Scalable Influence Estimation in Continuous-Time Diffusion Networks

Nan Du [1] Le Song [1] Manuel Gomez Rodriguez[2] Hongyuan Zha[1]

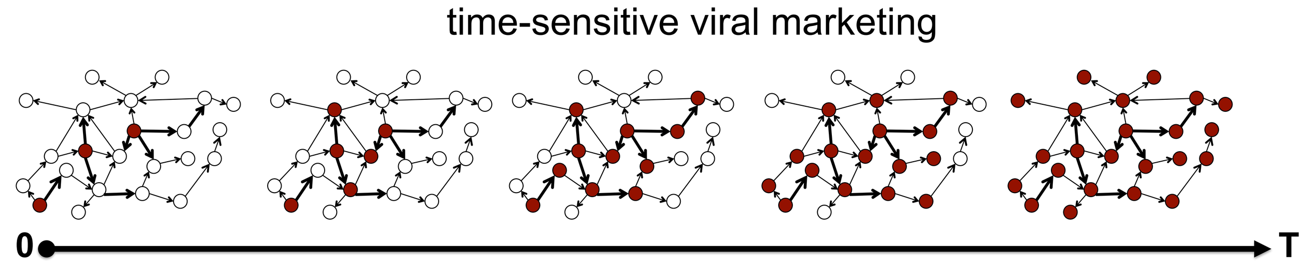
[1]Georgia Institute of Technology, [2]Max Planck Institute for Intelligent Systems

Georgia Institute of Technology

Motivation

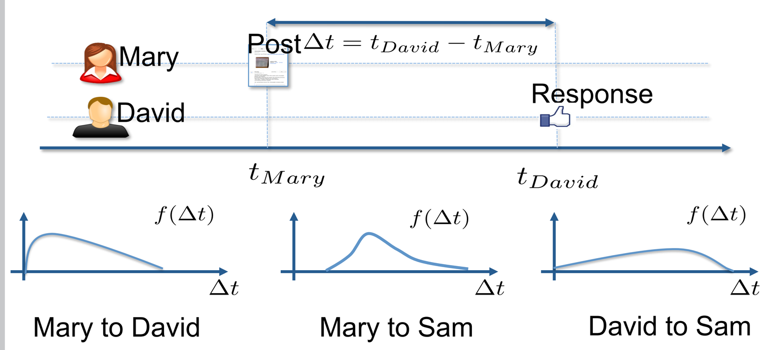
- Question : How can we optimize the selection of the earlier nodes to trigger, *within a time window T*, the largest expected number of follow-ups ?

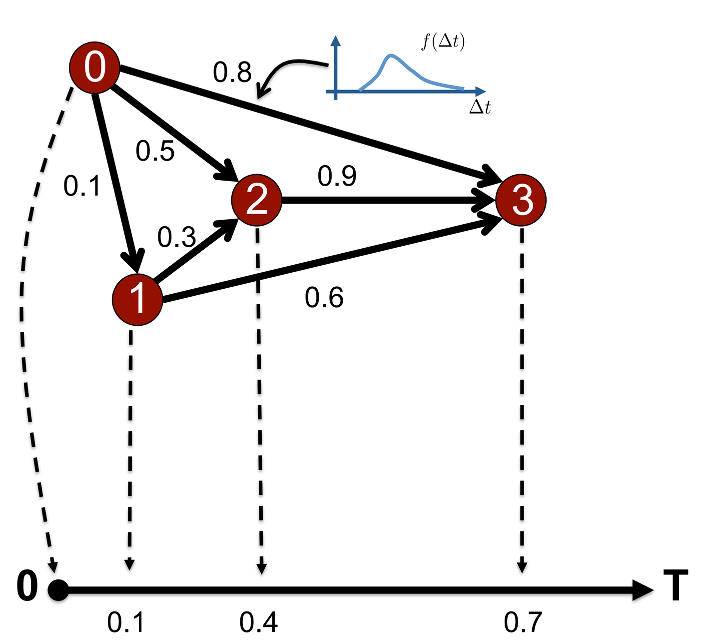
time-sensitive viral marketing



Continuous-time Independent Cascade Model

- Infection : an event occurs to a node, e.g., adopting a product.
- Pairwise conditional density $f_{ji}(t_j|t_i) = f_{ji}(t_i - t_j)$ over time





Absolute Infection Time Perspective

- For each node i, let random variable t_i represent the infection time.
- The influence value of sources $\mathcal{A}$ by time T is

$$\sigma(\mathcal{A}, T) = \mathbb{E}\left[\sum_{i\in\mathcal{V}} \mathbb{I}\{t_i \le T\}\right] = \sum_{i\in\mathcal{V}} \Pr\{t_i \le T\}.$$

- Directed graphical model representation

$$p(\{t_i\}_{i\in\mathcal{V}}) = \prod_{i\in\mathcal{V}} p(t_i|\{t_j\}_{j\in\pi_i}),\quad \pi_i \text{ is the set of parents.}$$

- Marginalization

$$\Pr\{t_i \le T\} = \int_0^\infty \cdots \int_{t_i=0}^T \cdots \int_0^\infty \left(\prod_{j\in\mathcal{V}} p(t_j|\{t_l\}_{l\in\pi_j})\right)\left(\prod_{j\in\mathcal{V}} dt_j\right).$$

Inter-event Time Perspective

- Aim to calculate $\mathbb{E}\left[\sum_{i\in\mathcal{V}} \mathbb{I}\{t_i \le T\}\right]$ directly.
- Mutually independent transmission times $\tau_{ji} = t_i - t_j$.

$$p\left(\{\tau_{ji}\}_{(j,i)\in\mathcal{E}}\right) = \prod_{(j,i)\in\mathcal{E}} f_{ji}(\tau_{ji}).$$

- A set of transmission times (or a particular configuration).

$$G := \{\tau_{ji}\}_{(j,i)\in\mathcal{E}} \backsim p\left(\{\tau_{ji}\}_{(j,i)\in\mathcal{E}}\right).$$

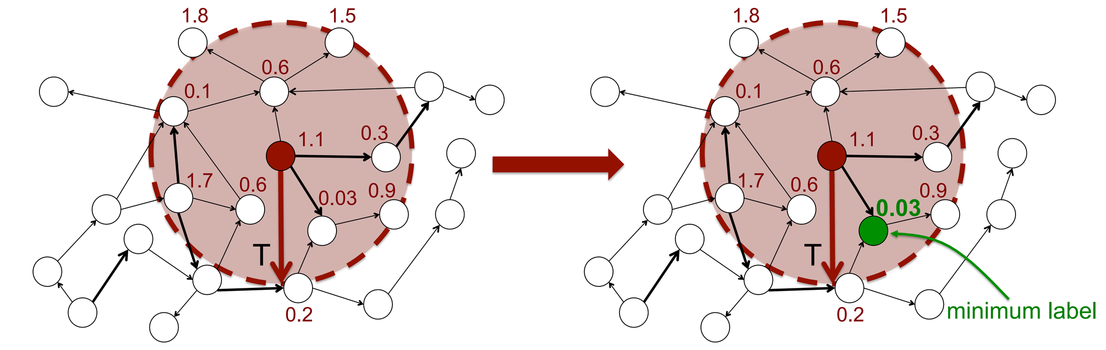
- Given G, t_i is the length of shortest path from all sources in $\mathcal{A}$ to i.
- Draw a set of $G^\ell \backsim p\left(\{\tau_{ji}\}_{(j,i)\in\mathcal{E}}\right)$

$$\Pr\{t_i \le T\} = \Pr\{\text{length of shortest path from } \mathcal{A} \to i \le T\}.$$

- Naive simulation requires $O(|\mathcal{V}|^2)$ time complexity.

Neighborhood Size Estimation

Given n i.i.d random variable $X^i \backsim e^{-x}$, the minimum $X_* \backsim ne^{-nx}$.



assign exponential random label r_i — find the minimum label r_*

- Find m such least labels, $\{r_*^u\}_{u=1}^m$ to estimate $|\mathcal{N}(\{j\}, T)| \approx \frac{m-1}{\sum_{u=1}^m r_*^u}$, convert counting problem to estimation problem !

Neighborhood Size Estimation

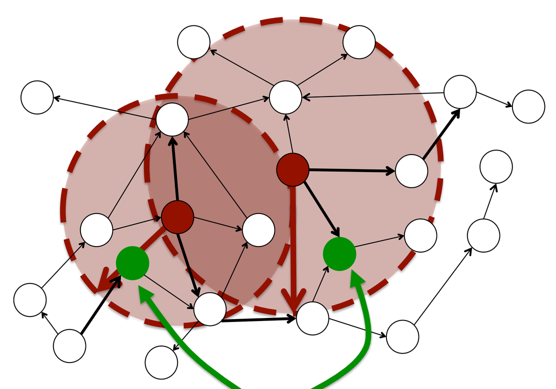
- Multiple sources $\mathcal{A}$

$$\mathcal{N}(\mathcal{A}, T) = \bigcup_{s\in\mathcal{A}} \mathcal{N}(s, T).$$

- Reuse least-label list for each single source $s \in \mathcal{A}$

$$r_* = \min_{i\in\mathcal{A}} \min_{j\in\mathcal{N}(i,T)} r_j$$

- Cohen's algorithm produces the lists for all nodes in time $\tilde{O}(|\mathcal{E}|)$.



minimum of the two least labels

Overall Algorithm ConTinEst

Influence function

$$\sigma(\mathcal{A}, T) = \mathbb{E}_{\{\tau_{ji}\}_{(j,i)\in\mathcal{E}}}\left[|\mathcal{N}(\mathcal{A}, T)|\right] = \mathbb{E}_{\{\tau_{ji}\}}\mathbb{E}_{\{r^1,\ldots,r^m\}|\{\tau_{ji}\}}\left[\frac{m-1}{\sum_{u=1}^m r_*^u}\right].$$

1. Sample n sets of random transmission times $\{\tau_{ij}^l\}_{(j,i)\in\mathcal{E}} \sim \prod_{(j,i)\in\mathcal{E}} f_{ji}(\tau_{ji})$
2. Given $\{\tau_{ij}^l\}_{(j,i)\in\mathcal{E}}$, sample m random labels $\{r_i^u\}_{i\in\mathcal{V}} \sim \prod_{i\in\mathcal{V}} \exp(-r_i)$.
3. Estimate $\sigma(\mathcal{A}, T)$ by $\hat{\sigma}(\mathcal{A}, T) \approx \frac{1}{n}\sum_{l=1}^n \left((m-1)/\sum_{u_l=1}^m r_*^{u_l}\right)$.

Overall Algorithm ConTinEst

Theorem : Draw the following number of sets of random transmission times

$$n \geqslant \frac{C\Lambda}{\epsilon^2}\log\left(\frac{2|\mathcal{V}|}{\delta}\right),$$

where Λ depends on $\mathcal{A}$ and T, and for each set of random transmission times, draw m sets of random labels. Then $|\hat{\sigma}(\mathcal{A}, T) - \sigma(\mathcal{A}, T)| \leqslant \epsilon$ uniformly for all $\mathcal{A}$ with $|\mathcal{A}| \leqslant C$, with probability at least $1 - \delta$.

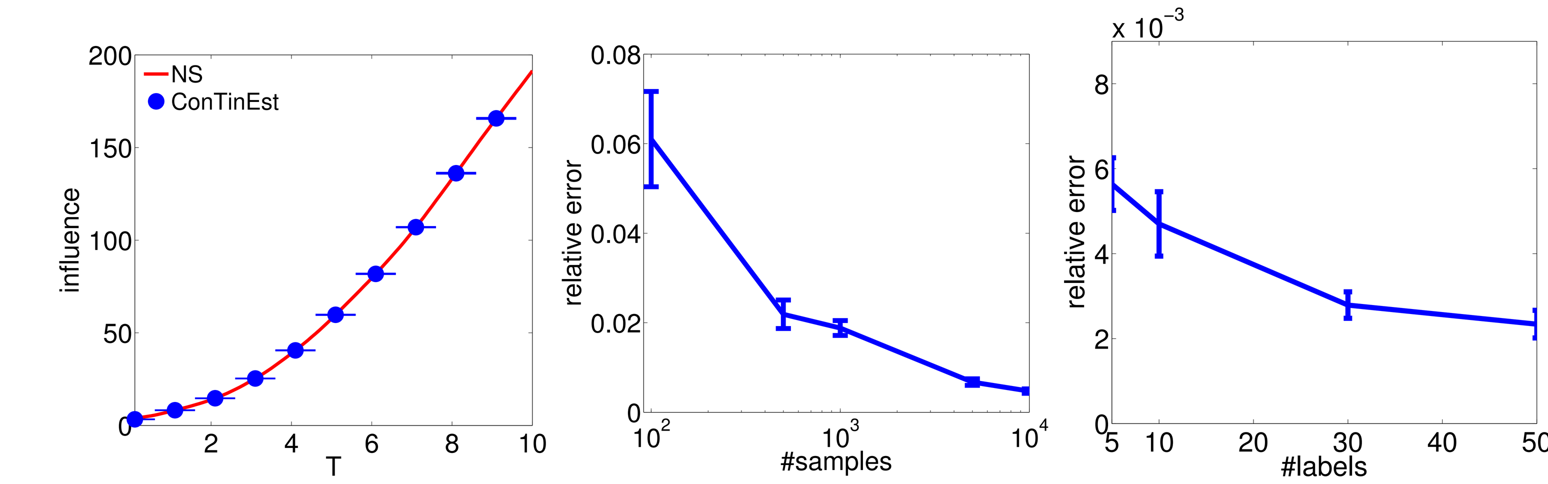
- Influence of larger source set $\mathcal{A}$ at the longer time window T requires more samples in the worst case.
- In practice : small $m = 5$ achieves good performance.

Influence Maximization ConTinEst

- Solve $\mathcal{A}^* = \mathrm{argmax}_{|\mathcal{A}|\leqslant C}\ \sigma(\mathcal{A}, T)$, which is NP-hard in general.
- $\sigma(\mathcal{A}, T)$ is a non-negative, monotonic, submodular function.
- Greedy algorithm achieves $(1 - 1/e)$ of the optimal value (OPT).
- The estimator $\hat{\sigma}(\mathcal{A}, T)$ induces some error ϵ.
- **Theorem** : Suppose the influence $\sigma(\mathcal{A}, T)$ for all $\mathcal{A}$ with $|\mathcal{A}| \le C$ are estimated uniformly with error ϵ and confidence $1 - \delta$, the greedy algorithm returns a set of sources $\hat{\mathcal{A}}$ such that $\sigma(\hat{\mathcal{A}}, T) \ge (1 - 1/e)OPT - 2C\epsilon$ with probability at least $1 - \delta$.

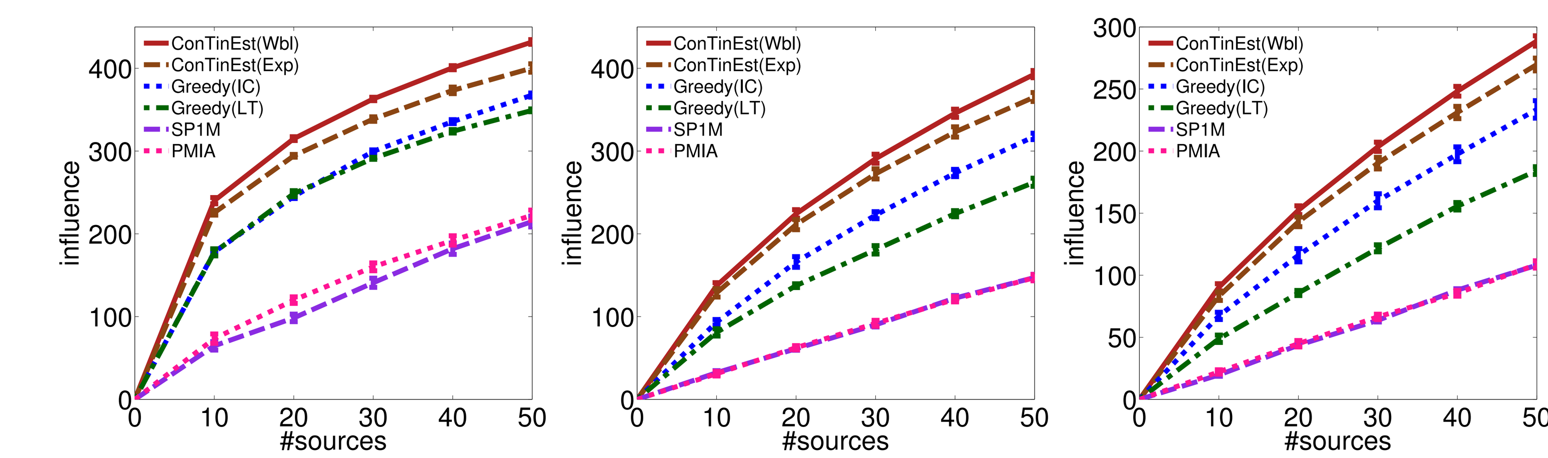
Experimental Evaluation : Synthetic Dataset

- Accuracy of the estimated influence (highest out-degree node).



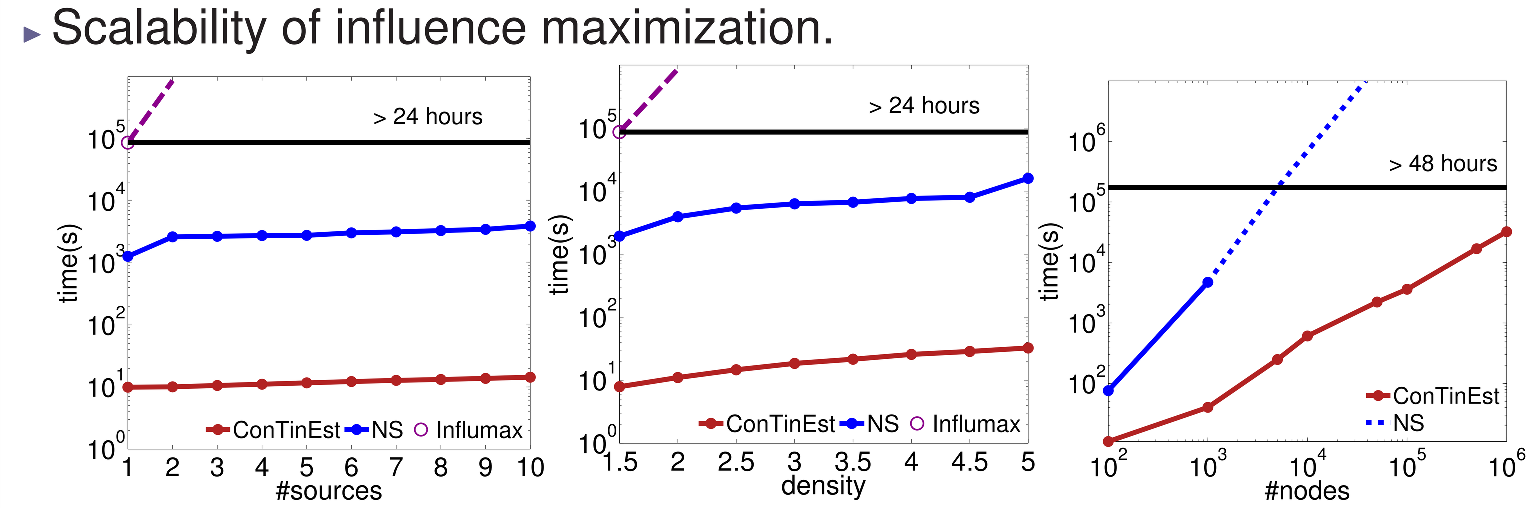
(a) Influence vs. time (b) Error vs. #samples (c) Error vs. #labels

- Quality of the selected nodes for influence maximization.



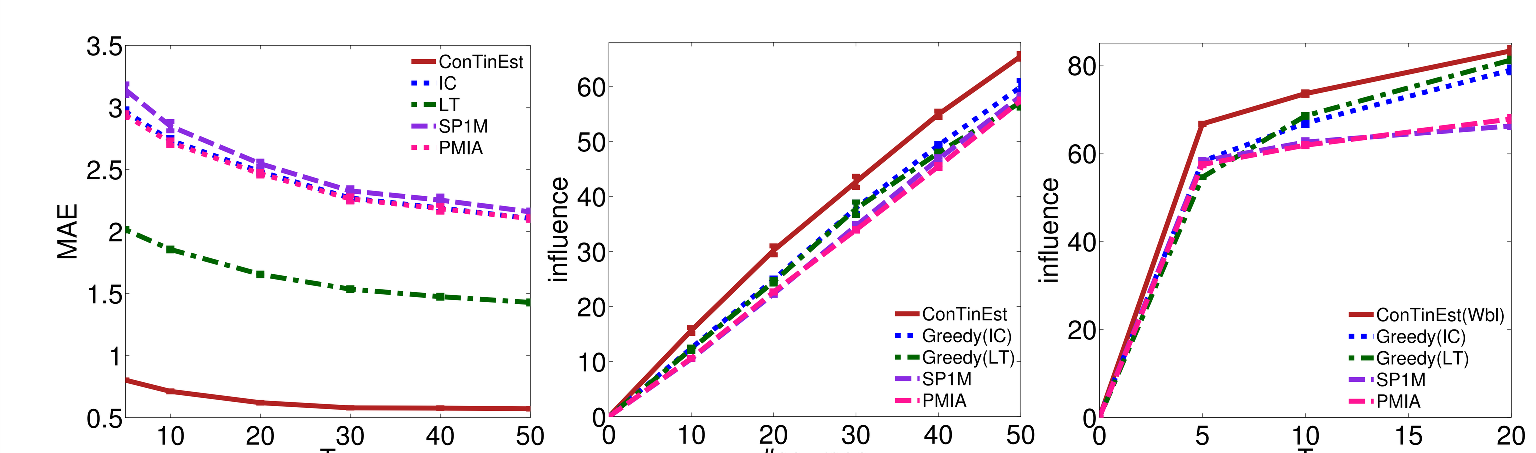
(a) Core-periphery (b) Random (c) Hierarchical

- Scalability of influence maximization.



(a) # sources (b) network density (c) network size

Experimental Evaluation : Real Dataset



(a) Estimation error (b) #sources (c) time

- 10,967 hyperlink cascades randomly splits into 80%-training and 20%-testing data.
- Infer network structures based on the training data.
- Evaluate estimated influence on the testing data.
  - Given node i, $\mathcal{C}(i)$ be the set of all cascades where i is the source.
  - Based on $\mathcal{C}(i)$, the total number of distinct nodes infected before T quantifies the real influence of node u up to time T.
  - Average across different cascades is the true influence.
- Repeat experiments for 10 times.