Simulating and Evaluating Generative Modeling and Collaborative Filtering in Complex Social Networks

Wen Dong
Air Force Research Laboratory
Dayton, OH, USA
wendong@gmail.com

Fairul Mohd-Zaid
Air Force Research Laboratory
Dayton, OH, USA
fairul.mohd-zaid@us.af.mil

ABSTRACT

We introduce a multi-agent simulation framework for modeling large-scale online social dynamics by combining retrieval-augmented large language models, generative embedding methods, and collaborative filtering. Our approach learns diverse agent embeddings to capture varying user behaviors and employs a multi-layer perceptron for user-content ranking. We compare three strategies—(1) a generative modeling approach that integrates agent embeddings and collaborative filtering, (2) an LLM-based method grounded in historical context, and (3) a reflection-based clustering technique—and evaluate them on metrics such as comment volume, tree depth, user engagement patterns, and topic distribution. Results show that generative embeddings coupled with collaborative filtering better approximate complex phenomena like localized influencers, specialized subcommunities, and emergent echo chambers. Moreover, our framework supports policy-driven experimentation by incorporating social regularizers (cohesion, polarization, and bias) to simulate scenarios ranging from tightly knit communities to more balanced, cross-cutting interactions. By integrating large-scale data with adaptable LLM-driven agents, this work provides a versatile, data-centric foundation for simulating and analyzing online social ecosystems at scale.

KEYWORDS

Generative Modeling; Collaborative Filtering; Large Language Models (LLMs); Social Network Analysis; Agent-Based Simulation; Online Community Behavior

ACM Reference Format:
Wen Dong and Fairul Mohd-Zaid. 2025. Simulating and Evaluating Generative Modeling and Collaborative Filtering in Complex Social Networks. In *Proc. of the 24th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2025), Detroit, Michigan, USA, May 19 – 23, 2025*, IFAAMAS, 10 pages.

1 INTRODUCTION

Communities emerge, strengthen, or fragment through a complex interplay of social, cultural, and technological forces—including large language models (LLMs), generative AI, and recommendation systems. As Pariser illustrates in *The Filter Bubble* [29], personalized content can deepen polarization by focusing on individual interests rather than shared narratives: for instance, two users searching "Egypt" might see protests versus tourism, revealing divergent realities. On the positive side, tight-knit communities can foster belonging, mutual support, and collective action, as seen in both historical contexts of print capitalism and modern social movements like #MeToo and #BlackLivesMatter [2, 57]. However, increased interaction among like-minded groups risks creating echo chambers that erode critical thinking and distort information [6, 18, 34, 45].

The business models driving many platforms often exacerbate this dynamic: by maximizing advertising revenue through hyper-targeted content, they can fragment information ecosystems and undermine a shared public sphere [8, 10, 28, 42, 46]. Ethically designed social media systems should mitigate these risks by facilitating inclusive dialogue, bridging communities, and promoting critical engagement with information. However, it is challenging to safely test interventions at scale in real-world platforms. To address this, we propose a multi-agent simulation architecture that explores strategies like adjusting content recommendations to connect polarized communities or facilitating dialogue to promote empathy and mutual understanding.

We leverage generative AI—specifically large language models (LLMs)—to simulate online interactions and analyze policy implications [1]. Since LLMs have limited context windows, we adopt retrieval-augmented generation, using vector stores to supply relevant information at each step. We also model agents' decision-making as state machines equipped with a suite of extensible microservices. Agent behaviors can then be fine-tuned via prompt engineering or gradient-based optimization to capture complex social phenomena.

Our contributions are threefold: (1) A multi-agent simulation framework for modeling social media dynamics using LLM-enhanced agents. (2) Techniques to overcome LLM context constraints via retrieval-augmented generation and state-based modeling. (3) A challenge-based evaluation framework and baselines to compare simulated agent behaviors with real-world patterns. The remainder of the paper is organized as follows: Section2 surveys related research on community dynamics, simulation methods, and LLM-based modeling. Section3 details our agent-simulation design, including architecture and generative components. Section4 describes the implementation, experimental setup, and evaluation results. Section5 concludes the paper.

2 BACKGROUND AND STATE OF THE ART

The impact of personalization algorithms and individual information-seeking on social networks—particularly their role in forming communities, amplifying divisions, distorting information, and creating filter bubbles—has been a key theme in research on social cohesion and polarization. For example, as Eli Pariser illustrates in The Filter

Proc. of the 24th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2025), Y. Vorobeychik, S. Das, A. Nowé (eds.), May 19 – 23, 2025, Detroit, Michigan, USA. © 2025 International Foundation for Autonomous Agents and Multiagent Systems (www.ifaamas.org).

Bubble, algorithms can isolate users within their own belief systems [29]. Similarly, in Republic.com 2.0 [45] Cass Sunstein makes a point that we just really listen and read those who are more or less like us, our type of democracy may decay even further than it has already. These dynamics have been analyzed in works such as Santos et al.'s study [35] on the influence of link recommendation algorithms on polarization and Liu et al.'s examination of how political typologies interact with content-based and colaborative filtering recommendation algorithms to shape users' exposure to diverse content [23]. Creating responsible AI goes beyond addressing immediate ethical issues like privacy, fairness, and safety [26] — it must also tackle emergent societal impacts that may not be immediately apparent.

From a statistical physics perspective [9], polarization in opinion dynamics occurs when agents influence each other more strongly if they share similar views, leading to homophily-driven extremes in social networks. Cultural formation arises through imitation and conformity, where individuals influence one another more if they already share many beliefs and behaviors, forming cultural norms. Language formation involves agents negotiating a shared vocabulary and grammar through repeated, context-rich interactions. Influential models in this domain include the Hegselmann-Krause model [17] and Axelrod's Cultural Model [3], both crucial to understanding social phenomena like polarization and cultural convergence.

AAMAS has a strong tradition of studying opinion dynamics through independent agent-based simulations [36, 43, 48, 50]. Recently, there is a surge in using large language models to enhance agent-based modeling and simulation, expanding the capabilities of simulations and opening up new applications and research directions.

In much of the existing literature, text serves as a crucial interface for agents' interactions with their environment and each other. Game rules, domain-specific knowledge, and contextual instructions are all embedded in textual formats [37]. As a result, tool usage [21], environmental observations, multi-modal capabilities, and inter-agent communication are typically handled through text-based interactions. This reliance on text as the primary medium enables agents to engage with complex environments flexibly [14, 30].

LLM-enhanced agents are tailored to align closely with human knowledge and exhibit personalized behaviors, achieved primarily through prompt engineering and fine-tuning. Prompt engineering sets the context for the agents' actions, offering domain-specific instructions, background knowledge, patterns for text generation, and illustrative task examples [30]. Fine-tuning, on the other hand, leverages large-scale domain-specific datasets or directly synthesized "human feedback" to refine agent responses and decision-making processes [13, 40].

The design of these LLM-enhanced agents aims to simulate complex behaviors that reflect human cognitive processes. Through text-based reasoning, they are capable of retaining and utilizing past experiences (memory) [22, 52, 56], introspecting and adjusting their behavior based on outcomes (reflection) [15, 39], and carrying out sequences of interconnected tasks that mimic human planning and workflows [30, 44]. This development marks a shift towards creating more dynamic, adaptive simulations that better capture the complexities of real-world interactions.

Evaluating LLM-enhanced simulations centers on accuracy, explainability, and ethical considerations. Accuracy is measured at both the individual and collective levels, often using real-world data as benchmarks [11, 15]. Explainability requires agents to transparently justify their actions and decisions [22, 30]. Ethical concerns are particularly pronounced for LLM-based agents, as their advanced capabilities pose more significant challenges than traditional agent-based models, including risks around bias, misuse, and unintended societal impacts [51].

For example, in the social media domain, an LLM-enhanced agent has been tasked to generate realistic social networks to model demographic homophily and address biases, using curated personas and three prompting methods: global network generation, local relationship assignment for each persona, and a sequential approach incorporating evolving network information [11]. LLMs are also used to simulate social media dynamics by generating content based on user profiles and evolving contexts. In one study, GPT-3.5-turbo replicates how users react to and modify messages before sharing, modeling information spread. Another uses ANES-based personas to simulate posting, liking, and commenting under varying exposure scenarios, such as engaging with opposing viewpoints or popular content [20, 47].

In human-computer interaction, natural language often interfaces with the environment via API primitives. One study employs LLMs to recall experiences, plan, and reflect on behaviors, creating agents that exhibit long-term coherence in a "The Sims"-like environment while interacting with other agents [30]. A key challenge for future research is managing the vast and evolving set of experiences to maintain authenticity and social dynamics. Other works task LLMs with decomposing goals into sub-goals and generating sequences of primitive API calls [56] or autonomously identifying new goals and writing code to achieve them [52].

We argue that while large language models enhance agents by enabling perception through language-based communication with other agents and information retrieval from the environment, and by heightening reasoning, autonomy, and human-like behavior through natural language and model heterogeneity, there remain significant challenges and future work needed to build systems from data effectively. These include developing generalizable LLM capabilities to support open platforms for training [49], fine-tuning, and deploying LLMs across diverse modeling and simulation tasks; creating techniques for robust and stable simulations that perform well in unforeseen scenarios and adversarial conditions [53]; and improving the efficiency of scaling up while managing computational costs. Our methodological work, therefore, focuses on exploring solutions to these challenges.

3 METHOD

AI can enhance critical thinking by combining semantic and structural insights: through advanced language models and retrieval-augmented generation, users can verify claims, explore diverse perspectives, and detect bias or misinformation. Additionally, network analysis methods—community detection, bridging centrality, and cross-community interaction metrics—uncover echo chambers and measure polarization, illuminating how information circulates across social ecosystems. Together, these tools help users challenge

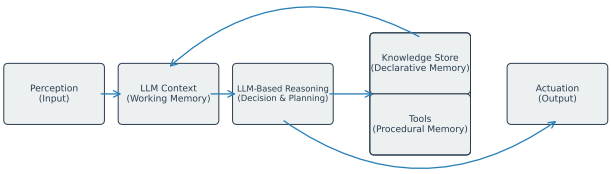


Figure 1: A cognitive architecture for LLM-enhanced agents, illustrating how perception, memory (declarative and procedural), reasoning, and actuation coordinate for adaptive, context-aware behavior.

their own assumptions while providing companies and governments with data-driven metrics to regulate platforms, optimize recommendations, and implement transparent, evidence-based policies fostering healthier online discourse. Building on these capabilities, our method employs three enabling technologies for modeling agent behavior in LLM-enhanced simulations: a cognitive architecture (§3.1) integrating working memory, reasoning, and extensible tools, generative modeling of agent behaviors through learned embedding distributions (§3.2), and specialized neural network tools for tasks such as collaborative filtering and complex analysis (§3.3).

3.1 Cognitive Architecture LLM-Empowered Agent

Recent advances in large language models have led to richer agent frameworks that combine multiple memory stores, reflection mechanisms, and specialized tools for sophisticated planning and user support [22, 30, 52]. Building on these ideas, our cognitive architecture explicitly weaves together LLM Context, Knowledge Store, Tools, Perception, and Actuation, under a central LLM-based planner. In doing so, it moves beyond simple chat or rule-based designs toward an integrated, extensible system that helps users detect misinformation, analyze diverse viewpoints, and manage information overload.

- *LLM Context (Working Memory):* This short-term scratchpad holds recent inputs and intermediate reasoning steps. Approaches like chain-of-thought prompting [54] and reflection [38] allow the agent to generate or refine its reasoning in real time. By monitoring the user's current discussion thread and any relevant "tools" outputs (e.g., fact checks), the LLM context enables on-the-fly critical thinking suggestions.
- *Knowledge Store (Long-Term Declarative Memory):* Unlike the context buffer, which is ephemeral, the knowledge store persistently retains broader facts or past experiences—often as embeddings in a vector database [11, 56]. This supports advanced retrieval-augmented generation (RAG) for critical thinking tasks: for instance, verifying a claim by retrieving authoritative sources, or highlighting contradictory evidence from previous sessions. Storing user models here also helps tailor the difficulty or diversity of newly recommended viewpoints.
- *Tools (Long-Term Procedural Memory):* Complex tasks—like sentiment analysis, clustering, or claim verification—are offloaded to specialized modules invoked via function calls. For instance, the agent might use: *Sentiment/Emotion Classifiers* to detect loaded language or manipulative rhetoric, *Embedding-Based Clustering* to reveal how the user's reading history may be overly homogeneous, flagging a potential echo chamber, *Multi-Document Summarizers* to produce short, multi-perspective digests of diverse news articles, *Fact-Check Retrieval Modules* to compare user-posted claims against verified sources or fact-check databases. In each case, results are returned to the LLM context for explanation or user-facing summaries [21, 54].
- *Perception (Input):* Incoming data—social media posts, user questions, sensor signals—flows into working memory. Although text inputs remain most common, we can integrate multimodal signals if accompanied by a tool that translates, say, an image or audio snippet into textual or semantic form [15]. The planner then decides whether to invoke further analysis (e.g., sentiment detection on a newly received tweet).
- *Actuation (Output):* Actuation materializes the agent's decisions: it might respond to the user, update a feed-ranking policy, or highlight contrasting sources to stimulate more balanced thinking. For example, if the agent's sentiment tool detects recurring anger or fear in a user's news feed, Actuation could prompt, "Would you like to see alternative perspectives on this topic?" or automatically inject relevant counter-articles.
- *LLM-Based Planning and Reasoning:* At the core is the planner, driven by a large language model [5, 30]. It interprets new perceptions, draws on the knowledge store, and selectively invokes tools to obtain deeper insights—like fact checks or multi-perspective summaries. This orchestration supports advanced "what-if" analyses. For instance, the agent can simulate how a small group of "bridge" articles might reduce polarization in a community or generate a multi-viewpoint summary to prevent one-sided consumption of news.

Figure 1 shows how each module fits into a feedback loop. The agent first perceives new input (e.g., an article or user request), which populates the LLM Context. The planner queries the Knowledge Store or calls Tools as needed—whether for embedding-based clustering, chain-of-thought summarization, or emotional content labeling. Once the LLM produces a response or recommended action, Actuation carries it out. New environmental changes or user feedback then cycle back as fresh perceptions, fostering iterative refinement [22, 38].

Empowering Critical Thinking and Diverse Exploration. Crucially, this architecture is not just for automating tasks; it also aims to guide the user toward thoughtful engagement. By combining knowledge retrieval, structural analysis (e.g., echo chamber detection), and multi-perspective content recommendations, the agent can nudge users to reflect on biases or broaden their exposure without overwhelming them. This mix of short-term reasoning (LLM context), long-term declarative memory (knowledge store), and procedural skills (tools) provides a scalable path for delivering interactive, real-time support in tasks such as:

- *Flagging Rhetorical Tricks:* Use emotion detection and fallacy classifiers to highlight loaded language and prompt a user to question its sources.
- *Suggesting Contrasting Viewpoints:* Dynamically retrieve alternative takes on a given topic, encouraging a more balanced mental model.

- *Summarizing Multiple Documents:* Help users synthesize conflicting news stories or research papers into a concise multi-perspective summary.

Integrations for Policy and Platform Research. The integrated design paves the way for realistic simulations of user behavior at scale. Because each agent can replicate real user traits (e.g., strongly partisan vs. open-minded), platform owners or policymakers can experiment with new ranking strategies or intervention policies and assess outcomes in a sandbox before enacting them in the real world. This aligns with emerging "generative agent" paradigms, where reflective reasoning and domain-specific tooling work together to reduce echo chambers and misinformation spread [52].

In summary, our cognitive architecture harnesses LLM-based reasoning, persistent knowledge, and specialized tools to support not only automation but also critical thinking and perspective-taking. It offers a cohesive, extensible foundation for building agents that guide users toward more informed, balanced, and open-minded interactions in complex social-media environments.

3.2 Generative Modeling of Agent Behavior through Learning Embedding Distributions

Recent advances in large language model (LLM) simulations have significantly expanded the modeling of human-like behaviors across domains such as social simulations, task automation, and interactive AI [7, 27]. Most state-of-the-art approaches rely on prompt engineering and supervised fine-tuning to align LLMs with human values, personalize responses, and support dynamic planning. However, these methods introduce notable limitations. Supervised fine-tuning requires carefully annotated data [32], which can be costly or mirror the biases of its curators [55]. Relying on fixed prompts and heuristics also reduces agents' adaptability and stability, risking errors in ambiguous or novel scenarios. Moreover, agents remain susceptible to adversarial prompts and out-of-distribution inputs, undermining their reliability and decision-making in real-world contexts [31].

To address these limitations, we propose a generative approach that learns and samples agent embeddings via diffusion models [19, 41], enabling flexible and probabilistic behavior generation. Unlike purely language-driven prompting, which depends on carefully crafted instructions and fine-tuning, our framework uses learned distributions over embeddings to capture both content and agent preferences. First, we generate an embedding that reflects the agent's potential behavior. We then rank stored memories by similarity and feed the top matches back into the LLM, prompting it to produce contextually relevant actions. For instance:

```
"The agent has previously exhibited these behaviors
   in similar situations: [memory1, memory2, ...,
memory-k]. Generate a new behavior that aligns with
 these examples, reflecting the observed patterns."
```

Because embeddings directly encode behavioral traits, interpolations and adjustments become more natural, reducing reliance on prompt-specific heuristics and expensive fine-tuning. This process resembles a recommendation system—linking stored experiences with ongoing contexts—and robustly handles diverse or novel scenarios. Technically, the diffusion model progressively adds noise to embeddings (forward process) and then removes it (reverse process), with a neural network predicting and subtracting this noise. Learning this distribution allows us to generate and refine embeddings in a flexible, data-driven manner, bridging the gap between static LLM prompting and contextually rich behavior modeling.

The diffusion model can be formulated in various ways. One common formulation defines the forward process as a stochastic differential equation (SDE) that introduces noise to an initial noise-free data point x_0 over time t: $dx_t = f(t, x_t)dt + g(t)dW_t$. Here, x_t is the noisy data at time t, $f(t, x_t)$ is the drift term (often set to 0 in diffusion models), $g(t)$ is the diffusion coefficient, and dW_t is the Wiener process (standard Brownian motion). A neural network $\epsilon_\theta(x_t, t)$, parameterized by θ, is trained to predict the noise ϵ_t from the perturbed data x_t by minimizing the weighted loss:: $L(\theta) = \mathbb{E}_{t,x_0,\epsilon}\left[\lambda^2(t)\left\|\epsilon_t - \epsilon_\theta\left(x_t, t\right)\right\|^2\right]$, where ϵ_t is sampled from a standard Gaussian distribution, and $W_t = \lambda(t)\epsilon_t$ represents the noise added to x_0 to generate x_t.

The reverse process is the time-reversed SDE, which reverses the noise added during the forward process. It is given by $dx_t = \left[f\left(t, x_t\right) - g(t)^2\nabla_{x_t}\log p_t\left(x_t\right)\right]dt + g(t)d\bar{W}_t$, where $d\bar{W}t$ is the reverse-time Brownian motion, and the score term $\nabla x_t \log p_t(x_t)$ is estimated by the neural network $\epsilon_\theta(x_t, t)$ trained in the forward process. To generate a sample from the learned distribution through the reverse process, we start with a noisy embedding $x_T \sim \mathcal{N}(0, I)$ and solve the reverse-time SDE using numerical techniques (e.g., Euler-Maruyama) to recover x_0, $x_{t-\Delta t} = x_t - \left[f\left(t, x_t\right) - g(t)^2\epsilon_\theta\left(x_t, t\right)\right]\Delta t + g(t)\sqrt{\Delta t}z$, where $z \sim \mathcal{N}(0, I)$. One drawback of diffusion model is that sampling from the reverse process is slow due to that many discrete-time steps are required. One solution to accelerate the sampling process is to use the corresponding probability flow ODE, $d\mathbf{x} = \left[\mathbf{f}(\mathbf{x}, t) - \frac{1}{2}g(t)^2\nabla_{\mathbf{x}}\log p_t(\mathbf{x})\right]dt$ whose trajectories have the same mariginal probability density $p_t(\mathbf{x})$. The likelihood of the data, $p_0(x)$, can be computed via the probability flow ODE formulation and the change-of-variable formula as $\log p_0(\mathbf{x}(0)) = \log p_1(\mathbf{x}(1)) - \frac{1}{2}\int_0^1 \frac{d\left[g^2(t)\right]}{dt}\,\mathrm{div}\, s_\theta(\mathbf{x}(t), t)dt$ Here, divergence term can be efficiently estimated using the Skilling-Hutchinson estimator $\mathrm{div}\, s_\theta(\mathbf{x}(t), t) = \mathbb{E}_{\boldsymbol{\epsilon}\sim\mathcal{N}(\mathbf{0},\mathbf{I})}\left[\boldsymbol{\epsilon}^\top J_{s_\theta}(\mathbf{x}(t), t)\boldsymbol{\epsilon}\right]$.

To learn the generative model of the embeddings from learning the diffusion dynamics $\epsilon_\theta(x_t, t)$, we use a multi-layer perceptron (MLP) with layer normalization for stability and enhanced training dynamics.

- Time Embedding: The time step t is encoded into a high-dimensional vector $e(t)$ using either sinusoidal functions (like in Transformer models) or a learned embedding. This embedding is concatenated or added to the hidden state at the input and hidden layers to ensure temporal context is integrated throughout the network.
- Input Layer: The concatenated vector of x_t and $e(t)$ forms the input, allowing the network to leverage both data and temporal information.
- Hidden Layers: A series of fully connected layers follow, each with layer normalization and a non-linear activation (e.g., ReLU or GELU). The hidden layers transform x_t and $e(t)$, learning their complex relationships.
- Output Layer: The network produces the noise component $\epsilon_\theta(x_t, t)$, which guides the reverse diffusion process for generating realistic samples.

The network architecture can be represented as $\epsilon_\theta(x_t, t) = \text{MLP}([x_t; e(t)])$, where $e(t)$ is the time embedding, ensuring temporal information is integrated throughout the MLP. By utilizing this architecture, the network gains the ability to dynamically adapt its predictions based on the temporal state t, providing context-aware outputs that are crucial for the diffusion process.

For the forward SDE, we use $d\mathbf{x} = \sigma^t d\mathbf{w}$, where σ^t scales the noise applied at time t through a Wiener process $d\mathbf{w}$. The conditional distribution $p_{0\to t}(\mathbf{x}(t) \mid \mathbf{x}(0))$ is Gaussian, given by $\mathcal{N}\left(\mathbf{x}(t); \mathbf{x}(0), \frac{1}{2\log\sigma}(\sigma^{2t}-1)\mathbf{I}\right)$. The weighting function for the loss in the noise model is $\lambda(t) = \sqrt{\frac{1}{2\log\sigma}(\sigma^{2t}-1)}$. The reverse-time SDE is consequently $d\mathbf{x} = -\sigma^{2t}\nabla_{\mathbf{x}}\log p_t(\mathbf{x})dt + \sigma^t d\overline{\mathbf{w}}$, and its corresponding probability flow ODE is $\frac{d\mathbf{x}}{dt} = -\sigma^{2t}s_\theta(\mathbf{x}, t)$.

By conditioning on certain contexts or modifying the noise level, embeddings can be sampled or interpolated between behaviors, allowing controlled generation of agent actions. This flexibility can be critical in modeling coherent transitions in agent behavior over time. This structure provides a concise way to model the generation of agent behavior in embedding space, guided by the learned distributions.

3.3 Integration of Neural Network Tools in LLM-Enhanced Agents

While LLMs excel at language-based tasks, they encounter limitations when intricate computation, dynamic adaptation, or real-time interaction are required. Tools, particularly neural networks designed for specific tasks, provide complementary functionality beyond the text-based scope of LLMs. For instance, specialized neural network models can simulate complex environments, perform probabilistic computations, or analyze structured data that would be challenging to handle through prompts alone.

Integrating tools via function calls extends LLM capabilities, such as data classification or embedding generation, acting as modular plugins without altering the core architecture. Neural network tools can be continuously trained and updated independently, aligning with MLOps principles. Within LLM cognitive architectures, these tools function as procedural memory, handling specific tasks while the LLM focuses on higher-level reasoning and planning.

This dual system, where the LLM manages high-level reasoning and tools provide specialized, adaptable functionality, balances generalization and stability. The agent can operate across various domains, with the neural network tools being retrained and optimized as needed without disrupting the core language model. This modularity is crucial for developing robust agent behavior, allowing iterative improvement and adaptability in response to evolving simulation scenarios and complex environments.

Deep MLP for Predicting User Vote. To model whether a Reddit user will submit or comment on specific content, a deep multi-layer perceptron (MLP) architecture is used. This network uses user embeddings and content embeddings as inputs, producing a probability score indicating potential user interaction with the content. The key components of the architecture are:

- User Embedding Layer: A learnable embedding vector representing user features (e.g., user history, preferences).
- Content Embedding Layer: A learnable embedding vector representing the features of the content (e.g., topic, style, previous engagements).
- Hidden Layers: A series of fully connected layers with non-linear activation functions (ReLU or GELU), and layer normalization to stabilize training.
- Output Layer: A sigmoid activation function produces a probability score indicating whether the user will interact (submit or comment) with the content.

The network is trained to minimize cross-entropy loss between predicted interaction scores and actual user behavior. The approach draws inspiration from collaborative filtering in recommendation systems [12, 33], predicting user-content relationships similar to user-item predictions. By incorporating non-linearities, the model captures complex user preferences and interactions more effectively.

However, unlike conventional recommendation system research which focuses on optimizing predictive accuracy, this model emphasizes policy research and agent behavior simulation. The simplified MLP structure aids in realistic agent simulation and analysis without the complexities of cutting-edge recommendation algorithms, making it easier to integrate into LLM-enhanced agent frameworks for studying polarization and social dynamics.

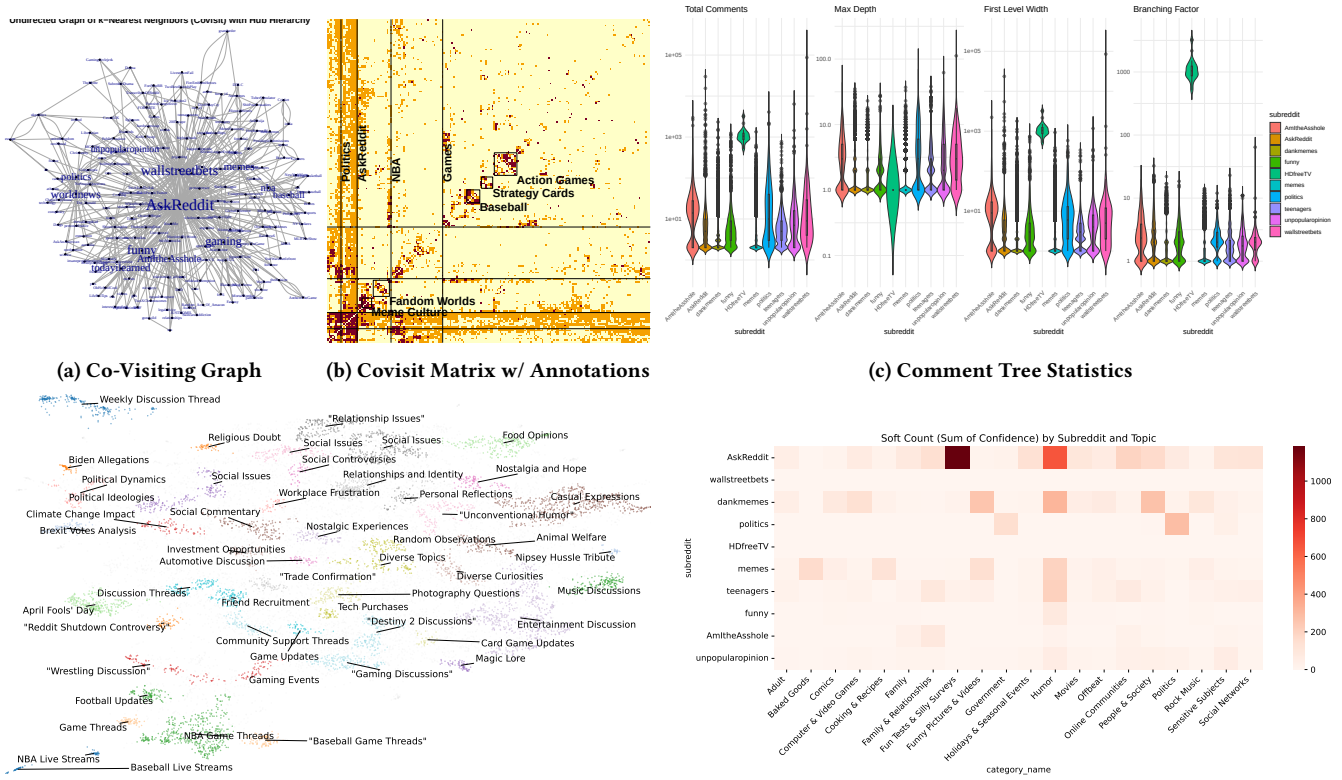
4 EXPERIMENT

To explore the dynamics of online communities and their behaviors, we use Reddit as a representative platform due to its diverse range of user interactions and content-sharing practices. Reddit was created as the "front page of the Internet," where users share content in subreddits based on interests (e.g., `r/science`, `r/gaming`). An upvote/downvote system ranks posts, with visibility driven by user engagement. With 73.1 million daily users, Reddit primarily serves for entertainment (72%), news (43%), brand following (17%), and networking (13%). Popular subreddits like `r/funny`, `r/AskReddit`, `r/gaming`, and `r/worldnews` drive significant activity, offering a rich space for studying social dynamics.

4.1 Reddit Dynamics and Structure

Reddit [4] exhibits a highly complex structure that operates on both macroscopic and microscopic levels. At the macroscopic level, the platform consists of a fractal-like network of trees and clusters, with `r/AskReddit` as the central hub. Surrounding this central hub are various subhubs, and further surrounding these subhubs are smaller, specialized hubs. Together, these top 200 subreddits, out of a total of 30,000, account for nearly half of all Reddit users and their interactions. This concentration of activity provides shared experiences and a sense of community for a significant portion of Reddit users.

If we examine these co-visiting patterns (Fig. 2a and Fig. 2b), we can already observe signs of information filtering and interaction dynamics. For instance, within the baseball cluster (Fig. 2b), subreddits such as `r/Braves`, `r/Cardinals`, `r/NewYorkMets`, and `r/NYYankees` form separate information bubbles. Although these subreddits do not interact directly with one another, they converge in larger baseball-focused communities like `r/baseball` and



(a) Co-Visiting Graph (b) Covisit Matrix w/ Annotations (c) Comment Tree Statistics

(d) Topic Clusters (e) Topics Heatmap

Figure 2: The macroscopic structure at the Reddit level, the microscopic structure of subreddit comment trees, and the distribution of topics among top subreddits. (a) Co-visiting patterns with subreddit hubs. (b) Covisit matrix showing clusters of shared visits. (c) Violin plots illustrating subreddit comment tree depth and breadth. (d) Topic Clusters showing semantic groups of submissions. (e) Topics Heatmap illustrating thematical overlapping across top-10 subreddits.

`r/fantasybaseball`. Similarly, patterns suggest that teenagers engage more with meme culture and fandom worlds than with subreddits centered around politics or relationships.

At the microscopic level, we observe a similar concentration of activity. Out of the millions of submissions posted daily, only a few thousand manage to capture half of the users' attention through comments and views, generating shared experiences. In essence, most of the submissions are peripheral, just as most subreddits are either niche or non-essential, much like real-world social interactions. Within each subreddit, the interaction patterns, as reflected in the structure of comment trees (Fig. 2c), vary significantly. The size, depth, and branching of these trees indicate different levels of engagement—some trees are wide and shallow, representing broad but superficial interactions, while others are deep with high branching factors, indicating more involved and intense discussions.

This fractal structure—both at the platform level and within individual subreddits—suggests that a few highly active subreddits and posts drive much of the shared experience on Reddit, creating both cohesive communities and isolated information bubbles. Through our analysis, we show how Reddit's complexity can be measured and managed, allowing us to identify important patterns in user interaction and information dissemination.

In addition to Figures 2a–2c, Figures 2d and 2e illustrate how embedding-based methods, combined with dimension reduction and clustering, reveal coherent thematic groupings in a single day of Reddit submissions. In Figure 2d, Uniform Manifold Approximation and Projection [25] projects large language model (LLM) embeddings into two dimensions, followed by Hierarchical Density-Based Spatial Clustering of Applications with Noise [24], with the LLM automatically labeling the resulting clusters (e.g., sports or politics). Meanwhile, Figure 2e uses the Google Cloud Platform Natural Language Processing application programming interface [16]—also driven by language-model embeddings—to classify submissions into the top topics across the ten most-commented subreddits, highlighting distinct coverage and overlap. These techniques help users break out of narrow "filter bubbles," fostering a more inclusive Reddit experience.

Incorporating sentiment analysis indicates that subreddits focused on lighthearted or encouraging themes (for instance, `r/aww` and `r/wholesomememes`) generally maintain a positive overall tone, while those centered on conflict or venting (e.g., `r/entitledparents` and `r/unpopularopinion`) tend toward more negative sentiments.

By integrating sentiment analysis with structural and language-based approaches, both platforms and researchers can more effectively detect tightly insulated echo chambers and direct users toward broader, multi-perspective information sources.

4.2 Implementation

In this experiment, we develop a framework to simulate Reddit-like agent interactions using generative models and a recommendation system to replicate real-world behaviors.

The preparation phase focuses on training generative models for content and user embeddings to capture interaction diversity. Using PaLM 2, we compute text chunk embeddings (e.g., titles, comments) and represent each user embedding as the mean of their textual contribution embeddings. A diffusion model is trained to produce both content and user embeddings, capturing latent social dynamics. Additionally, a recommendation system modeled as a multi-layer perceptron (MLP) is trained to score content relevance for users. This MLP uses both user and content embeddings as inputs, with regularization terms to balance cohesion, polarization, and diversity.

During simulation, agents are initialized with embeddings generated by the user diffusion model. The context (e.g., time of day) determines whether an agent will submit new content or interact with existing content. If submitting, a content retrieval step selects candidate text chunks from a vector store, and an LLM generates a new post based on these examples. If commenting is favored, a personalized content feed is ranked by the MLP, and the agent's decision to comment, upvote, or downvote is driven by sentiment and the recommendation score.

For computational efficiency, interactions are generated in batches within short 10-minute windows. This manages memory and processing constraints but may limit interaction depth in real-time. Simulation quality is assessed by comparing generated content trees' properties (e.g., size, depth) with real data. In policy research applications, the framework supports modifying recommendation regularization terms to influence cohesion and polarization.

The algorithm pseudocode is given in Alg. 1. This implementation efficiently simulates Reddit-like interactions through agent-based modeling, generative content generation, and personalized recommendations, allowing for exploration of social dynamics in a scalable manner.

Algorithm 1 Simulation of Agent Interactions on Reddit

Input: User embeddings, content embeddings
Train generative models for user/content embeddings
Train recommendation model (MLP) using embeddings
Initialize agents with embeddings from user diffusion model
for each time window t_w **do**
  for each agent a in agents **do**
    if should_submit(a, t_w) **then**
      Retrieve example submission s from vector store
      Generate new submission $\hat{s}$ with LLM based on s
      Submit $\hat{s}$ as new content
    else
      Obtain personalized feed F for a via MLP
      Rank content in F based on recommendation score
      Decide comment/upvote/downvote on content
      Generate response using LLM and post
    end if
  end for
  if end of batch processing **then**
    Flush data to storage
  end if
end for
Evaluate simulation against real-world data

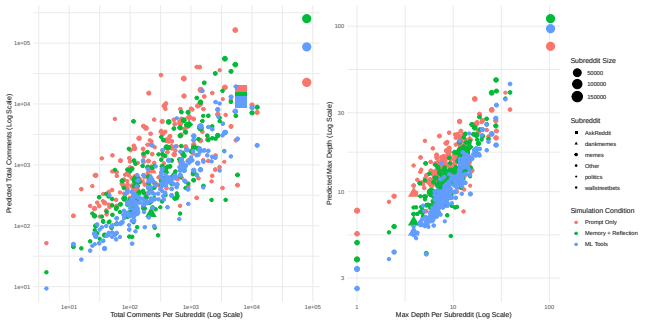


Figure 3: Simulation performance across subreddits: The left panel compares predicted vs. real total comments, and the right panel shows maximum comment tree depth. Subreddit sizes reflect total interactions, with distinct shapes for the top 5. Points near the diagonal indicate higher prediction accuracy.

4.3 Evaluating Social Simulation Fidelity

In our evaluation, we focus on two key statistics: total comments and maximum depth of comment trees. Total comments reflect the overall level of user engagement, while maximum depth shows how deeply users engage in discussions, with deeper trees indicating more substantial back-and-forth interactions. To make these metrics more representative of the interaction dynamics, we use weighted statistics where the weight of each submission is proportional to the number of comments it receives. This ensures that the most active and visible discussions, which contribute more significantly to the community's experience, are appropriately emphasized in the evaluation.

To evaluate performance, we compare three simulation scenarios. First, Prompt Only uses a large language model (LLM) that generates responses based on similar examples retrieved through semantic search in a vector store, reflecting a few-shot learning approach. The LLM predicts user engagement and submission behavior directly from these examples, with predictions averaged across multiple rounds. Second, Memory + Reflection improves upon the first approach by prompting the LLM to reflect on engagement patterns hierarchically, allowing it to abstract themes and save those insights for future interactions. This reflection-based approach helps the LLM contextualize why certain topics gather more user interaction. Lastly, ML Tools employs a deep neural network, as described in our method section, which serves as a

more numerically precise collaboration filter. This network captures the complexities of user-content interactions in a way that the LLM alone cannot, allowing for more accurate predictions of both engagement levels and discussion depth.

The comparison of the three simulation approaches (Fig. 3) shows that the "Prompt Only" method underperforms, particularly in predicting deeper interactions and engagement levels, especially for larger subreddits. The "Memory + Reflection" approach improves prediction accuracy by abstracting from examples, but still struggles with deeper discussions. The "ML Tools" method, which integrates a neural network alongside the LLM, performs best, accurately capturing both total comments and interaction depth across a variety of subreddits. This highlights the advantage of combining LLMs with specialized tools to simulate more realistic and complex social behaviors.

4.4 Modeling Cohesion, Polarization, and Bias

We incorporate cohesion, polarization, and bias as core social dynamics in training a multi-layer perceptron (MLP)-based recommendation system. By treating these dimensions as regularization terms, we explore how varying content recommendations affect user engagement, community clustering, and exposure diversity, thereby offering a flexible platform for policy research. Specifically:

- **Cohesion** measures how closely users in the same community connect and share interests. In network terms, it is indicated by the density of connections among frequently interacting users; semantically, it reflects similarity in user embeddings. High cohesion suggests a tightly knit group with shared narratives.
- **Polarization** reflects how sharply users split into opposing communities with minimal cross-group contact. On the network side, it appears as high modularity (well-separated clusters); semantically, it manifests as divergent topic distributions, with each group consuming content aligned with its stance.
- **Bias** represents the skew in a user's content exposure toward existing beliefs. Network bias is how often a user's feed matches their current views, and semantic bias is measured by alignment between content and the user's embedding. High bias indicates an echo-chamber environment that continuously reinforces familiar perspectives.

Our approach uses a "policy" collaborative filter that builds on a baseline filter trained to reflect real-world data. We gather training examples from simulated interactions, where each agent's behavior is shaped by learned embeddings (e.g., from a diffusion model) and the policy filter's recommendations. During each update, the policy filter balances standard engagement accuracy with cohesion, polarization, and bias targets by including these social metrics in its loss function. This continuous learning loop refines recommendations to achieve desired outcomes—such as increasing content diversity or strengthening in-group bonding—while still aligning with observed user behavior.

Adjusting the regularization weights in the policy filter allows us to simulate interventions ranging from promoting cross-community exposure to mitigating echo chambers. For instance, reducing bias magnifies the variety of content users receive but can dilute intra-community cohesion. These controlled experiments reveal trade-offs among engagement, cohesion, polarization, and bias, offering insights for platform designers and policymakers interested in shaping healthier or more inclusive online communities.

Table 1: Comparison of social metrics across regularization settings in the MLP model. Network and semantic cohesion, polarization, and bias are compared. The baseline model has no regularization, while the other columns show the impact of specific regularization techniques.

Metric	Baseline	+Cohesion	-Polarity	-Bias
Connection Density	0.40	0.55	0.48	0.42
Embedding Similarity	0.30	0.45	0.40	0.32
Modularity	0.65	0.58	0.42	0.63
Topic Divergence	0.60	0.52	0.40	0.57
Content Alignment	0.75	0.70	0.68	0.55
Engagement Acc (%)	85%	83%	81%	84%

In Table 1, the base model demonstrates lower cohesion and higher modularity, indicating more fragmented interactions. When cohesion is emphasized, users cluster more tightly; however, polarization also increases as groups become more insular. Incorporating bias regularization reduces overexposure to similar content, which decreases user bias and slightly improves engagement accuracy, striking a balance between social cohesion and diversity. These findings indicate that by affecting the balance of social metrics in the recommendation model, we can manipulate the structure and behavior of online communities, offering a valuable tool for studying and potentially mitigating the effects of echo chambers and polarization on social media platforms.

5 CONCLUSIONS AND DISCUSSIONS

We introduced a multi-agent simulation framework that integrates large language models, generative embeddings, and collaborative filtering to capture Reddit-style social dynamics at scale. This approach employs retrieval-augmented generation to adapt LLMs for domain-specific tasks, handle large corpora, and focus on complex social contexts. By comparing generative agents against simpler baselines, we showed how trainable neural network components can balance user engagement with broader policy goals (e.g., curbing echo chambers). Our results highlight three key insights: (1) popularity emerges from the interplay of content, audience, and influencers; (2) shared experiences coalesce into distinctive community identities; and (3) flexible deep-learning modules allow continuous adaptation to large, evolving datasets. The complete source code and documentation are publicly available at `https://github.com/wendongml/LLM-SocSim`, facilitating further research on simulating social media interventions and evaluating complex policy scenarios.

DISCLAIMER

The views expressed are those of the authors and do not reflect the official guidance or position of the United States Government, the Department of Defense or of the United States Air Force.

The content or appearance of hyperlinks does not reflect an official DoD, Air Force, Air Force Research Laboratory position or endorsement of the external websites, or the information, products, or services contained therein.

REFERENCES

[1] Alberto Acerbi and Joseph M Stubbersfield. 2023. Large language models show human-like content biases in transmission chain experiments. *Proceedings of the National Academy of Sciences* 120, 44 (2023), e2313790120.

[2] Benedict Anderson. 1983. *Imagined Communities: Reflections on the Origin and Spread of Nationalism*. Verso, London, UK.

[3] Robert Axelrod. 1997. The dissemination of culture: A model with local convergence and global polarization. *Journal of conflict resolution* 41, 2 (1997), 203–226.

[4] Jason Baumgartner, Savvas Zannettou, Brian Keegan, Megan Squire, and Jeremy Blackburn. 2020. The pushshift reddit dataset. In *Proceedings of the international AAAI conference on web and social media*, Vol. 14. AAAI Press, Palo Alto, CA, USA, 830–839.

[5] Marcel Binz and Eric Schulz. 2023. Using cognitive psychology to understand GPT-3. *Proceedings of the National Academy of Sciences* 120, 6 (2023), e2218523120.

[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] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language Models are Few-Shot Learners. *arXiv preprint arXiv:2005.14165* (2020).

[8] Talha Burki. 2019. Vaccine misinformation and social media. *The Lancet Digital Health* 1, 6 (2019), e258–e259.

[9] Claudio Castellano, Santo Fortunato, and Vittorio Loreto. 2009. Statistical physics of social dynamics. *Reviews of modern physics* 81, 2 (2009), 591–646.

[10] Abhijnan Chakraborty, Gourab K Patro, Niloy Ganguly, Krishna P Gummadi, and Patrick Loiseau. 2019. Equality of voice: Towards fair representation in crowdsourced top-k recommendations. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*. 129–138.

[11] Serina Chang, Alicja Chaszczewicz, Emma Wang, Maya Josifovska, Emma Pierson, and Jure Leskovec. 2024. LLMs generate structurally realistic social networks but overestimate political homophily. *arXiv preprint arXiv:2408.16629* (2024).

[12] Paul Covington, Jay Adams, and Emre Sargin. 2016. Deep neural networks for youtube recommendations. In *Proceedings of the 10th ACM conference on recommender systems*. ACM, San Francisco, CA, USA, 191–198.

[13] Yann Dubois, Chen Xuechen Li, Rohan Taori, Tianyi Zhang, Ishaan Gulrajani, Jimmy Ba, Carlos Guestrin, Percy S Liang, and Tatsunori B Hashimoto. 2024. Alpacafarm: A simulation framework for methods that learn from human feedback. *Advances in Neural Information Processing Systems* 36 (2024), 30039–30069.

[14] Daocheng Fu, Xin Li, Licheng Wen, Min Dou, Pinlong Cai, Botian Shi, and Yu Qiao. 2024. Drive like a human: Rethinking autonomous driving with large language models. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*. 910–919.

[15] 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).

[16] Google. 2023. Google Cloud Platform Natural Language Processing Documentation. https://cloud.google.com/natural-language/docs. Accessed: 2023-10-01.

[17] Rainer Hegselmann and Ulrich Krause. 2002. Opinion dynamics and bounded confidence: Models, analysis and simulation. *Journal of Artificial Societies and Social Simulation* 5, 3 (2002), 1–33.

[18] Natali Helberger, Kari Karppinen, and Lucia D'acunto. 2018. Exposure diversity as a design principle for recommender systems. *Information, communication & society* 21, 2 (2018), 191–207.

[19] Jonathan Ho, Ajay Jain, and Pieter Abbeel. 2020. Denoising diffusion probabilistic models. *Advances in neural information processing systems* 33 (2020), 6840–6851.

[20] Yuxuan Hu, Gemju Sherpa, Lan Zhang, Weihua Li, Quan Bai, Yijun Wang, and Xiaodan Wang. 2024. An LLM-enhanced Agent-based Simulation Tool for Information Propagation. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence (IJCAI-24), Demonstrations Track*. IJCAI, Hobart, Australia; Auckland, New Zealand; Jilin, China.

[21] Chenliang Li, He Chen, Ming Yan, Weizhou Shen, Haiyang Xu, Zhikai Wu, Zhicheng Zhang, Wenmeng Zhou, Yingda Chen, Chen Cheng, et al. 2023. ModelScope-Agent: Building Your Customizable Agent System with Open-source Large Language Models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*. 566–578.

[22] Nian Li, Chen Gao, Mingyu Li, Yong Li, and Qingmin Liao. 2024. Econagent: large language model-empowered agents for simulating macroeconomic activities. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 15523–15536.

[23] Ping Liu, Karthik Shivaram, Aron Culotta, Matthew A Shapiro, and Mustafa Bilgic. 2021. The interaction between political typology and filter bubbles in news recommendation algorithms. In *Proceedings of the Web Conference 2021*. 3791–3801.

[24] Leland McInnes, John Healy, and Steve Astels. 2017. HDBSCAN: Hierarchical Density Based Clustering. *The Journal of Open Source Software* 2, 11 (2017), 205. https://doi.org/10.21105/joss.00205

[25] Leland McInnes, John Healy, and James Melville. 2018. UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction. In *Proceedings of the 1st International Conference on Learning Representations (ICLR)*. https://arxiv.org/abs/1802.03426

[26] Silvia Milano, Mariarosaria Taddeo, and Luciano Floridi. 2020. Recommender systems and their ethical challenges. *Ai & Society* 35 (2020), 957–967.

[27] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. *Advances in neural information processing systems* 35 (2022), 27730–27744.

[28] Dimitris Paraschakis. 2016. Recommender systems from an industrial and ethical perspective. In *Proceedings of the 10th ACM conference on recommender systems*. ACM, San Francisco, CA, USA, 463–466.

[29] Eli Pariser. 2011. *The Filter Bubble: What the Internet is Hiding from You*. Penguin Press, New York, NY, USA.

[30] 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 *Proceedings of the 36th annual acm symposium on user interface software and technology*. ACM, San Francisco, CA, USA, 1–22.

[31] Ethan Perez, Saffron Huang, Francis Song, Trevor Cai, Roman Ring, John Aslanides, Amelia Glaese, Nat McAleese, and Geoffrey Irving. 2022. Red Teaming Language Models with Language Models. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*. 3419–3448.

[32] Ethan Perez, Sam Ringer, Kamile Lukosiute, Karina Nguyen, Edwin Chen, Scott Heiner, Craig Pettit, Catherine Olsson, Sandipan Kundu, Saurav Kadavath, et al. 2023. Discovering language model behaviors with model-written evaluations. In *Findings of the Association for Computational Linguistics: ACL 2023*. 13387–13434.

[33] Shaina Raza and Chen Ding. 2019. Progress in context-aware recommender systems—An overview. *Computer Science Review* 31 (2019), 84–97.

[34] Urbano Reviglio. 2017. Serendipity by design? How to turn from diversity exposure to diversity experience to face filter bubbles in social media. In *Internet Science: 4th International Conference, INSCI 2017, Thessaloniki, Greece, November 22-24, 2017, Proceedings 4*. Springer, 281–300.

[35] Fernando P Santos, Yphtach Lelkes, and Simon A Levin. 2021. Link recommendation algorithms and dynamics of polarization in online social networks. *Proceedings of the National Academy of Sciences* 118, 50 (2021), e2102141118.

[36] Fernando P Santos, Francisco C Santos, Jorge M Pacheco, and Simon A Levin. 2021. Social network interventions to prevent reciprocity-driven polarization. In *Proceedings of the 20th International Conference on Autonomous Agents and MultiAgent Systems*. 1643–1645.

[37] Murray Shanahan, Kyle McDonell, and Laria Reynolds. 2023. Role play with large language models. *Nature* 623, 7987 (2023), 493–498.

[38] Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. 2023. Reflexion: Language Agents with Verbal Reinforcement Learning. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*. 8634–8652.

[39] Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. 2024. Reflexion: Language agents with verbal reinforcement learning. *Advances in Neural Information Processing Systems* 36 (2024).

[40] Karan Singhal, Shekoofeh Azizi, Tao Tu, S Sara Mahdavi, Jason Wei, Hyung Won Chung, Nathan Scales, Ajay Tanwani, Heather Cole-Lewis, Stephen Pfohl, et al. 2023. Large language models encode clinical knowledge. *Nature* 620, 7972 (2023), 172–180.

[41] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. 2021. Score-Based Generative Modeling through Stochastic Differential Equations. In *International Conference on Learning Representations*. openreview, online. https://openreview.net/forum?id=PxTIG12RRHS

[42] Kamal Souali, Abdellatif El Afia, and Rdouan Faizi. 2011. An automatic ethical-based recommender system for e-commerce. In *2011 International Conference on Multimedia Computing and Systems*. IEEE, 1–4.

[43] Jean Springsteen, William Yeoh, and Dino Christenson. 2024. Algorithmic Filtering, Out-Group Stereotype, and Polarization on Social Media. In *Proceedings of the 23rd International Conference on Autonomous Agents and Multiagent Systems*. 1782–1790.

[44] Haotian Sun, Yuchen Zhuang, Lingkai Kong, Bo Dai, and Chao Zhang. 2024. Adaplanner: Adaptive planning from feedback with language models. *Advances in Neural Information Processing Systems* 36 (2024).

[45] Cass R. Sunstein. 2007. *Republic.com 2.0*. Princeton University Press, Princeton, NJ, USA.

[46] Tiffany Ya Tang and Pinata Winoto. 2016. I should not recommend it to you even if you will like it: the ethics of recommender systems. *New Review of Hypermedia and Multimedia* 22, 1-2 (2016), 111–138.

[47] Petter Törnberg, Diliara Valeeva, Justus Uitermark, and Christopher Bail. 2023. Simulating social media using large language models to evaluate alternative news feed algorithms. *arXiv preprint arXiv:2310.05984* (2023).

[48] Anshul Toshniwal and Fernando P Santos. 2023. Opinion Dynamics in Populations of Converging and Polarizing Agents. In *Proceedings of the 2023 International Conference on Autonomous Agents and Multiagent Systems*. 2763–2765.

[49] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971* (2023).

[50] Alan Tsang and Kate Larson. 2014. Opinion dynamics of skeptical agents. In *Proceedings of the 2014 international conference on Autonomous agents and multi-agent systems*. 277–284.

[51] Avish Vijayaraghavan and Cosmin Badea. 2024. Minimum levels of interpretability for artificial moral agents. *AI and Ethics* (2024), 1–17.

[52] Guanzhi Wang, Yuqi Xie, Yunfan Jiang, Ajay Mandlekar, Chaowei Xiao, Yuke Zhu, Linxi "Jim" Fan, and Anima Anandkumar. 2024. Voyager: An Open-Ended Embodied Agent with Large Language Models. Reviewed on OpenReview: https://openreview.net/forum?id=ehfRiF0R3a. *Transactions on Machine Learning Research* (March 2024). https://voyager.minedojo.org Equal contribution: Yunfan Jiang, Ajay Mandlekar. Equal advising: Linxi "Jim" Fan, Anima Anandkumar.

[53] Jindong Wang, Xixu Hu, Wenxin Hou, Hao Chen, Runkai Zheng, Yidong Wang, Linyi Yang, Wei Ye, Haojun Huang, Xiubo Geng, et al. 2024. On the Robustness of ChatGPT: An Adversarial and Out-of-distribution Perspective. *Data Engineering* (2024), 48.

[54] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V. Le, and Denny Zhou. 2022. Chain-of-Thought Prompting Elicits Reasoning in Large Language Models. In *Advances in Neural Information Processing Systems*, Vol. 35. 24824–24837.

[55] Laura Weidinger, Jonathan Uesato, Jakub Bielecki, Glenn van den Driessche, Mike Chrzanowski, Dmitriy Krasheninnikov, Martin Chadwick, Rohen Shah Gur, Amanda Glaese, Ruben Tréger, et al. 2021. Ethical and social risks of Large Language Models. *arXiv preprint arXiv:2112.04359* (2021).

[56] Xizhou Zhu, Yuntao Chen, Hao Tian, Chenxin Tao, Weijie Su, Chenyu Yang, Gao Huang, Bin Li, Lewei Lu, Xiaogang Wang, et al. 2023. Ghost in the minecraft: Generally capable agents for open-world environments via large language models with text-based knowledge and memory. *arXiv preprint arXiv:2305.17144* (2023).

[57] Ethan Zuckerman. 2013. *Rewire: Digital Cosmopolitans in the Age of Connection*. W. W. Norton & Company.