Int. J. Nonlinear Anal. Appl. In Press, 1–12
ISSN: 2008-6822 (electronic)
http://dx.doi.org/10.22075/ijnaa.2026.37384.5440

Leveraging Reinforcement Learning with Greedy Path Mining for Target Group Influence Maximization in Social Networks

Neda Binesh[a,*], Mahdieh Haddadi[a]

[a]Faculty of Mathematics, Statistics and Computer Science Semnan University, Semnan, Iran

(Communicated by Majid Eshaghi Gordji)

Abstract

Social networks are characterized by the formation of groups, where decisions are often influenced by majority consensus. This paper deals with the issue of Target Group Influence Maximization (TGIM). The objective of TGIM is to select a set of k influential nodes to maximize the activation of target group members through the dissemination of specific content or information.

We propose a novel algorithm, **Target Group Influence Maximization using Reinforcement Learning with Greedy Path mining (TGRLGP)**, to address the TGIM problem. TGRLGP leverages reinforcement learning techniques to identify optimal paths from target nodes to potential influencers. Experimental results demonstrate the superior performance of the proposed TGRLGP algorithm over existing methods. Extensive experiments on four benchmark networks show that TGRLGP consistently achieves the highest influence spread among the compared approaches, improving influence spread by up to 4.76% and by 2.12% on average over the strongest baseline.

Keywords: Q-learning; Information diffusion ; Independent cascade; Seed selection ; Monte Carlo rollout;Heuristic search.

1 Introduction

Social networks constitute online platforms designed to facilitate the creation, sharing, and dissemination of information and ideas among individuals and groups. These networks function as interconnected nodes, enabling the rapid propagation of information. A salient characteristic of social networks is their capacity to influence individuals and groups through the dissemination of opinions and perspectives. This influence can be strategically targeted to specific demographics. Furthermore, social networks have become essential tools for commercial purposes, allowing businesses to persuade consumers, raise awareness, and stimulate sales on a broader scale. In recent times, online social networks have emerged as effective channels for promoting new products and cultivating relationships between businesses and consumers. By leveraging these platforms, companies can reach a vast audience while minimizing advertising expenditures, thereby maximizing their impact.

Companies can utilize social networks like Instagram to identify influential nodes with a substantial follower base to promote their products. By distributing product samples and encouraging these influencers to endorse the product,

*Corresponding author
Email addresses: neda_binesh@semnan.ac.ir (Neda Binesh), m.haddadi@semnan.ac.ir (Mahdieh Haddadi)

Received: April 2025 *Accepted:* June 2026

companies can trigger a cascading effect, leading to widespread adoption and increased sales. This targeted approach allows companies to optimize their marketing efforts.

Consequently, determining the significance of nodes and identifying influential nodes within a social network is a crucial challenge in the field of Influence Maximization (IM). The goal of IM is to select a set of "seed nodes" that can maximize the spread of influence across the network. The IM problem is computationally challenging, classified as NP-hard. Researchers have extensively investigated this problem, employing various techniques to address it, as detailed in [19, 31, 32].

Group Influence Maximization (GIM) is a specialized area within the broader domain of influence maximization. The primary objective in GIM is to strategically select an initial set of k active nodes (individuals) within a network. These chosen nodes serve as influencers, with the overarching goal of maximizing the expected number of influenced groups. Unlike traditional influence maximization, which focuses on maximizing individual impact by targeting specific individuals, GIM adopts a broader approach, emphasizing the activation of entire groups rather than isolated individuals [17].

In certain contexts, organizations may prioritize a specific subset within a social network, often referred to as the "target group". This subset could represent users of a particular product, residents of a specific geographic region, individuals with particular political inclinations, or similar categories. Such scenarios introduce a novel challenge in the influence maximization domain: identifying influential nodes to activate the target group. This concept is known as Target Group Influence Maximization (TGIM). In this study, we focus on TGIM, aiming to activate an entire group rather than targeting individuals in isolation.

2 Preliminaries

Table 1 includes the necessary notations and symbols in this paper.

Notations	Meaning
k	Number of spreaders
$G/V/E$	A social network/ set of nodes/ set of links
n/m	Number of nodes/ number of links/ number of candidate nodes
$deg(i)$	degree of node i in undirected graphs and out-degrees in directed graphs
$bn(i)$	number of neighbours of node i whose degrees are greater than or equal to deg(i)
ND	neighborhood-degree index
V'	Set of destination nodes including candidate nodes
$l = \|V'\|$	Number of candidate nodes
H	Target groups: a predefined subset of nodes that we aim to activate
h	size of target groups, $h = \|H\|$
$Q \in \mathbb{R}^{n\times n}$	Q-value matrix in reinforcement learning approach
$R \in \mathbb{R}^{n\times n}$	Reward matrix in reinforcement learning approach
γ	discount factor (future reward weight)
s'/a'	next state /candidate next action
$P \in \mathbb{R}^{h\times l}$	Path matrix including the normalized path score from each target to each destination node
M	Diffusion model
$A_0/A_t/A_T$	Initial spreaders set / Active nodes set in time t / Final active nodes set
R	Number of iterations in Monte Carlo simulation
IS	Influence Spread (average number of final active nodes in R iterations)
$\beta(u, v)$	Probability of information transfer from node u to node v in the IC model
$W(u, v)$	weight of the edge from node u to node v; in unweighted graphs, $W(u, v) = 1$ for all edges.
α	probability of node activation in the IC model.
$\pi = (v_0, v_1, \ldots, v_T)$	Sampled greedy path from a target node to a destination node
$A^*(v)$	Set of greedy successors at node v: $A^*(v) = \arg\max_{a\in V} Q(v, a)$
M	Number of sampled paths (rollouts) for greedy path sampling
$L_{\max}$	Maximum rollout length (steps per rollout) used to avoid non-terminating trajectories
score(π)	Path score defined as $\sum_{t=0}^{T-1} Q(v_t, v_{t+1})$

Table 1: Frequent notations used across the paper.

2.1 Social Networks

A social network is typically modeled as a graph $G = (V, E)$, where the set V represents individuals (nodes), and the set E denotes the edges, signifying social connections or relationships. The neighbors of a node u in this network are the nodes directly connected to u by an edge. The degree of u, denoted by $\deg(u)$, represents the number of edges incident to u. The Neighborhood-degree of u is defined as follows [5]:

$$ND(u) = \frac{\deg(u)}{bn(u) + 1}, \tag{2.1}$$

Remark. The term $bn(u) + 1$ avoids division by zero; in particular, when $bn(u) = 0$, $ND(u) = deg(u)$.

In this definition, $bn(u)$ indicates the number of neighbours of the node u whose degrees are greater than or equal to $\deg(u)$. A higher value of $ND(u)$ indicates that the node u is a more effective spreader. If $bn(u) = 0$, it means that node u is better than its neighbors, and in this case, $ND(u) = \deg(u)$. Consequently, nodes with the highest degrees will be selected as anchor nodes. When two nodes have equal degrees, the one with the least $bn(u)$ value is considered better.

It is worth noting that bn is a vector similar to the vector LI used in the LIR algorithm [24].

2.2 Independent Cascade Diffusion model

In this framework, the method of information dissemination in networks is crucial and has been studied under the title of *dissemination models*. In the general framework of all dissemination models, each user has one of two statuses: passive or active. By starting to publish news, only the primary broadcasting nodes are active, and other nodes are inactive. Throughout the propagation process, active nodes attempt to activate other nodes using various methods.

A random diffusion model (with discrete time steps) for a social network $G = (V, E)$ is the random process of generating the active set A_t per broadcast set of primary influencers A. Dissemination models classify users as either inactive or active, with information released in time steps $t = 0, 1, 2, \ldots, T$. In the context of influence maximization, effective nodes (often referred to as influencers) become active at time $t = 0$ and attempt to activate other previously inactive nodes. The nodes that eventually become active at step t are denoted as A_t.

Diffusion models can be categorized into two main types: **progressive** and **non-progressive** [33]. In progressive models, an active node remains active in the next steps. Conversely, non-progressive models allow activated nodes to return to an inactive state. Progressive models are commonly employed to simulate the acceptance of new technologies or products, such as the adoption of smartphones or the viewership of a new movie. On the other hand, non-progressive models typically capture the spread of ideas and opinions related to events, where nodes may change their state and opinion based on new information from the network. It is noteworthy that many algorithms designed to maximize the impact of these models tend to utilize forward (progressive) models.

Two of the most common diffusion models are **independent cascade diffusion models (IC)** [24, 13, 12] and **linear threshold (LT)** [19], which differ in the way of activating other inactive nodes. Different branches of the influence maximization problem have emerged, with the common goal of determining effective nodes for various purposes. One of the most used models is the Independent Cascade Model (IC).

In the IC model, each user v is independently activated by each of their outgoing neighbours (friends) with a probability of influence $\beta(u, v)$ for the edge (u, v). To determine the probability of propagation and weight the edges of the network, the following equation is utilized:

$$\beta(u, v) = 1 - (1 - \alpha)^{W(u,v)} \tag{2.2}$$

In this equation, α holds significance as it represents a parameter governing the speed of node activation, akin to the news that influences the model.

2.3 Problem definition

Here we explain the **IM** and **TGIM** problems and what should be known about them.

The influence expansion function: In the social network $G = (V, E)$ for a set of users $S \in V$ and the set of network parameters P, it is represented by $\sigma_{G,P}(S)$. Its value is equal to the mathematical expectation of the number of users who are affected by the set S under the parameters of the problem P.

The Influence Maximization (IM) problem: For a social network G, diffusion model M, and a positive number k, the problem of maximizing the influence involves finding the set A_0 containing k initial nodes such that the spread of the influence $\sigma_{G,M}(A_0)$ is maximized. Here, P is the diffusion model M.

The Target Group Influence Maximization (TGIM) problem: For a social network G, diffusion model M, a positive number k, and a target group H, the problem of maximizing the influence involves finding the set A_0 containing k initial nodes such that the spread of the influence $\sigma_{G,M,H}(A_0)$ is maximized.

The problem revolves around maximizing local influence within a network while targeting a specific number of individuals, target group H, rather than the entire network. The goal is to determine a set of broadcasting nodes, keeping their number within a specified limit k, that will optimize the activation percentage of the predefined target group. This problem formulation encapsulates the challenge of strategically selecting nodes to efficiently influence a specific subset of individuals in the network. The proposed approach tries to use reinforcement learning to solve this **TGIM** problem.

2.4 Reinforcement Learning and Q-learning

Reinforcement Learning (RL) is a learning paradigm wherein an agent acquires the skill of making sequential decisions by engaging with an environment. Through its actions, the agent obtains feedback in the form of rewards or penalties. The primary objective of RL is to learn an optimal policy that maximizes cumulative rewards over time. The agent explores the environment through trial and error, takes actions based on its current state, and receives feedback. Utilizing this feedback, the agent updates its policy to enhance decision-making for future actions.

RL algorithms commonly employ functions or value functions to estimate rewards or expected values associated with different states and actions. These estimations empower the agent to learn and refine its decision-making abilities. Through repeated interactions and learning, the agent's policy gradually converges towards an optimal solution, fostering intelligent and adaptive behavior in complex and dynamic environments.

Q-learning is an algorithm employed in reinforcement learning. It operates in a model-free, value-based, and off-policy manner. The primary objective of Q-learning is to identify the optimal sequence of actions based on the current state of an agent. The letter "Q" signifies quality, representing the value of an action in terms of maximizing future rewards. Unlike model-based approaches that rely on transition and reward functions, Q-learning learns from experience without explicitly using such functions. Instead, it trains a value function to assess the relative importance of different states and guide action selection [30].

Through successive iterations, Q-learning updates its estimate of the value of each action in each state, known as the **Q-value**, based on the observed rewards and the transition probabilities. By iteratively refining its Q-values, Q-learning converges towards an optimal policy that dictates the best action to take in each state to maximize long-term rewards.

3 RELATED WORK

In **Influence Maximization**, one typically begins by specifying a predetermined quantity k to represent the number of influencers or spreaders to be chosen. These spreaders can be identified using either local or global methods. Local methods involve selecting each influencer independently, without regard for interactions with other influencers, whereas global methods consider the interplay among all influencers [5]. Noteworthy local approaches include the degree centrality index (Degree) [19, 1, 29], k-shell decomposition [2, 27, 28], semi-local centrality (LC) [7], and the Local Index Rank algorithm (LIR) [24]. Successful global methods encompass the betweenness centrality (BC) index [11], closeness centrality (CC) index [36, 3], eigenvector centrality [10], spreaders inside communities [4, 14, 26, 16, 23, 25], IM based on overlapping community detection algorithms [18, 22], the CR method [34], the generalized closeness centrality index (GCC) [25], and HIPA [20]. These methods offer diverse strategies for pinpointing influential nodes in complex networks, thereby optimizing influence spread across various contexts.

Kempe et al. pioneered the exploration of social influence maximization as a combinatorial optimization problem, delving into its computational intricacies under the LT and IC diffusion models [19]. In their investigations, they assumed the social influence function, denoted as σ, to be both submodular and monotone. The proposed greedy strategy is a key aspect of their approach.

To address scalability challenges, Leskovec et al. introduced a Cost-Effective Lazy Forward (CELF) scheme, leveraging the submodularity property of the social influence function. Their key insight was grounded in the observation that the marginal gain in influence spread for any node in the current iteration is constrained by its marginal gain in

previous iterations. Exploiting this idea allowed them to significantly reduce the number of evaluations of the influence estimation function (σ), leading to a notable improvement in running time [21]. In CELF, each node u in the network keeps track of some information in a table, including its marginal gain in influence for the current seed set, the node with the highest gain encountered so far, and other details.

Chen, Wei, et al. proposed enhancements to the basic greedy algorithm, including techniques to accelerate influence simulations and enhance the accuracy of influence estimates. Their contributions aimed to make influence maximization more scalable and efficient for large-scale social networks [8].

Wu, Yang, et al. focused on maximizing influence in social networks based on group information (GIM). They applied the greedy algorithm in this context, considering group dynamics and information to optimize influence spread. This work expanded the applicability of influence maximization techniques to scenarios involving group behavior [6].

Cordasco and the team put forward a quick and effective method to choose target sets in undirected social networks. While this method gives the best solutions for trees, cycles, and complete graphs, it particularly stands out for its performance in real-life social networks compared to other existing methods [9].

There are other extensions to the influence maximization, for example, an extension to the traditional Influence Maximization (IM) problem was proposed introducing the concept of a supplementary Independent Cascade (IC) model. This model accounts for scenarios where the diffusion probability of one item increases as users adopt supplementary products in advance [35].

One of the approaches that has recently been used in this area is reinforcement learning. For example, in [15] the authors proposed the Q-learning-based DOM (QDOM) framework to address Dynamic Opinion Maximization (DOM) in signed social networks. This framework includes an activated dynamic opinion model based on stateless Q-learning theory for opinion propagation and a Q-learning-based seeding algorithm to identify seed nodes efficiently.

4 PROPOSED METHOD

In this study, the approach involves leveraging reinforcement learning to determine the optimal path from a source node to some destination nodes within a social network. The destinations are some candidate nodes, typically those with high degrees or influential positions. We want to excerpt some of these destinations as influencers (seed nodes). So the final influencers will select from these destination nodes which are capable of effectively reaching and activating a predetermined target group of individuals. In this context, each node in the graph serves as a state, and the objective is to identify the most effective route from the source to the best destination.

Here we use the ND index from equation (1) as the local score. $V^{'}$ is the set of destination nodes which are selected based on the local score (ND). We show the size of $V^{'}$ as l.

In Q-learning approach, we should construct a reward matrix which is shown with R. This matrix contains reward values representing the agent's experience when transitioning from one state to another. A value of 0 signifies the presence of an edge between two nodes, while a value of -1 indicates the absence of edges between states. Notably, a value of 100 denotes an edge connected to the destination node, emphasizing its significance in the learning process.

Utilizing the reward matrix R, a corresponding matrix Q can be constructed, and its values are updated iteratively using the following equation:

$$Q(s,a) = R(s,a) + \gamma \max_{a'} Q(s',a'). \tag{4.1}$$

In equation (4.1), matrix Q contains Q-values, such that $Q(s,a)$ is the value of state s and action a. R is the reward matrix and $R(s,a)$ is the reward value at state s and action a. $\gamma \in [0,1]$ denotes the discount factor (i.e., the weight of future returns). In our experiments, $\gamma = 0.9$. The learning rate is not used in this formulation since Q is updated by a direct Bellman-style backup.

Here, s' is the next state reached from state s after taking action a, and a' is a candidate action available in state s', thus $\max_{a'} Q(s',a')$ denotes the maximum Q-value over all actions in the next state. The formula basically says the value of matrix Q at state s and action a is equal to the sum of the corresponding value in matrix R and the discount factor, multiplied by the maximum value of matrix Q for all possible actions in the next state.

After obtaining the learned action-value matrix Q, we compute, for each target node, a greedy trajectory toward the destination nodes. To this end, we employ the *Greedy Path Selection (GPS)* procedure summarized in Table 2.

Algorithm: Greedy Path Selection (GPS)

Input: matrix Q , Target group H, destination set V', number of rollouts M, maximum path length $L_{\max}$.
Output: Path matrix P.

Initialize $P[u, d] \leftarrow 0$, $\forall u \in H$, $\forall d \in V'$.
for $r = 1$ to M **do**
  for each $u \in H$ **do**
    Set $\pi \leftarrow [u]$, $v \leftarrow u$, and $\ell \leftarrow 0$.
    while $v \notin V'$ and $\ell < L_{\max}$ **do**
      $A^* \leftarrow \{a : Q[v, a] = \max_b Q[v, b]\}$.
      Randomly choose $a \in A^*$ with uniform probability.
      Update $\pi \leftarrow \pi \oplus [a]$, $v \leftarrow a$, $\ell \leftarrow \ell + 1$.
    end while
    if $v \in V'$ **then**
      $d \leftarrow v$.
      if $|\pi| = 1$ **then** $P[u, d] \leftarrow P[u, d] + Q[u, u]$.
      else $P[u, d] \leftarrow P[u, d] + \sum_{t=0}^{|\pi|-2} Q[\pi_t, \pi_{t+1}]$.
    end if
  end for
end for
$P \leftarrow P/M$.
for each $u \in H$ **do**
  $s_u \leftarrow \sum_{d \in V'} P[u, d]$.
  if $s_u > 0$ **then** normalize row u: $P[u, d] \leftarrow P[u, d]/s_u$ for all $d \in V'$.
end for
return P.

Table 2: Greedy path sampling algorithm used to compute the destination score matrix P from the learned Q-value matrix.

For each target node $u \in H$, GPS generates a path by iteratively moving from the current node v to one of its best next-step candidates according to Q. Let

$$A^*(v) \;=\; \arg\max_{a \in V} Q(v, a) \tag{4.2}$$

denote the set of maximizers in row v of Q (i.e., all next nodes achieving the maximum Q-value). If $|A^*(v)| > 1$, GPS resolves the tie by random selection (uniformly) among the candidates in $A^*(v)$, yielding a *greedy path sampling* behavior. The procedure stops when a destination node is reached, i.e., when $v \in V'$.

To guarantee termination, we cap the rollout length by $L_{\max}$ steps. In all experiments, we set $L_{\max} = n$ (number of nodes), which is sufficiently large to avoid premature truncation while preventing infinite trajectories in cyclic graphs. If a rollout terminates without reaching any destination (due to truncation), it does not contribute to the destination scores.

Let $\pi = (v_0, v_1, \ldots, v_T)$ denote a sampled path for target u, where $v_0 = u$ and $v_T \in V'$ is the reached destination. We define the path score as the cumulative sum of Q-values along the path:

$$\text{score}(\pi) \;=\; \sum_{t=0}^{T-1} Q(v_t, v_{t+1}). \tag{4.3}$$

For each target–destination pair (u, d), we aggregate the scores of all rollouts that end at destination d:

$$P(u, d) \;\leftarrow\; P(u, d) + \text{score}(\pi) \quad \text{whenever } v_T = d. \tag{4.4}$$

After running M rollouts, we optionally average by M and then apply row-wise normalization to obtain a distribution over destinations for each target:

$$P(u,d) \leftarrow \frac{P(u,d)}{\sum_{d' \in V'} P(u,d')} \qquad \forall u \in H,\ \forall d \in V', \tag{4.5}$$

whenever the denominator is nonzero. Therefore, P is a row-stochastic matrix: each row corresponds to a target node and sums to one. In particular, $P(i,j)$ represents the *normalized aggregated path score* from the i-th target node to the j-th destination node, estimated via repeated greedy rollouts with random tie-breaking.

Subsequently, a column-wise summation is performed on matrix P to derive the score vector S, where each element represents the aggregate score of a potential destination node. In the final stage, the top-k nodes with the highest values in S are selected as the initial seeds. Table 3 outlines the formal steps of the proposed TGRLGP algorithm, which are summarized as follows:

Algorithm: TGRLGP
Inputs: network G, target group H, number of seeds k
Output: Seed nodes
Phase 1: Compute the neighborhood-degree (ND) of each network node based on equation (2.1).
Phase 2: Select destination nodes from among those with the highest ND scores (e.g., top 10% of nodes).
Phase 3: Construct reward matrix R and run Q-learning approach based on (4.1) to compute matrix Q.
Phase 4: Using the GPS procedure, perform M greedy rollouts per target node with random tie-breaking, accumulate the sampled path scores into the destination matrix P, and normalize each row of P.
Phase 5: Aggregate matrix P column-wise to derive the final score vector S.
Phase 6: Select the k nodes with the highest aggregate values in S as the final seed set.

Table 3: TGRLGP Algorithm

Figure 1 shows the graphic abstract of the proposed method on a sample network with 8 nodes. The destinations are selected based on the ND index and are shown in green color. After running TGRLGP, the score of node 3 is more than 1 to activate the yellow target nodes. Finally, node 3 with red color is selected as the seed node.

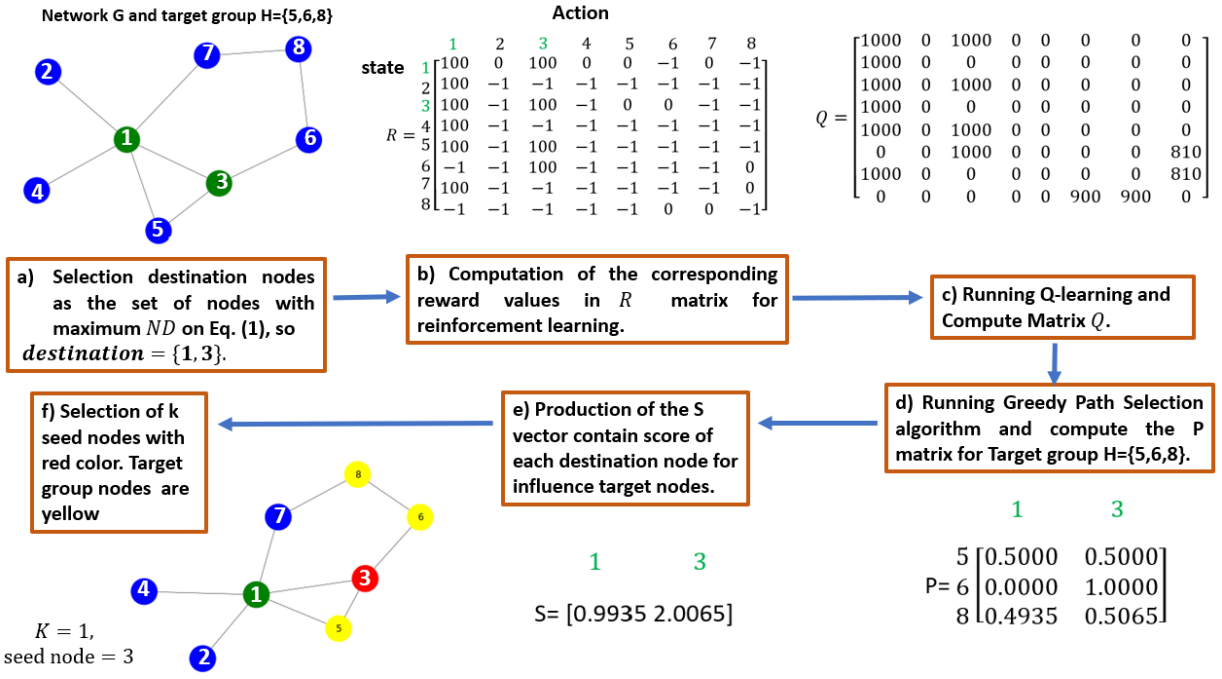

Figure 1: Graphical abstract of **TGRLGP** on a sample network with 8 nodes, target group $H = \{5,6,8\}$, and $k = 1$. The number of destination nodes selected in Phase 2 is 2.

5 EXPERIMENTS AND RESULT

In this section, we compare TGRLGP with some other algorithms on some popular datasets as described in Table 4. To ensure the statistical significance of the results and to achieve convergence in the Greedy Path Selection (GPS) process, the number of rollouts is set to $M = 2000$. Furthermore, the maximum rollout length is set to $L_{\max} = n$, allowing the agent to explore potential paths across the entire network diameter. Other parameters, such as the discount factor γ, are maintained at 0.9 as previously discussed.

Network's Name	Number of Nodes (n)	Number of Edges (m)
Zachary	34	78
Dolphin	62	159
Jazz	198	2742
Football	115	613
Facebook	4039	88234

Table 4: The properties of the used datasets

First, we use the Zachary karate club network with $l = |V'| = 0.15 \times n$, where the set of destination nodes is $V' = \{34, 1, 33, 3, 2\}$. Figure 2 shows this network. In Figure 2, the target group is highlighted in yellow, while the red node denotes the seed selected by TGRLGP. The green node represents the seed identified by the Degree centrality measure. Notably, the Degree-based approach selects node 34, which is overlooked by our method. As illustrated, although node 34 has a high degree, it is topologically distant from the target nodes, rendering it ineffective for localized influence propagation within the target group.

We can see matrix P corresponding to TGRLGP in Table 5. These values show the ideal probabilities of influencing each target with each destination. The corresponding S vector is as follows:

$$S = [0.332, 6.186, 0.344, 1.36, 0.778]$$

and the maximum value in S is 6.18, which belongs to node 1 which is selected by TGRLGP. We can see that this node is close to the target group, while the node with the highest degree is not an ideal choice.

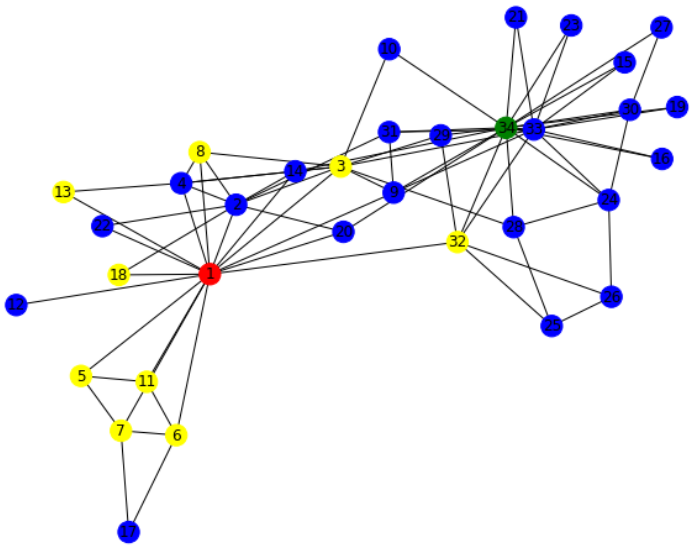

Figure 2: The Zachary Karate Club network highlighting the target group and the selected seed nodes.

Target	Destination				
	34	1	33	3	2
3	0	0	0	1	0
5	0	1	0	0	0
6	0	1	0	0	0
7	0	1	0	0	0
11	0	1	0	0	0
13	0	1	0	0	0
18	0	0.528	0	0	0.472
8	0	0.334	0	0.36	0.306
32	0.332	0.324	0.344	0	0

Table 5: The P matrix for Figure 2

Figure 3 shows the jazz musician network. In this network, we first perform clustering and then select some nodes randomly from one cluster as the target group, shown in yellow.

In this figure, the seed nodes from TGRLGP are colored in red, indicating their optimal placement among the target group nodes. Green nodes represent high-degree nodes selected by the Degree algorithm, which do not correspond well to target group nodes and cannot influence them effectively. The seeds from other algorithms are listed in Table 6.

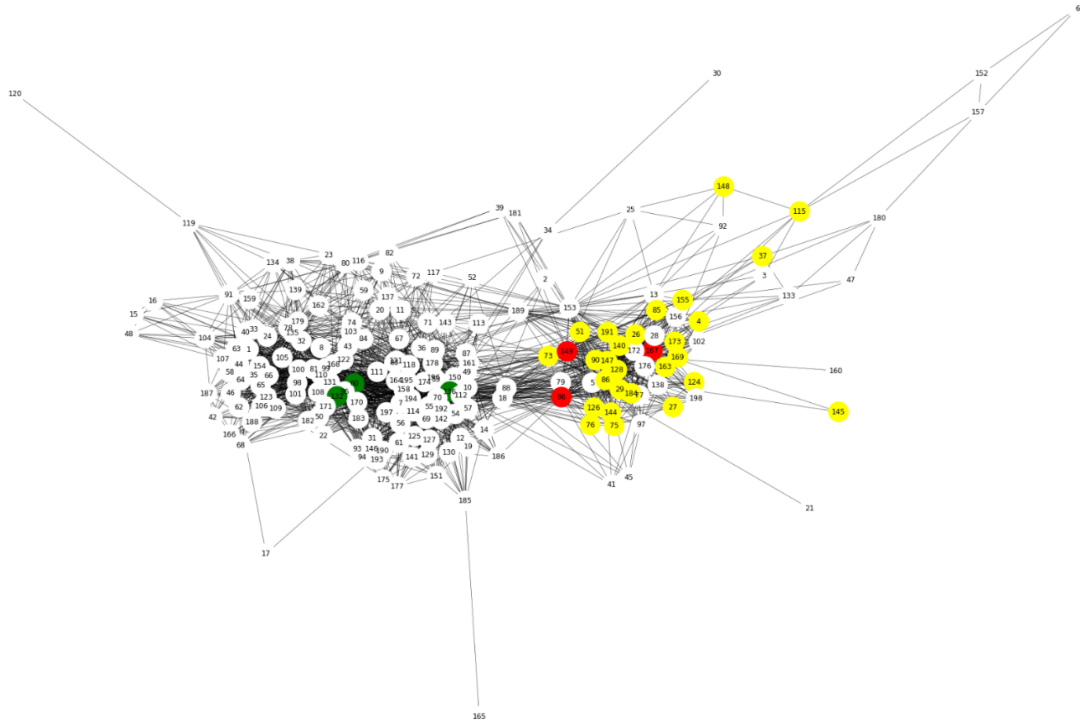

Figure 3: Jazz musician network. Target group nodes are yellow, spreaders of **TGRLGP** are red, and Degree influencers are green.

Beyond these graphical results, we compare the algorithms using the Influence Spread (IS) measure. IS is a well-known metric that counts the final active nodes after a diffusion process. If $A_T(i)$ is the set of final active nodes in

the i-th experiment ($i \in \{1, 2, 3, \ldots, R\}$) under a given diffusion model, then:

$$IS = \frac{\sum_{i=1}^{R} |A_T(i)|}{R}$$

The larger this measure, the better the algorithm's performance. Table 6 shows the IS values for various algorithms such as Degree, LIR, Local Centrality (LC), GCC, HIPA, and the Greedy approach (TGIM) compared with our TGRLGP algorithm. We use the IC diffusion model with a fixed α, as mentioned in the first column. The number of seeds (k) is also specified. The second column reports the target group H and its size ($h = |H|$). The target groups for Zachary and Jazz networks correspond to Figures 2 and 3, respectively. The selected seed nodes by each algorithm are listed, demonstrating that for various target groups, TGRLGP achieves the highest IS score.

Network	**Target Group**	**Algorithm**	**Seeds**	**IS (%)**
Zachary, $\alpha = 0.1$, $k = 1$	3,5,6,7,11, 13,18,8,32 and h=9	Degree	34	40.15
		LIR	34	38.57
		LC	1	56.26
		GCC	6	21.45
		HIPA	34	38.22
		TGRLGP	1	**56.9**
		Greedy (TGIM)	6	21.42
Dolphin $\alpha = 0.1$, $k = 3$	All nodes and h=62	Degree	15, 46, 38	69.49
		LIR	15, 18, 46	81.71
		LC	15, 38, 46	68.59
		GCC	15, 35, 55	74.09
		HIPA	15, 38, 46	69.35
		TGRLGP	15, 18, 21	**85.6**
		Greedy (TGIM)	1, 11, 15	57.96
Jazz, $\alpha = 0.07$, $k = 3$	Various nodes and h=28	Degree	60,132,136	97.2
		LIR	1,2,136	97.32
		LC	60,132,136	97.16
		GCC	54,132,136	97.23
		HIPA	136,60,132	97.14
		TGRLGP	167,149,96	**98.54**
		Greedy (TGIM)	115, 4, 167	97.77
Facebook, $\alpha = 0.01$, $k = 10$	First 800 nodes and h=800	Degree	107,1663,1684,1800,1888,1912,2347,2543,3437,4039	94.04
		LIR	1,2,3,4,5,107,686,1912,3437,3980	95.21
		LC	107,1199,1352,1431,1663,1800,1888,1912,2347,2543	94.28
		GCC	107,483,1663,1800,1888,1912,2347,2543,3437,4039	92.74
		HIPA	107,1684,1912,3437,4039,2543,1888,2347,1800,483	94.24
		TGRLGP	107,686,483,384,414,376,56,713,475,67	**96.91**
		Greedy (TGIM)	1,194,73,48,53,54,119,88,92,126	71.42

Table 6: Comparison of the proposed method (**TGRLGP**) with other algorithms based on IS measure.

6 CONCLUSION

In this paper, we introduce the Reinforcement Learning Approach for the Target Group influence maximization problem. The proposed algorithm (TGRLGP) is designed to identify influential nodes within a target group by leveraging Q-learning methodology, augmented with the optimal path from each target node to the candidate nodes (destinations) which is computed using Greedy Path Selection (GPS) algorithm. Through this innovative approach,

TGRLGP aims to maximize the local coverage of target group nodes, thereby enhancing the potential impact of influence propagation within the network.

To evaluate the effectiveness of TGRLGP, we conducted extensive experiments across four benchmark networks. Additionally, we compared the performance of TGRLGP against six state-of-the-art algorithms commonly used for influence maximization tasks. Our findings consistently demonstrate that TGRLGP achieves superior influence spread in the majority of experimental scenarios. The success of TGRLGP underscores its efficacy in identifying influential nodes within target groups, offering promising prospects for various applications in network analysis and influence maximization strategies.

References

[1] Mohammed Alshahrani, Zhu Fuxi, Ahmed Sameh, Soufiana Mekouar, and Sheng Huang, *Efficient algorithms based on centrality measures for identification of top-k influential users in social networks*, Information Sciences **527** (2020), 88–107.

[2] Frank Schweitzer Antonios Garas and Shlomo Havlin, *A k-shell decomposition method for weighted networks*, New Journal of physics **14** (2012), 083030.

[3] Niloofar Arazkhani, Mohammad Reza Meybodi, and Alireza Rezvanian, *An efficient algorithm for influence blocking maximization based on community detection*, 2019 5th international conference on web research (ICWR), IEEE, 2019, pp. 258–263.

[4] Esmaeil Bagheri, Gholamhossein Dastghaibyfard, and Ali Hamzeh, *Fsim: A fast and scalable influence maximization algorithm based on community detection*, International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems **26** (2018), no. 03, 379–396.

[5] Neda Binesh and Mehdi Ghatee, *Distance-aware optimization model for influential nodes identification in social networks with independent cascade diffusion*, Information Sciences **581** (2021), 88–105.

[6] Neda Binesh, Mahdieh Haddadi, and Niloofar Arazkhani, *Target group influence maximization using reinforcement learning approach*, 2024 10th International Conference on Web Research (ICWR), IEEE, 2024, pp. 130–136.

[7] Duanbing Chen, Linyuan Lü, Ming-Sheng Shang, Yi-Cheng Zhang, and Tao Zhou, *Identifying influential nodes in complex networks*, Physica a: Statistical mechanics and its applications **391** (2012), no. 4, 1777–1787.

[8] Wei Chen, Yifei Yuan, and Li Zhang, *Scalable influence maximization in social networks under the linear threshold model*, 2010 IEEE international conference on data mining, IEEE, 2010, pp. 88–97.

[9] Gennaro Cordasco, Luisa Gargano, Marco Mecchia, Adele A Rescigno, and Ugo Vaccaro, *Discovering small target sets in social networks: a fast and effective algorithm*, Algorithmica **80** (2018), 1804–1833.

[10] Paramita Dey, Agneet Chaterjee, and Sarbani Roy, *Influence maximization in online social network using different centrality measures as seed node of information propagation*, Sādhanā **44** (2019), 1–13.

[11] L. C. Freeman, *A set of measures of centrality based on betweenness.*, Sociometry **40** (2012), 35–41.

[12] Jacob Goldenberg, Barak Libai, and Eitan Muller, *Talk of the network: A complex systems look at the underlying process of word-of-mouth*, Marketing Letters **12** (2001), 211–223.

[13] Jacob Goldenberg, Barak Libai, and Eitan Muller, *Using complex systems analysis to advance marketing theory development: Modeling heterogeneity effects on new product growth through stochastic cellular automata*, Academy of Marketing Science Review **9** (2001), no. 3, 1–18.

[14] Maoguo Gong, Chao Song, Chao Duan, Lijia Ma, and Bo Shen, *An efficient memetic algorithm for influence maximization in social networks*, IEEE Computational Intelligence Magazine **11** (2016), no. 3, 22–33.

[15] Qiang He, Yingjie Lv, Xingwei Wang, Jianhua Li, Min Huang, Lianbo Ma, and Yuliang Cai, *Reinforcement-learning-based dynamic opinion maximization framework in signed social networks*, IEEE Transactions on Cognitive and Developmental Systems **15** (2022), no. 1, 54–64.

[16] Masoud Jalayer, Morvarid Azheian, and Mehrdad Agha Mohammad Ali Kermani, *A hybrid algorithm based on community detection and multi attribute decision making for influence maximization*, Computers & Industrial Engineering **120** (2018), 234–250.

[17] Smita Ghosh Jianming Zhu and Weili Wu, *Group influence maximization problem in social networks*, IEEE Transactions on Computational Social Systems **6** (2019), 1156–1164.

[18] Hamed Kalantari, Mehdi Ghazanfari, Mohammad Fathian, and Kamran Shahanaghi, *Multi-objective optimization model in a heterogeneous weighted network through key nodes identification in overlapping communities*, Computers & Industrial Engineering **144** (2020), 106413.

[19] David Kempe, Jon Kleinberg, and Éva Tardos, *Maximizing the spread of influence through a social network*, Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining, 2003, pp. 137–146.

[20] Sahar Kianian and Mehran Rostamnia, *An efficient path-based approach for influence maximization in social networks*, Expert Systems with Applications **167** (2021), 114168.

[21] Jure Leskovec, Andreas Krause, Carlos Guestrin, Christos Faloutsos, Jeanne VanBriesen, and Natalie Glance, *Cost-effective outbreak detection in networks*, Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining, 2007, pp. 420–429.

[22] Weimin Li, Kexin Zhong, Jianjia Wang, and Dehua Chen, *A dynamic algorithm based on cohesive entropy for influence maximization in social networks*, Expert Systems with Applications **169** (2021), 114207.

[23] Weimin Li, Heng Zhu, Shaohua Li, Hao Wang, Hongning Dai, Can Wang, and Qun Jin, *Evolutionary community discovery in dynamic social networks via resistance distance*, Expert Systems with Applications **171** (2021), 114536.

[24] Dong Liu, Yun Jing, Jing Zhao, Wenjun Wang, and Guojie Song, *A fast and efficient algorithm for mining top-k nodes in complex networks*, Scientific reports **7** (2017), no. 1, 43330.

[25] Huan-Li Liu, Chuang Ma, Bing-Bing Xiang, Ming Tang, and Hai-Feng Zhang, *Identifying multiple influential spreaders based on generalized closeness centrality*, Physica A: Statistical Mechanics and its Applications **492** (2018), 2237–2248.

[26] Wei-Xue Lu, Chuan Zhou, and Jia Wu, *Big social network influence maximization via recursively estimating influence spread*, Knowledge-Based Systems **113** (2016), 143–154.

[27] Giridhar Maji, *Influential spreaders identification in complex networks with potential edge weight based k-shell degree neighborhood method*, Journal of Computational Science **39** (2020), 101055.

[28] Giridhar Maji, Sharmistha Mandal, and Soumya Sen, *A systematic survey on influential spreaders identification in complex networks with a focus on k-shell based techniques*, Expert Systems with Applications **161** (2020), 113681.

[29] Mohammad Reza Meybodi Niloofar Arazkhani and Alireza Rezvanian, *Influence blocking maximization in social network using centrality measures*, Proceedings of the Conference on Knowledge-based Engineering and Innovation, 2019.

[30] P. Watkins, C.J.C.H.& Dayan, *Q-learning*, Machin Learning **8** (1992), 279—292.

[31] Laks V. S. Lakshmanan Wei Chen, Carlos Castillo, *Information and influence propagation in social networks*, Springer, 2013.

[32] Wenguo Yang, Yapu Zhang, and Ding-Zhu Du, *Influence maximization problem: properties and algorithms*, Journal of Combinatorial Optimization **40** (2020), no. 4, 907–928.

[33] Dongxiang Zhang Yuchen Li and Kian-Lee Tan, *Real-time targeted influence maximization for online advertisements*, Proceedings of the VLDB Endowment, 2015, pp. 1070–1081.

[34] Dayong Zhang, Yang Wang, and Zhaoxin Zhang, *Identifying and quantifying potential super-spreaders in social networks*, Scientific reports **9** (2019), no. 1, 14811.

[35] Yapu Zhang, Jianxiong Guo, Wenguo Yang, and Weili Wu, *Supplementary influence maximization problem in social networks*, IEEE Transactions on Computational Social Systems **11** (2023), no. 1, 986–996.

[36] Zhiying Zhao, Xiaofan Wang, Wei Zhang, and Zhiliang Zhu, *A community-based approach to identifying influential spreaders*, Entropy **17** (2015), no. 4, 2228–2252.