电子科技大学

University of Electronic Science and Technology of China

SEISMIC: A Self-Exciting Point Process Model for Predicting Tweet Popularity

Reporter: Ruiqi Yang

Data Mining Lab, Big Data Research Center, UESTC

➢ Introduction

purpose: for predicting tweet popularity

i.e. to predict how many reshares a given post will ultimately receive.

conceptual model: information cascade

it simplifies users' resharing behavior & helps explain aggregating effects of individuals

statistical model: self-exciting point process

- what's information cascade?

An information cascade occurs when a person observes the actions of others and then—despite possible contradictions in his/her own private information signals—engages in the same acts.

condition:

1, decision 2, limited action space 3, sequentially 4, private 5, infer

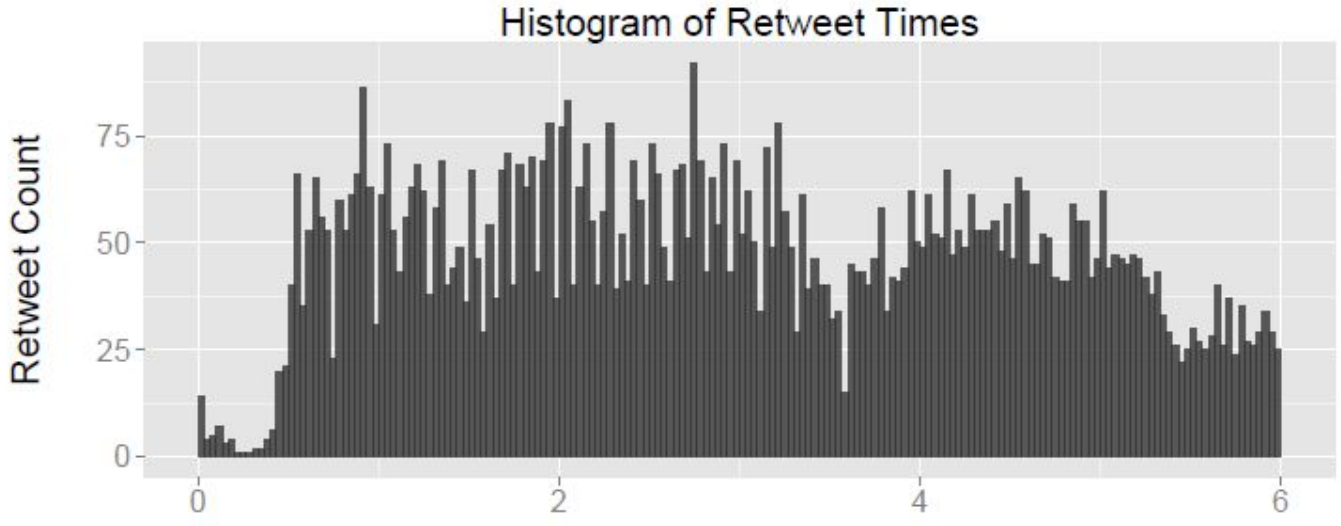
example: spring outing

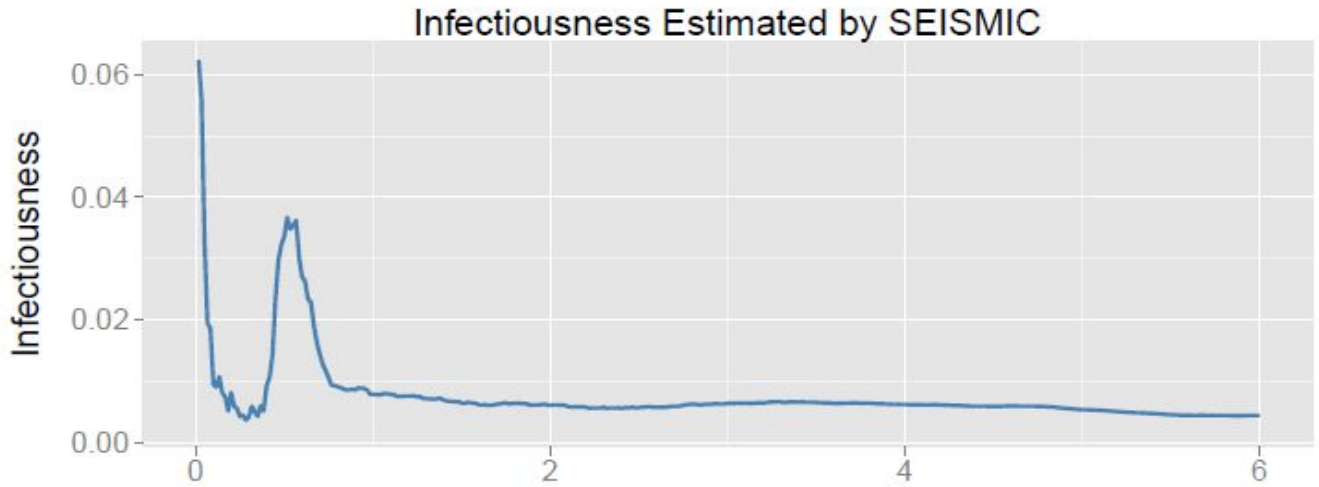
- what's about self-exciting point process?

  a point process is a type of random process for which any one realisation consists of a set of isolated points either in time or geographical space, or in even more general spaces.

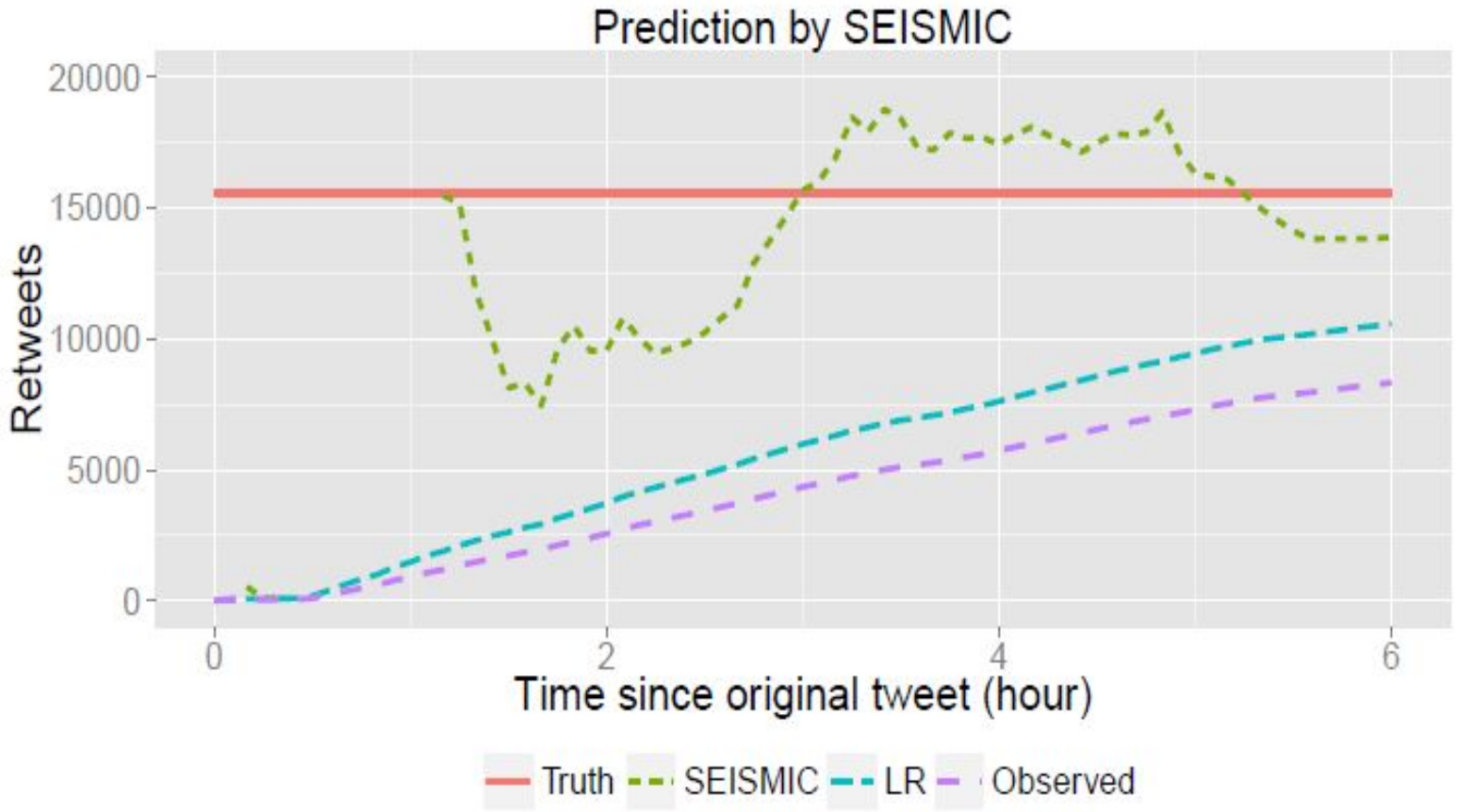
  example: the arrival of customers in a queue

  self-excting: all the previous instances(i.e. reshares) influence the future evolution of the process.





intuitily, it seems that if the prediction is precise, the trends in the two pictures should matain the same.



- related work

the two classifications of recent models for predicting size of information cascades : feature based and point process based methods.

point process: directly models the formation of an information cascade in a network. a major distinction between SEISMIC and Hawkes processes based method is that the process intensity λ_t in the former depends on another stocastic process p_t

feature based methods:

features - content features, original poster features, network structural features and temporal features.

learning methods - probabilistic collaborative filtering, regression trees, content-based models, and passive-aggressive algorithms.

laborious feature engineering and extensive training are crutial

The SEISMIC model

➢ Human reaction time

In order to predict the cascade size, we need to know how long it takes for a person to reshare a post.

probability density $\Phi(s)$, is also called a memory kernel because it measures a physical/social system's memory of stimuli.

the distribution of human response time:
heavy-tailed in social networks,
usually the tail follows a power-law with exponent $\in [1,2]$
or a log-normal distribution

$\Phi(s)$ only needs to be estimated once per network

➢ Post infectiousness

We assume each post w is associated with a time dependent, intrinsic infectiousness parameter $p_t(w)$

most existing methods studying self-exciting point processes assume p_t to be fixed over time.

a phase transition phenomenon at certain critical threshold.

in reality, R_t is always bounded due to the finite size of the network.

this is an extension to the standard self-exciting point process(also called the Hawkes process)

The definition of the intensity λ_t of R_t, which simply measures the rate of obtaining an additional reshare at time t.

$$\lambda_t = \lim_{\Delta \to 0} \frac{P(R_{t+\Delta} - R_t = 1)}{\Delta}$$

$$\lambda_t = p_t \cdot \sum_{t_i \le t, i \ge 0} n_i \Phi(t - t_i)$$

➢Predicting information cascades

sample-function density is defined as the joint probability of the number of reshares in the time interval $[t_0, t)$ and the density of their occurrence times.

$$P(R = r, t_1, ..., t_r) = \prod_{i=1}^{R_t} \{\lambda_{t_i} \cdot \exp\{-\int_{t_{i-1}}^{t_i} \lambda_s ds\}\}$$

$$= \prod_{i=1}^{R_t} \lambda_{t_i} \cdot \exp\{-\int_{t_0}^{t} \lambda_s ds\}.$$

by taking derivative of the ln:

$$\frac{d \ln P(R = r, t_1, ..., t_r)}{dp} = \sum_{i=1}^{R_t} \frac{1}{p} - \int_{t_0}^{t} \sum_{t_i \le s, i \ge 0} n_i \Phi(s - t_i) ds = 0$$

$$\Rightarrow \hat{p}_t = \frac{R_t}{\sum_{i=0}^{R_t} n_i \int_{t_i}^{t} \Phi(s - t_i) ds}$$

the denominator, denoted as N_t^e hereafter, can be interpreted as the accumulative "effective" number of exposed users to the post.

take $t \to \infty$, $\int_{t_i}^{t} \Phi(s - t_i) ds \to 1$

$$\hat{p}_{\infty} = \frac{R_{\infty}}{\sum_{i=0}^{R_{\infty}} n_i} = \frac{1}{\frac{1}{R_{\infty}} \sum_{i=0}^{R_{\infty}} n_i} \approx \frac{1}{n_*}$$

Thus, the assumption that p_t to be a constant over time is unreasonable because most posts will have the same infectiousness in the end. What's more, cannot explain the bursty and volatile dynamics information cascades.

weighting kernel $K_t(s)$

$$K_t(s) = \max\{0, 1 - \frac{2s}{t}\}, s > 0.$$

then,

$$\hat{p}_t = \frac{\int_{t_0}^{t} K_t(t-s)dR_s}{\int_{t_0}^{t} K_t(t-s)dN_s^e} = \frac{\sum_{i=1}^{R_t} K_t(t-t_i)}{\sum_{i=0}^{R_t} n_i \int_{t_i}^{t} K_t(t-s)\Phi(s-t_i)ds}$$

1, it quickly discards the unstable and potentially explosive period at the beginning.

2, it takes into account posts in a larger window size as time t increases

3, it up-weights the most recent posts and gradually down-weights older posts.

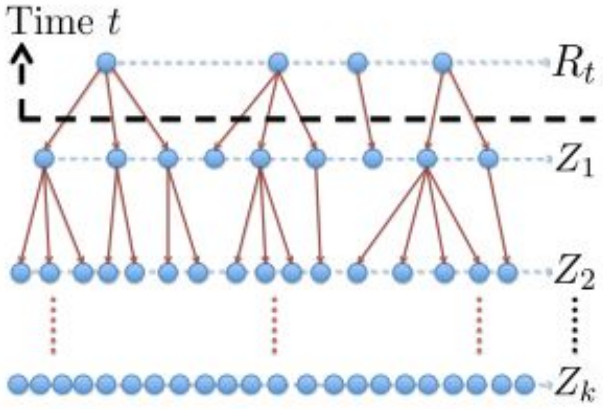
assume the (out-)degrees in the network are i.i.d with expectation n_* and the infectiousness parameter is a constant for $s \geq t$

$$\mathbb{E}[R_\infty | \mathcal{F}_t] = \begin{cases} R_t + \dfrac{p(N_t - N_t^e)}{1 - pn_*}, & \text{if } p < \dfrac{1}{n_*}, \\ \infty, & \text{if } p \geq \dfrac{1}{n_*}. \end{cases}$$

given Z_1, the sequence of random variables Z_k defines a Galton-Watson tree with offspring expectation $\mu = n_* p$

$$\forall k > 1, \mathbb{E}\left[Z_{k+1} | Z_k\right] = \mu \, Z_k$$

$$\mathbb{E}\left[\sum_{k=1}^{\infty} Z_k \middle| Z_1\right] = \frac{Z_1}{(1-\mu)} = \frac{Z_1}{(1-n_* p)}.$$



$$\mathbb{E}[R_\infty | \mathcal{F}_t] = R_t + \mathbb{E}\left[\sum_{k=1}^{\infty} Z_k\right] = R_t + \frac{\mathbb{E}[Z_1]}{(1-n_* p)},$$

Algorithm 1 SEISMIC: Predict final cascade size

Purpose: For a given post at time t, predict its final reshare count

Input: Post resharing information: t_i and n_i for $i = 0, \ldots, R_t$.

Algorithm:

$N_t = 0, N_t^e = 0$

for $i = 0, \ldots, R_t$ **do**

N_t += n_i

N_t^e += $n_i \int_{t_i}^{t} \phi(s - t_i) ds$ (Sec. 3.1)

end for

$\hat{R}_\infty(t) = R_t + \alpha_t \hat{p}_t (N_t - N_t^e)/(1 - \gamma_t \hat{p}_t n_*)$ (Alg. 2)

Deliver: $\hat{R}_\infty(t)$

➢Details of Algorithm

$$\int_{t_i}^{t} K_t(t-s)\Phi(s-t_i)ds$$

$$\phi(s) = \begin{cases} c & \text{if } 0 < s \le s_0, \\ c(s/s_0)^{-(1+\theta)} & \text{if } s > s_0. \end{cases}$$

$$K_t(s) = \max\left\{1 - \frac{2s}{t}, 0\right\}, \quad s > 0.$$

1, $0 < t - t_i \le s_0$

2, $s_0 < t - t_i \le win$

3, $win < t - t_i \le win + s_0$

4, $win + s_0 < t - t_i$

$$win = \frac{t}{2}$$

$$\int_{t_1}^{t_2} K_t(t-s)\Phi(s-t_i)ds$$

$$= c(t_2 - t \cdot t_2 / win + 1/win \cdot t_2^2/2) - c(t_1 - t \cdot t_1 / win + 1/win \cdot t_1^2/2) \cdots\cdots\cdots\cdots (1)$$

or

=

$$c \cdot s_0^{1+\theta} \cdot (t_2 - t_i)^{-\theta} \cdot (t_i/win - \theta + (\theta-1) \cdot t/win - \theta/win \cdot t_2 + 1)/((\theta-1)\cdot\theta) -$$

$$c \cdot s_0^{1+\theta} \cdot (t_1 - t_i)^{-\theta} \cdot (t_i/win - \theta + (\theta-1) \cdot t/win - \theta/win \cdot t_1 + 1)/((\theta-1)\cdot\theta) \cdots (2)$$

case1: linear(ti, t)

case2: linear(ti, ti+s0) + power(ti+s0, t)

case3: linear(t-win, ti+s0) + power(ti+s0, t)

case4: power(t-win, t)

given an example：

t=3600,win=1800,{(0,33),(800,46828),(1600,208),(3000,37),(3500,137)}

$$1,0 < 3600 - t_i \le 300$$

$$2,300 < 3600 - t_i \le 1800$$

$$3,1800 < 3600 - t_i \le 2100$$

$$4,2100 < 3600 - t_i$$

$$\tilde{R}_t = 0, \tilde{R}_t += K_t(3600) + K_t(2800) + K_t(2000) + K_t(600) + K_t(100)$$

$$\tilde{N}_t^e = 0, \tilde{N}_t = 33 + 46828 + 208 + 37 + 137 = 47243$$

$$\tilde{N}_t^e += 33 \cdot power(1800,3600) +$$
$$46828 \cdot power(1800,3600) +$$
$$208 \cdot [linear(1800,1900) + power(1900,3600)] +$$
$$37 \cdot [linear(3000,3300) + power(3300,3600)] +$$
$$137 \cdot linear(3500,3600)$$

$$p_t = \tilde{R}_t / \tilde{N}_t^e, R_\infty = R_t + p_t(N_t - N_t^e)/(1 - p_t n_*)$$

- computational complexity of SEISMIC

  the overall computational cost of SEIMIC is linear in the observed number of reshares R_t of a given post by time t.

Algorithm 2 Compute real-time infectiousness $\hat{p}(t)$

Purpose: For a given post w, calculate infectiousness p_t with information about w prior to time t

Input: Post resharing information: t_i and n_i for $i = 0, \ldots, R_t$.

Algorithm:

$\tilde{R}_t = 0$, $\tilde{N}_t^e = 0$

for $i = 0, \ldots, R_t$ **do**

$\tilde{R}_t$ += $K_t(t - t_i)$

end for

for $i = 0, \ldots, R_t$ **do**

$\tilde{N}_t^e$ += $n_i \int_{t_i}^{t} K_t(t - s)\phi(s - t_i)ds$ (Sec. 4.1)

end for

$p_t = \tilde{R}_t / \tilde{N}_t^e$

Deliver: p_t

EXPERIMENTS

- Data description

  complete set of over 3.2 billion tweets and retweets on Twitter from 2011.10.7 to 2011.11.7.

  tweet id | posting time | retweet time | number of followers

  choose a subset of tweets with at least 50 retweets.

  166,076 tweets satisfies.

  training set: tweets of the first 7days

  testing set: tweets of the next 8 days

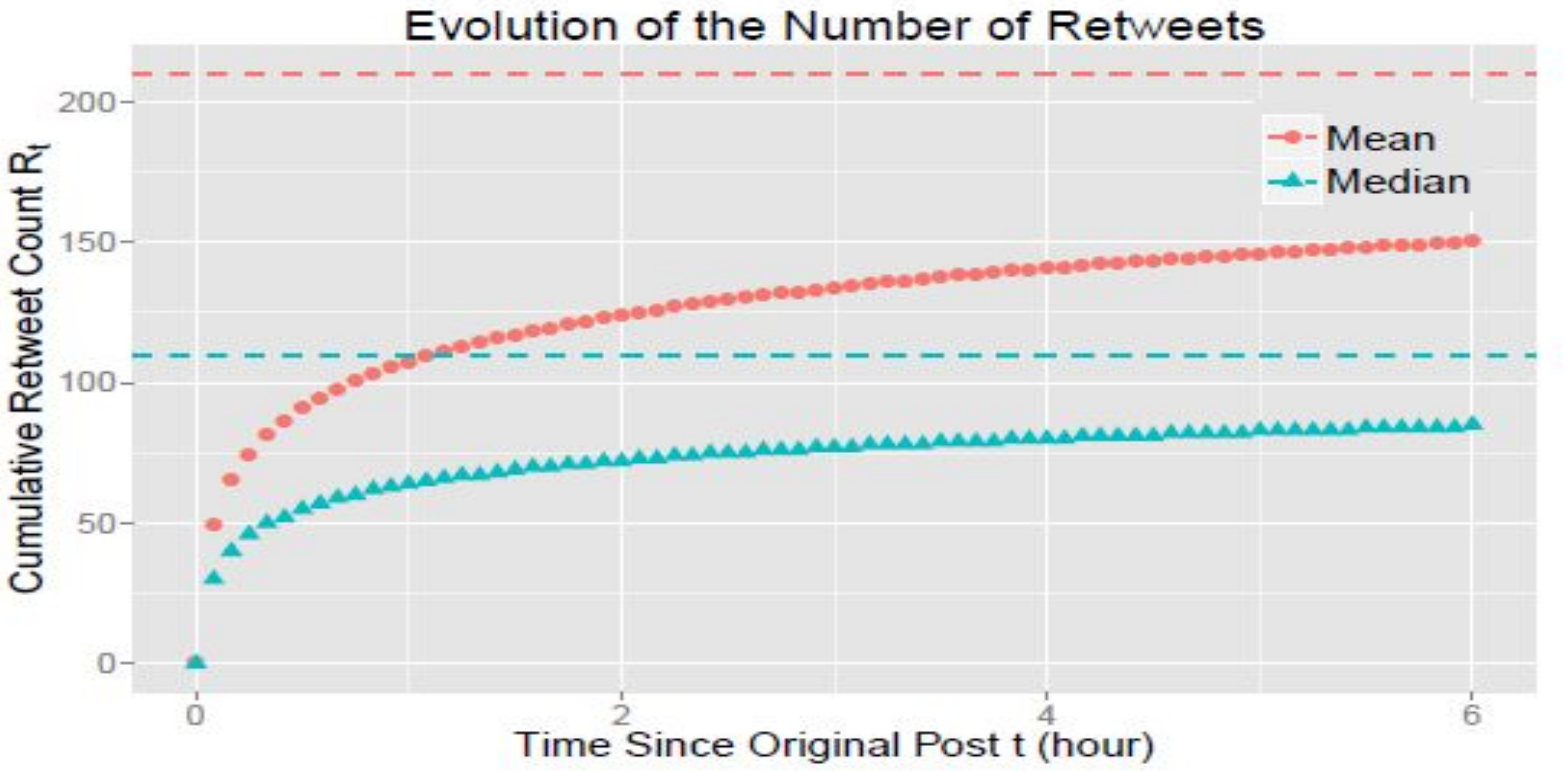


Figure 3: Convergence of the mean and media cumulative retweet count R_t as a function of time.The horizontal lines correspond to mean and median final retweet count $R_{14\,\text{days}}$. On average, a tweet receives 75% of its retweets in the first 6 hours.

15 tweets in the training set are choosen to use the distribution of all their retweet times as memory kernel. All the original posters have an overwhelming number of followers.

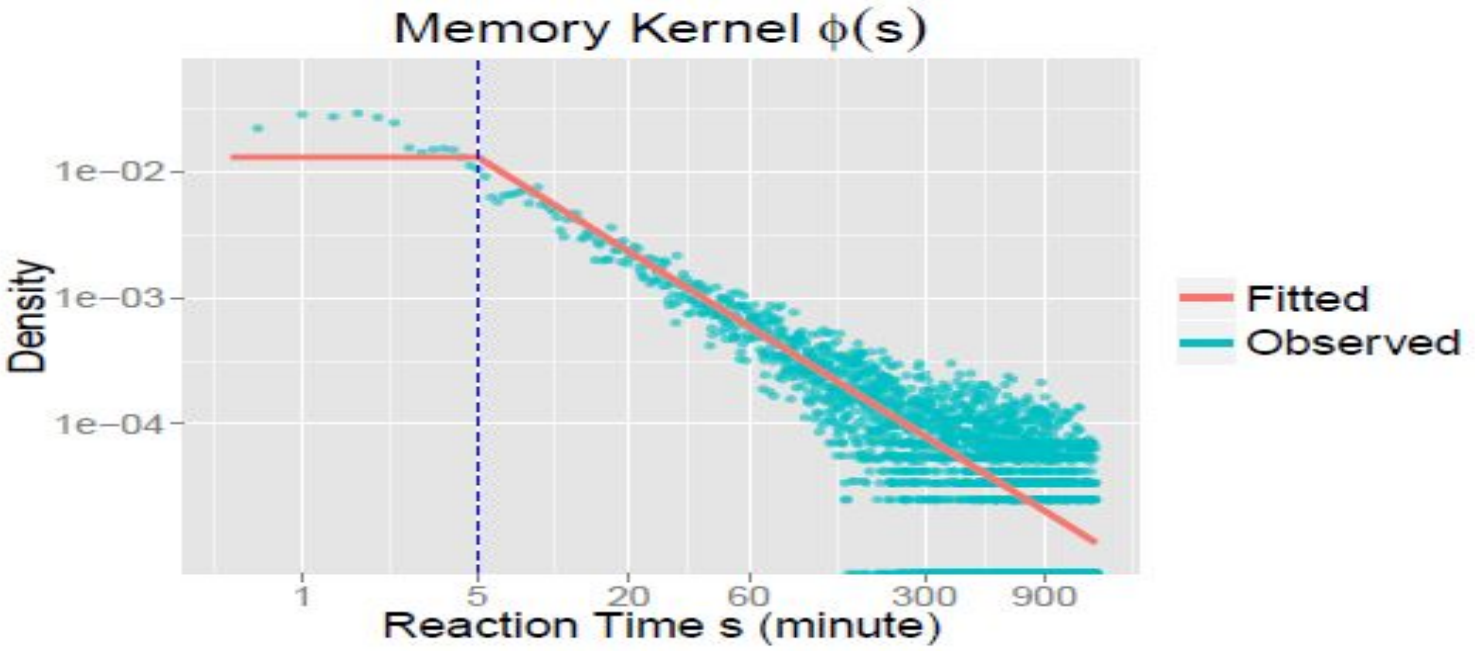


Figure 4: Reaction time distribution and the estimated memory kernel $\phi(s)$. The reaction time is plotted on logarithmic axes, hence the linear trend suggests a power law decay.

$$\phi(s) = \begin{cases} c & \text{if } 0 < s \leq s_0, \\ c(s/s_0)^{-(1+\theta)} & \text{if } s > s_0. \end{cases}$$

$$\theta = 0.2314843\,, s_0 = 300, c = 0.0006265725$$

$$\gamma_t n_* = 20$$

time (minute)	5	10	15	20	30
α	0.389	0.803	0.772	0.709	0.680
time (minute)	60	120	180	240	360
α	0.562	0.454	0.378	0.352	0.326

Table 2: Values α_t used in Algorithm 1.

➢Baseline for comparision

·Linear regression(LR)

$$\log R_\infty = \alpha_t + \log R_t + \epsilon,$$

·Linear regression with degree(LR-D)

$$\log R_\infty = \alpha_t + \beta_{1,t} \log R_t + \beta_{2,t} \log N_t + \beta_{3,t} \log n_0 + \epsilon$$

·Dynamic Poisson Model(DPM)

$$\lambda_t = \lambda_{t_{\text{peak}}} (t - t_{\text{peak}})^\gamma \quad t_{\text{peak}} = \arg\max_{s<t} \lambda_s.$$

·Reinforced Poisson Model(RPM)

$$\lambda_t = c f_\gamma(t) r_\alpha(R_t) \qquad f_\gamma(t) \propto t^{-\gamma} (\gamma > 0)$$

- Evaluation metrics
  - ·Absolute Percentage Error(APE)

$$\mathrm{APE}(w,t) = \frac{|\hat{R}_\infty(w,t) - R_\infty(w)|}{R_\infty(w)}.$$

  - ·Kendall-tau Rank Correlation

$$\tau = \frac{2P}{\frac{1}{2}n(n-1)} - 1 = \frac{4P}{n(n-1)} - 1$$

  - ·Breakout Tweet Coverage

- example of Kendall-tau Rank Correlation

1 2 3 4 5 6 7 8

3 4 1 2 5 7 8 6

P = 5 + 4 + 5 + 4 + 3 + 1 + 0 + 0 = 22

$$\tau = \frac{88}{56} - 1 = \frac{44}{28} - 1 = 0.57$$

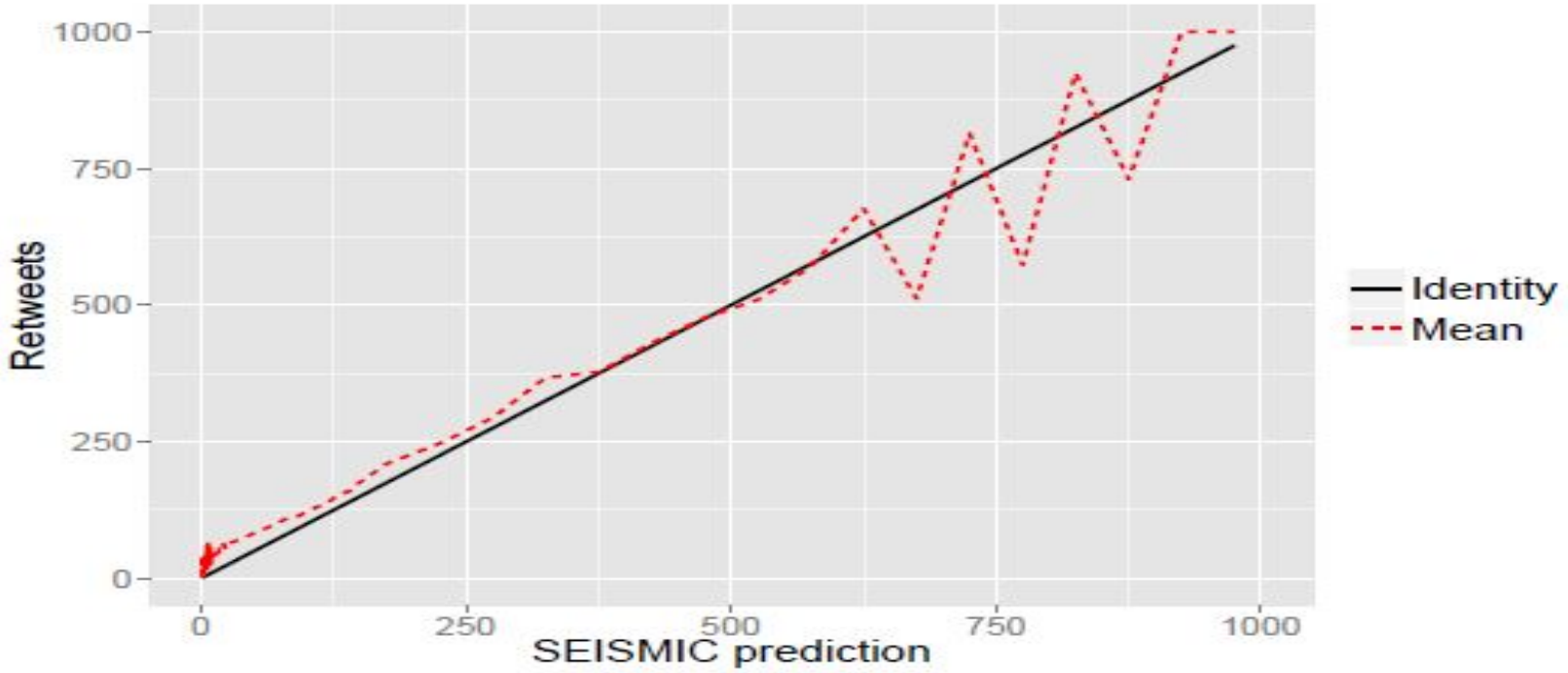


Figure 5: Predicted final retweet counts nicely follow the ground-truth retweet counts, which suggests SEISMIC provides an unbiased estimate of the final retweet count. The dashed red curve is obtained by binning the tweets according to the prediction and then computing the average number of retweets in each bin.

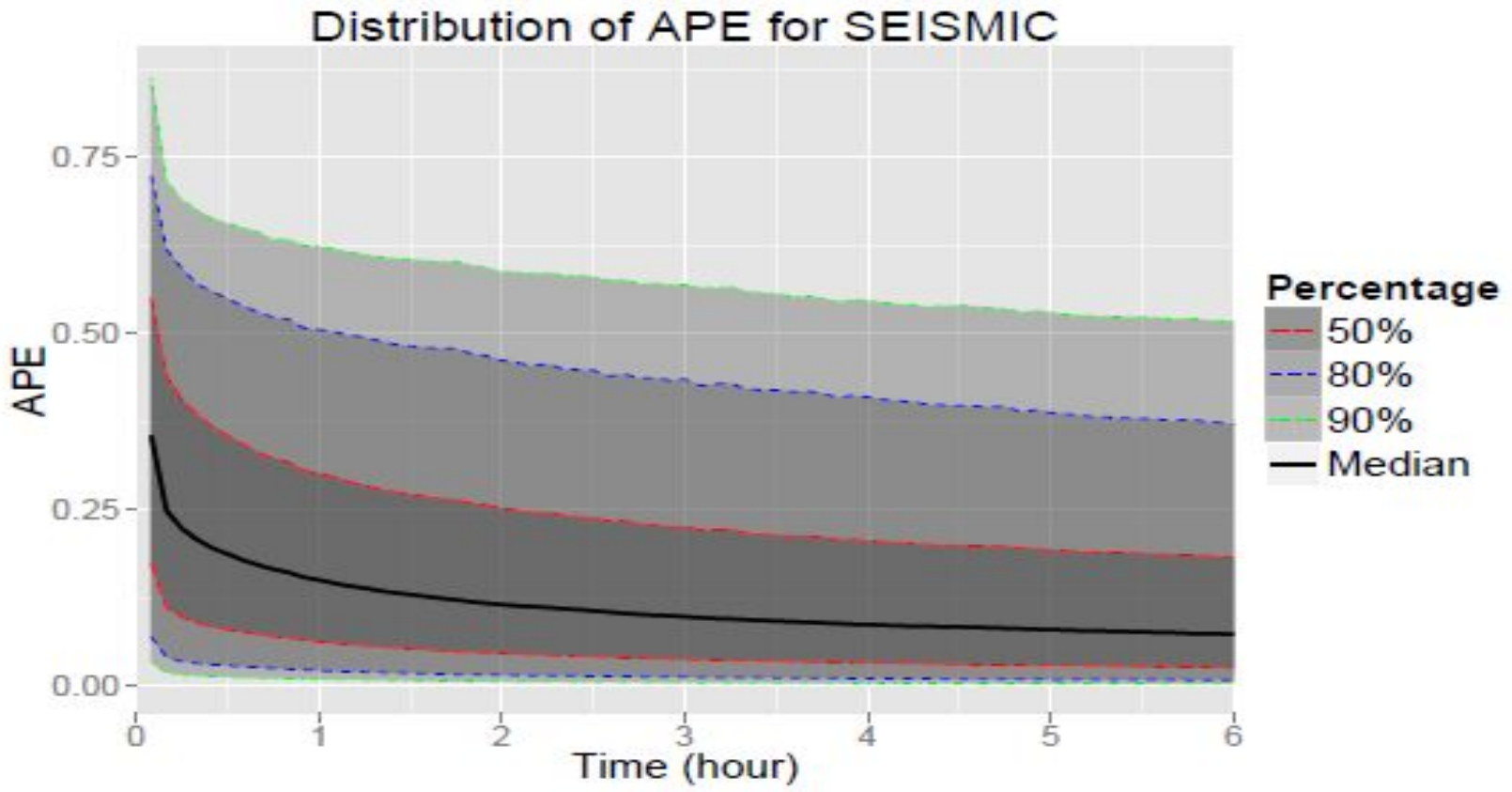


Figure 6: Absolute Percentage Error (APE) of SEISMIC on the test set. We plot the median and the middle 50th, 80th, 90th percentiles of the distribution of APE across the tweets.

- identifying outbreaks

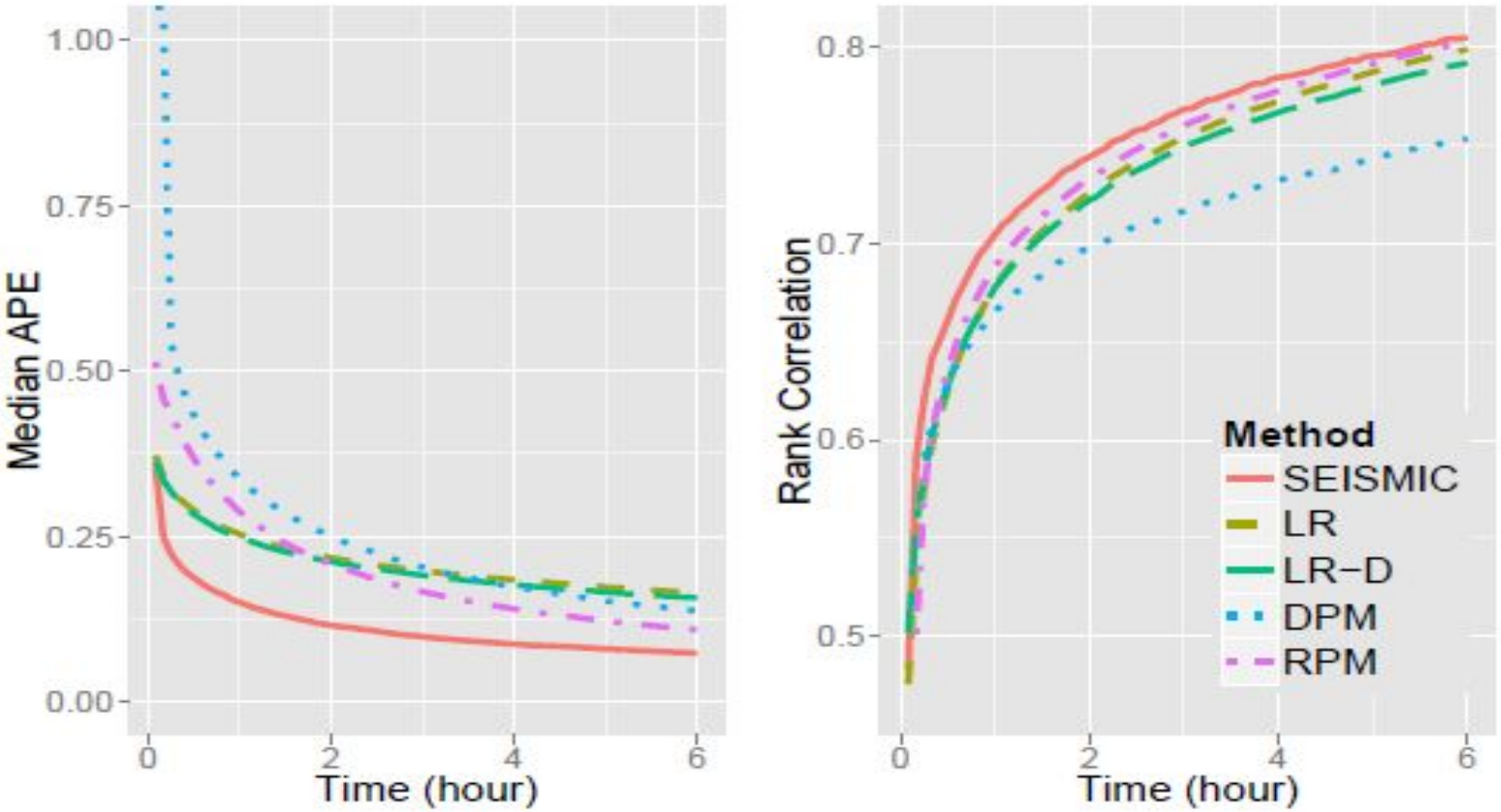


Figure 7: Median Absolute Percentage Error (APE) and Kendall's Rank Correlation of SEISMIC and the baselines as a function of time. SEISMIC consistently gives best performance.

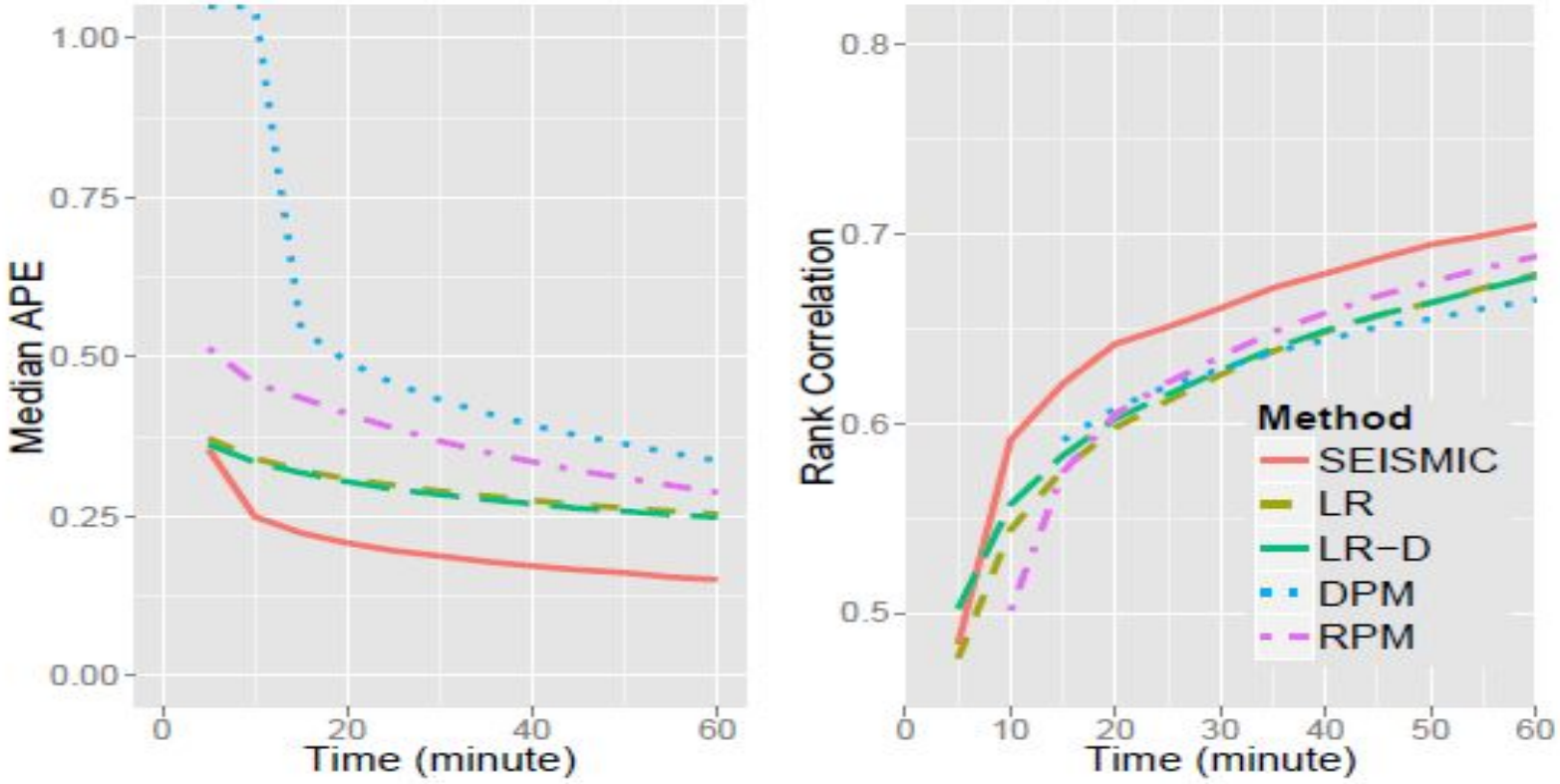


Figure 8: Zoom-in of Figure 7: Median APE and Rank Correlation for the first 60 minutes after the tweet was posted. SEISMIC performs especially well compared to the baselines early in the tweet's lifetime.

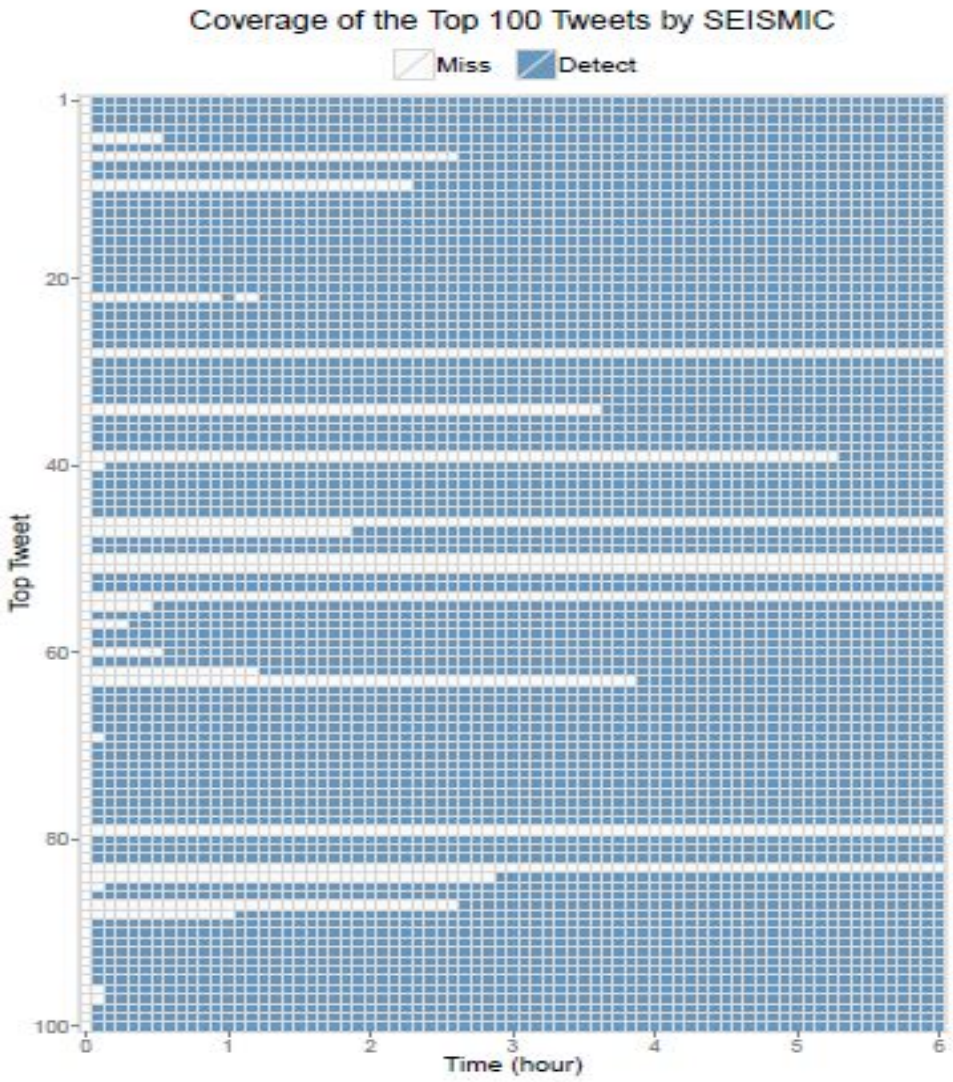


Figure 9: Coverage of top 100 most retweeted tweets. Each row represents a tweet. White blocks indicate that a given tweet was not covered by SEISMIC's predicted list of top-500 tweets at time t, and blue indicates successful coverage.

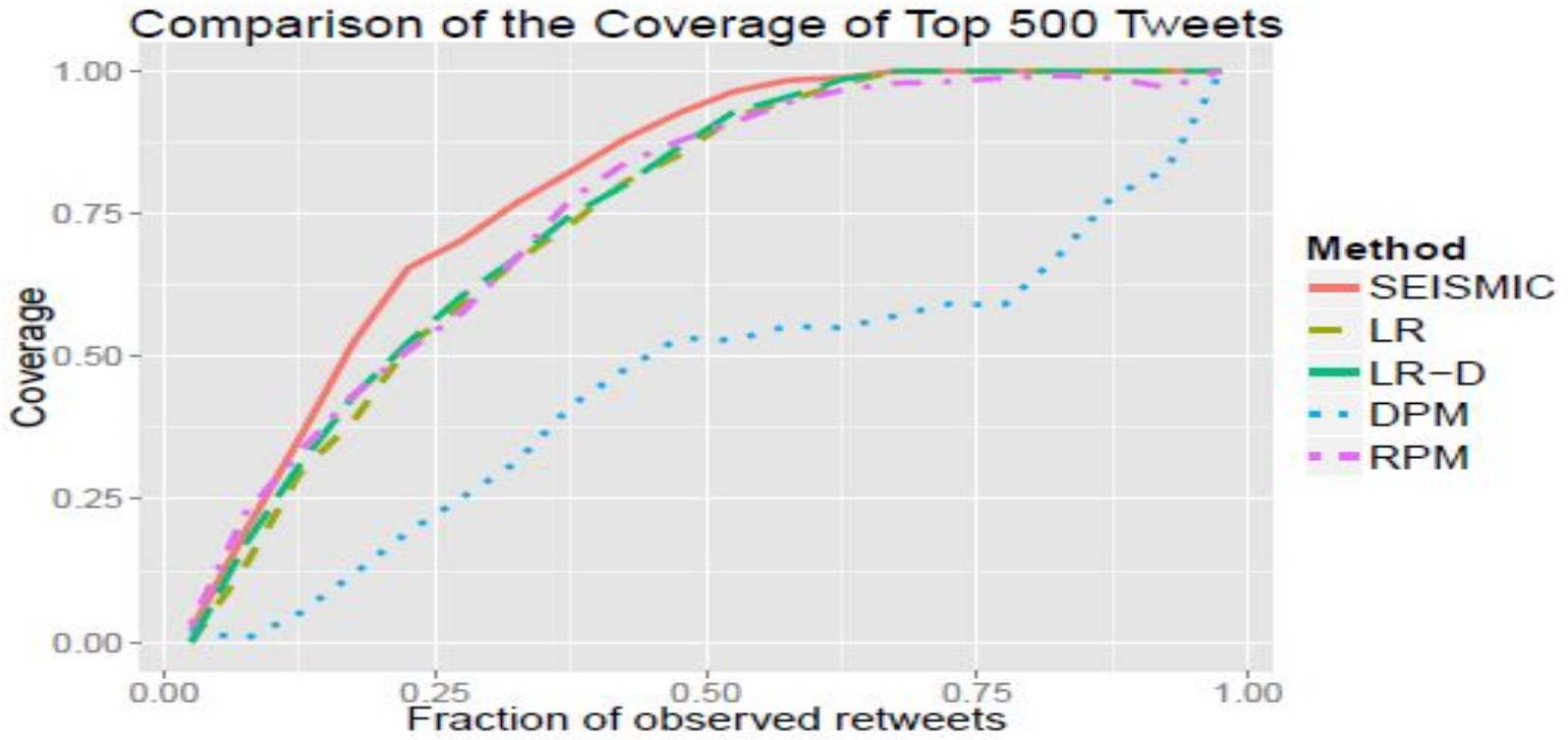


Figure 10: Coverage of top 500 tweets (L^*_{500}) by various methods. SEISMIC exhibits clear improvement over all methods after about 10% of retweets are observed. All methods except DPM achieve perfect coverage after 65% of retweets are observed.

- Future work:

  *if network structure is available, newly exposed followers replace n_i

  *if features is available, we can develop a feature-based prior of p_t

  *if user timezone is available, they can help modify the p_t

Thanks

Ruiqi Yang