Robust Influence Maximization

Wei Chen
Microsoft Research
weic@microsoft.com

Tian Lin
Tsinghua University
lint10@mails.tsinghua.edu.cn

Zihan Tan
IIIS, Tsinghua University
zihantan1993@gmail.com

Mingfei Zhao
IIIS, Tsinghua University
mingfeizhao@hotmail.com

Xuren Zhou
The Hong Kong University of Science and Technology
xzhouap@cse.ust.hk

ABSTRACT

In this paper, we address the important issue of uncertainty in the edge influence probability estimates for the well studied influence maximization problem — the task of finding k seed nodes in a social network to maximize the influence spread. We propose the problem of robust influence maximization, which maximizes the worst-case ratio between the influence spread of the chosen seed set and the optimal seed set, given the uncertainty of the parameter input. We design an algorithm that solves this problem with a solution-dependent bound. We further study uniform sampling and adaptive sampling methods to effectively reduce the uncertainty on parameters and improve the robustness of the influence maximization task. Our empirical results show that parameter uncertainty may greatly affect influence maximization performance and prior studies that learned influence probabilities could lead to poor performance in robust influence maximization due to relatively large uncertainty in parameter estimates, and information cascade based adaptive sampling method may be an effective way to improve the robustness of influence maximization.

Keywords

social networks, influence maximization, robust optimization, information diffusion

1. INTRODUCTION

In social and economic networks, *Influence Maximization* problem has been extensively studied over the past decade, due to its wide applications to viral marketing [13, 19], outbreak detection [22], rumor monitoring [6], etc. For example, a company may conduct a promotion campaign in social networks by sending free samples to the initial users (termed as seeds), and via the word-of-mouth (WoM) effect, more and more users are influenced by social links to join the campaign and propagate messages of the promotion. This problem is first introduced by Kempe et al. [19] under an algorithmic framework to find the most influential seeds, and they propose the *independent cascade* model and *linear threshold* model, which consider the social-psychological factors of information diffusion to simulate such a random process of adoptions.

Since Kempe et al.'s seminal work, extensive researches have been done on influence maximization, especially on improving the efficiency of influence maximization in the independent cascade model [11, 10, 16, 4, 28], all of which assume that the ground-truth influence probabilities on edges are exactly known. Separately, a number of studies [26, 27, 15, 25, 24] propose learning methods to extract edge influence probabilities. Due to inherent data limitation, no learning method could recover the exact values of the edge probabilities, and what can be achieved is the estimates on the true edge probabilities, with confidence intervals indicating that the true values are within the confidence intervals with high probability. The uncertainty in edge probability estimates, however, may adversely affect the performance of the influence maximization task, but this topic has left mostly unexplored. The only attempt addressing this question is a recent study in [18], but due to a technical issue as explained in [18], the results achieved by the study is rather limited.

In this paper, we utilize the concept of robust optimization [3] in operation research to address the issue of influence maximization with uncertainty. In particular, we consider that the input to the influence maximization task is no longer edge influence probability on every edge of a social graph, but instead an interval in which the true probability may lie. Thus the input is actually a parameter space Θ, which is the product of all intervals on all edges. For any seed set S, let $\sigma_\theta(S)$ denote the *influence spread* of S under parameter setting $\theta \in \Theta$. Then we define *robust ratio* of S as $g(\Theta, S) = \min_{\theta \in \Theta} \frac{\sigma_\theta(S)}{\sigma_\theta(S^*_\theta)}$, where S^*_θ is the optimal seed set achieving the maximum influence spread under parameter θ. Intuitively, robust ratio of S indicates the (multiplicative) gap between its influence spread and the optimal influence spread under the worse-case parameter $\theta \in \Theta$, since we are unsure which $\theta \in \Theta$ is the true probability setting. Then our optimization task is to find a seed set of size k that maximize the robust ratio under the known parameter space Θ — we call this task *Robust Influence Maximization (RIM)*.

It is clear that when there is no uncertainty on edge probabilities, which means Θ collapses to the single true parameter θ, RIM degenerates to the classical influence maximization problem. However, when uncertainty exists, solving RIM may be a more difficult task. In this paper, we first pro-

KDD '16, August 13 - 17, 2016, San Francisco, CA, USA

ISBN 978-1-4503-4232-2/16/08…$15.00

DOI: http://dx.doi.org/10.1145/2939672.2939745

pose an algorithm LUGreedy that solves the RIM task with a solution-dependent bound on its performance, which means that one can verify its performance after it selects the seed set (Section 3). We then show that if the input parameter space Θ is only given and cannot be improved, it is possible that even the best robust ratio in certain graph instances could be very small (e.g. $O(\log n/\sqrt{n})$ with n being the number of nodes in the graph). This motivates us to study sampling methods to further tighten parameter space Θ, and thus improving the robustness of our algorithm (Section 4). In particular, we study both uniform sampling and adaptive sampling for improving RIM performance. For uniform sampling, we provide theoretical results on the sample complexity for achieving a given robust ratio of the output seed set. For adaptive sampling, we propose an information cascade based sampling heuristic to adaptively bias our sampling effort to important edges often traversed by information cascades. Through extensive empirical evaluations (Section 5), we show that (a) robust ratio is sensitive to the width of the confidence interval, and it decreases rapidly when the width of the confidence interval increases; as a result prior studies that learned edge probabilities may result in poor robust ratio due to relative large confidence intervals (and thus high uncertainty); (b) information cascade based adaptive sampling method performs better than uniform sampling and other baseline sampling methods, and can significantly improve the robustness of the influence maximization task.

In summary, the contribution of our paper includes: (a) proposing the problem of robust influence maximization to address the important issue of uncertainty in parameter estimates adversely impacting the influence maximization task; (b) providing the LUGreedy algorithm that guarantees a solution-dependent bound; and (c) studying uniform and adaptive sampling methods to improve robust influence maximization.

Due to space constraint, the proofs of some technical results are omitted. The complete proofs of all results can be found in the full technical report [9].

1.1 Additional Related Work

Influence maximization has been extensively studied and we already point out a number of closely related studies to our work in the introduction. For a comprehensive survey, one can refer to the monograph [8]. We discuss a few most relevant work in more detail here.

To the best of our knowledge, the study by He and Kempe [18] is the only attempt prior to our work that also tries to address the issue of uncertainty of parameter estimates impacting the influence maximization tasks. However, besides the similarity in motivation, the technical treatments are quite different. First, their central problem, called influence difference maximization, is to find a seed set of size k that maximizes the additive difference between the two influence spreads of the *same* seed set using different parameter values. Their purpose is to see how large the influence gap could be due to the uncertainty in parameter space. However, our goal is still to find the best possible seed set for influence maximization purpose, while considering the adverse effect of the uncertainty, and thus we utilize the robust optimization concept and use the worse-case multiplicative ratio between the influence spread of the chosen seed set and the optimal seed set as our objective function. Second, their influence difference maximization turns out to be hard to approximate to any reasonable ratio, while we provide an actual algorithm for robust influence maximization that has both a theoretical solution-dependent bound and performs reasonably well in experiments. Third, we further consider using sampling methods to improve RIM, which is not discussed in [18].

In the context of robust optimization, Krause et al.'s work on robust submodular optimization [20] is possibly the closest to ours. Our RIM problem can be viewed as a specific instance of robust submodular optimization studied in [20]. However, due to the generality of problem scope studied in [20], they show strong hardness results and then they have to resolve to bi-criteria solutions. Instead, we are working on a particular instance of robust submodular optimization, and their bi-criteria solution may greatly enlarge the selected seed set size, which may not be allowed in our case. Furthermore, they work on finite set of submodular functions, but in our case our objective function is parametrized with θ from a continuous parameter space Θ, and it is unclear how their results work for the continuous case.

In a parallel work that will appear in the same proceeding, He and Kempe study the same subject of robust influence maximization [17], but they follow the bi-criteria approximation approach of [20], and thus in general their results are orthogonal to ours. In particular, they use essentially the same objective function, but they work on a finite set of influence spread functions Σ, and require to find $k \cdot \ln |\Sigma|$ seeds to achieve $1 - 1/e$ approximation ratio comparing to the optimal seed set of size k; when working on continuous parameter space Θ, they show that it is equivalent to a finite spread function space of size 2^n and thus requiring $\Theta(kn)$ seeds for a bi-criteria solution, which renders the bi-criteria solution useless. Thus their bi-criteria approach is suitable when the set of possible spread functions Σ is small.

Adaptive sampling for improving RIM bears some resemblance to pure exploration bandit research [5], especially to combinatorial pure exploration [7] recently studied. Both use adaptive sampling and achieve some optimization objective in the end. However, the optimization problem modeled in combinatorial pure exploration [7] does not have a robustness objective. Studying robust optimization together with combinatorial pure exploration could be a potentially interesting topic for future research. Another recent work [21] uses online algorithms to maximize the expected coverage of the union of influenced nodes in multiple rounds based on online feedbacks, and thus is different from our adaptive sampling objective: we use feedbacks to adjust adaptive sampling in order to find a seed set nearly maximizing the robust ratio after the sampling is done.

2. MODEL AND PROBLEM DEFINITION

As in [19], the *independent cascade (IC)* model can be equivalently modeled as a stochastic diffusion process from seed nodes or as reachability from seed nodes in random live-edge graphs. For brevity, we provide the live-edge graph description below. Consider a graph $G = (V, E)$ comprising a set V of nodes and a set E of directed edges, where every edge e is associated with probability $p_e \in [0, 1]$, and let $n = |V|$ and $m = |E|$. To generate a random live-edge graph, we declare each edge e as *live* if flipping a biased random coin with probability p_e returns success, declare e as *blocked* otherwise (with probability $1 - p_e$). The randomness on all edges are mutually independent. We define the subgraph L

Algorithm 1 Greedy(G, k, θ)

Input: Graph G, budget k, parameter vector θ

$S_0 \leftarrow \emptyset$
for $i = 1, 2, \ldots, k$ **do**
  $v \leftarrow \arg\max_{v \notin S_i} \{\sigma_\theta(S_{i-1} \cup \{v\}) - \sigma_\theta(S_{i-1})\}$
  $S_i \leftarrow S_{i-1} \cup \{v\}$
end for
return S_k

consisting of V and the set of live edges as the (random) *live-edge graph*. Given any set $S \subseteq V$ (referred as *seeds*), let $R_L(S) \subseteq V$ denote the *reachable set* of nodes from S in live-edge graph L, i.e., (1) $S \subseteq R_L(S)$, and (2) for a node $v \notin S$, $v \in R_L(S)$ iff there is a path in L directing from some node in S to v.

For convenience, we use *parameter vector* $\theta = (p_e)_{e \in E}$ to denote the probabilities on all edges. The *influence spread* function $\sigma_\theta(S)$ is defined as the expected size of the reachable set from S, that is

$$\sigma_\theta(S) := \sum_L \Pr_\theta[L] \cdot |R_L(S)|,$$

where $\Pr_\theta[L]$ is the probability of yielding live-edge graph L under vector θ. From [19], we know that the influence spread function is non-negative ($\forall S \subseteq V$, $\sigma_\theta(S) \geq 0$), monotone ($\forall S \subseteq T \subseteq V$, $\sigma_\theta(S) \leq \sigma_\theta(T)$), and submodular ($\forall S \subseteq T \subseteq V$, $\forall v \in V$ $\sigma_\theta(S \cup \{v\}) - \sigma_\theta(S) \geq \sigma_\theta(T \cup \{v\}) - \sigma_\theta(T)$).

The well-known problem of *Influence Maximization* raised in [19] is stated in the following.

Problem 1 (Influence Maximization [19]). *Given a graph $G = (V, E)$, parameter vector $\theta = (p_e)_{e \in E}$ and a fixed budget k, we are required to find a seed set $S \subseteq V$ of k vertices, such that the influence spread function $\sigma_\theta(S)$ is maximized, that is,*

$$S^*_\theta := \arg\max_{S \subseteq V, |S| = k} \sigma_\theta(S).$$

It has been shown that Influence Maximization problem is NP-hard [19]. Since the objective function $\sigma_\theta(S)$ is submodular, we have a $(1 - \frac{1}{e})$ approximation using standard greedy policy Greedy(G, k, θ) in Algorithm 1 (assuming a value oracle on function $\sigma_\theta(\cdot)$). Let S^g_θ be the solution of Greedy(G, k, θ). As a convention, we assume that both optimal seed set S^*_θ and greedy seed set S^g_θ in this paper are of fixed size k implicitly.

On the other hand, it is proved by Feige [14] that such an approximation ratio could not be improved for k-max cover problem, which is a special case of the influence maximization problem under the IC model.

However, the knowledge of the probability on edges is usually acquired by learning from the real-world data [26, 27, 15, 25, 24], and the obtained estimates always have some inaccuracy comparing to the true value. Therefore, it is natural to assume that, from observations of edge e, we can obtain the statistically significant neighborhood $[l_e, r_e]$, i.e., the *confidence interval* where the true probability p_e lies in with high probability. This confidence interval prescribes the uncertainty on the true probability p_e of the edge e, and such uncertainty on edges may adversely impact the influence maximization task. Motivated by this, we study the problem of *robust influence maximization* as specified below.

Suppose for every edge e, we are given an interval $[l_e, r_e]$ ($0 \leq l_e \leq r_e \leq 1$) indicating the range of the probability, and the ground-truth probability $p_e \in [l_e, r_e]$ of this edge is unknown. Denote $\Theta = \times_{e \in E}[l_e, r_e]$ as the *parameter space* of network G, and $\theta = (p_e)_{e \in E}$ as the latent parameter vector. Specifically, let $\theta^-(\Theta) = (l_e)_{e \in E}$ and $\theta^+(\Theta) = (r_e)_{e \in E}$ as the minimum and maximum parameter vectors, respectively, and when the context is clear, we would only use θ^- and θ^+. For a seed set $S \subseteq V$ and $|S| = k$, define its *robust ratio* under parameter space Θ as

$$g(\Theta, S) := \min_{\theta \in \Theta} \frac{\sigma_\theta(S)}{\sigma_\theta(S^*_\theta)}, \tag{1}$$

where S^*_θ is the optimal solution of size k when the probability on every edge is given by θ.

Given Θ and solution S, the robust ratio $g(\Theta, S)$ characterizes the *worst-case* ratio of influence spread of S and the underlying optimal one, when the true probability vector θ is unknown (except knowing that $\theta \in \Theta$). Then, the *Robust Influence Maximization* (RIM) problem is defined as follows.

Problem 2 (Robust Influence Maximization). *Given a graph $G = (V, E)$, parameter space $\Theta = \times_{e \in E}[l_e, r_e]$ and a fixed budget k, we are required to find a set $S \subseteq V$ of k vertices, such that robust ratio $g(\Theta, S)$ is maximized, i.e.,*

$$S^*_\Theta := \arg\max_{S \subseteq V, |S| = k} g(\Theta, S) = \arg\max_{S \subseteq V, |S| = k} \min_{\theta \in \Theta} \frac{\sigma_\theta(S)}{\sigma_\theta(S^*_\theta)}.$$

The objective of this problem is to find a seed set S^*_Θ that has the largest robust ratio, that is, S^*_Θ should maximize the worst-case ratio between its influence spread and the optimal influence spread, when the true probability vector θ is unknown. When there is no uncertainty, which means Θ collapses to the true probability θ, we can see that the RIM problem is reduced back to the original influence maximization problem.

In RIM, the knowledge of the confidence interval is assumed to be the input. Another interpretation is that, it can be viewed as given an estimate of probability vector $\hat{\theta} = (\hat{p}_e)_{e \in E}$ with a perturbation level δ_e on each edge e, such that the true probability $p_e \in [\hat{p}_e - \delta_e, \hat{p}_e + \delta_e] = [l_e, r_e]$, which constitutes parameter space $\Theta = \times_{e \in E}[l_e, r_e]$. Notice that, in reality, this probability could be obtained via edge samplings, i.e., we make samples on edges and compute the fraction of times that the edge is live. On the other hand, we can also observe information cascades on each edge when collecting the trace of diffusion in the real world, so that the corresponding probability can be learned.

However, when the amount of observed information cascade is small, the best robust ratio $\max_S g(\Theta, S)$ for the given Θ can be low so that the output for a RIM algorithm does not have a good enough guarantee of the performance in the worst case. Then a natural question is, given Θ, how to further make samples on edges (e.g., activating source node u of an edge (u, v) and see if the sink node v is activated through edge e) so that $\max_S g(\Theta, S)$ can be efficiently improved? To be specific, how to make samples on edges and output Θ' and S' according to the outcome so that (a) with high probability the true value θ lies in the output parameter space Θ', where the randomness is taken according to θ, and (b) $g(\Theta', S')$ is large. This sub-problem is called *Sampling for Improving Robust Influence Maximization*, and will be addressed in Section 4.

Algorithm 2 LUGreedy(G, k, Θ)

Input: Graph $G = (V, E)$, budget k, parameter space $\Theta = \times_{e\in E}[l_e, r_e]$

1: $S^g_{\theta^-} \leftarrow$ Greedy(G, k, θ^-)
2: $S^g_{\theta^+} \leftarrow$ Greedy(G, k, θ^+)
3: **return** $\arg\max_{S\in\{S^g_{\theta^-}, S^g_{\theta^+}\}} \{\sigma_{\theta^-}(S)\}$

3. ALGORITHM AND ANALYSIS FOR RIM

Consider the problem of RIM, parameter space $\Theta = \times_{e\in E}[l_e, r_e]$ is given, and we do not know the true probability $\theta \in \Theta$. Let $\theta^- = (l_e)_{e\in E}$ and $\theta^+ = (r_e)_{e\in E}$.

Our first observation is that, when Θ is a single vector ($l_e = r_e$, $\forall e \in E$), it is trivially reduced to the classical Influence Maximization problem. Therefore, we still have the following hardness result on RIM [19, 14]:

THEOREM 1. *RIM problem is* **NP***-hard, and for any $\varepsilon > 0$, it is* **NP***-hard to find a seed set S with robust ratio at least $1 - \frac{1}{e} + \varepsilon$.*

To circumvent the above hardness result, we develop algorithms that achieves reasonably large robust ratio. When we are not allowed to make new samples on the edges to improve the input interval, it is natural to utilize the greedy algorithm of submodular maximization in [19] (i.e., Algorithm 1) as the subroutine to calculate the solution. In light of this, we first propose Lower-Upper Greedy Algorithm and the solution-dependent bound for $g(\Theta, S)$, and then discuss $g(\Theta, S)$ in the worst-case scenario.

3.1 Lower-Upper Greedy Algorithm

Given parameter space $\Theta = \times_{e\in E}[l_e, r_e]$ with the minimum and maximum parameter vectors $\theta^- = (l_e)_{e\in E}$ and $\theta^+ = (r_e)_{e\in E}$, our *Lower-Upper Greedy algorithm* (LUGreedy(G, k, Θ)) is described in Algorithm 2 which outputs the best seed set S^{LU}_Θ for the minimum parameter vector θ^- such that

$$S^{\mathsf{LU}}_\Theta := \arg\max_{S\in\{S^g_{\theta^-}, S^g_{\theta^+}\}} \{\sigma_{\theta^-}(S)\}. \tag{2}$$

To evaluate the performance of this output, we first define the *gap ratio* $\alpha(\Theta) \in [0, 1]$ of the input parameter space to be

$$\alpha(\Theta) := \frac{\sigma_{\theta^-}(S^{\mathsf{LU}}_\Theta)}{\sigma_{\theta^+}(S^g_{\theta^+})}. \tag{3}$$

Then, LUGreedy achieves the following result:

THEOREM 2 (SOLUTION-DEPENDENT BOUND). *Given a graph G, parameter space Θ and budget limit k,* LUGreedy *outputs a seed set S^{LU}_Θ of size k such that*

$$g(\Theta, S^{\mathsf{LU}}_\Theta) \ge \alpha(\Theta)\left(1 - \frac{1}{e}\right),$$

where $\alpha(\Theta) := \frac{\sigma_{\theta^-}(S^{\mathsf{LU}}_\Theta)}{\sigma_{\theta^+}(S^g_{\theta^+})}$.

PROOF. For any seed set S, $g(\Theta, S) = \min_{\theta\in\Theta} \frac{\sigma_\theta(S)}{\sigma_\theta(S^g_\theta)}$ by definition. Obviously, it is a fact that $\sigma_\theta(S)$ is monotone on θ for any fixed S. From the definition of optimal solutions and the greedy algorithm, we can get $\sigma_\theta(S^*_\theta) \le \sigma_{\theta^+}(S^*_\theta) \le \sigma_{\theta^+}(S^*_{\theta^+}) \le \frac{\sigma_{\theta^+}(S^g_{\theta^+})}{1-1/e}$. Moreover, it can be implied that

$$g(\Theta, S) \ge \min_{\theta\in\Theta} \frac{\sigma_\theta(S)}{\sigma_{\theta^+}(S^g_{\theta^+})}\left(1 - \frac{1}{e}\right) = \frac{\sigma_{\theta^-}(S)}{\sigma_{\theta^+}(S^g_{\theta^+})}\left(1 - \frac{1}{e}\right).$$

Use seed set S^{LU}_Θ from LUGreedy, and it follows immediately that $g(\Theta, S^{\mathsf{LU}}_\Theta) \ge \frac{\sigma_{\theta^-}(S^{\mathsf{LU}}_\Theta)}{\sigma_{\theta^+}(S^g_{\theta^+})}\left(1 - \frac{1}{e}\right) = \alpha(\Theta)\left(1 - \frac{1}{e}\right)$. □

We refer $\alpha(\Theta)(1 - \frac{1}{e})$ as the *solution-dependent bound* of $g(\Theta, S^{\mathsf{LU}}_\Theta)$ that LUGreedy achieves, because it depends on the solution S^{LU}_Θ. The good thing is that it can be evaluated once we have the solution, and then we know the robust ratio must be at least this lower bound. Note that the bound is good if $\alpha(\Theta)$ is not too small, and thus it in turn indicates that the influence spread $\sigma_\theta(S^{\mathsf{LU}}_\Theta)$ we find has a good performance under any probability vector $\theta \in \Theta$.

It is worth remarking that the choice of using $\alpha(\Theta) = \sigma_{\theta^-}(S^{\mathsf{LU}}_\Theta)/\sigma_{\theta^+}(S^g_{\theta^+})$ as a measurement is for the following reasons: (a) Intuitively, $S^g_{\theta^-}$ is expected to be the best possible seed set we can find that maximizes $\sigma_{\theta^-}(\cdot)$; (b) Meanwhile, we consider $S^g_{\theta^+}$ as a potential seed set for the later theoretical analysis (in the proof of Theorem 6), which requires the alignment of the same seed set for the numerator and denominator. Thus, $\alpha(\Theta) \ge \max\{\sigma_{\theta^-}(S^g_{\theta^-}), \sigma_{\theta^-}(S^g_{\theta^+})\}/\sigma_{\theta^+}(S^g_{\theta^+})$. In particular, when θ^+ and θ^- tend to the same value θ, RIM is tending towards the classical Influence Maximization, and thus the influence spread $\sigma_\theta(S^{\mathsf{LU}}_\Theta)$ can be close to the best possible result $\sigma_\theta(S^g_\theta)$. The approach adopted by LUGreedy is similar to the sandwich approximation used in [23].

The following example shows that for certain problem instances, the gap ratio $\alpha(\Theta)$ of LUGreedy could match the robust ratio $g(\Theta, S^{\mathsf{LU}}_\Theta)$, which also matches the best possible robust ratio $\max_{|S|=k} g(\Theta, S)$.

EXAMPLE 1. *Consider a graph $G = (V, E)$ where the set of nodes are equally partitioned into $2k$ subsets $V = \cup_{i=1}^{2k} V_i$ such that every V_i contains $t + 1$ nodes. Let $V_i = \{v_i^j \mid 1 \le j \le t + 1\}$ and set $E = \cup_{i=1}^{2k} E_i$ where $E_i = \{(v_i^1, v_i^j) \mid 2 \le j \le t + 1\}$. That is, every (V_i, E_i) forms a star with v_i^1 being the node at the center, all stars are disconnected from one another. For the parameter space we set the interval on every edge to be $[l, r]$. When* LUGreedy *selects k nodes, since all v_i^1's have the same (marginal) influence spread, w.l.o.g., suppose that* LUGreedy *selects $\{v_1^1, v_2^1, \ldots, v_k^1\}$. Then if we set the true probability vector $\theta \in \Theta$ s.t. $p_e = l$ for every $e \in \cup_{i=1}^{k} E_i$, and $p_e = r$ for every $e \in \cup_{i=k+1}^{2k} E_i$, it is easy to check that $\max_{|S|=k} g(\Theta, S) = g(\Theta, S^{\mathsf{LU}}_\Theta) = \alpha(\Theta) = \frac{1+tl}{1+tr}$.*

The intuition from the above example is that, when there are many alternative choices for the best seed set, and these alternative seed sets do not have much overlap in their influence coverage, the gap ratio $\alpha(\Theta)$ is a good indicator of the best possible robust ratio one can achieve.

In the next subsection, we will show that the best robust ratio could be very bad for the worst possible graph G and parameter space Θ, which motivates us to do further sampling to improve Θ.

3.2 Discussion on the robust ratio

For the theoretical perspective, we show in this part that if we make no assumption or only add loose constraints to the

input parameter space Θ, then no algorithm will guarantee good performance for some worst possible graph G.

THEOREM 3. *For RIM,*

1. *There exists a graph $G = (V, E)$ and parameter space $\Theta = \times_{e\in E}[l_e, r_e]$, such that*

$$\max_{|S|=k} g(\Theta, S) = \max_{|S|=k} \min_{\theta\in\Theta} \frac{\sigma_\theta(S)}{\sigma_\theta(S^*_\theta)} = O\left(\frac{k}{n}\right).$$

2. *There exists a graph $G = (V, E)$, constant $\delta = \Theta\left(\frac{1}{n}\right)$ and parameter space $\Theta = \times_{e\in E}[l_e, r_e]$ where $r_e - l_e \le \delta$ for every $e \in E$, such that*

$$\max_{|S|=k} g(\Theta, S) = O\left(\frac{\log n}{n}\right).$$

3. *Consider random seeds set $\tilde{S}$ of size k. There exists a graph $G = (V, E)$, constant $\delta = \Theta\left(\frac{1}{\sqrt{n}}\right)$ and parameter space $\Theta = \times_{e\in E}[l_e, r_e]$ where $r_e - l_e \le \delta$ for every $e \in E$, we have*

$$\max_{\Omega} \min_{\theta\in\Theta} \mathbb{E}_{\tilde{S}\in\Omega}\left[\frac{\sigma_\theta(\tilde{S})}{\sigma_\theta(S^*_\theta)}\right] = O\left(\frac{\log n}{\sqrt{n}}\right),$$

where Ω is any probability distribution over seed sets of size k, and $\mathbb{E}_{\tilde{S}\in\Omega}[\cdot]$ is the expectation of random set $\tilde{S}$ taken from the distribution Ω.

In the first case, we allow the input Θ to be an arbitrary parameter space. It is possible that $\Theta = \times_{e\in E}[0, 1]$ for some graph G, which means there is no knowledge at all for edge probabilities. Then any seed set may achieve $O\left(\frac{k}{n}\right)$-approximation of the optimal solution in the worst case. Intuitively, a selected seed set S may rarely activate other nodes (i.e., $O(k)$), while optimal solution (to the latent θ) may cover almost the whole graph (i.e., $\Omega(n)$).

In the second case, an additional constraint is assumed on the parameter space $\left\|\theta^+ - \theta^-\right\|_\infty \le \delta$, i.e., for every $e \in E$, $r_e - l_e \le \delta$, to see if we could obtain a better performance when δ is small. However, even though δ is in the order of $O(1/n)$, the robust ratio can be as small as $O(\log n/n)$. The proof is related to the phase transition in the Erdős-Rényi graph for the emergence of giant component. In particular, if we have a graph G consisting of two disconnected, equal-sized Erdős-Rényi random graphs with edge probabilities close to the critical value of generating a giant connected component, then whenever we select a seed in one component, that component could be just below the threshold resulting in $O(\log n)$ influence spread while the other component is just above the threshold leading to $\Theta(n)$ influence spread. Thus, the worst-case ratio for any one-node seed set is always $O(\log n/n)$. A similar discussion can be found in [18].

In the third case, we allow the algorithm to be randomized, namely the output seed set $\tilde{S}$ is a random set of size k. Even in this case, the robust ratio could be as bad as $O(\log n/\sqrt{n})$.

4. SAMPLING FOR IMPROVING RIM

From the previous section, we propose LUGreedy algorithm to check the solution-dependent bound of the robust ratio, and point out the worse-case bound could be small if Θ is not assumed to be tight enough.

Theorem 3 in the previous subsection points out that the best possible robust ratio $\max_S g(\Theta, S)$ can be too low so that the output for RIM could not provide us with a satisfying seed set in the worst case. Then a natural question is: given the input Θ, can we make efficient samples on edges so that Θ is narrowed into Θ' (this means the true $\theta \in \Theta'$ with high probability) and then output a seed set S' that makes $g(\Theta', S')$ large? This problem is called *Sampling for Improving RIM*.

In this section we study both uniform sampling and adaptive sampling for improving RIM. According to the Chernoff's bound, the more samples we make on an edge, the narrower the confidence interval we get that guarantees the true probability to be located within the confidence interval with a desired probability of confidence. After sampling to get a narrower parameter space, we could use LUGreedy algorithm to get the seed set.

4.1 Uniform Sampling

In Sampling for improving RIM, the goal is to design a sampling and maximization algorithm $\mathcal{A}$ that outputs Θ' and S' such that with high probability the robust ratio of S' in Θ' is large. After sampling edges, we can use Chernoff's bound to compute the confidence interval, and the confidence interval can be further narrowed down with more samples. However, the key issue is to connect the width of confidence interval with the stability of influence spread. We propose two ideas exploiting properties of additive and multiplicative confidence interval respectively to this issue, and incorporate into Uniform Sampling algorithm (in Algorithm 3) with theoretical justification (in Theorem 6).

Our first idea is inspired by the following lemma from [12] to build the connection in the additive form.

LEMMA 4 (LEMMA 7 IN [12]). *Given graph G and parameter space Θ such that $\forall\theta_1, \theta_2 \in \Theta$, $\|\theta_1 - \theta_2\|_\infty \le \delta$, then, $\forall S \subseteq V$,*

$$|\sigma_{\theta_1}(S) - \sigma_{\theta_2}(S)| \le mn\delta.$$

We use a tight example (in the order of $|V|$ and $|E|$) to illustrate the connection and give an insight of this lemma as follows. Consider graph $G = (V, E)$ with $|V| = n$ and $|E| = m$ ($m \gg n$). Let G be two disjoint cycles, each containing exactly $\frac{n}{2}$ nodes and $\frac{n}{2}$ edges. We arbitrarily assign the rest $m - n$ edges between two cycles. Then, for every edge e in the cycle, the interval is $l_e = r_e = 1$, and $l_e = 0$, $r_e = \delta$ for those between two cycles, which constitutes $\Theta = \times_{e\in E}[l_e, r_e]$. Suppose $\delta > 0$ is sufficiently small, and let budget $k = 1$. For any single-node set S, it is easy to check that for $\theta^- = (l_e)_{e\in E}$, $\sigma_{\theta^-}(S) = \frac{n}{2}$, and for $\theta^+ = (r_e)_{e\in E}$, $\sigma_{\theta^+}(S) \approx \frac{n}{2} + \frac{n}{2}(m-n)\delta$, thus $|\sigma_{\theta^+}(S) - \sigma_{\theta^-}(S)| \approx \frac{1}{2}n(m-n)\delta$ in this case. As a comparison, from Lemma 4, we know that $|\sigma_{\theta^+}(S) - \sigma_{\theta^-}(S)| \le mn\delta$.

Therefore, the above lemma establishes the guidance that we may sample every edge for sufficient times to shrink their confidence intervals in Θ, and feed LUGreedy with Θ as same as solving RIM, then the performance is guaranteed by Theorem 2, which matches our intuition that LUGreedy performs well with the satisfactory Θ.

On the other hand, our second idea is to use the multiplicative confidence interval to reduce the fluctuation of influence spread, then LUGreedy still applies. The next lemma is crucial to achieve this goal.

Algorithm 3 US-RIM

Input: Graph $G = (V, E)$, budget k, (ϵ, γ)
Output: Parameter space Θ_{out}, seed set S_{out}

for all $e \in E$ **do**
  Sample e for t times, and observe $x_e^1, \ldots, x_e^t$
  $p_e \leftarrow \frac{1}{t}\sum_{i=1}^{t} x_e^i$, and set δ_e according to Theorem 6
  $r_e \leftarrow \min\{1, p_e + \delta_e\}$, $l_e \leftarrow \max\{0, p_e - \delta_e\}$
end for
$\Theta_{\text{out}} \leftarrow \times_{e\in E}[l_e, r_e]$
$S_{\text{out}} \leftarrow \mathsf{LUGreedy}(G, k, \Theta_{\text{out}})$
return $(\Theta_{\text{out}}, S_{\text{out}})$

LEMMA 5. *Given graph $G = (V, E)$ and parameter space Θ. If there exists $\lambda \geq 0$, for all edge $e \in E$, s.t., $r_e \leq (1+\lambda)l_e$, then for any nonempty set $S \subseteq V$,*

$$\frac{\sigma_{\theta^+}(S)}{\sigma_{\theta^-}(S)} \leq (1+\lambda)^n, \tag{4}$$

and

$$\max_{|S|=k} \min_{\theta\in\Theta} \frac{\sigma_\theta(S)}{\sigma_\theta(S^*_\theta)} \geq (1+\lambda)^{-n}. \tag{5}$$

In this lemma, the ratio of influence spread can be bounded based on the relation of l_e and r_e in the multiplicative form.

To unify both ideas mentioned above, we propose *Uniform Sampling for RIM* algorithm (US-RIM) in Algorithm 3, and the theoretical result is presented in Theorem 6. Basically, the algorithm samples every edge with the same number of times, and use LUGreedy to obtain the seed set. We set different t and δ_e for the two ideas. Henceforth, we explicitly refer the first setting as *Uniform Sampling with Additive form* (US-RIM-A), and the second one as *Uniform Sampling with Multiplicative form* (US-RIM-M).

THEOREM 6. *Given a graph $G = (V, E)$, budget k, and accuracy parameter $\epsilon, \gamma > 0$, let $n = |V|$ and $m = |E|$, then for any unknown ground-truth parameter vector $\theta = (p_e)_{e\in E}$, Algorithm US-RIM outputs (Θ_{out}, S_{out}) such that*

$$g(\Theta_{out}, S_{out}) \geq \left(1 - \frac{1}{e}\right)(1-\epsilon),$$

with $\Pr[\theta \in \Theta_{out}] \geq 1-\gamma$, where the randomness is taken according to θ, if we follow either of the two settings:

1. *Set $t = \frac{2m^2n^2\ln(2m/\gamma)}{k^2\epsilon^2}$, and for all e, set $\delta_e = \frac{k\epsilon}{mn}$;*
2. *Assume we have p' such that $0 < p' \leq \min_{e\in E} p_e$, set $t = \frac{3\ln(2m/\gamma)}{p'}\left(\frac{2n}{\ln(1/1-\epsilon)} + 1\right)^2$, and for all edge, set $\delta_e = \frac{1}{n}p_e \log\frac{1}{\gamma}$.*

In general, the total number of samples summing up all edges is $O(\frac{m^3n^2\log(m/\gamma)}{k^2\epsilon^2})$ for US-RIM-A, and $O(\frac{mn^2\log(m/\gamma)}{p'\epsilon^2})$ for US-RIM-M with an additional constant p', the lower bound probability on all edge probabilities. The difference is that the former has a higher order of m, and the latter requires the knowledge of p' and has an extra dependency on $O(1/p')$. Since the sample complexity for both settings can be calculated in advance, one may compare the values and choose the smaller one when running the uniform sampling algorithm. An intuitive interpretation is that: (1) with high probability ($\geq 1-\gamma$), the algorithm always outputs an $(1-\frac{1}{e}-\epsilon)$-approximation solution guaranteed by US-RIM-A; (2) if $p' = \Omega(\frac{k^2}{m^2})$ (it is a loose assumption naturally satisfied in practice), we may choose US-RIM-M to achieve better sample complexity.

4.2 Non-uniform and Adaptive Sampling

In a real network, the importance of edges in an influence diffusion process varies significantly. Some edges may have larger influence probability than others or connect two important nodes in the network. Therefore, in sampling it is crucial to sample edges appropriately. Moreover, we can adapt our sampling strategy dynamically to put more sampling effort on critical edges when we learn the edge probabilities more accurately over time.

For convenience, given graph $G = (V, E)$, we define *observation set* $\mathcal{M} = \{M_e\}_{e\in E}$ as a collection of sets, where $M_e = \{x_e^1, x_e^2, \cdots, x_e^{t_e}\}$ denotes observed values of edge e via the first t_e samples on edge e. We allow that a parameter space $\Theta_0 \subseteq \times_{e\in E}[0,1]$ is given, which can be obtained by some initial samples $\mathcal{M}_0$ (e.g., uniformly sample each edge of the graph for a fixed number of times).

The following lemma is used to calculate the confidence interval, which is a combination of additive and multiplicative Chernoff's Bound. We adopt this bound in the experiment since some edges in the graph have large influence probability while others have small ones, but using either additive or multiplicative bound may not be good enough to obtain a small confidence interval. The following bound is adapted from [1] and is crucial for us in the experiment.

LEMMA 7. *For each $e \in E$, let $M_e = \{x_e^1, x_e^2, \ldots, x_e^{t_e}\}$ be samples of e in $\mathcal{M} = \{M_e\}_{e\in E}$, and t_e be the sample number. Given any $\gamma > 0$, let confidence intervals for all edges be $\Theta = \times_{e\in E}[l_e, r_e]$, such that, for any $e \in E$,*

$$l_e = \min\left\{\hat{p}_e + \frac{c_e^2}{2} - c_e\sqrt{\frac{c_e^2}{4} + \hat{p}_e},\ 0\right\}$$
$$r_e = \max\left\{\hat{p}_e + \frac{c_e^2}{2} + c_e\sqrt{\frac{c_e^2}{4} + \hat{p}_e},\ 1\right\},$$

where $\hat{p}_e = \frac{\sum_{i=1}^{t_e} x_e^i}{t_e}$, $c_e = \sqrt{\frac{3}{t_e}\ln\frac{2m}{\gamma}}$. Then, with probability at least $1-\gamma$, the true probability $\theta = (p_e)_{e\in E}$ satisfies that $\theta \in \Theta$.

Our intuition for non-uniform sampling is that the edges along the information cascade of important seeds determine the influence spread, and henceforth they should be estimated more accurately than other edges not along important information cascade paths. Thus, we use the following *Information Cascade Sampling* method to select edges. Starting from the seed set S, once node v is activated, v will try to activate its out-neighbors. In other words, for every out-edge e of v, denote t_e as the number of samples, then e will be sampled once to generate a new observation $x_e^{t_e}$ based on the latent Bernoulli distribution with success probability p_e, and t_e will be increased by 1. The process goes on until the end of the information cascade.

We propose *Information Cascade Sampling for RIM* (ICS-RIM) algorithm in Algorithm 4, which adopts information cascade sampling described above to select edges.

Algorithm 4 is an iterative procedure. In the i-th iteration, Lemma 7 is used to compute the confidence interval

Algorithm 4 ICS-RIM(τ): Information Cascade Sampling

Input: Graph $G = (V, E)$, budget k, initial sample $\mathcal{M}_0$, threshold κ, γ.
Output: Parameter space Θ_{out}, seed set S_{out}

$i \leftarrow 0$
repeat
  Get Θ_i based on $\mathcal{M}_i$ (see Lemma 7).
  $S^{\text{LU}}_{\Theta_i} = \text{LUGreedy}(G, k, \Theta_i)$
  $\mathcal{M}_{i+1} \leftarrow \mathcal{M}_i$
  for $j = 1, 2, \ldots, \tau$ **do**
    Do information cascade with the seed set $S^{\text{LU}}_{\Theta_i}$
    During the cascade, once $v \in V$ is activated, sample all out-edges of v and update $\mathcal{M}_{i+1}$
  end for
  $i \leftarrow i + 1$
until $\alpha(\Theta_i) > \kappa$
$S_{\text{out}} \leftarrow S^{\text{LU}}_{\Theta_{i-1}}$
$\Theta_{\text{out}} \leftarrow \Theta_{i-1}$
return $(\Theta_{\text{out}}, S_{\text{out}})$

Θ_i from observation set $\mathcal{M}_i$. Then according to Θ_i, we find the lower-upper greedy set $S^{\text{LU}}_{\Theta_i}$ and use information cascade to update observation set $\mathcal{M}_{i+1}$ by absorbing new samples.

Since the robust ratio $g(\Theta, S^{\text{LU}}_{\Theta_i})$ cannot be calculated efficiently, we will calculate $\alpha(\Theta)$ (defined in (3)) instead. In our algorithm, we use a pre-determined threshold κ ($\kappa \in (0, 1)$) as the stopping criteria. Therefore, for S_{out}, the robust ratio $g(\Theta, S_{\text{out}}) \geq \alpha(\Theta)\left(1 - \frac{1}{e}\right) > \kappa\left(1 - \frac{1}{e}\right)$ is guaranteed by Theorem 2, and the true probability $\theta \in \Theta_{\text{out}}$ holds with probability at least $1 - \gamma$ due to Lemma 7.

Compared with information cascade sampling method, calculating a greedy set is time-consuming. Therefore in Algorithm 4, we call LUGreedy once every τ rounds of information cascades to reduce the cost.

5. EMPIRICAL EVALUATION

We conduct experiments on two datasets, Flixster[1] and NetHEPT[2] to verify the robustness of influence maximization and our sampling methods.

5.1 Experiment Setup

5.1.1 Data Description

Flixster. The Flixster dataset is a network of American social movie discovery service (www.flixster.com). To transform the dataset into a weighted graph, each user is represented by a node, and a directed edge from node u to v is formed if v rates one movie shortly after u does so on the common movie. The dataset is analyzed in [2], and the influence probability are learned by the topic-aware model. We use the learning result of [2] in our experiment, which is a graph containing 29357 nodes and 212614 directed edges. There are 10 probabilities on each edge, and each probability represents the influence from the source user to the sink on a specific topic. Since most movies belong to at most two topics, we only consider 3 out of 10 topics in our experiment, and get two induced graphs whose number of edges are 23252 and 64934 respectively. For the first graph, probabilities of topic 8 are directly used as the ground truth parameter (termed as Flixster(Topic 8)). For the second graph, we mix the probabilities of Topic 1 and Topic 4 on each edge evenly to obtain the ground-truth probability (termed as as Flixster(Mixed)). After removing isolated nodes, the number of nodes in the two graphs are 14473 and 7118 respectively.

In [2], the probability for every edge (u, v) is learned by rating cascades that reach u and may or may not reach v, and in this cases we view that edge (u, v) are sampled. According to the data reported in [2], on average every edge is sampled 318 times for their learning process. We then use 318 samples on each edge as our initial sample $\mathcal{M}_0$.

NetHEPT. The NetHEPT dataset [11] is extensively used in many influence maximization studies. It is an academic collaboration network from the "High Energy Physics-Theory" section of arXiv form 1991 to 2003, where nodes represent the authors and each edge in the network represents one paper co-authored by two nodes. It contains 15233 nodes and 58891 undirected edges (including duplicated edges). We remove those duplicated edges and obtain a directed graph $G = (V, E), |V| = 15233, |E| = 62774$ (directed edges). Since the NetHEPT dataset does not contain the data of influence probability on edges, we set the probability on edges according to the *weighted cascade* model [19] as the ground truth parameter, i.e., $\forall e = (v, u) \in E$, let x_u be the in-degree of u in the edge-duplicated graph, y_e be the number of edges connecting node v and u, then the true probability is $p_e = 1 - (1 - \frac{1}{x_u})^{y_e}$. Following the same baseline of Flixster, we initially sample each edge for 318 times as $\mathcal{M}_0$.

5.1.2 Algorithms

We test both the uniform sampling algorithm US-RIM and the adaptive sampling algorithm ICS-RIM, as well as another adaptive algorithm OES-RIM (Out-Edge Sampling) as the baseline (to be described shortly). Each algorithm is given a graph G and initial observation set $\mathcal{M}_0$. Note that the method to estimate the parameter space based on sampling results in Algorithm 3 and Algorithm 4 are different. In order to make the comparison meaningful, in this section, for all three algorithms, a common method according to Lemma 7 is used to estimate the parameter space. In all tests, we set the size of the seed set $k = 50$. To reduce the running time, we use a faster approximation algorithm PMIA (proposed in [10]) to replace the well known greedy algorithm purposed in [19] in the whole experiment. The accuracy requirement $\gamma = o(1)$ is set to be $\gamma = m^{-0.5}$ where m is the number of edges.

US-RIM. The algorithm is slightly modified from Algorithm 3 for a better comparison of performance. The modified algorithm proceeds in an iterative fashion: In each iteration, the algorithm makes τ_1 samples on each edge, updates Θ according to Lemma 7 and computes $\alpha(\Theta)$. The algorithm stops when $\alpha(\Theta) \geq \kappa = 0.8$. τ_1 is set to 1000, 1000, 250 for NetHEPT, Flixster(Topic 8), Flixster(Mixed), respectively to achieve fine granularity and generate visually difference of $\alpha(\Theta)$ in our results.

[1] http://www.cs.sfu.ca/~sja25/personal/datasets/
[2] http://research.microsoft.com/en-us/people/weic/projects.aspx

ICS-RIM. As stated in Algorithm 4, in each iteration, the algorithm do $\tau_2 = 5000$ times information cascade sampling based on the seed set from the last iteration, and then it updates Θ according to Lemma 7, computes $\alpha(\Theta)$ and uses LUGreedy algorithm to compute the seed set for the next round. The algorithm stops when $\alpha(\Theta) \geq \kappa = 0.8$.

OES-RIM. This algorithm acts as a baseline, and it proceeds in the similar way to ICS-RIM. Instead of sampling information cascades starting from the current seed set as in ICS-RIM, OES-RIM only sample *out-edges* from the seed set. More specifically, in each iteration, the algorithm samples 5000 times of all out-edges of the seed set from last iteration, for the three graphs respectively, and then it updates Θ according to Lemma 7, computes $\alpha(\Theta)$ and uses LUGreedy algorithm to compute the seed set for the next round. Note that for OES-RIM, $\alpha(\Theta)$ remains small (with the increase of the number of samples) and cannot exceed the threshold κ even the iteration has been processed for a large number of times, therefore we will terminate it when $\alpha(\Theta)$ is stable.

5.1.3 $\bar{\alpha}$ as a Upper Bound

Theorem 2 shows that $\alpha(\Theta)\left(1-\frac{1}{e}\right)$ is a lower bound for the robust ratio $g(\Theta, S^{\mathsf{LU}}_{\Theta})$. We would also like to find some upper bound of $g(\Theta, S^{\mathsf{LU}}_{\Theta})$: If the upper bound is reasonably close to the lower bound or match in trend of changes, it indicates that $\alpha(\Theta)\left(1-\frac{1}{e}\right)$ is a reasonable indicator of the robust ratio achieved by the LUGreedy output S^{LU}_{Θ}. For any $\theta \in \Theta$, we define $\bar{\alpha}(\Theta, \theta) = \frac{\sigma_\theta(S^{\mathsf{LU}}_{\Theta})}{\sigma_\theta(S^g_\theta)}$. The following shows that $\bar{\alpha}(\Theta, \theta)$ is an upper bound for $g(\Theta, S^{\mathsf{LU}}_{\Theta})$:

$$\bar{\alpha}(\Theta,\theta) = \frac{\sigma_\theta(S^{\mathsf{LU}}_{\Theta})}{\sigma_\theta(S^g_\theta)} \geq \frac{\sigma_\theta(S^{\mathsf{LU}}_{\Theta})}{\sigma_\theta(S^*_\theta)} \geq \min_{\theta' \in \Theta} \frac{\sigma_{\theta'}(S^{\mathsf{LU}}_{\Theta})}{\sigma_{\theta'}(S^*_{\theta'})} = g(\Theta, S^{\mathsf{LU}}_{\Theta}).$$

The next question is how to find a $\theta = (\theta_e)_{e\in E} \in \Theta$ to make the upper bound $\bar{\alpha}(\Theta, \theta)$ as small as possible. In our experiments, we use the following two heuristics and take their minimum.

The first heuristic borrows the intuition from Example 1, which says that the gap ratio $\alpha(\Theta)$ is close to the robust ratio $g(\Theta, S^{\mathsf{LU}}_{\Theta})$ when (a) there are two disjoint seed sets with similar influence spead, (b) their cascade overlap is small, and (c) the reachable edges from one seed set use lower end parameters values while the reachable edges from the other seed set use upper end parameters. Thus in our heuristic, we use PMIA algorithm to find another seed set S' of k nodes when we remove all nodes in S^{LU}_{Θ}. We then do information cascades from both S^{LU}_{Θ} and S' for an equal number of times. Finally, for every edge e, if it is sampled more in the information cascade with seed set S^{LU}_{Θ} than with S', we set $\theta_e = l_e$, otherwise we set $\theta_e = r_e$. The second heuristic is a variant of the first one, where we run a number of information cascades from S^{LU}_{Θ}, and for any edge e that is sampled in at least 10% of cascades, we set $\theta_e = l_e$, otherwise we set $\theta_e = r_e$.

Other more sophisticated heuristics are possible, but it could be a separate research topic to find tighter upper bound for the robust ratio, and thus we only use the simple combination of the above two in this paper, which is already indicative. We henceforth use $\bar{\alpha}(\Theta)$ to represent the upper bound found by the minimum of the above two heuristics.

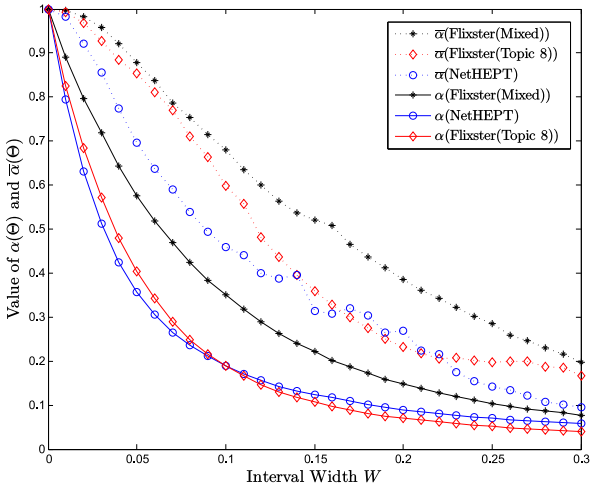

Figure 1: $\alpha(\Theta)$ and $\bar{\alpha}(\Theta)$ for different widths of confidence interval W.

5.2 Results

5.2.1 $\alpha(\Theta)$ and $\bar{\alpha}(\Theta)$ with Predetermined Intervals

In the first experiment we explore the relationship between the width of confidence interval $\Theta = \times_{e\in E}[l_e, r_e]$ and $\alpha(\Theta)$ together with $\bar{\alpha}(\Theta)$. For a given interval width W, we set $l_e = \min\{p_e - \frac{W}{2}, 0\}, r_e = \max\{p_e + \frac{W}{2}, 1\}\ \forall e \in E$, where p_e is the ground-truth probability of e. Then we calculate $\alpha(\Theta)$ and $\bar{\alpha}(\Theta)$. We vary the width W to see the trend of changes of $\alpha(\Theta)$ and $\bar{\alpha}(\Theta)$. Figure 1 reports the result on the three graphs with seed set size $k = 50$.

First, we observe that as the parameter space Θ becomes wider, the value of both $\alpha(\Theta)$ and $\bar{\alpha}(\Theta)$ become smaller, which matches our intuition that larger uncertainty results in worse robustness. Second, there is a sharp decrease of $\alpha(\Theta)$ between $W \in [0, 0.1]$ and a much slower decrease afterwards for all three graphs. The decrease of $\bar{\alpha}(\Theta)$ is not as sharp as that of $\alpha(\Theta)$ but the decrease also slows down with larger W after 0.2. The overall trend of $\alpha(\Theta)$ and $\bar{\alpha}(\Theta)$ suggests that the robust ratio may be sensitive to the uncertainty of the parameter space, and only when the uncertainty of the parameter space reduces to a certain level that we can obtain reasonable guarantee on the robustness of our solution.

As a comparison, we know that the average number of samples on each edge is 318 for the learned probabilities in the Flixster dataset. This corresponds to an average interval width of 0.293 for topic 8 and 0.265 for the mixed topic. At these interval widths, $\alpha(\Theta)$ values are approximately 0.04 and 0.08 respectively for the two graphs, and $\bar{\alpha}(\Theta)$ are approximately 0.12 and 0.2 respectively. This means that, even considering the upper bound $\bar{\alpha}(\Theta)$, the robust ratio is pretty low, and thus the learned probabilities reported in [2] may result in quite poor performance for robust influence maximization.

Of course, our result of $\alpha(\Theta)$ and $\bar{\alpha}(\Theta)$ is only targeted at the robustness of our LUGreedy algorithm, and there could exist better algorithm having higher robustness performance at the same uncertainty level. Finding a better RIM algorithm seems to be a difficult task, and we hope that our study could motivate more research in searching for

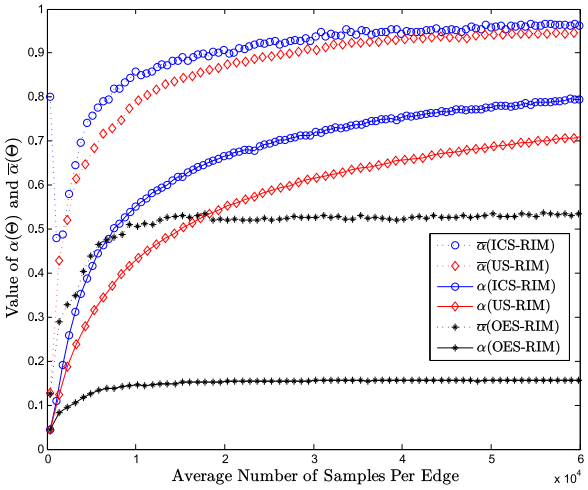

Figure 2: $\alpha(\Theta)$ and $\bar{\alpha}(\Theta)$ for different average number of samples per edge on graph NetHEPT.

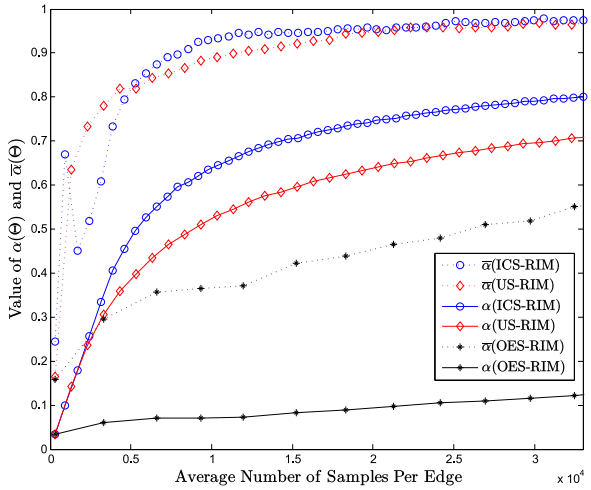

Figure 3: $\alpha(\Theta)$ and $\bar{\alpha}(\Theta)$ for different average number of samples per edge on graph Flixster(Topic 8).

such better RIM algorithms. Besides S^{LU}_{Θ}, we also independently test the classical greedy seed set S^g_θ for $\theta = (p_e)_{e\in E}$ on the lower parameter vector θ^- (that is $\frac{\sigma_{\theta^-}(S^g_\theta)}{\sigma_{\theta^+}(S^g_{\theta^+})}$ versus $\alpha(\Theta)$), and the average performance on each data point is 2.45%, 1.05%, 6.11% worse than S^{LU}_{Θ} for Flixster(Mixed), Flixster(Topic 8) and NetHEPT, respectively. Therefore, it shows that S^{LU}_{Θ} outperforms σ^g_θ in the worse-case scenario, and henceforth we only use S^{LU}_{Θ} in the following experiments.

5.2.2 *Results for Sampling algorithms*

Figures 2, 3 and 4 reports the result of $\alpha = \alpha(\Theta)$ and $\bar{\alpha} = \bar{\alpha}(\Theta)$ for the three tested graphs respectively, when the average number of samples per edge increases. For better presentation, we trim all figures as long as $\alpha(\mathsf{US\text{-}RIM}) = 0.7$. (For example, in Flixster(Topic 8), US-RIM requires 77318 samples in average for α to reach 0.8, while ICS-RIM only needs 33033, and for OES-RIM α sticks to 0.118.)

For the sampling algorithms, after the i-th iteration, the observation set is updated from $\mathcal{M}_{i-1}$ to $\mathcal{M}_i$, and the average number of samples per edge in the network is calculated. Markers on each curve in these figures represent the result after one iteration of the corresponding sampling algorithm.

The results on all three graphs are consistent. First, for each pair of $\alpha(\Theta)$ and $\bar{\alpha}(\Theta)$, even though there is still some gap, indicating either the lower bound or the upper bound may not be tight yet, the trends on both $\alpha(\Theta)$ and $\bar{\alpha}(\Theta)$ are consistent: Both increase with the number of samples, even with similar slopes at each point; and among different algorithms, the ranking order and relative change are consistent with both $\alpha(\Theta)$ and $\bar{\alpha}(\Theta)$. All these consistency suggests that gap ratio $\alpha(\Theta)$ could be used as an indicator for the robustness of Algorithm LUGreedy, and it is reasonable to use $\alpha(\Theta)$ in comparing the performance of different algorithms.

Second, comparing the performance of three algorithms, we see that both US-RIM and ICS-RIM are helpful in improving the robust ratio of the selected seed set, and ICS-RIM is better than US-RIM, especially when the sample size increases. The baseline algorithm OES-RIM, however, performs significantly poorer than the other two, even though it is also an adaptive algorithm as ICS-RIM. The reason is

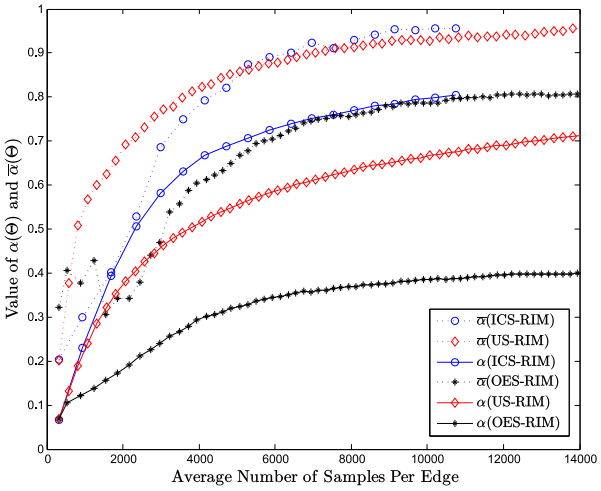

Figure 4: $\alpha(\Theta)$ and $\bar{\alpha}(\Theta)$ for different average number of samples per edge on graph Flixster(Mixed).

that, the lower-upper greedy set S^{LU}_{Θ} changes little after a certain number of iterations in OES-RIM, and thus only a small number of edges (out edges of S^{LU}_{Θ}) are repeatedly sampled. The probabilities on these edges are already estimated very accurately while other edge probabilities are far from accurate. It is the inaccurate edges that make $\alpha(\Theta)$ and the best robust ratio small. In contrast, ICS-RIM uses information cascades to sample not only edges directly connecting to the seed set but also edges that can be potentially reached. This suggests that it is important for a sampling method to balance the sampling between critical edges and other potentially useful edges in order to achieve better robustness in influence maximization.

Overall, the results suggest that information cascade based sampling method stands out as a competitive choice when we can adaptively sample more edges to achieve better robustness. If adaptive sampling is not possible, predetermined uniform sampling may also perform reasonably well.

6. CONCLUSION

In this paper, we propose the study of robust influence maximization to address the impact of uncertainty in edge probability estimates that would inevitably occur in practice to the influence maximization task. We propose the LUGreedy algorithm with a proven solution-dependent bound, and further propose sampling methods, in particular information cascade based adaptive sample method to effectively reduce the uncertainty and increase the robustness of the LUGreedy algorithm. The experimental results validate the usefulness of the LUGreedy algorithm and the information cascade based sampling method ICS-RIM. Moreover, the results indicate that robustness may be sensitive to the uncertainty of parameter space, and learning algorithms may need more data to achieve accurate learning results for robust influence maximization.

Our work opens up a number of research directions. First, it is unclear what could be the upper bound of the best robust ratio given an actual network and learned parameter space. Answering this question would help us to understand whether robust influence maximization is intrinsically difficult for a particular network or it is just our algorithm that does not perform well. If it is the latter case, then an important direction is to design better robust influence maximization algorithms. Another direction is how to improve sampling methods and learning methods to achieve more accurate parameter learning, which seems to be crucial for robust influence maximization. In summary, our work indicates a big data challenge on social influence research — the data on social influence analysis is still not big enough, such that the uncertainty level in model learning may result in poor performance for influence maximization. We hope that our work could encourage further researches to meet this challenge from multiple aspects including data collection, data analysis, and algorithm design.

Acknowledgment

The research of Wei Chen is partially supported by the National Natural Science Foundation of China (Grant No. 61433014).

7. REFERENCES

[1] A. Badanidiyuru, R. Kleinberg, and A. Slivkins. Bandits with knapsacks. In *FOCS 2013*.

[2] N. Barbieri, F. Bonchi, and G. Manco. Topic-aware social influence propagation models. *Knowledge and information systems*, 37(3):555–584, 2013.

[3] A. Ben-Tal and A. Nemirovski. Robust optimization–methodology and applications. *Mathematical Programming*, 92(3):453–480, 2002.

[4] C. Borgs, M. Brautbar, J. T. Chayes, and B. Lucier. Maximizing social influence in nearly optimal time. In *SODA 2014*.

[5] S. Bubeck, R. Munos, and G. Stoltz. Pure exploration in finitely-armed and continuous-armed bandits. *Theoretical Computer Science*, 412:1832–1852, 2011.

[6] C. Budak, D. Agrawal, and A. El Abbadi. Limiting the spread of misinformation in social networks. In *WWW 2011*.

[7] S. Chen, T. Lin, I. King, M. R. Lyu, and W. Chen. Combinatorial pure exploration of multi-armed bandits. In *NIPS 2014*.

[8] W. Chen, L. V. Lakshmanan, and C. Castillo. Information and influence propagation in social networks. *Synthesis Lectures on Data Management*, 5(4):1–177, 2013.

[9] W. Chen, T. Lin, Z. Tan, M. Zhao, and X. Zhou. Robust influence maximization. *CoRR*, abs/1601.06551, 2016.

[10] W. Chen, C. Wang, and Y. Wang. Scalable influence maximization for prevalent viral marketing in large-scale social networks. In *KDD 2010*.

[11] W. Chen, Y. Wang, and S. Yang. Efficient influence maximization in social networks. In *KDD 2009*.

[12] W. Chen, Y. Wang, and Y. Yuan. Combinatorial multi-armed bandit and its extension to probabilistically triggered arms. *CoRR*, abs/1407.8339, 2014.

[13] P. Domingos and M. Richardson. Mining the network value of customers. In *KDD 2001*.

[14] U. Feige. A threshold of ln n for approximating set cover. *Journal of the ACM (JACM)*, 45(4):634–652, 1998.

[15] A. Goyal, F. Bonchi, and L. V. Lakshmanan. Learning influence probabilities in social networks. In *WSDM 2010*.

[16] A. Goyal, W. Lu, and L. V. Lakshmanan. Celf++: optimizing the greedy algorithm for influence maximization in social networks. In *WWW 2011*.

[17] X. He and D. Kempe. Robust influence maximization. In *KDD 2016*.

[18] X. He and D. Kempe. Stability of Influence Maximization. *ArXiv e-prints*, Jan. 2015.

[19] D. Kempe, J. Kleinberg, and É. Tardos. Maximizing the spread of influence through a social network. In *KDD 2003*.

[20] A. Krause, H. B. McMahon, C. Guestrin, and A. Gupta. Robust submodular observation selection. *JMLR*, 9:2761–2801, 2008.

[21] S. Lei, S. Maniu, L. Mo, R. Cheng, and P. Senellart. Online influence maximization. In *KDD 2015*.

[22] J. Leskovec, A. Krause, C. Guestrin, C. Faloutsos, J. VanBriesen, and N. Glance. Cost-effective outbreak detection in networks. In *KDD 2007*.

[23] W. Lu, W. Chen, and L. V. Lakshmanan. From competition to complementarity: comparative influence diffusion and maximization. In *VLDB 2015*.

[24] P. Netrapalli and S. Sanghavi. Learning the graph of epidemic cascades. In *SIGMETRICS 2012*.

[25] M. G. Rodriguez, D. Balduzzi, and B. Schölkopf. Uncovering the temporal dynamics of diffusion networks. In *ICML 2011*.

[26] K. Saito, R. Nakano, and M. Kimura. Prediction of information diffusion probabilities for independent cascade model. In *Knowledge-Based Intelligent Information and Engineering Systems*, pages 67–75. Springer, 2008.

[27] J. Tang, J. Sun, C. Wang, and Z. Yang. Social influence analysis in large-scale networks. In *KDD 2009*.

[28] Y. Tang, X. Xiao, and Y. Shi. Influence maximization: near-optimal time complexity meets practical efficiency. In *SIGMOD 2014*.