Analyzing Competitive Influence Maximization Problems with Partial Information

Yishi Lin, John C.S. Lui
The Chinese University of Hong Kong
Background: Word-of-Mouth

In social interactions, we influence each other.

**Foodie @Sydney, Australia**
Awesome strawberry watermelon cake with rose scented cream!!
Background: Viral Marketing

- **Assumption**: the *word-of-mouth* effect
- **Idea**: exploiting the *social influence* for marketing
- **Targeting** “influencers” who are likely to produce the word-of-mouth diffusion
Background: Classical *Independent Cascade* model

- **Single source *Independent Cascade* (IC) model** (Kempe et al. KDD’03)
  - Initially, a set of "seed nodes" $S$ are activated.
  - Influenced node $u$ influences its neighbor $v$ with probability $p_{uv}$.
  - **Influence spread $\sigma(S)$**: the expected number of influenced nodes

---

**Background:** Classical *Influence Max. Problem*

**Input:** $G$ and $k$

**Problem (Influence Maximization)**
Select $k$ seed nodes so to maximize the expected spread of influence.

**Output:**
Seed set of size $k$

**Under the IC model:**
- The IM problem is NP hard. 😞
- Even computing $\sigma(S)$ is #P hard. 😞
Motivation

- Competition among products

- **Partial information**: It is not always possible to have full information about viral marketing strategies of the competitor.
Main Contributions

**General Competitive Independent Cascade Model**
- Many specific models proposed previously are its special cases
  - Distance-based model (Carnes et al. ICEC’2007)
  - Wave propagation model (Carnes et al. ICEC’2007)
  - Campaign-Oblivious Independent Cascade model (Budak et al. WWW’11)

**General Competitive Influence Maximization Problem**
- Assuming only partial knowledge about competitor’s seeding strategy

**General algorithmic framework**
- It solves the general problem.
- It works for any specific instances of the general model.
Model
General Competitive Independent Cascade Model

- **Network** $G = (V, E)$:
  - Every edge $e_{uv}$ is associated with a probability $p_{uv}$.

- **Sources**: two competing sources $A$ and $B$.

- **State of a node**: Susceptible, Inf$_A$ or Inf$_B$
  - “Influenced” cannot change its state.

- **Seeds / initial adopters**: $S_A \subseteq V, S_B \subseteq V$
  - We assume $S_A \cap S_B = \emptyset$. 
Model
General Competitive Independent Cascade Model

- **Given seeds:** $S_A$ and $S_B$
- **Determine propagation results**
  - Active edges $E_a$: edge $e_{uv}$ is “active” w.p. $p_{uv}$.
  - Node $u$ will be in the same state as that of *one of its nearest seeds* in $G = (V, E_a)$.
  - A specific model should specify how the influence propagates in detail.
- **The expected influence**
  - $\sigma(S_B \mid S_A) = \mathbb{E}_{E_a} \left[ \# \text{ of nodes in state Inf}_B \right]$
- **Assumption**
  - monotonicity and submodularity of $\sigma(S_B \mid S_A)$

$S_A = \{3\} \quad S_B = \{5\}$
Problem Definition

Competitive Influence Maximization problem with Partial information (CIMP)

Input:
- $G$, $k$, propagation model
- Competitor’s seed distribution $D_A$

Problem
Select a set $S_B^*$ of $k$ nodes so that the expected spread of influence of source $B$ under the presence of competitors, $\sigma(S_B^*|D_A)$, is maximized.

$$\sigma(S_B^*|D_A) = \mathbb{E}_{S_A \sim D_A} [\sigma(S_B^*|S_A)]$$

Output:
Seed set $S_B$ of size $k$
Problem Definition

Competitive Influence Maximization problem with Partial information (CIMP)

The CIMP problem is **NP hard. 😞**
Even computing $\sigma(S_B|D_A)$ is **#P hard. 😞**

Solution

Two-phase Competitive Influence Maximization (TCIM)
TCIM: Estimating the Expected Influence

Random **Reverse Accessible Pointed Graph** (RAPG)

**Input:**
- Random root \( r \)
- Random seeds \( S_A \sim \mathcal{D}_A \)
- Random active subgraph \( g \)

**Output:** \( R = (V_R, E_R, S_{R,A}) \)
- \( V_R \): nodes that might influence \( r \) in \( g \)
- \( E_R \): all shortest paths from \( V_R \) to \( r \) in \( g \)
- \( S_{R,A} = S_A \cap V_R \): seeds of source \( A \) in \( R \).

\[
g = G\{e_1, e_6, e_7\}, \quad S_A = \{4,9\}
\]

\[
R = (V_R, E_R, S_{R,A})
\]
TCIM: Estimating the Expected Influence

- **Reverse Accessible Pointed Graph** $R$
- Seed set $S_B$
- Specific competitive propagation model

"Score" of $S_B$ in $R$:
$\Pr[S_B \text{ influences the root of } R]$

Distance-based Model
"Score" of $S_B$ in $R$: $1/2$

$S_A = \{9\}$
$S_B = \{11\}$

(Lemma 1) $n \cdot \mathbb{E}_R[\text{Score}_R(S_B)] = \sigma(S_B|\mathcal{D}_A)$

$\mathbb{E}[\text{"Score" of } S_B \text{ in a random } R]$
$= \Pr[S_B \text{ influences a random node in } G]$
TCIM: High Level Ideas

(Lemma 1) $n \cdot \mathbb{E}_R[\text{Score}_R(S_B)] = \sigma(S_B|\mathcal{D}_A)$

Chernoff-Hoeffding Bound

---

Estimating Influence

- Given $\mathcal{D}_A$ and a sufficiently large number of random RAPG instances,
  $n \cdot \text{ave}[\text{score}_R(S_B)] \approx \sigma(S_B|\mathcal{D}_A)$.

Node Selection  Monotone & Submodular

- We know how to estimate $\sigma(S_B|\mathcal{D}_A)$
- We greedily add nodes to $S_B$ with the goal of maximizing $\sigma(S_B|\mathcal{D}_A)$

---

Phase 1: Parameter estimation

Estimate and refine the number of RAPG instances we need.

Phase 2: Node Selection

1. Generate enough RAPGs
2. Selects seeds for source $B$
TCIM: Main Results

(Theorem 4) Two-phase Competitive Influence Maximization

**Practical performance guarantee**
- \( \sigma(S_B|\mathcal{D}_A) \geq (1 - 1/e - \epsilon) \cdot \sigma(S_{B}^{OPT}|\mathcal{D}_A) \), with probability at least \( 1 - n^{-\ell} \)
- the best approximate ratio one could obtain in polynomial time

**Practical efficiency**
- \( O((c(\ell + k)(m + n) \log n)/\epsilon^2) \)
- the value of \( c \) is related to the specific GCIC model
Application: A Special Case of the GCIC model

Distance-based Model (Carnes et al. ICEC’2007)

- Given $S_A$, $S_B$ and a set of active edges $E_a$.
- Probability that source $B$ influences node $u$:

$$\frac{\text{# of } u\text{'s nearest seeds of source } B}{\text{# of } u\text{'s nearest seeds of both sources}}$$

- TCIM Complexity: $O\left(\frac{k(\ell + k)(m + n) \log n}{\epsilon^2}\right)$

$$c = O(k)$$

$S_A = \{3\}, S_B = \{4,5\}$
Experiments

Comparison among the TCIM framework and previous methods

- **Dataset**: a *Facebook-like social networks* (1,899 nodes and 20,296 directed edges)

- **Baselines**:
  - CELF (Leskovec et al. ICDM’07): a greedy method
  - CELF++ (Goyal et al. WWW’11): a greedy method
  - DegreeDiscount (Chen et al. KDD’09): a heuristic method

- **Settings**:
  - For each edge $e_{uv} \in E$, we set $p_{uv} = 1/d_v$ (IC-Weighted Cascade model).
  - We select 50 nodes using single source influence maximization method for source $A$. 

Comparison among the TCIM framework and previously methods

The influence spread of $S_B$ returned by TCIM, CELF and CELF++ are comparable.

Up to 4 orders of magnitude speedup
Experimental Results

Results on larger datasets

- The NetHEPT collaboration network (15,233 nodes and 58,891 undirected edges)
- The Epinion social network (508,837 directed relationships among 75,879 users)

Remarks:

1. When $\epsilon = 0.5$, TCIM finishes within 7 seconds for the NetHEPT dataset and finishes within 23 seconds for the Epinion dataset.
2. If we do not require a very tight approximation ratio, we could choose a larger $\epsilon$. 

Figure 7: Results on large datasets: Running time versus $\epsilon$ under three propagation models. ($|S_A| = 50, k = 50, \ell = 1$)
Experimental Results

Results on larger datasets

- The NetHEPT collaboration network (15,233 nodes and 58,891 undirected edges)
- The Epinion social network (508,837 directed relationships among 75,879 users)

Remarks:
1. With the increase of $k$, the running time of TCIM tends to drop first, because the number of RAPG instances needed decreases.
2. TCIM is especially efficient for large $k$.

Figure 6: Results on large datasets: Running time versus $k$ under three propagation models. ($|S_A| = 50$, $\epsilon = 0.1$, $\ell = 1$)
Experiments: TCIM with partial information

| Competitor’s strategy | influence given explicit $S_A$ selected by different methods ($|S_A| = 50$) |
|-----------------------|---------------------------------------------------------------------------------|
|                       | COICM                                                                              | Wave propagation model                                                    |
|                       | greedy  | degree  | centrality  | average  | greedy  | degree  | centrality  | average  |
| **dataset**           |        |         |             |          |        |         |             |          |
| *NetHEPT*             | mixed method | 599.82  | 632.23       | 657.49   | **629.85** | 586.58  | 624.41       | 650.39   | **620.46** |
|                       | greedy | 658.38  | 515.72       | 519.50   | **564.53** | 644.53  | 525.70       | 515.37   | **561.87** |
|                       | degree | 400.18  | 702.93       | 622.15   | **575.09** | 372.58  | 693.95       | 613.98   | **560.17** |
|                       | centrality | 233.14  | 478.74       | 763.43   | **491.77** | 201.72  | 462.66       | 752.97   | **472.45** |
| **Epinion**           | mixed method | 2781.71 | 4603.63      | 10683.26 | **6022.87** | 2773.17 | 4494.80      | 10517.00 | **5928.32** |
|                       | greedy | 4440.93 | 3958.87      | 6372.13  | **4923.98** | 4265.87 | 3813.06      | 6377.30  | **4818.74** |
|                       | degree | 3130.99 | 5473.33      | 7283.28  | **5295.87** | 2983.56 | 5299.18      | 7258.24  | **5180.33** |
|                       | centrality | 224.93  | 2809.74      | 12078.70 | **5037.79** | 204.01  | 2721.87      | 12075.78 | **5000.55** |

Table 1: Expected influence of seeds $S_B$ returned by the TCIM framework given the “mixed method distribution” (mixed method) as seed distribution for source $A$ or given the guess of explicit seeds of $A$. Seeds “greedy” for source $A$ is the set of nodes selected by single source influence maximization algorithm. The set “degree” for source $A$ (resp. “centrality”) denotes the top 50 nodes ranked by (out)degree (resp. closeness centrality). ($k = 50, \epsilon = 0.1, \ell = 1$)
Conclusion

➢ General problem formulation
  ◦ General Competitive Independent Cascade (GCIC) model
  ◦ Competitive Influence Maximization problem with Partial information (CIMP)

➢ General Two-phase Competitive Influence Maximization (TCIM) framework
  ◦ It solves the CIMP problem under the GCIC model.
  ◦ With probability at least $1 - n^{-\ell}$, it guarantees a $(1 - 1/e - \epsilon)$-approximate solution.
  ◦ It runs in $O((c(\ell + k)(n + m) \log n)/\epsilon^2)$ expected time, where $c$ depends on the specific propagation model.

➢ We conduct extensive experiments using real datasets. For example,
  ◦ When $S_A$ is given explicitly, we achieve up to four orders of magnitude speedup as compared to previous algorithms with the same quality guarantee.
Thank you!