

Influence Maximization for Social Good: Use of Social Networks in Low Resource Communities

Ph.D. Dissertation Proposal
submitted by

Amulya Yadav

July 2016

Guidance Committee

Milind Tambe (Chairperson)
Kristina Lerman
Aram Galstyan
Eric Rice (Outside Member)
Dana Goldman (Outside Member)

Abstract

The field of influence maximization involves finding “*influential*” members of a social network. This field has seen a lot of progress in recent years, with a significant number of models and algorithms having been proposed in previous work. Unfortunately, most of these models and algorithms do not address challenges that are crucial in many real-world domains such as uncertainties in network structure, selection of nodes in multiple stages, etc. This begets the following question: Can current state-of-the-art influence maximization techniques be deployed successfully in these domains? If not, can influence maximization algorithms, which take into account the aforementioned real-world challenges, be designed? Finally, can these influence maximization algorithms be deployed successfully in the real world for society’s benefit?

This thesis attempts to answer all these questions by focusing on domains where influence maximization can potentially be used for social good. In particular, this thesis is strongly motivated by the following domain: raising awareness about Human Immunodeficiency Virus (HIV) among homeless youth. HIV infections are ten times more common among homeless youth than among youth with stable homes, mainly due to their involvement in high-risk activities such as unprotected sex, sharing drug needles, etc. To prevent spread of HIV, many homeless shelters conduct intervention camps for raising awareness about HIV prevention and treatment practices among homeless youth. However, manpower constraints faced by these shelters mean that they can only intervene upon a small set ($\sim 3-4$) of youth in every intervention. Moreover, these shelters prefer a series of small sized intervention camps organized sequentially.

With this domain in mind (which is just one of many possible domains), this thesis makes the following technical contributions: (i) we provide a definition of the Dynamic Influence Maximization Under Uncertainty (or DIME) problem, which models the problem faced by homeless shelters accurately; (ii) we propose a novel Partially Observable Markov Decision Process (POMDP) model for solving the DIME problem; (iii) we design two scalable POMDP algorithms (PSINET and HEALER) for solving the DIME problem, since conventional POMDP solvers fail to scale up to sizes of interest; and (iv) we test our algorithms effectiveness in the real world by conducting a pilot study with actual homeless youth in Los Angeles. The success of this pilot (as explained later) shows the promise of using influence maximization for social good on a larger scale.

Further, we also conduct extensive experimentation by simulating our algorithms on (i) real-world networks of homeless youth; and (ii) artificially generated networks like Erdos-Renyi, Watts-Strogatz, etc. Our results show that our POMDP algorithms (i) are able to successfully scale up to real-world network sizes; (ii) are robust to changes in input parameter values; and (iii) our algorithm’s performance over baselines improves as network sizes scale up.

In the future, we plan to improve the ease of our system’s adoption by homeless shelter officials by designing an explanation system for justifying solutions generated by our algorithms in an intuitive manner. We also plan to explore hybrid approaches (i.e., combination of POMDPs and greedy methods) to solve the DIME problem. Finally, we also plan to conduct a much larger study with 900 homeless youth to further validate the effectiveness of our algorithms.

Chapter 1

Introduction

Since ancient times, humans have intertwined themselves into various social networks. These networks can be of many different kinds, such as friendship based networks, consumer/seller networks, professional networks, etc. Besides these networks being used for more direct reasons (e.g., friendship based networks used for connecting with old and new friends, consumer/seller networks used for finding out highly rated sellers providing cheap deals, etc.), these networks also play a critical role in the formulation and propagation of opinions and ideas among the people in that network. In recent times, this property of social networks has been exploited by companies to popularize their products among consumers, and by election campaigners to win support among voters, among other examples.

From a computer science perspective, the question of finding out the most “*influential*” people in a social network is well studied in the field of influence maximization, which looks at the problem of selecting the K (an input parameter) most influential nodes in a social network (represented as a graph), who will be able to influence the most number of people in the network within a given time period. Influence in these networks is assumed to spread according to a known *influence model* (popular ones are independent cascade [23] and linear threshold [4]). Since the field’s inception in 2003 by Kempe et. al. [16], influence maximization has seen a lot of progress over the years [23, 18, 4, 5, 2, 33, 1, 19, 3, 22, 12, 13, 36, 11, 10].

Unfortunately, most models and algorithms from previous work do not address challenges that are crucial in many real-world domains such as uncertainties in network structure, selection of nodes in multiple stages, etc. Specifically, most previous work suffers from four potential major limitations. First, almost every previous work focuses on single-shot decision problems, where only a single subset of graph nodes is to be chosen and then evaluated for influence spread. Instead, most realistic applications of influence maximization would require selection of nodes in multiple stages (as we illustrate later). Second, the state of the network is not known; thus, the selection of nodes in multiple stages (which is unhandled in previous work) introduces additional uncertainty about which network nodes are influenced at a given point in time, which complicates the node selection procedure. Third, network structure is assumed to be known with certainty in

most previous work, which is untrue in reality, considering that there is always noise in any network data collection procedure. Fourth, most previous work in influence maximization has used two probabilistic influence models (the independent cascade model and the linear threshold model) for their mathematical elegance, but their effectiveness in modeling the influence spread process remains invalidated in the real world.

This thesis attempts to resolve these limitations and is motivated by an important domain, where influence maximization could be used for social good: raising awareness about Human Immunodeficiency Virus (HIV) among homeless youth. HIV-AIDS is a dangerous disease which claims 1.5 million lives annually [35], and homeless youth are particularly vulnerable to HIV due to their involvement in high risk behavior such as unprotected sex, sharing drug needles, etc. [7]. To prevent the spread of HIV, many homeless shelters conduct intervention camps, where a select group of homeless youth are trained as “peer leaders” to lead their peers towards safer practices and behaviors, by giving them information about safe HIV prevention and treatment practices. These peer leaders are then tasked with spreading this information among people in their social circle.

However, due to financial/manpower constraints, the shelters can only organize a limited number of intervention camps. Moreover, in each camp, the shelters can only manage small groups of youth ($\sim 3-4$) at a time (as emotional and behavioral problems of youth makes management of bigger groups difficult). Thus, the shelters prefer a series of small sized camps organized sequentially [29]. As a result, the shelter cannot intervene on the entire target (homeless youth) population. Instead, it tries to maximize the spread of awareness among the target population (via word-of-mouth influence) using the limited resources at its disposal. To achieve this goal, the shelter uses the friendship based social network of the target population to strategically choose the participants of their limited intervention camps. Unfortunately, the shelters’ job is further complicated by a lack of complete knowledge about the social network’s structure [27]. Some friendships in the network are known with certainty whereas there is uncertainty about other friendships.

Thus, the shelters face an important challenge: they need a sequential plan to choose the participants of their sequentially organized interventions. This plan must address three key points: (i) it must deal with network structure uncertainty; (ii) it needs to take into account new information uncovered during the interventions, which reduces the uncertainty in our understanding of the network; and (iv) the intervention approach should address the challenge of gathering information about social networks of homeless youth, which usually costs thousands of dollars and many months of time [29].

The contributions of this thesis are as follows. First, we use the homeless youth domain to motivate the definition of the Dynamic Influence Maximization Under Uncertainty (DIME) problem [38], which models the aforementioned challenge faced by the homeless shelters accurately. Infact, the sequential selection of network nodes in multiple stages in DIME sets it apart from

any other previous work in influence maximization [23, 18, 4, 5]. Second, we introduce a novel Partially Observable Markov Decision Process (POMDP) based model for solving DIME, which takes into account uncertainties in network structure and evolving network state. Third, since conventional POMDP solvers fail to scale up to sizes of interest (our POMDP had 2^{300} states and $\binom{150}{6}$ actions), we design two scalable (and more importantly, “*deployable*”) algorithms, which use our POMDP model to solve the DIME problem.

Our first algorithm PSINET [37] relies on the following key ideas: (i) compact representation of transition probabilities to manage the intractable state and action spaces; (ii) combination of the QMDP heuristic with Monte-Carlo simulations to avoid exhaustive search of the entire belief space; and (iii) voting on multiple POMDP solutions, each of which efficiently searches a portion of the solution space to improve accuracy. Unfortunately, even though PSINET was able to scale up to real-world sized networks, it completely failed at scaling up in the number of nodes that get picked in every round (intervention). To address this challenge, we designed HEAL, our second algorithm. HEAL [38] hierarchically subdivides our *original POMDP* at two layers: (i) In the top layer, graph partitioning techniques are used to divide the *original POMDP* into *intermediate POMDPs*; (ii) In the second level, each of these *intermediate POMDPs* is further simplified by sampling uncertainties in network structure repeatedly to get *sampled POMDPs*; (iii) Finally, we use aggregation techniques to combine the solutions to these simpler POMDPs, in order to generate the overall solution for the *original POMDP*.

Finally, we have also tested HEAL’s performance in a real-world pilot study, in collaboration with Safe Place for Youth¹, a homeless shelter in Los Angeles which provides food and lodging to homeless youth. We enrolled 60 homeless youth from Safe Place for Youth and conducted multiple interventions and post-intervention surveys to evaluate the effectiveness of HEAL at spreading awareness about HIV. Our pilot results showed that HEAL was able to propagate information to 66% of all surveyed youth. More importantly, HEAL was successful in behavior change as well, with a 25% increase in the number of youth who get tested for HIV regularly. These results point to the usability of HEAL in the real world. More importantly, it illustrates one way (among many others) in which social networks and influence maximization can be harnessed for doing social good.

¹<http://safeplaceforyouth.nationbuilder.com/>

Chapter 2

Background and Related Work

2.1 Related Work

The influence maximization problem, as stated by Kempe et. al. [16], takes in a social network as input (in the form of a graph), and outputs a set of K ‘seed nodes’ which maximize the expected influence spread in the social network within T time steps. Note that the expectation of influence spread is taken with respect to a probabilistic influence model (explained later), which is also provided as input to the problem.

There are many algorithms for finding ‘seed sets’ of nodes to maximize influence spread in networks [16, 23, 2, 33, 22, 12, 13, 36, 11, 10]. However, all these algorithms assume *no uncertainty in the network structure* and select a single seed set. In contrast, we select several seed sets sequentially in our work to select intervention participants, as that is a natural requirement arising from our homeless youth domain. Also, our work takes into account uncertainty about the network structure and influence status of network nodes (i.e., whether a node is influenced or not). Finally, unlike most previous work [16, 23, 2, 33, 22, 12, 13, 36, 11, 10], we use a different influence model as we explain later.

There is another line of work by Golovin et. al. [14], which introduces adaptive submodularity and discusses adaptive sequential selection (similar to our problem). They prove that a Greedy algorithm provides a $(1 - 1/e)$ approximation guarantee. However, unlike our work, they assume no uncertainty in network structure. Also, while our problem can be cast into the adaptive stochastic optimization framework of [14], our influence function is not adaptive submodular (as shown later), because of which their Greedy algorithm loses its approximation guarantees.

2.2 Network Representation and Influence Model

We represent social networks as directed graphs (consisting of *nodes* and *directed edges*) where each *node* represents a person in the social network and a *directed edge* between two nodes A and B (say) represents that node A considers node B as his/her friend. We assume directed-ness of

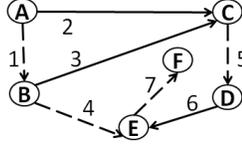


Figure 2.1: Uncertain Network

edges as sometimes homeless shelters assess that the influence in a friendship is very much uni-directional; and to account for uni-directional follower links on Facebook. Otherwise friendships are encoded as two uni-directional links.

Uncertain Network: The uncertain network is a directed graph $G = (V, E)$ with $|V| = N$ nodes and $|E| = M$ edges. The edge set E consists of two disjoint subsets of edges: E_c (the set of certain edges, i.e., friendships which we are certain about) and E_u (the set of uncertain edges, i.e., friendships which we are uncertain about). Note that uncertainties about friendships exist because HEALER’s Facebook application misses out on some links between people who are friends in real life, but not on Facebook.

To model the uncertainty about missing edges, every uncertain edge $e \in E_u$ has an existence probability $u(e)$ associated with it, which represents the likelihood of “existence” of that uncertain edge. For example, if there is an uncertain edge (A, B) (i.e., we are unsure whether node B is node A ’s friend), then $u(A, B) = 0.75$ implies that B is A ’s friend with a 0.75 chance. In addition, each edge $e \in E$ (both certain and uncertain) has a propagation probability $p(e)$ associated with it. A propagation probability of 0.5 on directed edge (A, B) denotes that if node A is influenced (i.e., has information about HIV prevention), it influences node B (i.e., gives information to node B) with a 0.5 probability in each subsequent time step (our full influence model is defined below). This graph G with all relevant $p(e)$ and $u(e)$ values represents an uncertain network and serves as an input to the DIME problem. Figure 2.1 shows an uncertain network on 6 nodes (A to F) and 7 edges. The dashed and solid edges represent uncertain (edge numbers 1, 4, 5 and 7) and certain (edge numbers 2, 3 and 6) edges, respectively.

Influence Model We use a variant of the independent cascade model [39]. In the standard independent cascade model, all nodes that get influenced at round t get a **single** chance to influence their un-influenced neighbors at time $t + 1$. If they fail to spread influence in this **single** chance, they don’t spread influence to their neighbors in future rounds. Our model is different in that we assume that nodes get **multiple** chances to influence their un-influenced neighbors. If they succeed in influencing a neighbor at a given time step t' , they stop influencing that neighbor for all future time steps. Otherwise, if they fail in step t' , they try to influence again in the next round. This variant of independent cascade has been shown to empirically provide a better approximation to real influence spread than the standard independent cascade model [6, 39]. Further, we assume that nodes that get influenced at a certain time step remain influenced for all future time steps.

2.3 POMDP

Partially Observable Markov Decision Processes (POMDPs) are a well studied model for sequential decision making under uncertainty [26]. Intuitively, POMDPs model situations wherein an agent tries to maximize its expected long term *rewards* by taking various *actions*, while operating in an environment (which could exist in one of several *states* at any given point in time) which reveals itself in the form of various *observations*. The key point is that the exact state of the world is not known to the agent and thus, these actions have to be chosen by reasoning about the agent's probabilistic beliefs (belief state). The agent, thus, takes an action (based on its current belief), and the environment transitions to a new world state. However, information about this new world state is only partially revealed to the agent through observations that it gets upon reaching the new world state. Hence, based on the agent's current belief state, the action that it took in that belief state, and the observation that it received, the agent updates its belief state. The entire process repeats several times until the environment reaches a terminal state (according to the agent's belief).

More formally, a full description of the POMDP includes the sets of possible environment *states*, the set of *actions* that the agent can take, and the set of possible *observations* that the agent can observe. In addition, the full POMDP description includes a *transition matrix*, for storing *transition probabilities*, which specify the probability with which the environment transitions from one *state* to another, conditioned on the immediate *action* taken. Another component of the POMDP description is the *observation matrix*, for storing *observation probabilities*, which specify the probability of getting different *observations* in different *states*, conditioned on the *action* taken to reach that *state*. Finally, the POMDP description includes a *reward matrix*, which specifies the agent's *reward* of taking *actions* in different *states*.

A POMDP policy Π provides a mapping from every possible belief state (which is a probability distribution over world states) to an action $a = \Pi(\beta)$. Our aim is to find an optimal policy Π^* which, given an initial belief β_0 , maximizes the expected cumulative long term reward over H horizons (where the agent takes an action and gets a reward in each time step until the horizon H is reached). Computing optimal policies offline for finite horizon POMDPs is PSPACE-Complete. Thus, focus has recently turned towards online algorithms, which only find the best action for the current belief state [24, 31]. Thus, online planning interleaves planning and execution at every time step.

Chapter 3

DIME Problem

We now provide some background information that helps us define a precise problem statement for DIME. After that, we will show some hardness results about this problem statement.

Given the *uncertain network* as input, we plan to run for T rounds (corresponding to the number of interventions organized by the homeless shelter). In each round, we will choose K nodes (youth) as intervention participants. These participants are assumed to be influenced post-intervention with certainty. Upon influencing the chosen nodes, we will ‘observe’ the true state of the *uncertain edges* (friendships) out-going from the selected nodes. This translates to asking intervention participants about their 1-hop social circles, which is within the homeless shelter’s capabilities [28].

After each round, influence spreads in the network according to our influence model for L time steps, before we begin the next round. This L represents the time duration in between two successive intervention camps. *In between rounds, we do not observe the nodes that get influenced during L time steps.* We only know that explicitly chosen nodes (our intervention participants in all past rounds) are influenced. Informally then, given an uncertain network $G_0 = (V, E)$ and integers T , K , and L (as defined above), our goal is to find an online policy for choosing *exactly* K nodes for T successive rounds (interventions) which maximizes influence spread in the network at the end of T rounds.

We now provide notation for defining an online policy formally. Let $\mathcal{A} = \{A \subset V \text{ s.t. } |A| = K\}$ denote the set of K sized subsets of V , which represents the set of possible choices that we can make at every time step $t \in [1, T]$. Let $A_i \in \mathcal{A} \forall i \in [1, T]$ denote our choice in the i^{th} time step. Upon making choice A_i , we ‘observe’ uncertain edges adjacent to nodes in A_i , which updates its understanding of the network. Let $G_i \forall i \in [1, T]$ denote the uncertain network resulting from G_{i-1} with *observed* (additional edge) information from A_i . Formally, we define a history $H_i \forall i \in [1, T]$ of length i as a tuple of past choices and observations $H_i = \langle G_0, A_1, G_1, A_2, \dots, A_{i-1}, G_i \rangle$. Denote by $\mathcal{H}_i = \{H_k \text{ s.t. } k \leq i\}$ the set of all possible histories of length less than or equal to i . Finally, we define an i -step policy $\Pi_i: \mathcal{H}_i \rightarrow \mathcal{A}$ as a function

that takes in histories of length less than or equal to i and outputs a K node choice for the current time step. We now provide an explicit problem statement for DIME.

Problem 1 *DIME Problem* *Given as input an uncertain network $G_0 = (V, E)$ and integers $T, K,$ and L (as defined above). Denote by $\mathcal{R}(H_T, A_T)$ the expected total number of influenced nodes at the end of round T , given the T -length history of previous observations and actions H_T , along with A_T , the action chosen at time T . Let $\mathbb{E}_{H_T, A_T \sim \Pi_T}[\mathcal{R}(H_T, A_T)]$ denote the expectation over the random variables $H_T = \langle G_0, A_1, \dots, A_{T-1}, G_T \rangle$ and A_T , where A_i are chosen according to $\Pi_T(H_i) \forall i \in [1, T]$, and G_i are drawn according to the distribution over uncertain edges of G_{i-1} that are revealed by A_i . The objective of DIME is to find an optimal T -step policy $\Pi_T^* = \operatorname{argmax}_{\Pi_T} \mathbb{E}_{H_T, A_T \sim \Pi_T}[\mathcal{R}(H_T, A_T)]$.*

We now analyze the hardness of computation in the DIME problem in the next two theorems. More details and proofs of these theorems can be found in Yadav et. al. [38].

Theorem 1 *The DIME problem is NP-Hard.*

Some NP-Hard problems exhibit nice properties that enable approximation guarantees for them. Golovin et. al. [14] introduced adaptive submodularity, an analog of submodularity for adaptive settings. Presence of adaptive submodularity ensures that a simply greedy algorithm provides a $(1 - 1/e)$ approximation guarantee w.r.t. the optimal solution defined on the *uncertain network*. However, as we show next, while DIME can be cast into the adaptive stochastic optimization framework of [14], our influence function is not adaptive submodular, because of which their Greedy algorithm does not have a $(1 - 1/e)$ approximation guarantee.

Theorem 2 *The influence function of DIME is not adaptive submodular.*

Chapter 4

POMDP Model

The theorems in the previous chapter show that DIME is a hard problem as it is difficult to even obtain any reasonable approximations. We model DIME as a POMDP [26] because of two reasons. First, POMDPs are a good fit for DIME as (i) we conduct several interventions sequentially, similar to sequential POMDP actions; and (ii) we have *partial observability* (similar to POMDPs) due to uncertainties in network structure and influence status of nodes. Second, POMDP solvers have recently shown great promise in generating near-optimal policies efficiently [31]. We now explain how we map DIME onto a POMDP.

States. A POMDP state in our problem is a pair of binary tuples $s = \langle W, F \rangle$ where W and F are of lengths $|V|$ and $|E_U|$, respectively. Intuitively, W denotes the influence status of network nodes, where $W_i = 1$ denotes that node i is influenced and $W_i = 0$ otherwise. Moreover, F denotes the existence of uncertain edges, where $F_i = 1$ denotes that the i^{th} uncertain edge exists in reality, and $F_i = 0$ otherwise.

Actions. Every choice of a subset of K nodes is a POMDP action. More formally, $A = \{a \subset V \text{ s.t. } |a| = K\}$. For example, in Figure 2.1, one possible action is $\{A, B\}$ (when $K = 2$).

Observations. Upon taking a POMDP action, we “*observe*” the ground reality of the uncertain edges outgoing from the nodes chosen in that action. Consider $\Theta(a) = \{e \mid e = (x,y) \text{ s.t. } x \in a \wedge e \in E_u\} \forall a \in A$, which represents the (ordered) set of uncertain edges that are observed when we take action a . Then, our POMDP observation upon taking action a is defined as $o(a) = \{F_e \mid e \in \Theta(a)\}$, i.e., the F-values of the observed uncertain edges. For example, by taking action $\{B, C\}$ in Figure 2.1, the values of F_4 and F_5 (i.e., the F-values of uncertain edges in the 1-hop social circle of nodes B and C) would be observed.

Rewards. The reward $R(s, a, s')$ of taking action a in state s and reaching state s' is the number of newly influenced nodes in s' . More formally, $R(s, a, s') = (\|s'\| - \|s\|)$, where $\|s'\|$ is the number of influenced nodes in s' .

Initial Belief State. The initial belief state is a distribution β_0 over all states $s \in S$. The support of β_0 consists of all states $s = \langle W, F \rangle$ s.t. $W_i = 0 \forall i \in [1, |V|]$, i.e., all states in which

all network nodes are un-influenced (as we assume that all nodes are un-influenced to begin with). Inside its support, each F_i is distributed independently according to $P(F_i = 1) = u(e)$.

Transition And Observation Probabilities. Computation of exact transition probabilities $T(s'|s, a)$ requires considering all possible paths in a graph through which influence could spread, which is $\mathcal{O}(N!)$ (N is number of nodes in the network) in the worst case. Moreover, for large social networks, the size of the transition and observation probability matrix is prohibitively large (due to exponential sizes of state and action space). Therefore, instead of storing huge transition/observation matrices in memory, we follow the paradigm of large-scale online POMDP solvers [31, 9] by using a generative model $\Lambda(s, a) \sim (s', o, r)$ of the transition and observation probabilities. This generative model allows us to generate on-the-fly samples from the exact distributions $T(s'|s, a)$ and $\Omega(o|a, s')$ at very low computational costs. Given an initial state s and an action a to be taken, our generative model Λ simulates the random process of influence spread to generate a random new state s' , an observation o and the obtained reward r . Details of the random process of simulation can be found in Yadav et. al. [37, 38].

First, we tried to solve this POMDP using current state-of-the-art solvers. Initial experiments with ZMDP [32] showed that state-of-the-art offline POMDP planners ran out of memory on 10 node graphs. Thus, we focused on POMCP [31], a state-of-the-art online POMDP solver. Unfortunately, even POMCP ran out of memory on 30 node graphs. In the next two chapters, we provide two different scalable POMDP algorithms (PSINET and HEAL) for solving the DIME problem.

Chapter 5

PSINET

As a first attempt at solving DIME, we proposed PSINET, a Monte Carlo (MC) sampling based online planner that simplifies the POMDP into smaller sub-problems and then uses various kinds of voting techniques to aggregate the solutions of the sub-problems.

The flow of PSINET proceeds in three distinct steps, as shown in Figure 5.1(a). First, graph sampling techniques are used to get many different graph instances, each of which is solved as an independent POMDP. Second, each of these independent POMDPs is run (in parallel) to find out the best action to take in each independent POMDP. Finally, all these independent POMDPs vote for the best action to be taken. These votes are then aggregated using different voting mechanisms (each different voting mechanism gives rise to different variants of PSINET, as we explain later), and the overall best action that wins this vote is executed by PSINET. We now go through each of the three distinct steps shown in Figure 5.1(a) in more detail.

Sampling Graphs First, we randomly keep or remove uncertain edges to create one graph instance. As a single instance might not represent the real network well, we instantiate the graph several times to create a set Δ of instances. Then, we use each of these instances to vote for the best action to be taken.

Finding Best Action For each sampled graph instance, we find the action which maximizes long term rewards averaged across η (we use $\eta=2^8$) MC simulations starting from states (particles) sampled from the current belief β . Each MC simulation samples a particle from β and chooses an action to take (choice of action is explained in Yadav et. al. [37]). Then, upon taking this action, we follow a uniform random rollout policy (until either termination, i.e., all nodes get influenced, or the horizon is breached) to estimate the long term reward, which we get by taking the “selected” action. Finally, we pick the action with the maximum average reward.

Voting Mechanisms Finally, each network instance votes for the best action (found in the previous step) to take in the uncertain graph and the overall best action (approximate) is chosen by aggregating these votes according to different voting schemes. We propose using the following three different voting schemes, each of which leads to a different variant of PSINET:

PSINET-S Each instance’s vote gets equal weight.

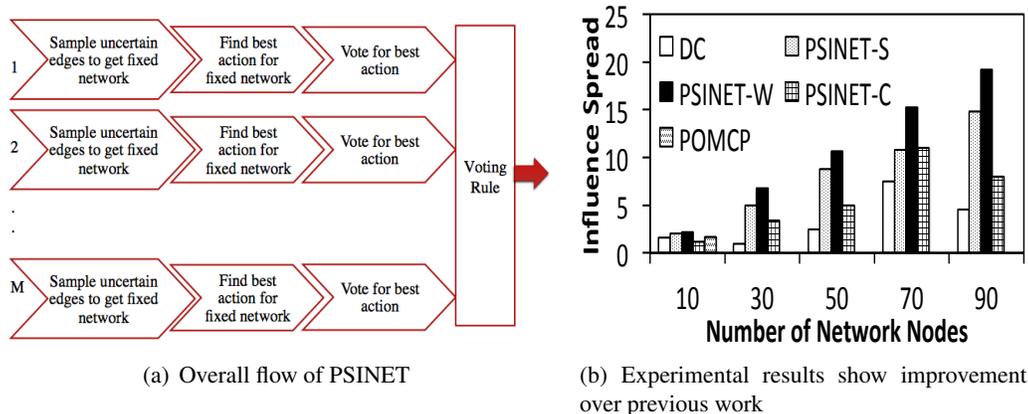


Figure 5.1: PSINET Flow and Results

PSINET-W Every instance’s vote gets weighted differently. This weighting scheme approximates the probabilities of occurrences of real world events by giving low weights to instances which removes either too few or too many uncertain edges, since those events are less likely to occur. Instances which remove exactly half the number of uncertain edges get the highest weight, since that event is most likely [37].

PSINET-C Given a ranking over actions, the Copeland rule makes pairwise comparisons among all actions, and picks the one preferred by a majority of instances over the highest number of other actions [25]. In PSINET-C, similar to Jiang. et. al. [15], rankings over actions are generated for each instance, and the winner decided using Copeland rule.

Figure 5.1(b) compares solution qualities of Degree Centrality (DC), POMCP and PSINET-(S,W and C) on BTER [30] networks of varying sizes. DC selects nodes in decreasing order of out-degrees, where every uncertain edge e adds $u(e)$ to the node degrees. We choose DC as our baseline as it is the current modus operandi of agencies working with homeless youth. The x-axis shows number of network nodes and the y-axis shows influence spread across varying horizons (number of interventions). This figure shows that all POMDP based algorithms beat DC by around 60%, which shows the value of our POMDP model. Further, it shows that PSINET-W beats PSINET-(S and C). Also, *POMCP runs out of memory on 30 node graphs*.

Unfortunately, even though PSINET was able to scale up to real-world sized networks, it completely failed at scaling up in the number of nodes that get picked in every round (intervention). Thus, while PSINET was successful in scaling up to the required POMDP state space, it failed to deal with the explosion in action space that occurred with scale up in the number of nodes picked per round. To address this challenge, we designed HEAL, which we present next.

Chapter 6

HEAL

HEAL solves the *original POMDP* using a novel *hierarchical ensembling heuristic*: it creates ensembles of imperfect (and smaller) POMDPs at *two* different layers, in a hierarchical manner (see Figure 6.1). HEAL’s *top layer* creates an ensemble of smaller sized *intermediate POMDPs* by subdividing the original *uncertain network* into several smaller sized *partitioned networks* by using graph partitioning techniques [20], which generates partitions that minimize the edges going across partitions (while ensuring that partitions have similar sizes). Since these partitions are “almost” disconnected, we solve each partition separately. Each of these partitioned networks is then mapped onto a POMDP, and these *intermediate POMDPs* form our *top layer* ensemble of POMDP solvers.

In the bottom layer, each *intermediate POMDP* is solved using TASP (**T**ree **A**ggregation for **S**equential **P**lanning), our novel POMDP planner, which subdivides the POMDP into another ensemble of smaller sized *sampled POMDPs*. Each member of this *bottom layer* ensemble is created by randomly sampling uncertain edges of the partitioned network to get a sampled network having no uncertain edges, and this sampled network is then mapped onto a *sampled POMDP*. Each sampled POMDP is solved using a novel tree search algorithm, which avoids the exponential branching factor seen in PSINET [37]. Finally, the solutions of POMDPs in both the *bottom* and *top layer* ensembles are aggregated using novel techniques to get the solution for HEAL’s original POMDP.

These heuristics enable scale up to real-world sizes (at the expense of sacrificing performance guarantees), as instead of solving one huge problem, we now solve several smaller problems. However, these heuristics perform very well in practice. Our simulations show that even on smaller settings, HEAL achieves a 100X speed up over PSINET, while providing a 70% improvement in solution quality; and on larger problems, *where PSINET is unable to run at all*, HEAL continues to provide high solution quality.

Bottom layer: TASP We now explain TASP, our new POMDP solver that solves each *intermediate POMDP* in HEAL’s bottom layer. Given an *intermediate POMDP* and the uncertain network it is defined on, as input, TASP goes through four steps. First, it creates an ensemble

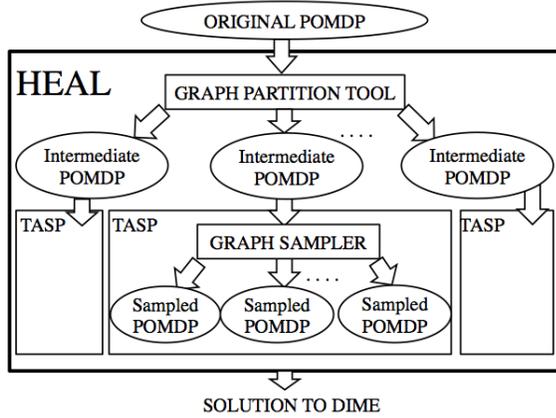


Figure 6.1: Hierarchical decomposition in HEAL

of smaller sized *sampled POMDPs*, by sampling uncertain edges of the input network to get an *instantiated* network. Each uncertain edge in the input network is randomly kept with probability $u(e)$, or removed with probability $1 - u(e)$, to get an *instantiated* network with no uncertain edges. We repeat this sampling process to get Δ (a variable parameter) different *instantiated* networks. These Δ different *instantiated* networks are then mapped onto to Δ different POMDPs, which form our ensemble of *sampled POMDPs*. Each *sampled POMDP* shares the same action space (defined on the input partitioned network) as the different POMDPs only differ in the sampling of uncertain edges.

Next, for each instantiated network $\delta \in [1, \Delta]$, we generate an α^δ list of rewards. The i^{th} element of α^δ gives the long term reward achieved by taking the i^{th} action in *instantiated* network δ . After these α^δ lists are generated, we find the expected reward r_i of taking the i^{th} action, by taking a reward expectation across the α^δ lists (for each $\delta \in [1, \Delta]$). For e.g., if $\alpha_1^{\delta_1} = 10$ and $\alpha_1^{\delta_2} = 20$, i.e., the rewards of taking the 1^{st} action in instantiated networks δ_1 and δ_2 (which occurs with probabilities $P(\delta_1)$ and $P(\delta_2)$) are 10 and 20 respectively, then the expected reward $r_1 = P(\delta_1) \times 10 + P(\delta_2) \times 20$. Note that $P(\delta_1)$ and $P(\delta_2)$ are found by multiplying existence probabilities $u(e)$ (or $1 - u(e)$) for uncertain edges that were kept (or removed) in δ_1 and δ_2 . Finally, the action $\kappa = \operatorname{argmax}_j r_j$ is returned by TASP. More details on the generation of α^δ lists can be found in Yadav et. al. [38].

Top layer: Using Graph Partitioning We now explain HEAL’s top layer, in which we use METIS [20], a state-of-the-art graph partitioning technique, to subdivide our original uncertain network into different partitioned networks. These partitioned networks form the ensemble of *intermediate POMDPs* (in Figure 6.1) in HEAL. Then, TASP (our bottom layer solver) is invoked on each intermediate POMDP independently, and their solutions are aggregated to get the final DIME solution. We try two different partitioning/aggregation techniques, which leads to two variants of HEAL:

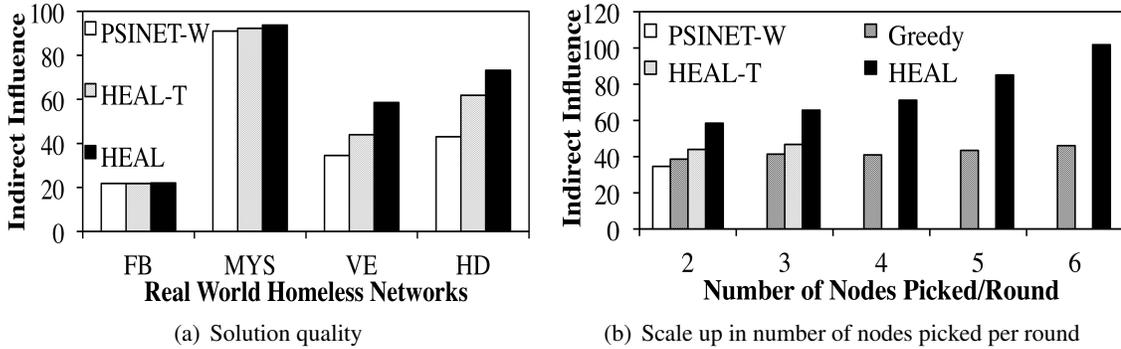


Figure 6.2: Experimental results show improvement over previous work

K Partition Variant (HEAL): Given the *uncertain* network G and the parameters K , L and T as input, we first partition the uncertain network into K partitions. In each round from 1 to T , we invoke the bottom layer TASP algorithm to select 1 node from each of the K clusters. These singly selected nodes from the K clusters give us an action of K nodes, which is given to shelter officials to execute. Based on the *observation* (about uncertain edges) that officials get while executing the action, we update the partition networks (which are input to the *intermediate POMDPs*) by either replacing the *observed* uncertain edges with certain edges (if the edge was *observed* to exist in reality) or removing the uncertain edge altogether (if the edge was *observed* to *not exist* in reality).

T Partition Variant (HEAL-T): Given the *uncertain* network G and the parameters K , L and T as input, we first partition the uncertain network into T partitions and TASP picks K nodes from the i^{th} partition ($i \in [1, T]$) in the i^{th} round.

Simulation Results Figure 6.2(a) shows the influence spread of different algorithms on four real world networks of homeless youth. The x-axis shows the four networks and the y-axis shows the influence spread achieved by the different algorithms. This figure shows that (i) HEAL outperforms all other algorithms on every network; (ii) it achieves $\sim 70\%$ improvement over PSINET-W in VE and HD networks; (iii) it achieves $\sim 25\%$ improvement over HEAL-T. The difference between HEAL and other algorithms is not significant in the FB) and MYS networks, as HEAL is already influencing almost all nodes in these two relatively small networks.

Figure 6.2(b) shows the influence spread achieved by HEAL, HEAL-T, Greedy and PSINET-W on the VE and HD networks respectively ($T = 5$), as we scale up values of K , i.e., number of nodes picked per round. The x-axis shows increasing K values, and the y-axis shows the influence spread. This figure show that (i) PSINET-W and HEAL-T fail to scale up – they cannot handle more than $K = 2$ and $K = 3$ respectively (thereby not fulfilling real world demands); (ii) HEAL outperforms all other algorithms, and the difference between HEAL and Greedy increases linearly with increasing K values.

Chapter 7

Real World Pilot Study

We conducted a pilot study with actual homeless youth in collaboration with Safe Place for Youth (a homeless shelter in Los Angeles) to test the effectiveness of HEAL in spreading information in a social network. We enrolled 62 homeless youth and constructed a social network using their Facebook contact and follower lists. We conducted three interventions, with four homeless youth selected in every intervention. For each intervention, HEAL would take the social network as input and come up with a recommendation of four network nodes as intervention attendees.

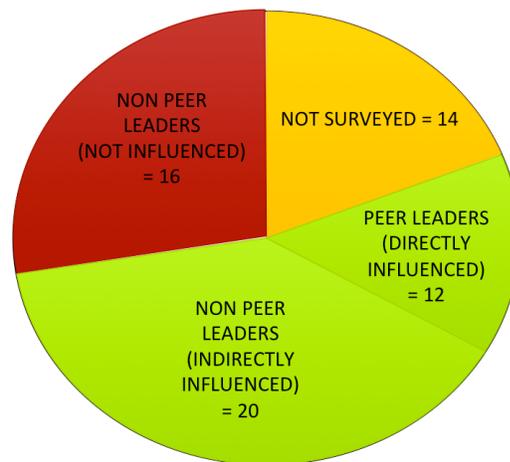


Figure 7.1: Result of real-world pilot study

Figure 7.1 shows that out of 62 homeless youth, 14 youth did not attend the post-intervention surveys. Out of the 48 surveyed youth, 36 youth reported that they received HIV related information from youth in our study. Thus, HEAL was able to influence 66% of all surveyed youth.

To the best of our knowledge, this is the first deployment of an influence maximization algorithm in the real world. Further, it is the first time that influence maximization has been applied for social good. The success of this pilot study illustrates one way (among many others) in which social networks and influence maximization can be harnessed for doing social good.

Chapter 8

Future Work

8.1 Explanation system for POMDPs

The goal of this thesis is to make influence maximization algorithms that actually get deployed in the real world for society’s benefit. One big impediment in front of this goal in the homeless youth domain, is the reluctance of homeless shelter officials to adopt state-of-the-art technology (i.e., our algorithms) in selecting their intervention attendees. Thus, there is a need to convince homeless shelter officials that our algorithms are very good at selecting “*influential*” attendees for their interventions.

Explanation systems for machine learning algorithms [34] have shown great promise in convincing end-users of the algorithm’s usefulness. These explanation systems generate an easy-to-understand explanation of the algorithm’s output for the end-user, which facilitates widespread adoption of the algorithm. Recently, explanation systems have also been built for Markov Decision Processes [17, 8]. However, no previous work has tackled the problem of building explanation systems for POMDPs, due to challenges of handling partial observability in POMDPs.

In the future, we plan to explore the feasibility of building such an explanation system for HEAL, our POMDP solver. Our goal is to be able to justify the solutions of HEAL to the homeless shelter officials in an intuitive manner. To that end, our first goal is to find out what kind of reasoning do officials (or humans in general) use to pick nodes in very simple graph settings. That will give us an understanding about what kinds of reasons are most likely to persuade humans and officials to adopt HEAL’s solutions.

8.2 Hybrid approaches

Recently, we investigated the Greedy algorithm introduced by Golovin et. al. [14] more carefully. This work (under submission) was led by another Ph.D. student¹, and resulted in a non-trivial adaptation of Golovin et. al.’s Greedy algorithm [14], which showed promising results. While

¹Bryan Wilder was first author in this research

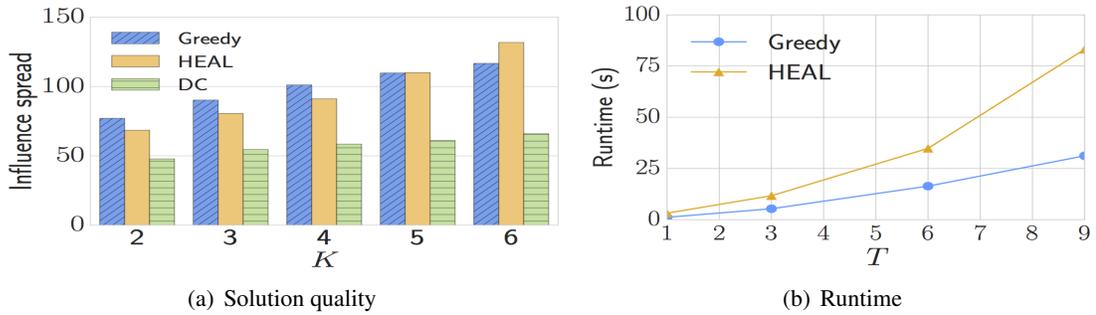


Figure 8.1: New greedy algorithm

this new Greedy algorithm has not been tested in the real world, it outperforms our POMDP algorithms (HEAL and PSINET) in simulation for some cases. As shown in Figure 8.1(a), this Greedy algorithm achieves more influence spread for smaller values of K (i.e., nodes picked per round). It also run much faster than HEAL (as shown in Figure 8.1(b)).

However, there are still cases where HEAL still outperforms this new Greedy algorithm (see large values of K in Figure 8.1(a)). This leads us to the following question: Can we combine ideas from HEAL and the new Greedy algorithm to design a “*hybrid*” algorithm which runs much faster than HEAL, and outperforms the Greedy algorithm in all cases? We would like to explore this question as part of future work.

8.3 Real world validation

We plan to conduct a much larger study with 900 homeless youth, to test the effectiveness of our algorithms. In this study, we plan to have four different groups of homeless youth (each group has its own social network) which are separated geographically, so as to ensure that the same homeless youth is not a member of two different social networks. We plan to conduct interventions about HIV awareness simultaneously in each of these four groups, with intervention attendees being selected by a different algorithm in each group. This study will allow us to compare the influence spread achieved by different algorithms such as HEAL, Greedy algorithms, Degree Centrality based approaches, uniform random selection, etc.

Further, the data collected in this study will allow us to validate the effectiveness of different influence spread models (such as independent cascade, linear threshold, or our own repeated activation model, etc.) in the real world. Similar to Lerman et. al. [21], information cascade data about who influenced whom, could be used from this study, which will allow us to fit different influence models to the data, and determine which model provides the best fit.

Bibliography

- [1] Shishir Bharathi, David Kempe, and Mahyar Salek. Competitive Influence Maximization in Social Networks. In *Internet and Network Economics*, pages 306–311. Springer, 2007.
- [2] Christian Borgs, Michael Brautbar, Jennifer Chayes, and Brendan Lucier. Maximizing Social Influence in Nearly Optimal Time. In *Proceedings of the Twenty-Fifth Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '14, pages 946–957. SIAM, 2014.
- [3] Allan Borodin, Yuval Filmus, and Joel Oren. Threshold Models for Competitive Influence in Social Networks. In *Internet and network economics*, pages 539–550. Springer, 2010.
- [4] Wei Chen, Chi Wang, and Yajun Wang. Scalable Influence Maximization for Prevalent Viral Marketing in large-scale Social Networks. In *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1029–1038. ACM, 2010.
- [5] Edith Cohen, Daniel Delling, Thomas Pajor, and Renato F Werneck. Sketch-based Influence Maximization and Computation: Scaling up with guarantees. In *Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management*, pages 629–638. ACM, 2014.
- [6] Jean-Philippe Cointet and Camille Roth. How Realistic Should Knowledge Diffusion Models Be? *Journal of Artificial Societies and Social Simulation*, 10(3):5, 2007.
- [7] National HCH Council. HIV/AIDS among Persons Experiencing Homelessness: Risk Factors, Predictors of Testing, and Promising Testing Strategies. www.nhchc.org/wp-content/uploads/2011/09/InFocus_Dec2012.pdf, December 2012.
- [8] Thomas Dodson, Nicholas Mattei, Joshua T Guerin, and Judy Goldsmith. An english-language argumentation interface for explanation generation with markov decision processes in the domain of academic advising. *ACM Transactions on Interactive Intelligent Systems (TiIS)*, 3(3):18, 2013.
- [9] Adam Eck and Leen-Kiat Soh. To Ask, Sense, or Share: Ad Hoc Information Gathering. In *Proceedings of the 2015 International Conference on Autonomous Agents and Multiagent Systems*, pages 367–376. International Foundation for Autonomous Agents and Multiagent Systems, 2015.
- [10] Aram Galstyan and Paul R Cohen. Influence propagation in modular networks. In *AAAI Spring Symposium: Social Information Processing*, pages 21–23, 2008.

- [11] Aram Galstyan, Vahe Musoyan, and Paul Cohen. Maximizing influence propagation in networks with community structure. *Physical Review E*, 79(5):056102, 2009.
- [12] Rumi Ghosh and Kristina Lerman. Leaders and negotiators: An influence-based metric for rank. In *ICWSM*, 2009.
- [13] Rumi Ghosh and Kristina Lerman. Community detection using a measure of global influence. In *Advances in Social Network Mining and Analysis*, pages 20–35. Springer, 2010.
- [14] Daniel Golovin and Andreas Krause. Adaptive Submodularity: Theory and Applications in Active Learning and Stochastic Optimization. *Journal of Artificial Intelligence Research*, 42:427–486, 2011.
- [15] A. X. Jiang, L. S. Marcolino, A. D. Procaccia, T. Sandholm, N. Shah, and M. Tambe. Diverse Randomized Agents Vote to Win. In *Proceedings of the 28th Neural Information Processing Systems Conference, NIPS 2014*, 2014.
- [16] David Kempe, Jon Kleinberg, and Éva Tardos. Maximizing the Spread of Influence through a Social Network. In *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 137–146. ACM, 2003.
- [17] Omar Zia Khan, Pascal Poupart, and James P Black. Minimal sufficient explanations for factored markov decision processes. Citeseer, 2009.
- [18] Masahiro Kimura and Kazumi Saito. Tractable Models for Information Diffusion in Social Networks. In *Knowledge Discovery in Databases: PKDD 2006*, pages 259–271. Springer, 2006.
- [19] Jan Kostka, Yvonne Anne Oswald, and Roger Wattenhofer. Word of Mouth: Rumor Dissemination in Social Networks. In *Structural Information and Communication Complexity*, pages 185–196. Springer, 2008.
- [20] Dominique LaSalle and George Karypis. Multi-threaded Graph Partitioning. In *Parallel & Distributed Processing (IPDPS), 2013 IEEE 27th International Symposium on*, pages 225–236. IEEE, 2013.
- [21] Kristina Lerman, Rumi Ghosh, and Tawan Surachawala. Social contagion: An empirical study of information spread on digg and twitter follower graphs. *arXiv preprint arXiv:1202.3162*, 2012.
- [22] Kristina Lerman, Xiaoran Yan, and Xin-Zeng Wu. The” majority illusion” in social networks. *PloS one*, 11(2):e0147617, 2016.
- [23] Jure Leskovec, Andreas Krause, Carlos Guestrin, Christos Faloutsos, Jeanne VanBriesen, and Natalie Glance. Cost-effective Outbreak Detection in Networks. In *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 420–429. ACM, 2007.
- [24] Sébastien Paquet, Ludovic Tobin, and Brahim Chaib-Draa. An Online POMDP Algorithm for Complex Multiagent Environments. In *Proceedings of the fourth international joint conference on Autonomous agents and multiagent systems*, pages 970–977. ACM, 2005.

- [25] Jean-Charles Pomerol and Sergio Barba-Romero. *Multicriterion Decision in Management: Principles and Practice*, volume 25. Springer Science & Business Media, 2000.
- [26] Martin L Puterman. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. John Wiley & Sons, 2009.
- [27] Eric Rice. The Positive Role of Social Networks and Social Networking Technology in the Condom-using Behaviors of Homeless Young People. *Public health reports*, 125(4):588, 2010.
- [28] Eric Rice, Anamika Barman-Adhikari, Norweeta G Milburn, and William Monro. Position-specific HIV risk in a Large Network of Homeless Youths. *American journal of public health*, 102(1):141–147, 2012.
- [29] Eric Rice, Anthony Fulginiti, Hailey Winetrobe, Jorge Montoya, Aaron Plant, and Timothy Kordic. Sexuality and Homelessness in Los Angeles public schools. *American Journal of Public Health*, 102, 2012.
- [30] C Seshadhri, Tamara G Kolda, and Ali Pinar. Community Structure and Scale-free Collections of Erdős-Rényi Graphs. *Physical Review E*, 85(5):056109, 2012.
- [31] David Silver and Joel Veness. Monte-Carlo Planning in large POMDPs. In *Advances in Neural Information Processing Systems*, pages 2164–2172, 2010.
- [32] Trey Smith. ZMDP Software for POMDP/MDP Planning. www.longhorizon.org/trey/zmdp/, March 2013.
- [33] Youze Tang, Xiaokui Xiao, and Yanchen Shi. Influence maximization: Near-Optimal Time Complexity meets Practical Efficiency. In *Proceedings of the 2014 ACM SIGMOD international conference on Management of data*, pages 75–86. ACM, 2014.
- [34] Nava Tintarev and Judith Masthoff. A survey of explanations in recommender systems. In *ICDE*, 2007.
- [35] UNAIDS. Report on the Global AIDS Epidemic. www.unaids.org/en/resources/campaigns/20121120_globalreport2012/globalreport/, March 2012.
- [36] Greg Ver Steeg and Aram Galstyan. Information-theoretic measures of influence based on content dynamics. In *Proceedings of the sixth ACM international conference on Web search and data mining*, pages 3–12. ACM, 2013.
- [37] A Yadav, L Marcolino, E Rice, R Petering, H Winetrobe, H Rhoades, M Tambe, and H Carmichael. Preventing HIV Spread in Homeless Populations Using PSINET. In *Proceedings of the Twenty-Seventh Conference on Innovative Applications of Artificial Intelligence (IAAI-15)*, 2015.
- [38] Amulya Yadav, Hau Chan, Albert Xin Jiang, Haifeng Xu, Eric Rice, and Milind Tambe. Using social networks to aid homeless shelters: Dynamic influence maximization under uncertainty. In *Proceedings of the 2016 International Conference on Autonomous Agents & Multiagent Systems*, pages 740–748. International Foundation for Autonomous Agents and Multiagent Systems, 2016.

- [39] Qiuling Yan, Shaosong Guo, and Dongqing Yang. Influence Maximizing and Local Influenced Community Detection based on Multiple Spread Model. In *Advanced Data Mining and Applications*, pages 82–95. Springer, 2011.