Viral marketing without tears: Limiting the harm caused by diffusing information to vulnerable users

Huiping Chen

huiping.chen@kcl.ac.uk

King's College London

Joint work with G. Loukides, J. Fan, H. Chan

London Stringology Days/London Algorithmic Workshop

February 8, 2019

Motivation (1/2): Social networks and viral marketing

- Social networks are powerful communication infrastructures
  - Facebook (1.94 billion monthly active users[1])
  - Twitter (313 million monthly active users[2])
- They allow diffusing information quickly to many users through word-of-mouth effects
  - good for advertising products or events through viral marketing
- The success of a viral marketing campaign on a social network can be measured by the number of influenced users

[1] http://newsroom.fb.com/company-info/

[2] https://about.twitter.com/company

Motivation (2/2): Influence maximization and its drawback

- **Influence maximization**
  - Find k users (*seeds*) that influence the largest number of users, according to a diffusion model

- **Drawback**: Some users (*vulnerable users*) may be harmed by information diffusion
  - Promoting alcoholic drinks to people with drinking problems
  - Promoting junk food to obese people

How to limit the influence to vulnerable users, while maximizing the influence to the non-vulnerable users (so that users and companies benefit from viral marketing)?

Contributions

- **Influence measure to quantify the quality of a seed-set**
  - Additive Smoothing Ratio (*ASR*)
- **Baseline Heuristics for finding an ASR-Maximizing seed-set**
  - *GR* natural greedy heuristic
  - GR_{MB}: a variation of *GR* (more efficient)
- **Approximation algorithm for finding an ASR-Maximizing seed-set**
  - *ISS* (Iterative Subsample with Spread bounds): an efficient approximation algorithm

Background (1/2): Set functions

Monotonicity

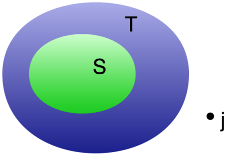
A function $f : 2^U \rightarrow \mathbb{R}$ is *monotone*, if $f(X) \leq f(Y)$ for all subsets $X \subseteq Y \subseteq U$, and *non-monotone* otherwise

Submodularity, supermodularity, and modularity

- A function $f : 2^U \rightarrow \mathbb{R}$ is **submodular**, if $\forall S \subseteq T \subseteq U$ and $j \in U \setminus T$:

$$f(S \cup \{j\}) - f(S) \geq f(T \cup \{j\}) - f(T) \quad (1)$$

- **supermodular**, if and only if $-f$ is submodular [3]
- **modular**, if Eq. 1 holds with equality



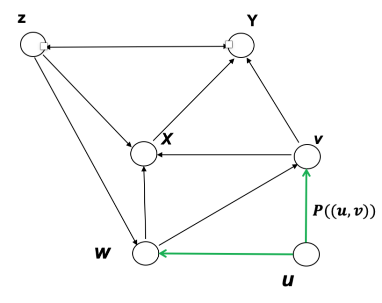
- diminishing returns property

Background(2/2): Graph representation and IC model

Social network as a graph

- Directed graph $G(V, E)$ that models a social network (at a certain time)
- V is partitioned into $\mathcal{N}$(non-vulnerable nodes) and $\mathcal{V}$(vulnerable nodes) and we assume ($\mathcal{N} \neq \varnothing$)

Independent Cascade (IC) model [2]



- Seed nodes are influenced at initial time point 0.
- At each next time point, each newly influenced node u activates its out-neighbor v independently, with probability $p((u, v))$.
- The process stops when no new nodes are activated.
- The **spread** (expected number of influenced users) for a seed-set S in the IC model is denoted with $\sigma(S)$.

Natural influence measures (1/2)

Difference

The difference $\sigma_{\mathcal{N}}(S) - \sigma_{\mathcal{V}}(S)$ between the spread of non-vulnerable and vulnerable users

Limitations

- It does not consider what fraction of all influenced users are vulnerable

Example

It favors promoting an alcoholic beverage to 140 users out of whom **40 have drinking problems**, instead of 59 users with no drinking problems, since $(140 - 40) - 40 > 59 - 0$.

- It cannot be used to find a seed-set S with approximately maximum $\sigma_{\mathcal{N}}(S) - \sigma_{\mathcal{V}}(S)$ [1]

Natural influence measures (2/2)

Ratio

The ratio $\frac{\sigma_{\mathcal{V}}(S)}{\sigma_{\mathcal{N}}(S)}$ between the spread of vulnerable and non-vulnerable users

Limitations

- It does not favor a seed-set that influences many non-vulnerable users (i.e., is good for viral marketing), among seed-sets that do not influence vulnerable users (does not distinguish seed-sets with $\sigma_V(S) = 0$).

Example

S_1 and S_2 do not influence users with drinking problems:

- S_1: 59 users with no drinking problems: $\frac{\sigma_{\mathcal{V}}(S_1)}{\sigma_{\mathcal{N}}(S_1)} = \frac{0}{59} = 0$
- S_2: 2 users with no drinking problems: $\frac{\sigma_{\mathcal{V}}(S_2)}{\sigma_{\mathcal{N}}(S_2)} = \frac{0}{2} = 0$

- It cannot be used to find a seed-set with small or zero $\sigma_{\mathcal{V}}(S)$ and large $\sigma_{\mathcal{N}}(S)$.

Our influence measure and problem definition

Additive Smoothing Ratio (*ASR*)

- $ASR(S,c) = \frac{\sigma_{\mathcal{N}}(S)+c}{\sigma_{\mathcal{V}}(S)+c}$, where S is a seed-set and $c > 0$ is a constant

Example

S_1: 59 users with no drinking problems, $ASR(S_1,1)=\frac{\sigma_{\mathcal{N}}(S_1)+1}{\sigma_{\mathcal{V}}(S_1)+1} = \frac{60}{1}$

S_2: 2 users with no drinking problems, $ASR(S_2,1)=\frac{\sigma_{\mathcal{N}}(S_2)+1}{\sigma_{\mathcal{V}}(S_2)+1} = \frac{3}{1}$

Problem definition

- Given $G(V,E)$ and $c > 0$, find a seed-set $S \subseteq V$ of size at most k with maximum $ASR(S,c)$

- NP-hard
- Cannot be approximated using algorithms for submodular and/or supermodular maximization because ASR is **non-monotone** and **neither submodular nor supermodular**.

Baseline heuristics (1/2)

GR (GReedy heuristic)

Input: $\mathcal{N} \subseteq V$, $\mathcal{V} \subseteq V$, graph G, parameter k, constant c

Output: Subset $S \subseteq \mathcal{N}$ of size $|S| \leq k$

$S_0 \leftarrow \{\}$; $i \leftarrow 0$

While $i < k$

Find a node $u \in \underset{v \in \mathcal{N} \setminus \{S_i\}}{\arg\max} \dfrac{\sigma_{\mathcal{N}}(S_i \cup v) - \sigma_{\mathcal{N}}(S_i) + c}{\sigma_{\mathcal{V}}(S_i \cup v) - \sigma_{\mathcal{V}}(S_i) + c}$

$S_{i+1} \leftarrow S_i \cup \{u\}$

$i \leftarrow i + 1$

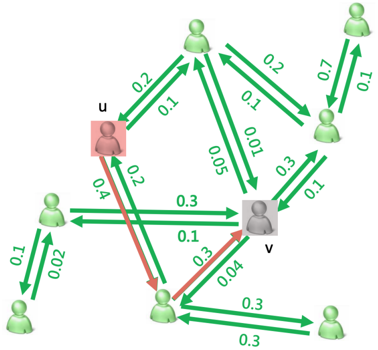
Return the subset $S \in \{S_1, \ldots, S_k\}$ with the largest *ASR*

Limitation: The computation of $\sigma_{\mathcal{N}}$ and $\sigma_{\mathcal{V}}$ is slow (all paths from S to $\mathcal{N}$ or $\mathcal{V}$ in the graph need to be considered)

Baseline heuristics (2/2)

GR_{MB}

- Differs from *GR* in that it estimates the spread efficiently using the **MIA** (Maximum Influence Arborescence) Batch-update method [6]
- **two orders of magnitude faster** on average than *GR*, but less effective in terms of *ASR*



- For any pair of nodes u and v, find the **maximum influence path** from u to v
- Estimate influence probability $P_S(u)$ as the union of maximum influence paths from S to u
- $\sigma_{\mathcal{N}} = \sum_{u \in \mathcal{N}} P_S(u)$
- $\sigma_{\mathcal{V}} = \sum_{u \in \mathcal{V}} P_S(u)$

The *ISS* approximation algorithm (1/3)

Main ideas

- We define submodular (easier to maximize) functions $ASR^{\mathbf{L}}$ and $ASR^{\mathbf{U}}$ that bound *ASR* from below and from above:

$$ASR^{\mathbf{L}}_{Y,c}(S) = \frac{\sigma_{\mathcal{N}}(S) + c}{\widehat{\sigma_{\mathcal{V},Y}}(S) + c} = \frac{\sigma_{\mathcal{N}}(S) + c}{\sigma_{\mathcal{V}}(Y) + \sum_{u \in S \setminus Y} \sigma_{\mathcal{V}}(\{u\}) - \sum_{u \in Y \setminus S} (\sigma_{\mathcal{V}}(Y) - \sigma_{\mathcal{V}}(Y \setminus \{u\})) + c}$$

$$ASR^{\mathbf{U}}_{Y,\pi^Y,c}(S) = \frac{\sigma_{\mathcal{N}}(S) + c}{\widetilde{\sigma_{\mathcal{V},\pi^Y}}(S) + c} = \frac{\sigma_{\mathcal{N}}(S) + c}{\sum_{u \in S} (\sigma_{\mathcal{V},Y,\pi^Y}(u)) + c}$$

  because $ASR(S, c)$ is non-monotone and non-submodular (difficult to maximize). The bounds are based on the modular bounds for submodular functions in [1].

- We select seeds from a sample of $\mathcal{N}$ of size approximately $\frac{|\mathcal{N}|}{k}$.

- Iterative construction of a seed-set, until *ASR* cannot improve.

The *ISS* approximation algorithm (2/3)

Simplified description of *ISS*

Input: $\mathcal{N} \subseteq V$, $\mathcal{V} \subseteq V$, graph G, parameter k, constant c

Output: Subset $S \subseteq \mathcal{N}$ of size $|S| \leq k$

$S_{pr} \leftarrow \{\}; S_{cur} \leftarrow \mathcal{N}$

While *true*

- $i \leftarrow 0; S_0^{\mathbf{O}} \leftarrow \{\}; S_0^{\mathbf{L}} \leftarrow \{\}; S_0^{\mathbf{U}} \leftarrow \{\}$
- **While** $i < k$
  - Uniform random sample with approximately $\frac{|\mathcal{N}|}{k}$ nodes
  - $S_{i+1}^{\mathbf{O}} \leftarrow$ add into $S_i^{\mathbf{O}}$ the node with max. marginal gain in ASR
  - $S_{i+1}^{\mathbf{L}} \leftarrow$ add into $S_i^{\mathbf{L}}$ the node with max. marginal gain in $ASR^{\mathbf{L}}_{S_{pr},c}$
  - $S_{i+1}^{\mathbf{U}} \leftarrow$ add into $S_i^{\mathbf{U}}$ the node with max. marginal gain in $ASR^{\mathbf{U}}_{S_{pr},\pi^{S_{pr}},c}$
  - $i \leftarrow i + 1$
- $S_{cur} \leftarrow$ best seed-set w.r.t ASR among $S_k^{\mathbf{O}}$, $S_k^{\mathbf{L}}$, $S_k^{\mathbf{U}}$
- **If** S_{cur} not better than S_{pr} w.r.t. ASR
  - **break**
- $S_{pr} \leftarrow S_{cur}$

Return S_{cur}

The *ISS* approximation algorithm (3/3)

- *ISS* constructs a seed-set with expected value of *ASR* no less than $\mathcal{M} \cdot 23\%$ of the optimal, where $\mathcal{M}$ depends on the constants c and k and the ASR^L function.

Theorem

ISS constructs a seed-set S such that:

$$\mathbb{E}[ASR(S,c)] \geq \max\left(\frac{\sigma_{\mathcal{V}}(S^*)+c}{\widehat{\sigma_{\mathcal{V},S_{pr}}}(S^*)+c}, \frac{c}{c+k\cdot \max_{u\in\mathcal{N}} \widehat{\sigma_{\mathcal{V},S_{pr}}}(\{u\})}\right)\cdot$$
$$\frac{1}{\mathrm{e}}\cdot(1-\frac{1}{\mathrm{e}})\cdot ASR(S^*,c)$$

where $S^* = \arg\max_{S\subseteq\mathcal{N},|S|\leq k} ASR(S,c)$, $\widehat{\sigma_{\mathcal{V},S_{pr}}}$ *is the modular upper bound used in* $ASR^{\mathbf{L}}$*, and the expectation is over every possible S constructed by ISS.*

Experimental setup

Evaluation of *GR*, GR_{MB}, *ISS*

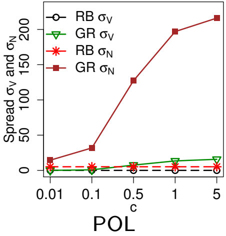
- **Competitors**:
  - *TIM* [5]: a heuristic for maximizing $\sigma_{\mathcal{N}}(S) - \sigma_{\mathcal{V}}(S)$,
  - *RB*: employs *Greedy* [4] to the subset of non-vulnerable nodes that influence no vulnerable nodes
- **Effectiveness measures**: $\sigma_{\mathcal{N}}$, $\sigma_{\mathcal{V}}$, *ASR*, $\frac{\sigma_{\mathcal{N}}}{|\mathcal{N}|}$, $1 - \frac{\sigma_{\mathcal{V}}}{|\mathcal{V}|}$
- **Efficiency measure**: Runtime

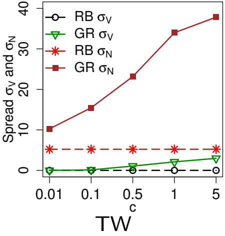
Datasets

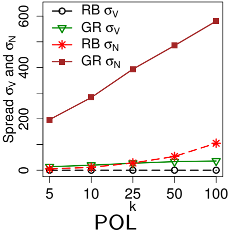
Dataset	# of nodes ($|V|$)	# of edges ($|E|$)	avg in-degree	max in-degree	# of vuln. nodes ($|\mathcal{V}|$)	θ
WI	7115	103689	13.7	452	100	0.01
TW	235	2479	10.5	52	25	0.01
POL	1490	19090	11.9	305	100	0.003
AB	840	10008	11.9	137	10	0.01

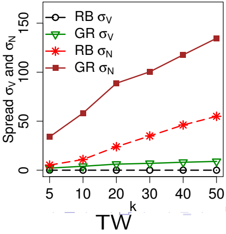
Comparison to *RB*

- *GR* constructs seed-sets that influence at least 5.5 and up to 38 times **more non-vulnerable nodes** than those constructed by *RB*, for different values of c and k



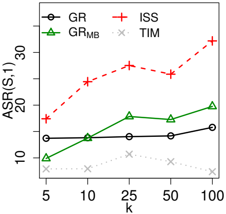




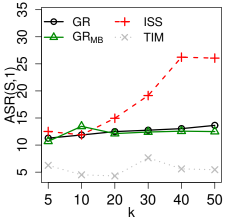


ASR with $c = 1$

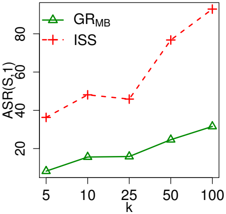
- All our algorithms substantially **outperform** *TIM*
- *ISS* outperformed all other method **3.5 times** on average over all datasets, k value and $|\mathcal{V}|$ values



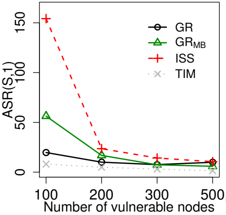
POL



TW



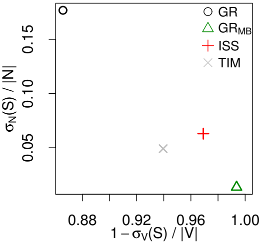
WI



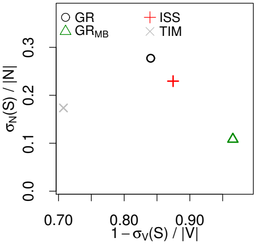
POL

Spread of Vulnerable and Non-vulnerable Nodes

- Each point (x, y) corresponds to the values $(1 - \frac{\sigma_{\mathcal{V}}(S)}{|\mathcal{V}|}, \frac{\sigma_{\mathcal{N}}(S)}{|\mathcal{N}|})$, referred to as *protection* and *utility* of a seed-set S



POL

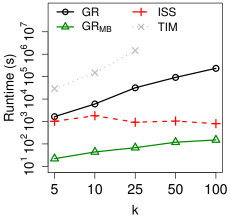


TW

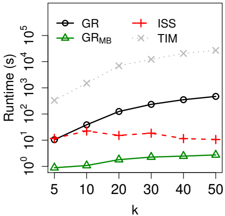
- All our algorithms substantially **outperformed** *TIM* in terms of $\sigma_{\mathcal{N}}$ and/or $\sigma_{\mathcal{V}}$
- *ISS* outperformed *TIM* with respect to **both protection and utility**, achieving overall better protection than *GR* and better utility than GR_{MB}

Efficiency

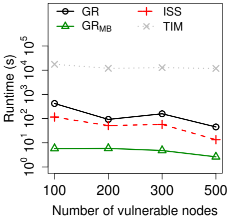
- Our methods are **faster** than *TIM* by at least one order of magnitude
- *TIM* is too slow (10 hours for $k = 50$ and a dataset with 235 nodes, and more than 17 days for larger datasets)



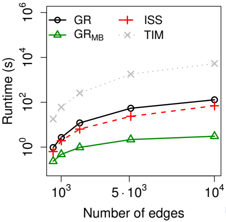
POL



TW



POL



AB

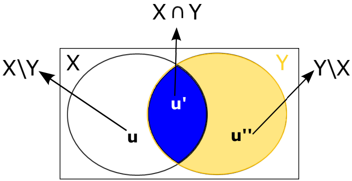
Conclusions

- Introduced the problem of performing viral marketing while limiting the influence to vulnerable nodes
- Proposed an influence measure and defined an optimization problem based on the measure
- Proposed two greedy baseline heuristics and the *ISS* approximation algorithm
- Experimentally showed that *ISS* outperforms *TIM* [5] and our baselines in terms of effectiveness and efficiency

Forthcoming IEEE AINA paper:
`https://kclpure.kcl.ac.uk/portal/files/104770966/VIM_paper_final.pdf`

Background (3/5): Modular bounds

- We review two bounds for a submodular function that are used in our approximation algorithm.
- The bounds are computed for a given subset $Y \subseteq U$.
- The bounds are modular and thus easier than f to optimize efficiently.



Modular upper bound [1]

- The *modular upper bound* $\widehat{f}_Y(X)$ of a submodular function $f : 2^U \rightarrow \mathbb{R}$ is a modular function [1]

$$\widehat{f}_Y(X) = f(Y) + \sum_{u \in X \setminus Y} (f(\{u\}) - f(\{\})) - \sum_{u \in Y \setminus X} (f(Y) - f(Y \setminus \{u\})) \quad (2)$$

where $Y \subseteq U$ is a given subset of U.

Background (4/5): Modular bounds

Modular lower bound [1]

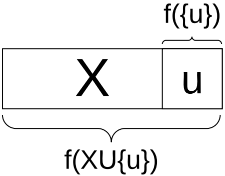
- The *modular lower bound* $\underbrace{f_{Y,\pi^Y}}(X)$ of a submodular function $f(X) : 2^U \to \mathbb{R}$ is a modular function

$$\underbrace{f_{Y,\pi^Y}}(X) = \sum_{u \in X} f_{Y,\pi^Y}(u) \tag{3}$$

where $Y \subseteq U$ is a given subset of U, π^Y is a random permutation of the elements of Y, π^Y_u is the prefix of π^Y, $\pi^Y_{u^-}$ is π^Y_u except u, and

$$f_{Y,\pi^Y}(u) = \begin{cases} f(\pi^Y_u) - f(\pi^Y_{u^-}), & \text{if } u \in Y \\ 0, & \text{otherwise} \end{cases} \tag{4}$$

Background (5/5): Modular bounds



Marginal gain of u

f(XU{u})-f(X)

- $X \to \pi^Y$
- $f(X \cup \{u\}) - f(X) \to f_{Y,\pi^Y}(u)$
  $= \begin{cases} f(\pi_u^Y) - f(\pi_{u^-}^Y), & \text{if } u \in Y \\ 0, & \text{otherwise} \end{cases}$
- $f(X) \to \widecheck{f}_{Y,\pi^Y}(X)$

References

R. Iyer and J. Bilmes.
Algorithms for approximate minimization of the difference between submodular functions, with applications.
In *UAI*, pages 407–417, 2012.

D. Kempe, J. Kleinberg, and E. Tardos.
Maximizing the spread of influence through a social network.
In *KDD*, pages 137–146, 2003.

A. Krause and D. Golovin.
Submodular function maximization.
In *Tractability*. 2013.

G. L. Nemhauser, L. A. Wolsey, and M. L. Fisher.
An analysis of approximations for maximizing submodular set functions.
Mathematical Programming, 14(1):265–294, 1978.

Ramakumar Pasumarthi, Ramasuri Narayanam, and Balaraman Ravindran.
Near optimal strategies for targeted marketing in social networks.
In *AAMAS*, pages 1679–1680, 2015.

C. Wang, W. Chen, and Y. Wang.
Scalable influence maximization for independent cascade model in large-scale social networks.
DMKD, 25(3):545–576, 2012.