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

Qingyuan Zhao\textsuperscript{1}, Murat A. Erdogdu\textsuperscript{1}, Hera Y. He\textsuperscript{1}, Anand Rajaraman\textsuperscript{2}, Jure Leskovec\textsuperscript{2}

Department of Statistics\textsuperscript{1} and Computer Science\textsuperscript{2}, Stanford University

KDD’15, Aug 12, 2015
Information cascade

An information cascade occurs when people engage in the same actions.

Source: wikimedia.org

Source: adweek.com
Twitter provides the ideal playground to study information cascades.

- **Start:** a Twitter user posts a 140-character message which can be seen by his/her followers.
- **Spread:** a tweet is forwarded in Twitter by another user.
Predicting cascades in real time

Goal

Given the tweet and retweets up to time $T$, predict its final popularity.
Predicting cascades in real time

**Goal**

Given the tweet and retweets up to time $T$, predict its final popularity.

**Applications**

- Ranking content.
- Detecting viral/breakout tweets.
- Understanding human social behavior.
Mathematical definitions

Data

- Relative retweet time $t_0 = 0, t_1, t_2, \ldots$
  - Number of retweets by time $t$: $R_t = \sum_{t_i \leq t} 1$. 
- Number of followers of each retweeter $n_0, n_1, n_2, \ldots$
  - Number of exposed users by time $t$: $N_t = \sum_{t_i \leq t} n_i$. 

Problem statement

Given $(R_t, N_t)$ for $0 \leq t \leq T$, predict $R_\infty$. 

Mathematical definitions

Data

- Relative retweet time $t_0 = 0, t_1, t_2, \ldots$
  - Number of retweets by time $t$: $R_t = \sum_{t_i \leq t} 1$.
- Number of followers of each retweeter $n_0, n_1, n_2, \ldots$
- Number of exposed users by time $t$: $N_t = \sum_{t_i \leq t} n_i$.

Problem statement

Given $(R_t, N_t)$ for $0 \leq t \leq T$, predict $R_\infty$. 
Approaches to cascade prediction

Broadly categorized into two groups:

- **Feature based methods (the majority):**
  - Feature engineering: temporal, network structure, content, user, . . .
  - Supervised learning: linear regression, collaborative filtering, regression trees, topic modeling, . . .

- **Point process based methods:**
  - Dynamic Poisson process, reinforced Poisson process
  - Our model (SEISMIC): self-exciting point process.
Approaches to cascade prediction

Broadly categorized into two groups:

- **Feature based methods (the majority):**
  - Feature engineering: temporal, network structure, content, user, ...

- **Point process based methods:**
Approaches to cascade prediction

Broadly categorized into two groups:

- **Feature based methods (the majority):**
  - Feature engineering: temporal, network structure, content, user, . . .
  - Supervised learning: linear regression, collaborative filtering, regression trees, topic modeling, . . .

- **Point process based methods:**
Approaches to cascade prediction

Broadly categorized into two groups:

- **Feature based methods (the majority):**
  - Feature engineering: temporal, network structure, content, user, . . .
  - Supervised learning: linear regression, collaborative filtering, regression trees, topic modeling, . . .

- **Point process based methods:**
  - Dynamic Poisson process, reinforced Poisson process
Approaches to cascade prediction

Broadly categorized into two groups:

- **Feature based methods (the majority):**
  - Feature engineering: temporal, network structure, content, user, . . .
  - Supervised learning: linear regression, collaborative filtering, regression trees, topic modeling, . . .

- **Point process based methods:**
  - Dynamic Poisson process, reinforced Poisson process
  - Our model (SEISMIC): self-exciting point process.
Example

Matt Bellamy
@mottbollomy

- Saddam Hussein
- Osama Bin Laden
- Col. Gaddafi
- Justin Bieber

5:40 AM - 20 Oct 2011

RETWEETS 16,258  FAVORITES 906
Example
SEISMIC (Self-Exciting Model of Information Cascades) is a flexible model of information cascades.

Highlights

- Generative model.
- Easy interpretation.
- Scalable: prediction takes $O(\# \text{ retweets})$.
- State-of-the-art performance.
Background: point processes

Point process models

\[ R_t \text{ is characterized by its intensity } \lambda_t = \lim_{\Delta \downarrow 0} \frac{\mathbb{P}(R_{t+\Delta} - R_t = 1)}{\Delta}. \]
Background: point processes

**Point process models**

\[ R_t \] is characterized by its intensity \( \lambda_t = \lim_{\Delta \downarrow 0} \frac{\mathbb{P}(R_t + \Delta - R_t = 1)}{\Delta} \).

**Examples**

- Poisson process: \( \lambda_t = \lambda \);
- Reinforced Poisson process\(^1\): \( \lambda_t = p \cdot \phi(t) \cdot g(R_t) \).

\(^1\) S. Gao, J. Ma, and Z. Chen. Modeling and predicting retweeting dynamics on microblogging platforms. In WSDM '15, 2015.
Background: point processes

Point process models

$R_t$ is characterized by its intensity $\lambda_t = \lim_{\Delta \downarrow 0} \frac{P(R_{t+\Delta} - R_t = 1)}{\Delta}$.

Examples

- Poisson process: $\lambda_t = \lambda$;
- Reinforced Poisson process$^1$: $\lambda_t = p \cdot \phi(t) \cdot g(R_t)$.

They are not suitable to model viral tweets.

---

Key steps of retweeting

- How often does a user check Twitter?
- What is the user’s probability of retweeting a given tweet?

\[ \lambda_t = p \cdot \sum_{t_i \leq t} n_i \phi(t-t_i), \quad t \geq t_0. \]
Key steps of retweeting

- How often does a user check Twitter?
  - Memory kernel (power law distribution).
- What is the user’s probability of retweeting a given tweet?
Key steps of retweeting

- How often does a user check Twitter?
  - Memory kernel (power law distribution).
- What is the user’s probability of retweeting a given tweet?
  - Tweet infectiousness.
**Key steps of retweeting**

- How often does a user check Twitter?
  - Memory kernel (power law distribution).
- What is the user’s probability of retweeting a given tweet?
  - Tweet infectiousness.

**Self-exciting point process**

- Infectiousness: “probability” of retweeting
  \[ \lambda_t = p \cdot \sum_{t_i \leq t} n_i \phi(t - t_i), \quad t \geq t_0. \]
- Self-exciting: “rate” of viewing
Time-varying infectiousness

- Fixed $p$ is not enough to model viral tweets.

- SEISMIC replaces $p$ by a smooth process $p_t$. 
We estimate $p_t$ by locally smoothing the maximum likelihood estimator (MLE):

- "Number of retweets"

$$
\hat{p}_t = \frac{\sum_{i=1}^{R_t} K_t(t - t_i)}{\sum_{i=0}^{R_t} n_i \int_{t_i}^{t} K_t(t - s) \phi(s - t_i) ds}.
$$

- "Number of views"
Predict popularity

**SEISMIC prediction formula**

Assume the out-degrees in the network have mean $n_*$ and the infectiousness parameter $p_t \equiv p$ for $t \geq T$. Then

$$
\mathbb{E}[R_\infty | \mathcal{F}_T] = \begin{cases} 
R_T + \frac{p(N_T - N^e_T)}{1 - pn_*}, & \text{if } p < \frac{1}{n_*} \\
\infty, & \text{if } p \geq \frac{1}{n_*}.
\end{cases}
$$

where $N^e_T = \sum_{i=0}^{R_T} n_i \int_{t_i}^T \phi(t - t_i)dt$.

See our paper for derivation.
Example

Histogram of Retweet Times

Prediction by SEISMIC

Time since original tweet (hour)

Retweets

Retweet Count

Final SEISMIC Cumulative
Experiments: dataset

- Raw dataset: all tweet and retweet activities from October 7 to November 7, 2011.
- Filter by:
  - Posted in the first 15 days.
  - English tweets;
  - No hashtag;
  - At least 50 retweets;
- End up with 166076 cascades (in total over 34 million tweets/retweets).
Baselines

We compare SEISMIC to four different baselines:

1. **LR**: linear regression
2. **LR-D**: linear regression with degree
3. **DPM**: dynamic Poisson model
4. **RPS**: reinforced Poisson model
Comparison: Absolute Percentage Error (APE)

\[ APE = \left| \hat{R}_\infty - R_\infty \right| / R_\infty. \]

15% vs 25% percentage error when observe 1 hour.
Comparison: Coverage of breakouts

- A list of **true top 500 tweets** with most retweets.
- Lists of **predicted top 500 tweets** at all time points.

70% vs 55% coverage when observe 25% retweets.
In conclusion, SEISMIC

- Effectively models information cascades by self-exciting point processes;
- Efficiently updates parameters and makes prediction;
- Outperforms several baselines and state-of-the-art.

Code and data available online at http://snap.stanford.edu/seismic.
Estimation of memory kernel $\phi(t)$

Figure 4: Plot of observed reaction time distribution and estimated memory kernel $\phi(s)$. The reaction time is plotted on a log scale, hence a linear trend in the plot suggests a power law decay in the distribution.
More detail: final tweak

- The prediction is unstable when $\hat{p}_t$ is close to $\frac{1}{n^*}$.
- The real $p_s$ is likely to decrease.

Stabilized prediction

$$\hat{R}_\infty(t) = R_t + \alpha_t \frac{\hat{p}_t (N_t - N^e_t)}{1 - \gamma_t \hat{p}_t n^*}$$

where $0 < \alpha_t, \gamma_t \leq 1$ are trained for the network.