Received April 19, 2005; revised August 26, 2015.

Abstract—Given a social network with nonuniform selection cost of the users, the problem of Budgeted Influence Maximization (BIM in short) asks for selecting a subset of the nodes within an allocated budget for initial activation, such that due to the cascading effect, influence in the network is maximized. In this paper, we study this problem with a variation, where a set of nodes are designated as target nodes, each of them is assigned with a benefit value, that can be earned by influencing them, and our goal is to maximize the earned benefit by initially activating a set of nodes within the budget. We call this problem as the Earned Benefit Maximization Problem. First, we show that this problem is NP-Hard and the benefit function is monotone, sub-modular under the Independent Cascade Model of diffusion. We propose an incremental greedy strategy for this problem and show, with minor modification it gives \((1-1/e)\)-factor approximation guarantee on the earned benefit. Next, by exploiting the sub-modularity property of the benefit function, we improve the efficiency of the proposed greedy algorithm. Then, we propose a hop-based heuristic method, which works based on the computation of the ‘expected earned benefit’ of the effective neighbors corresponding to the target nodes. Finally, we perform a series of extensive experiments with four real-life, publicly available social network datasets. From the experiments, we observe that the seed sets selected by the proposed algorithms can achieve more benefit compared to many existing methods. Particularly, the hop-based approach is found to be more efficient than the other ones for solving this problem.

Index Terms—Social Network, Earned Benefit, Target Nodes, Greedy Algorithm, Effective Nodes.

I. INTRODUCTION

Social Networks are an interconnected structure among a group of agents [1]. These are effective platforms, where word-of-mouth effect happens at a large scale and information, ideas, rumors etc. disseminates widely and rapidly [2], [3]. This phenomenon has been exploited by the E-Commerce houses for promoting their brands among people [4], [5]. The key problem is that which users initially to choose for initiating the diffusion process such that the influence in the network gets maximized. Formally this problem is called as the Social Influence Maximization Problem [6]. Due to the wider applications in different domains such as viral marketing [7], social recommendation [8], market basket analysis [9], prediction of hot topics [10] this problem has been studied in different variations. Please, look into [11] [12] for recent surveys. The social influence happens due to the cascading process in the underlying network [13], [14], and this has huge impact, because human decisions from personal (which place to visit and which restaurant to explore?) to political (which political party to vote in the coming election?) are influenced by their neighbors, at least to some extent. To study the diffusion process in a social network several diffusion models have been studied. Please look into [15] for recent survey.

One of the recently introduced variants of the SIM Problem is the problem of Budgeted Influence Maximization [16]. This problem assumes that users of the network have nonuniform selection cost, which signifies the amount of incentive need to be paid if a user is selected as a seed node. A fixed amount of budget is allocated for the seed set selection process, and the job is to choose highly influential seed nodes within the budget to maximize the influence. There are a few solution methodologies available in the literature for this problem, such as directed acyclic graph-based heuristic by Nguyen et al. [16], Sample Average Aggregate Scheme by Guney et al. [17], ComBIM by Banerjee et al. [18]. In all these studies, it is implicitly assumed that the influencing each user is equally important, though commercial campaigns are targeted in nature, which means a specific brand is to be advertised towards a specific set of users. Because, advertising a brand towards a set of people who do not have any interest towards it does not make any sense. On the other hand, there are some studies in the literature that consider the target user in the influence maximization process [19] [20] [21]. To the best of our knowledge, none of the targeted influence maximization studies considers non-uniform selection cost of the users.

In target advertisement scenarios, influencing different target user leads to the different amount of benefit and from the advertisers perspective, the main goal is to maximize the total earned benefit. Motivated by this practical scenario, we study the Earned Benefit Maximization Problem (EBM Problem), where the target users are associated with a benefit value, users are associated with a selection cost, and a fixed budget is given. The goal is to select a seed set within the budget to maximize the earned benefit. There are previous studies on this problem by us. In [22], we came up with an integer programming formulation for this problem. In [23], [24], we proposed a ranking approach with the exploitation of community structure for this problem. However, in this study our approach to this problem is very different. We start by studying the properties of the earned benefit function
and propose a number of solutions followed by experiments. Particularly, we make the following contributions in this paper.

- We extend the BIM Problem by considering the notion of target users with non-uniform benefit values and propose the ‘Earned Benefit Maximization Problem’.
- For the EBM Problem, we propose an ‘incremental greedy strategy’ and show with minor modification, this methodology leads to \((1 - \frac{1}{\sqrt{e}})\) factor approximation guarantee on the earned benefit.
- We show that the benefit function is monotone and sub-modular under IC Model of diffusion and exploit this property to improve the efficiency of the incremental greedy algorithm.
- Using the concept of ‘expected earned benefit’ of a node, we propose an efficient hop-based heuristic for solving this problem.
- We conduct a set of extensive experiments with four real-world publicly available social network datasets for showing the effectiveness and efficiency of the proposed methodologies.

Rest of the paper is organized as follows. Section II describes some of the recent studies from the literature. Section III presents preliminary definitions, formally defines the EBM Problem and state its hardness result. The proposed methodologies for solving this problem have been described in Section IV. Section V contains the experimental evaluations of the proposed methodologies, and finally, in Section VI, we conclude this study and give the future directions.

II. RELATED WORK

In this section, we present some closely related studies from the literature. This study closely related with the SIM Problem and its variants and more particularly for the targeted users.

a) Social Influence Maximization and Its Variants: Given a social network of users, which nodes should be chosen for initially injecting the information that causes the maximum influence in the network? This problem is known as the social influence maximization. Initially, this problem was identified in the context of viral marketing by Damingos and Richardson [25]. However, Kempe et al. [6] were the first to investigate the computational issues of this Problem and proved that it is NP-Hard and proposed an incremental greedy algorithm, which admits \((1 - \frac{1}{e})\)-factor approximation ratio. Their study triggers a vast amount of research on the SIM Problem and hence, a plenty of solution methodologies are available in the literature, such as Cost-Effective Lazy Forward (CELF) [26], CELF++ [27], SIMPATH [28], Two-Phase Influence Maximization (TIM) [29], Influence Maximization Via Martingales (IMM) [30], Influence Ranking and Influence estimation (IRIE) [31], different community-based solution methodologies [32] [33], different non-traditional optimization algorithms such as genetic algorithm [34], discrete particle swarm optimization [35] and many more. Also, there are several variants of this problem studied in the literature, such as \(\lambda\)-coverage problem [36], budgeted influence maximization problem [16], [18] and many more.

b) Social Influence Maximization for the Targeted Users: Recently, the problem of influence maximization for the targeted users in the social network has been addressed by the researchers. Li et al. [19] studied this problem, and they considered target users, who are relevant to a particular keyword. There solution methodology for this problem was based on the construction of reverse influence set and its indexing. Song et al. [37] addressed the targeted influence maximization problem by considering the geographical location of the users and the time deadline within which the users should be influenced. Recently, Wen et al. [38] studied this problem focusing on mainly two issues: how to capture the social influence among the target user and develop an efficient scheme that can offer the wider influence spread among the target users. Wang et al. [39] also solved the same problem by considering the impact of budget on the influence spread and incorporating efficient sampling techniques.

However, to the best of the authors’ knowledge, none of the existing studies on targeted influence maximization problem considers the nonuniform benefit associated with each target users and the nonuniform selection cost of users. In this paper, we study the Earned Benefit Maximization Problem, where the target users are associated with non-uniform benefit value and non-uniform selection cost of the users.

III. BACKGROUND AND PROBLEM DEFINITION

We consider that the social network is represented as a weighted graph \(G(V, E, P)\), where the vertex set \(V(G)\) is the set of users of the network and the edge set, \(E(G)\) represents the set of social ties among the users. \(P\) is the edge weight function that assigns each edge to its influence probability, i.e., \(P : E(G) \rightarrow (0, 1)\). For any edge \((u_i u_j) \in E(G)\), we denote its influence probability as \(P_{u_i, u_j}\). This signifies the probability that the user \(u_i\) will be able to influence \(u_j\). We denote the number of nodes and edges of \(G\) by \(n\) and \(m\), respectively. Next, we briefly describe the Independent Cascade Model, which we consider as the underlying diffusion model in our study.

A. The Independent Cascade Model

The Independent Cascade Model is one of the models, which has been predominantly used in influence maximization literature [6], [29], [30]. Here, the diffusion of information starts from a set of nodes selected initially and known as the seed nodes. All the nodes of the network are ignorant of the information and the seed nodes are informed at time \(t = 0\). Now, from these seed nodes, information is diffused by the following rules:

- information is diffused in discrete time steps,
- a node can be either one of the two states: ‘active’ (‘influenced’) or inactive (‘uninfluenced’),
- a node can change its state from inactive to active however, not the vice-versa.
- once a node is influenced, it will remain in this state.

Each active node (say, \(u_i\)) at current time stamp (say \(t\)) will get a chance to activate its currently inactive neighbors \((u_j \in N(u_i))\) and \(u_j\) is inactive) with probability as their edge
weight. If any one of them succeeds, then \( u_j \) will become an active node at time \( t+1 \). Only the recently active node can take part in the triggering process. This process stops when no more node activation is possible. Next, we introduce the Earned Benefit Maximization Problem.

### B. The Earned Benefit Maximization Problem

In this problem, along with the social network \( G(V,E,P) \), a subset of the users \( D \) is given as the target users. Each of them is associated with a benefit, which can be earned by influencing the corresponding target user. This can be characterized by the benefit function \( b : D \rightarrow \mathbb{R}^+ \). For any \( u \in D \) his benefit is denoted as \( b(u) \) and for any \( u \notin D \), \( b(u) = 0 \). For a seed set \( S \), the set of nodes influenced by it is denoted by \( I(S) \). As the diffusion of information under IC Model is a probabilistic process, the influence of a seed set is measured in terms of expectation. Hence, the number of influenced nodes due to the seed set \( S \) is \( \sigma(S) = E[I(S)] \), where, \( \sigma(.) \) is the social influence function [6]. Now, the earned benefit by the seed set \( S \) is defined as \( \beta(S) = \sum_{u \in D \cap I(S)} b(u) \). Here, \( \beta(.) \) is the earned benefit function, that maps each subset of the nodes to is expected earned benefit value, i.e., \( \beta : 2^V \rightarrow \mathbb{R}_{\geq 0} \).

In real-world campaigns, earned benefit maximization is done by conducting an information diffusion process. As the real-life social networks are formed by the rational human agents, if a user is selected as seed, incentivization is required. This can be characterized by the cost function \( C : V(G) \rightarrow \mathbb{R}^+ \). Selection cost associated with the node \( u \) is denoted as \( C(u) \). For a subset of nodes \( S \), their selection cost is denoted as \( C(S) = \sum_{u \in S} C(u) \), and a fixed amount of budget \( B \) is allocated for seed set selection. Hence, the problem here is to choose a subset \( S \) from \( V(G) \) to maximize the function \( \beta(S) \) subject to the constraint \( C(S) \leq B \). Formally, the problem can be expressed as follows:

\[
\text{EBM Problem} \\
\text{Input:} \text{Social Network } G(V,E,P), \text{Target Nodes } D, \text{Cost Function } C, \text{Benefit Function } b, \text{and Budget } B. \\
\text{Problem:} \text{Find out the seed set } (S) \text{ such that } \sum_{u \in S} C(u) \leq B \text{ and for any other seed set } S' \text{ with } \sum_{u \in S'} C(u) \leq B, \beta(S) \geq \beta(S').
\]

The EBM problem is basically the generalization of the BIM problem, which is NP-Hard under the IC Model of diffusion [16]. Hence, Theorem 1 holds.

#### Theorem 1. The EBM Problem is NP-Hard under Independent Cascade Model of diffusion.

This result motivates us to design suitable approximation algorithm and heuristic solution for this problem. We discuss them in the next section.

### IV. PROPOSED METHODOLOGY

In this section, we present our proposed methodologies for the EBM problem. Prior to that, we establish two properties of the benefit function, which will be used subsequently.

#### A. Properties of the Benefit Function

As mentioned previously, the benefit earned by a given seed set \( S \) is defined as \( \beta(S) = \sum_{u \in D \cap I(S)} b(u) \). So, the benefit function can be thought of as a set function, which is defined on the ground set \( V(G) \), i.e., \( \beta : 2^V \rightarrow \mathbb{R}^+ \). Now, we prove two important properties of the benefit function, namely monotonicity and sub-modularity. These two properties are exploited for proving the approximation guarantee of Algorithm 2.

**Definition 1** (Non-negativity and Monotonicity of Set Function). A set function \( f(.) \) defined over the ground set \( V(G) \) is said to be non-negative if \( \forall S \subseteq V(G) \), \( f(S) \geq 0 \) and monotone if \( \forall S \subseteq T \), \( f(S) \leq f(T) \).

**Lemma 1.** The benefit function, \( \beta(.) \) is non-negative and monotone under IC Model of diffusion.

**Proof.** It is reported in the literature that the social influence function is non-negative and monotone [6]. As \( \forall u \in V(G) \), \( b(u) \geq 0 \), it is trivial to observe that \( \forall S \subseteq V(G) \), \( \beta(S) \geq 0 \). By the monotonicity property of \( \sigma(.) \), \( \forall S \subseteq S' \),

\[
I(S) \subseteq I(S') \\
\Rightarrow D \cap I(S) \subseteq D \cap I(S') \\
\Rightarrow \sum_{u \in D \cap I(S)} b(u) \leq \sum_{u \in D \cap I(S')} b(u) \\
\Rightarrow \beta(S) \leq \beta(S')
\]

which means \( \beta(.) \) is monotone. This completes the proof. \( \square \)

**Definition 2** (Sub-modularity of Set Function). A set function \( f(.) \) defined over the ground set \( V(G) \) is said to be sub-modular if \( \forall S \subseteq T \subset V(G) \) and \( \forall u \in V(G) \setminus T \), the following condition is met:

\[
f(S \cup \{u\}) - f(S) \geq f(T \cup \{u\}) - f(T)
\]

**Lemma 2.** The benefit function, \( \beta(.) \) is sub-modular under IC Model of diffusion.

**Proof.** In the literature, it is mentioned that the social influence function, \( \sigma(.) \) is sub-modular under the IC Model of diffusion [6]. Let us assume that \( S \subseteq T \subseteq V(G) \) and \( u \in V(G) \setminus S \). Now, from the definition of \( \beta(.) \), we have

\[
\beta(S \cup \{u\}) - \beta(S) = \sum_{u \in D \cap I(S \cup \{u\})} b(u) - \sum_{u \in D \cap I(S)} b(u)
\]

By simple set theoretic interpretation, we can write

\[
I(S \cup \{u\}) = (I(S) \cup I(\{u\})) \setminus (I(S) \cap I(\{u\})) \\
\Rightarrow D \cap I(S \cup \{u\}) = (D \cap I(S)) \cup (D \cap I(\{u\})) \setminus (D \cap I(S \cap I(\{u\}))
\]

Hence,

\[
\beta(S \cup \{u\}) - \beta(S) = \sum_{u \in D \cap I(S \cup \{u\})} b(u) - \sum_{u \in D \cap I(S)} b(u) - \sum_{u \in D \cap I(\{u\})} b(u) + \sum_{u \in D \cap I(S \cap I(\{u\}))} b(u) \\
\Rightarrow \beta(S \cup \{u\}) - \beta(S) = \sum_{u \in D \cap I(\{u\})} b(u) - \sum_{u \in D \cap I(S \cap I(\{u\}))} b(u)
\]
We obtain the inequality required to show sub-modularity of the Algorithm 1 is simple to understand, it does not give any bounded approximation guarantee on the earned benefit and we demonstrate this claim with an example.

**Example 1.** Let us assume, a network with \( p + 1 \) nodes \( V(G) = \{u, v_1, v_2, \ldots, v_p\} \), where \( u \) is an isolated node and the remaining nodes connected within themselves with each edge having the diffusion probability 1. The entire vertex set of the network is the target node set, i.e., \( D = V(G) \). Benefit associated with each target node is 1. For each \( v_i \), its associated selection cost is \( p \) and the selection cost of \( u \) is \((1-\epsilon)\), where \( 0 < \epsilon < 1 \). The allocated budget for the seed set selection process is \( p \). The optimal algorithm for this problem should select any \( v_i \) node and achieve the earned benefit of amount \( p \) by influencing all the remaining \( v_i \) nodes. However, as Algorithm 1 selects the seed node based on the marginal gain in the earned benefit per unit cost, it will select the node \( u \) and not any \( v_i \). For the node \( u \), the value of \( \Delta_\beta(S|u) \), when \( S = \phi \), is \( \frac{1}{1-\epsilon} \). On the other hand, for any \( v_i \), the value of \( \Delta_\beta(S|v_i) \) is 1. As \( \frac{1}{1-\epsilon} > 1 \), Algorithm 1 selects the node \( u \). After selecting the node \( u \), the remaining budget will be \( p + \epsilon - 1 \), which is less than \( p \). Within this budget none of the \( v_i \) nodes can be selected as each of them has the selection cost \( p \). Hence, Algorithm 1 terminates by earning the benefit 1 and returning an unutilized budget of amount \( p + \epsilon - 1 \). The approximation ratio of the Algorithm 1 is define as

\[
\text{Ratio}_{\text{Alg1}} = \frac{\text{Benefit earned by the seed set selected by Algorithm 1}}{\text{Benefit earned by the optimal seed set}}
\]

In this example the value of \( \text{Ratio}_{\text{Alg1}} \) is \( \frac{1}{p} \). If the value of \( p \) is arbitrarily large, then the approximation ratio of Algorithm 1 becomes very very less. Hence, Theorem 2 holds.

**Theorem 2.** Algorithm 1 does not provide any constant approximation guarantee.

Now, we present two important inequities on the iterative performance of Algorithm 1 and this result will be used subsequently.

**Lemma 3.** After each iteration of the ‘while’ loop \( i = 1, 2, \ldots, p + 1 \), the following inequality always holds

\[
\beta(S_i) - \beta(S_{i-1}) \geq \frac{C(u_i)}{B} (\beta(S^{\text{opt}}) - \beta(S_{i-1})).
\]

**Proof.** The value of \( \beta(S^{\text{opt}}) - \beta(S_{i-1}) \) is no more than the sum of the benefit values of the target nodes that are influenced by the seed nodes in \( S^{\text{opt}} \), however, not by the nodes in \( S_{i-1} \).

For each node in \( S^{\text{opt}} \setminus S_{i-1} \), the earned benefit to cost ratio could be at most \( \frac{B_i}{C(u_i)} \), where \( B_i \) is the earned benefit by the nodes in \( S_i \) but not in \( S_{i-1} \). This is because \( S_i \) maximizes this ratio in Algorithm 1. Since the total selection cost of the nodes in \( S^{\text{opt}} \setminus S_{i-1} \) is bounded by the budget \( B \), the total earned benefit due to the target nodes in \( I(S^{\text{opt}}) \setminus I(S_{i-1}) \) can be at most \( B \frac{B_i}{C(u_i)} \). Hence, we have

\[
\beta(S^{\text{opt}}) - \beta(S_{i-1}) \leq B \frac{B_i}{C(u_i)}
\]

By definition, we have

\[
B_i = \beta(S_i) - \beta(S_{i-1})
\]

From the Equations (5) and (6), we have
This completes the proof.

Lemma 4. In any arbitrary iteration \(i = 1, 2, \ldots, p + 1\) of the While loop from Line 2 to 9 of Algorithm 1, the following condition will be true

\[
\beta(S_i) \geq [1 - \prod_{i=1}^{i-1}(1 - \frac{C(u_i)}{B})]\beta(S_{opt})
\]

Proof. We prove this statement by the method of induction on the iteration of the ‘while’ loop. For the first iteration, i.e., at \(i = 1\), we need to show,

\[
\beta(S_1) \geq \frac{C(u_1)}{B} \beta(S_{opt})
\]

From Lemma 3, by putting \(i = 0\) in Equation (4) we have,

\[
\beta(S_1) - \beta(S_0) \geq \frac{C(u_1)}{B} (\beta(S_{opt}) - \beta(S_0))
\]

As we are starting with an empty seed set, hence \(S_0 = \emptyset\) and \(\beta(S_0) = 0\). This clearly implies that \(\beta(S_1) \geq \frac{C(u_1)}{B} \beta(S_{opt})\).

Now, suppose the statement holds till \((i - 1)^{th}\) iteration. We show that the statement holds in the \(i^{th}\) iteration as well. Now,

\[
\beta(S_i) = \beta(S_{i-1}) + (\beta(S_i) - \beta(S_{i-1}))
\]

\[
\geq \beta(S_{i-1}) + \frac{C(u_i)}{B} (\beta(S_{opt}) - \beta(S_{i-1}))
\]

\[
= \left(1 - \frac{C(u_i)}{B}\right) \beta(S_{i-1}) + \frac{C(u_i)}{B} \beta(S_{opt})
\]

\[
\geq \left(1 - \frac{C(u_i)}{B}\right) \beta(S_{i-1}) - \prod_{k=1}^{i-1}(1 - \frac{C(u_k)}{B})\beta(S_{opt}) + \frac{C(u_i)}{B} \beta(S_{opt})
\]

\[
\geq \left[1 - \prod_{k=1}^{i}(1 - \frac{C(u_k)}{B})\right]\beta(S_{opt})
\]

Here, the first inequality is due to Lemma 3 and the second one is due to inductive hypothesis.

Algorithm 1 can be modified for yielding a constant approximation ratio on the earned benefit. Let \(S_C\) be the seed set generated by the Algorithm 1. \(u_{max}\) be the node that has the highest individual benefit gain. We compare the earned benefit, when the seed set is \(S_C\) and the node is \(u_{max}\). We return the seed set that maximizes the earned benefit. Algorithm 2 formally states the procedure.

Algorithm 2 Modified Incremental Greedy Algorithm

Input: Social Network \(G(V,E,P)\), Target Nodes \(D\), Cost Function \(C\), Benefit Function \(b\), and Budget \(B\).

Output: The seed set \(S \subseteq V(G)\) such that \(\sum_{u \in S} C(u) \leq B\).

1: \(S \leftarrow \emptyset\)
2: \(S_C = \text{Seed Set selected by Algorithm 1}\)
3: \(u_{max} = \arg \max_{v \in V(G), C(v) \leq B} b(v)\)
4: \(S = \arg \max_{S_C \cup \{u_{max}\}} (\beta(S_C), \beta(u_{max}))\)
5: return \(S\)

Algorithm 2 provides bounded approximation guarantee, which is stated in Theorem 3.

Theorem 3. \(S^A\) is the seed set selected by Algorithm 2 and \(S_{opt}\) be the optimal seed set, then \(\beta(S^A) \geq (1 - \frac{1}{\sqrt{e}}) \beta(S_{opt})\), where \(e = \sum_{x=1}^{\infty} \frac{1}{x^2}\) In other words, Algorithm 2 provides an approximation guarantee of \((1 - \frac{1}{\sqrt{e}})\).

Proof. The strategy of this proof has been used previously for proving the approximation bound of the Budgeted Maximum Coverage Problem by Khuller et al. [40]. Here, we prove the statement by case-wise analysis of Algorithm 2.

Case I
If there exists one node \(u \in V(G)\), which has the earned benefit \(\beta(u)\), and \(\beta(u)\) is found to be greater than equal to \(\beta(S_{opt})\), then \(u\) will be selected as \(u_{max}\) in Algorithm 2. In this case, the approximation ratio of Algorithm 2 will be as follows:

\[
\frac{\beta(S_{opt})}{\beta(S^A)} \geq \frac{\beta(S_{opt})}{\beta(S_{opt})} = \frac{1}{2}
\]

Case II
If Case I does not happen, then there does not exist any \(u \in V(G)\), for which \(\beta(u)\) is greater than \(\beta(S_{opt})\). This can be divided into two sub-cases.

Case IIa
Now, if we have \(C(S^A) < \frac{B}{2}\), then \(\forall u \in V(G) \setminus S\), \(C(u) > \frac{B}{2}\). Hence, no more node can be added to \(S^A\). Otherwise, the budget constraint will be violated. Without the loss of generality, let us assume that \(S_{opt} \neq S^A\). In this case, \(S_{opt}\) can contain one extra node without violating the budget constraint. Now, as the function \(\beta(.)\) is sub-modular, hence,

\[
\beta(S_{opt} \cap S^A) \geq \beta(S_{opt})
\]

As \(\forall u \in V(G), \beta(u) < \frac{\beta(S_{opt})}{2}\). This clarifies that \(\beta(S_{opt} \cap S^A) \geq \frac{\beta(S_{opt})}{2}\). As \(S_{opt} = S^A \cup \{v\}\) and \(v \notin S^A\), \(S_{opt} \cap S^A = S^A\). This essentially means \(\beta(S^A) \geq \frac{\beta(S_{opt})}{2}\). In this case, the approximation ratio of the Algorithm 2 will be as follows:

\[
\frac{\beta(S_{opt})}{\beta(S^A)} \geq \frac{\beta(S_{opt})}{\beta(S_{opt})} \geq \frac{1}{2}
\]

Case IIb
If \(C(S^A) \geq \frac{B}{2}\), we first observe that for \(n\) real numbers \(a_1, a_2, \ldots, a_n\) and \(\sum_{i=1}^{n} a_i \geq \alpha A\), the function \(\prod_{i=1}^{n} (1 - \frac{a_i}{A})\) attains its maximum value, when \(a_i = \frac{\alpha A}{n}\). Hence, by Lemma 4, we have

\[
\beta(S^A) \geq [1 - \prod_{i=1}^{A} (1 - \frac{C(u_i)}{B})] \beta(S_{opt})
\]

\[
\geq [1 - \frac{1}{\sqrt{e}}] \beta(S_{opt})
\]

Hence, the worst case performance guarantee of Algorithm 2 is \((1 - \frac{1}{\sqrt{e}})\). This proves the statement.

Now, we investigate the time requirement of Algorithms 1 and 2. For both of them, it is easy to observe that the time requirement is heavily dependent on the earned benefit calculation for a given seed set. It is reported in the literature that counting the number of influenced nodes for a given seed set is \#P-Hard problem [6]. With this argument, we can say that for a given seed set \(S\), computing the exact
value of the earned benefit is also \(\#P\)-Hard. Hence, we estimate this value, the way influence of a seed set is estimated [6]. First, a number (say \(R\)) of sampled graphs of \(G\), i.e., \(G_1, G_2, \ldots, G_R\) are generated, and for all \(p \in [R]\), for all \((u_i, u_j) \in E(G_p), (u_i, u_j) \in E(G_p)\) with probability \(P_{u_i, u_j}\) and \((u_i, u_j) \notin E(G_p)\) with probability \((1 - P_{u_i, u_j})\). Now, earned benefit is computed in all of the sampled graphs and the average value is returned as its approximate value, which is given in Equation 7

\[
\beta_G(S) = \sum_{p=1}^{R} \beta_{G_p}(S) / R
\]  

(7)

If \(|S| = k\), then traversing \(R\) subgraphs will require \(O(k(m + n)R)\) time. Let, \(C_{\text{min}}\) be the minimum selection cost among all the users. Maximum number of possible iterations of the While loop (Line 2 to 9) in Algorithm 1 is \(\frac{B}{C_{\text{min}}}\). Hence, \(k \leq \frac{B}{C_{\text{min}}}\). In Algorithm 1, in each iteration, maximum number of times earned benefit estimations are done is \(O(n)\). Hence, the total number of times earned benefit estimations are of \(O(\frac{B}{C_{\text{min}}}n)\). The required computational time for Algorithm 1 is \(O(\frac{B}{C_{\text{min}}}^2 n(m + n)R)\).

In Algorithm 2, along with the incremental greedy strategy, the node, which can grab the maximum earned benefit has to be found out (Line 3 of Algorithm 2). This can be done \(O(n)\) earned benefit estimations with a single seed node, and this will take \(O(n(m + n)R)\) time. Hence, running time of Algorithm 2 is of \(O(\frac{B}{C_{\text{min}}}^2 n(m + n)R + n(m + n)R) \approx O(\frac{B}{C_{\text{min}}}^2 n(m + n)R)\). If we do the on-line sampling of the input social network for sampled graph generation, then only one subgraph is required per iteration. For storing, this network will take \(O(n + m)\) space. Storing the seed set requires \(O(\frac{B}{C_{\text{min}}}R)\) space. Hence, the total amount of space required by both Algorithms 1 and 2 is \(O(m + n + \frac{B}{C_{\text{min}}}R)\) and the number of seed nodes is generally found to be much more less than the number of nodes, i.e., \(\frac{B}{C_{\text{min}}} << n\). Hence, \(O(m + n + \frac{B}{C_{\text{min}}}) \approx O(m + n)\). Hence, Theorem 4 holds.

**Theorem 4.** Algorithms 1 and 2 have the running time of \(O(\frac{B}{C_{\text{min}}}^2 n(m + n)R)\) and space requirement of \(O(m + n)\).

### C. Improving the Efficiency of Algorithm 2

Though Algorithm 2 provides a provable approximation bound on the earned benefit, it is highly inefficient, as it estimates the earned benefit many times. Here, we present an improved version of Algorithm 2 in Algorithm 3 by removing redundant earned benefit estimations due to the exploitation of the sub-modularity property of the earned benefit function.

#### Algorithm 3 Incremental Greedy Algorithm with Improve Performance in terms of Efficiency (IGAIP).

**Input:** Social Network \(G(V, E, P)\), Target Nodes \(D\), Cost Function \(C\), Benefit Function \(b\), and Budget \(B\).

**Output:** The seed set \(S \subseteq V(G)\) such that \(\sum_{u \in S} C(u) \leq B\).

1: \(S \leftarrow \emptyset\)
2: while \(\exists u \in V(G) \setminus S\) and \(C(S \cup \{u\}) \leq B\) do
3: for All \(u \in V(G) \setminus S\) do
4: \(\text{Cur}_u = \text{False}\)
5: end for
6: while True do
7: \(w \leftarrow \arg\max_{u \in V(G): C(S \cup \{u\}) \leq B} \Delta b(u|S)\)
8: if \(\text{Cur}_w = \text{True}\) then
9: \(S = S \cup \{w\}\)
10: \(\text{break}\)
11: else
12: \(\Delta b(w|S) = \beta(S \cup \{w\}) - \beta(S)\)
13: \(\text{Cur}_w = \text{True}\)
14: end if
15: end while
16: end while
17: return \(S\)

In Lemma 2, it has been shown that the earned benefit function \(\beta(.)\) is sub-modular and this implies that the marginal gain in earned benefit for a non-seed node (say \(u\)) with respect to the seed set in \(i\)-th iteration \((S^i)\) will always be more than that of with respect to the seed set in \((i + 1)\)-th iteration. In Algorithm 3, in the first iteration of the while loop (Line 2), the earned benefit by the nodes in \(V(G)\) individually is computed, sorted them in descending order, and put the node with the highest individual earned benefit in the seed set. Now, in the next iteration on words, during the computation of the marginal gain of the non-seed nodes in descending order of their marginal earned benefit, as soon as we get a node, whose marginal gain in the current iteration is more than that in the previous iteration of the next node in the sorted list, then we include the first node and move to the next iteration. This is because, as the benefit function is sub-modular, even if we compute the marginal gain, earned benefit is computed for the second and the subsequent nodes, it cannot be more than the values in the previous iteration. This process is iterated, until the budget is exhausted. One important point to observe here is that, escaping the unnecessary benefit function evaluation does not result in loosing approximation guarantee in the quality of the selected seed set. This exploitation of the sub-modularity property results in significant improvement in the efficiency of our proposed methodology, as we observe in our experiments.

#### D. Efficient Heuristic Solution

Though Algorithm 3 is quite efficient, it is not enough to deal with large real-life social networks. Here, we propose an efficient heuristic solution for the EBM Problem. Before stating the procedure, first we state one important aspect of social influence. In social networks, influence of a node is bounded within 2 to 3 hops, which is called as the influence...
zone of a node [41] [42]. According to Goel et al. [43], in a diffusion cascade, less than 10% of the influenced nodes
resides more than hop count 2 from any seed node. These existing results reported in the literature motivate us to design
algorithm considering the locality of influence effect. Based
on this principle, to influence a target node, there should be
at least one seed node within a few hop count. In this context,
we define h-hop neighbor of a node as follows:

**Definition 4** (h-hop Neighbor). For a node \( u \in V(G) \), its
h-hop neighbor is defined as the set of nodes that are at most
at a distance of \( h \) from \( u \) and denoted as \( N^h(u) \), i.e., \( N^h(u) = \{ u_j | \text{dist}(u_j, u) \leq h \} \).

Among the nodes present in h-hop neighbor set of a target
node, there can be many nodes, whose influence probability
to the target node is extremely low. Hence, those nodes probably
cannot be able to influence the target node. To identify such
nodes, it is important to compute the influence probability. For
a target node \( u_i \), here we describe the procedure for computing
\( P_{u_j \rightarrow u_i} \), where \( u_j \in N^h(u_i) \). We construct the breadth first
search tree upto depth \( h \) rooted at the node \( u_i \). Now, for any
node \( u_j \) other than root of the tree, the value of
the following equation can be computed by the following equation:

\[
P_{u_j \rightarrow u_i} = [1 - \prod_{u \in N(u_i)} (1 - P_{u_j \rightarrow u})] P_{u_i \rightarrow u_j} \tag{8}
\]

In Equation (8), the value of \( P_{u_j \rightarrow u} \) can be recursively
computed, until the child of the currently processing node is
\( u_j \). For details, please look into [41]. Now, it is effective
to identify among the nodes in the \( N^h(u_i) \) which are effective
for influencing the target user \( u_i \). Here, we define the Effective
h-hop neighbors as follows.

**Definition 5** (Effective h-hop Neighbors). Given a target node
\( u_i \in D \) and an \( \alpha \in [0, 1] \), the effective h-hop neighbor(s) of
\( u_i \) is a subset of its h-hop neighbors and denoted as \( N^h_\alpha(u_i) \).
For \( u_j \in N^h_\alpha(u_i) \), the node \( u_j \) is an effective h-hop neighbor
of the node \( u_i \), if \( P_{u_j \rightarrow u_i} \geq \alpha \), i.e., \( N^h_\alpha(u_i) = \{ u_j | u_j \in
N^h(u_i) \land P_{u_j \rightarrow u_i} \geq \alpha \} \).

For any node \( u_j \), the main criterion to be included in the
seed set is how much benefit it can earn. If the node
is one of the target nodes, then the benefit associated with
this node is surely be earned and along with this, if there
are some target nodes (say \( u_i \)) within a few hop distance,
benefit corresponding to that node may be earned, however,
it depends upon the influence probability \( P_{u_j \rightarrow u_i} \). Now, we
define the earned benefit of a node as follows.

**Definition 6** (Earned Benefit of a Node). For a node \( u_j \in
V(G) \), its earned benefit \( \mathcal{EB}(u_j) \) is defined as the amount of
benefit that can be earned by including this node in the seed
set. It has two components. One is the direct benefit due to this
node. The other one is the expected benefit due to influencing
nearby target nodes. Mathematically, it can be expressed as
follows:

\[
\mathcal{EB}(u_j) = b(u_j) + \sum_{u_i \in N^h(u_j) \land P_{u_j \rightarrow u_i} \geq \alpha} P_{u_j \rightarrow u_i} b(u_i) \tag{9}
\]

There are two components in the right hand side of Equation
(9). The first part is due to the benefit associated with this
particular node and the second part signifies the ‘expected
earned benefit’, i.e., the expected benefit due to the influence
of the nodes within the few of distance of the node under
consideration.

Now, we describe the hop-based heuristic for solving the
EBM Problem. First, we create an array for storing the
expected earned benefit of each individual node and initialized
with 0 for non-target nodes and associated benefit value for
the target nodes (from Lines 1 to 6 of Algorithm 4). Then, for
a target node, we compute the effective h-hop neighbors (from
Lines 8 to 14). Then, for each of these nodes, we compute the
expected benefit that can be earned by influencing the target
node and sum it up. This process is repeated for each of the
target nodes. Next, we divide the earned benefit of each target
node by its selection cost and sort the nodes in descending
order based on this earned benefit value. Finally, we choose
the seed node from this sorted list until the budget is exhausted.
Algorithm 4 describes this procedure.

Now, we analyze the time and space requirement of Algo-
rum 4 by assuming it as a sparse and d-regular graph.
For initializing the array \( \mathcal{EB} \) requires \( O(n) \) time (Line 1 to
6). Now, for a target node \( u_i \) in a d-regular graph, number of
nodes and edges within the hop \( h \) is \( O(d^{h+1}) \). Hence,
performing breadth first search from \( u_i \) upto depth \( h \) requires
\( O(d^{h+1}) \) time. For computing the influence probability from
each node \( u_j \in N^h(u_i) \) to \( u_i \), i.e., \( P_{u_j \rightarrow u_i} \), and comparing with
\( \alpha \) requires \( O(hd^h) \) time. In the worst case, all the h-hop-neighbor
nodes may be the effective h-hop neighbor nodes. Then,
for computing the earned benefit by influencing the target
node \( u_i \) requires \( O(d^{h+1}) \) time. The same process is iterated
over all the target nodes. Hence, the execution from Lines
7 to 18 of Algorithm 4 requires \( O(|D|d^{h+1}(dh^h + d^{h+1})) \).

The formal statement is stated in Theorem 5.

**Theorem 5.** Algorithm 4 has the running time of \( O(n \log n +
|D|d^{h+1}(dh^h + d^{h+1})) \) and space requirement of \( O(n + d^{h+1}).

V. EXPERIMENTAL EVALUATION

In this section, we report the experimental evaluation of
our proposed methodologies. Initially, we start with a brief
description of the datasets.

**A. Dataset Description**

In our experiments, we use the following four publicly
available social network datasets.
Algorithm 4 A Hop-Based Heuristic for the EBM Problem

Input: Social Network $G(V, E, P)$, Target Nodes $D$, Cost Function $C$, Benefit Function $b$, Hop Count $h$, Cut off Probability $\alpha$, and Budget $B$.

Output: The seed set $S \subseteq V(G)$ such that $\sum_{u \in S} C(u) \leq B$.

1. Create Vector $(EB, n, 0)$
2. for All $u \in V(G)$ do
3. if $u \notin D$ then
4. $EB(u) = \beta(u)$
5. end if
6. end for
7. for Each $u \in D$ do
8. $N^h(u) = \{v|\text{dist}(uv) \leq h\}$
9. for Each $w \in N^h(u)$ do
10. Compute $P_{w \rightarrow u}$ using Equation (8)
11. if $P_{w \rightarrow u} \geq \alpha$ then
12. $N^h_E(u) = N^h_E(u) \cup \{(w, P_{w \rightarrow u})\}$
13. end if
14. end for
15. for Each $(w, P_{w \rightarrow u}) \in N^h_E(u)$ do
16. $EB(u) = EB(u) + \beta(u).P_{w \rightarrow u}$
17. end for
18. end for
19. for Each $w \in V(G)$ do
20. $EB(u) = \frac{EB(u)}{C(u)}$
21. end for
22. $V = \text{Sort } V(G)$ based on $EB$
23. $i = 1$
24. $S \leftarrow \phi$
25. while $B \geq 0$ do
26. if $B \geq C(V[i])$ then
27. $S = S \cup \{V[i]\}$
28. $B = B - C(V[i])$
29. end if
30. $i = i + 1$
31. end while
32. return $S$

- **Email-Eu-core network Dataset** [44], [45]: The network is generated based on the e-mail exchanges among different departments from a large European research institution. There is an edge between the users $u_i$ and $u_j$, if there is an e-mail exchange between them.
- **Facebook Network Dataset** [46]: This dataset was collected from survey participants using a Facebook app. Each user of the network is represented by a node, and two vertices are connected by an edge, if the corresponding users are friends of each other in Facebook.
- **Physics Network Dataset** [47]: This is an academic collaboration network among the researchers of physics section of arxiv.org. Two users are connected by an undirected edge, if they are co-author in at least one paper.
- **Epinions** [47]: This is a who-trust-whom on-line social network of a general consumer review site: Epinions.com. There is a directed edge from the user $u_i$ to $u_j$, if the user $u_i$ trusts $u_j$.

Among them, the first, second and fourth one are downloaded from Stanford Social Network Analysis http://snap.stanford.edu/data/index.html and the third one is from https://www.microsoft.com/en-us/research/people/weic/#selected-projects. Here, we give a brief description of each of the datasets. These datasets have been extensively used in social influence maximization research [7]. Table I gives a basic statistics of the described datasets.

### TABLE I

| Dataset Name               | $|V(G)|$ | $|E(G)|$ | Avg Deg | Avg Clus Coeff |
|----------------------------|--------|--------|---------|---------------|
| Email-Eu-core network      | 1005   | 25571  | 25,443  | 0.3994        |
| Facebook Dataset           | 4037   | 88254  | 43,6910 | 0.6055        |
| PHY Network                | 37154  | 231584 | 12,466  | 0.2371        |
| Epinions                   | 75879  | 508379 | 15,6345 | 0.1378        |

#### B. Parameter Settings

1) **Diffusion Probability**: In this paper, we consider the following two diffusion probability settings.

- **Uniform Setting**: In this setting, $\forall (u_i, u_j) \in E(G)$, $P_{u_i \rightarrow u_j} = P$, and $P \in (0, 1]$. We set the value of $P$ as 0.1. This value has also been used in the literature, in many studies [27].

- **Trivalency Setting**: In this setting, each edge is assigned diffusion probability uniformly at random from the set $\{0.1, 0.01, 0.001\}$.

On the other hand,

2) **Target Nodes**: In this study, we select 20% of the nodes as target nodes and they are chosen uniformly at random. We adopt this settings from [48].

3) **Cost and Benefit**: In this study, we follow two different settings.

- First one is the random setting, where the selection cost of the nodes and the earned benefit of the target nodes are selected from the intervals $[1, 50]$ and $[50, 100]$, respectively, uniformly at random. We adopt this setting from [16] and call it the random setting.

- Secondly, the influence ability of a node is directly proportional to its degree. Naturally, the selection cost of a node should be proportional to its degree. We adopt another settings from [48]. By this setting, we compute the selection cost of the node $u_i$ is computed as

$$C(u_i) = \frac{n \cdot \text{deg}(u_i)}{2.m},$$

(10)

and in this case, the benefit of each target node is considered as 1. We call this setting as the ‘degree proportional’ setting.
4) Budget: In our study, based on the two different cost assignment settings, we adopt two different budget settings. In case of random setting, we consider the budget values starting from 2000 continued till 16,000, and each time is incremented by 2,000. In the ‘degree proportional setting’, we start with the budget value of 100 and continued until 800 with a gap of 100.

5) Hop Count and Cut Off Probability: In Algorithm 4, we use a hop count \( h \) and cut off probability \( \alpha \) for computing the effective nodes. In our experiments, we choose the value of \( h \) as 2 and the value of \( \alpha \) as 0.1. We adopt these settings from [41].

C. Algorithms in the Experiment

Here, we have listed out the algorithms that we have listed out for the experimentation.

1) Algorithms proposed in this paper:
   - **Incremental Greedy Approach with Approximation Guarantee** (IGAAG): This is basically the Algorithm 2 of this paper, which returns either the set of nodes chosen incrementally by Algorithm 1 or the node that causes maximum individual benefit gain.
   - **Incremental Greedy Approach with Improved Performance** (IGAIP): This is the Algorithm 3 of this paper, which improves the Algorithm 2 by exploiting the sub-modularity property of the benefit function.
   - **Hop-Based Heuristic** (HBB): This is the Algorithm 4 of this paper, which works based on the computation of expected earned benefit of the nodes that are within the \( h \)-hop (for a given value of \( h \)) of the target nodes.

2) Baseline Algorithms:
   - **Maximum Degree Heuristic** (Max_DEG): In this method, the maximum degree nodes within the budget is returned as the seed set. This method has been used in previous studies as well [6].
   - **Degree Discount Heuristic** (DEG_DIS): This is a popular heuristic for the SIM Problem proposed by Chen et al. [49]. In this heuristic, if \( u \) is a seed node and \( \langle uv \rangle \in E(G) \), then the degree of \( v \) will be discounted by \( 2t_v + (d_v - t_v)t_v \frac{\mathcal{P}_{u-v}}{v} \), where \( t_v \) is the number of neighbors of \( v \) currently in the seed set, and \( d_v \) is the degree of \( v \). This method has been used in many previous studies [50].
   - **Single Discount Heuristic** (SIN_DEG): This a variant of degree discount heuristic proposed by Chen et al. [49]. In this heuristic, if \( u \) is a seed node and \( \langle uv \rangle \in E(G) \), then the degree of \( v \) will be discounted by 1. This method has been used as a baseline in previous studies [51] [52].
   - **Prefix excluded Maximum Influence Arbarence** (PMIA): This is one of the state-of-the-art and popular heuristic for influence maximization problem proposed by Chen et al. [7] [53].
   - **ComPBRA**: This is a recently developed community-based solution framework for the EBM Problem developed by Banerjee et al. [24].

All the algorithms have been implemented in ‘Python 3.4’ along with ‘NetworkX 1.9.1’. We have carried out all the experiments in a high performance computing cluster with 5 nodes and each of them having 64 cores and 64 GB of RAM running in Centos 6.7 environment. As, the Algorithm 2 (IGAAG) is quite inefficient, we don’t execute this on the larger datasets (e.g., Physics Network Dataset, Epinions).

D. Experimental Results and Discussion

The main goal of our experimentation is to make a comparative study of the proposed as well as baseline methods in terms of performance. It is measured as the amount of earned benefit obtained by influencing the target users due to the initial activation of the seed nodes selected by different algorithms. We also report the computational time requirement by different algorithms for selecting the seed sets.

1) Performance on Earned Benefit: First row of Figure 1 shows the budget vs. earned benefit plot for the ‘email-Eu-core’ dataset. Based on the random and degree proportional setting, the maximum benefit that can be earned is 13912 and 179, respectively. From the results, it is observed that there is a gap in the earned benefit between the existing methods and the methods proposed in this paper. The gap is even significant in tri-valency setting compared to the uniform setting. As an example, in uniform setting \( (p_c = 0.1) \) with random cost and benefit assignment for \( B = 16000 \), among the existing methods from the literature, the seed set selected by ComPBRA leads to more earned benefit and the amount is 12231, which is 73.34% of the maximum possible. On the other hand, among the proposed methodologies, the seed set selected by the IGAAG leads to more amount of earned benefit 13912, which is 91.37% of maximum possible.

In degree proportional setting, for \( B = 16000 \), in tri-valency setting among the existing methods, the seed set selected by both PMIA and ComPBRA leads to the earned benefit of 167 (83.5% of the maximum possible), whereas the same for both IGAAG and IGAIP is 172 (86% of the maximum possible).

Next, we report the results for the ‘Facebook’ dataset in the second row of Figure 1. In this dataset also, we observe that the seed set selected by the proposed methodologies leads to more earned benefit compared to the existing methods. As an example, when the budget value is 16000, under random cost and benefit with tri-valency setting, among the existing methods the seed set selected by ComPBRA leads to the earned benefit of 15578. However, the same due to the seed set selected by the proposed hop-based heuristic is 20450, which is almost 31% more. Now, under the degree proportional cost and tri-valency setting, when the budget value is 16000, among the existing methods the earned benefit due to the seed set by ComPBRA is 328, and the same by the hop-based heuristics is 426. Next, we show the results for the ‘Physics Collaboration Network’ dataset in the third row of Figure 1. In this dataset also, we observe a significant gap in the earned benefit between the existing methods and the methods proposed in the literature. The gap is more in case of tri-valency setting. As an example, for \( B = 16000 \), under random cost with uniform influence probability setting, among the existing methods, the seed set selected by PMIA leads to maximum amount of earned benefit which is 42817 and
the same by the hop-based heuristic is 43568. In tri-valency setting, for $B = 16000$, the seed set selected by SIN_DIS leads to the earned benefit, which is equal to 15352, and the same obtained by hop-based heuristic is 20955. This is approximately 36% more compared to the SIN_DIS method.

Next, we report the results for the ‘Epinions’ dataset in the last row of Figure 1. In this dataset also, we observe a significant difference between the earned benefit due to the seed sets selected by the baseline methods and the methods proposed in this paper. As an example, for the highest budget ($B = 16000$), in uniform setting under random cost and benefit assignment seed set selected by the ComPBRA leads to the earned benefit of worth 74561, and the same in case of the ‘hop-based heuristic’ is 86564, which is almost 16% more compared to the ComPBRA. Similarly, in degree proportional setting under tri-valency diffusion model, the seed set selected by the ComPBRA leads to the earned benefit of amount 688 and the same for the ‘hop-based heuristic’ is 911, which is almost 32% more compared to the ComPBRA.

From the results, it is observed that the seed set selected
by the proposed methodologies can lead to more amount of earned benefit compared to the existing methods considered in this paper. Next, we report the computational time of the proposed and baseline methods.

2) Computational Time: Table II reports the computational time required for selecting the seed sets by different methodologies. From the reported results, it is observed that the IGAAG method can achieve an approximation guarantee, the computational time requirement is quite impractical. However, the IGAIP method overcomes this issue by improving it upto 220 times faster compared to IGAAG. However, the hop-based heuristic is much more efficient and also scalable compared to both IGAAG and IGAIP, while achieving the almost similar amount of earned benefit, in some instances compared to both IGAAG and IGAIP, while achieving the almost similar amount of earned benefit, in some instances even more.

Among the baseline methods, the MAX_DEG is the fastest one, as it returns the high degree nodes within the budget. The DEG_DIS and SIN_DIS methods take more time compared to the MAX_DEG method. Among the existing methods, PMIA is seen to be the fastest.

Now, in real-life applications of this problem, such as 'computational advertisement', ‘viral marketing’ etc. from the advertisers point of view, the main priority will be the earned benefit. However, the methodology used for seed set selection purpose should be able to perform this task with a reasonable computational time. From the experimental evaluation, it is established that among the proposed methodologies, the hop-based heuristic is far ahead compared to the existing methods.

VI. CONCLUSION AND FUTURE DIRECTION

In this paper, we have studied the ‘Earned Benefit maximization problem’, where a subset of nodes of the input social network are designated as target nodes and each of them is associated with a benefit value. Each node of the network is associated with a selection cost and the seed selection has to be done within an allocated budget with an aim to maximize the earned benefit. For this problem, we propose an \((1 - \frac{1}{e})\) factor approximation algorithm. By exploiting the sub-modularity of the benefit function, we improve the efficiency of this algorithm. To deal with the large scale social networks, we propose a hop-based heuristic solution for this problem. Reported results demonstrate that the seed set selected by the proposed methodologies leads to more amount of earned benefit compared to the existing methods. Now, this study can be extended in several directions. First of all, our study can be carried forward by considering the time varying nature of the real-world social networks. Secondly, the approximation bound that we have provided for our proposed methodology can be improved by more sophisticated analysis. Moreover, it will be interesting to see up with a game theoretic model of this problem.

REFERENCES

[1] A. Abraham, A.-E. Hassanien, and V. Snášel, Computational social network analysis: Trends, tools and research advances.. Springer Science & Business Media, 2009.

[2] F. Chierichetti, S. Lattanzi, and A. Panconesi, “Rumor spreading in social networks,” Theoretical Computer Science, vol. 412, no. 24, pp. 2602–2610, 2011.

[3] J. Kleinberg, “The convergence of social and technological networks,” Communications of the ACM, vol. 51, no. 11, pp. 66–72, 2008.

[4] T. N. Dinh, H. Zhang, D. T. Nguyen, and M. T. Thai, “Cost-effective viral marketing for time-critical campaigns in large-scale social networks,” IEEE/ACM Transactions on Networking (ToN), vol. 22, no. 6, pp. 2001–2011, 2014.

[5] A. Baghernejad and R. Parekh, “Combining behavioral and social network data for online advertising,” in 2008 IEEE International Conference on Data Mining Workshops. IEEE, 2008, pp. 837–846.

[6] D. Kempe, J. Kleinberg, and É. Tardos, “Maximizing the spread of influence through a social network,” in Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2003, pp. 137–146.

[7] W. Chen, C. Wang, and Y. Wang, “Scalable influence maximization for prevalent viral marketing in large-scale social networks,” in Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2010, pp. 1029–1038.

[8] M. Ye, X. Liu, and W.-C. Lee, “Exploring social influence for recommendation: a generative model approach,” in Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval. ACM, 2012, pp. 671–680.

[9] A. Monteserin and M. G. Armentano, “ Influence-based approach to market basket analysis,” Information Systems, 2018.

[10] L. Jiang, J. Liu, D. Zhou, Q. Zhou, X. Yang, and G. Yu, “Predicting the evolution of hot topics: A solution based on the online opinion dynamics model in social media,” IEEE Transactions on Systems, Man, and Cybernetics: Systems, 2018.

[11] Y. Li, J. Fan, Y. Wang, and K.-L. Tan, “Influence maximization on social graphs: A survey,” IEEE Transactions on Knowledge and Data Engineering, 2018.

[12] S. Banerjee, M. Jenamani, and D. K. Pratihar, “A survey on influence maximization in a social network,” Knowl Inf Syst, 2013.

[13] Y.-M. Li and Y.-L. Shiu, “A diffusion mechanism for social advertising over microblogs,” Decision Support Systems, vol. 54, no. 1, pp. 9–22, 2012.

[14] A. Guille, H. Hacid, C. Favre, and D. A. Zighed, “Information diffusion in online social networks: A survey,” ACM Sigmod Record, vol. 42, no. 2, pp. 17–28, 2013.

[15] M. Li, X. Wang, K. Gao, and S. Zhang, “A survey on information diffusion in online social networks: Models and methods,” Information, vol. 8, no. 4, p. 118, 2017.

[16] H. Nguyen and R. Zheng, “On budgeted influence maximization in social networks,” IEEE Journal on Selected Areas in Communications, vol. 31, no. 6, pp. 1084–1094, 2013.

[17] E. Güney, “On the optimal solution of budgeted influence maximization problem in social networks,” Operational Research, pp. 1–15, 2017.

[18] S. Banerjee, M. Jenamani, and D. K. Pratihar, “Combining a community-based solution approach for the budgeted influence maximization problem,” Expert Systems with Applications, vol. 125, pp. 1–13, 2019.
