Temporal Centrality Metrics for Graph Streams

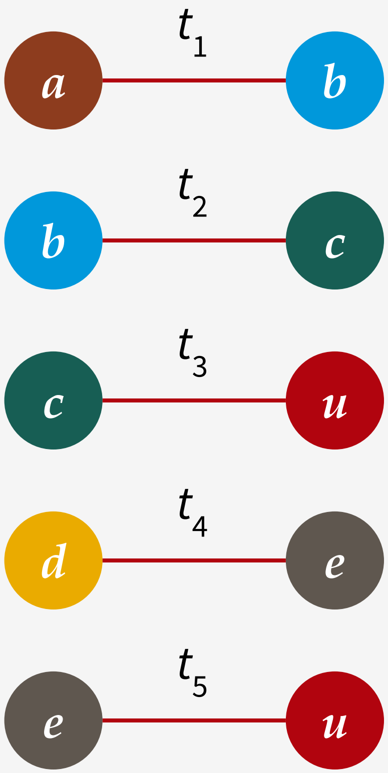
Róbert Pálovics

MTA SZTAKI

Stanford University

Temporal networks

- Network changes over time
- Edge stream
  - time series of edges: each link has a timestamp
  - edges may occur several times
  - example: Twitter mention network

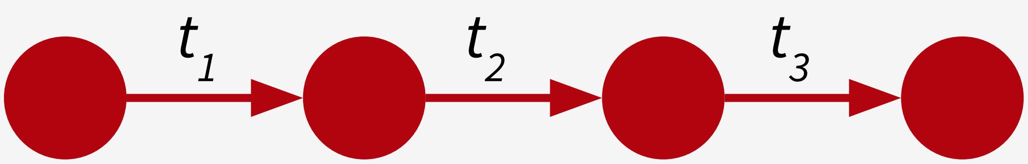


Objective

- Define network centrality metrics that are
  - temporal, reflect changes in the edge stream
  - online updatable

Time respective paths

- Adjacent edges that are ordered in time
- Models a flow, e.g.
  - information flow in social networks
  - flow of funds or goods in the economy
- Concept
  - delay t_2-t_1 is small, then flow is more likely

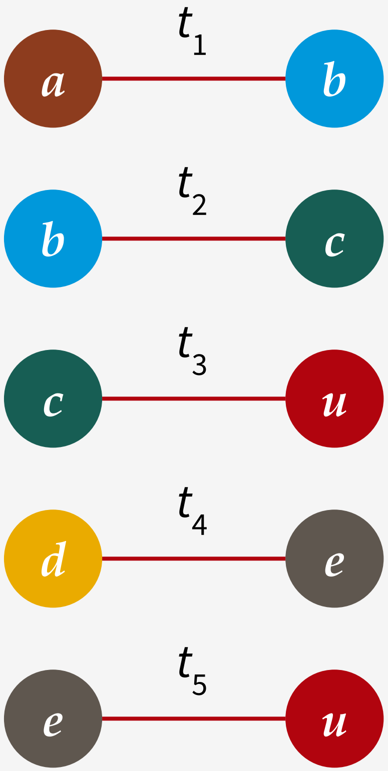


$t_1 < t_2 < t_3$

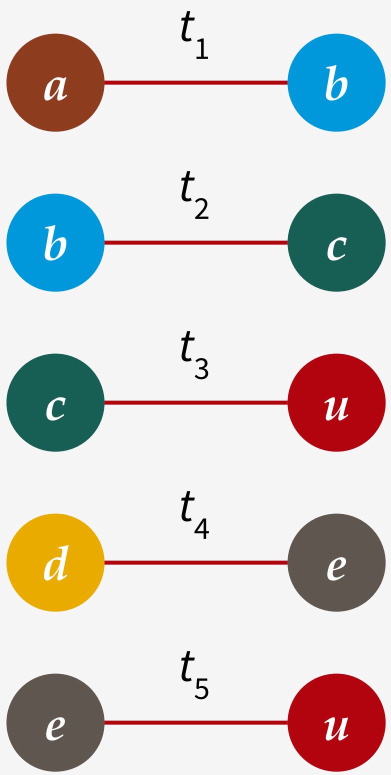
Temporal Katz Centrality

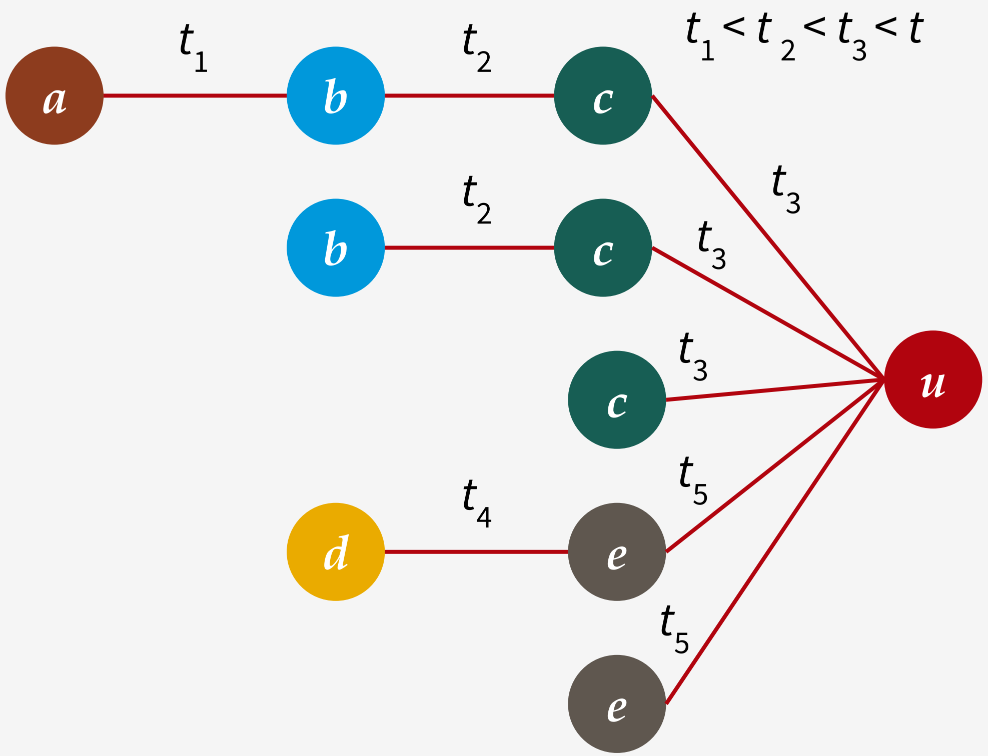
Definition: weighted sum of all time respecting walks that end in node *u*

Temporal Katz Centrality



Temporal Katz Centrality





Temporal Katz Centrality

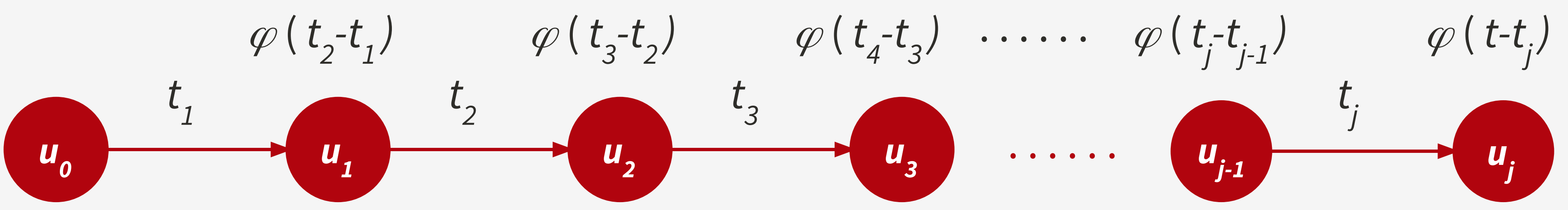
- Centrality for node *u* at time *t*

$$r_u(t) := \sum_{v} \sum_{\substack{\text{temporal paths } z \\ \text{from } v \text{ to } u}} \Phi(z,t)$$

- where $\Phi(z,t)$ is the weight of a single path:

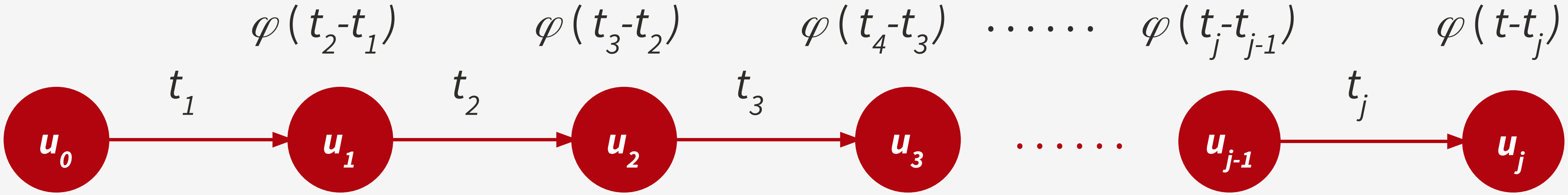
$$\Phi(z,t) := \prod_{i=1}^{j} \varphi(t_{i+1} - t_i)$$

- where edges appeared at $(t_1, t_2, \ldots, t_j)$ for walk z



Weighting functions

- Constant $\beta < 1$
  - $\varphi(\tau) = \beta$
  - walk length penalized with β
  - $\Phi(z,t) = \beta^{|z|}$
- Exponential decay
  - $\varphi(\tau) = \beta \exp\{-c\,\tau\}$
  - as $\varphi(a) \cdot \varphi(b) = \varphi(a+b)$, for an arbitrary path
  - $\Phi(z,t) = \beta \exp\left(-c\left[t - t_j\right]\right) \ldots \beta \exp\left(-c\left[t_2 - t_1\right]\right) = \beta^{|z|} \exp\left(-c\left[t - t_1\right]\right)$

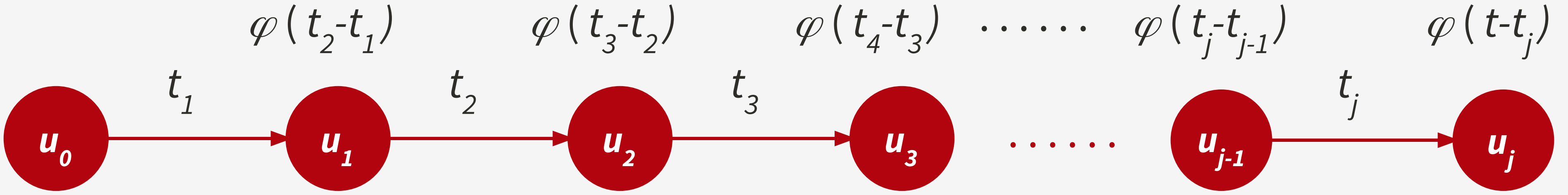


Temporal Katz Centrality

$$r_u(t) := \sum_{v} \sum_{\substack{\text{temporal paths } z \\ \text{from } v \text{ to } u}} \Phi(z, t)$$

$$\Phi(z, t) := \prod_{i=1}^{j} \varphi(t_{i+1} - t_i)$$

$$\Phi(z, t) = \beta \exp\left(-c\left[t - t_j\right]\right) \ldots \beta \exp\left(-c\left[t_2 - t_1\right]\right) = \beta^{|z|} \exp\left(-c\left[t - t_1\right]\right)$$



Relation to Katz Centrality

- Katz centrality

$$\vec{\text{Katz}} = \mathbf{1} \cdot \sum_{k=0}^{\infty} \beta^k A^k,$$

$$\vec{\text{Katz}}(u) := \sum_{v} \sum_{k=0}^{\infty} \beta^k |\{\text{paths of length } k \text{ from } v \text{ to } u\}|$$

- Given an underlying graph with edge set of size *E*
- We sample uniform random *T* edges

Goal: calculate the expected value of temporal Katz centrality

Expected value - $\varphi := \beta$

- Given an underlying graph with edge set of size *E*
- We sample uniform random *T* edges
- The expected number of times the edges of a *given* path of length *k* appear in a given order:

$$s_{T,k} = \binom{T}{k} \cdot E^{-k}$$

as a given edge has a probability of 1/E to appear at a given position

Expected value - $\varphi := \beta$

- The expected number of times the edges of a *given* path of length k appear in a given order:

$$s_{T,k} = \binom{T}{k} \cdot E^{-k}$$

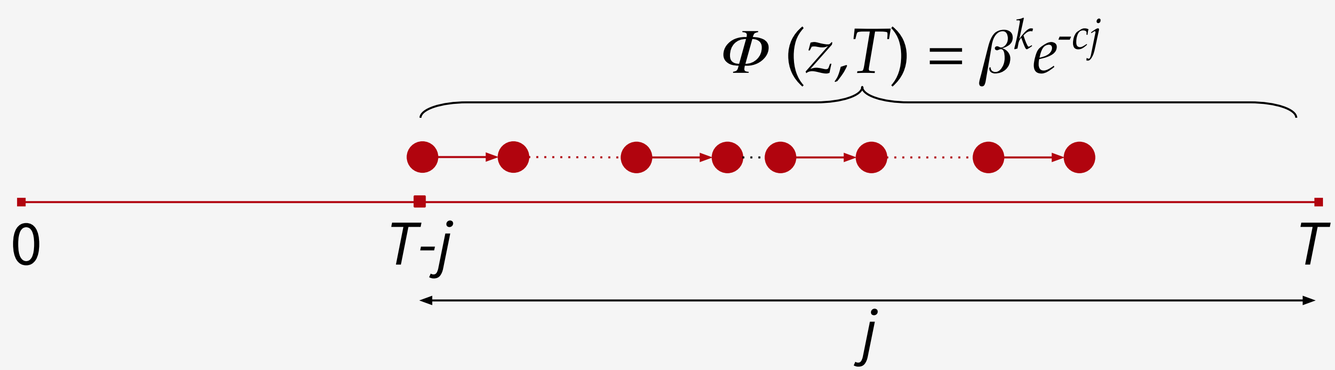
$$\overrightarrow{\mathrm{TemporalKatz}} = \mathbf{1} \cdot \sum_{k=0}^{K} \beta^k A^k \binom{T}{k} \cdot E^{-k} \simeq \mathbf{1} \cdot \sum_{k=0}^{K} \beta^k A^k (T/E)^k / k!$$

Expected value - exponential decay

- $s_{T,k}$: the expected total weight of a *given* path of length *k*

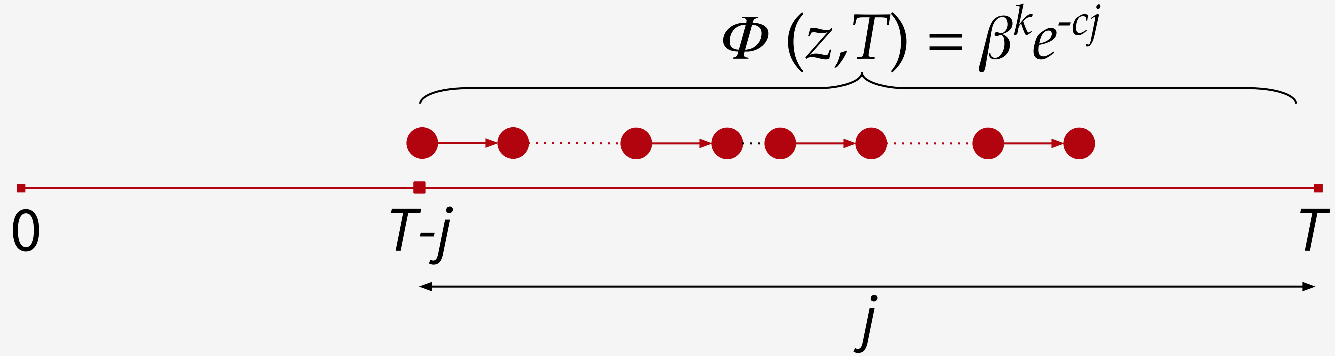
$$\vec{\text{TemporalKatz}} = \lim_{T\to\infty} \mathbf{1} \cdot \sum_{k=0}^{K} A^k s_{T,k} = \mathbf{1} \cdot \sum_{k=0}^{K} A^k \lim_{T\to\infty} s_{T,k}$$

- Each occurrence of a path of length *k* starting at time $(T - j)$ has the weight $\beta\, k \exp(-cj)$



Expected value - exponential decay

- Each occurrence of a path of length k starting at time $(T - j)$ has the weight $\beta\, k$ exp $(-cj)$



$$s_{T,k} = \beta^k \frac{1}{E^k} \sum_{j=k}^{T} \binom{j-1}{k-1} e^{-cj}$$

Expected value - exponential decay

Since $\sum_{n=m}^{\infty} \binom{n}{m} x^n = x^m/(1-x)^{m+1}$,

$$\begin{aligned}
\lim_{T\to\infty} s_{T,k} &= \lim_{T\to\infty} \left(\frac{\beta}{E}\right)^k \sum_{j=k}^{T} \binom{j-1}{k-1} e^{-cj} \\
&= \left(\frac{\beta}{E}\right)^k e^{-c} \sum_{j=k}^{\infty} \binom{j-1}{k-1} e^{-c(j-1)} \\
&= \left(\frac{\beta}{E}\right)^k \frac{e^{-ck}}{(1-e^{-c})^k} = \left(\frac{\beta}{E}\right)^k \frac{1}{(e^c-1)^k}.
\end{aligned}$$

Expected value - exponential decay

$$\overrightarrow{\text{TemporalKatz}} = \mathbf{1} \cdot \sum_{k=0}^{K} A^k \lim_{T \to \infty} s_{T,k} = \mathbf{1} \cdot \sum_{k=0}^{K} A^k \left(\frac{\beta}{E}\right)^k \left(\frac{1}{e^c - 1}\right)^k.$$

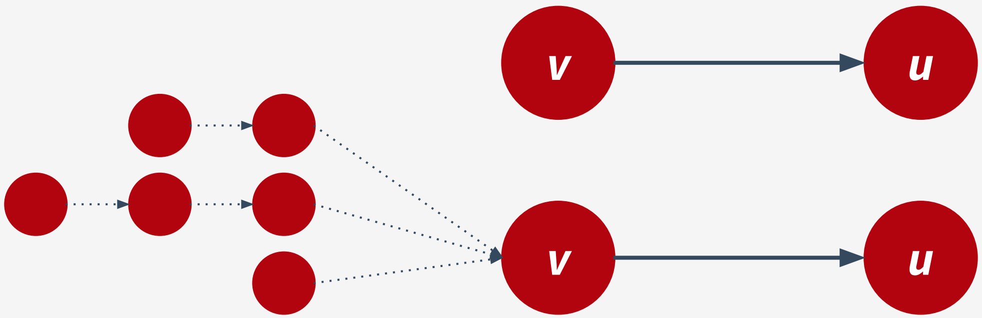
- let *c*:= *c'*/*E* with *c'* <<*E*
- hence *c*/*E* << 1 and exp{*c*} = exp{*c'*/*E*} ≈ 1 + c'/*E*

$$\overrightarrow{\text{TemporalKatz}} = \mathbf{1} \cdot \sum_{k=0}^{K} A^k \left(\frac{\beta}{E}\right)^k \left(\frac{1}{1 + c'/E - 1}\right)^k = \mathbf{1} \cdot \sum_{k=0}^{K} A^k \left(\frac{\beta}{c'}\right)^k$$

Temporal Katz converges to static Katz on uniformly sampled edge streams

Temporal Katz Centrality - computation

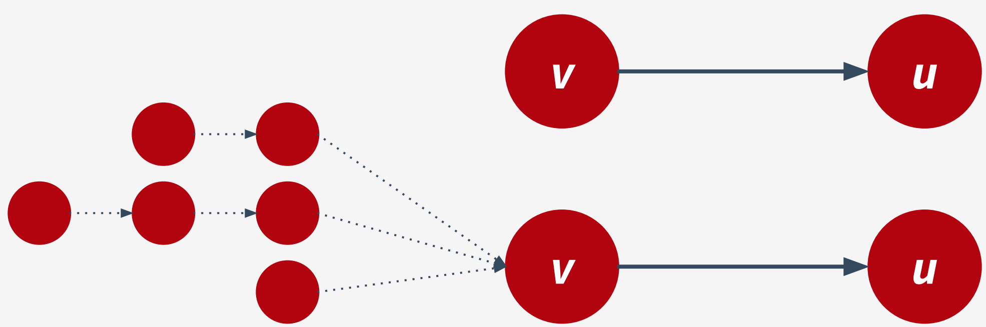
- When edge *vu* appears at time t_{vu}
- The centrality of node *u* at time *t* increases as
  - a new time respecting walk appears
  - all walks that ended in in *v* continue via edge *vu* to *u*



Temporal Katz Centrality - computation

- When edge *vu* appears at time t_{vu}
- The centrality of node *u* at time *t* increases as
  - a new time respecting walk appears
  - all walks that ended in in *v* continue via edge *vu* to *u*
- Hence the total increase is

$$(1 + r_v(t_{vu})) \cdot \varphi(t - t_{vu})$$



Temporal Katz Centrality - computation

- Recursive definition

$$r_u(t) = \sum_{vu \in E(t)} (1 + r_v(t_{vu}))\, \varphi(t - t_{vu})$$

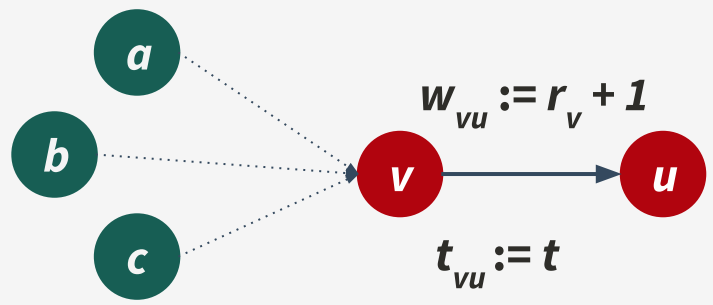
- Note that $w_{vu} := r_v(t_{vu})$ does not depend on time!

Temporal Katz Centrality - computation

- For each node u we initialize $r(u) := 0$
- We maintain $r_u(t)$, w_{vu} and t_{vu}
- When edge uv appears
  - we calculate the current value of r_v

$$r_v := \sum_{zv \in E(t)} w_{zv} \cdot \varphi(t - t_{zv})$$

  - $w_{vu} := r_v + 1$
  - $t_{vu} := t$



Special case: exponential decay

- as $\varphi(a) \cdot \varphi(b) = \varphi(a + b)$

$$
\begin{aligned}
r_v &:= r_v \cdot \varphi(t - t_v); \\
r_u &:= r_u \cdot \varphi(t - t_u) + (r_v + 1) \cdot \beta; \\
t_u &:= t, \quad t_v := t,
\end{aligned}
$$

update r_v

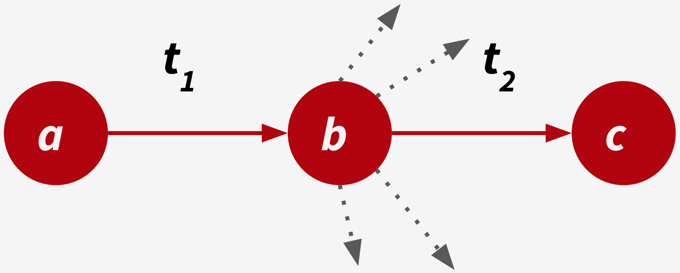
add new walks ending in *u*

update old walks ending in *u*

- no need to store w_{vu}

Related work

- Related result: Temporal PageRank
- Polina Rozenstein & Aris Gionis
- Different φ () weighting function
  - for adjacent edges (a, b, t_1) and (b, c, t_2)
  - L := number of edges (b, x, t) where $t_1 < t < t_2$
  - $\varphi \sim |\alpha|^L$, $\alpha < 1$

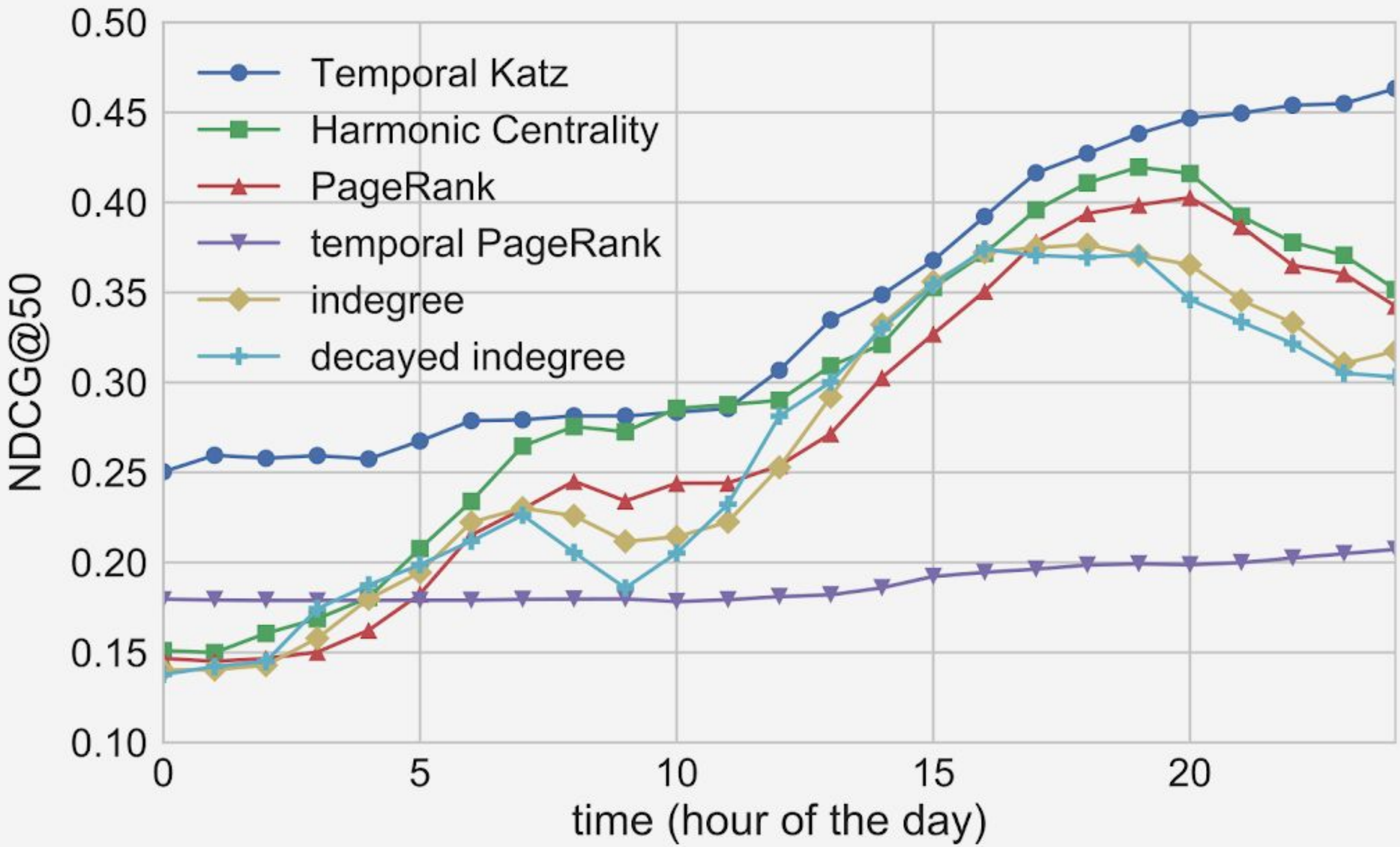


Experiments on Twitter data

- Centrality metrics are difficult to evaluate overall
- Current need: temporal network with temporal labels
- Data: Twitter mentions during a tennis tournament
  - edges: user mentions
  - labels: players participating on a given day

1	Roland-Garros	@rolandgarros	0
2	Stanislas Wawrinka	@stanwawrinka	1
3	Andy Murray	@andy_murray	1
4	Simona Halep	@Simona_Halep	0
5	Rafa Nadal	@RafaelNadal	1
6	Dominic Thiem	@ThiemDomi	1
7	Timea Bacsinszky	@TimeaOfficial	0
8	Rohan Bopanna	@rohanbopanna	0
9	Ana Ivanovic	@AnaIvanovic	0
10	WTA	@WTA	0
11	Gaby Dabrowski	@GabyDabrowski	0
12	Tennis Channel	@TennisChannel	0
13	Rafa Nadal Academy	@rnadalacademy	0
14	Karolina Pliskova	@KaPliskova	0
15	yonex.com	@yonex_com	0
16	Gusti Fernandez	@gustifernandez4	0
17	rolandgarrosFR	@rolandgarros_FR	0
18	Eurosport.es	@Eurosport_ES	0
19	ATP World Tour	@ATPWorldTour	0
20	Caroline Garcia	@CaroGarcia	0

Experiments on Twitter data

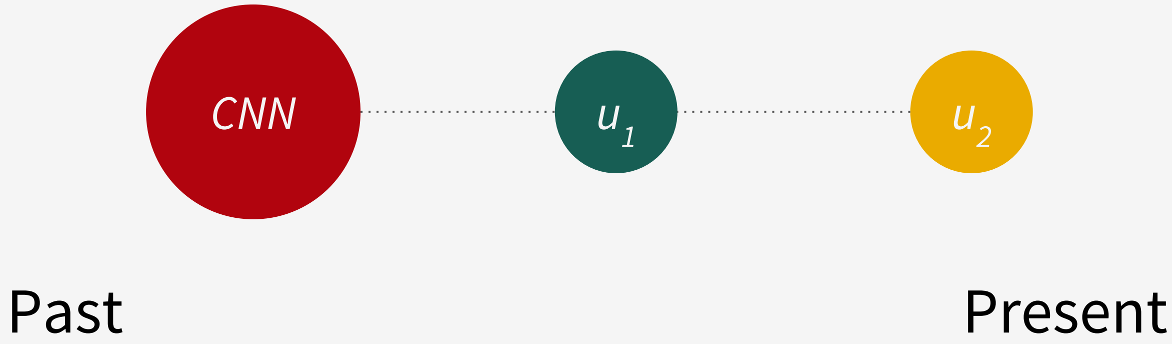


Temporal Katz Centrality - Summary

- Defined over edge time series
- Sum of time-respecting walks ending in a node
- Related to static Katz-centrality
- Online updateable

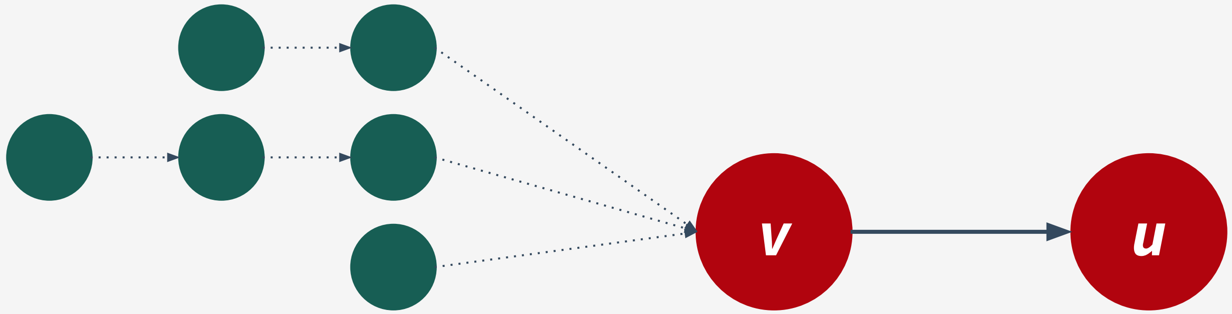
Temporal Katz - ongoing work

- Weighted count of walks starting from a node
- Past node may be more relevant, e.g. on Twitter:



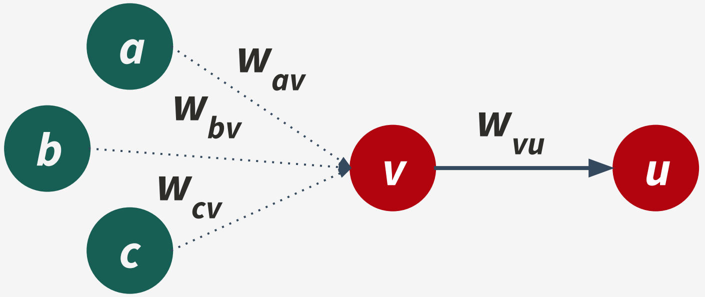
Temporal Katz - ongoing work

- Issue: no simple online update rule
- When *vu* edge appears, all prev. nodes in the temporal walks should be updated



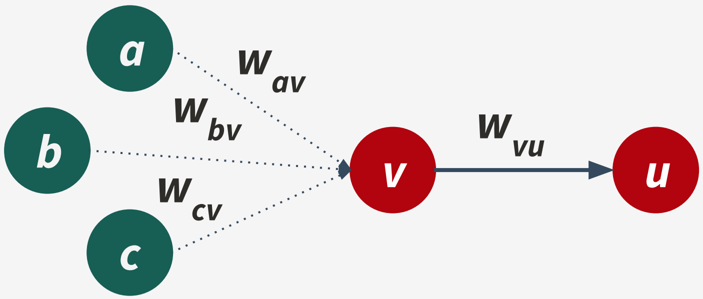
Temporal Katz - ongoing work

- Compute temporal Katz, maintain w_{vu}
- i.e. for every edge (v, u,t_{vu}) store the weighted sum of walks from v to u at time t_{vu}
- Define a time resp. random walk over the graph
  - at node v
  - stop with probability $1 / (1 + r_v(t_{vu})) = 1 / (1 + w_{vu})$
  - if continue, select edges with $t < t_{vu}$ proportional to their weight



Temporal Katz - ongoing work

- Define a time resp. random walk over the graph
  - at node *v*
  - stop with probability $1 / (1 + r_v(t_{vu})) = 1 / (1 + w_{vu})$
  - if continue, select edges with $t < t_{vu}$ proportional to their weight



Results in a time respecting walk sampled proportional to its weight

Temporal Katz - ongoing work

- Hence if we maintain the edge weights
- Anytime walks can be sampled proportional to their weights
- By sampling walks we can approximate the number of tr. walks starting from a node

Network Node Embeddings

Network node embeddings - ongoing work

- Find embedding of nodes to *d* dimensions
- Similar nodes in the graph have embeddings that are close together
- Already existed, e.g. recSys MF, graph factorization
- Revisited: random walk based methods

Network node embeddings - ongoing work

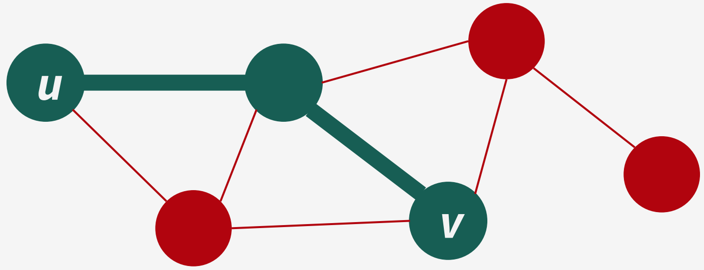
- Simple graph factorization loss function

$$L = \Sigma_{uv}[\, p_u p_v - A_{uv}]^2$$

  - model parameters: p_u
  - optimization via SGD
- Random walk based methods

$$L = \Sigma_{D\,(u,v)} - \log(\, \mathrm{softmax}(p_u p_v\,)\,)$$

  - D is a set of generated random walks, e.g. k from each node

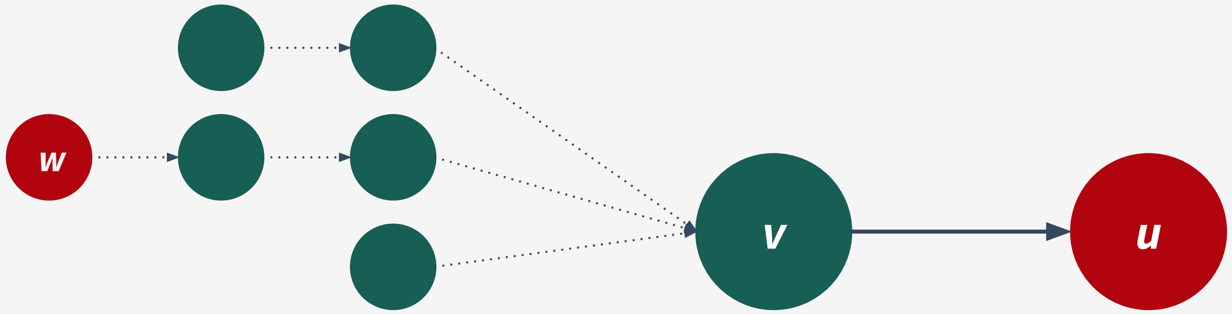


Network node embeddings - ongoing work

- Our objective
  - define an edge stream based
  - online updateable model
  - that generates temporal embeddings
  - adapts to concept drift

Network node embeddings - ongoing work

- Process edges in temporal order
- For each edge (*u*,*v*)
  - learn that p_u and p_v are similar
  - start random trw. walks (*w*,...,*u*,*v*) to the past ending in (*u*,*v*)
  - learn for (*w*,*v*) that p_u and p_v are similar



Summary

- Temporal Katz
  - Defined over edge time series
  - Sum of time-respecting walks ending in a node
  - Related to static Katz-centrality
  - Online updateable
  - Ongoing work: sum of paths starting from a common node in the path
- Network embeddings
  - Time respective walk based online updates