Kurdistan Journal of Applied Research (KJAR)
Print-ISSN: 2411-7684 | Electronic-ISSN: 2411-7706

Original Article

Balancing Speed and Accuracy in Influence Maximization: A Reinforcement Learning Solution

Khabat Atar [a], **Amir Sheikhahmadi** [b], **Keyhan Khamforoosh** [c*]

[a] Department of Science Computer, Faculty of Health and Science, Koya University, Koya, Iraq.
[b] Department of Engineering, RMIT University, Melbourne, Australia.
[c] Department of Computer Engineering, Islamic Azad University, Sanandaj, Iran.

Submitted: 29 May 2025
Revised: 23 July 2025
Accepted: 1 September 2025

*** Corresponding Author**:
k.khamforoosh@iau.ac.ir

Keywords: Influence maximization, Q-learning, Genetic algorithm, Weighted networks, independent cascade model.

How to cite this paper: K. Atar, A. Shekhahmadi, K. Khamforoosh, "Balancing Speed and Accuracy in Influence Maximization: A Reinforcement Learning Solution", KJAR, vol. 10, no. 2, pp: 250 -265 Dec 2025, doi: 10.24017/science.2025.2.16

Abstract: Influence maximization involves selecting an optimal subset of nodes within a graph to activate as many nodes as possible in a network. This approach is categorized as non-polynomial time, and no specific algorithm is currently available to run efficiently within a reasonable time frame, especially for large-scale networks. Numerous methods have been introduced to resolve this challenge, including greedy algorithms, structural heuristics, and metaheuristic approaches. Although greedy algorithms and their improved versions achieve high accuracy, they often suffer from poor scalability and slow execution times on large graphs. In contrast, structural methods offer faster computation but at the cost of reduced accuracy. Metaheuristic algorithms, while promising, face difficulties in balancing speed and accuracy due to the expansive search space inherent in complex social networks. This study introduces a novel method that leverages Q-learning, a reinforcement learning technique, to optimize influence maximization. The proposed method narrows down the search space by focusing on high-degree influential nodes. It dynamically updates the Q-table by assigning rewards and penalties based on the nodes' impact during influence propagation, modeled using the Independent Cascade framework. This approach effectively balances exploration and exploitation, enabling the identification of a highly influential seed set with improved efficiency. Experiments conducted on various real-world datasets show that the Q-learning-based method significantly reduces execution time compared to genetic, particle swarm optimization, random, degree centrality, and K-shell algorithms while achieving higher influence spread in most cases. These results underscore the promise of reinforcement learning techniques in addressing complex network optimization problems such as influence maximization.

1. Introduction

A complicated network is a network that exhibits irregular, non-trivial topological characteristics and is frequently encountered in real-world systems [1]. These networks exhibit complex connection structures between their parts that cannot be readily understood using simple mathematical models. Different types of real-life networks, such as biological interactions, information diffusion, and social dynamics, are complex networks. The main challenge in these networks is identifying the most influential nodes, which play a key role in applications such as viral marketing, epidemic control, and opinion propagation. This challenge, known as influence maximization (IM), involves selecting a subset of nodes that can maximize the spread of information under a given diffusion model. However, IM is NP-hard, making exact solutions computationally infeasible for large-scale networks [2].

Greedy algorithms offer high accuracy but are computationally expensive and not scalable to large networks. Structural heuristics, such as degree centrality and k-shell decomposition, are efficient but often lack precision. Metaheuristic methods such as particle swarm optimization (PSO) attempt to balance speed and accuracy, yet they struggle with vast search spaces in complex networks. These limitations underscore the need for a more adaptive, scalable, and learning-based solution—one that can overcome the uncertainty and dynamic nature of influence spread more effectively [1, 2].

To overcome these challenges, we propose a reinforcement learning-based solution using Q-learning. This method enables the agent to learn an optimal node election strategy by interacting with the environment, receiving feedback through the simulation of the independent cascade (IC) model. Unlike static heuristics, our approach can adapt over time and make informed, step-by-step decisions that maximize the long-term spread of influence.

Q-learning is especially suitable for the IM challenge for a number of reasons. First, it is model-free, meaning it can operate without a predefined mathematical model of the environment, which is ideal for influence spread, which is often uncertain. Second, it supports delayed rewards, aligning with the nature of IM, where the impact of a node may only become evident after several steps. Third, Q-learning allows for reward-driven learning, where the effectiveness of each node can be evaluated based on actual diffusion outcomes. This enables the algorithm to refine its decisions over time. Finally, Q-learning balances exploration and exploitation, helping discover hidden combinations of influential nodes that simpler algorithms might overlook.

We validated our approach using four real-world datasets representing diverse network types: an academic genealogy network of computer science PhDs, a scientific collaboration network, a semantic network from Roget's Thesaurus, and a protein-protein interaction network. Our method was compared against widely used baselines—random selection, degree centrality, k-shell, and PSO—based on influence spread and execution time. The contributions of this paper are as follows:

- We introduced a novel approach relying on Q-learning to solve the influence maximization problem by optimizing the selection of influential nodes in social networks. This approach outperforms traditional algorithms by delivering faster execution and higher accuracy, making it a more efficient and effective solution.
- We reduced the search space by preprocessing the data to focus on high-degree nodes, which are more influential. This approach improves the overall efficiency of the algorithm.
- The Q-table is dynamically updated using rewards and penalties based on each node's influence in the diffusion process. In contrast, the IC model calculates the influence to adjust rewards in the learning algorithm.
- The proposed method effectively balances exploration and exploitation, achieving a balanced trade-off between execution speed and accuracy in influence spread.
- A comprehensive evaluation on multiple real-world datasets shows that the Q-learning method generally outperforms genetic algorithms, PSO, degree-based, and K-shell methods in both speed and effectiveness.
- A sensitivity analysis on the activation threshold parameter identifies 0.1 as the optimal value for maximizing the algorithm's performance.

The remaining parts of this paper will include a review of recent advancements in influence maximization in social networks, as discussed in section 2. Section 3 outlines the method we developed using Q-learning. Section 4 introduces the datasets used in the paper, and section 5 examines the results and comparisons. Section 6 discusses our method in comparison to current algorithms and details its benefits. Ultimately, section 7 includes the summary and potential future projects.

2. Related Works

Social networks represent the relationships among individuals that enable the flow of information. Compared to traditional information dissemination methods, social networks offer a faster and more scalable environment for communication, which has led to the rise of viral marketing [3, 4].

Influence Maximization (IM) is the problem of selecting a small subset of nodes in a network to maximize the spread of influence under a given propagation model. Two commonly used models in this domain are the IC and the linear threshold (LT) models. Kempe *et al.* [5] first demonstrated that IM under these models is NP-hard and proposed a basic greedy algorithm, the kernelized greedy algorithm (KGA), to approximate a solution. Chen *et al.* [6] later improved this with the cost-effective lazy forward (CELF) optimization, which significantly reduces computation time while maintaining accuracy. Despite these advances, greedy algorithms are still computationally expensive for large-scale networks due to the repeated influence spread simulations. In one study, a greedy algorithm is proposed to solve the cost distribution under budget (CDB) problem in continuous influence maximization in social networks [7]. In each iteration, the algorithm calculates the influence increment of each node. It updates the cost distribution accordingly to find the optimal allocation that maximizes total influence within a given budget. Experimental results show that this method significantly improves expected influence spread.

Metaheuristic algorithms, including genetic algorithms and PSO, offer a balance between speed and performance by more intelligently exploring the solution space than pure heuristics. However, they struggle with scalability as the network size and solution space grow [8].

To address this limitation, heuristic methods were introduced. These methods, such as degree centrality, k-shell decomposition, and degree discount, aim to approximate node influence using topological features [9-11]. While they are significantly faster, they generally sacrifice accuracy, especially in networks with complex structures or high clustering. For example, the community-based influence maximization algorithm – heuristic (CMIA-H) and the campaign-oblivious independent cascade model – out-arborescence (CMIA-O) are heuristic techniques that enhance scalability by focusing on positive seed selection [12].

Pruning-based techniques have also emerged, aiming to modify the network structure by removing or adding edges to control the flow of influence. These methods are still in early development but show promise for targeted influence control [12].

A more recent line of research focuses on fair and balanced influence maximization. Traditional IM approaches may result in biased outcomes, where only specific communities or user groups are exposed to information. Garimella *et al.* [13] and Becker *et al.* [14] introduced the concept of fair influence maximization (FIM), proposing algorithms to ensure equitable information exposure. Subsequent studies have proposed models such as CEA-FIM and BIM-DRL, which incorporate community structure and deep reinforcement learning to achieve fairness while maintaining efficiency [15, 16].

Competitive opinion maximization seeks to select influential seed nodes in social networks to spread favorable opinions in the presence of competitors. This paper introduces a Q-learning-based framework (QOMF) that models dynamic opinion changes and unknown competitor strategies using a multistage seeding process. Experiments show that QOMF significantly outperforms existing methods in maximizing relative effective opinions [17]. With the advent of deep learning and graph representation techniques, reinforcement learning (RL) has emerged as a potent method for IM. RL-based models such as S2V-DQN and graph combinatorial optimization (GCOMB) utilize graph embeddings to represent node features and learn optimal seed selection strategies through trial and error [18]. These methods can adapt to dynamic network changes and model uncertainty in influence propagation. CoreQ, for example, addresses the scalability challenge by using a k-core hierarchy to limit the search space, while learning influence overlap and structure-aware policies [19].

Another RL-based approach introduces new metrics such as connectivity strength and effective distance, which analyze both local and global structural contributions. This method aggregates multi-hop neighborhood information and significantly improves the accuracy of identifying influential nodes [20].

Finally, online influence maximization methods have emerged to deal with evolving social networks. These techniques estimate edge activation probabilities in real-time and dynamically update seed node selection strategies using feedback from the environment. Deep RL (DRL) is beneficial in this context due to its ability to learn optimal policies without requiring complete knowledge of network dynamics [21]. In one study, the authors introduced a framework called budgeted influence maximization using deep reinforcement learning (BIM-DRL), which leverages deep reinforcement learning to address the challenge of balanced influence maximization in social networks [22]. The goal of this approach is to prevent the formation of filter bubbles and echo chambers through balanced seed node selection and consideration of entity correlations. The novelty of another study lies in proposing a DRL-based model with transfer learning to address competitive influence maximization on unknown social networks by jointly learning when to explore the network and how to select influential seed nodes optimally [23]. The authors of another work proposed a novel propagation model and designed a new algorithm that captures these dynamics, outperforming existing methods in both performance, time complexity [24], and memory [25],

One study introduced a dual coupled graph neural network for seed selection in social networks, which intelligently extracts node information by combining deep reinforcement learning and graph neural networks, and selects the optimal strategy to maximize the influence spread. Another proposed method involves a new model called novel influence network embedding (NINE) for network representation, as well as the NINE influence maximization algorithm for seed selection, which accurately models the diffusion behavior and social influence of nodes [26]. The innovation introduced in another article includes a voting-based influence maximization method where nodes with different degrees have distinct voting powers, and the influence of 2-hop neighbors is discounted to reduce overlap. It also enhances efficiency by only updating the voting scores of nodes whose scores may change, avoiding unnecessary calculations [27].

The novelty of this paper lies in introducing a cross-layer IC model to capture inter-layer information propagation in multilayer social networks. It also proposes an algorithm that integrates differential evolution with deep reinforcement learning and multilayer network embedding to identify the optimal seed set more effectively [28].

In summary, while classical algorithms provide foundational techniques for influence maximization, recent developments in reinforcement learning and fairness-aware models have expanded the capabilities of IM methods to handle large, complex, and dynamic. Table 1 briefly presents the existing studies in this field, the problems addressed, and the proposed solutions.

Table 1: Summary of influence maximization research.

Paper No.	Topic	Method / Model	Summary
[3]	Viral marketing	Social networks	Faster and more scalable medium for information dissemination compared to traditional methods.
[5]	Influence maximization (IM)	IC & LT models + greedy algorithm (KGA)	IM is NP-hard; a basic greedy algorithm was proposed to approximate the solution.
[6]	Improved greedy algorithm	CELF	Significantly reduces computation time while maintaining accuracy.
[7]	Continuous influence maximization	Greedy algorithm for CDB	Optimizes budget allocation by computing influence increment per iteration.
[8, 9, 10]	Heuristic methods	Degree centrality, K-shell, degree discount	Faster but less accurate in complex or highly clustered networks.
[11]	Metaheuristic algorithms	Genetic algorithms, PSO	Balance speed and performance, but face scalability issues.
[12]	Pruning & scalable techniques	CMIA-H, CMIA-O	Control influence flow by modifying network structure; focus on positive seed selection.
[13, 14]	Fair influence maximization (FIM)	Fair algorithms	Ensure equitable exposure of information across communities.
[15, 16]	Advanced FIM models	CEA-FIM, BIM-DRL	Incorporate community structure and deep RL to ensure fairness and efficiency.
[17]	Competitive opinion maximization	QOMF (Q-learning)	Model dynamic opinions and competitor strategies using multi-stage seeding.
[18]	Reinforcement learning in IM	S2V-DQN, GCOMB	Use graph embeddings and RL to learn optimal seed selection strategies.
[19]	Scalability in RL	CoreQ	Uses the K-core hierarchy to limit the search space and learn efficient policies.

Table 1: continue

[20]	Accuracy in RL	IC-SNI	Introduces connectivity strength and effective distance metrics for better node identification.
[21]	Online influence maximization	Deep RL	Learns optimal seed strategies in dynamic networks using real-time feedback.
[22]	Influence maximization	Deep RL	Preventing the formation of filter bubbles and echo chambers through balanced seed node selection and consideration of entity correlations.
[23]	Influence maximization	Deep RL	Proposing a DRL-based model with transfer learning to address competitive influence maximization on social networks
[24]	Influence maximization	Deep RL	Proposing a novel propagation model and designing a new algorithm that captures these dynamics, outperforming existing methods in both performance and time complexity.
[25]	Influence maximization	Deep RL	Proposing methods, including a new model called Novel Influence Network Embedding for network representation and the NINE Influence Maximization algorithm for seed selection
[26]	Influence maximization	Deep RL	Introducing a Dual Coupled Graph Neural Network (DGN) for seed selection in social networks
[27]	Influence maximization	Voting-based influence maximization	Voting-based influence maximization method where nodes with different degrees have distinct voting powers, and the influence of 2-hop neighbors is discounted to reduce overlap

3. Material and Methods

In the problem of maximizing influence, the search space is extensive, as the goal is to select the optimal subset of network nodes to initiate the spread process. Considering all network nodes in the search space would slow down the algorithm and reduce its accuracy; a few nodes are connected to multiple others, whereas a small portion of nodes with high centrality have connections to many others and can greatly impact information spread in the network [29]. Given these characteristics, the proposed method preprocesses the problem space by narrowing it down to a reduced group of highly impactful nodes, where the optimization occurs. Due to the intricate structure of these networks, the Q-learning algorithm is employed to address the issue of influence spread optimization. Figure 1 shows the flowchart of the proposed method.

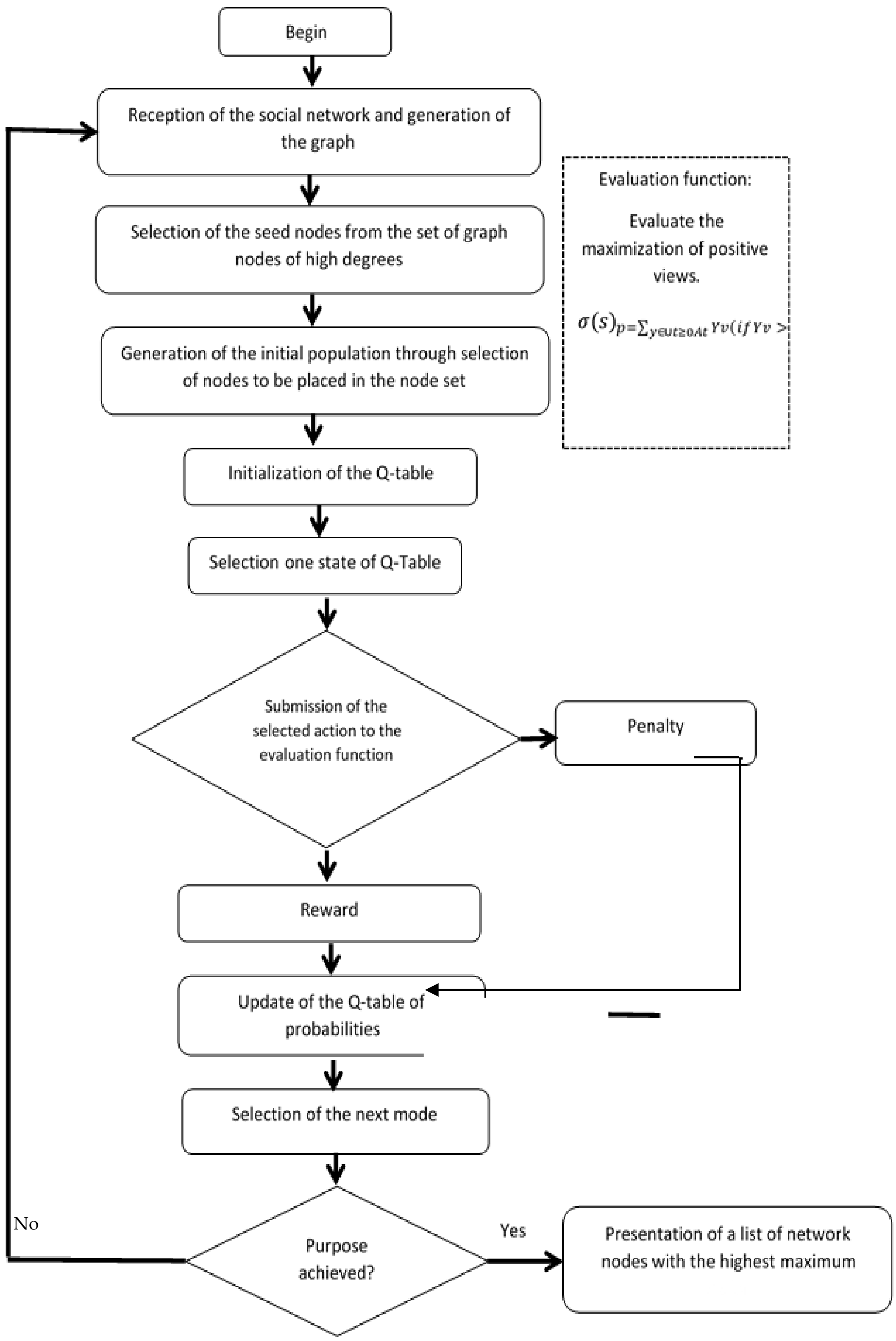

Figure 1: The flowchart of the proposed approach.

3.1. Formulation of the Problem

The social network graph input is a directed graph represented as G = (V, E), with V being the network entities set, and E representing the directed edges showing relationships between the entities. A directed edge $e_{ij} \in E$ going from node vi to node v_j indicates that v_i has an influence on v_j. According to Equation 1, the goal is to find a subset S of network nodes that can activate the most nodes using the IC propagation model [12].

$$\sigma(S^*) = \max_{S \subseteq V}\{\sigma(S)\} \quad (1)$$

The IC propagation model is employed for the function σ(.), assigning a nonnegative real number to each subset S to show its propagation potential [30]. In the linear threshold model, the algorithm computes the spread of influence in a social network as follows:

Inputs: The graph G (V, E, W), where V is the set of nodes, E is the set of edges, and W represents edge weights indicating the influence between nodes. Additionally, the seed set S and the thresholds θ_vFor each node, the information is provided.

- Initially, set k=0, and the initial set of influenced nodes $\phi(t_0)$ is assigned to the seed set S. In this step, the seed nodes are influenced by default.
- The algorithm then enters an iterative loop, continuing until the influenced set $\phi(t_k)$ is no longer empty.
- In each iteration, k is incremented by 1, and the set $\phi(t_k)$ is reset to an empty set.
- The algorithm proceeds by considering all nodes v not in $\phi(t_{k-1})$. For each node v, it computes the fraction of its neighbors that are influenced (i.e., belong to $\phi(t_{k-1})$).
- The fraction of influenced neighbors is computed by counting the number of neighbors of node v that have already been influenced, and dividing that by the total number of neighbors. If this fraction exceeds the threshold θ_v of node v, then node v is added to the set of influenced nodes for this iteration, $\phi(t_k)$.
- This process continues for all nodes, updating the set of influenced nodes in each iteration.
- Once the process terminates, the total spread of influence is computed as $\boldsymbol{\phi(t_k) = \cup_{j=0}^{k} \phi(t_j)}$, representing the union of all influenced nodes across iterations.
- The final output is the size of the influence spread $|\phi(k)|$, which gives the total number of influenced nodes.

3.2. Computation of the Q-table

The issue is resolved by utilizing the Q-learning algorithm for the method being suggested. To achieve this goal, the issue of IM is transformed into one that can be addressed with this algorithm. RL represents a machine learning technique where an agent gains experience by engaging with its surroundings, receiving feedback through trial and error. In supervised learning and reinforcement learning, there is an emphasis on linking input and output variables. Nevertheless, in reinforcement learning, rewards and punishments serve as indicators of good and bad conduct, as they guide the agent towards the appropriate actions needed to carry out a task, in contrast to supervised learning. Q-learning is a reinforcement learning method that aims to determine a specific policy by acquiring knowledge of an action-value function, thereby adapting actions according to varying circumstances. One advantage of this approach is that the function mentioned earlier can be acquired without requiring a predetermined environment model.

One significant alteration in Q-learning when applied to tackle the influence maximization issue relates to the manner in which the Q-table is generated and its quality. In this table, the options chosen by the core members determine the content of the columns, while the rows represent the various states. Each cell contains the highest anticipated future reward for the specific state and action. For instance, a group of nodes (81, 17, 19, 1, 8) with elevated degrees is given as the starting choices. Next, the selection of potential members for the starting core determines the preferred solutions. Figure 2 displays an example of the agent search technique when analyzing various states to choose the initial core members.

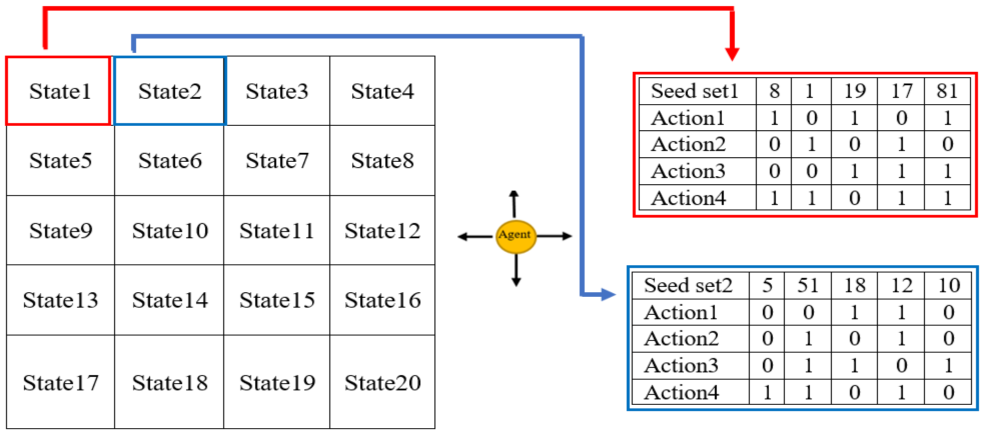

Figure 2: Agent search method in the examined states (various seed set choices).

Every score in the Q-table represents the anticipated reward the agent will receive when following the best policy and taking a specific action. The Q-table functions similarly to a poker game, where the agent determines the optimal action by looking at the highest score in each row corresponding to each state. Afterwards, the calculation of the value for every Q-table element is performed. The Q-function, also called the action-value function, receives the state and action as input and provides the expected future reward for the given state and action (according to the Bellman equation).

$$\underbrace{\mathbf{NewQ(s,a)}}_{\text{New Q-Value for state and action}} = \underbrace{\mathbf{Q(s,a)}}_{\text{Current Q-value}} + \underbrace{\alpha}_{\text{Learning Rate}}[\underbrace{\mathbf{R(s,a)}}_{\text{Reward for action}} + \underbrace{\gamma}_{\text{Discount Rate}}\underbrace{\mathbf{maxQ'(S',a')}}_{\text{Minimum predicted reward regard to new s' and all possible actions in state}} - \mathbf{Q(s,a)}] \quad \mathbf{(2)}$$

In the proposed method, each state (selected initial core) is sent to the evaluation function for each action chosen, and the propagation values are computed by the evaluation function, i.e., the IC propagation model. Throughout this study, the calculated values of positive influence spread are used as the evaluation function.

3.3. *Detailed Steps of the Proposed Method*

- The problem input involves a social network G = (V, E, W), where V is the set of nodes, E is the set of edges, and W is the set of edge weights, which represent the influence of each node on the others. The weights are numbers between 0 and 1, with a larger number indicating greater influence.
- Given the large size of the graph of the problem, preprocessing takes place in this step. For that purpose, nodes of higher degrees are selected from among the graph nodes as candidates for membership in the initial core or seed set S.
- The initial population is generated, where nodes are selected from among those of degrees higher than or equal to the threshold, to be placed in the initial core or seed set S. For the generation of the initial population, a Boolean vector $\boldsymbol{B} = (\boldsymbol{b_1}, \cdots, \boldsymbol{b_N}),\ \boldsymbol{b_i} \in \{\mathbf{0}, \mathbf{1}\} : \boldsymbol{b_i} = \mathbf{1} \leftrightarrow \boldsymbol{v_i} \in \boldsymbol{S}$ is used to decide on the selection of the nodes for membership in the initial core.
- The Q-table is initialized, where different states of seed set member selection specify the selected solutions. The presence or absence of the nodes in the initial core makes up the columns of the Q-table, and the seed sets (initial cores) constitute the rows.
- A state in the Q-table is selected as the current state.
- An action a is selected in the current state.

- Experience is gained as the action selected by each state (seed set chosen) is sent to the evaluation function, by which propagation is computed (observation of the output state s and reward).
- The Q-table is subsequently updated with the new reward and penalty values using the Bellman equation to update Q (s, r). If the optimal solution is identified, the algorithm proceeds to the final step (End); otherwise, it returns to step 3.
- End

3.4. Datasets

To evaluate the proposed method, four publicly available datasets consisting of directed weighted graphs were selected. These datasets are extracted from real-world data and have been extensively used in prior research. The datasets include:

- **CSphd-genealogy**: genealogical data of computer science PhDs.
- **Netscience-collaboration**: a network of scientists collaborating on network theory.
- **Roget-dictionary**: derived from Roget.net, based on the roget.dat file from the Stanford GraphBase, containing cross-references found in Roget's Thesaurus.
- **YST**: a protein-protein interaction network constructed by V. Batagelj.

A summary of the key statistics of these datasets is shown in table 2.

Table 2: Dataset characteristics used for evaluation.

Dataset	Number of Nodes	Number of Edges	Average Degree	Max Degree	Min Degree
CSphd-genealogy	1882	1740	1.85	46	1
Netscience-collaboration	1461	2742	3.75	34	1
Roget-dictionary	1010	3649	7.23	28	1
YST	2361	7182	6.08	66	1

4. Results

The proposed approach is evaluated using the following criteria:

- **Influence Spread:** The node count activated by the selected seed set members. Higher values indicate better performance.
- **Time Efficiency:** The total runtime from the start of seed selection to the completion of propagation. This reflects how quickly the method selects the seed set.
- **Sensitivity Analysis:** The activation threshold was varied to assess robustness. The optimal threshold value was found to be 0.1.
- **Comparison with Other Methods:** The method was tested against cutting-edge solution algorithms, including community-based shell decomposition , centrality-based methods, moment of influence and recommendation random node, discrete particle swarm optimization , and dynamic generalized genetic algorithms .

As shown in figures 3 and 4, in the YST.csv dataset, which represents a complex network with an asymmetric node structure, AI-based methods such as Q-Learning and PSO have demonstrated significantly better performance in identifying key nodes for influence propagation, activating more than 750 nodes. In contrast, simpler methods, such as random or K-Shell, activated fewer than 100 nodes. Although PSO achieved the highest activation count, its relatively long execution time makes it less efficient compared to Q-Learning, which required minimal processing time. Therefore, considering the properties of the network and the level of influence spread, Q-Learning is the most suitable choice, as it provides performance close to PSO with a much lower execution time, indicating its effectiveness in accurately identifying strategically essential nodes.

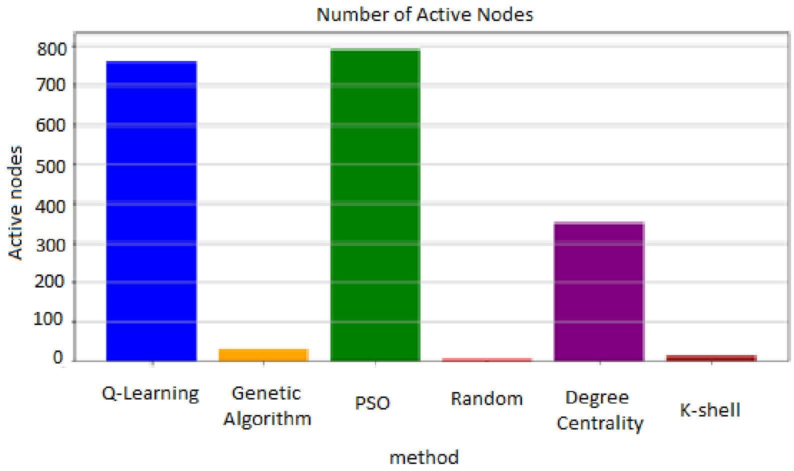

Figure 3: Comparison of active nodes with the YST dataset and different algorithm.

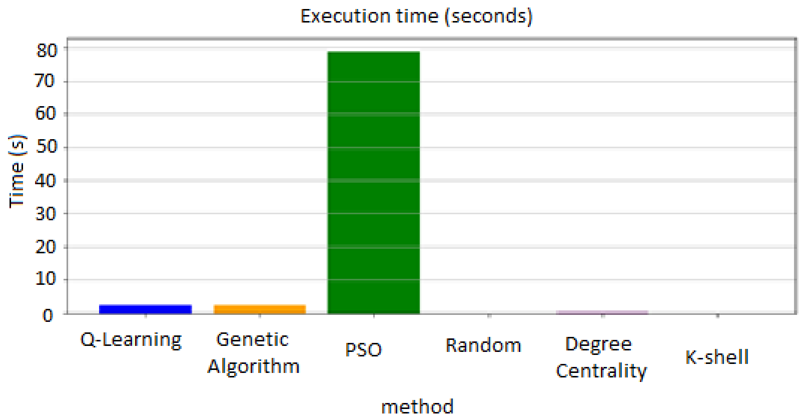

Figure 4: Comparison of execution time with the YST dataset and different algorithm.

In the Roget.csv dataset, various methods, including Q-Learning, Genetic Algorithm, PSO, and random activation, produced a similar number of active nodes, approximately 600. This suggests that the network structure may permit significant influence from numerous seed selections. However, PSO, despite having the highest activation (about 610 nodes), requires a significantly higher execution time (around 20 seconds), making it less efficient. In contrast, Q-Learning and genetic algorithm provide nearly the same level of influence spread with much lower execution times, offering better performance-efficiency trade-offs. Methods such as Degree Centrality and K-Shell activated far fewer nodes (110 and 90, respectively), showing limited effectiveness. Therefore, considering both effectiveness and execution time, Q-Learning emerges as the most balanced and efficient method for this dataset. This is clearly illustrated in figures 5 and 6.

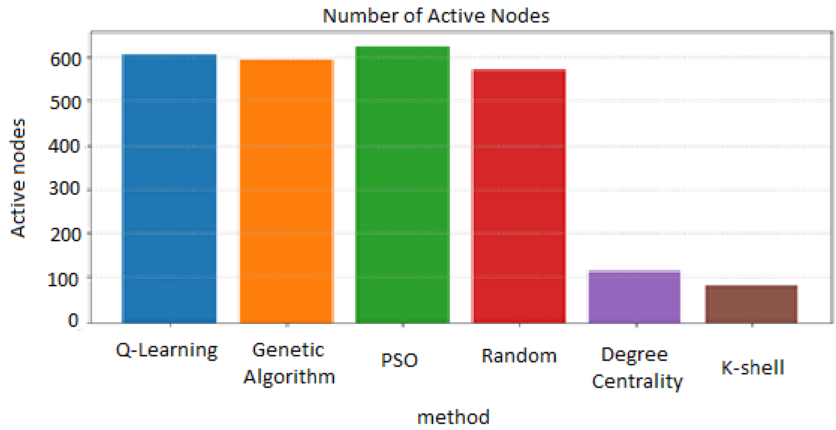

Figure 5: Comparison of active nodes with the Roget dataset and different algorithm.

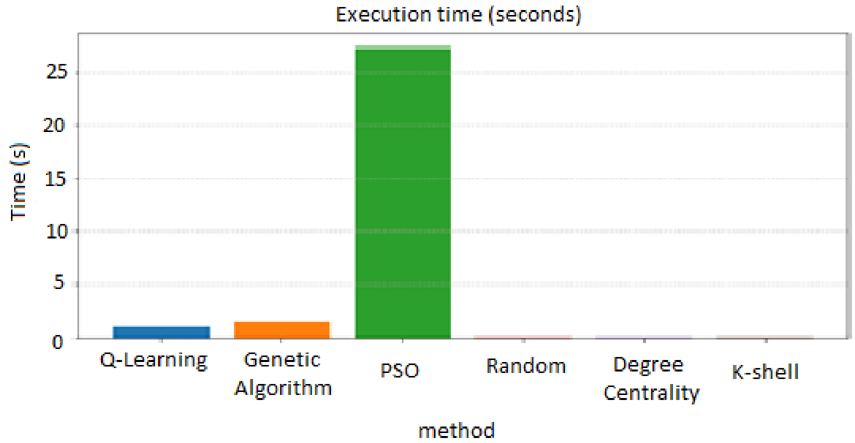

Figure 6: Comparison of execution time with the Roget dataset and different algorithm.

In the CSphd.csv dataset, which represents an academic network with a relatively limited structure and few key nodes, the Q-Learning method outperforms others by activating around 30 nodes. While PSO ranked second with approximately 23 activated nodes, its execution time was significantly higher (around 35 seconds), making it less efficient. Other methods, such as genetic algorithm, random, degree centrality, and K-Shell, activated fewer than 15 nodes, indicating low effectiveness in this type of network. Therefore, considering both the higher number of activated nodes and the much shorter execution time, Q-Learning can be considered as the most effective choice for this dataset, demonstrating strong capability in identifying strategic nodes even in small and specialized networks. This is illustrated in figures 7 and 8.

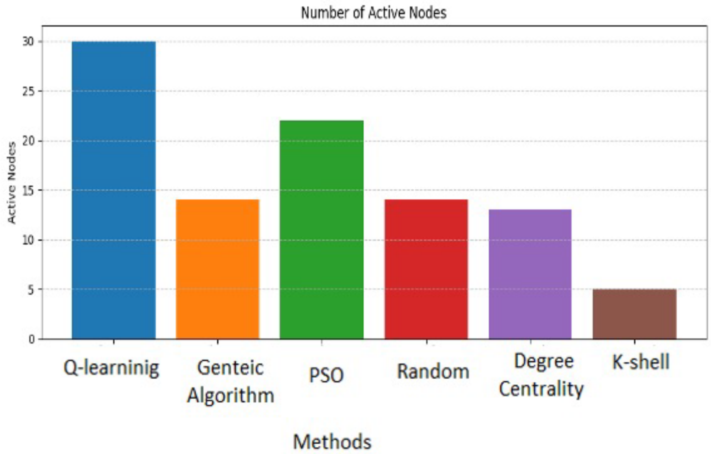

Figure 7: Comparison of active nodes with CSphd dataset and different algorithm.

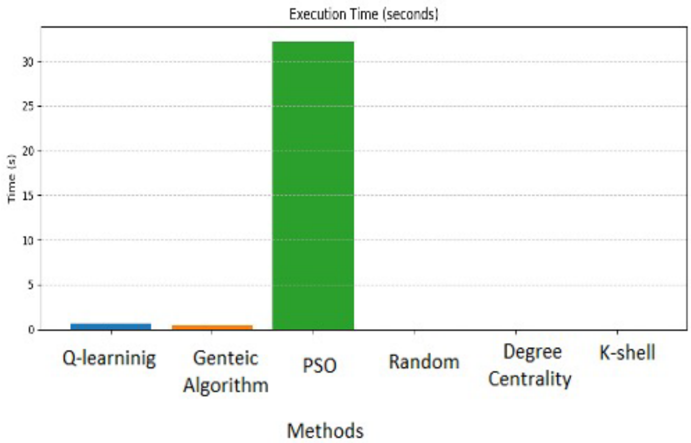

Figure 8: Comparison of the execution time with the CSphd dataset and different algorithms.

Figures 9 and 10 illustrate the results of applying different methods to the Netscience.csv dataset. The Degree Centrality method achieved the highest influence among all methods by activating approximately 198 nodes, while its execution time remained minimal. Although AI-based methods such as Q-Learning (with approximately 120 activated nodes) and PSO (with approximately 103 activated nodes) also showed relatively good performance, especially in the case of PSO, the high execution time (around 26 seconds) reduced their efficiency compared to faster methods. Therefore, in this particular network, Degree Centrality is identified as the most effective method, as it achieved the highest spread with minimal computational cost and without requiring complex calculations.

In the Netscience.csv dataset, the network appears to have a centralized structure or a skewed degree distribution, indicating that a limited set of nodes possessed an extremely high degree (number of connections), and these nodes play a crucial role in information diffusion or influence. In such networks, degree centrality performs well, as it selects explicitly these high-degree nodes as key influencers. In datasets like CSphd or YST, where the network structure may be more sparse, symmetric, or multi-centered, node degree is not always a dependable indicator of influence. Therefore, degree centrality cannot perform effectively in those cases.

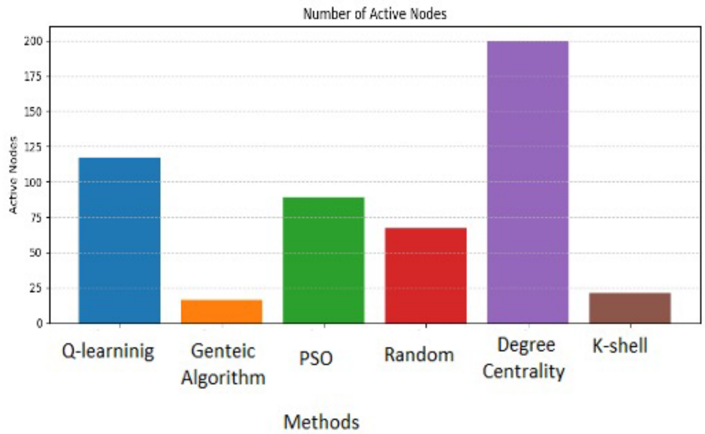

Figure 9: Comparison of active nodes with Netscience dataset and different algorithm.

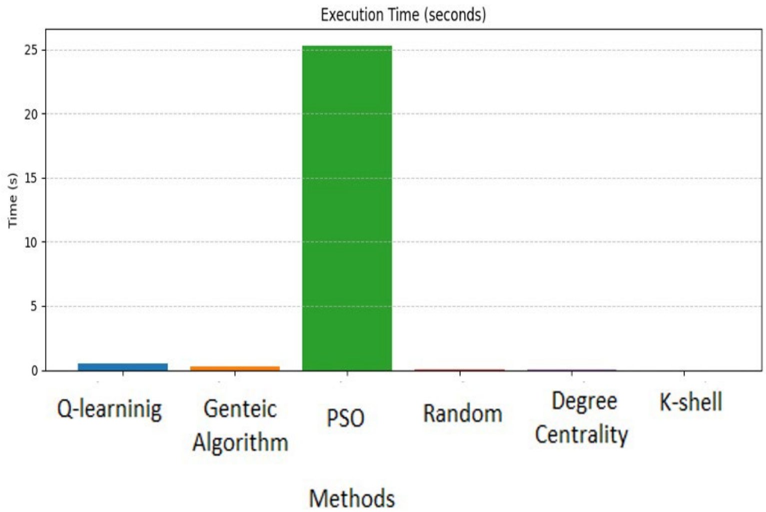

Figure 10: Comparison of the execution time with the Netscience and dataset different algorithms.

The results indicated that the proposed Q-learning algorithm outperformed other methods in most cases. The Q-learning method consistently achieved a higher influence spread. The degree centrality method performed similarly in the YST dataset, possibly due to the network's higher average degree. Still, Q-learning generally leads in other datasets such as Netscience and CSphd. Heuristic algorithms, such as random and degree centrality, have the lowest execution times but lower influence spread. Metaheuristic approaches such as PSO and Genetic Algorithms require significantly more time due to extensive search processes. The proposed Q-learning method offers a balance with moderate runtime and superior effectiveness.

5. Discussion

The results indicated that the Q-learning algorithm effectively balanced maximizing influence spread with maintaining computational efficiency. In datasets with complex structures and lower average degrees, simpler algorithms tend to underperform, whereas Q-learning leverages reinforcement learning to adaptively improve seed selection. Sensitivity analysis confirmed that setting the activation threshold to 0.1 optimizes performance across all tested datasets, underscoring the importance of parameter tuning.

In summary, the proposed method shows a competitive advantage over existing approaches by achieving a higher influence spread without significantly increasing runtime, making it a compelling strategy for influence maximization in social networks. We proposed a novel approach based on Q-learning to solve the IM problem in large-scale social networks. Our approach effectively addressed the challenge of selecting an optimal seed set to maximize influence spread. By leveraging reinforcement learning, the approach dynamically updates the Q-table based on feedback from the network propagation process. We demonstrated that by preprocessing the data to focus on high-degree nodes and utilizing the IC model, the Q-learning algorithm achieved both high accuracy and efficiency. The results from our experiments across various real-world datasets, such as CSphd-genealogy and Netscience-collaboration, showed that our method can consistently outperform traditional algorithms, such as degree centrality and PSO, in terms of influence spread while maintaining a reasonable execution time. Our approach offers significant advantages in both scalability and effectiveness, addressing the limitations of existing techniques. Unlike greedy or structural heuristics, which face scalability or accuracy issues, the Q-learning algorithm effectively balances exploration and exploitation, allowing it to adaptively identify the most influential nodes over time. Furthermore, through sensitivity analysis, we identified the optimal activation threshold to maximize performance, highlighting the importance of parameter tuning in real-world applications. The proposed method has the potential to be employed in multiple areas, like viral marketing, epidemic control, and opinion dynamics, where influence spread is a critical factor.

Looking ahead, several avenues for future work can further improve the proposed approach. To further develop and enhance the reinforcement learning method proposed in this paper, several measures can be taken in future research. A hybrid structural-content method can be presented for the selection of the initial core members. Given that influence is a relative notion, depending totally on content, the suggested reinforcement learning technique can also utilize contextual data. The proposed reinforcement learning approach can be assessed using different propagation models. For instance, the dependent cascade model can be used instead of the IC method, as the former considers the earlier efforts made by the nodes to activate their neighbors, unlike the latter. One potential direction is the integration of DRL techniques to improve the model's capacity to manage larger, more complex networks by learning richer, more abstract representations of influence propagation. Additionally, exploring hybrid approaches that combine Q-learning with other optimization techniques, such as genetic algorithms [31-33] or particle swarm optimization [34, 35], may lead to even more efficient and scalable solutions. Another interesting extension would be to incorporate temporal dynamics into the model, where the influence spread might vary over time, reflecting the evolving nature of social networks. Such advancements could make the approach more adaptable to real-time network environments, further enhancing its practical applicability.

6. Conclusions

In this study, a novel Q-learning-based algorithm was proposed to address the influence maximization problem in large-scale social networks. Experiments on real-world datasets such as CSphd, Netscience, and YST demonstrated that this method outperforms common algorithms like PSO and Degree Centrality in terms of influence spread while maintaining lower execution times. This balance between accuracy and efficiency makes the algorithm a suitable choice for applications such as viral marketing, epidemic control, and social dynamics analysis. Despite these successes, future improvements—such as integrating advanced propagation models and employing deep reinforcement learning—could further enhance the scalability and accuracy of the approach.

Author contribution: Khabat atar: Conceptualization, Investigation, Methodology. **Amir Sheikhahmadi:** Writing–original draft, Writing – review & editing, Conceptualization, Methodology**.** Project administration. **Keyhan Khamforoosh:** Conceptualization, Investigation, Methodology, Project administration.

Data availability: Data will be available upon reasonable request by authors.

Conflicts of interest: The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Funding: The authors did not receive support from any organization for the conducting of the study.

References

[1] S. Banerjee, M. Jenamani, and D. K. Pratihar, "A survey on influence maximization in a social network," *Knowledge and Information Systems*, vol. 62, no. 9, pp. 3417–3455, Sep. 2020, doi: 10.1007/s10115-020-01461-4

[2] D. H. Zanette, "Dynamics of rumor propagation on small-world networks," *Physical Review E*, vol. 65, no. 4, p. 041908, Apr. 2002, doi: 10.1103/PhysRevE.65.041908

[3] M. Czuba and P. Bródka, "Rank-refining seed selection methods for budget constrained influence maximization in multilayer networks under linear threshold model," *Social Network Analysis and Mining*, vol. 15, no. 1, p. 46, Apr. 2025. doi: 10.1007/s13278-025-01454-7

[4] A. Mohammadi, K. Khamforoosh. "Influence maximization in social networks using learning automata", *International Journal of Computer Applications*, 129(8):4-10, Nov. 2015, doi:10.5120/ijca2015906898

[5] D. Kempe, J. Kleinberg, and É. Tardos, "Maximizing the spread of influence through a social network," in *Proceedings of the 9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 137–146, Aug. 2003, doi: 10.1145/956750.956769

[6] T. Chen, S. Yan, J. Guo, and W. Wu, "ToupleGDD: A fine-designed solution of influence maximization by deep reinforcement learning," *IEEE Transactions on Computational Social Systems*, pp. 1–12, Oct. 2023, doi: 10.1109/TCSS.2023.3245670

[7] L. Wei, Z. Deng, Y. Chen, and L. Chen, "A Greedy Descent Method for Budget Constrained Continuous Influence Maximization in Online Social Network," in *IEEE Transactions on Computational Social Systems*, PP. 1-21, Apr. 2025, doi: 10.1109/TCSS.2025.3551651.

[8] H. Kaur and J. He, "Blocking negative influential node set in social networks: From host perspective," *Transactions on Emerging Telecommunications Technologies*, vol. 28, p. e4723, Jun. 2017, doi: 10.1002/ett.4723

[9] I. H. Sarker, "Machine learning: Algorithms, real-world applications and research directions," *SN Computer Science.*, vol. 2, no. 3, pp. 1–21, Mar. 2021, doi: 10.1007/s42979-021-00592-x.

[10] J. Leskovec, A. Krause, C. Guestrin, C. Faloutsos, J. VanBriesen, and N. Glance, "Cost effective outbreak detection in networks," in Proceedings of the 13th ACMSIG international conference on Knowledge discovery and data mining, Aug. 2007, pp.420-429, https://doi.org/10.1145/1281192.1281239

[11] T. Chen, S. Yan, J. Guo, and W. Wu, "ToupleGDD: A fine-designed solution of influence maximization by deep reinforcement learning," *IEEE Transactions on Computational Social Systems*, pp. 1–12, 2023, doi: 10.1109/TCSS.2023.3245670

[12] B. Guo, F. Z. Chen, and M. Q. Li, "A multi-objective optimization approach for influence maximization in social networks," in *Proceedings of the 24th International Conference on Industrial Engineering and Engineering Management*, Springer, May 2019, pp. 706–715, doi: 10.1007/978-981-13-3402-3_74

[13] K. Garimella, A. Gionis, N. Parotsidis, and N. Tatti, "Balancing information exposure in social networks," in *Proceedings of the 31st International Conference on Neural Information Processing Systems (NeurIPS)*,May 2017, pp. 4666–4674, doi: 10.5555/3294996.3295219

[14] R. Becker, F. Corò, G. D'Angelo, and H. Gilbert, "Balancing spreads of influence in a social network," in *Proceedings of the 34th AAAI Conference on Artificial Intelligence*, vol. 34, Feb, 2020, pp. 3–10, doi: 10.1609/aaai. v34i01.5327

[15] K. Ma, X. Xu, H. Yang, R. Cao and L. Zhang, "Fair Influence Maximization in Social Networks: A Community-Based Evolutionary Algorithm," in *IEEE Transactions on Emerging Topics in Computing,* vol. 13, no. 1, pp. 262-275, Jan.-Mar. 2025, doi: 10.1109/TETC.2024.3403891.

[16] S. Yang, Q. Du, G. Zhu, J. Cao, L. Chen, W. Qin, and Y. Wang, "Balanced influence maximization in social networks based on deep reinforcement learning," *Neural Networks*, vol. 169, pp. 334–351, Jan. 2024, doi: 10.1016/j.neunet.2023.10.030

[17] Q. He *et al*., "Multistage Competitive Opinion Maximization With Q-Learning-Based Method in Social Networks," in *IEEE Transactions on Neural Networks and Learning Systems*, vol. 36, no. 4, pp. 7158-7168, Apr. 2025, doi: 10.1109/TNNLS.2024.3387293.

[18] J. J. Lotf, M. A. Azgomi, and M. R. E. Dishabi, "An improved influence maximization method for social networks based on genetic algorithm," *Physica A: Statistical Mechanics and its Applications.* vol. 586, p. 126480, Jan. 2022, doi: 10.1016/j.physa.2021.126480.

[19] W. Ahmad and B. Wang, "A learning-based influence maximization framework for complex networks via K-core hierarchies and reinforcement learning," Expert Systems with Applications, vol. 259, p. 125393, Jan. 2025, doi: 10.1016/j.eswa.2024.125393

[20] S. Nandi, M. C. Malta, G. Maji, S. Saha, R. K. Mishra, and N. Ganguly, "IC-SNI: Measuring nodes' influential capability in complex networks through structural and neighboring information," *Knowledge. Information System*, vol. 67, pp. 1309–1350, Apr. 2025, doi: 10.1007/s10115-024-02262-9.

[21] N. Song, W. Sheng, Y. Sun, T. Lin, Z. Wang, Z. Xu, and F. Zhang , "Online dynamic influence maximization based on deep reinforcement learning," *Neurocomputing*, vol. 618, p. 129117, Feb. 2025, doi: 10.1016/j.neucom.2024.129117.

[22] S. Yang, et al., "Balanced influence maximization in social networks based on deep reinforcement learning," *Neural Networks*, vol. 169, pp. 334–351, Jan., 2024, doi: 10.1016/j.neunet.2023.10.030.

[23] K. Ali, C.-Y. Wang, M.-Y. Yeh, and Y.-S. Chen, "Addressing competitive influence maximization on unknown social network with deep reinforcement learning," in *2020 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining* (ASONAM), The Hague, Netherlands, pp. 196–203, Aug., 2020, doi: 10.1109/ASONAM49781.2020.9381471.

[24] J. Wang, Z. Cao, C. Xie, Y. Li, J. Liu, and Z. Gao, "DGN: Influence maximization based on deep reinforcement learning," *The Journal of Supercomputing*, vol. 81, no. 1, p. 130, Jan. 2025, doi.org/10.1007/s11227-024-06300-9.

[25] F. Wang, Z. Zhu, P. Liu, and P. Wang, "Influence maximization in social netwnhhjhork considering memory effect and social reinforcement effect," *Future Internet*, vol. 11, no. 4, p. 95, Apr. 2019, doi: 10.3390/fi11040095.

[26] Z. Wang, X. Chen, X. Li, Y. Du, and X. Lan, "Influence maximization based on network representation learning in social network," *Intelligent Data Analysis*, vol. 26, no. 5, pp. 1321–1340, Sep. 2022, doi.org/10.3233/IDA-216149.

[27] J. Tang, C. Li, L. Liu, T. Xu, and Y. Yao, "A novel evolutionary deep reinforcement learning algorithm for the influence maximization problem in multilayer social networks," *Chaos, Solitons & Fractals*, vol. 200, p. 116967, Nov. 2025, doi: 10.1016/j.chaos.2025.116967.

[28] P. Liu, L. Li, S. Fang, and Y. Yao, "Identifying influential nodes in social networks: A voting approach," *Chaos, Solitons & Fractals*, vol. 152, p. 111309, Nov. 2021, doi: 10.1016/j.chaos.2021.111309.

[29] A. Zareie, A. Sheikhahmadi, and K. Khamforoosh, "Influence maximization in social networks based on TOPSIS," *Expert Systems with Applications*, vol. 108, pp. 96–107, Oct. 2018, doi: 10.1016/j.eswa.2018.05.001.

[30] Adesokan, A. B. Rahman, and E. E. Tsiropoulou, "INFLUTRUST: Trust-based influencer marketing campaigns in online social networks," *Future Internet*, vol. 16, no. 7, p. 222, jun. 2024, doi: 10.3390/fi16070222.

[31] A. C. Khosroshahi, S. T. Afshord, B. Zarei, and B. Arasteh, "Influence maximization in social networks using improved genetic algorithm," *IEEE Access*, Jul., 2025, doi:10.1109/ACCESS.2025.3589189.

[32] J. J. Lotf, M. A. Azgomi, and M. R. Dishabi, "An improved influence maximization method for social networks based on genetic algorithm," *Physica A: Statistical Mechanics and its Applications*, vol. 586, p. 126480, Jan. 15, 2022, doi: 10.1016/j.physa.2021.126480.

[33] D. Bucur and G. Iacca, "Influence maximization in social networks with genetic algorithms," in *European Conference on the Applications of Evolutionary Computation*, Cham: Springer International Publishing, Mar., 2016, pp. 379–392, doi: 10.1007/978-3-319-77538-8_9.

[34] H. Khavandi, B. N. Moghadam, J. Abdollahi, and A. Branch, "Maximizing the impact on social networks using the combi-*nation of PSO and GA algorithms," Future Generation in Distributed Systems, vol. 5, pp. 1–3 ,Jun., 2023, https://doi.org/10.82553/josc.2025.140309071191740*.

[35] A. Maleki Ghalghachi and M. Roayaei, "deep q-learning enhanced variable neighborhood search for influence maximization in social networks," *International Journal of Web Research*, vol. 7, no. 2, pp. 23–36, Apr. 2024, doi: 10.22133/ijwr.2024.459158.1219.