STREAMING INFLUENCE MAXIMIZATION IN SOCIAL NETWORKS BASED ON MULTI-ACITION CREDIT DISTRIBUTION

Qilian Yu*, Hang Li*, Yun Liao^, and Shuguang Cui*

*Dept. of ECE, University of California, Davis, USA ^Dept. of ECE, University of California, San Diego, USA

Introduction

In a social network, influence maximization is the problem of identifying a set of users that own the maximum influenceability across the network. In this paper, a novel credit distribution (CD) based model, termed as the multi-action CD (mCD) model, is introduced to quantify the influence ability of each user. Based on this model, influence maximization is formulated as a submodular maximization problem under a knapsack constraint, which is NP-hard. An efficient streaming algorithm is developed to achieve *(1/3-ε)* approximation of the optimality. Experiments conducted on real Twitter dataset demonstrate that the mCD model enjoys high accuracy compared to the conventional CD model in estimating the total number of people who get influenced in a social network. Furthermore, compared to the greedy algorithm, the proposed single-pass streaming algorithm achieves similar performance in terms of influence maximization, while running several orders of magnitude faster.

Model Design

Multi-action Event Logs. The same action for one particular user could be recorded for multiple times if the user performs this action repeatedly.

Direct Credit. This credit is what user *u* assigns to user *v* when *u* takes the same action *a* after *v*.

$$R_{u,a} = \sum_{v \in N_{in}(u,a)} \exp\left(-\Delta t_{v,u}(a)/\tau_{v,u}\right), \quad \Delta t_{v,u}(a) = 1/\textstyle\sum_{t \in \mathcal{T}_{v,u}(a)} \left(t_1(u,a) - t\right)^{-1}$$

$$\gamma_{v,u}(a) = \begin{cases} \exp\left(-\frac{\Delta t_{v,u}(a)}{\tau_{v,u}}\right) \cdot R_{u,a}^{-1}, & (v,u) \in \mathcal{E}(a); \\ 0, & \text{otherwise.} \end{cases}$$

Indirect Credit. Suppose that *(v, w)* and *(w, u)* are in *E(a)* such that *v* and *u* are connected indirectly. Then, user *u* may assign all indirect credit to *v* via *w* as $\gamma_{v,w}(a) \cdot \gamma_{w,u}(a)$.

Average Credit. $\Gamma_{v,u}(a) = \sum_{w \in N_{in}(u,a)} \Gamma_{v,w}(a) \cdot \gamma_{w,u}(a)$

Problem Formulation

Budgeted Influence Maximization Problem. We consider the budget of selecting users into the influencer set *S* as the major constraint, where the budgeted influence maximization problem could be cast as

$$\begin{aligned} \underset{\mathcal{S} \subseteq \mathcal{V}}{\text{maximize}} \quad & \sigma_{mCD}(\mathcal{S}) = \textstyle\sum_{u \in \mathcal{V}} \kappa_{\mathcal{S},u} \\ \text{subject to} \quad & g^T I_{\mathcal{S}} \leq b. \end{aligned} \tag{1}$$

Proposition 1. $\sigma_{mCD}(\mathcal{S}) \leq |\cup_{a \in \mathcal{A}} \mathcal{V}(a)|.$

Therefore problem (1) is to find a subset *S* from the ground set *V* to maximize a lower bound of the total number of users that finally get influenced over all actions.

Algorithm

The algorithm is divided into the following modules.

Model Learner is designed to learn the parameters that is the mathematical average time delay between each pair *(v, u)* over all actions, and *Au(a)*, the frequency of *u* taking action *a*, from the training dataset before solving the optimization problem, such that the algorithm can deal with a newly arriving dataset or test set much more efficiently.

Log Scanner scans the new or test set of data to calculate the total credit assigned to user *v* by *u* for action *a* by using the already learned parameters from the training set.

Problem Solver solves the influence maximization problem and outputs the seed set.

Special Case

We start with a cardinality constraint as a special case of the knapsack constraint (by applying the same weight for every user). Given *k* as the cardinality limit for *S*, the simplified problem (also known as the conventional influence maximization problem) is cast as

$$\begin{aligned} \underset{\mathcal{S} \subseteq \mathcal{V}}{\text{maximize}} \quad & \sigma_{mCD}(\mathcal{S}) \\ \text{subject to} \quad & |\mathcal{S}| \leq k. \end{aligned} \tag{2}$$

Algorithm 1 STREAMING_ALGORITHM(k, UC)

1: **for** each $x \in \mathcal{V}$
2: $m := \max\{m, \sigma_{mCD}(\{x\})\}$
3: $\mathcal{O} := \{(1+\epsilon)^i | i \in \mathbb{Z}, m \leq (1+\epsilon)^i \leq 2k \cdot m\}$.
4: **for** $c \in \mathcal{O}$
5: **if** marginal gain of c is over $\frac{c}{2k}$ and $|\mathcal{S}_c| < k$
6: $\mathcal{S}_c := \mathcal{S}_c \cup \{x\}$.
7: **end if**
8: **end for**
9: **end for**
10: **return** $\mathcal{S} := \text{argmax}_{\mathcal{S}_c, c \in \mathcal{O}} \sigma_{mCD}(\mathcal{S}_c)$.

Lemma 1. There exists a value *c* in set *O* such that *(1 − ε)OPT ≤ c ≤ OPT*, with *OPT* denoting the optimal value for problem (2).

Theorem 1. Algorithm 1 produces a solution *S* such that the set function value of the solution set is larger than *(1/2-ε)OPT*.

General Case

Next, to solve problem (1), we modify:

The threshold in line 5 of Algorithm 1 to $\frac{2q_x}{3b}$.

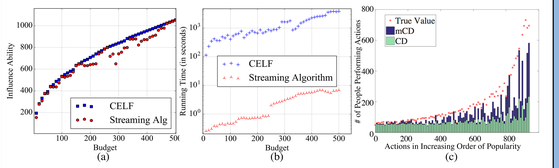
$q \in \mathcal{Q} := \{(1+3\epsilon)^i | i \in \mathbb{Z}, \frac{m}{1+3\epsilon} \leq (1+3\epsilon)^i \leq 2b \cdot m\}$

Keeps searching for a particular user who has dominated influences.

Experimental Results

We conduct experiments on a Twitter dataset containing about 17,000 users and 100 actions to evaluate the mCD model and the corresponding streaming algorithm.

We are interested in the following performance metrics: 1) the influence ability of the seed set provided by our proposed streaming algorithm; 2) the gap between the output influence ability and the number of people that truly get influenced; and 3) the running time of the algorithm.



a) Influence Ability Comparison. b) Running Time Comparison.
c) Estimated Influence for Actions in Test Set.

Conclusion

In this work, we extended the conventional CD model to the mCD model in dealing with the multi-action event logs and analyzing the influence ability of users in social networks.

We re-designed the credit assignment method in the CD model by utilizing a modified harmonic mean to handle multi-action event logs, which achieves a higher accuracy in estimating the total number of people that get influenced. Based on this new model, an efficient streaming algorithm was developed with *(1/3-ε)*-approximation of the optimal value for the corresponding budgeted influence maximization problem.

Experiments showed that the mCD model is more accurate compared to the conventional CD model, and the proposed algorithm