



Aalto University  
School of Science  
and Technology

## Mining temporal networks

Aristides Gionis<sup>1</sup> Polina Rozenshtein<sup>2</sup>

<sup>1</sup> Aalto University, Finland

<sup>2</sup> Nordea Data Science Lab, Finland

EDBT Summer school 2019 tutorial

September 6, 2019

# tutorial website

<https://rozenbsp.github.io/temporal-networks-tutorial>

# agenda

Part I : introduction and motivation

Part II : models of temporal networks

Part III : algorithmic frameworks

Part IV : data mining problems

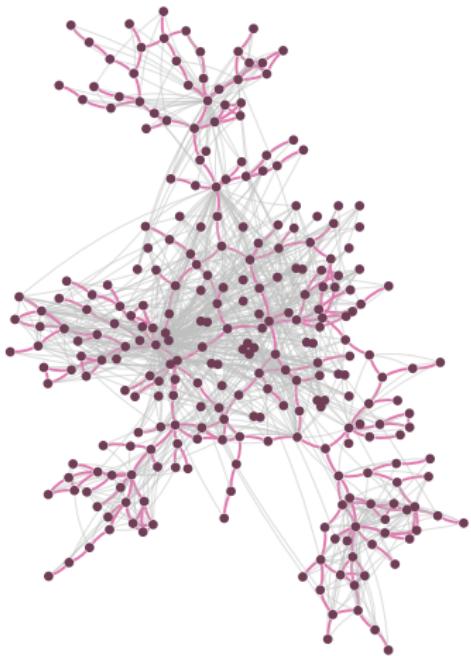
Part V : future challenges

# part I

## introduction and motivation

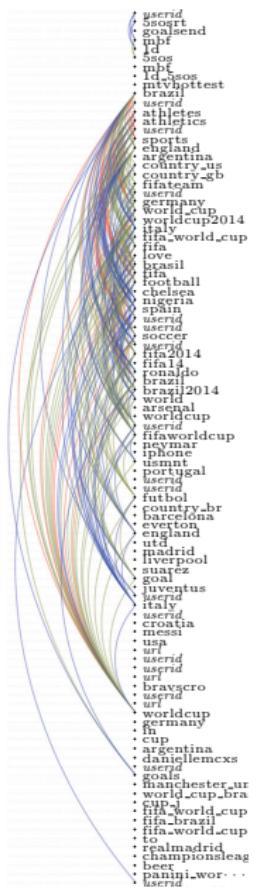
# interconnected world

- networks model **objects** and their **relations**
- many different **network types**
  - social
  - informational
  - technological
  - biological
  - ...



# impact of network science

- online communication networks and social media
  - implications in
    - knowledge creation
    - information sharing
    - education
    - democracy
    - society as a whole



# research questions in network science

- structure discovery
  - communities, summarization, events, role mining
- study complex dynamic phenomena
  - evolution, information diffusion, opinion formation, structural prediction
- develop novel applications
- design efficient algorithms

## traditional view

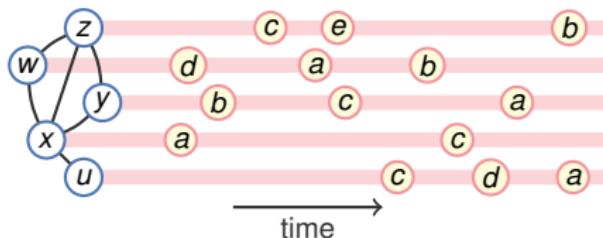
- networks represented as pure graph-theory objects
  - no additional vertex / edge information
- emphasis on **static networks**
- **dynamic** settings model **structural changes**
  - vertex / edge additions / deletions

# temporal networks

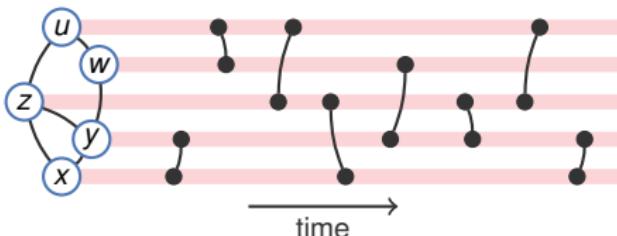
- ability to collect and store large volumes of network data
- available data have **fine granularity**
- lots of **additional information** associated to vertices/edges
- network topology is **relatively stable**, while lots of **activity** and **interaction** is taking place
- giving rise to **new concepts**, **new problems**, and **new computational challenges**

# modeling activity in networks

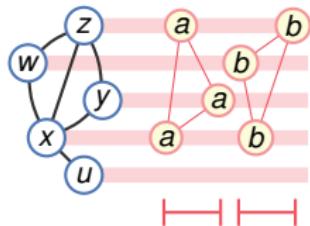
1. network nodes **perform actions** (e.g., posting messages)



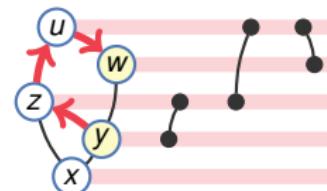
2. network nodes **interact** with each other  
(e.g., a “like”, a repost, or sending a message to each other)



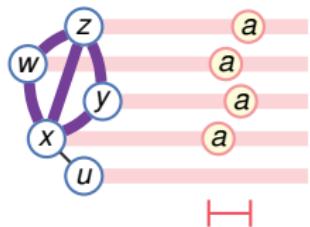
# many novel and interesting concepts



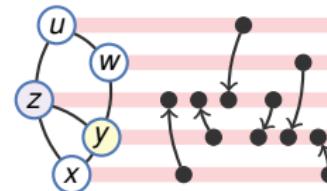
new pattern types



temporal information paths



new types of events



network evolution

## temporal networks — objectives

- identify new concepts and new problems
- develop algorithmic solutions
- demonstrate relevance to real-world applications

# terminology

- we use term “**temporal networks**”, but terminology is not standardized
- term “**X Y**” can be encountered in the literature, where

**X :**

temporal  
dynamic  
(time-)evolving  
time-varying  
time-dependent  
evolutionary

**Y :**

networks  
graphs

- some combinations have distinct meaning, but not always

# examples of temporal networks

[Holme, 2015]

- human communication networks
  - phone, email, text messages, etc.
- human proximity networks
  - recorded by various sensors and devices, e.g., bluetooth, wifi, etc.
  - patient-referral networks, i.e., how patients are transferred between wards of a hospital system
  - sexual contact networks
- animal proximity networks
  - obtained via RFID devices
  - livestock or wildlife

## examples of temporal networks — cnt'd

[Holme, 2015]

- bibliographic networks
  - collaboration and citation networks
- economic networks
  - credit card transactions
  - trade networks of countries
  - bitcoin transactions
- travel and transportation networks
  - airline connections, bus transport, bike-sharing systems

## examples of temporal networks — cnt'd

[Holme, 2015]

- brain networks
  - temporal correlations of the oxygen levels of brain regions as measured by fMRI scanning
- biological networks
  - genes involved in different interactions that change over time
  - current challenges, as one cannot measure precisely when two proteins interact with each other, but technology is improving

# agenda

Part I : introduction and motivation

Part II : models of temporal networks

Part III : algorithmic frameworks

Part IV : data mining problems

Part V : future challenges

## part II

# models of temporal networks

# representation of temporal networks

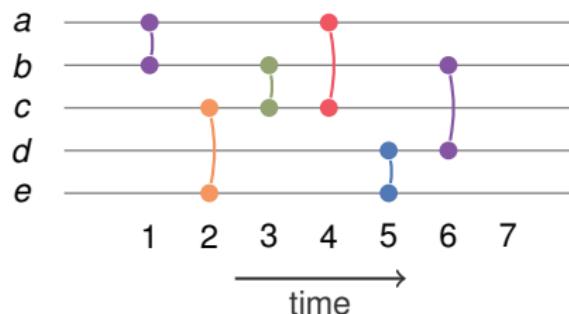
## 1. sequence of interactions

- a temporal network is represented as  $G = (V, E)$ 
  - with set of nodes  $V$ , and
  - set of edges  $E = \{(u, v, t)\}$ , with  $u, v \in V$  and  $t \in \mathbb{R}$
  - if interactions have duration, then  $E = \{(u, v, t, \lambda)\}$
- this is a lossless representation — no information is lost
- also known as sequence of contacts, or sequence of (temporal) edges

# representation of temporal networks

## 1. sequence of interactions

- visual representation of a temporal network as a sequence of interactions



# representation of temporal networks

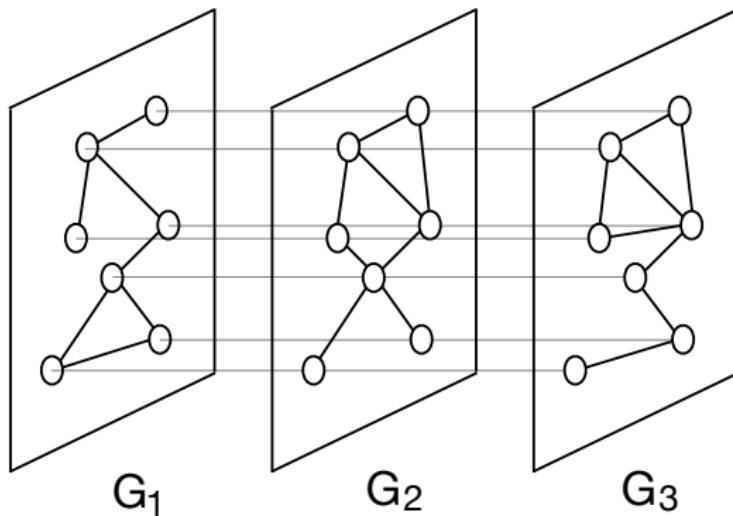
## 2. sequence of static graphs

- sequence  $G_1, \dots, G_T$ 
  - where  $G_t = (V_t, E_t)$ , with  $t = 1, \dots, T$
  - typically assume that nodes are fixed, i.e.,  $V_t = V$
  - $E_t$  are the edges that occur in time interval  $t$
- advantages: static graph analysis methods can be applied
- disadvantages: the representation assumes quantization into time intervals
  - thus, representation depends on quantization parameters, e.g., seconds, minutes, hours, days, etc.
  - coarse resolution may lead to information loss
  - fine resolution may lead to sparse (or even empty) static graphs

# representation of temporal networks

## 2. sequence of static graphs

- visual representation of a temporal network as a sequence of static graphs



# representation of temporal networks

## 3. time series of contacts

- a time-series for each pair of nodes in the network
- equivalent representation with sequence of interactions

## 4. tensor representation

- tensor representing node  $\times$  node  $\times$  time information
- can apply powerful tensor-algebra techniques
- a complication is that time is directed, while tensor algebra assumes that indices can be relabeled (breaking the time ordering)

# representation of temporal networks

[Casteigts et al., 2012]

5. time-varying graphs defined as  $G = (V, E, T, p, \lambda)$ ,

where

- $V$ : set of nodes
- $E \subseteq V \times V$ : set of edges
- $T$ : a time domain
- $p : E \times T \rightarrow \{0, 1\}$ : a presence function
- $\lambda : E \times T \rightarrow \mathbb{R}$ : a latency function

- general definition that can be used to model graph datasets in different applications
  - transportation networks, communication networks, social networks

# representation of temporal networks

## 6. stream graphs and link streams [Latapy et al., 2018]

- a formalization for modeling interactions over time
- a stream graph is defined as  $G = (T, V, W, E)$ , where
  - $T$ : a time domain
  - $V$ : a set of nodes
  - $W \subseteq T \times V$ : a set of temporal nodes
  - $E \subseteq T \times V \times V$ : a set of links
    - s.t.,  $(t, u, v) \in E$  implies  $(t, u) \in W$  and  $(t, v) \in W$
- formalization is **self-consistent** : relations between concepts are preserved
  - e.g., can define clustering coefficient using density
- formalization **generalizes** usual concepts of graph theory
  - e.g., line graphs, k-cores, cliques, density, centralities

# temporal networks vs. dynamic graphs

- **dynamic graphs** is a standard model typically studied in theoretical computer science
  - e.g., [Henzinger et al., 1999, Thorup, 2000]
- dynamic graphs are represented as a **sequence** of **edge additions** and/or **edge deletions**
- $G_0$  is the initial graph, and  $G_i$  is the graph resulting after the  $i$ -th edge addition/deletion operation
- **objective:** efficient maintenance of graph properties
  - e.g., connectivity, shortest paths, spanners, etc.

## temporal networks vs. dynamic graphs

- in dynamic-graph studies, the properties of interest refer to **individual graph snapshots**  $G_i$ , not considering the whole **graph evolution**
- emphasis on **computational efficiency**
  - computation time **per operation**
  - e.g., cost of maintaining a minimum spanning tree per edge additions/deletions
  - or, cost of maintaining a data structure that allows to answer short-path queries
- **dynamic graph** model captures **topological changes**, not interactions
  - e.g., dynamic graphs can be used to model friendship additions/deletions in a social network, but not discussions or other interactions

# temporal networks vs. dynamic graphs

- dynamic graphs resemble sequence of interactions model
- main difference lies on which graph properties we study
- for dynamic graphs we typically consider properties on graph snapshots
  - i.e., minimum spanning tree on the current snapshot
- for temporal graphs we typically consider properties that span a time interval
  - i.e., a temporal pattern
- disclaimer: in this tutorial we do not consider dynamic graphs
  - however, it is a well-developed area with rich literature

# dynamic networks

- in the context of graph generation models, we consider dynamic networks
  - e.g., Barabási-Albert, forest-fire, copying model, etc.
- similar to dynamic graphs, as data are seen as a sequence of node/edge additions (typically no deletions)
- node/edge addition are governed by a probabilistic model, not arbitrary, or worst case, as in algorithmic models
- emphasis again on network topology, i.e., how certain network structures emerge
  - e.g., scale-free distribution, small world, etc.
- disclaimer: in this tutorial we do not consider dynamic networks

# graph streams

- setting inspired by **data streams**  
[Muthukrishnan et al., 2005]
- recall the **data-stream model**:
  - data are presented as a **sequence of data items** (potentially infinite)
  - assume a **small number of passes** typically constant or just one pass
  - assume **small memory** compared to data size e.g., poly-logarithmic
  - assume **fast computation** per data item processed e.g., constant or poly-logarithmic

## graph streams

- a graph stream is a graph dataset in the data-stream model  
e.g., sequence of interactions (temporal network), or  
sequence of edge additions/deletions (dynamic graph)
- thus, a graph stream is not a representation model, instead  
it refers to the underlying computational model
- thus, we can study questions of mining temporal networks  
in the graph-stream model

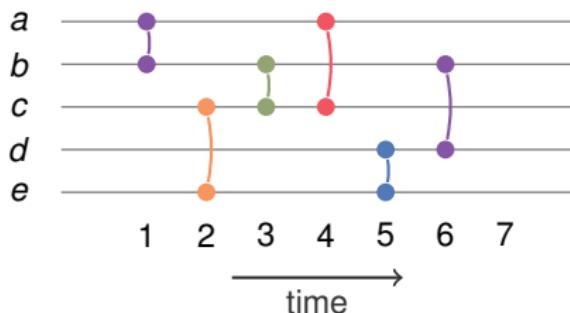
## dynamic graph algorithms on streaming model

- well-studied model
- extensive survey [McGregor, 2014]
- different settings considered
  - node/edge additions (**incremental**)
  - node/edge additions/deletions (**fully-dynamic**)
  - updating weights/labels is a special case of the fully-dynamic model
  - **sliding-window setting**: consider only edges from latest interval of fixed length
  - algorithms can be **deterministic** or randomized

## time-respecting paths

- a fundamental concept in analysis of temporal networks
  - used in studies of information propagation, or epidemics spreading
- a time-respecting path is a sequence of temporal edges, such that
  - consecutive edges share a common node, and
  - time stamps of temporal edges are non-decreasing
- intuitively, a piece of information (or disease) can propagate in the network only over time-respecting paths

## time-respecting paths — example



$(c, e, 2), (e, d, 5), (d, b, 6)$  is a time-respecting path from *e* to *b*  
 $(c, b, 3), (b, a, 1)$  is not a time-respecting path

## static expansion of a temporal network

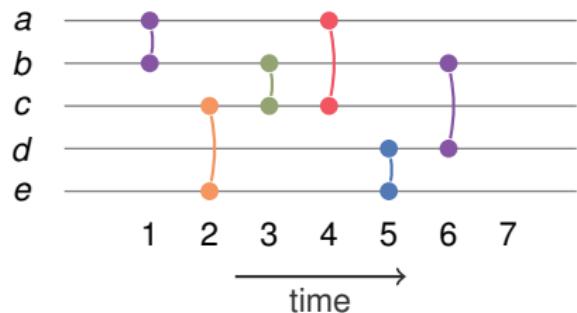
- a transformation of a temporal network to a directed (static) network so that time-respecting paths in the temporal network correspond to directed (static) paths in the directed (static) network
- how to create such a transformation?

## static expansion of a temporal network

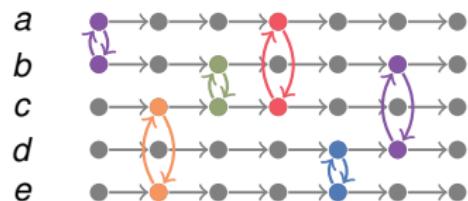
- a transformation of a temporal network to a directed (static) network so that time-respecting paths in the temporal network correspond to directed (static) paths in the directed (static) network
- how to create such a transformation?
  1. create a copy of each node for each time instance
  2. create a directed edge from the  $(t - 1)$ -th copy of  $u$  to the  $t$ -th copy of  $u$ , for all nodes  $u$ , and all time instances  $t$
  3. create directed edges for the temporal edges

# static expansion of a temporal network

example



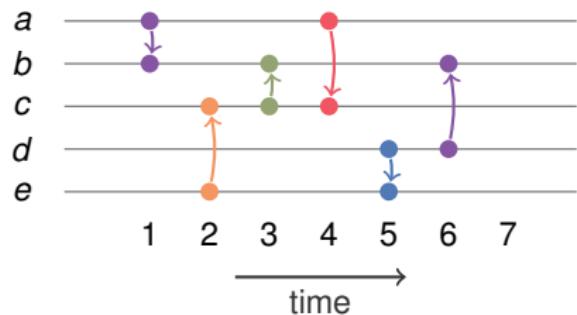
(a) representation of a temporal network



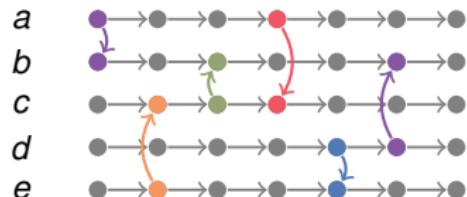
(b) static expansion of temporal network

# static expansion of a temporal network

example



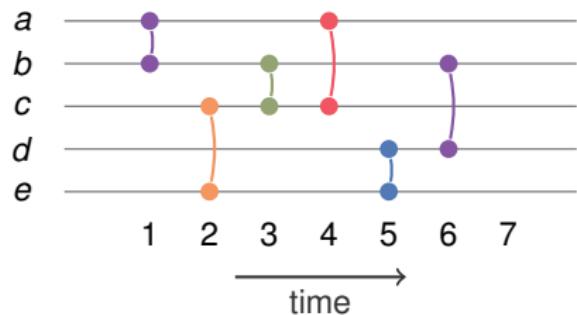
(a) representation of a temporal network



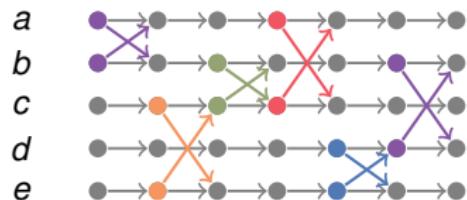
(b) static expansion of temporal network; directed edges

# static expansion of a temporal network

example



(a) representation of a temporal network



(b) static expansion of temporal network; delays

# reachability, connectivity, and connected components

- defined as in static graphs, but using time-respecting paths
- **reachability**:
  - used to study infection spreading and information cascades
- **connectivity**: as in directed (static) graphs is not symmetric
  - distinguish strong and weak connectivity
  - in addition, we can define transitive connectivity:  
a subgraph is transitively connected if time-respecting paths from  $u$  to  $v$  and  $v$  to  $w$  imply a time-respecting path from  $u$  to  $w$

## minimum temporal paths

different notions of minimum temporal paths rely on time-respecting paths

- earliest-arrival path : a path from  $x$  to  $y$  with earliest arrival time
- latest-departure path : a path from  $x$  to  $y$  with latest departure time
- fastest path : path from  $x$  to  $y$  with minimum elapsed time
- shortest path : fastest path from  $x$  to  $y$  in terms of overall traversal time required on edges

[Wu et al., 2014]

## diameter, network efficiency

- **diameter**: shortest latency of time-respecting paths over connected pairs [Chaintreau et al., 2007]
  - restricted on connected pairs, as real data have many disconnected pairs
- **network efficiency**: the harmonic mean of latency over all pairs [Tang et al., 2009]
  - **discussion**: what is the motivation for **harmonic mean**?

## diameter, network efficiency

- **diameter**: shortest latency of time-respecting paths over connected pairs [Chaintreau et al., 2007]
  - restricted on connected pairs, as real data have many disconnected pairs
- **network efficiency**: the harmonic mean of latency over all pairs [Tang et al., 2009]
  - **discussion**: what is the motivation for **harmonic mean**?
  - it combines average latency and reachability ratio

## centrality measures

- many centrality measures on static graphs use distances
- closeness centrality :  $C_c(u) = \frac{n-1}{\sum_{v \neq u} d(u,v)}$
- betweenness centrality :  $C_b(u) = \frac{\sum_{v \neq u \neq w} p_u(v,w)}{\sum_{v \neq u \neq w} p(v,w)}$
- for temporal networks we replace distance with shortest latency time-respecting path
- analogues of Katz centrality and PageRank have also been defined
- discussion : how do these centrality measures on temporal networks compare with their static analogues?

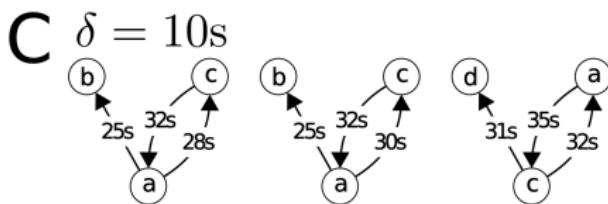
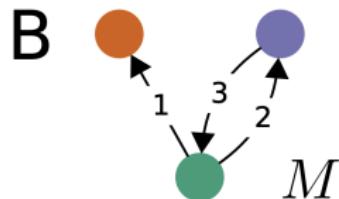
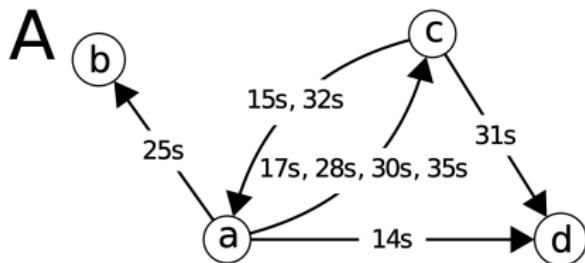
# temporal motifs

- temporal motif counting

[Paranjape et al., 2017, Kovanen et al., 2013]:

- temporal motif is a small subgraph with temporally ordered edges (and/or interval or delay constraints)

# temporal motifs



$\delta$ -temporal motif: a sequence of directed temporally ordered edges which appear within a time window  $\delta$

[Paranjape et al., 2017]

# agenda

Part I : introduction and motivation

Part II : models of temporal networks

Part III : algorithmic frameworks

Part IV : data mining problems

Part V : future challenges

## part III

algorithmic frameworks for temporal network  
analysis

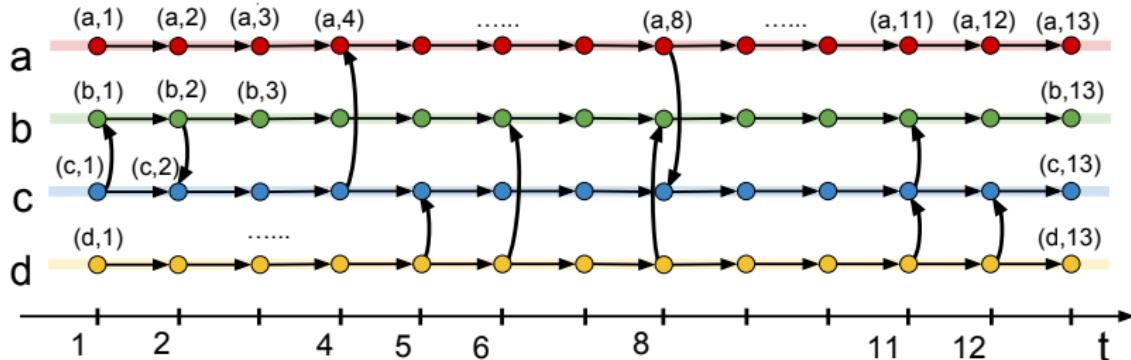
# frameworks

adopted traditional frameworks

- static expansion graphs
- dynamic graphs
- time-series
- labeled graphs

# static expansion graphs

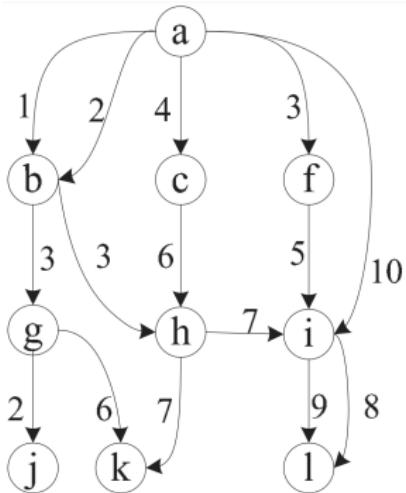
- static graph of time-stamped nodes and time-forwarding edges  $G_e = (V_e, E_e)$
- $V_e = \{(v, t) \mid v \in V, t \in T\}$ , where  $T$  is the set of all possible timestamps
- edges  $E_e$  : interactions between the temporal nodes  $V_t$



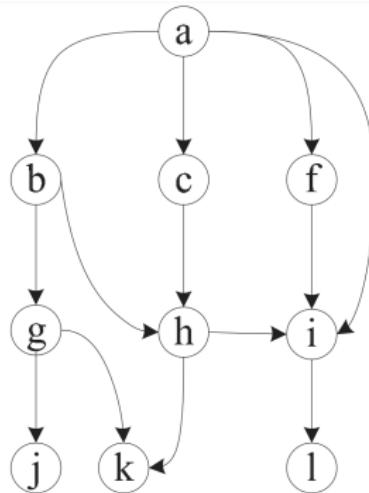
# static expansion graphs

- static expansion graph is a **directed acyclic graph (DAG)**
- **standard graph algorithms** (BFS, DFS, Dijkstra, Bellman-Ford) can be adopted for finding:
  - fastest temporal paths,  
shortest temporal paths, and  
weighted combinations
  - journeys and walks
- **upstream**, **downstream** reachability sets

## time-respecting paths



(a) Temporal Graph



(b) Static Graph

- some paths in the static graph **are not meaningful** in the temporal graph
- e.g.,  $a-b-g-j$  is not time-respecting path
- what is the shortest path from  $a$  to  $\ell$ ?

## minimum temporal paths

different notions of minimum temporal paths rely on time-respecting paths

- earliest-arrival path : a path from  $x$  to  $y$  with earliest arrival time
- latest-departure path : a path from  $x$  to  $y$  with latest departure time
- fastest path : path from  $x$  to  $y$  with minimum elapsed time
- shortest path : fastest path from  $x$  to  $y$  in terms of overall traversal time required on edges

[Wu et al., 2014]

## earliest-arrival path

- temporal graph  $G = (V, E)$
  - source vertex  $x$ , starting time  $t_s$
  - array  $T$  of size  $|V|$  to record arrival times to each node
  - $T[x] = t_s$  and  $T[v] = \infty$ , for nodes other than source
  - process edges  $(u, v, t, \lambda)$  in temporal order
    - if  $t \geq T[u]$  ( $u$  is already reached from  $x$ )
      - check if the edge creates the earliest-seen-so-far path from  $x$  to  $v$  and update  $T[v]$ :
- $$T[v] = \min(T[v], t + \lambda)$$

[Wu et al., 2014]

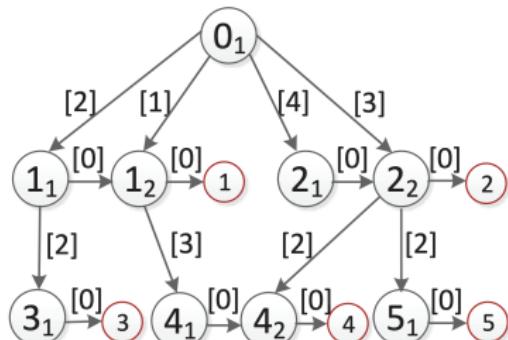
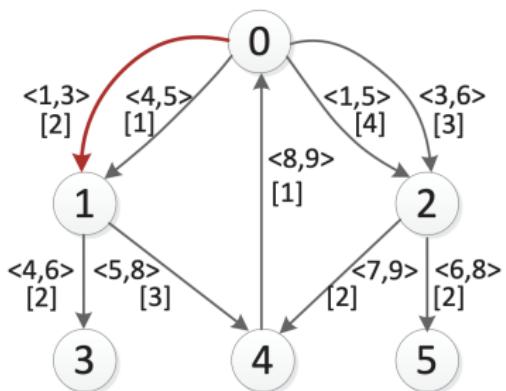
## latest-departure path

- temporal graph  $G = (V, E)$
- sink vertex  $x$ , ending time  $t_s$
- same process as for earliest-arrival path, but
- process edges in reversed temporal order
- add new interaction to the path if it does not violate temporal order

[Wu et al., 2014]

# minimum spanning trees

- $\text{MST}_a$  : minimum spanning tree with earliest-arrival times  
each temporal path from the root to the node is the earliest arrival path
- $\text{MST}_w$  : minimum spanning tree with smallest total weight  
or with the smallest number of hops: directed Steiner tree.

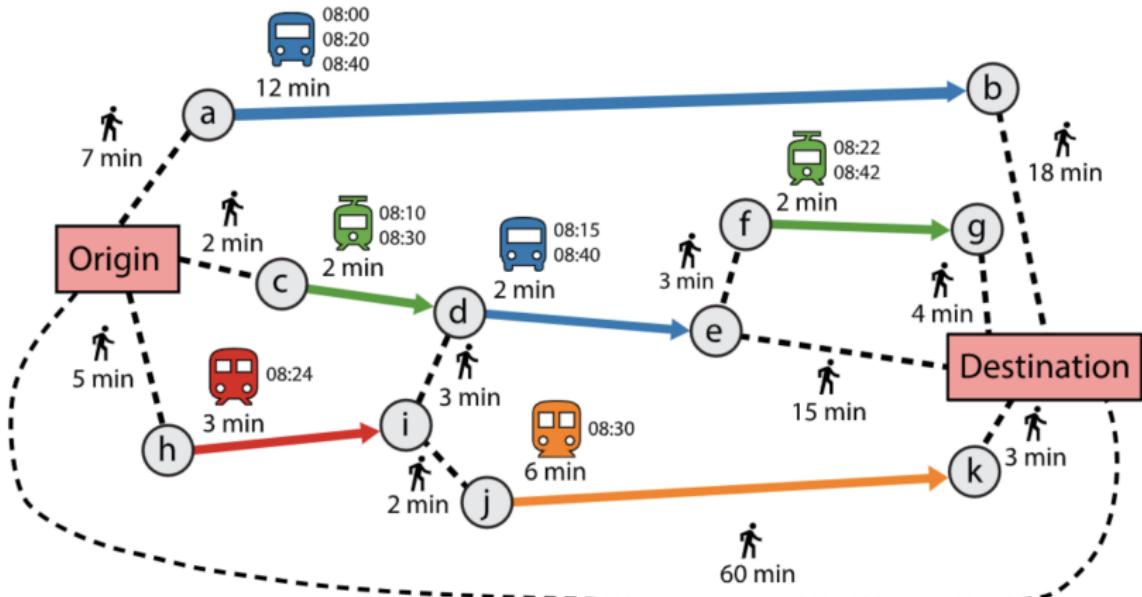


[Huang et al., 2015]

# applications of temporal paths

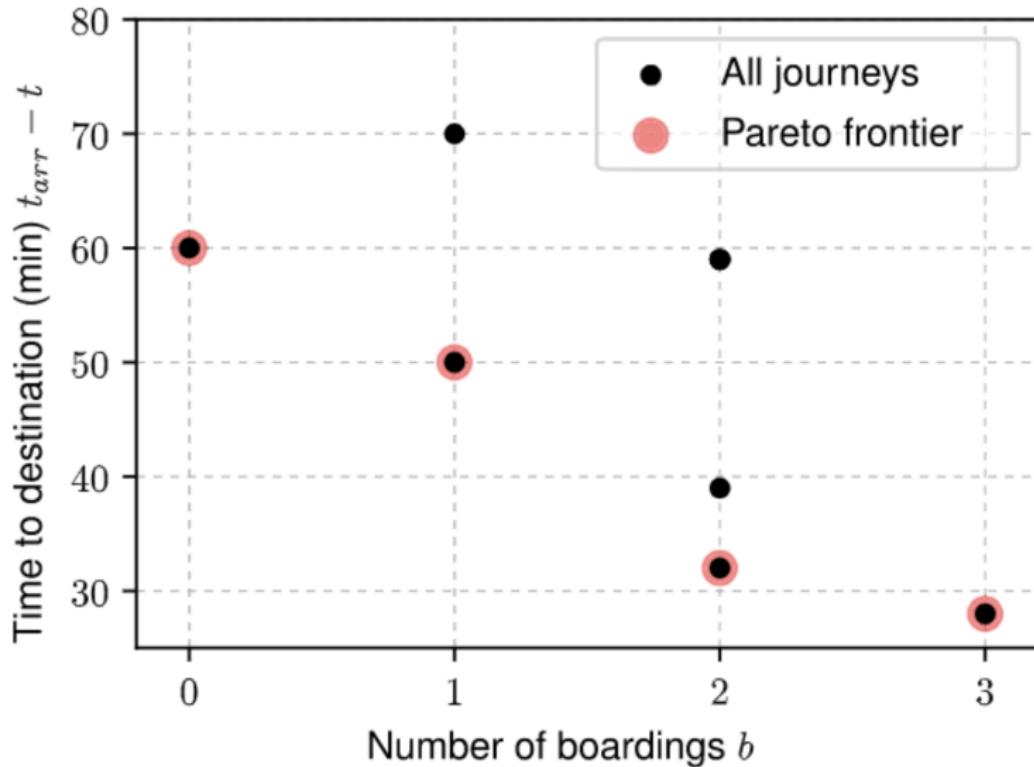
- temporal reachability problems
  - diffusion simulation, centrality measures
- directed spanning or Steiner trees
  - reconstruction of diffusion
- **drawback**: large size of expansion graph may lead to high computational complexity and large memory consumption
- **challenge**: scalable algorithms and approximations

# applications — transportation temporal networks

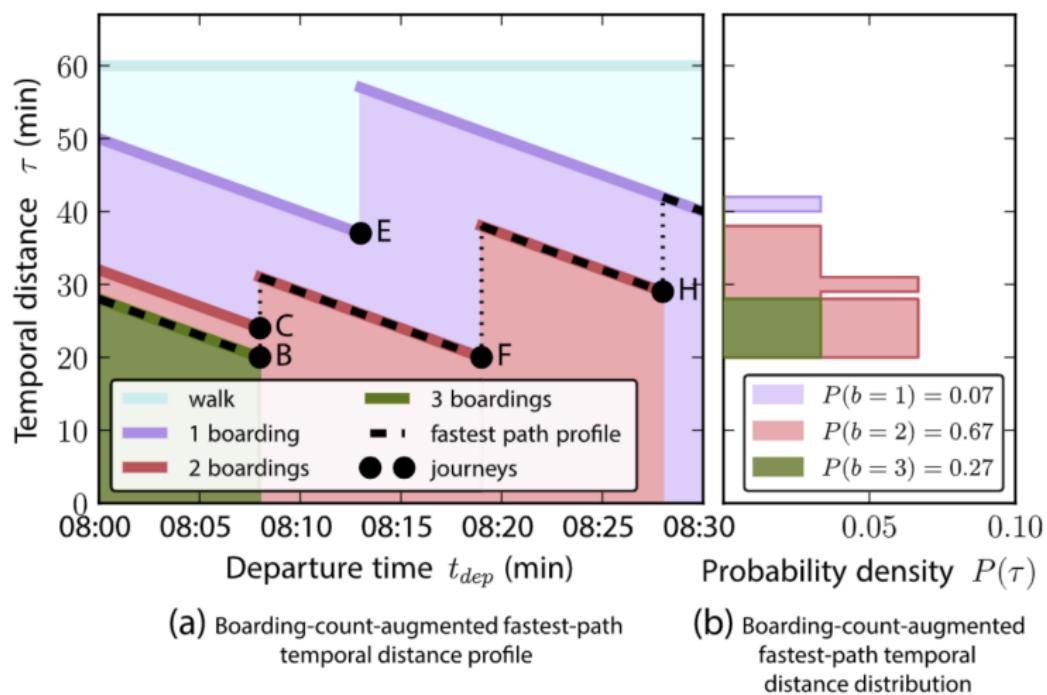


[Kujala et al., 2018]

## Pareto-optimal journeys



# Boarding-count-augmented temporal-distance profiles



## dynamic graph algorithms on streaming model

- well-studied model
- extensive survey [McGregor, 2014]
- different settings considered
  - node/edge additions (**incremental**)
  - node/edge additions/deletions (**fully-dynamic**)
  - updating weights/labels is a special case of the fully-dynamic model
  - **sliding-window setting**: consider only edges from latest interval of fixed length
  - algorithms can be **deterministic** or randomized

# dynamic graph algorithms on streaming model

[McGregor, 2014]

|                      | Insert-Only                             | Insert-Delete  | Sliding Window (width $w$ )                   |
|----------------------|---|--|---|
| Connectivity         | Deterministic [27]                      | Randomized [5]   | Deterministic [22]                            |
| Bipartiteness        | Deterministic [27]                      | Randomized [5]   | Deterministic [22]                            |
| Cut Sparsifier       | Deterministic [2, 8]                    | Randomized [6, 31]   | Randomized [22]                               |
| Spectral Sparsifier  | Deterministic [8, 46]                   | Randomized<br>$\tilde{O}(n^{5/3})$ space [7]                     | Randomized<br>$\tilde{O}(n^{5/3})$ space [22] |
| $(2t - 1)$ -Spanners | $O(n^{1+1/t})$ space [11, 23]           | Only multiple pass<br>results known [6]                          | $O(\sqrt{w}n^{(1+1/t)})$ space [22]           |
| Min. Spanning Tree   | Exact [27]                              | $(1 + \epsilon)$ -approx. [5]<br>Exact in $O(\log n)$ passes [5] | $(1 + \epsilon)$ -approx. [22]                |
| Unweighted Matching  | 2-approx. [27]<br>1.58 lower bound [42] | Only multiple pass<br>results known [3, 4]                       | $(3 + \epsilon)$ -approx. [22]                |
| Weighted Matching    | 4.911-approx. [25]                      | Only multiple pass<br>results known [3, 4]                       | 9.027-approx. [22]                            |

Table 1: Single-Pass, Semi-Streaming Results: Algorithms use  $O(n \text{ polylog } n)$  space unless noted otherwise.

# sliding-window neighborhood profiles

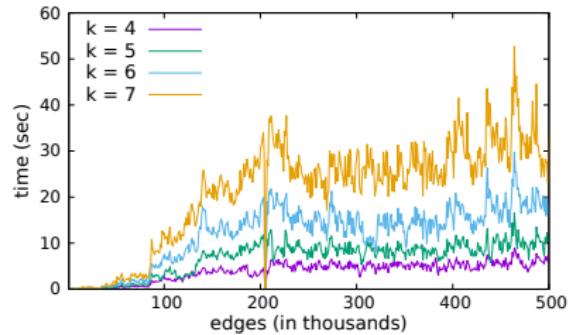
- temporal network  $G = (V, E)$
- stream of edges  $E = \langle (u_1, v_1, t_1), (u_2, v_2, t_2), \dots \rangle$   
with  $t_1 \leq t_2 \leq \dots$
- sliding window length  $w$
- snapshot network  $G(t, w)$  at time  $t$  contains all edges  
with time-stamps in  $(t - w, t]$

problem :

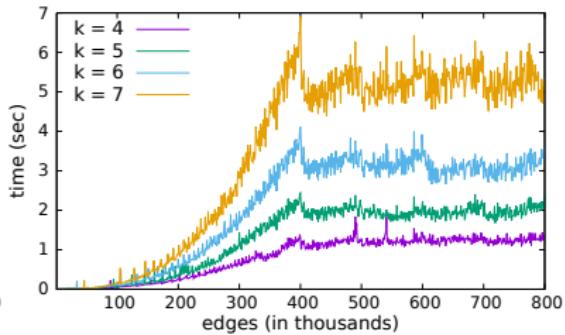
given node  $u$ , window length  $w$ , and distance  $r$ , how many  
nodes in  $G(t, w)$  are within distance  $r$  from  $u$  at time  $t$ ?

[Kumar et al., 2015]

## empirical evaluation — running time



(c) Higgs



(d) DBLP

### contrast (DBLP)

- offline HyperANF : 3.6 sec / sliding window
- proposed approach : 0.003 sec / sliding window

[Kumar et al., 2015]

# time-series analysis

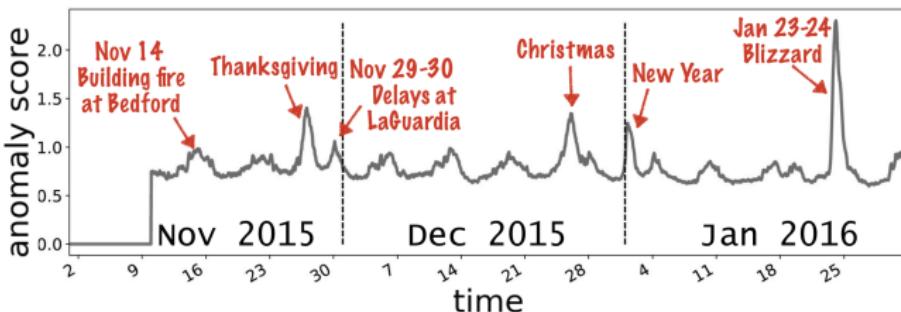
- view a temporal network as a (multivariate) time series
  - calculate **temporal profile** of nodes, edges, or a whole network
  - calculate **distance** between adjacent snapshots and analyze the resulting time series
- **distance**: edit distance, node-profile distances, vector-space distance
- applications in **change-point detection**, anomaly detection, evolutionary pattern mining

# event detection in time series

- given a sequence of graphs  $G_t$
- a function to calculate the vertex affinity matrix  $S$ , where  $s_{ij}$  indicates the influence vertex  $i$  has on vertex  $j$
- a set of time points for detected events is  
$$\{t \in T \mid d(G_t, G_{t+1}) \geq \delta\}$$

where

$$d(G_t, G_{t+1}) = \sqrt{\sum_{i=1}^n \sum_{j=1}^n (\sqrt{s_{t,ij}} - \sqrt{s_{t+1,ij}})^2}$$



## time-series analysis

- anomaly detection survey [Ranshous et al., 2015]
- approach does not solve all the problems, as it does not capture the network topology
- **possible work-around:** use more topology embeddings metrics (larger neighborhoods, influence measures, eigenvectors, . . . )

# labeled graphs

- edges are labeled with occurrence timestamps
- applications of classic graph-theoretical problems
  - coloring, routing, network flow, covering, etc.
- “any property of a graph labeled from a discrete set of labels corresponds to some temporal property if interpreted appropriately”[Michail, 2016]

## labeled graphs

- for example, consider a **proper edge coloring**
  - a coloring of the edges in which no two adjacent edges share a common color
- corresponds to a temporal network where no two adjacent edges share a common time-label
  - i.e., no two adjacent edges ever appear at the same time
- **limitation:** labels are independent, timestamps are not

# theoretical aspects of temporal graphs

- how is the complexity of classic combinatorial optimization problems changes when time is added?
- some old results: the max-flow min-cut theorem holds with unit capacities for time-respecting paths [Berman, 1996]
- a number of recent attempts
  - sliding window vertex cover [Akrida et al., 2018]
  - sliding window graph coloring [Mertzios et al., 2018]
  - maximal matching [Mertzios et al., 2019]
- etc.

## theoretical aspects of temporal graphs

- there are many models for abstracting temporal networks
- challenge: which models are most general and most useful?

# agenda

Part I : introduction and motivation

Part II : models of temporal networks

Part III : algorithmic frameworks

Part IV : data mining problems

Part V : future challenges

# part V

## data mining problems

# data mining problems

- community detection
- event detection
- finding important nodes
- epidemics analysis and influence spreading
- network summarization
- ...

community detection

# community detection in static graphs

- static graphs: extensive survey [Fortunato and Hric, 2016]
- standard community definitions
  - a community is a set of nodes, which are closer to each other than to the rest of the network
  - a community is a dense network subgraph
- general definition [Coscia et al., 2011]
  - a community in a complex network is a set of entities that share some closely correlated sets of actions with the other entities of the community
- typical problem settings
  - a single community vs. network partition
  - overlapping vs. non-overlapping communities

# community detection in static graphs

## partition measures

- modularity : the difference between the actual number of edges and the expected
- cut : number of edges between a community and the rest of the graph
- ratio cut : cut normalized by the number of edges of community nodes
- ...

## single-community measures

- average degree :  $\frac{|E(S)|}{2|S|}$
- density :  $\frac{2|E(S)|}{|S|(|S|-1)}$
- conductance :  $\frac{\text{cut}(S, \bar{S})}{\min\{\text{vol}(S), \text{vol}(\bar{S})\}}$
- ...

# community detection in temporal networks

temporal information gives rise to several issues

- **temporal localization**: concise time interval or intervals, whole time history
- **behaviour**: single-appearance, recurring, persistent, evolutionary patterns, smoothness
- partition of the **topology network** vs. partition of the **time history**
- **online** vs. **offline**
- application-specific settings

# community detection in temporal networks

- proposed taxonomies
  - [Aynaud et al., 2013]
  - [Aggarwal and Subbian, 2014]
  - [Enugala et al., 2015]
  - [Renaud and Naoki, 2016]
  - [Hartmann et al., 2016]
  - [Rossetti and Cazabet, 2018]
  - [Dakiche et al., 2019]
  - ...

## temporal communities : temporal assumptions

prior model, which describes what is the temporal behavior of interesting community structures, e.g.,

- small/large covering intervals of community interactions
- frequent patterns
- persistent patterns

# evolutionary patterns : vocabulary

evolutionary patterns of communities in the network

[Dakiche et al., 2019]

- birth
- death
- growth
- contraction
- merge
- split
- continue
- resurgence

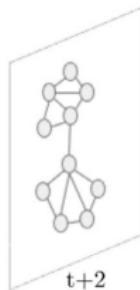
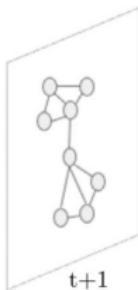
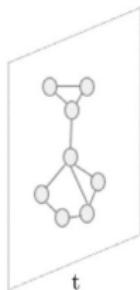
## temporal communities: taxonomy

we follow a recent survey on community detection

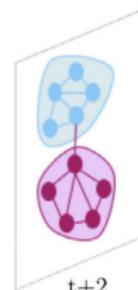
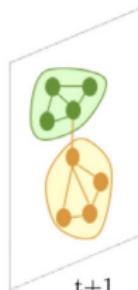
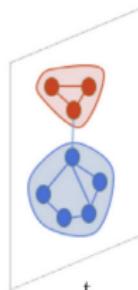
[Dakiche et al., 2019]

- independent community detection and matching
  - first detect communities at each timestamp
  - then match them across different timestamps

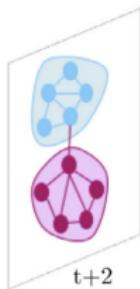
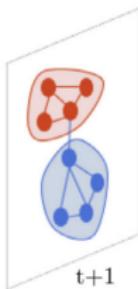
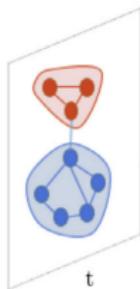
# independent community detection and matching



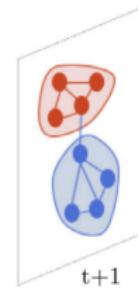
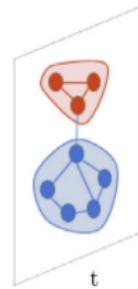
(1) A dynamic network consisting of three snapshots



(2) Community detection in each snapshot



(3) Match communities of  $t$  and  $t+1$



(4) Match communities of  $t+1$  and  $t+2$

$$\begin{array}{c} \bullet = \bullet \\ \bullet = \bullet \end{array}$$

$$\begin{array}{c} \bullet = \bullet \\ \bullet = \bullet \end{array}$$

# typical evolutionary patterns



$t + \tau$

$C_1(t + \tau)$

$t$

a) Growth



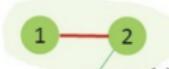
$C_1(t)$

$t + \tau$

$C_1(t + \tau)$

$t$   $t + \tau$

b) Contraction



$t + \tau$

$C_1(t + \tau)$

$t$

$C_1(t)$

c) merging



$C_1(t)$

$t + \tau$

$C_1(t + \tau)$

$t$   $t + \tau$

d) splitting



$t + \tau$

$C_2(t + \tau)$

$t$

$C_2(t)$

e) Birth

$C_1(t)$

$t$

$t + \tau$

$C_1(t + \tau)$

$C_1(t)$

$C_1(t)$

$t + \tau$

$C_1(t)$

$C_1(t + \tau)$

$C_1(t + \tau)$

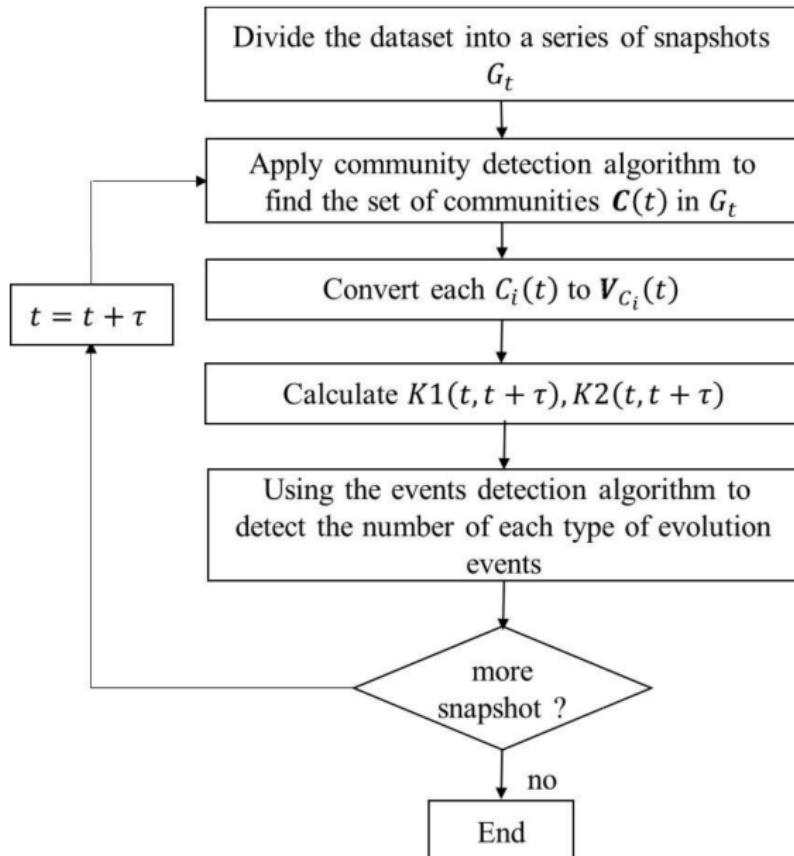
$t + \tau$

$C_1(t)$

$C_1(t)$

[Sun et al., 2015]

# procedure



# independent community detection and matching

## advantages

- reuses unmodified traditional community detection methods
- possible to use existing similarity measures

## disadvantages

- instability of community-detection algorithms

# temporal communities: taxonomy

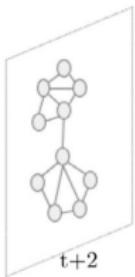
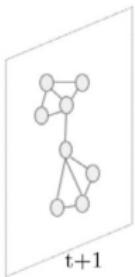
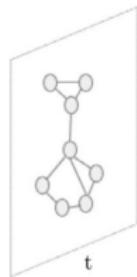
[Dakiche et al., 2019]

- **dependent community detection**

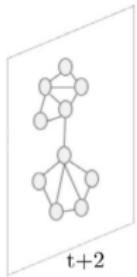
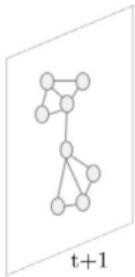
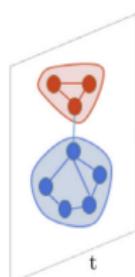
detect communities at time  $t$  based on

- network topology at  $t$ , and
- communities at time  $t - 1$

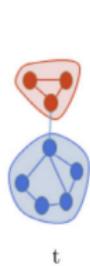
# dependent community detection



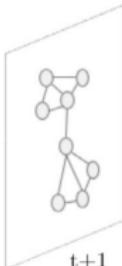
(1) A dynamic network consisting of three snapshots



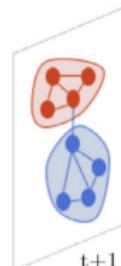
(2) Community detection in the first snapshot



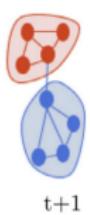
and



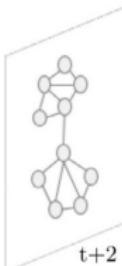
→



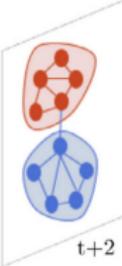
(3) Community detection at  $t+1$  using graph of  $t+1$  and communities of  $t$



and



→



(4) Community detection at  $t+2$  using graph of  $t+2$  and communities of  $t+1$

## Louvain algorithm

- a fast greedy approach based on modularity optimization
- two phases repeated iteratively
  - initially, each node in network is a community
  - then, for each node  $i$ , consider its neighbor  $j$  and compute the gain of modularity of putting  $i$  into the community of  $j$
  - node  $i$  is placed into the community with the largest gain, if the gain is positive

[Blondel et al., 2008]

## Louvain algorithm

- on the **second phase**, each community is considered as a super-node
  - the edges between these super-nodes are contracted and re-weighted by the number of edges between them
- the two phases are repeated until there is **no improvement** in modularity
- the algorithm is **extremely fast**

[Blondel et al., 2008]

## history-dependent approach

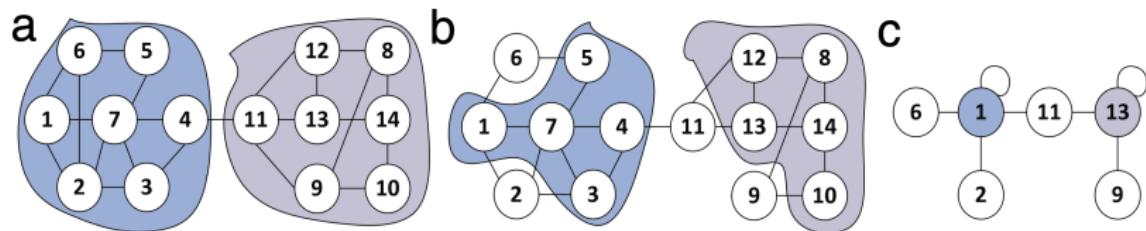
### idea

- for two consecutive time steps, there only few edges that affect the community structure
- if the connections of all the nodes in the same community at time step  $t - 1$  keep unchanged at time step  $t$ , they are still in the same community at time step  $t$
- thus, no need to break that super-node

[He and Chen, 2015]

## history-dependent approach

- find all communities in snapshot  $t = 1$
- for  $t = 2$ :
  - if a node's connection change, then remove it from its super-node and add as single node
  - leave all other nodes inside the super-node
  - re-weight the edges



[He and Chen, 2015]

# dependent community detection

## advantages

- a solution for the problem of instability
- improved computational complexity

## disadvantages

- traditional community detection methods are no longer directly applicable

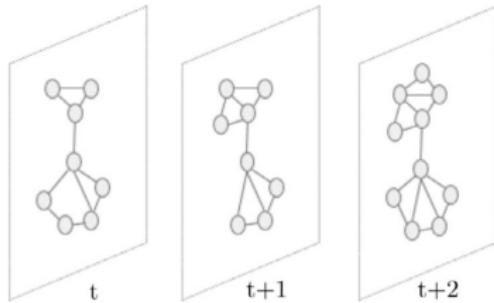
## temporal communities: taxonomy

[Dakiche et al., 2019]

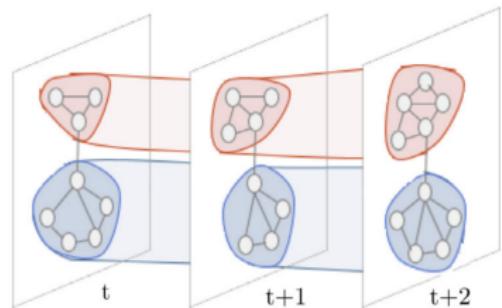
simultaneous community detection on all snapshots

- construct a static expansion graph
  - add edges between instances of nodes in different timestamps
- run a standard community detection on the resulting graph

# simultaneous community detection on all snapshots



(1) A dynamic network consisting of three snapshots



(2) Community detection on all snapshots

[Dakiche et al., 2019]

## simultaneous community detection

- algorithm based on some basic **assumptions** about individual behavior and group membership
- assumptions
- **gradual changes** : nodes change community affiliation infrequently
  - **reliable true positive** : members of the same community mostly interact with each other
  - **negligible false positive** : members of different communities rarely interact with each other

[Tantipathananandh and Berger-Wolf, 2011]

# simultaneous community detection

## costs

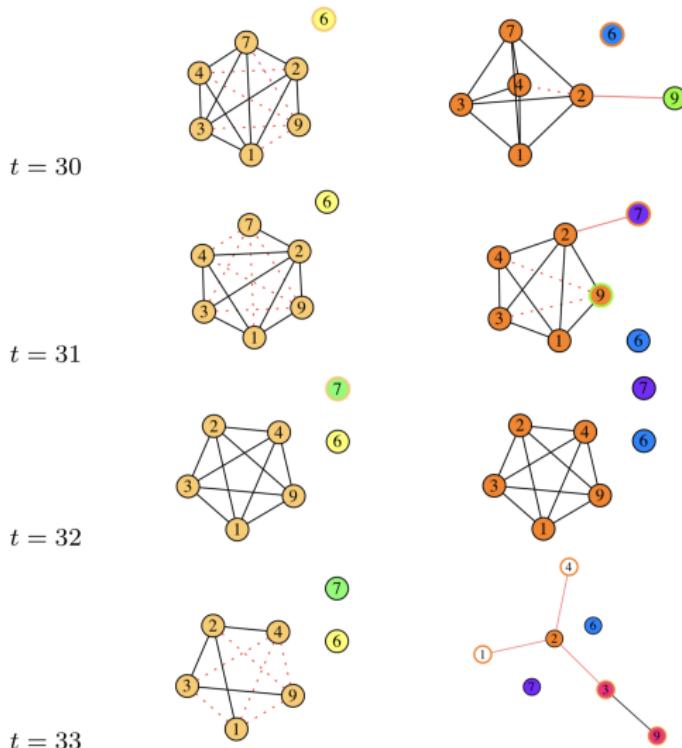
- switching cost: each node  $u$  incurs cost  $C_{sw}$  when changing community affiliation
- false negative cost: two nodes incur cost  $C_{fn}$  when belong to the same community but do not interact
- false positive cost: two nodes incur cost  $C_{fp}$  when belong to different communities but do interact

## resulting problem

- find a partition into clusters that minimizes the total cost of switching, false negative, and false positive

[Tantipathananandh and Berger-Wolf, 2011]

# simultaneous community detection



$$(C_{sw}, C_{fn}, C_{fp}) = (5, 1, 5) \text{ vs. } (C_{sw}, C_{fn}, C_{fp}) = (1, 1, 5)$$

[Tantipathananandh and Berger-Wolf, 2011]

## simultaneous community detection on all snapshots

### advantages

- provides a solution for the problem of instability

### disadvantages

- no possibility to track community evolution in a network evolving in real time

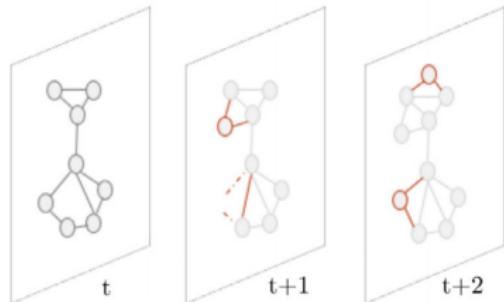
## temporal communities: taxonomy

[Dakiche et al., 2019]

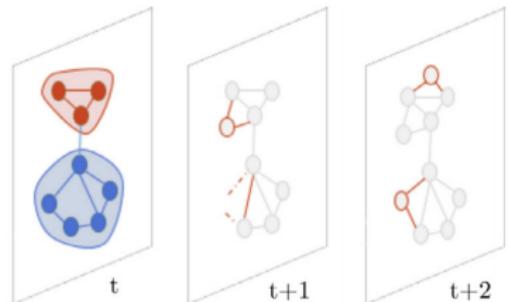
### dynamic community detection

- update previously discovered communities according to network modifications

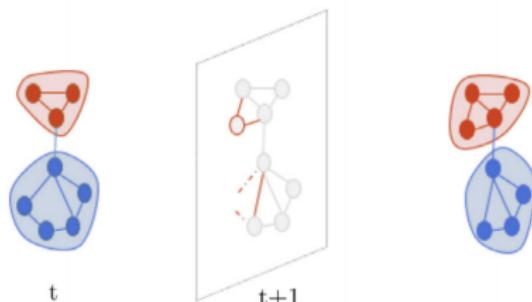
# dynamic community detection



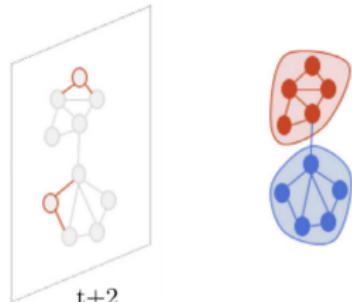
(1) Temporal network: an initial snapshot and sequence of modifications



(2) Community detection on first snapshot



(3) Update communities of t according to modifications of t+1



(4) Update communities of t+1 according to modifications of t+2

# dynamic community detection

## advantages

- provides a solution for the problem of instability
- light-weight methods to track communities

## disadvantages

- possibility to drift towards invalid communities

event detection

## event detection

- given a network representing some kind of activity
  - network of social interactions
  - social-media feed
  - transportation network
- an event can be generally defined as an activity with some prominent **qualitative** or **quantitative difference** from the **background activity**
  - bursting news about major natural disasters
  - abnormally high traffic in the city
  - an emerging new discussion topic in social media

# applications

- news spread in social media faster than in traditional news media [Sakaki et al., 2010, Dou et al., 2012]
- weather or traffic condition warning systems
- early notification about influential social events
- understanding causal relations, semantics, and dynamics of what is happening

comprehensive survey on event detection in dynamic networks  
[Ranshous et al., 2015]

## temporal event detection

- identify **atypical** time intervals and/or time instances
- temporal records
  - time sequences (time-ordered records) or
  - time series (equally-spaced in time sequences)
- number of interactions, tweets, reposts, purchases, check-ins, or some other measures in absolute values or aggregated per time unit

## temporal event detection

- time series may represent a temporal network
  - **topological** characteristics of each snapshot
  - **distance** between two consecutive graph snapshots

# temporal event detection: standard approaches

## abnormality score

- the likelihood that an interval contains an event can be estimated by comparing an abnormality score on the interval

[Heins and Stern, 2014]

## predictive models

- learn a predictive model and find intervals and time points whose behavior differ from the predicted one

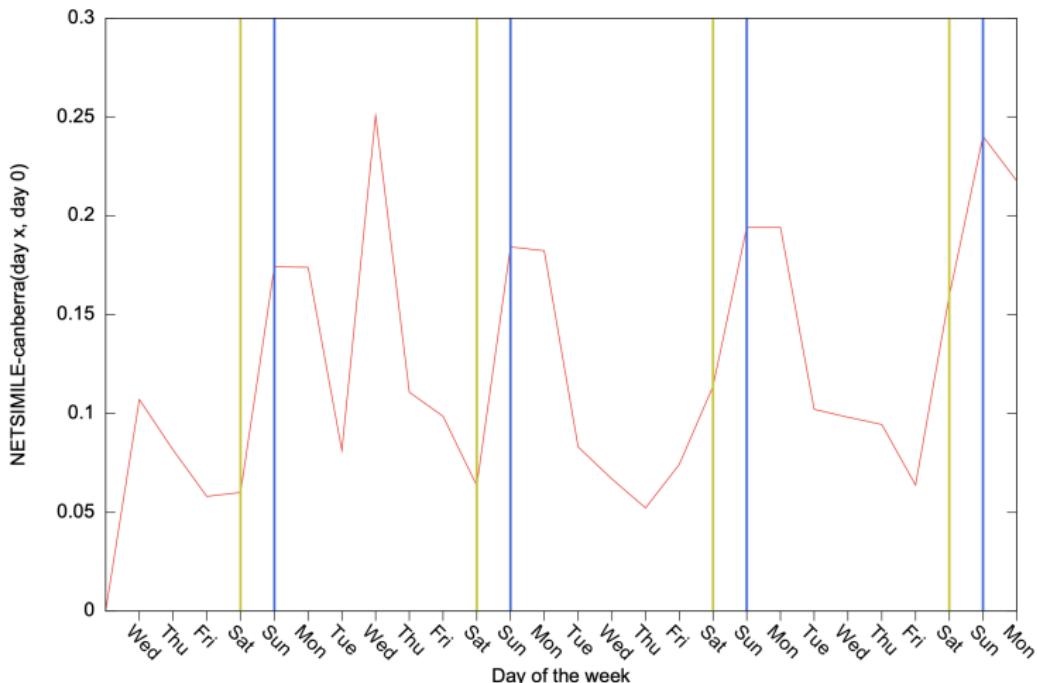
[Hunter and McIntosh, 1999, Gensler and Sick, 2017]

## Netsimile

- an event exists in  $G_{j+1}$ , if  $G_{j+1}$  is very different than  $G_j$
- for each node calculate 7 local and egonet-based measures
  - degree
  - clustering coefficient
  - average degree of neighbours
  - average clustering coefficient of neighbours
  - number of edges in the egonet
  - number of edges outgoing from the egonet
  - number of neighbours of the egonet
- combine into a signature vector and compare

[Berlingerio et al., 2012]

# Netsimile algorithm



(a) NetSimile between each day and day 0 in Yahoo! IM

# spatiotemporal event detection

detailed survey [Shi and Pun-Cheng, 2019]

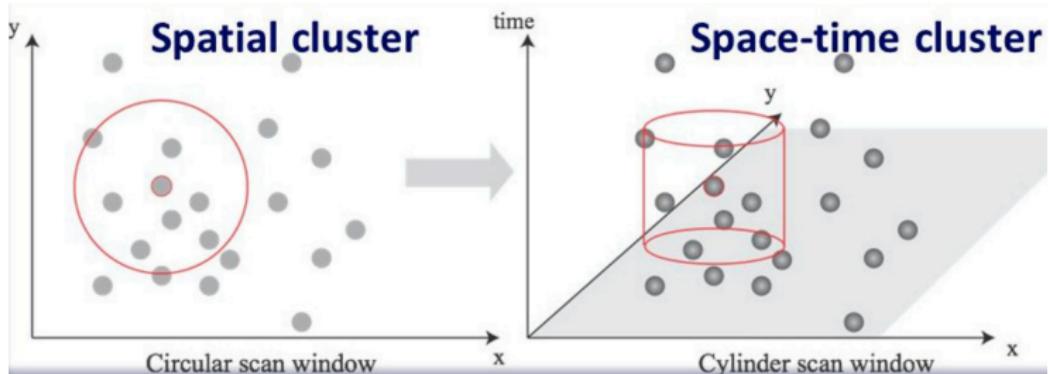
- consider time and the (geo-)location of an event
- sources of spatial data
  - GPS devices / smart phones
  - geo-tagged messages in online social networks
- typical approaches model the data as a set of geo-locations associated with activity measurements
- given a set of locations with activity measures, find a subset of locations that are close to each other and have abnormal activity pattern
- in spatiotemporal setting, one is also interested in finding the time interval (moment) of an event

## spatiotemporal event detection: scan statistics

- a classic family of methods is **spatial** and **spatiotemporal scan statistics**
- **scan** over the **space** and **time** windows to identify regions of data generated by some process

# spatiotemporal event detection: scan statistics

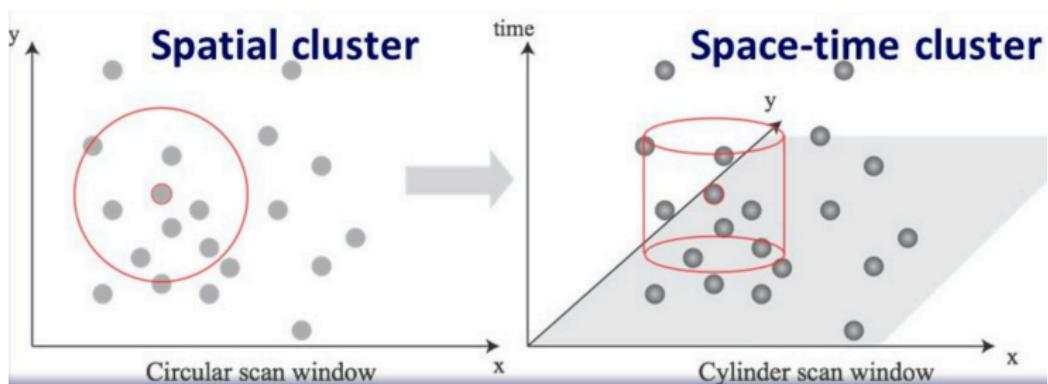
- a seminal paper : **spatial scan statistics** [Kulldorff, 1997]
- scan a circular spatial window and test the non-randomness of data against Poisson or Bernoulli baseline process



[Takahashi et al., 2004]

# spatiotemporal event detection: scan statistics

- later the approach was extended to **spatiotemporal** scans with **cylindric** windows
- similar works explore **other types** of statistics and tests [Neill, 2006, Qian et al., 2014].



[Takahashi et al., 2004]

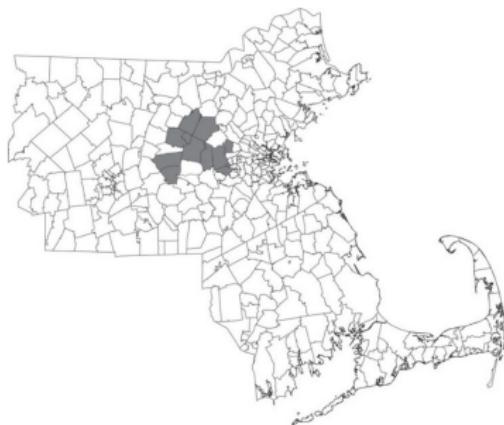
## flexible scans

- flexible spatial scan-statistics
- first, divide the entire area into many small regions
  - the location of each region is the administrative population centroid
- next, the set of irregularly shaped windows: concentric circles and connected regions
  - $k$  is a pre-specified maximum length of cluster
- similar idea is used in the flexible space-time scan statistics
- both of these are fitted to a small cluster size

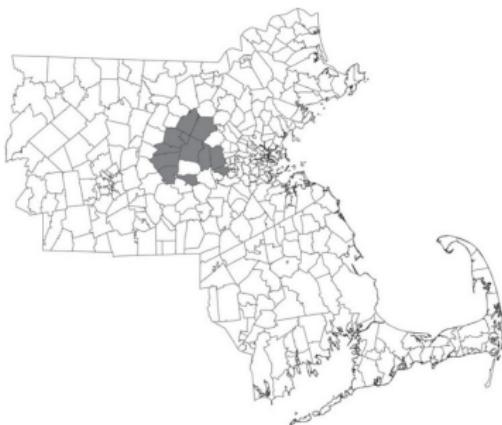
[Takahashi et al., 2008]

# flexible scans

simulated disease maps in the Tokyo Metropolitan area



(c) Respiratory (flexible on Aug.12)



(d) Respiratory (flexible on Aug.15)

[Takahashi et al., 2008]

## structural event

- structural event:
  - set of interconnected abnormal nodes
  - no assumptions on geodesic distances
- e.g., the edge weights represent similarity of nodes
  - similarities between twitter users in preferences, language, frequently visited locations, etc.
- scan extension to graph model [Liu et al., 2016b]
- scan through a graph neighborhood — a set of interconnected nodes
- dense subgraph detection
  - e.g., [Charikar, 2000, Khuller and Saha, 2009]

## semantic event detection

- define event as an emerging/bursting/unusual topic in social media, or
- use textual information to supplement and support event detection
  - meaning of the event
  - more robust event detection
- simplest use of textual information monitor the frequencies of separate key words [Lappas et al., 2012]
- efficient for predefined events, vocabulary is known
- more general approach: topic modeling to identify the event vocabulary
- combine with other event-related information
  - e.g., the geo-tags of tweets

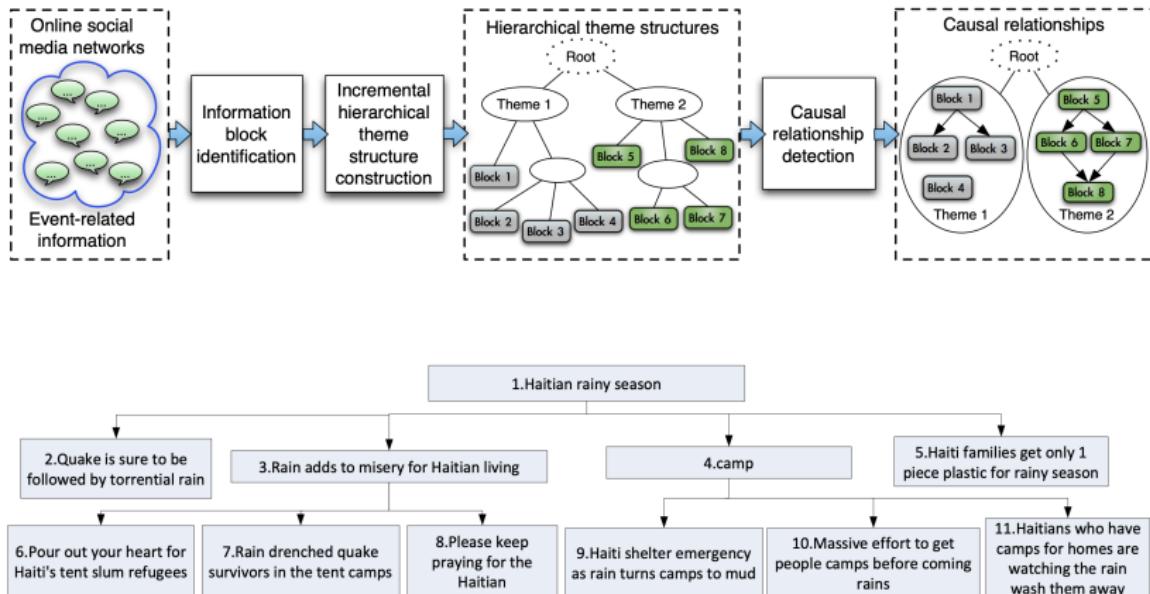
[Hong et al., 2012, Kling et al., 2014]

## ETree

- aggregate **semantically similar** (based on  $n$ -grams) tweets into information blocks
- model an event in twitter as a **tree** of **information hierarchy**, where nodes are subtopics
- each subtopic is a directed graph of **information blocks**, where edges are potential causal relationships
- the causal estimates rely on content **similarity** and **temporal** relevance
- **assemble** a topic tree by greedy heuristic

[Gu et al., 2011]

# ETree



[Gu et al., 2011]

finding important nodes

# PageRank

- classic approach for measuring node importance
- listed in the top-10 most important data-mining algorithms

[Wu et al., 2008]

- numerous applications
  - ranking web pages
  - trust and distrust computation
  - finding experts in social networks
  - ...

## static PageRank

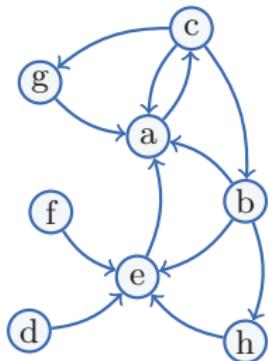
- graph  $G = (V, E)$
- corresponding row-stochastic matrix  $P \in \mathbb{R}^{n \times n}$
- personalization vector  $\mathbf{h} \in \mathbb{R}^n$
- PageRank is the **stationary distribution** of a random walk, with restart probability  $(1 - \alpha)$

$$\pi(u) = \sum_{v \in V} \sum_{k=0}^{\infty} (1 - \alpha) \alpha^k \sum_{\substack{z \in \mathcal{Z}(v, u) \\ |z|=k}} h(v) \Pr[z \mid v]$$

where,  $\mathcal{Z}(v, u)$  is the set of all paths from  $v$  to  $u$

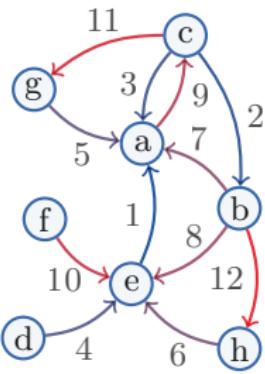
and  $\Pr[z \mid v] = \prod_{(i,j) \in z} P(i, j)$

## motivating example



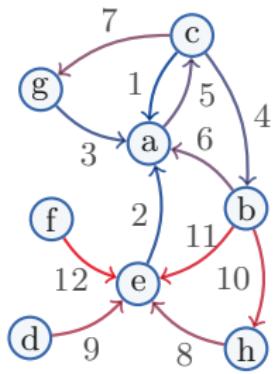
(a)

static network



(b)

temporal network

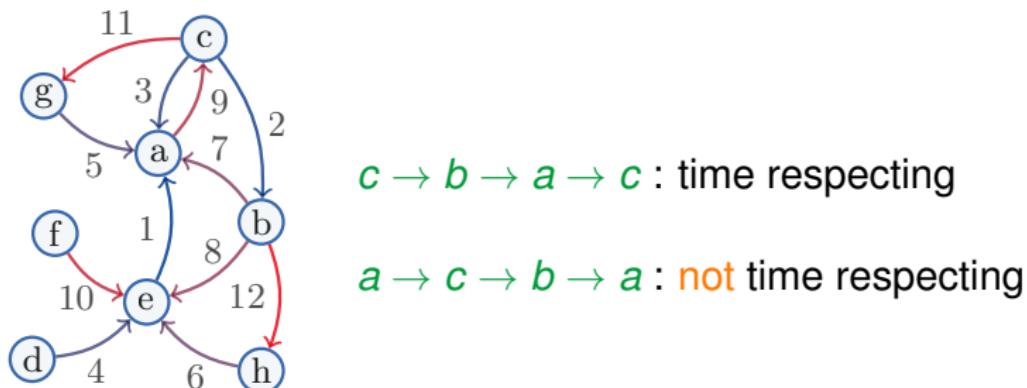


(c)

temporal network

# temporal PageRank

- make a random walk only on temporal paths
  - e.g., time-respecting paths
  - time-stamps increase along the path



# temporal PageRank

- intuition : probability of visiting node  $u$  at time  $t$  given a random walk on temporal paths
- need to model probability of following next temporal edge
  - we use an exponential distribution
- temporal PageRank definition

$$r(u, t) = \sum_{v \in V} \sum_{k=0}^t (1 - \alpha) \alpha^k \sum_{\substack{z \in \mathcal{Z}^T(v, u | t) \\ |z|=k}} \Pr'[z | t]$$

$\mathcal{Z}^T(v, u | t)$  set of temporal paths from  $v$  to  $u$  until time  $t$

## static vs. temporal PageRank

- computation:  
simple online algorithm iterating over edges
- temporal PageRank is designed to capture changes  
in network dynamics and concept drifts
- proposition :  
if the edge distribution is stable, then  
as  $T \rightarrow \infty$ , the temporal PageRank on  $G$   
converges to the static PageRank on  $G_S$ ,  
with personalization vector equal to weighted out-degree

[Rozenshtein and Gionis, 2016]

diffusion analysis and influence spreading

# diffusion analysis and influence spreading

- propagation models
  - used to study disease spreading or information cascade in the network
- activity spreading: virus, information, idea, rumor
- applications: epidemiology, information security, marketing
- why use models?
  - facilitate mathematical analysis of propagation processes
  - have intuitive interpretation
  - proven to be realistic by empirical studies
- extensive survey in the book [Shakarian et al., 2015]

# standard models

- susceptible-infected (SI) model
  - SIR, SIRS, other variants
- independent cascade (IC) model
- linear threshold (LT) model
- shortest path (SP) model

## static models: assumptions

- all models have similar implicit assumptions on temporality:
  1. uniform time steps
  2. interactions happen at each time step and are independent

## drawbacks of static models

- large heterogeneity in the time instances of real interactions

[Barabasi, 2005, Candia et al., 2008,  
Leskovec and Horvitz, 2008]

- burstiness in communication patterns
- periodic activity changes
- causal relationships between interactions

## temporal propagation models

- intuitive **extensions** from **static graphs** to temporal graphs
- add distributions (e.g., Poisson or power-law) of the **intervals between interactions** (latencies)

[Vazquez et al., 2007, Min et al., 2011]

- **realistic generalizations** of well-studied models

[Karsai et al., 2011, Candia et al., 2008]

- continuous time, partially observed graph
- develop **mathematical analysis** for novel and generalized models

[Harris, 2002, Fernández-Gracia et al., 2011]

## typical problem formulations

- immunization strategies
- influence maximization
- seed and cascade reconstruction

## static immunization strategies

- how to stop or prevent a viral diffusion?
- main aspects differentiating the research works:
  - assumptions about the spreading model
  - assumptions about the network structure
  - whether the whole network is observable
- both assumptions on the network structure and on the infection propagation are crucial
- results may not hold for any general network and real infection

[Newman, 2003, Pastor-Satorras and Vespignani, 2002a].

## static immunization strategies

- simple model-blind strategies, such as random immunization, perform moderately well in different scenarios  
[Pastor-Satorras and Vespignani, 2002b, Madar et al., 2004]
- better results on real-world networks can be achieved by immunizing nodes with high connectivity  
[Pastor-Satorras and Vespignani, 2002b, Dezső and Barabási, 2002].
- requires explicit knowledge of the network structure and it is impractical for real applications

## static immunization strategies

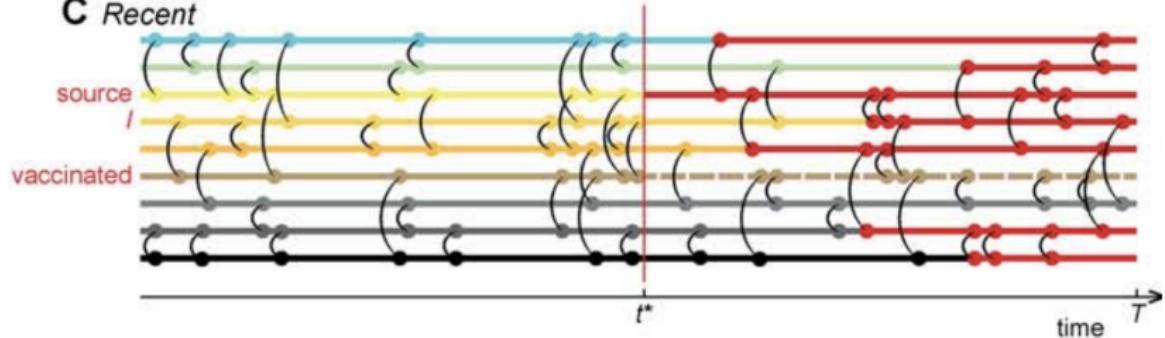
- [Cohen et al., 2003] overcomes this drawback by employing acquaintance immunization strategy:
- immunization of random neighbors of randomly selected nodes leads to immunization of the most central nodes without knowing any global information about the network

# temporal immunization strategies

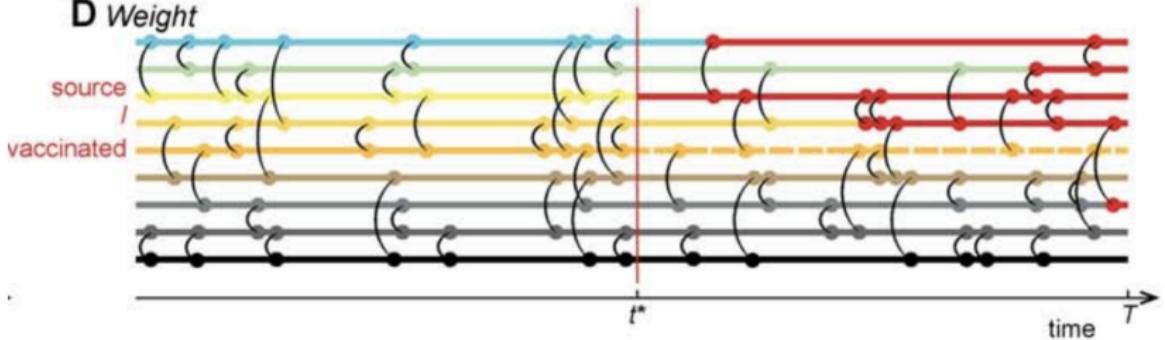
- adjust successful static strategies
- e.g., Cohen's neighborhood vaccination scheme  
[Lee et al., 2012]
- two vaccination strategies
- recent:
  - ask a random individual  $i$  to name its most recent contact and vaccinate this person
- weight:
  - ask a random individual  $i$  to name its most frequent contact since some time  $t$

## 2 protocols

C Recent



D Weight



## static influence maximization

- how to select the initial set of infected nodes (**seeds**), such that the speed, size, or other spread characteristics are optimized
- applications in **marketing** and **network design**
- influence maximization problem was introduced by [Kempe et al., 2003] in the **IC** and **LT** models
- find a set of  **$k$  seed nodes**, such that the expected number of nodes activated by the infection cascade is **maximized**

## static influence maximization

- NP-hard [Kempe et al., 2003]
- simple greedy algorithm with approximation guarantee
- influence maximization problem was been studied for many different variants of other models, constraints, and objective functions
- many practical heuristics and approximations

[Chen et al., 2009, Chen et al., 2010, Tang et al., 2014]

# temporal influence maximization

- **intuitive** approach to reflect temporality:
  - sequence of graphs (or snapshots)
  - each **time step** of propagation corresponds to propagation over the **corresponding graph**
  - all interactions within one time step happen **simultaneously**
- related papers:

[Aggarwal et al., 2012, Zhuang et al., 2013,  
Gayraud et al., 2015]

## temporal influence maximization

- another approach:
- incorporate time into the diffusion model as distribution of intervals between the interactions
- different types of models and interval distributions
  - [Chen et al., 2012, Liu et al., 2012, Rodriguez and Schölkopf, 2012, Du et al., 2013]
- the most realistic approachable setting?
- the latest promising research:
  - infer propagation model parameters from the data

[Rodriguez et al., 2011, Gomez-Rodriguez et al., 2016]

## seed and cascade reconstruction

- given some observed data about the infection
  - e.g., a small subset of infected nodes, the goal is to find the most probable seed nodes
- other versions:
  - find the most probable cascades
- the order of infection (who got infected from whom)
- these works are data-driven:
  - it is essential that the assumed propagation model matches the actual infection flow in the network

# seed and cascade reconstruction

- applications:
  - epidemiology (who was the patient zero?)
  - influencer discovery  
(who was the source of information?)
- a number of different approaches
  - find a single source under the SI model  
[Shah and Zaman, 2011]
  - multiple seeds [Prakash et al., 2012]
  - $k$  seeds under the IC model [Lappas et al., 2010]
- the most recent papers
  - take advantage of the recorded infection order  
[Sefer and Kingsford, 2016].

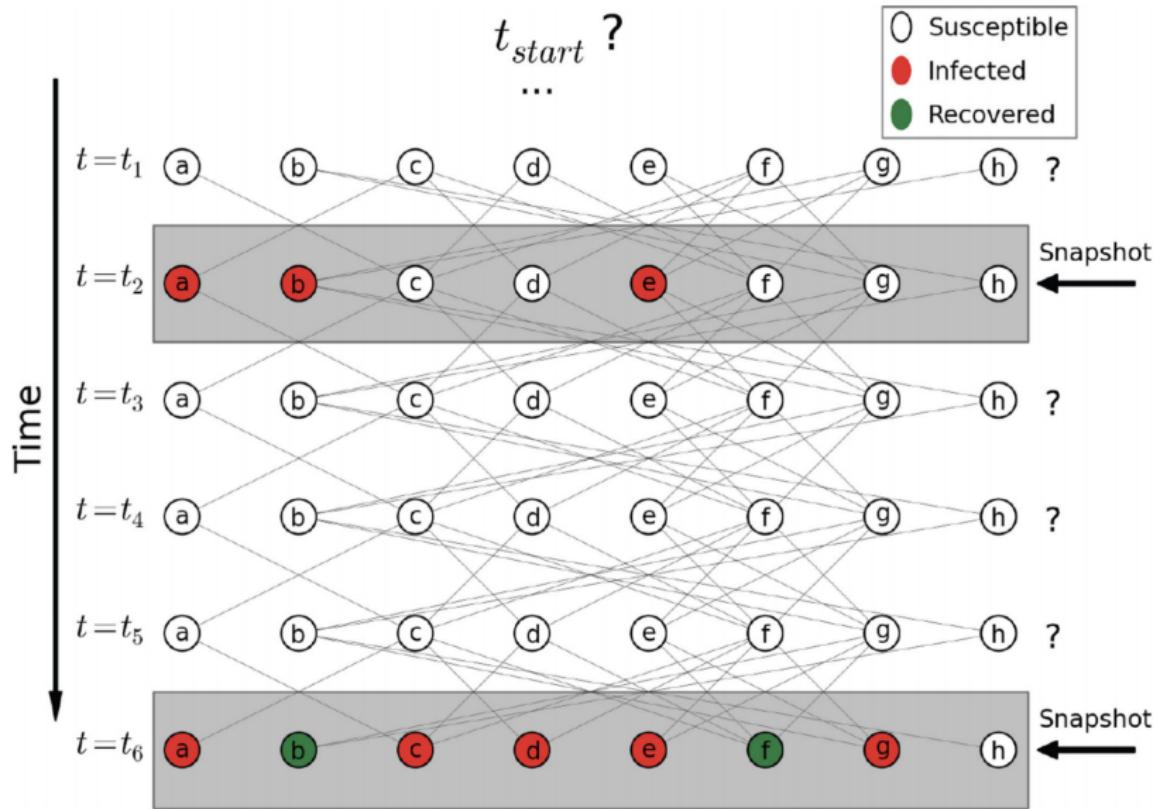
## temporal reconstruction

- the problems formulated **in this setting** tend to be either
  - **oversimplified** versions of static reconstruction or
  - become **too hard** or **ill-posed**
- knowing the history of interactions allow to reconstruct feasible paths of infection and prune unfeasible
- any noise or missing information adds uncertainty
- need more assumptions about the **noise** and information **available**

## temporal reconstruction

- some problem formulations :
- reconstruct the cascade given the **sequence** of graph **snapshots** along with **node-status** information  
[Feizi et al., 2016, Sefer and Kingsford, 2016]
- reconstruct an **SI cascade** from one **sampled** snapshot with **all** information  
[Sundareisan et al., 2015]
- while there are methods to handle **partially observed cascade** for static graphs, in temporal graphs most of works rely on **noise-free data**
- the knowledge of the **diffusion model** is crucial
- see survey paper: [Holme, 2015]

# history reconstruction



network summarization

# network summarization

- aims to **simplify** and **explain** the **high-level** structure of complex real graphs
- many different problem formulations and techniques:
  - recent **survey** [Liu et al., 2016a]

# motivation and applications

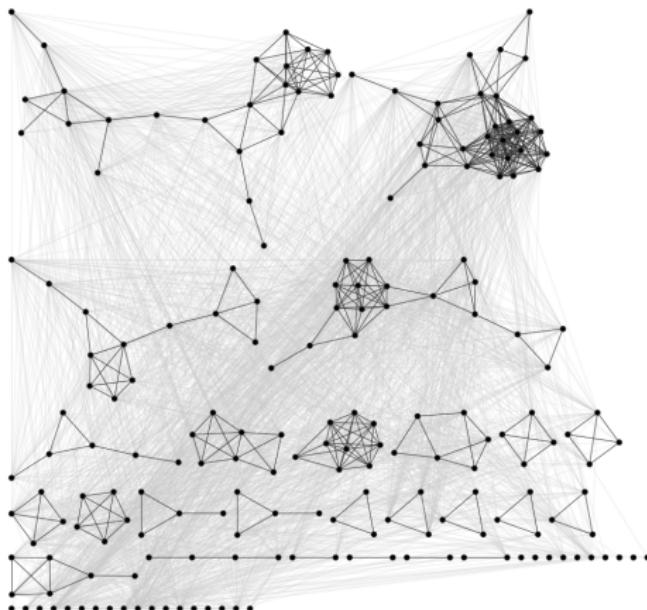
- fast and interactive large-graph analysis:
  - summaries decrease space and memory required for the storage and processing of real-world networks
- clear human understandable visualization
- noise elimination: filter out insignificant structural fluctuations in networks and preserve only prominent patterns

# approaches to summarization

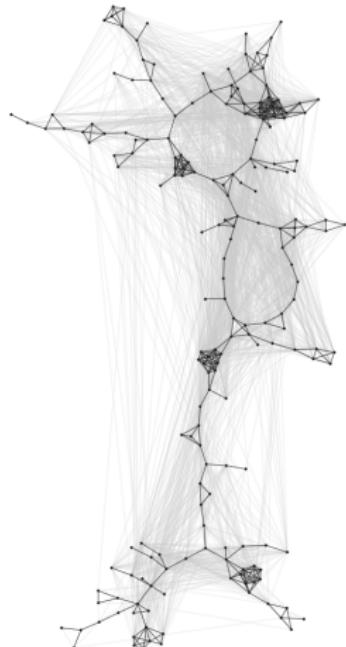
- sparsification
- aggregation / compression
- non-graph summary

# sparsification

- remove somewhat **unimportant** edges or/and nodes
- preserving certain **local** or/and **global** structures
- important properties to preserve are **cuts**, **community structures**, **distances**, **spectral properties**, etc.



(a) Quadrilateral Simmelian Backbone



(b) Quadrilateral Simmelian Backbone with UMST

[Hamann et al., 2016]

# sparsification

- sparsification problems are often formulated as **optimization** problems:
  - minimize some kind of graph **approximation** (reconstruction) error
  - while sparsifying **as much as possible**
- another common approach are **heuristic** strategies
- **survey**: [Hamann et al., 2016]

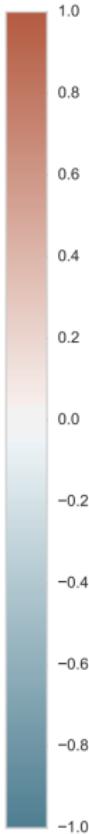
## some comparison

- random edge (RE)
- triangle counts (Tri)
- Jaccard similarity (JS) [Satuluri et al., 2011]
- simmelian backbones (TS, QLS) [Nick et al., 2013]
- edge forest fire (EFF) [Leskovec and Faloutsos, 2006]
- algebraic distance (AD) [Chen and Safro, 2011]
- local degree (LD) [Hamann et al., 2016]
- “local” versions of all mentioned methods  
[Hamann et al., 2016]

[Hamann et al., 2016]

# some comparison

| MOD | 0.4 | 0.46 | 0.39 | 0.38 | 0.42 | 0.39 | 0.44 | 0.41 | 0.24 | -0.13 | 0.026    | -0.025 | -0.00022 | 0.013    |        |
|-----|-----|------|------|------|------|------|------|------|------|-------|----------|--------|----------|----------|--------|
| +   | Ad  | 0.74 | 0.38 | 0.37 | 0.37 | 0.37 | 0.4  | 0.39 | 0.31 | -0.14 | -0.075   | -0.087 | 0.00016  | -0.0094  |        |
| +   | +   | LAD  | 0.36 | 0.44 | 0.4  | 0.45 | 0.42 | 0.47 | 0.21 | -0.17 | 0.046    | -0.018 | -0.00011 | 0.021    |        |
| +   | +   | +    | ls   | 0.83 | 0.84 | 0.7  | 0.93 | 0.77 | 0.81 | -0.19 | -0.15    | -0.18  | 0.0002   | -0.03    |        |
| +   | +   | +    | +    | +    | Ls   | 0.75 | 0.83 | 0.84 | 0.92 | 0.57  | -0.25    | 0.034  | -0.041   | 0.00014  | 0.011  |
| +   | +   | +    | +    | +    | +    | 4s   | 0.88 | 0.85 | 0.76 | 0.68  | -0.13    | -0.11  | -0.14    | 3.2e-05  | -0.017 |
| +   | +   | +    | +    | +    | +    | +    | Ls   | 0.76 | 0.84 | 0.48  | -0.19    | 0.034  | -0.028   | -3.4e-05 | 0.015  |
| +   | +   | +    | +    | +    | +    | +    | qs   | 0.88 | 0.71 | -0.18 | -0.059   | -0.11  | 9.2e-05  | -0.011   |        |
| +   | +   | +    | +    | +    | +    | +    | +    | Lqs  | 0.53 | -0.19 | 0.05     | -0.017 | -9.5e-05 | 0.017    |        |
| +   | +   | +    | +    | +    | +    | +    | +    | +    | Tr   | 0.21  | -0.51    | -0.4   | 6.5e-05  | -0.086   |        |
| -   | -   | -    | -    | -    | -    | -    | -    | -    | +    | Ld    | -0.4     | -0.19  | -0.00015 | -0.041   |        |
| +   | -   | +    | -    | +    | -    | +    | -    | +    | -    | EFF   | 0.46     | 5e-05  | 0.097    |          |        |
| -   | -   | -    | -    | -    | -    | -    | -    | -    | -    | LEFF  | -0.00038 | 0.076  |          |          |        |
|     |     |      |      |      |      |      |      |      |      | RE    | 8.8e-05  |        |          |          |        |
| +   | -   | +    | -    | +    | -    | +    | -    | +    | -    | +     | +        | +      |          | LRE      |        |



## aggregation / compression

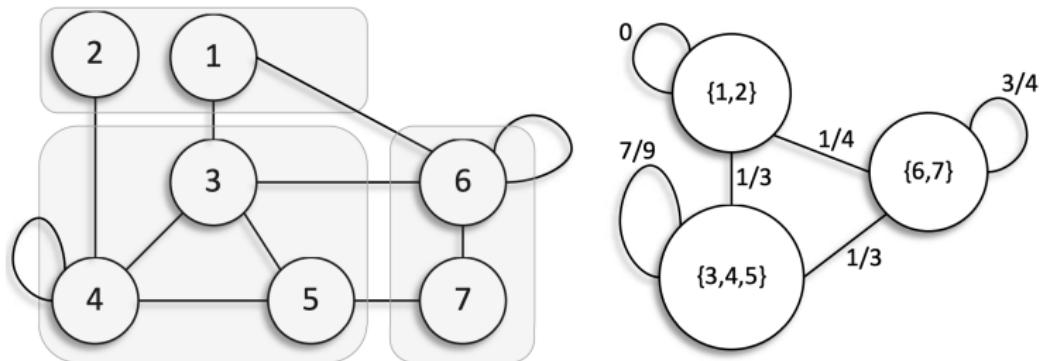
- super graph:
  - nodes are grouped into **supernodes** and
  - edges between the super nodes form **superedges**
- graph aggregation can be formulated as an optimization problem
  - minimizing reconstruction error
  - preserve some properties
- common heuristic is to build a supergraph based on **clustering**

[Abello et al., 2006, Clémenton et al., 2012]

## aggregation / compression

- some examples:
  - node aggregation to approximate node degree and eigenvector centrality  
[LeFevre and Terzi, 2010, Riondato et al., 2017]
  - edge aggregation to preserve the weights of superedges or strengths of the paths  
[Toivonen et al., 2011]

## compression example



- graph  $G = (V, E)$
- number  $k$
- $A_G$  : adjacency matrix of  $G$
- $k$ -summary  $S$  of  $G$  is a complete undirected weighted graph  $S = (V', V' \times V')$
- where  $V'$  is a disjoint  $k$ -partition of  $V$

## non-graph summary

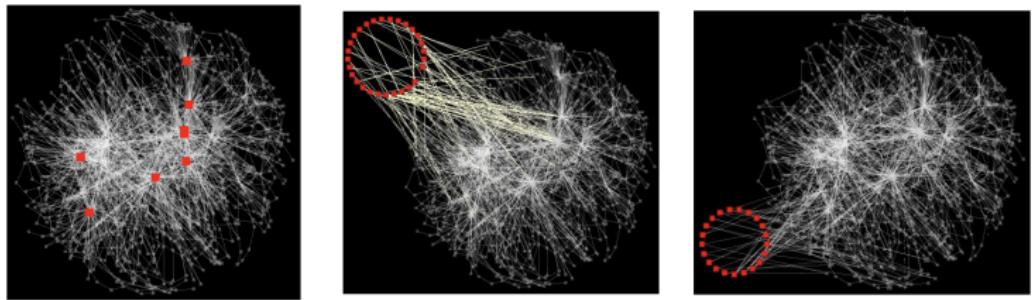
- represent some **interesting**, **characterizing**, or otherwise **important** structures observed in the graph
    - e.g. a set of tightly interconnected nodes (**communities**)
    - graph can be summarized as a set of communities, **ignoring other parts**
- [Lancichinetti et al., 2011, Perozzi and Akoglu, 2018]

## non-graph summary

- other examples:
  - motif counting  
(counting small subgraphs of restricted size)  
[Itzhack et al., 2007]
  - finding frequent subgraphs  
[Jiang et al., 2013]
- other approaches develop specialized vocabulary to encode a large graph.
- e.g., summarize by a set of chains, stars, cliques, and bipartite cores  
[Koutra et al., 2015]
- this framework can be further extended to domain-specific vocabulary constructed by an expert

# vocabulary-based summarization

- vocabulary: full and near cliques (fc, nc), full and near bipartite cores (fb, nb), stars (st), and chains (ch)
- encode the graph using MDL-base encoding:  
 $\text{graph} = \text{vocabulary} + \text{noise}$



- more approaches in the survey [Liu et al., 2016a]

[Koutra et al., 2015]

# temporal graph summarization

adaptation of existing techniques

- frequent subgraph mining: find persistent graph patterns over a collection of snapshots
- do not take into account how the instances of the same subgraph are located in time
- sequential pattern mining: search for time-ordered patterns in the sequence of snapshots
- network evolutionary patterns  
[Berlingerio et al., 2009, Wackersreuther et al., 2010]
- ignores structural patterns
- time-series analysis: gather node- and structure-dependent statistics over time
- apply segmentation techniques [Ye and Keogh, 2009]
- does not consider network structure

# temporal techniques

- summarization of both structural and temporal aspects
- how to define a summary?
- many possible options:
  - a summary can be a short temporal sequence of small graphs,
  - a concise presentation of evolutionary patterns,
  - a representative collection of temporally and topologically frequent patterns
- one common approach to summary definition:
- summary should consist of
  - small structurally “interesting” subgraphs
  - with non-trivial temporal behavior

# temporal motifs

- temporal motif counting

[Paranjape et al., 2017, Kovanen et al., 2013]:

- temporal motif is a small subgraph with temporally ordered edges (and/or interval or delay constraints)

- some other works explore temporal graphlets

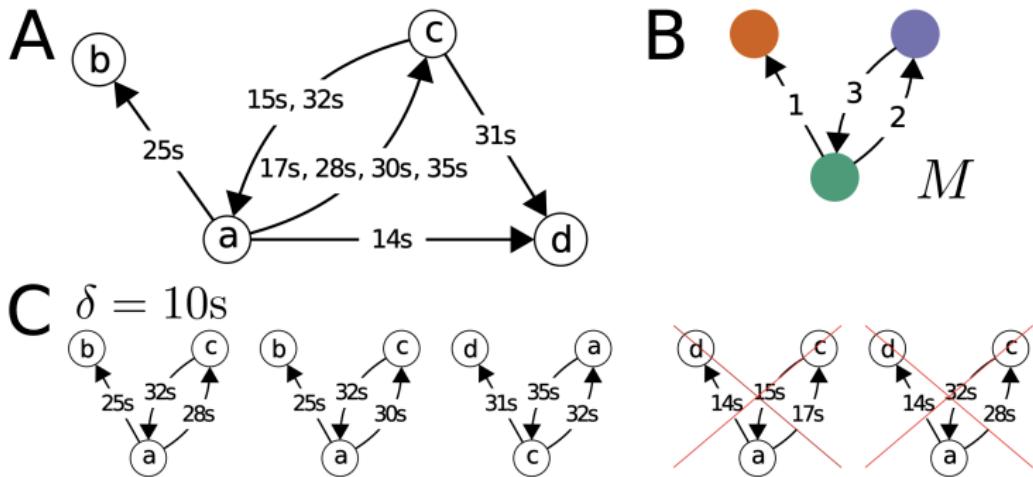
- time constrained causal subgraphs

[Hulovatyy et al., 2015]

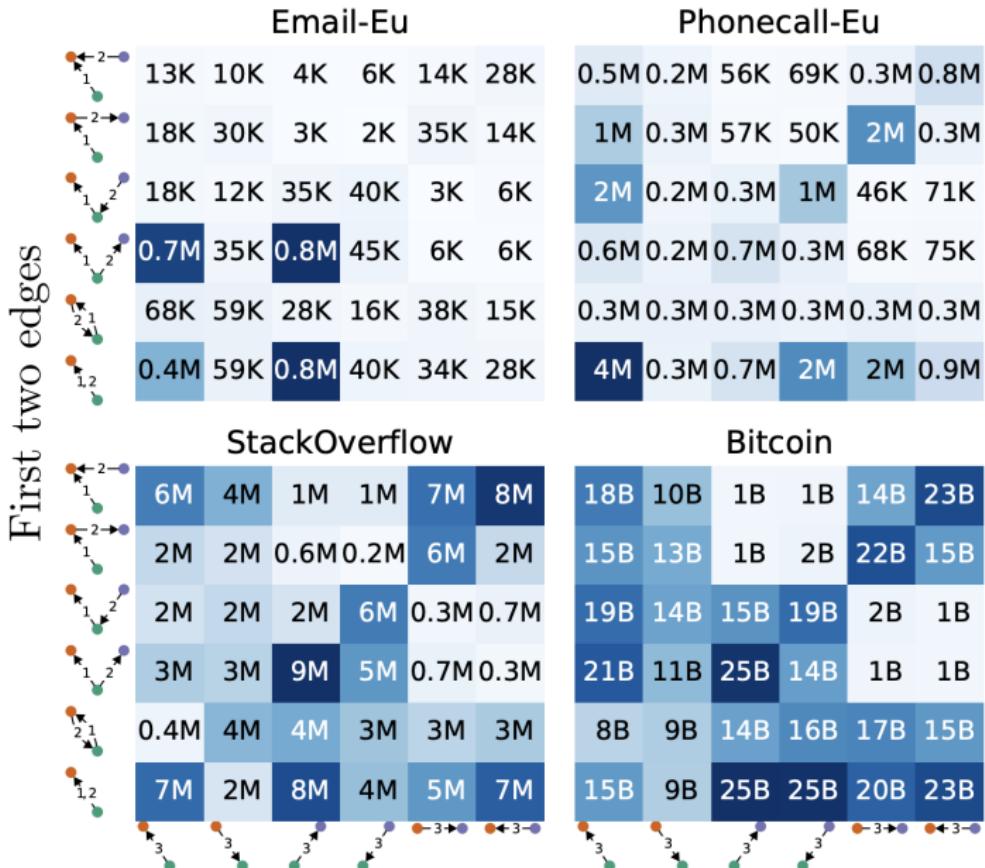
and cyclic patterns

[Lahiri and Berger-Wolf, 2008]

# temporal motifs



[Paranjape et al., 2017]



## vocabulary-based summarization

- summarize a temporal graph as a set:
  - subgraphs of a special “most non-random” shape (stars, cliques, bipartite cores, chains), and
  - behavioural temporal patterns (flickering, periodic, oneshot, ranged, and constant patterns)
- use MDL principle to encode whole temporal network by the vocabulary plus noise

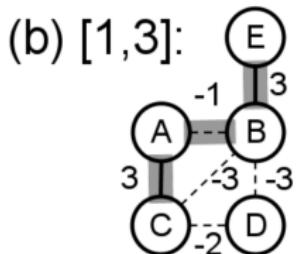
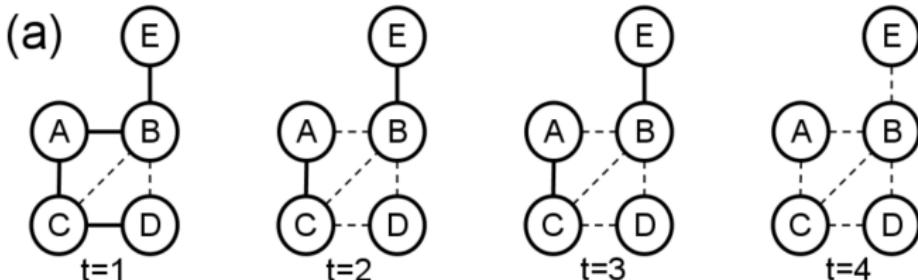
[Shah et al., 2015]

## larger structures

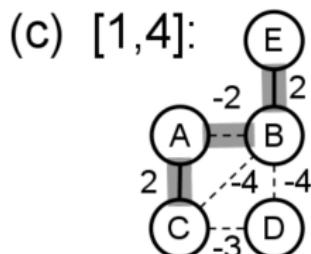
- use larger structures to summarize the network:
  - communities
  - spanning graphs
  - backbones
  - cores
- common approach:
  - given a sequence of graphs  
(snapshot, or sliding-window aggregation)
  - search for communities that are coherent and/or persistent in time
- different measures of community quality and temporal smoothness are used

[Pietilänen and Diot, 2012, He and Chen, 2015]
- the resulting summary is a trade-off between structural quality and historical consistency

## temporal backbones



$$\text{score}(\{\text{AB}, \text{AC}, \text{BE}\}, 1, 3) = 5$$



$$\text{score}(\{\text{AB}, \text{AC}, \text{BE}\}, 1, 4) = 2$$

[Bogdanov et al., 2011]

## influence-based summarization

- summarizes the flow of information propagation:
  - find influential nodes and information-forwarding connections
- OSNet [Qu et al., 2014]:
  - processes a temporal network in a streaming fashion
  - outputs the subgraphs of influential nodes
  - node importance is calculated based on temporal spreading trees
- [Lin et al., 2008] identify influential nodes and interactions in temporal multi-view social networks
  - networks with edges between different types of entities, e.g., users, photos, and comments
  - explain the evolution of topics over time

# agenda

Part I : introduction and motivation

Part II : models of temporal networks

Part III : algorithmic frameworks

Part IV : data mining problems

Part V : future challenges

## part V

# future challenges

## temporal community detection: challenges

- large number of **problem formulations** and variants
- lack of **fundamental theoretical** treatment
  - most of the approaches are **heuristics**
  - many are combinations of **several** ideas and algorithms
  - require **many** parameters and attention to implementation details
- hard to compare methods and choose one for an application
  - **few datasets** with ground-truth temporal communities
  - synthetic generators are built on **various assumptions**
  - **no** standard benchmarks
- a large number of **quality metrics** to calculate and compare
- may be **misleading** if a method is not designed for that particular community definition

## event detection: challenges

- actively **evolving** area, **application- and data-oriented**
  - families of problems and methods are considered only for the **specific** sources of data
    - e.g., a large body of research is focused on the analysis of **Twitter** data [Atefah and Khreich, 2015]
  - no **unified classification** for problem settings, research questions, and data requirements
    - recent classifications are based on **various** aspects:
    - event **definitions**, **online or retrospective detection**, **specified or unspecified** event detection, etc.
- [Cordeiro and Gama, 2016, Goswami and Kumar, 2016]

## event detection: more challenges

- speed and quality:
- online streaming event-detection techniques are demanded for nearly real-time event detection
- quality: both false events and missed events may have a high price
- more use of multi-modal data:
- text: complex semantic and sentiment analysis is rare
- high-resolution interaction patterns: “who talked to whom about what and what happened then” are also often not considered

## diffusion analysis: challenges

- influence maximization and immunization strategies:
  - what is the most realistic approachable setting?
- models:
  - temporal diffusion models are proposed, but the theoretical properties of many of them are not yet well studied
  - the applications and limitations are not yet well understood
- immunization strategies:
  - not extensively studied yet
  - most of the approaches are based on heuristics

## summarization: challenges

- meaningful summary vocabulary
- diversity of summarizing substructures is vast  
[Perozzi and Akoglu, 2018, Koutra et al., 2015,  
Jiang et al., 2013])
- which summaries are preferable and in which applications?
- summaries useful for a general network exploration by a  
non-expert analyst?

## summarization: more challenges

- fast and light-weighted algorithms
- interactive analysis
- have a hierarchical structure, which is possible to browse
  - similar to a visual analytic tool OntoVis, which constructs some type of graphical summaries  
[Shen et al., 2006]
- multi-level summarizations:
- use all available attributes in the temporal networks
  - text, geotags, propagation patterns...

# references

-  **Abdelhaq, H., Sengstock, C., and Gertz, M. (2013).**  
Eventweet: Online localized event detection from twitter.  
*Proceedings of the VLDB Endowment*, 6(12):1326–1329.
-  **Abello, J., Van Ham, F., and Krishnan, N. (2006).**  
Ask-graphview: A large scale graph visualization system.  
*IEEE transactions on visualization and computer graphics*, 12(5):669–676.
-  **Aggarwal, C. and Subbian, K. (2014).**  
Evolutionary network analysis: A survey.  
*ACM Computing Surveys (CSUR)*, 47(1):10.
-  **Aggarwal, C. C., Lin, S., and Yu, P. S. (2012).**  
On influential node discovery in dynamic social networks.  
In *Proceedings of the 2012 SIAM International Conference on Data Mining*, pages 636–647. SIAM.

## references (cont.)

-  Aggarwal, C. C. and Subbian, K. (2012).  
Event detection in social streams.  
*In Proceedings of the 2012 SIAM international conference on data mining*, pages 624–635. SIAM.
-  Akrida, E. C., Mertzios, G. B., Spirakis, P. G., and Zamaraev, V. (2018).  
Temporal vertex cover with a sliding time window.  
*arXiv preprint arXiv:1802.07103*.
-  Atefah, F. and Khreich, W. (2015).  
A survey of techniques for event detection in twitter.  
*Computational Intelligence*, 31(1):132–164.
-  Aynaud, T., Fleury, E., Guillaume, J., Wang, Q., Ganguly, N., Mukherjee, A., Mitra, B., Peruani, F., and Choudhury, M. (2013).  
Dynamics on and of complex networks.

## references (cont.)

-  Barabasi, A.-L. (2005).

The origin of bursts and heavy tails in human dynamics.  
*Nature*, 435(7039):207.
-  Batal, I., Fradkin, D., Harrison, J., Moerchen, F., and Hauskrecht, M. (2012).

Mining recent temporal patterns for event detection in multivariate time series data.  
In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 280–288. ACM.
-  Berlingerio, M., Bonchi, F., Bringmann, B., and Gionis, A. (2009).

Mining graph evolution rules.  
In *joint European conference on machine learning and knowledge discovery in databases*, pages 115–130. Springer.
-  Berlingerio, M., Koutra, D., Eliassi-Rad, T., and Faloutsos, C. (2012).

Netsimile: A scalable approach to size-independent network similarity.  
*arXiv preprint arXiv:1209.2684*.

## references (cont.)



Berman, K. A. (1996).

Vulnerability of scheduled networks and a generalization of menger's theorem.

*Networks: An International Journal*, 28(3):125–134.



Blondel, V. D., Guillaume, J.-L., Lambiotte, R., and Lefebvre, E. (2008).

Fast unfolding of communities in large networks.

*Journal of statistical mechanics: theory and experiment*,  
2008(10):P10008.



Bogdanov, P., Mongiovì, M., and Singh, A. K. (2011).

Mining heavy subgraphs in time-evolving networks.

In *Data Mining (ICDM), 2011 IEEE 11th International Conference on*,  
pages 81–90. IEEE.

## references (cont.)

-  Candia, J., González, M. C., Wang, P., Schoenharl, T., Madey, G., and Barabási, A.-L. (2008).  
Uncovering individual and collective human dynamics from mobile phone records.  
*Journal of physics A: mathematical and theoretical*, 41(22):224015.
-  Casteigts, A., Flocchini, P., Quattrociocchi, W., and Santoro, N. (2012).  
Time-varying graphs and dynamic networks.  
*International Journal of Parallel, Emergent and Distributed Systems*, 27(5):387–408.
-  Chaintreau, A., Mtibaa, A., Massoulie, L., and Diot, C. (2007).  
The diameter of opportunistic mobile networks.  
In *Proceedings of the 2007 ACM CoNEXT conference*, page 12. ACM.

## references (cont.)

-  Charikar, M. (2000).  
Greedy approximation algorithms for finding dense components in a graph.  
In *International Workshop on Approximation Algorithms for Combinatorial Optimization*, pages 84–95. Springer.
-  Chen, J. and Safro, I. (2011).  
Algebraic distance on graphs.  
*SIAM Journal on Scientific Computing*, 33(6):3468–3490.
-  Chen, W., Lu, W., and Zhang, N. (2012).  
Time-critical influence maximization in social networks with time-delayed diffusion process.  
In *AAAI*, volume 2012, pages 1–5.

## references (cont.)

-  Chen, W., Wang, C., and Wang, Y. (2010).  
Scalable influence maximization for prevalent viral marketing in large-scale social networks.  
In *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1029–1038. ACM.
-  Chen, W., Wang, Y., and Yang, S. (2009).  
Efficient influence maximization in social networks.  
In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 199–208. ACM.
-  Cléménçon, S., De Arazoza, H., Rossi, F., and Tran, V. C. (2012).  
Hierarchical clustering for graph visualization.  
*arXiv preprint arXiv:1210.5693*.
-  Cohen, R., Havlin, S., and Ben-Avraham, D. (2003).  
Efficient immunization strategies for computer networks and populations.  
*Physical review letters*, 91(24):247901.

## references (cont.)

-  Cordeiro, M. and Gama, J. (2016).  
Online social networks event detection: a survey.  
In *Solving Large Scale Learning Tasks. Challenges and Algorithms*, pages 1–41. Springer.
-  Coscia, M., Giannotti, F., and Pedreschi, D. (2011).  
A classification for community discovery methods in complex networks.  
*Statistical Analysis and Data Mining: The ASA Data Science Journal*, 4(5):512–546.
-  Dakiche, N., Tayeb, F. B.-S., Slimani, Y., and Benatchba, K. (2019).  
Tracking community evolution in social networks: A survey.  
*Information Processing & Management*, 56(3):1084–1102.
-  Dezső, Z. and Barabási, A.-L. (2002).  
Halting viruses in scale-free networks.  
*Physical Review E*, 65(5):055103.

## references (cont.)

-  Dong, X., Mavroeidis, D., Calabrese, F., and Frossard, P. (2015).  
Multiscale event detection in social media.  
*Data Mining and Knowledge Discovery*, 29(5):1374–1405.
-  Dou, W., Wang, X., Ribarsky, W., and Zhou, M. (2012).  
Event detection in social media data.  
In *IEEE VisWeek Workshop on Interactive Visual Text Analytics-Task Driven Analytics of Social Media Content*, pages 971–980.
-  Du, N., Song, L., Rodriguez, M. G., and Zha, H. (2013).  
Scalable influence estimation in continuous-time diffusion networks.  
In *Advances in neural information processing systems*, pages 3147–3155.
-  Enugala, R., Rajamani, L., Ali, K., and Kurapati, S. (2015).  
Community detection in dynamic social networks: a survey.  
*International Journal of Research and Applications*, 2(6):278–285.

## references (cont.)

-  Eswaran, D., Faloutsos, C., Guha, S., and Mishra, N. (2018).  
Spotlight: Detecting anomalies in streaming graphs.  
In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 1378–1386. ACM.
-  Feizi, S., Médard, M., Quon, G., Kellis, M., and Duffy, K. (2016).  
Network infusion to infer information sources in networks.  
*arXiv preprint arXiv:1606.07383*.
-  Fernández-Gracia, J., Eguíluz, V. M., and San Miguel, M. (2011).  
Update rules and interevent time distributions: Slow ordering versus no ordering in the voter model.  
*Physical Review E*, 84(1):015103.
-  Fortunato, S. and Hric, D. (2016).  
Community detection in networks: A user guide.  
*Physics reports*, 659:1–44.

## references (cont.)

-  Gayraud, N. T., Pitoura, E., and Tsaparas, P. (2015).  
Diffusion maximization in evolving social networks.  
In *Proceedings of the 2015 ACM on Conference on Online Social Networks*, pages 125–135. ACM.
-  Gensler, A. and Sick, B. (2017).  
Performing event detection in time series with swiftevent: an algorithm with supervised learning of detection criteria.  
*Pattern Analysis and Applications*, pages 1–20.
-  Gomez-Rodriguez, M., Song, L., Du, N., Zha, H., and Schölkopf, B. (2016).  
Influence estimation and maximization in continuous-time diffusion networks.  
*ACM Transactions on Information Systems (TOIS)*, 34(2):9.
-  Goswami, A. and Kumar, A. (2016).  
A survey of event detection techniques in online social networks.  
*Social Network Analysis and Mining*, 6(1):107.

## references (cont.)

-  Gu, H., Xie, X., Lv, Q., Ruan, Y., and Shang, L. (2011).  
Etree: Effective and efficient event modeling for real-time online social media networks.  
*In Proceedings of the 2011 IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology-Volume 01*, pages 300–307. IEEE Computer Society.
-  Hamann, M., Lindner, G., Meyerhenke, H., Staudt, C. L., and Wagner, D. (2016).  
Structure-preserving sparsification methods for social networks.  
*Social Network Analysis and Mining*, 6(1):22.
-  Harris, T. E. (2002).  
*The theory of branching processes*.  
Courier Corporation.
-  Hartmann, T., Kappes, A., and Wagner, D. (2016).  
Clustering evolving networks.  
*In Algorithm Engineering*, pages 280–329. Springer.

## references (cont.)

-  He, J. and Chen, D. (2015).  
A fast algorithm for community detection in temporal network.  
*Physica A: Statistical Mechanics and its Applications*, 429:87–94.
-  Heins, K. and Stern, H. (2014).  
A statistical model for event sequence data.  
In *Artificial Intelligence and Statistics*, pages 338–346.
-  Henzinger, M. R., King, V., and King, V. (1999).  
Randomized fully dynamic graph algorithms with polylogarithmic time per operation.  
*Journal of the ACM (JACM)*, 46(4):502–516.
-  Holme, P. (2015).  
Modern temporal network theory: a colloquium.  
*The European Physical Journal B*, 88(9):234.

## references (cont.)

-  Hong, L., Ahmed, A., Gurumurthy, S., Smola, A. J., and Tsoutsouliklis, K. (2012).  
Discovering geographical topics in the twitter stream.  
In *Proceedings of the 21st international conference on World Wide Web*, pages 769–778. ACM.
-  Huang, S., Fu, A. W.-C., and Liu, R. (2015).  
Minimum spanning trees in temporal graphs.  
In *Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data*, pages 419–430. ACM.
-  Hulovatyy, Y., Chen, H., and Milenković, T. (2015).  
Exploring the structure and function of temporal networks with dynamic graphlets.  
*Bioinformatics*, 31(12):i171–i180.

## references (cont.)

-  Hunter, J. and McIntosh, N. (1999).  
Knowledge-based event detection in complex time series data.  
*In Joint European Conference on Artificial Intelligence in Medicine and Medical Decision Making*, pages 271–280. Springer.
-  Itzhack, R., Mogilevski, Y., and Louzoun, Y. (2007).  
An optimal algorithm for counting network motifs.  
*Physica A: Statistical Mechanics and its Applications*, 381:482–490.
-  Jiang, C., Coenen, F., and Zito, M. (2013).  
A survey of frequent subgraph mining algorithms.  
*The Knowledge Engineering Review*, 28(1):75–105.
-  Karsai, M., Kivelä, M., Pan, R. K., Kaski, K., Kertész, J., Barabási, A.-L., and Saramäki, J. (2011).  
Small but slow world: How network topology and burstiness slow down spreading.  
*Physical Review E*, 83(2):025102.

## references (cont.)

-  Kempe, D., Kleinberg, J., and Tardos, É. (2003).  
Maximizing the spread of influence through a social network.  
In *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 137–146. ACM.
-  Khuller, S. and Saha, B. (2009).  
On finding dense subgraphs.  
In *International Colloquium on Automata, Languages, and Programming*, pages 597–608. Springer.
-  Kleinberg, J. (2003).  
Bursty and hierarchical structure in streams.  
*Data Mining and Knowledge Discovery*, 7(4):373–397.
-  Kling, C. C., Kunegis, J., Sizov, S., and Staab, S. (2014).  
Detecting non-gaussian geographical topics in tagged photo collections.  
In *Proceedings of the 7th ACM international conference on Web search and data mining*, pages 603–612. ACM.

## references (cont.)

-  Koutra, D., Kang, U., Vreeken, J., and Faloutsos, C. (2015).  
Summarizing and understanding large graphs.  
*Statistical Analysis and Data Mining: The ASA Data Science Journal*,  
8(3):183–202.
-  Kovanen, L., Karsai, M., Kaski, K., Kertész, J., and Saramäki, J. (2013).  
Temporal motifs.  
In *Temporal Networks*, pages 119–133. Springer.
-  Kujala, R., Weckström, C., Mladenović, M. N., and Saramäki, J. (2018).  
Travel times and transfers in public transport: Comprehensive accessibility analysis based on pareto-optimal journeys.  
*Computers, Environment and Urban Systems*, 67:41–54.
-  Kulldorff, M. (1997).  
A spatial scan statistic.  
*Communications in Statistics-Theory and Methods*, 26(6):1481–1496.

## references (cont.)



Kumar, R., Calders, T., Gionis, A., and Tatti, N. (2015).

Maintaining sliding-window neighborhood profiles in interaction networks.

In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 719–735. Springer.



Kunneman, F. and van den Bosch, A. (2014).

Event detection in twitter: A machine-learning approach based on term pivoting.



Kunneman, F. and Van den Bosch, A. (2015).

Automatically identifying periodic social events from twitter.

In *Proceedings of the International Conference Recent Advances in Natural Language Processing*, pages 320–328.



Lahiri, M. and Berger-Wolf, T. Y. (2008).

Mining periodic behavior in dynamic social networks.

In *Data Mining, 2008. ICDM'08. Eighth IEEE International Conference on*, pages 373–382. IEEE.

## references (cont.)

-  Lancichinetti, A., Radicchi, F., Ramasco, J. J., and Fortunato, S. (2011).  
Finding statistically significant communities in networks.  
*PloS one*, 6(4):e18961.
-  Lappas, T., Terzi, E., Gunopulos, D., and Mannila, H. (2010).  
Finding effectors in social networks.  
In *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1059–1068. ACM.
-  Lappas, T., Vieira, M. R., Gunopulos, D., and Tsotras, V. J. (2012).  
On the spatiotemporal burstiness of terms.  
*Proceedings of the VLDB Endowment*, 5(9):836–847.
-  Latapy, M., Viard, T., and Magnien, C. (2018).  
Stream graphs and link streams for the modeling of interactions over time.  
*Social Network Analysis and Mining*, 8(1):61.

## references (cont.)

-  Lee, S., Rocha, L. E., Liljeros, F., and Holme, P. (2012).  
Exploiting temporal network structures of human interaction to effectively immunize populations.  
*PloS one*, 7(5):e36439.
-  LeFevre, K. and Terzi, E. (2010).  
Grass: Graph structure summarization.  
In *Proceedings of the 2010 SIAM International Conference on Data Mining*, pages 454–465. SIAM.
-  Leskovec, J. and Faloutsos, C. (2006).  
Sampling from large graphs.  
In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 631–636. ACM.
-  Leskovec, J. and Horvitz, E. (2008).  
Planetary-scale views on a large instant-messaging network.  
In *Proceedings of the 17th international conference on World Wide Web*, pages 915–924. ACM.

## references (cont.)

-  Li, J., Tai, Z., Zhang, R., Yu, W., and Liu, L. (2014).  
Online bursty event detection from microblog.  
In *Proceedings of the 2014 IEEE/ACM 7th International Conference on Utility and Cloud Computing*, pages 865–870. IEEE Computer Society.
-  Lin, Y.-R., Sundaram, H., and Kelliher, A. (2008).  
Summarization of social activity over time: people, actions and concepts in dynamic networks.  
In *Proceedings of the 17th ACM conference on Information and knowledge management*, pages 1379–1380. ACM.
-  Liu, B., Cong, G., Xu, D., and Zeng, Y. (2012).  
Time constrained influence maximization in social networks.  
In *Data Mining (ICDM), 2012 IEEE 12th International Conference on*, pages 439–448. IEEE.
-  Liu, Y., Dighe, A., Safavi, T., and Koutra, D. (2016a).  
A graph summarization: A survey.  
*arXiv preprint arXiv:1612.04883*.

## references (cont.)

-  Liu, Y., Zhou, B., Chen, F., and Cheung, D. W. (2016b).  
Graph topic scan statistic for spatial event detection.  
*In Proceedings of the 25th ACM International Conference on Information and Knowledge Management*, pages 489–498. ACM.
-  Madar, N., Kalisky, T., Cohen, R., Ben-avraham, D., and Havlin, S. (2004).  
Immunization and epidemic dynamics in complex networks.  
*The European Physical Journal B*, 38(2):269–276.
-  McGregor, A. (2014).  
Graph stream algorithms: a survey.  
*ACM SIGMOD Record*, 43(1):9–20.
-  Mertzios, G. B., Molter, H., Niedermeier, R., Zamaraev, V., and Zschoche, P. (2019).  
Computing maximum matchings in temporal graphs.  
*arXiv preprint arXiv:1905.05304*.

## references (cont.)

-  Mertzios, G. B., Molter, H., and Zamaraev, V. (2018).  
Sliding window temporal graph coloring.  
*arXiv preprint arXiv:1811.04753.*
-  Michail, O. (2016).  
An introduction to temporal graphs: An algorithmic perspective.  
*Internet Mathematics*, 12(4):239–280.
-  Min, B., Goh, K.-I., and Vazquez, A. (2011).  
Spreading dynamics following bursty human activity patterns.  
*Physical Review E*, 83(3):036102.
-  Muthukrishnan, S. et al. (2005).  
Data streams: Algorithms and applications.  
*Foundations and Trends® in Theoretical Computer Science*, 1(2):117–236.

## references (cont.)

-  Neill, D. B. (2006).  
Detection of spatial and spatio-temporal clusters.  
In *Tech Rep CMU-CS-06-142, PhD thesis*. Carnegie Mellon University.
-  Newman, M. E. (2003).  
The structure and function of complex networks.  
*SIAM review*, 45(2):167–256.
-  Nick, B., Lee, C., Cunningham, P., and Brandes, U. (2013).  
Simmelian backbones: Amplifying hidden homophily in facebook networks.  
In *Proceedings of the 2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, pages 525–532. ACM.
-  Paranjape, A., Benson, A. R., and Leskovec, J. (2017).  
Motifs in temporal networks.  
In *Proceedings of the Tenth ACM International Conference on Web Search and Data Mining*, pages 601–610. ACM.

## references (cont.)

-  Pastor-Satorras, R. and Vespignani, A. (2002a).  
Epidemics and immunization in scale-free networks.  
*arXiv preprint cond-mat/0205260*.
-  Pastor-Satorras, R. and Vespignani, A. (2002b).  
Immunization of complex networks.  
*Physical Review E*, 65(3):036104.
-  Perozzi, B. and Akoglu, L. (2018).  
Discovering communities and anomalies in attributed graphs: Interactive visual exploration and summarization.  
*ACM Transactions on Knowledge Discovery from Data (TKDD)*, 12(2):24.
-  Pietilänen, A.-K. and Diot, C. (2012).  
Dissemination in opportunistic social networks: the role of temporal communities.  
In *Proceedings of the thirteenth ACM international symposium on Mobile Ad Hoc Networking and Computing*, pages 165–174. ACM.

## references (cont.)

-  Prakash, B. A., Vreeken, J., and Faloutsos, C. (2012).  
Spotting culprits in epidemics: How many and which ones?  
In *Data Mining (ICDM), 2012 IEEE 12th International Conference on*, pages 11–20. IEEE.
-  Qian, J., Saligrama, V., and Chen, Y. (2014).  
Connected sub-graph detection.  
In *Artificial Intelligence and Statistics*, pages 796–804.
-  Qu, Q., Liu, S., Jensen, C. S., Zhu, F., and Faloutsos, C. (2014).  
Interestingness-driven diffusion process summarization in dynamic networks.  
In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 597–613. Springer.
-  Ranshous, S., Shen, S., Koutra, D., Harenberg, S., Faloutsos, C., and Samatova, N. F. (2015).  
Anomaly detection in dynamic networks: a survey.  
*Wiley Interdisciplinary Reviews: Computational Statistics*, 7(3):223–247.

## references (cont.)

-  Rayana, S. and Akoglu, L. (2016).  
Less is more: building selective anomaly ensembles.  
*ACM Transactions on Knowledge Discovery from Data (TKDD)*,  
10(4):42.
-  Renaud, L. and Naoki, M. (2016).  
*A Guide To Temporal Networks*, volume 4.  
World Scientific.
-  Riondato, M., García-Soriano, D., and Bonchi, F. (2017).  
Graph summarization with quality guarantees.  
*Data Mining and Knowledge Discovery*, 31(2):314–349.
-  Rodriguez, M. G., Balduzzi, D., and Schölkopf, B. (2011).  
Uncovering the temporal dynamics of diffusion networks.  
*arXiv preprint arXiv:1105.0697*.

## references (cont.)

-  Rodriguez, M. G. and Schölkopf, B. (2012).  
Influence maximization in continuous time diffusion networks.  
*arXiv preprint arXiv:1205.1682*.
-  Rossetti, G. and Cazabet, R. (2018).  
Community discovery in dynamic networks: a survey.  
*ACM Computing Surveys (CSUR)*, 51(2):35.
-  Rozenshtein, P. and Gionis, A. (2016).  
Temporal pagerank.  
In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 674–689. Springer.
-  Sakaki, T., Okazaki, M., and Matsuo, Y. (2010).  
Earthquake shakes twitter users: real-time event detection by social sensors.  
In *Proceedings of the 19th international conference on World wide web*, pages 851–860. ACM.

## references (cont.)

-  Satuluri, V., Parthasarathy, S., and Ruan, Y. (2011).  
Local graph sparsification for scalable clustering.  
In *Proceedings of the 2011 ACM SIGMOD International Conference on Management of data*, pages 721–732. ACM.
-  Sefer, E. and Kingsford, C. (2016).  
Diffusion archeology for diffusion progression history reconstruction.  
*Knowledge and information systems*, 49(2):403–427.
-  Shah, D. and Zaman, T. (2011).  
Rumors in a network: Who's the culprit?  
*IEEE Transactions on information theory*, 57(8):5163–5181.
-  Shah, N., Koutra, D., Zou, T., Gallagher, B., and Faloutsos, C. (2015).  
Timecrunch: Interpretable dynamic graph summarization.  
In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1055–1064. ACM.

## references (cont.)

-  Shakarian, P., Bhatnagar, A., Aleali, A., Shaabani, E., and Guo, R. (2015).  
*Diffusion in social networks.*  
Springer.
-  Shen, Z., Ma, K.-L., and Eliassi-Rad, T. (2006).  
Visual analysis of large heterogeneous social networks by semantic and structural abstraction.  
*IEEE transactions on visualization and computer graphics*,  
12(6):1427–1439.
-  Shi, Z. and Pun-Cheng, L. S. (2019).  
Spatiotemporal data clustering: A survey of methods.  
*ISPRS International Journal of Geo-Information*, 8(3):112.

## references (cont.)

-  Sun, Y., Tang, J., Pan, L., and Li, J. (2015).  
Matrix based community evolution events detection in online social networks.  
*In 2015 IEEE International Conference on Smart City/SocialCom/SustainCom (SmartCity)*, pages 465–470. IEEE.
-  Sundaresan, S., Vreeken, J., and Prakash, B. A. (2015).  
Hidden hazards: Finding missing nodes in large graph epidemics.  
*In Proceedings of the 2015 SIAM International Conference on Data Mining*, pages 415–423. SIAM.
-  Takahashi, K., Kulldorff, M., Tango, T., and Yih, K. (2008).  
A flexibly shaped space-time scan statistic for disease outbreak detection and monitoring.  
*International Journal of Health Geographics*, 7(1):14.
-  Takahashi, K., Yokoyama, T., and Tango, T. (2004).  
Flexscan: Software for the flexible spatial scan statistic.  
*National Institute of Public Health, Japan*.

## references (cont.)

-  Tang, J., Musolesi, M., Mascolo, C., and Latora, V. (2009).  
Temporal distance metrics for social network analysis.  
In *Proceedings of the 2nd ACM workshop on Online social networks*, pages 31–36. ACM.
-  Tang, Y., Xiao, X., and Shi, Y. (2014).  
Influence maximization: Near-optimal time complexity meets practical efficiency.  
In *Proceedings of the 2014 ACM SIGMOD international conference on Management of data*, pages 75–86. ACM.
-  Tantipathananandh, C. and Berger-Wolf, T. Y. (2011).  
Finding communities in dynamic social networks.  
In *2011 IEEE 11th International Conference on Data Mining*, pages 1236–1241. IEEE.

## references (cont.)

-  Thorup, M. (2000).  
Near-optimal fully-dynamic graph connectivity.  
In *Proceedings of the thirty-second annual ACM symposium on Theory of computing*, pages 343–350. Citeseer.
-  Toivonen, H., Zhou, F., Hartikainen, A., and Hinkka, A. (2011).  
Compression of weighted graphs.  
In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 965–973. ACM.
-  Vazquez, A., Racz, B., Lukacs, A., and Barabasi, A.-L. (2007).  
Impact of non-poissonian activity patterns on spreading processes.  
*Physical review letters*, 98(15):158702.
-  Wackersreuther, B., Wackersreuther, P., Oswald, A., Böhm, C., and Borgwardt, K. M. (2010).  
Frequent subgraph discovery in dynamic networks.  
In *Proceedings of the Eighth Workshop on Mining and Learning with Graphs*, pages 155–162. ACM.

## references (cont.)

-  Wu, H., Cheng, J., Huang, S., Ke, Y., Lu, Y., and Xu, Y. (2014).  
Path problems in temporal graphs.  
*Proceedings of the VLDB Endowment*, 7(9):721–732.
-  Wu, X., Kumar, V., Quinlan, J. R., Ghosh, J., Yang, Q., Motoda, H., McLachlan, G. J., Ng, A., Liu, B., Philip, S. Y., et al. (2008).  
Top 10 algorithms in data mining.  
*KAIS*.
-  Ye, L. and Keogh, E. (2009).  
Time series shapelets: a new primitive for data mining.  
In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 947–956. ACM.
-  Zhuang, H., Sun, Y., Tang, J., Zhang, J., and Sun, X. (2013).  
Influence maximization in dynamic social networks.  
In *Data Mining (ICDM), 2013 IEEE 13th International Conference on*, pages 1313–1318. IEEE.