

A reinforcement learning approach to edge suggestion for fair information access on social networks

Fangzheng Wang^a, Yanhao Wang^a,*, Jingjing Wang^b, Minghao Zhao^a

^a School of Data Science and Engineering, East China Normal University, Shanghai, 200062, China

^b College of Computer Science and Engineering, Changsha University, Changsha, 410022, China

ARTICLE INFO

Dataset link: <https://snap.stanford.edu/data>, <https://github.com/fzzf09/FIAES-RL>

Keywords:

Social network analysis
Edge augmentation
Reinforcement learning
Influence maximization
Algorithmic fairness

ABSTRACT

Fairness in information access on social networks has been actively investigated in recent years. Most existing studies on this topic focus on the problem of *Fair Influence Maximization* (FIM), which aims to select a set of seed nodes such that a propagation campaign initiated by them has fair influence spread across different groups. Although FIM approaches can guarantee fair access to specific information, they cannot resolve the inherent disparities in information access between different groups arising from graph structures. To address this issue, we study the problem of augmenting the graph structure via edge suggestion towards fairer information access at the group level. Specifically, we formulate a new optimization problem called *Fair Information Access via Edge Suggestion* (FIAES) that identifies a set of at most b non-existing edges to be added into the graph such that they not only maximally increase the total influence spread, but also ensure fairness in the sense that the influence spreads within different groups are proportional to their population sizes. Since FIAES is NP-hard and cannot be approximated within any constant factor, unless $P = NP$, we propose FIAES-RL, a reinforcement learning-based algorithm for edge selection that strikes a balance between influence and fairness objectives. Finally, with extensive experimentation on four synthetic and real-world networks, we demonstrate that FIAES-RL outperforms several state-of-the-art baseline methods for fairness-aware edge suggestion, reducing inequity in information access while significantly boosting information propagation.

1. Introduction

Social networks have become an important platform in various aspects of modern life, including communication [1], advertising [2], and event organization [3]. As information is typically propagated through word of mouth among users on social networks, the problem of finding a few “key influencers” that can trigger the largest expected spread of information in the network, called *Influence Maximization* [4] (IM), has been extensively investigated in the past two decades and has been widely used in real-world propagation campaigns. However, a substantial amount of evidence [5–8] suggests that the adoption of IM may cause disparate effects in information access between different groups: small and marginalized groups will find it much more difficult to receive information compared to advantaged groups through IM. When the information to be disseminated is about jobs [9], public health [5], and agricultural knowledge [10], inequity in access can exacerbate unfairness in opportunities and resources and further widen the social gap between different groups.

To alleviate inequity in information access, several *Fairness-aware IM* (FIM) problems [5,6,11–16] have recently received considerable attention. FIM approaches focus mainly on designing *fair seeding schemes*; that is, how to select a set of seed nodes that not only maximize the total influence spread but also reduce the influence gap between different groups. Although they ensure the equity of access to a specific propagation campaign, they cannot resolve the inherent unfairness in information access at its root causes. This is because such disparities arise from graph structures: Information dissemination triggered by IM approaches flows disproportionately to advantaged groups with more central nodes [7] that are more likely to be chosen as seeds or closer to seed nodes. FIM approaches address this problem by choosing some less central nodes as seeds to achieve sufficient influence spread on disadvantaged groups, nevertheless, without changing their inferior network positions [6].

Therefore, recent studies [7,17–21] have focused on augmenting the graph structure, usually through edge suggestion, to optimize the

* Corresponding author.

E-mail addresses: fzwang@stu.ecnu.edu.cn (F. Wang), yhwang@dase.ecnu.edu.cn (Y. Wang), wangjingjing2019@hnu.edu.cn (J. Wang), mhzhao@dase.ecnu.edu.cn (M. Zhao).

<https://doi.org/10.1016/j.knosys.2025.113719>

Received 7 December 2024; Received in revised form 21 April 2025; Accepted 6 May 2025

Available online 21 May 2025

0950-7051/© 2025 Elsevier B.V. All rights are reserved, including those for text and data mining, AI training, and similar technologies.

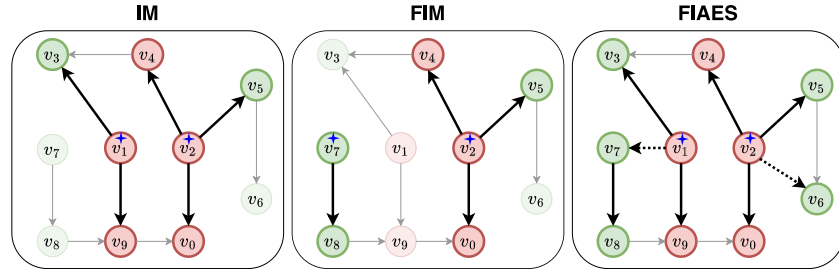


Fig. 1. Comparison of IM, FIM, and FIAES. We use red and green colors to denote the two groups of users in a social network and mark the seed users with blue stars. Then, users who are influenced and not influenced by the propagation campaign are denoted in heavier and lighter colors, respectively. In the vanilla IM problem, we pick $S = \{v_1, v_2\}$ as seed users, who triggers a large yet imbalanced influence spread between two groups. In the FIM problem, we replace a seed v_1 with a less centralized user v_7 , which reduces the influence spread while making it more balanced between groups. In our FIAES problem, we suggest two new edges to the graph so that, with the same seed set $S = \{v_1, v_2\}$, the influence spread is maximized and balanced between groups.

disparities in information access between groups. Although these methods have gained some success in improving the information access of disadvantaged groups by adding new edges to the graph, they still suffer from two limitations. First, some of them [7,20] adopt diffusion models that differ from those in IM problems, e.g., the *Independent Cascade* (IC) and *Linear Threshold* (LT) models [4], and thus become less effective when optimizing information flows triggered by IM methods. Second, they typically suggest edges that maximally increase the influence spread within the most disadvantaged group. However, such edge suggestions are often suboptimal for enhancing graph connectivity and boosting overall information flows in the graph. Furthermore, they may proportionally increase the influence spread across all groups, thus not eliminating information gaps.

In order to address the above challenges, we propose new approaches to augment the graph by edge suggestion towards fairer information access at the group level. Specifically, we formulate the *Fair Information Access via Edge Suggestion* (FIAES) problem, which finds a set of b non-existing edges in the graph to maximally boost the total influence spread in the IC model subject to a fairness constraint that the influence spreads within different groups should be proportional to their population sizes. We compare the FIAES problem with IM and FIM problems in Fig. 1. We show that the FIAES problem is NP-hard and cannot be approximated within any constant factor, unless $P = NP$. Unlike existing methods based on heuristics [18] or linear programming [17] for edge suggestion, we propose a novel reinforcement learning-based approach, called FIAES-RL, to the problem. Specifically, by simplifying the information diffusion process under the IC model and relaxing the fairness and budget constraints as Lagrange regularization terms in the reward function, we reformulate the FIAES problem as a Markov Reward Process (MRP) [22,23]. Then, we utilize a policy learning approach to edge selection by maximizing the regularized reward in the above MRP using a Multi-Layer Perceptron (MLP) network. By dynamically adjusting regularization parameters during the optimization process, FIAES-RL strikes a tunable balance between the objectives of influence (i.e., boosting the spread of total influence) and fairness (i.e., improving equity of access to information between different groups). Finally, through extensive experimentation on four real-world and synthetic networks, we demonstrate that FIAES-RL outperforms state-of-the-art baseline methods for fairness-aware edge suggestion: It not only brings about a greater increase in influence spread but also a more significant reduction in information gaps.

Our key contributions are summarized as follows:

- We formally define the FIAES problem and show its NP-hardness and inapproximability.
- We propose a new reinforcement learning-based algorithm, FIAES-RL, for FIAES and analyze its time complexity.
- We conduct extensive experiments on four real-world and synthetic datasets to show the effectiveness and efficiency of FIAES-RL compared to state-of-the-art methods for fairness-aware edge suggestion.

The remainder of this paper is organized as follows. First, we review existing work related to ours in Section 2. Then, we define the FIAES problem and show its NP-hardness in Section 3. Next, we propose the FIAES-RL algorithm in Section 4. We further present the experimental setup and results in Section 5. Finally, we conclude the paper in Section 6.

2. Related work

In this section, we review existing studies that are relevant to this work in the following four aspects: influence maximization with reinforcement learning, fair influence maximization, influence maximization through edge suggestion, and achieving fair information access through edge suggestion.

Influence Maximization with Reinforcement Learning: Since the seminal work of Kempe et al. [4], the IM problem has been studied extensively over the last two decades. Due to the vast literature on IM, we will not elaborate on the related work here and refer the interested reader to comprehensive surveys [24–26]. In recent years, increasing efforts have been devoted to addressing the IM problem using reinforcement learning (RL). Lin et al. [27] and Ali et al. [28,29] utilized RL in competitive IM problems. They formulated the competition between different agencies as an adversarial game and used RL to find the best seeding policy to maximize the profit of one agent, given that the opponent’s policy is known. Tian et al. [30] proposed an RL approach to the topic-aware IM problem. Kamarthi et al. [31] proposed an RL framework to discover effective sampling heuristics on unknown social networks. Wang et al. [32] and Li et al. [33] proposed IM algorithms via graph embedding and RL with high efficiency in large-scale and dynamic social networks. Chen et al. [34] and Wang et al. [35] further incorporated coupled graph neural networks (GNNs) for graph embedding with RL to improve IM performance. Liu et al. [36] designed an end-to-end framework that combines GNN as the encoder and RL as the decoder for the IM problem. Chen et al. [37] considered using RL to learn an IM policy on historical networks, which can be generalized to unseen networks. Ma et al. [38] considered combining an evolutionary algorithm (EA) with RL for the IM problem. Yang et al. [39] proposed using RL to balance the influence maximization of multiple entities to avoid the emergence of filter bubbles and echo chambers. Ghosh et al. [40] addressed the problem of group influence maximization (GIM) and proposed a robust framework for GIM using deep reinforcement learning. Wu et al. [41] proposed a DRL-based algorithm that utilized graph topology and observable user behaviors for incentive allocation in social networks. Jiang et al. [42] proposed a DRL approach to dynamically suppress the spread of rumors on social networks by blocking key users in multiple stages. However, the above methods consider neither fairness in IM nor the optimization of graph structures for IM.

Fair Influence Maximization: In recent years, fairness has been considered an increasingly important issue in IM problems. Tsang et al. [5]

introduced *maximin group fairness* into IM, with the aim of maximizing the minimum influence spread of any of the groups. Then, they proposed a multi-objective Frank–Wolfe algorithm with an approximation guarantee for the problem. Fish et al. [6] also considered *maximin fairness* in IM at the individual level, i.e., maximizing the minimum probability of any node being reached by the propagation process, and proposed greedy heuristics for the problem. Farnad et al. [11] proposed a general framework for IM problems with different concepts of fairness, including *equality*, *equity*, *maximin*, and *diversity*, using mixed integer programming (MIP). Rahmattalabi et al. [12] studied FIM from the perspective of social welfare theory and formulated group fairness via welfare maximization. They proved that their formulated FIM problem was monotone and submodular and could be solved approximately via submodular maximization. Becker et al. [14] proposed a randomized seeding strategy for (maximin) FIM. Khajehnejad et al. [43] and Feng et al. [15] applied graph embedding techniques to FIM problems. Lin et al. [44] proposed an attribute-based reverse influence sampling framework to balance influence spread and group fairness. Ali et al. [13] considered the time-criticality of information propagation and aimed to achieve balanced group-level influences within a time limit. Rui et al. [16] designed an improved $(1 - 1/e - \epsilon)$ -approximation algorithm for IM with welfare fairness based on reverse influence sampling. However, all aforementioned methods focus on fair seed selection but do not consider optimizing graph structures.

Influence Maximization through Edge Suggestion: Another line of work considers how to increase information flows in social networks through edge suggestions. Chaoji et al. [45] first formulated the problem of adding a bounded number of edges per node to maximize the influence of given seed nodes. D’Angelo et al. [46] defined the *Influence Maximization with Augmentation* (IMA) problem that adds a limited number of edges incident to a given set of seeds to maximize their influence spread under the IC model. Coro et al. [47] extended the IMA problem to the LT model. They all showed the NP-hardness of the defined problems and devised approximation algorithms for them. Yu et al. [48] proposed to balance relevance and spread enhancement in friend recommendation. Yang et al. [49] considered a generalized non-submodular IMA problem and designed a sandwich algorithm with a data-dependent approximation guarantee. Yu et al. [50] adopted a problem formulation similar to that defined by Yang et al. [49]. By showing that a restricted form of the problem is submodular, they proposed greedy algorithms with an approximation guarantee. Chen et al. [51] proposed an improved approximation algorithm with higher efficiency and scalability for the IMA problem. The above studies focus merely on optimizing graph connectivity via edge suggestion without considering fairness in information access.

Achieving Fair Information Access through Edge Suggestion: Finally, a few recent studies reduced disparities in information access through edge suggestion. Jalali et al. [7] defined the information unfairness between groups based on their flows received through random walks and proposed the MaxFair algorithm to reduce such unfairness with edge augmentation. Mashayekhi et al. [20] proposed the GREASE algorithm for edge suggestion to reduce graph imbalance by minimizing the earth mover’s distance between two groups of nodes. Note that the two methods adopted information diffusion models different from the IC model in this work. Becker et al. [18] adopted *maximin group fairness* and proposed adding edges to maximize the minimum coverage among all groups under the IC model. Bashardoust et al. [19] quantified structural bias in social networks through the lens of information flows under the IC model and proposed the BC-Chord algorithm to select edges to reduce such bias by increasing the access of least-advantaged nodes. Bhaskara et al. [21] studied a problem similar to that defined by Bashardoust et al. [19] and proposed a bicriteria approximation algorithm. Swift et al. [17] studied a problem similar to our FIAES problem: How to find a set of new edges to maximize content spread under the fairness constraint that the spread should be distributed proportionally to group sizes? They designed linear programming-based algorithms for this problem. The above methods are the most relevant and will be compared to our FIAES-RL algorithm in the experiments.

3. Preliminaries

In this section, we first introduce the basic concepts on social networks, information propagation, and fairness. Then, we formally define the FIAES problem on the basis of these concepts. Finally, we show the NP-hardness and inapproximability of the FIAES problem. Before presenting technical details, we list the frequently used notations in this paper in Table 1.

Graph Model: A social network is represented as a graph $G = (V, E)$, where V is the set of n nodes (users) and $E \subseteq V \times V$ is the set of edges (relationships). Here, the graph G can be directed or undirected. We assume that the nodes are associated with sensitive attributes (e.g., sex, race, and age) to divide them into m groups $V_1, V_2, \dots, V_m$ with $\bigcup_{i=1}^m V_i = V$ and $V_i \cap V_j = \emptyset$ for any $i \neq j$. To generate a set of candidates for edge suggestion, we adopt the popular “friends of friends” strategy [17,45,48]: Given a graph G , the candidate edges E_c include those that connect two nodes within two hops, but do not exist in E . Formally, the set of candidate edges generated by the “friends of friends” strategy is denoted as

$$E_c = \{(u, v) \in V \times V \setminus E \mid \exists w \in V, (u, w) \in E \wedge (w, v) \in E\}.$$

The intuition behind this strategy is that two users who do not share any common friends are less likely to become friends. Note that any other methods for candidate edge generation can also be integrated with our formulation.

Independent Cascade (IC) Model: To describe the information diffusion process on social networks, we adopt the IC model that has been widely used for IM-related problems [4,24]. The IC model assigns each edge $(u, v) \in E$ with a probability $p_{u,v} \in [0, 1]$ that indicates the influence strength between u and v . The stochastic diffusion process begins with an initial set S of active nodes (i.e., *seeds*) at step 0. Then, it unfolds in discrete steps $t = 1, 2, \dots$. At each step t , if a node u becomes active at step $t - 1$, it has a single chance to activate each of its inactive neighbors v with a success probability $p_{u,v}$. If u successfully activates v , v will try to activate its inactive neighbors at the next step $t + 1$. Regardless of whether u succeeds or fails in activating v , it cannot make any activation attempts in subsequent steps. When there are multiple neighbors who attempt to activate a node at the same step, different attempts will be performed sequentially in an arbitrary order. The process terminates when no more activations are possible. The active nodes after the diffusion process denote the users who are influenced by the propagation campaign. Given a graph $G = (V, E)$, a seed set $S \subseteq V$, and a subset $V' \subseteq V$, we use the influence spread function $\mathbb{I}_G(S, V')$ to denote the expected number of active nodes from V' in a diffusion process initiated by S under the IC model. Since computing $\mathbb{I}_G(S, V')$ exactly is known to be #P-hard [52], several methods, such as Monte-Carlo simulation [4], maximum influence path (MIP) [52], and reverse influence sampling (RIS) [53], have been proposed to estimate $\mathbb{I}_G(S, V')$ in polynomial time. In the subsequent section, we will devise a new method to estimate $\mathbb{I}_G(S, V')$ using a Markov Reward Process (MRP).

Fairness Model: Following the existing literature on group fairness in information access on social networks [11,17,19], we adopt the notion of *equity* as our fairness model. This requires that the expected influence spread within each group is proportional to its population size. Formally,

Definition 1 (Equity). Given a set of m groups $V_1, V_2, \dots, V_m$ on G and a seed set S , the influence spread under the IC model is equitable if for any $i, j \in [m]$, we have $\mathbb{I}_G(S, V_i)/|V_i| = \mathbb{I}_G(S, V_j)/|V_j|$.

The FIAES Problem: Based on the above concepts, we define the problem of *Fair Information Access via Edge Suggestion* (FIAES) as follows.

Table 1
List of frequently used notations in this paper.

Symbol	Description
G	A graph representing the social network
V	The set of users in G
E	The set of edges in G
V_i	A subset of users in V belong to the i th group
E_c	The set of candidate edges for the FIAES problem
$p_{u,v}$	The probability that node u successfully activates node v under the IC model
S	A set of seed nodes under the IC model
$\mathbb{I}_G(S, V)$	The expected number of active nodes in V influenced by S under IC model
b	The budget that limits the number of edges to add in the FIAES problem
$\mathcal{X}$	A finite set of states in the MRP
$\mathcal{T}$	The state transition function in the MRP
$\mathcal{R}$	The reward function in the MRP
$\mathbf{x}_t$	A $ V $ -dimensional vector denoting the activation status of each node at step t
$\mathbf{E}$	The adjacency matrix of G
$\mathbf{A}$	The activation probability matrix, where $A[i, j]$ denotes p_{ij}
$\mathbf{r}_i$	The group indicator vector for V_i
π	A policy in the MRP that indicates the selected edge set for the FIAES problem
$\mathbf{E}_\pi$	A policy matrix parameterizing the selection probability of candidate edges

Definition 2 (FIAES). For a graph $G = (V, E)$ with m groups $V_1, V_2, \dots, V_m$, a seed set $S \subseteq V$, a set $E_c \subseteq V \times V \setminus E$ of candidate edges, and a budget constraint $b \in \mathbb{Z}_+$, find a set $E' \subseteq E_c$ of at most b edges to be added to G as $G' = (V, E \cup E')$ such that (1) $\mathbb{I}_{G'}(S, V)$ is maximized and (2) the equity condition in Definition 1 is satisfied on G' w.r.t. S . Or formally,

$$\max_{E' \subseteq E_c} \mathbb{I}_{G'}(S, V) \quad (1)$$

$$\text{s.t. } \frac{\mathbb{I}_{G'}(S, V_i)}{|V_i|} = \frac{\mathbb{I}_{G'}(S, V_j)}{|V_j|}, \forall i, j \in [m]; \quad (2)$$

$$|E'| \leq b. \quad (3)$$

Then, the following theorem indicates the computational hardness and inapproximability of the FIAES problem.

Theorem 1. The FIAES problem is NP-hard and cannot be approximated within any constant factor unless $P = NP$.

Proof. First, since the *Influence Maximization with Augmentation* (IMA) problem, which has been proven to be NP-hard in [46], is a special case of the FIAES problem when there is only one group in V , the FIAES problem is NP-hard as well.

Second, we will show the inapproximability of FIAES from the fact that computing an (exact or approximate) solution to FIAES in polynomial time implies a polynomial-time algorithm for the EXACT COVER problem, which is known to be NP-complete [54]. Given a collection S of subsets of a ground set X , an exact cover is a sub-collection S^* of S such that each element in X is contained in exactly one subset of S^* . An instance $\mathcal{I}_{EC} = (S, X, b)$ of EXACT COVER is a decision problem that asks whether an exact cover of size $b \in \mathbb{Z}_+$ exists. Let us construct an instance $\mathcal{I}_{FI} = (G_0, S, E_c, b)$ of FIAES whose solution corresponds exactly to that of an instance $\mathcal{I}_{EC}$ of EXACT COVER (see Fig. 2 for illustration).

- (1) Add $|X| + 2|S|$ nodes to the node set V_0 of a directed graph G_0 , where $|X|$ nodes $v_1, v_2, \dots, v_{|X|}$ correspond to the elements $1, 2, \dots, |X|$ in X and two nodes u_i, u'_i correspond to each subset S_i in S .
- (2) Divide the nodes in V_0 into $|X| + 1$ groups: $V_i = \{v_i\}$ for each $i = 1, 2, \dots, |X|$ and $V_{|X|+1} = \{u_1, \dots, u_{|S|}, u'_1, \dots, u'_{|S|}\}$.
- (3) Add an edge (u_i, v_j) with an influence probability $p_0 = \frac{|S|+b}{2|S|}$ to the edge set E_0 of G_0 if an element j is contained in the set S_i .
- (4) Set the seed set $S = \{u'_1, u'_2, \dots, u'_{|S|}\}$, the set of candidate edges $E_c = \{(u'_i, u_i) \mid i \in [|S|]\}$, in which each edge is assigned with an influence probability 1, and a budget constraint b .

Obviously, the above construction procedure runs in polynomial time. Next, we show the equivalence between a solution to $\mathcal{I}_{EC}$ and its corresponding solution to $\mathcal{I}_{FI}$. If a sub-collection S^* is an exact cover of size b for $\mathcal{I}_{EC}$, the corresponding edge set $E'_0 = \{(u'_i, u_i) \mid S_i \in S^*\}$ w.r.t. S^* is a valid solution to $\mathcal{I}_{FI}$ because we have

$$\frac{\mathbb{I}_{G'_0}(S, V_{|X|+1})}{|V_{|X|+1}|} = \frac{|S| + b}{2|S|} = p_0 \quad \text{and} \quad \frac{\mathbb{I}_{G'_0}(S, V_i)}{|V_i|} = \frac{|S| + b}{2|S|} = p_0$$

for each $i \in [|X|]$. On the other hand, if a sub-collection S' of size b is not an exact cover for $\mathcal{I}_{EC}$, for its corresponding edge set $E''_0 = \{(u'_i, u_i) \mid S_i \in S'\}$, there must exist an element j that is either not covered or covered more than one time by S' . In the prior case, we have $\frac{\mathbb{I}_{G''_0}(S, V_j)}{|V_j|} = 0$; in the latter case, we have $\frac{\mathbb{I}_{G''_0}(S, V_j)}{|V_j|} > p_0$. In both cases, E''_0 is not a valid solution to $\mathcal{I}_{FI}$ because the equity constraint is not satisfied. Furthermore, if the size of S' is greater than b , E''_0 is not a valid solution because it violates the budget constraint. Similarly, if the size of S' is smaller than b , no matter whether S' is an exact cover or not, we have $0 < \frac{\mathbb{I}_{G''_0}(S, V_{|X|+1})}{|V_{|X|+1}|} < p_0$. Since $\frac{\mathbb{I}_{G''_0}(S, V_i)}{|V_i|}$ can only be 0 or at least p_0 for each $i \in [|X|]$, the equity constraint cannot be satisfied and E''_0 is also not a valid solution.

Therefore, if there exist a polynomial-time algorithm $\mathcal{A}$ to find an (exact or approximation) solution to FIAES, by applying the above construction procedure, $\mathcal{A}$ can be used to solve the EXACT COVER problem in polynomial time, which is impossible unless $P = NP$. $\square$

4. Methodology

In this section, we present a *Reinforcement Learning* (RL) based algorithm, called FIAES-RL, for the FIAES problem. The framework of FIAES-RL is illustrated in Fig. 3. Generally, FIAES-RL consists of two components: *Reward Agent* and *Policy Network*. The reward agent calculates the estimated influence of each group w.r.t. a given seed set and a policy for edge suggestions by reformulating FIAES as a *Markov Reward Process* (MRP). Then, taking the reward values calculated by the agent, the policy network updates the current policy for edge selection using a *Multi-Layer Perceptron* (MLP) network. The above two steps are performed iteratively and the policy is returned to obtain the final set of edge suggestions after a given number of iterations. Subsequently, we first describe the reformulation of FIAES as an MRP in Section 4.1. Then, we present the policy network in Section 4.2. Finally, we show the detailed procedure and time complexity of FIAES-RL in Section 4.3.

4.1. FIAES reformulation as MRP

We define an MRP $\mathcal{M} = \langle \mathcal{X}, \mathcal{T}, \mathcal{R}, \gamma \rangle$, where $\mathcal{X}$ is a finite set of states, $\mathcal{T}$ is the state transition, $\mathcal{R}$ is the reward function, and $\gamma \in (0, 1]$ is the

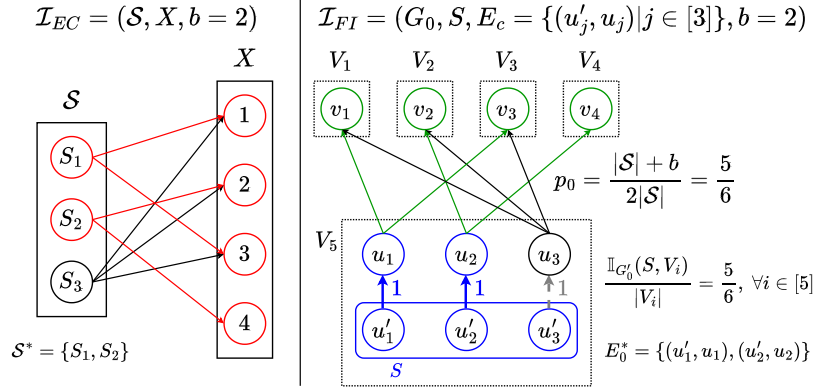


Fig. 2. Example of constructing an instance $\mathcal{I}_{FI}$ of FIAES from an instance $\mathcal{I}_{EC}$ of EXACT COVER for the proof of inapproximability in Theorem 1.

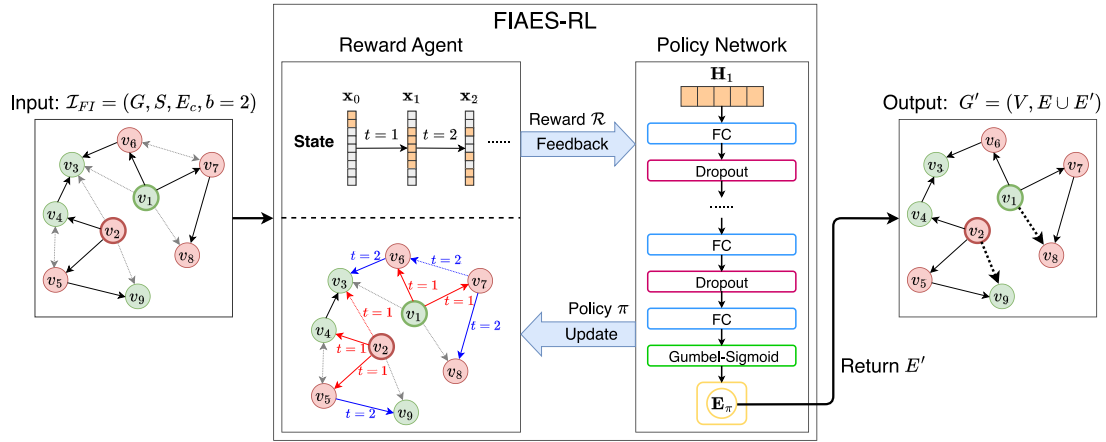


Fig. 3. Framework of the FIAES-RL algorithm.

discount rate, to estimate the diffusion process from the seed set $\mathcal{S}$ to different groups in G as follows.

(i) *State $\mathcal{X}$* : The state $\mathbf{x}_t$ at step $t = 0, 1, \dots, T$ is denoted by a $|V|$ -dimensional vector. When $t = 0$, for a given seed set $\mathcal{S}$, we define $\mathbf{x}_0 = [x_1, x_2, \dots, x_{|V|}]$, where $x_j = 1$ if node $v_j \in \mathcal{S}$ and $x_j = 0$ otherwise. For $t > 0$, we define $\mathbf{x}_t$ to indicate the probability that each node is activated by the diffusion process at step t .

(ii) *Transition $\mathcal{T}$* : We define a transition matrix $\mathbf{E}$ w.r.t. the edge set E , where $\mathbf{E}$ is a matrix of size $|V| \times |V|$ with $\mathbf{E}[i, j] = 1$ if there is an edge $e_{ij} = (v_i, v_j) \in E$ and $\mathbf{E}[i, j] = 0$ otherwise. In addition, we denote the activation probability of each edge by a matrix $\mathbf{A}$ of size $|V| \times |V|$, where each entry $\mathbf{A}[i, j]$ is the probability p_{ij} that v_i activates v_j during the diffusion process under the IC model. For each state $\mathbf{x}_t$, we compute its subsequent state $\mathbf{x}_{t+1}$ as $\mathbf{x}_t^T (\mathbf{E} \odot \mathbf{A})$, where $\odot$ is the Hadamard product; that is, given a vector $\mathbf{x}_t$ at step t , the vector $\mathbf{x}_{t+1}$ indicates the activation probability of each node at step $t + 1$. Finally, we have $\Pr[\mathbf{x}_{t+1} | \mathbf{x}_t, \mathbf{E}, \mathbf{A}] = 1$, since the stochastic diffusion process is converted into deterministic matrix operations.

(iii) *Reward Function $\mathcal{R}$* : In our formulation, the reward should estimate the information received by each group w.r.t. a seed set $\mathcal{S}$ in the diffusion process. Therefore, we define the reward function $\mathcal{R}_i$ for each group V_i to represent the sum of rewards obtained by reaching the nodes in V_i during the transition process from the initial state $\mathbf{x}_0$ to the final state $\mathbf{x}_T$. Here, T represents the maximum step in the MRP. To calculate rewards by groups, we define m group indicator vectors $\mathbf{r}_1, \mathbf{r}_2, \dots, \mathbf{r}_m$. Each $\mathbf{r}_i = [r_{i1}, r_{i2}, \dots, r_{i|V|}]$ is a $|V|$ -dimensional vector, where each $r_{ij} = 1$ if the node $v_j \in V_i$ and $r_{ij} = 0$ otherwise. Then, for each group V_i and an initial state $\mathbf{x}_0$, we compute the reward function $\mathcal{R}_i$ of V_i as $\sum_{t=0}^{T-1} \gamma^t \mathbf{x}_t^T (\mathbf{E} \odot \mathbf{A}) \mathbf{r}_i$. We use the reward function $\mathcal{R}_i$ to estimate the influence spread function $\mathbb{I}_{G'}(\mathcal{S}, V_i)$ under the IC model.

Since different groups are assumed to be non-overlapping, the total reward $\mathcal{R}$ can be easily calculated as $\sum_{i=1}^m \mathcal{R}_i$. Especially, we set the discount rate $\gamma = 1$ in the reward function; that is, the total reward is the sum of rewards over all steps without discounts. This setting matches exactly the FIAES formulation in Definition 2, which is based on the total number of activated nodes after the information diffusion process is complete and does not take into account the step in which a node is activated.

With the above MRP formulation, the objective of FIAES is transformed as learning a policy π to select an edge set $E' \in E_c$ that maximizes the reward function $\mathcal{R}$ under budget and fairness constraints. We denote the policy π with a parameterized matrix $\mathbf{E}_\pi \in [0, 1]^{|V| \times |V|}$, where each entry $\mathbf{E}_\pi[i, j]$ indicates the probability that a candidate edge $e_{ij} = (v_i, v_j)$ will be selected. For a given policy π , the transition matrix should be updated to $\mathbf{E} + \mathbf{E}_c \odot \mathbf{E}_\pi$, where $\mathbf{E}_c \in \{0, 1\}^{|V| \times |V|}$ is the adjacency matrix w.r.t. the candidate edge set E_c . As such, the reward function of each group V_i w.r.t. the policy π is

$$\mathcal{R}_i^\pi = \sum_{t=0}^{T-1} \gamma^t \mathbf{x}_t^T ((\mathbf{E} + \mathbf{E}_c \odot \mathbf{E}_\pi) \odot \mathbf{A}) \mathbf{r}_i$$

and the total reward function w.r.t. the policy π is $\mathcal{R}^\pi = \sum_{i=1}^m \mathcal{R}_i^\pi$ accordingly. Now, FIAES is reformulated to find the optimal strategy π^* to maximize the total reward function $\mathcal{R}^{\pi^*}$ while satisfying the equity and budget constraints, i.e.,

$$\max_{\pi} \mathcal{R}^\pi \quad \text{s.t.} \quad \mathcal{R}_i^\pi = \frac{|V_i|}{|V|} \cdot \mathcal{R}^\pi, \forall i \in [m] \quad \text{and} \quad \sum_{i=1}^m \sum_{j=1}^{|V|} \mathbf{E}_\pi[i, j] \leq b. \quad (4)$$

In Eq. (4), we have $\pi^* = \arg \max_{\pi} \mathcal{R}^\pi$.

Then, since the constrained optimization problem in Eq. (4) can still not be solved directly, we use the augmented Lagrangian method to

transform it into an unconstrained optimization problem. Specifically, we transform the two constraints in Eq. (4) as two regularization terms:

$$\alpha(\pi) = \sum_{i=1}^m |\mathcal{R}_i^\pi - \frac{|V_i|}{|V|} \cdot \mathcal{R}^\pi| \quad \text{and} \quad \beta(\pi) = \sum_{i=1}^{|V|} \sum_{j=1}^{|V|} \mathbf{E}_\pi[i, j] - b.$$

Intuitively, the values of $\alpha(\pi)$ and $\beta(\pi)$ measure how far the policy π is from satisfying the equity and budget constraints. The problem in Eq. (4) is thus transformed to the following problem:

$$\min_{\pi} \mathcal{L}(\pi) = -\mathcal{R}^\pi + \lambda_1 \alpha(\pi) + \frac{\sigma_1}{2} \alpha^2(\pi) + \frac{1}{2\sigma_2} \left[\max \{ \lambda_2 + \sigma_2 \beta(\pi), 0 \}^2 - \lambda_2^2 \right], \quad (5)$$

where $\sigma_1, \sigma_2 > 0$ are the penalty coefficients and $\lambda_1, \lambda_2 > 0$ are the Lagrange multipliers for the first and second regularization terms, respectively. At each iteration, the Lagrange multipliers are updated as follows:

$$\lambda_1^{new} \leftarrow \lambda_1^{old} + \sigma_1 \alpha(\pi) \quad \text{and} \quad \lambda_2^{new} \leftarrow \max \{ \lambda_2^{old} + \sigma_2 \beta(\pi), 0 \}. \quad (6)$$

4.2. The policy network

Next, we show how to learn a policy π by solving the optimization problem in Eq. (5). Since Eq. (5) is a non-convex optimization problem, we adopt a neural network (called *policy network*) to solve it. In our implementation, we design an MLP network with five hidden layers as the policy network. We do not adopt a more complex network architecture or add more hidden layers because they only marginally improve performance while incurring higher computational overhead. Specifically, the policy network $\mathcal{MLP}_\theta$ receives the initial state $\mathbf{x}_0$ and the transition matrix $\mathbf{E}_0$ as input. Then, it transforms $\mathbf{x}_0$ into a diagonal matrix $\mathbf{D}_{\mathbf{x}_0}$ with $\mathbf{D}_{\mathbf{x}_0}[j, j] = 1$ if $x_j = 1$ and $\mathbf{D}_{\mathbf{x}_0}[j, j] = 0$ otherwise for each $j \in [|V|]$. It generates a feature matrix $\mathbf{H}^{(1)}$ by concatenating $\mathbf{D}_{\mathbf{x}_0}$ and $\mathbf{E}_0$, i.e., $\mathbf{H}^{(1)} = \mathbf{D}_{\mathbf{x}_0} \oplus \mathbf{E}_0$, where $\oplus$ denotes the concatenation operation of two matrices, as input to the first fully connected layer of $\mathcal{MLP}_\theta$. Then, each fully connected layer is associated with a dropout layer [55] to calculate a hidden feature matrix, denoted as

$$\mathbf{H}^{(i+1)} = \text{Dropout}(\phi(\mathbf{H}^{(i)} \theta^{(i)})), \quad \forall i = 1, 2, \dots, 5.$$

Here, $\phi(\cdot)$ can denote any non-linear activation function, for which we use the hyperbolic tangent (tanh) function. Finally, it computes a policy matrix $\mathbf{E}_\pi$ from the last hidden feature matrix $\mathbf{H}^{(6)}$ using the sigmoid activation function $\psi(\cdot)$ enhanced by the Gumbel-Max trick [56], i.e., $\mathbf{E}_\pi = \psi(\phi(\mathbf{H}^{(6)} \theta^{(6)}))$. Especially, the purpose of employing the Gumbel-sigmoid function is to make the parameterized matrix $\mathbf{E}_\pi$ remain differentiable for parameter updates using gradients while being discretized into $\{0, 1\}$ in the final output. The re-parameterization using Gumbel-sigmoid is defined as

$$\mathbf{E}_\pi[i, j] = \frac{1}{1 + \exp((w_{ij} + g_j)/\tau)}, \quad \forall i, j \in [|V|]. \quad (7)$$

Here, g_j is a random variable drawn from Gumbel(0, 1) and w_{ij} represents the original weight of a candidate edge e_{ij} obtained from the hidden feature matrix. We initialize all w_{ij} 's to 0 in $\mathbf{E}_0$ to make the choice of edges independent of the input state. In Eq. (7), $\tau > 0$ is the temperature parameter that determines the degree to which the output $\mathbf{E}_\pi[i, j]$ is discretized. During the training process, the value of τ gradually decreases over epochs. When τ approaches 0, the output obtained from the Gumbel-sigmoid distribution is close to a one-hot vector. Finally, the parameters θ in the policy network $\mathcal{MLP}_\theta$ are $\theta = \{\theta^{(1)}, \theta^{(2)}, \dots, \theta^{(6)}\}$.

The policy network $\mathcal{MLP}_\theta$ returns a parameterized policy matrix $\mathbf{E}_\pi$ after each training epoch. Then, we use $\mathbf{E}_\pi$ to update the transition matrix as $\mathbf{E}_k \leftarrow \mathbf{E} + \mathbf{E}_c \odot \mathbf{E}_\pi$ for the k th epoch ($k = 1, 2, \dots, K$) and calculate the value of the loss function $\mathcal{L}(\pi)$. Next, we take the derivative of the loss function $\mathcal{L}$ with respect to $\mathbf{E}_\pi$ and update all parameters θ of the policy network $\mathcal{MLP}_\theta$ along the gradient direction to minimize the value of $\mathcal{L}$. The above procedure is performed iteratively until a pre-defined number K of epochs is reached.

Algorithm 1: FIAES-RL

Input: Graph $G = (V, E)$ with groups $V_1, \dots, V_m$, set E_c of candidate edges, activation matrix $\mathbf{A}$, seed set S , budget b

Output: Edge suggestion $E' \subseteq E_c$

Parameters: $\tau = 1$; $\lambda_1, \lambda_2 = 0$ and σ_1, σ_2 ; θ ; constraint violation thresholds ϵ_1, ϵ_2 ; step size η ; temperature attenuation factor ν

- 1 Generate the initial policy matrix $\mathbf{E}_\pi$ using Eq. (7) with $w_{ij} = 0$ for all i, j 's;
- 2 **for** $k = 1, 2, \dots, K$ **do**
- 3 Obtain the policy matrix $\mathbf{E}_\pi$ from the $(k-1)$ -th epoch;
- 4 Update the transition matrix $\mathbf{E}_k \leftarrow \mathbf{E} + \mathbf{E}_c \odot \mathbf{E}_\pi$;
- 5 Set the loss function as $\mathcal{L}_k(\pi) = -\mathcal{R}^\pi$;
- 6 **if** $k \geq K_{es}$ **then**
- 7 Set $\mathcal{L}_k(\pi)$ as $\mathcal{L}_k(\pi) + \lambda_1 \alpha(\pi) + \frac{1}{2} \sigma_1 \alpha(\pi)^2$;
- 8 **if** $|\alpha(\pi)| > \epsilon_1$ **then**
- 9 Update the value of λ_1 as Eq. (6);
- 10 **if** $k \geq K_{bs}$ **then**
- 11 Set $\mathcal{L}_k(\pi)$ as $\mathcal{L}_k(\pi) + \frac{1}{2\sigma_2} \left(\max \{ \lambda_2 + \sigma_2 \beta(\pi), 0 \}^2 - \lambda_2^2 \right)$;
- 12 **if** $|\beta(\pi)| > \epsilon_2$ **then**
- 13 Update the value of λ_2 as Eq. (6);
- 14 **if** $k \geq K_{ts}$ **then**
- 15 Set $\tau \leftarrow \tau \nu$;
- 16 Update the policy network parameters $\theta \leftarrow \theta - \eta \nabla \mathcal{L}_k(\pi)$;
- 17 **return** E' by selecting the top- b edges with the highest values in $\mathbf{E}_\pi$;

4.3. The FIAES-RL algorithm

Algorithm 1 provides a detailed description of the procedure of FIAES-RL. We adopt an adaptive training strategy that optimizes different components of the loss function in different epochs. First of all, we add the total reward function $\mathcal{R}^\pi$ to the loss function $\mathcal{L}_k$ across all epochs k from 1 to the pre-defined limit K . Then, we add the regularization terms w.r.t. the equity and budget constraints to the loss function $\mathcal{L}_k$ from the K_{es} th and K_{bs} th epochs, respectively. When the two regularization terms are added, we check whether the values of the constraint functions $\alpha(\pi), \beta(\pi)$ exceed the constraint violation thresholds ϵ_1, ϵ_2 in each epoch. If so, we will update the Lagrange multipliers λ_1, λ_2 according to Eq. (6). Furthermore, from the K_{ts} th epoch and so on, we discretize the policy matrix $\mathbf{E}_\pi$ by gradually annealing the temperature τ of the Gumbel-sigmoid function, ensuring that the entries of $\mathbf{E}_\pi$ can be close to $\{0, 1\}$ after K epochs. We note that there does not exist a universal scheme to assign the parameters and they should be fine-tuned for different datasets and problem settings to achieve satisfactory outcomes. In each epoch, we obtain $\mathbf{E}_\pi$ from the previous epoch and the loss function $\mathcal{L}_k$ by summing up its corresponding components as described above. Then, the parameters θ of $\mathcal{MLP}_\theta$ are updated via stochastic gradient descent. In addition, we set $\epsilon_1 = +\infty$ and $\epsilon_2 = 0$ in the implementation. This allows us to manually control the importance of the equity constraint by only adjusting the value of σ_1 , and to set the same value of σ_2 across different datasets and problem settings.

Time Complexity: Next, we analyze the time complexity of FIAES-RL. In each epoch, we need to calculate the reward function $\mathcal{R}_i$ for each group V_i . The operation takes $O(T \cdot |V|^2)$ time on matrices of size $|V| \times |V|$ and $|V|$ -dimensional vectors over T iterations. Then, it takes $O(m)$ such operations to compute the rewards of m groups and the total reward. Since the total number of parameters in $\mathcal{MLP}_\theta$ is $O(|V|^2)$, the time complexity of parameter updates in each epoch is $O(|V|^2)$. Therefore, the time complexity per epoch is $O(mT \cdot |V|^2)$ and the overall complexity of FIAES-RL is $O(mTK \cdot |V|^2)$.

Table 2
Statistics of network datasets used in the experiments.

Dataset	$ V $	$ E $	m	$ E_c $	p
Antelope Valley	500	1697	2	5221	0.5
Email-Eu	1005	25,571	2	97,240	0.3
Epinions	1000	21,795	2	95,228	0.3
Facebook	4039	88,234	3	716,134	0.1

5. Experiments

In this section, we conduct extensive experiments to evaluate the performance of the FIAES-RL algorithm. We first describe the experimental setup in Section 5.1. Then, we compare the FIAES-RL algorithm with the state-of-the-art baselines in Section 5.2. Finally, we conduct the parameter sensitivity analysis for the FIAES-RL algorithm in Section 5.3.

5.1. Experimental setup

Datasets: We use the following four publicly available network datasets in the experiments:

- **Antelope Valley** is a synthetic network proposed in [5]. It models a campaign for obesity prevention intervention in the Antelope Valley region of California, USA. The nodes are labeled with a sensitive attribute *gender* and divided into two groups.
- **Email-Eu** is a real network downloaded from the SNAP collection [57]. It denotes the email communications within an European research institution. We divide the nodes into two groups by the departments to which they belong (Dep. ID 0-20 for one group and Dep. ID 21-41 for the other group).
- **Epinions** is also a real network downloaded from SNAP. It is constructed from user interactions on Epinions, a consumer review website, where each edge represents the trust relationships between two users. Since the ground-truth user groups or communities are not available, we run the Louvain method [58] to partition the nodes into two groups.
- **Facebook** is another real network downloaded from SNAP. It denotes the friendships between users on Facebook. The users are divided into three groups based on an anonymized attribute.

We summarize the statistics of these datasets in Table 2, where $|V|$, $|E|$, and m are the numbers of nodes, edges, and groups in the network, $|E_c|$ is the number of candidate edges in E_c generated using the “*friends of friends*” strategy, and p is the edge activation probability in the IC model.

Baselines: We compare the FIAES-RL algorithm with the following five baselines for fairness-aware edge suggestion:

- **LP-Advanced** [17] is a linear programming-based algorithm that is also proposed to suggest edges to maximize the influence spread with an equity constraint.
- **BC-Chord** [19] is a heuristic edge suggestion algorithm to reduce the access diameter of the network by improving the influence probability of the least advantaged node.
- **Bicriteria** [21] is a bicriteria approximation algorithm proposed for the same problem as defined in [19].
- **MaxFair** [7] aims to balance the information flows with different groups triggered by random walks.
- **GREASE** [20] reduces the imbalance between two groups by reducing the Earth Mover’s Distance (EMD) of their distributions.

Note that we do not compare with the algorithms in [18] as they cannot be completed within one day in our settings.

Implementation: All experiments were carried out on a server with a 14-core Intel Xeon(R) Platinum 8362 processor @3.6 GHz, 64 GB DDR4 memory and a Nvidia RTX 3090 GPU running Ubuntu 18.04 LTS. All algorithms were implemented in Python 3 with the TensorFlow 2 library for policy network training. For each baseline, we used the implementation published by the original authors or implemented it ourselves according to the original paper. Moreover, we followed the default parameter setting as suggested in the original paper for each baseline. For FIAES-RL, the default parameters for each dataset are presented in Table 3. Note that all parameters have been fine-tuned over specific ranges, and the ones with the best trade-offs between effectiveness and efficiency are used. Detailed analysis of the parameter sensitivity of FIAES-RL is presented in Section 5.3. By default, we set the budget b for each algorithm to 400, 1000, 1500, and 3000 on the *Antelope Valley*, *Email-Eu*, *Epinions*, and *Facebook* datasets, respectively. All algorithms were run ten times on each graph, and the average and standard deviation and each metric were reported. Our code is publicly available at <https://github.com/fzzf09/FIAES-RL>.

Evaluation Metrics: We evaluate the effectiveness of each method in two aspects: *Influence Boost Ratio* (IBR) and *Equity score* (EQ) and the efficiency of each method in terms of *running time*. The IBR of a set E' of suggested edges is defined as the increase in the expected influence spread of a seed set S in the updated graph $G' = (V, E \cup E')$ compared to its expected influence spread in G , i.e.,

$$\text{IBR}(E') = \frac{\mathbb{I}_{G'}(S, V) - \mathbb{I}_G(S, V)}{\mathbb{I}_G(S, V)}. \quad (8)$$

We select nodes with the highest degrees as the set S of seeds in each network and run 1000 Monte-Carlo simulations under the IC model to estimate its influence spread. Then, the EQ of a set E' of suggested edges measures how close the group-wise influence spread after edge augmentation is to satisfying the equity constraint in Definition 2. Specifically, we have

$$\xi_i(E') = \frac{\mathbb{I}_{G'}(S, V_i)}{|V_i|} \text{ and } \text{EQ}(E') = \min_{i,j \in [m]} \min \left\{ \frac{\xi_i(E')}{\xi_j(E')}, \frac{\xi_j(E')}{\xi_i(E')} \right\}. \quad (9)$$

5.2. Comparison with state-of-the-art baselines

Overall Performance: We present the performance of all algorithms across the four datasets in terms of IBR and EQ by varying the size of the seed set S from 3 to 15 in Table 4. Generally, FIAES-RL consistently outperforms other baselines in almost all cases in terms of IBR and EQ: On the one hand, it always achieves significantly higher IBRs than all other baselines; on the other hand, it also increases the initial EQs to at least 0.899, which are also the highest among all algorithms in all cases except for the *Email-Eu* dataset when $|S| = 3$. LP-Advanced ranks second among all methods in most cases because its optimization objectives are similar to those of FIAES. The gap between FIAES-RL and LP-Advanced clearly indicates that our RL-based method shows significantly better empirical performance than the LP-based method in [17]. BC-Chord sometimes achieves better EQs than LP-Advanced but provides worse IBRs. This means that improving the least advantaged node can reduce group-level information disparity but cannot optimize graph connectivity. Bicriteria improves upon BC-Chord in terms of IBRs at the expense of lower EQs. Finally, MaxFair and GREASE typically cannot provide satisfactory edge suggestions because they do not adopt the IC model to describe the information diffusion process and cannot provide accurate influence estimations for FIAES.

We also present the running time of each algorithm when $|S| = 6$ in Table 5. The results for other seed set sizes are omitted because the running time is generally similar for each method. BC-Chord always runs the fastest among all algorithms because it adopts a simple greedy strategy for edge selection without requiring any optimization procedure. FIAES-RL runs slightly slower than LP-Advanced since it takes more time to train the policy network than LP-Advanced to solve

Table 3

Default parameters of FIAES-RL on each dataset in the experiments.

Parameter	Antelope Valley	Email-Eu	Epinions	Facebook
Time horizon T	15	10	10	6
Penalty coefficients (σ_1, σ_2)	(1, 0.5)	(2, 1)	(2, 1)	(5, 2)
Number of epochs K	700	1500	1500	2400
Budget scheduling K_{bs}	0	0	0	0
Equity scheduling K_{es}	100	300	300	800
Temperature scheduling K_{ts}	50	100	100	200
Temperature attenuation factor ν	0.999	0.995	0.995	0.99

Table 4

Performance of different fairness-aware edge suggestion algorithms in terms of IBR and EQ across the four networks with the default budgets as indicated in Section 5.1 (i.e., $b = 400, 1000, 1500$, and 3000 on *Antelope Valley*, *Email-Eu*, *Epinions*, and *Facebook*, respectively) when the size of the seed set S is set to 3, 6, 9, 12, 15. The best and second-best results in each setting are highlighted by **bold** and underlined fonts.

Dataset	S	Initial	FIAES-RL		LP-Advanced		BC-Chord		Bicriteria		MaxFair		GREASE	
		EQ	IBR	EQ	IBR	EQ	IBR	EQ	IBR	EQ	IBR	EQ	IBR	EQ
Antelope valley	3	0.729	1.155±0.094	0.909±0.035	<u>1.108±0.114</u>	0.816±0.039	0.605±0.082	0.762±0.056	0.989±0.091	0.792±0.086	0.529±0.061	0.786±0.069	0.708±0.071	<u>0.821±0.044</u>
	6	0.867	1.043±0.063	0.965±0.018	<u>0.997±0.113</u>	0.846±0.024	0.568±0.073	0.902±0.011	0.829±0.082	0.825±0.078	0.549±0.055	0.821±0.062	0.656±0.065	<u>0.928±0.039</u>
	9	0.879	0.964±0.032	0.968±0.014	0.863±0.062	<u>0.894±0.035</u>	0.443±0.051	0.884±0.032	0.776±0.073	0.846±0.070	0.481±0.049	0.830±0.055	0.579±0.058	<u>0.872±0.034</u>
	12	0.835	0.819±0.049	0.939±0.028	<u>0.740±0.079</u>	0.869±0.034	0.402±0.057	0.841±0.026	0.707±0.064	<u>0.887±0.061</u>	0.452±0.042	0.857±0.048	0.514±0.051	<u>0.871±0.029</u>
	15	0.755	0.699±0.067	0.911±0.017	<u>0.652±0.054</u>	0.865±0.048	0.347±0.035	0.763±0.042	0.562±0.055	0.856±0.053	0.383±0.036	0.795±0.041	0.479±0.044	<u>0.823±0.024</u>
EmailâEu	3	0.882	0.535±0.072	<u>0.959±0.018</u>	<u>0.463±0.050</u>	0.923±0.030	0.268±0.028	0.964±0.017	0.319±0.030	0.895±0.022	0.226±0.020	0.879±0.019	0.272±0.015	0.857±0.014
	6	0.873	0.495±0.059	<u>0.974±0.013</u>	<u>0.412±0.046</u>	0.908±0.028	0.281±0.026	<u>0.933±0.016</u>	0.294±0.027	0.852±0.020	0.193±0.018	0.903±0.017	0.257±0.014	0.872±0.013
	9	0.883	0.436±0.036	0.982±0.011	<u>0.357±0.032</u>	<u>0.956±0.026</u>	0.270±0.025	0.917±0.015	0.329±0.025	0.918±0.019	0.176±0.017	0.872±0.016	0.300±0.013	0.923±0.012
	12	0.865	0.403±0.034	0.987±0.005	<u>0.333±0.021</u>	<u>0.960±0.024</u>	0.214±0.024	0.913±0.014	0.323±0.024	0.943±0.018	0.184±0.016	0.893±0.016	0.299±0.012	0.904±0.011
	15	0.859	0.341±0.023	0.980±0.009	<u>0.301±0.020</u>	<u>0.924±0.022</u>	0.256±0.022	0.888±0.013	<u>0.312±0.023</u>	0.897±0.017	0.166±0.015	0.845±0.015	0.228±0.011	0.873±0.010
Epinions	3	0.816	0.590±0.028	0.903±0.022	<u>0.549±0.027</u>	0.842±0.024	0.157±0.015	<u>0.883±0.022</u>	0.433±0.025	0.851±0.020	0.164±0.018	0.878±0.019	0.149±0.018	0.862±0.017
	6	0.802	0.606±0.026	<u>0.912±0.021</u>	<u>0.552±0.025</u>	0.836±0.022	0.151±0.014	0.870±0.020	0.385±0.023	0.843±0.018	0.142±0.017	0.865±0.018	0.156±0.017	0.873±0.016
	9	0.789	0.612±0.024	0.899±0.020	<u>0.577±0.023</u>	0.831±0.020	0.144±0.013	0.858±0.018	0.393±0.022	0.839±0.016	0.156±0.016	0.851±0.017	0.145±0.016	0.848±0.015
	12	0.776	0.571±0.022	<u>0.911±0.018</u>	<u>0.542±0.021</u>	0.823±0.018	0.137±0.012	<u>0.848±0.016</u>	0.367±0.020	0.834±0.014	0.116±0.015	0.838±0.016	0.139±0.015	0.798±0.014
	15	0.835	0.529±0.020	<u>0.913±0.017</u>	<u>0.489±0.019</u>	<u>0.884±0.016</u>	0.136±0.011	0.883±0.014	0.380±0.021	0.853±0.012	0.117±0.014	0.873±0.015	0.176±0.014	0.815±0.013
Facebook	3	0.850	0.165±0.013	<u>0.971±0.010</u>	0.122±0.012	0.931±0.013	0.092±0.010	<u>0.938±0.013</u>	0.139±0.011	0.872±0.010	0.093±0.009	0.916±0.011	0.115±0.011	0.919±0.010
	6	0.836	0.191±0.014	<u>0.957±0.011</u>	<u>0.146±0.013</u>	0.928±0.012	0.113±0.011	<u>0.933±0.012</u>	0.140±0.021	0.874±0.011	0.115±0.010	0.892±0.011	0.109±0.010	0.921±0.009
	9	0.871	0.188±0.014	<u>0.943±0.012</u>	<u>0.160±0.014</u>	0.924±0.013	0.099±0.010	<u>0.937±0.012</u>	0.124±0.011	0.901±0.010	0.074±0.008	0.875±0.011	0.102±0.009	0.905±0.010
	12	0.867	0.131±0.012	<u>0.939±0.011</u>	<u>0.123±0.011</u>	0.916±0.012	0.103±0.010	<u>0.923±0.011</u>	0.102±0.010	0.911±0.010	0.084±0.009	0.885±0.010	0.097±0.009	0.899±0.010
	15	0.893	0.149±0.013	0.953±0.012	<u>0.110±0.010</u>	<u>0.933±0.011</u>	0.077±0.008	0.930±0.010	0.101±0.010	0.912±0.009	0.081±0.009	0.891±0.010	0.095±0.009	0.908±0.009

Table 5

Running time (in seconds) of each algorithm across the four networks when $|S| = 6$. All other settings remain the same as those in Table 4.

Dataset	FIAES-RL	LP-Advanced	BC-Chord	Bicriteria	MaxFair	GREASE
Antelope Valley	6.255±1.804	<u>2.221±0.518</u>	1.195±0.214	7.145±1.932	10.126±2.044	8.190±1.638
Email-Eu	64.863±9.263	<u>19.380±2.765</u>	12.740±2.109	75.519±8.074	105.808±10.521	100.334±9.867
Epinions	76.541±11.585	<u>40.531±5.023</u>	12.900±2.230	80.943±9.115	115.997±13.469	106.269±12.055
Facebook	1830.562±146.377	<u>1594.116±130.985</u>	40.627±6.231	3290.717±285.740	5126.563±412.681	3478.562±301.464

linear programs. However, FIAES-RL has a higher efficiency than the remaining three baselines, which also involve optimizations or matrix operations in the edge selection process. Overall, FIAES-RL strikes the best balance between the influence and fairness objectives while running in a reasonable time.

Effect of Budget b : Fig. 4 presents the performance of different algorithms by varying budget b across the four datasets. We observe trends similar to those in Tables 4 and 5. FIAES-RL outperforms all baselines in terms of IBR and EQ in almost all cases. LP-Advanced mostly ranks second in terms of IBR but has moderate equity scores on the *Epinions* and *Facebook* datasets. The other four baselines show inferior performance for FIAES due to misalignments in the notions of fairness and the information diffusion models. For FIAES-RL, when the budget b increases (i.e., more new edges can be inserted into the graph), the IBR naturally increases, while the EQ slightly decreases. This is because the same value of the equity penalty coefficient σ_1 is used across different budgets. As the reward values increase with b , the weight of the equity regularization term is relatively reduced. In addition, the running time of FIAES-RL remains nearly stable with b because the policy network training and reward computation processes are almost unaffected by b . For algorithms that perform iterative procedures to pick each edge one by one, the running time is generally linear w.r.t. b .

5.3. Parameter sensitivity analysis

Effect of Equity Penalty Coefficient σ_1 : For FIAES-RL, the trade-off between influence and equity objectives can be achieved by adjusting the value of σ_1 . When $\sigma_1 = 0$, the regularization term for the equity constraint will not affect the optimization process; with a greater value of σ_1 , the penalty for violating the equity constraint increases. In Fig. 5, we present the performance of FIAES-RL in terms of IBR and EQ by varying the value of σ_1 . We can observe that when the value of σ_1 increases, the IBR decreases while the EQ increases consistently on all four datasets. In practice, by simply tuning the value of σ_1 , FIAES-RL can easily trade off between the two objectives of boosting influence and ensuring equity.

Convergence Analysis and Effect of Number of Epochs K : Fig. 6 illustrates the loss value $\mathcal{L}(\pi)$ of FIAES-RL over epochs when training the policy network using the default parameters outlined in Table 3. In the initial training phase, the loss value consisting of $-\mathcal{R}(\pi)$ and the budget penalty term decreases rapidly and converges. Then, we observe a surge in the loss value at the K_{es} th iteration due to the inclusion of the equity penalty term. Finally, the loss value stabilizes and converges again.

We also illustrate the performance of FIAES-RL in terms of IBRs and EQs by varying the number of epochs K in Fig. 7. Here, we scale the

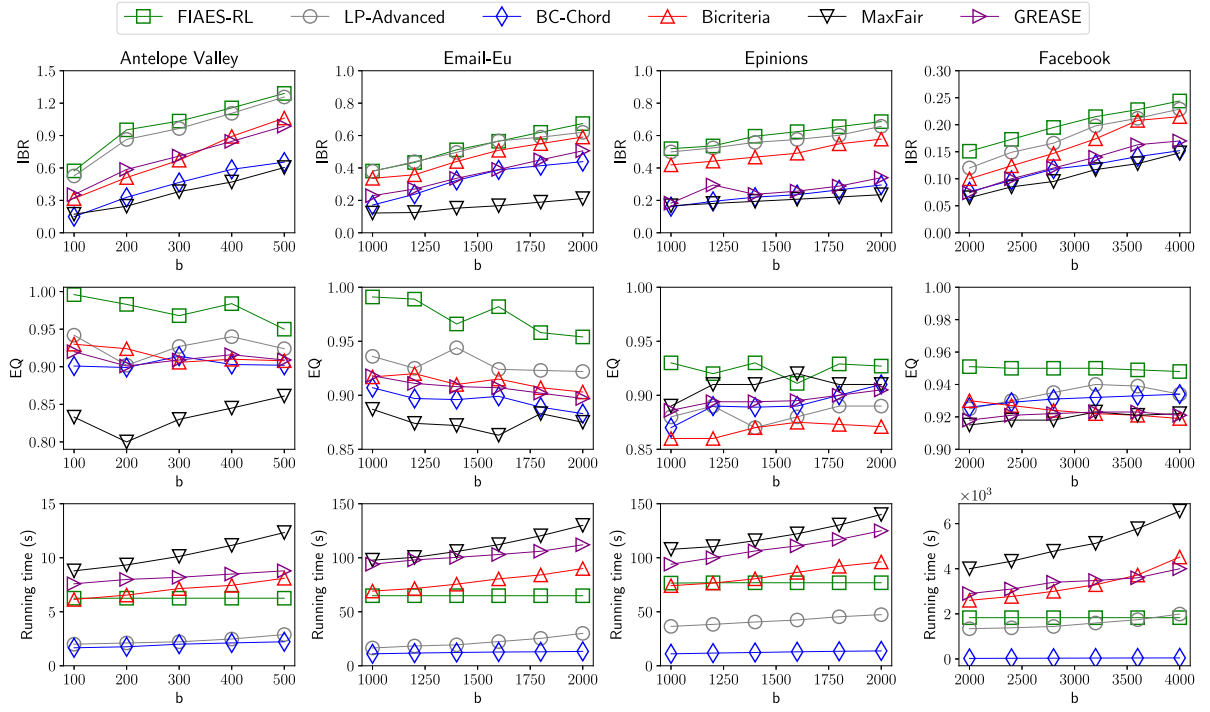


Fig. 4. Performance of different algorithms in terms of IBR, EQ, and running time across four networks by varying the budget b when $|S| = 6$.

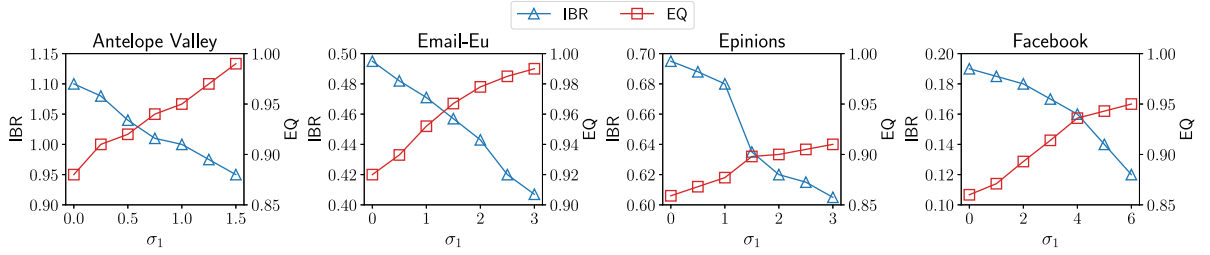


Fig. 5. IBRs and EQs of FIAES-RL by varying equity penalty coefficient σ_1 .

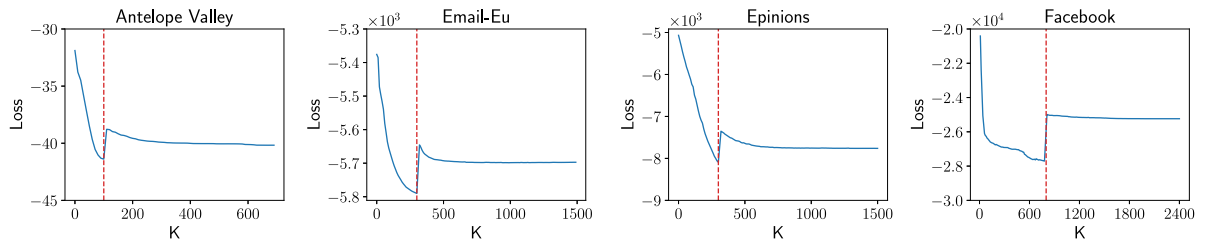


Fig. 6. Loss value $\mathcal{L}(\pi)$ of FIAES-RL over epochs when training the policy network using the default parameters outlined in Table 3.

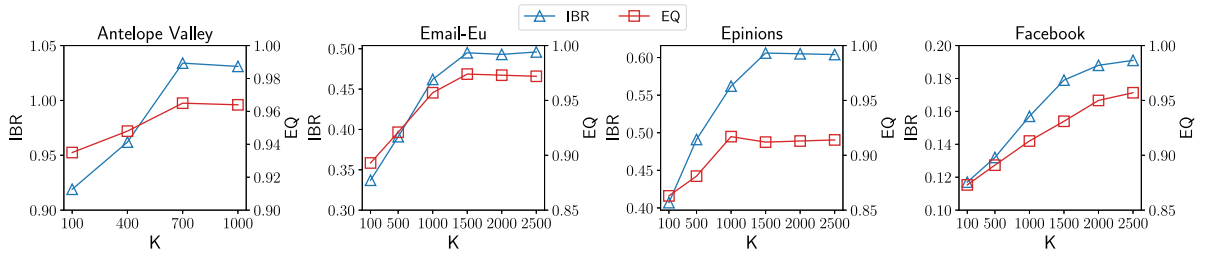


Fig. 7. IBRs and EQs of FIAES-RL with varying the number of epochs K .

Table 6

IBRs and EQs of FIAES-RL with varying the time horizon T , budget penalty coefficient σ_2 , and temperature attenuation factor ν . The best results in each setting are highlighted in **bold font**.

Parameter	Value	Antelope Valley		Email-Eu		Epinions		Facebook	
		IBR	EQ	IBR	EQ	IBR	EQ	IBR	EQ
T	2	0.567±0.041	0.884±0.035	0.243±0.022	0.904±0.021	0.462±0.025	0.877±0.023	0.131±0.012	0.922±0.015
	4	0.779±0.058	0.891±0.027	0.339±0.027	0.928±0.017	0.523±0.026	0.891±0.019	0.152±0.013	0.934±0.011
	6	1.051±0.086	0.941±0.014	0.487±0.029	0.968±0.010	0.591±0.024	0.906±0.013	0.191±0.014	0.957±0.011
	8	1.055±0.093	0.951±0.015	0.505±0.030	0.963±0.010	0.595±0.023	0.908±0.013	0.182±0.012	0.961±0.009
	10	1.049±0.094	0.960±0.013	0.495±0.027	0.974±0.009	0.606±0.022	0.912±0.012	0.187±0.013	0.960±0.010
	15	1.043±0.083	0.965±0.012	0.497±0.028	0.968±0.011	0.607±0.024	0.909±0.011	0.195±0.014	0.944±0.011
σ_2	0.1	0.879±0.067	0.924±0.019	0.381±0.021	0.922±0.018	0.419±0.020	0.873±0.017	0.097±0.009	0.918±0.018
	0.5	1.043±0.081	0.965±0.012	0.434±0.025	0.958±0.012	0.576±0.021	0.917±0.012	0.118±0.011	0.927±0.009
	1.0	1.048±0.094	0.954±0.014	0.495±0.027	0.974±0.009	0.606±0.022	0.912±0.012	0.145±0.012	0.948±0.010
	1.5	1.050±0.091	0.960±0.013	0.482±0.026	0.976±0.009	0.595±0.023	0.915±0.011	0.182±0.012	0.951±0.010
	2.0	1.055±0.093	0.953±0.014	0.503±0.028	0.960±0.010	0.602±0.021	0.917±0.011	0.191±0.014	0.957±0.011
ν	0.90	0.263±0.031	0.884±0.027	0.323±0.020	0.875±0.020	0.245±0.019	0.872±0.016	0.127±0.011	0.915±0.019
	0.95	0.986±0.071	0.936±0.018	0.428±0.025	0.903±0.016	0.435±0.022	0.874±0.014	0.149±0.012	0.933±0.010
	0.99	1.018±0.078	0.941±0.014	0.459±0.026	0.922±0.014	0.578±0.022	0.905±0.013	0.191±0.014	0.957±0.011
	0.995	1.049±0.094	0.957±0.013	0.495±0.027	0.974±0.009	0.606±0.022	0.912±0.012	0.186±0.013	0.960±0.010
	0.999	1.043±0.083	0.965±0.012	0.493±0.027	0.970±0.010	0.598±0.021	0.915±0.011	0.189±0.013	0.952±0.010

values of K_{es} and K_{ts} proportionally to K . Generally, when the value of K is too small, FIAES-RL cannot achieve convergence, leading to inferior performance. As the value of K increases, the performance of FIAES-RL first increases and then gradually becomes stable. Since the running time of FIAES-RL increases nearly linearly with K , we typically pick the smallest K with stable performance on each dataset. On the *Facebook* dataset, we do not use a larger value of K than 2400 by default due to the efficiency issue.

Effects of Time Horizon T , Budget Penalty Coefficient σ_2 , and Temperature Attenuation Factor ν : Table 6 demonstrates the performance of FIAES-RL in terms of IBR and EQ by varying the time horizon T , budget penalty coefficient σ_2 , and temperature attenuation factor ν across four datasets. As can be seen, when the value of T is too small, the MRP does not accurately estimate the influence spread and thus FIAES-RL cannot achieve good performance. But the gain of increasing T becomes marginal when $T \geq 6$. Since the running time of FIAES-RL grows with T , we pick an appropriate value of T for each dataset to balance between the effectiveness and efficiency of FIAES-RL. Then, when the value of σ_2 is excessively small, the policy returned by FIAES-RL can violate the budget constraint b . To provide a set of edge suggestions of size at most b , we remove edges with the smallest weights in E_π , thus reducing the performance of FIAES-RL. Similarly, a small value of ν leads to a rapid discretization process, significantly impeding the performance of FIAES-RL. However, when the values of σ_2 and ν are large enough, their effects on the performance of FIAES-RL are insignificant. In addition, their values also hardly impact the time efficiency of FIAES-RL. Therefore, we choose large enough values of σ_2 and ν on each dataset to ensure stable performance of FIAES-RL.

6. Conclusion

In this paper, we formulated the problem of *Fair Information Access via Edge Suggestion* (FIAES) on social networks and showed its NP-hardness and inapproximability. Then, we proposed a reinforcement learning-based algorithm, FIAES-RL, for this problem. Through extensive experiments on four synthetic and real-world datasets, we demonstrated that FIAES-RL significantly outperforms state-of-the-art baselines for fairness-aware edge suggestion in terms of boosting the influence spread and reducing the information gap. In future work, we plan to extend our framework to incorporate different notions of fairness at both the group and individual levels. We will also attempt to improve the scalability of our approach on larger networks.

CRedit authorship contribution statement

Fangzheng Wang: Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Methodology. **Yanhao Wang:** Writing – review & editing, Supervision, Project administration, Investigation, Funding acquisition, Conceptualization. **Jingjing Wang:** Writing – review & editing, Validation, Project administration. **Minghao Zhao:** Writing – review & editing, Project administration, Funding acquisition.

Declaration of competing 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.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China under Grant Number 62202169 and the Shanghai Sailing Program 23YF1410600.

Data availability

The datasets are downloaded from <https://snap.stanford.edu/data>. The code is publicly available at <https://github.com/fzzf09/FIAES-RL>.

References

- [1] C.L. Coyle, H. Vaughn, Social networking: Communication revolution or evolution? Bell Labs Tech. J. 13 (2) (2008) 13–17.
- [2] E. Bakshy, D. Eckles, R. Yan, I. Rosenn, Social influence in social advertising: evidence from field experiments, in: Proceedings of the 13th ACM Conference on Electronic Commerce, 2012, pp. 146–161.
- [3] K. Feng, G. Cong, S.S. Bhowmick, S. Ma, In search of influential event organizers in online social networks, in: Proceedings of the 2014 ACM SIGMOD International Conference on Management of Data, 2014, pp. 63–74.
- [4] D. Kempe, J. Kleinberg, É. 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, 2003, pp. 137–146.
- [5] A. Tsang, B. Wilder, E. Rice, M. Tambe, Y. Zick, Group-fairness in influence maximization, in: Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, 2019, pp. 5997–6005.
- [6] B. Fish, A. Bashardoust, D. Boyd, S. Friedler, C. Scheidegger, S. Venkatasubramanian, Gaps in information access in social networks? in: The World Wide Web Conference, 2019, pp. 480–490.
- [7] Z.S. Jalali, W. Wang, M. Kim, H. Raghavan, S. Soundarajan, On the information unfairness of social networks, in: Proceedings of the 2020 SIAM International Conference on Data Mining, 2020, pp. 613–521.

- [8] A. Saxena, G. Fletcher, M. Pechenizkiy, FairSNA: Algorithmic fairness in social network analysis, *ACM Comput. Surv.* 56 (8) (2024) 213:1–213:45.
- [9] Z.K. Yaseen, Y. Marwan, The influence of social media on recruitment and selection process in SMEs, *J. Small Bus. Enterpr. Dev.* 4 (2) (2016) 21–27.
- [10] L. Beaman, A. Dillon, Diffusion of agricultural information within social networks: Evidence on gender inequalities from Mali, *J. Dev. Econ.* 133 (2018) 147–161.
- [11] G. Farnad, B. Babaki, M. Gendreau, A unifying framework for fairness-aware influence maximization, in: *Companion Proceedings of the Web Conference 2020*, 2020, pp. 714–722.
- [12] A. Rahmattalabi, S. Jabbari, H. Lakkaraju, P. Vayanos, M. Izenberg, R. Brown, E. Rice, M. Tambe, Fair influence maximization: A welfare optimization approach, *Proc. AAAI Conf. Artif. Intell.* 35 (13) (2021) 11630–11638.
- [13] J. Ali, M. Babaei, A. Chakraborty, B. Mirzasoleiman, K.P. Gummadi, A. Singla, On the fairness of time-critical influence maximization in social networks, *IEEE Trans. Knowl. Data Eng.* 35 (3) (2023) 2875–2886.
- [14] R. Becker, G. D'Angelo, S. Ghobadi, H. Gilbert, Fairness in influence maximization through randomization, *J. Artificial Intelligence Res.* 73 (2022) 1251–1283.
- [15] Y. Feng, A. Patel, B. Cautis, H. Vahabi, Influence maximization with fairness at scale, in: *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2023, pp. 4046–4055.
- [16] X. Rui, Z. Wang, J. Zhao, L. Sun, W. Chen, Scalable fair influence maximization, *Adv. Neural Inf. Process. Syst.* 36 (2023) 66675–66691.
- [17] I.P. Swift, S. Ebrahimi, A. Nova, A. Asudeh, Maximizing fair content spread via edge suggestion in social networks, *Proc. VLDB Endow.* 15 (11) (2022) 2692–2705.
- [18] R. Becker, G. D'Angelo, S. Ghobadi, Improving fairness in information exposure by adding links, *Proc. AAAI Conf. Artif. Intell.* 37 (12) (2023) 14119–14126.
- [19] A. Bashardoust, S. Friedler, C. Scheidegger, B.D. Sullivan, S. Venkatasubramanian, Reducing access disparities in networks using edge augmentation, in: *Proceedings of the 2023 ACM Conference on Fairness, Accountability, and Transparency*, 2023, pp. 1635–1651.
- [20] Y. Mashayekhi, B. Kang, J. Lijffijt, T.D. Bie, GREASE: Graph imbalance reduction by adding sets of edges, *IEEE Trans. Knowl. Data Eng.* 36 (4) (2024) 1611–1623.
- [21] A. Bhaskara, A. Crane, M.M.H.U. Mazumder, B.D. Sullivan, P. Yalamanchili, Optimizing information access in networks via edge augmentation, 2024, *CoRR* abs/2407.02624.
- [22] G.S. Ramachandran, I. Brugere, L.R. Varshney, C. Xiong, GAEA: Graph augmentation for equitable access via reinforcement learning, in: *Proceedings of the 2021 AAAI/ACM Conference on AI, Ethics, and Society*, 2021, pp. 884–894.
- [23] M. Nie, D. Chen, D. Wang, Reinforcement learning on graphs: A survey, *IEEE Trans. Emerg. Top. Comput. Intell.* 7 (4) (2023) 1065–1082.
- [24] Y. Li, J. Fan, Y. Wang, K. Tan, Influence maximization on social graphs: A survey, *IEEE Trans. Knowl. Data Eng.* 30 (10) (2018) 1852–1872.
- [25] S. Banerjee, M. Jenamani, D.K. Pratihari, A survey on influence maximization in a social network, *Knowl. Inf. Syst.* 62 (9) (2020) 3417–3455.
- [26] Y. Li, H. Gao, Y. Gao, J. Guo, W. Wu, A survey on influence maximization: From an ML-based combinatorial optimization, *ACM Trans. Knowl. Discov. Data* 17 (9) (2023) 133:1–133:50.
- [27] S.-C. Lin, S.-D. Lin, M.-S. Chen, A learning-based framework to handle multi-round multi-party influence maximization on social networks, in: *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2015, pp. 695–704.
- [28] K. Ali, C.-Y. Wang, M.-Y. Yeh, 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*, 2020, pp. 196–203.
- [29] K. Ali, C. Wang, Y. Chen, Leveraging transfer learning in reinforcement learning to tackle competitive influence maximization, *Knowl. Inf. Syst.* 64 (8) (2022) 2059–2090.
- [30] S. Tian, S. Mo, L. Wang, Z. Peng, Deep reinforcement learning-based approach to tackle topic-aware influence maximization, *Data Sci. Eng.* 5 (1) (2020) 1–11.
- [31] H. Kamarthi, P. Vijayan, B. Wilder, B. Ravindran, M. Tambe, Influence maximization in unknown social networks: Learning policies for effective graph sampling, in: *Proceedings of the 19th International Conference on Autonomous Agents and MultiAgent Systems*, 2020, pp. 575–583.
- [32] C. Wang, Y. Liu, X. Gao, G. Chen, A reinforcement learning model for influence maximization in social networks, in: *Proceedings of the 26th International Conference on Database Systems for Advanced Applications, Part II*, 2021, pp. 701–709.
- [33] H. Li, M. Xu, S.S. Bhowmick, S.R. Joty, C. Sun, J. Cui, PIANO: Influence maximization meets deep reinforcement learning, *IEEE Trans. Comput. Soc. Syst.* 10 (3) (2023) 1288–1300.
- [34] T. Chen, S. Yan, J. Guo, W. Wu, ToupleGDD: A fine-designed solution of influence maximization by deep reinforcement learning, *IEEE Trans. Comput. Soc. Syst.* 11 (2) (2024) 2210–2221.
- [35] J. Wang, Z. Cao, C. Xie, Y. Li, J. Liu, Z. Gao, DGN: influence maximization based on deep reinforcement learning, *J. Supercomput.* 81 (1) (2025) 130.
- [36] C. Liu, C. Fan, Z. Zhang, Finding influencers in complex networks: An effective deep reinforcement learning approach, *Comput. J.* 67 (2) (2024) 463–473.
- [37] H. Chen, W. Qiu, H.-C. Ou, B. An, M. Tambe, Contingency-aware influence maximization: A reinforcement learning approach, in: *Proceedings of the Thirty-Seventh Conference on Uncertainty in Artificial Intelligence*, 2021, pp. 1535–1545.
- [38] L. Ma, Z. Shao, X. Li, Q. Lin, J. Li, V.C.M. Leung, A.K. Nandi, Influence maximization in complex networks by using evolutionary deep reinforcement learning, *IEEE Trans. Emerg. Top. Comput. Intell.* 7 (4) (2023) 995–1009.
- [39] S. Yang, Q. Du, G. Zhu, J. Cao, L. Chen, W. Qin, Y. Wang, Balanced influence maximization in social networks based on deep reinforcement learning, *Neural Netw.* 169 (2024) 334–351.
- [40] S. Ghosh, T. Chen, W. Wu, Enhanced group influence maximization in social networks using deep reinforcement learning, *IEEE Trans. Comput. Soc. Syst.* 12 (2) (2025) 573–585.
- [41] S. Wu, W. Li, Q. Bai, GAC: A deep reinforcement learning model toward user incentivization in unknown social networks, *Knowl.-Based Syst.* 259 (2023) 110060.
- [42] J. Jiang, X. Chen, Z. Huang, X. Li, Y. Du, Deep reinforcement learning-based approach for rumor influence minimization in social networks, *Appl. Intell.* 53 (17) (2023) 20293–20310.
- [43] M. Khajehnejad, A.A. Rezaei, M. Babaei, J. Hoffmann, M. Jalili, A. Weller, Adversarial graph embeddings for fair influence maximization over social networks, in: *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence*, 2020, pp. 4306–4312.
- [44] M. Lin, L. Sun, R. Yang, X. Liu, Y. Wang, D. Li, W. Li, S. Lu, Fair influence maximization in large-scale social networks based on attribute-aware reverse influence sampling, *J. Artificial Intelligence Res.* 76 (2023) 925–957.
- [45] V. Chaoji, S. Ranu, R. Rastogi, R. Bhatt, Recommendations to boost content spread in social networks, in: *Proceedings of the 21st International Conference on World Wide Web*, 2012, pp. 529–538.
- [46] G. D'Angelo, L. Severini, Y. Velaj, Recommending links through influence maximization, *Theoret. Comput. Sci.* 764 (2019) 30–41.
- [47] F. Coró, G. D'Angelo, Y. Velaj, Link recommendation for social influence maximization, *ACM Trans. Knowl. Discov. Data* 15 (6) (2021) 94:1–94:23.
- [48] Z. Yu, C. Wang, J. Bu, X. Wang, Y. Wu, C. Chen, Friend recommendation with content spread enhancement in social networks, *Inf. Sci.* 309 (2015) 102–118.
- [49] W. Yang, J. Ma, Y. Li, R. Yan, J. Yuan, W. Wu, D. Li, Marginal gains to maximize content spread in social networks, *IEEE Trans. Comput. Soc. Syst.* 6 (3) (2019) 479–490.
- [50] L. Yu, G. Li, L. Yuan, Maximizing boosted influence spread with edge addition in online social networks, *ACM/IMS Trans. Data Sci.* 1 (2) (2020) 9:1–9:21.
- [51] X. Chen, Y. Song, J. Tang, Link recommendation to augment influence diffusion with provable guarantees, in: *Proceedings of the ACM Web Conference 2024*, 2024, pp. 2509–2518.
- [52] W. Chen, C. Wang, Y. 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*, 2010, pp. 1029–1038.
- [53] C. Borgs, M. Brautbar, J.T. Chayes, B. Lucier, Maximizing social influence in nearly optimal time, in: *Proceedings of the Twenty-Fifth Annual ACM-SIAM Symposium on Discrete Algorithms*, 2014, pp. 946–957.
- [54] M.R. Garey, D.S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*, W.H. Freeman, New York, NY, USA, 1979.
- [55] N. Srivastava, G.E. Hinton, A. Krizhevsky, I. Sutskever, R. Salakhutdinov, Dropout: a simple way to prevent neural networks from overfitting, *J. Mach. Learn. Res.* 15 (1) (2014) 1929–1958.
- [56] E. Jang, S. Gu, B. Poole, Categorical reparameterization with Gumbel-Softmax, in: *The 5th International Conference on Learning Representations*, 2017, pp. 1–12.
- [57] J. Leskovec, A. Krevl, *SNAP Datasets: Stanford large network dataset collection*, 2014, <http://snap.stanford.edu/data>.
- [58] V.D. Blondel, J.-L. Guillaume, R. Lambiotte, E. Lefebvre, Fast unfolding of communities in large networks, *J. Stat. Mech. Theory Exp.* 2008 (10) (2008) P10008.