“Got to have faith!”: The DEvOTION algorithm for delurking in social networks

Roberto Interdonato
DIMES, University of Calabria
87036 Arcavacata di Rende (CS) - Italy
Email: rinterdonato@dimes.unical.it

Chiara Pulice
DIMES, University of Calabria
87036 Arcavacata di Rende (CS) - Italy
Email: cpulice@dimes.unical.it

Andrea Tagarelli
DIMES, University of Calabria
87036 Arcavacata di Rende (CS) - Italy
Email: tagarelli@dimes.unical.it

***Abstract*—Lurkers are silent members of a social network (SN) who gain benefit from others' information without significantly giving back to the community. The study of lurking behaviors in SNs is nonetheless important, since these users acquire knowledge from the community, and as such they are social capital holders. Within this view, a major goal is to *delurk* such users, i.e., to encourage them to more actively be involved in the SN. Despite delurking strategies have been conceptualized in social science and human-computer interaction research, no computational approach has been so far defined to turn lurkers into active participants in the SN. In this work we fill this gap by presenting a delurking-oriented targeted influence maximization problem under the linear threshold (LT) model. We define a novel objective function, in terms of the lurking scores associated with the nodes in the final active set, and we show it is monotone and submodular. We provide an approximate solution by developing a greedy algorithm, named DEvOTION, which computes a k-node set that maximizes the value of the delurking-capital-based objective function, for a given minimum lurking score threshold. Results on SN datasets of different sizes have demonstrated the significance of our delurking approach via LT-based targeted influence maximization.**

***Keywords*—silent users, lurking, targeted influence maximization, LurkerRank, participation and engagement in social networks**

I. Introduction

All large-scale on-line social networks are characterized by a “participation inequality” principle, i.e., a disequilibrium between the niche of super contributors and the crowd of silent users, which just observe ongoing discussions, read posts, watch videos and so on. In other words, the real audience of a SN does not actively contribute; rather, it *lurks*. A *lurker* is hence a silent member of a SN who gains benefit from others' information and services without significantly giving back to the community [3]. In this respect, a major goal is to *delurk* such users, i.e., to develop a mix of strategies aimed at encouraging lurkers to return their acquired social capital, through a more active participation to the community life. This has the important long-term effect of helping sustain the SN over time with fresh ideas and perspectives.

Social science and human-computer interaction research communities have widely investigated the main causes that explain lurking behaviors. Lurking is often due to a subjective reticence (rather than malicious motivations) to contribute to the community wisdom; a lurker often simply feels that gathering information by browsing is enough without the need of being further involved in the community. Lurking can be expected or even encouraged because it allows users (especially newcomers) to learn or improve their understanding of the etiquette of an online community [3]. Sun et al. [12] have identified four types of factors related to lurking, namely: environmental influence determined by the online community, personal preference related to an individual's personality, relationships between the individual and the community, and privacy/security considerations. By contrast, few suggestions have been given about *how to turn lurkers into participants/contributors*. *Delurking actions* can be broadly categorized into four types [12]: reward-based external stimuli (e.g., tangible or intangible rewards), encouragement information (e.g., welcome statements, introduction to the netiquette rules), improvement of the usability and learnability of the system, and guidance from elders/master users to help lurkers become familiar with the system as quickly as possible. As studied in [1], the trustworthiness or credibility that lurkers perceive with regard of some members of the community represents a key psychological factor to persuade lurkers to change their silent status. In addition, a further incentive for delurking would be represented by a habit of the community of nurturing newcomers and novices.

The above considerations prompted us to investigate an effective computational approach to define a delurking strategy, in which lurkers are persuaded to be more actively engaged in the SN community by other users. Our key idea in this work is *to conceptualize a delurking approach under a graph-based information diffusion model*. Research on information diffusion in SNs (cf. [5] for a recent survey) is well-established due to a plethora of methods that have been developed in the last years, mostly upon two seminal models, namely Independent Cascade (IC) and Linear Threshold (LT) [8]. These assume that an information diffusion process would unfold in a static, directed graph, where each node can be “activated” or not (under a progressive assumption) according to some rules. Intuitively, the activation of a node means that a user is influenced by other users so to “become aware of” or “adopt” a piece of information.

In this work, we address a special case of the *influence maximization* (IM) problem, namely *targeted IM*. In an IM framework the general objective is to find a set of initially activated users (also called seed users) which can maximize the spread of information through the network ([8], [4], [10]). In our instance of targeted IM, lurkers are regarded as the target of the diffusion process and the goal is to find a set of nodes capable of maximizing the likelihood of “activating” them. It should be noted that there is a relatively small body of works on IM that involves some notion of *target* of the diffusion process and they all stem from perspectives different from ours. More in detail, our proposed approach aims at maximizing the probability of activating a target set which can be arbitrarily large, by discovering a seed set which is neither fixed and singleton (e.g., [7], [6], [17]) nor it has constraints related to the topological closeness to a fixed initiator (e.g., [17]). Other approaches (e.g., [9], [10], [18], [16]) can be considered as related to targeted IM since they introduce in the diffusion process dimensions concerning the users' profiles; however, our proposal does not make any assumption on the network structure, nor it depends on user characterization based on topic-biased or categorical distributions. While some analogy could in principle be found between the notion of conformity in [10] and the proposed one of lurker activation, our method does not rely on sentiment analysis like [10] does.

It is worth emphasizing that all the aforementioned works focus on the IC diffusion model, while we propose a targeted IM problem under the LT model. We believe that a natural motivation to use LT (rather than IC or IC-based models) stands in the ability of this model to reflect the cumulative effect of exposure to multiple sources of influence. This can be profitably exploited to maximize the likelihood of changing the status of a user into a more active role in the SN. We recall that a node v can be influenced by each neighbor u according to a weight $b(u,v)$ such that $\sum_{u \in N^{in}(v)} b(u,v) \leq 1$, where $N^{in}(v)$ is the in-neighbor set of node v. At the very beginning of the diffusion process, each node v is assigned a threshold uniformly at random from $[0,1]$. This represents the weighted fraction of v's neighbors that must become active in order for v to become active itself. Intuitively, the higher is this threshold, the harder will be the task of enrolling v in a new trend, since the total influence weight must exceed its threshold.

Our key intuition is that the outcome of a diffusion process can actually lead to the delurking of silent users, provided that the information to be diffused towards the target lurkers represents a well-established delurking action. The activation of nodes regarded as lurkers can be seen as delurking those users. To this end, existing *lurker ranking* algorithms [15], [13] can profitably be exploited to mine lurking behaviors in the network, and hence to associate users with a value (*lurking score*) indicating her/his lurking status. Therefore, the lurking score of a user would represent the gain deriving from delurking that user. Our main contributions are summarized as follows:

— We propose the first computational framework that addresses the problem of delurking in social networks.

— We define the *delurking-oriented targeted influence maximization* problem. A key novelty in the formulated optimization problem is the objective function, which is defined in terms of the cumulative amount of the lurking scores associated with the nodes in the final active set, or *delurking capital*.

— Our delurking-oriented targeted IM problem shares the computational intractability with classic IM problems. However, since the proposed objective function is shown to be monotone and submodular under the LT model, we provide a greedy algorithm (with typical $1 - 1/e - \epsilon$ approximation ratio), dubbed DEvOTION, which computes a k-node set that maximizes the delurking capital in the network, for a given minimum lurking threshold. We also point out that, to the best of our knowledge, our approach is the first to address a targeted IM problem under the LT diffusion model.

— We evaluate DEvOTION over SN datasets of different characteristics and sizes, assessing its performance in terms of estimation of the delurking capital, distribution of target nodes in the final active set, and execution time.

II. Delurking-oriented Targeted Influence Maximization

A. Problem statement

Let $\mathcal{G}_0 = \langle \mathcal{V}, \mathcal{E} \rangle$ be the directed graph representing a SN, with set of nodes $\mathcal{V}$ and set of edges $\mathcal{E}$. According to the lurking graph models given in [15], [13], any edge (u,v) means that v is is "consuming" or "receiving" information from u. Let $\mathcal{G} = \mathcal{G}_0(b,\ell) = \langle \mathcal{V}, \mathcal{E}, b, \ell \rangle$ denote a directed weighted graph representing the information diffusion graph associated with the social graph $\mathcal{G}_0$, where $b : \mathcal{E} \rightarrow \mathbb{R}^*$ is an edge weighting function, and $\ell : \mathcal{V} \rightarrow \mathbb{R}^*$ is a node weighting function. Such weighting functions rely on a pre-existing solution of a LurkerRank instance applied to the social network graph $\mathcal{G}_0 = \langle \mathcal{V}, \mathcal{E} \rangle$ [15], [13]. We shall provide formal details about these functions in Section II-C, while here we give intuition into them. For any edge (u,v), the weight $b(u,v)$ indicates how much node u has contributed to the v's lurking score calculated by LurkerRank, which resembles a measure of "influence" produced by u to v. Also, the node weight $\ell(v)$ indicates the status of v as lurker, such as the higher the lurker ranking score of v the higher should be $\ell(v)$.

We denote with $LS \in [0,1]$ a threshold value that indicates the *minimum lurking score* that a node in the network must have in order to be regarded as a target node. Moreover, given any seed set $S \subseteq \mathcal{V}$, we denote with $\mu(S)$ the *final active set*, i.e., the set of nodes that are active at the end of the diffusion process starting from S.

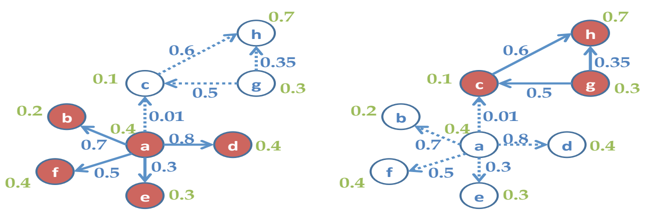
Upon the above defined quantities, we introduce a measure that will be essential to the definition of our targeted IM problem. This measure, we call *delurking capital* cumulated via S, is proportional to the amount of the lurking scores over all target nodes that are activated by the seed set S.

Definition 1: Given a set $S \subseteq \mathcal{V}$, the ***delurking capital*** $DC(\mu(S))$ associated with the final active set $\mu(S) \subseteq \mathcal{V}$ is defined as:

$$DC(\mu(S)) = \sum_{v \in \mu(S) \setminus S \ \wedge \ \ell(v) \geq LS} \ell(v) \quad (1)$$

Note that in Eq. 1 we do not consider nodes that belong to the seed set S. The ratio behind this choice is that the selection of the seed set should not be biased by nodes with highest lurking scores; this will be made clear next in the definition of our objective function, which in fact relies on $DC(\mu(S))$.

We now formally define our proposed problem of delurking-oriented targeted influence maximization. Afterwards, we discuss the diffusion model and properties of the proposed objective function, and we end this section by presenting our developed algorithm.



(a) $\mu(S)$ with $S = \{a\}$ (b) $\mu(S)$ with $S = \{g\}$

Fig. 1. Final active sets (filled circles) for different seeds. Edge weights (values in blue) and node weights (values in green) are computed by functions b and ℓ. To avoid cluttering of the figure, the node activation thresholds used by LT model here coincide with the node weights. (Best viewed in color).

Definition 2: ***Delurking-oriented Targeted Influence Maximization Problem***. Given $\mathcal{G} = \langle \mathcal{V}, \mathcal{E}, b, \ell \rangle$, a diffusion model on $\mathcal{G}$, a budget k, and a lurking threshold LS, find a seed set $S \subseteq \mathcal{V}$ with $|S| \leq k$ of nodes (users) such that, by activating them, the final active set $\mu(S) \subseteq \mathcal{V}$ will have the maximum delurking capital:

$$S = \underset{S' \subseteq \mathcal{V} \ s.t. \ |S'| \leq k}{\text{argmax}} DC(\mu(S')) \qquad (2)$$

B. Properties of the proposed objective function

The objective function of the problem in Eq. 2 differs from the ones in classic IM as it is defined in terms of the cumulative amount of the scores associated with the activated (target) nodes, i.e., $DC(\mu(S))$, instead of the size of the final active set ($|\mu(S)|$, commonly known as spread).

Example in Fig. 1 helps us motivate the different outcome obtained via LT for the classic IM and the proposed targeted IM based on LS. We assume for the sake of simplicity of the example that the node weights correspond to both the node activation thresholds (used in the LT model) and the lurking scores (ℓ). As shown on the left of the figure, the seed set $S = \{a\}$ is the best one to accomplish the influence maximization task as it causes four nodes to be activated during the process. However, the optimal seed set can be different in our setting of targeted IM: for example, if we set $LS = 0.6$, then the best seed set is $S = \{g\}$. Indeed, node h is the only one with lurking score ($\ell(h)$) greater than 0.6. This node is eventually activated (at the second step of the diffusion process) by the seed g due to the influence exerted jointly with c (which is in turn activated by g at the first step of the process).

The problem in Def. 2 preserves the complexity of the IM problem and, as a result, it is computationally intractable, i.e., it is NP-hard. However, as for the classic IM problem, a greedy solution can be designed provided that the natural diminishing property holds for the considered problem. It is well-known that for many diffusion models, including LT, the function $\sigma(A)$ mapping any subset $A \subseteq \mathcal{V}$ of nodes to the size of the final active set $\mu(A)$ satisfies monotonicity and submodularity by exploiting the *equivalent live-edge graph model* [8]. Upon this, we next provide a theoretical result that proves that the function $DC(\mu(A))$ mapping each active set $\mu(A) \subseteq \mathcal{V}$ to its overall delurking capital, is monotone and submodular, for any $LS \in [0, 1]$.

Proposition 1: The delurking capital function defined in Eq. 1 is monotone and submodular under the LT model.

Proof sketch. By exploiting the equivalence between LT and the live-edge model shown in [8], for any set $A \subseteq \mathcal{V}$ we can express the expected delurking capital of the final active set $\mu(A)$ in terms of reachability under the live-edge graph:

$$DC(\mu(A)) = \sum_{\forall X} \Pr(X) DC(R^X(A)) \qquad (3)$$

where $\Pr(X)$ is the probability that a hypothetical live-edge graph X is selected from all possible live-edge graphs, and $R^X(A)$ is the set of nodes that are reachable in X from A. Since for all $v \in \mathcal{V}$, $\ell(v)$ is a non-negative value, $DC(R^X(A))$ is clearly monotone and submodular. Thus, the expected delurking capital under LT is a non-negative linear combination of monotone submodular functions, and hence it is monotone and submodular. □

As a consequence of the above result, a greedy approach can be applied in order to provide an approximate solution for the problem in Def. 2. We shall present it in Section II-D.

C. Modeling the information diffusion graph

As anticipated in Section II-A, the node weight $\ell(v)$ should reflect the status of v as lurker, such as the higher the lurker ranking score of v the higher should be $\ell(v)$. We define the node weighting function $\ell(\cdot)$ upon scaling and normalizing the stationary distribution produced by the LurkerRank algorithm over $\mathcal{G}_0$ [15], [13]. The scaling compensates for the fact that the lurking scores produced by LurkerRank, although distributed over a significantly wide range (as reported in [13]), might be numerically very low (e.g., order of $1e$-3 or below). Moreover, we introduce a small smoothing constant in order to avoid that the highest lurking scores are mapped exactly to 1. Formally, for each node $v \in \mathcal{V}$, we define the *node lurking value* $\ell(v) \in [0, 1)$ as follows:

$$\ell(v) = \frac{\widetilde{\pi_v} - min_\pi}{(max_\pi - min_\pi) + \epsilon_\pi} \qquad (4)$$

where $\widetilde{\pi}$ denotes the stationary distribution of the lurker ranking scores (π) divided by the base-10 power of the order of magnitude of the minimum value in π, $\widetilde{\pi_v}$ is the value of $\widetilde{\pi}$ corresponding to node v, $max_\pi = \max_{u \in \mathcal{V}} \widetilde{\pi_u}$, $min_\pi = \min_{u \in \mathcal{V}} \widetilde{\pi_u}$, and ϵ_π is a smoothing constant proportional to the order of magnitude of the max_π value.

In order to define the edge weights so that they express a notion of strength of influence from a node to another (as normally required in an information diffusion model), we again exploit information derived from the ranking solution obtained by LurkerRank as well as from the structural properties of the social graph. Our key idea is to calculate the weight on edge $(u, v) \in \mathcal{E}$ proportionally to the fraction of the original lurking score of v given by its in-neighbor u [15], [13]:

$$b_0(u, v) = \left[\sum_{w \in N^{in}(v)} \frac{out(w)}{in(w)} \pi_w \right]^{-1} \frac{out(u)}{in(u)} \pi_u \qquad (5)$$

The above edge weight definition has however two limitations: (i) it is independent of the v's lurking score, therefore the contribution of node u will be the same value for each of its out-neighbors with identical set of in-neighbors; (ii)

Algorithm 1 DElurking Oriented Targeted Influence maximizatiON - DEvOTION

Input: A graph $\mathcal{G} = \langle \mathcal{V}, \mathcal{E}, b, \ell \rangle$, a budget (seed set size) k, a lurking threshold $LS \in [0, 1]$, a path pruning threshold $\eta \in [0, 1]$.
Output: Seed set S.

1: $S \leftarrow \emptyset$
2: $T \leftarrow \mathcal{V}$ *{nodes that can reach target nodes}*
3: $TargetSet \leftarrow \emptyset$ *{stores the target nodes at current iteration}*
4: **for** $u \in \mathcal{V}$ **do**
5: **if** $\ell(u) \geq LS$ **then**
6: $TargetSet \leftarrow TargetSet \cup \{u\}$
7: **end if**
8: **end for**
9: **while** $|S| < k$ **do**
10: $bestSeed, bestSeed.DC \leftarrow -1$ *{keeps track of the node with the highest spread}*
11: **for** $u \in T \setminus S$ **do**
12: $u.DC \leftarrow 0$ *{initializes each node's spread to zero}*
13: **end for**
14: $T \leftarrow \emptyset$;
15: **for** $u \in TargetSet \setminus S$ **do**
16: backward$(\langle u \rangle, 1, \ell(u))$
17: **end for**
18: **if** $bestSeed \neq -1$ **then**
19: $S \leftarrow S \cup \{bestSeed\}$
20: **else**
21: **break**
22: **end if**
23: **end while**
24: **return** S

25: **procedure** backward$(\mathcal{P}, pp, score)$
26: $u \leftarrow \mathcal{P}.last()$
27: $T \leftarrow T \cup \{u\}$
28: **while** $v \in N^{in}(u) \ \wedge \ v \notin S \cup \mathcal{P}.nodeSet()$ **do**
29: $pp \leftarrow pp \times b(v, u)$ *{updates the path probability}*
30: **if** $pp \geq \eta$ **then**
31: $v.DC \leftarrow v.DC + pp \times score$
32: **if** $v.DC > bestSeed.DC$ **then**
33: $bestSeed \leftarrow v$ *{sets the best seed node as v}*
34: **end if**
35: backward$(\mathcal{P}.append(v), pp, score)$
36: **end if**
37: **end while**

the constraint on the sum of incoming edge weights will be strictly equal to 1, which is not necessarily required. Therefore, we define the actual edge weighting function $b(\cdot)$ upon a modification of Eq. 5 according to the above remarks:

$$b(u, v) = b_0(u, v) \times e^{\ell(v)-1} \tag{6}$$

Equation 6 fulfills the requirement $\sum_{u \in N^{in}(v)} b(u, v) \leq 1$, moreover it takes into account $\ell(v)$ such that the resulting weight on (u, v) will be decreased (with exponential smoothing) for higher $\ell(v)$. This can be explained since the more a node acts as a lurker, the more active in-neighbors are needed to activate that node.

D. The DEvOTION algorithm

Algorithm 1 shows our proposed greedy method, named DEvOTION (stands for **DE**lurking **O**riented **T**argeted **I**nfluence maximizati**ON**). Following the lead of the study in [4], DEvOTION exploits as well the search for shortest paths in the diffusion graph, however in a *backward* fashion. Along with the information diffusion graph $\mathcal{G}$, the budget integer k, and the minimum lurking score LS, DEvOTION takes in input a real-valued threshold η. This parameter is used to control the size of the neighborhood within which paths are enumerated: in fact, the majority of influence can be captured by exploring the paths within a relatively small neighborhood; note that for higher η values, less paths are explored (i.e., paths are pruned earlier) leading to smaller runtime but with decreased accuracy in spread estimation.

The key idea behind DEvOTION is to perform a backward visit of the graph starting from the nodes identified as target (i.e., the nodes u with $\ell(u) \geq LS$). To this end, all nodes are initially examined to compute $TargetSet$ (Lines 4-8). In order to yield a seed set S of size at most k, DEvOTION works as follows. During each iteration of the main loop (Lines 9-19), DEvOTION computes the set T of nodes that reach the target ones and keeps track, into the variable $bestSeed$, of the node with the highest marginal gain (i.e., delurking capital DC). This is found at the end of each iteration upon calling the subroutine backward over all nodes in $TargetSet$ that do not belong to the current seed set S (Lines 15-17). This subroutine takes a path $\mathcal{P}$, its probability pp, and the lurking score of the end node in the path (i.e., a target node), and extends $\mathcal{P}$ as much as possible (i.e., as long as pp is not lower than η). Initially (Line 16), a path is formed by one target node, with probability 1. Then (Lines 28-37), the path is extended by exploring the graph backward, adding to it one, unexplored in-neighbor v at time, in a depth-first fashion. The path probability is updated (Line 29) according to the LT-equivalent "live-edge" model [4], [8], and so the delurking capital (Line 31). The process is continued until no more nodes can be added to the path.

Consider the example in Fig. 1, where the target set is $\{h\}$. By applying DEvOTION, assuming for simplicity to set $k = 1$ and $\eta = 0$, the target node can be reached via a (with $a.DC = [0.01 \times 0.6] \times 0.7$), c (with $c.DC = 0.6 \times 0.7$), and g (with $g.DC = [0.35 + 0.5 \times 0.6] \times 0.7$). Node g has the highest DC, since it has the largest chance of success ($0.35 + 0.5 \times 0.6$), and hence it is chosen as seed node.

Note that moving backward from the target nodes has two positive side effects: indeed, as all the paths starting from nodes that cannot reach any target are ignored, we get a further path pruning that reduces DEvOTION running time without affecting its accuracy. It is also worth noting that (i) if LS is set to zero and the node lurking values are uniformly distributed, our problem reduces to classic IM problem, and (ii) if the target set includes a single node, i.e. $|TargetSet| = 1$, our problem is equivalent to a single-target IM problem.

Estimating delurking capital. The overall delurking capital and influence weight received by the target nodes, given a seed set, is estimated by using a variant of the *SimPath-Spread* procedure [4], which has been adapted to account for our objective function. More in detail, the DC spread is updated according to the lurking scores of the target nodes reached by the seed set S and the probability of the considered paths. Moreover, for each node u belonging to the target set, its total influence weight is computed by summing up the weight of each scanned path ending in it.

III. Experimental Evaluation

A. Data and evaluation methodology

We used the publicly available *Google+*, *FriendFeed* and *Instagram* datasets to conduct our analysis. *Google+* (about

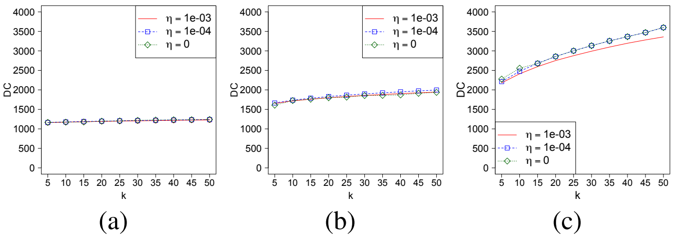


Fig. 2. Delurking capital in function of k and η, with LS-*perc* set to (a) 5%, (b) 10%, and (c) 25%, on *Instagram*.

100k nodes, 13.5M links) dataset was studied in [11], *FriendFeed* (about 500K nodes, 19.5M links) refers to the latest (2010) version of the dataset studied in [2], while *Instagram* (about 55K nodes, 1M links) is our dump studied in [14].[1]

We will use symbol LS-*perc* to denote a percentage value that determines the setting of LS such that the selected target set corresponds to the top-LS-*perc* of the distribution of node lurking scores (ℓ values). We will show results that correspond to LS-*perc* $\in \{5\%, 10\%, 25\%\}$. To implement the node and edge weighting functions (i.e., ℓ and b), we used the stationary distribution produced by the in-neighbors-driven PageRank-based LurkerRank algorithm on the social network graph.

We studied the impact of parameters η, k and LS-*perc* on the proposed model. Moreover, we compared DEvOTION with three baseline algorithms, dubbed Random, LargestDegree and bottom-LR baselines: Random calculates the delurking capital obtained for 100 randomly selected seed sets for each value of k, then the final delurking capital is averaged over the number of random extractions; LargestDegree selects, for each value of k, the k nodes in the graph with the largest out-degree; bottom-LR algorithm selects, for each value of k, the k nodes with the lowest lurking value, i.e., the k most active users in the network.[2]

B. Results

Impact of parameters. We evaluated the performance of DEvOTION in terms of delurking capital obtained by varying all three parameters involved, i.e., k, LS-*perc*, η. Focusing first on the impact of η, we varied it from $1e$-03 to 0 (no path pruning). Figure 2 shows results obtained on *Instagram*, for different settings of η. It can be noted that no significant effects on DC are yielded by varying η and keeping fixed the other two parameters; only for a relatively large target set (e.g., LS-*perc* $= 25\%$) and $k \geq 30$, we observe a slight reduction in DC for $\eta = 1e$-03. This indicates that no significant gain in spread (DC) is obtained for lower values of η. This fact achieves particular importance when it is coupled with the time performance results shown in Fig. 6: several orders of magnitude in the runtime are saved when $\eta > 0$, and from this perspective $\eta = 1e$-03 is also to be preferred to $\eta = 1e$-04. Hereinafter in our discussion, unless otherwise specified, we assume that η is set to $1e$-03.

Concerning the other two parameters, as expected, the delurking capital increases with the size of the target set (LS-*perc*) and the size of the seed set (k). The increasing trend with k is clearer for higher LS-*perc* (e.g., for LS-*perc* $= 25\%$, $DC \simeq 2180$ with $k = 5$ up to $DC \simeq 3359$ for $k = 25$).

Figure 3 shows the DC performance on the largest datasets. Besides the obviously higher values of DC than on *Instagram*, the growing trend of DC in function of k is confirmed and even more evident for higher LS-*perc* on both *FriendFeed* and *Google+*. This would indicate that the larger the network size, the higher the impact of k on the delurking capital obtained. More importantly, we observe on all datasets that for smaller target sets (i.e., LS-*perc* $\leq 10\%$), a significant fraction of delurking capital can be achieved using low k.

We also investigated how the target nodes are distributed in the final active set, for varying k and LS-*perc*. To this aim, we analyzed the density distributions $pdf(x)$ with variable x modeling the vector of total influence weights associated with the nodes in the final active set. Figure 4 shows that the plotted distributions are generally roughly bimodal, for every k and LS-*perc*. The two major peaks of each distribution are located at a low regime (i.e., $0.0 \lessapprox x \lessapprox 0.4$) and mid-high regime (i.e., $0.6 \lessapprox x \lessapprox 0.8$) of total influence weights. More specifically, while the density at the low regimes always shows a significant variance and increases with higher LS-*perc*, the opposite situation is observed for mid-high regimes. The latter finding is important as it implies that a high total influence (within 0.9, for LS-*perc* $= 5\%$, and 0.7, for LS-*perc* $> 5\%$) is always achieved by a large (for LS-*perc*=5%) or roughly half (for LS-*perc*=25%) fraction of the target set.

Comparison with baselines. Figure 5 shows the delurking capital obtained by the three baselines in function of k and LS-*perc*, on *Instagram*. Both Random and bottom-LR methods achieve very low DC (up to 10.5 for the Random, and 37.9 for bottom-LR, with LS-*perc* $= 25\%$), although the growth in function of k evolve with different trends. In the Random case, the delurking capital follows a roughly linear

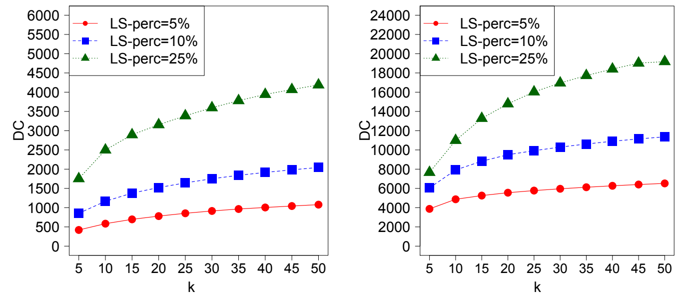


Fig. 3. Delurking capital in function of k for varying LS-*perc*, on *Google+* (left) and *FriendFeed* (right).

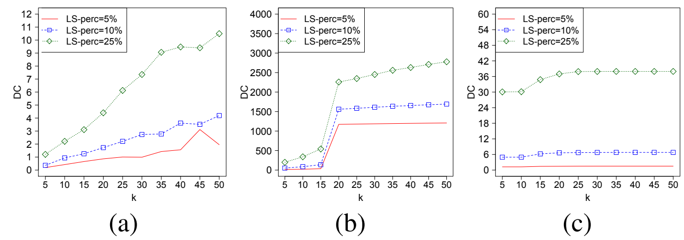
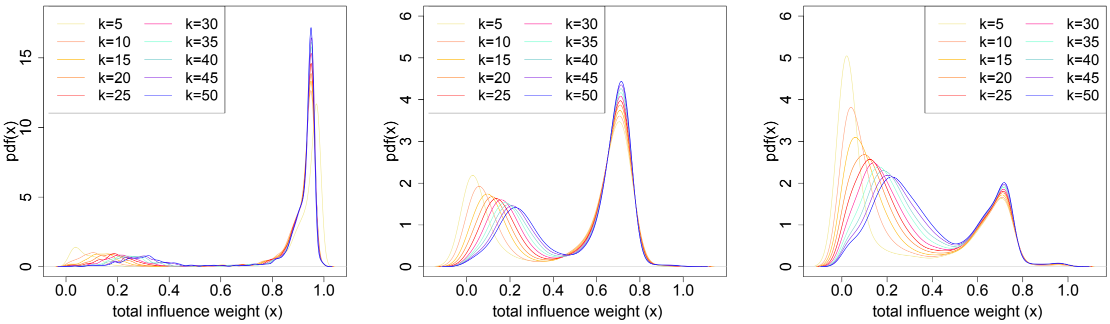


Fig. 5. Delurking capital in function of k and LS-*perc*: (a) Random, (b) LargestDegree, and (c) bottom-LR baselines, on *Instagram*.

[1]More details on the evaluation data are available at `http://uweb.dimes.unical.it/tagarelli/devotion/`.

[2]Experiments were carried out on an Intel Core i7-3960X CPU @3.30GHz, 64GB RAM machine. All algorithms were written in Python.



(a) (b) (c)

Fig. 4. Density distributions of total influence weight, for varying k, with LS-*perc* set to (a) 5%, (b) 10%, and (c) 25%, on *Instagram*. (Best viewed in color).

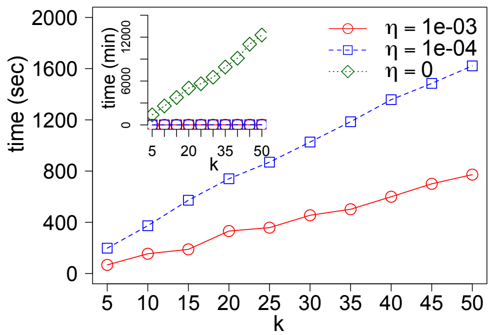


Fig. 6. Execution time of DEvOTION in function of k, with $LS\text{-}perc = 5\%$ and for varying η, on *Instagram*. The inset shows the overall results that include $\eta = 0$, while the main plot zooms in to focus on $\eta \neq 0$.

trend by increasing k, except for some fluctuation naturally due to the random selection of the seed set. By contrast, bottom-LR exhibits a behavior that is nearly constant with respect to k; this is actually not surprising since choosing globally low-ranked lurkers (i.e., users regarded as highly active in the SN), does not ensure a good influence spread towards a selected target set, which can mainly be ascribed to connectivity features of the SN. The best performing baseline is LargestDegree, which shows a rapidly increasing ramp for $k \approx 15$, reaching DC values that are of similar order of magnitude as DEvOTION although always significantly lower than our algorithm. Moreover, even though peaking seeds with high out-degree might increase the probability of spreading the influence towards the target set, the reachability of target nodes is not guaranteed, like for the other baselines.

IV. Conclusion

We proposed the first computational approach to delurking silent users in SNs. We defined a novel targeted IM problem in which the objective function to be maximized is defined in terms of delurking capital of the target users. We proved that the proposed objective function is monotone and submodular, by using the LT-equivalent live-edge graph model, and developed an approximate algorithm, DEvOTION, to solve the problem under consideration. Our algorithm has shown to be robust w.r.t. the pruning of paths to be explored in the graph. A significant fraction of delurking capital can be achieved already with a small seed set, even for large network datasets. DEvOTION is also robust w.r.t. the size of both the seed set and target set in terms of total influence received by the target nodes. We finally point out that the proposed approach can easily be generalized to deal with other targeted IM scenarios.

References

[1] J. Bishop, “Increasing participation in online communities: a framework for human-computer interaction,” *Computers in Human Behavior*, 23:1881–1893, 2007.

[2] F. Celli, F. M. L. D. Lascio, M. Magnani, B. Pacelli, and L. Rossi, “Social Network Data and Practices: The Case of Friendfeed,” in *Proc. SBP*, 2010, pp. 346–353.

[3] N. Edelmann, “Reviewing the definitions of “lurkers” and some implications for online research,” *Cyberpsychology, Behavior, and Social Networking*, 16(9):645–649, 2013.

[4] A. Goyal, W. Lu, and L. V. S. Lakshmanan, “SIMPATH: An Efficient Algorithm for Influence Maximization under the Linear Threshold Model,” in *Proc. IEEE ICDM*, 2011, pp. 211–220.

[5] A. Guille, H. Hacid, C. Favre, and D. A. Zighed, “Information diffusion in online social networks: a survey,” *SIGMOD Rec.*, 42(2):17–28, 2013.

[6] B. Guler, B. Varan, K. Tutuncuoglu, M. S. Nafea, A. A. Zewail, A. Yener, and D. Octeau, “Optimal strategies for targeted influence in signed networks,” in *Proc. ASONAM*, 2014, pp. 906–911.

[7] J. Guo, P. Zhang, C. Zhou, Y. Cao, and L. Guo, “Personalized influence maximization on social networks,” in *Proc. CIKM*, 2013, pp. 199–208.

[8] D. Kempe, J. M. Kleinberg, and E. Tardos, “Maximizing the spread of influence through a social network,” in *Proc. KDD*, 2003, pp. 137–146.

[9] C. Lagnier, L. Denoyer, E. Gaussier, and P. Gallinari, “Predicting information diffusion in social networks using content and user’s profiles,” in *Proc. ECIR*, 2013, pp. 74–85.

[10] H. Li, S. S. Bhowmick, A. Sun, and J. Cui, “Conformity-aware influence maximization in online social networks,” *The VLDB Journal*, 24:117–141, 2015.

[11] J. J. McAuley and J. Leskovec, “Learning to Discover Social Circles in Ego Networks,” in *Proc. NIPS*, 2012, pp. 548–556.

[12] N. Sun, P. P.-L. Rau, and L. Ma, “Understanding lurkers in online communities: a literature review,” *Computers in Human Behavior*, 38:110–117, 2014.

[13] A. Tagarelli and R. Interdonato, “Lurking in social networks: topology-based analysis and ranking methods,” *Social Network Analysis and Mining*, 4(230), 2014.

[14] A. Tagarelli and R. Interdonato, “Understanding lurking behaviors in social networks across time,” in *Proc. ASONAM*, 2014, pp. 51–55.

[15] A. Tagarelli and R. Interdonato, ““Who’s out there?”: Identifying and Ranking Lurkers in Social Networks,” in *Proc. ASONAM*, 2013, pp. 215–222.

[16] F. Tang, Q. Liu, H. Zhu, E. Chen, and F. Zhu, “Diversified social influence maximization,” in *Proc. ASONAM*, 2014, pp. 455–459.

[17] D. Yang, H. Hung, W. Lee, and W. Chen, “Maximizing acceptance probability for active friending in online social networks,” in *Proc. KDD*, 2013, pp. 713–721.

[18] J. Zhou, Y. Zhang, and J. Cheng, “Preference-based mining of top-k influential nodes in social networks,” *Future Generation Computer Systems*, 31:40–47, 2014.