Learning with Temporal Point Processes: Models and Inference

Alexey Zaytsev

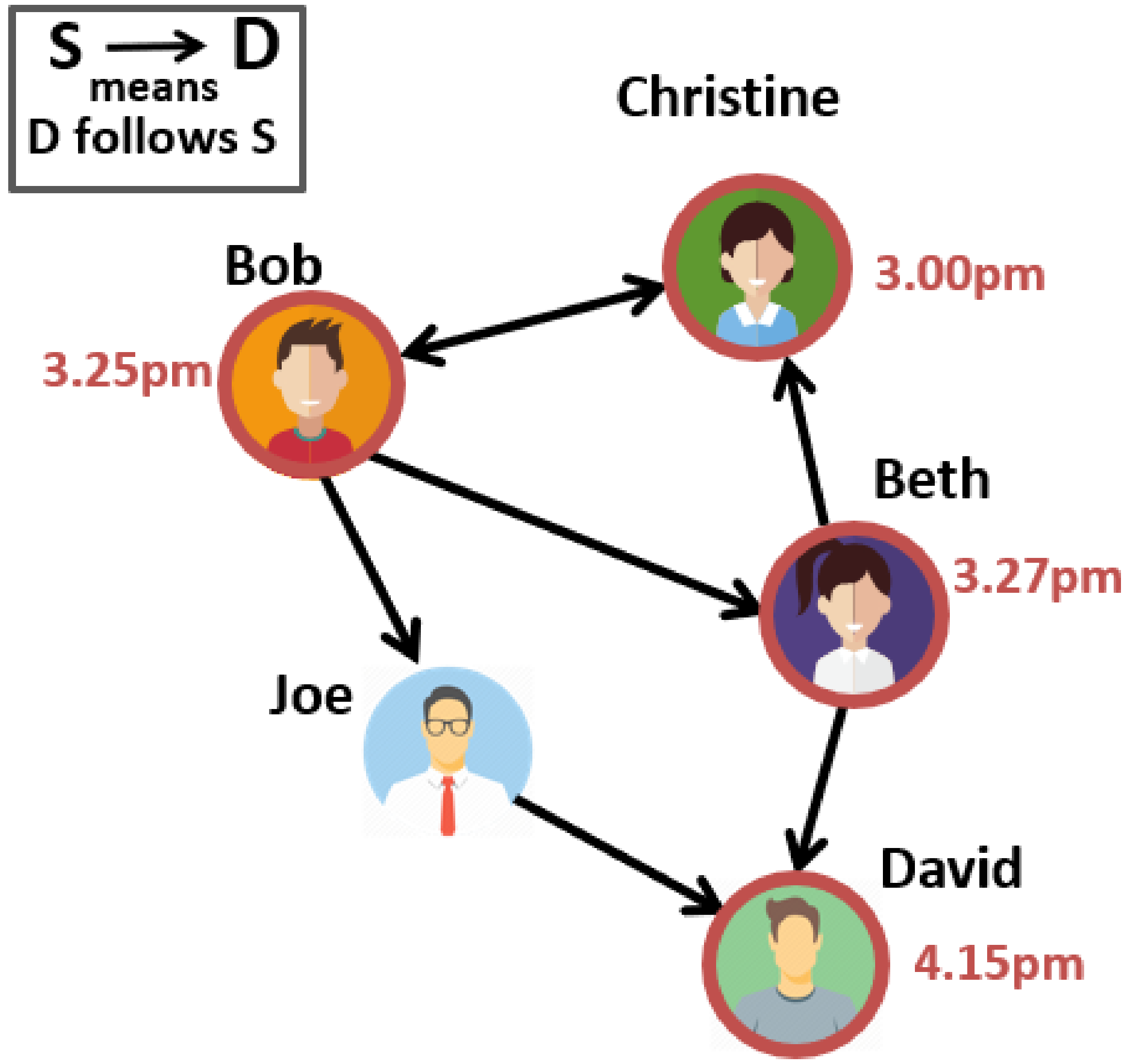
Skoltech

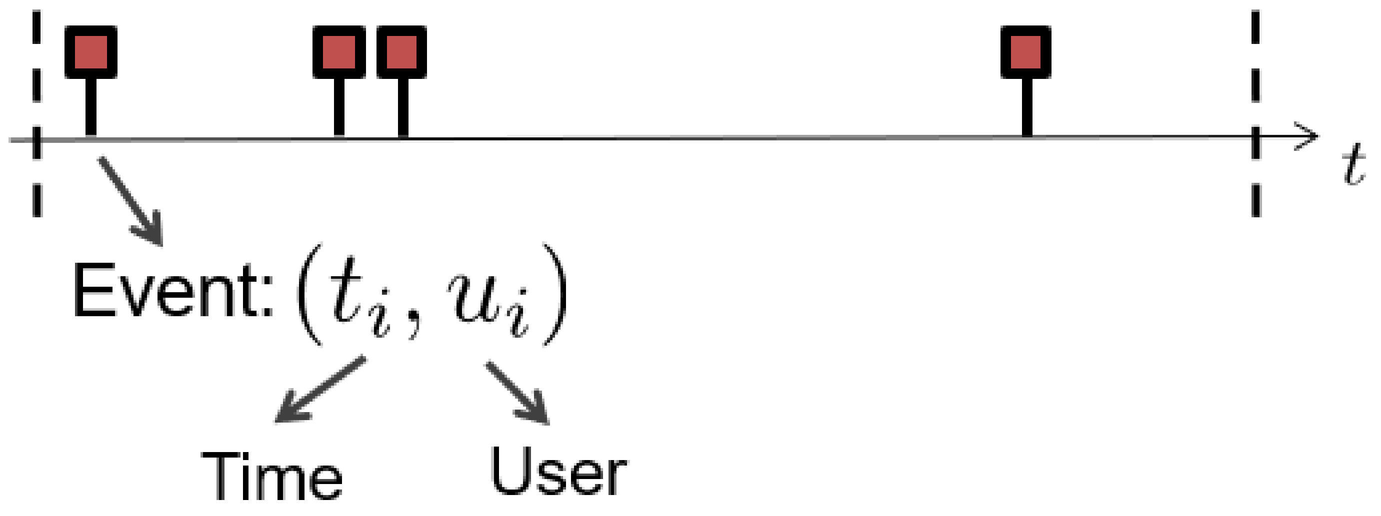
Based on ICML Tutorial, July 2018 by Isabel Valera

1. Modeling event sequences

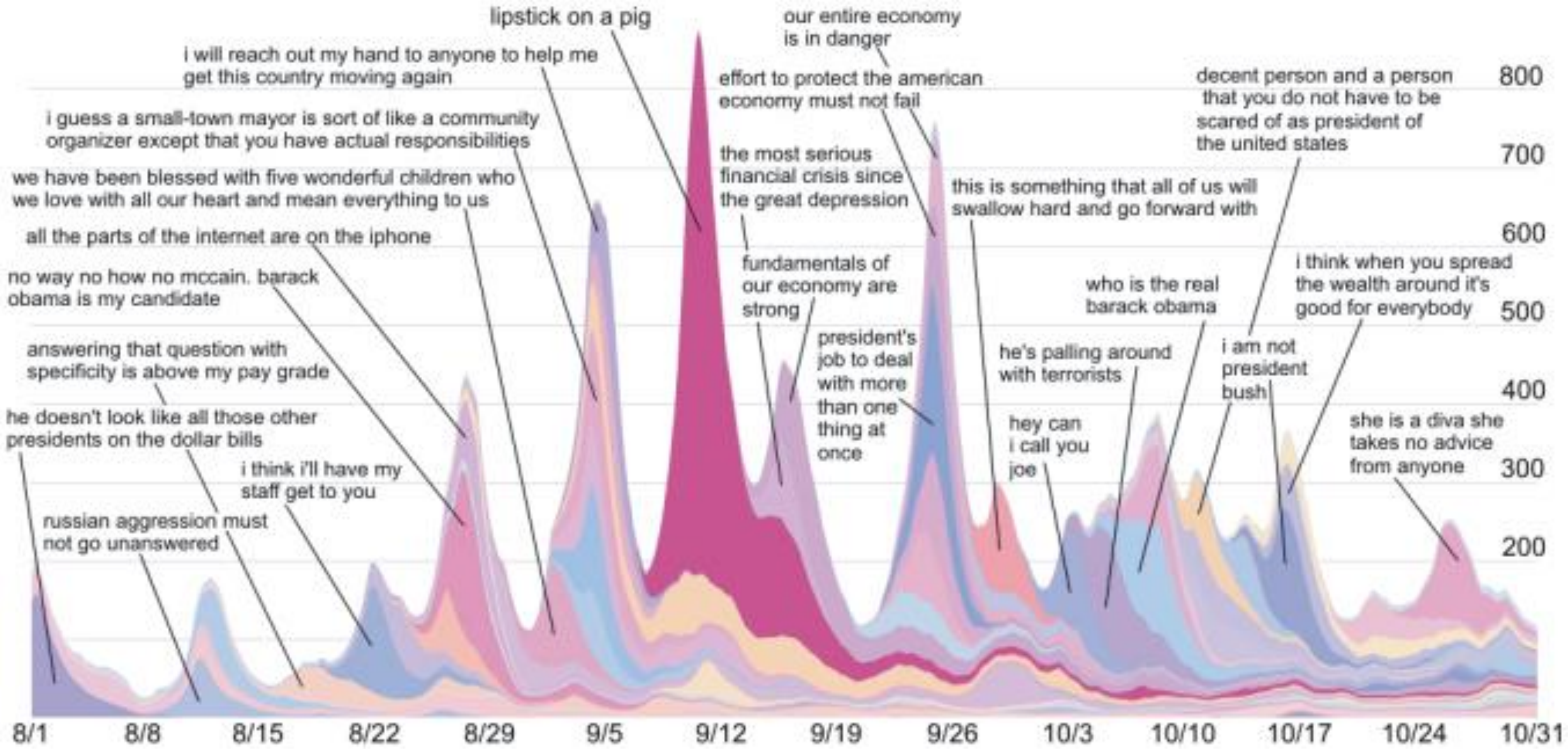
Skoltech

Event sequences as cascades



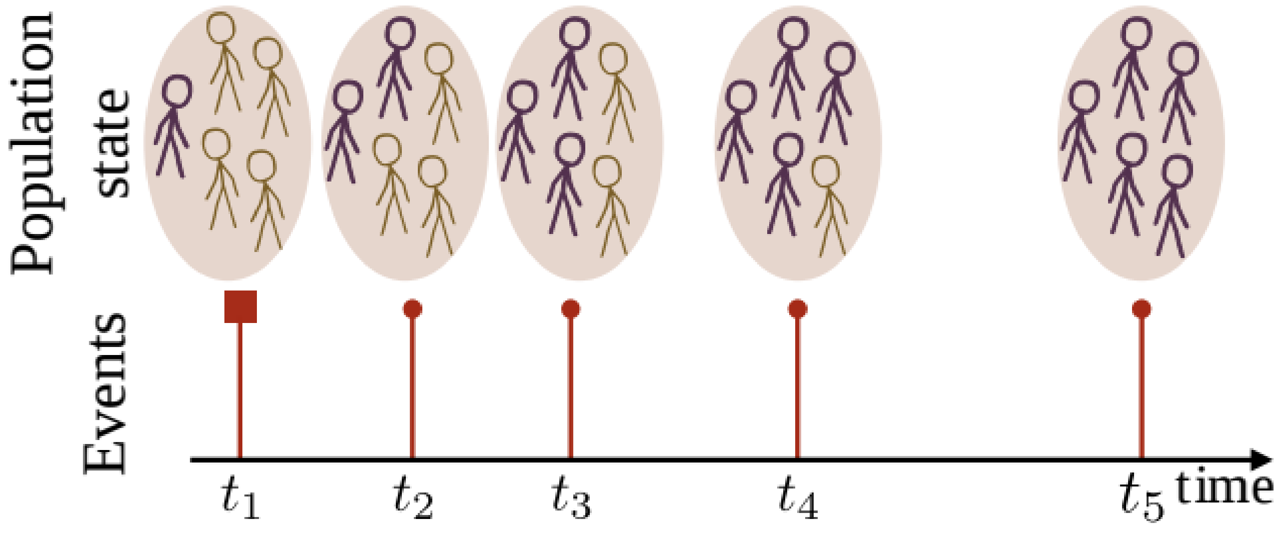


Information Diffusion



[Leskovec et al., 2009]

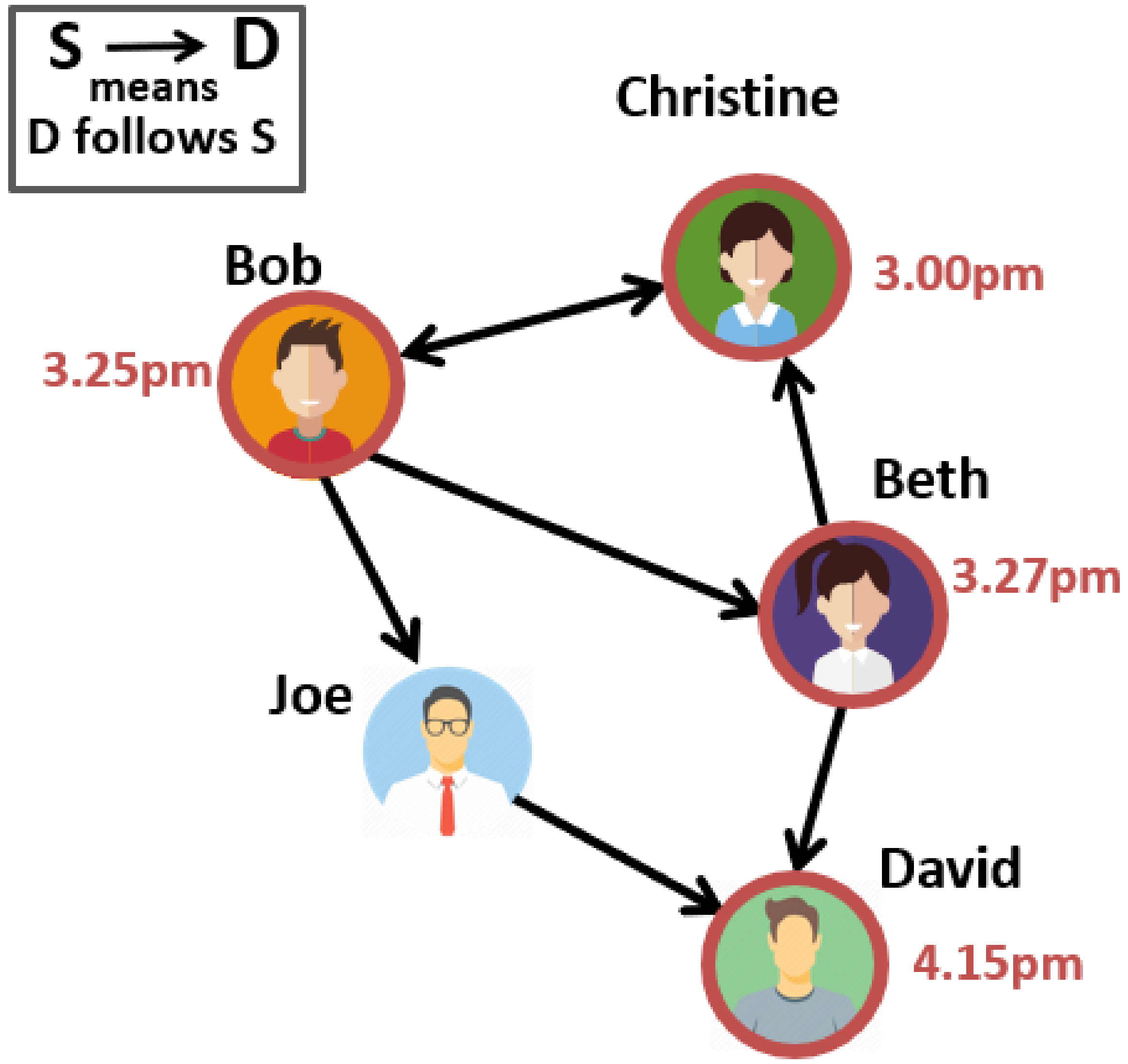
Disease Diffusion

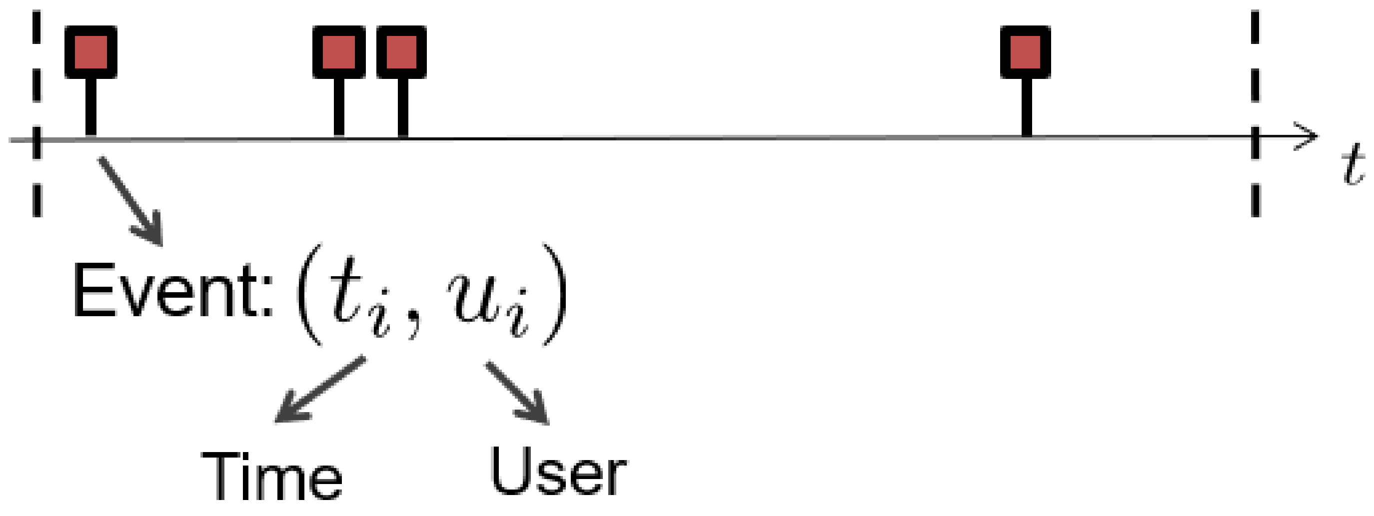


[Rizoiu et al., 2018]

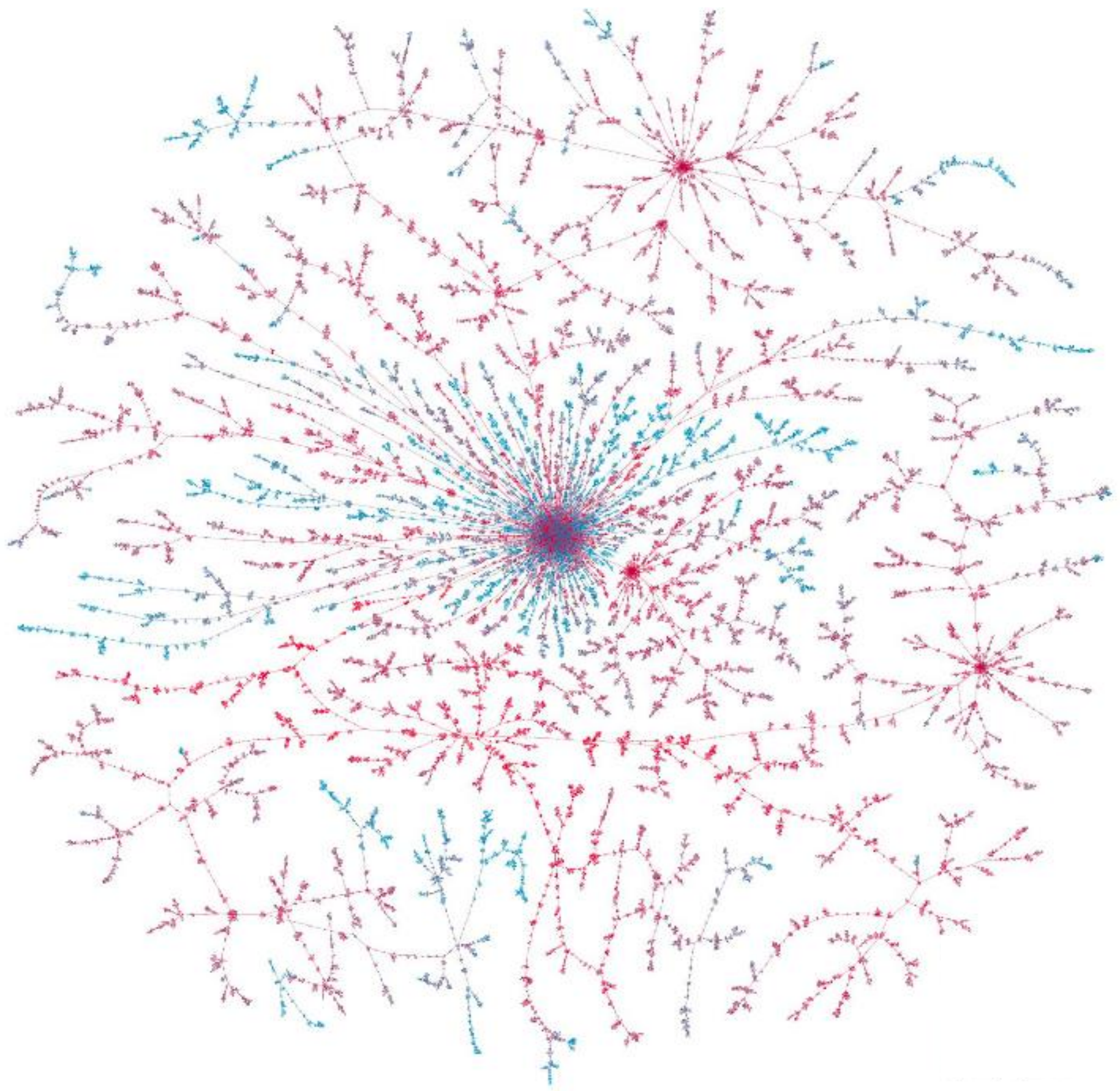
Skoltech

An example: idea adoption





They can have an impact
in the off-line world

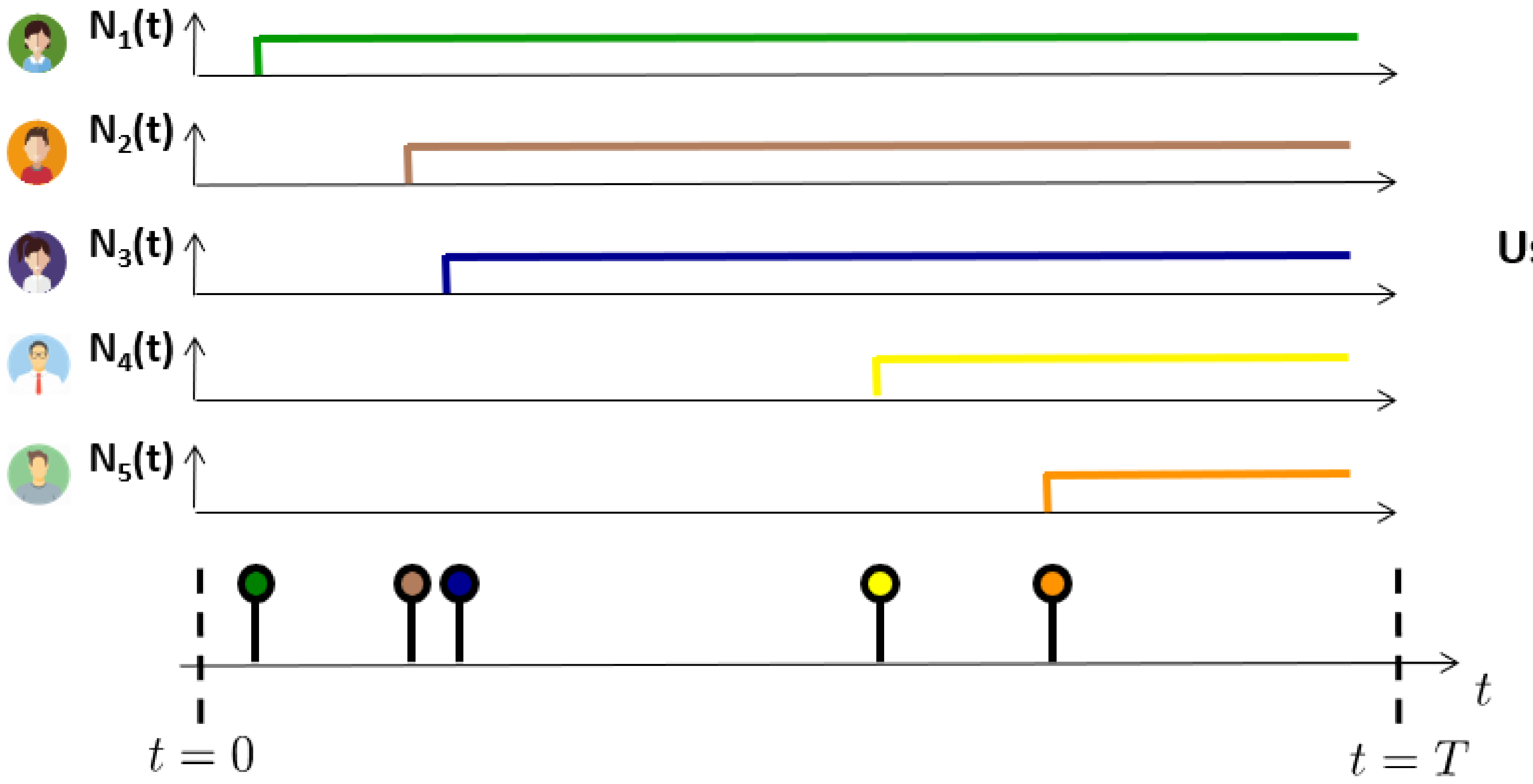


theguardian

Click and elect: how fake news helped Donald Trump win a real election

Infection cascade representation

We represent an infection cascade using **terminating temporal point processes**:



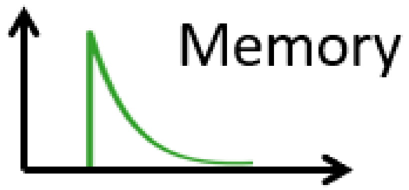
Infection event:

$$(u_i, m_i, t_i)$$

Infection intensity

Source
(given, not modeled)

Follow-ups
(modeled)



$$\lambda_u^*(t) = \underbrace{(1 - N_u(t))}_{\text{Users get infected only once}} \sum_{v \in [m]} b_{vu} \underbrace{\sum_{e_i \in \mathcal{H}_v(t)} \kappa(t - t_i)}_{\text{Previous infections of user v}}$$

Influence from user v on user u

[Gomez-Rodriguez et al., ICML 2011]

Model inference from multiple cascades

Conditional intensities

$$\lambda_u^*(t)$$

Diffusion log-likelihood

$$\mathfrak{L} = \sum_{u=1}^{n} \log \lambda_u^*(t_u) - \int_0^T \lambda_u^*(\tau)\, d\tau$$

Maximum likelihood approach to find model parameters!

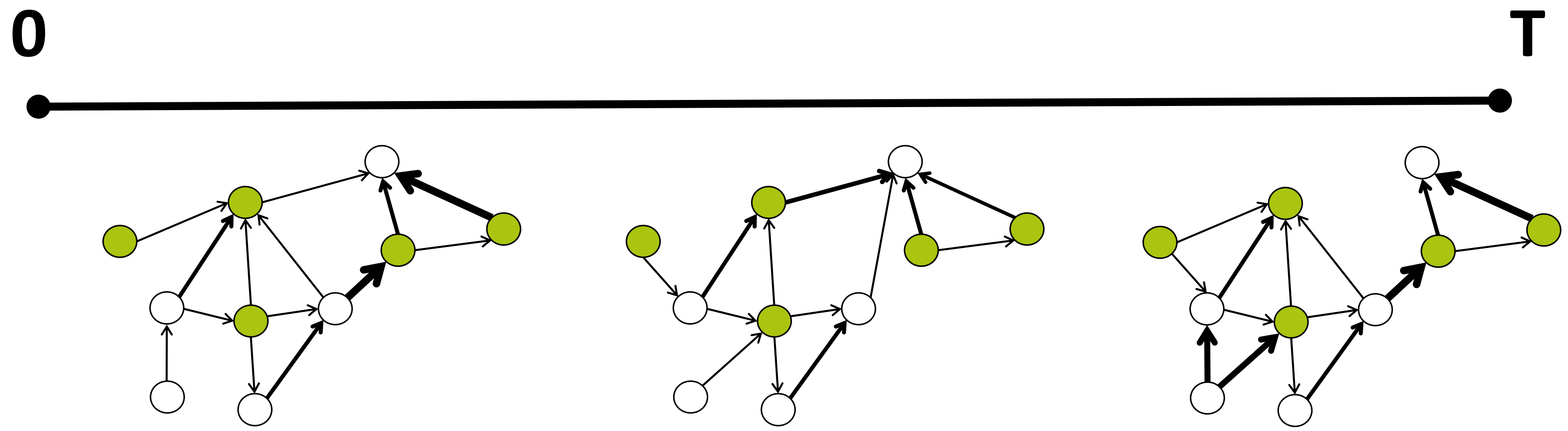
Sum up log-likelihoods of multiple cascades!

Theorem. For any choice of parametric memory, the **maximum likelihood** problem is **convex in B**.

Dynamic influence

In some cases, influence change over time:

Propagation over networks with variable influence

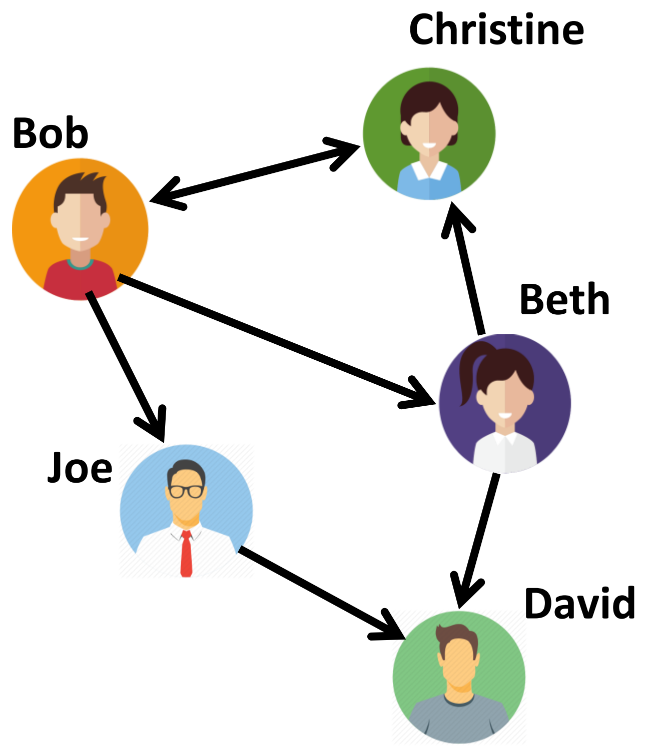


Properties are similar to static influence

[Gomez-Rodriguez et al., WSDM 2013]

Recurrent events: beyond cascades

Up to this point, each user is only infected once, and event sequences can be seen as cascades.



In general, users perform recurrent events over time. E.g., people repeatedly express their opinion online:

CNN
How social media is revolutionizing debates

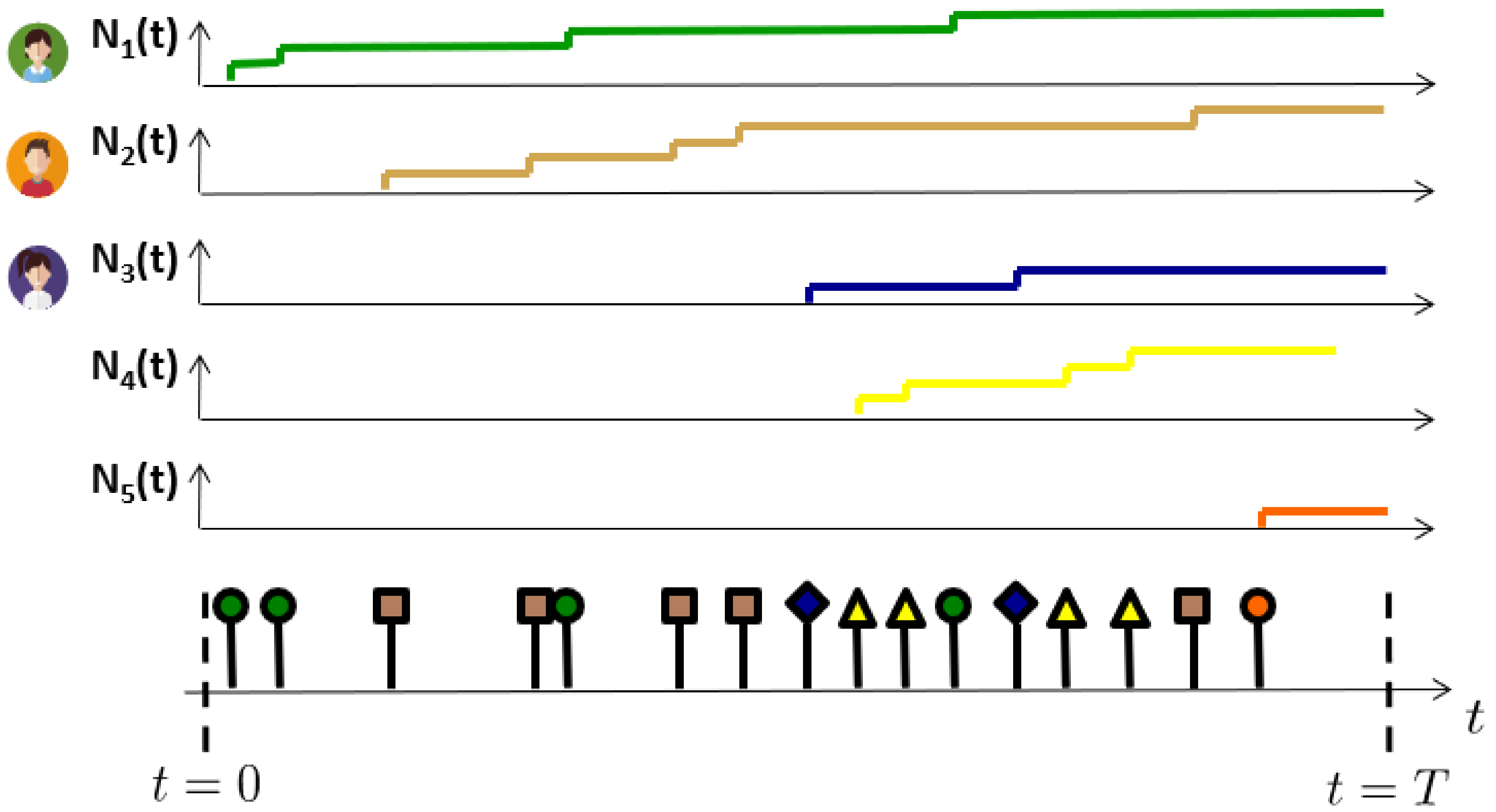
The New York Times
Social Media Are Giving a Voice to Taste Buds

TC
Twitter Unveils A New Set Of Brand-Centric Analytics

The New York Times
Campaigns Use Social Media to Lure Younger Voters

Recurrent events representation

We represent messages using **nonterminating temporal point processes**:



Recurrent event:

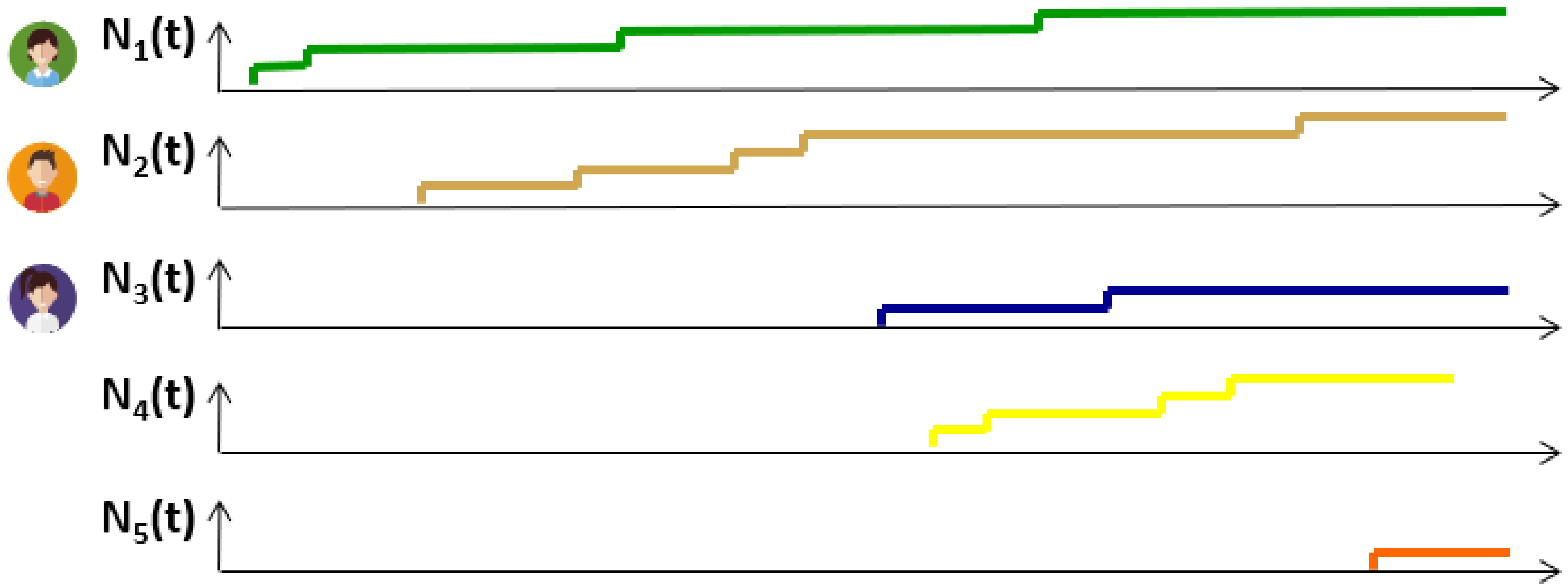
$$(u_i, t_i)$$

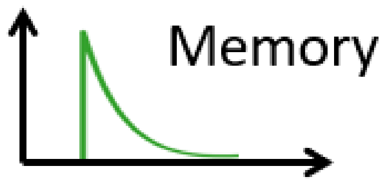
User ↙ ↘ Time

[Farajtabar et al., NIPS 2014]

Skoltech

Recurrent events intensity





Cascade sources!

$$\underbrace{\lambda_u^*(t)}_{\text{User's intensity}} = \underbrace{\mu_u}_{\text{Events on her own initiative}} + \sum_{v \in [m]} b_{vu} \underbrace{\sum_{e_i \in \mathcal{H}_v(t)} \kappa(t - t_i)}_{\text{Previous messages by user v}}$$

Influence from user v on user u

Hawkes process

[De et al., NIPS 2016]

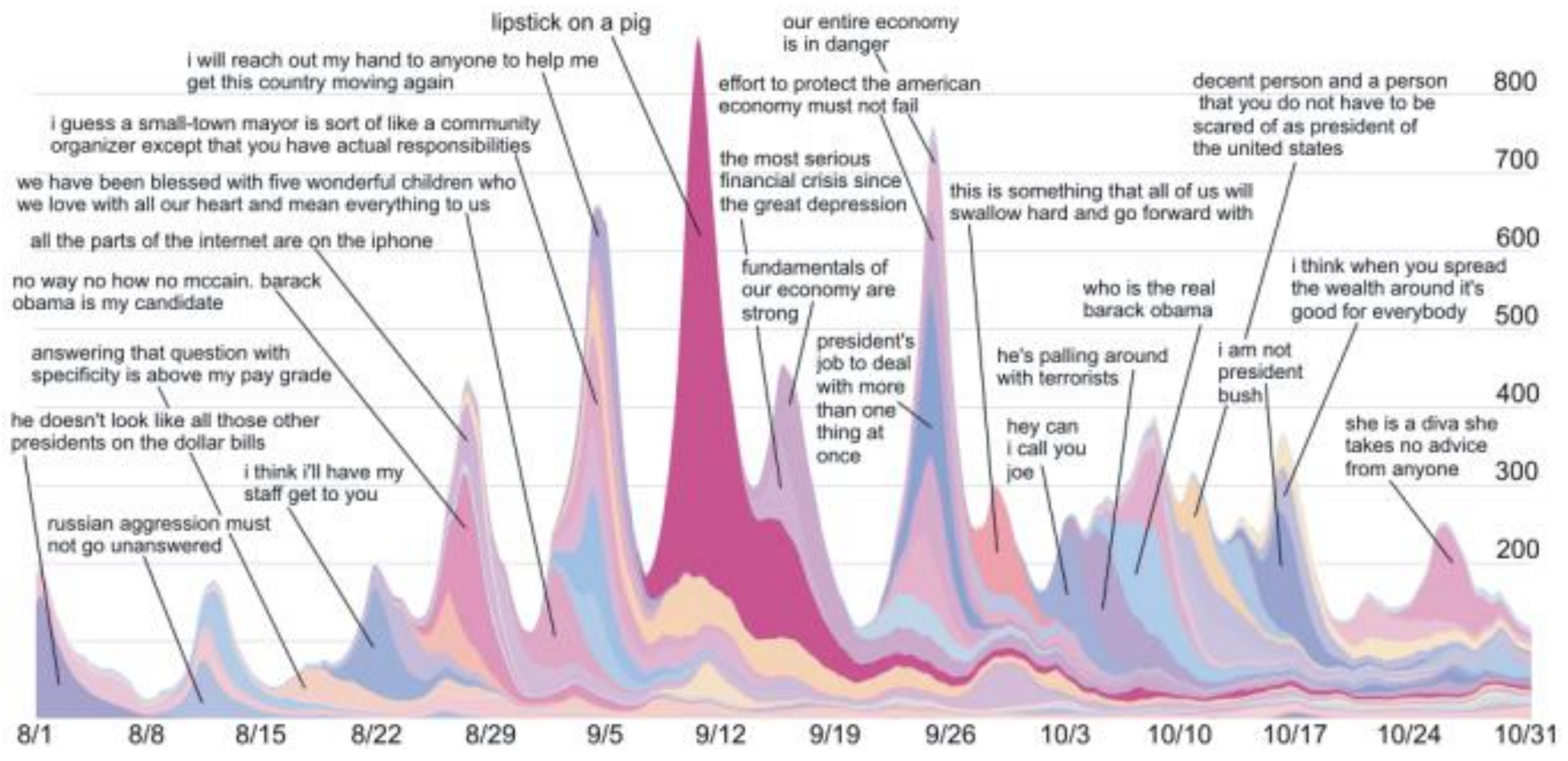
2. Clustering event sequences

Skoltech

Event sequences

So far, we have assumed the cascade (topic, meme, etc.) that each event belongs to was known.

Often, the cluster (topic, meme, etc.) that each event in a sequence belongs to is not known:



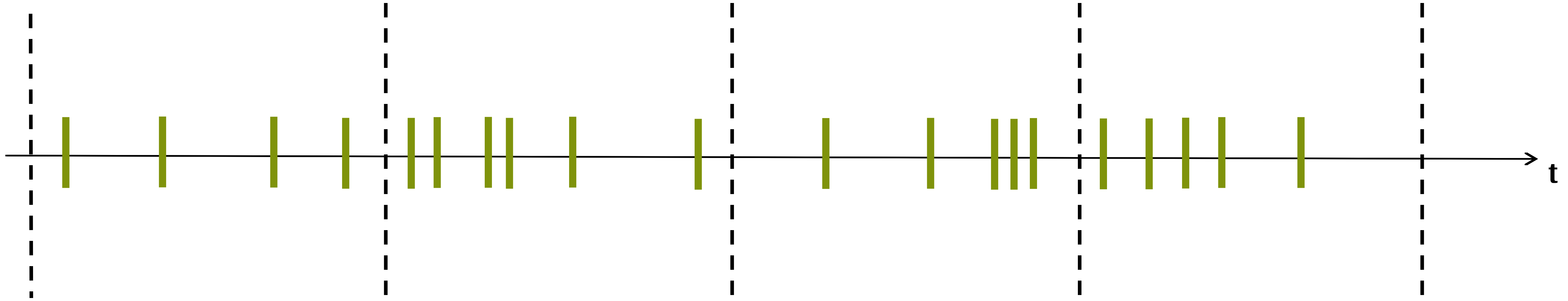
Politics

Music

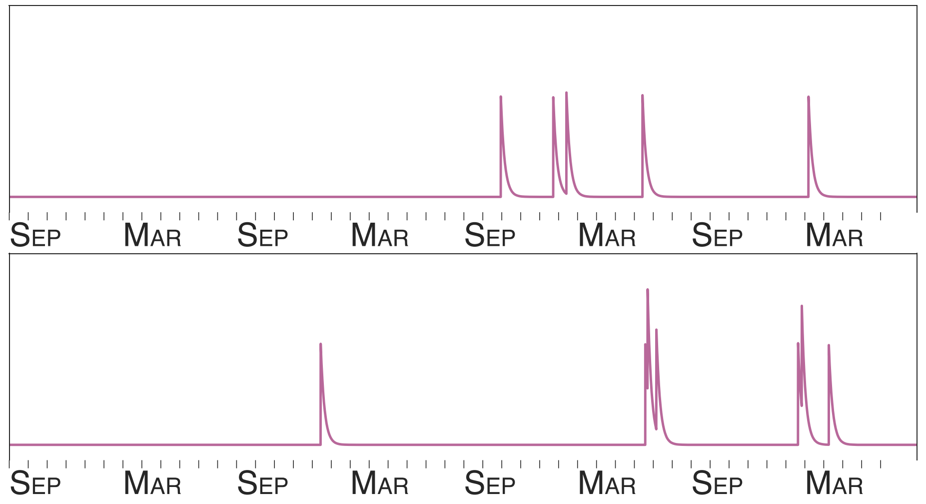
Politics

Clustering event sequences

Assume the event cluster to be hidden and aim to automatically learn the cluster assignments from the data:



Bayesian methods to cluster event sequences in the context of:



Learning

Online News

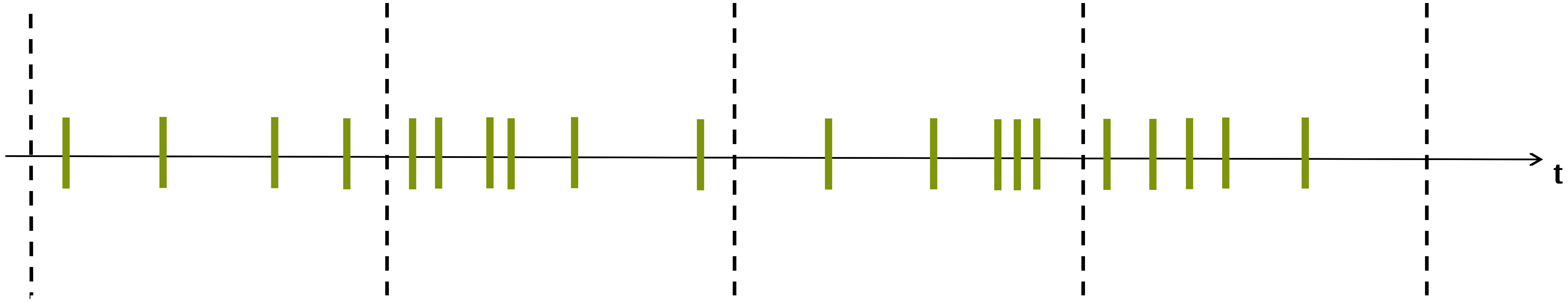
Method	DMHP
ICU Patient	**0.3778**
IPTV User	**0.2004**

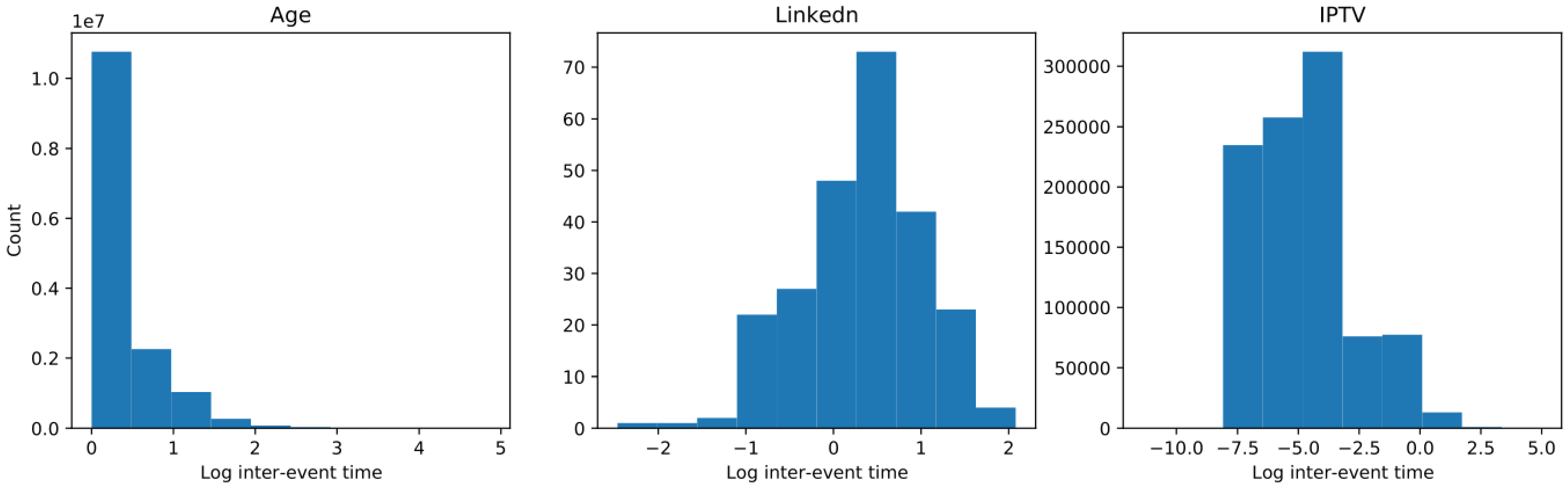
Health care

[Du et al., 2015; Mavroforakis et al., 2017; Xu & Zha, 2017]

Clustering event sequences

Assume the event cluster to be hidden and aim to automatically learn the cluster assignments from the data:





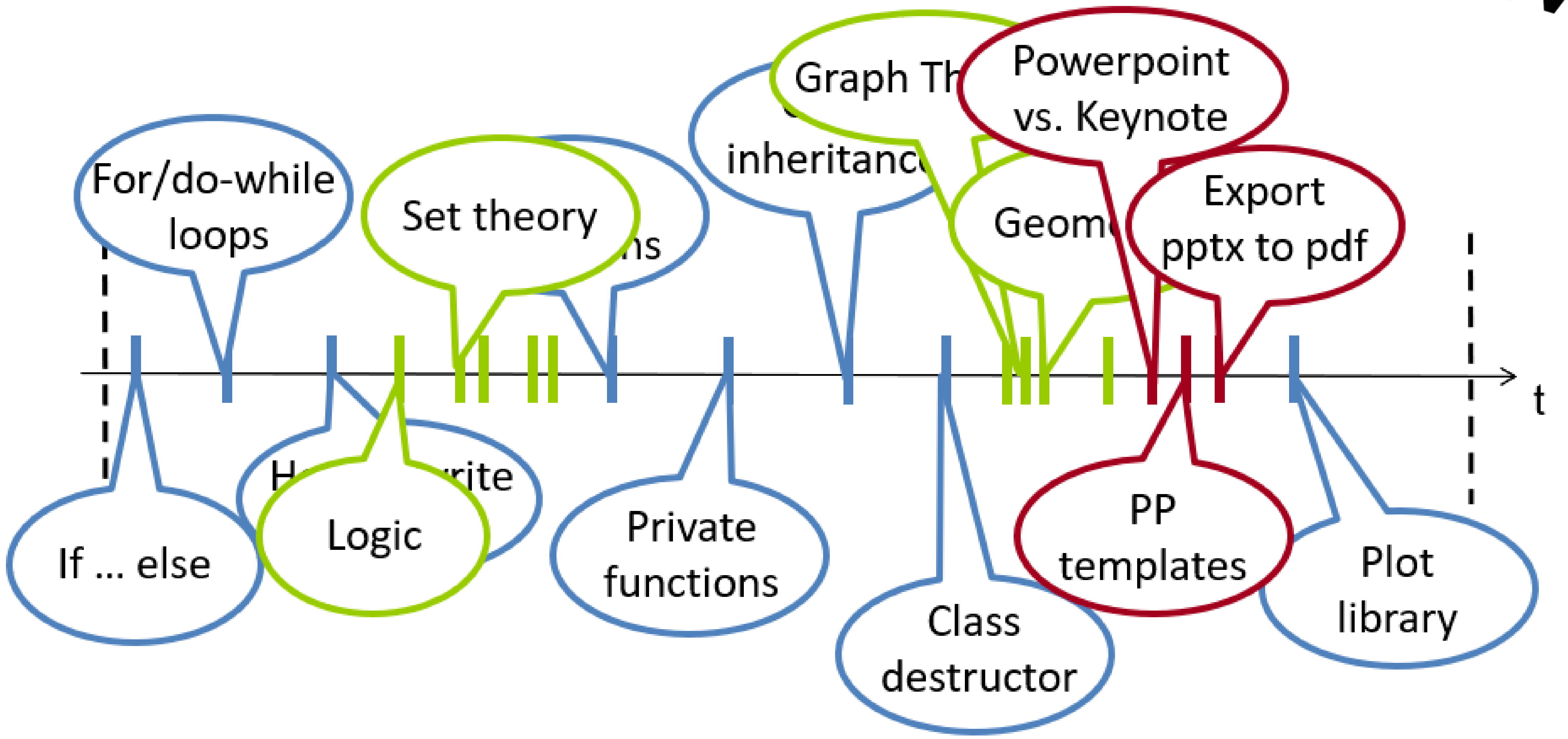
Financial transactions **Job changes from Linkedin** **TV programs, IPTV**

[Zhuzhel et al., 2021]

Hierarchical Dirichlet Hawkes process

1st year computer science student

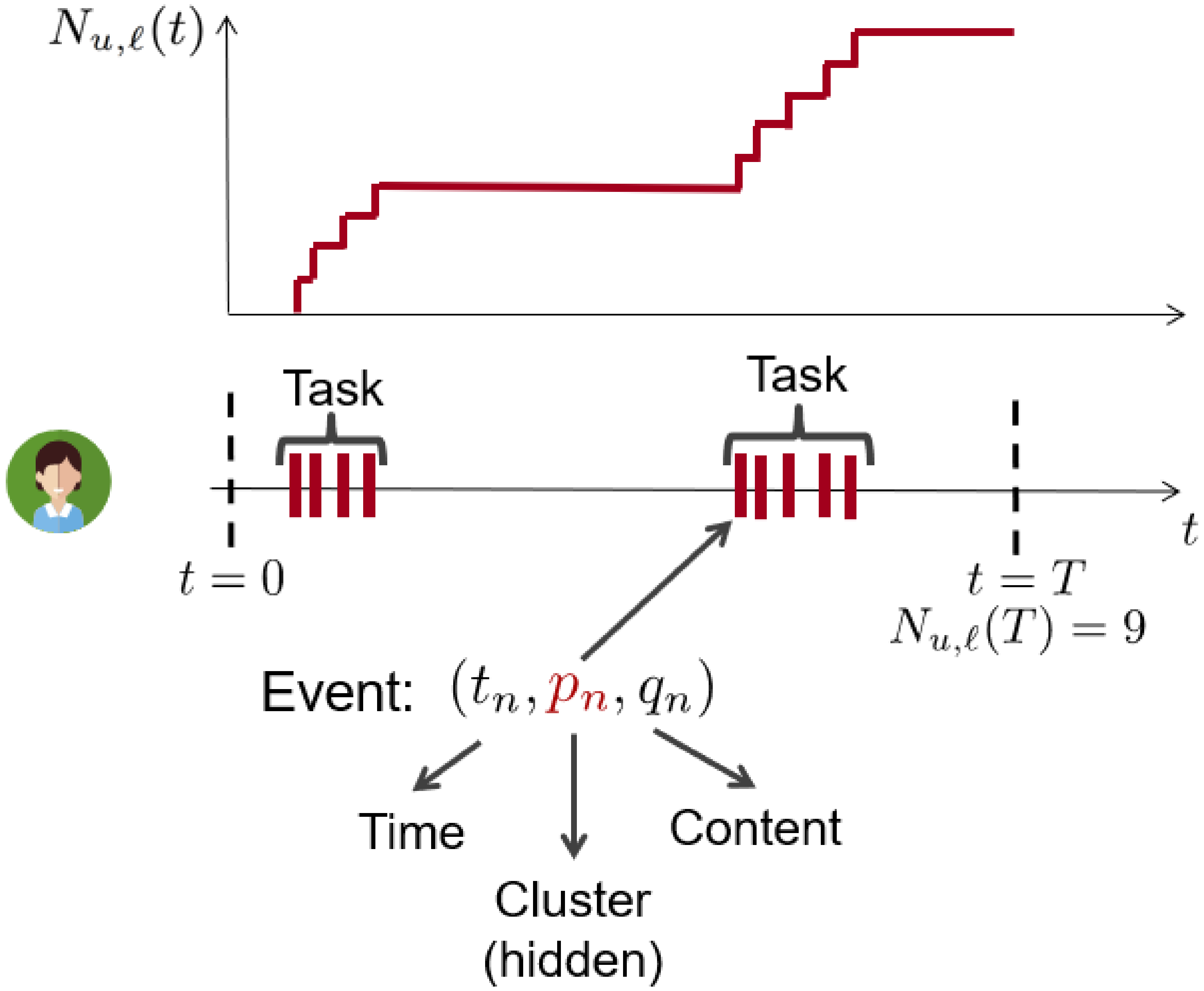
- Introduction to programming
- Discrete math
- Project presentation



[Mavroforakis et al., WWW 2017]

Events representation

We represent the events using marked temporal point processes:



[Mavroforakis et al., WWW 2017]

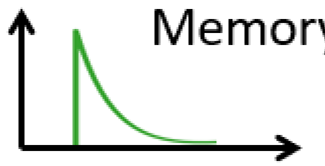
Skoltech

Cluster intensity

New cascade rate

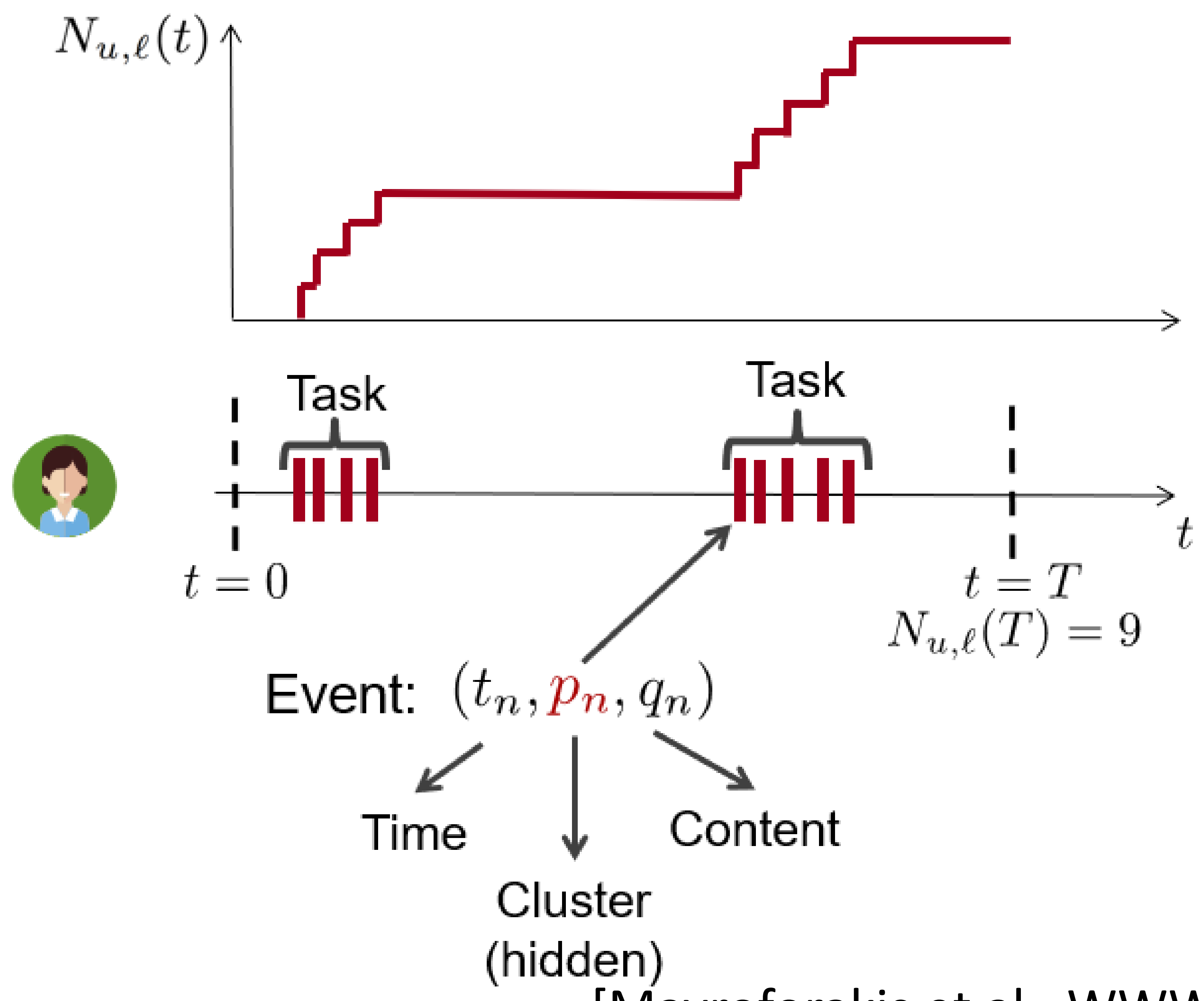
Cluster popularity

Memory



$$\underbrace{\lambda^*_{u,\ell}(t)}_{\substack{\textbf{Intensity} \\ \textbf{or rate} \\ \textbf{(events / hour)}}} = \underbrace{\mu_u \pi_\ell}_{\substack{\textbf{Own} \\ \textbf{initiative}}} + \underbrace{\sum_{j:t_j \in \mathcal{H}_{u,\ell}(t)} k_{\theta_\ell}(t - t_j)}_{\textbf{Follow-up}}$$

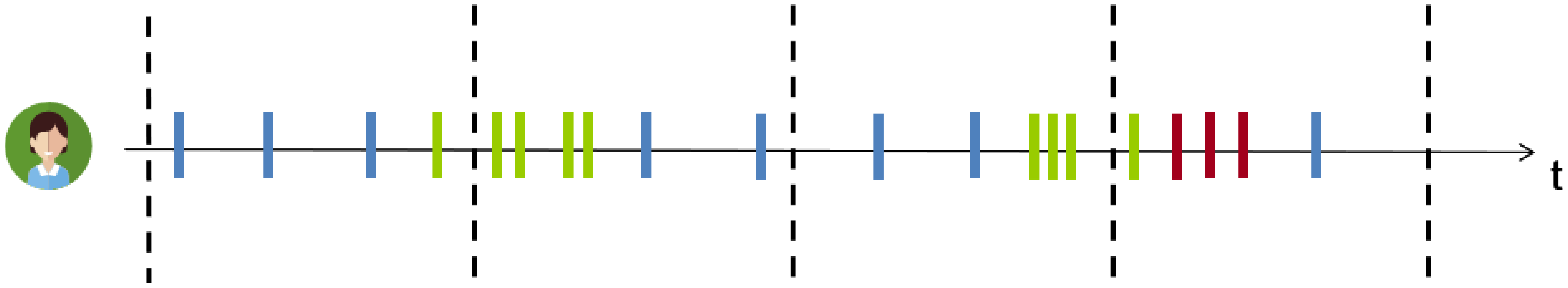
Hawkes process



[Mavroforakis et al., WWW 2017]

User events intensity

Users adopt more than one cluster:



A user's learning events as a multidimensional Hawkes:

Time cluster

$$(t_n, p_n) \sim Hawkes \begin{pmatrix} \lambda^*_{u,1}(t) \\ \vdots \\ \lambda^*_{u,\infty}(t) \end{pmatrix}$$

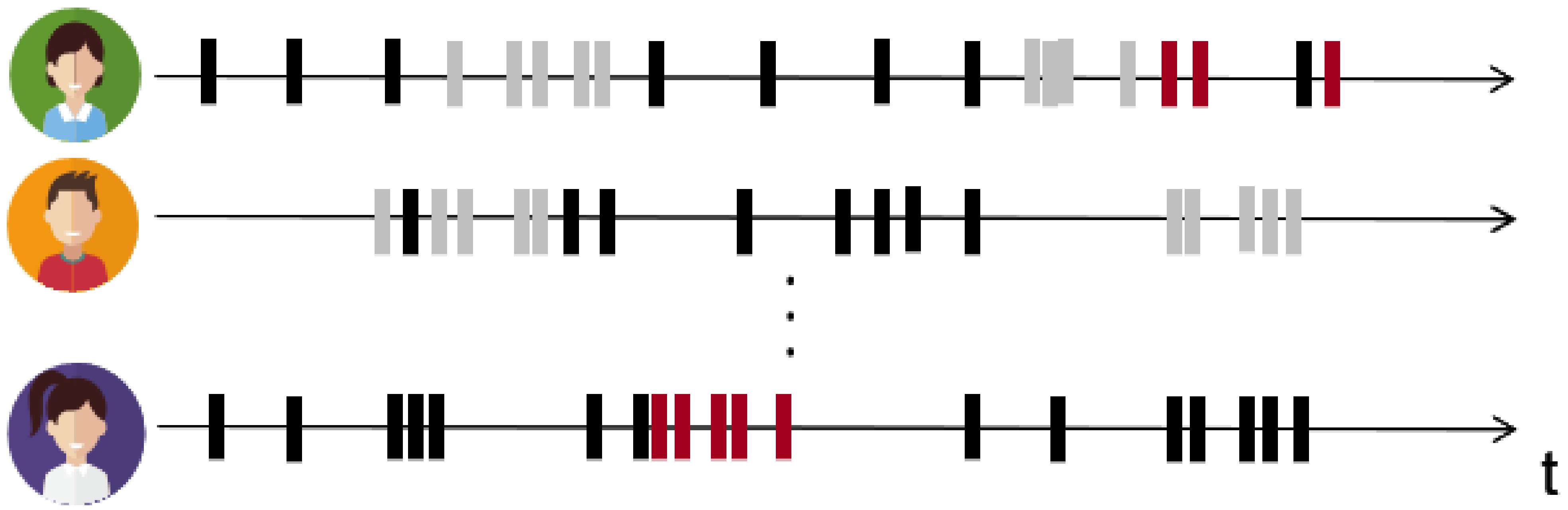
Content $\rightarrow$ $q_n = \boldsymbol{\omega}$ $\quad \omega_j \sim Multinomial(\boldsymbol{\theta}_p)$

Skoltech

[Mavroforakis et al., WWW 2017]

People share same clusters

Different users adopt same clusters



Cluster distribution from a Dirichlet process:

- Infinite # of clusters.
- Shared parameters across users.

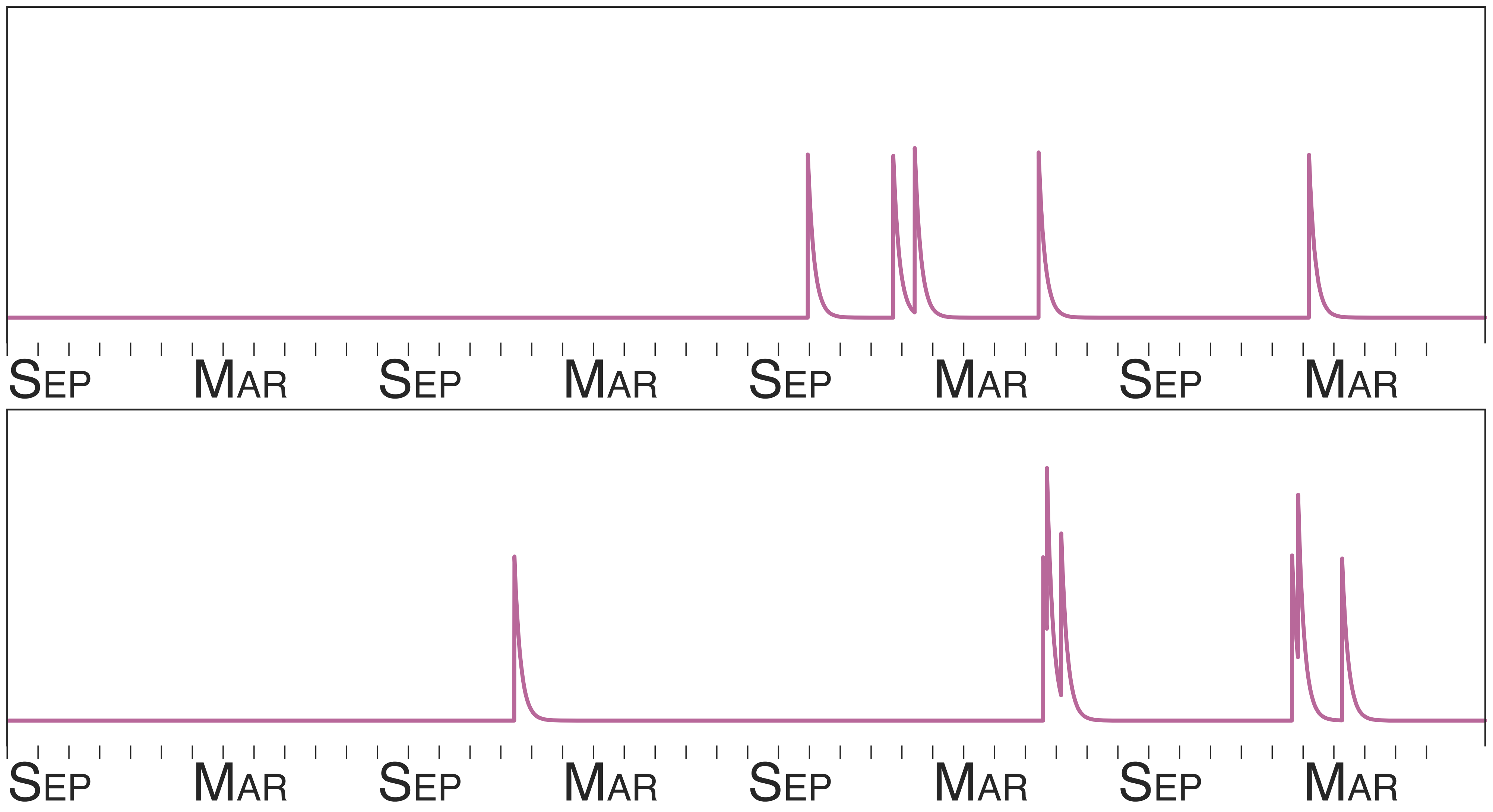
Efficient model inference using Sequential Monte-Carlo!

[Mavroforakis et al., WWW 2017]

Learning cluster (I): Version Control

Content

Intensities

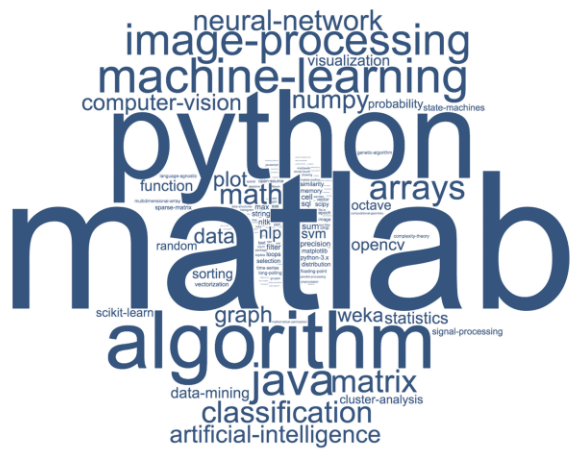


Version control tasks tend to be specific,
quickly solved after performing few questions

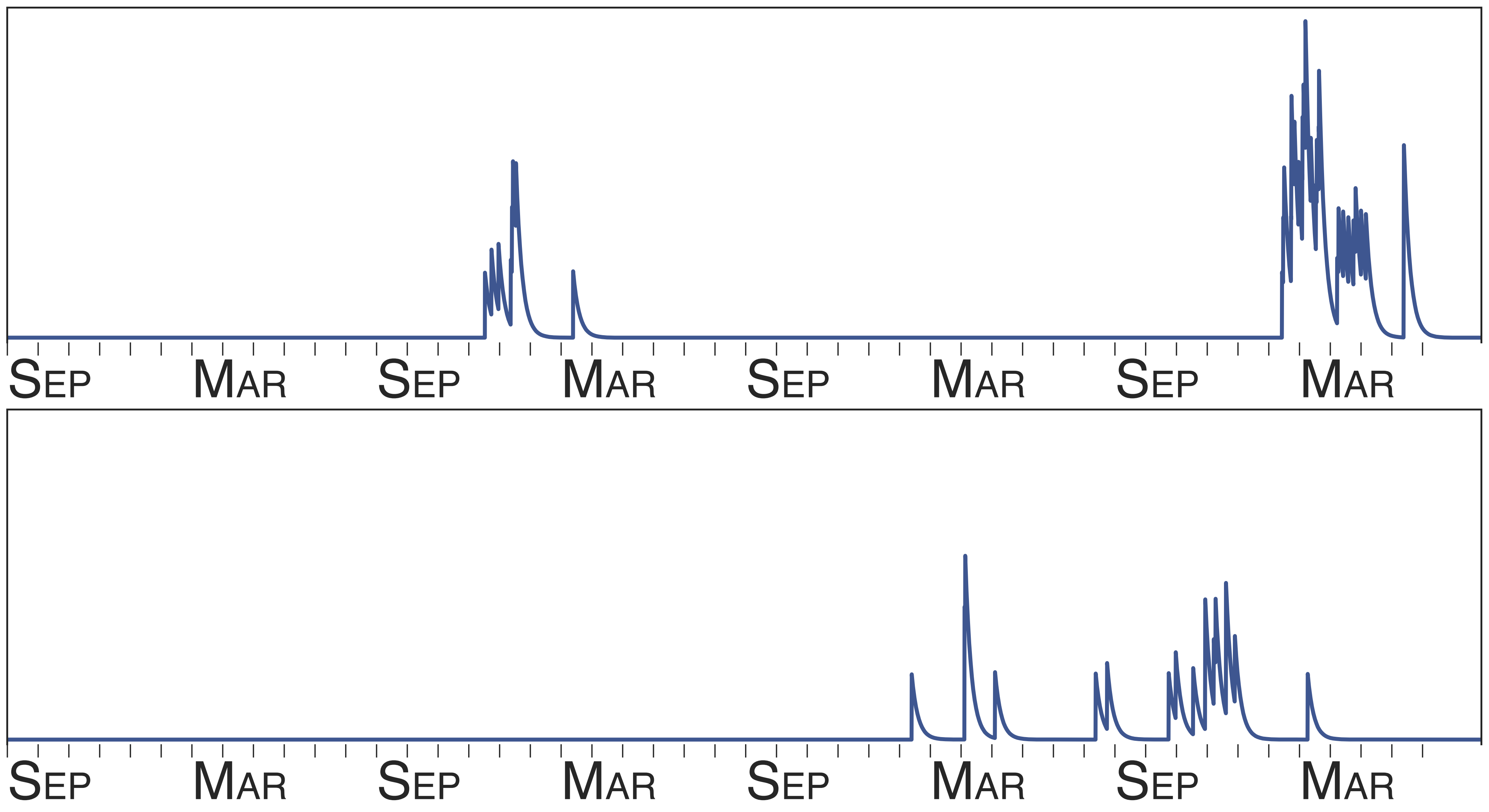
[Mavroforakis et al., WWW 2017]

Learning cluster (II): Machine learning

Content



Intensities

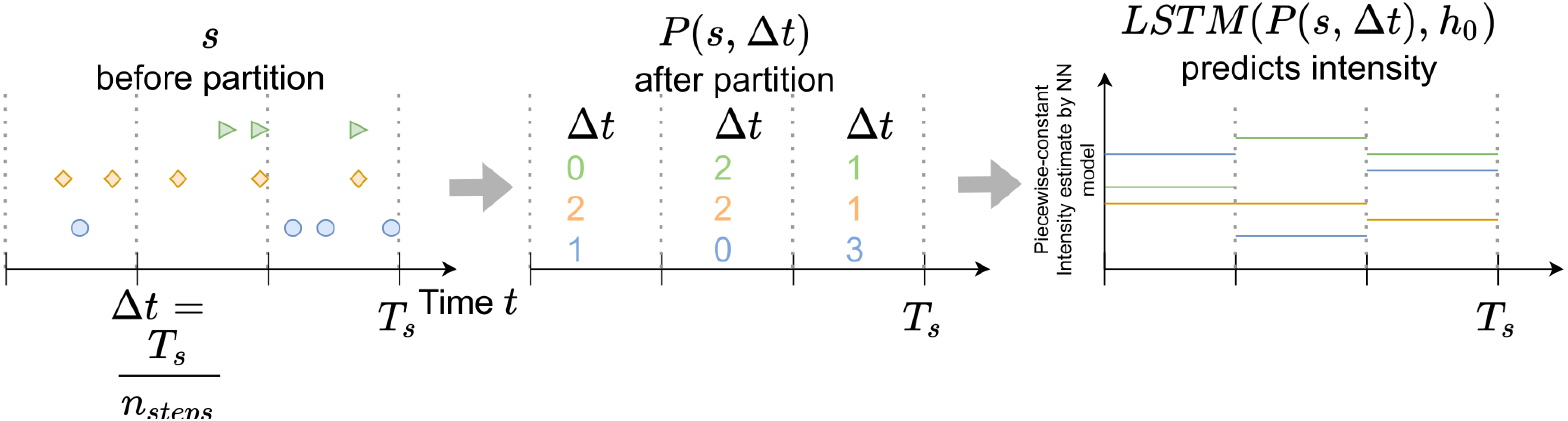


stackoverflow

Machine learning tasks tend to be more complex and require asking more questions

[Mavroforakis et al., WWW 2017]

COHORTNEY for events clustering

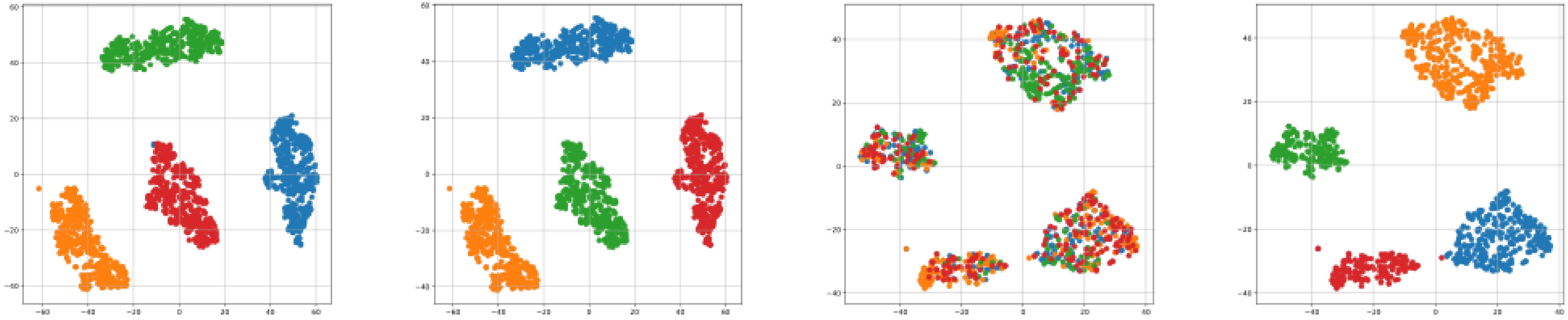


Analytical likelihood

EM algorithm for selection of parameters and labeling

[Zhuzhel et al., 2021]

COHORTNEY for events clustering



(a) (b) (c) (d)

True (a) and learned (b) clusters
for synthetic data

True (c) and learned (d) clusters
for real AGE data

[Zhuzhel et al., 2021]

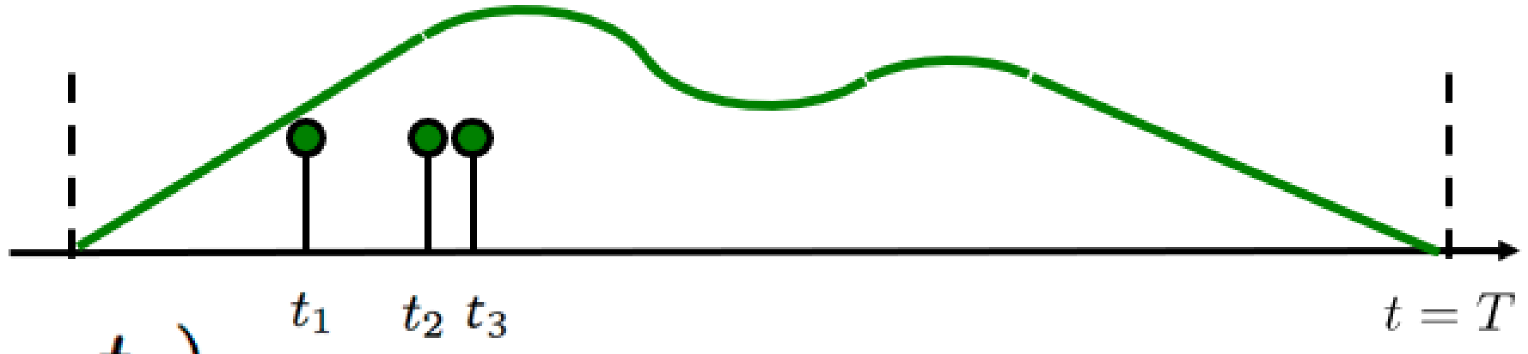
3. Capturing complex dynamics

Skoltech

Towards real-world temporal dynamics

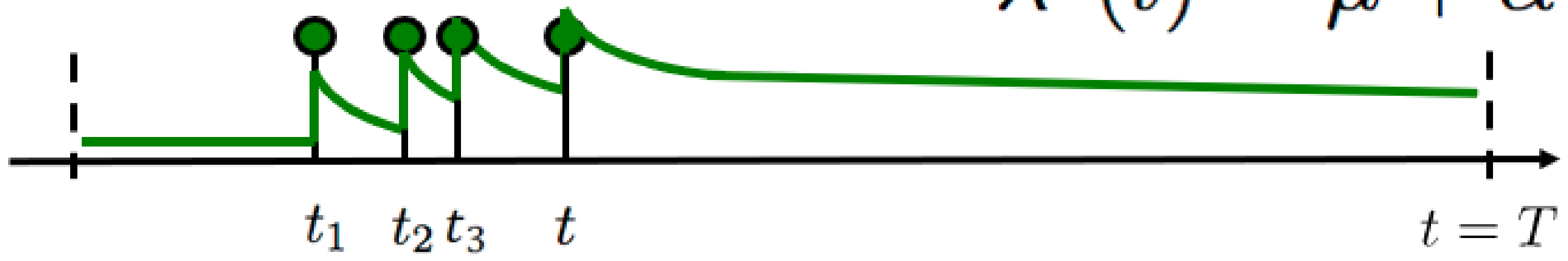
Up to now, we have focused on simple temporal dynamics (and intensity functions):

$$\lambda^*(t) = \mu$$



$$\lambda^*(t) = \sum_j \alpha_j \, k(t - t_j)$$

$$\lambda^*(t) = \mu + \alpha \sum_{t_i \in \mathcal{H}(t)} \kappa_\omega(t - t_i)$$



Recent works make use of RNNs to capture more complex dynamics

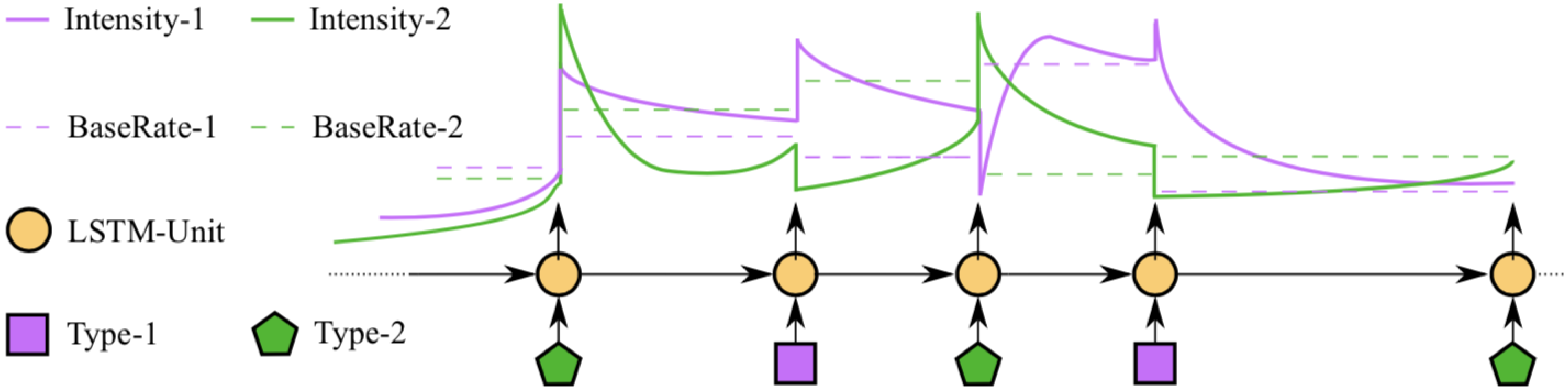
[Du et al., 2016; Dai et al., 2016; Mei & Eisner, 2017; Jing & Smola, 2017; Trivedi et al., 2017; Xiao et al., 2017a; 2018]

Neural Hawkes process

1) History effect does not need to be additive

2) Allows for complex memory effects

   such as delays



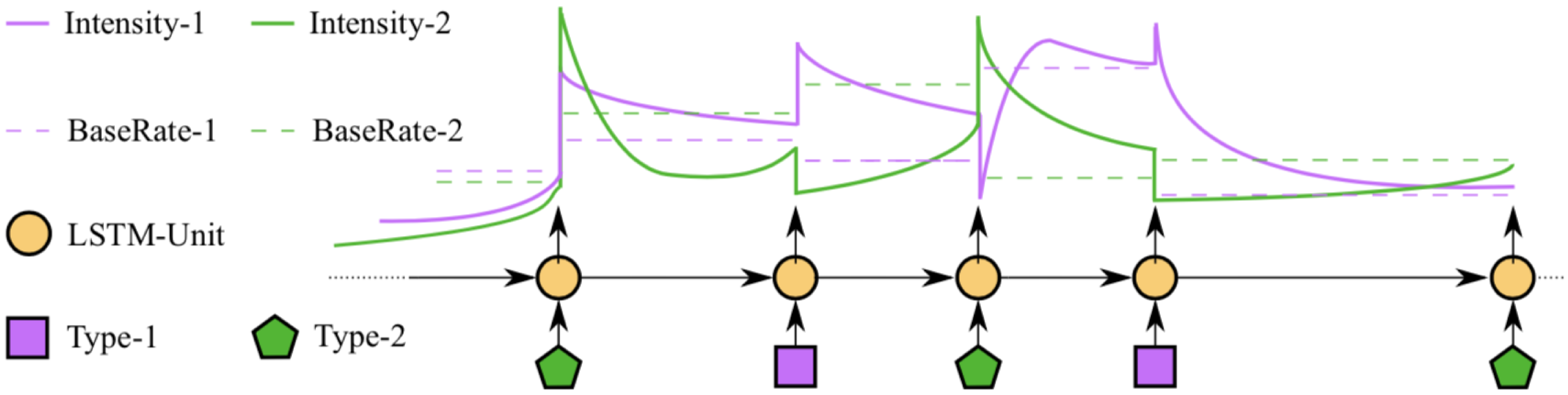
[Mei & Eisner, NIPS 2017]

Neural Hawkes process

$$\lambda_u(t) = f_u(\mathbf{w}_u^\top \mathbf{h}(t)) \qquad \mathbf{h}(t) = \overbrace{\mathrm{RNN}(\mathcal{H}(t))}^{\text{Memory}}$$

Excitation & inhibition

Parametric learning using stochastic gradient descent

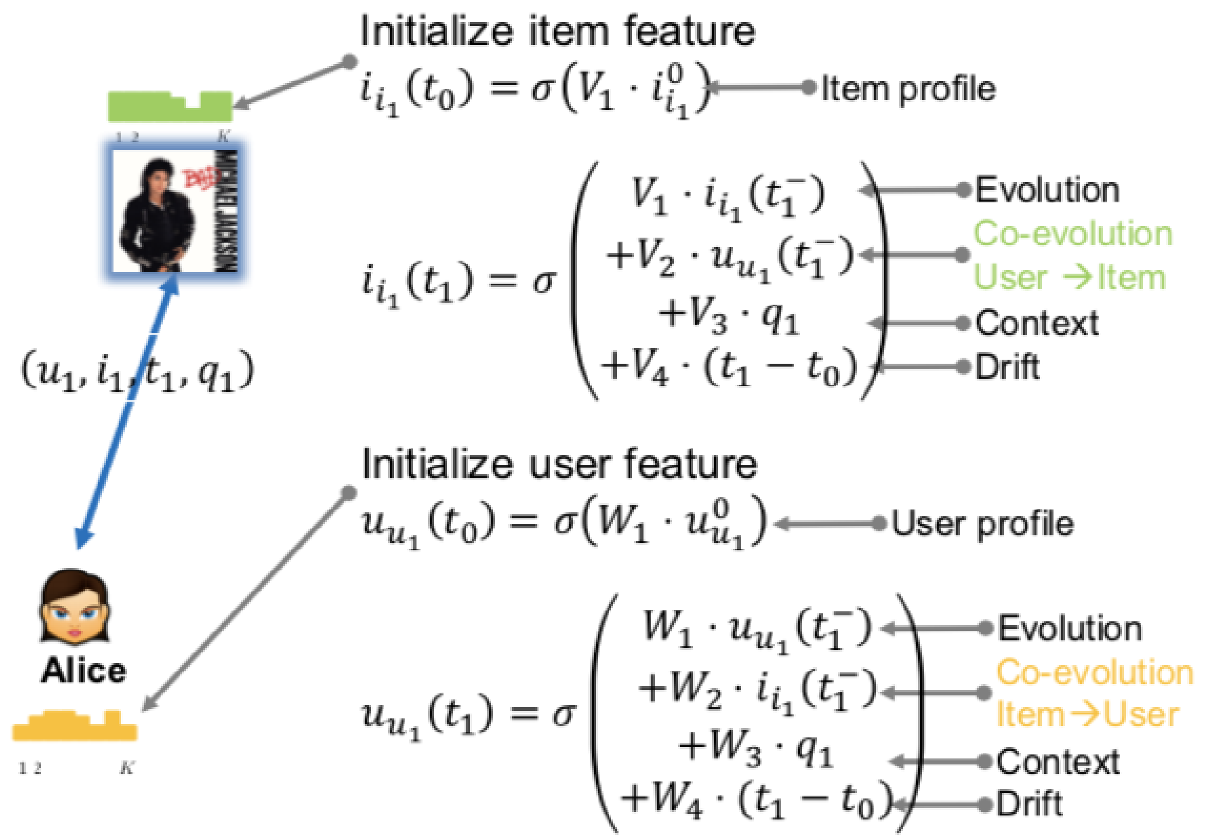


[Mei & Eisner, NIPS 2017]

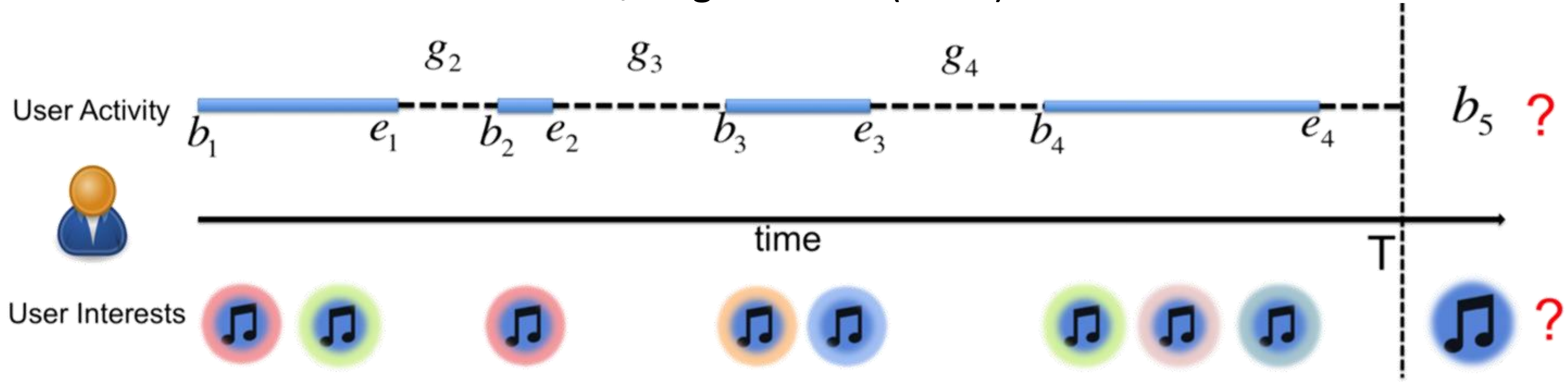
Applications (I): Predictive Models

Know-Evolve, Trivedi et al. (2017)

Coevolutionary Embedding, Dai et al. (2017)

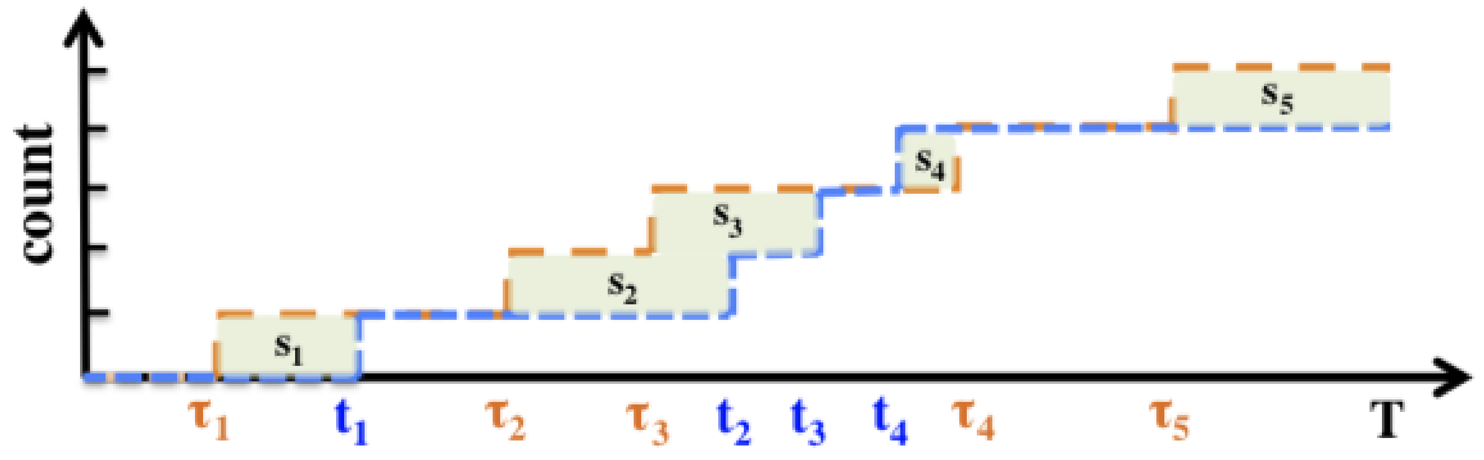


Neural Survival Recommender, Jing & Smola (2017)



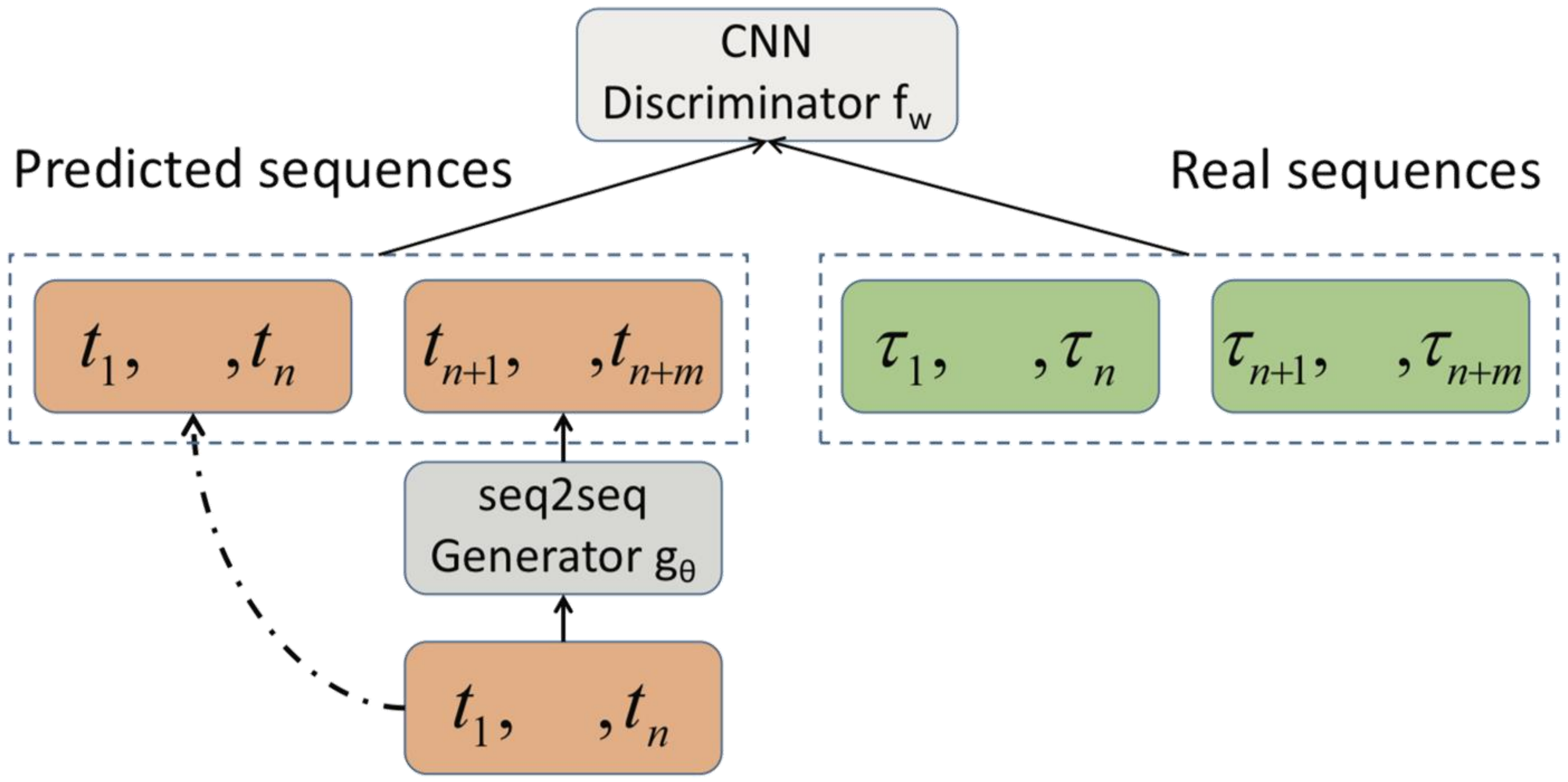
Applications (II): Generative Models

Key idea: Intensity- and likelihood-free models



Wasserstein-Distance for Temporal Point Processes

GAN architecture



Skoltech

[Xiao et al., 2017 & 2018]

4. Causal reasoning on event sequences

Skoltech

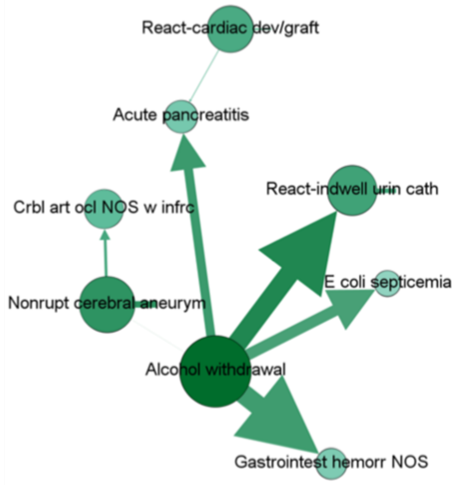
Temporal point processes beyond prediction

So far, we have focused on models that improve predictions:

Link prediction

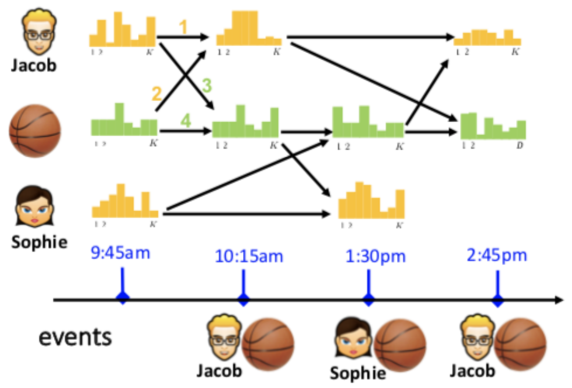
[Trivedi et al., 2017]

Community detection



[Xiao et al., 2017]

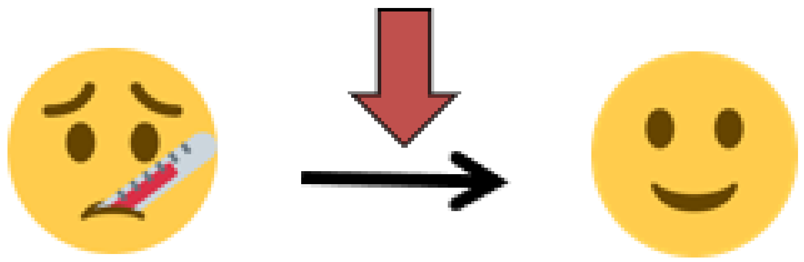
Recommendations



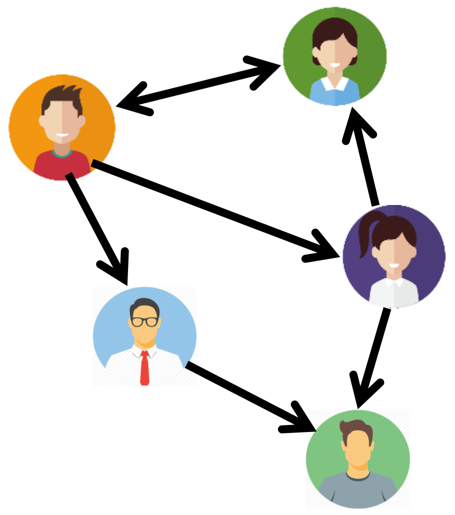
[Dai et al., 2017]

Recent works have focused on performing causal inference using event sequences:

Treatment effect



Granger causality graph



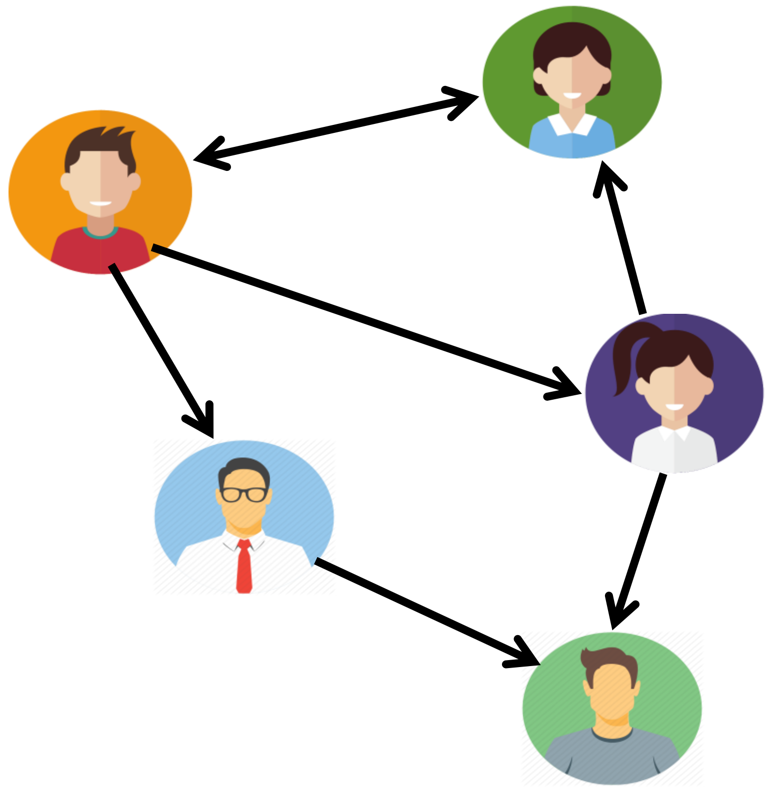
[Xu et al., 2016; Achab et al., 2017; Kuśmierczyk & Gomez-Rodriguez, 2018]

Uncovering Causality from Hawkes Processes

Multivariate Hawkes process:

$$N(t) = \sum_{u \in \mathcal{U}} N_u(t)$$

$$\lambda_u(t) = \mu_u + \sum_{v \in \mathcal{U}} \int_0^t \underbrace{k_{u,v}(t-t')}_{\text{Effect of v's past events on u}} dN_v(t')$$



Granger causality:

"X causes Y in the sense of Granger causality if forecasting future values of Y is more successful while taking X past values into account"

[Granger, 1969]

Skoltech

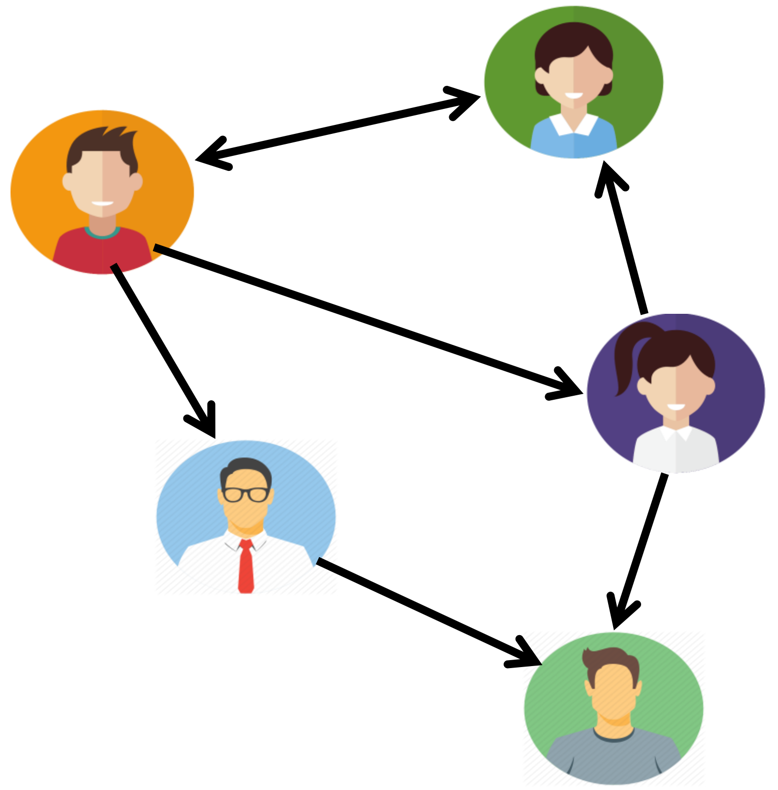
[Achab et al., ICML 2017]

Uncovering Causality from Hawkes Processes

Multivariate Hawkes process:

$$N(t) = \sum_{u \in \mathcal{U}} N_u(t)$$

$$\lambda_u(t) = \mu_u + \sum_{v \in \mathcal{U}} \int_0^t \underbrace{k_{u,v}(t-t')}_{\text{Effect of v's past events on u}} dN_v(t')$$



Granger causality on multivariate Hawkes processes:

“ $N_v(t)$ does not Ganger-cause $N_u(t)$ w.r.t. $N(t)$ if and only if $k_{u,v}(\tau) = 0$ for $\tau \in \Re^+$ "

[Eichler et al., 2016]

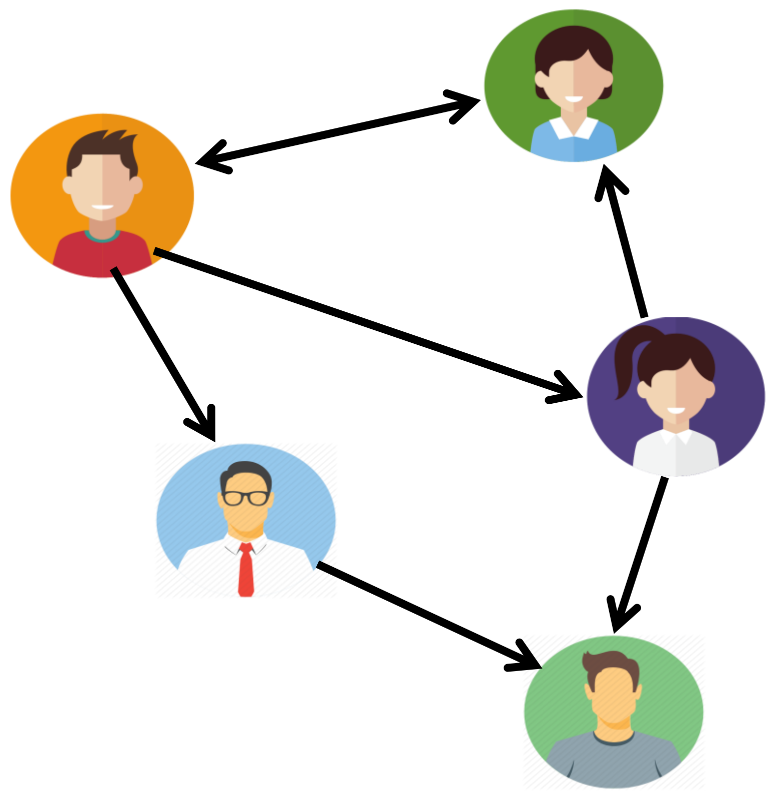
Skoltech

[Achab et al., ICML 2017]

Uncovering Causality from Hawkes Processes

Goal is to estimate $G = [g_{uv}]$, where:

$$g_{uv} = \int_0^{+\infty} k_{u,v}(\tau)d\tau \geq 0 \text{ for all } u, v \in \mathcal{U}$$



Average total # of events of node *u* whose *direct* ancestor is an event by node *v*

Then, $G = [g_{uv}]$ quantifies the *direct causal relationship* between nodes.

Skoltech

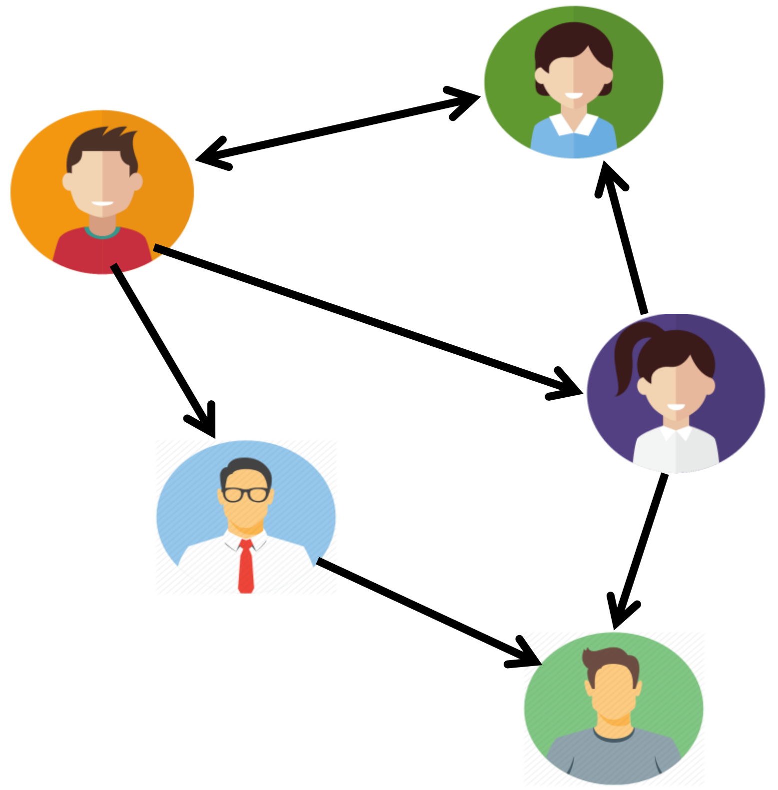
[Achab et al., ICML 2017]

Uncovering Causality from Hawkes Processes

Goal is to estimate $G = [g_{uv}]$, where:

$$g_{uv} = \int_0^{+\infty} k_{u,v}(\tau)d\tau \geq 0 \text{ for all } u, v \in \mathcal{U}$$

Average total # of events of node *u* whose *direct* ancestor is an event by node *v*



Then, $G = [g_{uv}]$ quantifies the *direct causal relationship* between nodes.

Key idea: Estimate G using the cumulants dN(t) of the Hawkes process.

Skoltech

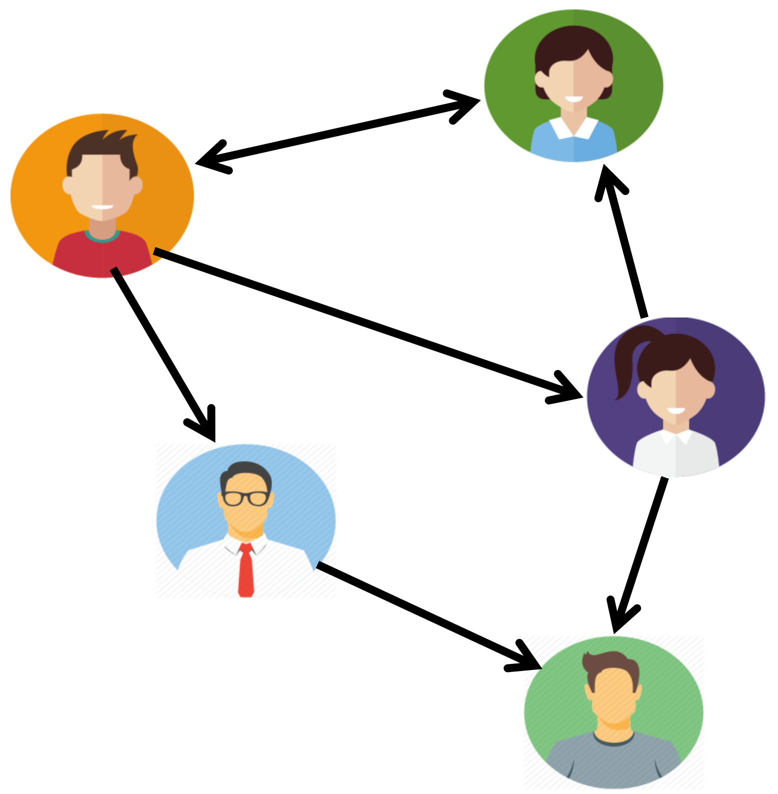
[Achab et al., ICML 2017]

Uncovering Causality from Hawkes Processes

Goal is to estimate $G = [g_{uv}]$, where:

$$g_{uv} = \int_0^{+\infty} k_{u,v}(\tau)d\tau \geq 0 \text{ for all } u, v \in \mathcal{U}$$

Average total # of events of node *u* whose *direct* ancestor is an event by node *v*



Then, $G = [g_{uv}]$ quantifies the *direct causal relationship* between nodes.

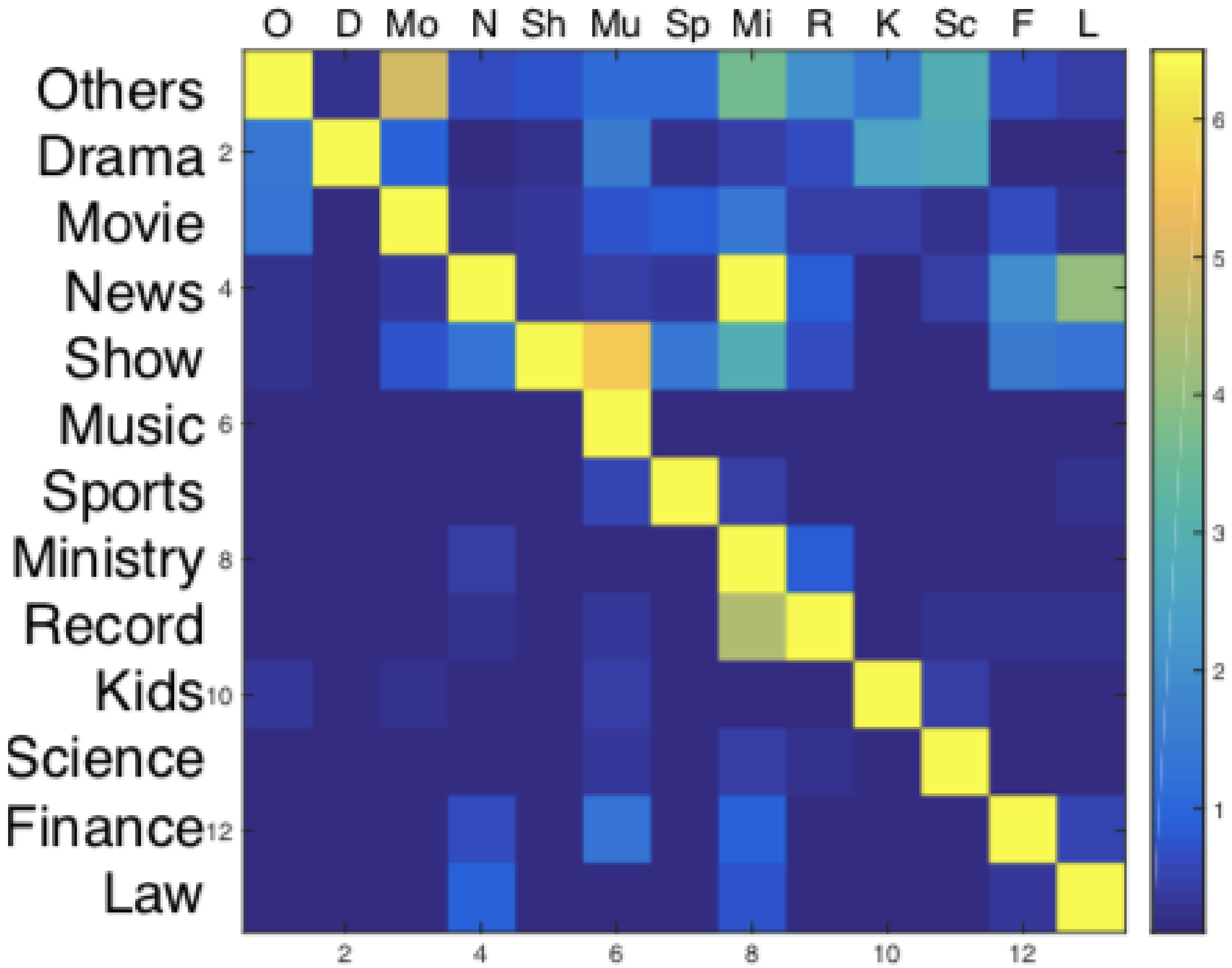
Non parametric Hawkes cumulant estimation method with TensorFlow implementation

Key idea: Estimate G using the cumulants dN(t) of the Hawkes process.

[Achab et al., ICML 2017]

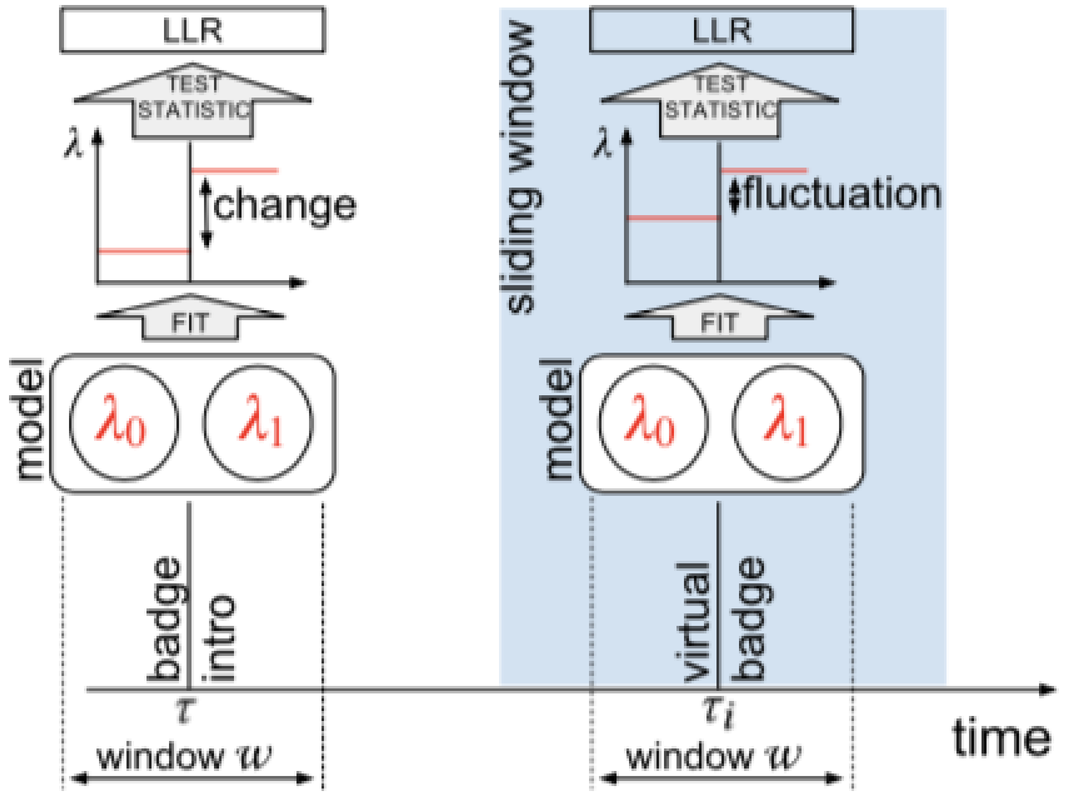
Causal reasoning: Applications

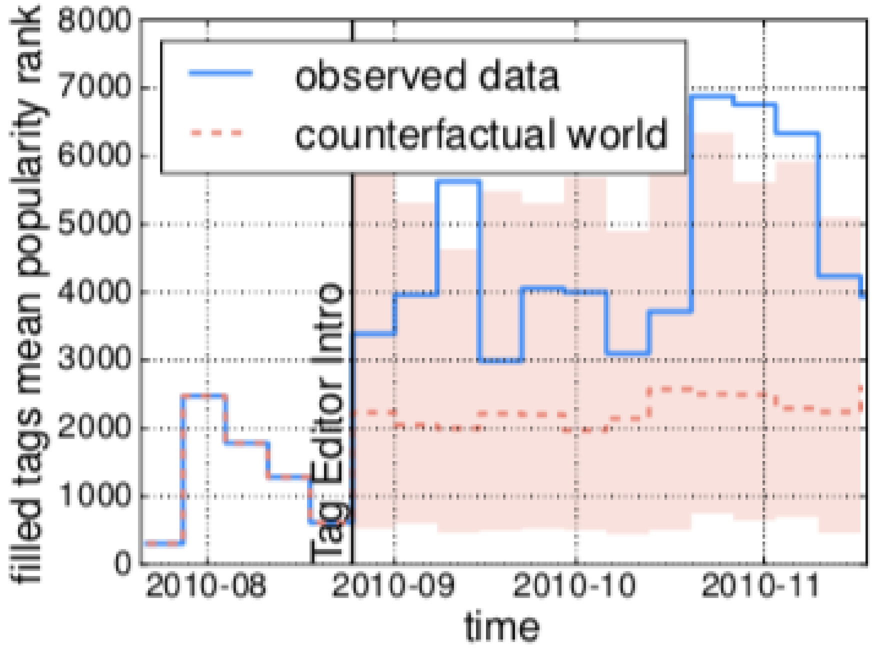
Infectivity matrix estimation



[Xu et al., 2016, ICML]

Effect of Badges





Tag wiki rank over time

[Kuśmierczyk & Gomez-Rodriguez, 2018]

Models and inference

1. Modeling event sequences

2. Clustering event sequences

3. Capturing complex dynamics

4. Causal reasoning on event sequences