RELINK: Edge Activation for Closed Network Influence Maximization via Deep Reinforcement Learning

Shivvrat Arya
shivvrat.arya@njit.edu
New Jersey Institute of Technology
Newark, United States

Smita Ghosh
sghosh3@scu.edu
Santa Clara University
Santa Clara, United States

Bryan Maruyama
bmaruyama@scu.edu
Santa Clara University
Santa Clara, United States

Venkatesh Srinivasan
vsrinivasan4@scu.edu
Santa Clara University
Santa Clara, United States

Abstract

Influence Maximization aims to select a subset of elements in a social network to maximize information spread under a diffusion model. While existing work primarily focuses on selecting influential *nodes*, these approaches assume unrestricted message propagation—an assumption that fails in *closed social networks*, where content visibility is constrained and node-level activations may be infeasible. Motivated by the growing adoption of privacy-focused platforms such as Signal, Discord, Instagram, and Slack, our work addresses the following fundamental question: *How can we learn effective edge activation strategies for influence maximization in closed networks?* To answer this question we introduce **Re**inforcement **L**earning for L**ink** Activation (RELINK), the first DRL framework for *edge-level* influence maximization in privacy-constrained networks. It models edge selection as a Markov Decision Process, where the agent learns to activate edges under budget constraints. Unlike prior node-based DRL methods, RELINK uses an edge-centric Q-learning approach that accounts for structural constraints and constrained information propagation. Our framework combines a rich node embedding pipeline with an edge-aware aggregation module. The agent is trained using an n-step Double DQN objective, guided by dense reward signals that capture marginal gains in influence spread. Extensive experiments on real-world networks show that RELINK consistently outperforms existing edge-based methods, achieving up to 15% higher influence spread and improved scalability across diverse settings.

CCS Concepts

• **Human-centered computing** → **Social network analysis**; • **Information systems** → **Social networks**; • **Networks** → **Online social networks**.

Keywords

Influence Maximization, Closed Networks, Edge Selection, Deep Reinforcement Learning, Social Network Analysis

CIKM '25, Seoul, Republic of Korea

ACM ISBN 979-8-4007-2040-6/2025/11
https://doi.org/10.1145/3746252.3761006

ACM Reference Format:
Shivvrat Arya, Smita Ghosh, Bryan Maruyama, and Venkatesh Srinivasan. 2025. RELINK: Edge Activation for Closed Network Influence Maximization via Deep Reinforcement Learning. In *Proceedings of the 34th ACM International Conference on Information and Knowledge Management (CIKM '25), November 10–14, 2025, Seoul, Republic of Korea.* ACM, New York, NY, USA, 12 pages. https://doi.org/10.1145/3746252.3761006

1 Introduction

The widespread adoption of digital social platforms has made information diffusion a central challenge in network science, marketing, and policy-making. Influence Maximization (IM)—the task of selecting a subset of network elements to maximize information spread under a diffusion model—has been extensively studied over the past two decades. Classical approaches [49] focus on identifying a small set of influential *nodes* (seeds) to initiate diffusion. However, these strategies often rest on strong assumptions—most notably, that content shared by a user is visible to all their neighbors—overlooking scenarios where senders can selectively restrict post visibility. Such assumptions fail to capture the increasingly constrained and intentional nature of real-world information sharing.

This trend is exemplified by the rise of private and semi-private platforms, often referred to as *closed social networks.* Applications such as Signal, GroupMe, and Discord enable users to create invitation-only groups for targeted conversations [3, 29, 42]. Instagram's Close Friends and private stories features allow selective sharing [15], while platforms such as Slack and Nextdoor support controlled communication within trusted circles [69, 72, 86, 87]. In response, major platforms—Meta, Microsoft (LinkedIn), and X (formerly Twitter)—have introduced privacy-centric features such as private groups, audience controls, and restricted visibility options [30, 64]. These developments show a broad shift toward more intentional and privacy-preserving modes of online interaction.

This paradigm shift necessitates a reassessment of classical IM strategies. Huang et al. [46] addressed this need by introducing and formalizing Influence Maximization in Closed Social Networks (IM-CSN), which recasts influence maximization under propagation constraints as an edge activation problem. Formally, IM-CSN seeks to select a budget-limited subset of edges whose activation maximizes the expected influence spread from a given seed set. This formulation accurately captures the dynamics of *closed social networks*, where the set of active nodes remains fixed and information flow is governed by user- or platform-level access constraints—factors not

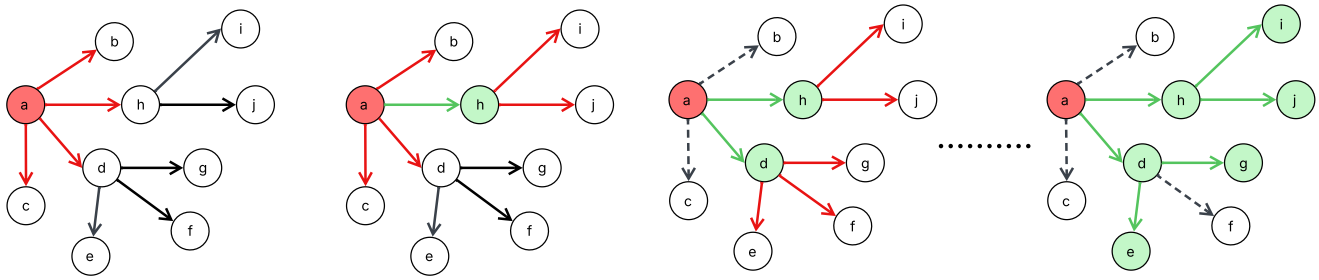


Figure 1: Influence Maximization in Closed Social Networks (IM-CSN): Edge selection for $k = 2$. Red: seed nodes and candidate edges; green: activated nodes and edges; dashed: pruned edges due to budget limits.

accounted for in traditional influence maximization frameworks designed for open networks [49]. Consequently, classical node-selection strategies are ill-suited for IM-CSN.

Example 1. *Figure 1 presents a toy example of edge selection in a closed network. Each node may recommend up to $k = 2$ neighbors to propagate information, and the seed set is $S = \{a\}$. Let C denote candidate edges and $\mathcal{F} \subseteq V$ the active nodes below their out-degree limit. At $t = 0$, $C = E(a)$ (red arrows, panel 1) and $\mathcal{F} = \{a\}$, where $E(a)$ is a's outgoing edges. While $C \neq \emptyset$, the algorithm selects the highest-priority edge $e = (u, v) \in C$, activates it (green), and updates $\mathcal{F} \leftarrow \mathcal{F} \cup \{v\}$, adding $E(v)$ to C. Panels 2–3 show iterations: activating (a, h) adds (h, i), (h, j); later, selecting (a, d) removes remaining outgoing edges of a ((a, b), (a, c)) from C due to the out-degree constraint. Panel 4 shows the final set of activated edges.*

Huang et al. [46] established that IM-CSN is NP-Hard, underscoring its intrinsic computational difficulty due to the combinatorial explosion in the space of possible edge insertions. Furthermore, their work showed that greedy strategies based on marginal influence gain are ineffective: the objective function is non-submodular, and computing marginal gains under the classical Independent Cascade (IC) model [49] is computationally prohibitive. To address these challenges, they introduced PSNA—the first algorithm specifically designed for IM-CSN —which augments the diffusion network using submodularity-preserving lower bounds on influence spread. While this surrogate objective enables tractable optimization, it introduces limitations: PSNA optimizes over approximations rather than true influence values, which can lead to suboptimal solutions. Moreover, computing these bounds is costly, especially in large networks, causing substantial runtime overhead.

To overcome these issues, we propose **Re**inforcement **L**earning for L**ink** Activation (RELINK), the first deep reinforcement learning (DRL) framework designed for edge-level influence maximization in closed social networks. Our approach leverages DRL to enable sequential, influence-aware edge activation—an essential capability for scalable decision-making in closed-network settings. While *exact* marginal gains are costly at test time [46], our method computes them during training, leveraging greater resources for richer learning signals. This allows the model to approximate true marginal gains, enabling more effective edge selection without computing them at inference time.

We formulate RELINK as a Markov Decision Process (MDP), where states correspond to partial cascades and actions to edge activations. The agent is trained using deep Q-learning with an n-step Double DQN objective, allowing end-to-end optimization under budget and propagation constraints. Inspired by the success of DRL in node-level influence maximization—where methods such as PIANO [59] and DREIM [62] have outperformed state-of-the-art approximation algorithms—our framework brings these advantages to the edge-level setting. The agent receives dense rewards based on *exact* marginal influence gains and employs an ε-greedy exploration policy, enabling it to learn effective strategies in the presence of non-submodular, long-range influence dynamics.

Our contributions are summarized as follows:

- To the best of our knowledge, we introduce the first end-to-end DRL-based solution for IM-CSN, addressing the unique challenges posed by edge-level activations under closed network constraints.
- We design a node embedding and aggregation pipeline that captures topological and influence-relevant features, and integrates multi-hop signals using an edge-aware module with gated updates.
- We demonstrate strong performance under tight budget constraints by learning to prioritize high-impact edges early in the decision process.
- We show that our method consistently outperforms state-of-the-art baselines in influence spread and scales better on large real-world networks.

2 Related Work

2.1 Traditional Influence Maximization

Influence maximization (IM) was introduced by Kempe et al. [49] as a combinatorial optimization problem under the Independent Cascade (IC) and Linear Threshold (LT) models. Subsequent research focused on node activation, developing scalable algorithms with approximation guarantees for unconstrained diffusion [4, 5, 8, 9, 11–13, 16, 18, 21, 34, 40, 41, 44, 45, 48, 61, 63, 66, 80, 82, 83, 88, 90]. This assumption often fails in modern networks, where privacy and limited propagation undermine node-centric methods. These limitations motivated edge-based IM, where modifying connections is more feasible than activating users [14, 25, 26, 46, 50, 93, 94]. Edge-centric methods leverage submodularity for IC and LT [26, 50], while addressing constraints such as per-node edge budgets [14] or candidate edge restrictions [93, 94]. However, these approaches assume unrestricted propagation, either through existing edges or newly inserted ones, and thus fail in *closed networks*, where diffusion is limited to a chosen subset of edges.

2.2 Closed-Network Influence Maximization

Huang et al. [46] formulated Influence Maximization in Closed Social Networks (IM-CSN), where diffusion is restricted by user- or platform-level access controls. This shifts the focus from node selection to edge activation under structural constraints, aiming to maximize influence within subgraphs—better reflecting privacy-aware platforms. The problem entails selecting a limited subset of original edges to boost seed influence, addressed through a two-stage method of topology expansion and influence-guided edge ranking, which yields strong empirical performance.

2.3 DRL for Traditional IM

To address limitations of approximation methods, recent work has explored Deep Reinforcement Learning (DRL) for IM, formulating it as a sequential decision-making task. PIANO [59] uses Deep Q-learning to select influential nodes and performs well in open networks, but its node-centric actions and reliance on unrestricted propagation make it unsuitable for IM-CSN. Other DRL-based approaches have been proposed for other IM variants, including group IM [33], rumor containment [54], competitive influence via community detection [23], and trust-aware reward shaping [53]. Yet they share these key limitations: (1) they define the action space over nodes rather than edges, and (2) their reward functions do not extend to edge activation scenarios. While DRIM [23] introduces quota-based policies, it operates at coarse community granularity, lacking support for fine-grained edge-level actions.

2.4 Motivation

Despite growing interest in DRL for IM, existing methods have not addressed edge activation in closed networks. The absence of adaptable, learning-driven mechanisms for edge-level decision-making under partial propagation represents a significant gap in the literature. To the best of our knowledge, this work is the first to formulate edge-level influence maximization in closed social networks as a deep reinforcement learning (DRL) problem. Prior efforts have either focused on node selection in open networks [59, 60] or used approximation algorithms for edge insertion in closed networks [46], but these either ignore edge-level decisions [59, 60] or lack a learnable, adaptive decision-making framework [46].

In contrast, our proposed method **Re**inforcement **L**earning for L**ink** Activation (RELINK) formalizes edge activation as the primary decision variable within a Markov Decision Process (MDP), enabling sequential budget-constrained optimization with controlled information propagation. By learning edge-level policies, RELINK addresses key limitations of prior work and extends DRL-based influence maximization to emerging privacy-sensitive social networks.

3 Preliminaries

In this section, we present the foundational concepts underlying edge-based influence maximization in closed network settings.

A **social network** is a directed graph $G = (V, E)$, where V is the set of nodes (users), and $E \subseteq V \times V$ is the set of edges representing connection links. Each edge $(u, v) \in E$ has a weight $w_{uv} \in [0, 1]$ that quantifies the probability of influence from node u to node v.

A **diffusion model** ψ defines the stochastic process by which influence propagates through a directed graph $G = (V, E)$. A widely used model is the Independent Cascade (IC) model [49], in which each directed edge (u, v) is associated with an independent activation probability $w_{u,v}$. The cascade starts with a set of seed nodes S active at time $t = 0$ and progresses in discrete rounds. In round $t \geq 1$, every node u activated at $t - 1$ receives exactly one chance to trigger each inactive out-neighbour v with probability $w_{u,v}$; successful attempts activate v in round t. A newly activated node joins the process in the next round and follows the same one-shot rule. Propagation terminates when no new activations occur. The *influence spread*, denoted by $\mathcal{I}_{(V,E)}(S)$ or equivalently $\mathcal{I}_G(S)$, is defined as the expected number of nodes activated at the end of the diffusion process governed by ψ, starting from the seed set S.

Definition 1. ***Marginal Influence Gain:*** *Given a subgraph $G' = (V', E')$ of the graph $G = (V, E)$, and a fixed seed set $S \subseteq V$, the* marginal influence gain *of adding an edge $e \in E \setminus E'$ to G' is defined as:*

$$\Delta(e; G', S) := \mathcal{I}_{(V',E'\cup\{e\})}(S) - \mathcal{I}_{(V',E')}(S),$$

where $\mathcal{I}_{(V',E')}(S)$ denotes the expected number of activated nodes from S under the diffusion model restricted to G'.

Definition 2. *Given a directed graph $G = (V, E)$ and an integer $k \geq 1$, a* **k-subnetwork** *is a subgraph $G_k = (V_k, E_k)$ such that $V_k \subseteq V$, $E_k \subseteq E$, and $\deg^+_{G_k}(v) \leq k$ for all $v \in V_k$, where $\deg^+_{G_k}(v)$ denotes the out-degree of v in G_k.*

For example, in the fourth panel of Figure 1, the subnetwork consisting of solid green edges forms a 2-subnetwork of the directed graph shown in the first panel.

Definition 3. **Influence Maximization in Closed Social Networks (IM-CSN)[46]**: *Given a directed graph $G = (V, E)$, a seed set $S \subseteq V$, a diffusion model, and an integer $k \geq 1$, the objective is to identify the optimal diffusion k-subnetwork $G^*_k = (V^*_k, E^*_k)$ that maximizes the expected influence spread from the seed set S. Formally,*

$$G^*_k = \underset{G_k \in \mathcal{P}(G)}{\arg\max}\ \mathcal{I}_{G_k}(S),$$

where $\mathcal{P}(G)$ denotes the set of all possible k-subnetworks of G.

We formulate the IM-CSN problem as a sequential decision-making task suitable for DRL. Given a directed graph $G = (V, E)$, a fixed seed set $S \subseteq V$, and an out-degree budget k, the agent interacts with the environment over discrete time steps $t = 1, \ldots, T$. At each step t, the agent begins with the current subgraph $G^{(t)} = (V^{(t)}, E^{(t)})$. It then selects an edge $a^{(t)} = (u, v) \in E \setminus E^{(t)}$ to activate, such that the out-degree constraint is preserved. The selected edge is added to the current edge set, updating the subgraph to $G^{(t+1)} = (V^{(t+1)}, E^{(t+1)})$, where $V^{(t+1)} = V^{(t)} \cup \{v\}$ and $E^{(t+1)} = E^{(t)} \cup \{(u, v)\}$. After each edge activation, the agent receives a reward equal to the marginal influence gain, $\Delta(a^{(t)}; G^{(t)}, S)$, with respect to the seed set S (Definition 1).

The episode terminates when no further valid actions remain, producing a k-subnetwork $G_k = (V_k, E_k)$. The objective is to learn a policy π_θ that maximizes the expected cumulative reward:

$$\mathbb{E}_{\pi_\theta}\left[\sum_{t=1}^{T} \Delta(a^{(t)}; G^{(t)}, S)\right]$$

The reward sequence forms a telescoping sum:

$$\begin{aligned}&\left(\mathcal{I}_{G^{(1)}}(S) - \mathcal{I}_{G^{(0)}}(S)\right) + \left(\mathcal{I}_{G^{(2)}}(S) - \mathcal{I}_{G^{(1)}}(S)\right) + \cdots \\ &+ \left(\mathcal{I}_{G^{(T)}}(S) - \mathcal{I}_{G^{(T-1)}}(S)\right) = \mathcal{I}_{G^{(T)}}(S) - \mathcal{I}_{G^{(0)}}(S),\end{aligned}$$

where $G^{(0)} = (S, \emptyset)$. Since $\mathcal{I}_{G^{(0)}}(S)$ is deterministic for a given S and does not depend on π_θ, maximizing the expected cumulative reward is equivalent to maximizing $\mathbb{E}_{\pi_\theta}[\mathcal{I}_{G^{(T)}}(S)]$, matching the IM-CSN objective (Definition 3).

This formulation captures key constraints of the closed network setting—specifically, node-level degree budgets and restricted influence propagation—and is well suited for DRL approaches that leverage sequential feedback. In the following sections, we elaborate on the DRL framework used to solve this problem.

4 Influence-Aware Embeddings

We introduce an influence-aware node embedding framework for IM-CSN. Our embedding integrates five feature classes that capture structural, relational, and probabilistic signals indicative of influence potential. By combining high-order graph features, community structure, and diffusion heuristics, it encodes both latent pathways and activation likelihoods in a scalable representation.

We first describe each of the five feature classes used in constructing the initial node representations, followed by an edge-aware aggregation module that propagates multi-hop influence signals through learned parameters.

4.1 Initial Node Representation

We construct high-dimensional node embeddings by aggregating features across five distinct feature classes:

(1) Community and Structural Role Features: We detect communities using the Louvain algorithm [10] and encode each node's community ID along with the size of its community. Structural features used in the embedding include approximate betweenness centrality [75] (identifying nodes on shortest paths), degree (connectivity), bridge score [92] (capturing cross-community connections), and local clustering coefficient [79]. Together, these features characterize a node's position within and across community boundaries—key determinants of influence propagation [17].

(2) Path-Based Features: We compute three role-based features per node: (i) the number of 2-hop neighbors, capturing local reachability; (ii) average path diversity, reflecting redundancy in influence pathways; and (iii) an RMPP-based feature [14] that approximates the most probable paths. These features capture influence propagation potential beyond immediate neighbors.

(3) Edge Weight Statistics: For each node v, we compute the mean (μ_v), maximum ($\max_v$), and variance (σ_v^2) of its outgoing edge weights. These statistics capture heterogeneity in influence strength across outgoing edges and are essential for modeling diffusion dynamics.

(4) Seed-Based Features: We encode three features: a binary indicator of whether the node is a seed, a binary indicator of its activation status, and the shortest-path distance to the nearest seed. These features provide localized context about proximity to influence sources and recent activation events.

(5) Node2Vec Embedding: To encode global structural similarity, we incorporate Node2Vec [36] embeddings computed over the weighted adjacency matrix using biased random walks.

To ensure a compact representation, we apply Singular Value Decomposition (SVD) [52] to reduce the final embedding ($\mathbf{h}$) to a fixed dimension d across all graphs, retaining the most informative components.

This embedding compactly encodes both structural context and influence dynamics. In contrast to prior work [47, 59, 60, 62], which often focuses on isolated feature types—such as basic topological descriptors, temporal patterns, or GNN-derived embeddings—we propose a unified representation that integrates community structure, structural roles, and path-based dynamics. Structural role features offer coarse-grained insights into diffusion pathways, while edge activation statistics capture fine-grained variability in influence strength. Seed-based features extend local context beyond binary seed indicators [59], and Node2Vec embeddings complement both structural and temporal cues used in prior approaches [47, 60]. Collectively, these features produce node embeddings that integrate graph topology with diffusion semantics, thereby overcoming key limitations of prior approaches.

4.2 Edge-Aware Node Embedding Aggregation

To generate aggregated embeddings for each edge $(i, j) \in E$, we first compute an edge relevance score α_{ij} based on the initial node embeddings $\mathbf{h}_i, \mathbf{h}_j \in \mathbb{R}^d$ and a trainable projection vector $\mathbf{a} \in \mathbb{R}^{2d}$:

$$\alpha_{ij} = \sigma\left(\mathbf{a}^\top [\mathbf{h}_i \| \mathbf{h}_j] \cdot w_{ij}\right),$$

where $[\cdot\|\cdot]$ denotes vector concatenation and $\sigma(\cdot)$ denotes the sigmoid function. The aggregated messages for node i are computed as:

$$\mathbf{m}_i^{(1)} = \sum_{j\in\mathcal{N}_i} \alpha_{ij} \cdot \mathbf{h}_j, \quad \mathbf{m}_i^{(2)} = \sum_{j\in\mathcal{N}_i^{(2)}} (A^2)_{ij} \cdot \mathbf{h}_j,$$

where

$$\mathcal{N}_i = \{j \in V \mid (i, j) \in E\}, \quad \mathcal{N}_i^{(2)} = \{k \in V \mid (\mathbf{A}^2)_{ik} > 0\}.$$

Here, $\mathbf{A}$ is the adjacency matrix of the G, and $\mathbf{A}^2$ encodes weighted two-hop paths via intermediate nodes.

The aggregated messages are transformed using dimensionality-preserving multi-layer perceptrons, α_1 and α_2:

$$\tilde{\mathbf{m}}_i^{(1)} = \alpha_1(\mathbf{m}_i^{(1)}), \qquad \tilde{\mathbf{m}}_i^{(2)} = \alpha_2(\mathbf{m}_i^{(2)}).$$

Node embeddings are updated using a gating mechanism that balances the original and aggregated representations:

$$\mathbf{g}_i = \sigma\left(\mathbf{W}_g\left[\mathbf{h}_i \,\|\, \left(\tilde{\mathbf{m}}_i^{(1)} + \tilde{\mathbf{m}}_i^{(2)}\right)\right]\right),$$

$$\mathbf{h}_i' = \mathbf{g}_i \odot \mathbf{h}_i + (1 - \mathbf{g}_i) \odot \left(\tilde{\mathbf{m}}_i^{(1)} + \tilde{\mathbf{m}}_i^{(2)}\right),$$

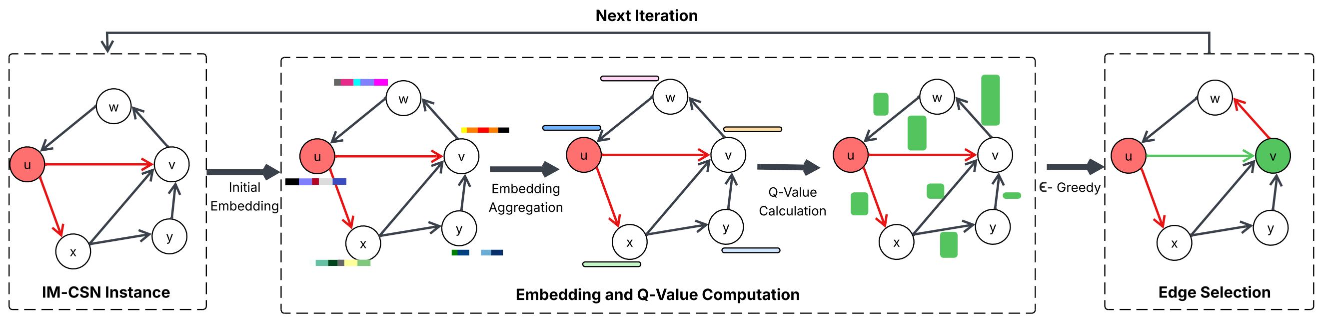


Figure 2: Overview of the RELINK framework. Given a IM-CSN instance, initial node embeddings are generated and aggregated to compute edge-level Q-values. Guided by the k-subnetwork structural constraint, the edge with the highest Q-value is selected in each iteration, and the process repeats until no candidate edges remain.

where $\odot$ denotes element-wise multiplication, and $\mathbf{W}_g \in \mathbb{R}^{d\times 2d}$ is a learnable weight matrix.

To obtain the final representation, we add a degree-normalized residual connection:

$$\mathbf{h}_i^f = \mathbf{h}_i' + \frac{1}{\sqrt{\deg(i)}} \cdot \alpha_s(\mathbf{h}_i),$$

where α_s is a dimensionality-preserving MLP and $\deg(i)$ denotes the degree of node i in the graph.

This edge-aware aggregation approach enables the model to incorporate both local and second-order structural signals, while prioritizing informative connections. By jointly leveraging multi-hop message passing, and gated integration with residual pathways, the resulting node embeddings are well-suited for capturing fine-grained influence dynamics and structural context—key properties for IM-CSN.

5 Learning Influence-Maximizing Subnetworks via Deep Q-Learning

We present RELINK (Figure 2), a deep reinforcement learning (DRL) framework that models the IM-CSN problem as a Markov Decision Process (MDP). The agent learns to sequentially construct an influence-maximizing k-subnetwork by optimizing a Q-function over the set of edges. This Q-function guides the construction of the subgraph $G_k = (V_k, E_k)$. We now describe the MDP formulation, Q-value computation, training algorithm, and inference-time decision procedure in detail.

5.1 Problem Formulation as an MDP

We formulate IM-CSN as a finite-horizon MDP and solve it using a DRL agent. The agent incrementally constructs a subgraph by selecting edges that maximize cumulative influence, subject to a k-outgoing edge constraint per node (Definition 2). The MDP is defined by the tuple $(\mathcal{S}, \mathcal{A}, \mathcal{T}, r, \gamma)$ with discount factor $\gamma \in [0, 1]$.

State ($S^{(t)}$): The state $S^{(t)}$ represents a partial subnetwork $G^{(t)} = (V^{(t)}, E^{(t)})$, where $E^{(t)} \subseteq E$ and $V^{(t)}$ includes all nodes incident to edges in $E^{(t)}$. A global seed set $S \subseteq V$ is fixed at start and remains unchanged throughout the episode. Each episode starts from the initial state $G^{(0)} = (S, \emptyset)$. The terminal state is reached when all nodes reachable from the seed set S in $G^{(t)}$ have exhausted their out-degree budget of k edges.

Action ($a^{(t)}$): Each action $a^{(t)} = (u, v)$ corresponds to activating a directed edge that is not currently in $E^{(t)}$. The admissible action set $\mathcal{A}(S^{(t)})$ includes edges that satisfy: (1) the out-degree constraint, $\deg^+_{G^{(t)}}(u) \leq k$, and (2) the requirement that u remains reachable from the seed set S in $G^{(t)}$.

Transition ($\mathcal{T}$): Applying action $a^{(t)} = (u, v)$ transitions the environment from $S^{(t)}$ to $S^{(t+1)}$ by adding the selected edge:

$$S^{(t+1)} = S^{(t)} \cup \{(u, v)\}, \quad G^{(t+1)} = (V^{(t)} \cup \{v\}, E^{(t)} \cup \{(u, v)\}).$$

Reward (r): The agent receives an immediate reward equal to the marginal gain in influence (Definition 1):

$$r(S^{(t)}, (u, v)) := \Delta((u, v); G^{(t)}, S) = \mathcal{I}_{G^{(t+1)}}(S) - \mathcal{I}_{G^{(t)}}(S),$$

We adopt an exact computation of influence under the IC model [49], thereby avoiding approximate techniques such as RMPP [14] and RIS [27]. Although computing true marginal gains (used for reward estimation) is computationally intensive, our method restricts this operation to the training phase, where greater computational resources and time are available. This enables more informative and stable learning signals.

We model the Q-function $Q^*(S^{(t)}, a^{(t)} = (u, v))$ using a neural approximator $\hat{Q}(S^{(t)}, a^{(t)} = (u, v); \Theta)$, which estimates the expected cumulative reward for selecting edge (u, v) in state $S^{(t)}$. The function is parameterized by Θ.

5.2 Computing Q-Values for Candidate Edges

To estimate Q-values, we first compute edge-level features using the aggregated node embeddings $\mathbf{h}^f \in \mathbb{R}^{|V|\times d}$ (from Sec. 4.2) and edge weights w_{ij}. For each edge (i, j), we compute the feature $\mathbf{f}_{ij}$ as follows:

$$\mathbf{z}_j = \phi\left(\beta_1(\mathbf{h}_j^f)\right), \quad \mathbf{z}_{ij} = \phi\left(\beta_2(w_{ij})\right), \quad \mathbf{f}_{ij} = \left[\mathbf{z}_j, |, \mathbf{z}_{ij}\right] \in \mathbb{R}^{d+1}$$

where $\beta_1 : \mathbb{R}^d \to \mathbb{R}^d$ and $\beta_2 : \mathbb{R} \to \mathbb{R}$ are learnable projections, and ϕ is the leaky-ReLU activation function. To discourage redundant edge activations, we apply a penalty ($\mathbf{p}_j$) based on the cumulative incoming edge weight at node j:

$$A_j = \sum_{i:(i,j)\in \mathrm{E}} w_{ij}, \quad \mathbf{p}_j = \phi\left(A_j \cdot \beta_3\right)$$

where β_3 is a learnable scalar. This penalty is subtracted from all edge features ending at node j, yielding adjusted features $\tilde{\mathbf{f}}_{ij} = \mathbf{f}_{ij} - \mathbf{p}_j$. The adjusted feature $\tilde{\mathbf{f}}_{ij}$ is then passed through a fully connected neural network $\mathcal{NN}(\cdot)$ to estimate the Q-value as:

$$\hat{Q}(S^{(t)}, (u,v); \Theta) = \mathcal{NN}(\tilde{\mathbf{f}}_{uv})$$

5.3 Training

To improve stability, mitigate overestimation bias, and accelerate convergence, we train $\hat{Q}$ using an n-step Double Deep Q-Learning (DDQN) approach [81, 85]. Specifically, we maintain two networks: an online network $\hat{Q}(\cdot\,;\Theta)$ and a target network $\hat{Q}(\cdot\,;\Theta^-)$.

At each training step, the agent samples a batch of n-step transition sequences. For each sequence i, starting from state $S_i^{(t)}$ and action $a_i^{(t)}$, the agent observes a reward trajectory $(r_i^{(t)}, r_i^{(t+1)}, \ldots, r_i^{(t+n-1)})$ and arrives at state $S_i^{(t+n)}$. The target Q-value is computed as:

$$y_i^{(t)} = \sum_{k=0}^{n-1} \gamma^k r_i^{(t+k)} + \gamma^n \hat{Q}\left(S_i^{(t+n)}, \arg\max_{a'} \hat{Q}(S_i^{(t+n)}, a'; \Theta); \Theta^-\right)$$

The online network parameters Θ are updated by minimizing the mean squared error (MSE) over a batch of B transitions, according to the following loss function:

$$\mathcal{L}(\Theta) = \frac{1}{B}\sum_{i=1}^{B}\left(\hat{Q}(S_i^{(t)}, a_i^{(t)}; \Theta) - y_i^{(t)}\right)^2$$

Training proceeds until convergence, with the target network parameters Θ^- periodically synchronized with the online network to stabilize target estimation.

To balance exploration and exploitation during training, we employ an ε-greedy policy: with probability ε, a random valid edge from $\mathcal{A}(S^{(t)})$ is selected; otherwise, the action maximizing $\hat{Q}(S^{(t)}, a^{(t)}; \Theta)$ is chosen. The exploration rate ε decays over time, gradually shifting the policy toward exploitation.

Throughout training, we restrict the action space to edges valid under the current state, ensuring that the learned policy remains compliant with the k-subnetwork structural constraint.

5.4 Inference

At test time, we begin with the given seed set S and $G^{(0)} = (S, \emptyset)$. At each decision step, we compute Q-values $\mathbf{Q} \in \mathbb{R}^{|E|}$ using the trained model. To enforce the k-subnetwork constraint, we apply an action mask that filters out invalid edges—specifically, those whose selection would violate the per-node out-degree limit by exceeding the maximum of k outgoing activations.

From the remaining valid actions, we select the edge with the highest Q-value:

$$a^* = \arg\max_{(u,v)\in\mathcal{A}(S^{(t)})} \hat{Q}(S^{(t)}, (u,v); \Theta).$$

The selected edge is activated and added to the current subgraph, updating the state. This process repeats iteratively until no valid actions remain. The final subnetwork $G_k = (V_k, E_k)$ is returned, and its influence is evaluated via $\mathcal{I}_{G_k}(S)$. Since each node may activate up to k edges, the maximum number of steps is bounded by $O(|V| \cdot k)$.

6 Experiments

We evaluate our proposed approach, RELINK, against state-of-the-art algorithms on the Influence Maximization in Closed Social Networks (IM-CSN) task. Empirical results demonstrate that the DRL-based RELINK framework consistently outperforms existing methods. This section details the experimental setup, including datasets, evaluation metrics, baselines, and implementation specifics. To guide the analysis, we pose several key questions throughout.

6.1 Experimental Setup

Training Dataset Description: To train RELINK, we construct synthetic datasets designed to reflect structural characteristics commonly observed in real-world social networks. We employ four generative models: Barabási–Albert preferential attachment networks [6], Watts–Strogatz small-world graphs [89], stochastic block models [43], and Erdős–Rényi random graphs [28]. For each model, we generate 10 network instances, yielding a total of 40 training graphs with node counts ranging from 25 to 150.

Testing Dataset Description: We evaluate all methods on 20 real-world social network datasets that exhibit diverse structural properties and vary widely in scale, spanning a broad spectrum of domains and graph sizes to enable comprehensive evaluation across heterogeneous settings. Table 1 provides a summary of node and edge counts for each dataset. Following the preprocessing protocol of Huang et al. [46], we convert each undirected edge into a pair of directed edges in opposite directions. For each network, we randomly sample seed sets S from the top 5% of nodes ranked by degree, with $|S| \in \{5, 50, 100, 200\}$, following the procedure in Huang et al. [46]. We also vary the parameter k over the set $\{5, 10, 20\}$, resulting in a total of 480 experimental configurations.

Diffusion Models: We evaluate all methods under the Independent Cascade (IC) model[1]. During training, edge activation probabilities are randomly sampled from the interval $[0, 1]$. For evaluation, we adopt two standard weighting schemes from the IM literature: (i) *random weights* [2, 7, 20, 32, 84, 91], where edge probabilities are sampled from $[0, 1]$, and (ii) *uniform weights* [2, 19, 20, 35, 49, 70], where all edge probabilities are fixed at 0.5.

Baseline Methods: We compare against four established methods from the Influence Maximization in Closed Social Networks (IM-CSN) literature [46]. The first three are traditional baselines originally developed for traditional influence maximization and subsequently adapted to the IM-CSN setting by Huang et al. [46]. These include: (i) the DEGREE-based method, which selects the top-k outgoing edges per source node based on the degrees of the target nodes; (ii) the FOF method, which prioritizes edges connecting node pairs that share the most common neighbors; and (iii) the RANDOM method, which selects k outgoing edges per node uniformly at random, with performance averaged over five runs. Additionally,

[1]RELINK can be readily adapted to a range of diffusion models—including the Linear Threshold [49], Trigger [39], and Weighted Cascade (WC) [1] models—by appropriately modifying the reward function before training the model.

Table 1: Dataset Characteristics

Dataset (Name)	Nodes	Edges
Residence hall (**moreno-oz**) [31, 55]	217	2674
Physicians (**moreno-innovation**) [24, 55]	241	1100
FilmTrust (**librec-filmtrust-trust**) [38]	874	1853
Email-Eu-core (**email-Eu-core**) [58, 95]	1005	25571
Adolesc. health (**moreno-health**) [55, 67]	2539	12971
Facebook (**ego-Facebook**) [65]	4039	88234
CiaoDVD (**librec-ciaodvd-trust**) [37]	4658	40133
Wikipedia vote (**wiki-Vote**) [57]	7115	103689
LastFM Asia (**lastfm-asia**) [78]	7624	27806
Facebook Large (**musae-facebook**) [76]	22470	171002
Twitter (**ego-twitter**) [65]	23370	33101
Google+ (**ego-gplus**) [65]	23628	39242
Math temporal (**sx-mathoverflow**) [71]	24818	506550
Deezer - RO (**ro-deezer**) [77]	41773	125827
Deezer - HU (**hu-deezer**) [77]	47538	222888
Deezer - HR (**hr-deezer**) [77]	54573	498203
Brightkite (**Bright kite**) [22]	58228	214078
Epinions (**soc-Epinions1**) [74]	75879	508841
Slashdot Zoo (**slashdot-zoo**) [55, 56]	79116	515399
Slashdot Feb'09 (**soc-sign-Slashdot**) [57]	82140	549202

we include Practical Subnetwork Augmentation (PSNA), a method proposed specifically for IM-CSN by Huang et al. [46].

Neural Network Architecture: For each learnable layer in the online and target networks, we employ a standardized architecture consisting of a two-layer multilayer perceptron (MLP) with 64 and 32 hidden units, respectively, unless otherwise specified in Sections 4.2 and 5.2. The embedding dimension (d), obtained via singular value decomposition (SVD), is fixed at 16. ReLU activations are applied to hidden layers, while output layers use linear activations. Target networks are updated using exponential moving average (EMA) [68], and the replay memory buffer is set to a capacity of 1M transitions. All models are trained using the Adam optimizer [51] and implemented in PyTorch [73]. All experiments are conducted on a machine equipped with an NVIDIA A40 GPU and an Intel(R) Xeon(R) Silver 4314 CPU.

Evaluation Criteria: For each method, we extract the selected subgraph and estimate its influence spread using 100,000 Monte Carlo simulations under the IC model [49]. Each experiment is repeated 10 times, and we report the average influence spread and total runtime across all runs.

6.2 Evaluating Known Methods vs RELINK:

Figure 3 presents a contingency table comparing our proposed method against existing methods for the IM-CSN task. We conduct 480 experiments across 20 real-world social networks. Each cell (i, j) in the table reports the number of instances (out of 480) in which the method in row i achieved a higher average influence spread than the method in column j. Notably, a single RELINK model—trained on the synthetic datasets described in Section 6.1—is evaluated across all combinations of these settings.

The color scale in Figure 3 visualizes relative performance: darker blue cells indicate that the row method outperforms the column method in most experiments, while darker red denotes the opposite. Each cell shows the raw count of wins, and both axes enumerate the compared methods.

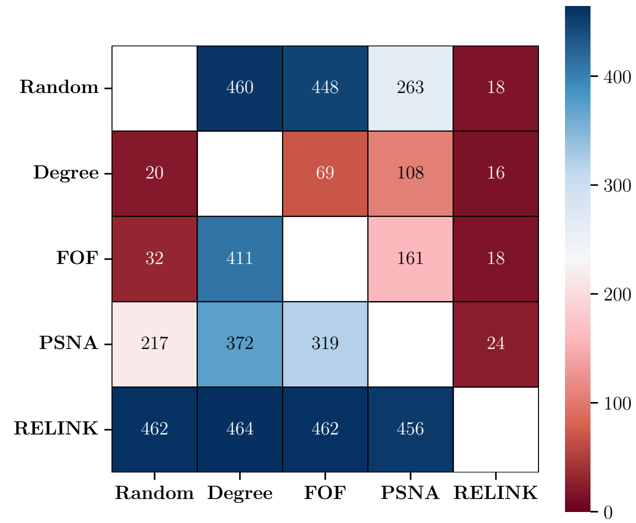


Figure 3: Contingency table comparing baseline methods against our approach (Relink). Each cell shows how often the row method outperformed the column in average influence spread. Darker blue indicates more wins; darker red indicates fewer.

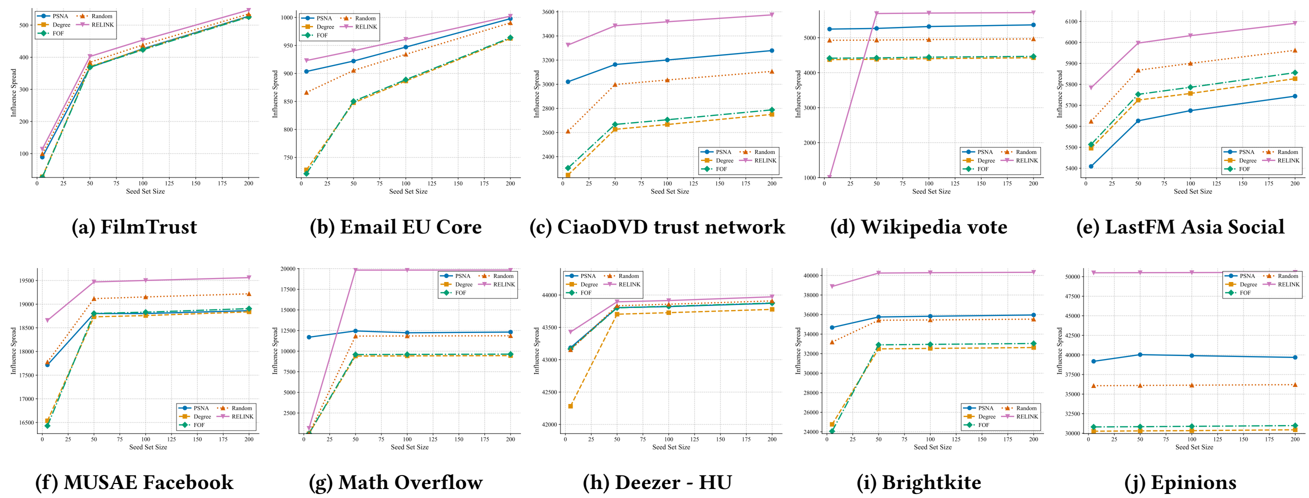
The contingency table reveals that RELINK consistently outperforms all methods in at least 90% of the experiments, as indicated by the dark blue shading in the last row. This highlights the robustness and effectiveness of RELINK, which achieves the highest average influence spread across the majority of experimental configurations. Among the remaining methods, Random ranks second, followed by PSNA, while FoF and Degree perform worst on IM-CSN.

Q1. How does RELINK perform in terms of influence spread compared to traditional IM baselines (Random, FoF, Degree) Figure 3 shows that RELINK consistently outperforms all traditional baselines. It achieves higher influence spread than Random in 462 out of 480 experiments, and similarly outperforms Degree and FoF in 464 and 462 cases, respectively. This relative ordering—Random outperforming FoF, which in turn outperforms Degree—is consistent with prior findings from Huang et al. [46].

Q2. How does RELINK compare to PSNA on the influence spread metric, given that PSNA is specifically designed for the IM-CSN task by Huang et al. [46]? RELINK outperforms PSNA in 456 out of 480 experiments, while PSNA performs better in only 24 cases. These results demonstrate the strong generalization ability of our method, which remains effective even when compared against a IM-CSN-specific baseline.

Influence Spread and Runtime Comparison

Figures 4 and 5 show comparisons of influence spread and runtime, respectively. Although our full evaluation covers 20 datasets, multiple edge budgets (k), and various weight initialization schemes, we report line plots for ten representative datasets across all seed set



(a) FilmTrust (b) Email EU Core (c) CiaoDVD trust network (d) Wikipedia vote (e) LastFM Asia Social

(f) MUSAE Facebook (g) Math Overflow (h) Deezer - HU (i) Brightkite (j) Epinions

Figure 4: Influence spread comparison for $k = 20$ and seed sets $|\mathcal{S}| \in \{5, 50, 100, 200\}$ across networks of varying sizes.

sizes under a fixed edge budget ($k = 20$) and *uniform* edge weights due to space constraints. In both plots, the x-axis indicates the seed set size ($|S|$), and the y-axis represents the evaluated metric—either influence spread or runtime.

The selected datasets include one small-scale network (*FilmTrust*), four medium-scale networks with fewer than 10,000 nodes (*email-Eu-core*, *CiaoDVD*, *Wikipedia Vote*, and *LastFM Asia*), and five large-scale networks (*MUSAE Facebook*, *Math Overflow*, *Deezer HU*, *BK* and *Epinions*).

As shown in Figure 4, RELINK consistently achieves the highest influence spread when the seed set size is small. While performance gaps narrow as $|S|$ increases, RELINK maintains superiority across most configurations. Among the baselines, Random and PSNA generally perform best, followed by FoF and Degree.

Q3. How does the runtime of RELINK compare to all baseline methods across different graph scales? Figure 5 presents runtime results across ten datasets. On small- and medium-scale graphs, RELINK incurs higher runtime compared to the other methods. This is expected, as our approach formulates edge selection as a sequential decision-making process, selecting one edge per step, in contrast to the other methods that select multiple edges in a single pass [46]. Additionally, the node embedding initialization introduces a fixed overhead.

However, as graph size increases, RELINK becomes increasingly competitive—often matching or outperforming PSNA in terms of runtime. This is largely due to the latter's computational complexity, $O(I(|V| + |E|) \log |V| + |V| d_m \log d_m)$, where I is the number of iterations until convergence, $|V|$ and $|E|$ denote the number of nodes and edges, and d_m is the maximum in-degree. As graph scale grows, PSNA's runtime deteriorates, whereas RELINK benefits from GPU-accelerated inference (see Section 5.4), enabling more favorable scalability. The traditional baselines—Random, Degree, and FoF—require negligible preprocessing and maintain consistently low runtime across all datasets.

6.3 Detailed Comparison of RELINK vs PSNA

To enable a fine-grained comparison with PSNA—a baseline specifically designed for the IM-CSN task—we present a heatmap in Figure 6, focusing on experiments with *random* edge weights. This visualization summarizes the relative performance of RELINK across all 20 datasets and parameter configurations.

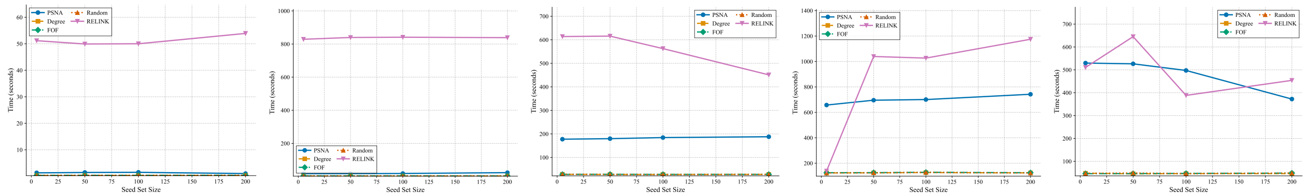
Each cell reports the ratio of average influence spread between the two methods, computed as $(\text{Inf}_{\text{RELINK}}/\text{Inf}_{\text{PSNA}})$. Datasets are sorted by graph size along the y-axis, while the x-axis enumerates combinations of edge budget (k) and seed set size ($|S|$). Blue cells indicate configurations where RELINK outperforms PSNA; red cells indicate the opposite; and white denotes comparable performance. Color intensity corresponds to the magnitude of the performance gap, with darker shades indicating larger differences.

The heatmap shows that RELINK method outperforms PSNA in most configurations, as indicated by the dominant blue cells. However, as the edge budget k increases, the performance gap narrows, with more light blue and white cells appearing on the right side of the heatmap, reflecting cases where PSNA catches up. On dense graphs such as *email-Eu-core*, *wiki-Vote*, and *sx-mathoverflow*, PSNA performs better for smaller seed set sizes. Nevertheless, RELINK method remains superior in most settings, frequently achieving influence spread improvements exceeding 15%.

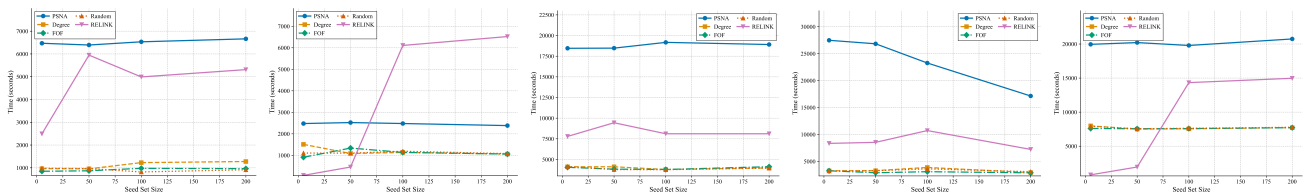
Q4. How does RELINK method compare to PSNA in initiating large-scale diffusion under constrained edge budgets (i.e., lower values of k)? A key observation is that under low edge budgets, RELINK significantly outperforms PSNA, as indicated by the dark blue cells on the left side of the heatmap. This trend underscores RELINK's strength in prioritizing high-impact edges early, making it particularly effective in constrained settings, whereas PSNA tends to close the gap under more relaxed budget conditions.

6.4 Training Time

As previously noted, we train a single model used across all experiments. Training takes 6,257.329 seconds for 2,000 epochs. Due to the use of true marginal gains, the training process is computationally



(a) FilmTrust **(b) Email EU Core** **(c) CiaoDVD trust** **(d) Wikipedia vote** **(e) LastFM Asia Social**



(f) MUSAE Facebook **(g) Math Overflow** **(h) Deezer SN - HU** **(i) Brightkite** **(j) Epinions**

Figure 5: Execution time comparison for $k = 20$ and seed sets $|\mathcal{S}| \in \{5, 50, 100, 200\}$ across networks of varying sizes.

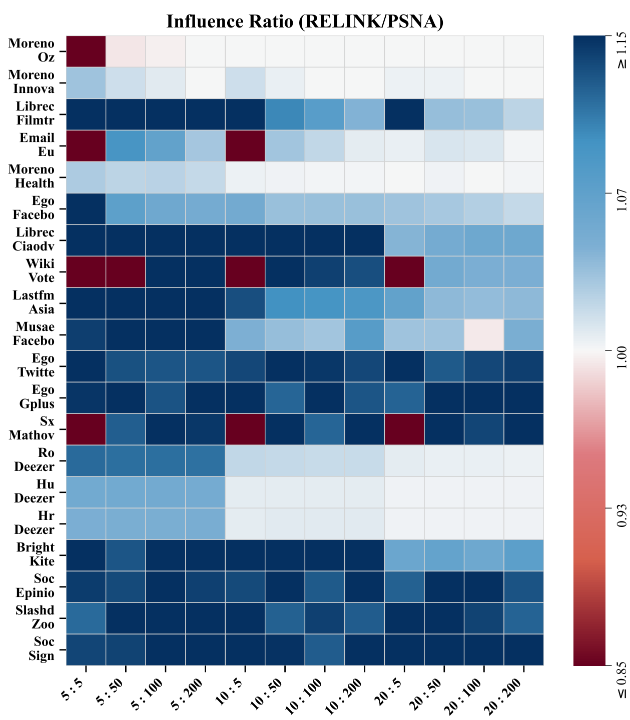


Figure 6: Heatmap of influence ratios ($\mathbf{Inf_{RELINK}/Inf_{PSNA}}$). X-axis: edge budget and seed set size (k:$|S|$); Y-axis: social networks. Blue: RELINK outperforms PSNA; red: PSNA outperforms. Darker shades indicate larger differences.

intensive. However, once trained, the model performs inference without requiring influence spread estimation or approximation (unlike PSNA), enabling greater scalability on large networks.

6.5 Summary

We evaluate RELINK on 20 real-world social networks under the Independent Cascade (IC) model, comparing it against four established baselines: Degree, FoF, Random, and PSNA. The model is trained once on synthetic graphs designed to reflect diverse structural properties and is evaluated across a wide range of settings by varying seed set sizes and edge budgets. RELINK consistently achieves higher average influence spread than all baselines. The gains are most prominent in low-budget scenarios and on large graphs, where the method excels at selecting high-impact edges early. As graph size increases, RELINK becomes increasingly competitive with, and often outperforms, PSNA—a method specifically designed for IM-CSN—in terms of runtime. These results highlight the robustness, scalability, and strong generalization of RELINK.

7 Conclusion and Future Work

We introduced RELINK, a deep reinforcement learning (DRL) framework for edge-level influence maximization in closed social networks. In contrast to traditional node-based strategies, our method targets settings where privacy or structural constraints make direct node activation infeasible. By framing the problem as a Markov Decision Process (MDP), RELINK allows the agent to sequentially activate edges to maximize cumulative influence spread, ultimately identifying an optimal k-subnetwork for diffusion. Our method integrates a rich node embedding pipeline, a sparse edge-aware aggregation module, and action masking for policy feasibility. Our empirical results demonstrate that RELINK consistently outperforms state-of-the-art baselines in both influence spread and scalability.

In future work, we aim to extend RELINK to settings with dynamic topologies and adaptive constraints. Further exploration of sample-efficient training strategies and theoretical analysis of policy optimality under submodular objectives also presents promising directions. Extending the framework to dynamic or time-evolving graphs, incorporating user-specific constraints, and integrating uncertainty in diffusion models are promising directions.

8 GenAI Usage Disclosure

We used generative AI tools (e.g., ChatGPT) solely for language edits—such as grammar, punctuation, and clarity improvement, comparable to traditional writing assistants like Grammarly. These tools were not used to generate original content or ideas. All substantive content was authored and verified by the authors.

References

[1] Zahra Aghaee and Sahar Kianian. 2020. Efficient influence spread estimation for influence maximization. *Social Network Analysis and Mining* 10, 1 (Sept. 2020), 81. https://doi.org/10.1007/s13278-020-00694-z TLDR: A heuristic-greedy algorithm named the HEDVGreedy algorithm was proposed for the influence maximization problem (IMP) in the social networks and it significantly reduces the running time of the computations under both the Independent Cascade and Weighted Cascade models in the eight real-world data sets..

[2] Akhil Arora, Sainyam Galhotra, and Sayan Ranu. 2017. Debunking the Myths of Influence Maximization: An In-Depth Benchmarking Study. In *Proceedings of the 2017 ACM International Conference on Management of Data, SIGMOD Conference 2017, Chicago, IL, USA, May 14-19, 2017*, Semih Salihoglu, Wenchao Zhou, Rada Chirkova, Jun Yang, and Dan Suciu (Eds.). ACM, 651–666. https://doi.org/10.1145/3035918.3035924

[3] Kumar Ashish, Yaswanth Bolisetty, Deepanshu Singh, and Amarinder Kaur. 2024. Encryption, Privacy, and Usability : A Comparative Evaluation of Leading Secure Messaging Platforms. *International Journal of Scientific Research in Computer Science, Engineering and Information Technology* (2024).

[4] Çigdem Aslay, Laks V. S. Lakshmanan, Wei Lu, and Xiaokui Xiao. 2018. Influence Maximization in Online Social Networks. In *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining, WSDM 2018, Marina Del Rey, CA, USA, February 5-9, 2018*, Yi Chang, Chengxiang Zhai, Yan Liu, and Yoelle Maarek (Eds.). ACM, 775–776. https://doi.org/10.1145/3159652.3162007

[5] Suman Banerjee, Mamata Jenamani, and Dilip Kumar Pratihar. 2019. ComBIM: A community-based solution approach for the Budgeted Influence Maximization Problem. *Expert Systems with Applications* 125 (2019), 1–13.

[6] Albert-László Barabási and Réka Albert. 2010. Emergence of Scaling in Random Networks. *ArXiv preprint* abs/10.1126 (2010). https://arxiv.org/abs/10.1126

[7] Ruben Becker, Gianlorenzo D'Angelo, and Sajjad Ghobadi. 2023. On the cost of demographic parity in influence maximization. In *Proceedings of the Thirty-Seventh AAAI Conference on Artificial Intelligence and Thirty-Fifth Conference on Innovative Applications of Artificial Intelligence and Thirteenth Symposium on Educational Advances in Artificial Intelligence (AAAI'23/IAAI'23/EAAI'23)*. AAAI Press, Article 1582, 9 pages. https://doi.org/10.1609/aaai.v37i12.26651

[8] Glenn S Bevilacqua and Laks VS Lakshmanan. 2022. A fractional memory-efficient approach for online continuous-time influence maximization. *The VLDB Journal* 31, 2 (2022), 403–429.

[9] Song Bian, Qintian Guo, Sibo Wang, and Jeffrey Xu Yu. 2020. Efficient algorithms for budgeted influence maximization on massive social networks. *Proceedings of the VLDB Endowment* 13, 9 (2020), 1498–1510.

[10] Vincent D. Blondel, Jean-Loup Guillaume, Renaud Lambiotte, and Etienne Lefebvre. 2008. Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment* 2008, 10 (2008), P10008. https://doi.org/10.1088/1742-5468/2008/10/P10008

[11] Christian Borgs, Michael Brautbar, Jennifer T. Chayes, and Brendan Lucier. 2014. Maximizing Social Influence in Nearly Optimal Time. In *Proceedings of the Twenty-Fifth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2014, Portland, Oregon, USA, January 5-7, 2014*, Chandra Chekuri (Ed.). SIAM, 946–957. https://doi.org/10.1137/1.9781611973402.70

[12] Taotao Cai, Jianxin Li, Ajmal Mian, Rong-Hua Li, Timos Sellis, and Jeffrey Xu Yu. 2020. Target-aware holistic influence maximization in spatial social networks. *IEEE Transactions on Knowledge and Data Engineering* 34, 4 (2020), 1993–2007.

[13] Bogdan Cautis, Silviu Maniu, and Nikolaos Tziortziotis. 2019. Adaptive Influence Maximization. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD 2019, Anchorage, AK, USA, August 4-8, 2019*, Ankur Teredesai, Vipin Kumar, Ying Li, Rómer Rosales, Evimaria Terzi, and George Karypis (Eds.). ACM, 3185–3186. https://doi.org/10.1145/3292500.3332285

[14] Vineet Chaoji, Sayan Ranu, Rajeev Rastogi, and Rushi Bhatt. 2012. Recommendations to boost content spread in social networks. In *Proceedings of the 21st World Wide Web Conference 2012, WWW 2012, Lyon, France, April 16-20, 2012*, Alain Mille, Fabien L. Gandon, Jacques Misselis, Michael Rabinovich, and Steffen Staab (Eds.). ACM, 529–538. https://doi.org/10.1145/2187836.2187908

[15] Sikky Shiqi Chen, Tai Lam, Kwok Lam, William Lo, David Chao, Ki Mak, Edmund Lam, Wai Tang, Hoi Chan, and Paul Yip. 2022. The Use of Close Friends on Instagram, Help-Seeking Willingness, and Suicidality Among Hong Kong Youth: Exploratory Sequential Mixed Methods Study. *Journal of Medical Internet Research* 24 (2022), e37695. https://doi.org/10.2196/37695

[16] Wei Chen, Chi Wang, and Yajun Wang. 2010. 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, Washington, DC, USA, July 25-28, 2010*, Bharat Rao, Balaji Krishnapuram, Andrew Tomkins, and Qiang Yang (Eds.). ACM, 1029–1038. https://doi.org/10.1145/1835804.1835934

[17] Wei Chen, Chi Wang, and Yajun Wang. 2010. 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, Washington, DC, USA, July 25-28, 2010*, Bharat Rao, Balaji Krishnapuram, Andrew Tomkins, and Qiang Yang (Eds.). ACM, 1029–1038. https://doi.org/10.1145/1835804.1835934

[18] Wei Chen, Yajun Wang, and Siyu Yang. 2009. Efficient influence maximization in social networks. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*. 199–208.

[19] Wei Chen, Yajun Wang, and Siyu Yang. 2009. Efficient influence maximization in social networks. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (Paris, France) *(KDD '09)*. Association for Computing Machinery, New York, NY, USA, 199–208. https://doi.org/10.1145/1557019.1557047

[20] Wei Chen, Yifei Yuan, and Li Zhang. 2010. Scalable Influence Maximization in Social Networks under the Linear Threshold Model. In *2010 IEEE International Conference on Data Mining*. 88–97. https://doi.org/10.1109/ICDM.2010.118

[21] Suqi Cheng, Huawei Shen, Junming Huang, Guoqing Zhang, and Xueqi Cheng. 2013. StaticGreedy: solving the scalability-accuracy dilemma in influence maximization. In *22nd ACM International Conference on Information and Knowledge Management, CIKM'13, San Francisco, CA, USA, October 27 - November 1, 2013*, Qi He, Arun Iyengar, Wolfgang Nejdl, Jian Pei, and Rajeev Rastogi (Eds.). ACM, 509–518. https://doi.org/10.1145/2505515.2505541

[22] Eunjoon Cho, Seth A. Myers, and Jure Leskovec. 2011. Friendship and mobility: user movement in location-based social networks. In *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Diego, CA, USA, August 21-24, 2011*, Chid Apté, Joydeep Ghosh, and Padhraic Smyth (Eds.). ACM, 1082–1090. https://doi.org/10.1145/2020408.2020579

[23] Tzu-Yuan Chung, Khurshed Ali, and Chih-Yu Wang. 2019. Deep reinforcement learning-based approach to tackle competitive influence maximization. In *Proc. of MLG workshop*.

[24] James Coleman, Elihu Katz, and Herbert Menzel. 1957. The diffusion of an innovation among physicians. *Sociometry* 20, 4 (1957), 253–270.

[25] Federico Coró, Gianlorenzo D'angelo, and Yllka Velaj. 2021. Link recommendation for social influence maximization. *ACM Transactions on Knowledge Discovery from Data (TKDD)* 15, 6 (2021), 1–23.

[26] Gianlorenzo D'Angelo, Lorenzo Severini, and Yllka Velaj. 2019. Recommending links through influence maximization. *Theoretical Computer Science* 764 (2019), 30–41.

[27] Ying-Hao Du, Wen-Jin Qiu, and Wei–Neng Chen. 2023. Influence Maximization with Reverse Influence Sampling and Evolutionary Algorithm. In *2023 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*. 783–790. https://doi.org/10.1109/SMC53992.2023.10394612

[28] P. Erdös and A. Rényi. 2006. *On the evolution of random graphs.* Princeton University Press, Princeton, 38–82. https://doi.org/doi:10.1515/9781400841356.38

[29] Ksenia Ermoshina and Francesca Musiani and. 2019. "Standardising by running code": the Signal protocol and de facto standardisation in end-to-end encrypted messaging. *Internet Histories* 3, 3-4 (2019), 343–363. https://doi.org/10.1080/24701475.2019.1654697

[30] Sindhu Kiranmai Ernala, Stephanie Yang, Yuxi Wu, Rachel Chen, Kristen Wells, and Sauvik Das. 2021. Exploring the Utility Versus Intrusiveness of Dynamic Audience Selection on Facebook. *Proceedings of the ACM on Human-Computer Interaction* 5 (2021), 1 – 30.

[31] Linton C Freeman, Cynthia M Webster, and Deirdre M Kirke. 1998. Exploring social structure using dynamic three-dimensional color images. *Social networks* 20, 2 (1998), 109–118.

[32] Sainyam Galhotra, Akhil Arora, and Shourya Roy. 2016. Holistic Influence Maximization: Combining Scalability and Efficiency with Opinion-Aware Models. In *Proceedings of the 2016 International Conference on Management of Data, SIGMOD Conference 2016, San Francisco, CA, USA, June 26 - July 01, 2016*, Fatma Özcan, Georgia Koutrika, and Sam Madden (Eds.). ACM, 743–758. https://doi.org/10.1145/2882903.2882929

[33] Smita Ghosh, Tiantian Chen, and Weili Wu. 2024. Enhanced Group Influence Maximization in Social Networks Using Deep Reinforcement Learning. *IEEE Transactions on Computational Social Systems* (2024).

[34] Amit Goyal, Wei Lu, and Laks VS Lakshmanan. 2011. Celf++ optimizing the greedy algorithm for influence maximization in social networks. In *Proceedings of the 20th international conference companion on World wide web*. 47–48.

[35] Amit Goyal, Wei Lu, and Laks V.S. Lakshmanan. 2011. CELF++: optimizing the greedy algorithm for influence maximization in social networks. In *Proceedings of the 20th International Conference Companion on World Wide Web* (Hyderabad, India) *(WWW '11)*. Association for Computing Machinery, New York, NY, USA, 47–48. https://doi.org/10.1145/1963192.1963217

[36] Aditya Grover and Jure Leskovec. 2016. node2vec: Scalable Feature Learning for Networks. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, August 13-17, 2016*, Balaji Krishnapuram, Mohak Shah, Alexander J. Smola, Charu C. Aggarwal, Dou Shen, and Rajeev Rastogi (Eds.). ACM, 855–864. https://doi.org/10.1145/2939672.2939754

[37] Guibing Guo, Jie Zhang, Daniel Thalmann, and Neil Yorke-Smith. 2014. ETAF: An extended trust antecedents framework for trust prediction. In *2014 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining, ASONAM 2014, Beijing, China, August 17-20, 2014*, Xindong Wu, Martin Ester, and Guandong Xu (Eds.). IEEE Computer Society, 540–547. https://doi.org/10.1109/ASONAM.2014.6921639

[38] Guibing Guo, Jie Zhang, and Neil Yorke-Smith. 2013. A Novel Bayesian Similarity Measure for Recommender Systems. In *IJCAI 2013, Proceedings of the 23rd International Joint Conference on Artificial Intelligence, Beijing, China, August 3-9, 2013*, Francesca Rossi (Ed.). IJCAI/AAAI, 2619–2625. http://www.aaai.org/ocs/index.php/IJCAI/IJCAI13/paper/view/6615

[39] Jianxiong Guo and Weili Wu. 2020. Influence Maximization: Seeding Based on Community Structure. *ACM Trans. Knowl. Discov. Data* 14, 6 (Sept. 2020), 66:1–66:22. https://doi.org/10.1145/3399661 TLDR: A novel seeding strategy based on community structure is proposed, and the IMCB-Framework, which is inspired by the idea of continuous greedy process and pipage rounding, and derive the best approximation ratio for this problem is proposed..

[40] Kai Han, Keke Huang, Xiaokui Xiao, Jing Tang, Aixin Sun, and Xueyan Tang. 2018. Efficient algorithms for adaptive influence maximization. *Proceedings of the VLDB Endowment* 11, 9 (2018), 1029–1040.

[41] Qiang He, Xingwei Wang, Zhencheng Lei, Min Huang, Yuliang Cai, and Lianbo Ma. 2019. TIFIM: A two-stage iterative framework for influence maximization in social networks. *Appl. Math. Comput.* 354 (2019), 338–352.

[42] Eva Heinrich and Lucila Carvalho. 2022. Fostering connections for professional identity formation: two case studies of Discord discipline-focused communities. *ASCILITE Publications* (2022). https://api.semanticscholar.org/CorpusID:253870695

[43] Paul Holland, Kathryn B. Laskey, and Samuel Leinhardt. 1983. Stochastic blockmodels: First steps. *Social Networks* 5 (1983), 109–137.

[44] Huimin Huang, Hong Shen, Zaiqiao Meng, Huajian Chang, and Huaiwen He. 2019. Community-based influence maximization for viral marketing. *Applied Intelligence* 49 (2019), 2137–2150.

[45] Keke Huang, Jing Tang, Kai Han, Xiaokui Xiao, Wei Chen, Aixin Sun, Xueyan Tang, and Andrew Lim. 2020. Efficient approximation algorithms for adaptive influence maximization. *The VLDB Journal* 29 (2020), 1385–1406.

[46] Shixun Huang, Wenqing Lin, Z. Bao, and Jiachen Sun. 2022. Influence Maximization in Real-World Closed Social Networks. *Proceedings of the VLDB Endowment* (2022). https://doi.org/10.48550/arXiv.2209.10286

[47] Yunming Hui, Shihan Wang, Melisachew Wudage Chekol, Stevan Rudinac, and Inez Maria Zwetsloot. 2024. Non-Progressive Influence Maximization in Dynamic Social Networks. *ArXiv preprint* abs/2412.07402 (2024). https://arxiv.org/abs/2412.07402

[48] Kyomin Jung, Wooram Heo, and Wei Chen. 2012. Irie: Scalable and robust influence maximization in social networks. In *2012 IEEE 12th international conference on data mining*. IEEE, 918–923.

[49] David Kempe, Jon Kleinberg, and Éva Tardos. 2003. Maximizing the spread of influence through a social network. In *Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (Washington, D.C.) *(KDD '03)*. Association for Computing Machinery, New York, NY, USA, 137–146. https://doi.org/10.1145/956750.956769

[50] Elias Boutros Khalil, Bistra Dilkina, and Le Song. 2014. Scalable diffusion-aware optimization of network topology. In *The 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '14, New York, NY, USA - August 24 - 27, 2014*, Sofus A. Macskassy, Claudia Perlich, Jure Leskovec, Wei Wang, and Rayid Ghani (Eds.). ACM, 1226–1235. https://doi.org/10.1145/2623330.2623704

[51] Diederik P. Kingma and Jimmy Ba. 2015. Adam: A Method for Stochastic Optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, Yoshua Bengio and Yann LeCun (Eds.). http://arxiv.org/abs/1412.6980

[52] V. Klema and A. Laub. 1980. The singular value decomposition: Its computation and some applications. *IEEE Trans. Automat. Control* 25, 2 (1980), 164–176. https://doi.org/10.1109/TAC.1980.1102314

[53] ANIKÓ KOPACZ. 2024. COMPETITIVE INFLUENCE MAXIMIZATION IN TRUST-BASED SOCIAL NETWORKS WITH DEEP Q-LEARNING. *Studia Universitatis Babes-Bolyai, Informatica* 69, 1 (2024).

[54] Gouri Kundu, Smita Ghosh, and Sankhayan Choudhury. 2024. Rumor Containment in Hypergraph Representation of Social Networks: A Deep Reinforcement Learning-Based Solution. *IEEE Transactions on Computational Social Systems* (2024).

[55] Jérôme Kunegis. 2013. Konect: the koblenz network collection. In *Proceedings of the 22nd international conference on world wide web*. 1343–1350.

[56] Jérôme Kunegis, Andreas Lommatzsch, and Christian Bauckhage. 2009. The slashdot zoo: mining a social network with negative edges. In *Proceedings of the 18th International Conference on World Wide Web, WWW 2009, Madrid, Spain, April 20-24, 2009*, Juan Quemada, Gonzalo León, Yoëlle S. Maarek, and Wolfgang Nejdl (Eds.). ACM, 741–750. https://doi.org/10.1145/1526709.1526809

[57] Jure Leskovec, Daniel P. Huttenlocher, and Jon M. Kleinberg. 2010. Signed networks in social media. In *Proceedings of the 28th International Conference on Human Factors in Computing Systems, CHI 2010, Atlanta, Georgia, USA, April 10-15, 2010*, Elizabeth D. Mynatt, Don Schoner, Geraldine Fitzpatrick, Scott E. Hudson, W. Keith Edwards, and Tom Rodden (Eds.). ACM, 1361–1370. https://doi.org/10.1145/1753326.1753532

[58] Jure Leskovec, Jon Kleinberg, and Christos Faloutsos. 2007. Graph evolution: Densification and shrinking diameters. *ACM transactions on Knowledge Discovery from Data (TKDD)* 1, 1 (2007), 2–es.

[59] Hui Li, Mengting Xu, Sourav S Bhowmick, Joty Shafiq Rayhan, Changsheng Sun, and Jiangtao Cui. 2022. PIANO: Influence maximization meets deep reinforcement learning. *IEEE Transactions on Computational Social Systems* 10, 3 (2022), 1288–1300.

[60] Hui Li, Mengting Xu, Sourav S Bhowmick, Changsheng Sun, Zhongyuan Jiang, and Jiangtao Cui. 2019. Disco: Influence maximization meets network embedding and deep learning. *ArXiv preprint* abs/1906.07378 (2019). https://arxiv.org/abs/1906.07378

[61] Xiang Li, J David Smith, Thang N Dinh, and My T Thai. 2019. Tiptop:(almost) exact solutions for influence maximization in billion-scale networks. *IEEE/ACM Transactions on Networking* 27, 2 (2019), 649–661.

[62] Changan Liu, Changjun Fan, and Zhongzhi Zhang. 2024. Finding influencers in complex networks: An effective deep reinforcement learning approach. *Comput. J.* 67, 2 (2024), 463–473.

[63] Wei Liu, Xin Chen, Byeungwoo Jeon, Ling Chen, and Bolun Chen. 2019. Influence maximization on signed networks under independent cascade model. *Applied Intelligence* 49, 3 (2019), 912–928.

[64] Pilar Manzanares-Lopez, Juan Pedro Muñoz-Gea, and Josemaria Malgosa-Sanahuja. 2014. Analysis of LinkedIn Privacy Settings - Are they Sufficient, Insufficient or Just Unknown?. In *International Conference on Web Information Systems and Technologies*.

[65] Julian J. McAuley and Jure Leskovec. 2012. Learning to Discover Social Circles in Ego Networks. In *Advances in Neural Information Processing Systems 25: 26th Annual Conference on Neural Information Processing Systems 2012. Proceedings of a meeting held December 3-6, 2012, Lake Tahoe, Nevada, United States*, Peter L. Bartlett, Fernando C. N. Pereira, Christopher J. C. Burges, Léon Bottou, and Kilian Q. Weinberger (Eds.). 548–556. https://proceedings.neurips.cc/paper/2012/hash/7a614fd06c325499f1680b9896beedeb-Abstract.html

[66] Marco Minutoli, Mahantesh Halappanavar, Ananth Kalyanaraman, Arun Sathanur, Ryan Mcclure, and Jason McDermott. 2019. Fast and scalable implementations of influence maximization algorithms. In *2019 IEEE International Conference on Cluster Computing (CLUSTER)*. 1–12.

[67] James Moody. 2001. Peer influence groups: identifying dense clusters in large networks. *Social networks* 23, 4 (2001), 261–283.

[68] Daniel Morales-Brotons, Thijs Vogels, and Hadrien Hendrikx. 2024. Exponential Moving Average of Weights in Deep Learning: Dynamics and Benefits. *Trans. Mach. Learn. Res.* 2024 (2024). https://openreview.net/forum?id=2M9CUnYnBA

[69] Stefan Müller. 2023. How Slack Facilitates Communication and Collaboration in Seminars and Project-Based Courses. *Journal of Educational Technology Systems* 51 (2023), 303 – 316.

[70] Lata Narayanan and Kangkang Wu. 2020. How to choose friends strategically. *Theoretical Computer Science* 811 (2020), 99–111. https://doi.org/10.1016/j.tcs.2018.07.013

[71] Ashwin Paranjape, Austin R. Benson, and Jure Leskovec. 2017. Motifs in Temporal Networks. In *Proceedings of the Tenth ACM International Conference on Web Search and Data Mining, WSDM 2017, Cambridge, United Kingdom, February 6-10, 2017*, Maarten de Rijke, Milad Shokouhi, Andrew Tomkins, and Min Zhang (Eds.). ACM, 601–610. https://doi.org/10.1145/3018661.3018731

[72] Megan Parker and Mary Dodge. 2024. The New Neighborhood Watch: An Exploratory Study of the Nextdoor App and Crime Narratives. *International Journal of Criminology and Sociology* (2024).

[73] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, Hanna M. Wallach, Hugo Larochelle, Alina Beygelzimer, Florence d'Alché-Buc, Emily B. Fox, and Roman Garnett (Eds.). 8024–8035. https://proceedings.neurips.cc/paper/2019/hash/bdbca288fee7f92f2bfa9f7012727740-Abstract.html

[74] Matthew Richardson, Rakesh Agrawal, and Pedro Domingos. 2003. Trust management for the semantic web. In *International semantic Web conference*. Springer,

351–368.
[75] Matteo Riondato and Evgenios M. Kornaropoulos. 2014. Fast approximation of betweenness centrality through sampling. In *Seventh ACM International Conference on Web Search and Data Mining, WSDM 2014, New York, NY, USA, February 24-28, 2014*, Ben Carterette, Fernando Diaz, Carlos Castillo, and Donald Metzler (Eds.). ACM, 413–422. https://doi.org/10.1145/2556195.2556224
[76] Benedek Rozemberczki, Carl Allen, and Rik Sarkar. 2021. Multi-scale attributed node embedding. *Journal of Complex Networks* 9, 2 (2021), cnab014.
[77] Benedek Rozemberczki, Ryan Davies, Rik Sarkar, and Charles Sutton. 2019. GEMSEC: graph embedding with self clustering. In *ASONAM '19: International Conference on Advances in Social Networks Analysis and Mining, Vancouver, British Columbia, Canada, 27-30 August, 2019*, Francesca Spezzano, Wei Chen, and Xiaokui Xiao (Eds.). ACM, 65–72. https://doi.org/10.1145/3341161.3342890
[78] Benedek Rozemberczki and Rik Sarkar. 2020. Characteristic Functions on Graphs: Birds of a Feather, from Statistical Descriptors to Parametric Models. In *CIKM '20: The 29th ACM International Conference on Information and Knowledge Management, Virtual Event, Ireland, October 19-23, 2020*, Mathieu d'Aquin, Stefan Dietze, Claudia Hauff, Edward Curry, and Philippe Cudré-Mauroux (Eds.). ACM, 1325–1334. https://doi.org/10.1145/3340531.3411866
[79] Jari Saramäki, Mikko Kivelä, Jukka-Pekka Onnela, Kimmo Kaski, and János Kertész. 2007. Generalizations of the clustering coefficient to weighted complex networks. *Physical Review E* 75, 2 (2007), 027105. https://doi.org/10.1103/PhysRevE.75.027105 Publisher: American Physical Society TLDR: A comparative study of the several suggestions of the clustering coefficient, which is one of the central characteristics in the complex network theory, is presented..
[80] Lichao Sun, Weiran Huang, Philip S. Yu, and Wei Chen. 2018. Multi-Round Influence Maximization. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD 2018, London, UK, August 19-23, 2018*, Yike Guo and Faisal Farooq (Eds.). ACM, 2249–2258. https://doi.org/10.1145/3219819.3220101
[81] R.S. Sutton and A.G. Barto. 1998. Reinforcement Learning: An Introduction. *IEEE Transactions on Neural Networks* 9, 5 (1998), 1054–1054. https://doi.org/10.1109/TNN.1998.712192
[82] Youze Tang, Yanchen Shi, and Xiaokui Xiao. 2015. Influence Maximization in Near-Linear Time: A Martingale Approach. In *Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data, Melbourne, Victoria, Australia, May 31 - June 4, 2015*, Timos K. Sellis, Susan B. Davidson, and Zachary G. Ives (Eds.). ACM, 1539–1554. https://doi.org/10.1145/2723372.2723734
[83] Youze Tang, Xiaokui Xiao, and Yanchen Shi. 2014. Influence maximization: near-optimal time complexity meets practical efficiency. In *International Conference on Management of Data, SIGMOD 2014, Snowbird, UT, USA, June 22-27, 2014*, Curtis E. Dyreson, Feifei Li, and M. Tamer Özsu (Eds.). ACM, 75–86. https://doi.org/10.1145/2588555.2593670
[84] Youze Tang, Xiaokui Xiao, and Yanchen Shi. 2014. Influence maximization: near-optimal time complexity meets practical efficiency. In *International Conference on Management of Data, SIGMOD 2014, Snowbird, UT, USA, June 22-27, 2014*, Curtis E. Dyreson, Feifei Li, and M. Tamer Özsu (Eds.). ACM, 75–86. https://doi.org/10.1145/2588555.2593670
[85] Hado van Hasselt, Arthur Guez, and David Silver. 2016. Deep Reinforcement Learning with Double Q-Learning. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, February 12-17, 2016, Phoenix, Arizona, USA*, Dale Schuurmans and Michael P. Wellman (Eds.). AAAI Press, 2094–2100. http://www.aaai.org/ocs/index.php/AAAI/AAAI16/paper/view/12389
[86] Pascal Vogel, Christian Grotherr, Christian Kurtz, and Tilo Böhmann. 2020. Conceptualizing Design Parameters of Online Neighborhood Social Networks. In *Wirtschaftsinformatik*.
[87] Dakuo Wang, Haoyu Wang, Mo Yu, Zahra Ashktorab, and Ming Tan. 2022. Group Chat Ecology in Enterprise Instant Messaging: How Employees Collaborate Through Multi-User Chat Channels on Slack. *Proc. ACM Hum.-Comput. Interact.* 6, CSCW1, Article 94 (2022), 14 pages. https://doi.org/10.1145/3512941
[88] Yanhao Wang, Qi Fan, Yuchen Li, and Kian-Lee Tan. 2017. Real-time influence maximization on dynamic social streams. *PVLDB* 10, 7 (2017), 805–816.
[89] Duncan J. Watts and Steven H. Strogatz. 1998. Collective dynamics of 'small-world' networks. *Nature* 393, 6684 (1998), 440–442. https://doi.org/10.1038/30918 TLDR: Simple models of networks that can be tuned through this middle ground: regular networks 'rewired' to introduce increasing amounts of disorder are explored, finding that these systems can be highly clustered, like regular lattices, yet have small characteristic path lengths, like random graphs..
[90] Hao-Hsiang Wu and Simge Küçükyavuz. 2018. A two-stage stochastic programming approach for influence maximization in social networks. *Computational Optimization and Applications* 69, 3 (2018), 563–595.
[91] Xunlian Wu, Anqi Zhang, Yining Quan, Qiguang Miao, and Peng Sun. 2024. Dynamic linear threshold model with decay for influence spread. https://doi.org/10.36227/techrxiv.20106149.v5
[92] Xuejun Yan, Yifan Wang, Yilong Zhang, and Xuefeng Zhang. 2023. A comprehensive bridge detection method for complex networks. *KSII Transactions on Internet and Information Systems (TIIS)* 17, 2 (2023), 496–515. https://doi.org/10.3837/tiis.2023.02.012
[93] Wenguo Yang, Shengminjie Chen, Suixiang Gao, and Ruidong Yan. 2020. Boosting node activity by recommendations in social networks. *Journal of Combinatorial Optimization* 40, 3 (2020), 825–847.
[94] Wenguo Yang, Jianmin Ma, Yi Li, Ruidong Yan, Jing Yuan, Weili Wu, and Deying Li. 2019. Marginal gains to maximize content spread in social networks. *IEEE Transactions on Computational Social Systems* 6, 3 (2019), 479–490.
[95] Hao Yin, Austin R. Benson, Jure Leskovec, and David F. Gleich. 2017. Local Higher-Order Graph Clustering. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Halifax, NS, Canada, August 13 - 17, 2017*. ACM, 555–564. https://doi.org/10.1145/3097983.3098069