YuLan-OneSim: Towards the Next Generation of Social Simulator with Large Language Models

Lei Wang, Heyang Gao, Xiaohe Bo, Xu Chen*, Ji-Rong Wen
Gaoling School of AI, Renmin University of China
Beijing Key Laboratory of Research on Large Models and Intelligent Governance
Engineering Research Center of Next-Generation Intelligent Search and Recommendation, MOE
wanglei154@ruc.edu.cn, xu.chen@ruc.edu.cn

Abstract

Leveraging large language model (LLM) based agents to simulate human social behaviors has recently gained significant attention. In this paper, we introduce a novel social simulator called YuLan-OneSim. Compared to previous works, YuLan-OneSim distinguishes itself in five key aspects: (1) **Code-free scenario construction**: Users can simply describe and refine their simulation scenarios through natural language interactions with our simulator. All simulation code is automatically generated, significantly reducing the need for programming expertise. (2) **Comprehensive default scenarios**: We implement 50 default simulation scenarios spanning 8 domains, including economics, sociology, politics, psychology, organization, demographics, law, and communication, broadening access for a diverse range of social researchers. (3) **Evolvable simulation**: Our simulator is capable of receiving external feedback and automatically fine-tuning the backbone LLMs, significantly enhancing the simulation quality. (4) **Large-scale simulation**: By developing a fully responsive agent framework and a distributed simulation architecture, our simulator can handle up to 100,000 agents, ensuring more stable and reliable simulation results. (5) **AI social researcher**: Leveraging the above features, we develop an AI social researcher. Users only need to propose a research topic, and the AI researcher will automatically analyze the input, construct simulation environments, summarize results, generate technical reports, review and refine the reports—completing the social science research loop. To demonstrate the advantages of YuLan-OneSim, we conduct experiments to evaluate the quality of the automatically generated scenarios, the reliability, efficiency, and scalability of the simulation process, as well as the performance of the AI social researcher. Code can be found at https://github.com/RUC-GSAI/YuLan-OneSim.

1 Introduction

Social science has long played a critical role in the advancement of human civilization by providing profound insights into human behavior, societal structures, and cultural dynamics. Social simulation has emerged as a foundational methodology for investigating complex social behaviors and uncovering patterns that are often difficult to observe directly in real-world settings Squazzoni et al. [2014]. For decades, agent-based modeling (ABM) has been prominent in social simulations Heath et al. [2009], Bianchi and Squazzoni [2015], but often fails to capture the intricacies of human cognitive processes and language-mediated interactions Gao et al. [2024a].

*Corresponding author.

39th Conference on Neural Information Processing Systems (NeurIPS 2025) Workshop: Scaling Environments for Agents (SEA).

Recently, large language models (LLMs) have opened new avenues for advancing social simulation. By training on large-scale language corpora, LLMs can exhibit human-like intelligence across a wide range of tasks Zhao et al. [2023], Achiam et al. [2023]. Motivated by these advancements, researchers have increasingly explored leveraging LLMs to build more realistic language-based social simulators, such as GenSim Tang et al. [2024], OASIS Yang et al. [2024a], and AgentSociety Piao et al. [2025].

In this paper, we introduce a novel LLM agent based social simulator, called "YuLan-OneSim". Compared to previous works, our simulator has five key advantages: (1) **Code-free scenario construction** using natural language, substantially reducing programming expertise requirements; (2) **Comprehensive default scenarios** comprising 50 pre-defined scenarios spanning eight domains (economics, sociology, politics, psychology, organizational studies, demography, law, and communication); (3) **Evolvable simulation** through our Verifier–Reasoner–Refiner–Tuner (VR²T) framework that integrates external feedback to evolve backbone LLMs; (4) **Large-scale simulation** supporting up to 100,000 agents via a fully responsive agent framework and distributed architecture; (5) **AI social researcher** that autonomously completes the entire research loop from idea generation to report production.

To realize the above features, our simulator is built based on four subsystems: the scenario auto-construction subsystem, simulation subsystem, feedback-driven evolving subsystem, and the AI social researcher subsystem. To begin with, the scenario auto-construction subsystem translates user requirements into executable code that directly drives the simulation. Based on this code, the simulation subsystem performs the corresponding simulation tasks and continuously oversees and evaluates the simulation process in real time. When simulation results are found to be unreliable, the feedback-driven evolving subsystem incorporates external feedback and retrains the underlying LLMs to improve the overall performance. Built upon these subsystems, the AI social researcher can autonomously complete the entire social science research loop—from idea generation and simulation scenario construction to results analysis and report generation and refinement.

In summary, the key contributions of this paper are as follows:

• We develop a novel social simulator named YuLan-OneSim, which supports code-free scenario construction, a comprehensive set of default scenarios, and evolvable and large-scale simulations.

• Based on YuLan-OneSim, we develop an AI social researcher capable of autonomously completing the entire social science research loop—from initial idea generation to final report production.

• To assess the efficiency and effectiveness of our simulator, we conduct extensive experiments to assess scenario generation quality, simulation reliability and scalability, and AI researcher performance.

2 Related Work

2.1 Traditional Social Simulation

Social simulation has long been valuable for understanding social phenomena through rule-based models. Classical works include Schelling's residential segregation model Schelling [1971], which showed that racial segregation emerges even with high tolerance for diversity; Axelrod's cultural dissemination model Axelrod [1997], demonstrating coexistence of local convergence and global diversity; and Palmer et al.'s artificial stock market Palmer et al. [1999], revealing how simple behaviors generate complex market dynamics. However, these methods rely on heuristic rules and simplistic functions, limiting their ability to capture real human cognitive complexity.

2.2 LLM Agent based Social Simulation

Recent advances in LLMs have enabled more sophisticated social simulators. Early works focused on specific scenarios with limited agents: Generative Agents Park et al. [2023] simulated 25 agents in daily life, EconAgent Li et al. [2023] modeled economic decisions with 100 agents, and various domain-specific simulators emerged for recommendation systems Wang et al. [2025], Zhang et al. [2024a], social networks Gao et al. [2023], conflicts Hua et al. [2023], and elections Zhang et al. [2024b]. The field has progressed toward general-purpose, large-scale simulators. GenSim Tang et al. [2024] supports customizable scenarios with up to 100,000 agents, while AgentScope Gao et al. [2024b] and OASIS Yang et al. [2024a] enable million-agent simulations. AgentSociety Piao et al. [2025] creates virtual cities with geographic realism, and SocioVerse Zhang et al. [2025] incorporates

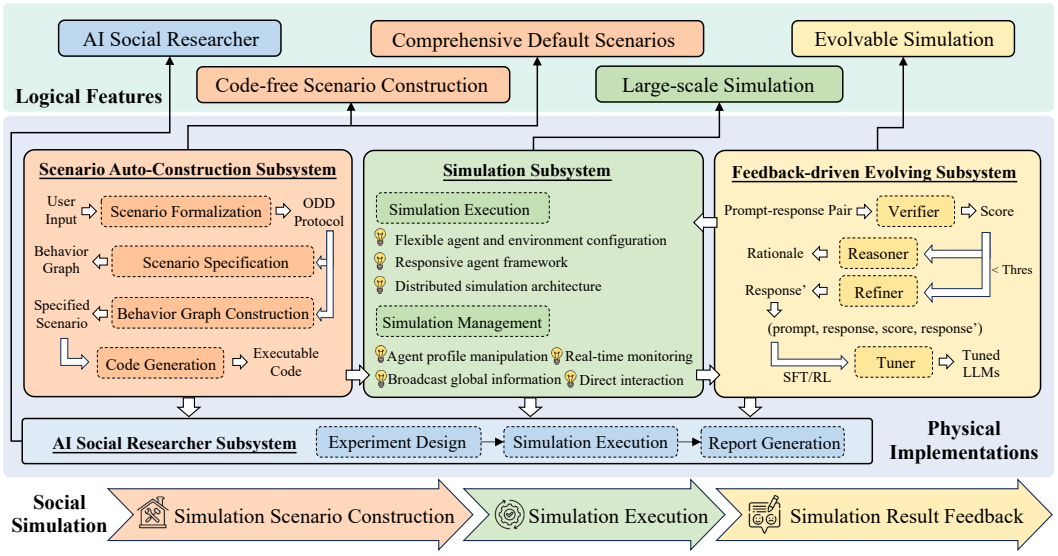

Figure 1: The overall framework of our simulator. The upper subfigure highlights the key features of our simulator (i.e., logical characteristics), while the lower subfigure illustrates its core subsystems (i.e., physical implementations). The correspondences between the features and their supporting subsystems are labeled by solid arrows.

10 million user profiles for survey simulations. YuLan-OneSim advances beyond existing works by enabling automatic scenario construction, autonomous evolution through feedback, and complete AI-driven research cycles with minimal human input. A detailed comparison between YuLan-OneSim and previous studies is provided in Table 5 in the Appendix.

3 YuLan-OneSim

3.1 Overview

The overall framework of our simulator is illustrated in Figure 1. Our simulator divides social simulation into three phases: (1) **Scenario construction**: implementing simulation programs including agent profiles and interaction behaviors; (2) **Simulation execution**: running simulations with large-scale capabilities and runtime support; (3) **Result feedback**: evaluating simulation performance with predefined metrics. Phases (2)-(3) form an iterative execution-feedback loop for continuous refinement. We extend this with an **AI social researcher** that integrates all phases: users provide research questions, which are translated into simulation inputs, executed, and synthesized into final reports.

Traditional scenario construction requires extensive manual coding, creating barriers for social science researchers lacking programming expertise. Our auto-construction subsystem enables users to describe scenarios in natural language while automatically generating executable code through a four-step framework: *scenario formalization*, *behavior graph construction*, *code generation*, and *agent specification*.

Scenario Formalization. We adopt the ODD (Overview, Design Concepts, Details) protocol Grimm et al. [2010] to formalize user requirements. An ODD-translation agent interacts with users through conversation, requesting clarification for ambiguities until sufficient information is gathered to finalize the protocol.

Behavior Graph Construction. The behavior graph $\mathcal{G} = \{\mathcal{N}, \mathcal{E}\}$ defines agent interactions, where nodes $n \in \mathcal{N}$ represent agent-type actions and edges $e \in \mathcal{E}$ indicate action triggers. We extract actions and events from the ODD protocol, then validate the graph structurally (ensuring connectivity from start to end nodes) and semantically (verifying alignment with the protocol).

Code Generation. We employ three strategies: (1) *General code structure*: base classes with common functionalities reduce LLM generation complexity; (2) *Graph-guided generation*: BFS traversal of

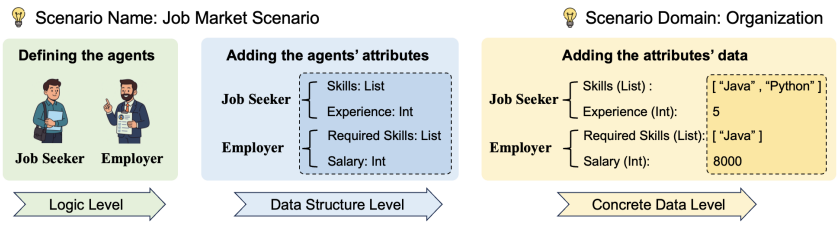

Figure 2: Progressive scenario construction: from logical level to data structure level, and finally to concrete data level.

the behavior graph enables progressive code generation; (3) *Iterative validation*: comprehensive error analysis and repair cycles ensure code quality.

Agent Specification. We decouple code and data, specifying environment data, agent profiles (with LLM or random sampling), and agent relationships. Validation ensures proper network connectivity for event flow.

This progressive construction from logical to data structure to concrete levels (Figure 2) ensures stable, high-quality scenario generation.

To further support social science researchers, we have developed a default scenario repository comprising 50 scenarios across eight domains as ready-to-use examples. A detailed list of these scenarios can be found in Table 6 in the Appendix.

3.2 Simulation Subsystem

The simulation subsystem is responsible for executing and managing the simulation process. From the execution perspective, it supports flexible configurations of the agents and environments. The dynamic interactions between agents and their environments are realized based on a responsive framework as well as a distributed simulation architecture, which readily accommodates large-scale agent simulations. From the management perspective, users can flexibly manipulate agent profiles, broadcast global information, and interact with agents through conversation. Additionally, we implement a monitoring module to observe simulation performance in real time.

3.2.1 Simulation execution

The simulation subsystem executes and manages simulations through flexible agent/environment configurations, a responsive framework, and distributed architecture for large-scale simulations. It also provides real-time monitoring and interactive management capabilities.

Agent Architecture. Each agent comprises four modules: (1) *Profile*: public attributes (name, occupation) accessible to others and private attributes (personality, preferences) influencing internal behavior; (2) *Memory*: customizable storage with strategy (sliding windows, long/short-term), storage type (vector databases, knowledge graphs), and operations (retrieval, reflection, forgetting); (3) *Planning*: three approaches including Chain-of-Thought (single-step reasoning), Belief-Desire-Intention (long-term goals), and Theory-of-Mind (interpersonal interactions); (4) *Action*: converts profile, memory, planning, and context into specific behaviors.

Environment Control. The environment manages simulation lifecycle through Start/End events and two execution modes: *round mode* (turn-based with synchronization) and *tick mode* (continuous asynchronous execution). It maintains shared variables and collects simulation data for analysis.

Responsive Framework. We employ an event-driven, asynchronous architecture centered on an event bus. Events encapsulate occurrences (agent actions, environmental changes) with contextual information. Agents subscribe to relevant event types, enabling natural representation of concurrent activities and complex causal relationships while supporting parallelism and modularity.

Distributed Architecture. Our Master-Worker design supports large-scale simulations through: (1) *Master node*: orchestrates workers, maintains agent registry, and manages global state; (2) *Worker nodes*: execute agent logic in parallel; (3) *High-performance communication*: gRPC-based Wang et al. [1993] with optimized batching; (4) *Topology-aware allocation*: co-locates frequently interacting agents to minimize cross-node communication; (5) *Adaptive routing*: hybrid approach with P2P

optimization for direct worker communication; (6) *Proxy interface*: consistent API for both distributed and single-machine deployments.

Interactive Management. The system enables: real-time agent profile modification, global message broadcasting for intervention studies, direct agent interaction for controlled experiments, and automatic real-time monitoring with graphical visualization of scenario-specific metrics.

3.3 Feedback-driven Evolving Subsystem

Due to the inherent limitations of LLMs (e.g., hallucination and bias), the simulation result at each step may be unreliable. Even more concerning, errors introduced in earlier stages of the simulation can accumulate and amplify as the process progresses.

To alleviate this problem, we incorporate an error correction mechanism based on system or human feedback. Specifically, we design a multi-agent framework comprising a verifier, a reasoner, a refiner, and a tuner, which collaboratively label simulation samples and re-train the backbone LLMs based on the annotated results (see Figure 1). To begin with, the system generates a prompt–response pair, denoted as (p, r). The verifier evaluates the correctness of r with respect to p using predefined metrics and assigns a score s. Next, the reasoner analyzes the rationale behind the score s and produces an explanation, denoted as res. If the score s falls below a predefined threshold thres, the refiner generates a corrected response r' based on the combination of p, r, s, and res. Following this process, for all underperforming prompt–response pairs, we collect the quadruplets $(\mathrm{p}, \mathrm{r}, \mathrm{s}, \mathrm{r}')$. At last, the tuner fine-tunes the backbone LLMs using supervised fine-tuning (SFT) or reinforcement learning (RL). For the former method, we only use the data $(\mathrm{p}, \mathrm{r}')$, while for the latter one, we leverage the complete data $(\mathrm{p}, \mathrm{r}, \mathrm{s}, \mathrm{r}')$.

Each component can be implemented using LLMs or humans. As user feedback accumulates, the simulator evolves toward more reliable, human-aligned simulations. This addresses the challenge that real-world social data is often inaccessible due to privacy, ethical constraints, and availability issues, providing a "social science gym" for controlled knowledge infusion into LLMs.

3.4 AI Social Researcher Subsystem

The AI social researcher enables end-to-end research automation: users input research topics (e.g., "rumor spread on social media"), and the system generates simulation-ready ODD protocols, executes simulations, and synthesizes technical reports through two collaborative modules, as illustrated in Figure 3.

Experiment Design Module transforms research topics into simulation scenarios via three agents: (1) *Inspiration Agent*: generates 3-5 candidate research questions and scenarios with detailed agent types, interactions, and parameters; (2) *Evaluation Agent*: scores scenarios (1-10) across six criteria (feasibility, complexity, research value, agent design, parameter space, insights) and selects the most promising one; (3) *ODD-generation Agent*: converts selected scenarios into standard ODD protocols, assigning domain classification and generating JSON-compatible specifications for direct simulator initialization.

Report Generation Module analyzes simulation results and produces structured research reports through four agents: (1) *Data analysis agent*: interprets results using visualizations and metrics, extracting patterns and contextualizing findings; (2) *Outline writing agent*: creates hierarchical report structure with sections for objectives, setup, results, and conclusions; (3) *Report writing agent*: composes full LaTeX reports following the outline with technical descriptions and data-driven insights; (4) *Reviewer agent*: evaluates reports across four dimensions (insight, structure, content, utility) with scores (1-5) and provides improvement feedback.

The system iteratively refines reports based on reviewer feedback until quality standards are met, enabling researchers to move efficiently from research ideas to comprehensive academic reports with minimal manual effort.

4 Experiments

To evaluate the effectiveness and efficiency of our simulator, we conduct extensive experiments, focusing on the following four research questions (**RQ**): **RQ 1**: How high is the quality of our scenario auto-construction subsystem? **RQ 2**: Are the simulation results of YuLan-OneSim reliable?

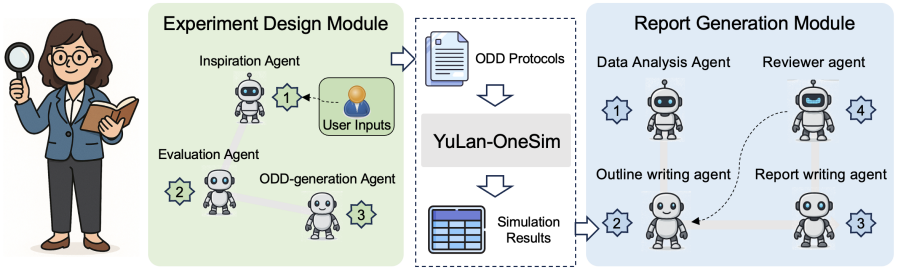

Figure 3: AI Social Researcher workflow with Experiment design and Report generation modules enabling complete research automation.

Table 1: Evaluation results on the scenario construction

Domain	BG-Rating	C-Rating	G-Time	G-Tokens	Files	Lines
Communication	5.00±0.00	4.17±0.37	362.51	20978.0	13.33	588.33
Demographics	4.86±0.35	4.29±0.45	844.19	25342.8	14.25	667.75
Economics	5.00±0.00	4.00±0.37	294.80	20487.4	12.60	581.60
Law	4.57±0.49	4.29±0.45	402.41	20760.8	13.83	615.67
Organization	4.50±0.50	4.00±0.00	412.77	17595.8	13.40	674.00
Politics	5.00±0.00	4.33±0.47	177.05	12491.7	12.33	458.50
Psychology	4.83±0.37	4.17±0.37	310.83	16056.8	12.88	540.25
Sociology	4.83±0.37	4.33±0.47	261.15	16122.7	12.57	518.29
Average	4.82±0.20	4.20±0.13	358.95	18080.7	13.71	570.66

RQ 3: Is the simulation process of YuLan-OneSim efficient and scalable? **RQ 4**: How effective is our AI social researcher?

In the following sections, we address each of these questions in detail.

4.1 Evaluation on the Scenario Auto-Construction Quality

We evaluate our auto-construction subsystem using 50 default scenarios across eight domains with metrics: Behavior Graph Rating (BG-Rating) and Code Rating (C-Rating) on 1-5 scales (detailed criteria in Appendix C), Generation Time (G-Time), and code volume metrics. BG-Rating evaluates the quality of intermediate behavior graphs that serve as the foundation for generating simulation code, scoring based on step identification accuracy and sequence correctness. C-Rating assesses the final generated simulation code quality, considering syntax correctness, executability, and workflow conformance. We empirically adopt GPT-4o as the code generation LLM due to its strong performance. To evaluate the quality of the automatically generated scenarios, we engaged three Ph.D. students to perform a manual evaluation. Results in Table 1 show strong performance: average BG-Rating of 4.82 and C-Rating of 4.20, demonstrating high-quality automated scenario construction. From an efficiency perspective, our system generates code at 50 tokens/second, over 14× faster than human programmers (3.5 tokens/second). Error analysis (detailed in Table 8 in the Appendix) reveals that logical errors dominate, including value access errors, instruction-action mismatches, and incorrect value assignments. These errors are typically straightforward to correct using standard debugging techniques and require minimal manual intervention.

Ablation Study. We conduct ablation experiments on eight representative scenarios (one from each domain) to validate our design choices. Three human annotators evaluate scenario quality on 1-5 scales. Table 2 compares our complete system with variants: without behavior graph (direct code generation), without graph validation, and without iterative refinement. Results demonstrate that each component contributes to quality improvement, with our complete system achieving consistent high scores (average 4.0) across all scenarios.

4.2 Evaluation on the Simulation Reliability

We validate reliability through social theory verification and real-world data alignment.

Social Theory Verification. We implement Axelrod's cultural dissemination model Axelrod [1997] where agents are arranged in an N×N grid, each possessing F cultural features taking one of q possible states. Agents interact with adjacent neighbors based on cultural similarity, potentially adopting

Table 2: Ablation study on scenario auto-construction components

Method	Econ.	Soc.	Poli.	Psy.	Org.	Demo.	Law	Comm.	Avg.
w/o Graph	2	3	3	3	3	3	3	3	2.88
w/o Validation	2	3	3	**4**	3	3	3	3	3.00
w/o Refinement	3	**4**	3	**4**	3	3	3	**4**	3.38
YuLan-OneSim	**4**	**4**	**4**	**4**	**4**	**4**	**4**	**4**	**4.00**

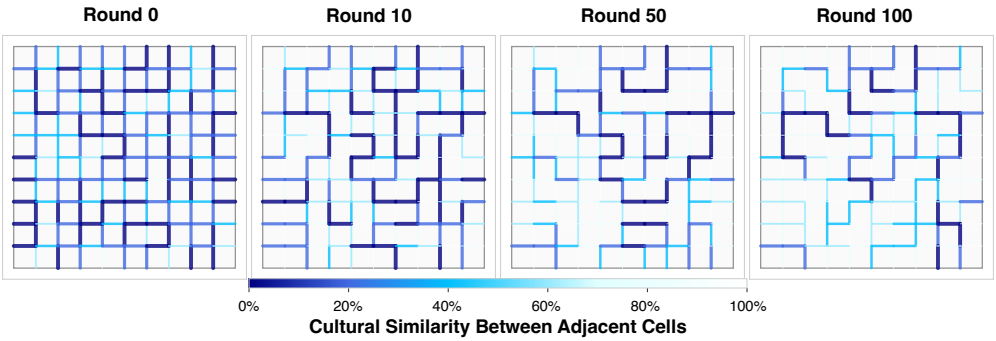

Figure 4: Cultural similarity evolution showing distinct cultural region formation over simulation rounds 0, 10, 50, and 100.

features during interactions. Following Axelrod's setting, we set N=10, F=5, q=5, with LLM-based agent interactions. Figure 4 shows distinct cultural boundaries emerging over 100 rounds. Within each region, neighboring agents exhibit high similarity (darker connections), while boundaries remain visible through lighter connections. This captures Axelrod's core insight—local interactions promote regional homogeneity while preserving global diversity. We quantify this using Local Convergence (LC) and Global Polarization (GP):

$$LC = \frac{1}{|E|} \sum_{(i,j)\in E} \frac{|F_i \cap F_j|}{|F|}, \quad GP = \frac{|C|}{N^2} \tag{1}$$

where E is the set of adjacent agent pairs, F_i and F_j represent cultural feature sets, $|F|$ is total features, $|C|$ is distinct cultural regions, and N^2 is total agents.

Figure 5 reveals key dynamics over 100 rounds. Initially, local convergence increases while global polarization decreases as agents form early clusters. Around round 15, a crossover occurs where local convergence continues rising while global polarization stabilizes at 0.35-0.40. This demonstrates local regions becoming increasingly homogeneous while maintaining distinct boundaries—confirming Axelrod's theoretical predictions with quantitative validation.

Real-world Data Alignment. We simulate Brazilian real estate markets Furtado [2018] with six agent types (individuals, families, government, banks, retail companies, real estate companies) across three interconnected markets: talent recruitment, retail goods, and real estate. Individuals submit resumes for recruitment, form families that consume products and properties, while companies make production and pricing decisions. We conducted 12 rounds (one year) and compared with actual Brazilian rental price distributions from 2020. Figure 6 shows strong alignment, particularly the multi-modal pattern and correspondence in lower price ranges (0.05-0.25). While some discrepancies exist in mid-range values (0.45-0.55) due to simplified real-world factor representation, results demonstrate our simulator's capability to approximate real-world economic distributions with reasonable accuracy.

4.3 Evaluation on the Simulation Efficiency and Scalability

In this section, we evaluate the efficiency and scalability of our simulator. Specifically, we continue using the Axelrod's cultural dissemination scenario from Section 4.2, but scale it up by setting N=320, F=5, and q=10, resulting in a simulation with approximately 100,000 agents. We conduct experiments on a server with 8 A100 40GB GPUs and 96 CPU cores, employing distributed architecture with one master node and 96 worker nodes. For LLM inference, we deploy 8 instances of Qwen2.5-7B-Instruct Yang et al. [2024b] using vLLM Kwon et al. [2023] across GPUs to maximize throughput.

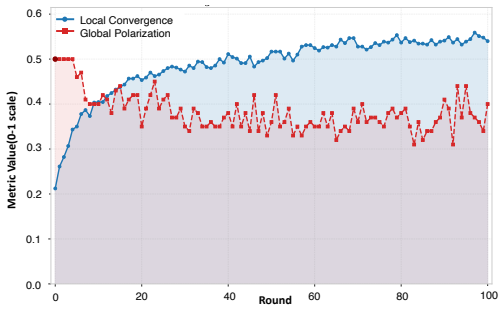

Figure 5: Local Convergence vs. Global Polarization evolution.

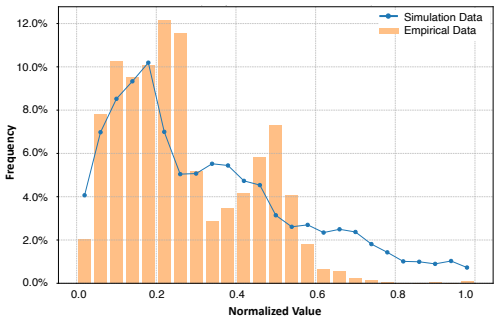

Figure 6: Simulated vs. empirical Brazilian housing price distributions showing strong alignment.

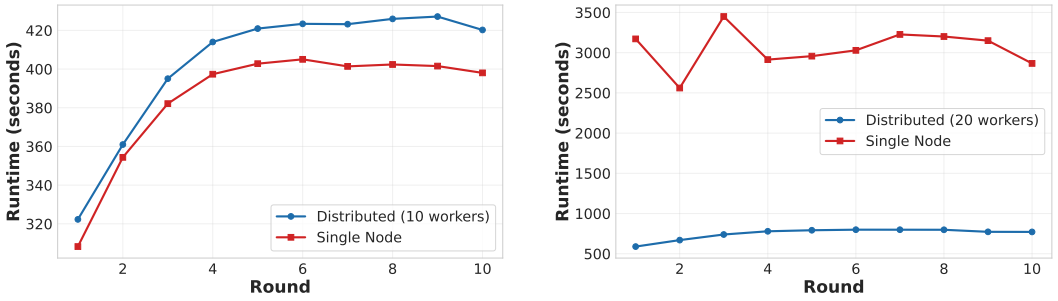

Figure 7: Scalability comparison between single-node and distributed architectures for different agent scales. Left: 5k agents with 10 workers. Right: 10k agents with 20 workers.

We run our simulator for 3 rounds and report the time cost and events count for each round, as well as the average across all rounds.

From Table 3, we can see: our simulator executes 100,000-agent simulations with an average time cost of 6,026 seconds per round, achieving 49.41 events per second throughput and processing over 294,000 interaction events per round. This performance enables comprehensive population-level social dynamics investigations previously constrained by computational limitations.

We further compare single-node vs. distributed performance across different scales (Figure 7):

Small-scale (5k agents): For 5k agents, single-node mode achieves an average runtime of 385.31 seconds, slightly outperforming distributed mode at 403.31 seconds. This indicates that for relatively small workloads, communication overhead in distributed systems may offset the efficiency gains from parallel processing.

Large-scale (10k agents): When the agent count increases to 10k, distributed system advantages become evident. The average runtime of 750.76 seconds is significantly lower than single-node mode's 3,052.61 seconds, demonstrating **nearly 4× performance improvement**. This result strongly validates our distributed architecture's scalability for high-concurrency, large-scale scenarios.

4.4 Evaluation on the AI Social Researcher

The AI Social Researcher is responsible for transforming user-specified research topics into valuable research questions and generating final technical reports. To evaluate its effectiveness, we assess both of these critical phases. We selected eight scenarios for our experiment, one from each domain implemented in our simulator.

To evaluate the quality of the designed scenarios, we assess four key aspects: relevance (alignment with the research topic), fidelity (accuracy in reflecting social phenomena), feasibility (practicality of implementation), and significance (research value). For the generated reports, we evaluate their insight (depth of analysis), structure (logical organization), content (completeness and correctness), and utility (practical applicability). We employ GPT-4o Hurst et al. [2024] to assess the outputs across these dimensions and provide an overall rating in the range of [1, 5].

Table 3: Simulation efficiency results showing time cost and events count. We also report throughput (events per second) and token processing rates.

Round	Time Cost (s)	#Events	Events/s	Input Tokens/s	Output Tokens/s
1	5,266	294,638	55.95	83,257	4,160
2	6,051	294,612	48.69	82,685	3,860
3	6,761	294,639	43.58	82,314	3,545
Average	6,026	294,630	49.41	82,752	3,855

Table 4: Evaluation results on scenario design and report quality. Scenario Design: Rel.=Relevance, Fid.=Fidelity, Fea.=Feasibility, Sig.=Significance, Ovr.=Overall. Report Quality: Ins.=Insight, Str.=Structure, Con.=Content, Uti.=Utility, Ovr.=Overall.

	Scenario Design Quality					Report Quality				
Domain	**Rel.**	**Fid.**	**Fea.**	**Sig.**	**Ovr.**	**Ins.**	**Str.**	**Con.**	**Uti.**	**Ovr.**
Economics	4.00	3.00	5.00	4.00	4.00	4.00	5.00	4.00	4.00	4.00
Sociology	5.00	4.00	5.00	4.00	4.50	3.00	4.00	3.00	3.00	3.00
Politics	3.00	3.00	4.00	4.00	3.50	3.00	4.00	4.00	3.00	3.50
Psychology	5.00	4.00	5.00	4.00	4.50	3.00	4.00	4.00	3.00	3.00
Organization	4.00	3.00	5.00	4.00	4.00	3.00	4.00	3.00	2.00	3.00
Demographics	4.00	3.00	5.00	4.00	4.00	3.00	3.00	3.00	3.00	3.00
Law	4.00	3.00	5.00	4.00	4.00	4.00	4.00	4.00	3.00	4.00
Communication	5.00	4.00	5.00	4.00	4.50	3.00	4.00	4.00	3.00	3.00
Average	4.25	3.38	4.88	4.00	4.13	3.25	4.00	3.63	3.00	3.31

The evaluation results are presented in Table 4. We can see: for the quality of the designed scenarios, our AI social researcher demonstrates strong performance across all metrics, with an average overall score of 4.13 out of 5. Particularly impressive was the system's ability to generate highly feasible simulation designs (average score: 4.88), indicating that the AI researcher can effectively translate conceptual research questions into implementable simulation scenarios. The system also excels in relevance (4.25), ensuring that generated scenarios accurately correspond to the intended research topics.

For the quality of the generated reports, our AI social researcher performs well in structural organization (average: 4.00) and content quality (average: 3.63). This indicates that the system can effectively organize analytical findings into coherent, well-structured reports with reasonable depth and accuracy. The highest-rated reports were for Auction Market Dynamics (Economics) and Court Trial Simulation (Law), both scoring 4.0 overall. The relative weakness in insight (3.25) and utility (3.00) scores highlights potential areas for enhancement. While the AI researcher can competently analyze simulation data and present findings, it could be further improved to generate deeper insights and more actionable recommendations from simulation results. This is particularly evident in scenarios like Labor Market Matching Process, which receives the lowest utility score (2.0).

Overall, the above evaluation demonstrates that our AI Social Researcher can successfully complete the full research cycle—from question formulation to report generation—with solid performance. The system excels particularly in translating research topics into feasible simulation designs, while showing room for improvement in extracting deeper insights from the simulation results.

5 Conclusion

In this paper, we have introduced a novel social simulator called YuLan-OneSim, which is featured in five aspects, that is, code-free scenario construction, comprehensive default scenarios, evolvable simulation, large-scale simulation and AI social researcher. We conduct extensive experiments to demonstrate the effectiveness and efficiency of our simulator. We believe YuLan-OneSim makes a significant step towards the next generation of LLM agent based social simulator. In the future, we plan to incorporate more social theories to regularize agent behaviors in our simulator, and will also incorporate more spatial and multi-modal information to make the simulator more applicable.

References

Flaminio Squazzoni, Wander Jager, and Bruce Edmonds. Social simulation in the social sciences: A brief overview. *Social Science Computer Review*, 32(3):279–294, 2014.

Brian Heath, Raymond Hill, and Frank Ciarallo. A survey of agent-based modeling practices (january 1998 to july 2008). *Journal of Artificial Societies and Social Simulation*, 12(4):9, 2009.

Federico Bianchi and Flaminio Squazzoni. Agent-based models in sociology. *Wiley Interdisciplinary Reviews: Computational Statistics*, 7(4):284–306, 2015.

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

Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, et al. A survey of large language models. *arXiv preprint arXiv:2303.18223*, 1(2), 2023.

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

Jiakai Tang, Heyang Gao, Xuchen Pan, Lei Wang, Haoran Tan, Dawei Gao, Yushuo Chen, Xu Chen, Yankai Lin, Yaliang Li, et al. Gensim: A general social simulation platform with large language model based agents. *arXiv preprint arXiv:2410.04360*, 2024.

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

Jinghua Piao, Yuwei Yan, Jun Zhang, Nian Li, Junbo Yan, Xiaochong Lan, Zhihong Lu, Zhiheng Zheng, Jing Yi Wang, Di Zhou, et al. Agentsociety: Large-scale simulation of llm-driven generative agents advances understanding of human behaviors and society. *arXiv preprint arXiv:2502.08691*, 2025.

Thomas C Schelling. Dynamic models of segregation. *Journal of mathematical sociology*, 1(2): 143–186, 1971.

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

Richard G Palmer, W Brian Arthur, John H Holland, and Blake LeBaron. An artificial stock market. *Artificial Life and Robotics*, 3:27–31, 1999.

Joon Sung Park, Joseph O'Brien, Carrie Jun Cai, Meredith Ringel Morris, Percy Liang, and Michael S Bernstein. Generative agents: Interactive simulacra of human behavior. In *Proceedings of the 36th annual acm symposium on user interface software and technology*, pages 1–22, 2023.

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

Lei Wang, Jingsen Zhang, Hao Yang, Zhi-Yuan Chen, Jiakai Tang, Zeyu Zhang, Xu Chen, Yankai Lin, Hao Sun, Ruihua Song, et al. User behavior simulation with large language model-based agents. *ACM Transactions on Information Systems*, 43(2):1–37, 2025.

An Zhang, Yuxin Chen, Leheng Sheng, Xiang Wang, and Tat-Seng Chua. On generative agents in recommendation. In *Proceedings of the 47th international ACM SIGIR conference on research and development in Information Retrieval*, pages 1807–1817, 2024a.

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

Wenyue Hua, Lizhou Fan, Lingyao Li, Kai Mei, Jianchao Ji, Yingqiang Ge, Libby Hemphill, and Yongfeng Zhang. War and peace (waragent): Large language model-based multi-agent simulation of world wars. *arXiv preprint arXiv:2311.17227*, 2023.

Xinnong Zhang, Jiayu Lin, Libo Sun, Weihong Qi, Yihang Yang, Yue Chen, Hanjia Lyu, Xinyi Mou, Siming Chen, Jiebo Luo, et al. Electionsim: Massive population election simulation powered by large language model driven agents. *arXiv preprint arXiv:2410.20746*, 2024b.

Dawei Gao, Zitao Li, Xuchen Pan, Weirui Kuang, Zhijian Ma, Bingchen Qian, Fei Wei, Wenhao Zhang, Yuexiang Xie, Daoyuan Chen, et al. Agentscope: A flexible yet robust multi-agent platform. *arXiv preprint arXiv:2402.14034*, 2024b.

Xinnong Zhang, Jiayu Lin, Xinyi Mou, Shiyue Yang, Xiawei Liu, Libo Sun, Hanjia Lyu, Yihang Yang, Weihong Qi, Yue Chen, et al. Socioverse: A world model for social simulation powered by llm agents and a pool of 10 million real-world users. *arXiv preprint arXiv:2504.10157*, 2025.

Volker Grimm, Uta Berger, Donald L DeAngelis, J Gary Polhill, Jarl Giske, and Steven F Railsback. The odd protocol: a review and first update. *Ecological modelling*, 221(23):2760–2768, 2010.

Xingwei Wang, Hong Zhao, and Jiakeng Zhu. Grpc: A communication cooperation mechanism in distributed systems. *ACM SIGOPS Operating Systems Review*, 27(3):75–86, 1993.

Bernardo Alves Furtado. Policyspace: agent-based modeling. 2018.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, et al. Qwen2. 5 technical report. *arXiv preprint arXiv:2412.15115*, 2024b.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph Gonzalez, Hao Zhang, and Ion Stoica. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the 29th Symposium on Operating Systems Principles*, pages 611–626, 2023.

Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*, 2024.

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

Xiangru Tang, Anni Zou, Zhuosheng Zhang, Ziming Li, Yilun Zhao, Xingyao Zhang, Arman Cohan, and Mark Gerstein. Medagents: Large language models as collaborators for zero-shot medical reasoning. *arXiv preprint arXiv:2311.10537*, 2023.

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

Fengli Xu, Jun Zhang, Chen Gao, Jie Feng, and Yong Li. Urban generative intelligence (ugi): A foundational platform for agents in embodied city environment. *arXiv preprint arXiv:2312.11813*, 2023.

Junkai Li, Yunghwei Lai, Weitao Li, Jingyi Ren, Meng Zhang, Xinhui Kang, Siyu Wang, Peng Li, Ya-Qin Zhang, Weizhi Ma, et al. Agent hospital: A simulacrum of hospital with evolvable medical agents. *arXiv preprint arXiv:2405.02957*, 2024.

Yangyang Yu, Zhiyuan Yao, Haohang Li, Zhiyang Deng, Yuechen Jiang, Yupeng Cao, Zhi Chen, Jordan Suchow, Zhenyu Cui, Rong Liu, et al. Fincon: A synthesized llm multi-agent system with conceptual verbal reinforcement for enhanced financial decision making. *Advances in Neural Information Processing Systems*, 37:137010–137045, 2024.

Zeyu Zhang, Jianxun Lian, Chen Ma, Yaning Qu, Ye Luo, Lei Wang, Rui Li, Xu Chen, Yankai Lin, Le Wu, et al. Trendsim: Simulating trending topics in social media under poisoning attacks with llm-based multi-agent system. *arXiv preprint arXiv:2412.12196*, 2024c.

Table 5: Comparison between YuLan-OneSim and previous works. In the "# Agents" column, we present the number of agents as reported in the original papers' experiments. In the "# Environments (Env)" column, "-" means that the simulator does not provide environments for the agents to take actions, for example, the simulator is purely based on conversations. The symbol ◯ indicates that while SOTOPIA-S4 allows users to configure simulation scenarios using natural language, it leverages user inputs as LLM prompts instead of converting them into executable simulation code. We use different colors to represent various stages in the field of LLM agent based social simulation.

Date	Name & Reference	Automatic Coding	# Agents	# Env	AI Researcher	Feedback Compatible
2023.04	Generative Agents Park et al. [2023]	×	10-100	1	×	×
2023.06	RecAgent Wang et al. [2025]	×	1000-10000	3	×	×
2023.07	S^3 Gao et al. [2023]	×	1000-10000	1	×	×
2023.10	Agent4Rec Zhang et al. [2024a]	×	1000-10000	1	×	×
2023.10	EconAgent Li et al. [2023]	×	10-100	1	×	×
2023.10	Acerbi et al. Acerbi and Stubbersfield [2023]	×	10-100	1	×	×
2023.11	WarAgent Hua et al. [2023]	×	10-100	1	×	×
2023.11	MedAgents Tang et al. [2023]	×	10-100	1	×	×
2023.11	Chuang et al. Chuang et al. [2023]	×	10-100	1	×	×
2023.12	UGI Xu et al. [2023]	×	100-1000	5	×	×
2024.05	Agent Hospital Li et al. [2024]	×	10-100	1	×	×
2024.07	FinCon Yu et al. [2024]	×	10-100	1	×	×
2024.10	TrendSim Zhang et al. [2024c]	×	1000-10000	1	×	×
2024.10	ElectionSim Zhang et al. [2024b]	×	1000-10000	1	×	×
2024.10	GenSim Tang et al. [2024]	×	$\geq$ 10000	4	×	✓
2024.11	OASIS Yang et al. [2024a]	×	$\geq$ 10000	2	×	×
2025.02	AgentSociety Piao et al. [2025]	×	$\geq$ 10000	4	×	×
2025.04	SocioVerse Zhang et al. [2025]	×	$\geq$ 10000	3	×	×
2025.04	SOTOPIA-S4 Zhou et al. [2025]	◯	100-1000	-	×	×
2025.05	YuLan-OneSim	✓	$\geq$ 10000	50	✓	✓

Xuhui Zhou, Zhe Su, Sophie Feng, Jiaxu Zhou, Jen-tse Huang, Hsien-Te Kao, Spencer Lynch, Svitlana Volkova, Tongshuang Sherry Wu, Anita Woolley, et al. Sotopia-s4: a user-friendly system for flexible, customizable, and large-scale social simulation. *arXiv preprint arXiv:2504.16122*, 2025.

Table 6: The default scenarios implemented in our simulator.

Domain	# Scenarios	Scenario Names
Economics	6	Auction market dynamics, Bank reserves, Cash flow, Collective action problem, Customer satisfaction and loyalty model, Rational choice theory
Sociology	6	Cultural capital theory, Norm formation theory, Social capital theory, Social relations theory, Social stratification network, Theory of planned behavior
Political	6	Electoral polarization system, Public opinion polling, Rebellion, Selective exposure theory, Simple policy implementation model, Voting
Psychology	6	Antisocial personality theory, Attribution theory, Cognitive dissonance theory, Conformity behavior model, Emotional contagion model, Metacognition theory
Organization	6	Decision theory, Hawthorne studies, Hierarchy of needs, Labor market matching process, Organizational change and adaptation theory, Scientific management theory
Demographics	7	Community health mobilization theory, Epidemic transmission network, Health belief model, Health inequality, Life course theory, Reciprocal altruism theory, SIR model
Law	7	Case law model, Court trial simulation, Self defense and excessive defense, Social contract theory, Tort law and compensation, Unjust enrichment, Work hours and overtime in labor law
Communication	6	Agenda setting theory, Cultural globalization, Diffusion of innovations, Information cascade and silence, Two step flow model, Uses and gratifications theory

Table 7: Criteria for evaluating the behavior graph

Score	Description
5	Fully accurately identifies all core steps. Extracted steps are completely consistent with the original description. Step sequence matches exactly 80–100% of key steps are correctly identified.
4	Accurately identifies the vast majority of core steps. Extracted steps are highly consistent with the original description. Step sequence is mostly correct 60–80% of key steps are correctly identified.
3	Identifies most core steps. Extracted steps are generally consistent with the original description. Some inaccuracies in step sequence 40–60% of key steps are correctly identified.
2	Identifies some core steps but with significant omissions. Extracted steps are partially inconsistent with the original description. Step sequence is mostly disordered 20–40% of key steps are correctly identified.
1	Fails to identify core steps of the workflow. Extracted steps are severely inconsistent with the original description. Step sequence is completely disordered. Fewer than 20% of key steps are correctly identified.

A Comparison to Related Work

B Default Scenario Details

C Evaluation Criteria and Error Analysis

D Limitations

The primary limitations of YuLan-OneSim stem from a few major challenges:

Table 8: Criteria for evaluating the simulation code

Score	Description
5	The generated code fully matches the workflow. No syntax errors. Runs flawlessly. Accurately reflects the workflow logic. Code quality: 80–100%. Clear code structure with good readability and maintainability. No manual modification required
4	The generated code highly conforms to the workflow. Virtually no syntax errors. Runs smoothly. Accurately reflects the workflow logic. Code quality: 60–80%. Requires minor manual adjustments to fix a few edge cases or detail issues.
3	The generated code generally conforms to the workflow. Contains some syntax or logic errors. Partially executable. Roughly reflects the workflow logic. Code quality: 40–60%. Requires manual revision of key algorithms and logic, fixing multiple code defects, and significant refactoring by developers
2	The generated code is partially related to the workflow. Contains severe syntax and logic errors. Difficult to execute. Reflects only a small portion of the workflow logic. Code quality: 20–40%. Requires rewriting 60–90% of the code. Major refactoring of core logic is needed. Multiple critical algorithm and business logic issues exist.
1	The generated code is unrelated to the workflow. Severe syntax errors. Cannot run at all. Does not reflect any workflow logic. Code quality: below 20%. Requires rewriting over 90% of the code. Almost unusable, essentially starting from scratch. Contains serious syntax and logical flaws. Developers need to completely redesign the code architecture.

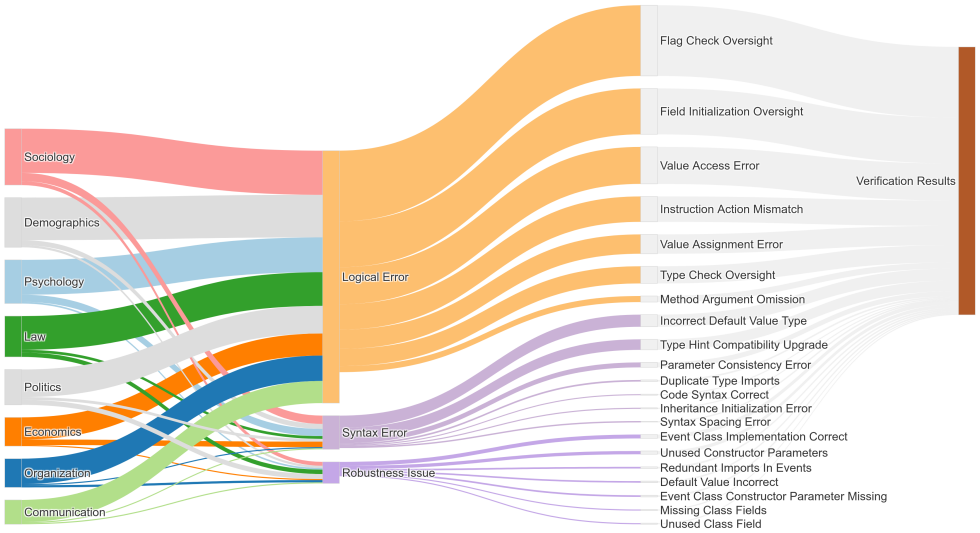

Figure 8: Illustration on the distribution of various error types—including logical errors, syntax errors, and robustness issues—across different domains.

(1) **LLM Inherent Reliability Issues:** Due to the inherent limitations of LLMs (e.g., hallucination and bias), errors introduced early in the simulation can accumulate and amplify as the process progresses, even with the feedback-driven evolving subsystem in place.

(2) **AI Social Researcher's Insight Depth:** As shown in our evaluation (Table 5), while the AI Social Researcher excels at report structure and feasibility, it currently demonstrates relative weakness in generating deeper insights (3.25 average) and high utility/actionable recommendations (3.00 average) from the simulation results.

(3) **Simplification of Real-World Factors:** In complex scenarios like the Brazilian real estate market simulation, the simplified representation of real-world factors may lead to minor discrepancies when compared to empirical data.

(4) **Future Scope:** Our current version lacks built-in integration of fine-grained spatial and multi-modal information, which is a planned future enhancement to broaden the simulator's applicability.

E Broader Impact

Our social simulator, YuLan-OneSim, primarily aims to offer a positive impact by advancing social simulation and democratizing social science research. It provides a controllable, reproducible, and accessible "social science gym" for researchers to investigate complex social phenomena without the need for extensive programming skills. By automatically completing the social science research loop, it significantly accelerates the research process from ideation to report generation.

We acknowledge the potential for misuse, as with any generative AI technology. However, since YuLan-OneSim is a research platform focused on simulating social dynamics and validating social theories, we believe the risks associated with its academic release are low. We encourage the community to use this platform responsibly and for ethical, academic purposes, focusing on its potential to advance our understanding of social systems.

F Crowdsourcing Details

The evaluation of the automatically generated scenarios in Section 4.1 involved a limited crowdsourcing effort with human experts. We engaged three doctoral students to serve as human annotators for scoring the scenarios based on the criteria outlined in Appendix C. All annotators were compensated at a rate of $15 per hour for their time and expertise. The evaluation process was conducted with the consent of the participants.