cities

- **Highly central cities (hubs)?**
- **Next promising city to establish a hub?**

Multi Type, Multi Feature Data Analysis

- Challenges
  - **Modeling**
  - **Flexible Analysis**
  - **Computation Efficiency, and**
  - **Scalability**

Modeling: Traditional Approach

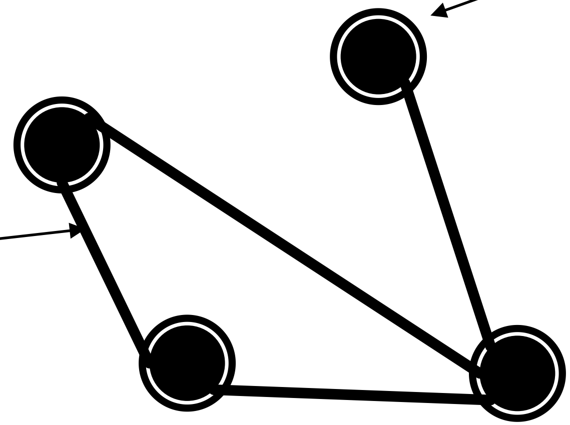
➢ Single Edge Monoplex (single, simple graph)

Entities as **Nodes**

Relationships as **Edges**

- **Weights** for Strength
- **Direction** for Information Flow

A single feature of the accident data set can be modeled



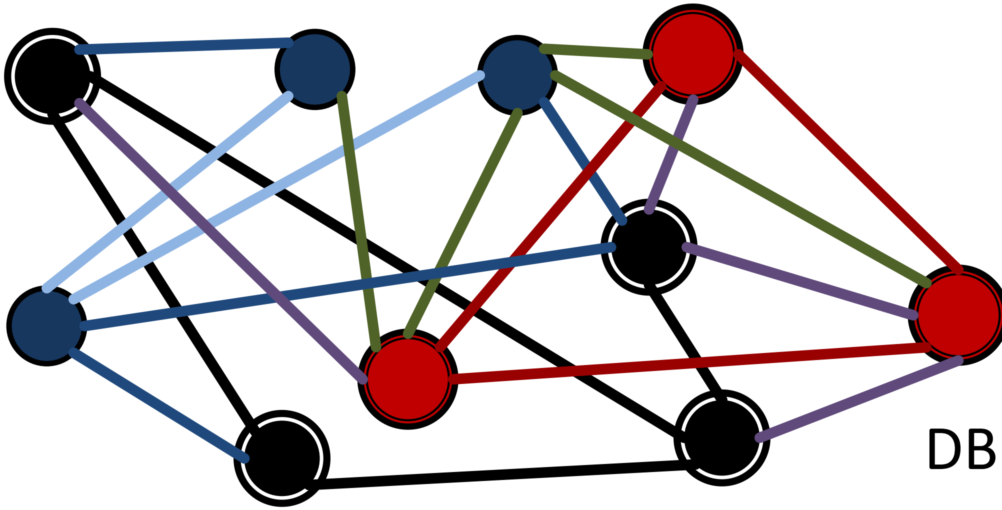
➢ Drawbacks

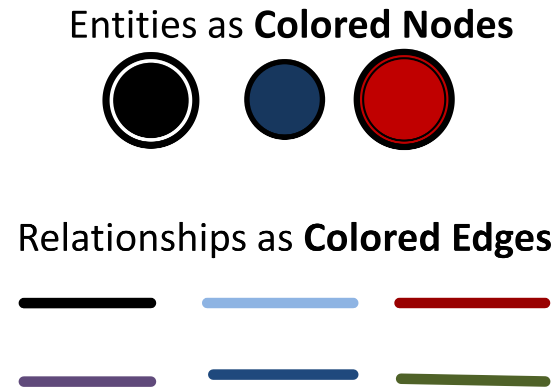
- Combining features is not straightforward
- Every **feature combination need to be analyzed separately**

➔ Difficult/cannot reuse computations

Modeling: Traditional Approach using attributed Graph

- Multi Edge/node type Monoplex





DBLP data set can be modeled this way

- Drawbacks
  - analysis wrt different feature combinations is difficult
  - Need to extract subgraphs for feature combinations
  - Convoluted Representation (difficult to understand)

Our Approach: Multilayer Networks (MLNs)

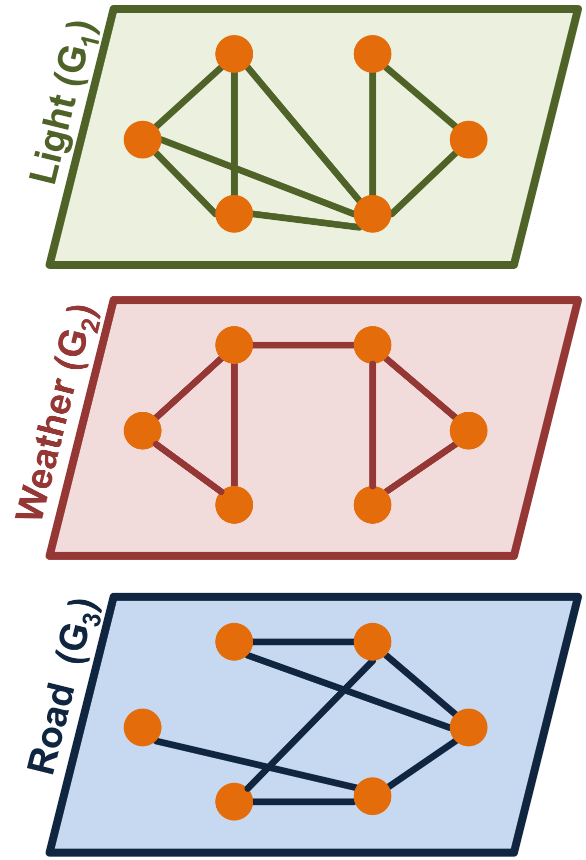
- Modeling
  - Use **Multiplexes or Multilayer Networks**
    - A ***network of networks or layers of networks***
    - Each layer/network represents a **single perspective or feature**
- **Computation challenges:**
  - **Can process individual layers**
  - **Develop techniques for composing partial results from each layer**
    - **With Loss less or high accuracy**
  - **Flexibility for analysis**
- MLNs Differentiated into **3 types**

MLN Modeling
(*Same Entities, Different Relationships*)

Homogeneous Multiplex

Multiple relationships among **same type of entities**

- **Similarity of disasters** (accidents, storms etc.) based on factors
- **Interaction among people** via various media (social media, calls etc.)
- **Connectivity among cities** based on different airlines



Accident Multiplex

MLN Modeling

(*Different Entities, Different Relationships)*

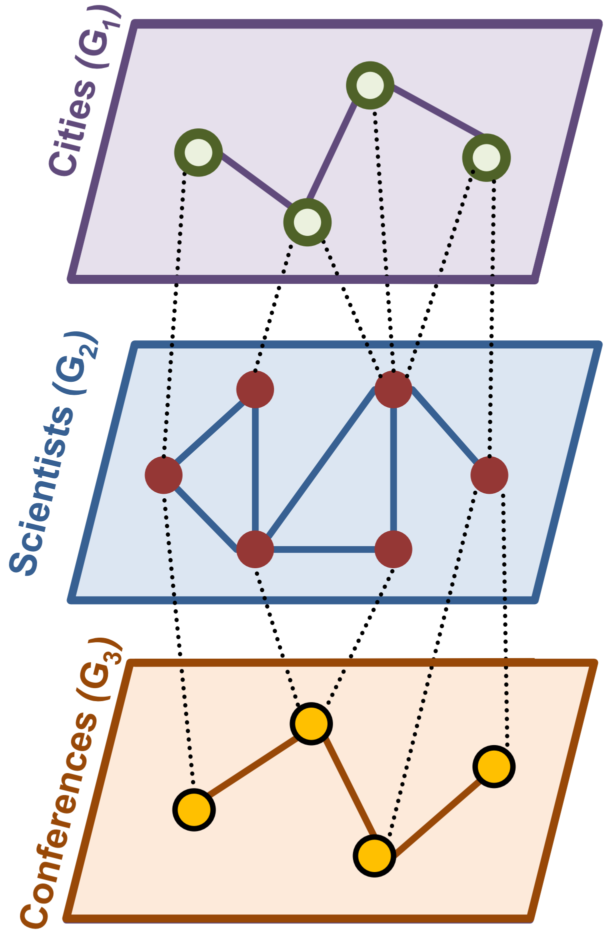
Heterogeneous Multiplex

Multiple relationships among **different types of entities**

- Residence, venue and attendance connectivity among ***city airline, scientist collaboration*** and ***similar conference*** networks

Hybrid Multiplex

- **Combination of the above two**



City-Scientist-Conference Multiplex

Benefits of Using MLNs

- **Flexible analysis**
  - **analyzing each layer or combinations**
    - **Homogeneous (Boolean, Linear etc.)**
    - **Heterogeneous (Projection, Type Independent, bipartite flow-based etc.)**
- **Parallel processing** can be leveraged using existing algorithms
- Ease of **handling the dataset incrementally**
  - **Addition** of new entities (nodes), relationship with existing entities (edges) and features/perspectives (layers)
- Amenable to decoupled approach
  - **Layers can be composed for arbitrary combination**

Computations using Multiplexes

- **Multiplex-based analysis is at a nascent stage**
  - Layers are either considered **individually or all layers are aggregated together**, in specific **sub-disciplines**
  - **Hardly any work on mining and querying multiplexes**
- Existing algorithms for a monoplex can be leveraged
  - Need to **generate, store and analyze each layer combination**
  - **N individual layers $\Rightarrow$ O(2^N) layer combinations!**
  - **Multiplexes have potential to reduce it to linear (or O(N)) complexity**
  - **Scalable**

Homogeneous Multiplex Computations

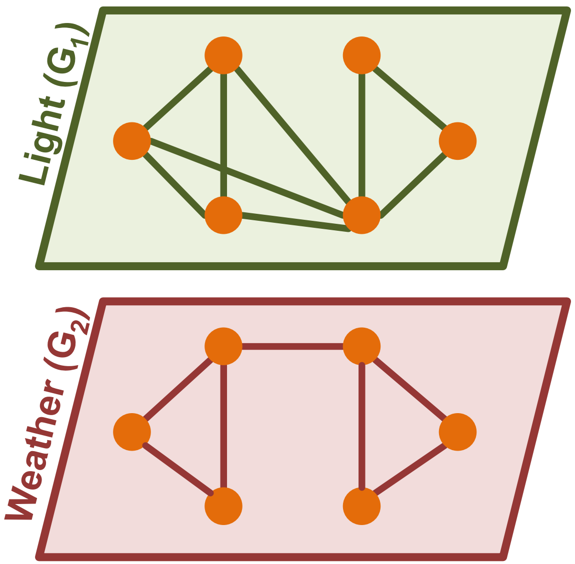
Multiplex Analysis

- **currently, we are considering:**

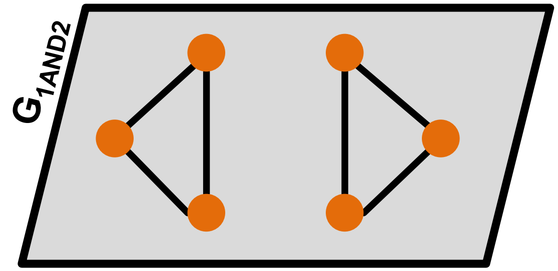
- **Communities:** Tightly connected group of nodes
  - **Effectiveness of accident prevention techniques**
    - Variation of **accident prone regions** over time

- **Hubs:** Highly central nodes
  - **Maximize the reach of an advertisement**
    - Most **influential people** across social media
  - **Identifying hubs of airlines**

Proposed Homogeneous Multiplex Layer Compositions through **Boolean Operations – AND, OR, NOT**

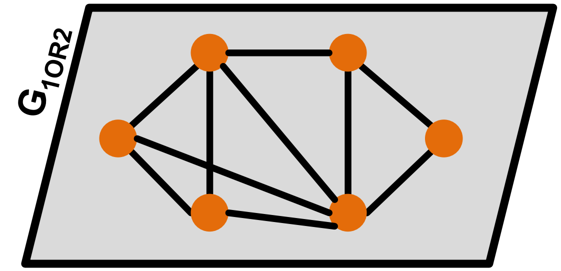


AND Composition



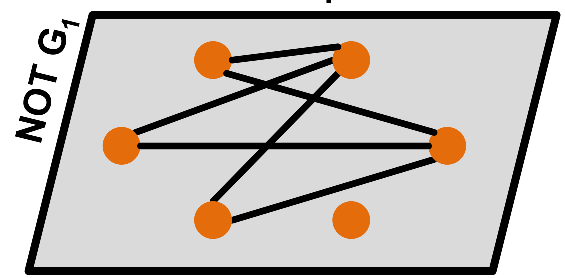
Relationships present in *all* layers

OR Composition



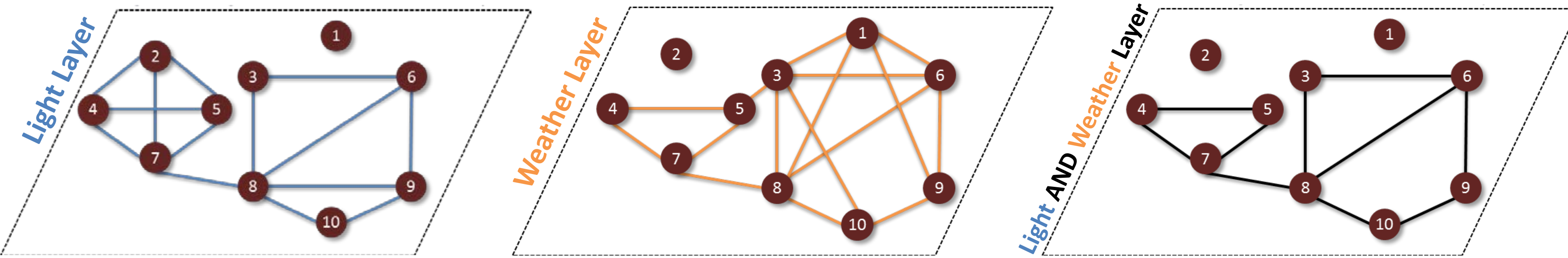
Relationships present in *at least one* layer

NOT Composition



Relationships *not* present in a layer

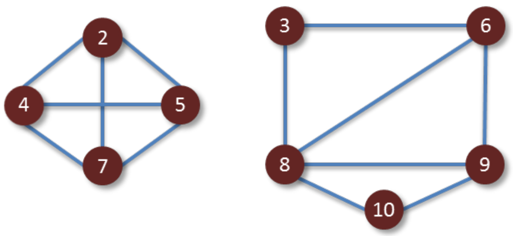
Actual Communities



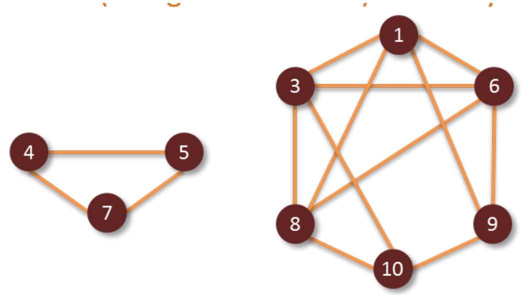
N Individual Layers and their Communities

Generate additional $O(2^N)$ AND Layer Compositions and their Communities

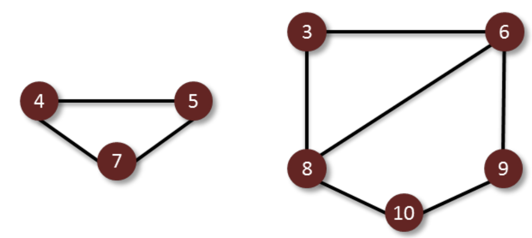
Communities (Light Layer)



Communities (Weather Layer)

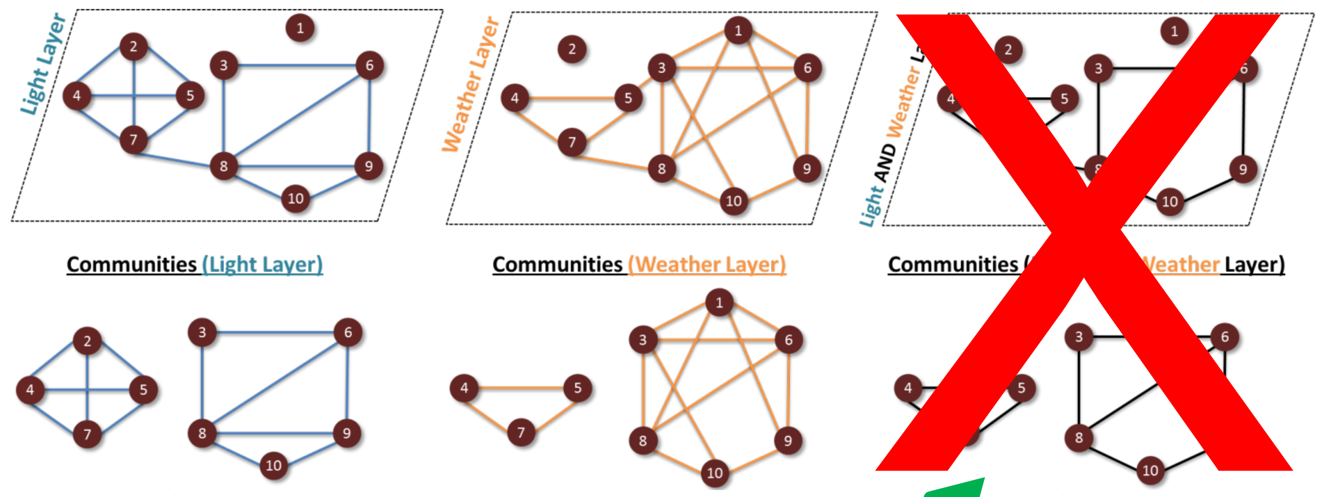


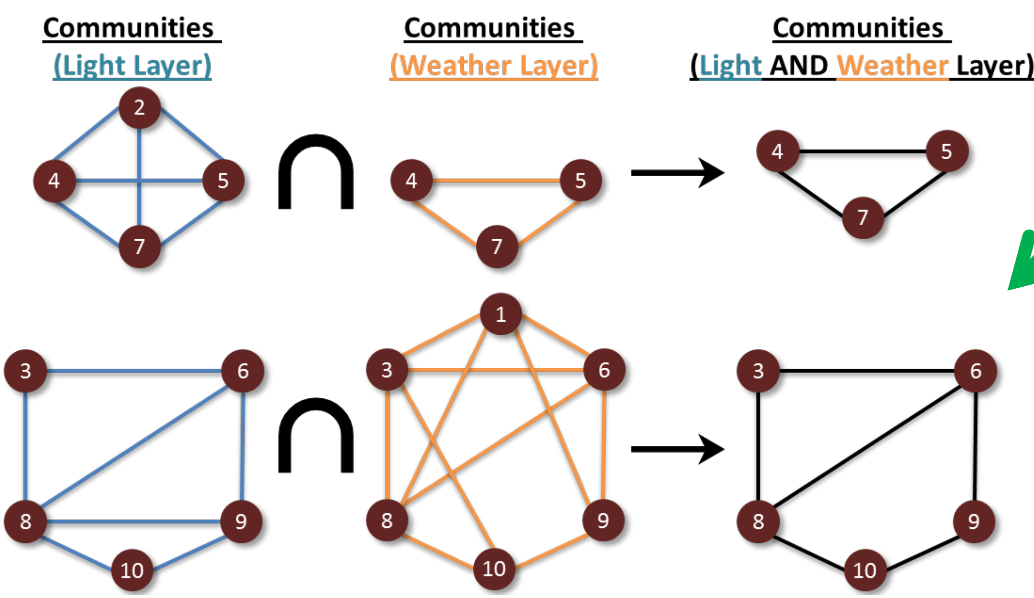
Communities (Light AND Weather Layer)



EXPENSIVE !

Aggregation Rule for Communities





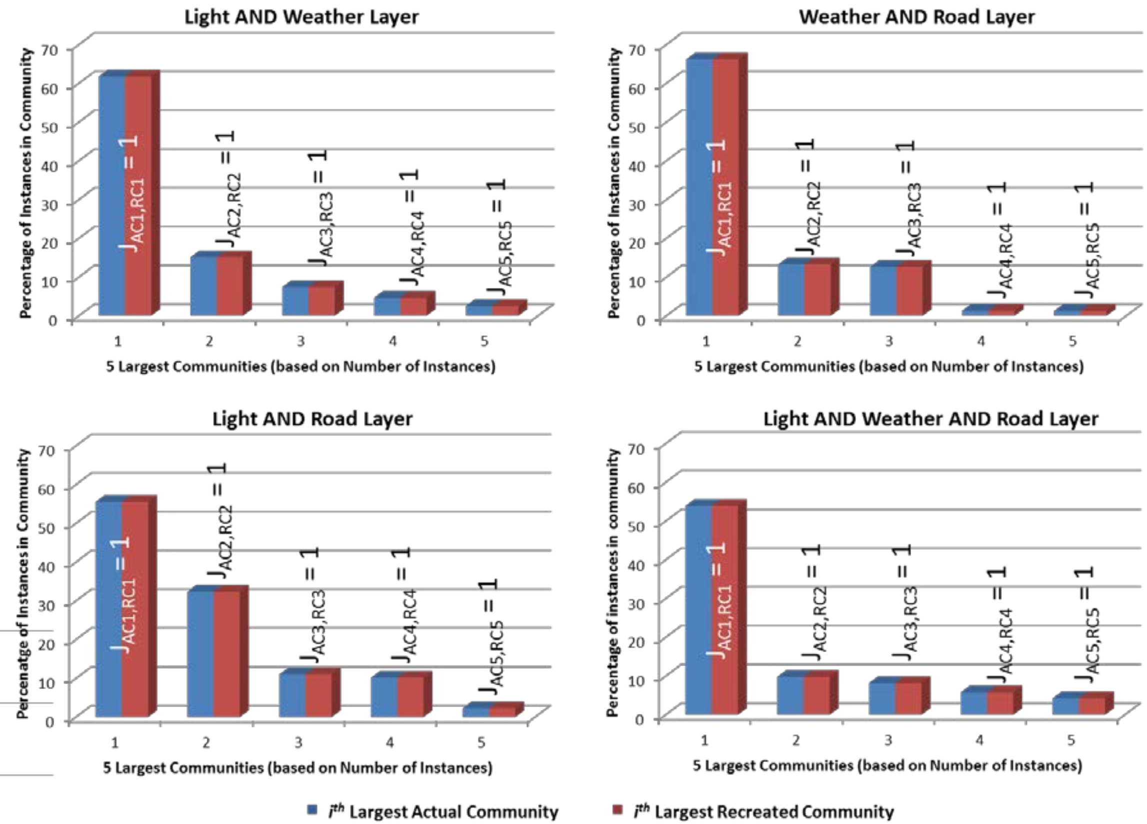
Aggregation Rule*

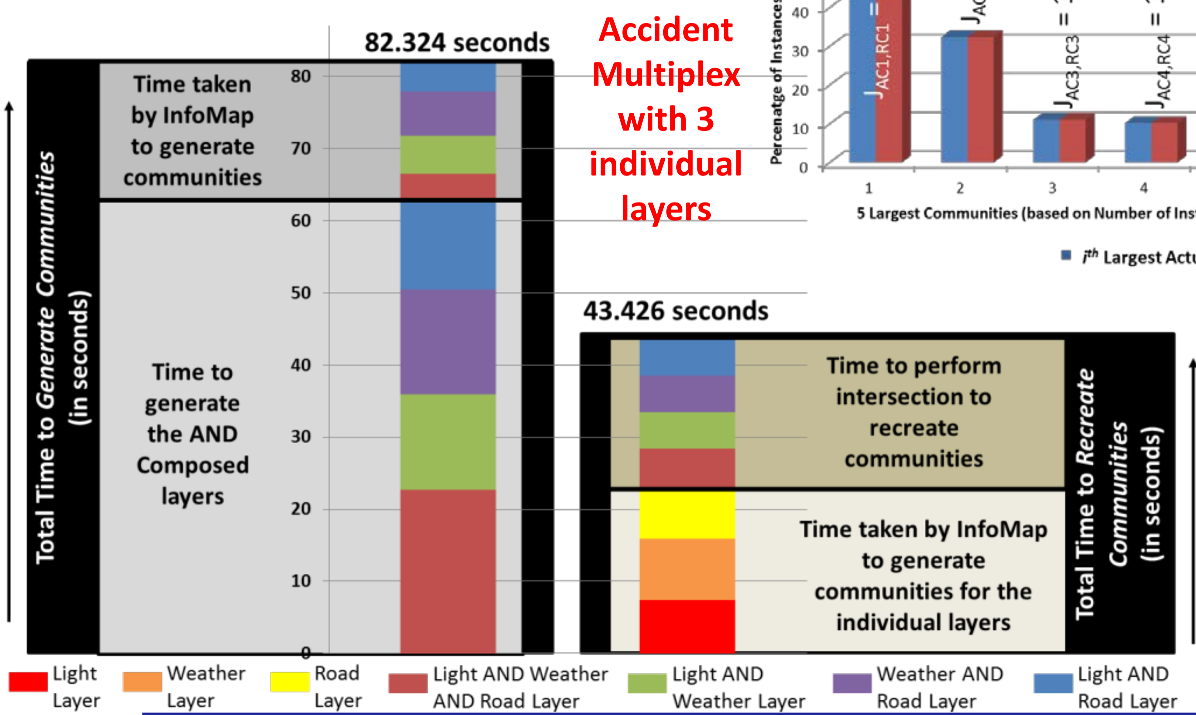
Community(L_1 AND L_2) =

Community (L_1) $\cap$ Community (L_2)

*with individual layers having self-preserving communities

Proposed an accurate node **intersection-based community re-creation technique for *any AND-composed multiplex layer* using layer-wise communities***





Reduced the overall computation time by over 40% *(with real-life multi-feature datasets – traffic accidents, storms)*

*with individual layers having self-preserving communities

Non self-preserving communities

- We have extended this work to non self-preserving communities
  - Need to take edges into account as well in addition to vertices
    - Takes more computation
  - Accuracy of 90% is achievable
  - Simple test for checking self-preserving property
  - Approach can be chosen based on desired accuracy
  - Also extended to OR and NOT combinations

Homogeneous Multiplex Computations

Computing Hubs using decomposition

Hub based insights into a dataset

- Most **influential people** across different communication platforms (Advertisement Agencies)
- Most **dominating accident locations** w.r.t poor lighting conditions and bad roads (Accident Prevention Measures)
- Highly **popular/preferred co-actors** for various genre combinations (Casting and Production Houses)

Solution: Generate **highly central vertices (i.e., hubs)** in the required individual or AND-composed multiplex layer

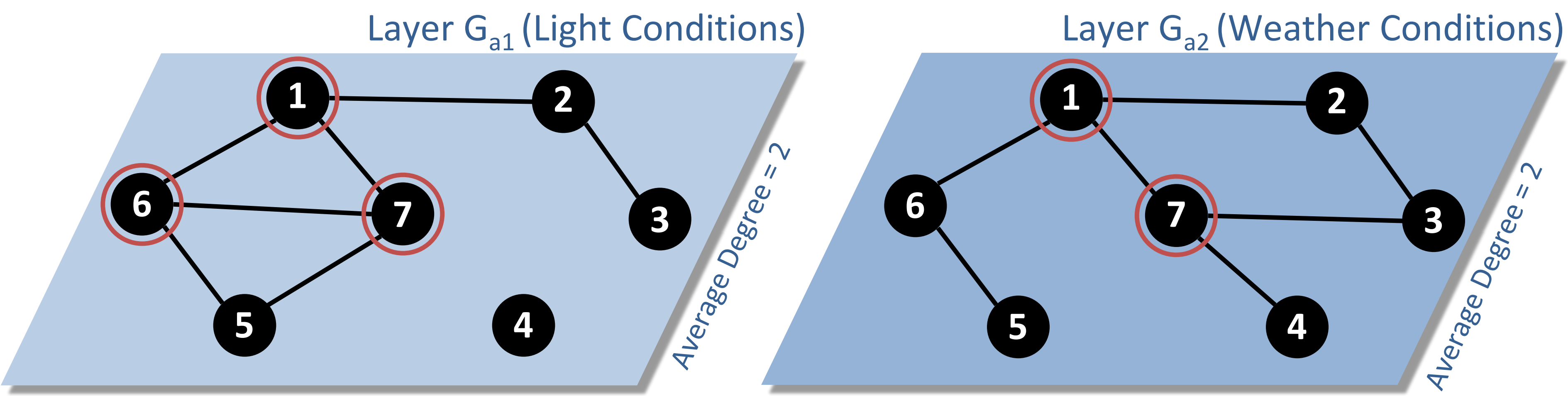
Problem Statement

"Identify the hub sets in any AND-composed layer by using information about the hubs from the participating individual layers"

- *Degree centrality (using average degree)*
- *Closeness centrality (using valued closeness)*

Quantifying Hubs
(Degree Centrality Hubs)

- Higher the degree, greater the influence on immediate neighborhood
- Degree Centrality Hub: A node having the degree above the average degree

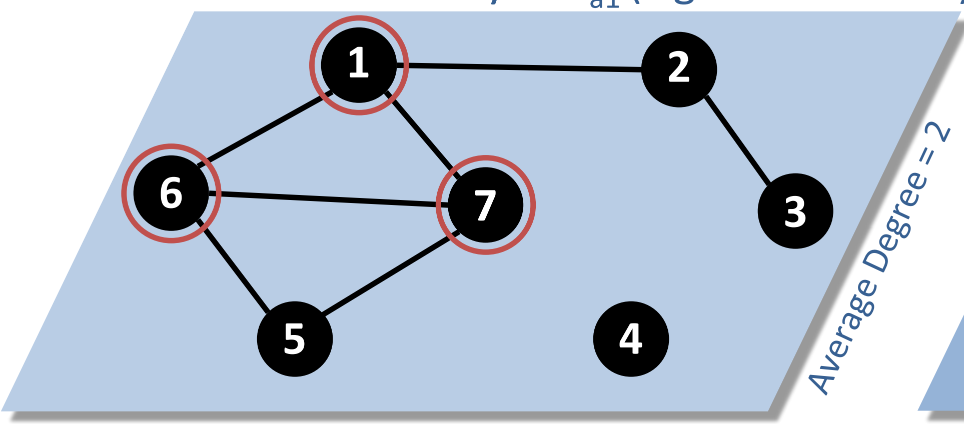


Proposed Solutions

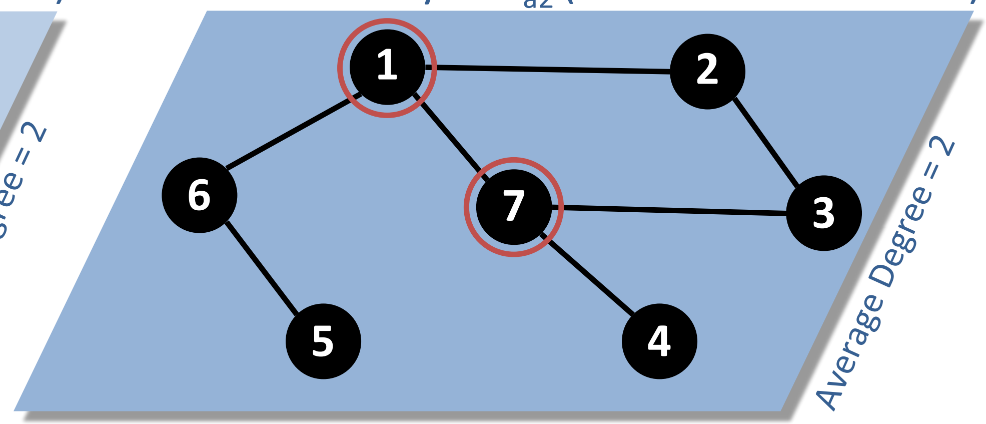
The Naïve Approach

Intersect the Layer-wise Hub Sets

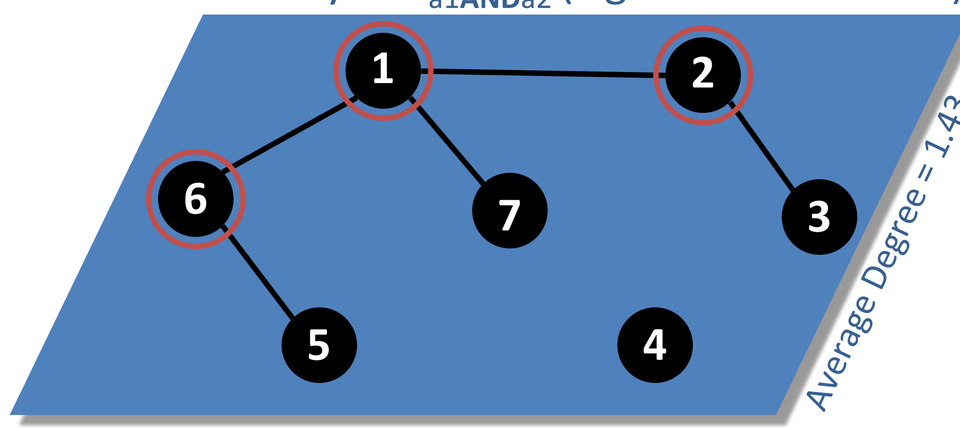
Layer G_{a1} (Light Conditions)



Layer G_{a2} (Weather Conditions)



Layer $G_{a1ANDa2}$ (Light AND Weather)



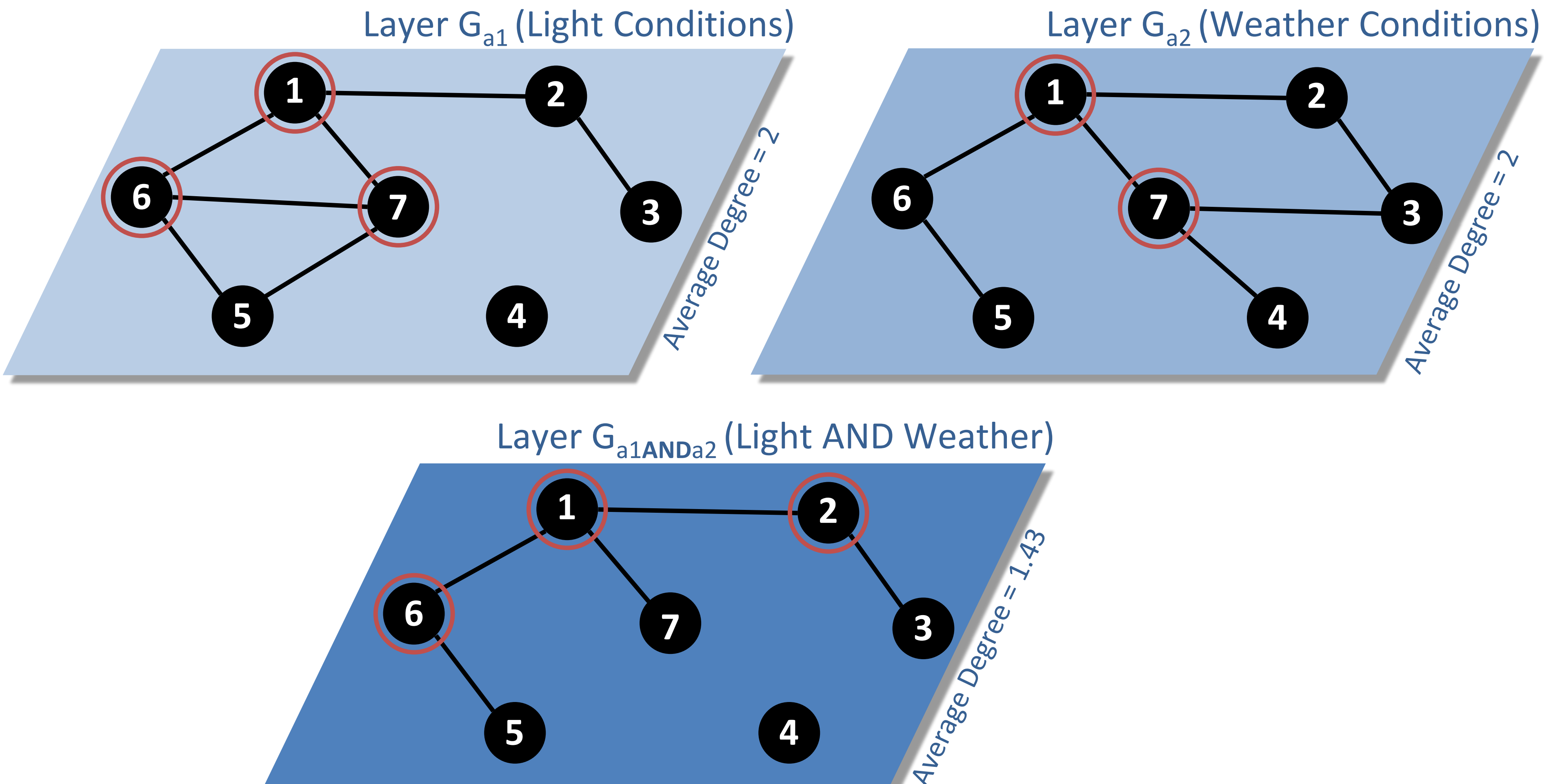
Estimated Hub Set 1 7

False Positive 7

False Negatives 2 6

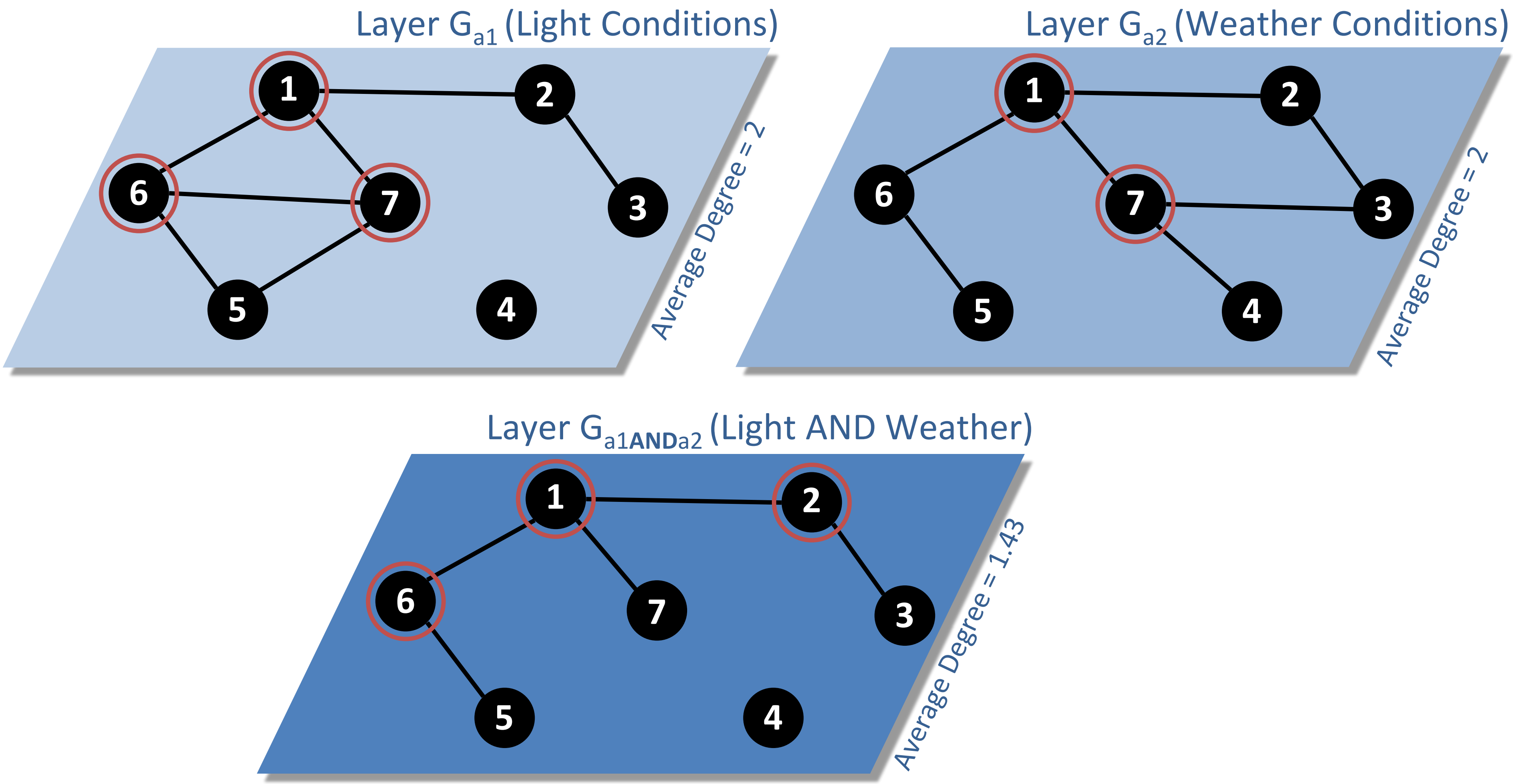
Non-Triviality of the Task (Case 1)

Hubs in individual layers may not be hubs in the AND-composed layer



Non-Triviality of the Task (Case 2)

Non-hubs in individual layers may be **hubs** in the AND-composed layer



Advantages of Proposed Heuristics

- Closeness Centrality based heuristic has also been proposed *(details in the paper)*
- Heuristics are **commutative and associative**
- **Flexible Composition**
  - **Any k-layer AND-composition hub set** can be estimated by using the **2-layer heuristic as a subroutine**, **in parallel**
- **Reduce computational complexity**
  - **Eliminate the need to generate, store and re-compute** degrees or shortest paths for $2^N - N$ layer compositions (N: number of layers)
  - **$2^N - N$ AND-composition hub sets** estimated by only **using N layer-wise hub sets and minimal neighborhood information**

Experimental Analysis

Experimental Setup

- **Datasets**
  - *Accident Multiplex:* 1000 random UK traffic accidents from 2014, 3 conditions-based layers (Light, Weather, Road Surface Conditions)
  - *IMDb Multiplex:* 5000 random actors, 3 genre-based layers (Comedy, Action, Drama)
- **Environment:** UBUNTU 13.10, 4GB RAM, C++ codes
- **Comparison Metrics**
  - **Accuracy**: **Jaccard Index** used to compare estimated (X) and actual (Y) hub sets, $J(X,Y) = |X \cap Y|/|X \cup Y|$
  - **Generation Time**
    - **Actual Hub Set**: Time to generate the AND-composed layer + Time to compute the hub set
    - **Estimated Hub Set**: Time to apply the proposed heuristic

Naïve Approach is not Accurate!

AND-Composed Layers	Degree Centrality	Closeness Centrality
$G_{m1ANDm2}$	59%	43.3%
$G_{m1ANDm3}$	67.9%	55.4%
$G_{m2ANDm3}$	54.4%	48.1%
$G_{m1ANDm2ANDm3}$	14.1%	13.5%
Overall	**48.9%**	**40.1%**

Low accuracies due to **presence of False Positives and Negatives** (IMDb Multiplex)

Performance of Heuristic DC1

High Accuracies due to absence of false positives, **Low hub generation times**

IMDb Multiplex

AND-Composed Layer	Accuracy	Hub Set Generation Time (secs)	
		Actual	Estimated by DC1
$G_{m1ANDm2}$	88.2%	0.0597	0.0302
$G_{m1ANDm3}$	74.6%	0.0681	0.0483
$G_{m2ANDm3}$	82.4%	0.0634	0.0385
$G_{m1ANDm2ANDm3}$	85.9%	0.0492	0.0226
Overall	**82.8%**	0.2403	0.1396 **(41.9%↓)**

Accident Multiplex

AND-Composed Layer	Accuracy	Hub Set Generation Time (secs)	
		Actual	Estimated by DC1
$G_{a1ANDa2}$	78.6%	0.0523	0.0166
$G_{a1ANDa3}$	77.5%	0.0423	0.0152
$G_{a2ANDa3}$	85.7%	0.0711	0.0152
$G_{a1ANDa2ANDa3}$	76.4%	0.0458	0.0147
Overall	**79.5%**	0.2115	0.0618 **(70.8%↓)**

Performance of Heuristic DC2

Better average degree estimate lead to **improved accuracies** as compared to DC1, **but at increased overhead costs**

Accident Multiplex

AND-Composed Layer (Actual Average Degree)	Average Degree $DC1_{est}$	$DC2_{est}$	% Change in Accuracy
$G_{a1AND a2}$ (11.2)	14.92	12.988	5.2%↑
$G_{a1AND a3}$ (10.18)	14.92	12.847	4.4%↑
$G_{a2AND a3}$ (14.35)	16.44	15.257	1.6%↑
$G_{a1AND a2AND a3}$ (9.28)	14.92	12.045	2.7%↑
Overall	–	–	3.5%↑

Improved Accuracy: Accident – 79.5%(DC1), 83.04%(DC2)
IMDb – 82.8%(DC1), 83.9% (DC2)

Fall in Overall Computation Time Savings: Accident – 70.8%(DC1), 58.4%(DC2)
IMDb – 41.9%(DC1), 12.2% (DC2)

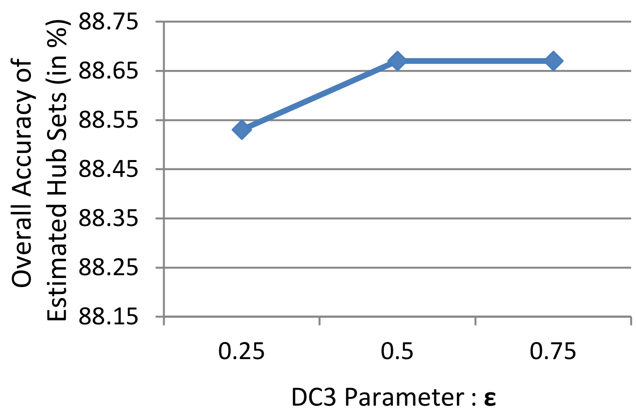
Performance of Heuristic DC3 with Parameter ϵ

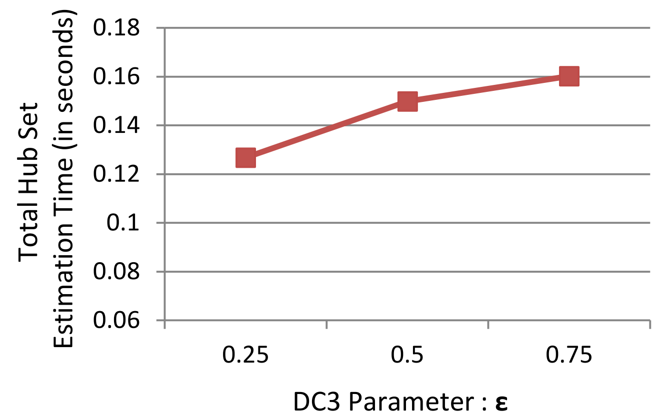
Increasing ϵ:

Overall accuracy increases as the number of false negatives are reduced.

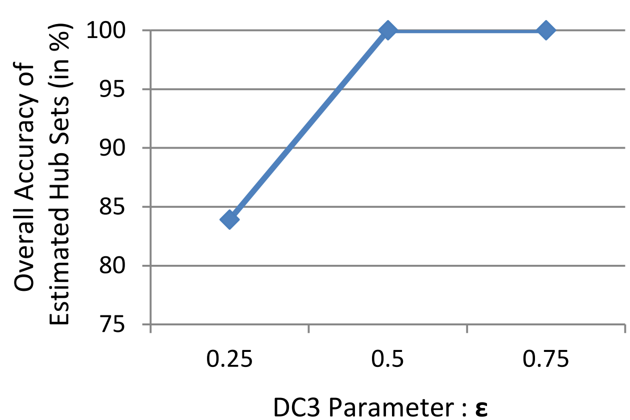
Increases Hub Estimation Times as more layer-wise non-hubs are carried forward

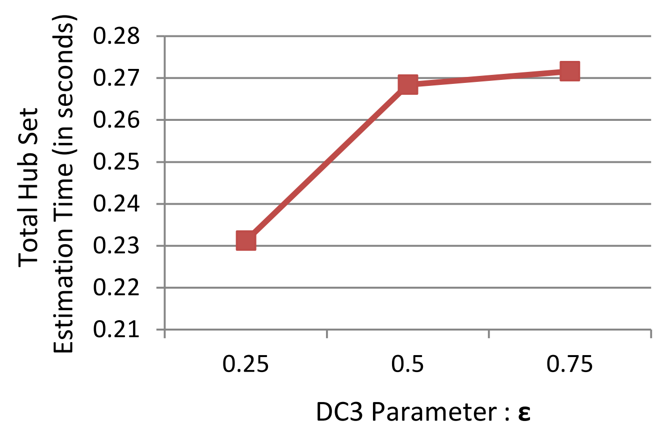
Accident Multiplex





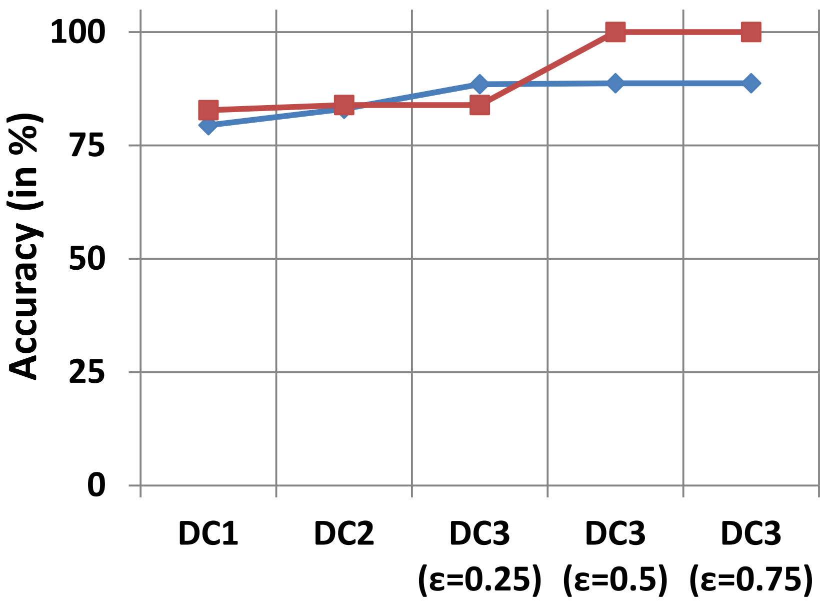
IMDb Multiplex

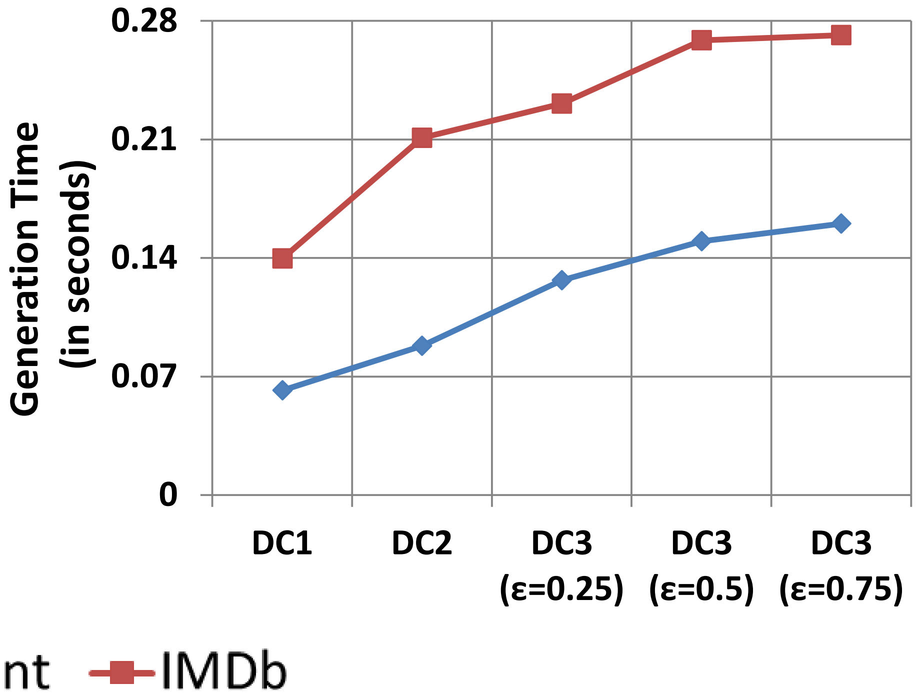




Comparison between DC1, DC2 and DC3

Trade-off between Accuracy and Savings in Computational Costs





Proposed **efficient heuristics to estimate the high centrality vertices for any AND-composed multiplex layer using the layer-wise hub sets**

Overall average **accuracy of at least 70-80%, Reduced the overall computation time by over 30%** *(with real-life multi-feature datasets – traffic accidents, IMDb)*

Homogeneous Multiplex Computations

(Related Publications)

- Scalable Holistic Analysis of Multi-Type, Data-Intensive Problems Using Multilayered Networks – *CoRR abs 2016 (ArXiv)*
- Efficient Community Re-creation in Multilayer Networks Using Boolean Operations – *ICCS 2017*
- FlexiComposer: Flexible Composition of Multilayer Network Communities using Boolean Operations – ***under preparation***
- HUBify: Efficient Estimation of Central Entities across Multiplex Layer Compositions – *ICDM-W 2017*
- *BDA 2017 paper*
- *Computing Communities in Heterogeneous multilayer networks: A Bipartite Graph approach*

  under preparation

Homogeneous Multiplex Computations (Case Study)

Analysis of airlines multiplex using degree and closeness centrality measures

Data Set

- **6 airline** websites were crawled to extract the **non-stop flights** that ply between US cities
  - **American Airlines**
  - **Spirit Airlines**
  - **Delta Air Lines**
  - **Southwest Airlines**
  - **Allegiant Air**
  - **Frontier Airlines**
- Routes active in **February 2018** have been considered

The Airline Multiplex

- **Nodes:** The same set of **214 US cities** were represented through nodes in each layer
- **Edges:** Two cities are connected by an *unweighted and undirected edge* in the i^{th} layer if there is a direct flight between them
- **Layer 1 (American Airlines)**
  - Number of Edges: 746
- **Layer 2 (Delta Airlines)**
  - Number of Edges: 689

American Airlines

DELTA

The Airline Multiplex

- **Layer 3 (Southwest Airlines)**
  - Number of Edges: 717
- **Layer 4 (Allegiant Airlines)**
  - Number of Edges: 379
- **Layer 5 (Frontier Airlines)**
  - Number of Edges: 346
- **Layer 6 (Spirit Airlines)**
  - Number of Edges: 189

Analysis Requirements

- Which are the most important cities (hubs) per airline carrier?
- Can the airlines be separated into major and minor airlines?
- Given an airline carrier, recommend the next city for its expansion?

Most Important Cities

- **Higher the degree of a node**, more is the number of flights plying from the corresponding city.
- **Higher is the closeness centrality of a node**, faster it is to travel from any other city to this particular city, in terms of number of intermediate flights.

Most Important Cities

American Airlines

Degree Centrality Hubs	Closeness Centrality Hubs
Dallas	Dallas
Charlotte	Chicago
Chicago	Charlotte
Washington	Philadelphia
Philadelphia	Phoenix

DELTA

Degree Centrality Hubs	Closeness Centrality Hubs
Atlanta	Atlanta
Minneapolis	Minneapolis
Detroit	Detroit
Salt Lake City	Salt Lake City
New York	New York

Southwest

Degree Centrality Hubs	Closeness Centrality Hubs
Chicago	Chicago
Denver	Denver
Baltimore	Baltimore
Dallas	Dallas
Las Vegas	Las Vegas

Most Important Cities

Allegiant

Degree Centrality Hubs	Closeness Centrality Hubs
Orlando	Orlando
Las Vegas	Tampa
Tampa	Las Vegas
Phoenix	Phoenix
Fort Myers	Fort Myers

Frontier Airlines

Degree Centrality Hubs	Closeness Centrality Hubs
Denver	Denver
Orlando	Orlando
Las Vegas	Austin
Austin	Las Vegas
Philadelphia	Philadelphia

Spirit

Degree Centrality Hubs	Closeness Centrality Hubs
Fort Lauderdale	Fort Lauderdale
Detroit	Las Vegas
Las Vegas	Orlando
Orlando	Detroit
Baltimore	Baltimore

Degree and closeness centrality measures are able to figure out the airline-wise important cities in terms of **maximizing neighborhood connectivity and minimizing overall travel across US cities**, respectively

Categorizing Airlines

- Based on the edge density of a layer, an airline was categorized into major or minor airline.

Airlines		Fleet Size	Revenue
Major Airlines *(higher edge density)*	American	950	$42.2 Billion
	Southwest	857	$21.2 Billion
	Delta	718	$41.2 Billion
Minor Airlines *(lower edge density)*	Allegiant	100	$1.4 Billion
	Frontier	78	$1.4 Billion
	Spirit	118	$2.6 Billion

Verified the categorization using the **fleet size and annual revenue** information. Thus, **edge density has a positive correlation** with these parameters.

Promising Cities for Business

- Business types varies from city to city
- For each category, heuristics were used to estimate the cities that will be hubs in the 3-layer AND composition
  - Tier I Cities (Hubs of Major Airlines)
  - Tier II Cities (Hubs of Minor Airlines)

Tier I Cities

Chicago
Phoenix
Los Angeles
Washington
Atlanta
New York
Dallas

- Good place to invest in restaurants, advertisement and brand enhancement
- People have good spending power.
- Large scale business can consider new headquarters here as easy connectivity to other city
- More availability of man power
- Larger audience size for business
- Job market is bigger

Tier II Cities

Cleveland
Columbus
Fort Myers
Orlando
Tampa
Kansas City
Las Vegas
New Orleans
Pittsburg

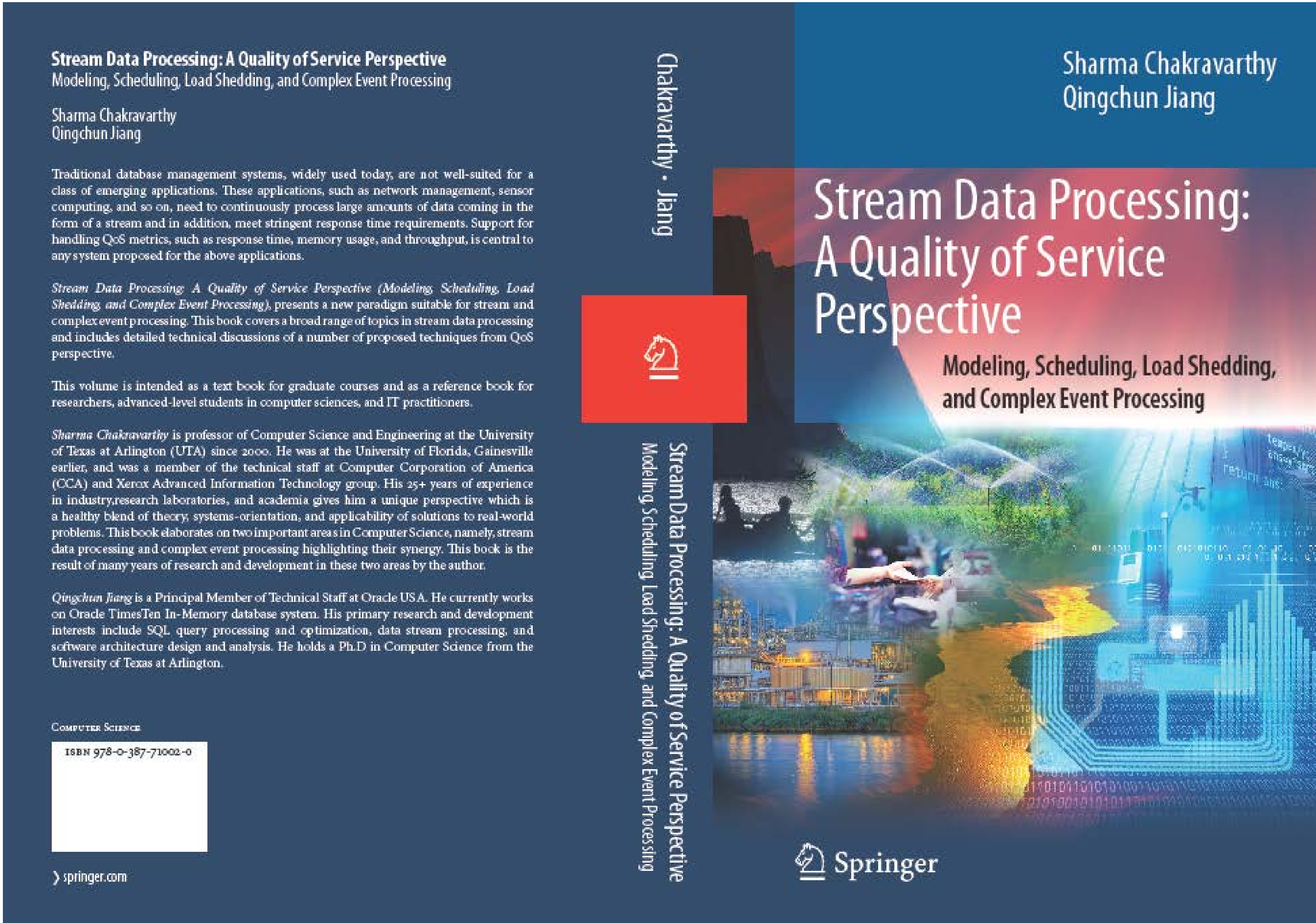
- Small scale industries will benefit because of,
  - Comparatively cheaper real estate rates
  - Decent population and Manpower
- Good place to host event
  - Helps to save budget on location cost but still have good footfall

Proposed heuristics efficiently generate the above set of hubs, by cutting down on both time and storage space required.

Thank You !!!

For more information visit:

http://itlab.uta.edu



Stream Data Processing: A Quality of Service Perspective
Modeling, Scheduling, Load Shedding, and Complex Event Processing

Sharma Chakravarthy
Qingchun Jiang

Traditional database management systems, widely used today, are not well-suited for a class of emerging applications. These applications, such as network management, sensor computing, and so on, need to continuously process large amounts of data coming in the form of a stream and in addition, meet stringent response time requirements. Support for handling QoS metrics, such as response time, memory usage, and throughput, is central to any system proposed for the above applications.

Stream Data Processing: A Quality of Service Perspective (Modeling, Scheduling, Load Shedding, and Complex Event Processing), presents a new paradigm suitable for stream and complex event processing. This book covers a broad range of topics in stream data processing and includes detailed technical discussions of a number of proposed techniques from QoS perspective.

This volume is intended as a text book for graduate courses and as a reference book for researchers, advanced-level students in computer sciences, and IT practitioners.

Sharma Chakravarthy is professor of Computer Science and Engineering at the University of Texas at Arlington (UTA) since 2000. He was at the University of Florida, Gainesville earlier, and was a member of the technical staff at Computer Corporation of America (CCA) and Xerox Advanced Information Technology group. His 25+ years of experience in industry,research laboratories, and academia gives him a unique perspective which is a healthy blend of theory, systems-orientation, and applicability of solutions to real-world problems. This book elaborates on two important areas in Computer Science, namely, stream data processing and complex event processing highlighting their synergy. This book is the result of many years of research and development in these two areas by the author.

Qingchun Jiang is a Principal Member of Technical Staff at Oracle USA. He currently works on Oracle TimesTen In-Memory database system. His primary research and development interests include SQL query processing and optimization, data stream processing, and software architecture design and analysis. He holds a Ph.D in Computer Science from the University of Texas at Arlington.

COMPUTER SCIENCE

ISBN 978-0-387-71002-0

› springer.com

Chakravarthy · Jiang

Stream Data Processing: A Quality of Service Perspective

Modeling, Scheduling, Load Shedding, and Complex Event Processing

Sharma Chakravarthy
Qingchun Jiang

Stream Data Processing: A Quality of Service Perspective

Modeling, Scheduling, Load Shedding, and Complex Event Processing

Springer