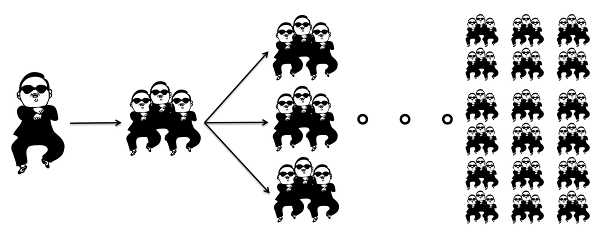
INFLUENCE FUNCTION LEARNING IN INFORMATION DIFFUSION NETWORKS

NAN DU   YINGYU LIANG   MARIA-FLORINA BALCAN   LE SONG
GEORGIA INSTITUTE OF TECHNOLOGY

Georgia Institute of Technology

MOTIVATION

- Problem : Given a set of influential earlier users, can we predict how many people will follow them in the future?



How many people will be influenced in the future ?

- Challenges
  - Latent social network structures
  - Unknown diffusion mechanism
  - Observing only temporal traces of information diffusion

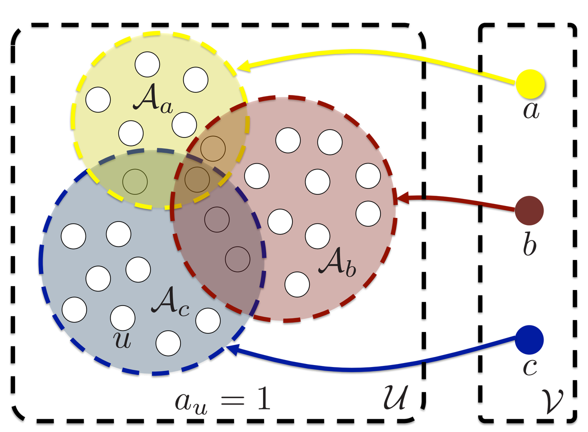
PREVIOUS TWO-STAGE SOLUTIONS

- Algorithm
  - Learn one of the following **diffusion models**
    - Discrete-Time independent cascade Model (DIC)
    - Linear Threshold Model (LT)
    - Continuous-Time independent cascade Model (CIC)
  - Calculate the influence from the chosen model
- Weakness
  - The diffusion model may be misspecified.
  - Need to learn both hidden networks and model parameters.
  - Influence calculation is challenging.

Can we avoid diffusion model learning & influence computation?

INFLUENCE FUNCTION

- Definition : $\sigma(\mathcal{S}) : 2^{\mathcal{V}} \mapsto \mathbb{R}_+$ of a set of nodes $\mathcal{S} \subseteq \mathcal{V}, |\mathcal{V}| = d$
  - $\sigma(\mathcal{S})$ is the expected number of infected nodes by set $\mathcal{S}$.
  - $\sigma(\mathcal{S})$ is common to many diffusion models.
- Property : $\sigma(\mathcal{S})$ is a **coverage function** for DIC, LT and CIC model
  - $\sigma(\mathcal{S}) = \sum_{u \in \bigcup_{s \in \mathcal{S}} \mathcal{A}_s} a_u$
  - a ground set $\mathcal{U}$ with weight $a_u \geqslant 0,\ u \in \mathcal{U}$
  - a collection of subsets $\{\mathcal{A}_s : \mathcal{A}_s \subseteq \mathcal{U}\}$ associated with each $s \in \mathcal{V}$



RANDOM REACHABILITY FUNCTION

- View the diffusion process as a node reachability problem in a random graph $\mathcal{G}$ sampled from a joint distribution induced by a diffusion model.
- Represent each sample $\mathcal{G}$ as a binary reachability matrix with

$$\boldsymbol{R}_{sj} = \begin{cases} 1, & j \text{ is reachable from source } s, \\ 0, & \text{otherwise.} \end{cases}$$

- Denote each set $\mathcal{S}$ as a binary vector $\chi_{\mathcal{S}} \in \{0,1\}^d$, $\chi_{\mathcal{S}}(s) = 1, s \in \mathcal{S}$
- Determine the reachability of node j from $\mathcal{S}$ by whether $\chi_{\mathcal{S}}^\top \boldsymbol{R}_{:j} \geq 1$
- Transform $\chi_{\mathcal{S}}^\top \boldsymbol{R}_{:j}$ into a binary function $\phi\left(\chi_{\mathcal{S}}^\top \boldsymbol{R}_{:j}\right) : 2^{\mathcal{V}} \mapsto \{0,1\}$, where $\phi(u) = \min\{u, 1\} : \mathbb{Z}_+ \mapsto \{0,1\}$ is a concave function
- Derive the influence of $\mathcal{S}$ in $\mathcal{G}$ as

$$\#(\mathcal{S}|\boldsymbol{R}) := \sum_{j=1}^{d} \phi\left(\chi_{\mathcal{S}}^\top \boldsymbol{R}_{:j}\right).$$

EXPECTATION OF RANDOM RECHABILITY FUNCTIONS

- Overall influence function

$$\mathbb{E}_{\boldsymbol{R} \sim p_{\boldsymbol{R}}}[\#(\mathcal{S}|\boldsymbol{R})] = \sum_{j=1}^{d} \mathbb{E}_{\boldsymbol{R} \sim p_{\boldsymbol{R}}}\left[\phi\left(\chi_{\mathcal{S}}^\top \boldsymbol{R}_{:j}\right)\right] = \sum_{j=1}^{d} \underbrace{\Pr\left\{\phi\left(\chi_{\mathcal{S}}^\top \boldsymbol{R}_{:j}\right) = 1 | \chi_{\mathcal{S}}\right\}}_{:= f_j(\chi_{\mathcal{S}})}$$

- Simple Learning Strategy
  Learn each $f_j(\chi_{\mathcal{S}})$ separately in parallel and sum them together.

RANDOM BASIS FUNCTION APPROXIMATION

- Denote $f_j(\chi_{\mathcal{S}}) = \mathbb{E}_{r \sim p_j(r)}\left[\phi(\chi_{\mathcal{S}}^\top r)\right]$ where $r := \boldsymbol{R}_{:j}$, and $p_j(r)$ is the marginal distribution of column j of $\boldsymbol{R}$ induced by $p_{\boldsymbol{R}}$.
- Let C be the minimum value such that $p_j(r) \leq C q_j(r)$.
- Draw K random binary vectors $\{r_1, r_2, \ldots, r_K\}$ from $q(r)$ such that

$$f^w(\chi_{\mathcal{S}}) = \sum_{k=1}^{K} w_k \phi(\chi_{\mathcal{S}}^\top r_k) = w^\top \phi(\chi_{\mathcal{S}}) \text{ subject to } \sum_k w_k = 1, w_k \geqslant 0$$

Lemma

Let $p_\chi(\chi_{\mathcal{S}})$ be a distribution of $\chi_{\mathcal{S}}$. If $K = O(\frac{C^2}{\epsilon^2}\log\frac{C}{\epsilon\delta})$ and $r_1, \ldots, r_K$ are drawn i.i.d. from $q_j(r)$, then with probability at least $1-\delta$, there exists an $f^w \in \widehat{\mathcal{F}}^w$ such that $\mathbb{E}_{\chi_{\mathcal{S}} \sim p_\chi}[(f_j(\chi_{\mathcal{S}}) - f^w(\chi_{\mathcal{S}}))^2] \leq \epsilon^2$.

- Propose $q_j(r) = \prod_{s=1}^{d} q_j(r(s))$ where $q_j(r(s))$ is the marginal distribution of the i-th dimension of r estimated by $q_j(r(s)) = \frac{1}{|\mathcal{D}_s^m|}\sum_{i \in \mathcal{D}_s^m} y_{ij}$, $\mathcal{D}_s^m := \{i : s \in \mathcal{S}_i\}$.

EFFICIENT LEARNING ALGORITHM

- Truncate f^w to avoid zero probability $f^{w,\lambda}(\chi_{\mathcal{S}}) = (1-2\lambda)f^w(\chi_{\mathcal{S}}) + \lambda$, λ is a small threshold value.
- Draw m i.i.d. cascades $\mathcal{D}^m := \{(\mathcal{S}_1, \mathcal{I}_1), \ldots, (\mathcal{S}_m, \mathcal{I}_m)\}$ with source set $\mathcal{S}_i$ and the respective set of influenced nodes $\mathcal{I}_i$.
- Let $y_{ji} = \mathbb{I}\{j \in \mathcal{I}_i\}$ denote whether node j is infected in cascade $\mathcal{I}_i$
- Learn the parameters w by maximizing the log-likelihood for each node j

$$\widehat{w} = \sum_{i=1}^{m} y_{ij} \log f^{w,\lambda}(\chi_{\mathcal{S}_i}) + (1 - y_{ij}) \log(1 - f^{w,\lambda}(\chi_{\mathcal{S}_i}))$$

$$\text{subject to } \sum_{k=1}^{K} w_k = 1, w_k \geq 0. \qquad (1)$$

by using convex optimization techniques.

OVERALL ALGORITHM INFLULEARNER

Algorithm 1 INFLULEARNER

input training data $\{(\mathcal{S}_i, \mathcal{I}_i)\}_{i=1}^{m}$, $\lambda \in (0, \frac{1}{4})$

1: **for** each node $j \in [d]$ **do**
2: sample K random features $\{r_1, \ldots, r_K\}$ from $q_j(r)$;
3: compute $\phi(\chi_{\mathcal{S}_i}) = (\phi(\chi_{\mathcal{S}_i}^\top r_1), \ldots, \phi(\chi_{\mathcal{S}_i}^\top r_K)), \forall i$;
4: Solve (1) using convex optimization;
5: $\widehat{f}_j^{w,\lambda}(\chi_{\mathcal{S}}) = \lambda + (1-2\lambda)(w^T)^\top \phi(\chi_{\mathcal{S}})$;
6: **end for**

output $\widehat{\sigma}(\mathcal{S}) = \sum_{j=1}^{d} \widehat{f}_j^{w,\lambda}(\chi_{\mathcal{S}})$;

SAMPLE COMPLEXITY

Suppose we set $\lambda = \tilde{O}(\frac{\epsilon}{d})$, $K = \tilde{O}(\frac{C^2 d^2}{\epsilon^2})$, and $m = \tilde{O}\left(\frac{C^2 d^3}{\epsilon^3}\right)$. Then with probability at least $1-\delta$ over the drawing of the random features, the output of Algorithm 1 satisfies $\mathbb{E}_{\mathcal{D}^m}\mathbb{E}_{p_\chi}\left[\left(\sum_{j=1}^{d} \widehat{f}_j^{w,\lambda}(\chi_{\mathcal{S}}) - \sigma(\mathcal{S})\right)^2\right] \leq \epsilon$.

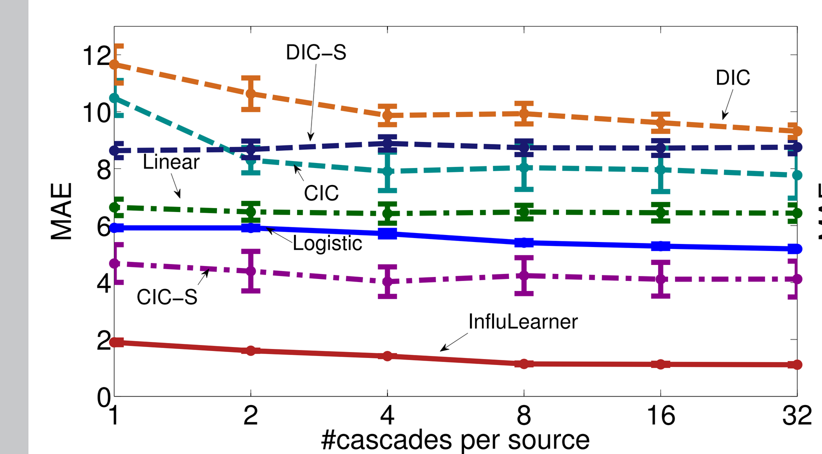
Intuitively, when the gap C between p_j and q_j is large, we need more random features and more training data to learn the weights.

EXPERIMENTAL EVALUATION : COMPETITORS

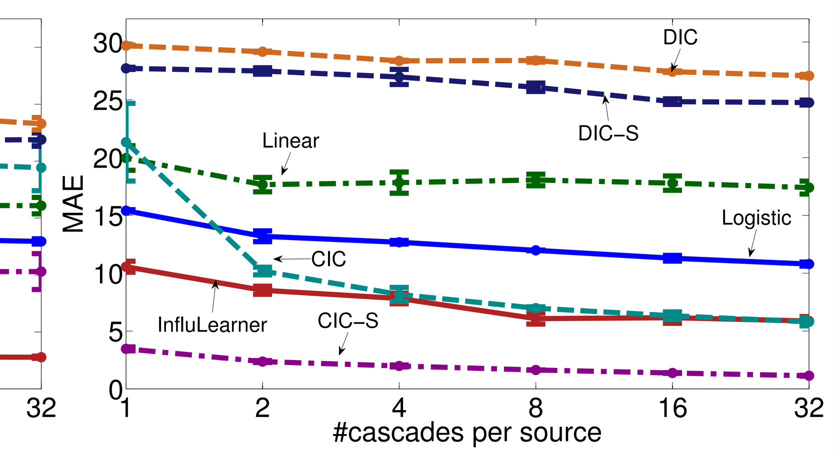
- **C**ontinuous-time **I**ndependent **C**ascade model with exponential pairwise transmission function (CIC).
- **C**ontinuous-time **I**ndependent **C**ascade model with exponential pairwise transmission function and given network **S**tructure (CIC-S).
- **D**iscrete-time **I**ndependent **C**ascade model (DIC).
- **D**iscrete-time **I**ndependent **C**ascade model with given network **S**tructure (DIC-S).
- Modified Logistic Regression
- Linear Regression

EXPERIMENTAL EVALUATION : SYNTHETIC DATA

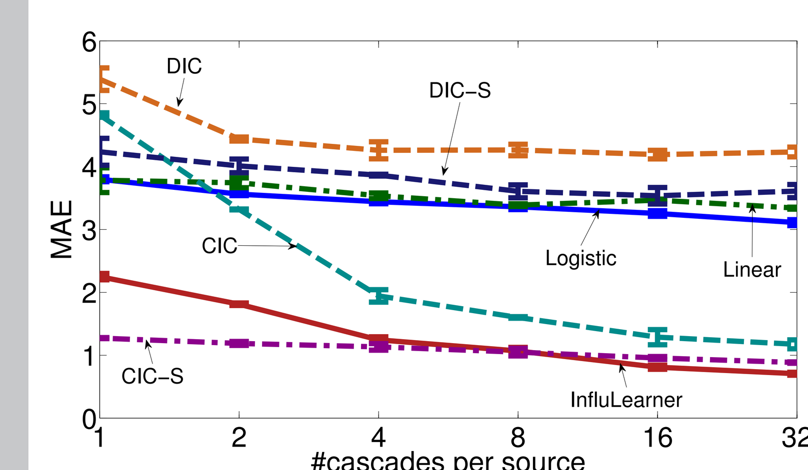
Robustness to model mis-specifications



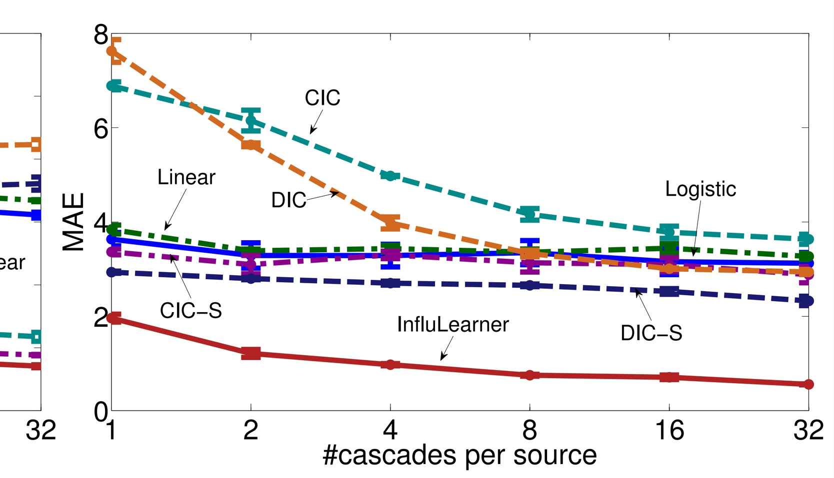
(a) Weibull Family (CIC)



(b) Exponential (CIC)

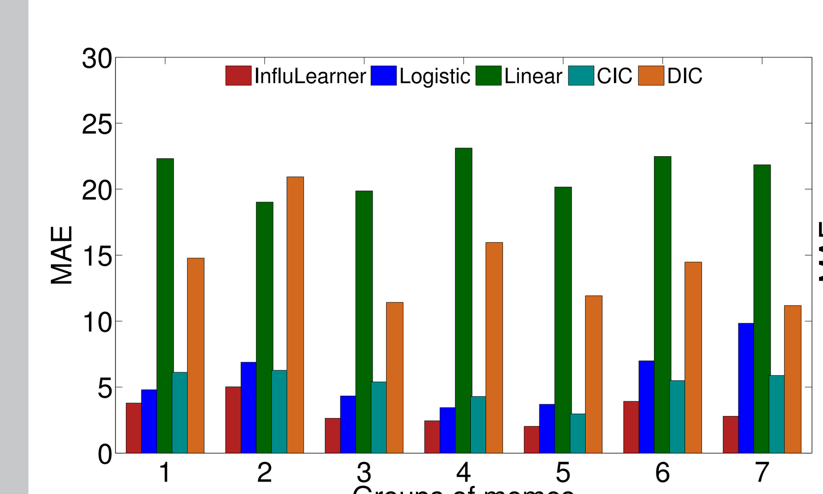


(c) DIC

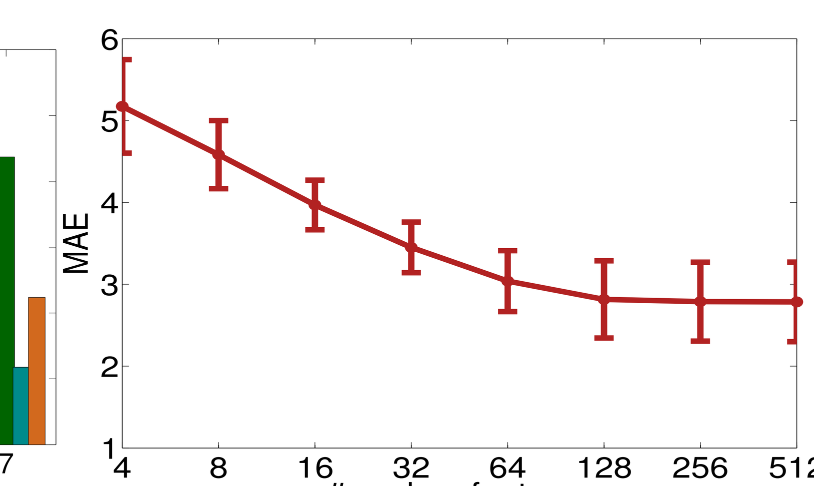


(d) LT

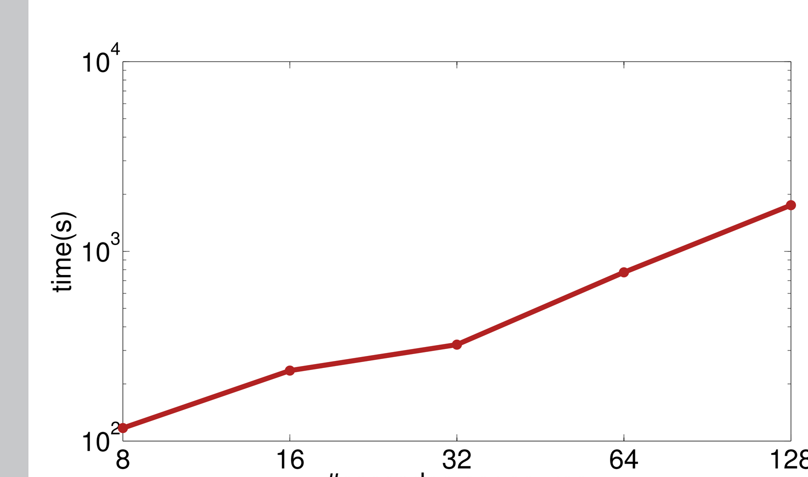
EXPERIMENTAL EVALUATION : REAL DATA



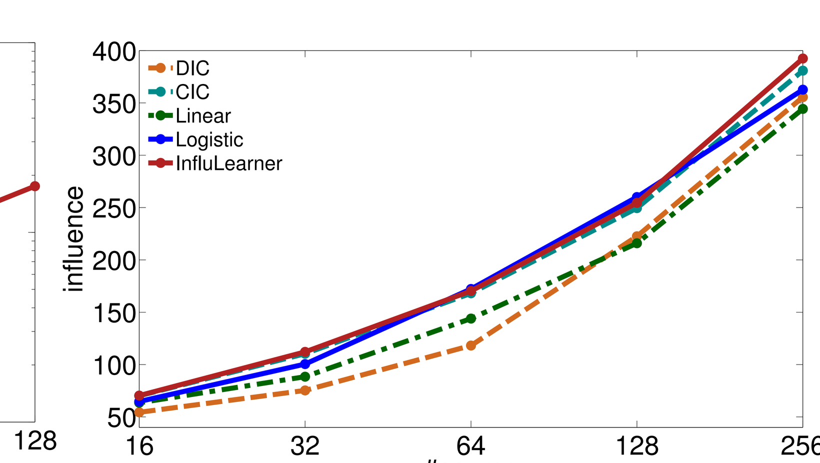
(a) MAE on real data



(b) Effect of random features



(c) Runtime



(d) Influence maximization

MAIL: DUNAN@GATECH.EDU