PolicySim: An LLM-Based Agent Social Simulation Sandbox for Proactive Policy Optimization

Renhong Huang
Zhejiang University
Hangzhou, China
renh2@zju.edu.cn

Ning Tang
Fudan University
Shanghai, China
ningtang24@m.fudan.edu.cn

Jiarong Xu*
Fudan University
Shanghai, China
jiarongxu@fudan.edu.cn

Yuxuan Cao
HKUST
Hong Kong, China
ycaoce@connect.ust.hk

Qingqian Tu
University of Nottingham
Nottingham, United Kingdom
psxqt2@nottingham.ac.uk

Sheng Guo
MyBank, AntGroup
Hangzhou, China
guosheng.guosheng@mybank.cn

Bo Zheng
MyBank, AntGroup
Hangzhou, China
guangyuan@mybank.cn

Huiyuan Liu
PowerChina Huadong Engineering Corporation Limited
Hangzhou, China
liu_hy8@hdec.com

Yang Yang
Zhejiang University
Hangzhou, China
yangya@zju.edu.cn

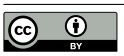
Abstract

Social platforms serve as central hubs for information exchange, where user behaviors and platform interventions jointly shape opinions. However, intervention policies like recommendation and content filtering, can unintentionally amplify echo chambers and polarization, posing significant societal risks. Proactively evaluating the impact of such policies is therefore crucial. Existing approaches primarily rely on reactive online A/B testing, where risks are identified only after deployment, making risk identification delayed and costly. LLM-based social simulations offer a promising pre-deployment alternative, but current methods fall short in realistically modeling platform interventions and incorporating feedback from the platform. Bridging these gaps is essential for building actionable frameworks to assess and optimize platform policies. To this end, we propose PolicySim, an LLM-based social simulation sandbox for the proactive assessment and optimization of intervention policies. PolicySim models the bidirectional dynamics between user behavior and platform interventions through two key components: (1) a user agent module refined via supervised fine-tuning (SFT) and direct preference optimization (DPO) to achieve platform-specific behavioral realism; and (2) an adaptive intervention module that employs a contextual bandit with message passing to capture dynamic network structures. Experiments show that PolicySim can accurately simulate platform ecosystems at both micro and macro levels and support effective intervention policy.

CCS Concepts

• **Computing methodologies** → **Modeling and simulation**.

*Corresponding author.



WWW '26, Dubai, United Arab Emirates.

ACM ISBN 979-8-4007-2307-0/2026/04
https://doi.org/10.1145/3774904.3792555

Keywords

Social Simulation; Large Language Model; Multi-Agent

ACM Reference Format:
Renhong Huang, Ning Tang, Jiarong Xu, Yuxuan Cao, Qingqian Tu, Sheng Guo, Bo Zheng, Huiyuan Liu, and Yang Yang. 2026. PolicySim: An LLM-Based Agent Social Simulation Sandbox for Proactive Policy Optimization. In *Proceedings of the ACM Web Conference 2026 (WWW '26), April 13–17, 2026, Dubai, United Arab Emirates.* ACM, New York, NY, USA, 12 pages. https://doi.org/10.1145/3774904.3792555

1 Introduction

In today's digital era, social platforms have become the core infrastructure for social interaction and information exchange [26, 29, 61]. Platform intervention policies fundamentally shape users' interactive behaviors including information consumption, content sharing, and social engagement, through intervention mechanisms such as recommender systems [30, 66, 68, 71], content filtering [53], and exposure control [15]. Collectively, these intervention mechanisms shape users' online experiences and exert a profound influence on opinion formation and decision-making processes [10, 15, 16].

However, inappropriate platform intervention policies can trigger unintended and even deleterious consequences, as social platforms' inherent tendency to amplify information can magnify the negative effects of such policies beyond their initial expectations [30, 53, 71]. For instance, numerous studies have demonstrated that recommender systems can foster the emergence of echo chambers and filter bubbles [6, 10], thereby suppressing cross-viewpoint dialogue and reducing stance diversity [5]. At the same time, platform interventions that prioritize user engagement may increase polarization and conflict, undermining public trust and attracting regulatory scrutiny [50]. Consequently, a key research challenge is to *assess the potential social impacts of intervention policies in a proactive and systematic manner prior to deployment.*

Existing evaluation methods primarily rely on the A/B testing to assess intervention policies [17], requiring direct deployment in real user environments to collect behavioral feedback, as illustrated in

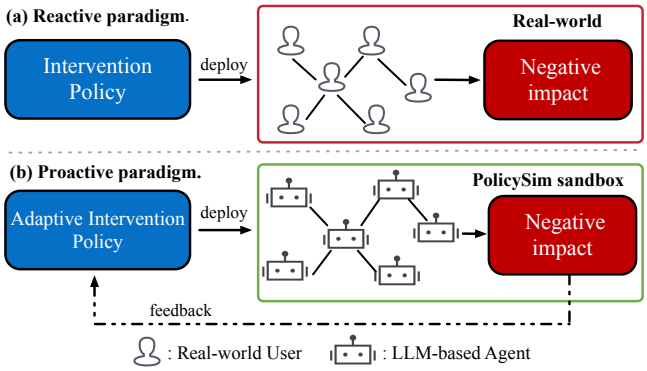


Figure 1: Compared to A/B testing, which *reactively* assesses interventions by deploying and learning only after outcomes are observed, PolicySim *proactively* assesses and optimizes intervention policies prior to deployment via feedback.

Figure 1(a). However, this paradigm suffers from several limitations: (i) evaluation is reactive rather than proactive; (ii) feedback loops are often delayed and therefore cannot keep pace with rapid platform dynamics; and (iii) directly testing in real-world environments may introduce uncontrollable and potentially irreversible harmful consequences. Thus, reliance on A/B testing alone is insufficient for the prospective evaluation of intervention policies, highlighting the need for proactive, risk-aware alternatives.

Recent advances in large language model (LLM)-based simulation, enabled by their strong generative and reasoning capabilities, offer a promising direction. Prior works include traditional agent-based models [23, 54] and LLM-based agent simulation frameworks [7, 38, 45, 67]. Both types of models have been used to study complex social phenomena, including opinion dynamics [8], economic systems [35], and social norm alignment [52]. Nevertheless, several challenges remain unresolved in current works. (i) Most simulations do not explicitly model platform intervention policies, making it difficult to accurately capture their effects. (ii) Their agent design often relies heavily on prompt engineering rather than realistic modeling of social media behavior, which limits the credibility of simulated outcomes. (iii) Existing frameworks lack principled mechanisms for leveraging simulation feedback to optimize real-world intervention policies, limiting the use of simulation outcomes for iterative improvement of real-world policies.

To bridge these gaps, we present PolicySim, a LLM-based multi-agent social simulation sandbox for the proactive assessment and optimization of intervention policies. The framework captures the bidirectional dynamics between intervention strategies and ecosystem evolution, as well as the influence of interventions on user behavior patterns. To achieve this, the framework introduces a user agent module and an intervention policy module. Specifically, the user agent module is introduced as a novel paradigm for training social agents that integrates supervised fine-tuning (SFT) with direct preference optimization (DPO). This unified approach jointly enhances the behavioral faithfulness of agents to platform-specific user data and improves the distinctiveness of user intent representations. Building upon the simulation, the intervention policy module is introduced to collect rewards from the sandbox and adaptively optimize intervention policies. This module employs a contextual bandit algorithm that balances exploration and exploitation while leveraging message passing to capture dynamic user networks. We incorporate multiple intervention policies, such as *recommender systems* and *exposure control*), deployed on X and Weibo platforms.

	Scale	Relation	IP.	AI.	Env.
PolicySim	1000	✓	✓	✓	X & Weibo
Oasis [67]	1M	✓	✓	✗	X & Reddit
Agent4rec [70]	1000	✓	✓	✗	Movie Rec
HiSim [46]	700	✗	✗	✗	X
Stopia [73]	2	✗	✗	✗	-

Table 1: Comparison of our system with recent social simulation frameworks. Scale: number of LLM agents; Relation: whether follow/follower links evolve; IP/AI: presence of intervention policy and adaptive interventions; Env.: underlying platform ("–" means unspecified).

Our contributions are summarized as follows:

- We propose PolicySim, a LLM-based multi-agent social simulation sandbox for the proactive assessment and optimization of intervention policies.
- To enhance LLM-based agent simulation fidelity, we propose, for the first time, a unified paradigm for training social agents that combines SFT and DPO, which ensures behavioral alignment with platform data while capturing diverse user intents.
- To enable adaptive optimization of intervention policy, we employ a contextual bandit framework that balances exploration and exploitation, augmented with message passing to capture dynamic network structures and information flows.
- Extensive experiments across multiple datasets verify the realism of agent behaviors and the effectiveness of intervention optimization, showing that PolicySim enables scalable and proactive assessment of intervention policies.

The paper is organized as follows. §2 presents the social simulation sandbox, comprising user-agent and intervention-policy modules. Building on this sandbox, §3 introduces adaptive intervention policy based on a bandit algorithm for automated feedback and optimization. Finally, §4 evaluates the effectiveness of our framework through simulation and intervention experiments.

2 Simulation Sandbox Framework

In this section, we first present the framework of PolicySim, as illustrated in Figure 2. We primarily follow the architecture of HiSim [46] to help us build the framework. The entire framework consists of the user agent module (see §2.1) and the intervention policy module (see §2.2). Each agent represents an LLM-powered user with authentic profile, accumulates memories across simulation rounds, and can simulate the interaction between intervention policies and social ecosystem evolution.

2.1 User Agent Module

To enable LLMs to simulate social user behaviors, we equipped agents with specialized modules including user profile, user behavior, memory, training and planning. Based on these designs, agents are empowered to emulate realistic user decision-making processes.

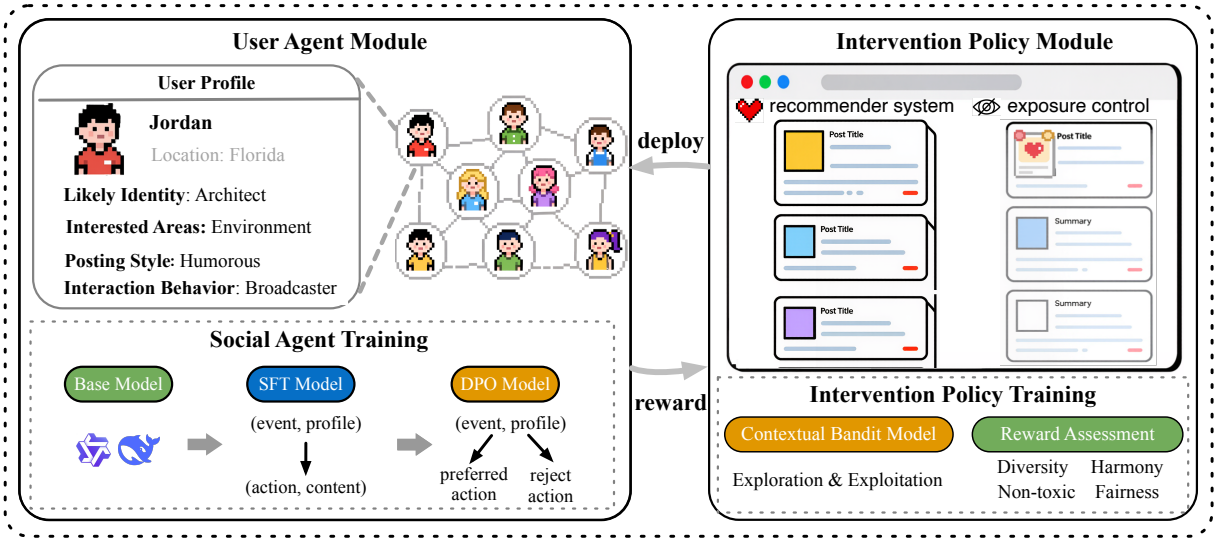


Figure 2: The architecture of PolicySim is composed of two main modules: the User Agent Module and the Intervention Policy Module. The User Agent Module contains detailed components including user profiles, memory, user relations, and behavioral models. The bottom-left panel illustrates how we train agent tailored for social media environments to capture realistic user behaviors. The Intervention Policy Module instantiates typical platform mechanisms such as recommender systems and exposure control. By simulating intervention policies within sandbox framework, the Target Reward Assessment component evaluates their performance and provides feedback, which are further utilized to adaptively optimize intervention policy.

2.1.1 User Profile. User profiles are crucial for capturing interests, behaviors, and preferences, supporting tasks such as personalized recommendation, behavior prediction, and social agent modeling [49, 64, 70]. As explicit profiles are mostly unavailable, we construct them from post content and user metadata, extracting four high-level attributes for agent prompting:

- Likely identity: Infers social or professional roles, enabling domain-specific terminology and perspectives.
- Interested areas: Identifies user interests from posts, hashtags, and retweeted accounts (e.g., a user engaging with # ClimateEmergency is labeled with an environment-related interest).
- Posting style. Linguistic traits inferred from posts, emojis, slang, and sentiment.
- Interaction behavior. Quantifies engagement via post/retweet ratios, reply frequency, and follower/following ratios, categorizing users by interaction roles.

Let U be the set of users. For each $u_i \in U$, we denote their profile by $\phi(u_i)$. Details on the extraction of the four high-level attributes and the social media metadata format are provided in Appendix A.2.

2.1.2 User Behavior. During each simulation round, agents undergo behaviors calling phase, interacting with the environment and executing various behaviors. Our agent architecture extends conventional social interactions [45] (tweet, retweet, reply, like, dislike, do nothing) by adding two relationship actions, *follow* and *unfollow*, enabling the network topology to evolve dynamically in real time according to agents' content preferences and behaviors. Unlike prior work [44, 45] executing a single action per round, we implement *Multiple Behavior Selection* in the behavior calling phase, capturing diverse engagement patterns and enhancing interaction variety to better reflect real-world social media dynamics. Detailed definitions of behaviors and the prompts used for behavior generation are provided in Appendix A.2.

2.1.3 Relation. The social media network is typically modeled as dynamic graph $G_t = (V, E_t)$ at simulation round t, where V is the set of users and E_t contains directed edges representing follow relationships. An edge $e_{ij} \in E_t$ indicates that agent u_i follows or interacts with agent u_j at round t. The network evolves over time, with edges updated dynamically across rounds due to *follow* and *unfollow* actions, yielding a time-varying topology $\{G_1, \ldots, G_T\}$.

2.1.4 User Stance. To model group opinion dynamics on social media, we quantify user agents' stances toward events using LLMs, which capture implicit attitudes and subtle textual patterns, and have shown high accuracy in stance detection [12, 32].

Specifically, a user's stance at simulation round t is discretely classified as $-1, 0, 1$ for negative, neutral, and positive, respectively. For user u_i, the discrete stance score is inferred as:

$$\bar{s}^t(u_i) = f_{\tau_1}\left(p(u_i), \text{Action}(u_i)[:t]\right) \in \{-1, 0, 1\}, \tag{1}$$

where f_{τ_1} denotes the LLM conditioned on prompt τ_1 and the user's history. Further, to reduce noise from LLM hallucinations [25], we apply an exponential moving average:

$$s^t(u_i) = \alpha s^{t-1}(u_i) + (1-\alpha)\bar{s}^t(u_i), \tag{2}$$

with smoothing coefficient $\alpha \in (0, 1)$ and initialization $s^0(u_i) = \bar{s}^0(u_i)$. A larger α emphasizes past stances in the current score.

2.1.5 Memory. Memory encodes, stores, and retrieves information to influence future actions [2, 56]. We model user memory

with short-term and long-term components to capture dynamic information retention and retrieval [47].

Short-term memory. This component handles temporary storage and fast processing. Humans typically retain only the most salient content, varying across messages and contexts. Therefore, for user u_i, the k-th short-term memory m_k is generated as:

$$m_k = f_{\tau_2}(c, \phi(u_i), m_{k-1}), \tag{3}$$

where c is post content and f_{τ_2} directs the LLM via prompt τ_2. Memories are stored in *memory pool* $\{m_k, \text{emb}(m_k), t\}$, including content, embedding by embedding model, and simulation round.

Long-term memory. Long-term memory retains information of lasting significance, such as past experiences or high-level insights. Retrieval samples memories from the pool, prioritizing those with high semantic relevance and accounting for temporal decay:

$$\Pr(m_k) \propto e^{-\lambda \Delta_t} \text{Sim}(\text{emb}(c), \text{emb}(m_k)), \tag{4}$$

where $\lambda > 0$ is a decay rate, Δ_t is the elapsed time, and $\text{Sim}(\cdot,\cdot)$ measures semantic similarity. Sampled short-term memory entries $\{m\}$ are integrated into long-term memory $\hat{m}_k$ via $\hat{m}_k = f_{\tau_3}(c, \phi(u_i), \{m\})$ with f_{τ_3} guiding the LLM using prompt τ_3.

To reduce computational cost during memory generation process, short posts below a length threshold bypass LLM processing and are directly added to the memory pool. Full algorithmic details are provided in Appendix A.2.

2.1.6 Agent Training and Planning. Existing multi-agent designs largely rely on prompt engineering. While these expert-crafted prompts can improve plausibility, they lack alignment with real-world social media data. Social media agents exhibit heterogeneous, context-sensitive behaviors with long-term dependencies. Therefore, to better capture authentic user behaviors, we train agents directly on platform data rather than relying on prompt-based heuristics, and we adopt a two-stage framework combining Supervised Fine-Tuning (SFT) and Direct Preference Optimization (DPO) to preserve both stylistic and behavioral consistency.

SFT as cold start. Given a collection of (event, user, action) tuples $\{(e_i, u_i, a_i)\}_{i=1}^M$, we first construct a dataset of instruction-action pairs $\mathcal{D}_{\text{SFT}} = \{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i=1}^M$. Each instruction sequence $\boldsymbol{x}_i$ concatenates the topic description and the associated user profile, i.e. $\boldsymbol{x}_i = [e_i, \phi(u_i)]$, which jointly defines the situational context and the characteristics of target user. The response $\boldsymbol{y}_i$ combines observed user action with its corresponding textual content. The policy model, parameterized by θ, is first trained to minimize the negative conditional log-likelihood over the response sequence:

$$\mathcal{L}_{\text{SFT}}(\theta) = -\sum_{i=1}^{N} \sum_{t=1}^{L_i} \log P\left(y_{i,t} \mid \boldsymbol{x}_i, \boldsymbol{y}_{i,<t}; \theta\right), \tag{5}$$

where L_i is the length of the response sequence for the i-th sample and $y_{i,t}$ denotes the t-th token in the sequence.

RL Post Training. With SFT as semantic grounding, we further employ Direct Preference Optimization (DPO) [51], explicitly aligning the agent's behaviours with desired social behaviors observed. We construct a preference dataset $\mathcal{D}_{\text{DPO}} = \{(\boldsymbol{x}_i, \boldsymbol{y}_i^+, \boldsymbol{y}_{i,j}^-)_{j=1}^J\}_{i=1}^M$, where $\boldsymbol{x}_i$ represents the same prompt used in SFT, $\boldsymbol{y}_i^+$ denotes the preferred response, and $\boldsymbol{y}_i^-$ is a rejected alternative generated by the pretrained model under the same prompt. Specifically, multiple candidate actions for a given (e_i, u_i) pair are produced by prompting the base model to generate. Among these, J samples that exhibit low semantic similarity to $\boldsymbol{y}_i^+$ or differ in action choice are retained as negative options to guide preference learning.

Starting from a reference policy π_{ref} initialized with the SFT model, DPO further optimizes the policy π_θ by maximizing the DPO loss $\mathcal{L}_{\text{DPO}}(\theta)$, formulated as:

$$-\mathbb{E}_{(\boldsymbol{x},\boldsymbol{y}^+,\boldsymbol{y}^-)\sim\mathcal{D}_{\text{DPO}}}\left[\log\sigma\Big(\beta\Big[\log\frac{\pi_\theta(\boldsymbol{y}^+ \mid \boldsymbol{x})}{\pi_{\text{ref}}(\boldsymbol{y}^+ \mid \boldsymbol{x})} - \log\frac{\pi_\theta(\boldsymbol{y}^- \mid \boldsymbol{x})}{\pi_{\text{ref}}(\boldsymbol{y}^- \mid \boldsymbol{x})}\Big]\Big)\right]. \tag{6}$$

This objective guides the policy with users' behavioral preferences, enabling the agent to align its responses with realistic social actions observed in human data.

We also compared different training strategies as well as prompt-instructed agents, the experiment results can be found in § 4. Furthermore, we utilize CoT reasoning [62] to enhance the interpretability of agent behaviors. In addition, we further provide a theoretical discussion on the rationale of using LLMs for learning compared to relying solely on ABM models as multi-agent systems, which would produce inseparable "users" rather than realistic user populations. Details are provided in Appendix A.4.

2.2 Intervention Policy Module

There are various types of intervention policies on social platforms, which can have profound impacts on user behavior, information propagation, and platform ecosystems. In this section, we introduce commonly adopted intervention policies in social media scenarios, along with the typical objectives these interventions aim to achieve.

2.2.1 Intervention Policy. In our work, we primarily focus on the following two intervention policies.

Recommender System. The recommender system acts as a control mechanism for regulating information access, thereby playing a pivotal role in shaping the dynamics of information flow within the platform [39]. Typically, the recommender system delivers relevant content aggregated from three primary sources [43]:

- Relational recommendation: Posts from users that an agent follows, reflecting direct social connections. A message posted at time t becomes visible to followers at $t+1$.
- Personalized recommendation: The dominant channel in social platforms, delivering content tailored to individual preferences by prioritizing posts semantically aligned with users' historical behavior [36] and profile representations.
- Headline recommendation: This channel provides non-personalized content such as trending topics or headline news, curated to highlight widely popular information.

By combining these channels, the recommender system regulates the information accessible to each user, shaping engagement patterns and balancing individual and global exposure.

Exposure control mechanisms. Exposure control mechanisms regulate the visibility of content to specific user groups, serving as a tool to simulate moderation, content prioritization, or fairness-oriented interventions [42, 59]. Formally, for a given user $u_i \in U$, we define its exposure probability at time step t as $\exp(u_i) \in [0, 1]$,

representing the likelihood that a post generated by u_j passes the platform's filtering. By adjusting parameter $\exp(u_i)$ for each user, the platform can increase or suppress the exposure of particular user groups, thereby emulating interventions such as promoting underrepresented content, reducing the spread of misinformation, or mitigating echo chamber effects [11].

2.2.2 Intervention Objective. The objectives of intervention policies in social media are to guide platform dynamics towards desirable outcomes. Specifically, our interventions are designed to achieve the following goals separately:

- **Promote cross-viewpoint interactions.** Encourage engagement between users with opposing stances, fostering diverse perspectives without increasing toxicity [45, 60].
- **Mitigate misinformation.** Limit the visibility and propagation of misleading content, improving information reliability.

3 Adaptive Intervention Policy

Built on the PolicySim sandbox, we obtain feedback from multi-agent interactions in the simulated environment. These signals guide the adaptive optimization of the intervention policy toward target objectives. We next formalize the problem definition.

Problem Definition: Consider a social platform $\mathcal{S}_t = (U, G_t, R)$ at round t, where U is the user set, G_t the social network induced by user interactions, and R the intervention policy. Let $\rho(\mathcal{S}_t)$ be a utility function measuring how well the platform objectives are achieved. As for adaptive intervention policy, agents interact under policy R, producing feedback signals that reflect changes in $\rho(\mathcal{S}_t)$. These signals are then used to adaptively update R within the action space $\mathcal{A}$ to maximize the expected utility $\mathbb{E}[\rho(\mathcal{S}_t)]$.

This problem is practically motivated. For instance, when the intervention policy involves recommender systems, it typically relies on large amounts of historical data for offline training [28, 31]. A key shortcoming of such policy is their inability to optimize using interaction signals derived from multi-agent environments.

3.1 Adaptive Interventions via RL

The adaptive intervention problem can be naturally framed as reinforcement learning task: By interacting with the environment, we can apply desired impact as a reward and use appropriate reinforcement learning algorithms to enable the agent to achieve the maximum cumulative reward in a dynamic environment.

To enable real-time adaptation to evolving user behaviors, we employ contextual multi-armed bandits, which provide a lightweight and flexible framework [4, 27, 34]. Formally, intervention process is modeled over T rounds. At each round t, the policy selects from n arms, $X^t = x_1^t, \ldots, x_n^t$, where each arm represents a candidate action from the action space $\mathcal{A}$. Pulling arm x_i^t yields a reward:$r_i^t = \psi(x_i^t) + \xi_i^t$, where ψ maps the arm's context to the reward, and ξ_i^t is zero-mean noise. Following prior work [4], rewards are bounded in $[0, 1]$.

Action Space. The action space depends on the type of intervention policy. For recommender systems, each arm represents a user-post pair, whereas for exposure control mechanisms, each arm corresponds to a user-probability pair. After defining the arm type, we construct a discrete action space by sampling candidate sets. For recommender systems, we sample the post set $P\text{cand} \subseteq P_t$ from historical posts and the user set $U\text{cand} \subseteq U$, preferring posts not previously recommended. The final candidate arms are the Cartesian product $U\text{cand} \otimes P\text{cand}$.

Context Design. Context embedding of each arm combines information from both the user and the post. For a user u_i, we define the context embedding as:

$$X_{\text{user}}(u_i) = \text{emb}\Big([\, \phi(u_i) : \{m\} \,] \Big), \tag{7}$$

where $\phi(u_i)$ is the user profile and $\{m\}$ represents the user's recent memory. X_{user} denotes the context embedding matrix for all users.

To capture social influence, we further propagate context embeddings across the social network, inspired by label propagation [74]. Let G_t be the social graph with adjacency matrix A and degree matrix D. We iteratively update user embeddings via:

$$X_{\text{user}}^k = \gamma X_{\text{user}}^{k-1} + (1-\gamma) D^{-1} A X_{\text{user}}^{k-1}, \tag{8}$$

where $\gamma \in [0, 1]$ balances self-information and neighbor aggregation. After k iterations, each user embedding incorporates from k-hop neighbors. Finally, we concatenate the propagated user embedding with the post's embedding to obtain the arm context.

Reward Assessment. The reward is equal to $\rho(\mathcal{S}_t)$, reflecting the objective of the adaptive intervention algorithm. Different intervention goals correspond to different reward formulations. We here provide reward for objectives in § 2.2.2:

(1) Promote cross-viewpoint interactions: Inspired by the commonly used evaluation metrics [46, 60, 67], the reward r_i^t for arm x_i^t balances: (1) engagement across opposing stances, (2) penalizing toxic interactions, and (3) preserving overall user engagement. Formally, given a post-user arm x_i^t and the reaction o_i^{t+1} from the receiver agent (see §2.1.2), the reward for arm x_i^t is

$$r_i^t = \frac{|s^t(u_s) - s^t(u_r)|}{2} h(o_i^{t+1}) \max\left(0, 1 - \mu, \tau(o_i^{t+1})\right), \tag{9}$$

balancing stance divergence between sender u_s and receiver u_r, penalizing toxic reactions $\tau(o_i^{t+1}) \in [0, 1]$ (via Perspective API [33]), and weighting the reaction type by engagement through $h(o_i^{t+1})$.

(2) Mitigate misinformation: Similarly, we compute the reward based on the environment's reaction. Specifically, for arm x_i^t,

$$r_i^t = \text{mis}^{t-1}(u) - \text{mis}^t(u), \tag{10}$$

where $\text{mis}^t(u)$ indicates whether user u is misinformed at round t.

Optimization Process: In the bandit optimization process, we draw inspiration from the works [4, 27] and consider both the exploitation and exploration framework, as detailed below.

For the exploitation aspect, actions are made based on the knowledge or experience already acquired. We here use a neural network g to learn the mapping $g_{\theta^t} : x_i^t \to r_i^t$, where the context of the arms is mapped to the reward with learnable parameters θ^t. After executing an arm x_i^t, we receive the reward r_i^t at the next simulation epoch $t+1$. Therefore, we perform gradient descent to update θ^t based on the collected feedback $\{x_i^t, r_i^t\}$.

In addition to exploiting the contexts, model should explore new possibilities in unknown environments to discover potentially better policies. Inspired by [4], we utilize neural network $\hat{g}_{\phi^t}$ to estimate the potential gain in terms of reward for exploration, where

potential gain measures the discrepancy between the observed reward and the predicted reward $r_i^t - g_{\theta^t}(x_i^t)$. A large positive potential gain indicates that the arm is more under-explored, while a small potential gain suggests an overestimation of the reward, making it less suitable for exploration.

Since the potential gain is bounded by the gradient of the predicted reward $\nabla_{\theta_t} g(\mathbf{x}_i^t)$ in [4], the gradient of the predicted reward is used as input to measure the potential gain $\hat{g}_{\phi^t} : \nabla_{\theta_t} g(x_i^t) \rightarrow r_i^t - g_{\theta^t}(x_i^t)$. After executing an arm x_i^t, we receive the reward r_i^t at the next simulation epoch $t+1$. Therefore, we perform gradient descent to update ϕ^t based on the collected feedback $\left\{\nabla_{\theta_t} g(x_i^t), r_i^t - g_{\theta^t}(x_i^t)\right\}$. Finally, to balance exploitation and exploration, we output score $g_{\theta^t}(x_i^t) + \hat{g}_{\phi^t}(\nabla_{\theta_t} g(x_i^t))$, which is used to select the arms by ranking.

4 Experiments

In this section, we evaluate PolicySim in two stages: first, the realism of the social simulation, and second, the effectiveness of adaptive intervention policy (Code is available at https://github.com/renH2/PolicySim.). Our experiments address two key questions:

1. How valid is the simulation generated by PolicySim? We propose several metrics to identify suitable agents for social simulation and validate PolicySim at both micro- and macro-levels.

2. Can the adaptive intervention policy effectively optimize platform outcomes? We test our adaptive intervention policy under various objectives, showing PolicySim can leverage environmental feedback to optimize policies.

4.1 Experimental Setup

4.1.1 Datasets. In our experiments, we use real-world social media datasets: TwiBot-20 dataset [14]. TwiBot-20, collected from July to September 2020, comprises 229K users, 33.5M tweets, and 456K follow links. We extract social activity events and relevant user groups while preserving their relational structures. In addition, we further conduct experiments on the Weibo dataset [41]. Detailed statistics of the dataset are provided in Appendix A.2, and the experiments conducted on the Weibo dataset are presented in Appendix A.3.

4.1.2 Simulation evaluation metric. The evaluation metrics for the simulation encompass both **micro-level** and **macro-level** perspectives. At the **micro-level**, we focus on assessing whether social agents' decision-making behaviors align with real-world patterns. Specifically, we evaluate four aspects:

- **Content quality.** We assess whether agents can produce realistic content by comparing generated posts with real user posts using multiple textual similarity metrics, including *BERTScore F1* and *BertSim* (BERT-based cosine similarity), to capture both semantic and lexical closeness.
- **Behaviour alignment.** We measure the ability of agents to replicate human-like behavioral actions (e.g., posting, retweeting) through prediction accuracy.
- **Self-consistency.** We measure whether agents can correctly identify their own generated posts by prediction accuracy, reflecting the self-consistency of behaviors.
- **Social capability.** We further employ large language models as evaluators (*LLM-as-a-Judge*) to rate each agent on three dimensions: (1) *Engagement*: the agent's ability to participate in natural, meaningful, and contextually appropriate interactions, such as replying, expressing opinions, and conveying emotions; (2) *Robustness*: the agent's capacity to maintain relevance, coherence, and naturalness across diverse social contexts, including discussions, debates, humor, and trending topics; (3) *Suitability*: an overall measure of the agent's behavioral realism, reflecting how convincingly the agent mimics authentic human activity.

Notably, *Engagement* and *Robustness* are measured from 1 to 4 ordinal scale, whereas *Suitability* is evaluated from 0 to 100 on continuous scale.

At the **macro level**, we measure how distribution of stances evolves over time to assess whether agents reproduce realistic opinion dynamics. In addition, we examine how intervention policies affects network-level phenomena such as polarization.

4.1.3 Backbones and Baselines. We evaluate PolicySim through both simulation and adaptive intervention experiments. For simulation, PolicySim is built upon five open-source LLMs used in a prompt manner, including GLM4-light, Llama-3-8B-Instruct, and the Qwen2.5 series. To examine the effect of each training component, we conduct ablations: (1) PolicySim-ϕ (without user profile generation); (2) PolicySim-SFT (only supervised fine-tuning); and (3) PolicySim-DPO (only DPO without SFT initialization). For intervention, we compare withϵ-greedy and UCB bandit baselines.

4.1.4 Implementation details. In practice, we set the attenuation coefficient α in Eq.(2) to 0.8, and use $\lambda = 1$, $k = 1$, $\mu = 4$, and $\beta = 0.5$ in all experiments. For the macro-level results in § 4.2 and Objective 1 in § 4.3, we use the social topic *Anti-abortion Legislation* as the simulation context, collecting 10 trigger news items from late June 2022 (after the overturning of *Roe v.Wade*) via *newsapi.ai* and summarizing them with GPT-4o. For Objective 2, we initialize 20% of users to post misinformation—"#wakeupamerica who needs a #gun registry when #obama has all your personal information"—and observe its spread in the simulated environment.

Owing to its strong capability on social interaction, The LLM we employ to power the user agents is Qwen2.5-3B-Instruct [58], with a maximum context length of 32,768 tokens. For model training, we use a LoRA adaptation of rank 64 to finetune the base model with a learning rate of 1×10^{-6}, batch size of 256. In the DPO stage, we set the temperature coefficient $\beta = 0.1$, learning rate 5×10^{-7} and sample size $J = 3$. Both stages are trained for up to 10 epochs until convergence. All experiments were conducted on 12 NVIDIA A100-PCIE-40GB GPUs. More details can be found in Appendix A.2.

4.2 Simulation Results

4.2.1 Micro-level evaluation. As shown in Table 2, we first report the micro-level simulation results of PolicySim. In terms of **content quality**, LLM-based agents effectively reproduce the linguistic and semantic characteristics of real Twitter posts, achieving a high *BertSim* score and *BERTScore F1*, demonstrating its ability to generate realistic, human-like text. For **behaviour alignment**, PolicySim improves the accuracy of reproducing human behavioral

Method	Content quality		Behavior alignment	Self-consistency	Social capability		
	BERTScore F1 ↑	*BertSim*↑	*Accuracy* ↑	*Accuracy* ↑	*Engagement* ↑	*Robustness* ↑	*Suitability* ↑
Random	28.55 ±4.18	74.42 ±12.08	36.11 ±25.30	21.20 ±1.65	2.65 ±0.55	2.31 ±0.54	2.11 ±0.14
GLM4-light [18]	46.01 ±10.35	81.95 ±13.64	48.33 ±27.64	47.20 ±27.64	2.91 ±0.40	2.70 ±0.57	57.64 ±0.50
Llama-3-8B-Instruct [1]	46.32 ±13.07	85.22 ±6.81	52.36 ±26.21	45.60 ±25.95	3.16 ±0.49	3.51 ±0.50	71.85 ±0.45
Qwen2.5-0.5B-Instruct [58]	46.64 ±23.99	81.38 ±9.03	47.78 ±22.74	24.30 ±19.52	2.95 ±0.45	2.43 ±0.57	47.22 ±0.50
Qwen2.5-3B-Instruct	48.26 ±12.57	85.91 ±6.35	60.56 ±26.87	40.40 ±26.98	3.17 ±0.47	2.65 ±0.61	52.41 ±0.50
Qwen2.5-7B-Instruct	49.48 ±11.77	85.52 ±6.63	50.56 ± 25.60	51.20 ±29.98	3.29 ±0.47	2.71 ±0.61	63.83 ±0.48
PolicySim-ϕ	45.16 ±14.91	80.15 ±8.23	58.33 ±29.95	27.20 ±21.82	3.04 ±0.42	2.52 ±0.57	55.17 ±0.50
PolicySim-SFT	52.66 ±15.53	86.77 ±7.55	54.44 ±22.91	56.40 ±25.83	3.00 ±0.42	2.42 ±0.53	44.83 ±0.50
PolicySim-DPO	47.95 ±13.24	83.20 ±8.13	53.89 ±24.41	50.40 ±25.37	3.14 ±0.47	2.67 ±0.58	61.27 ±0.49
PolicySim	58.05 ±15.96	88.06 ±6.32	65.56 ±19.71	56.00 ±25.61	3.20 ±0.44	2.73 ±0.61	59.44 ±0.49

Table 2: Micro-level performance of social simulation across different backbones and PolicySim on the TwiBot-20 dataset. Underline indicates the best result and results are averaged over five runs with standard deviations.

Method	Objective 1			Objective 2
	Stance	Toxicity ↓	Cross interactions ↑	Misinformation ratio ↓
Origin	0.014 (0.47)	0.0556	0.04	40%
ϵ-greedy	0.184 (0.42)	0.0426	0.14	26%
UCB	0.026 (0.34)	0.0628	0.50	30%
PolicySim	0.376 (0.48)	0.0386	0.56	24%

Table 3: Performance of different intervention policies across different objectives. Underline indicates the best performance.

patterns by 8.26% over random and backbone baselines, highlighting the effectiveness of modeling user interaction during agent training. PolicySim also exhibit strong self-consistency, with an accuracy gain of 10.15%, indicating coherence between generated content and underlying behavioral preferences. Regarding **social capability**, PolicySim achieves superior scores in *Engagement* and *Robustness*, demonstrating its ability to emulate context-aware and socially coherent user behaviors. Notably, Llama-3-8B-Instruct performs better on *Suitability*, likely due to its instruction tuning or pre-traning data that promote more socially appropriate responses.

As for the comparison between PolicySim and other baseline models, the results reveal that the Qwen series exhibits consistent improvements with increasing model scale across most metrics (except for *BertSim* and accuracy of behavior alignment), confirming the applicability of scaling laws in social simulation tasks. Besides, the performance of PolicySim–ϕ, which removes the user profile module, declines apparently compared to PolicySim, underscoring the pivotal role of user profiling in shaping personalized agent behaviors. Finally, the inferior performance of PolicySim-DPO relative to PolicySim-SFT and PolicySim indicates that supervised fine-tuning serves as a necessary foundation for acquiring the core structure, style, and knowledge of social interactions before applying DPO to achieve optimal performance.

4.2.2 Macro-level evaluation. We simulate real-world dynamics by chronologically injecting trigger news events into the environment. Each news event is designed to shift public opinion. For example, in rounds 0 and 4, agents are exposed to news headlines "The Supreme Court overturns Roe v. Wade" and "The Biden administration announced plans to continue covering abortion medication," respectively. In round 7, large-scale protests demanding federal action to restore abortion rights are introduced. Agents supporting Anti-Abortion Legislation are labeled as having a positive stance, while opponents are labeled as negative.

As shown in Figure 3, the mean stance across agents exhibits a rapid initial decline, followed by a gradual increase, mirroring the expected public opinion trajectory. This validates the realism of our simulation in capturing macro-level stance dynamics. Moreover, the std. of stance scores increases over rounds, indicating a polarization effect as agents form more extreme opinions. Notably, when the intervention policy (i.e., recommender system) is applied, polarization intensifies: users are increasingly exposed to homogeneous content, reinforcing existing beliefs and amplifying divisions.

4.2.3 Scalability. Figure 4 illustrates the execution time across different numbers of agents over 10 simulation rounds. The overall runtime is primarily determined by the latency of LLM inference (or API calls) and the execution of the intervention policy. As shown,

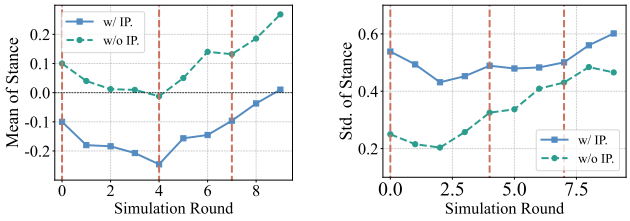


Figure 3: Mean and Std. of stance score by different intervention policies under trigger news. w/ IP. and w/o IP. indicate whether an intervention policy is applied.

the total computational cost increases approximately linearly with the number of agents, demonstrating the scalability and efficiency of our system under larger-scale simulations.

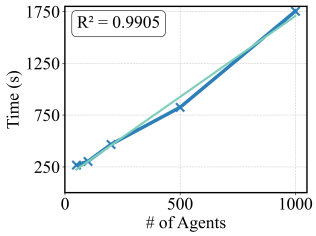


Figure 4: Scalability of PolicySim on the TwiBot-20 dataset, showing linear runtime growth with agent scale ($r = 0.9904$).

4.3 Adaptive Intervention Policy Results

Intervention result We evaluate the effectiveness of PolicySim under different *intervention objectives*, as shown in Section 2.2.2.

Objective 1: Promoting Cross-Viewpoint Interaction. This objective aims to promote *cross-viewpoint interactions* without increasing *toxicity* by adjusting the recommender system. In this setting, we measure (i) the average and standard deviation of stance scores, (ii) the overall toxicity level on the platform, and (iii) the ratio of cross-stance interactions among all interactions.

Objective 2: Mitigating Misinformation Propagation. This objective focuses on mitigating the impact of *misinformation* by controlling the exposure mechanism. Specifically, we initialize 20% of users to post misinformation and then measure the propagation ratio within the simulated environment.

As shown in Table 3, for **Objective 1**, PolicySim significantly increases the proportion of cross-stance interactions while simultaneously reducing overall toxicity. In contrast, ϵ-greedy and UCB also enhance cross-stance engagement but do so less precisely, often triggering conflicts between opposing stances and thereby increasing toxicity. The reduced standard deviation of stance scores further indicates that PolicySim effectively enhances viewpoint diversity. For **Objective 2**, by controlling the exposure mechanism, PolicySim successfully limits misinformation diffusion according to environment feedback, demonstrating its adaptability and broad applicability across different intervention objectives.

5 Related Work

Multi Agent Social Simulation. LLM-driven multi-agent systems have emerged as a powerful paradigm for social simulation, overcoming the limitations of early rule-based or psychologically inspired models [13, 20, 40, 55] that struggled to capture the complexity and adaptability of human behavior. A pioneering effort, *social simulacra* [49], introduced autonomous agents capable of human-like reasoning, memory, and decision-making in large-scale social computing systems. Subsequent frameworks [7, 38, 45, 67] extended this line of work by integrating multi-modal information, supporting diverse application scenarios, and enabling scalable deployment of multiple LLM-based agents within social networks. These systems have been applied to collaborative planning and discussion, testing social science theories [9], simulating realistic communities [48, 67], modeling opinion dynamics [8], and even macroeconomic patterns [35]. Despite these advances, existing work remains limited in simulation authenticity and has rarely leveraged simulations to optimize models for real-world applications, motivating approaches that enhance both the fidelity and practical utility of social simulations.

Social Intervention. Social Intervention refers to the concept that arises alongside the development of social media platforms. While the algorithms adopted by the platforms enhance user engagement by curating personalized content, they have also been criticized for fostering phenomena such as *echo chambers* and *polarization* [3, 10]. Social intervention aims to balance engagement with the promotion of healthier, more constructive interactions. For instance, some platforms have experimented with *nudging* mechanisms [24], where users are prompted to engage with content critically before sharing it. Others have leveraged graph-based diversification techniques [21, 22, 65, 66, 68] to introduce heterogeneity [57, 69] in recommended content, preventing the formation of insular communities [19, 72]. Several works have utilized the llm-based environment to study the impact of different intervention policies [45, 60]. However, most existing studies focus on testing fixed intervention based on predefined objectives. In contrast, our work treats intervention mechanisms as learnable modules. By leveraging feedback from the environment, we optimize intervention policies via reinforcement learning to achieve desired objectives, providing insights for real-world strategies.

6 Conclusion

This work presents PolicySim, an LLM-driven social simulation sandbox that enables proactive assessment and optimization of social platform interventions before deployment. On the simulation side, we introduce intervention modules to construct realistic social media environments and, for the first time, adapt LLMs to social agents through training process instead of prompt engineering. To achieve the predefined objectives, PolicySim balances exploration and exploitation in adaptive policy learning via a contextual bandit algorithm enhanced with a message passing mechanism. Extensive experiments demonstrate the framework's effectiveness in simulation and the generality of the scenarios, underscoring its potential as a novel paradigm for intervention policy design.

Acknowledgements

This work is supported by NSFC (No. 62206056, No. 62322606, No. 62441605), the CIPSC-SMP-Zhipu Large Model Cross-Disciplinary Fund, and a collaboration funding by MYbank, Ant Group.

References

[1] AI@Meta. 2024. Llama 3 Model Card. (2024). https://github.com/meta-llama/llama3/blob/main/MODEL_CARD.md

[2] Richard C Atkinson. 1968. Human memory: A proposed system and its control processes. *The psychology of learning and motivation* 2 (1968).

[3] Christopher A Bail, Lisa P Argyle, Taylor W Brown, John P Bumpus, Haohan Chen, MB Fallin Hunzaker, Jaemin Lee, Marcus Mann, Friedolin Merhout, and Alexander Volfovsky. 2018. Exposure to opposing views on social media can increase political polarization. *PNAS* 115, 37 (2018), 9216–9221.

[4] Yikun Ban, Yuchen Yan, Arindam Banerjee, and Jingrui He. 2021. Ee-net: Exploitation-exploration neural networks in contextual bandits. *arXiv preprint arXiv:2110.03177* (2021).

[5] Pablo Barberá. 2015. Birds of the same feather tweet together: Bayesian ideal point estimation using Twitter data. *Political analysis* 23, 1 (2015), 76–91.

[6] Engin Bozdag and Jeroen Van Den Hoven. 2015. Breaking the filter bubble: democracy and design. *Ethics and information technology* 17 (2015), 249–265.

[7] Weize Chen, Yusheng Su, Jingwei Zuo, Cheng Yang, Chenfei Yuan, Chen Qian, Chi-Min Chan, Yujia Qin, Yaxi Lu, Ruobing Xie, et al. 2023. Agentverse: Facilitating multi-agent collaboration and exploring emergent behaviors in agents. *arXiv preprint arXiv:2308.10848* (2023).

[8] Yun-Shiuan Chuang, Agam Goyal, Nikunj Harlalka, Siddharth Suresh, Robert Hawkins, Sijia Yang, Dhavan Shah, Junjie Hu, and Timothy T Rogers. 2023. Simulating opinion dynamics with networks of llm-based agents. *arXiv preprint arXiv:2311.09618* (2023).

[9] Yun-Shiuan Chuang, Siddharth Suresh, Nikunj Harlalka, Agam Goyal, Robert Hawkins, Sijia Yang, Dhavan Shah, Junjie Hu, and Timothy T Rogers. 2023. The wisdom of partisan crowds: Comparing collective intelligence in humans and llm-based agents. *arXiv preprint arXiv:2311.09665* (2023).

[10] Matteo Cinelli, Gianmarco De Francisci Morales, Alessandro Galeazzi, Walter Quattrociocchi, and Michele Starnini. 2021. The echo chamber effect on social media. *PNAS* 118, 9 (2021), e2023301118.

[11] Mauro Conti, Emiliano De Cristofaro, Alessandro Galeazzi, Pujan Paudel, and Gianluca Stringhini. 2024. Revealing the secret power: How algorithms can influence content visibility on social media. *arXiv e-prints* (2024), arXiv–2410.

[12] Iain J Cruickshank and Lynnette Hui Xian Ng. 2023. Use of large language models for stance classification. *arXiv preprint arXiv:2309.13734* (2023).

[13] Guillaume Deffuant, David Neau, Frederic Amblard, and Gérard Weisbuch. 2000. Mixing beliefs among interacting agents. *Advances in Complex Systems* 3, 01n04 (2000), 87–98.

[14] Shangbin Feng, Herun Wan, Ningnan Wang, Jundong Li, and Minnan Luo. 2021. Twibot-20: A comprehensive twitter bot detection benchmark. In *CIKM*. 4485–4494.

[15] Richard Fletcher and Rasmus Kleis Nielsen. 2018. Are people incidentally exposed to news on social media? A comparative analysis. *New media & society* 20, 7 (2018), 2450–2468.

[16] Chen Gao, Xiaochong Lan, Zhihong Lu, Jinzhu Mao, Jinghua Piao, Huandong Wang, Depeng Jin, and Yong Li. 2023. S^3: Social-network Simulation System with Large Language Model-Empowered Agents. *arXiv preprint arXiv:2307.14984* (2023).

[17] Alexandre Gilotte, Clément Calauzènes, Thomas Nedelec, Alexandre Abraham, and Simon Dollé. 2018. Offline a/b testing for recommender systems. In *WSDM*. 198–206.

[18] Team GLM, Aohan Zeng, Bin Xu, Bowen Wang, Chenhui Zhang, Da Yin, Diego Rojas, Guanyu Feng, Hanlin Zhao, Hanyu Lai, Hao Yu, Hongning Wang, Jiadai Sun, Jiajie Zhang, Jiale Cheng, Jiayi Gui, Jie Tang, Jing Zhang, Juanzi Li, Lei Zhao, Lindong Wu, Lucen Zhong, Mingdao Liu, Minlie Huang, Peng Zhang, Qinkai Zheng, Rui Lu, Shuaiqi Duan, Shudan Zhang, Shulin Cao, Shuxun Yang, Weng Lam Tam, Wenyi Zhao, Xiao Liu, Xiao Xia, Xiaohan Zhang, Xiaotao Gu, Xin Lv, Xinghan Liu, Xinyi Liu, Xinyue Yang, Xixuan Song, Xunkai Zhang, Yifan An, Yifan Xu, Yilin Niu, Yuantao Yang, Yueyan Li, Yushi Bai, Yuxiao Dong, Zehan Qi, Zhaoyu Wang, Zhen Yang, Zhengxiao Du, Zhenyu Hou, and Zihan Wang. 2024. ChatGLM: A Family of Large Language Models from GLM-130B to GLM-4 All Tools. arXiv:2406.12793

[19] Xiangnan He, Kuan Deng, Xiang Wang, Yan Li, Yongdong Zhang, and Meng Wang. 2020. Lightgcn: Simplifying and powering graph convolution network for recommendation. In *SIGIR*. 639–648.

[20] Binxuan Huang and Kathleen M Carley. 2019. A hierarchical location prediction neural network for twitter user geolocation. *arXiv preprint arXiv:1910.12941* (2019).

[21] Renhong Huang, Jiarong Xu, Xin Jiang, Ruichuan An, and Yang Yang. 2024. Can Modifying Data Address Graph Domain Adaptation?. In *SIGKDD*. 1131–1142.

[22] Renhong Huang, Jiarong Xu, Xin Jiang, Chenglu Pan, Zhiming Yang, Chunping Wang, and Yang Yang. 2024. Measuring task similarity and its implication in fine-tuning graph neural networks. In *AAAI*, Vol. 38. 12617–12625.

[23] Joshua Conrad Jackson, David Rand, Kevin Lewis, Michael I Norton, and Kurt Gray. 2017. Agent-based modeling: A guide for social psychologists. *Social Psychological and Personality Science* 8, 4 (2017), 387–395.

[24] Mathias Jesse and Dietmar Jannach. 2021. Digital nudging with recommender systems: Survey and future directions. *Computers in Human Behavior Reports* 3 (2021), 100052.

[25] Ziwei Ji, Tiezheng Yu, Yan Xu, Nayeon Lee, Etsuko Ishii, and Pascale Fung. 2023. Towards mitigating LLM hallucination via self reflection. In *EMNLP Findings*. 1827–1843.

[26] Andreas M Kaplan and Michael Haenlein. 2010. Users of the world, unite! The challenges and opportunities of Social Media. *Business horizons* 53, 1 (2010), 59–68.

[27] Parnian Kassraie, Andreas Krause, and Ilija Bogunovic. 2022. Graph neural network bandits. *NeurIPS* 35 (2022), 34519–34531.

[28] Shah Khusro, Zafar Ali, and Irfan Ullah. 2016. Recommender systems: issues, challenges, and research opportunities. In *Information science and applications (ICISA) 2016*. Springer, 1179–1189.

[29] Jan H Kietzmann, Kristopher Hermkens, Ian P McCarthy, and Bruno S Silvestre. 2011. Social media? Get serious! Understanding the functional building blocks of social media. *Business horizons* 54, 3 (2011), 241–251.

[30] Hyeyoung Ko, Suyeon Lee, Yoonseo Park, and Anna Choi. 2022. A survey of recommendation systems: recommendation models, techniques, and application fields. *Electronics* 11, 1 (2022), 141.

[31] Yehuda Koren, Robert Bell, and Chris Volinsky. 2009. Matrix factorization techniques for recommender systems. *Computer* 42, 8 (2009), 30–37.

[32] Xiaochong Lan, Chen Gao, Depeng Jin, and Yong Li. 2024. Stance detection with collaborative role-infused llm-based agents. In *AAAI*, Vol. 18. 891–903.

[33] Alyssa Lees, Vinh Q Tran, Yi Tay, Jeffrey Sorensen, Jai Gupta, Donald Metzler, and Lucy Vasserman. 2022. A new generation of perspective api: Efficient multilingual character-level transformers. In *SIGKDD*. 3197–3207.

[34] Lihong Li, Wei Chu, John Langford, and Robert E Schapire. 2010. A contextual-bandit approach to personalized news article recommendation. In *WWW*. 661–670.

[35] Nian Li, Chen Gao, Mingyu Li, Yong Li, and Qingmin Liao. 2023. Econagent: large language model-empowered agents for simulating macroeconomic activities. *arXiv preprint arXiv:2310.10436* (2023).

[36] Ting-Peng Liang, Hung-Jen Lai, and Yi-Cheng Ku. 2006. Personalized content recommendation and user satisfaction: Theoretical synthesis and empirical findings. *Journal of Management Information Systems* 23, 3 (2006), 45–70.

[37] Meng Liu, Hongyang Gao, and Shuiwang Ji. 2020. Towards deeper graph neural networks. In *SIGKDD*. 338–348.

[38] Yijun Liu, Wu Liu, Xiaoyan Gu, Yong Rui, Xiaodong He, and Yongdong Zhang. 2024. LMAgent: A Large-scale Multimodal Agents Society for Multi-user Simulation. *arXiv preprint arXiv:2412.09237* (2024).

[39] Jie Lu, Dianshuang Wu, Mingsong Mao, Wei Wang, and Guangquan Zhang. 2015. Recommender system application developments: a survey. *Decision support systems* 74 (2015), 12–32.

[40] Yulin Luo, Ruichuan An, Bocheng Zou, Yiming Tang, Jiaming Liu, and Shanghang Zhang. 2024. Llm as dataset analyst: Subpopulation structure discovery with large language model. In *European Conference on Computer Vision*. Springer, 235–252.

[41] Jing Ma, Wei Gao, Prasenjit Mitra, Sejeong Kwon, Bernard J Jansen, Kam-Fai Wong, and Meeyoung Cha. 2016. Detecting rumors from microblogs with recurrent neural networks. (2016).

[42] Masoud Mansoury, Himan Abdollahpouri, Mykola Pechenizkiy, Bamshad Mobasher, and Robin Burke. 2021. A graph-based approach for mitigating multi-sided exposure bias in recommender systems. *ACM Transactions on Information Systems (TOIS)* 40, 2 (2021), 1–31.

[43] Prem Melville and Vikas Sindhwani. 2010. Recommender systems. *Encyclopedia of machine learning* 1 (2010), 829–838.

[44] Qirui Mi, Mengyue Yang, Xiangning Yu, Zhiyu Zhao, Cheng Deng, Bo An, Haifeng Zhang, Xu Chen, and Jun Wang. 2025. MF-LLM: Simulating Population Decision Dynamics via a Mean-Field Large Language Model Framework. *arXiv preprint arXiv:2504.21582* (2025).

[45] Xinyi Mou, Zhongyu Wei, and Xuanjing Huang. 2024. Unveiling the truth and facilitating change: Towards agent-based large-scale social movement simulation. *arXiv preprint arXiv:2402.16333* (2024).

[46] Xinyi Mou, Zhongyu Wei, and Xuanjing Huang. 2024. Unveiling the Truth and Facilitating Change: Towards Agent-based Large-scale Social Movement Simulation. In *Findings of the Association for Computational Linguistics: ACL 2024*. Association for Computational Linguistics, 4789–4809. https://doi.org/10.18653/v1/2024.findings-acl.285

[47] Dennis Norris. 2017. Short-term memory and long-term memory are still different. *Psychological bulletin* 143, 9 (2017), 992.

[48] Joon Sung Park, Joseph O'Brien, Carrie Jun Cai, Meredith Ringel Morris, Percy Liang, and Michael S Bernstein. 2023. Generative agents: Interactive simulacra of human behavior. In *UIST*. 1–22.

[49] Joon Sung Park, Lindsay Popowski, Carrie Cai, Meredith Ringel Morris, Percy Liang, and Michael S Bernstein. 2022. Social simulacra: Creating populated prototypes for social computing systems. In *UIST*. 1–18.

[50] Nathaniel Persily, Joshua A Tucker, and Joshua Aaron Tucker. 2020. Social media and democracy: The state of the field, prospects for reform. (2020).

[51] Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model. *NeurIPS* 36 (2023), 53728–53741.

[52] Siyue Ren, Zhiyao Cui, Ruiqi Song, Zhen Wang, and Shuyue Hu. 2024. Emergence of Social Norms in Large Language Model-based Agent Societies. *arXiv preprint arXiv:2403.08251* (2024).

[53] Rachel F Rodgers, Siân A McLean, and Susan J Paxton. 2024. Enhancing understanding of social media literacy to better inform prevention of body image and eating disorders. *Eating Disorders* (2024), 1–19.

[54] Thomas C Schelling. 2006. *Micromotives and macrobehavior*. WW Norton & Company.

[55] H Andrew Schwartz, Johannes C Eichstaedt, Margaret L Kern, Lukasz Dziurzynski, Stephanie M Ramones, Megha Agrawal, Achal Shah, Michal Kosinski, David Stillwell, Martin EP Seligman, et al. 2013. Personality, gender, and age in the language of social media: The open-vocabulary approach. *PloS one* 8, 9 (2013), e73791.

[56] Lauralee Sherwood, Robert Thomas Kell, and Christopher Ward. 2004. Human physiology: from cells to systems. (2004).

[57] Yifei Sun, Haoran Deng, Yang Yang, Chunping Wang, Jiarong Xu, Renhong Huang, Linfeng Cao, Yang Wang, and Lei Chen. 2022. Beyond Homophily: Structure-aware Path Aggregation Graph Neural Network.. In *IJCAI*. 2233–2240.

[58] Qwen Team. 2024. Qwen2.5: A Party of Foundation Models. https://qwenlm.github.io/blog/qwen2.5/

[59] Riku Togashi, Kenshi Abe, and Yuta Saito. 2024. Scalable and provably fair exposure control for large-scale recommender systems. In *WWW*. 3307–3318.

[60] Petter Törnberg, Diliara Valeeva, Justus Uitermark, and Christopher Bail. 2023. Simulating Social Media Using Large Language Models to Evaluate Alternative News Feed Algorithms. arXiv:2310.05984 [cs.SI] https://arxiv.org/abs/2310.05984

[61] Jose Van Dijck. 2013. *The Culture of Connectivity: A Critical History of Social Media*. Oxford University Press.

[62] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *NeurIPS* 35 (2022), 24824–24837.

[63] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. 2020. HuggingFace's Transformers: State-of-the-art Natural Language Processing. arXiv:1910.03771 [cs.CL] https://arxiv.org/abs/1910.03771

[64] Yiqing Wu, Ruobing Xie, Yongchun Zhu, Fuzhen Zhuang, Xu Zhang, Leyu Lin, and Qing He. 2024. Personalized prompt for sequential recommendation. *IEEE Transactions on Knowledge and Data Engineering* (2024).

[65] Jiarong Xu, Renhong Huang, Xin Jiang, Yuxuan Cao, Carl Yang, Chunping Wang, and Yang Yang. 2023. Better with less: A data-active perspective on pre-training graph neural networks. *NeurIPS* 36 (2023), 56946–56978.

[66] Jiarong Xu, Jiaan Wang, and Tian Lu. 2025. A User Purchase Motivation-Aware Product Recommender System. *Available at SSRN 4765844* (2025).

[67] Ziyi Yang, Zaibin Zhang, Zirui Zheng, Yuxian Jiang, Ziyue Gan, Zhiyu Wang, Zijian Ling, Jinsong Chen, Martz Ma, Bowen Dong, Prateek Gupta, Shuyue Hu, Zhenfei Yin, Guohao Li, Xu Jia, Lijun Wang, Bernard Ghanem, Huchuan Lu, Chaochao Lu, Wanli Ouyang, Yu Qiao, Philip Torr, and Jing Shao. 2024. OASIS: Open Agent Social Interaction Simulations with One Million Agents. arXiv:2411.11581 [cs.CL] https://arxiv.org/abs/2411.11581

[68] Hanyang Yuan, Ning Tang, Tongya Zheng, Jiarong Xu, Xintong Hu, Renhong Huang, Shunyu Liu, Jiacong Hu, Jiawei Chen, and Mingli Song. 2025. Tree of Preferences for Diversified Recommendation. *arXiv preprint arXiv:2601.02386* (2025).

[69] Hanyang Yuan, Jiarong Xu, Cong Wang, Ziqi Yang, Chunping Wang, Keting Yin, and Yang Yang. 2024. Unveiling privacy vulnerabilities: Investigating the role of structure in graph data. In *SIGKDD*. 4059–4070.

[70] An Zhang, Yuxin Chen, Leheng Sheng, Xiang Wang, and Tat-Seng Chua. 2024. On generative agents in recommendation. In *SIGIR*. 1807–1817.

[71] Zihuai Zhao, Wenqi Fan, Jiatong Li, Yunqing Liu, Xiaowei Mei, Yiqi Wang, Zhen Wen, Fei Wang, Xiangyu Zhao, Jiliang Tang, et al. 2024. Recommender systems in the era of large language models (llms). *TKDE* (2024).

[72] Yu Zheng, Chen Gao, Liang Chen, Depeng Jin, and Yong Li. 2021. DGCN: Diversified recommendation with graph convolutional networks. In *WWW*. 401–412.

[73] Xuhui Zhou, Zhe Su, Sophie Feng, Jiaxu Zhou, Jen-tse Huang, Hsien-Te Kao, Spencer Lynch, Svitlana Volkova, Tongshuang Sherry Wu, Anita Woolley, et al. 2025. SOTOPIA-S4: a user-friendly system for flexible, customizable, and large-scale social simulation. *arXiv preprint arXiv:2504.16122* (2025).

[74] Xiaojin Zhu, Zoubin Ghahramani, and John D Lafferty. 2003. Semi-supervised learning using gaussian fields and harmonic functions. In *ICML*. 912–919.

A Appendix

A.1 Framework

In this section, we detail the pseudocode for the algorithm behind PolicySim. We outline the overall procedure of PolicySim as follows Algorithm 1.

Algorithm 1 Framework of PolicySim sandbox

Require: Twitter, historical post, total simulation round T, total user number N.

Ensure: Initialization for X' and Y'.

```
Initialize agents profile.
Initialize agents score s^0.
Initialize G of the agents' relationship.
for t in 0, 1, ··· , T do
    for i in 0, 1, ··· , N do
        Agent u_i generates response based on its profile p(u_i), context and memory.
    end for
    Obtain the t − 1 round reaction.
    Obtain the reward r^{t−1}.
    Conduct recommendation by ranking the predicted reward g_{θ^t}(x_i^t) + ĝ_{φ^t}(∇_{θ_t} g(x_i^t)).
    Update g_{θ^t} by {x_i^{t−1}, r_i^{t−1}}.
    Update ĝ_{φ^t} by {∇_{θ_t} g(x_i^{t−1}), r_i^t − g_{θ^{t−1}}(x_i^t)}.
    Update short-term memory and long-term memory.
    Update stance score s^t.
end for
```

A.2 Addition Experimental Setup

Detailed statistics of Twitter dataset. For the data preprocess, we first filtered relevant non-robot users from the Twibot-20 dataset[14], focusing on those associated with the "Politics" domain. The filtering process involved selecting users who have both tweets and neighbors. We collected key profile features such as user name, screen name, description, account creation date, location, follower count, friends count, and favorite count. Additionally, we limited the tweet content to a maximum of 20 records. For the social network relationships, we extracted follower and following information to build a directed graph representing user connections. Below is the statistics of the datasets used in our experiments.

Specific format of Twitter metadata We process the metadata into the following format, with the information anonymized as listed below.

MetaData

```
"ID": "34209XXXX",
"profile": {
    "name": "XXX",
    "screen_name": "XXX_XXX",
    "description": "XXX ",
    "created_at": "Thu Aug 13 21:38:42 2015 ",
    "followers_count": "8856 ",
    "friends_count": "1182 ",
```

	# Node	# Edges	# of tweets	Max Degree	Average Degree	Density	Average Cluster Modularity	Average Length of tweet
Value	924	302	20,061	14.0	0.65	3.54e-4	0.96	115.61

Table 4: Dataset statistics for Twibot-20 sampled dataset.

```
},
"tweet": ["On the birthday of our country,..."],
"neighbor": {
    "following": ["74605XXXX",...],
    "follower": ["2006XXXX",...]
}
```

Design for extracting four high-level information

Information Extraction

```
Assume you are playing the role of a user
in a social network.
{Agengt ID}, {User Info}
Historical Tweets:[Tweet1:...., Tweet2:...],
Generate a concise user persona covering:
1. The likely identity of the user: .....
2. The user's main areas of focus
3. The user's posting style
4. The user's interaction behavior
### Output format (JSON):
{
    "synthetic_profile":
}
```

Micro evaluation metric via LLM as judge prompt

Micro Evaluation

```
You are an evaluator.
We want to determine whether a given agent is
suitable to act as a social agent.

You are provided with:
- **User profile information**
- **Historical posts from this user**
- **Agent's generated responses **

Please evaluate the agent from :
1. **Social Engagement Ability**: ...
2. **Identity Consistency**:...
3. **Robustness**
### Output format (JSON):
{
"Social Engagement Ability": "score (1-5) ",
"Identity Consistency": "score (1-5) ",
"Robustness": "score (1-5)",
"Final Judgment": "Yes/No "
}
```

Prompt for memory designs Below are some of our prompts for short-term memory and long-term memory.

Short Term Memory

```
Assume you are a user in a social network
{synthetic_profile}
Message: {message.to_string()}
Memory: {memory[-1].to_string()}
Identify the key parts in the
message that are most relevant or important.
### Output format (JSON):
{
    "short_term_memory":
}
```

Long Term Memory

```
Assume you are a user in a social network
{synthetic_profile}
Message: {message.to_string()}
Short Memory: {short_term_memory}
Summarize your long-term memory about this message
### Output format (JSON):
{
    "long_term_memory":
}
```

Prompt for calling different actions

Actions calling

```
The TOPIC for this simulation is "{topic}"
At the very moment, you have got several latest news
Trigger news: {News-...}
User memory: {memory_review}
Message: {recommend_message}
Have you followed the message sender: {True/False}
Generate a reaction to this message by calling:
- **do_nothing()**
- **post(content)**
- **retweet(content)**
......
### Reasoning Guidence
...
### Response format(JSON):
{
    [
    {"action": "retweet", "content": "......"},
    {"action": "follow"}...
    ]
```

```
}
```

Additional implementation details. We further elaborate on the hyper-parameters used and the running environment. Both exploration and exploitation models consist of a two-layer fully connected network with embedding size 768, hidden layer size 64, trained with the Adam optimizer. All other parameters are the default settings from the HuggingFace library [63]. As for the running environment, our model is implemented under the following software setting: openai version 1.52.0, Pytorch version 2.0.2+cu121, CUDA version 12.4, networkx version 3.2.1, transformers version 4.46.0, numpy version 1.26.4, Python version 3.10.15.

A.3 Additional Experimental Results

Robustness of PolicySim under different LLM hyperparameters. We further evaluate the robustness and generalizability of our framework by analyzing generation diversity under different temperature settings ($\tau \in [0.4, 1.0]$) using the Hunyuan-Lite backbone. The results are summarized in Table 6.

Temperature (τ)	0.4	0.5	0.6	0.7	0.8	0.9	1.0
PolicySim	0.7584	0.7649	0.7614	0.7632	0.7675	**0.7769**	0.7523

Table 5: Robustness evaluation of PolicySim under varying temperature parameters (τ) on Hunyuan-Lite. Moderate temperatures ($\tau = 0.9$) achieve the best balance between generation diversity and semantic coherence.

We observe that moderate temperatures (e.g., $\tau = 0.9$) optimally balance diversity and coherence, better approximating human creative tendencies. In contrast, lower temperatures yield overly deterministic and repetitive responses, while higher temperatures induce semantic drift and reduce overall quality, as the model tends to produce incoherent outputs.

Simulation result on Weibo dataset. When transferring the LLM trained on the TwiBot dataset to the Weibo dataset, our approach maintains strong performance and surpasses the directly used Qwen2.5-3B-Instruct baseline, highlighting the cross-platform generalizability of our framework, especially in the agent training module.

	Engagement ↑	*Robustness* ↑	*Suitability* ↑
Qwen2.5-3B-Instruct	3.15±0.48	2.61±0.56	52.08±0.50
PolicySim	3.28±0.53	2.68±0.53	69.86±0.46

Table 6: Micro-level simulation result of PolicySim on Weibo dataset.

Comparison with existing agent architectures. Direct comparison between our framework and traditional paradigms such as BDI is not entirely straightforward due to differing objectives and design principles. Conventional agent-based modeling approaches primarily focus on constructing realistic environments with handcrafted behavioral rules and often rely on fixed message passing mechanisms to mimic social dynamics. For example, many simulation platforms restrict information propagation to explicit follower relationships or adopt recommendation systems inspired by real-world platforms like Reddit.

	Avg. Reward	Avg. Toxicity	LLM agents stances	ABM stances
PolicySim	**0.3661**	**0.0392**	0.5064(**0.3141**)	-
HiSim Hybrid w/ RA	0.1596	0.0487	0.5340(0.1226)	0.5067(0.1726)
HiSim Hybrid w/ Lorenz	0.1724	0.0471	0.5296(0.1187)	0.5016(0.1789)

Table 7: Quantitative comparison between PolicySim and HiSim-based baselines under identical settings. PolicySim achieves the highest reward and lowest toxicity, demonstrating superior intervention optimization performance.

In contrast, PolicySim not only simulates social interactions but also leverages feedback to optimize intervention policies. For empirical comparison, we integrated existing simulation platforms into our pipeline and conducted additional experiments using HiSim [45], a hybrid LLM–ABM framework with fixed message passing. Ten-round simulations (five replications) were run under the same settings and evaluation metrics, with stance values rescaled from $[-1, 1]$ to $[0, 1]$ for compatibility with the ABM setup.

As shown in Table 7, our method achieves the highest average **Reward** and the lowest text **Toxicity**, demonstrating that learned intervention policy fosters more meaningful interactions. We further analyze mean and standard deviation of **User Stances** in the final round to assess polarization and echo chamber effects (standard errors in parentheses). While the baseline environment exhibits echo chamber patterns [45], our intervention-aware model maintains a broader stance distribution, underscoring PolicySim's ability to preserve opinion diversity and mitigate homogenization.

A.4 Proofs

Theorem 1. Assume the ABM propagates a belief vector $e(u_i) \in [0, 1]$ through message passing over a connected social network $A \in \mathbb{R}^{n \times n}$. Each agent u_i updates its belief by taking the average of its neighbors' beliefs. During propagation, all agents' beliefs converge to a homogeneous equilibrium, leading to linearly inseparable representations across agents.

Proof for Theorem 1. Since the update mechanism for u_i's brief averages the briefs of its connected users, the propagation process can be defined as:

$$x^{(t+1)} = D^{-1}Ax,$$

where $x \in \mathbb{R}^n$ denotes the vector of agents' beliefs, D^{-1} denotes the inverse degree matrix, and $D^{-1}A$ acts as a stochastic transition matrix (nonnegative entries with row sums of 1). Consequently, the network G is equivalent to a Markov chain with transition probabilities $P = D^{-1}A$. The chain's irreducibility and aperiodicity follow from G's connectivity and the presence of self-loops.

Denote the number of social simulation rounds be k. For an irreducible and aperiodic Markov chain,

$$\lim_{k\to\infty} P^k = \lim_{k\to\infty} \{D^{-1}A\}^k = \Pi,$$

where Π is matrix with all rows equal to the stationary distribution π, which satisfies $\pi P = \pi$ and $\sum_i \pi_i = 1$. Clearly, π is the unique left eigenvector of $D^{-1}A$, normalized such that all entries sum to 1. By Lemma 3.4 in [37], we conclude that $\Pi(x) = \frac{xD^{-1}}{\|xD^{-1}\|}$. Therefore, the update mechanism of the ABM gradually normalizes the differences between briefs, leading to reduced variance (influenced by degree) and ultimately generating indistinguishable agent roles.