Adaptive Crawling with Cautious Users

Xiang Li*, Tianyi Pan†‡, Guangmo (Amo) Tong§, Kai Pan¶
*Department of Computer Engineering, Santa Clara University, Santa Clara, CA, USA
†Google Inc, Mountain View, CA, USA
‡ CISE Department, University of Florida, Gainesville, FL, USA
§Department of Computer and Information Sciences, University of Delaware, DE, USA
¶Department of Logistics and Maritime Studies, The Hong Kong Polytechnic University, Hung Hom, Kowloon, Hong Kong
Email: xli8@scu.edu, tianyipan@google.com, amotong@udel.edu, kai.pan@polyu.edu.hk

Abstract—**In Online Social Networks (OSNs), privacy issue is a growing concern as more and more users are sharing their candid personal information and friendships online. One simple yet effective attack aims at private user data is to use socialbots to befriend the users and crawl data from users who accept the attackers' friend requests. With the attackers involving, individual users' preference and habit analysis is available, hence it is easier for the attackers to trick the users and befriend them. To better protect private information, some cautious, high-profile users may refer to their friends' decisions when receiving a friend request. The aim for this paper is to analyze the vulnerability of OSN users under this attack, in a more realistic setting that the high profile users having a different friend request acceptance model. Specifically, despite the existing probabilistic acceptance models, we introduce a deterministic linear threshold acceptance model for the cautious users such that they will only accept friend requests from users sharing at least a certain number of mutual friends with them. The model makes the cautious users harder to befriend with and complicates the attack. Although the new problem with multiple acceptance models is non-submodular and has no performance guarantee in general, we introduce the concept of adaptive submodular ratio and establish an approximation ratio under certain conditions. In addition, our results are also verified by extensive experiments in real-world OSN data sets.**

Index Terms—**Adaptive Crawling, Online Social Network, Adaptive Non-submodular Optimization**

I. Introduction

The Online Social Networks (OSNs) have become a rich source for user information and then a target for attackers, because many users of OSNs provide genuine information about themselves in their online profiles. In order to harvest private user information, attackers can simply befriend the normal users [1] since users usually give their friends the access to a large portion of their personal profiles. Although the attack is simple, it can be used to support severe attacks like spear phishing and account compromise with the collected user information [2]. Because of its importance, there is an emerging need of analyzing the attacks that befriend users and the vulnerability of OSNs to support future protection schemes. The problem of crawling information via befriending users, termed as Adaptive Crawling, has been studied in literature [3], [2], [4], [5], [6], [7]. The name adaptive is from the assumption that the attackers only have incomplete knowledge of the OSN and their knowledge of the OSN is updated each time a friend request is accepted.

One important aspect of the problem is how the users accept friend requests. The existing works all assume the users accept friend requests according to a probability. The probability either follows a known distribution [3], [4], [5] or an empirical linear function with the number of mutual friends between the user and the attacker as the only variable [2], [6], [7]. The assumption leads to sound theoretical guarantees, yet it treats all users the same. In reality, as the attackers can deliberately design the accounts to look inviting for normal users to accept the friend requests [8], some high-profile users may not make acceptance/rejection decisions based on the requests alone as they are more cautious about their private information. Instead, they may observe how many friends of them are already friends of the request sender, and only accept the request if the number of mutual friends is above a threshold. Under this setting, we reach to a deterministic linear threshold model for the high-profile and cautious users, which is new for the adaptive crawling problem.

In this paper, our focus is to study the adaptive crawling with cautious users problem (ACCU). The goal of ACCU is to obtain the most benefit from the users by sending a fixed number of friend requests. As discussed above, it considers a more practical situation that both probabilistic and deterministic friend request acceptance models can co-exist. In addition to the challenge imposed by the incomplete OSN information as in the existing adaptive crawling problems, there are two main challenges specific to ACCU. Firstly, the existence of multiple friend request acceptance models makes it difficult to design a general yet efficient strategy to befriend the users. The decision of sending requests to the cautious users needs special consideration. Secondly, with the cautious users, the objective function of ACCU is no longer submodular, which means that the well-known greedy algorithm cannot guarantee the $(1-1/e)$ approximation ratio and we need to find alternatives to bound the theoretical performance of the algorithm that solves ACCU.

To deal with the challenges, we introduce the notion of adaptive submodular ratio, which extends the submodular ratio in [9] and characterizes how "submodular" an objective function is in the adaptive setting. With the help of the ratio, we are able to theoretically bound the performance of the greedy algorithm to ACCU in certain cases. Our contributions are summarized as follows.

- We formally define the ACCU problem, which is a more realistic variation of the existing adaptive crawling problems.
- We propose an efficient greedy-based solution, adaptive benefit maximization (ABM) to ACCU and theoretically bound its performance under certain conditions with the introduction of the novel notion, adaptive submodular ratio.
- We demonstrate the superior performance of ABM to several alternative algorithms via extensive experiments. Further, we conduct sensitivity analysis and provide insights on how the introduction of cautious users can shape the behavior of attackers.

Related Works. Existing literature [10], [8], [11] showed that the information of users in OSNs can be crawled via socialbots. A manually controlled socialbot can successfully befriend important users such as members of security agencies [10], and automated socialbots can effectively befriend a large number of users [8] or even infiltrate organizations [11]. Therefore, understanding the crawling attacks by socialbots is crucial [12].

The main line of research toward the crawling attacks [3], [2], [4], [5], [6], [7] aims at understanding the befriending strategies of the attackers, which can in turn reveal the key users to protect. In [3], [2], [6], [7], the authors studied the strategy that the attacker sends one request at a time. In [4], however, sending multiple requests at the same time was considered for higher efficiency of the attack. Further, [5] discussed a collaborative attack with multiple socialbots. The existing works all assumed probabilistic friend request acceptance models. In [3], [4], [5], the probability is random following a distribution, while [2], [6], [7] considered an empirical acceptance function that the the probability increases with more mutual friends between the attacker and the user.

A few works analyzed the performance of greedy algorithm for optimization problems, when the objective function is non-submodular. A curvature based concept was introduced in [13] for non-adaptive maximization problem and extended in [6], [7] to the adaptive context. The submodular ratio [9] is another approach to handle non-adaptive non-submodular objectives, it was further discussed in [14]

Organization The rest of the paper is organized as follows. In Section II, we define the related models in OSNs and formally define our ACCU problem. We then discuss our solution to ACCU in Section III and analyze its performance guarantee. Section IV presents our experimental results and Section V concludes the paper.

II. Problem Formulation

In this section, we introduce the various models used in the paper and formally define the problem.

A. Models

Network Model. We abstract the Online Social Network as an undirected graph $G = (V, E, p)$, where the node set V is the collection of users and the set E is the collection of user friendship relations. As the attacker usually does not have access to the complete network topology, the edges in the network are all probabilistic. The function $p : E \to [0, 1]$ defines the link existence probabilities for all edges. For example, $p(u, v) = p(v, u)$ is the probability that an edge exists between nodes u and v. For simplicity, we also write it as p_{uv} or p_{vu}. We denote $s \in V$ as the attacker, who initially has no connections to other nodes. When the attacker successfully befriends a user u, we add the relation to E with $p_{su} = p_{us} = 1$.

From the attacker's point of view, there are three groups of users: (1) friends (2) friend-of-friends and (3) strangers.

- **Friends.** The users are direct neighbors of the attacker s. We denote the set as $F = \{u|u \in V, (s, u) \in E, u \neq s\}$.
- **Friend-of-friends.** The users are two-hop neighbors of the attacker. We denote the set as $FOF = \{v|N(v) \cap N(s) \neq \emptyset, (v, s) \notin E, v \in V, v \neq s\}$ where $N(v)$ is the neighborhood of node v.
- **Strangers.** The users are not friends nor friends-of-friends of the attacker. We denote the set as $S = \{w|N(w) \cap N(s) = \emptyset, w \in V, w \neq s\}$.

Friend Request Acceptance Models Based on how cautious the users are, they may have different behavior in accepting a friend request. Less cautious users may accept an arbitrary friend request, while a more cautious user will only accept a friend request from someone they share a certain number of mutual friends with. Hence, we propose to use different acceptance models for the two types of users. Denote the subset of less cautious (or reckless) users as V_R and the set of more cautious users as V_C. where $V_R \cup V_C = V$ and $V_R \cap V_C = \emptyset$. A friend request to $u \in V_R$ is accepted with probability q_u. For a user $v \in V_C$, however, friend request acceptance is not based on probability. Instead, we introduce a threshold $\theta_v \in \mathbb{Z}^+$. A friend request from s is accepted if and only if $|N(v) \cap N(s)| \geq \theta_v$. Without loss of generality, We assume that each user $v \in V_C$ has enough friends in V_R so that it is not impossible to befriend them (if it is not the case, we can simply remove the users from the network as they will not impact the attack at all). In other words, $|N(v) \cap V_R| \geq \theta_v, \forall v \in V_C$. Also, we assume that the links among users in V_C can be neglected as the links are not likely to be utilized in the attack. That is, $N(v) \cap V_C = \emptyset, \forall v \in V_C$.

Benefit Model. The benefits that an attacker can collect from the users are based on how the users are related to the attacker. For user $u \in F$, the attacker can collect the benefit of $B_f(u)$. If the user u is in the set FOF, the amount of benefit to the attacker is $B_{fof}(u)$. For the same user u, we have $B_f(u) \geq B_{fof}(u)$ as all information accessible by a friend-of-friend is also accessible by a friend, but not vice versa. Additionally, we assume that befriending the cautious users is much more beneficial than befriending other users. The reason is that the cautious users are usually high-profile users, so that being their friend is more valuable. We argue that this assumption is valid. Since the network is gigantic and the resource available to the attacker is limited, it is beyond the scope of the attack to befriend a cautious user who has limited benefit to the attacker. Hence, we can ignore them and limit V_C to high-profile users with a high benefit.

B. Problem Definition

Based on the above models, the goal of the attacker s is to obtain the most information benefit from the friends and friends-of-friends, with a budget k on the number of friend requests to be sent.

Definition 1 (Adaptive Crawling with Cautious Users (ACCU))**.** *Given a social network $G = (V, E, p)$, where V is the set of user accounts, E is the set of potential friendships between users and p is the link existence probability function, the benefits B_f, B_{fof}, friend request acceptance parameters q, θ as defined earlier. The problem asks us to find a sequence of at most $k \in \mathbb{Z}^+$ users $Q = \{v_1, v_2, \ldots, v_k\}, v_1, \ldots, v_k \in V$ to befriend with, such that the total expected benefit gain from friends F and friends-of-friends FOF is maximized.*

Notice that the sequence Q is selected iteratively. Let $Q_i = \{v_1, \ldots, v_i\}$, the decision for picking node v_i is made after all the response of the request to nodes in Q_{i-1} are observed and the attacker's knowledge to the network is updated accordingly. We term the whole decision making process as an adaptive attack strategy π. When a user u accepts the friend request of s, the neighborhood of u, $N(u)$, will be available to s and is no longer probabilistic.

In order to deal with the stochastic nature of the problem, we apply adaptive stochastic optimization as in earlier works [2], [5], [6], [4]. First, we introduce our notations. Since whether a user $u \in V_R$ will accept the friend request sent by s and the friendship relations among normal users are all unknown to the attacker, we will introduce random variables to depict the randomness. For each user $u \in V_R$, let $X_u \in \{0, 1, ?\}$ denote the state of u. $X_u = 1$ means that u accepts the friend request from s, 0 means the rejection of the friend request and ? represents the unknown state, that is, u has not received a request from s yet. Notice that we do not define X for users in V_C as the acceptance criteria for them are deterministic. Similarly, for each edge $(u, v) \in E$, we introduce the random variable $X_{uv} \in \{0, 1, ?\}$. The state is ? when the friend request to u or v was rejected or not sent yet. When the friend request to u or v is accepted, whether the edge exists or not is revealed. $X_{uv} = 1$ if the edge (u, v) exists and 0 otherwise. Initially, the state of all Xs and Ys should be ?. Let Ω be the collection of all possible states of G and $\phi = \{X_v\}_{v \in V} \cup \{X_{uv}\}_{(u,v) \in E} \to \Omega$ be a possible state, called a realization. Also, let $\phi(u)$ be the state of node u and $\phi(u, v)$ the state of edge (u, v) under realization ϕ. The realizations must be consistent. That is, each node and edge in each realization must be in one and only one of the states $\{0, 1, ?\}$. Clearly there are many possible realizations. We denote $\Pr[\phi]$ as the probability distribution followed by the realizations. Also, let Φ be a random realization and $\Pr[\phi] = \Pr[\Phi = \phi]$ over all realizations.

Under the notations, when the attacker s sends a friend request to u, the state $\Phi(u)$ will be observed. If $\Phi(u) = 1$, the states $\Phi(u, v), v \in V$ will be observed. Let $Q(\pi, \phi)$ denote the sequence of nodes selected by strategy π under realization ϕ. When part of the states is available to the attacker, we represent the observations as a partial realization ω. Also, we use $\text{dom}(\omega)$ to refer to the domain of ω, that is, the set of nodes and edges already observed in ω. We call a partial realization ω consistent with a realization ϕ if they are equal everywhere in the domain of ω and write it as $\phi \sim \omega$. When ω, ω' are both consistent with some ϕ and $\text{dom}(\omega) \subseteq \text{dom}(\omega')$, we call ω a sub-realization of ω' or simply write $\omega \subseteq \omega'$.

We can calculate the total benefit gain of a strategy π under realization ϕ as follows:

$$f(\pi, \phi) = \sum_{u \in F(\pi, \phi)} B_f(u) + \sum_{v \in FOF(\pi, \phi)} B_{fof}(v) \quad (1)$$

where $F(\pi, \phi)$ and $FOF(\pi, \phi)$ denotes the friend and friend-of-friend set of the attacker with strategy π under realization ϕ, respectively.

Thus, we can formulate the problem ACCU as:

$$\begin{aligned} &\max \mathbb{E}[f(\pi, \Phi)|\Phi] \\ &s.t.\ \mathbb{E}[|Q(\pi, \Phi)||\Phi] \leq k \end{aligned} \quad (2)$$

III. The Adaptive Greedy Algorithm and Its Theoretical Guarantees

In this section, we provide our first algorithm towards the ACCU problem, followed by the proofs of its theoretical guarantee.

A. The Greedy Algorithm

In a typical greedy algorithm, the elements are picked iteratively to maximize the marginal gain. In ACCU, however, this approach may not be effective enough since the existence of cautious users V_C makes the benefit function non-submodular (we will discuss about the adaptive submodular property later in this section). Hence, we need to take into account how the friendship with a user will facilitate befriending the cautious users when making the greedy selections. In order to do so, we define a potential function for calculating the marginal gain of befriending a user, which considers both the direct benefit gain and the indirect gain of increasing the chance of befriending the cautious users later. The potential function is defined as follows:

$$P(u|\omega) = q(u)(w_D P_D + w_I P_I)$$

Where P_1 denotes the direct benefit gain and P_2 denotes the indirect benefit gain. w_D, w_I are tunable parameters that denotes the relative importance of direct/indirect gains. Note that the potential function is applicable to all users. For users in V_C, P_I is always 0 since we assume that the cautious user are not connected to each other. Specifically:

$$P_D = B_f(u) - \mathbf{1}_{FOF}(u)B_{fof}(u) + \left(\sum_{v \in N(u) \setminus N(s)} p_{uv}(1 - \mathbf{1}_{FOF}(v))B_{fof}(v) \right).$$

The indicator function $\mathbf{1}_{FOF}(u) = 1$ if $u \in FOF$ and $\mathbf{1}_{FOF}(u) = 0$ otherwise.

$$P_I = \sum_{v \in N(u) \cap V_C, \theta_v > |N(s) \cap N(v)|} p_{uv} \frac{(B_f(v) - B_{fof}(v))}{\theta_v - |N(s) \cap N(v)|}$$

The indirect gain favors the users who (1) are friends with more cautious users that are not yet friends of the attacker (2) the cautious friends of the user have high benefit and (3) the number of mutual friends among the cautious friends of the user and the attacker is close to reach the thresholds. Note that we remove the friend-of-friend benefit from indirect gain since if the cautious user is not yet a friend-of-friend of the attacker, the potential benefit is considered in P_D already.

With the potential function, we are ready to describe our greedy algorithm, Adaptive Benefit Maximization (ABM). At a high-level, ABM iteratively performs the following two steps until a total of k requests are sent: (1) Greedy Selection: the potential gain of each user $u \notin F$ is calculated and the user u^* with the highest gain is selected and (2) Observation: a friend request is sent to the selected user u^* in the first stage and the result is observed. If the friend request is accepted, the attacker is also able to observe the neighborhood of u^*, $N(u^*)$. The updated knowledge is used to aid the attacker for future decisions.

Algorithm 1: Adaptive Benefit Maximization (ABM)

```
Input: Graph $G = (V, E, p)$, $B_f, B_{fof}, q, \theta w_1, w_2$, and
       $k \in \mathbb{Z}^+$
Output: A sequence $Q$ of users for $s$ to be friend with.
$Q \leftarrow \emptyset; \omega \leftarrow \emptyset$
while $|Q| < k$ do
    foreach $u \in V \setminus Q$ do
        $P(u|\omega) = q(u)(w_D P_D + w_I P_I)$
    Select $u^* \in \arg\max_u \Delta(u|\omega)$
    Add $u^*$ to $Q$
    Send a friend request to $u^*$
    if $u^*$ accepts the friend request then
        Update $\omega$ with new observed $N(u^*)$
Return $Q$
```

B. Theoretical Guarantee

In this section, we first show why the benefit function of the ACCU problem is not adaptive submodular and hence the renowned $(1-1/e)$ approximation ratio is not applicable to the greedy algorithm described above. Then, we extend existing notions from non-adaptive non-submodular optimization to show that an approximation exists for the greedy algorithm of ACCU under certain conditions.

Before starting our analysis, we introduce several notations and definitions.

We already defined the benefit of a strategy π under realization ϕ in equation (1). Here we slightly abuse the notation to denote the gain of a partially executed strategy. Assume the observed partial realization is ω, then the benefit of the partially executed strategy under realization ϕ is $f(\text{dom}(\omega), \phi)$. With this notation, we can define the expected marginal gain of befriending a user u conditioned on having ω, which is as follows:

$$\Delta(u|\omega) = \mathbb{E}[f(\text{dom}(\omega) \cup \{u\}, \Phi) - f(\text{dom}(\omega), \Phi) | \Phi \sim \omega]$$

Next, we state the following definitions, which are defined in [15].

Definition 2 (Strongly Adaptive Monotone). *A function $f(\cdot)$ is strongly adaptive monotone with respect to the distribution $\Pr[\phi]$ if the following condition holds. For all ω, all $v \notin$ dom(ω), and all possible states o of node v such that $\Pr[\Phi(v) = o | \Phi \sim \omega] > 0$, we have:*

$$\mathbb{E}[f(dom(\omega), \Phi) | \Phi \sim \omega] \leq \mathbb{E}[f(dom(\omega) \cup \{v\}, \Phi) | \Phi \sim \omega, \Phi(v) = o] \quad (3)$$

Definition 3 (Adaptive Submodularity). *A function $f(.)$ is adaptive submodular w.r.t the distribution $\Pr[\phi]$ of all realizations if for all ω and ω' such that $\omega \subseteq \omega'$ and for all $v \in V \setminus dom(\omega')$, we have:*

$$\Delta(v|\omega) \geq \Delta(v|\omega') \quad (4)$$

According to [15], an adaptive greedy algorithm achieves the $(1 - 1/e)$ ratio when the objective function is strongly adaptive monotone and adaptive submodular. Unfortunately, the benefit function of ACCU is not adaptive submodular. Consider the following example with only two normal users v_1, v_2 and an attacker s. v_1 is a cautious user who will only accept the friend request of s if s and v_1 share at least 1 mutual friend. Also let $B_f(v_1) > B_{fof}(v_1) > 0$. v_2 is not a cautious user and will accept the friend request from s with probability 1. Then, consider two partial realizations ω_1, ω_2. $\omega_1 = \emptyset$, which means that the attacker has not sent any friend requests. $\omega_2 = \{v_2, (v_1, v_2)\}$, which means that the attacker already sent a friend request to v_2. v_2 accepted the request, thus s observed that the edge (v_1, v_2) exists. Clearly $\omega_1 \subseteq \omega_2$. Based on the above definitions, $\Delta(v_1|\omega_1) = 0$, since v_1 will not accept the friend request in any realization. However, $\Delta(v_1|\omega_2) = B_f(v_1) - B_{fof}(v_1) > 0$ since v_1 will accept the friend request from s when its friend v_2 is also a friend of s. Since $\Delta(v_1|\omega_2) > \Delta(v_1|\omega_1)$ and $\omega_1 \subseteq \omega_2$, adaptive submodularity does not hold.

Because the objective is non-submodular, the immediate idea is to adopt an existing proof technique for obtaining an approximation ratio for adaptive non-submodular problems.

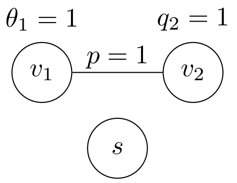


Fig. 1: An example showing the benefit function of ACCU is non-submodular

One such technique was the adaptive total primal curvature defined in [6], [7]. It captures the total change in the marginal gain of befriending a user u under realizations ω, ω' such that $\omega \subseteq \omega'$.

The adaptive total primal curvature can be calculated as

$$\Gamma(u \mid \omega', \omega) = \frac{\Delta(u \mid \omega')}{\Delta(u \mid \omega)}$$

It was showed in [7] that the greedy algorithm has a $(1-(1-\frac{1}{\delta k})^k)$ approximation ratio for a non-submodular maximization problem, where

$$\delta \geq \Gamma(u \mid \omega', \omega), \forall u, \omega, \omega'$$

.

However, this technique is not able to guarantee a ratio for the greedy algorithm of ACCU. Consider a cautious user u_1 and two partial realizations $\omega_1 = \emptyset$ and ω_2 that satisfies $|N(u) \cap \text{dom}(\omega_2)| \geq \theta_{u_1}$. That is, at least θ_{u_1} users in u_1's neighborhood are friends of the attacker s under partial realization ω_2. Clearly $\omega_1 \subseteq \omega_2$. Based on our friend request acceptance model for cautious users, the request will be accepted with probability 1 under ω_2 and will never be accepted under ω_1. Hence, $\Delta(u_1|\omega_1) = 0$ and $\Delta(u_1|\omega_2) > 0$, which makes $\Gamma(u_1 \mid \omega_2, \omega_1)$ unbounded. Thus, δ, which is the upper bound of all Γs, can be positive infinite. In this case, the ratio $(1-(1-\frac{1}{\delta k})^k)$ will go to 0.

Note that if the friend request acceptance model of cautious users can be more generalized, that is, accept the attacker's request with probability q_1 when the number of mutual friends is less than the threshold and increase the probability to q_2 when the number of mutual friends reaches the threshold, an approximation ratio do exist with $\delta = \max_{u \in V_C} \frac{q_2^u}{q_1^u}$. When q_1^u for all $u \in V_C$ are positive, δ is bounded and we have a non-zero approximation ratio. As a numerical example, when $\delta = 10$ and $k = 20$, the ratio is 0.095. In practice, however, δ is likely to be unbounded as it is highly possible that someone will not accept the friend request from a stranger (so $q_1 = 0$).

With the above analysis, we need a new technique to analyze the performance of the greedy algorithm of ACCU in its current form. The submodularity ratio was introduced in [9] and further discussed in [14]. It is defined for a set function $f(.)$ (which is not adaptive) and is the largest scalar λ such that the following inequality holds:

$$\sum_{u \in T \setminus S} \rho_{\{u\}}(S) \geq \lambda \rho_T(S), \forall T, S \subseteq V \tag{5}$$

In the inequality, for two subsets $S_1, S_2 \subseteq V$, $\rho_{S_1}(S_2) = f(S_1 \cup S_2) - f(S_2)$. As in [14], $\lambda \in [0,1]$ for non-decreasing functions. The higher the λ, the more submodular the non-decreasing function $f(.)$ is. $f(.)$ is submodular when $\lambda = 1$.

Now consider a single realization ϕ of ACCU and two arbitrary sub-realizations of ϕ, ω_1, ω_2. We can rewrite inequality (5) as:

$$\sum_{u \in \text{dom}(\omega_2) \setminus \text{dom}(\omega_1)} \rho_{\{u\}}(\text{dom}(\omega_1)) \geq \lambda \rho_{\text{dom}(\omega_2)}(\text{dom}(\omega_1)),$$
$$\forall \omega_1, \omega_2, \text{ s.t. } \phi \sim \omega_1, \phi \sim \omega_2 \tag{6}$$

We claim that we will always have a positive λ under certain conditions.

Lemma 1. *If $B_f(u) - B_{fof}(u) > 0, \forall u \in V$, λ is strictly positive under realization ϕ.*

Proof. Observe that λ will only reach 0 if the lhs of (6) is 0 and $\Delta(\text{dom}(\omega_2)|\omega_1) > 0$. When $B_f(u) - B_{fof}(u) > 0$ holds for all $u \in V$, the lhs of (6) will never be 0 as each individual $\Delta(u|\omega_1)$ must be positive since $u \notin \text{dom}(\omega_1)$. Therefore, $\lambda > 0$ under realization ϕ. □

We can then define a realization-specific adaptive submodular ratio (RASR) λ_ϕ with inequality (6).

Definition 4 (RASR). *The RASR of function f under realization ϕ is the largest scalar λ_ϕ such that inequality* (6) *holds.*

With the definition of RASR, we have our adaptive submodular ratio definition, as follows.

Definition 5 (Adaptive Submodular Ratio). *The adaptive submodular ratio λ of function f is the smallest RASR of all possible realizations.*

$$\lambda = \min_{\phi} \lambda_\phi$$

Corollary 1. *$\lambda > 0$ if $B_f(u) - B_{fof}(u) > 0, \forall u \in V$.*

Proof. As Lemma 1 holds for all realizations when the conditions $B_f(u) - B_{fof}(u) > 0, \forall u \in V$ are met, $\lambda > 0$ is guaranteed. □

In the following, we will prove the approximation guarantee of our ABM algorithms towards ACCU, under the condition that $w_I = 0$ and $B_f(u) - B_{fof}(u) > 0, \forall u \in V$. The reason why we need $w_I = 0$ is that the theoretical result requires a pure greedy strategy based on the direct benefit function, so the indirect benefits are not considered in the proof. However, we will demonstrate in our experiments that taking indirect benefits into account will improve the performance of the algorithm, so the gap between the solution of the ABM algorithm in general and the optimal solution is smaller than the one stated here in practice.

We will proceed for the approximation ratio in two steps. Firstly, we will develop a relation among the optimal strategy, the adaptive greedy strategy based on our ABM algorithm with $w_I = 0$ and a single greedy selection step. The optimal strategy sends exactly k requests, while the greedy strategy may send an arbitrary number of requests. Secondly, we will remove the single greedy selection step from the formula and obtain a relation between the optimal and greedy strategies, which will serve as the approximation guarantee.

For convenience, we denote $Q(\pi,\phi)$ to be the sequence of users to send friend requests under realization ϕ for policy π. It is equivalent to $\text{dom}(\omega(\pi,\Phi))$ where $\omega(\pi,\phi)$ is the sub-realization resulted from applying policy π to realization ϕ.

For the proof, we introduce the policy concatenation operation. For policies π_1,π_2, the concatenation $\pi_1@\pi_2$ is a policy that first sends requests based on π_1 and then sends requests based on π_2. That is, for a realization ϕ, we first obtain the two sequences $Q(\pi_1,\phi), Q(\pi_2,\phi)$ based on the individual policies, next send out the requests to the users in $Q(\pi_1,\phi)$ sequentially, then send requests to the users in $Q(\pi_2,\phi)\backslash Q(\pi_1,\phi)$ while maintaining the original order.

We start with the commutative property of policy concatenation, in the case of greedy and optimal policies.

Lemma 2. *For greedy policy π_1, optimal policy π_2 and realization ϕ, we have $f(\pi_1@\pi_2,\phi)=f(\pi_2@\pi_1,\phi)$ when $B_f(u)-B_{fof}(u)>0,\forall u\in V$.*

Proof. First of all, both policies $\pi_{12}=\pi_1@\pi_2$ and $\pi_{21}=\pi_2@\pi_1$ will send requests to the same set of users $Q(\pi_1,\phi)\cup Q(\pi_2,\phi)$ and their only difference is the order of friend requests that are sent. Clearly, if $F(\pi_{12},\phi)=F(\pi_{21},\phi)$, then the benefits of the two strategies are the same. Hence, in order for the benefits to be different, we must see a difference in $F(\pi_{12},\phi)$ and $F(\pi_{21},\phi)$.

Without loss of generality, let's assume that there exists a user v such that $v\in F(\pi_{12},\phi)$ but $v\notin F(\pi_{21},\phi)$. We claim that the user must not belong to the reckless user set V_R. As users in V_R accept friend requests based on probability, when the requests to them are sent has no impact on whether the requests are accepted or not. For the same realization, we must have v in both $F(\pi_{12},\phi), F(\pi_{21},\phi)$ or in neither of them. Then, such a user must belong to the cautious user set V_C.

We then claim that it is not possible for such a user to exist. If the request to v is rejected when executing policy π_{21}, it means that it was rejected when executing a policy that has it in sequence Q. [1] However, as the friend request acceptance model for cautious users is deterministic, any policy should know that the request will be rejected before it was sent. Hence, when there exists enough users in V_R to befriend with and to collect benefits, this situation will never happen for the greedy algorithm, which aims at the largest marginal gain, or for the optimal solution. In the extreme case that both greedy and optimal policies have to send requests to cautious users without getting benefit, the requests in π_{12} or π_{21} will both be rejected and will not impact the equation. Based on the above arguments, the commutative property of concatenation for greedy and optimal policies is proved. □

[1] It is possible that both policies decide to send request to user v, but this fact is not necessary for the claim.

Next, we consider the optimal policy π_k^* that sends k requests and the greedy policy π_l^g that sends l requests. Let $f_{\text{avg}}(\pi)=\mathbb{E}[f(\pi,\Phi)|\Phi]$ for simplicity.

Lemma 3.

$$f_{avg}(\pi_k^*)-f_{avg}(\pi_l^g)\le\frac{k}{\lambda}(f_{avg}(\pi_{l+1}^g)-f_{avg}(\pi_l^g))$$

Proof. It is clear that f is monotone. Hence, we have

$$f_{\text{avg}}(\pi_k^*)\le f_{\text{avg}}(\pi_k^*@\pi_l^g)=f_{\text{avg}}(\pi_l^g@\pi_k^*)\tag{7}$$

The equality is due to Lemma 2. Then,

$$\begin{aligned}
&f_{\text{avg}}(\pi_l^g@\pi_k^*)\\
&=\mathbb{E}[f(\pi_l^g@\pi_k^*,\Phi)|\Phi]\\
&=\mathbb{E}[f(\pi_l^g,\Phi)+\rho_{Q(\pi_k^*,\Phi)}(Q(\pi_l^g,\Phi))|\Phi]\\
&\le\mathbb{E}[f(\pi_l^g,\Phi)+\frac{1}{\lambda_\Phi}\sum_{u\in Q(\pi_k^*,\Phi)\backslash Q(\pi_l^g,\Phi)}\rho_{\{u\}}(Q(\pi_l^g,\Phi))|\Phi] &(8)\\
&\le\mathbb{E}[f(\pi_l^g,\Phi)+\frac{1}{\lambda}\sum_{u\in Q(\pi_k^*,\Phi)\backslash Q(\pi_l^g,\Phi)}\rho_{\{u\}}(Q(\pi_l^g,\Phi))|\Phi] &(9)\\
&\le\mathbb{E}[f(\pi_l^g,\Phi)+\frac{k}{\lambda}\rho_{\{Q(\pi_{l+1}^g,\Phi)\backslash Q(\pi_l^g,\Phi)\}}(Q(\pi_l^g,\Phi))|\Phi] &(10)\\
&=\mathbb{E}[f(\pi_l^g,\Phi)|\Phi]\\
&\quad+\mathbb{E}[\frac{k}{\lambda}(f(\pi_{l+1}^g,\Phi)-f(\pi_l^g,\Phi))|\Phi]\\
&=f_{\text{avg}}(\pi_l^g)+\frac{k}{\lambda}(f_{\text{avg}}(\pi_{l+1}^g)-f_{\text{avg}}(\pi_l^g)) &(11)
\end{aligned}$$

Inequality (8) is due to the definition of RASR and inequality (9) is based on adaptive submodular ratio. Inequality (10) is from the property of the greedy algorithm: the marginal benefit from the greedy selection, the single user in set $Q(\pi_{l+1}^g,\Phi)\backslash Q(\pi_l^g,\Phi)$, must be higher than the marginal benefit from all other users that are not selected yet. Combining (7) and (11), we can obtain the desired result. □

Finally, we apply the result of Lemma 3 to obtain the approximation ratio of the greedy algorithm.

Theorem 1.

$$(1-e^{-l\lambda/k})f_{avg}(\pi_k^*)<f_{avg}(\pi_l^g)$$

When $l=k$, Alg. 1 has approximation ratio of $1-e^{-\lambda}$ when $w_I=0$ and $B_f(u)-B_{fof}(u)>0,\forall u\in V$.

Proof. As in [15], denote $\delta_l=f_{\text{avg}}(\pi_k^*)-f_{\text{avg}}(\pi_l^g)$. Then we can rewrite the result of Lemma 3 as:

$$\delta_l\le\frac{k}{\lambda}(\delta_l-\delta_{l+1})$$

Hence, $\delta_{l+1}\le(1-\frac{\lambda}{k})\delta_l$ and we have

$$\delta_l\le(1-\frac{\lambda}{k})^l\delta_0\le e^{-l\lambda/k}\delta_0$$

We utilized the inequality $(1-x)^l < e^{-lx}$ to obtain the result. Since $\delta_0 = f_{\text{avg}}(\pi_k^*)$, we have

$$f_{\text{avg}}(\pi_k^*) - f_{\text{avg}}(\pi_l^g) \leq e^{-l\lambda/k} f_{\text{avg}}(\pi_k^*)$$

Therefore,

$$(1 - e^{-l\lambda/k}) f_{\text{avg}}(\pi_k^*) \leq f_{\text{avg}}(\pi_l^g)$$

When $l = k$, $1 - e^{-l\lambda/k}$ reduces to $1 - e^{-\lambda}$ and hence the result. □

Note that we recovers the result in [9] in the adaptive setting. Although the adaptive submodular ratio is problem-specific, we are able to provide some insights about what λ may be based on the subset of cautious users, V_C.

Observation 1. *When $V_C = \emptyset$, $\lambda = 1$.*

When there exists no cautious users, the problem ACCU degenerates to a problem that is similar to those considered in [4], [5] and is submodular. Hence λ reaches 1.

Based on the observation, λ is only related to the sub-realizations that involve the cautious users. Since λ is the minimum of all λ_ϕs, we can focus on one realization that results in the lowest λ_Φ. When discussing such a realization, we will treat it as a deterministic graph and use notations similar to the original definition of submodular ratio in (5).

We first consider the simplest case that has only one cautious user v_c. Then, we must have $v_c \in T, v_c \notin S$ for the two sets S, T in (5) in order to obtain the minimum λ. If v_c is included in S or in none of the sets, the lhs will be at least $\rho_T(S)$ and λ cannot be the minimum. In this case, we derive the following results for λ.

Lemma 4. *When $V_C = \{v_c\}$, let $B'(u) = B_f(u) - \mathbf{1}_{|N(u)\backslash N(v_c)|>1} B_{fof}(u)$, where $\mathbf{1}_{|N(u)\backslash N(v_c)|>1} B_{fof}(u) = 1$ if $|N(u)\backslash N(v_c)| > 1$ and 0 otherwise.*
If $d_{v_c} = 1$ and $N(v_c) = \{u\}$, then

$$\lambda = \frac{B'(u)}{B_f(v_c) + B'(u)}$$

If $d_{v_c} > 1$, then

$$\lambda = \min\{ \min_{U \in N(v_c), |U| = \theta_{v_c}} \frac{\sum_{u \in U} B'(u)}{B_f(v_c) + \sum_{u \in U} B'(u)}, \tag{12}$$

$$\min_{u^* \in N(v_c)} \frac{B'(u^*)}{B'(v_c) + B'(u^*)} \} \tag{13}$$

Proof. When $d_{v_c} = 1$, we have $v_c, u \in T$ and $v_c, u \notin S$, as in this case we will have v_c accepting the request when sending all requests in T and rejecting the request when the requests are sent individually. In order to minimize λ, we want to reduce the benefit collected from u given the request sent in S. If u has friends other than v_c, we will include those friends in S, so that sending a friend request to u will only obtain $B_f(u) - B_{fof}(u)$ benefit. Considering the fact that u may not have other friends, we write the benefit from u as $B'(u)$. Also, we claim that $T = \{v_c, u\}$. If we add other users to T, we will increase both the denominator and numerator by the same amount, which will result in a larger fraction. Combining the analysis, we obtain the λ value when $d_{v_c} = 1$.

For $d_{v_c} > 1$, we consider two situations. 1) T contains v_c and θ_{v_c} friends of v_c and $S \cap N(v_c) = \emptyset$. In this situation, by similar reasoning as in the case of $d_{v_c} = 1$, we obtain (12). Note that d_{v_c} may be larger than θ_{v_c} and we only need the size θ_{v_c} subset of $N(v_c)$ that results in the smallest λ. Any subset larger than θ_{v_c} will increase λ. 2) T contains v_c and only one user in $N(v_c)$, while S contains exactly $\theta_{v_c} - 1$ friends of v_c. In this situation, we need to find a single friend of v_c to include in T for the smallest λ, hence we obtain (13). Since we cannot determine the relationship between (12) and (13) without the actual benefit values, we take the minimum between them as λ. □

Note that when $B_f(u) - B_{fof}(u) = 0$ for some $u \in V$, it is possible that $\lambda = 0$. That is why we require the condition of $B_f(u) - B_{fof}(u) > 0, \forall u \in V$.

When we have more than 1 cautious users, we can still easily calculate λ when the cautious users do not have mutual friends: we first calculate a λ for each cautious user as if it is the only cautious user as in Lemma 4, then take the minimum of all such λs. However, λ may be very small if one friend is shared by many cautious users.

Lemma 5. *When $u \in \cap_{i=1,\ldots,r} N(v_c^i)$, λ is upper bounded by*

$$\frac{B_f(u)}{\sum_{i=1,\ldots,r} B'(v_c^i) + B_f(u)}$$

Proof. we can easily achieve the upper bound by having $u, v_c^1, \ldots, v_c^r$ in T and let S include exactly $\phi_{v_c^i} - 1$ friends for each cautious user v_c^i. □

IV. Experiment Results

In this section, we first demonstrate the performance of the ABM algorithm by comparing it against several alternative algorithms in various data sets. The data sets are summarized in Table I. Then, we study how the parameters in the ABM algorithm impact the befriending strategy. Finally, we examine the behavior of the ACCU problem by sensitivity analysis.

A. Experiment Setup

Parameter Selection. Throughout the experiments, we assign edge existence probabilities and friend request acceptance probabilities uniformly randomly between $[0, 1)$. The edge existence probabilities are only generated for edges in the data sets. Also, we only generate friend request acceptance probabilities between the attacker and the users in V_R. As for the benefits, we fix $B_f(u) = 2, \forall u \in V_R$ and $B_{fof}(u) = 1, \forall u \in V$. The reason why we keep the friend-of-friend benefit of users in V_R and V_C the same is that the cautious users tend not to share much to their indirect friends. The benefit of successfully befriending a cautious user varies in the experiments. Other variables include the threshold $\theta_v, \forall v \in V_c$ and the weights w_D, w_I of the ABM algorithm.

Cautious User Selection. Since we have limited information of whether the users are cautious or not, we select them

Network	Nodes	Edges	Kind
Facebook	4k	88k	Social
Slashdot	77k	905k	Social
Twitter	81k	1.77M	Social
DBLP	317k	1.05M	Collaboration

TABLE I: Statistics of the data sets. All networks are from SNAP [16].

randomly among nodes having degree within the range of $[10, 100]$. The reasoning behind the filtering is that nodes with really high degrees are not likely to be cautious, while nodes with low degrees are usually not important and can be ignored by the attacker. Also, we iteratively select the cautious users to make sure that there are no direct edges among them. In each network, we select 100 cautious users.

Algorithms for Comparison. We compare ABM with the following algorithms:

- MaxDegree: Iteratively pick the target users with highest degree in the network.
- PageRank: Pick target users in the network with highest PageRank scores.
- Random: Randomly select target users. The result of the random algorithm is averaged over 100 runs to mitigate fluctuation.

Note that we do not explicitly compare with the existing greedy algorithm [3], [2], [6] since ABM converges to the classical greedy with $w_D = 1, w_I = 0$.

Because of the randomness in both data set generation and algorithm execution, we generate 100 sample networks for each data set and run the algorithms 30 times in each sample network. The presented results are averaged over all outputs.

B. Performance of the Algorithms

We first compare ABM with the alternative methods using the most straightforward metric: the amount of benefits collected by the attacker. For all data sets, we let $B_f(u) = 50, \forall u \in V_c$. The threshold θ_v is set as 30% of the degree of the corresponding user. As for the weights for ABM, we set $w_D = w_I = 0.5$ to equally emphasize on the direct and indirect benefits.

In Fig. 2, we illustrate the amount of benefits obtained varying the number of friend requests k in all five data sets. We can observe that ABM has significant better performance comparing with the other algorithms in most of the times. The random algorithm is constantly the worst except for higher number of friend requests in the Facebook data set. The PageRank algorithm and MaxDegree algorithms are in the middle, with PageRank always slightly better than MaxDegree, except for the DBLP dataset, where the difference is much larger. An interesting finding is that the lines of ABM in Slashdot and Twitter share a common behavior: the lines are not perfectly concave. There exists one segment in each line showing convexity. It means that the average marginal benefit gain for sending one friend request within that segment is lower than the gain of some requests sent later.

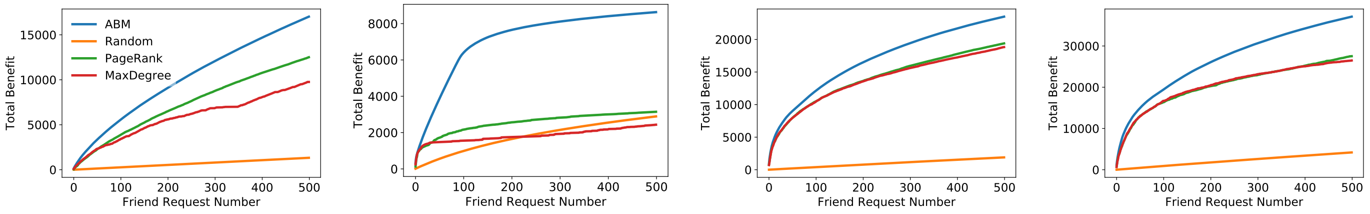
To find the cause for this behavior, we consider the average marginal gain of every friend request, breaking it down to benefits from cautious users and those from reckless users in Fig. 3. The idea behind this change is that the behavior did not show up in existing literature when the cautious users do not exist. From Fig. 3, we conclude that the decrease in marginal gain is directly related to the process of befriending the cautious users. In each data set, the orange region indicates a high concentration of requests sent to cautious users (and hence higher benefits from them). We can observe that the orange regions in Slashdot and Twitter heavily overlap with regions that have lower average marginal gain than later requests (around request 50 for Slashdot and around request 80 for Twitter). The root reason is that the potential function of ABM (with $w_D = w_I = 0.5$) assigns high values to the friends of cautious users when the cautious users are not friends of the attacker yet. The high values lead to ABM sending requests to those users, but the actual gain from them can be low. The reason why Facebook and DBLP did not clearly show this behavior are different. For Facebook, befriending a cautious user is quite significant, since there exists only a few users with very high degree. Therefore, the high gains from cautious users mitigate the low gains from the friends of cautious users, and overall the marginal gain is monotonically decreasing. For DBLP, however, there are too many users with high benefits. Based on the potential function, they are favored over the cautious users. Hence, ABM selects very limited amount of cautious users in DBLP and maintains a monotonically decreasing marginal gain.

C. Impact of Parameter Selection in ABM

In the previous subsection, we notice that ABM may send requests to users that result in lower marginal gain than later requests in some data sets. Since the target selection is determined by the potential function, we will explore how the selection of w_D, w_I affects the performance of the ABM algorithm in this subsection.

We first consider the overall benefit after sending 500 requests and the number of cautious friends in the Twitter data set. We let w_I vary between 0 and 0.6 and set $w_D = 1 - w_I$. The result is displayed in Fig. 4. It is clear that the number of cautious friends grows monotonically with w_I, however, higher w_I may not lead to higher benefit. The benefit peaks at $w_I = 0.2$ and drops when w_I is larger or smaller. Hence, it is necessary to achieve a balance between direct and indirect benefit for better performance. Note that $w_I = 0$ corresponds to the pure greedy strategy as in earlier adaptive crawling papers (e.g. [4]). With a proper parameter setting, ABM outperforms pure greedy in the ACCU problem.

We also illustrate the times that the cautious users become friends of the attacker in Fig. 5. Specifically, we consider the fraction of times request X is sent to a cautious user among all generated graphs and algorithm execution repetitions, where $X \in [1, 500]$. We can observe that with a higher w_I, ABM not only chooses to befriend more cautious users, but also tends to befriend the cautious users earlier. When w_I is large,



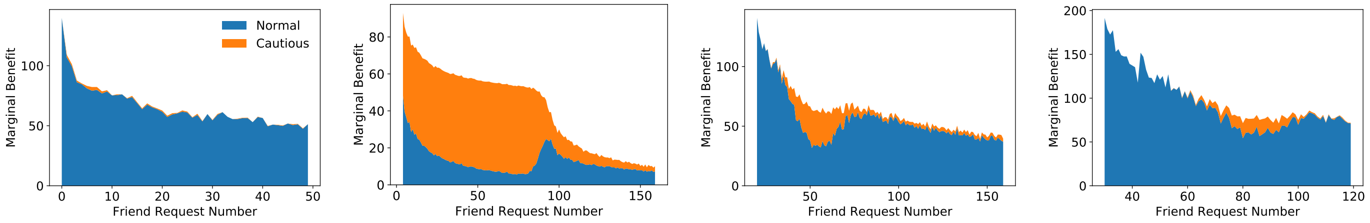
(a) DBLP

(b) Facebook

(c) Slashdot

(d) Twitter

Fig. 2: Amount of benefits obtained



(a) DBLP

(b) Facebook

(c) Slashdot

(d) Twitter

Fig. 3: Average marginal benefit from cautious and reckless users.

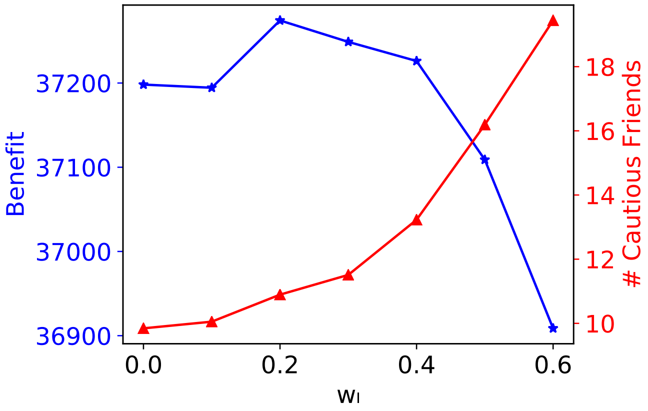


Fig. 4: Benefits and # cautious users obtained using ABM in Twitter, varying w_I.

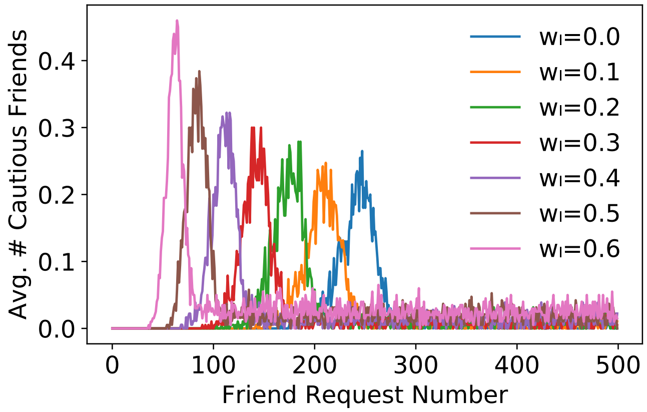


Fig. 5: Fraction of requests sent to cautious users.

ABM may over-emphasize the importance of cautious users and prioritize befriending them, which may be detrimental for overall benefit.

D. Sensitivity Analysis of the Modeling Parameters

In this subsection, we test the impact of two modeling parameters: the friend benefit $B_f(u)$ and the acceptance threshold of the cautious users.

In the following figures, we use heat maps to measure the collected benefit and the number of cautious friends when both parameters are changing. The experiments are run on Twitter data set with $k = 500, w_D = w_I = 0.5$.

In Fig. 6 and Fig. 7, we illustrate how the collected benefit and number of cautious friends change with the two modeling parameters. In general, higher friend benefit and lower acceptance threshold for cautious users lead to higher total benefit and more cautious friends. The only exception is when the friend benefit is only 20 for cautious users. In this case, if we make befriending cautious users harder by increasing the acceptance threshold, we actually obtain higher benefit. The reason is similar to what we discussed in the earlier section: over-emphasizing the importance of cautious users can sometimes degrade performance.

V. Conclusion

In this paper, we introduced the cautious users to consider the Adaptive Crawling problem in a more realistic setting. The friend request acceptance model of the cautious users broke submodularity in existing problems, making the new problem,

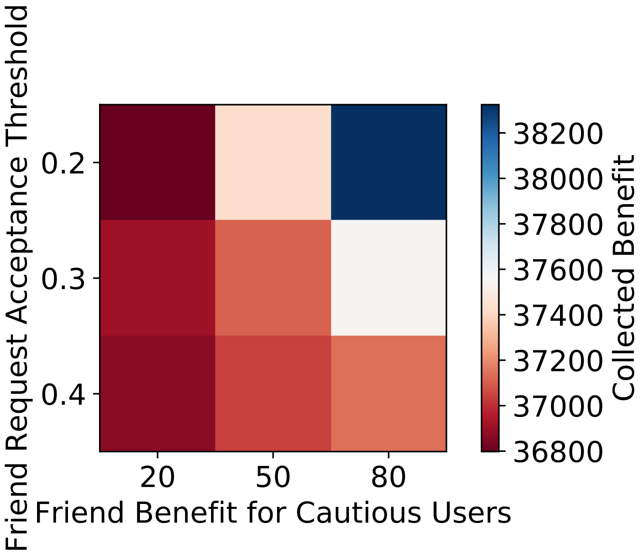


Fig. 6: Heat map for benefits when varying friend benefit and acceptance threshold for cautious users.

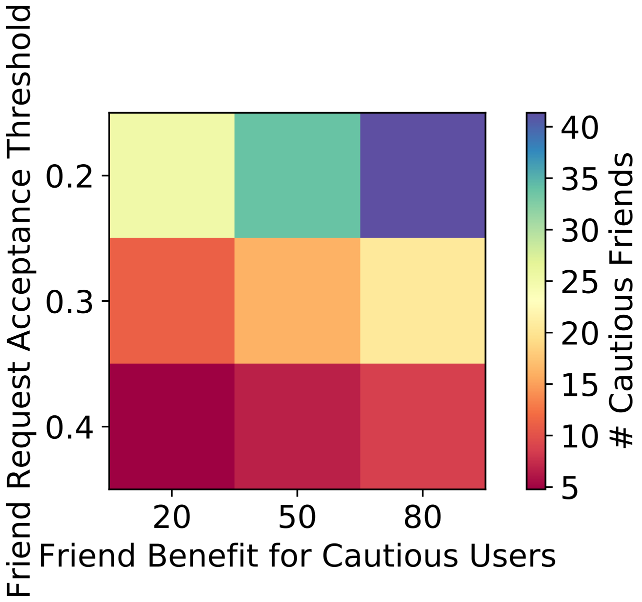


Fig. 7: Heat map for # cautious friends when varying friend benefit and acceptance threshold for cautious users.

named ACCU, more complicated. Nonetheless, we proposed a new notion, the adaptive submodular ratio, based on which we could provide a theoretical bound for the greedy algorithm that solves ACCU in certain conditions. We evaluated the performance of our algorithm, ABM, in various data sets, demonstrated its advantage over a few alternative algorithms. Also, we performed sensitivity analysis to provide insights on the ACCU problem.

References

[1] E. Novak and Q. Li, "A survey of security and privacy in online social networks," *College of William and Mary Computer Science Technical Report*, pp. 1–32, 2012.

[2] X. Li, J. D. Smith, T. N. Dinh, and M. T. Thai, "Privacy issues in light of reconnaissance attacks with incomplete information," in *Web Intelligence (WI), 2016 IEEE/WIC/ACM International Conference on*. IEEE, 2016, pp. 311–318.

[3] H. T. Nguyen and T. N. Dinh, "Targeted Cyber-attacks: Unveiling Target Reconnaissance Strategy via Social Networks," in *Proceedings of the IEEE Int Conf. on Computer Com., Security and Privacy in BigData Workshop*, ser. INFOCOM BigSecurity 2016, 2016.

[4] X. Li, J. D. Smith, and M. T. Thai, "Adaptive reconnaissance attacks with near-optimal parallel batching," in *Distributed Computing Systems (ICDCS), 2017 IEEE 37th International Conference on*. IEEE, 2017, pp. 699–709.

[5] X. Li, J. D. Smith, T. N. Dinh, and M. T. Thai, "Adaptive crawling with multiple bots: A matroid intersection approach," in *INFOCOM 2018-IEEE Conference on Computer Communications, IEEE*. IEEE, 2018, pp. 1–9.

[6] X. Li, J. D. Smith, T. Pan, T. N. Dinh, and M. T. Thai, "Quantifying privacy vulnerability to socialbot attacks: An adaptive non-submodular model," *IEEE Transactions on Emerging Topics in Computing*, 2018.

[7] J. D. Smith, A. Kuhnle, and M. T. Thai, "An approximately optimal bot for non-submodular social reconnaissance," in *Proceedings of the 29th on Hypertext and Social Media*. ACM, 2018, pp. 192–200.

[8] Y. Boshmaf, I. Muslukhov, K. Beznosov, and M. Ripeanu, "The Socialbot Network: When Bots Socialize for Fame and Money," in *Proceedings of the 27th Annual Computer Security Applications Conference*, ser. ACSAC '11. ACM, 2011, pp. 93–102.

[9] A. Das and D. Kempe, "Submodular meets spectral: Greedy algorithms for subset selection, sparse approximation and dictionary selection," *arXiv preprint arXiv:1102.3975*, 2011.

[10] T. Ryan and G. Mauch, "Getting in bed with robin sage," in *Black Hat Conference*, 2010, pp. 1–8.

[11] A. Elyashar, M. Fire, D. Kagan, and Y. Elovici, "Homing socialbots: intrusion on a specific organization's employee using socialbots," in *Proceedings of the 2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*. ACM, 2013, pp. 1358–1365.

[12] I. Jeun, Y. Lee, and D. Won, "A Practical Study on Advanced Persistent Threats," in *Computer Applications for Security, Control and System Engineering*, ser. Communications in Computer and Information Science. Springer Berlin Heidelberg, 2012, no. 339, pp. 144–152.

[13] Z. Wang, B. Moran, X. Wang, and Q. Pan, "Approximation for maximizing monotone non-decreasing set functions with a greedy method," *Journal of Combinatorial Optimization*, vol. 31, no. 1, pp. 29–43, 2016.

[14] A. A. Bian, J. M. Buhmann, A. Krause, and S. Tschiatschek, "Guarantees for greedy maximization of non-submodular functions with applications," *arXiv preprint arXiv:1703.02100*, 2017.

[15] D. Golovin and A. Krause, "Adaptive submodularity: Theory and applications in active learning and stochastic optimization," *Journal of Artificial Intelligence Research*, pp. 427–486, 2011.

[16] J. Leskovec and A. Krevl, "SNAP Datasets: Stanford large network dataset collection," http://snap.stanford.edu/data, Jun. 2014.