Clustering & Summarizing Temporal Graphs

Prithwish Basu

Lead Scientist

Raytheon BBN Technologies

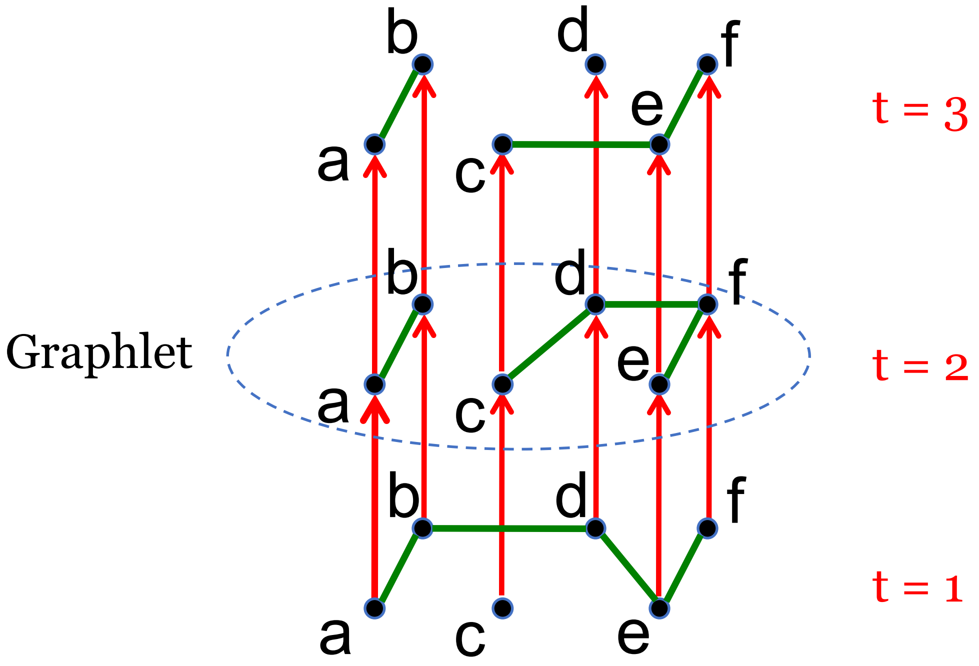
prithwish.basu@raytheon.com

This document does not contain technology or technical data controlled under either the U.S. International Traffic in Arms Regulations or the U.S. Export Administration Regulations.

Temporal Graph Models

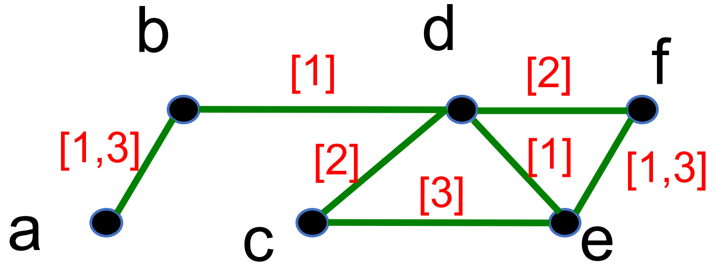
Stacked Graph

[Basu et al 2010]



Evolving Graph

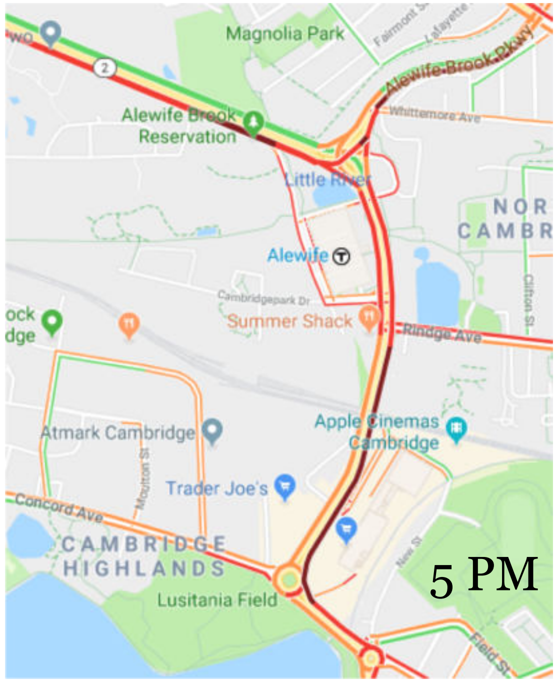
[Ferreira 2004]

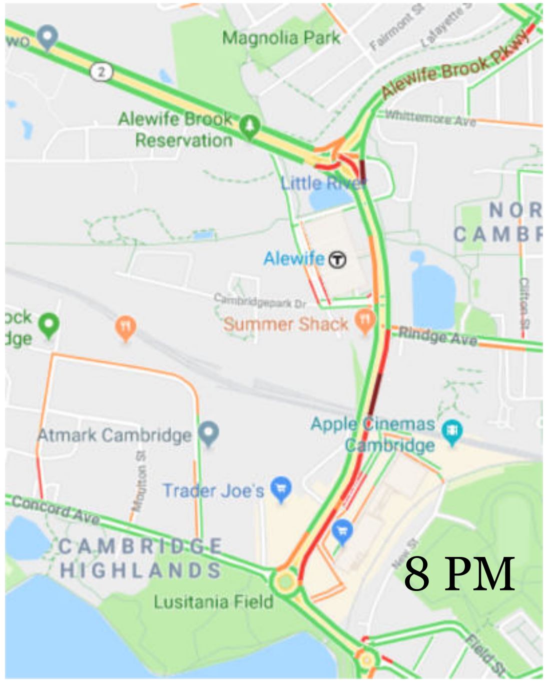


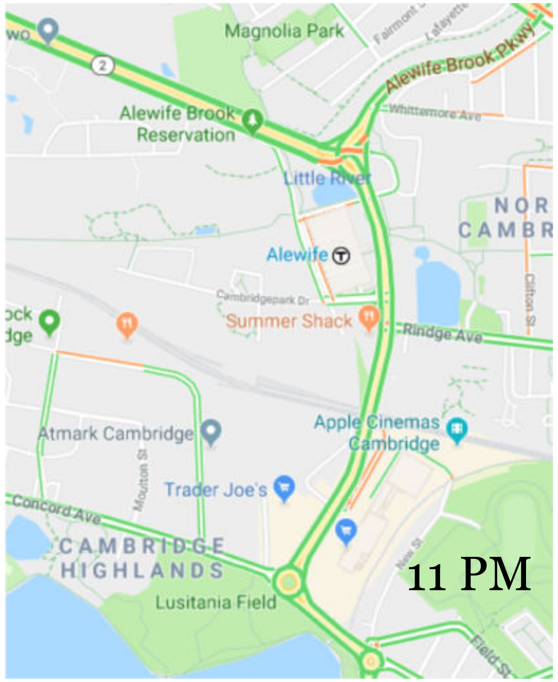
Application Scenario I

Temporal traffic graphs

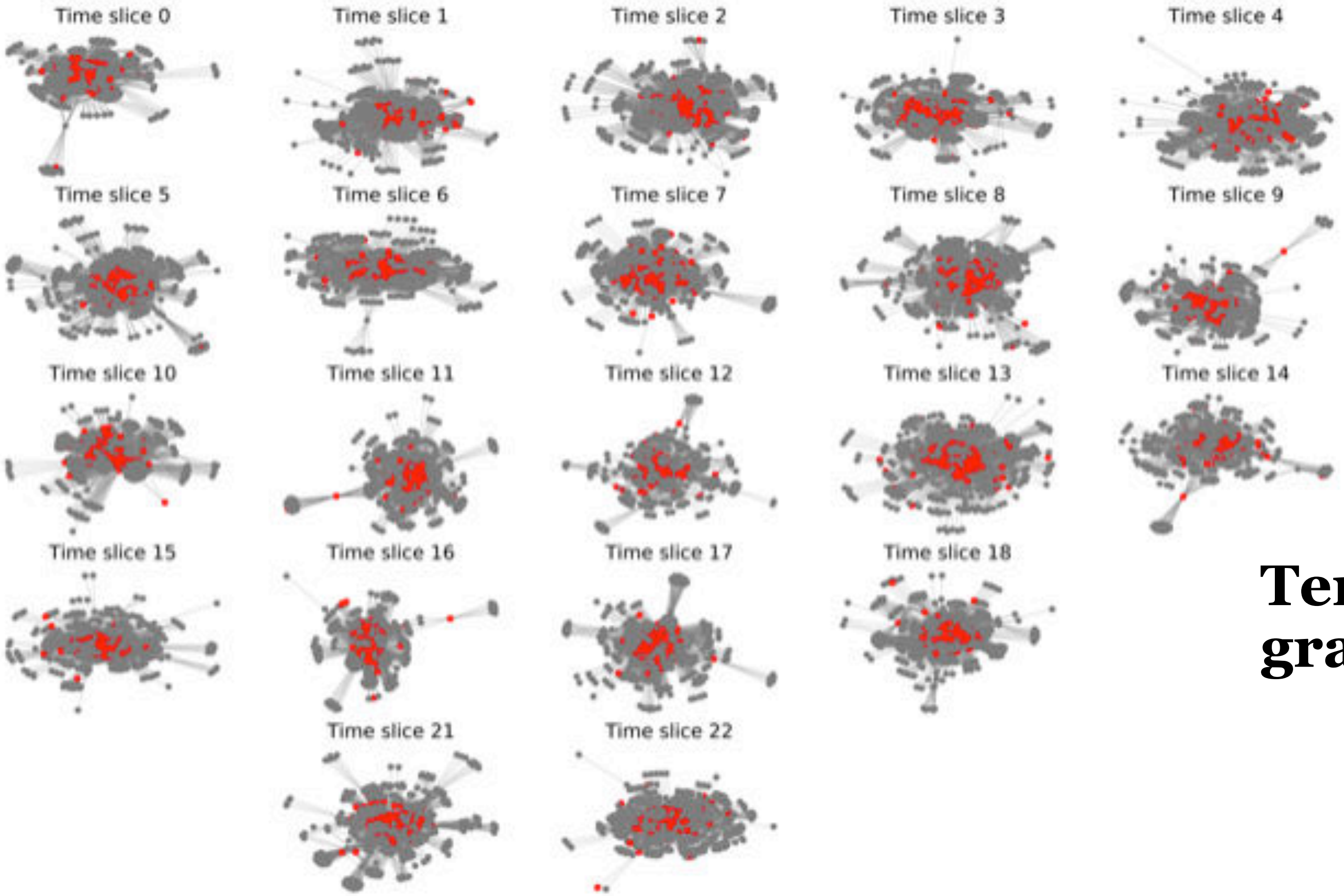
Nodes: intersections, **Edges**: road segments
Graphlet: corresponds to all traffic data in a time unit (e.g., hour)
Node attributes: average flow of vehicles through an intersection
Edge attributes: average speed of vehicles







Application Scenario II



Temporal entity relationship graphs

- **Nodes**: entities (in news articles)
- **Edges**: co-mentions in a news article
- **Graphlet**: corresponds to all data published in a contiguous time window

Problems

- Detect the existence of significant groups of nodes over space & time
  - ➢ Can reveal important information about the network's function
- Unsupervised clustering of nodes in a Temporal Graph
  - ➢ Find clusters that tend to persist over both space and time
  - ➢ The nodes in a cluster should be adjacent / loosely adjacent in the space dimension but need not be contiguous in the time dimension
    - o e.g., Alewife Brook Parkway is congested on Tuesday & Thursday afternoons
- Summarizing Temporal Graph signals
  - ➢ Given a temporal graph with real-valued node attributes, construct a smaller graph that is a skeleton for the larger graph
- Detecting significant graphlets
  - ➢ Which significant sets of nodes tend to be present (with certain attributes) whenever a temporal graph is seen to have a certain overall behavior?

Temporal Graph as Tensors

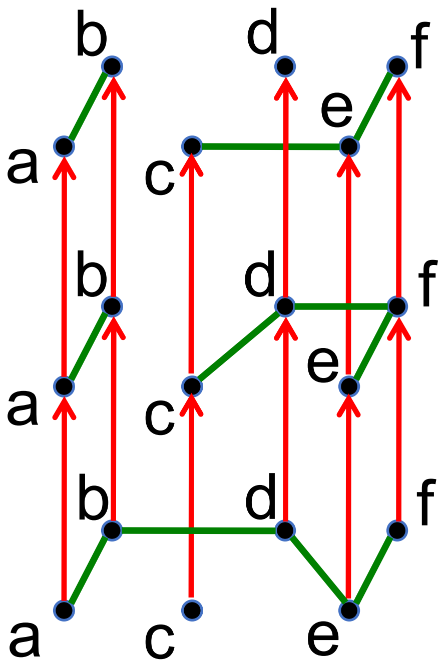
NETWORK SCIENCE CTA

Raytheon BBN Technologies

Temporal graph $G[1..T] = (V, E_1 \cup E_2 \cup \cdots \cup E_T)$

- Labeled edges: { `source ID`, `destination ID`, `time slice label`, `duration` } etc.
- Multiple *categorical* or *numeric* labels can be attached to each edge.
- One of the labels is the time slice index in which the edge is active

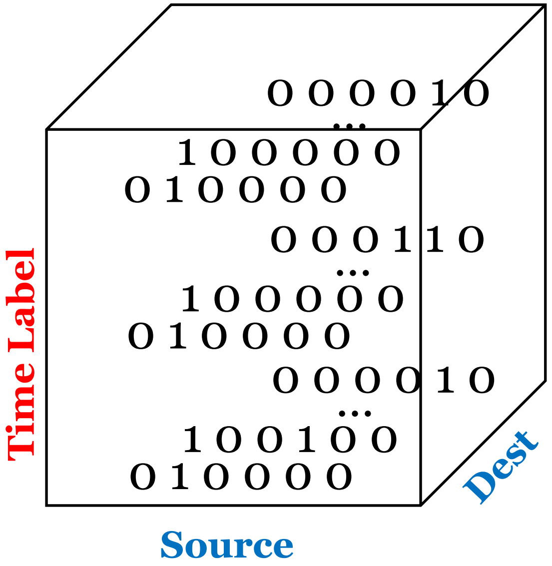
Temporal Graph



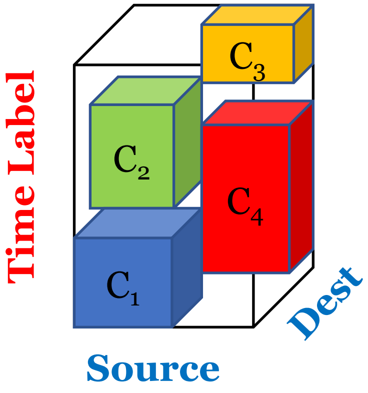
Edge list

T	S	D
0	a	b
0	b	d
0	d	e
0	e	f
1	a	b
1	c	d
1	d	f
1	e	f
2	a	b
2	c	e
2	e	f

3-mode Tensor



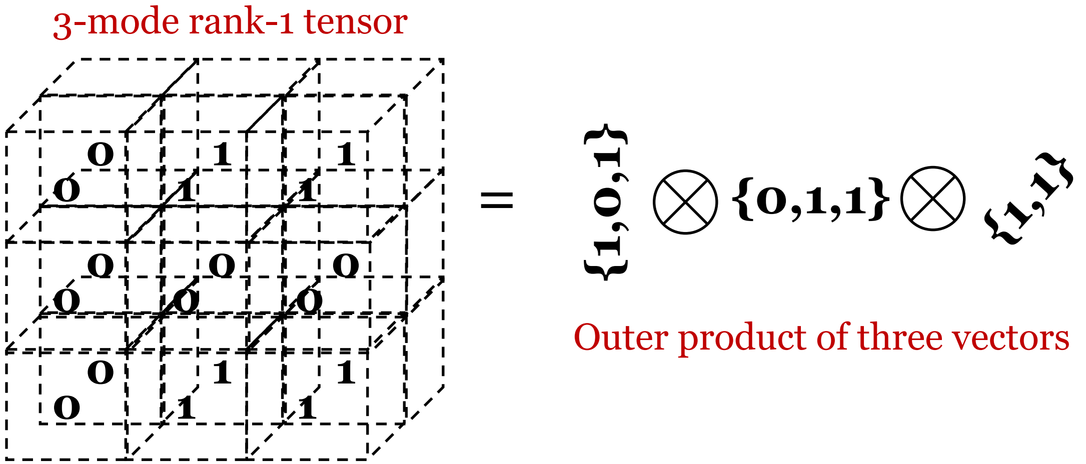
Clusters?



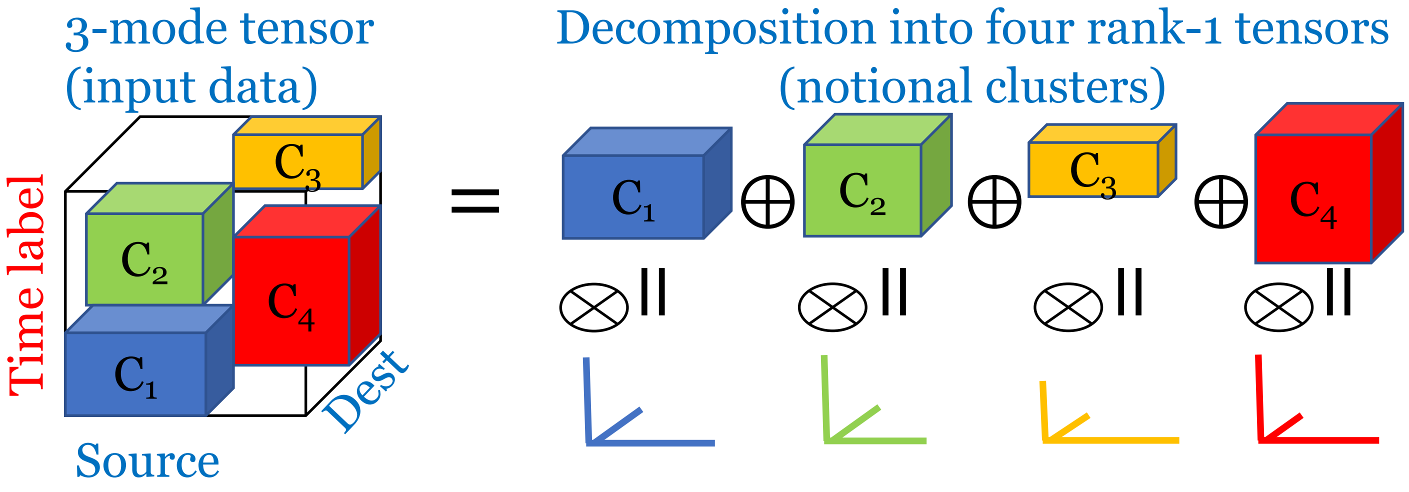
Tensor-based Clustering (COM²)

Araujo, Papadimitriou, Günnemann, Faloutsos, Basu, Swami, Papalexakis, and Koutra, "Com²: Fast Automatic Discovery of Temporal ('Comet') Communities," *PAKDD* 2014.

- **Basic idea**: Represent the tensor as a union of a set of smaller tensors, which can potentially overlap
  - Smaller tensors have rank-1 and hence can be expressed cheaply
    - A 3-mode tensor $G = \{0,1\}^{I_1 \times I_2 \times I_3}$ is rank-1 if it can be written as an outer product of 3 vectors
    - $G = a^{(1)} \otimes a^{(2)} \otimes a^{(3)}$



- **Iterated tensor decomposition**
  - Yields smaller rank-1 tensors (how?)
  - Each rank-1 tensor ⇒ cluster



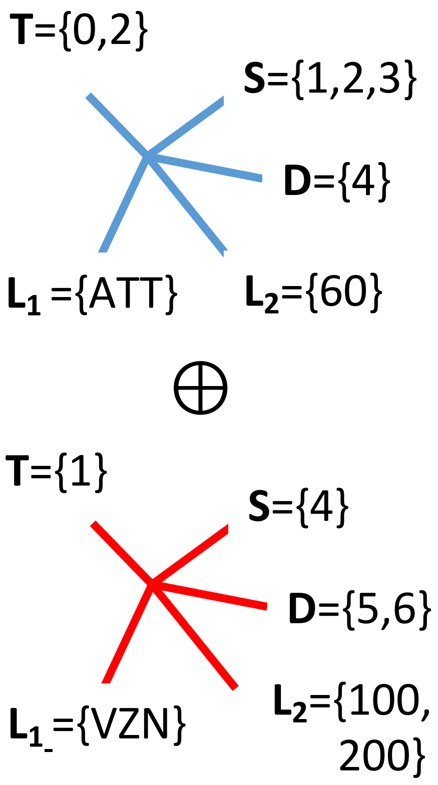
Tensor Decomposition Example

Example tensor

T	S	D	L_1	L_2
0	1	4	ATT	60
0	2	4	ATT	60
0	3	4	ATT	60
1	4	5	VZN	100
1	4	5	VZN	200
1	4	6	VZN	100
1	4	6	VZN	200
2	1	4	ATT	60
2	2	4	ATT	60
2	3	4	ATT	60

=

Rank-1 5-tensors



COM² Rationale

Minimum Description Length (MDL) principle from Information Theory

- Achieve compression: minimize total number of bits required
  - ➢ to encode the model + to describe the data given this model + to describe the outlier data
- Automatically trades off the model complexity and its goodness of fit to the data

Optimization problem

- Cost of encoding a cluster $C_i = (T_i, S_i, D_i)$ (a rank-1 tensor):
  - ➢ $L_1(C_i) = L_{\mathbb{N}}(|T_i|) + L_{\mathbb{N}}(|S_i|) + L_{\mathbb{N}}(|D_i|) + |T_i| \log |T| + |S_i| \log |S| + |D_i| \log |D|$
- Cost of encoding a set of clusters
  - ➢ C: $L_2(C) = L_{\mathbb{N}}(|C|) + \sum_{C_i \in C} L_1(C_i)$
- Cost of encoding the model errors in data $\underline{X}$

$$L_3(\underline{X}|C) = L_{\mathbb{N}}\left(\left\|\underline{X}^C - \underline{X}\right\|_F^2\right) + \left\|\underline{X}^C - \underline{X}\right\|_F^2 \cdot (\log|T| + \log|S| + \log|D|)$$

where $\underline{X}^C = \vee_{C_i \in C} I^{C_i}$ is the reconstructed tensor

- Find $C^* \subseteq \{0,1\}^{|T|} \times \{0,1\}^{|S|} \times \{0,1\}^{|D|}$ such that $C^* = \arg\min_C [L_2(C) + L_3(\underline{X}|C)]$
  - ➢ This is NP-hard, hence, we find good approximate clusters sequentially

COM² Algorithm

Algorithm Steps

1. Spot candidate nodes by efficient rank-1 approximation of tensor $\underline{X}$.

- $a_i \leftarrow \sum_{j=1}^{|S|} \sum_{k=1}^{|D|} X_{i,j,k} b_j c_k$
- $b_j \leftarrow \sum_{i=1}^{|T|} \sum_{k=1}^{|D|} X_{i,j,k} a_i c_k$ $\qquad \cong$ *Alternating Least Squares method for rank-1 results*
- $c_k \leftarrow \sum_{i=1}^{|T|} \sum_{j=1}^{|S|} X_{i,j,k} a_i b_j$

2. Start with a well-connected entry (a_i, b_j, c_k); grow/shrink a cluster by using score vectors a, b, c as bias in hill-climbing search and scoring each accept/reject move by MDL cost.
3. Deflate tensor by removing the cluster found in step 2.
4. Repeat above steps until $\underline{X = 0}$ (convergence is guaranteed since MDL cost is decreasing)

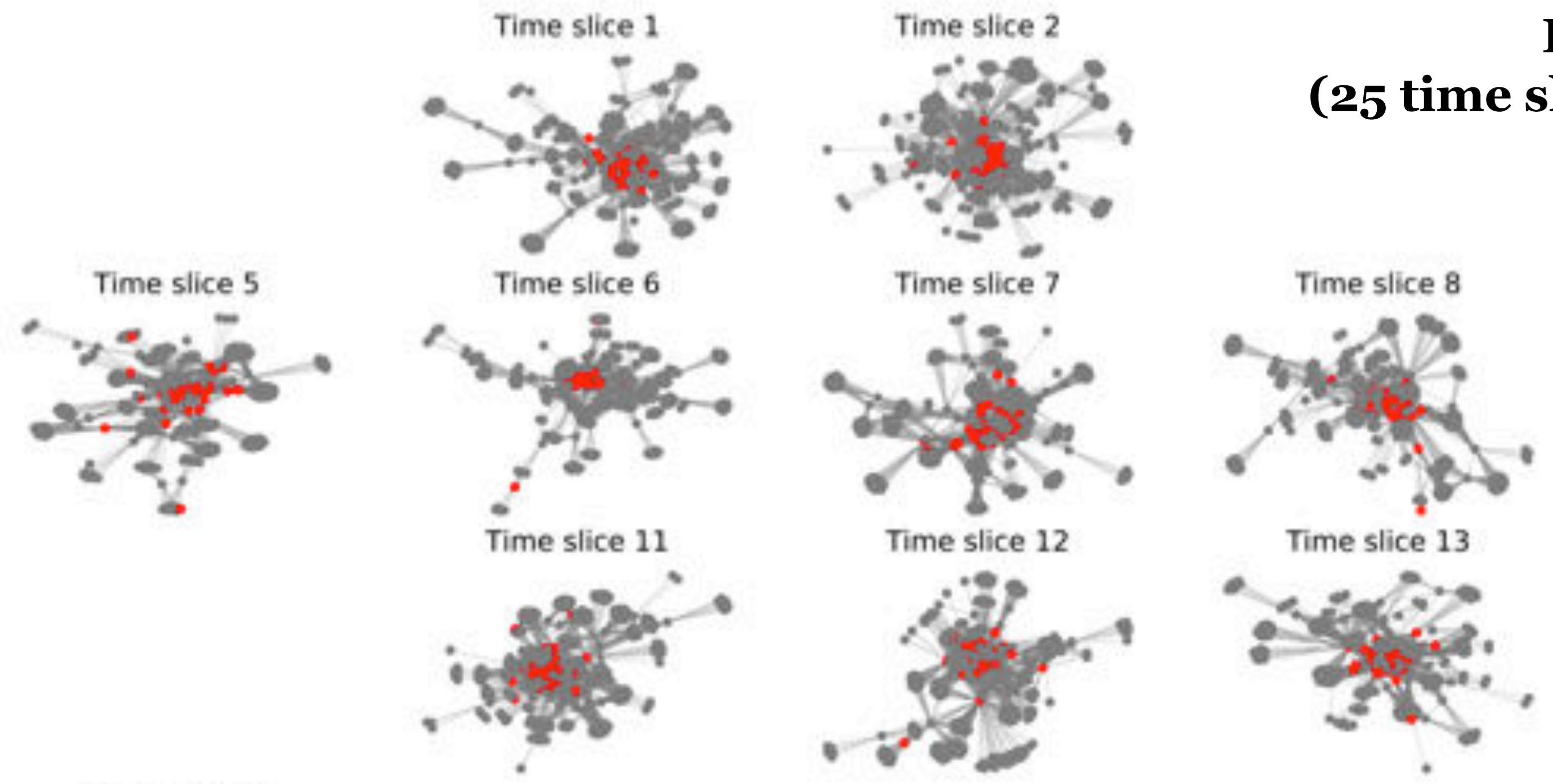
Benefits of COM²

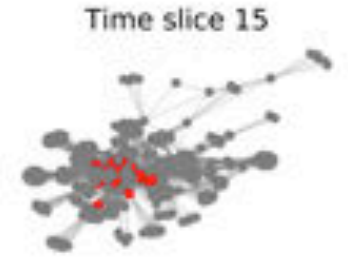
- Parameter free
- Time complexity: $O(|C|(E + L \log L \log N))$
  - N: length of biggest mode; L: size of biggest cluster; $|C|$: number of clusters
  - $|C|, N, L \ll E$ implies linear scaling with number of non-zero elements (E)

Case study: Information Network

Raytheon BBN Technologies

Boston Blizzard Data Set
(25 time slices, 7816 entities, 200100 edges)

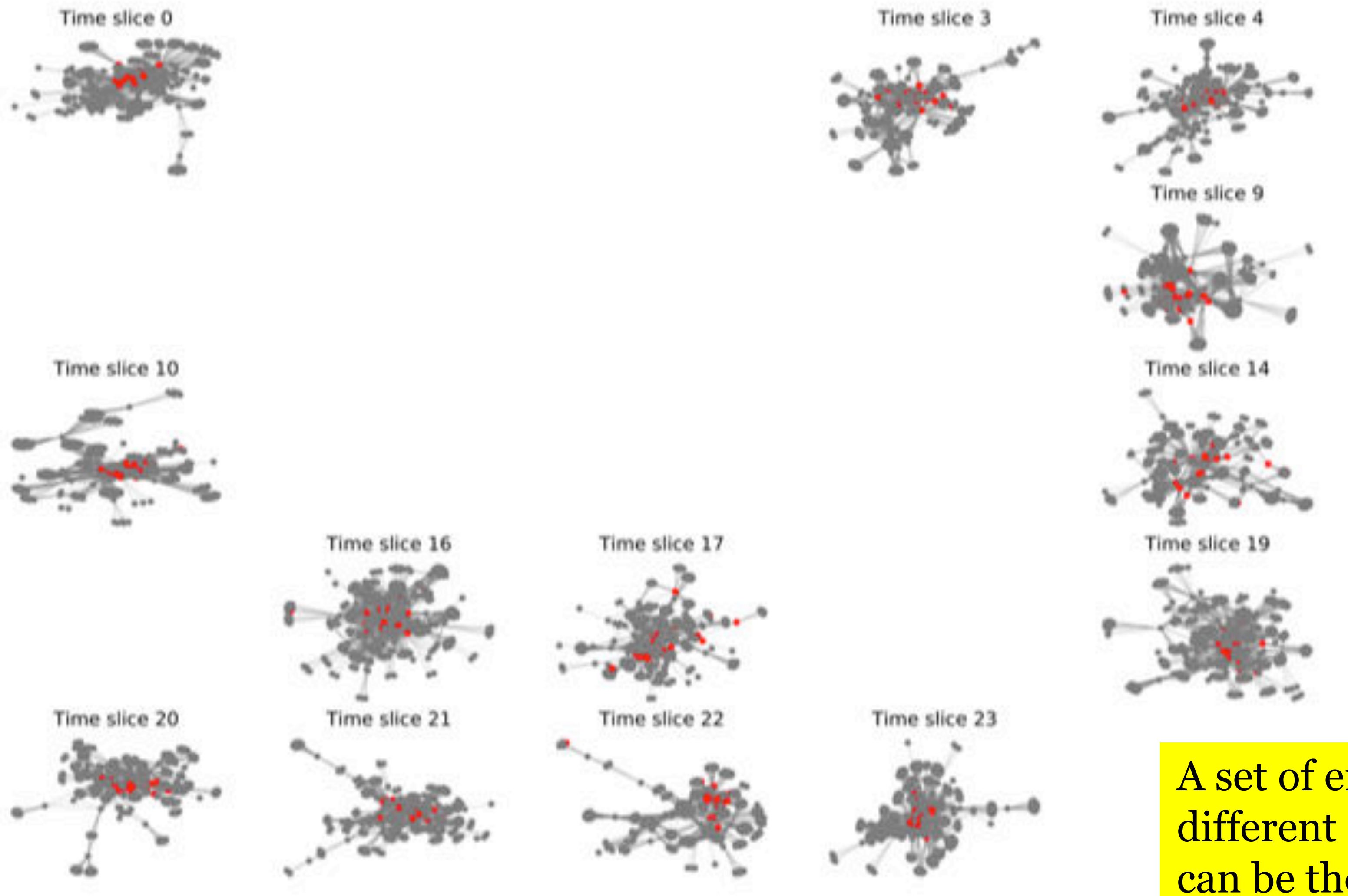




A set of entities mentioned multiple times in different news articles over a period of time can be thought of as belonging to a "story".

Cluster ID: 1
times: [1, 2, 5, 6, 7, 8, 11, 12, 13, 15]
entities: ['Rhode Island', 'Suffolk County', 'United States', 'Lunenburg', 'Amtrak', 'Faneuil Hall', 'Washington', 'Andrew Cuomo', 'Massachusetts', 'Scarborough', 'BOSTON', 'New York', 'Bob Paglia', 'Gorham', 'Logan Airport', 'William Pittman', 'Jeff Russell', 'Cuomo', 'National Weather Service', 'Bostons Logan Airport', 'Denise Gorham', 'National Guard', 'Northeast', 'Massachusetts Cape Cod', 'Reuters', 'Portland', 'Gary Szatkowski', 'Samuel Adams', 'New England', 'Long Island', 'the Sons of Liberty', 'Marthas Vineyard', 'Louis Uccellini', 'New Hampshire', 'Nantucket', 'New York City', 'eastern New England', 'Auburn', 'Worcester', 'U.S.', 'Boston Common', 'AP', 'Boston', 'Cape Cod', 'Framingham', 'Newport', 'East Coast', 'Maine', 'New Jersey', 'Mike Spigarolo', 'Montauk', 'NEW YORK', 'Connecticut', 'Whitman', 'Mount Holly', 'Charlie Baker', 'Redcoats', 'Providence', 'Maureen Keller', 'Trumbull', 'Bill de Blasio', 'south coast', 'Philadelphia', 'Chris Christie', "Martha's Vineyard", 'Brandon Bhajan', 'Susanne Payot', 'Marshfield']
number of articles: 283

Case study: Information Network



Boston Blizzard Data Set (25 time slices, 7816 entities, 200100 edges)

Cluster ID: 2
times: [0, 3, 4, 9, 10, 14, 16, 17, 19, 20, 21, 22, 23]
entities: ['Auburn', 'Rhode Island', 'Lunenburg', 'Andrew Cuomo', 'Massachusetts', 'Worcester', 'the National Weather Service', 'New York', 'Boston', 'National Weather Service', 'Northeast', 'Baker', 'Maine', 'New Jersey', 'Portland', 'Connecticut', 'Charlie Baker', 'New England', 'Long Island', 'Providence', 'Scituate', 'Marshfield', 'Mass.', 'New Hampshire', 'Nantucket', 'New York City']
number of articles: 259

A set of entities mentioned multiple times in different news articles over a period of time can be thought of as belonging to a "story".

Ongoing research

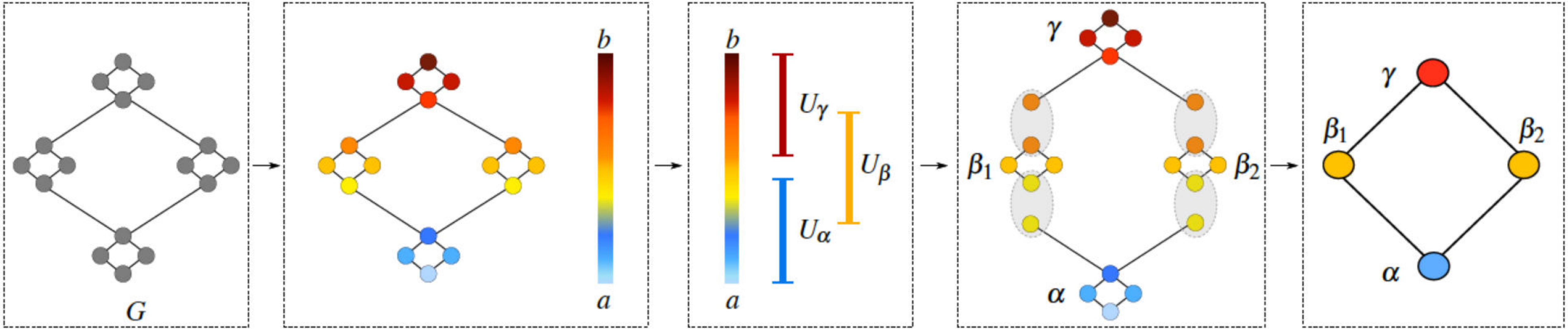
- How to handle weighted tensors?
  - Some edges may be more important than others
- Can multiple overlapping clusters be combined to create better clusters?
  - By post-processing output from COM^2?
- Measuring the quality of clusters
  - Obtaining ground truth can be very challenging

Summarizing Temporal Graph signals

Given a temporal graph with real-valued node attributes, construct a smaller graph that is a **skeleton** for the larger graph

Topological Summarization

- Summarize a topological space $\mathbb{x}$ with respect to a *lens*, i.e., attached data $f: \mathbb{x} \to \mathbb{R}$
- **Mapper** algorithm
  - ➢ For 3D point-clouds [G. Singh, F. Memoli, and G. Carlsson, 2007]
  - ➢ Simple extension to graphs [M. Hajij, B. Wang, and P. Rosen, 2018]



Graph signal $f: V \to [a, b]$

Cover elements of cover U

Connected components of subgraphs induced by node sets $\{f^{-1}(U_i)\}$ cover G, $f^*(U) = \{\alpha, \beta_1, \beta_2, \gamma\}$

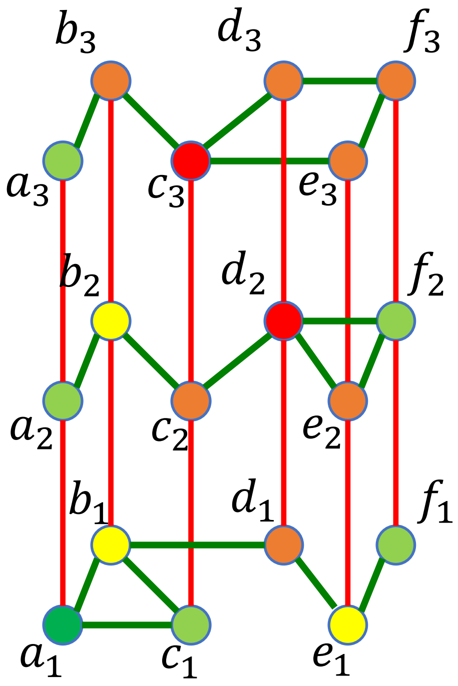
Nerve Graph of $f^*(U)$ (Mapper graph)

Skeletonizing Temporal Graphs

NETWORK SCIENCE CTA

Raytheon BBN Technologies

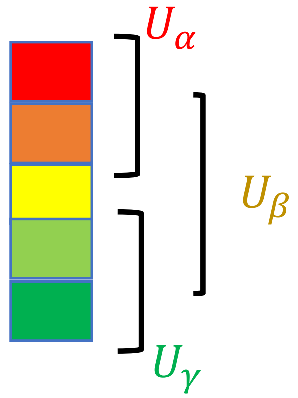
Smooth temporal graph signals



$G[1,3] = (V, E_1 \uplus E_2 \uplus E_3)$

$f: V \to [0,1]$

Cover $U = \{U_\alpha, U_\beta, U_\gamma\}$



$U_\alpha \cap U_\beta \neq \emptyset$

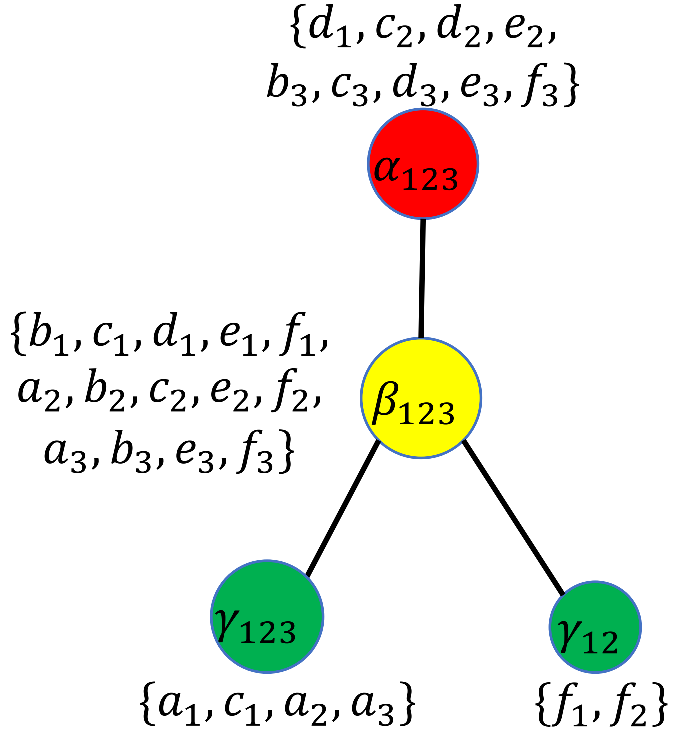
$U_\beta \cap U_\gamma \neq \emptyset$

$U_\gamma \cap U_\alpha = \emptyset$

$U_\alpha \cup U_\beta \cup U_\gamma = [0,1]$

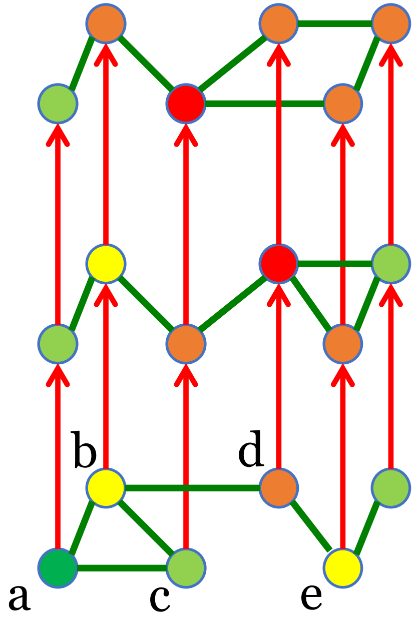
Nerve Graph of path-connected components of $\{f^{-1}(U_\alpha), f^{-1}(U_\beta), f^{-1}(U_\gamma)\}$

(ignoring temporal directionality)



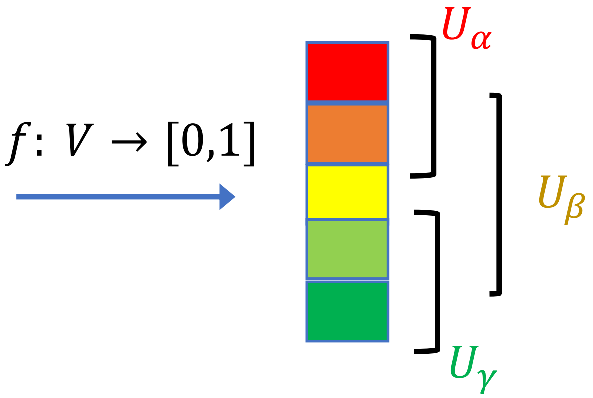
Skeletonizing Temporal Graphs

Smooth temporal graph signals



$G[1,3] = (V, E_1 \uplus E_2 \uplus E_3)$

Cover $U = \{U_\alpha, U_\beta, U_\gamma\}$



$U_\alpha \cap U_\beta \neq \emptyset$

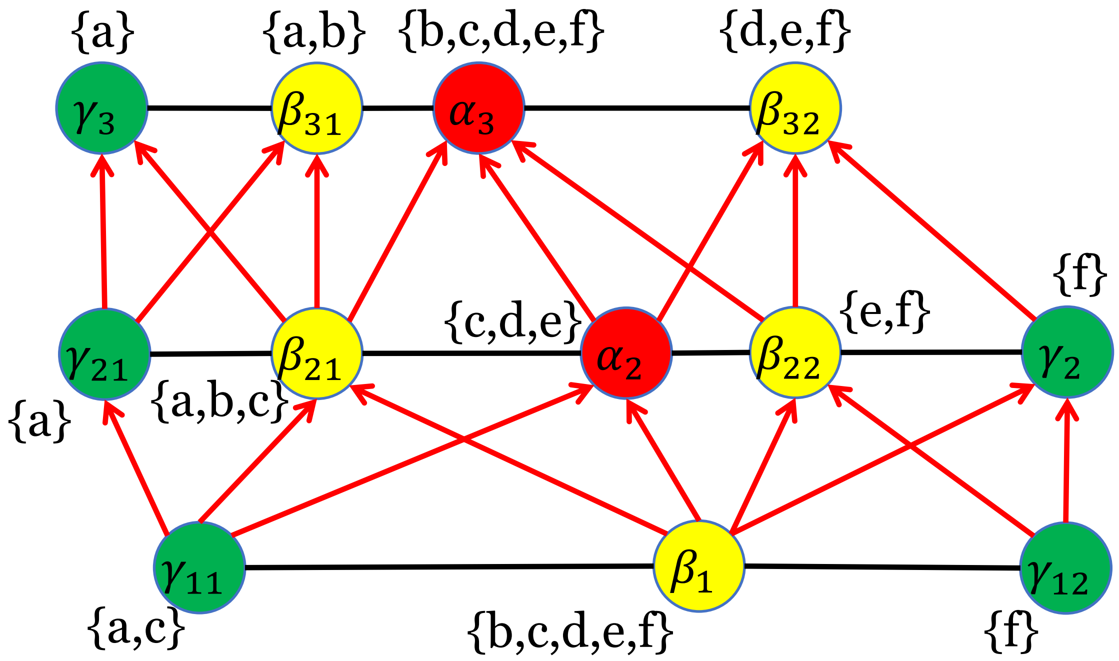
$U_\beta \cap U_\gamma \neq \emptyset$

$U_\gamma \cap U_\alpha = \emptyset$

$U_\alpha \cup U_\beta \cup U_\gamma = [0,1]$

Nerve Graph of path-connected components of $\{f^{-1}(U_\alpha), f^{-1}(U_\beta), f^{-1}(U_\gamma)\}$

(not ignoring temporal directionality)



Ongoing research

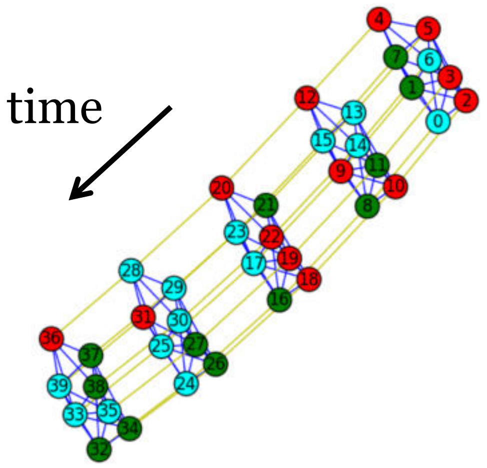
- Application: Visualization of large complex temporal networks
  - What's the measure of the quality of visualization?
- Application: Solving complex network flow problems
  - Flow on smaller skeleton graphs may be much cheaper to compute but have errors
- Measuring quality of summarization
  - Treat it as a lossy graph compression problem of a special type
  - Compressed nodes are just collections of nodes

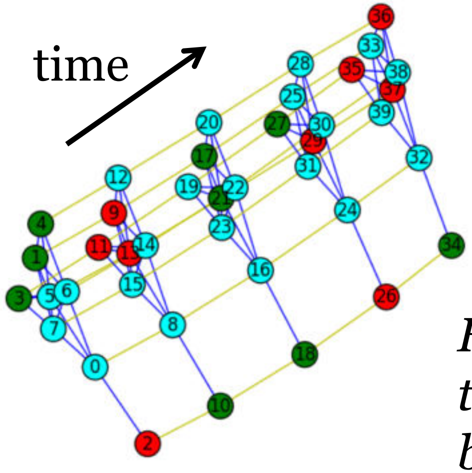
Detecting significant graphlets in Temporal Graphs

Which significant sets of nodes tend to be present (with certain attributes) whenever a temporal graph is seen to have a certain overall behavior?

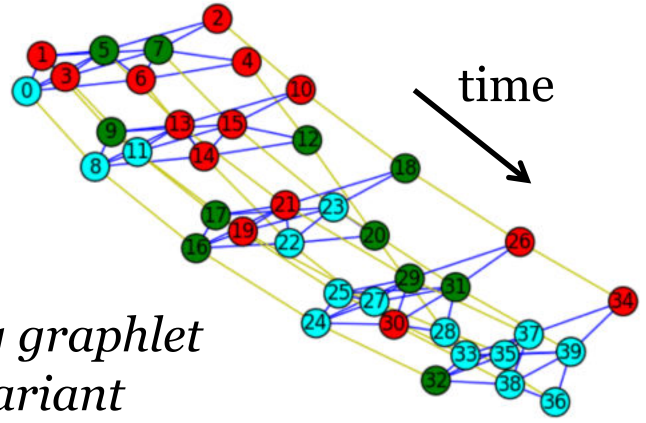
Discovering patterns

Detect node-valued patterns in a temporal graph that are positively correlated with an overall target behavior, e.g., congestion





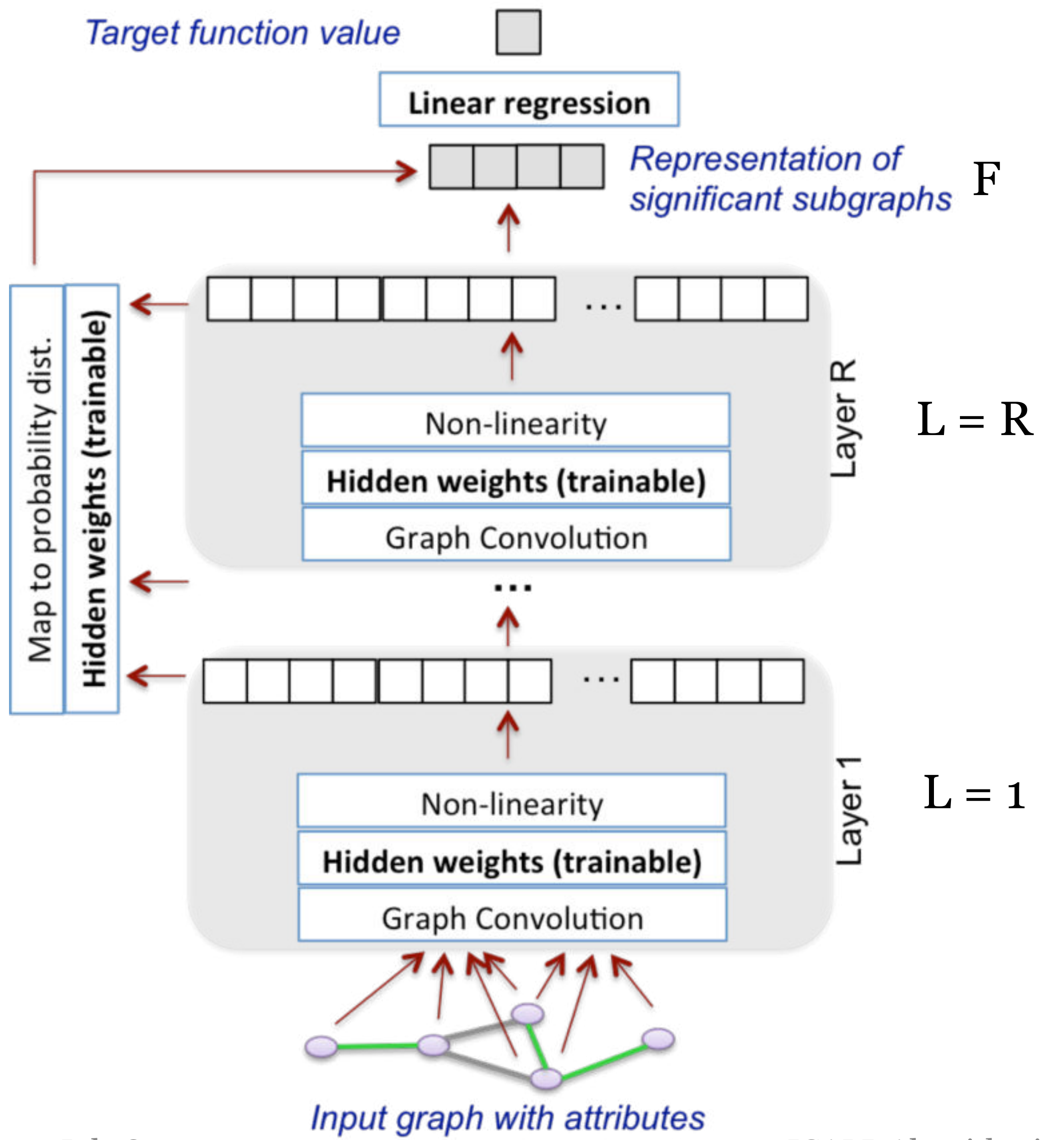
Here, the underlying graphlet topology is time-invariant but node values change



Can we detect a subgraph of 3 cyan nodes that exists through most time slices?

- These are imprecise variants of node-value based subgraph isomorphism
- So, we turn to Deep Neural Networks

Graph-convolutional DNN



GCNN training procedure (following Duvenaud et al.'s molecular fingerprinting architecture)

- Inputs
  - Temporal graph **G**; radius **R**; max deg $\mathbf{\Delta}$
  - hidden weights $\mathbf{H_1^1}, \ldots, \mathbf{H_R^\Delta}$
    - tracks contributions at each value of *hop-distance* & *node degree*
  - output weights $\mathbf{W_1}, \ldots, \mathbf{W_R}$
- Representation vector $\mathbf{F} \leftarrow \mathbf{0}$
- $\forall_{\mathbf{u} \in \mathbf{G}}\, \mathbf{x_u}$:= Attributes of **u**
- For each layer indexed $\mathbf{L \in [1,R]}$
  - For each node $\mathbf{u \in G}$
    - $\mathbf{x_1, x_2, \ldots, x_N}$ = neighbors(**u**) // $N \leq \Delta$
    - $\mathbf{y} := \mathbf{x_u} + \sum_1^N \mathbf{x_i}$ // Convolution
    - $\mathbf{x_u} := \sigma(\mathbf{y\, H_L^N})$ // Non-linearity
    - $\mathbf{i}$:= softmax($\mathbf{x_u W_L}$) // Map to prob.
    - $\mathbf{F} := \mathbf{F} + \mathbf{i}$ // Add to repr. vector
- Return vector **F**

Supervised training

- For each training network sample (temporal graph)
  - Mark a binary target regarding whether or not there is a cyan-colored 3-node subgraph in that sample
- Experimental setup
  - N_{train} = 54 (with supervised target patterns), $N_{validate}$ = 9, N_{test} = 18
  - Params: F = 20, R = 4, batch size = 9, SGD Adam step size = e^{-6}
- Training results
  - 500 iterations (Training RMSE: 0.0953)

True value	0	1	0	0	0	1	0	1	0	1
Prediction	0.05	1.14	0.05	0.05	0.08	1.14	0.05	1.13	0.07	1.12

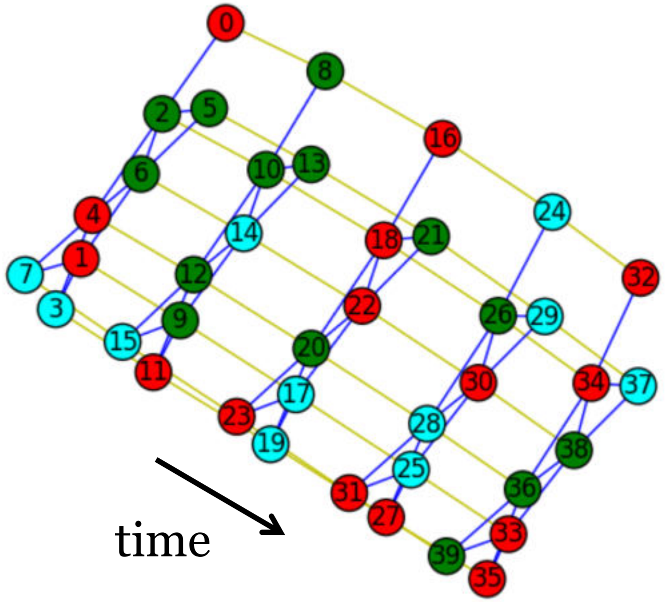
Regression Results

Raytheon
BBN Technologies

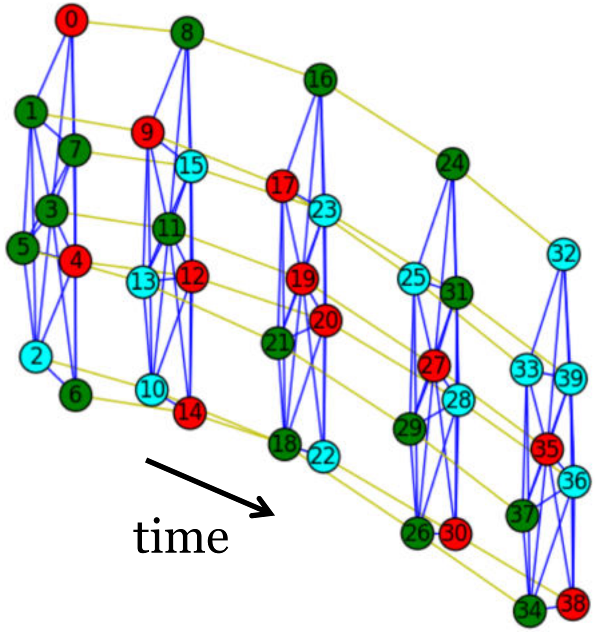
Test RMSE: 0.1643

True value	0	0	1	0	1	0	1	0	0	1	1	1	1	0	0	0	0	0
Prediction	0.26	0.14	1.0	0.14	0.96	0.18	1.07	0.16	0.16	1.03	1.10	1.03	0.63	0.20	0.14	0.16	0.19	0.14

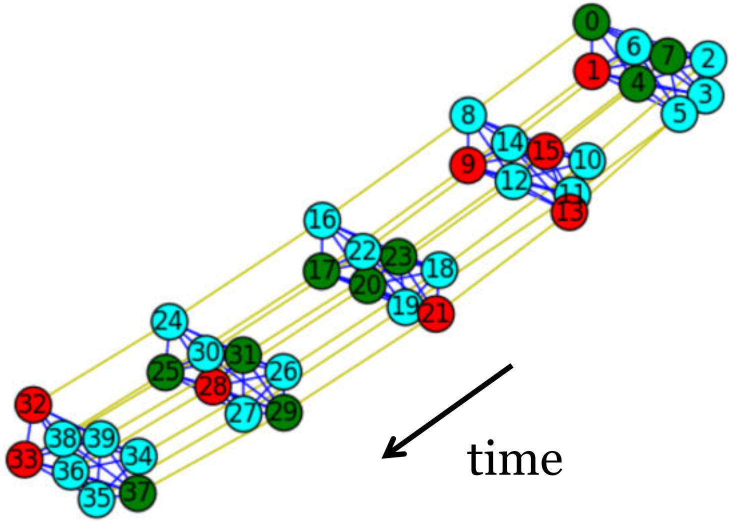
No cyan path present



No cyan path present

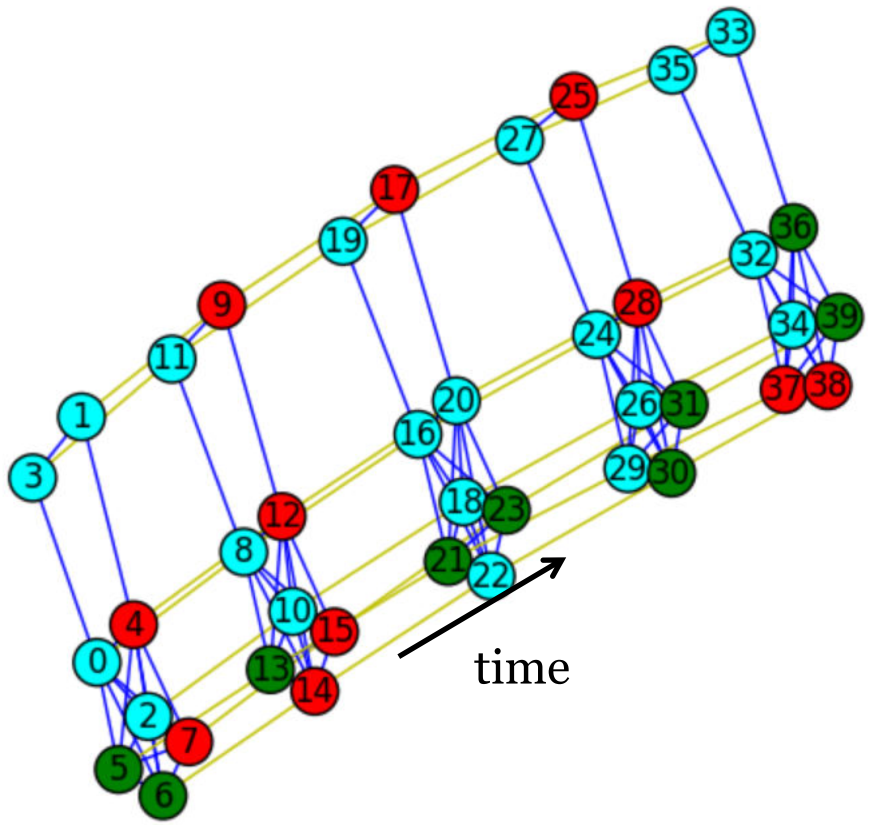


Cyan {2— 5—6} present through 5 time steps



Detection of significant patterns

- GCNN identified a subgraph within a given graph sample
  - ➢15 cyan nodes and 2 green nodes spread over time
  - ➢This pattern has the most significant correlation with the overall graph behavior of "congested"
  - ➢Activation coefficient = 0.93
- Testing on real data is underway



Temporal graph sample containing the pattern with the best recognition

Summary

- Presented multiple techniques to detect patterns in real-world temporal graphs
  - Unsupervised method (tensor-based clustering)
  - Supervised method (GCNN based training and subgraph recognition)
  - Summarization for visualization
- Illustrated some applications
  - Identifying coherent stories from within a sea of news articles
  - Identifying congestion patterns in traffic networks
  - Many more possible…

ευχαριστώ
Thank You!

Research was sponsored by the Army Research Laboratory and was accomplished under Cooperative Agreement Number W911NF-09-2-0053 (the ARL Network Science CTA). The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Army Research Laboratory or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation here on.