Machine learning for Dynamic Social Network Analysis

Applications: Control

Manuel Gomez Rodriguez
Max Planck Institute for Software Systems

IIT Hyderabad, December 2017

Outline of the Seminar

REPRESENTATION: TEMPORAL POINT PROCESSES

1. Intensity function
2. Basic building blocks
3. Superposition
4. Marks and SDEs with jumps

APPLICATIONS: MODELS

1. Information propagation
2. Opinion dynamics
3. Information reliability
4. Knowledge acquisition

APPLICATIONS: CONTROL

1. Influence maximization
2. Activity shaping
3. When to post
4. When to fact check

This lecture

Applications: Control

1. Influence maximization
2. Activity shaping
3. When to post
4. When to fact check

Influence maximization

Can we find the **most influential** group of users?

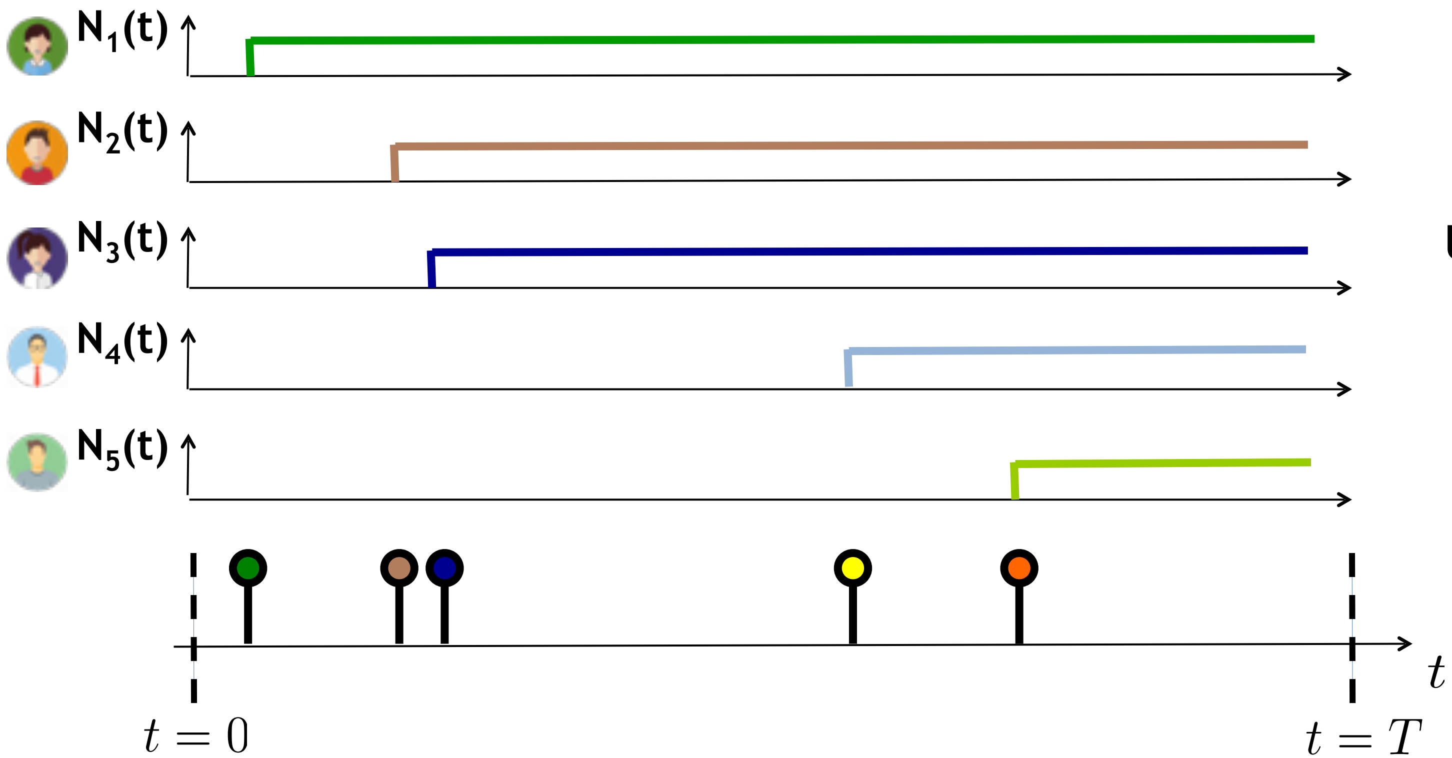
Why this goal?

Forbes

Why Word Of Mouth Marketing Is The Most Important Social Media

Idea adoption representation (Lecture 2)

We represent an idea adoptions using **terminating temporal point processes:**

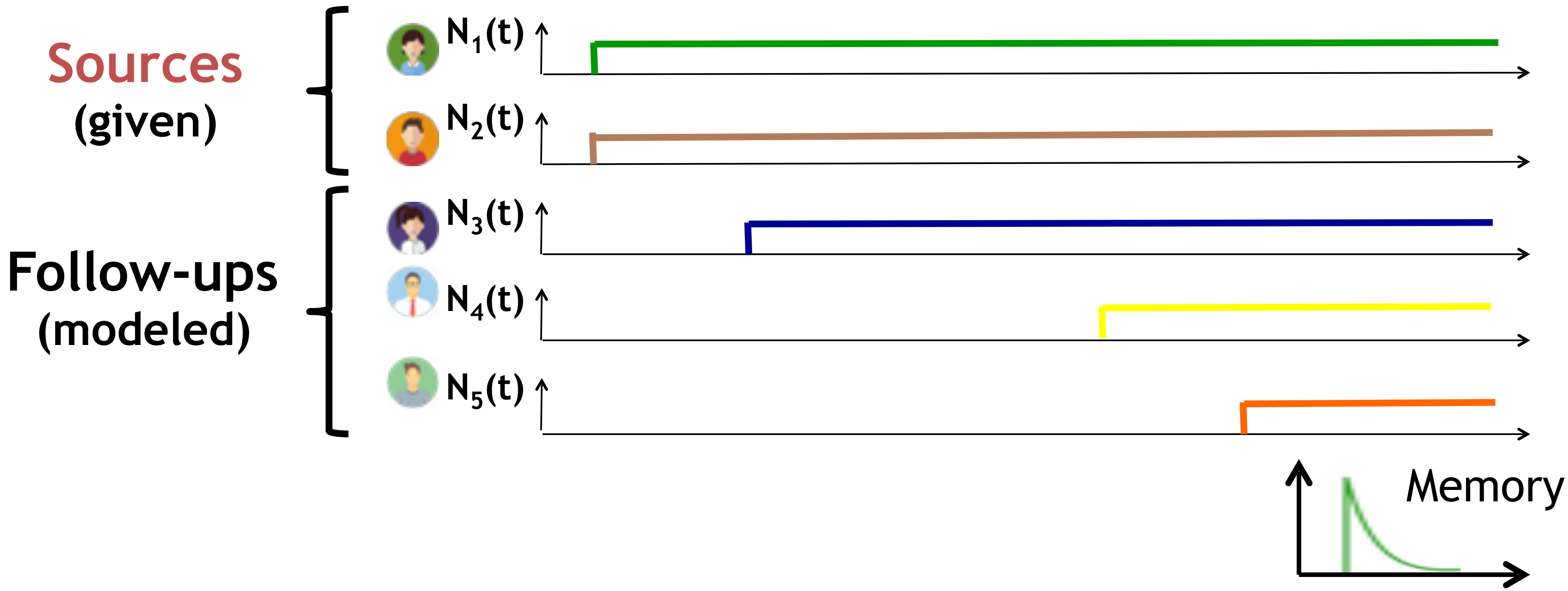


Idea adoption:

$$(u_i, m_i, t_i)$$

User Idea Time

Idea adoption intensity (Lecture 2)



$$\lambda_u^*(t) = \underbrace{(1-N(t))}_{\text{Adopt idea only once}} \sum_{v\in[m]} b_{vu} \underbrace{\sum_{e_i\in\mathcal{H}_v(t)} \kappa(t-t_i)}_{\text{Previous messages by user v}}$$

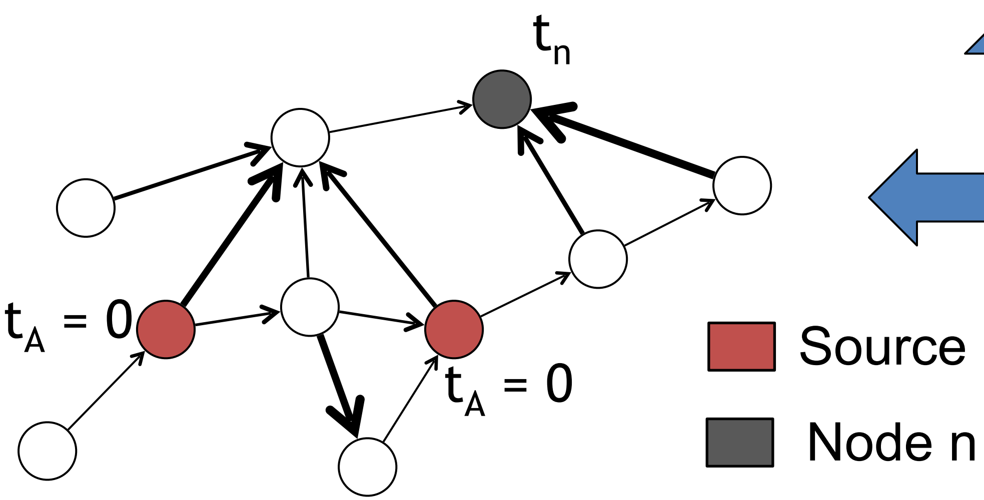
Influence from user v on user u

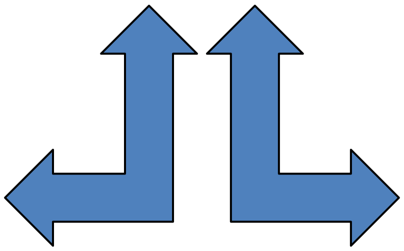
[Gomez-Rodriguez et al., ICML 2011]

Influence of a set of sources

Influence of a set of source nodes A:

$$\sigma(A;T) = \underbrace{\mathbb{E}N(A;T)}_{\text{Average number of nodes who follow-up by time T}} = \sum_{n=1}^{N} \underbrace{P(t_n \leq T|A)}_{\text{Probability that a node n follows up}}$$





It depends on the nodes intensity

$$\lambda_u^*(t)$$

(previous slide)

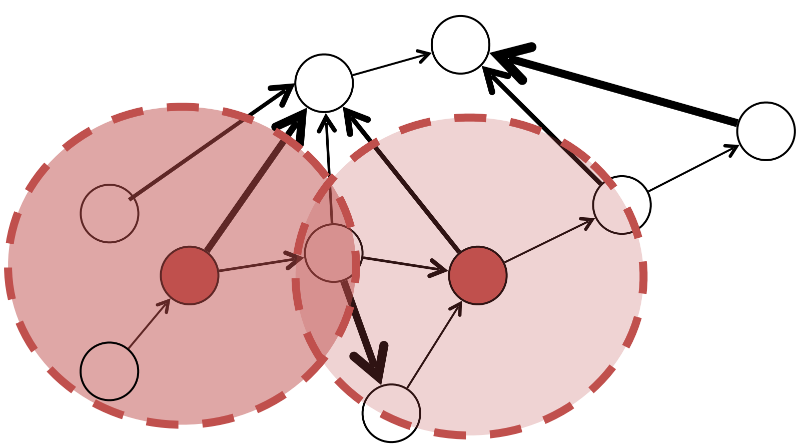
Influence estimation: exact vs. approx.

$$\sigma(A;T) = \mathbb{E}N(A;T) = \sum_{n=1}^{N} P(t_n \leq T | A)$$

Approximate Influence Estimation

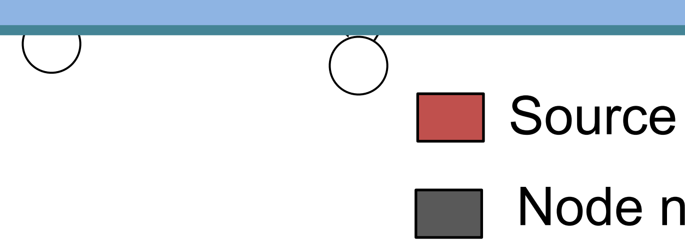
Exact Follow-up

Can be exponential in network size, **not scalable!**

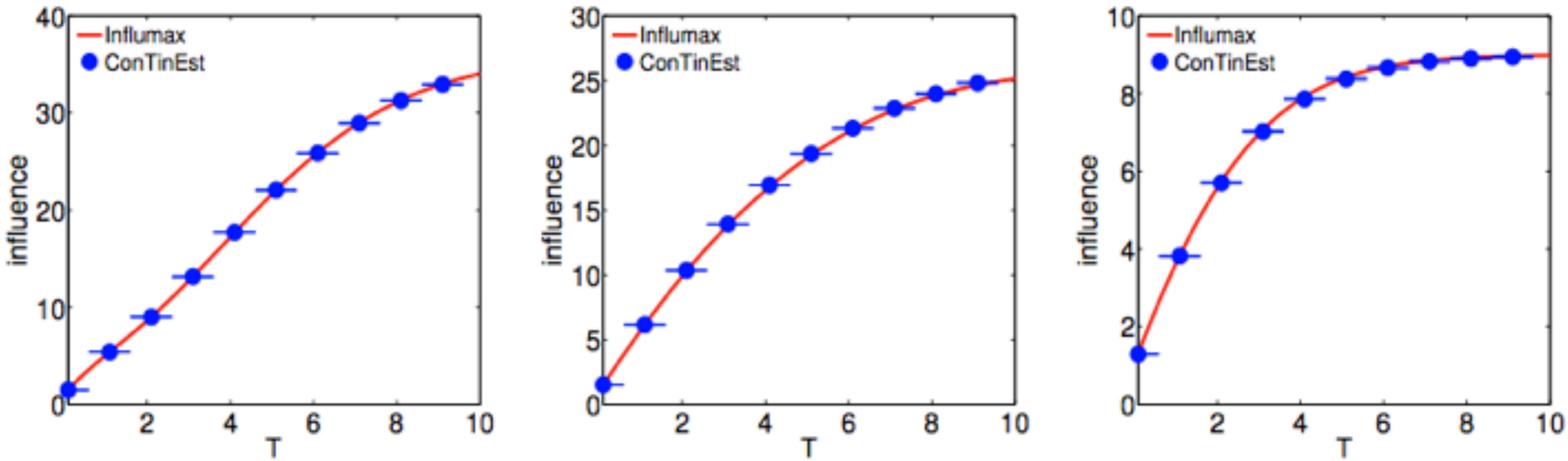


Key idea

Neighborhood



How good are approximate methods?



(a) Core-periphery (b) Random (c) Hierarchical

Not only **theoretical guarantees**, but they also **work well in practice.**

[Du et al., NIPS 2013

Maximizing the influence

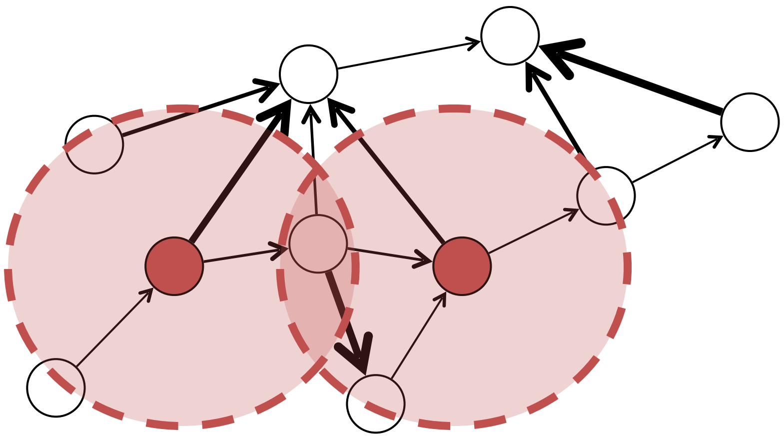
Once we have defined influence, what about finding **the set of source nodes that maximizes influence**?

$$A^* = \underset{|A| \leq k}{\operatorname{argmax}}\, \sigma(A; T)$$

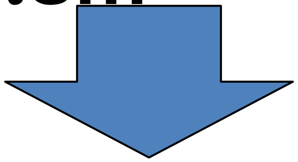
Theorem. For a wide variety of influence models, the influence maximization problem is NP-hard.

[Du et al., NIPS 2013

NP-hardness of influence maximization



The influence maximization can be reduced to a **Set Cover problem**

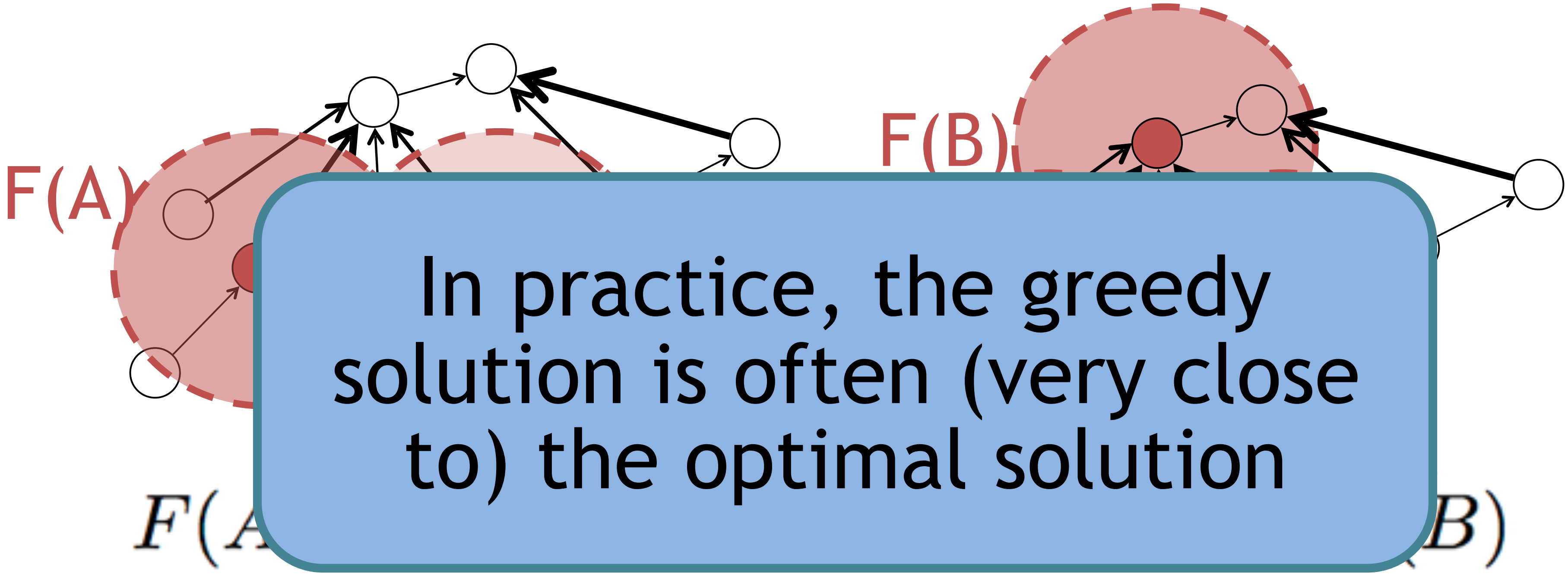


Set Cover is a **well-known NP-hard problem**

[Du et al., NIPS 2013

Submodularity of influence maximization

The influence functio $\sigma(A;T)$ satisfies a **natural diminishing property: submodularity!**



In practice, the greedy solution is often (very close to) the optimal solution

Consequence: Greedy algorithm with **63% provable guarantee**

[Du et al., NIPS 2013

Applications: Control

1. Influence maximization
2. Activity shaping
3. When to post
4. When to fact check

Activity shaping

Can we **steer users' activity** in a **social network in general?**

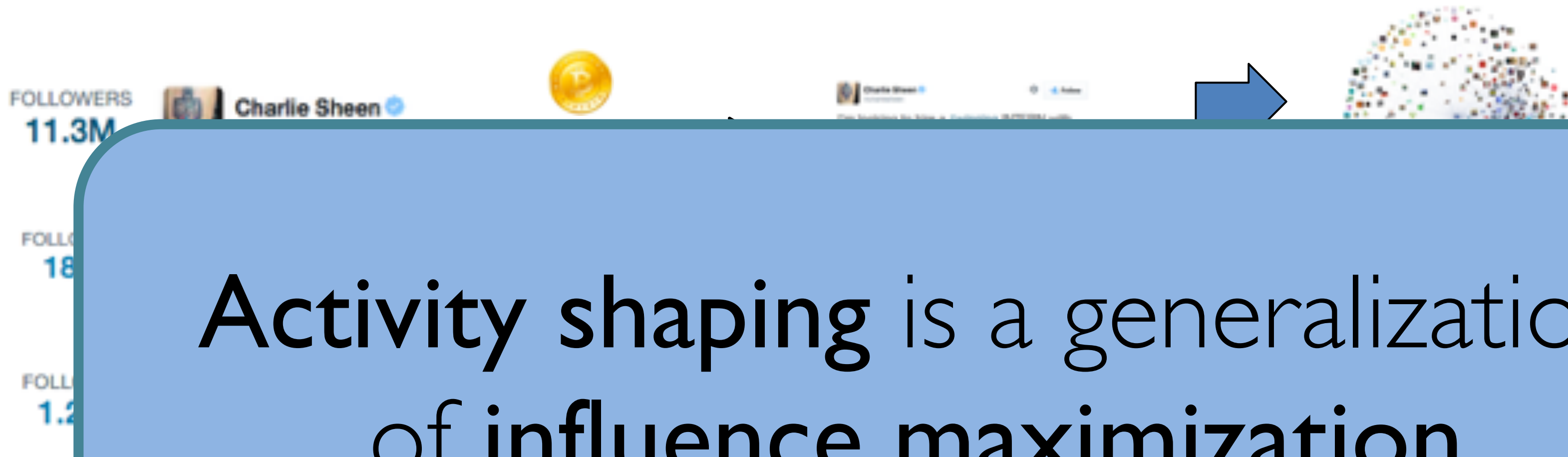
Why this goal?

TIME **Twitter Stock Tumbles After Drop in User Engagement**

HUFF POST **7 Ways to Increase Your Social Media Engagement**

Activity shaping vs influence maximization

Related to
Influence Maximization Problem

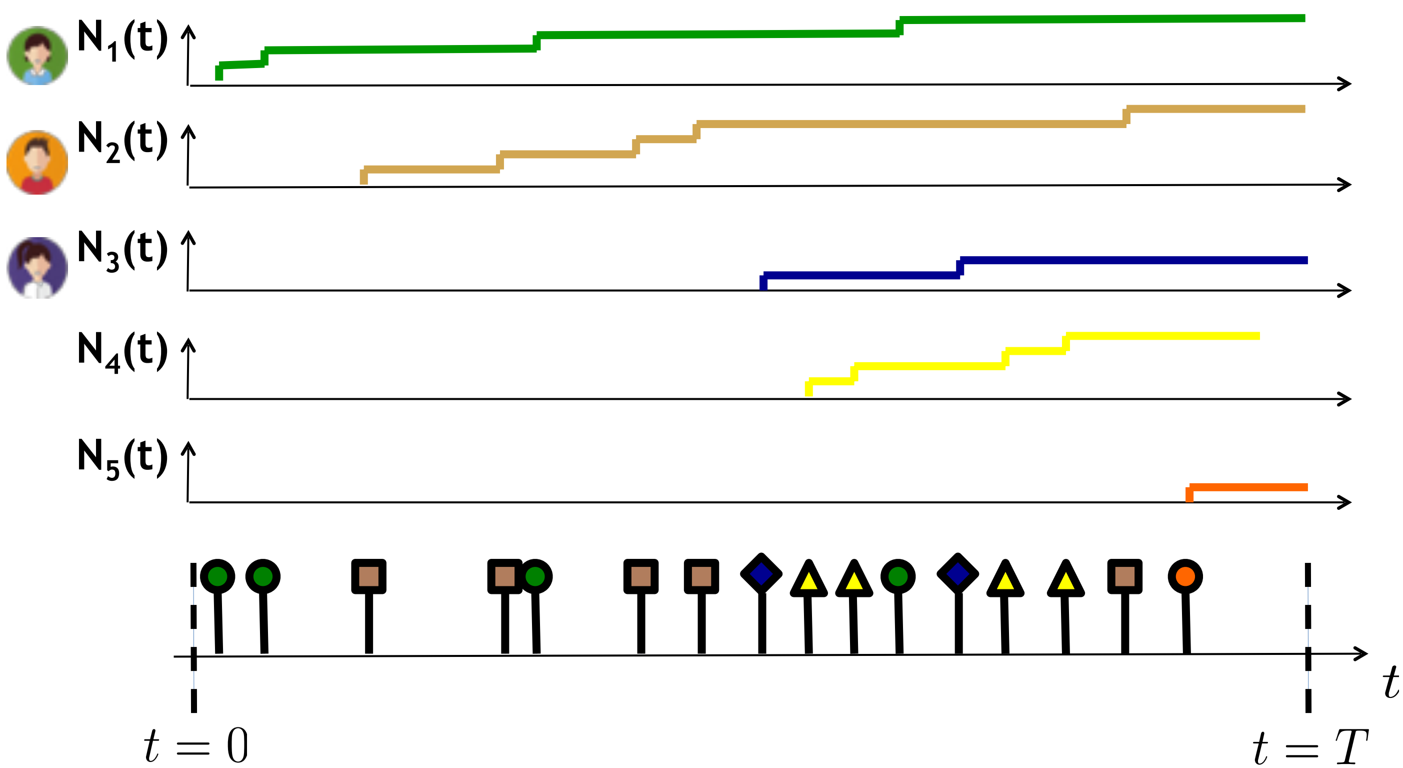


Activity shaping is a generalization of **influence maximization**

Influence Maximization	incentive	same piece of information	maximizing adoption
Activity Shaping	Variable incentive	Multiple times multiple pieces, recurrent!	Many different activity shaping tasks

Event representation (Lecture 2)

We represent messages using
nonterminating temporal point processes:



Recurrent event

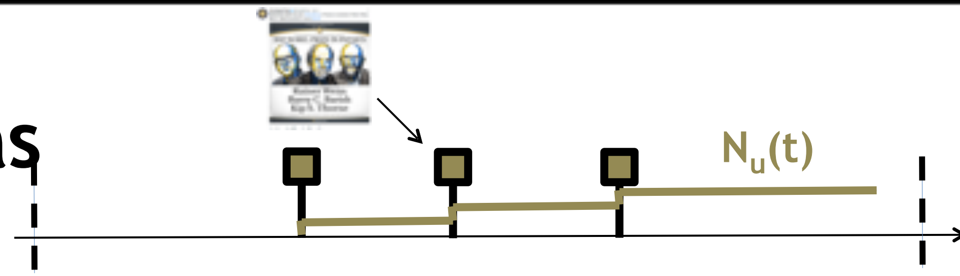
(u_i, t_i)

User **Time**

[Zarezade et al., Allerton 2017

Multidimensional Hawkes process

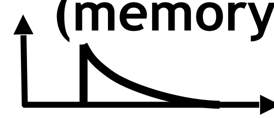
For each user u, actions as a counting process $N_u(t)$



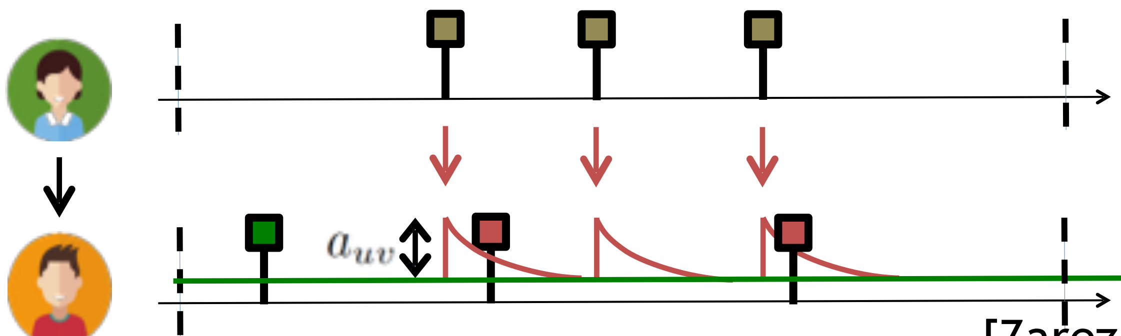
Intensities or rates
(Actions per time unit)

User influence matrix

Non-negative kernel (memory)

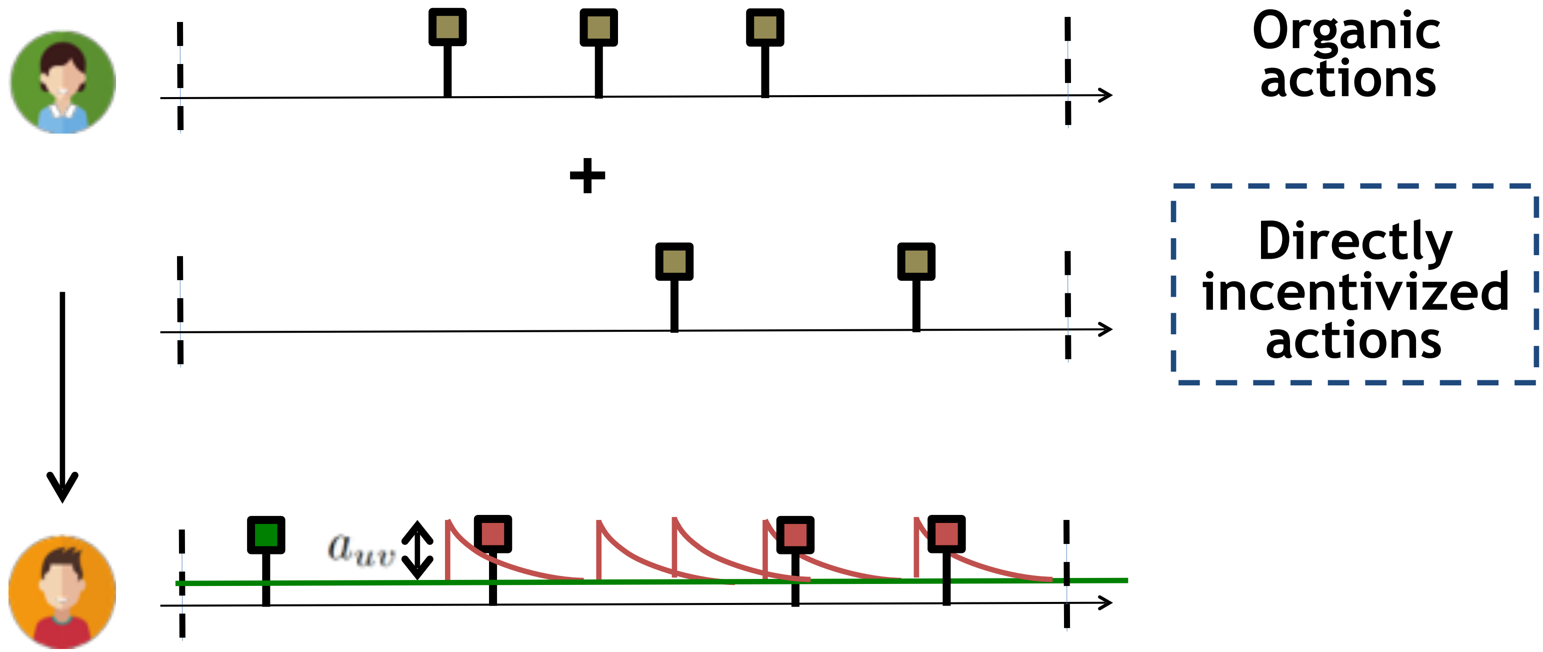


$$\boldsymbol{\lambda}^*(t) = \underbrace{\boldsymbol{\mu}_0}_{\text{Exogenous actions}} + \underbrace{\boldsymbol{A}\int_0^t \kappa(t-s)\, d\boldsymbol{N}(s)}_{\text{Endogenous actions}}$$



[Zarezade et al., Allerton 2017]

Steering endogenous actions



$$\boldsymbol{\lambda}^*(t) = \boldsymbol{\mu}_0 + \boldsymbol{A}\int_0^t \kappa(t-s)\, d\boldsymbol{N}(s) + \boldsymbol{A}\int_0^t \kappa(t-s)\, d\boldsymbol{M}(s)$$

Directly incentivized actions

Intensities of directly incentivized actions $\mathbb{E}[d\boldsymbol{M}(t)|\mathcal{H}(t)] = \boldsymbol{u}(t)\, dt$

[Zarezade et al., Allerton 2017

Cost to go & Bellman's principle of optimality

Optimization problem

$$\underset{\boldsymbol{u}(t_0,t_f]}{\text{minimize}} \quad \mathbb{E}_{(\boldsymbol{N},\boldsymbol{M})(t_0,t_f]}\left[\phi(\boldsymbol{\lambda}(t_f)) + \int_{t_0}^{t_f} \overbrace{\ell(\boldsymbol{\lambda}(t),\boldsymbol{u}(t))}^{\text{Loss}}\, dt\right]$$

$$\text{subject to} \quad u_i(t) \geq 0, \ \forall t \in (t_0, t_f], \ i = 1, \ldots, n$$

Dynamics defined by Jump SDEs

$$d\boldsymbol{\lambda}(t) = [w\boldsymbol{\mu}_0 - w\boldsymbol{\lambda}(t)]\,dt + \boldsymbol{A}\,d\boldsymbol{N}(t) + \boldsymbol{A}\,d\boldsymbol{M}(t)$$

To solve the problem, we first define the corresponding **optimal cost-to-go:**

$$J(\boldsymbol{\lambda}(t),t) = \min_{\boldsymbol{u}(t,t_f]} \mathbb{E}_{(\boldsymbol{N},\boldsymbol{M})(t,t_f]}\left[\phi(\boldsymbol{\lambda}(t_f)) + \int_{t}^{t_f} \ell(\boldsymbol{\lambda}(s),\boldsymbol{u}(s))\, ds\right]$$

The cost-to-go, evaluated at t_0, recovers the optimization problem!

[Zarezade et al., Allerton 2017]

Hamilton-Jacobi-Bellman (HJB) equation

Lemma. The **optimal cost-to-go** satisfies **Bellman's Principle of Optimality**

$$J(\boldsymbol{\lambda}(t), t) = \min_{\boldsymbol{u}(t,t+dt]} \{\mathbb{E}_{(\boldsymbol{N},\boldsymbol{M})(t,t+dt]}\left[J(\boldsymbol{\lambda}(t+dt), t+dt)\right] + \ell(\boldsymbol{\lambda}(t), \boldsymbol{u}(t))\, dt\}$$

$$dJ(\boldsymbol{\lambda}(t), t) = J(\boldsymbol{\lambda}(t+dt), t+dt) - J(\boldsymbol{\lambda}(t), t)$$

$$0 = \min_{\boldsymbol{u}(t,t+dt]} \{\mathbb{E}_{(\boldsymbol{N},\boldsymbol{M})(t,t+dt]}\left[dJ(\boldsymbol{\lambda}(t), t)\right] + \ell(\boldsymbol{\lambda}(t), \boldsymbol{u}(t))\, dt\}$$

$$d\boldsymbol{\lambda}(t) = [w\boldsymbol{\mu}_0 - w\boldsymbol{\lambda}(t)]\, dt + \boldsymbol{A}\, d\boldsymbol{N}(t) + \boldsymbol{A}\, d\boldsymbol{M}(t)$$

Hamilton-Jacobi-Bellman (HJB) equation

Partial differential equation in J (with respect to λ and t)

[Zarezade et al., Allerton 2017

Solving the HJB equation

Consider a quadratic loss

$$\ell(\boldsymbol{\lambda}(t), \boldsymbol{u}(t)) = \underbrace{-\frac{1}{2}\boldsymbol{\lambda}^T(t)\,\boldsymbol{Q}\,\boldsymbol{\lambda}(t)}_{\text{Rewards organic actions}} + \underbrace{\frac{1}{2}\boldsymbol{u}^T(t)\,\boldsymbol{S}\,\boldsymbol{u}(t)}_{\text{Penalizes directly incentivizes actions}}$$

We propose $J(\boldsymbol{\lambda}(t), t)$ and then show that the optimal intensity is:

$$\boldsymbol{u}^*(t) = -\boldsymbol{S}^{-1}\left[\boldsymbol{A}^T\boldsymbol{g}(t) + \boldsymbol{A}^T\boldsymbol{H}(t)\boldsymbol{\lambda}(t) + \frac{1}{2}\,\mathrm{diag}(\boldsymbol{A}^T\boldsymbol{H}(t)\boldsymbol{A})\right]$$

Computed offline once!

Closed form solution to a first order ODE

Solution to a matrix Riccati differential equation

[Zarezade et al., Allerton 2017

The Cheshire algorithm

Intuition

Steering actions means sampling action user & times from u*(t)

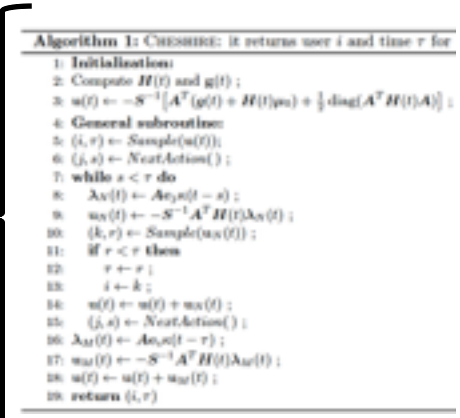
More in detail

Since the intensity function u*(t) is stochastic, we sample from it using:

- Superposition principle
- Standard thinning

It only requires $1^T N(t_f)$ ng from inhomog. Poisson!

Easy to implement


Algorithm 1: CHESHIRE: it returns user i and time τ for

```
Initialization:
Compute H(t) and g(t) ;
u(t) ← −S^{-1}[A^T(g(t) + H(t)μ_0) + ½ diag(A^T H(t) A)] ;
General subroutine:
(i, τ) ← Sample(u(t));
(j, s) ← NextAction() ;
while s < τ do
    λ_N(t) ← A e_j κ(t − s) ;
    u_N(t) ← −S^{-1} A^T H(t) λ_N(t) ;
    (k, r) ← Sample(u_N(t)) ;
    if r < τ then
        τ ← r ;
        i ← k ;
    u(t) ← u(t) + u_N(t) ;
    (j, s) ← NextAction() ;
λ_M(t) ← A e_i κ(t − τ) ;
u_M(t) ← −S^{-1} A^T H(t) λ_M(t) ;
u(t) ← u(t) + u_M(t) ;
return (i, τ)
```


[Zarezade et al., Allerton 2017

Experiments on real data

Experiments on five Twitter datasets (users) where actions are tweets and retweets

1. Fit model parameters

$$d\boldsymbol{\lambda}(t) = [w\boldsymbol{\mu}_0 - w\boldsymbol{\lambda}(t)]\,dt + \boldsymbol{A}\,d\boldsymbol{N}(t)$$

exogeneous rate **influence matrix**

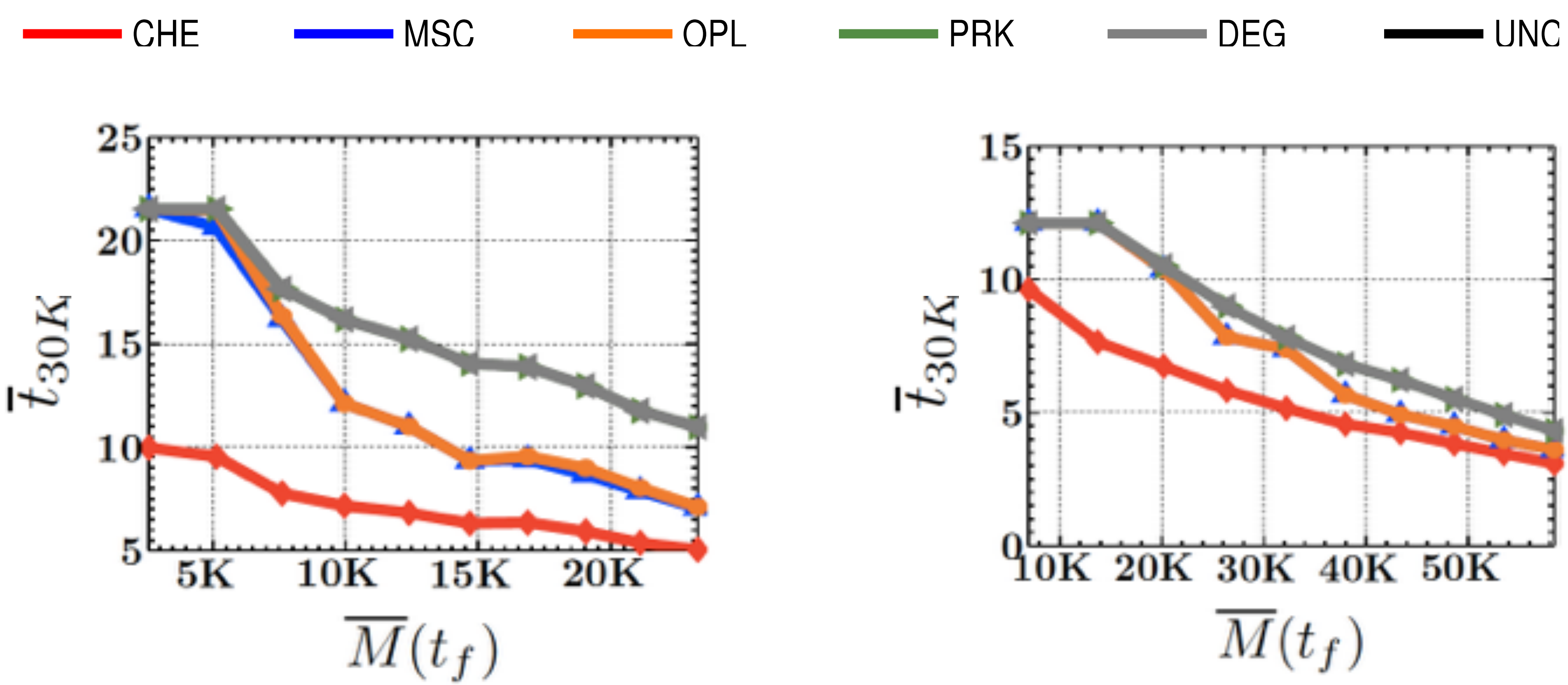
2. Simulate steering endogenous actions

$$d\boldsymbol{\lambda}(t) = [w\boldsymbol{\mu}_0 - w\boldsymbol{\lambda}(t)]\,dt + \boldsymbol{A}\,d\boldsymbol{N}(t) + \boldsymbol{A}\,d\boldsymbol{M}(t)$$

directly incentivized tweets chosen by each method

[Zarezade et al., Allerton 2017]

Performance vs. # of incentivized tweets



Sports, $M(t_f) \approx$ 5k

Series, $M(t_f) \approx$ 5k

Cheshire (in red) reaches 30K tweets 20-50% faster than the second best performer

[Zarezade et al., Allerton 2017

Applications: Control

1. Influence maximization
2. Activity shaping
3. When to post
4. When to fact check

Social media as a broadcasting platform

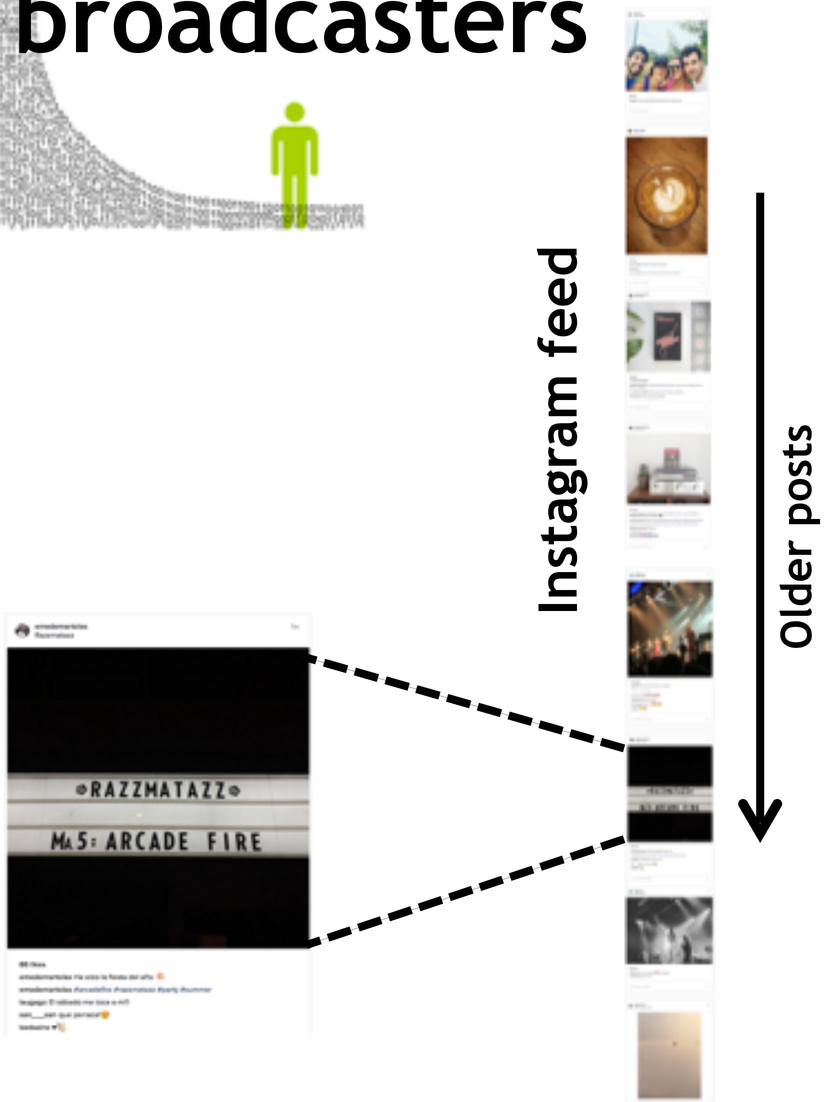
Everybody can build, reach and broadcast information to their own audience

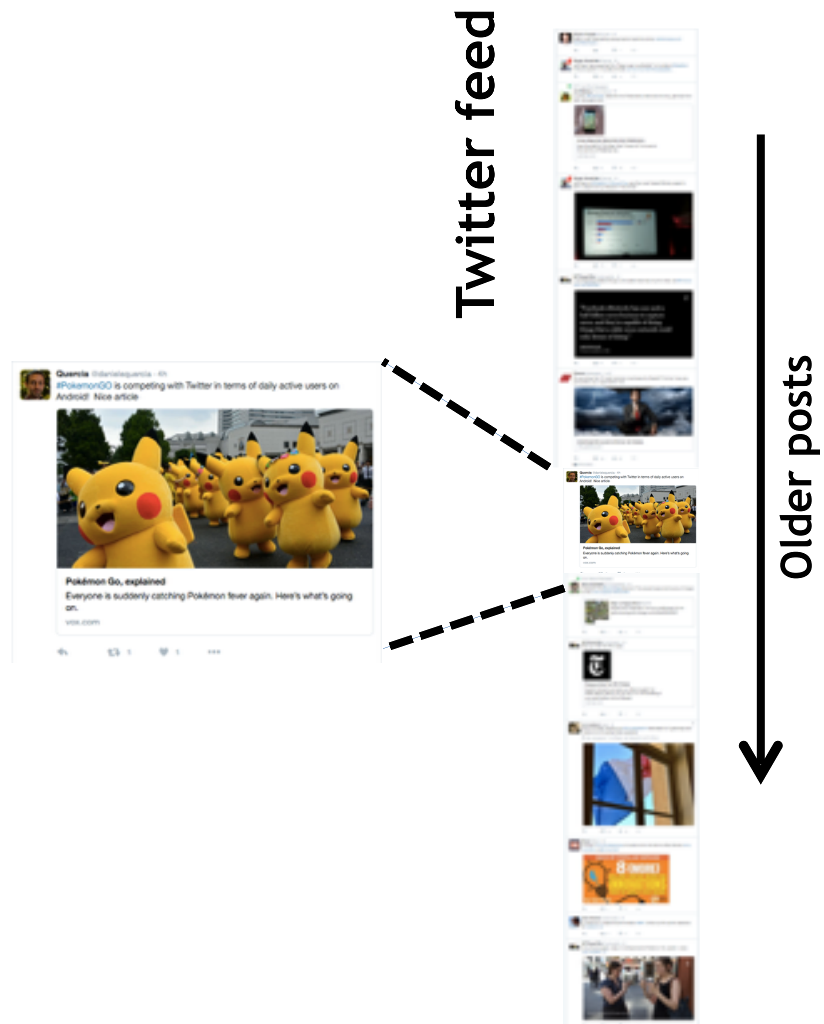
Broadcasted content

Audience reaction

Attention is scarce

Social media users follow many broadcasters





What are the best times to post?

THE HUFFINGTON POST

THE BLOG

The Best Times to Post on Social Media

Tech.Mic

Here Are the Be

So Your Picture

COMPANY

s To Post On

HubSpot

The Best Tim

Twitter, LinkedIn & Other Social Media Sites [Infographic]

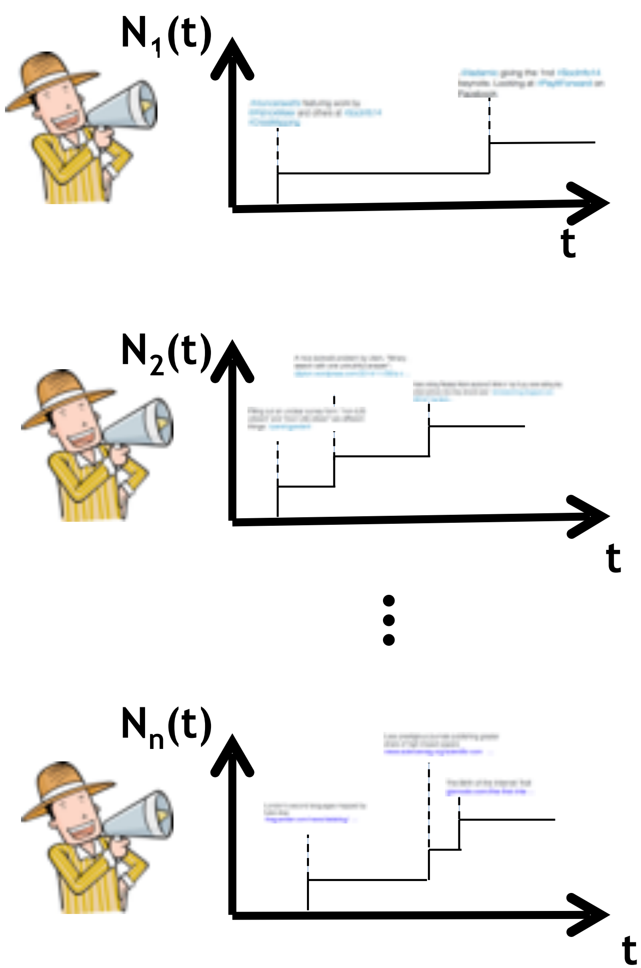
Forbes

For Brands And PR: When Is The Best Time To Post On Social Media?

Can we design an algorithm that tell us when to post to achieve **high visibility?**

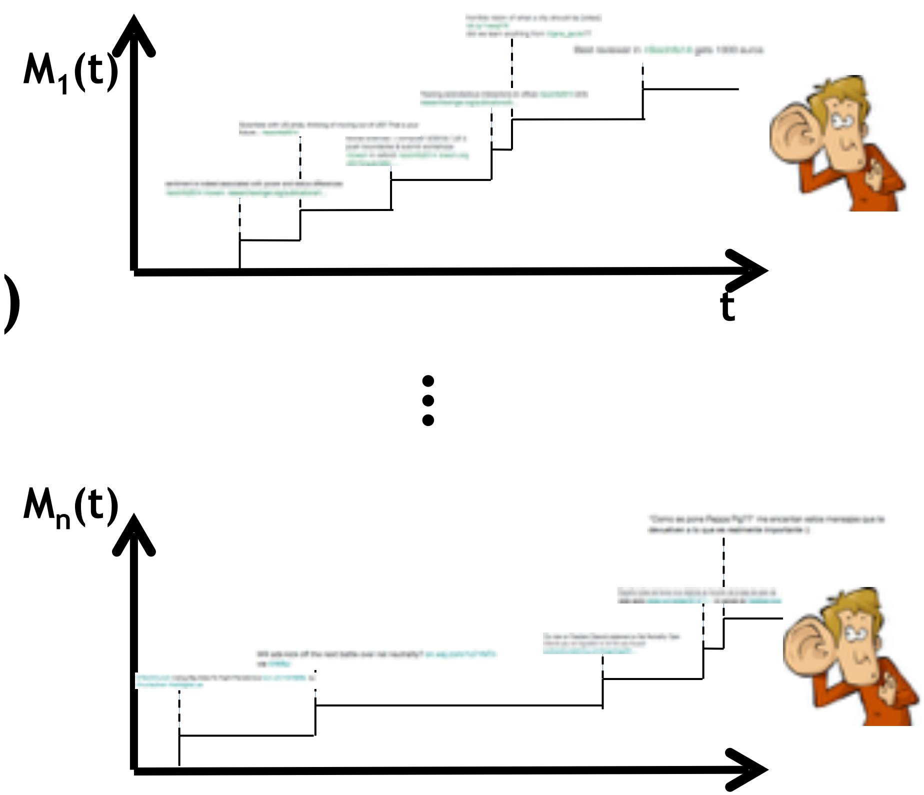
Representation of broadcasters and feeds

Broadcasters' posts as a counting process N(t)



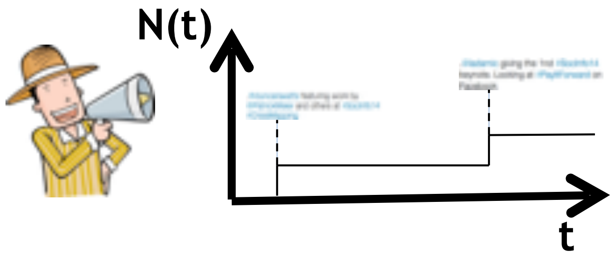
$$M(t) = A^T N(t)$$

Users' feeds as sum of counting processes M(t)



[Zarezade et al., WSDM 2017]

Broadcasting and feeds intensities



$$\mathbb{E}[\underbrace{dN(t)}_{\in\{0,1\}}|\mathcal{H}(t)] = \underbrace{\boldsymbol{\mu}(t)}\, dt \;\Rightarrow\; \mathbb{E}[\underbrace{dM(t)}_{\in\{0,1\}}|\mathcal{H}(t)] = \underbrace{\boldsymbol{\gamma}(t)}_{A^T\boldsymbol{\mu}(t)}\, dt$$

Broadcaster intensity function (tweets / hour)

Feed intensity function (tweets / hour)

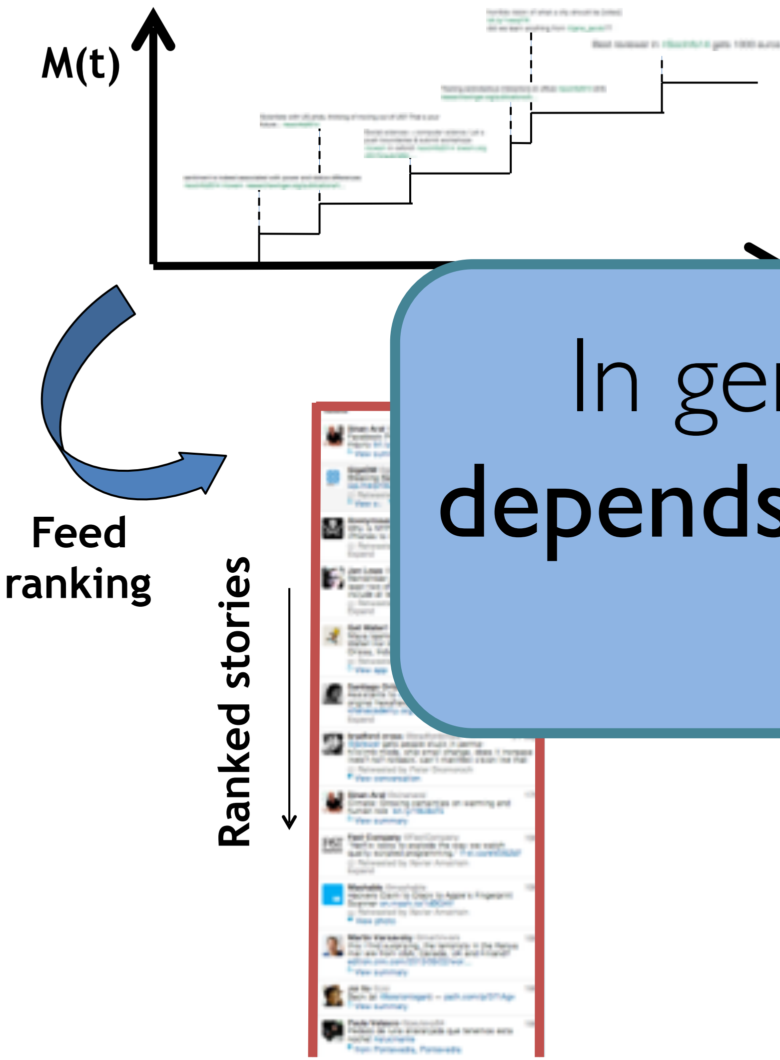
Given a broadcaster i and her followers

$$\boldsymbol{M}_{\setminus i}(t) = A^T \boldsymbol{N}(t) - A_i N_i(t)$$
$$\gamma_{j\setminus i}(t) = \gamma_j(t) - \mu_i(t)$$

Feed due to other broadcasters

[Zarezade et al., WSDM 2017]

Definition of visibility function



Visibility of broadcaster i at follower j

Position of the highest ranked tweet by broadcaster i in follower j's wall

$r_{ij}(t) = 0$ $r_{ij}(t') = 4$ $r_{ij}(t'') = 0$

In general, the **visibility depends** on the **feed ranking mechanism!**

Post by broadcaster u

Post by other broadcasters

[Zarezade et al., WSDM 2017]

Optimal control of temporal point processes

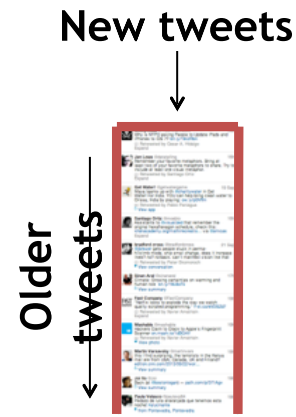
Formulate the when-to-post problem as a novel stochastic optimal control problem **(of independent interest)**

Visibility and feed dynamics	→	Optimizing visibility	→	Experiments
↕		↕		↕
System of stochastic equations with jumps		Optimal control of jumps		Twitter

[Zarezade et al., WSDM 2017]

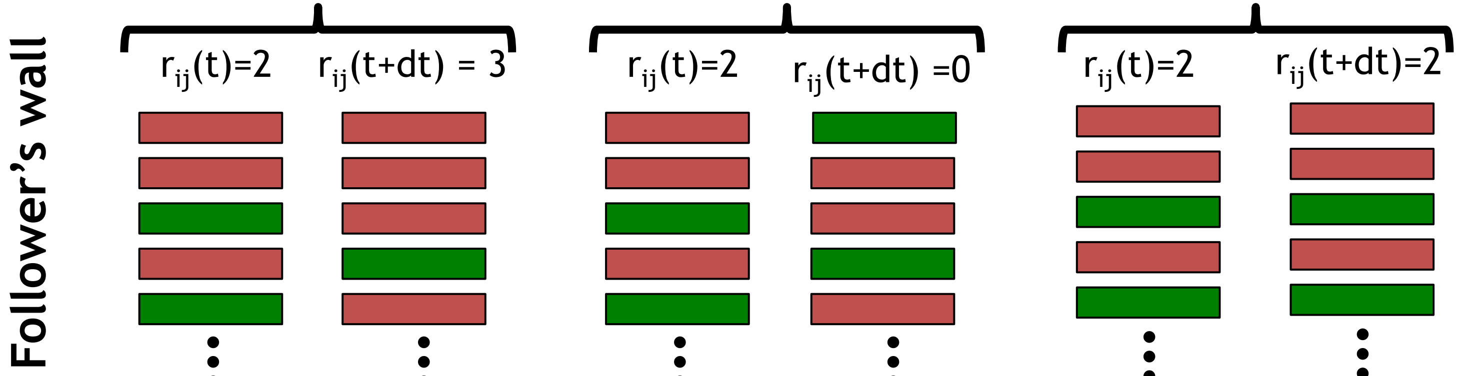
Visibility dynamics in a FIFO feed (I)

Reverse chronological order



$$\underbrace{r_{ij}(t+dt)}_{\text{Rank at t+dt}} = \underbrace{(r_{ij}(t)+1)dM_{j\setminus i}(t)(1-dN_i(t))}_{\text{Other broadcasters post a story and broadcaster i does not post}} + 0 + \underbrace{r_{ij}(t)(1-dM_{j\setminus i}(t))(1-dN_i(t))}_{\text{Nobody posts a story}}$$

Broadcaster i posts a story and other broadcasters do not post



[Zarezade et al., WSDM 2017]

Visibility dynamics in a FIFO feed (II)

$$r_{ij}(t+dt) = \quad (r_{ij}(t)+1)dM_{j\setminus i}(t)(1-dN_i(t)) + \; 0 + \; r_{ij}(t)(1-dM_{j\setminus i}(t))(1-dN_i(t))$$

Zero-one law $dN_i(t)dM_{j\setminus i}(t) = 0$

$$dr_{ij}(t) = -r_{ij}(t)\,dN_i(t) + dM_{j\setminus i}(t)$$

$r_{ij}(t+dt) - r_{ij}(t)$

Broadcaster i posts a story

Other broadcasters posts a story

Stochastic differential equation (SDE) with jumps

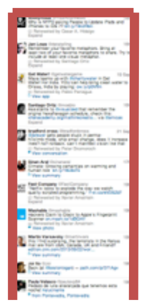
OUR GOAL:

Optimize $r_{ij}(t)$ over time, so that it is small, by controlling $dN_i(t)$ through the intensity $\mu_i(t)$

[Zarezade et al., WSDM 2017]

Feed dynamics

$N_i(t) \quad M_{j\backslash i}(t) = M(t) \rightarrow \gamma_{j\backslash i}(t) = \lambda(t)$



We consider a **general intensity:**

(e.g. Hawkes, inhomogeneous Poisson)

$$\lambda^*(t) = \underbrace{\lambda_0(t)}_{\text{Deterministic arbitrary intensity}} + \underbrace{\alpha \int_0^t g(t-s)dN(s)}_{\text{Stochastic self-excitation}}$$

Jump stochastic differential equation (SDE)

$$\left\{ d\lambda^*(t) = \left[\lambda_0'(t) + w\lambda_0(t) - w\lambda^*(t)\right] dt + \alpha dN_i(t) \right.$$

[Zarezade et al., WSDM 2017]

The when-to-post problem

$N_i(t) \quad M_{j\backslash i}(t) \qquad N_i(t) \quad M_{j\backslash i}(t) \qquad N_i(t) \quad M_{j\backslash i}(t)$

$\mu_i(t) = u(t) \rightarrow N_i(t)$

...

$dr_{ij}(t) = -r_{ij}(t)\, dN_i(t) + dM_{j\backslash i}(t)$ $\quad dr_{ij}(t) = -r_{ij}(t)\, dN_i(t) + dM_{j\backslash i}(t)$ $\quad dr_{ij}(t) = -r_{ij}(t)\, dN_i(t) + dM_{j\backslash i}(t)$

Optimization problem

$$\underset{u(t_0,t_f]}{\text{minimize}} \quad \mathbb{E}_{(N_i, \boldsymbol{M}_{\backslash i})(t_0,t_f]}\left[\overbrace{\phi(\boldsymbol{r}(t_f))}^{\text{Terminal penalty}} + \int_{t_0}^{t_f} \overbrace{\ell(\boldsymbol{r}(\tau), u(\tau))}^{\text{Nondecreasing loss}} d\tau\right]$$

$$\text{subject to} \quad u(t) \geq 0 \quad \forall t \in (t_0, t_f],$$

Dynamics defined by Jump SDEs

$$dr(t) = -r(t)\, dN(t) + dM(t)$$

$$d\lambda(t) = [\lambda_0'(t) + w\lambda_0(t) - w\lambda(t)]\, dt + \alpha\, dM(t)$$

[Zarezade et al., WSDM 2017]

Bellman's Principle of Optimality

Lemma. The **optimal cost-to-go** satisfies **Bellman's Principle of Optimality**

$$J(r(t), \lambda(t), t) = \min_{u(t,t+dt]} \mathbb{E}\left[J(r(t+dt), \lambda(t+dt), t+dt)\right] + \ell(r(t), u(t))\, dt$$

$$J(r(t+dt), \lambda(t+dt), t+dt) = J(r(t), \lambda(t), t) + dJ(r(t), \lambda(t), t)$$

$$0 = \min_{u(t,t+dt]} \mathbb{E}\left[dJ(r(t), \lambda(t), t)\right] + \ell(r(t), u(t))\, dt$$

$$dr(t) = -r(t)\, dN(t) + dM(t)$$
$$d\lambda(t) = [\lambda_0'(t) + w\lambda_0(t) - w\lambda(t)]\, dt + \alpha\, dM(t)$$

Hamilton-Jacobi-Bellman (HJB) equation

Partial differential equation in J (with respect to r, λ and t)

[Zarezade et al., WSDM 2017]

Solving the HJB equation

Consider a quadratic loss

$$\ell(r(t), u(t)) = \frac{1}{2} s(t)\, r^2(t) + \frac{1}{2} q\, u^2(t)$$

Favors some periods of times (e.g., times in which the follower is online)

Trade-offs visibility and number of broadcasted posts

We propose $J(r(t), \lambda(t), t)$ and then show that the optimal intensity is:

$$u^*(t) = q^{-1}\left[J(r(t), \lambda(t), t) - J(0, \lambda(t), t)\right]$$
$$= \sqrt{s(t)/q}\, r(t)$$

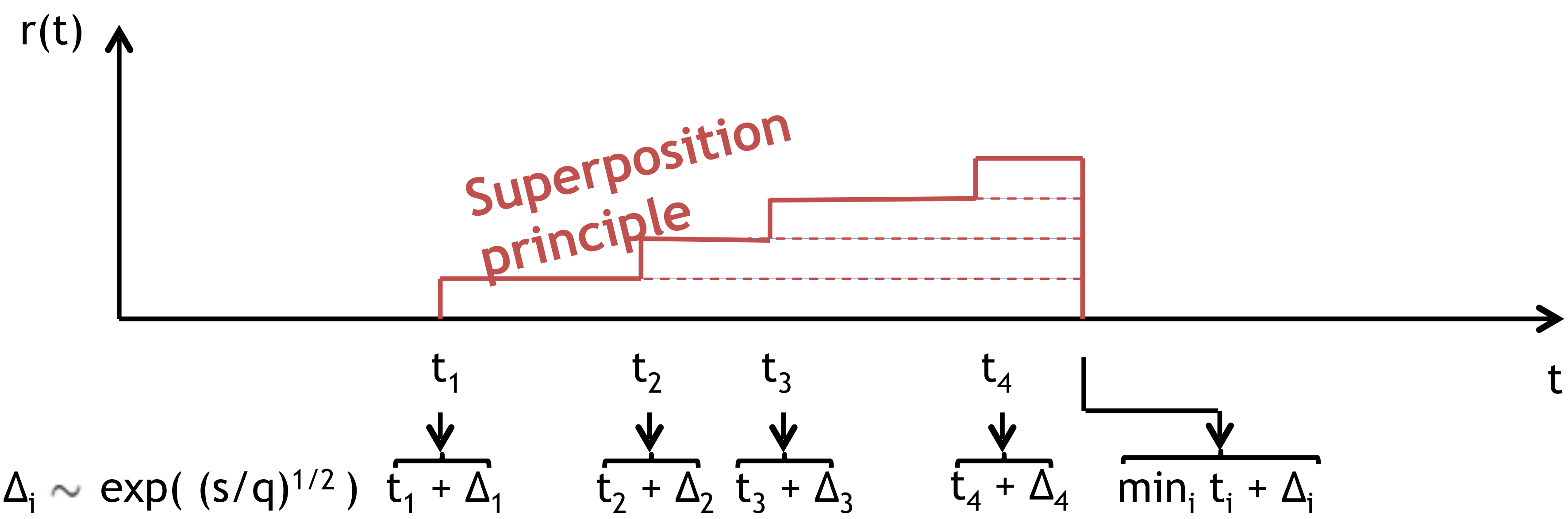
It only depends on the current visibility!

[Zarezade et al., WSDM 2017]

The RedQueen algorithm

Consider $s(t) = s$ $\longrightarrow$ $u^*(t) = (s/q)^{1/2}\, r(t)$

How do we sample the next time?



It only requires sampling $M(t_f)$ times!

[Zarezade et al., WSDM 2017]

When-to-post for multiple followers

Consider n followers and a quadratic loss:

$$\ell(\boldsymbol{r}(t), u(t), t) = \sum_{i=1}^{n} \frac{1}{2} s_i(t) r_i^2(t) + \frac{1}{2} q\, u^2(t)$$

Fav

(e.g., tim

d

Then, w

is:

We can easily adapt the efficient sampling algorithm to multiple followers!

$u^*(t$

$i=1$

It only depends on the current visibilities!

[Zarezade et al., WSDM 2017]

Novelty in the problem formulation

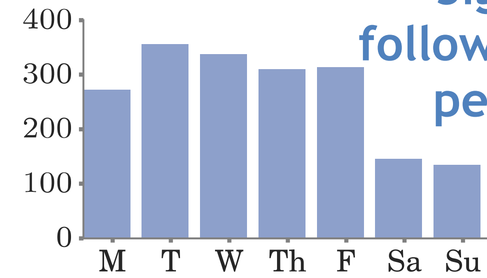
The problem formulation is unique in two key technical aspects:

I. The control signal is a conditional intensity
Previous work: time-varying real vector

II. The jumps are doubly stochastic
Previous work: memory-less jumps

[Zarezade et al., WSDM 2017]

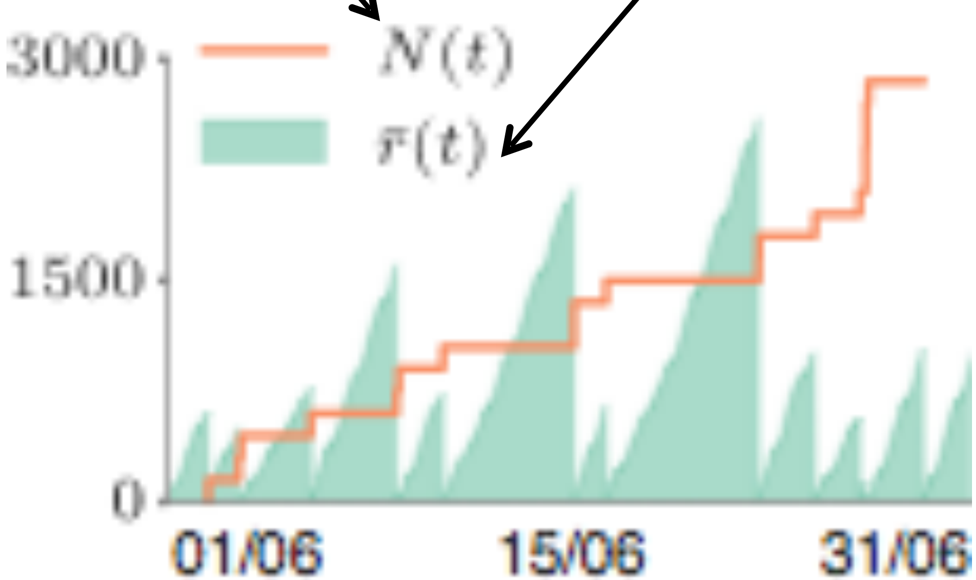
Case study: one broadcaster

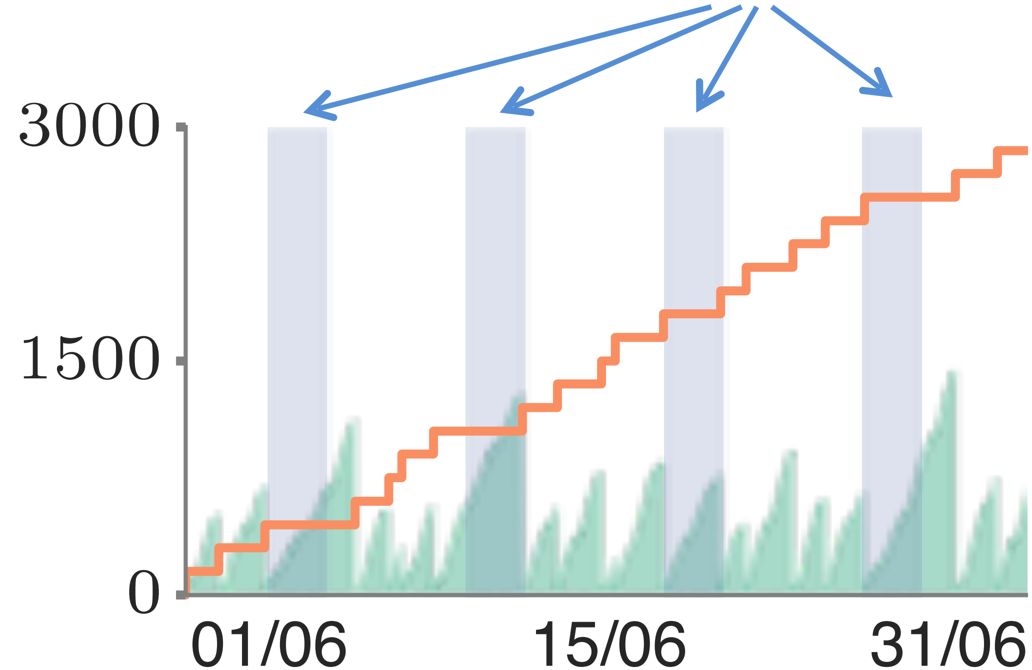


Significance: followers' retweets per weekday

Broadcaster's posts

Average position over time





True posts

$\frac{1}{T}\int_0^T \bar{r}(t)dt = 698.04$

40% lower!

REDQUEEN

$\frac{1}{T}\int_0^T \bar{r}(t)dt = 425.25$

[Zarezade et al., WSDM 2017]

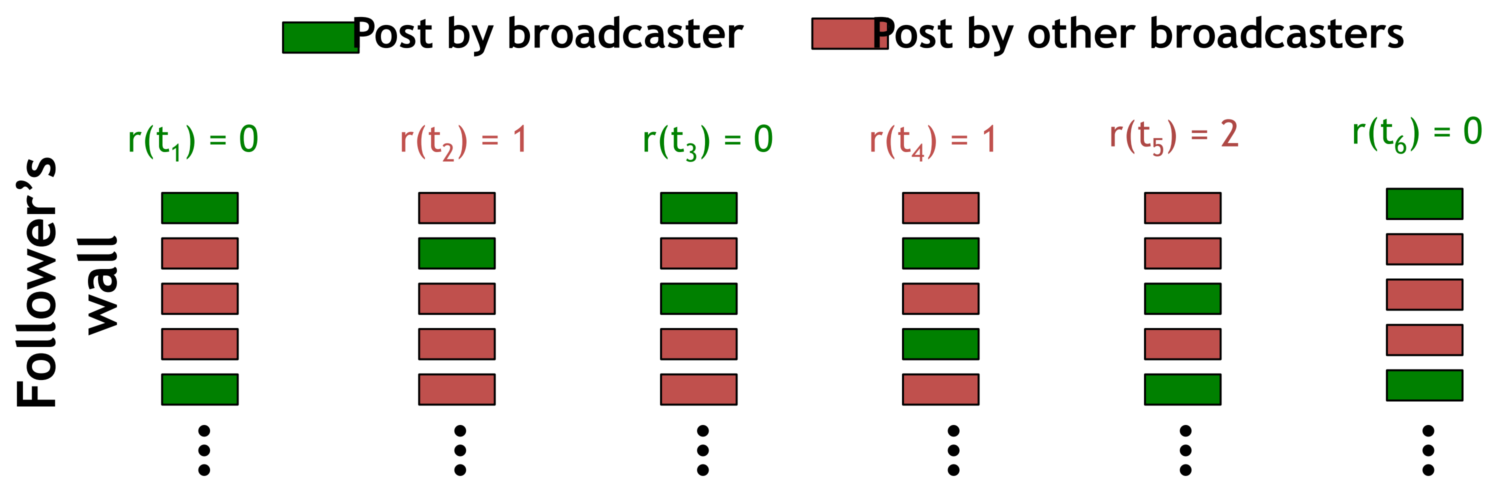
Evaluation metrics

Position over time

$$\int_0^T r(t)dt$$

Time at the top

$$\int_0^T \mathbb{I}(r(t)<1)dt$$

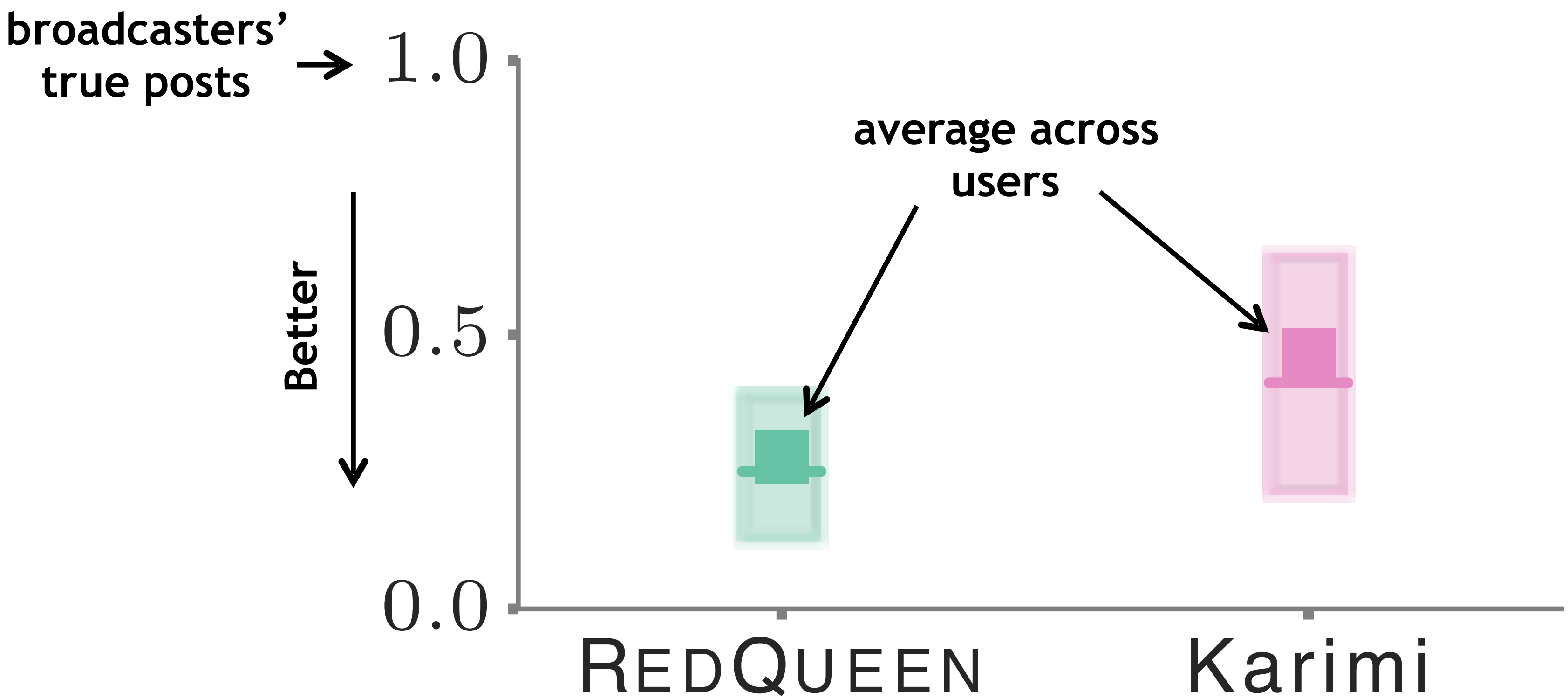


Position over time = $0x(t_2 - t_1) + 1x(t_3 - t_2) + 0x(t_4 - t_3) + 1x(t_5 - t_4) + 2x(t_6 - t_5)$

Time at the top = $(t_2 - t_1) + 0 + (t_4 - t_3) + 0 + 0$

[Zarezade et al., WSDM 2017]

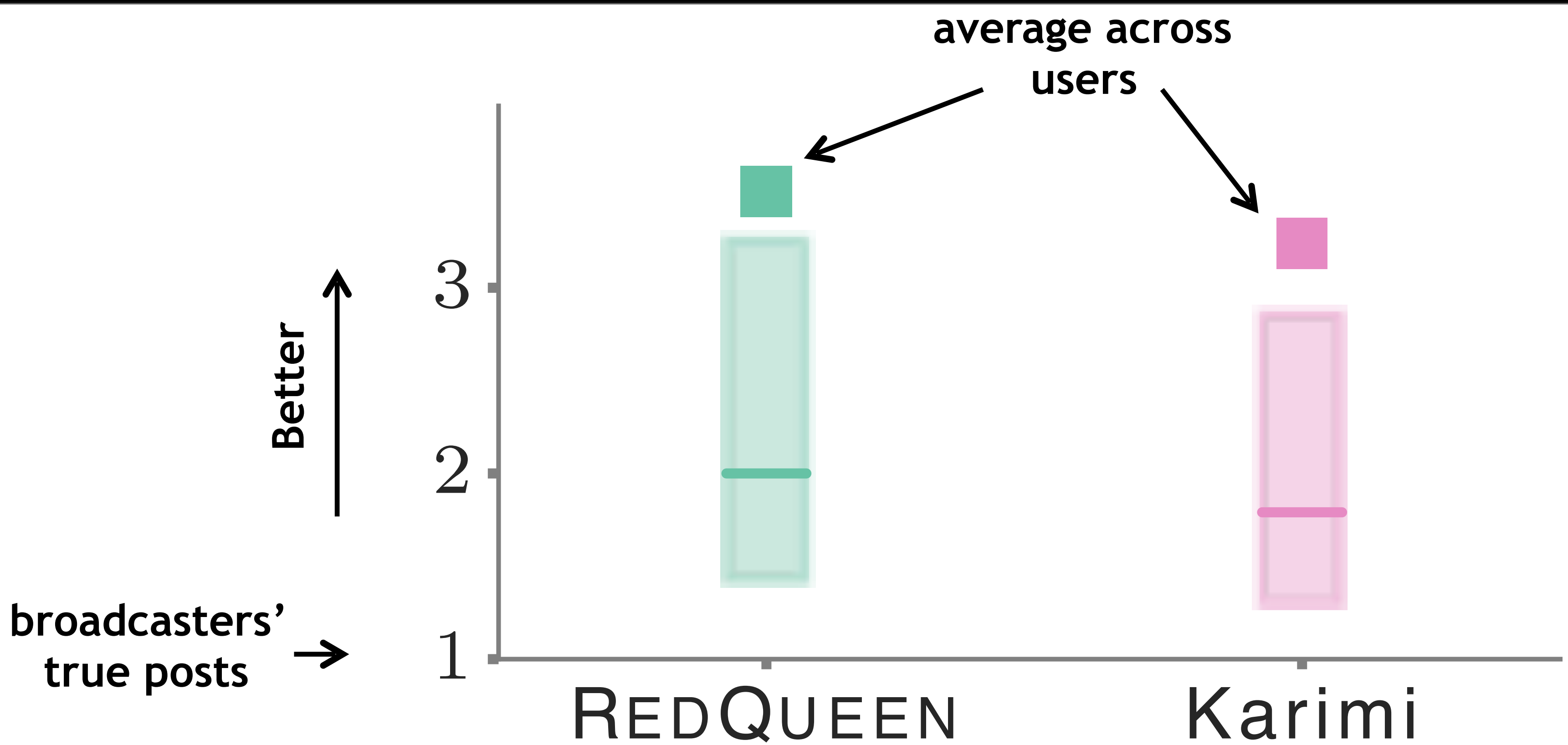
Position over time



It achieves **(i) 0.28x lower average position, in average, than the broadcasters' true posts** and **(ii) lower average position for 100% of the users.**

[Zarezade et al., WSDM 2017]

Time at the top



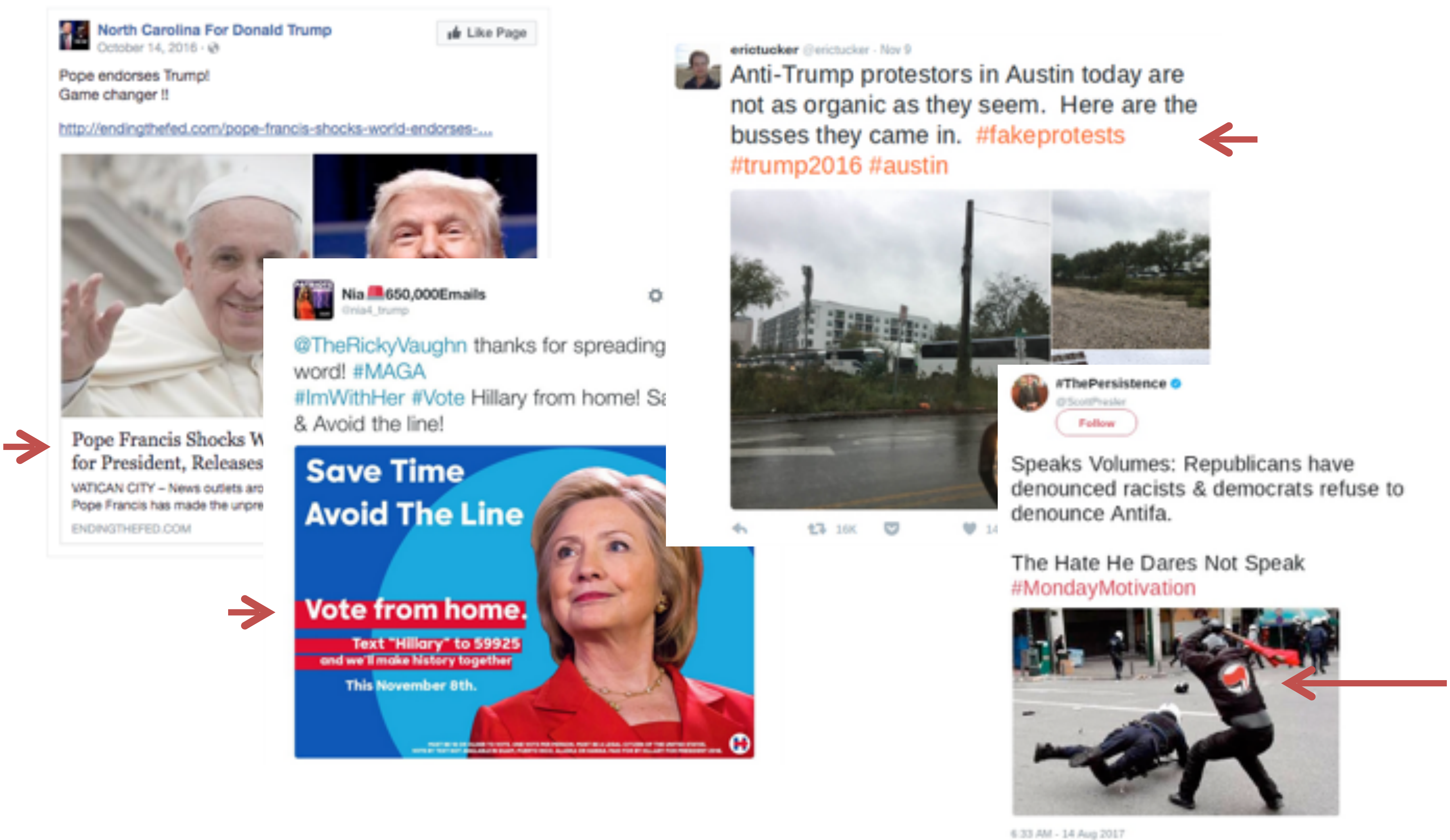
It achieves **(i) 3.5x higher time at the top, in average, than the broadcasters' true posts** and **(ii) higher time at the top for 99.1% of the users.**

[Zarezade et al., WSDM 2017]

Applications: Control

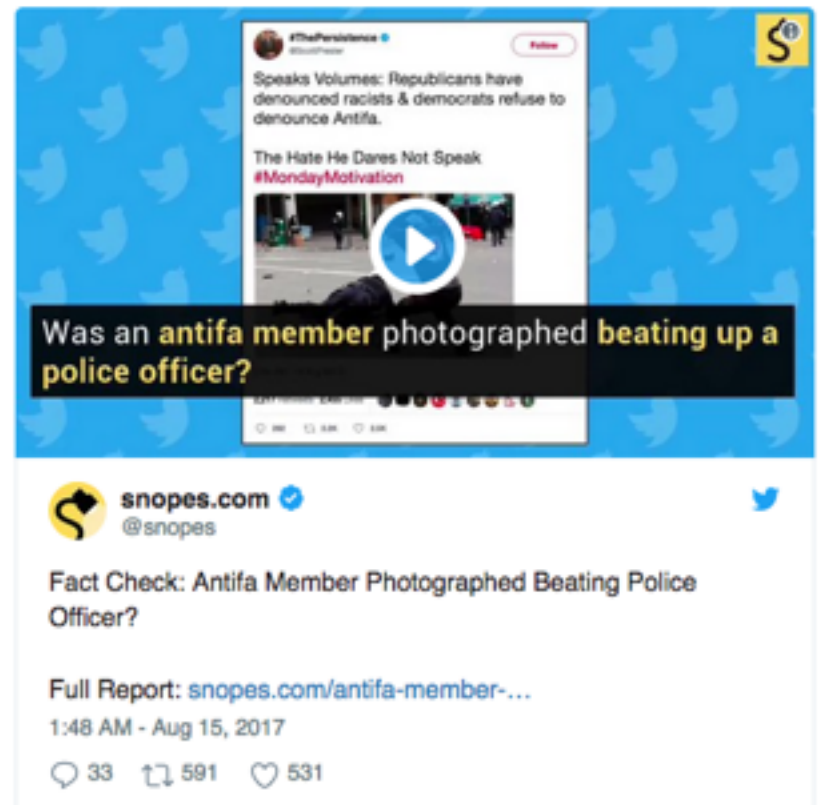
1. Influence maximization
2. Activity shaping
3. When to post
4. When to fact check

Opinionated, inaccurate, fake news



Solution: Resort to fact-checks by 3rd parties

Send information to trusted third parties (e.g., Snopes) for fact-checking



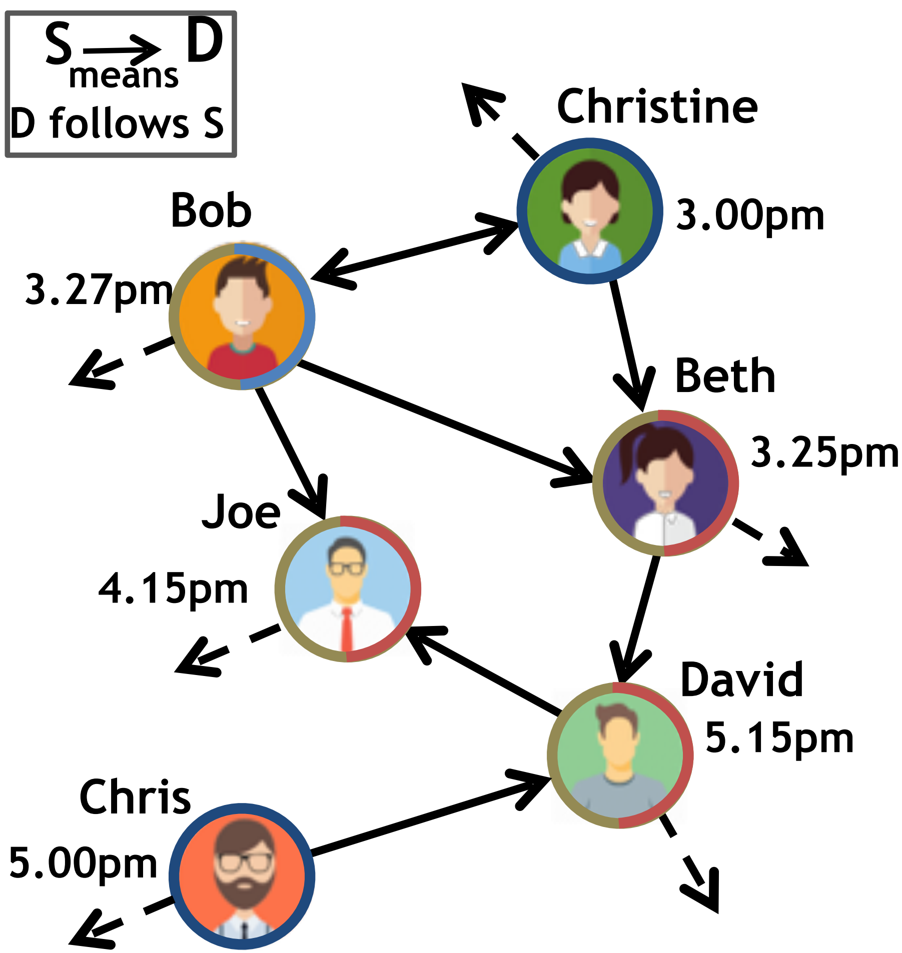
Challenges

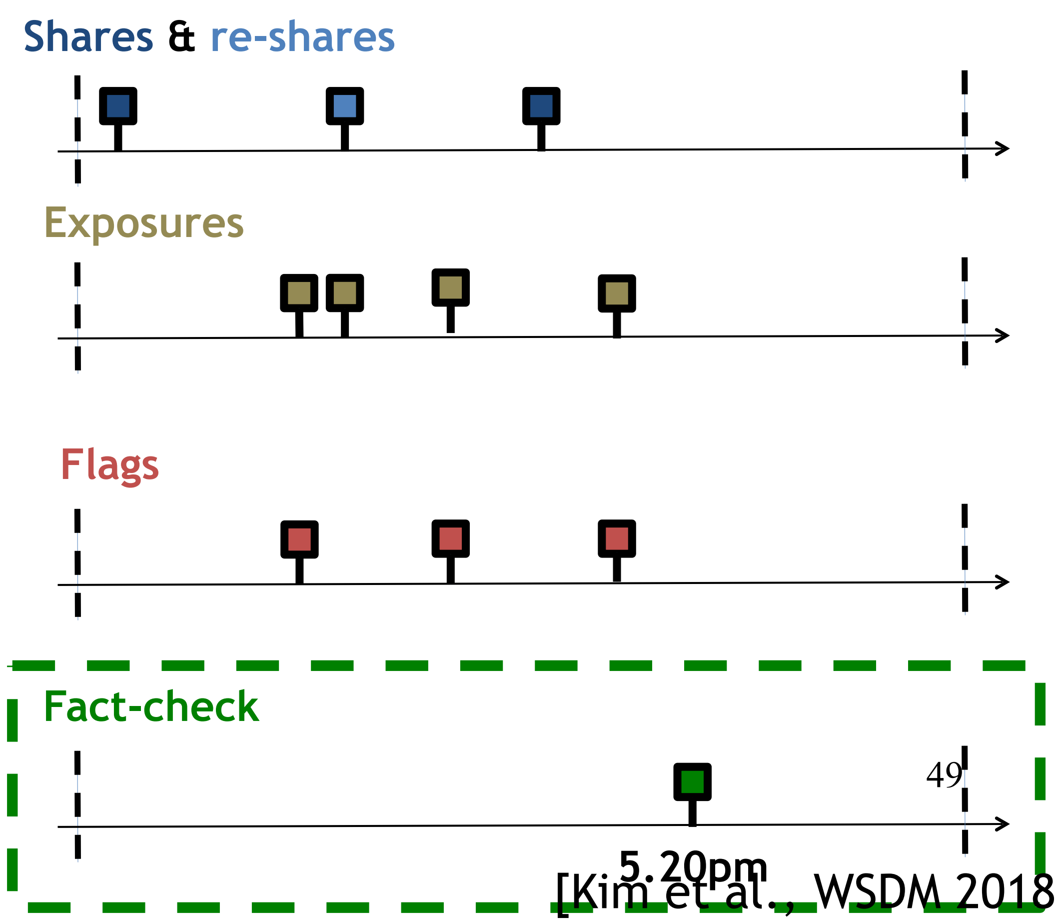
1. **Fact-checking** is ***costly***
2. Which **content** to **fact-check?**
3. What to do **after fact-checking?**

[Kim et al., WSDM 2018]

Detect & prevent = flags by crowd + fact check

Major social networking sites are testing the following mechanism





[Kim et al., WSDM 2018]

Challenges

Previous procedure faces several challenges:

Uncertainty on the number of **exposures** → This talk: **Probabilistic exposure models**

Flags can be **manipulated** → **Robust flag aggregation**

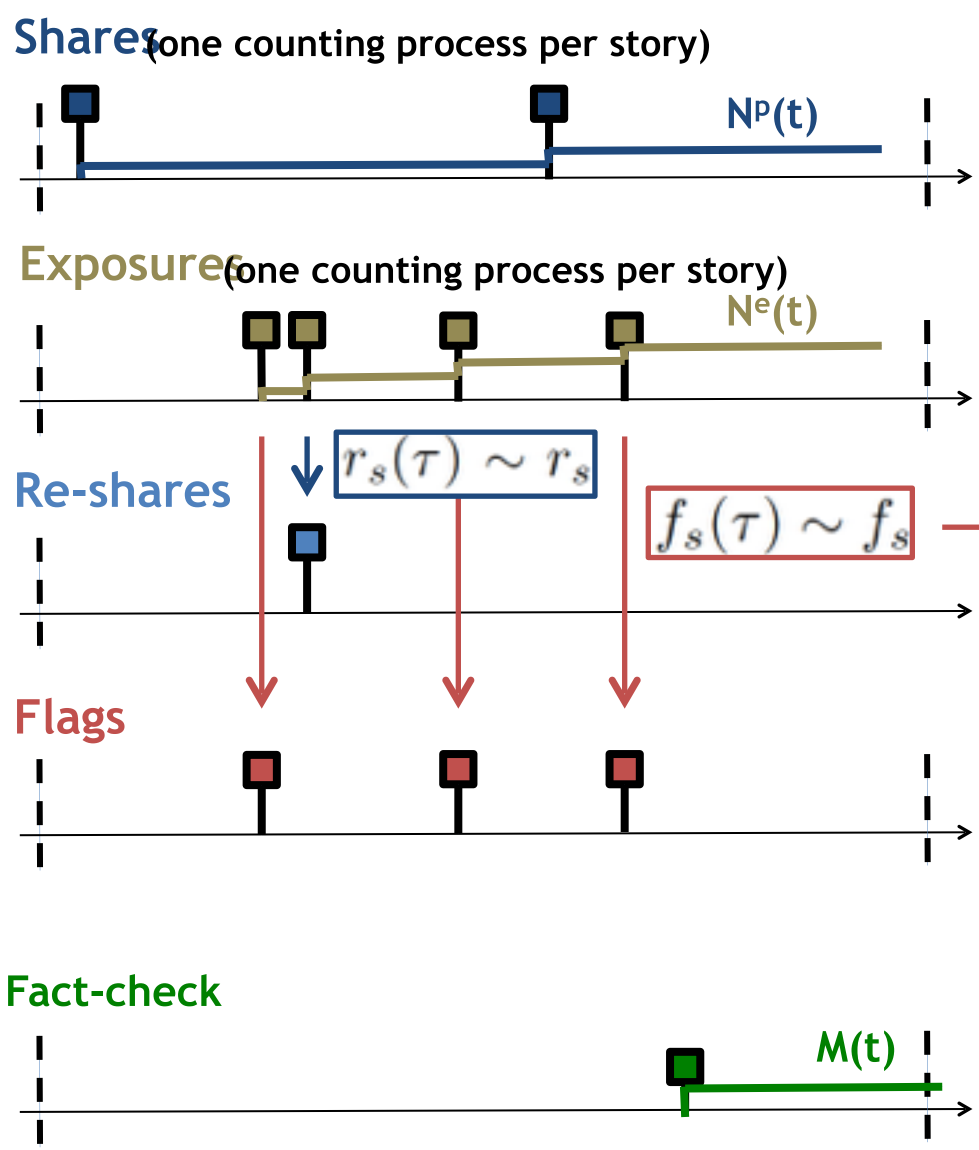
Fact-checking is costly

Tradeoff between flags & exposures

→ This talk: **Optimal fact-checking**

[Kim et al., WSDM 2018]

Procedure representation & modeling



Intensities or rates
(Shares/exposures per time unit)

$$\mathbb{E}[d\boldsymbol{N}^p(t)|\mathcal{H}(t)] = \boldsymbol{\lambda}^p(t)dt$$

$$\mathbb{E}[d\boldsymbol{N}^e(t)|\mathcal{H}(t)] = \boldsymbol{\lambda}^e(t)dt$$

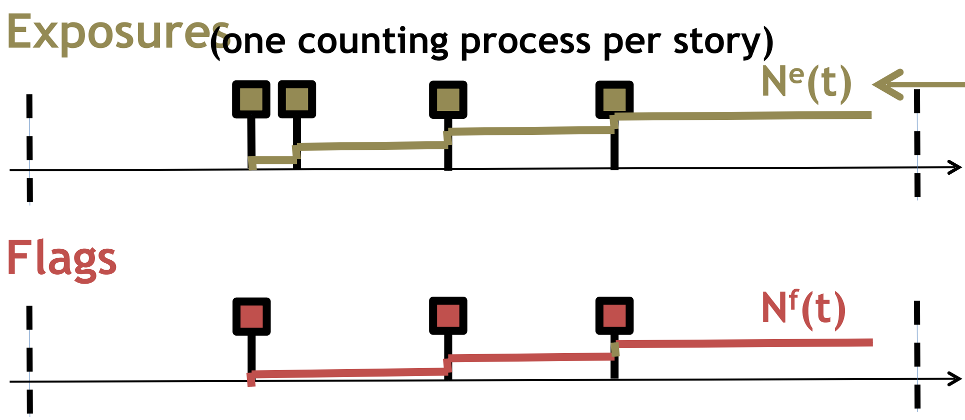
It is unknown
It depends on whether a story is misinformation
!

Survival rates
(Fact-checks per time unit)

$$\mathbb{E}[d\boldsymbol{M}(t)|\mathcal{H}(t)] = \boldsymbol{u}(t) \odot (1 - \boldsymbol{M}(t))dt$$

[Kim et al., WSDM 2018]

Rate of misinformation



$$\boxed{f_s(\tau) \sim f_s}$$

$$N_s^f(t) := \int_0^t f_s(\tau)\, dN_s^e(\tau)$$

Average number of users **exposed to misinformation** by time t:

Estimated from historical data

$$\bar{N}_s^m(t) := p_{m|s,f=1} N_s^f(t) + p_{m|s,f=0}\left(N_s^e(t) - N_s^f(t)\right)$$

$$dN_s^f(t) = f_s(t)\, dN_s^e(t)$$

$$f_s | N_s^f, N_s^e \sim Beta\left(\alpha + N_s^f(t), \beta + N_s^e(t) - N_s^f(t)\right)$$

Rate of misinformation:

$$\lambda_s^m(t)dt = \mathbb{E}[d\bar{N}_s^m(t) | \mathcal{H}(t)]$$

Conditional!
Allows for posteriors!

[Kim et al., WSDM 2018]

When to fact-check?

(Survival) rates
(Fact-checks per time unit)

$$\mathbb{E}[d\boldsymbol{M}(t)|\mathcal{H}(t)] = \boldsymbol{u}(t) \odot (1 - \boldsymbol{M}(t))dt$$

Find survival rates that minimize the

This is a stochastic optimal control problem for jump SDEs

S

$$\text{subject to} \quad \boldsymbol{u}(t) \geq 0 \quad \forall t \in (t_0, t_f],$$

Trade-off fact-checks vs misinformation

…given the dynamics of exposures, shares, reshares & flags

[Kim et al., WSDM 2018]

Solving the optimal control problem

We define the optimal cost-to-go J(…):

$$J(M(t), N^e(t), N^f(t), N^p(t), \lambda^e(t), t) = \underset{u(t,t_f]}{min} \mathbb{E}\left[\phi(\hat{\lambda}^m(t_f)) + \int_t^{t_f} \ell(\hat{\lambda}^m(\tau), u(\tau))d\tau\right]$$

Bellman's Principle of Optimality

$$0 = \min_{u(t,t+dt]} \left\{\mathbb{E}\left[dJ(M(t), N(t), N^f(t), N^p(t), \lambda^e(t), t)\right] + \ell(\hat{\lambda}^m(t), u(t))dt\right\}$$

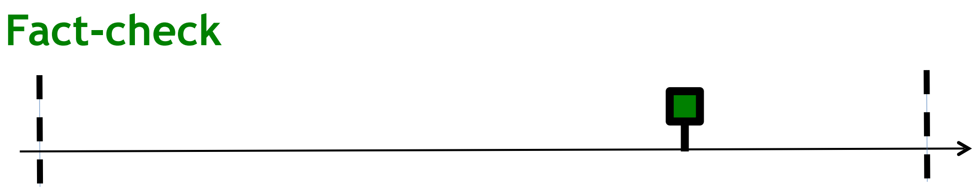
Dynamics
$dM(t)$, $dN(t)$, $dN^f(t)$, $dN^p(t)$, $d\lambda^e(t)$

Hamilton-Jacobi-Bellman (HJB) Equation

Partial differential equation in J (wrt M, N, N^f, N^p, λ^e, and t

[Kim et al., WSDM 2018]

Optimal solution for fact-checking



$$\mathbb{E}[d\boldsymbol{M}(t)|\mathcal{H}(t)] = \boldsymbol{u}(t) \odot (1 - \boldsymbol{M}(t))dt$$

For a general family of shares and exposure intensities

Given an additive quadratic loss, the optimal fact intensity for each story is given by

$$u_s^*(t) = q_s^{-\frac{1}{2}} \; \hat{\lambda}_s^m(t)$$

Parameter that trades-off fact-checks vs misinformation

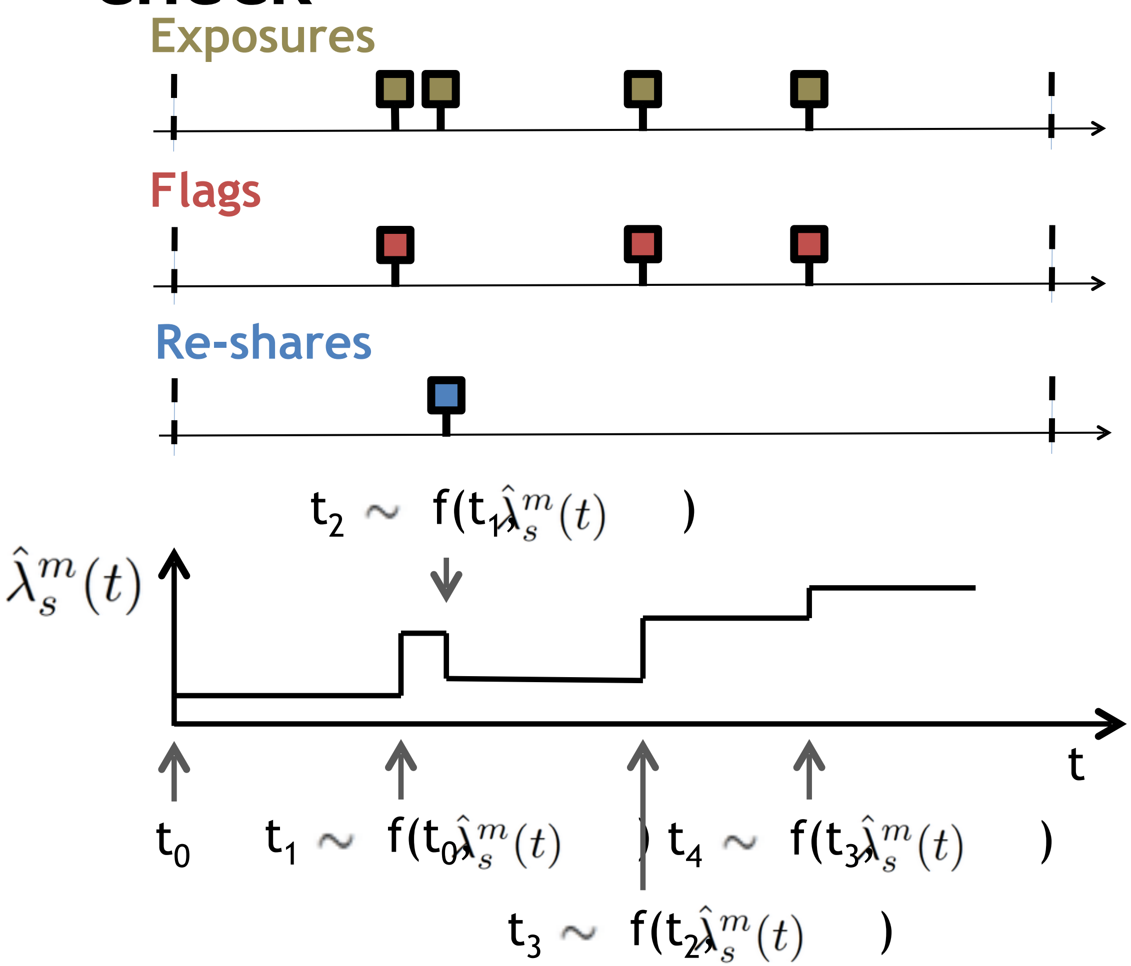
It only depends on the current rate of misinformation!

[Kim et al., WSDM 2018]

The CURB algorithm

Intuition

Adaptive planning of the time to fact check



The function f(..) uses:

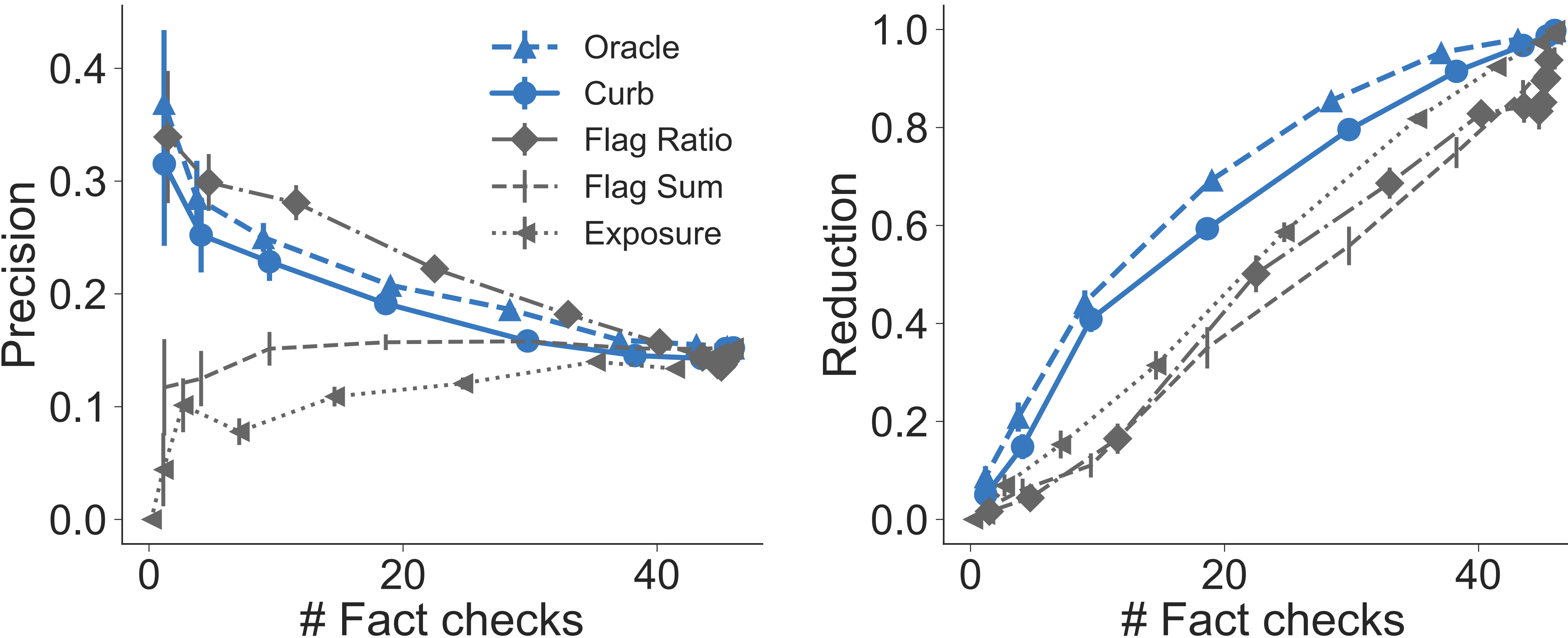
- → Superposition principle
- → Standard thinning

It only requires sampling $O(N^e(t_f))$ times!

Easy to implement

[Kim et al., WSDM 2018]

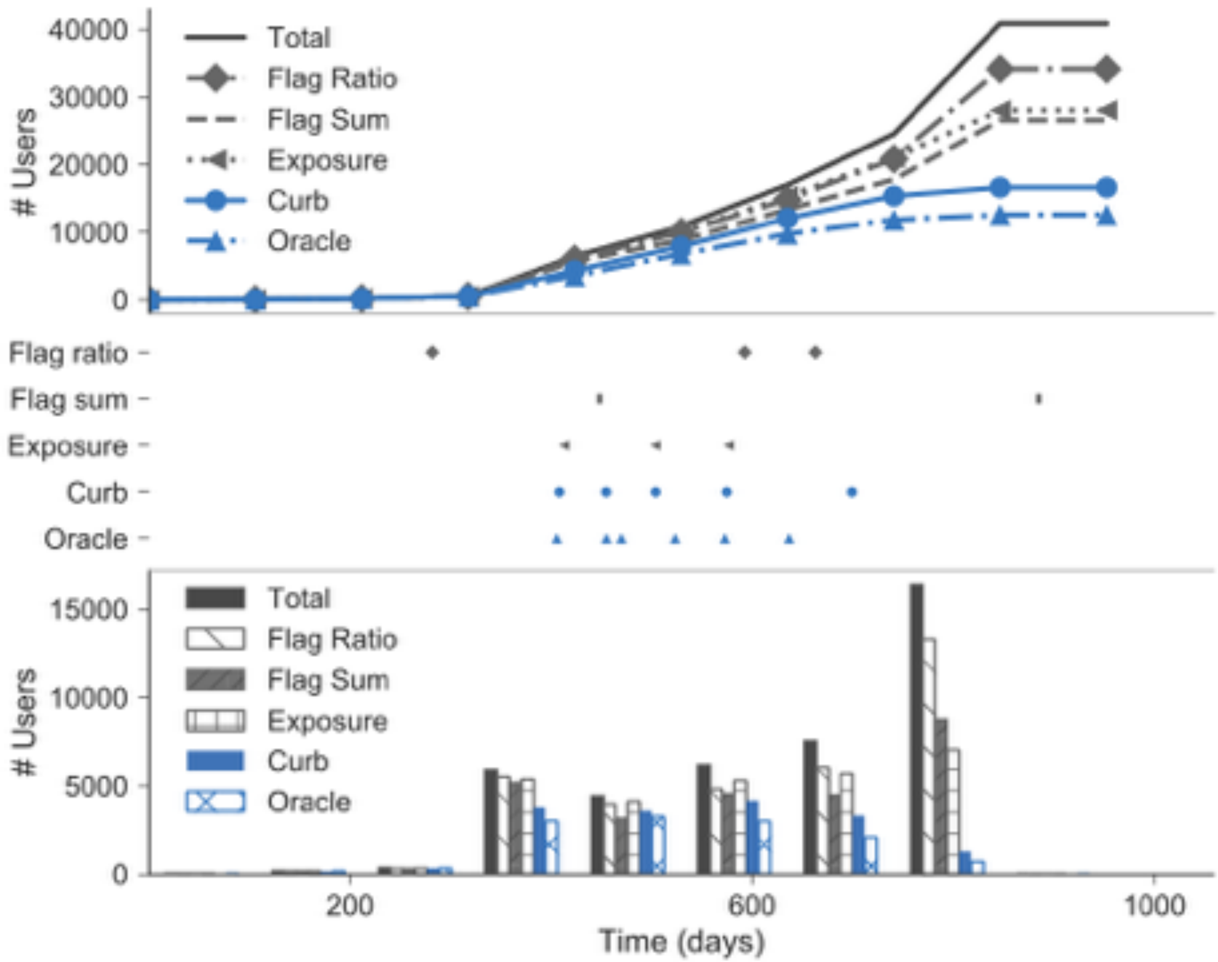
Performance vs # of fact checks (Twitter)



CURB and the oracle achieve optimal tradeoff between precision & misinformation reduction

[Kim et al., WSDM 2018]

Misinformation reduction over time (Twitter)



Both CURB and the oracle prevent the spread of misinformation before it becomes viral

[Kim et al., WSDM 2018]

Representation: Temporal Point processes

1. Intensity function
2. Basic building blocks
3. Superposition
4. Marks and SDEs with jumps

Applications: Models

1. Information propagation
2. Opinion dynamics
3. Information reliability
4. Knowledge acquisition

Applications: Control

1. Influence maximization
2. Activity shaping
3. When to post
4. When to fact check

Thanks!

Interested?
Internships/PhDs/postdoc positions at
learning.mpi-sws.org