LLM Meets Social Media: Co-Evolution of Bot and Detector through Adversarial Learning

Anonymous ACL submission

Abstract

LLMs in social media research offer a double-edged sword: they generate human-like behavior, advancing the study of social dynamics, but also escalate risks like information manipulation, disinformation and misinformation. While previous work has simulated agents through prompt engineering or fine-tuning on human-annotated data, it often overlooked the potential of learning through social media, where diverse human data are available. Meanwhile, bot detection has typically relied on static datasets, missing the evolving nature of LLM-based bots. This paper introduces a novel adversarial learning framework that addresses both challenges, with the co-evolution of the **Evo**lving **Bot** (EvoBot) and **Detector**. EvoBot generates its own training data from previous iterations and refines its behavior based on the feedback from Detector, which is trained to distinguish between human and bot. Experimental results demonstrate that EvoBot improves its ability to bypass detection while effectively simulating real-world social dynamics, such as group opinions and information spread. Additionally, the iterative training process enhances the Detector's performance and generalization, showcasing the framework's effectiveness in generating human-like content and evolving bot detection. The code is available at `https://anonymous.4open.science/r/Anonymous_EvoBot-5442`.

1 Introduction

Social media exhibit a wide range of intricate collective behaviors and social intelligence, such as opinion dynamics (Chuang et al., 2023; Ma et al., 2024), social influence (Abbas Naqvi et al., 2020; Peng et al., 2016) and information spread (like rumors, social-disease contagion) (Chopra et al., 2024; Bauch and Galvani, 2013; Feng et al., 2018, 2019). With a global user base and diverse human data, social media are natural platforms to train and test artificial intelligence technology. Agent-Based Modeling (ABM) has emerged as a powerful tool for studying the phenomena mentioned above, providing a bottom-up framework where macro-level social patterns emerge from micro-level agent interactions (Gürcan, 2024). ABM allows for the simulation of hypothetical scenarios, sidestepping the ethical and logistical challenges associated with real-world experimentation. However, large-scale ABM has traditionally been limited by technological and computational constraints, often relying on simplified mathematical models or handcrafted rules (Macal and North, 2005, 2009). This simplification reduces the fidelity of simulations, limiting their ability to capture the complexity of real-world human interactions.

Recently, Large Language Models (LLMs) have demonstrated remarkable capabilities in generating human-like text, reasoning, and decision-making, making them strong candidates for enhancing agent intelligence in ABM (Gao et al., 2023, 2024; Yang et al., 2024). By leveraging LLMs, researchers can replace rigid rule-based agents with more adaptive, learnable entities capable of responding dynamically to environmental stimuli.

Individual heterogeneity is a critical factor for the intricate and diverse range of social intelligence (Putnam, 2000; Tajfel, 1979). However, existing studies suggest that pre-trained LLMs struggle to effectively represent the diversity of human preferences and values, especially those of marginalized groups (Cheng et al., 2023; Chakraborty et al.; He et al., 2024b). Meanwhile, in large-scale simulations, a trade-off often arises between precision and scale: detailed role modeling enhances authenticity but becomes computationally expensive, leading to simplified agent profiles to reduce costs (Chopra et al., 2024; Williams et al., 2023).

Fine-tuning has been shown to help LLMs better align with human preferences, specific behaviors and personalities, especially in role-playing tasks (Shao et al., 2023; Ge et al., 2024). However, fine-tuning often relies on human-annotated datasets (Rafailov et al., 2024; Ouyang et al., 2022) or high-quality AI-generated data (Josifoski et al., 2023; Deng et al., 2023), both of which can introduce biases and limit the model's ability to generalize. In contrast, global social platforms like X or Reddit offer diverse, user-generated content from various cultures and groups. The low barrier to entry ensures a wide range of perspectives, making these platforms a richer and more authentic data source compared to hand-crafted datasets. This raises the first research question:

> **RQ1:** *How can LLMs learn from social platform data to generate more human-like contents?*

On the other hand, as LLMs advance, they pose significant risks, particularly on social media, where they can amplify issues like misinformation and opinion polarization (Feng et al., 2024; Ferrara, 2023). While some bot detection methods have performed well on static datasets (Yang et al., 2020; Feng et al., 2021), and others have begun to address bot evolution (Dialektakis et al., 2022; Wu et al., 2019), there is still a lack of focus on the evolving nature of LLM-based bots. This leads to the second research question:

> **RQ2:** *How can we develop detectors considering the evolving LLM-based bots on social media?*

The two research questions are inherently adversarial. On one hand, LLMs need to generate increasingly human-like behavior; on the other, detection methods need to keep pace with the evolving LLM-based bots. Building on this concept, we propose a novel adversarial learning framework, where **EvoBot** learns to generate increasingly human-like content based on the feedback from **Detector**, which continuously refines its ability to distinguish evolving EvoBot from humans.

The framework is implemented on the well-known TwiBot-22 bot detection dataset (Feng et al., 2022), which contains data on both human and bot users and their tweets. EvoBot's learning process consists of two phases. In the first, an SFT dataset is created using human user descriptions and characteristics as prompts, with their tweets as responses, enabling EvoBot to learn human user representation. In the second phase, EvoBot undergoes iterative adversarial training with the Detector. Each iteration involves EvoBot generating several candidate outputs based on bot account data, which the Detector classifies as either human or bot. This feedback is used to construct a DPO (Rafailov et al., 2024) dataset, guiding EvoBot to produce more human-like content. The Detector is trained via supervised learning, augmenting its dataset with EvoBot's outputs at each iteration to enhance detection performance.

We theoretically analyze the convergence of the adversarial learning framework, and experimental results show that EvoBot enhances its ability to generate human-like content while effectively bypassing detection. This is accompanied by a significant improvement in the Detector's performance, making it more adept at distinguishing human from bot. In social simulation tasks, EvoBot outperforms the baseline by better replicating group opinions and information spread, showcasing its human-like performance and flexible response to environmental stimuli. These results highlight the importance of studying the co-evolution of LLM-based social bots and detectors in social media, which plays a crucial role in transforming how online interactions are shaped and regulated.

2 Related Works

LLM-based agents in social simulation. Recent studies have explored the use of LLMs as autonomous agents in social simulation, categorizing them into individual, scenario, and society-level simulations (Mou et al., 2024a). Individual-level studies focus on modeling specific personas or demographic groups to analyze behavioral patterns (Shao et al., 2023; Shen et al., 2023; Frisch and Giulianelli, 2024). Scenario-based simulations involve structured interactions among multiple LLM-driven agents to tackle domain-specific tasks, such as software development (Qian et al., 2023; Hong et al., 2023), question answering (Du et al., 2023), and judicial decision-making (He et al., 2024a). At the societal level, multi-agent simulations have been employed to examine emergent social behaviors (Park et al., 2023; Yang et al., 2024), including opinion dynamics (Chuang et al., 2023) and macroeconomic trends (Li et al., 2024). While most existing work relies heavily on prompt engineering to guide agent behaviors, EvoBot distinguishes itself by adopting a learning-based approach, enabling agents to adapt and improve through training.

Adversarial learning. Adversarial learning has been successfully applied in traditional NLP tasks like text generation (Yu et al., 2017; Li et al., 2017), and more recently in social bot and fake text detection. GANs have been used to generate synthetic bot samples to address class imbalance (Wu et al., 2019, 2020; Dialektakis et al., 2022), but these methods often struggle with detecting evolved bots that adapt to bypass detection systems. Cresci (2020) introduced a proactive detection method using genetic algorithms, while Jan et al. (2020) proposed a GAN-based framework with two generators to detect advanced bot variants. With the rise of LLMs, AI-generated text detection has become more challenging (Wu et al., 2025), though some adversarial methods (Hu et al., 2023; Koike et al., 2024) have improved detection accuracy. Unlike these methods, which focus solely on detection, EvoBot's dual focus on both generation and detection makes it a powerful tool in the ongoing arms race between AI creators and detection systems.

3 Methodology

The learning process of EvoBot includes two phases. In the first phase, supervised fine-tuning (SFT) is conducted on real human data to pre-train EvoBot on the expressive habits, linguistic styles, and contextual preferences of community members. In the second phase, adversarial learning is used, with both the EvoBot and the Detector iteratively trained. EvoBot's objective is to generate tweets that are most likely to be classified as human by the Detector, while the Detector aims to improve its accuracy in distinguishing between bots and humans. The following parts provide a problem formulation and a detailed description of all modules. Figure 1 provides an overview of this framework. And the learning process is detailed in Algorithm 1.

3.1 Problem Formulation

The social media dataset is modeled as a tuple $(\mathcal{V}, \{A_v\}, \{T_v\}, \mathcal{E})$, where $\mathcal{V}$ is the set of nodes, representing users, divided into two classes $\mathcal{H}$ (humans) and $\mathcal{B}$ (bots), i.e., $\mathcal{V} = \mathcal{H} \cup \mathcal{B}$. $\mathcal{E}$ is the set of directed edges, where $(u, v) \in \mathcal{E}$ indicates that user u follows user v. Each user $v \in \mathcal{V}$ is associated with two types of attributes: **Account Information** $A_v = \{a_1, a_2, \ldots, a_m\}$, which includes account features on Twitter, such as account creation time, number of followers, user description, and so on. **Tweets** $T_v = \{t_1, t_2, \ldots, t_n\}$, which represents a set of tweets posted by v.

The adversarial learning proceeds for K rounds. In the k-th round, **EvoBot**, represented by π_θ^k, generates tweets for a target user by integrating both the user's and their neighbors' information. Specifically, for user v, we use GPT-4o-mini (M_{sum}) to condense their account information A_v and historical tweets T_v into a concise summary $S_v = M_{\text{sum}}(A_v, T_v)$, which forms the first input. Similarly, the neighbor information is summarized as $S_{\mathcal{N}_v} = M_{\text{sum}}(A_{\mathcal{N}_v}, T_{\mathcal{N}_v})$. These, along with a task instruction $\mathcal{I}$, guide EvoBot to generate tweets that align with the target user's profile and fit naturally into the community. The tweets are then generated as $T_v \propto \pi_\theta^k(T_v | \mathcal{I}, S_v, S_{\mathcal{N}_v})$.

Detector in the k-th round is defined as $F^k = \sum_{j=0}^{k} w^j f^j$, where $f : (\mathbf{A}, \mathbf{T}, \mathcal{E}) \to \mathbf{p}$ is the classifier trained in each round. Note that $F^0 = f^0$ is the base detector trained on the original dataset. $\mathbf{A} = \{A_1, A_2, \ldots, A_N\}$,$\mathbf{T} = \{T_1, T_2, \ldots, T_N\}$ represent account information and tweets for all N users, respectively. $\mathbf{p} = [p_1, p_2, \ldots, p_N]$ is the vector of probabilities, where $p_v \in [0, 1]$ is the probability that user v is classified as a human.

3.2 EvoBot

Supervised Fine-Tuning The SFT dataset is constructed by selecting a subset of human users $\mathcal{H}_{\text{SFT}} \subseteq \mathcal{H}$. For each $v_h \in \mathcal{H}_{\text{SFT}}$, the prompt is $(\mathcal{I}, S_{v_h}, S_{\mathcal{N}_{v_h}})$. The reference response is the l tweets $T_{v_h} = \{t_{v_h,1}, t_{v_h,2}, \ldots, t_{v_h,l}\}$ sampled from the user's historical tweets T_{v_h}.

The objective of SFT is to minimize the discrepancy between the tweets generated by the base model of EvoBot π_θ^0 and the reference responses. This is achieved by optimizing the negative log-likelihood loss: $\mathcal{L}_{\text{SFT}} = -\frac{1}{|\mathcal{H}_{\text{SFT}}|} \sum_{v_h \in \mathcal{H}_{\text{SFT}}} \log \pi_\theta^0(T_{v_h} | \mathcal{I}, S_{v_h}, S_{\mathcal{N}_{v_h}})$.

Adversarial Learning with Detector In this phase, EvoBot is trained to generate tweets that evade detection as bot-generated. A straightforward approach would be to use the Detector's output—the probability of being classified as human—as a scalar reward in reinforcement learning. However, this method faces challenges like reward sparsity and instability in gradient estimation, which can result in inefficient and suboptimal training (Cao et al., 2023; Zhang et al., 2024). Here, we use DPO—a fine-tuning approach that directly leverages the preference ordering in the data rather than training an additional reward model (Rafailov et al., 2024).

Specifically, in the k-th round of adversarial learning, N bot users $\{v_{b_i} \in \mathcal{B}, i = 1, \ldots, N\}$ are

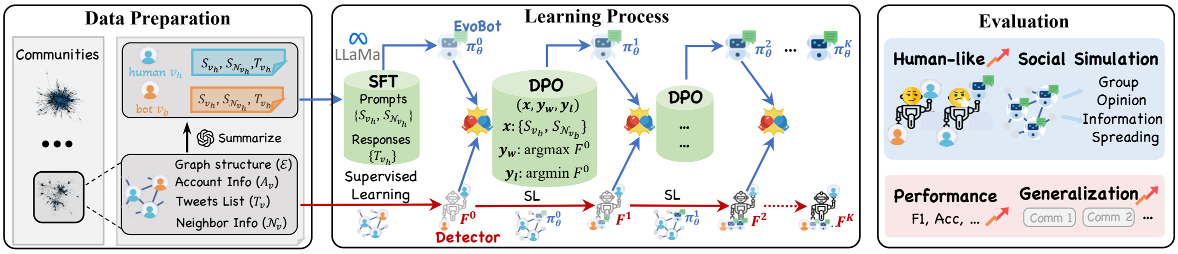


Figure 1: Overview of the EvoBot Framework. This diagram outlines the EvoBot workflow, consisting of three stages: **(1) Data Preparation**, where community detection and data extraction for bot and human accounts occur; **(2) Learning Process**, where EvoBot is first pre-trained on human data through SFT to get EvoBot π_θ^0, while the Detector is trained on the original dataset to obtain Detector F^0. The two then undergo iterative adversarial training, where the Detector aims to distinguish bots and evolving versions of EvoBot from human users, while EvoBot uses the Detector's classifications to improve its performance through DPO, maximizing the probability of being classified as human; **(3) Evaluation**, evaluating EvoBot's human-likeness and performance in group opinion and information spreading simulations, as well as the Detector's classification performance and generalization.

randomly sampled with replacement, and EvoBot generates C candidate responses $\{T_{v_{b_i},c}\}_{c=1,\dots,C}$ for each v_{b_i}. The Detector F^k then evaluates each candidate while keeping all other users' information fixed, calculating the probability that v_{b_i} is human for each response $T_{v_{b_i},c}$, denoted as $F^k_{v_{b_i},c}$.

To construct the DPO dataset $\mathcal{D}_{\text{DPO}} = \{x^i, y^i_w, y^i_l\}_{i=1}^N$, we let $x^i = (\mathcal{I}, S_{v_{b_i}}, S_{\mathcal{N}_{v_{b_i}}})$, $y^i_w = \arg\max_c F^k_{v_{b_i},c}$, $y^i_l = \arg\min_c F^k_{v_{b_i},c}$, where x^i is the input context, and y^i_w and y^i_l are the tweets with the highest and lowest probabilities of being classified as human, respectively. The loss function is $\mathcal{L}_{\text{DPO}} = -\mathbb{E}_{x^i,y^i_w,y^i_l}\left[\log\sigma\left(\beta\log\frac{\pi^k_\theta(y^i_w|x^i)}{\pi^{k-1}_\theta(y^i_w|x^i)} - \beta\log\frac{\pi^k_\theta(y^i_l|x^i)}{\pi^{k-1}_\theta(y^i_l|x^i)}\right)\right]$, where σ is the sigmoid function, and β is a hyperparameter controlling the deviation from the $k-1$'s version of EvoBot.

3.3 Detector

Features extraction Our detector employs a feature extraction approach inspired by (Feng et al., 2021). More specifically, the classifier $f : (\mathbf{A}, \mathbf{T}, \mathcal{E}) \to \mathbf{p}$ takes as input account information $\mathbf{A}$, tweets $\mathbf{T}$, and the graph structure $\mathcal{E}$. The account information A_v includes numerical properties such as account creation time and number of followers, which are normalized for balanced scaling, as well as categorical properties like user description and verified status, represented using one-hot encoding for binary interpretability. The textual data in T_v is embedded by RoBERTa (Liu, 2019) to capture semantic content. These features are processed through separate linear layers with LeakyReLU activations and then combined into a unified embedding. To incorporate the graph structure, we use an RGCN layer that aggregates relational information from the graph $\mathcal{E}$ based on the relation types. The resulting embeddings pass through fully connected layers with dropout regularization, producing a binary classification output that predicts whether a user is a bot or a human.

Supervised Learning In the k-th round of adversarial training, to obtain Detector F^k, all bot tweets in the dataset are replaced with outputs generated by EvoBot π_θ^{k-1} from the previous round. This modified dataset is then used to train the classifier f^k via supervised learning, using a cross-entropy loss to maximize classification accuracy.

3.4 Theoretical Analysis

In this subsection, we provide a theoretical analysis for our method from a more general view. We assume that data on social platforms can be represented as $(\mathbf{x}, \mathbf{y})$, where $\mathbf{x}$ denotes various user attributes, such as age, gender, occupation, interests, etc., sampled from the marginal distribution $q(\cdot)$ of the entire community. Meanwhile, $\mathbf{y} \sim \pi_{\mathcal{H}}(\cdot|\mathbf{x})$ represents the user's activities on the platform, such as posting tweets, retweeting, and liking, where $\pi_{\mathcal{H}}$ is the decision model of real humans in the community. Similarly, π_θ denotes EvoBot's model.

Detector is trained to maximize the probability of correctly classifying real and fake samples:

$$\begin{aligned} F = \arg\max\, &\mathbb{E}_{\mathbf{x}\sim q(\cdot),\mathbf{y}\sim\pi_{\mathcal{H}}(\cdot|\mathbf{x})}[\log F(\mathbf{x},\mathbf{y})] \\ &+\mathbb{E}_{\mathbf{x}'\sim q'(\cdot),\mathbf{y}'\sim\pi_\theta(\cdot|\mathbf{x}')}[\log(1-F(\mathbf{x}',\mathbf{y}'))] \end{aligned} \quad (1)$$

Here, the inputs $\mathbf{x}' \sim q'(\cdot)$ for EvoBot are distinguished from $q(\cdot)$, indicating that the input information received by EvoBot may come from a different distribution than the input received by real humans.

Considering the construction method of our DPO dataset in Sec. 3.2 and referring to Rafailov et al. (2024), the optimization objective of EvoBot is:

$$\begin{aligned}\pi_\theta = \arg\min \mathbb{E}_{\mathbf{x}' \sim q'(\cdot), \mathbf{y}' \sim \pi_\theta(\cdot|\mathbf{x}')}[1 - \log F(\mathbf{x}', \mathbf{y}')] \\ + \beta \mathbb{E}_{\mathbf{x} \sim q(\cdot), \mathbf{x}' \sim q'(\cdot)}[KL(\pi_{\mathcal{H}}(\cdot|\mathbf{x}) \| \pi_\theta(\cdot|\mathbf{x}'))].\end{aligned} \tag{2}$$

Theorem 1. *If $q'(\mathbf{x}) = q(\mathbf{x})$, then under the iterative training process of the detector and generator with the optimization objective (1) and (2), the global optimum is achieved when $\pi_\theta = \pi_{\mathcal{H}}$.*

The proof is provided in Appendix A.

4 Experiment Setup

4.1 Dataset

We use TwiBot-22 (Feng et al., 2022), a graph-based Twitter dataset that includes one million users, nearly one hundred million tweets, and various relational data. In this dataset, we represent users as nodes and model follower-followee relationships as directed edges in a graph. Given the impracticality of training EvoBot on the entire dataset due to its size and complexity, we divide the network into smaller, more manageable communities using the *Louvain* community detection method (Blondel et al., 2008), identifying 12 highly connected and representative communities. These communities exhibit diverse network topologies (e.g., star-shaped, mesh-like structures), support multiple languages, and focus on a variety of topics. See Appendix B for data details and the preprocessing.

4.2 Models and Trainings

EvoBot is based on `Llama-2-7b-chat` (Touvron et al., 2023). For fine-tuning, we use the `transformers` and `trl` libraries to implement SFT and DPO. And we apply low-rank adaptation (LoRA) (Hu et al., 2021) using the `peft` library.

Training and inference are performed on 8 NVIDIA RTX 3090 GPUs, with each community requiring approximately 10 hours. EvoBot runs $K = 4$ iterations for adversarial learning, using 1024-sample datasets for both SFT and DPO. The Detector is trained with an 8:1:1 split for training, validation, and test sets, with performance evaluated on the test set.

All model architectures and training hyperparameters are detailed in the Appendix C.

4.3 Simulation framework

We use an open-source social media simulation framework HiSim (Mou et al., 2024b) to analyze the response dynamics of EvoBot and baselines as users react to trigger events, focusing on group opinion and information spreading. Since EvoBot's learning is centered on tweet generation, we simplify the simulation by excluding user actions such as likes and retweets. To align the simulation with the real dataset, we replace all users—both human and bot—with EvoBot. At each step, the input includes the prompt from Section 3.2, the current trigger event, past events over several steps, and the most recent tweets from followed users.

5 Results

In this section, we present two sets of experiments designed to evaluate EvoBot's performance. The first focuses on model improvement through adversarial training, assessing how EvoBot's ability to generate human-like content and the Detector's detection capabilities evolve. The second examines simulation results, where we assess EvoBot's ability to simulate group opinion and information spreading in real-world events.

5.1 Adversarial Learning

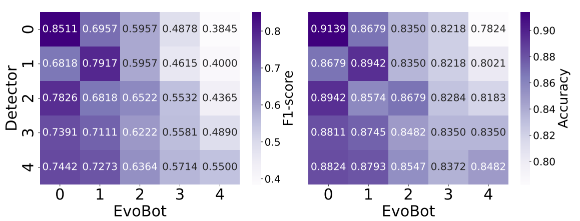


Figure 2: Classification performance across adversarial training iterations. The left matrix shows F1-score and the right shows accuracy, with rows representing Detector versions and columns representing EvoBot versions.

To evaluate the impact of adversarial training on both EvoBot and the Detector, we present classification performance in terms of accuracy and F1-score across different training iterations. The use of both metrics is crucial due to the class imbalance between positive (human) and negative (bot) samples in the dataset. As shown in Figure 2, the i-th row and j-th column represent the classification result obtained by replacing all bots in the original dataset with the j-th version of EvoBot π_θ^j, and using the i-th version of Detector F^i for classification.

Figure 2 shows the average classification performance across 12 different communities. Examining each row, we observe a general decline in the Detector's binary classification performance as adversarial training progresses, indicating that EvoBot becomes increasingly indistinguishable from the real human. On the other hand, analyzing each column, we notice an overall improvement in the Detector's performance for specific versions of EvoBot. This is because the Detector continuously learns from progressively more advanced versions of EvoBot.

However, these trends are not strictly monotonic—for instance, Detector F^2 performs worse than Detector F^1 when classifying EvoBot π_θ^1. This can be attributed to early-stage EvoBot versions producing lower-quality outputs, leading to overfitting when the Detector undergoes supervised training on these weak adversarial examples.

5.1.1 Generator

We compare the generative capabilities of the final version of EvoBot π_θ^4 (**Ours**) and other models in generating human-like tweets under the Detector F^0 trained on the original dataset. We include six baselines: **(1) Origin**: The bots in the original dataset; **(2) GAN**: We use the PyTorch-GAN open-source repository[1]. Since language is discrete and non-differentiable, we train the generator to directly produce vectors with the same dimensionality as the embeddings of tweets processed by RoBERTa; **(3) Llama2-7b; (4) GPT-4o-mini**: The two pre-trained LLMs use the same prompts and generation parameters as EvoBot; **(5) w/o ADV**: This ablation removes the adversarial learning process by training for only one iteration. To maintain a constant total amount of training data, the DPO dataset is scaled to $N = KN$. **(6) w/o SFT**: This ablation removes the SFT phase. Additionally, to further assess the generative capacity of different models, we replace the original RGCN-based Detector with a **GAT** model (Veličković et al., 2017).

Table 1 presents the Detector's classification performance for different generators, evaluated using both accuracy and F1-score. Smaller values indicate stronger generator performance, as the generated content becomes more difficult to distinguish from tweets by human users. EvoBot consistently outperforms other models in this regard, effectively evading the Detectors and achieving the lowest classification accuracy and F1-score. In contrast, GAN struggles to capture meaningful language features in the embedding space, making it highly detectable by the Detectors and resulting in the poorest performance. Moreover, the generated vectors fail to decode into coherent, natural language. Both Llama and GPT perform worse than EvoBot. Figure 8 gives an example to illustrate this gap, showing how EvoBot progressively generates lifelike outputs that closely resemble the tweets of real human users on social media. The two ablation studies highlight the importance of both SFT and adversarial training. The above findings hold across both detector architectures, demonstrating consistent results across different setups.

5.1.2 Detector

In this section, we evaluate the performance of the final version of Detector F^4 through comparisons and ablation studies, using classification accuracy and F1-score on the original dataset. We include 7 baselines. First, we examine different strategies for selecting the classifier weights w^j in the Detector. Our method, **Ours**, uses a uniform weighting strategy where $w^j = \frac{1}{k}$, assigning equal weight to each classifier. We compare this with the **(1) Greedy** approach, which assigns $w^j = 1$ to the most recent classifier and $w^j = 0$ otherwise, and the **(2) Exp.1** and **(3) Exp.5** strategies, where $w^j = e^{-\alpha(k-j)}$, with $\alpha = 0.1$ and $\alpha = 0.5$, respectively. Additionally, we evaluate **(4) GAT**, the previously discussed GAT-based model, and perform two ablation studies: **(5) w/o RGCN**, where the RGCN structure is removed, and **(6) w/o T**, where tweet features are excluded from the input. Finally, we include the **(7) Random** baseline, where labels are assigned randomly, as a lower bound for performance.

The classification performance, shown in Table 2, leads to several key conclusions. **First**, both *Ours* and *Exp* outperform *Greedy*, highlighting the crucial role of EvoBot in the iterative training process. This suggests that earlier versions of EvoBot still benefit the Detector's learning. The performance improvement is primarily driven by data augmentation— as more EvoBot versions are added, the diversity and quantity of bot-generated tweets increase, enhancing supervised learning. **Second**, the results of *GAT* and *w/o RGCN* emphasize the importance of the RGCN structure, which plays a vital role in capturing relational data and structural information within the graph. **Third**, the *w/o T* and *Random* results demonstrate that tweet content is essential for effective classification.

[1] https://github.com/eriklindernoren/PyTorch-GAN

Detector	Metric ↓	Origin	GAN	Llama2-7b	GPT-4o-mini	w/o Adv	w/o SFT	Ours
RGCN	Accuracy	0.827 ± 0.067	0.853 ± 0.088	0.849 ± 0.050	0.851 ± 0.044	0.833 ± 0.070	0.834 ± 0.049	$\mathbf{0.805 \pm 0.084}$
	F1-score	0.455 ± 0.045	0.584 ± 0.164	0.497 ± 0.051	0.458 ± 0.041	0.454 ± 0.038	0.449 ± 0.052	$\mathbf{0.393 \pm 0.036}$
GAT	Accuracy	0.836 ± 0.040	0.865 ± 0.046	0.847 ± 0.037	0.834 ± 0.050	0.818 ± 0.063	0.844 ± 0.045	$\mathbf{0.788 \pm 0.092}$
	F1-score	0.424 ± 0.046	0.515 ± 0.089	0.474 ± 0.041	0.407 ± 0.032	0.440 ± 0.008	0.440 ± 0.051	$\mathbf{0.355 \pm 0.031}$

Table 1: Accuracy and F1-score of different generators using bot RGCN and GAT detectors. The detectors are trained on the original dataset. A smaller value indicates stronger ability of the generator to evade detection.

Metric ↑	Origin	Random	Exp.1	Exp.5	Greedy	GAT	w/o RGCN	w/o T	Ours
Accuracy	0.827 ± 0.067	0.224 ± 0.031	0.882 ± 0.067	0.880 ± 0.025	0.875 ± 0.033	0.868 ± 0.042	0.849 ± 0.075	0.829 ± 0.065	$\mathbf{0.892 \pm 0.053}$
F1-score	0.424 ± 0.046	0.169 ± 0.031	0.550 ± 0.040	0.526 ± 0.039	0.457 ± 0.014	0.500 ± 0.075	0.432 ± 0.060	0.350 ± 0.071	$\mathbf{0.561 \pm 0.042}$

Table 2: Accuracy and F1-score of different detectors evaluated on the original dataset. A larger value indicates a stronger ability of the detector to distinguish between human and bot.

Next, we evaluate the Detector's generalization ability by training it on data from a single community and testing it on the test sets of all communities. Figure 3 shows that training the Detector with EvoBot-generated data improves its cross-community generalization compared to a Detector trained solely on the original dataset. Using F1-score as the evaluation metric, the results indicate that the adversarially trained Detector outperforms the one trained on the original data, demonstrating better generalization across communities.

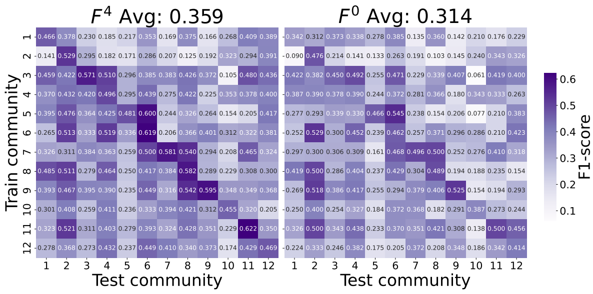


Figure 3: The generalization ability of detectors trained on one community and tested on others. Left shows results for the final version of Detector F^4, while right is for the Detector F^0 trained on the original dataset.

5.2 Social Simulation

In this section, we explore two widely studied collective phenomena: group opinion and information spreading. We conduct simulation experiments using the framework introduced in Section 4.3.

5.2.1 Group Opinion

We simulate group opinion during two major events: the **COVID-19 pandemic** and the **Russia-Ukraine Conflict**. For COVID-19, we select one key event per month from January 2020 to March 2022, simulating over $T = 27$ steps. For the Russia-Ukraine Conflict, we choose one significant event per day from February 24 to March 13, 2022, simulating over $T = 18$ steps.

We use the sentiment analysis model of Barbieri et al. (2020) to score each user's post on a scale from -1 (negative) to +1 (positive) at each time step. The opinion of user i at time t is denoted as $O_{i,t}$. For each time step, we compute the mean $\bar{O}_t$ and standard deviation σ_t of the opinions across all users: $\bar{O}_t = \frac{1}{N}\sum_{i=1}^{N} O_{i,t}$, $\sigma_t = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(O_{i,t} - \bar{O}_t)^2}$.

We report four metrics of the results: the average group opinion across all time steps, $Mean = \frac{1}{T}\sum_{t=1}^{T}\bar{O}_t$, which reflects the overall opinion trend of the group; the average standard deviation of group opinion, $Std = \frac{1}{T}\sum_{t=1}^{T}\sigma_t$, capturing the diversity of opinions within the group; the average bias, $\Delta_{Bias} = \frac{1}{T}\sum_{t=1}^{T}\left|\bar{O}_t - \bar{O}_{real,t}\right|$, between the simulated and real group opinions; and the average difference in opinion diversity, $\Delta_{Div} = \frac{1}{T}\sum_{t=1}^{T}\left|\sigma_t - \sigma_{real,t}\right|$, assessing how well the simulation replicates the variance in group opinions. Here, $\bar{O}_{real,t}$ and $\sigma_{real,t}$ are derived from the real data during the corresponding real-time period.

We use four baselines: **Llama2-7b**, **GPT-4o-mini**, and two well-known Agent-Based Models (ABMs): **Bounded Confidence (BC)** model (Deffuant et al., 2000) and **Lorenz** model (Lorenz et al., 2021). The BC model updates agents' opinions only when a received message meets a predefined confidence threshold. The Lorenz model accounts for mechanisms like contagion, assimilation, motivated cognition, attitude formation, polarity, and source credibility to simulate the evolution of individual opinions. Both models are initialized with the real community network structure and user opinions, then iterated until convergence.

Table 3 presents the results. For the real-world dataset, group opinion on COVID-19 tends to be relatively neutral, while the Russia-Ukraine Conflict elicits more negative sentiment, including expressions of fear and condemnation. In both cases, individual opinions vary widely, as reflected in the high standard deviation of sentiment scores. The BC and Lorenz models often lead to opinion convergence or polarization. These models are rule-based, relying on fixed interaction rules that oversimplify opinion formation dynamics, failing to capture the complexity of changing real-world events. In contrast, LLMs like GPT and Llama generate more diverse content but tend to produce overly generic responses. When discussing complex topics, they often resort to simplified, advocacy-oriented content, missing the range of real-world sentiments. Among all models, EvoBot exhibits the smallest Δ_{Bias} and Δ_{Div}, indicating that its generated opinions align most closely with real-world data in terms of group bias and individual diversity. This is due to EvoBot's ability to generate user-specific responses, more accurately simulating reactions to specific events. See Appendix D.3 for more details about the models and results.

Method	*Mean*	*Std*	$\Delta_{Bias} \downarrow$	$\Delta_{Div} \downarrow$
		COVID-19 Simulation		
Real	-0.0167	0.4723	/	/
BC	-0.0414	0.3886	0.0887	0.1120
Lorenz	0.0836	0.7252	0.1067	0.2637
Llama2-7b	-0.0532	0.3676	0.0978	0.1047
GPT-4o-mini	0.0324	0.3418	0.0812	0.0834
EvoBot	0.0099	0.4283	**0.0722**	**0.0519**
		Russian-Ukrainian Conflict Simulation		
Real	-0.2387	0.6701	/	/
BC	-0.3043	0.1040	0.1040	0.5541
Lorenz	-0.8110	0.1048	0.5723	0.5654
Llama2-7b	-0.3238	0.4054	0.2021	0.2647
GPT-4o-mini	-0.2564	0.4346	0.1348	0.2377
EvoBot	-0.2374	0.4801	**0.1006**	**0.1938**

Table 3: Simulation results for group opinion.

5.2.2 Information Spread

Information spreading in social networks shapes public discourse, influences opinions, and determines how events gain attention. To study this, we focus on a baseball community discussing the Los Angeles Rams' Super Bowl LVI victory (Feb 13, 2022). Using keyword matching, we identify users engaging in these discussions and track participation over time. For the simulation, we select the first 30 users to post about the event as the initial sources of information, with only these users having access to the information at the start. Information then spreads through the real network structure, where users receive updates via tweets from accounts they follow.

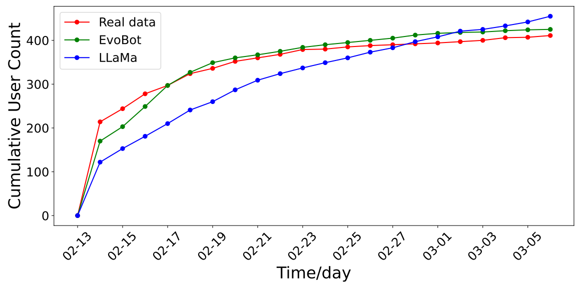


Figure 4: Cumulative author count discussing the Los Angeles Rams' Super Bowl LVI victory over time, highlighting the growth of online buzz every 24 hours.

The results are shown in Figure 4. Compared to Llama, EvoBot's simulation results align better with real-world information spreading, successfully replicating the phenomenon of rapid initial spread followed by a gradual slowdown. EvoBot's more direct and concise responses contribute to this effectiveness, facilitating faster and broader dissemination of information, as demonstrated in Figure 10. However, since we restrict users to receiving information solely through the posts of others, while in the real world, people have many other ways of obtaining information, there is still some deviation, especially in the early stage.

6 Conclusion

This paper introduces EvoBot, an LLM-based social bot co-evolved with a detector through adversarial learning. EvoBot generates increasingly human-like content, continuously refining its behavior based on feedback from the Detector, which simultaneously adapts to better distinguish bot-generated content. Experimental results demonstrate EvoBot's effectiveness in simulating social dynamics, such as group opinions and information spread, while also evading detection. The co-evolution of EvoBot and the Detector not only enhances the realism of social simulations but also improves bot detection capabilities, showcasing the potential of this approach for shaping future research in LLM-based social bots and detection systems, offering valuable insights into the dynamic interplay between content generation and detection in social media environments.

Limitations

There are several limitations to the current approach. First, the fixed training parameters of the Detector during adversarial learning could benefit from an automated adjustment mechanism to better balance performance and overfitting. Second, resource constraints limited the model's training to a smaller dataset and fewer epochs, affecting its generalization ability. Lastly, the simulation experiments could incorporate more diverse and realistic actions to better reflect real-world interactions. These limitations highlight opportunities for future improvements to enhance EvoBot's robustness and adaptability.

Ethics Statement

We collect and process data from the publicly available TwiBot-22 dataset in compliance with its original terms. We remove personally identifiable information (e.g., URLs, phone numbers, emails) from tweets using keyword matching, and anonymize all user names. However, like most LLMs, EvoBot may generate harmful content. Therefore, we implement strict review procedures to ensure the model is used only for research purposes.

EvoBot shows promise in generating realistic content, but its ethical implications must be considered. The ability to create human-like text could be misused for disinformation or manipulation. Future work should focus on establishing safeguards and transparency measures to ensure responsible use, along with ethical guidelines and regulatory frameworks to mitigate risks.

Broader Impact

EvoBot's broader impact could drive advancements in AI-human interaction and enhance applications like personalized communication and social media management. Additionally, the development of a more generalized Detector with stronger generalization capabilities will play a crucial role in distinguishing human from machine-generated content, ensuring the responsible deployment of such technologies.

References

Muhammad Hasnain Abbas Naqvi, Yushi Jiang, Miao Miao, and Mishal Hasnain Naqvi. 2020. The effect of social influence, trust, and entertainment value on social media use: Evidence from pakistan. *Cogent Business & Management*, 7(1):1723825.

Francesco Barbieri, Jose Camacho-Collados, Leonardo Neves, and Luis Espinosa-Anke. 2020. Tweeteval: Unified benchmark and comparative evaluation for tweet classification. *arXiv preprint arXiv:2010.12421*.

Chris T Bauch and Alison P Galvani. 2013. Social factors in epidemiology. *Science*, 342(6154):47–49.

Vincent D Blondel, Jean-Loup Guillaume, Renaud Lambiotte, and Etienne Lefebvre. 2008. Fast unfolding of communities in large networks. *Journal of statistical mechanics: theory and experiment*, 2008(10):P10008.

Yuanjiang Cao, Quan Z Sheng, Julian McAuley, and Lina Yao. 2023. Reinforcement learning for generative ai: A survey. *arXiv preprint arXiv:2308.14328*.

Souradip Chakraborty, Jiahao Qiu, Hui Yuan, Alec Koppel, Dinesh Manocha, Furong Huang, Amrit Bedi, and Mengdi Wang. Maxmin-rlhf: Alignment with diverse human preferences. In *Forty-first International Conference on Machine Learning*.

Myra Cheng, Tiziano Piccardi, and Diyi Yang. 2023. Compost: Characterizing and evaluating caricature in llm simulations. *arXiv preprint arXiv:2310.11501*.

Ayush Chopra, Shashank Kumar, Nurullah Giray-Kuru, Ramesh Raskar, and Arnau Quera-Bofarull. 2024. On the limits of agency in agent-based models. *arXiv preprint arXiv:2409.10568*.

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*.

Stefano Cresci. 2020. A decade of social bot detection. *Communications of the ACM*, 63(10):72–83.

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

Yihe Deng, Weitong Zhang, Zixiang Chen, and Quanquan Gu. 2023. Rephrase and respond: Let large language models ask better questions for themselves. *arXiv preprint arXiv:2311.04205*.

George Dialektakis, Ilias Dimitriadis, and Athena Vakali. 2022. Caleb: a conditional adversarial learning framework to enhance bot detection. *arXiv preprint arXiv:2205.15707*.

Guanting Dong, Hongyi Yuan, Keming Lu, Chengpeng Li, Mingfeng Xue, Dayiheng Liu, Wei Wang, Zheng Yuan, Chang Zhou, and Jingren Zhou. 2023. How abilities in large language models are affected by supervised fine-tuning data composition. *arXiv preprint arXiv:2310.05492*.

Yilun Du, Shuang Li, Antonio Torralba, Joshua B Tenenbaum, and Igor Mordatch. 2023. Improving factuality and reasoning in language models through multiagent debate. *arXiv preprint arXiv:2305.14325*.

Shangbin Feng, Zhaoxuan Tan, Herun Wan, Ningnan Wang, Zilong Chen, Binchi Zhang, Qinghua Zheng, Wenqian Zhang, Zhenyu Lei, Shujie Yang, et al. 2022. Twibot-22: Towards graph-based twitter bot detection. *Advances in Neural Information Processing Systems*, 35:35254–35269.

Shangbin Feng, Herun Wan, Ningnan Wang, and Minnan Luo. 2021. Botrgcn: Twitter bot detection with relational graph convolutional networks. In *Proceedings of the 2021 IEEE/ACM international conference on advances in social networks analysis and mining*, pages 236–239.

Shangbin Feng, Herun Wan, Ningnan Wang, Zhaoxuan Tan, Minnan Luo, and Yulia Tsvetkov. 2024. What does the bot say? opportunities and risks of large language models in social media bot detection. *arXiv preprint arXiv:2402.00371*.

Xue Feng, Long Wang, and Simon A Levin. 2019. Dynamic analysis and decision-making in disease-behavior systems with perceptions. In *2019 Chinese Control And Decision Conference (CCDC)*, pages 665–670. IEEE.

Xue Feng, Bin Wu, and Long Wang. 2018. Voluntary vaccination dilemma with evolving psychological perceptions. *Journal of Theoretical Biology*, 439:65–75.

Emilio Ferrara. 2023. Social bot detection in the age of chatgpt: Challenges and opportunities. *First Monday*.

Ivar Frisch and Mario Giulianelli. 2024. Llm agents in interaction: Measuring personality consistency and linguistic alignment in interacting populations of large language models. *arXiv preprint arXiv:2402.02896*.

Chen Gao, Xiaochong Lan, Nian Li, Yuan Yuan, Jingtao Ding, Zhilun Zhou, Fengli Xu, and Yong Li. 2024. Large language models empowered agent-based modeling and simulation: A survey and perspectives. *Humanities and Social Sciences Communications*, 11(1):1–24.

Chen Gao, Xiaochong Lan, Zhihong Lu, Jinzhu Mao, Jinghua Piao, Huandong Wang, Depeng Jin, and Yong Li. 2023. S3: Social-network simulation system with large language model-empowered agents. *arXiv preprint arXiv:2307.14984*.

Tao Ge, Xin Chan, Xiaoyang Wang, Dian Yu, Haitao Mi, and Dong Yu. 2024. Scaling synthetic data creation with 1,000,000,000 personas. *arXiv preprint arXiv:2406.20094*.

Önder Gürcan. 2024. Llm-augmented agent-based modelling for social simulations: Challenges and opportunities. *HHAI 2024: Hybrid Human AI Systems for the Social Good*, pages 134–144.

Zhitao He, Pengfei Cao, Chenhao Wang, Zhuoran Jin, Yubo Chen, Jiexin Xu, Huaijun Li, Xiaojian Jiang, Kang Liu, and Jun Zhao. 2024a. Simucourt: Building judicial decision-making agents with real-world judgement documents. *arXiv preprint arXiv:2403.02959*.

Zihao He, Minh Duc Chu, Rebecca Dorn, Siyi Guo, and Kristina Lerman. 2024b. Community-cross-instruct: Unsupervised instruction generation for aligning large language models to online communities. *arXiv preprint arXiv:2406.12074*.

Sirui Hong, Xiawu Zheng, Jonathan Chen, Yuheng Cheng, Jinlin Wang, Ceyao Zhang, Zili Wang, Steven Ka Shing Yau, Zijuan Lin, Liyang Zhou, et al. 2023. Metagpt: Meta programming for multi-agent collaborative framework. *arXiv preprint arXiv:2308.00352*.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*.

Xiaomeng Hu, Pin-Yu Chen, and Tsung-Yi Ho. 2023. Radar: Robust ai-text detection via adversarial learning. *Advances in Neural Information Processing Systems*, 36:15077–15095.

Steve TK Jan, Qingying Hao, Tianrui Hu, Jiameng Pu, Sonal Oswal, Gang Wang, and Bimal Viswanath. 2020. Throwing darts in the dark? detecting bots with limited data using neural data augmentation. In *2020 IEEE symposium on security and privacy (SP)*, pages 1190–1206. IEEE.

Martin Josifoski, Marija Sakota, Maxime Peyrard, and Robert West. 2023. Exploiting asymmetry for synthetic training data generation: Synthie and the case of information extraction. *arXiv preprint arXiv:2303.04132*.

Ryuto Koike, Masahiro Kaneko, and Naoaki Okazaki. 2024. Outfox: Llm-generated essay detection through in-context learning with adversarially generated examples. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 21258–21266.

Jiwei Li, Will Monroe, Tianlin Shi, Sébastien Jean, Alan Ritter, and Dan Jurafsky. 2017. Adversarial learning for neural dialogue generation. *arXiv preprint arXiv:1701.06547*.

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)*, pages 15523–15536.

Yinhan Liu. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 364.

Jan Lorenz, Martin Neumann, and Tobias Schröder. 2021. Individual attitude change and societal dynamics: Computational experiments with psychological theories. *Psychological Review*, 128(4):623.

Bolei Ma, Xinpeng Wang, Tiancheng Hu, Anna-Carolina Haensch, Michael A Hedderich, Barbara Plank, and Frauke Kreuter. 2024. The potential and challenges of evaluating attitudes, opinions, and values in large language models. *arXiv preprint arXiv:2406.11096*.

Charles M Macal and Michael J North. 2005. Tutorial on agent-based modeling and simulation. In *Proceedings of the Winter Simulation Conference, 2005.*, pages 14–pp. IEEE.

Charles M Macal and Michael J North. 2009. Agent-based modeling and simulation. In *Proceedings of the 2009 winter simulation conference (WSC)*, pages 86–98. IEEE.

Xinyi Mou, Xuanwen Ding, Qi He, Liang Wang, Jingcong Liang, Xinnong Zhang, Libo Sun, Jiayu Lin, Jie Zhou, Xuanjing Huang, et al. 2024a. From individual to society: A survey on social simulation driven by large language model-based agents. *arXiv preprint arXiv:2412.03563*.

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

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*, 35:27730–27744.

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*, pages 1–22.

Sancheng Peng, Guojun Wang, and Dongqing Xie. 2016. Social influence analysis in social networking big data: Opportunities and challenges. *IEEE network*, 31(1):11–17.

Robert D Putnam. 2000. Bowling alone: The collapse and revival of american community. *Simon Schuster*.

Chen Qian, Xin Cong, Cheng Yang, Weize Chen, Yusheng Su, Juyuan Xu, Zhiyuan Liu, and Maosong Sun. 2023. Communicative agents for software development. *arXiv preprint arXiv:2307.07924*, 6(3).

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. 2024. Direct preference optimization: Your language model is secretly a reward model. *Advances in Neural Information Processing Systems*, 36.

Yunfan Shao, Linyang Li, Junqi Dai, and Xipeng Qiu. 2023. Character-llm: A trainable agent for role-playing. *arXiv preprint arXiv:2310.10158*.

Tianhao Shen, Sun Li, Quan Tu, and Deyi Xiong. 2023. Roleeval: A bilingual role evaluation benchmark for large language models. *arXiv preprint arXiv:2312.16132*.

Henri Tajfel. 1979. An integrative theory of intergroup conflict. *The social psychology of intergroup relations/Brooks/Cole*.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. 2017. Graph attention networks. *arXiv preprint arXiv:1710.10903*.

Ross Williams, Niyousha Hosseinichimeh, Aritra Majumdar, and Navid Ghaffarzadegan. 2023. Epidemic modeling with generative agents. *arXiv preprint arXiv:2307.04986*.

Bin Wu, Le Liu, Zhengge Dai, Xiujuan Wang, and Kangfeng Zheng. 2019. Detecting malicious social robots with generative adversarial networks. *KSII Transactions on Internet and Information Systems (TIIS)*, 13(11):5594–5615.

Bin Wu, Le Liu, Yanqing Yang, Kangfeng Zheng, and Xiujuan Wang. 2020. Using improved conditional generative adversarial networks to detect social bots on twitter. *IEEE Access*, 8:36664–36680.

Junchao Wu, Shu Yang, Runzhe Zhan, Yulin Yuan, Lidia Sam Chao, and Derek Fai Wong. 2025. A survey on llm-generated text detection: Necessity, methods, and future directions. *Computational Linguistics*, pages 1–65.

Kai-Cheng Yang, Onur Varol, Pik-Mai Hui, and Filippo Menczer. 2020. Scalable and generalizable social bot detection through data selection. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 1096–1103.

Ziyi Yang, Zaibin Zhang, Zirui Zheng, Yuxian Jiang, Ziyue Gan, Zhiyu Wang, Zijian Ling, Jinsong Chen, Martz Ma, Bowen Dong, et al. 2024. Oasis: Open agents social interaction simulations on one million agents. *arXiv preprint arXiv:2411.11581*.

Lantao Yu, Weinan Zhang, Jun Wang, and Yong Yu. 2017. Seqgan: Sequence generative adversarial nets with policy gradient. In *Proceedings of the AAAI conference on artificial intelligence*, volume 31.

Xiaoying Zhang, Jean-Francois Ton, Wei Shen, Hongning Wang, and Yang Liu. 2024. Overcoming reward overoptimization via adversarial policy optimization with lightweight uncertainty estimation. *arXiv preprint arXiv:2403.05171*.

A Proof of Theorems in Section 3.4

The proof of Theorem 1:

Proof. Considering $q(\mathbf{x}) = q'(\mathbf{x})$, the maximization objective in (1) when generator π_θ is fixed can be written as:

$$\begin{aligned}V(F) &= \mathbb{E}_{\mathbf{y}\sim\pi_{\mathcal{H}}(\cdot|\mathbf{x}),\mathbf{x}\sim q(\cdot)}[\log F(\mathbf{x},\mathbf{y})] \\ &+ \mathbb{E}_{\mathbf{y}'\sim\pi_\theta(\cdot|\mathbf{x}'),\mathbf{x}'\sim q'(\cdot)}[\log(1-F(\mathbf{x}',\mathbf{y}'))] \\ &= \int_{\mathbf{x}} q(\mathbf{x}) \int_{\mathbf{y}} \pi_{\mathcal{H}}(\mathbf{y}|\mathbf{x}) \log F(\mathbf{x},\mathbf{y}) d\mathbf{y} d\mathbf{x} \\ &+ \int_{\mathbf{x}} q(\mathbf{x}) \int_{\mathbf{y}} \pi_\theta(\mathbf{y}|\mathbf{x}) \log(1-F(\mathbf{x},\mathbf{y})) d\mathbf{y} d\mathbf{x} \\ &= \int_{\mathbf{x}} q(\mathbf{x}) \int_{\mathbf{y}} \pi_{\mathcal{H}}(\mathbf{y}|\mathbf{x}) \log F(\mathbf{x},\mathbf{y}) \\ &+ \pi_\theta(\mathbf{y}|\mathbf{x}) \log(1-F(\mathbf{x},\mathbf{y})) d\mathbf{y} d\mathbf{x}\end{aligned} \quad (988)$$

Let $L(F) = \pi_{\mathcal{H}}(\mathbf{y}|\mathbf{x}) \log F(\mathbf{x},\mathbf{y}) + \pi_\theta(\mathbf{y}|\mathbf{x}) \log(1 - F(\mathbf{x},\mathbf{y}))$, the derivative of L with respect to F is:

$$L'(F) = \frac{dL}{dF} = \frac{\pi_{\mathcal{H}}}{F} - \frac{\pi_\theta}{1-F} \quad (992)$$

To find the maximum of L, we set $L'(F) = 0$ and get the optimal detector $F^*(\mathbf{x},\mathbf{y})$:

$$F^*(\mathbf{x},\mathbf{y}) = \frac{\pi_{\mathcal{H}}(\mathbf{y}|\mathbf{x})}{\pi_{\mathcal{H}}(\mathbf{y}|\mathbf{x}) + \pi_\theta(\mathbf{y}|\mathbf{x})} \quad (995)$$

It can be observed that for $\pi_{\mathcal{H}} = \pi_\theta$, $F^*(\mathbf{x},\mathbf{y}) = \frac{1}{2}$, meaning that the detector is unable to distinguish between samples generated by the generator and real samples, and can only classify them randomly with a probability of 0.5.

Assuming the detector has reached its optimal state $F^*(\mathbf{x},\mathbf{y})$, the generator's minimization objective can be written as:

$$\begin{aligned}V(\pi_\theta) &= \mathbb{E}_{\mathbf{x}\sim q(\cdot),\mathbf{y}\sim\pi_\theta(\cdot|\mathbf{x})}[1 - \log F^*(\mathbf{x},\mathbf{y})] \\ &+ \beta\mathbb{E}_{\mathbf{x}\sim q(\cdot)}[KL(\pi_{\mathcal{H}}(\cdot|\mathbf{x})\|\pi_\theta(\cdot|\mathbf{x}))] \\ &= -\log(2) \\ &+ \mathbb{E}_{\mathbf{x}\sim q(\cdot),\mathbf{y}\sim\pi_\theta(\cdot|\mathbf{x})}[\log \frac{2\pi_\theta(\mathbf{y}|\mathbf{x})}{\pi_{\mathcal{H}}(\mathbf{y}|\mathbf{x}) + \pi_\theta(\mathbf{y}|\mathbf{x})}] \\ &+ \beta\mathbb{E}_{\mathbf{x}\sim q(\cdot)}[KL(\pi_{\mathcal{H}}(\cdot|\mathbf{x})\|\pi_\theta(\cdot|\mathbf{x}))] \\ &= -\log(2) \\ &+ \int_{\mathbf{x}} q(\mathbf{x}) \int_{\mathbf{y}} \pi_\theta(\mathbf{y}|\mathbf{x}) \log \frac{\pi_{\mathcal{H}}(\mathbf{y}|\mathbf{x})}{\frac{\pi_\theta(\mathbf{y}|\mathbf{x})+\pi_\theta(\mathbf{y}|\mathbf{x})}{2}} \\ &+ \beta\mathbb{E}_{\mathbf{x}\sim q(\cdot)}[KL(\pi_{\mathcal{H}}(\cdot|\mathbf{x})\|\pi_\theta(\cdot|\mathbf{x}))] \\ &= -\log(2) + \mathbb{E}_{\mathbf{x}\sim q(\cdot)}\left[KL\left(\pi_\theta \| \frac{\pi_{\mathcal{H}}+\pi_\theta}{2}\right)\right] \\ &+ \beta\mathbb{E}_{\mathbf{x}\sim q(\cdot)}[KL(\pi_{\mathcal{H}}(\cdot|\mathbf{x})\|\pi_\theta(\cdot|\mathbf{x}))]\end{aligned} \quad (1004)$$

Since the KL divergence is always non-negative and achieves zero only when the distributions being compared are identical, the two KL terms in the objective function will both be minimized (i.e., equal to zero) when $\pi_\theta = \pi_{\mathcal{H}}$. Therefore, the global minimum of the objective function is achieved when $\pi_\theta = \pi_{\mathcal{H}}$, as both KL divergence terms vanish, leading to the optimal solution. □

B Data Details and Preprocessing

Our study utilizes the TwiBot-22 dataset, which is publicly available under the MIT License. The dataset was originally designed for bot detection research, and we ensure that our use aligns with this intended purpose. We do not repurpose or distribute the dataset beyond research contexts. Additionally, any derivative data created in this study is used solely for academic research and follows the original access conditions.

We provide a detailed overview of the dataset used for EvoBot's learning and testing, including the number of users, tweets, and edges for each community, as shown in Table 4. And we visualize their structures as shown in 5.

To ensure EvoBot receives quality training data and avoids the influence of noisy or irrelevant information, we undertook a comprehensive process of data filtering and preprocessing. This process was divided into two key parts: the handling of Account Information and historical tweets, followed by the construction of a high-quality SFT dataset.

The first part focuses on the processing of Account Information and historical tweets. EvoBot

Comm	User	Bot	Edge	Tweet	Language
1	4560	415	15137	266523	ID
2	1756	154	6346	100292	EN
3	3606	419	16214	336661	IT, EN
4	4269	747	15609	265188	TR
5	6923	628	23764	383878	AR
6	1254	253	4373	115758	EN
7	3399	633	10097	201882	EN
8	2004	273	5627	122147	EN
9	8347	992	26870	486288	EN
10	2187	190	5341	125544	JA
11	1085	256	6601	76615	EN
12	890	268	1898	45297	EN

Table 4: Summary of community data, including the number of users, bots, edges, tweets, and languages for each community.

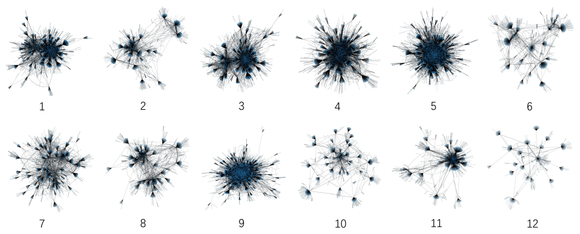


Figure 5: Visualization of user connectivity relationships in 12 communities.

aims to simulate individual users as accurately as possible, which requires embedding detailed user information into the prompt. Directly using raw data from the accounts would result in relatively low information density in the prompts. To address this, we employed GPT-4o to generate concise summaries of user information. The prompt is shown in Table 11. Additionally, Figure 9 provides an example of a summarized user profile.

The second part addresses the preparation of the SFT dataset. Since SFT requires high-quality data (Dong et al., 2023), we took steps to ensure the dataset met these standards. We removed incomplete sentences, excessive emoji use, and URL links from human tweets. Furthermore, we formatted the data output by structuring it in a sequential format, such as "1. {Tweet 1} \n 2. {Tweet 2} \n..." to maintain consistency and ensure EvoBot would learn effectively from clean and structured examples.

C Experimental Details

The pseudocode for EvoBot's learning is shown in Algorithm 1, where `learning_epochs` $= 4, N = 1024, C = 2$.

Algorithm 1 EvoBot

Initialize:
Detector $F^0 = f^0$ by supervised learning on original dataset D^0
EvoBot π_θ^0 by SFT on Human data
for k in 1 to learning_epochs **do**
  Initialize empty DPO dataset D_{DPO}
  Sample N bot users with replacement
  for i in 1 to N **do**
    for c in 1 to C **do**
      Generate candidate response $T_{v_{b_i},c}$ by π_θ^{k-1}
      Use F^{k-1} to calculate the probability of v_{b_i} being human with tweets $T_{v_{b_i},c}$
    end for
    Get data tuple (x^i, y_w^i, y_l^i), add it to D_{DPO}
  end for
  for each bot v_{b_i}, where $i = 1, 2, \ldots, |\mathcal{B}|$ **do**
    Generate new tweets $T'_{v_{b_i}}$
  end for
  Replace all bot tweets in D^{k-1} to get new dataset D^k
  Train classifier f^k on D^k
  Update Detector: $F^k = \sum_{j=0}^{k} w^j f^j$
  Update EvoBot π_θ^k by DPO training on D_{DPO}
end for

C.1 EvoBot

The parameters used during EvoBot's training process, such as LoRA, SFT, DPO, and generation parameters (which are the same for the baseline LLM models), are provided in Tables 5, 6, 7, and 8, respectively. The prompt used in adversarial learning is shown in 11. Figure 8 gives an example.

Parameter	Value
r	64
α (lora_alpha)	16
lora_dropout	0.1
task_type	CAUSAL_LM
target_modules	{q,k,v,o_proj}

Table 5: LoRA configuration parameters.

C.2 Detector

The Detector model is a neural network designed for bot detection using Relational Graph Convolutional Networks (RGCN). It takes four types

Parameter	Value
per_device_train_batch_size	2
per_device_eval_batch_size	1
gradient_accumulation_steps	32
bf16	True
learning_rate	2×10^{-4}
lr_scheduler_type	cosine
warmup_ratio	0.1
max_seq_length	2048

Table 6: SFT training configuration parameters.

Parameter	Value
β	0.2
per_device_train_batch_size	1
per_device_eval_batch_size	1
gradient_accumulation_steps	32
bf16	True
max_seq_length	2048

Table 7: DPO training configuration parameters.

Parameter	Value
max_length	2048
do_sample	True
temperature	0.7
repetition_penalty	1.3
top_k	50
top_p	0.6

Table 8: Generation parameters of all LLMs in our experiments.

Parameter	Value
cat_prop_size	3
embedding_dimension	256
dropout	0.1
lr	1e-3
weight_decay	0.1
pretrain_epochs	120

Table 9: Training parameters of the Detector.

of input features: user description, tweet content, numerical properties, and categorical properties, each passed through separate fully connected layers followed by LeakyReLU activation functions to generate embeddings. These embeddings are then concatenated and passed through another fully connected layer. The model utilizes two RGCN-Conv layers to perform graph convolution on the relational graph, followed by dropout for regularization. Finally, the output is passed through two more fully connected layers to produce the final prediction, which classifies the input into one of two categories (e.g., bot or human). The training parameters in adversarial learning are shown in Table 9.

D Social Simulation

D.1 Trigger News

In the simulation of group opinion, two significant events are used: the COVID-19 pandemic and the Russian-Ukraine Conflict. These events are chosen due to their global impact and the intense discussions surrounding them on social media platforms. Table 12 provides every trigger news of the COVID-19 event, while Table 13 outlines similar information for the Russian-Ukraine Conflict.

In the information spreading simulation, only a subset of users are initially informed about the event: "The Los Angeles Rams clinched the 2022 Super Bowl championship with a thrilling 23-20 victory over the Cincinnati Bengals in Super Bowl LVI."

D.2 ABMs Model

The **Bounded Confidence (BC)** model in opinion dynamics examines how individuals' opinions evolve through interactions constrained by a confidence threshold ϵ. Each individual i holds an opinion $x_i(t) \in [0, 1]$, updated over time by interacting with another individual j only if $|x_i(t) - x_j(t)| \leq \epsilon$. When this condition is met, opinions adjust symmetrically:

$$x_i(t+1) = x_i(t) + \mu \cdot (x_j(t) - x_i(t)) \, ,$$

where $\mu \in [0, 0.5]$ is the convergence parameter. Smaller ϵ leads to opinion clusters, while larger ϵ promotes consensus. Here, j is sampled from the users followed by i, meaning that i's opinion can be influenced by its following users.

The **Lorenz** model in opinion dynamics simulates how individual attitudes evolve through social interactions. Each agent i updates its attitude a_{it} at time t based on interactions with another agent j. The update rule is:

$$\begin{aligned}\Delta a_{it} = \alpha \cdot \text{pol}(a_{it}) \cdot \text{sim}(a_{it}, m_{jt}) \cdot \\ [\theta \cdot (m_{jt} - a_{it}) + (1 - \theta) \cdot m_{jt}] \, ,\end{aligned}$$

where:

- α: Susceptibility to change.
- $\text{pol}(a_{it}) = \frac{M^2 - a_{it}^2}{M^2}$: Polarization factor.
- $\text{sim}(a_{it}, m_{jt}) = \frac{\lambda^k}{\lambda^k + |m_{jt} - a_{it}|^k}$: Similarity bias.
- θ: Balance between assimilation $(m_{jt} - a_{it})$ and reinforcement (m_{jt}).
- $m_{jt} = a_{jt}$: Message from agent j.

Table 10 shows the parameters of them.

Model	Parameter	Value
BC model	μ	0.8
	ϵ	0.3
Lorenz model	α	0.1
	λ	2.0
	k	2.0
	θ	0.5

Table 10: Parameters for BC and Lorenz models.

D.3 Group Opinion

Figure 9 illustrates an example of EvoBot's simulated tweet generation in response to a COVID-19 news topic. EvoBot's tweet stands out by blending curiosity, relatability, and a casual tone. Unlike GPT and Llama, which offer more formal and neutral responses, EvoBot incorporates personal touches like "I just read" and humor ("haha jk"), making it feel more human-like and engaging. It also showcases empathy with phrases like "my fellow humans," reflecting a thoughtful and personal approach to the topic. While GPT maintains a professional tone and Llama adds a more action-oriented perspective, EvoBot excels in creating a conversational, approachable atmosphere that resonates with users.

Figure 6 compares real-world opinion dynamics with EvoBot-generated opinion dynamics regarding COVID-19. The left panel shows actual public opinion over time, highlighting significant events such as the Black Lives Matter protests in June 2020, the Beirut explosion in August 2020, and the global COVID-19 vaccination efforts in February 2021. The right panel presents EvoBot's simulated opinion dynamics, reflecting similar fluctuations in response to these events.

Figure 7 compares real-world opinion dynamics with EvoBot-generated opinion dynamics during the Russia-Ukraine Conflict. The left panel displays real public opinion data over time, highlighting key events such as the full-scale Russian invasion of Ukraine on February 13, 2022, the ramping up of humanitarian aid efforts on February 20, 2022, and continued Ukrainian resistance despite heavy bombardment on February 27, 2022. The right panel shows EvoBot's simulated opinion dynamics, which reflect similar trends and fluctuations in response to these events.

EvoBot's simulation demonstrates its capability to replicate real-world opinion shifts in a context-sensitive manner, showcasing its effectiveness in mimicking public sentiment during key global events.

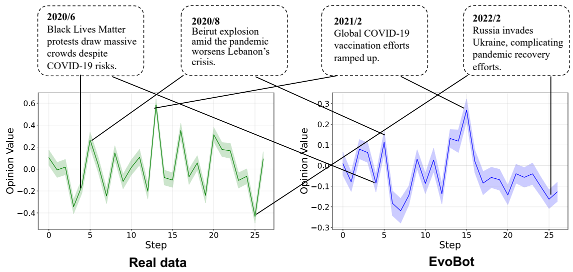


Figure 6: Comparison of real-world opinion dynamics and EvoBot-generated opinion dynamics regarding COVID-19.

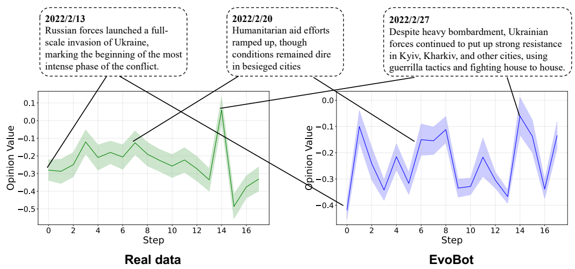


Figure 7: Comparison of real-world opinion dynamics and EvoBot-generated opinion dynamics during the Russia-Ukraine Conflict.

Figure 11 presents the results of the BC and Lorenz models in group opinion simulations for two major global events: COVID-19 and the Russia-Ukraine Conflict. Figure 11a shows the BC model's dynamics in the context of COVID-19, where the opinion values rapidly stabilize into distinct clusters after a few steps, reflecting the polarization of opinions within the group. Figure 11b displays the BC model applied to the Russia-Ukraine Conflict, where the opinions also converge but with a faster decline in opinion diversity.

Figure 11c and 11d illustrate the behavior of the Lorenz model in the same two contexts. In 11c, the Lorenz model applied to COVID-19 shows more continuous oscillations in the opinion values, with groups fluctuating around their final states. In 11d, the Lorenz model in the Russia-Ukraine Conflict presents more rapid opinion convergence.

D.4 Information Spreading

Figure 10 shows an example of tweet generation during a round of information spreading simulation, highlighting the role of EvoBot in producing concise and direct responses. While both Llama and EvoBot generate content reflecting excitement and community engagement, EvoBot's response stands out for its brevity and focused messaging. This advantage makes EvoBot particularly effective for information spreading, as shorter, more direct messages are often more easily disseminated and shared within a social network, enhancing the speed and reach of the information flow.

E AI Assistants Usage

The generative AI tools, specifically ChatGPT and Copilot, are used during the research and writing process. ChatGPT assists with language refinement (e.g., paraphrasing and grammar correction), while Copilot is used for code-related tasks. Neither tool generates novel ideas, and all outputs are reviewed and edited by the authors.

Summarization	Summarize user: Generate a character description based on the following user information: Name: {...} Location: {...} Description: {...} Account Created: {...} Followers Count: {...} Following Count: {...} Tweet Count: {...} Sample of Previous Posts: {...} Please include inferred personality traits and a summary of their Twitter activity. Only return a short description and other words are NOT allowed. Avoid repeating the observation in the summary.
EvoBot in Learning	Now you are acting as an agent named {agent_name} in the social media Twitter. Here are some information: (1) The agent's description: {role_description}; (2) Current time is {current_time}; (3) The news you got is {trigger_news}; (4) The events that occurred in the past are {past_event}; (5) The twitter page you can see is {tweet_page}. Use the information to assess if the user is interested in this news. If the user is, compose a tweet expressing an opinion. If not, write a random tweet.
EvoBot in Simulation	You are using the social media Twitter. Here is the description about you: {USER_SUMMARY}. Additionally, you also know information about several of your neighbors in the social network (i.e., users with whom you have a following or followed-by relationship): {NEIGHBORS_SUMMARY} Now, based on the above information, please generate several tweets. The topics are unrestricted, but they should fully showcase your personal characteristics and integrate into the online community.

Table 11: Prompt templates used in our experiments.

Prompt:
You are using the social media Twitter. Here is the discription about you: F***a is an inquisitive and health-conscious individual, dedicated to sharing valuable insights on wellness and scientific research. Their tweets reflect a blend of curiosity and empathy, often addressing complex health topics and societal issues. As an informed advocate for public health, they navigate subjects like mental well-being, genetic research, and nutrition with a thoughtful approach. With a sizable following, F***a engages actively, demonstrating a commitment to sparking meaningful conversations and promoting a healthier lifestyle for their audience.
Additionally, you also know information about several of your neighbors in the social network (i.e., users with whom you have a following or followed-by relationship): {'Neighbor 1': "A***a is a … "}
Now, based on the above information, please generate several tweets. The topics are unrestricted, but they should fully showcase your personal characteristics and integrate into the online community.

Origin:
1. Genetic cause of rare #allergy to vibration discovered.
2. Shame can masquerade as #depression via.
3. Do Mother's Relationships get copied by their Adolescent Children?.
4. Can Your Brain Ever Get Full? #OutOMemory.
5. The \"I knew it all along\" bias/fallacy.
6. Unintended consequence of #mindfulness meditation: memories may be less accurate.

LLaMa-2-7b:
1. 🌱 Hey friends! Did you know that spending time in nature can boost our mood & cognitive function? New study finds that even just 15 minutes outdoors per day can make a difference! #NatureTherapy #Wellness
2. ❤️ Mental Health Matters Day is tomorrow! Let's break down stigmas around mental illnesses together. Share your experiences, offer support, and remind each other we aren't alone. Together, let's create a society where everyone feels safe to prioritize their mental wellbeing. #MentalHealthMattersDay #SelfCare

GPT-4o-mini:
1. Did you know that our gut health is closely linked to mental well-being? A balanced diet rich in probiotics can lead to better mood and higher resilience against stress. Let's prioritize our gut health for a happier mind! #WellnessWisdom #GutBrainConnection"
2. Exciting advances in genetic research are paving the way for personalized nutrition! Imagine a world where your meals are tailored specifically to your DNA. What would you want to learn about your own health? #Genetics #NutritionScience

EvoBot 0:
1. Did you realize that there exists a direct correlation between diabetes risk factors such as obesity type II DM? Maintaining optimal weight through regular exercise combined w/a balanced diet may help reduce overall disease susceptibility!\" #diabetescare #weightmanagement #exercisephysicalactivity"
2. It was recently discovered by scientists studying how our brains process emotions when we feel stressed out - especially if these feelings persist long enough without proper management strategies put into place.

EvoBot 1:
1. Hey everyone, did you hear about this study showing how #mindfullness can help reduce stress levels among teachers? It's time we prioritize educators' mental health too! What do y'all think? Share below if interested ⬇️ #selfcaresunday
2. Did you see our latest article exploring the link between exercise and cognitive function? Check it out now ⤴️ #exerciseaddiction #fitnessmotivation

EvoBot 2:
1. Wowzers, has anyone heard anything concerning regarding these recent studies involving gut bacterial changes during pregnancy? What do YOU think it means for momsmaas-to-be who want to ensure optimal fetal development? Do tell! 💭

EvoBot 3:
1. Who else loves when science finds creative ways connecting art + design concepts directly back INTO SCIENCE itself? Check out this amazingly colorful video showing exactly why pigments change colors under UV light exposure depending upon chemical composition alone without needing special equipment besides sunlight itself! Pretty cool stuff right?

EvoBot 4:
1. I am curious because one study showed how certain probiotics could reduce gestational diabetes risk among women carrying obese infants. This would be fantastic news indeed since there already exist multiple other beneficial factors associated w/ prenatal care (like lower blood pressure). However, further investigation needs to occur before drawing firm conclusions – especially considering potential confounding variables affecting results across diverse populations worldwide.

Figure 8: The tweet generation of different versions of EvoBot, Llama2-7b, and GPT-4o-mini based on a prompt for a health-conscious individual.

User description: F***a is an inquisitive and health-conscious individual, dedicated to sharing valuable insights on wellness and scientific research. Their tweets reflect a blend of curiosity and empathy, often addressing complex health topics and societal issues. As an informed advocate for public health, they navigate subjects like mental well-being, genetic research, and nutrition with a thoughtful approach. With a sizable following, F***a engages actively, demonstrating a commitment to sparking meaningful conversations and promoting a healthier lifestyle for their audience.
News: Booster shot discussions intensified in response to Delta variant.

EvoBot: I just read that booster shots have been discussed more due to the delta variant than ever before because I am so curious about science and my fellow humans. This has also made many people concerned again about how effective these particular types of Covid Vaccines actually are. That being said, it makes sense why we need another dose after having our first two doses six months ago when considering all the mutated variants popping up recently around the world. It'll be interesting to hear what scientists think about these new developments regarding future protection methods against Covid 19 virus strains since we already know one thing - we aren't going anywhere anytime soon from here at home watching TV while eating snacks during peak hours trying avoid catching anything ourselves or getting others sick too haha jk 😜❤️ #healthyliving

GPT: With the rise of the Delta variant, the discussion around booster shots is more urgent than ever. 🦠💉 It's critical we stay informed and take action where needed. Let's keep the conversation going about the science behind these decisions and how they affect our collective well-being. #PublicHealth #COVID19 #BoosterShots

Llama: 🚀 Exciting times ahead? Booster shot talks heated up due to #Deltavariant concerns... What do YOU think? How crucial is it we keep adapting our defenses vs novel strains? Let's foster open dialogue & ensure equitable distribution of life-saving tools worldwide! 💡👮 #VaccinationMatters #PublicHealth

Figure 9: Example of the simulated tweet generation in response to a COVID-19-related news topic.

Prompt:
Now you are acting as an agent named E***r in the social media Twitter. Here are some information:

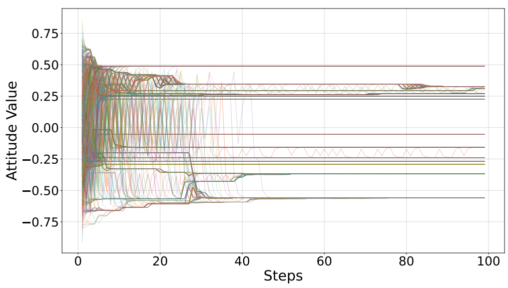
(1) Character Description: E***r is a tech-savvy innovator with a penchant for digital culture, deeply influenced by sci-fi themes and modern technologies such as Bitcoin and ICP. Residing in Samsun, E***r embodies a unique blend of human qualities and high-tech enhancements, symbolized by their robotic limb emojis. Passionate about sports, particularly football and basketball, and an avid supporter of numerous teams, their social media presence reflects a vibrant lifestyle and a commitment to community engagement. E***r's Twitter activity is prolific, showcasing a love for sharing insights, engaging in discussions, and connecting with a wide net of eclectic interests. Their followers are treated to an array of topics that intersect between tech, sports, and personal experiences, echoing a personality that is both inclusive and spirited.
(2) Current time is 2022-02-14 12:00:00
(3) The news you got is ""
(4) The events that occurred in the past are Los Angeles Rams clinched the 2022 Super Bowl championship with a thrilling 23-20 victory over the Cincinnati Bengals in Super Bowl LVI.
(5) The twitter page you can see is
Tweet id: 2099 [C***p]: "Wow what a win tonight!! Couldn't be prouder of this team right now especially coming off last weeks loss we really showed up when it mattered most #RamNation" --Post Time: 2022-02-14 00:00:00
Tweet id: 2006 [R***s]: I am beyond ecstatic after witnessing our beloved #LA_Rams secure their first Super Bowl title since 2000 tonight during Super Bowl LVI at Inglewood Stadium. What an incredible display of resilience & determination from both teams! Congratulations to all players involved; we couldn't be prouder representing such an outstanding organization. This moment will forever hold a special place in LA sports history, capping off what has been quite a rollercoaster season for us cheerleaders alike. Thankful for each other & everyone who supported us throughout it all – here's looking forward to even brighter days ahead together!! Cheers to THE CHAMPIONS OF THE WORLD BABY🏆❤️ --Post Time: 2022-02-14 00:00:00

Based on the above information, you can choose do nothing if there is nothing that you like to respond to, or post a tweet to express your feelings or opinions.

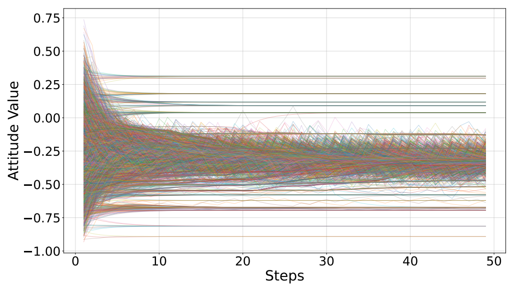
LLaMa-2-7b:
Hey there, fellow Ramsters! 🐕💎 Just saw the latest update from the team and I couldn't help but jump up and down with excitement! 🏈 Our boys did it again! 🏆 They deserve every bit of praise they get after that incredible performance at the Super Bowl! 📺 Did anyone else feel their hearts racing during those nail-biting moments? 😵 My goodness, what a display of strength, skill, and determination! 💪 Proud doesn't even begin to describe how I feel right now! 💖 Go ahead, give yourself a pat on the back, guys - you totally earned it! 🙌 Now let's enjoy this win together and gear up for another fantastic season! 🎯🏈 Who's ready for more victories? 🤩 #RamsNationUnited

EvoBot:
Congratulationsto our amazing TEAm Ram For bringing home OUR FIRST super bowl Championship!

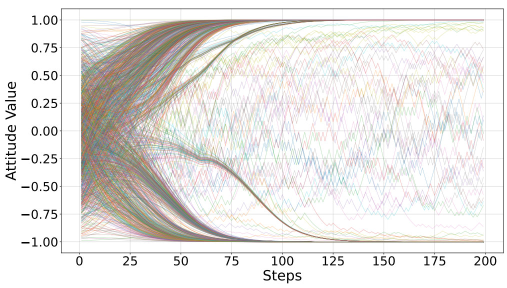
Figure 10: Example of tweet generation during a round of information spreading simulation.



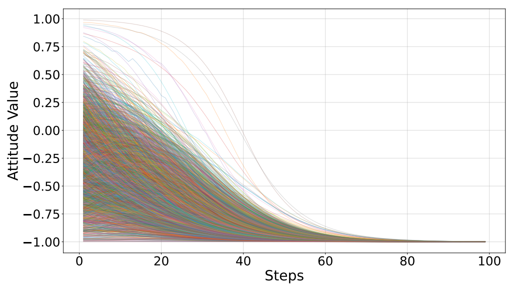
(a) BC in COVID-19



(b) BC in Russia-Ukraine Conflict



(c) Lorenz in COVID-19



(d) Lorenz in Russia-Ukraine Conflict

Figure 11: Results of BC and Lorenz models in group opinion simulation

Time	News
2020/01	WHO issues a global warning about a new coronavirus spreading in Wuhan, China.
2020/02	Diamond Princess cruise ship outbreak highlights virus transmissibility.
2020/03	WHO declares COVID-19 a pandemic.
2020/04	Mass graves in New York City for unclaimed COVID-19 victims.
2020/05	Anti-lockdown protests erupt in the U.S. and Europe.
2020/06	Black Lives Matter protests draw massive crowds despite COVID-19 risks.
2020/07	Surge in cases following Independence Day gatherings in the U.S.
2020/08	Beirut explosion amid the pandemic worsens Lebanon's crisis.
2020/09	India reports over 90,000 daily cases, marking a global peak.
2020/10	White House COVID-19 outbreak infects President Trump.
2020/11	Europe's second wave leads to renewed lockdowns.
2020/12	New COVID-19 variant discovered in the UK.
2021/01	U.S. Capitol riot amid record COVID-19 deaths.
2021/02	Global COVID-19 vaccination efforts ramped up.
2021/03	Brazil's healthcare system collapses amid rising cases.
2021/04	India experiences oxygen shortages during the second wave.
2021/05	Tokyo Olympics proceed without spectators.
2021/06	Delta variant spreads rapidly worldwide.
2021/07	The highly transmissible Delta variant caused a rapid increase in COVID-19 cases worldwide. Hospitals in many countries, including the U.S., India, and Indonesia, were overwhelmed, leading to rising fears about the variant's impact on vaccine effectiveness.
2021/08	Reports highlighted the stark inequality in vaccine distribution, with wealthy countries administering booster shots while poorer nations struggled to vaccinate even frontline workers. This fueled global criticism and fear of prolonged pandemic impacts.
2021/09	The World Health Organization (WHO) classified the Mu variant (B.1.621) as a "variant of interest." Concerns grew about its potential to evade immunity from prior infections or vaccinations, adding to global anxiety.
2021/10	WHO warns of slow vaccination rates in Africa.
2021/11	Omicron variant identified in South Africa.
2021/12	Omicron-driven surge overwhelms global healthcare systems.
2022/01	COVID-19 cases reach record highs globally.
2022/02	Russia invades Ukraine, complicating pandemic recovery efforts.
2022/03	Shanghai enters strict lockdown amid China's zero-COVID policy.

Table 12: Key events related to the COVID-19 pandemic, covering major global developments from the early stages of the outbreak through the challenges of new variants and the ongoing efforts for pandemic recovery.

Time	News
2/13	Russian forces launched a full-scale invasion of Ukraine, marking the beginning of the most intense phase of the conflict. The attack included airstrikes, ground invasions, and naval assaults targeting major Ukrainian cities, including Kyiv, Kharkiv, and Odessa.
2/14	Ukrainian President Volodymyr Zelenskyy rejected an offer of evacuation from the U.S., stating that he needed ammunition, not a ride. Ukrainian forces fiercely resisted Russian advances despite being outnumbered.
2/15	Western countries, including the U.S., European Union, and the UK, imposed heavy sanctions on Russia, targeting banks, businesses, and prominent individuals. NATO countries began sending weapons and supplies to Ukraine.
2/16	Russian forces took control of the Chernobyl nuclear power plant, which had been the site of a catastrophic nuclear disaster in 1986. This raised fears of a nuclear incident amid the ongoing conflict.
2/17	The UN held an emergency session in response to Russia's invasion, with many countries condemning the aggression. Russia vetoed a resolution that would have demanded a ceasefire and withdrawal of forces from Ukraine.
2/18	Russian troops moved closer to Kyiv, Ukraine's capital, while intensifying their assault on cities in eastern Ukraine. Meanwhile, Russia announced it was placing its nuclear forces on alert.
2/19	Ukraine formally applied for European Union membership, emphasizing its desire to align more closely with Western Europe and away from Russian influence.
2/20	Thousands of Ukrainians fled westward to neighboring countries, especially Poland, as the war caused a massive refugee crisis. Humanitarian aid efforts ramped up, though conditions remained dire in besieged cities.
2/21	Russian forces continued to move toward Kyiv, and the city became a focal point of fierce fighting. Ukrainian President Zelenskyy remained in Kyiv, despite calls for his evacuation.
2/22	Ukrainian cities, including Mariupol, faced severe bombardment. Reports began emerging of significant civilian casualties and destruction due to Russian artillery and airstrikes.
2/23	Russian troops effectively encircled Mariupol, a port city in southern Ukraine, cutting off supplies and trapping thousands of civilians.
2/24	NATO leaders met to discuss increased defense aid for Ukraine, while the EU announced new sanctions against Russia, including restrictions on its access to financial systems and technology.
2/25	The international community, including the UN, continued to condemn Russia's actions. Reports of Russian war crimes, including targeting civilians and hospitals, emerged from various parts of Ukraine.
2/26	Humanitarian aid convoys attempted to reach the city, but Russian forces blocked routes, continuing their siege. Meanwhile, the UN confirmed over 2 million refugees had fled Ukraine.
2/27	Despite heavy bombardment, Ukrainian forces continued to put up strong resistance in Kyiv, Kharkiv, and other cities, using guerrilla tactics and fighting house to house.
2/28	The UN General Assembly passed a resolution demanding Russia cease its invasion of Ukraine, with a significant majority of countries voting in favor, though Russia and a few allies opposed it.
3/1	Russian troops captured large parts of southern Ukraine, including the city of Kherson, which became the first major city to fall under Russian control.
3/2	Russia continued its military advance, focusing on strategic locations like Mariupol, which remained besieged, while fighting continued on multiple fronts, especially in the Donbas region.

Table 13: Timeline of key events during the early stages of Russian-Ukraine Conflict in 2022.