Towards a Principled Multi-Agent Approach to Simulating Social Platforms

Stephanie Anneris Malvicini[1,2][0009−0001−1685−3580], Gerardo Ignacio Simari[2,3][0000−0003−3185−4992], and Maria Vanina Martinez[1][0000−0003−2819−4735]

[1] Artificial Intelligence Research Institute (IIIA-CSIC), Barcelona, Spain
{stephanie.malvicini, vmartinez}@iiia.csic.es
[2] Dept. of Comp. Sci. and Eng., Universidad Nacional del Sur (UNS), Argentina
gis@cs.uns.edu.ar
[3] Inst. for Comp. Sci. and Eng. (ICIC UNS-CONICET), Argentina

Abstract. Simulating social platforms has become increasingly important, as access to real-world data is becoming more limited, and controlled experimentation on real platforms is often impractical, expensive, or simply infeasible. Traditional agent-based models (ABM) and Belief-Desire-Intention (BDI) architectures offer transparency and a theoretical grounding; however, they are not especially suited to operate in environments in which rich natural language is used. Recent Large Language Model (LLM)–driven agentic approaches enable natural language interaction, but often at the expense of interpretability, control, and reproducibility. In this work, we propose a general Social Platform Simulator framework leveraging a Social Agent model that supports a spectrum of agent implementations, ranging from fully LLM-based to structured symbolic architectures. The framework explicitly models key components of social platforms, including the social network, content management, filtering algorithms, and shared knowledge, enabling modular, controlled, and extensible simulations of social media–like environments. We demonstrate the expressiveness of the framework by analyzing both a minimalist LLM-based instantiation and a hybrid architecture that allows LLMs to be part of all internal processes of a classical BDI framework, which we refer to as "BDI+", and represents a balance between the control and transparency of BDI architectures and the expressive power of LLMs. The proposed framework supports reproducible social simulations and the generation of synthetic datasets that can be used for benchmarking in social data ecosystems. By means of a case study on affective polarization, we show how BDI+ facilitates structured interventions, heterogeneous agent roles, and the explicit analysis of internal agent states.

Keywords: Hybrid Agent Architectures · Social Platform Simulation.

1 Introduction

Social platforms have attracted significant attention within the scientific community, spanning disciplines such as sociology, computer science, economics, and

political science. This interest is well justified: these platforms increasingly shape how we act, what we think, and how we interact, influencing nearly every aspect of modern life. In recent years, particular attention has been given to misinformation and polarization due to the risks they pose, including threats to democratic processes. Historically, many studies were able to rely on real-world data from social platforms to conduct empirical research. However, this has become increasingly challenging, especially after X (formerly Twitter) restricted access to its free API. Although some open-access platforms remain available, they lack the scale and diversity needed to represent the broader population adequately. Moreover, using real platforms makes it difficult to test specific interventions or to construct controlled scenarios tailored to particular research questions. This is where simulations play a crucial role: they simplify and structure the world to focus on key variables, allowing researchers to study smaller, less detailed, and less complex problems while retaining their essential dynamics [13]. Computational social science simulations enable the modeling and understanding of social processes without requiring real human data, which is often costly, limited, or poorly aligned with the phenomena under investigation [13]. Additionally, simulations allow researchers to test interventions in a controlled environment and to explore hypotheses before conducting expensive and logistically challenging experiments with real participants.

For decades, formal agent and multi-agent models have been developed and widely employed in social simulation. Agent-based modeling (ABM) has been extensively used to study the dynamics of complex social and environmental systems [2]. Another influential framework is the Belief-Desire-Intention (BDI), providing a symbolic architecture for representing agents' mental states and deliberative processes [22]. These approaches offer strong theoretical foundations and interpretability, but are generally domain-agnostic and are not especially suited for processing and reasoning over natural language, which limits their ability to engage in unconstrained interactions like the ones found in social media environments. Recent years have witnessed increased interest in so-called agentic AI, approaches that largely differ from traditional model-centric architectures relying heavily on large language models (LLMs) as their central component. In this paradigm, a language model serves as the agent's core reasoning engine and interacts with its environment through a set of external tools, enabling actions such as information retrieval, code execution, and other task-specific operations.[4]

In this work, we propose a general *Social Platform Simulator* framework leveraging a *Social Agent model*, designed to support implementations across a spectrum ranging from lightweight agents, such as those used in ABM or LLM-based approaches, to fully formal agents. The proposal is sufficiently expressive to simulate a wide range of social scenarios while remaining simple, interpretable, and easy to implement. We demonstrate how a fully LLM-based multi-agent platform can be instantiated within our modeling framework, and we then use an extended BDI model, which we refer to as "BDI+", that combines the control and structure of formal models with the expressive power of LLMs. This balance

[4] `https://lilianweng.github.io/posts/2023-06-23-agent`

enables modular simulation of complex social media–like environments while facilitating the generation of synthetic datasets for benchmarking in social data ecosystems. Furthermore, it allows the analysis of latent variables such as agents' internal states or emotions that are typically inaccessible in real-world data.

2 Related Work

The literature on social simulations is extensive, spanning both general frameworks and context-specific studies. On the simpler, more symbolic side, [23] proposes an agent architecture based on decision-making for modeling and simulating user behavior to analyze communication dynamics on social media. Similarly, [14] adopts Social Judgment Theory to operationalize attitude shifts and study polarization dynamics. A more rule-based, general approach is presented in [8], where the authors demonstrate how Network Knowledge Bases can model information flow in social networks via belief revision operators.

Some approaches that are related to social platform simulation integrate symbolic and sub-symbolic reasoning. For example, [11] presents a multi-agent system addressing cybersecurity issues in social networks, combining symbolic reasoning for detecting malicious behavior with sub-symbolic components, such as Machine Learning (ML) classifiers for user type identification. [5] provides a survey on the combination of BDI models and ML, for different tasks including plan selection, experience learning, decision-making, intention selection, and belief reasoning. Similarly, [1] proposes a framework to analyze user and community sentiment over time on social platforms, leveraging BERT for topic and sentiment detection. Another line of research integrates symbolic models with LLMs. [15] goes beyond standard probabilistic diffusion models by embedding language-level user behavior into simulations using LLMs to study information propagation. [12] extends BDI agents with human communication capabilities via LLMs, and [7] combines BDI with LLMs for explainable human–robot interaction. [16] proposes an architecture that bootstraps agent reasoning via reinforcement learning and allows natural language instructions for BDI agents using a natural language inference model for plan selection. While not strictly social simulations, [18] illustrates the benefits of LLM integration in cognitive architectures by proposing SOFAI-LM, a hybrid system using metacognitive feedback to combine fast LLMs with slower reasoning modules for enhanced problem-solving and reasoning efficiency, demonstrating that SOFAI-LM enables LLMs to match or outperform standalone large reasoning models in accuracy while maintaining significantly lower inference time. At the other extreme, some approaches rely entirely on LLMs. [25] presents SOTOPIA-S^4, a scalable LLM-based social simulation platform supporting multi-turn, multi-party interactions for social science experiments. [24] introduces GenSim, a large-scale, error-correctable LLM-agent simulation platform for complex social behaviors, while [20] develops a conceptual framework for disinformation research using LLM-based agents.

Despite these advances, classical ABMs and ML-based approaches have inherent limitations. ABMs are often simple and heavily dependent on empirical

data for parameterization, making them less generalizable. Minimalist ABMs frequently reduce agent validity, resulting in implausible simplifications guided by rationality theory with only limited psychological realism [3]. When agents are designed solely based on expected behaviors rather than grounded in theory, they can become arbitrary, domain-specific, and poorly comparable [3]. ML approaches can be effective for targeted tasks, but often fail to generalize beyond their training datasets, whereas LLMs, although flexible, lack robustness, domain knowledge, and explainability. They are susceptible to inconsistent responses, adversarial inputs, and biases, raising concerns about fairness, harmful outputs, and transparency [10]. Our goal is thus to propose a sufficiently generic architecture capable of simulating diverse social network phenomena while retaining basic components that ensure simplicity and ease of implementation. Our thesis is that combining the generalizability of symbolic approaches with the adaptability of ML and LLM components, we can balance realism, interpretability, and practical usability.

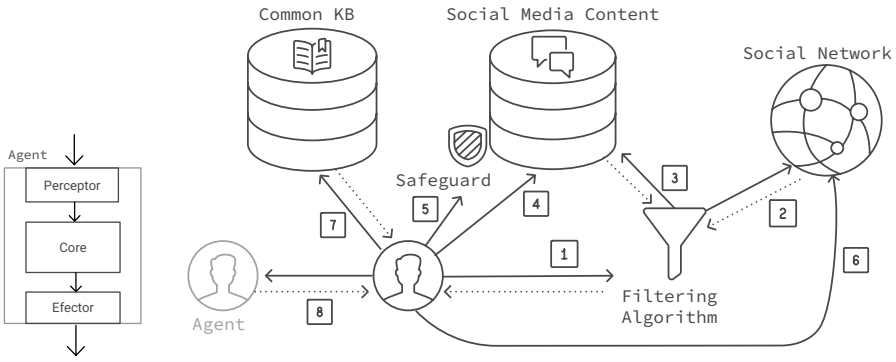
3 A Model for Simulating Social Platforms

We propose a Social Platform Simulator framework leveraging a general Social Agent model, and demonstrate how the latter can be instantiated to support different degrees of control over the agents' internal reasoning and decision making. In this section we describe the general model and, in Section 4, we illustrate the model's flexibility, presenting two concrete implementations: a lightweight approach based solely on LLMs, and a hybrid approach that integrates a traditional BDI architecture with LLM-based components.

3.1 Social Agents

In the most basic conceptualization, an intelligent agent consists of three fundamental components that together enable autonomous interaction with its environment: (i) *perceptors* that gather information about the current state of the environment; (ii) a *core* (or decision-making module) that interprets perceptual input, maintains internal state or beliefs, and selects appropriate actions based on goals and reasoning mechanisms; and (iii) *effectors* that execute the chosen actions, producing changes in the environment [6]. It is important to note that, since our agent will operate within a social platform where information is primarily exchanged in natural language, both the perceptor and effector modules may include components for translating between natural language and the agent's internal representations. For the purposes of this work, these components will largely rely on LLMs to interpret and generate language effectively.

The proposed Social Agent model follows the aforementioned structure, as shown in Figure 1a, and implements a set of predefined actions that allow the general and expected interactions within a text based social platform. Those actions, based on [11], are the following:



(a) (b)

Fig. 1: (a) Proposed Social Agent model. (b) Proposed Social Platform Simulator – 1: Content retrieval action; 2: Social Network request; 3: Social Media Content request; 4: React action; 5: Post action; 6: Update links action; 7: Ask action; 8: Read public profile action.

- *Retrieve content*: Get relevant content from the social platform, whether requesting a feed update or searching for a specific term or authorship.
- *Post*: Publish content to the platform by creating a new post, re-posting or commenting on an existing message.
- *React*: Interact with an existing message by adding a reaction.
- *Ask*: Query the platform for shared information, norms, conventions, etc.
- *Update links*: Follow or unfollow an agent within the platform.
- *Read public profile*: Request the publicly visible profile data of another agent.

This abstract architecture provides a flexible foundation for constructing social agents tailored to different application domains and levels of formalization. It allows the internal components to be implemented with varying degrees of structure and control. Below we show a simple instantiation relying exclusively on LLMs, as well as a more formal one that embeds a BDI architecture within the agent core, using LLMs as supporting components. This highlights how the proposed model accommodates a spectrum of design choices, from highly flexible but opaque reasoning to more structured and controllable behavior.

3.2 Social Platform Simulator

The proposed Social Platform Simulator comprises five main components, illustrated in Figure 1b and defined as follows:

Social Network: Responsible for maintaining the network structure, representing the set of agents and their relationships. Following the approach of [9], we model the social network as a directed graph, consisting of a set of nodes (agents)

and a set of edges denoting the relationships between them. Each edge has an associated weight denoting the strength of the relation.

Social Media Content: Responsible for managing all published posts (messages) and their associated attributes (e.g., creator, timestamp, likes, re-posts, reactions, etc.). It includes a safeguard mechanism to prevent the publication of undesirable content such as nudity or hate speech. In our model, a **message** is the primary artifact used for information exchange. In this analysis, we focus exclusively on text-based messages, although a message could take other forms. Each message has a unique *id* to identify it, a reference to the agent *creator* of the message, a *content* and a *timestamp*. It may optionally reference another message, identifying the *original* or parent message to which the current message responds or relates, such as a reply or comment. It also contains a list of *reactions*, consisting in pairs author of the reaction and reaction type (for example, "like" or "love"). Every message in the Social Media Content is created based on a list of *topics* and a set of *variables* that further describe the content.

Filtering Algorithm: Responsible for determining the content that each agent is exposed to. It filters and prioritizes messages based on the agents' profiles and personal data. As agents interact within the platform, they generate data that the algorithm can leverage to self-adapt and enhance personalization [4]. Agents may request the algorithm to update their feed or to search for specific terms or authorship. In all cases, the Filtering Algorithm processes the Social Network and its Content modules to retrieve and display the most relevant information.

Common Knowledge Base: It can be seen as a shared static repository that stores background information about the environment, such as norms, conventions, rituals, organizational structures, and factual knowledge, serving as a fixed reference for reasoning and decision-making.

4 A Proof-of-Concept Instantiation of the Model

Our proposed model can be instantiated in many different ways, offering varying levels of control and detail not only within the simulator modules, but also on the agent side. For example, agents may be as simple as the ABMs commonly used in sociology, or more complex architectures such as the previously mentioned BDI framework. The system may also be heterogeneous, combining different agent implementations within the same simulation. We illustrate this flexibility by first presenting a simple approach from the literature where all control is delegated to an LLM, and then describing a simple extension of the BDI architecture that incorporates LLM support at different stages of the reasoning process.

4.1 Lightweight LLM-Driven Agents

A straightforward implementation of the proposed architecture, in which an LLM subsumes all cognitive functions of the agent, is [19], where authors develop a multi-agent platform with agents interacting in a simulated online discussions.

Each agent is defined by a persona description (specifying characteristics such as behaviors, goals, beliefs, values, and personality traits to make the agent more realistic and relatable) along with a political standpoint (indicating the degree of alignment with a political group) and demographic attributes. Agents interact by reading discussion threads and generating responses. Before and after the discussions, they complete questionnaires designed to measure affective variables of interest. All tasks, including perception of the environment, reasoning, responding to questionnaires, and natural language text generation, are performed by the same LLM instance. As a result, reasoning and decision-making are implicit and fully embedded in the LLM's internal representations, making them difficult to inspect or constrain. The platform is simple and lacks a network representation. There is effectively no filtering algorithm, as it simulates only a single discussion thread, and agents read all messages in the order they appear. There is also no explicit common knowledge base; however, since the authors specify to the LLM that the discussion is X (formerly Twitter)-like, we can assume that common knowledge is implicitly represented in the LLM's training data.

Despite these limitations, the study demonstrates (in Experiment 1 [19]) how the platform can be used to examine variations in affectivity and polarization among non-partisan agents when reading threads initiated by a Democratic leader and followed by responses from Republicans. In this context, the platform functions both as a simulator, providing data for analyzing non-partisan affectivity changes, and as an agent (the non-partisan) leveraging the LLM to "reason" about its own affective state. Experiment 1 is conducted as follows: the authors create 50 agent configurations, each consisting of 10 Republican agents and 1 non-partisan agent. For each agent, they design a persona description, along with demographic attributes (gender, race, ethnicity, education level, and age group) aligned with distributions from the US Census Bureau. A feeling-thermometer–like questionnaire is used to measure agents' levels of love and hate toward both Republicans and Democrats on a scale from 0 to 10. An agent is considered to identify a party as its in-group if its level of love toward that party is greater than or equal to 5, and is considered polarized if it has an in-group and a level of hate toward the out-group greater than 5. A non-partisan agent is defined as one that has no in-group. Each simulation begins with the agents interacting in an X thread initiated by a tweet from Joe Biden, following a round robin order. After the simulations, the authors observe significant shifts in the non-partisan agents' attitudes: increased love toward Republicans ($+3.08$) and increased hate toward Democrats ($+1.74$), decreased love toward Democrats (-1.02) and decreased hate toward Republicans (-1.38). As a result, 64% of non-partisan agents adopt a Republican in-group identity, 4% adopt a Democratic one, and 62% become polarized.

Although the platform serves as a very simple testbed, it effectively shows that straightforward scenarios can be executed to gain initial insights into a problem, benefiting from the flexibility of LLMs and their capacity to generate and analyze human-like natural language text. However, it has several limitations: it lacks cognitive structures and learning mechanisms to capture long-term

changes, provides no explanation for why agents behave as they do, and poses challenges for replicability and the simulation of more complex scenarios. While this implementation demonstrates the potential of LLMs for simulating social interactions and measuring affective outcomes, it leverages perception, reasoning, memory, and generation into a single opaque component. This design limits interpretability, control, and extensibility. These limitations motivate the need for an architecture that preserves the expressive power of LLMs while reintroducing modularity, transparency, and structured decision-making.

4.2 BDI-based Agents

For the purposes of this paper, BDI+ denotes a modern interpretation of the traditional BDI architecture in which the core components are preserved, while LLMs are incorporated as flexible support mechanisms that can be selectively employed at different stages of the agent's operation. BDI+ is introduced to address the limitations of fully LLM-controlled agents and of classical BDI systems, which typically rely on symbolic reasoning and predefined rules and therefore struggle in rich, language-based environments such as online social platforms. Unlike prior work [12], where LLMs primarily act as transducers between the agent and the environment, our approach allows LLMs to influence multiple stages of the BDI cycle, including belief updating, deliberation, and intention selection. This design preserves the modularity and interpretability of BDI while leveraging the expressive and generative capabilities of LLMs.

Figure 2 illustrates the core components of the proposed BDI+ architecture. Each agent maintains a set of Desires, representing abstract goals or motivations. Through a deliberation process, a subset of these desires is selected and transformed into Intentions, which are the goals the agent actively commits to pursue. The Plan Generator then constructs a plan for each intention based on the agent's Local Knowledge Base, which stores beliefs about the agent itself, other agents, and the environment. Each plan consists of an ordered sequence of parameterized actions (as defined in Section 3.1). For example, an action that establishes a social link requires the identifier of a target agent as a parameter. When actions are executed, the agent interacts with the environment and receives feedback, which is stored in memory. Memory updates follow a defined policy and trigger corresponding updates to the Local Knowledge Base, allowing the agent's beliefs to evolve over time. In addition, each agent exposes a set of public attributes (e.g., username, profile description) that can be queried by other agents, as well as a unique identifier. While these elements closely follow the classical BDI framework, the LLM module spans multiple components of the architecture, enabling language-based reasoning and adaptive decision-making throughout the agent's control loop. Depending on the application, LLMs may be employed to support different stages of the reasoning process, including belief acquisition and revision from unstructured textual information, the extraction of latent agent attributes such as affective states or preferences that can be incorporated into the Local Knowledge Base, deliberation and intention selection, plan generation and adaptation, action parameterization, and reasoning about

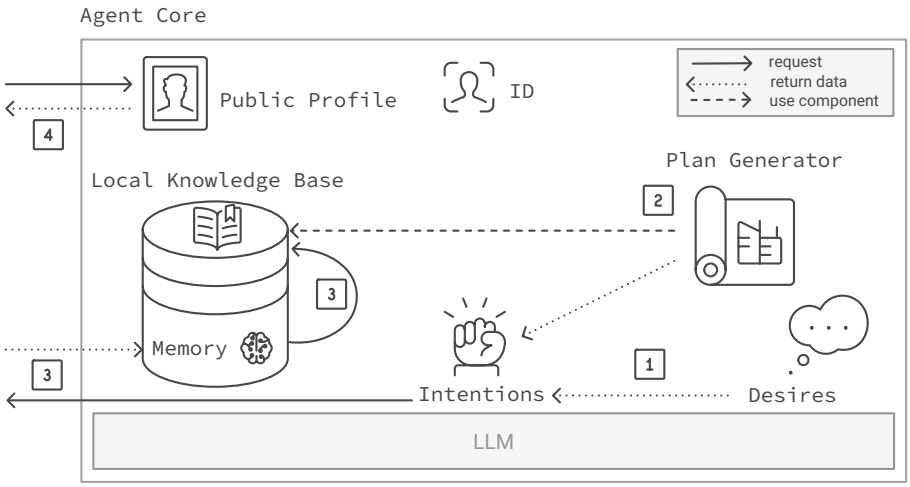


Fig. 2: BDI+ Agent Core – 1: Deliberation converts Desires into Intentions. 2: Plan Generator creates plans associated with Intentions based on the Local KB. Each plan consists of a sequence of parameterized actions. 3: Execution of an action and retrieval of corresponding content. Memory gets updated, which triggers an update to the Local KB. 4: Public profile gets requested and retrieved.

the beliefs, intentions, or likely behavior of other agents. This flexibility allows to selectively augment symbolic mechanisms with LLM-based reasoning where explicit rules are unavailable, difficult to engineer, or insufficient to capture the complexity of social and linguistic contexts.

An Example Implementation

To demonstrate the feasibility of the BDI+ approach, we implemented a simplified version[5] of the architecture using Jason 3.3, an interpreter for the AgentSpeak(L) language [21], one of the most influential BDI-based programming languages. Jason was selected not only because of its mature implementation of the BDI paradigm, but also because of its highly modular architecture. The platform allows developers to redefine or replace virtually any stage of the agent reasoning cycle, including belief revision, event selection, plan selection, communication with other agents, intention selection, and action execution. This flexibility makes Jason particularly suitable for exploring hybrid architectures in which LLMs augment or support traditional symbolic reasoning components. Beyond these core reasoning mechanisms, we also exploit Jason's extensibility through custom internal actions and architecture interfaces to implement the perceptual and action transducers responsible for interpreting and generating natural-language content. These extensions enable the integration of arbitrary translation mechanisms between symbolic representations and external inputs

[5] Available at: `https://github.com/ICBD-ICIC/Jason/tree/main/experiments`

and outputs. Although our prototype employs an LLM for these translation tasks, the architecture remains model-agnostic and can readily accommodate alternative approaches, including symbolic, rule-based, or multimodal systems capable of processing images, videos, or other forms of content. To support experimentation, we also implemented a simplified version of the simulation environment in Java, including the social-network management, content storage and feed filtering. The clear separation between the agent architecture and the simulation environment improves reproducibility and supports the incremental extension of both the platform and the agents' capabilities in future studies.

Our goal was to replicate, as closely as possible, Experiment 1 described in Section 4.1, and then illustrate how BDI+ enables extensions that are difficult to carry out in a fully LLM-driven architecture. We implemented two agent types, Republicans and Non-partisan, each one initialized with the same attributes used in the original study: political standpoint, persona description, and demographic characteristics. For each of the 50 experimental configurations, we instantiated 10 Republican agents and 1 non-partisan agent, using the same values reported in the original study. Interaction dynamics were reproduced by the usage of an orchestrator agent that enforces a Round Robin posting order, ensuring sequential replies within the simulated discussion thread. We employed the same language model and configuration (gemini-2.0-flash with temperature 0.7) for both affectivity assessment and text generation. Prior to the discussion, agents initialize their levels of love and hate toward the Republican and Democratic parties using a feeling thermometer-style questionnaire. Unlike the original implementation, each questionnaire is modeled as a separate BDI action with its own LLM prompt. This design emphasizes the modularity of BDI+, as LLM-based reasoning can be selectively replaced or augmented by alternative mechanisms. From the perspective of the BDI+ architecture (Figure 2), this use of the LLM can be understood as part of the Local Knowledge Base update process. In particular, the LLM is not used to perform symbolic reasoning over explicitly represented beliefs, but rather to infer affective information that is difficult to obtain through predefined rules or symbolic mechanisms alone. The model is therefore employed to uncover latent affective attributes of the agent, providing quantitative estimates of its levels of love and hate toward each political group, which can then be represented and manipulated as explicit beliefs within the agent's internal state.

From the original 50 configurations, we discarded runs in which the non-partisan agent was initialized with a love value greater than 5 for any party (even after multiple re-initializations), resulting in 42 valid runs. Across these runs, non-partisan agents exhibited increased love toward Republicans (+0.17), increased hate toward Democrats (+0.40), decreased love toward Democrats (−0.26), and increased hate toward Republicans (+0.14). As a result, 14.29% of non-partisan agents adopted a Republican in-group identity, none adopted a Democratic one, and 2.38% became polarized. Differences with respect to the original experiment can be attributed to factors such as independent questioning, prompt variations, and the inherent non-determinism of LLM outputs.

To illustrate the additional capabilities enabled by BDI+, we extended the simulation by introducing two additional agents representing botnet accounts. These agents have the explicit goal of reducing the level of hostility in the discussion. In this simplified implementation, bot agents periodically read the Republican agents' conversation and generate responses aimed at de-escalation. More sophisticated policies could be implemented by triggering interventions based on detected linguistic or affective cues. Under this extended setup, all 42 runs were again successful. The resulting affective changes were smaller in magnitude: increased love toward Republicans (+0.07), increased hate toward Democrats (+0.26), decreased love toward Democrats (−0.17), and increased hate toward Republicans (+0.02). As a result, 7.14% of non-partisan agents adopted a Republican in-group identity, none adopted a Democratic one, and 2.38% became polarized. These results suggest that, in our experiment, the bot agents substantially reduce shifts in group identification among non-partisan agents, although they do not affect the proportion of polarized agents. The types of interventions that BDI+ allows us to implement are difficult to replicate in purely LLM-driven approaches, as they are highly sensitive to prompt content and length which could lead to unintended outcomes and radicalized behaviors.

5 Conclusion and Future Work

We presented a general Social Platform Simulator framework together with a flexible Social Agent model that supports a wide spectrum of agent implementations, ranging from lightweight agents to structured symbolic architectures. By explicitly modeling key platform components—such as the social network, content management, filtering algorithms, shared knowledge, and the basic actions available to agents, the proposed architecture enables controlled, modular, and extensible simulations of social media–like environments.

We demonstrated two concrete instantiations of this framework: a minimalist, fully LLM-based approach drawn from prior work, and a more capable BDI+ architecture, which integrates LLMs throughout the classical BDI cycle. While the LLM-only implementation showcases the expressive and natural-language capabilities of generative models, BDI+ preserves interpretability, modularity, and fine-grained control over agent reasoning. BDI+ is particularly useful for the robust implementation of heterogeneous agent roles, targeted interventions, and explicit tracking of internal agent states, which are essential for studying complex social phenomena such as polarization and misinformation. Moreover, because agents interact primarily through natural-language messages, the framework naturally supports mixed simulations involving both artificial agents and human participants.

The modular design of the simulator facilitates systematic experimentation with individual components—such as recommendation mechanisms, content filtering strategies, or decision-making policies—without requiring changes to the rest of the system. This flexibility enables the construction of simpler, more scalable agent models with predictable behavior, partially mitigating the repro-

ducibility and control challenges associated with purely LLM-driven approaches. Additionally, the explicit representation of agents, actions, and network structure enables detailed analyses of information diffusion and social dynamics that would be difficult to obtain from opaque, fully generative systems.

Although this work focuses on text-based interactions, the simulator's modular architecture supports plug-and-play extensions that enable more realistic simulations. Real social platforms are inherently multimodal, with images and videos playing a substantial role in shaping user behavior and information spread. Future extensions could incorporate ML or other methods to extract knowledge from multimodal data, represent it in a formalized manner, and integrate it into the agents' reasoning processes. Another important future direction is the use of the simulator as a controlled generator of synthetic data for social media research. In increasingly constrained real-world data environments, such simulators can serve as generators of benchmark datasets for evaluating models, in some cases matching or even surpassing the utility of human-generated data [17].

Acknowledgments. This work was supported in part by the Spanish MCIN/AEI grant (CHIST-ERA iTrust) project PCI2022-135010-2,project PID2022-139835NB-C21, PIE 2023-5AT010 CSIC, and Universidad Nacional del Sur (UNS) grant PGI 24/ZN057.

References

1. Bonifazi, G., Cauteruccio, F., Corradini, E., Marchetti, M., Terracina, G., Ursino, D., Virgili, L.: A framework for investigating the dynamics of user and community sentiments in a social platform. Data & Knowledge Engineering **146**, 102183 (2023)
2. Brugière, A., Nguyen-Ngoc, D., Drogoul, A.: Handling multiple levels in agent-based models of complex socio-environmental systems: A comprehensive review. Frontiers in Applied Mathematics and Statistics **8**, 1020353 (2022)
3. Conte, R., Paolucci, M.: On agent-based modeling and computational social science. Frontiers in psychology **5**, 668 (2014)
4. Eg, R., Özlem Demirkol Tønnesen, Tennfjord, M.K.: A scoping review of personalized user experiences on social media: The interplay between algorithms and human factors. Computers in Human Behavior Reports **9**, 100253 (2023)
5. Erduran, Ö.I.: Machine learning for cognitive BDI agents: A compact survey. In: Proc. ICAART. pp. 257–268 (2023)
6. Franklin, S., Graesser, A.: Is it an agent, or just a program?: A taxonomy for autonomous agents. In: Proc. ATAL (1996)
7. Frering, L., Steinbauer-Wagner, G., Holzinger, A.: Integrating belief-desire-intention agents with large language models for reliable human–robot interaction and explainable artificial intelligence. Eng. Appl. Artif. Intell. **141**, 109771 (2025)
8. Gallo, F.R., Simari, G.I., Martinez, M.V., Falappa, M.A., Santos, N.A.: Reasoning about sentiment and knowledge diffusion in social networks. IEEE Internet Computing **21**(6), 8–17 (2017)
9. Gallo, F.R., Simari, G.I., Martinez, M.V., Santos, N.A., Falappa, M.A.: Local belief dynamics in network knowledge bases. ACM Trans. Comput. Logic **23**(1) (2021)

10. Gao, C., Lan, X., Li, N., Yuan, Y., Ding, J., Zhou, Z., Xu, F., Li, Y.: Large language models empowered agent-based modeling and simulation: A survey and perspectives. Humanities and Social Sciences Communications **11**(1), 1–24 (2024)
11. Garcia, A.C., Martinez, M.V., Deagustini, C.A., Simari, G.I.: A multi-agent system for addressing cybersecurity issues in social networks. In: ENIGMA@KR. pp. 43–54 (2023)
12. Gatti, A., Mascardi, V., Ferrando, A.: ChatBDI: think BDI, talk LLM. In: Proc. AAMAS (2025)
13. Gilbert, N., Troitzsch, K.: Simulation for the social scientist. McGraw-Hill Education (UK) (2005)
14. Haque, A., Ajmeri, N., Singh, M.P.: Understanding dynamics of polarization via multiagent social simulation. AI & society **38**(4), 1373–1389 (2023)
15. Hu, Y., Sherpa, G., Zhang, L., Li, W., Bai, Q., Wang, Y., Wang, X.: An LLM-enhanced agent-based simulation tool for information propagation. In: Proc. IJCAI (2024)
16. Ichida, A.Y., Meneguzzi, F., Cardoso, R.C.: BDI agents in natural language environments. In: Proc. IFAAMAS (2024)
17. Kar, A., Prakash, A., Liu, M.Y., Cameracci, E., Yuan, J., Rusiniak, M., Acuna, D., Torralba, A., Fidler, S.: Meta-sim: Learning to generate synthetic datasets. In: Proc. IEEE ICCV (2019)
18. Khandelwal, V., Rossi, F., Murugesan, K., Miehling, E., Campbell, M., Ramamurthy, K.N., Horesh, L.: Language models coupled with metacognition can outperform reasoning models. arXiv preprint arXiv:2508.17959 (2025)
19. Malvicini, S.A., Gajewska, E., Derbent, A., Budzynska, K., Chudziak, J.A., Martinez, M.V.: A natural language agentic approach to study affective polarization. In: Proc. ICAART, To Appear (2026)
20. Pastor-Galindo, J., Nespoli, P., Ruipérez-Valiente, J.A.: Large-language-model-powered agent-based framework for misinformation and disinformation research: opportunities and open challenges. IEEE Security & Privacy **22**(3), 24–36 (2024)
21. Rao, A.S.: Agentspeak(l): BDI agents speak out in a logical computable language. In: Agents Breaking Away. pp. 42–55. Springer Berlin Heidelberg (1996)
22. Rao, A.S., Georgeff, M.P., et al.: BDI agents: From theory to practice. In: ICMAS. vol. 95, pp. 312–319 (1995)
23. Rodermund, S.C., Lorig, F., Berndt, J.O., Timm, I.J.: An agent architecture for simulating communication dynamics in social media. In: Proc. MATES (2017)
24. Tang, J., Gao, H., Pan, X., Wang, L., Tan, H., Gao, D., Chen, Y., Chen, X., Lin, Y., Li, Y., et al.: Gensim: A general social simulation platform with large language model based agents. In: Proc. ACL (2025)
25. Zhou, X., Su, Z., Feng, S., Zhou, J., Huang, J.t., Kao, H.T., Lynch, S., Volkova, S., Wu, T., Woolley, A., et al.: SOTOPIA-S4: a user-friendly system for flexible, customizable, and large-scale social simulation. In: Proc. ACL. pp. 350–360 (2025)