Profit Maximization in Signed Social Networks ⋆

Poonam Sharma[0009−0009−2608−9414] and Suman Banerjee[0000−0003−1761−5944]

Department of Computer Science and Engineering, Indian Institute of Technology Jammu, Jammu & Kashmir, India
{poonam.sharma,suman.banerjee}@iitjammu.ac.in

Abstract. In recent times, *Online Social Networks* have been predominantly used by commercial houses for promoting their products. The key approach in this context is to choose a limited number of highly influential users such that if they become the initial adopter of the product then the influence (and hence profit) in the network gets maximized. Existing studies on this problem consider that all the links of the network contain positive influence probabilities only. However, in practice, a social network contains links for both positive as well as negative probabilities, and such networks are called *Signed Social Networks*. In this paper, we study the profit maximization problem in the context of signed social networks under the popular *Polarity Linked Information Diffusion (PLID) Model*. We show that, in our case, the profit function is non-negative, non-monotone, and non-submodular. We propose two solution approaches for this problem. The first one is an iterative greedy approach, which works based on marginal gain computation, and the other one is a local search-based approach, which iteratively improves the quality of the solution. Both methods have been analyzed to understand their time and space complexity. A number of experiments have been carried out to evaluate the effectiveness and efficiency of the proposed solution approaches. We observe that in most of the instances the seed set selected by the proposed solution approaches leads to more profit compared to the existing methods.

Keywords: Profit · Seed Set · Diffusion Model · Signed Social Network.

1 Introduction

Due to the advancement of wireless networks and internet technology, the use of online social networks is ubiquitous in human life. Many commercial houses grabbed this opportunity to make promotional advertisements for their products using social media platforms [1,2]. The key approach here is to select a limited number of highly influential users such that if they become initial adopters (also called seed nodes) of their products, it will lead to significant influence among their customers. In this problem, each user of the network is associated with a selection cost and benefit. This benefit can be earned if the user is influenced.

⋆ The work of Dr. Suman Banerjee is supported by the Research Initiation Grant (Grant No.: SG100047) supported by Indian Institute of Technology Jammu.

The key problem here is to choose a subset of the users from the network with the allocated budget such that their initial activation leads to maximum profit. This problem remained the central theme of research in social network analysis for the last one decade.

One of the important property of a social network is the diffusion of information [3]. In this phenomenon, the information, ideas, innovations, etc., are shared by individual users, and information shared by an influential user may influence an ordinary individual with high probability. To study this diffusion process several models have been introduced and studied. Among those, two popular models are the Independent Cascade Model (ICM) and the Linear Threshold Model (LTM) [4]. However, one important limitation of these two models is that they can model the diffusion process when the polarity of the edges has not been considered. In practice, there are social networks where some links are of positive polarity and some links are of negative polarity. Such networks are called *Signed Social Networks*. To model the information diffusion process, there exists a few studies [5–7]. One such model is the *Polarity Linked Information Diffusion* [8].

In most of the existing studies of the Profit Maximization Problem, it has been implicitly assumed that the links in the underlying network contain positive influence probabilities [9]. Several problems have been studied on signed social networks, including community detection, link prediction, social influence maximization, and many more. To the best of our knowledge, the problem of profit maximization has not been studied on signed social networks. In this paper, we bridge this gap by studying the Profit Maximization Problem where the underlying network is signed in nature. In particular, we make the following contributions in this paper.

- To the best of our knowledge, this is the first study on the Profit Maximization Problem where the underlying network is signed in nature.
- We pose this problem as a discrete optimization problem and propose two heuristic solution approaches.
- All the proposed solution methodologies have been implemented on real-world signed social network datasets to show the effectiveness of the proposed solution approach.

The rest of the paper has been arranged as follows. Section 2 gives a background and defines our problem formally. Section 3 describes our proposed solution approaches, and Section 4 describes the experimental evaluations of the proposed solution approaches. Finally, Section 5 concludes our study and gives future research directions.

2 Background and Problem Definition

In this section, we describe the background of our problem and define it formally. Initially, we start by describing Signed Social Networks.

2.1 Signed Social Networks

In our study, we model a signed social network by a directed, weighted graph $\mathcal{G}(V, E, \mathcal{W}, \mathcal{P})$ where the vertex set of $\mathcal{G}$; i.e.; $V(\mathcal{G}) = \{u_1, u_2, \ldots, u_n\}$ denotes the set of users of the network. The edge set $\mathcal{E}(\mathcal{G})$ represents the social ties among the users; i.e.; $(u_i u_j)$ is an edge in $\mathcal{G}$ if the users u_i and u_j has the social relation. Let, n and m denotes the number of nodes and edges of $\mathcal{G}$, respectively. There are two edge weight functions $\mathcal{W}$, and $\mathcal{P}$. $\mathcal{W}$ assigns each edge to its corresponding influence probability; i.e.; $\mathcal{W} : \mathcal{E}(\mathcal{G}) \longrightarrow (0, 1]$. In the case of a signed social network, every edge is associated with a polarity. In our study, we assume that there are three discrete levels of polarity: positive (denoted by $+1$), negative (denoted by -1), and neutral (denoted by 0). $E^+(\mathcal{G})$ and $E^-(\mathcal{G})$ denotes the set of edges of $\mathcal{G}$ having positive and negative polarity, respectively. Hence, $E(\mathcal{G}) = E^+(\mathcal{G}) \cup E^-(\mathcal{G})$. m^+ and m^- denotes the number of edges of $\mathcal{G}$ having positive and negative polarity, respectively. For any edge $e \in E(\mathcal{G})$, $\mathcal{W}(e)$ and $\mathcal{P}(e)$ denote the influence probability and the polarity of the edge, respectively.

2.2 Diffusion in Signed Social Networks

In the signed social network, each node will have two probabilities. The first one is to be influenced positively and the other one is to be influenced negatively. The polarity of the edges is also used in the diffusion process. The probability that any node v will be influenced positively and negatively from the seed set $\mathcal{S}$ is denoted as $\mathcal{P}^+_{\mathcal{S}\to v}$ and $\mathcal{P}^-_{\mathcal{S}\to v}$, respectively. For any seed set $\mathcal{S}$ and for all the nodes of the network their positive and negative influence probabilities can be represented as vectors $\mathbb{P}^+_{\mathcal{S}} = [\mathbb{P}^+_{\mathcal{S}\to u_1}, \mathbb{P}^+_{\mathcal{S}\to u_1}, \ldots, \mathbb{P}^+_{\mathcal{S}\to u_n}]$ and $\mathbb{P}^-_{\mathcal{S}} = [\mathbb{P}^-_{\mathcal{S}\to u_1}, \mathbb{P}^-_{\mathcal{S}\to u_2}, \ldots, \mathbb{P}^-_{\mathcal{S}\to u_n}]$. In this paper, we adopt the diffusion model in the signed social network proposed by Li et al. In this diffusion model for any node $v \in \mathcal{S}$, its positive influence probability from the seed set $\mathcal{S}$ is 1. For any node $v \notin \mathcal{S}$, its positive and negative influence will depend on the positive and negative influence probabilities of the neighbors' of node v. The damping factor of a node $v \in V(\mathcal{G})$ is df_v defined as $1/\text{in-degree}(v)$ where $\text{in-degree}(v) = |\{u \in V(\mathcal{G}) \mid (u, v) \in E\}|$. So, given that, we know the positive and negative influence probabilities of the neighbors' of the node v, Equations 1, 2, 3 and 4.

$$\mathcal{W}^+_{\mathcal{S}\to v} = 1, v \in S \tag{1}$$

$$\mathcal{W}^-_{\mathcal{S}\to v} = 0, v \in S \tag{2}$$

$$\mathcal{W}^+_{\mathcal{S}\to v} = df^+_v \left(\sum_{(u,v)\in E^+} W_{u,v} \mathcal{P}^+_{\mathcal{S}\to u} + \sum_{(u,v)\in E^-} W_{u,v} \mathcal{P}^-_{\mathcal{S}\to u} \right), v \notin S \tag{3}$$

$$\mathcal{W}^{-}_{\mathcal{S}\to v} = df^{-}_{v}\left(\sum_{(u,v)\in E^{+}} W_{u,v}\mathcal{P}^{-}_{\mathcal{S}\to u} + \sum_{(u,v)\in E^{-}} W_{u,v}\mathcal{P}^{+}_{\mathcal{S}\to u}\right), v \notin S \quad (4)$$

The calculation of final influence typically would go through multiple iterations. So, for the calculation of positive influence of node $v \notin S$ do as following for $(t+1)$-th iteration:

$$\mathcal{W}^{+,(t+1)}_{\mathcal{S}\to v} = df^{+}_{v}\left(\sum_{(u,v)\in E^{+}} W_{u,v}\mathcal{P}^{+,(t)}_{\mathcal{S}\to u} + \sum_{(u,v)\in E^{-}} W_{u,v}\mathcal{P}^{-,(t)}_{\mathcal{S}\to u}\right), v \notin S \quad (5)$$

Similarly, for the calculation of negative influence of node $v \notin S$ do the following for $(t+1)$-th iteration:

$$\mathcal{W}^{-,(t+1)}_{\mathcal{S}\to v} = df^{-}_{v}\left(\sum_{(u,v)\in E^{+}} W_{u,v}\mathcal{P}^{-,(t)}_{\mathcal{S}\to u} + \sum_{(u,v)\in E^{-}} W_{u,v}\mathcal{P}^{+,(t)}_{\mathcal{S}\to u}\right), v \notin S \quad (6)$$

At the end of the diffusion process, every node of the network contains the positive and negative influence probabilities from the seed set. The diffusion process ends when the influence probabilities of the nodes can not be updated further. In this paper, we study the profit maximization problem in the context of signed social networks. Subsequently, we define this problem formally.

2.3 Profit Maximization Problem in Signed Social Networks Problem

We consider that all the users of the network are associated with their respective cost and benefit values. In practice, the cost of a node signifies the incentive demand of that user, i.e., if that user is chosen as a seed user, how much incentive needs to be given to that user? On the other hand, the benefit associated with the user signifies the amount of benefit that can be earned. These are formalized as cost and benefit functions, and they are mathematically represented as $\mathcal{C} : V(\mathcal{G}) \longrightarrow \mathbb{R}^{+}$, and $b : V(\mathcal{G}) \longrightarrow \mathbb{R}^{+}$, respectively. For any user u of the network, its cost and benefit are denoted by $\mathcal{C}(u)$ and $b(u)$, respectively. For any subset of the nodes $S \subseteq V$, the cost of S is denoted by $\mathcal{C}(S)$ and defined as $\mathcal{C}(S) = \sum\limits_{u\in S} \mathcal{C}(u)$. Consider any arbitrary seed set $S \subseteq V(\mathcal{G})$ which has been used for the diffusion process. Consider for any node u, $\mathcal{W}^{+}_{S\to u}$ and $\mathcal{W}^{-}_{S\to u}$ are the positive and negative influence probabilities of the node u from the seed set S. Now, we state the notion of earned benefit by a seed set in the context of signed social networks in Definition 1.

Definition 1 (Earned Benefit by a Seed Set). *Given a signed social network $\mathcal{G}(V, E, \mathcal{W}, \mathcal{P})$, and a seed set $S \subseteq V$, the earned benefit by the seed set is defined as the sum of the expected earned benefit of all the nodes of the network by* *the seed set S. It is denoted by $\beta(S)$ and can be mathematically defined using Equation 7.*

$$\beta(S) = \sum_{u \in V} (\mathcal{W}^{+}_{S \rightarrow u} - \mathcal{W}^{-}_{S \rightarrow u}) \cdot b(u) \tag{7}$$

Based on the definition of earned benefit by a seed set, we define the earned profit by a seed set in Definition 2.

Definition 2 (Earned Profit by a Seed Set). *Given a signed social network $\mathcal{G}(V, E, \mathcal{W}, \mathcal{P})$, and a seed set $S \subseteq V$ the Earned Profit by a Seed Set S is denoted by $\phi(S)$ and mathematically defined in Equation 8.*

$$\phi(S) = \beta(S) - \mathcal{C}(S) \tag{8}$$

Our goal is to select a seed set S within the allocated budget such that the earned profit is maximized. Formally, we call this problem as the Profit Maximization in Signed Social Network Problem which has been stated in Definition 3.

Definition 3 (Profit Maximization in Signed Social Networks). *Given a signed social network $\mathcal{G}(V, E, \mathcal{W}, \mathcal{P})$ along with the cost and benefit function $\mathcal{C} : V(\mathcal{G}) \longrightarrow \mathbb{R}^{+}$, and $b : V(\mathcal{G}) \longrightarrow \mathbb{R}^{+}$, respectively and a fixed budget $\mathcal{B}$, this problem asks to choose a subset of the users $S \subseteq V$ as the seed set for initial activation such that the total selection cost of the users is less than or equal to the allocated budget, also the earned profit by the seed set, i.e., $\phi(S)$ is maximized. Mathematically, this problem can be posed as follows:*

$$S^{OPT} \longleftarrow \underset{S \subseteq V \text{ and } \mathcal{C}(S) \leq \mathcal{B}}{argmax} \phi(S) \tag{9}$$

Here, S^{OPT} denotes optimal seed set of budget $\mathcal{B}$ in $\mathcal{G}$. From the computational point of view, this problem can be posed as follows:

PROFIT MAXIMIZATION IN SIGNED SOCIAL NETWORKS

Input: A Signed Social Network $\mathcal{G}(V, E, \mathcal{W}, \mathcal{P})$, a budget $\mathcal{B}$.
Task: Find out the seed set $\mathcal{S} \subseteq V(\mathcal{G})$ such that $\mathcal{C}(\mathcal{S}) \leq \mathcal{B}$ and $\phi(\mathcal{S})$ is maximized.

2.4 Set Functions and Its Properties

It can be observed that both the profit and benefit functions are a mapping from all possible subsets of the vertex set to positive real numbers, including zero. Such functions are called *Set Functions*. A set function f defined on the ground set $\mathcal{X} = \{x_1, x_2, \ldots, x_n\}$ and this is a mapping from all possible subsets of $\mathcal{X}$ to a positive real number including 0; i.e., $f : 2^{\mathcal{X}} \longrightarrow \mathbb{R}^{+}_{0}$. There are several properties of the set function such as non-negativity, monotonicity, submodulariy, etc. A set function f defined on the ground set $\mathcal{X}$ is said to be non-negative if for all $\mathcal{X}^{'} \subseteq \mathcal{X}$, $f(\mathcal{X}^{'}) \geq 0$. f is said to be monotone if for all $\mathcal{X}^{'} \subseteq \mathcal{X}$ and $x \in \mathcal{X}^{'} \setminus \mathcal{X}$, $f(\mathcal{X}^{'} \cup \{x\}) \geq f(\mathcal{X}^{'})$. f is said to be submodular if for all $\mathcal{X}_1 \subseteq \mathcal{X}_2 \subseteq \mathcal{X}$ and $x \in \mathcal{X} \setminus \mathcal{X}_2$, $f(\mathcal{X}_1 \cup \{x\}) - f(\mathcal{X}_1) \geq f(\mathcal{X}_2 \cup \{x\}) - f(\mathcal{X}_2)$.

3 Proposed Solution Approaches

In this section, we describe two proposed solution approaches. The first one is an iterative approach which works based on the marginal profit computation. The other approach detects the community structure first and then applies the incremental greedy approach in each of the communities. Before that we mention some of the properties of the profit and benefit function.

3.1 Properties of the Profit and Benefit Functions

We establish the non-negativity, monotonicty, and sub-modularity, property of the profit and benefit function. Please refer to Figure 1 for the example where $\mathcal{C}(u_1) = 2$ and $\mathcal{C}(u_2) = \mathcal{C}(u_3) = 1$. Also, $b(u_1) = b(u_3) = 1$ and $b(u_2) = 4$.

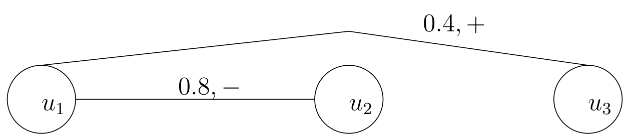

Fig. 1. Example figure to establish the properties of the benefit an profit function

Table 1. Benefit and Profit Calculation from Seed Set S

S	$\mathcal{W}^{\mathcal{E}}_{\mathcal{S}\to u_1}$	$\mathcal{W}^{\mathcal{E}}_{\mathcal{S}\to u_2}$	$\mathcal{W}^{\mathcal{E}}_{\mathcal{S}\to u_3}$	$\beta(\mathcal{S})$	$\phi(\mathcal{S})$
$\{\}$	0	0	0	0	0
u_1	1	-0.8	0.4	-1.8	-3.8
u_2	-0.2	1	-0.08	3.72	2.72
u_3	0.4	-0.32	1	0.12	-0.88
u_1, u_2	1	1	0.4	5.4	2.4
u_1, u_3	1	-0.8	1	5.4	2.4
u_2, u_3	-0.2	1	1	-1.2	-3.2
u_1, u_2, u_3	1	1	1	6	2

Lemma 1. *$\beta(.)$ and $\phi(.)$ may be positive or negative.*

Proof. As shown in Table 1, $S = \{u_2\}$, so, $\beta(S) = 3.72$ and $\phi(S) = 2.72$. In this case, $\beta(S)$ is greater than zero. Again, in the Table 1, $\beta(S) = -1.8$ and $\phi(S) = -3.8$ for $S = \{u_1\}$. Hence, both the functions $\beta(\cdot)$ and $\phi(\cdot)$ may be positive and negative.

Lemma 2. *Both $\beta(\cdot)$ and $\phi(\cdot)$ are non-monotone.*

Proof. The proof is by counter-example. So we provide an example where there exists a $S \subseteq V$ and $u \in V \setminus S$ such that $\beta(S) > \beta(S \cup \{u\})$ and also there exists a $S^{'} \subseteq V$ and $u^{'} \in V \setminus S^{'}$ such that $\phi(S^{'}) > \phi(S^{'} \cup \{u^{'}\})$.

Consider the following example with reference to Figure 1 and Table 1. Let $S = \{u_3\}$, and $u = \{u_2\}$ such that $u \in V(G) \setminus S$. So, $\beta(S) = 0.12$ and $\beta(S \cup \{u\}) = -1.2$ calculated in Table 1. Here, $\beta(S) \geq \beta(S \cup \{u\})$, which means $\beta(\cdot)$ is non-monotone.

Consider the following example. Let $S = \{u_2\}$, and $u = \{u_1\}$. As shown in Table 1, $\phi(S) = 2.72$ and $\phi(S \cup \{u\}) = 2.4$. Here, $\phi(S) > \phi(S \cup \{u\})$ and this means $\phi(\cdot)$ is non-monotone.

Lemma 3. *$\beta(\cdot)$ and $\phi(\cdot)$ is neither submodular nor supermodular.*

Proof. This proof is also by counter example. Let $S = \{\}$, $T = \{u_1\}$ and $u = \{u_2\}$, where $u \notin T$. So, $\beta(S \cup \{u\}) - \beta(S) = 3.72$ and $\beta(T \cup \{u\}) - \beta(T) = 7.2$ are calculated in Table 1. Similarly, Let $S = \{\}$, $T = \{u_2\}$ and $u = \{u_1\}$, where $u \notin T$. So, $\phi(S \cup \{u\}) - \phi(S) = -3.8$ and $\phi(T \cup \{u\}) - \phi(T) = -0.32$ are calculated in Table 1. We observe that submodularity is not followed.

Let $S = \{u_1\}$, $T = \{u_1, u_2\}$ and $u = \{u_3\}$, where $u \notin T$. So, $\beta(S \cup \{u\}) - \beta(S) = 7.2$ and $\beta(T \cup \{u\}) - \beta(T) = 0.6$ are calculated in Table 1. Similarly, Let $S = \{\}$, $T = \{u_1, u_3\}$ and $u = \{u_2\}$, where $u \notin T$. So, $\phi(S \cup \{u\}) - \phi(S) = 2.72$ and $\phi(T \cup \{u\}) - \phi(T) = -0.4$ are calculated in Table 1. We observe that supermodularity for $\beta(\cdot)$ and $\phi(\cdot)$ is not followed.

$\therefore$ Both $\beta(\cdot)$ and $\phi(\cdot)$ are neither submodular nor supermodular.

3.2 Incremental Greedy Approach

For a seed set S, and a non-seed user $u \in V \setminus S$, first we define the notion of *marginal gain* in Definition 4.

Definition 4 (Marginal Profit Gain of a non-seed Node). *Given a signed social network $\mathcal{G}(V, E, \mathcal{W}, \mathcal{P})$ and a seed set S, the marginal profit gain for any non-seed node $u \in V \setminus S$ is denoted by $\Delta(u|S)$ and defined as the difference between the earned profit when u is included in the seed set and when it is not. Mathematically, this can be mathematically posed as follows:*

$$\Delta(u|S) = \phi(S \cup \{u\}) - \phi(S) \tag{10}$$

Based on the definition of marginal profit gain, we describe an incremental greedy approach for our problem. In this approach, we start with an empty seed set, and we select seed nodes in an iterative manner. In each iteration,

Algorithm 1: Incremental Greedy Algorithm for Profit Maximization in Signed Network

Data: Signed Social Network $\mathcal{G}$, Budget $\mathcal{B}$
Result: The Seed Set S with $\mathcal{C}(S) \leq \mathcal{B}$
1 Initialize $S \leftarrow \emptyset$;
2 **while** *True* **do**
3   Find $u^* \leftarrow \underset{u \in V \setminus S \text{ and } \mathcal{C}(u) \leq \mathcal{B}}{argmax} \frac{\phi(S \cup \{u\}) - \phi(S)}{C(u)}$
4   **if** $\Delta(u^*|S) > 0$ **then**
5         $S \leftarrow S \cup \{u^*\}$
6         $\mathcal{B} \leftarrow \mathcal{B} - \mathcal{C}(u^*)$
7   **end**
8 **end**
9 Return S

we select the node that causes the maximum marginal profit gain, and this value is positive. That node is added to the seed set, and the budget is updated accordingly. Algorithm 1 describes this approach in the form of pseudocode.

Now, we analyze Algorithm 1 to understand its time and space requirements. It is important to observe that the running time of Algorithm 1 depends on two things: how many times the `while loop` will execute and given a seed set how much time it will take to finish the diffusion process. It can be observed that in the worst case, the number of times the while loop will be executed will be of $\mathcal{O}(n)$. In the worst case, the size of the seed set can be of $\mathcal{O}(n)$. It can be easily observed that computing the marginal profit gain for any non-seed node based on the PLID Model will take $\mathcal{O}(m \cdot n)$ time. As there can be $\mathcal{O}(n)$ many non-seed nodes, hence the execution of Line 3 in one iteration of the `while` loop will take $\mathcal{O}(m \cdot n^2)$ time. The conditional checking at Line 4 will take $\mathcal{O}(m \cdot n)$ time and the instructions at Line 5 and 6 will take $\mathcal{O}(1)$ time. Hence, the time requirement by Algorithm 1 will be of $\mathcal{O}(n \cdot (m \cdot n^2 + m \cdot n + 1))$ which reduces to $\mathcal{O}(m \cdot n^3)$.

The additional space requirement of Algorithm 1 is to store the set S which is of $\mathcal{O}(n)$, and also to store the marginal profit gains of the non-seed nodes which can also be $\mathcal{O}(n)$. Hence the total space requirement will be of $\mathcal{O}(n)$. Hence, Theorem 1 holds.

Theorem 1. *The computational time and space requirements of Algorithm 1 will be of $\mathcal{O}(m \cdot n^3)$ and $\mathcal{O}(n)$, respectively.*

3.3 A Local Search Algorithm

In this section, we propose a local search-based approach to solve our problem. The working strategy of this approach is as follows: It starts with an initial solution and iteratively updates the solution. Consider i-th iteration which starts with the set S^i, and now we describe how to construct the seed set S^{i+1}. First, we find out the node in S^i whose removal would lead to the least decrease in the profit earned per unit cost. Let the node be $u^{'}$. So, we do the computation as mentioned in Equation 11.

$$u^{'} \longleftarrow \underset{u \in S^i}{argmin} \frac{\phi(S^i) - \phi(S^i \setminus \{u\})}{\mathcal{C}(u)} \tag{11}$$

Accordingly, the current seed set S^i is updated as $S^i \setminus \{u^{'}\}$. As the node $u^{'}$ is excluded from the current seed set, hence the cost of amount $\mathcal{C}(u^{'})$ is available. On the other hand, we find out a node that is not in S^i (i.e., from the set $V(\mathcal{G}) \setminus S^i$) such that its inclusion in the updated seed set leads to maximum gain in the earned profit. Let the node be $u^{''}$. So, we do the computation mentioned in Equation 12.

$$u^{''} \longleftarrow \underset{u \in V(\mathcal{G}) \setminus S^i, \mathcal{C}(u) \leq \mathcal{C}(u^{'})}{\arg\max} \frac{\phi(\mathcal{M} \cup \{u\}) - \phi(\mathcal{M})}{\mathcal{C}(u)} \quad \text{where} \quad \mathcal{M} = (S^i \setminus \{u^{'}\}) \tag{12}$$

The node $u^{''}$ is included in the seed set. Now, it can be observed that the cost of the new seed set obtained by removing the node $u^{'}$ and adding the node $u^{''}$ will always be less than the allocated budget $\mathcal{B}$. Let us call the new seed set as $S^{'}$. It can be observed that the set $S^{'}$ will be the seed set at the beginning of the $(i+1)$-th. Hence, $S^{i+1} = S^{'}$. If, $\phi(S^{'}) > \phi(S)$, that means the profit earned by the new seed set at the end of i-th iteration is more than the seed set that we have started with at the beginning of the i-th iteration. So, the quality of the new seed set is better than the old one. Hence, the new seed set has been updated as the current seed set. This process has been iterated as long as the improvement in the quality of the seed set is possible. Algorithm 2 gives the pseudocode of the proposed solution approach.

From the description of this algorithm, it is evident that its running time and its performance will be dependent on how we generate the initial solution. In this study, we have used the following two methods to generate the initial solution. In the first method, we randomly pick up nodes from the nodes which are not already there in the seed set as long as budget permits. In the other method, we pick up high degree nodes with in the allocated budget. The observations have been demonstrated in the experimental section.

Now, we analyze Algorithm 2. Consider for an initialization process for the seed set, the time requirement is of $\mathcal{O}(X)$. The statements in Line 2 of Algorithm 2 will be of $\mathcal{O}(1)$ time. Now, question arises how many times the `while` loop at Line 3 runs. Consider this is $\mathcal{O}(itr\_num)$. In the worst case, the size of cur_sol and new_sol will be of $\mathcal{O}(n)$. To find the node $u^{'}$, there can be $\mathcal{O}(n)$ many marginal profit gain computations. Each one of them will take $\mathcal{O}(m \cdot n^2)$ time. Hence, computational time requirement to find the node $u^{'}$ will be of $\mathcal{O}(m \cdot n^2)$. It is easy to verify that the time requirement to execute the instructions mentioned in Line 6 and 7 will take $\mathcal{O}(1)$ time. Similar way, finding the node $u^{''}$ will take $\mathcal{O}(m \cdot n^2)$ time. All the remaining statements will take $\mathcal{O}(1)$ time to execute. Hence, the computational time requirement of Algorithm 2 will be of $\mathcal{O}(X + itr\_num \cdot (1 + m \cdot n^2 + 1 + m \cdot n^2 + 1))$ which is reduced to $\mathcal{O}(X + itr\_num \cdot (m \cdot n^2))$. As mentioned previously, X is the time requirement of the method which has

Algorithm 2: Local Search Algorithm for Profit Maximization in Signed Networks

Data: Signed Social Network $\mathcal{G}$, Initial Seed Set S_0, Budget $\mathcal{B}$
Result: The Seed Set S with $\mathcal{C}(S) \leq \mathcal{B}$

```
S_0 ⟵ Generate an Initial Solution such that C(S_0) ≤ B
Curr_Sol ⟵ S_0, B' ⟵ B − C(S_0)
while True do
    new_Sol ⟵ Curr_Sol
    Find u' ⟵ argmin_{u ∈ Curr_Sol} (φ(Curr_Sol) − φ(Curr_Sol \ {u})) / C(u)
    new_Sol ⟵ Curr_Sol \ {u'}
    B' ⟵ B' + C(u')
    Find u'' ⟵ argmax_{u ∈ V(G) \ new_Sol, C(u) ≤ C(u')}
        (φ((new_Sol \ {u'}) ∪ {u}) − φ((new_Sol \ {u'}))) / C(u)
    new_Sol ⟵ new_Sol ∪ {u''}
    if φ(new_Sol) > φ(Curr_Sol) then
        Curr_Sol ⟵ new_Sol
    else
        break,
    end
end
Return S
```

been used to generate the initial seed set. In the RANDOM Method, we randomly choose a node from the set of non-seed nodes and include it to the seed set until the budget is exhausted. This method will take $\mathcal{O}(n)$ time to execute. In the HIGH DEGREE Method, first the nodes are sorted based on its degree value and then high degree nodes are chosen as seed nodes until the budget is exhausted. This method will take $\mathcal{O}(n^2)$ time to generate the initial seed set. Hence, the total time requirement of Algorithm 2 will be of $\mathcal{O}(itr\_num \cdot m \cdot n^2)$. Additional space requirement of Algorithm 2 is to store the sets new_Sol, $Curr\_Sol$ and to store the marginal profit gains of the non seed nodes. All of them will take $\mathcal{O}(n)$ in the worst case. Hence, Theorem 2 holds.

Theorem 2. *Irrespective of RANDOM or HIGH DEGREE Method for generating the initial seed set, Algorithm 2 will take $\mathcal{O}(itr\_num \cdot m \cdot n^2)$ computational time and $\mathcal{O}(n)$ space.*

4 Experimental Evaluation

In this section, we describe the experimental evaluation of the proposed solution approach. Initially, we start by describing the datasets.

Dataset Description The following datasets have been used in our experiments.

- **Bitcoin Alpha**: This is who-trusts-whom network of people who trade using Bitcoin on a platform called **Bitcoin Alpha**. Members of Bitcoin Alpha rate other members on a scale of -10 (total distrust) to +10 (total trust) in steps of 1. This is the first explicit weighted signed directed network available for research. This dataset has 93% positive edges, and the remaining are negative. [10, 11]
- **Bitcoin-OTC**: This is who-trusts-whom network of people who trade using Bitcoin on a platform called **Bitcoin OTC**. The rating of members is done likewise Bitcoin Alpha. This dataset has 89% positive edges and remaining are negative edges. [10, 11]

Table 2 lists the datasets used in our experiments. These datasets are available to download at https://snap.stanford.edu/.

Table 2. Signed Social Network Dataset Statistics

Dataset	Number of Nodes, $\|V\|$	Number of Edges, $\|E\|$	Number of Positive Edges, $\|E^+\|$	Number of Negative Edges, $\|E^-\|$	Maximum Degree, Δ	Average Degree, δ
Bitcoin Alpha	3783	24186	22650	1536	888	12.79
Bitcoin OTC	5881	35592	32029	3563	1298	12.10

Experimental Setup For any edge $e \in E(\mathcal{G})$, $\mathcal{W}(e)$ and $\mathcal{P}(e)$ denote the influence probability and the polarity of the edge, respectively. Our datasets come with only the corresponding polarity of the edges in the adjacency list. Since the influence probability is not available, we use the following popular methods for the edges in the graphs of our experiments.

- **Uniform**: In this method, we assign the same diffusion probability to all edges in the graph. For our experiments, we have used a probability value 0.01.
- **Weighted Cascade**: The in-degree $d(v)$ of a node v is utilized to generate the diffusion probabilities. So, the diffusion probability $\mathcal{P}_{\mathcal{S}\rightarrow v}$ for an edge (u, v) is $1/d(v)$.
- **Trivalency**: This method has a set of three diffusion probabilities, and for every edge (u, v), one of the diffusion probabilities is randomly selected and assigned to it. The three diffusion probabilities for our experiments are $\{0.1, 0.01, 0.001\}$.

Cost and Benefit: A node's cost and benefit are assigned randomly from the intervals [50, 100] and [800, 1000], respectively.

Methods Compared We compare the performance of the proposed solution approaches with the following approaches:

- **Incremental Greedy Algorithm:** In the Incremental Greedy Algorithm, we iteratively add a node to the seed set on the positive marginal gain. This method is proposed in this paper in detail.
- **Local Search Algorithm:** It is our proposed method explained in Algorithm 2 in Section 3.
- **ICP Greedy:** This method follows the original Independent Cascade Model [7]. It adopts a greedy algorithm for profit maximization.
- **High Degree:** A baseline method that selects top-ℓ nodes based on the high degree within the limits of the allocated budget $\mathcal{B}$.
- **Random:** This method randomly selects the ℓ random nodes from the signed graph within the limits of the allocated budget $\mathcal{B}$.

Goals of the Experimentation Now we describe the goals of the experimentation:

- **Research Question 1:** Once the budget value increases, how does the earned profit increase for different methodologies (proposed as well as baseline) across different diffusion probability settings?
  Answer: With the increase of the budget, the profit earned increases. For instance, in Figure 2 (a), in LS - Random algorithm, profit earned for budget 500 is 9432, and for budget value 2000 it is 29621. Likewise, for all baseline algorithms, we observe that with an increase in the budget, the profit earned also increases.
- **Research Question 2:** How does the quality of the initial solution affect the quality of the final solution for our local search-based approach?
  Answer: In Figure 4, the initial profit earned is at iteration no. 1, and with every next iteration, the profit earned increases till the profit earned is not increased at some iteration.
- **Research Question 3:** Once the budget increases, how does the computational time requirement for the seed set selection change for different solution methodologies?
  Answer: With the increase in the budget, the computational time also increases. This observation is applicable to all the solution methodologies shown in Figure 3.

Results and Discussions In this section, we describe the experimental results and answer all the research questions mentioned previously. Figure 2 (a)-(e) shows how the earned profit changes when the budget value increases for different datasets and influence probability settings. Among the proposed solution approaches, in the uniform probability setting Figure 2 (a) & (d), the seed set selected by the single greedy approach leads to more profit earned compared to the other methods. Among the existing methods, the seed set selected by the ICP-Greedy led to the maximum profit for the weighted cascade setting shown in Figure 2 (b) & (e). As an example, for the budget of 2500, the profit earned by the seed set selected by the single greedy and ICP-Greedy is 558723.83

and 673897.10, respectively, which is 16% more compared to the best baseline method. In the Figure 2 (c) & (f) trivalency setting, among the proposed solution approaches, the local search-based approach where the initial solution has been generated by choosing high degree nodes, the final profit earned is maximum compared to the baseline methods.

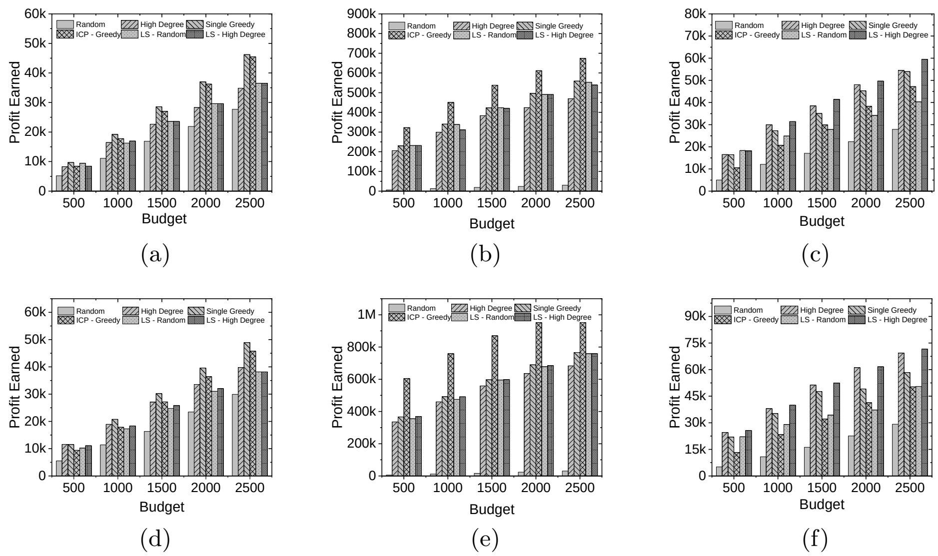

Fig. 2. Budget Vs. Profit Earned Plots:- Influence Probability: (a) Uniform (0.01), (b) Weighted Cascade & (c) Trivalency for Bitcoin Alpha dataset & Influence Probability: (d) Uniform (0.01), (e) Weighted Cascade & (f) Trivalency for Bitcoin-OTC dataset

Another observation is regarding the computational time of baselines and proposed solution approaches shown in Figure 3. In Figure 2 (a), it can be seen that by the local search-based approach selecting initial solution using random and high degree method profit earned is $2-20\%$ and $12-20\%$, respectively, less than the single greedy algorithm. For the same comparison setting, in Figure 3, the computational time is $25-45\%$ and $65-69\%$, respectively more in the single greedy approach. Next, in Figure 2 (b) for the weighted cascade setting, we observe that ICP-Greedy earns the highest profit, but it takes a maximum of 99% more time to give at most 30% more profit compared to the local search-based approach can be seen on Figure 3. In the trivalency setting shown in Figure 3 (c), the single greedy approach takes $61-96\%$ more time and earns $9-18\%$ less profit than the local search-based approach (Figure 2(c)). Therefore, the observation from Figure 3 (a)-(c) for the bitcoin alpha dataset is that the local search-based high degree method takes less time compared to the single greedy and ICP-greedy approach. For bitcoin-otc dataset, in Figure 3, the single greedy algorithm takes 93% more time to calculate the profit earned with a

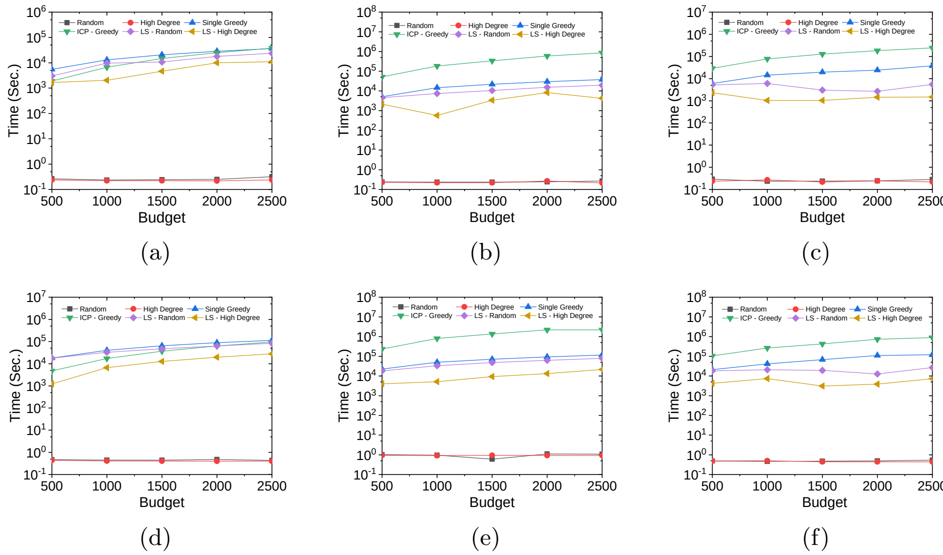

Fig. 3. Budget Vs. Time (Sec.) Plots:- Influence Probability: (a) Uniform (0.01), (b) Weighted Cascade & (c) Trivalency for Bitcoin Alpha dataset & Influence Probability: (d) Uniform (0.01), (e) Weighted Cascade & (f) Trivalency for Bitcoin-OTC dataset

trade-off of 3% decrease only in the profit earned by the seed set from the local search-based approach where the initial solution is generated by choosing high degree nodes for budget 500. Similarly, the ICP-Greedy approach takes $61-74\%$ more time than the local search-based approach selecting nodes in the initial solution from the high degree nodes. With this mentioned computational time, we found a maximum 16% decrease in profit earned by our proposed local search-based approach. Figure 3 (c) shows a weighted cascade setting, in which we can see the single greedy approach taking $18-31\%$ more time compared to our local search-based approach selecting the initial solution by the random method. However, the profit earned by our mentioned approach produces a maximum of 3% less profit than the single greedy approach. So, we can use a local search-based approach when the time is crucial to calculate profit. Though the ICP-Greedy approach produces the highest profit, the computational time is $92-97\%$ more than the local search-based approach selecting seed set nodes by random method, with the $20-41\%$ decrease in the profit earned. The same can be said about the local search-based approach selecting seed set nodes by a high degree method compared to the ICP-Greedy approach. Figure 3 (f) shows the trivalency setting, where the single greedy approach takes $79-96\%$ more time but gives $10-25\%$ less profit than the proposed local search-based approach. Similarly, ICP-Greedy takes 95% more time but earns 93% less profit than the local search-based approach having seed set nodes in the initial solution from the high degree method. Our proposed local search-based approach is analyzed to understand the improvement in the profit earned from the initial solution to the final solution. The final solution is a set of nodes from which the profit earned can not be improved further. It can be seen in Figure 4 for budget value 500 that with every next iteration, the profit earned increases. The initial solution generated from random and high degree method is further processed by the local search-approach, giving promising results in every next iteration. For instance in Figure 4 (d), the profit earned by local search-based random method at iteration no. 1 is 4597.2 and at iteration no. 8 it is 10209.5. This profit earned is not increased in the next iteration, so it continues to be the same at iteration no. 9. The profit earned by the high degree method at iteration no. 1 is 334840.6 which later increased to 369048.7 by the proposed approach at iteration no. 4. Overall, it can be seen that the high degree method gives more profit in most cases compared to the random method in less number of iterations, both performed under the proposed local search-based approach.

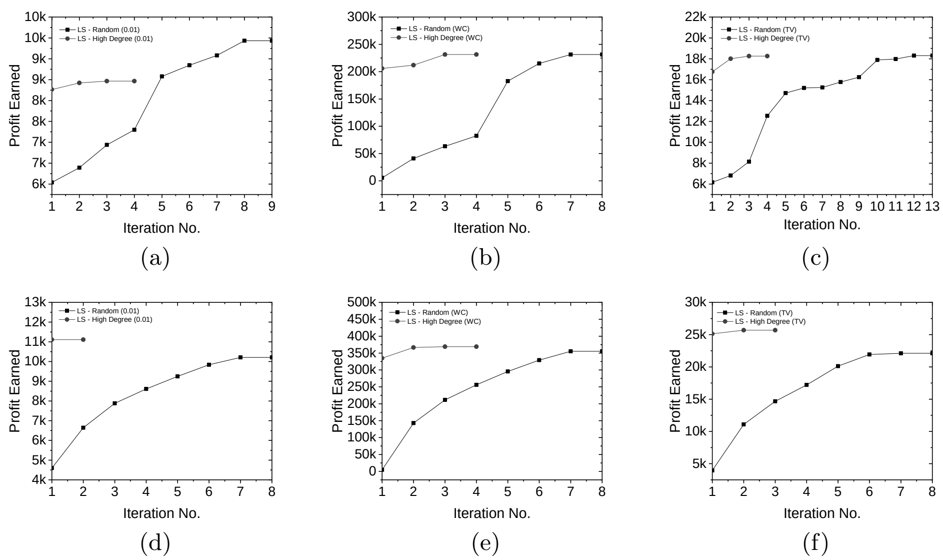

Fig. 4. Iteration No. Vs. Profit Earned Plots:- Influence Probability: (a) Uniform (0.01), (b) Weighted Cascade & (c) Trivalency for Bitcoin Alpha dataset & Influence Probability: (d) Uniform (0.01), (e) Weighted Cascade & (f) Trivalency for Bitcoin-OTC dataset

5 Concluding Remarks

In this paper, we have studied the problem of profit maximization in the context of a signed social network under the Polarity Linked Information Diffusion

Model. We observed that both the benefit and profit functions may be monotone or non-monotone, submodular or non-submodular. We have proposed two approaches to solve the problem namely, incremental greedy and local search approaches to solve the problem. The proposed approaches have been analyzed to obtain their time and space complexities. We have conducted a large number of experiments with real-world signed social network datasets and the results are reported. We have observed that the proposed solution approaches lead to more amount of profit compared to the existing methods with reasonable computational overhead.

References

1. M. A. Shareef, B. Mukerji, Y. K. Dwivedi, N. P. Rana, and R. Islam, "Social media marketing: Comparative effect of advertisement sources," *Journal of Retailing and Consumer Services*, vol. 46, pp. 58–69, 2019.
2. F. Li, J. Larimo, and L. C. Leonidou, "Social media in marketing research: Theoretical bases, methodological aspects, and thematic focus," *Psychology & Marketing*, vol. 40, no. 1, pp. 124–145, 2023.
3. E. Bakshy, I. Rosenn, C. Marlow, and L. Adamic, "The role of social networks in information diffusion," in *Proceedings of the 21st international conference on World Wide Web*, 2012, pp. 519–528.
4. P. Shakarian, A. Bhatnagar, A. Aleali, E. Shaabani, R. Guo, P. Shakarian, A. Bhatnagar, A. Aleali, E. Shaabani, and R. Guo, "The independent cascade and linear threshold models," *Diffusion in Social Networks*, pp. 35–48, 2015.
5. L. S. Alla and A. S. Kare, "Opinion maximization in signed social networks using centrality measures and clustering techniques," in *International Conference on Distributed Computing and Intelligent Technology*. Springer, 2023, pp. 125–140.
6. M. Hosseini-Pozveh, K. Zamanifar, and A. R. Naghsh-Nilchi, "Assessing information diffusion models for influence maximization in signed social networks," *Expert Systems with Applications*, vol. 119, pp. 476–490, 2019.
7. D. Li, Z.-M. Xu, N. Chakraborty, A. Gupta, K. Sycara, and S. Li, "Polarity related influence maximization in signed social networks," *PloS one*, vol. 9, no. 7, p. e102199, 2014. [Online]. Available: https://doi.org/10.1371/journal.pone.0102199
8. D. Li and J. Liu, "Modeling influence diffusion over signed social networks," *IEEE Transactions on Knowledge and Data Engineering*, vol. 33, no. 2, pp. 613–625, 2019.
9. J. Tang, X. Tang, and J. Yuan, "Profit maximization for viral marketing in online social networks: Algorithms and analysis," *IEEE Transactions on Knowledge and Data Engineering*, vol. 30, no. 6, pp. 1095–1108, 2017.
10. S. Kumar, F. Spezzano, V. Subrahmanian, and C. Faloutsos, "Edge weight prediction in weighted signed networks," in *Data Mining (ICDM), 2016 IEEE 16th International Conference on*. IEEE, 2016, pp. 221–230.
11. S. Kumar, B. Hooi, D. Makhija, M. Kumar, C. Faloutsos, and V. Subrahmanian, "Rev2: Fraudulent user prediction in rating platforms," in *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining*. ACM, 2018, pp. 333–341.