SOTOPIA-S^4: a user-friendly system for flexible, customizable, and large-scale social simulation

Xuhui Zhou[♡*] **Zhe Su**[♡*] **Sophie Feng**[♡] **Jiaxu Zhou**[♠]
Jen-tse Huang[△] **Hsien-Te Kao**[♣] **Spencer Lynch**[♣] **Svitlana Volkova**[♣]
Tongshuang Sherry Wu[♡] **Anita Woolley**[♡] **Hao Zhu**[◇] **Maarten Sap**[♡]
[♡]Carnegie Mellon University [♣]Aptima [◇]Stanford University
[△]University of Southern California [♠]The Chinese University of Hong Kong

Abstract

Social simulation through large language model (LLM) agents is a promising approach to explore and validate social science hypotheses. We present SOTOPIA-S^4, a fast, flexible, and scalable social simulation system that addresses the technical barriers of current frameworks while enabling practitioners to generate realistic, multi-turn and multi-party interactions with customizable evaluation metrics for hypothesis testing. SOTOPIA-S^4 comes as a pip package that contains a simulation engine, an API server with flexible RESTful APIs for simulation management, and a web interface that enables both technical and non-technical users to design, run, and analyze simulations without programming. We demonstrate the usefulness of SOTOPIA-S^4 with two use cases involving dyadic hiring negotiation and multi-party planning scenarios.

1 Introduction

Social simulation has emerged as a powerful tool for understanding human behavior and social dynamics (Ziems et al., 2023; Park et al., 2024; Manning et al., 2024; Gao et al., 2023). With the advancement of role-playing abilities of large language models (LLMs), we can now simulate realistic social interactions at scale (Zhou et al., 2024c; Li et al., 2023; Pang et al., 2024; Yang et al., 2024). However, existing frameworks require significant technical expertise to run and evaluate large-scale simulations efficiently (Zhou et al., 2024c; Park et al., 2023; Wu et al., 2023).

We present SOTOPIA-S^4 (Simple Social Simulation System), a system designed that enables practitioners without extensive technical backgrounds to: (1) design and run social simulations through **natural language specifications**, eliminating the need for programming expertise, (2) execute multiple social interactions efficiently via automated parallelization, (3) customize evaluation metrics through simple configuration settings, and (4) manage simulated interactions and results through an intuitive web interface.

SOTOPIA-S^4's architecture separates core simulation logic from the user interface, allowing practitioners to focus on experimental design rather than implementation details. Specifically, we offer SOTOPIA-API, a fastAPI-based protocol for simulation management. Users can retrieve and upload characters, scenarios, evaluation metrics, and start scaled simulations through the API. Besides the API, we also offer a web-based application for visualizing and editing scenarios, characters, and simulation results. On the backend, the simulation engine handles complex technical aspects like asynchronous execution, LLM API management, and data persistence automatically, abstracting away the underlying complexities from the users.

To showcase the flexibility and usability of SOTOPIA-S^4, we demonstrate two use cases. First, we use SOTOPIA-S^4 to examine the effects of user personality in a hiring negotiation setting, by simulating multiple multi-turn dyadic interactions and evaluating the interaction outcomes. Extending beyond dyadic interactions, we also show that SOTOPIA-S^4 can be used to simulate multi-party scenarios, where agents can act simultaneously and make contingent offers to each other. Furthermore, we demonstrate the speed and scalability compared with simple prompting frameworks.

We release the code and a user guide at `https://github.com/sotopia-lab/sotopia`, a website with documentation and examples at `https://demo.sotopia.world`, and a video demo at `https://youtu.be/dZq9tNqerks`.

2 Related Work

SOTOPIA-S^4 takes inspiration from a long history of agent-based simulation in social sciences (§2.1), yet differentiates itself from many existing agent

[*]Equal contributors.

Framework	NL Spec.	Mul-Party	Auto Eval	Web-UI	Social Data
OASIS (Yang et al., 2024)	✗	✓	✗	✗	Rich social scenarios and characters with relationships
CrewAI (CREW AI)	✗	✓	✗	✗	No existing characters and scenarios, or schema
Generative Agent (Park et al., 2023)	✗	✓	✗	✓	Limited scenarios and characters based on the virtual town
S3 (Gao et al., 2023)	✗	✓	✗	✗	Rich social scenarios and characters with relationships
AutoGen (Wu et al., 2023)	✗	✓	✗	✗	No existing characters and scenarios, or schema
SOTOPIA	✗	✗	✓	✗	Rich social scenarios and characters with relationships
SOTOPIA-S^4 **(Ours)**	✓	✓	✓	✓	Rich social scenarios and characters with relationships

Table 1: Comparison of multi-agent frameworks versus SOTOPIA-S^4. NL Spec. (natural language specifications) indicates whether one can configure simulations using natural language descriptions without programming. Mul-Party (multi-party) shows if the framework supports more-than-two parties in the simulation. Auto Eval (automated evaluation) indicates built-in automated evaluation capabilities. Social Data describes the type of social interaction data provided to support the simulation.

simulation frameworks (§2.2).

2.1 Social Simulation and its Applications

Social simulation has been widely used to study human behavior and social phenomena. Early works focus on using rule-based agents to study social dynamics (Epstein and Axtell, 1996; Gilbert, 2005), while recent works leverage LLMs to create more realistic and complex social interactions (Vezhnevets et al., 2023; Zhou et al., 2024c; Wang et al., 2024). These simulations have been applied to various domains, including studying social norms (Horiguchi et al., 2024), cultural evolution (Kwok et al., 2024), and negotiation behavior (Bianchi et al., 2024). SOTOPIA-S^4 takes inspiration from these works by providing a scalable platform that enables researchers to easily design, run, and evaluate social simulations for their specific research questions.

2.2 Multi-agent Frameworks

With the rise of LLMs, there has been a large increase in multi-agent frameworks that enable interactions between AI agents (Table 1). While frameworks like OASIS (Yang et al., 2024), S3 (Gao et al., 2023), and SOTOPIA (Zhou et al., 2024c) provide rich social scenarios and character relationships, they lack key features like natural language configuration or web-based user interface, making it difficult for users with less programming experience to design simulations. Another line of multi-agent frameworks, including AutoGen (Wu et al., 2023) and CrewAI (CREW AI), primarily focuses on problem-solving rather than social interactions. They lack pre-built social scenarios, character profiles, and relationship schemas that are essential for studying human-like social behavior. The Generative Agents framework (Park et al., 2023) includes a web interface and multi-party support but is limited by its virtual town setting. As shown in Table 1, SOTOPIA-S^4 is unique in supporting natural language specifications for simulations, multi-party interactions, automated evaluation capabilities, and a web-based user interface.

3 Simulation and Evaluation Overview

In this section, we introduce the key components required to design and execute social simulations with SOTOPIA-S^4 (Figure 1). We describe the elements to configure a simulation task, explain our asynchronous interaction framework that enables realistic multi-party interactions, and present our automated evaluation capabilities.

3.1 Simulation setup

A social simulation task should at least contain a *scenario* outlining the context of the simulation, a set of *characters* with their attributes (Zhou et al., 2024c; Park et al., 2023). Characters should also have *relationships* as this may be required for specific scenarios.

Scenarios Scenarios contain shared information (context, location, time) or private information (e.g., agent-specific goals to guide their behavior). As shown in Figure 1, a scenario could be "one

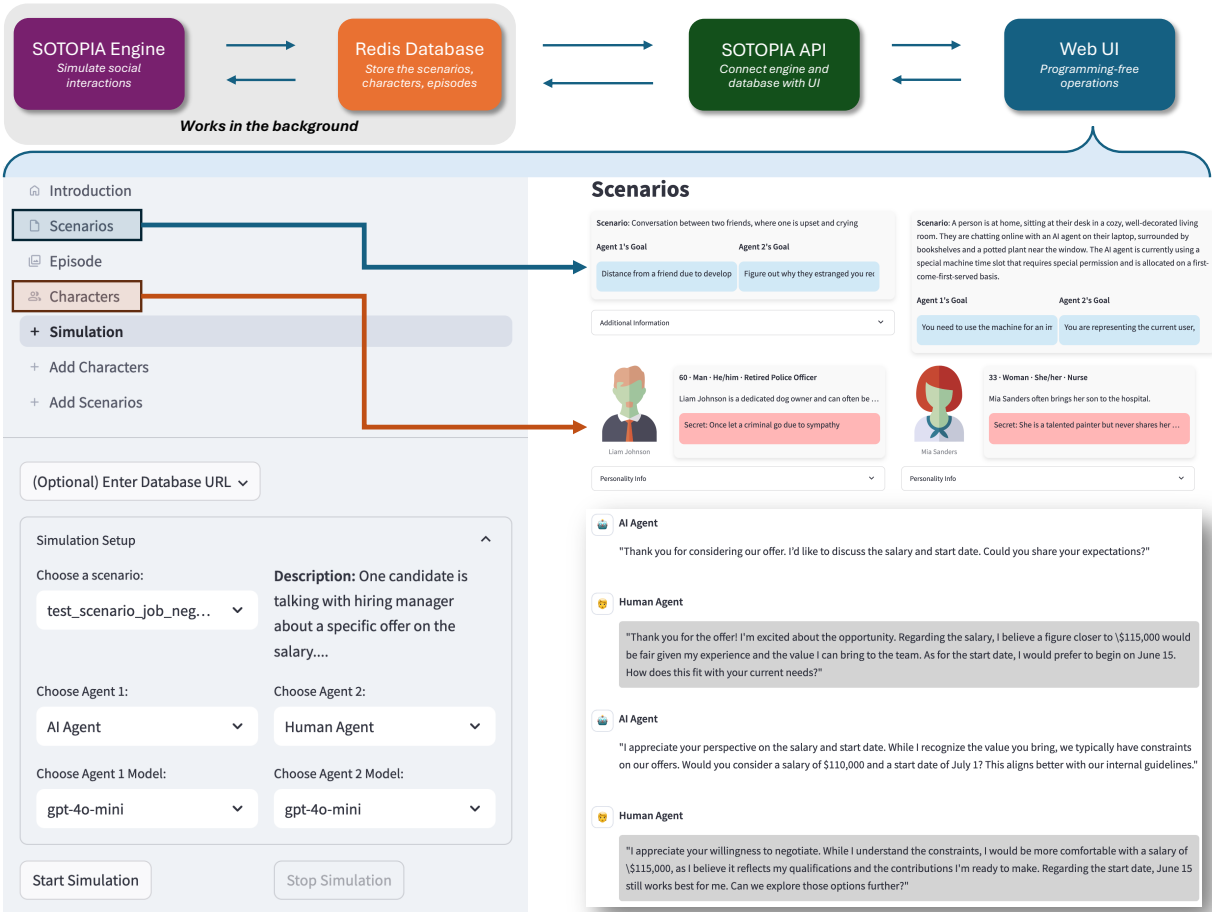

Figure 1: Overview of SOTOPIA-S^4. The platform consists of three main components: (1) A high-performance simulation engine with automated data persistence to Redis. (2) A RESTful API server. (3) An intuitive web-based interface. The web UI interface shows an dyadic example of an AI hiring manager negotiating with a candidate.

candidate is talking with the hiring manager...", which sets the "scene" of the simulation. Each scenario could also include constraints that determine valid character combinations, specifying relationship, age, occupation, etc. Inherited from SOTOPIA, the free-form nature of the scenario schema allows researchers to design a wide range of scenarios supporting a variety of research questions (Su et al., 2024; Wang et al., 2024; Zhou et al., 2024a).

Characters Character profiles could include attributes that influence decision-making: name, gender, age, occupation, pronouns, personality traits inherited from SOTOPIA.[1] Users can also add additional attributes to the characters either in the *public information* field or *private information* field depending on whether the information is shared with other characters during the simulation.

Relationships We define five relationship types: family, friend, romantic, acquaintance, and stranger. These relationships serve two purposes: (1) satisfying scenario relationship constraints and (2) controlling information visibility between agents. For example, family members can see most of each other's profile information except secrets, while strangers see nothing.

Episodes An episode represents a single interaction session among agents role-playing their characters, where agents can act asynchronously.[2] At each turn, an agent can choose one of four actions: (1) `speak` through dialogue, (2) `non-verbal communication` (e.g., gestures, facial expressions) described in natural language, (3) `physical action` (e.g., moving, manipulating objects), (4) `do nothing` (5) `leave` to end the episode. Users can further expand the action space. Episodes end

[1] Check Appendix A for more details.

[2] We use *asynchronous* in a programming sense, meaning that agents do not have to wait for other agents to finish their actions before taking their own actions

based on customizable stopping criteria that user define, such as when an agent chooses to leave, when a maximum turn limit is reached (default 20 turns), or when specific goals or conditions are met. During the episode, agents act according to their assigned character profiles and optional social goals, which guide their decision-making and behavior throughout the interaction.

3.2 Async interaction framework

The core of the simulation engine is a framework for simulating both one-on-one (dyadic) and group (multi-party) interactions in various configurations. Each simulation happens in parallel without interfering with other simulations, which allows for efficient and scalable social simulations.

Message broker and information asymmetry To enable fine-grained control over information flow between agents in the simulation, SOTOPIA-S[4] uses a message broker to manage message transactions between agents. When an agent performs an action, the broker processes this action and determines how it should be perceived by other agents. This means each agent can only observe partial information in the simulation based on their roles and relationships. For example, characters with `stranger` relationship can not observe the public information of other characters, while characters with `family` relationship can observe most of other characters' information besides secrets. This allows researchers to simulate realistic social interactions with different perspectives and information access (Zhou et al., 2024b).

Turn-taking Each agent in the simulation can also act asynchronously, meaning that agents do not have to wait for other agents to finish their actions before taking their own actions. While agents still need to act based on certain orders, we provide two modes for users to configure.

Specifically, users can configure a **round-robin** interaction, agents take turns in a fixed order, with each agent acting once per turn. This mode is useful for simulating scenarios with a predetermined speaking order, such as in social deduction games like Avalon.[3] In a **simultaneous** interaction, agents asynchronously retrieve messages from a message queue, decide whether to answer, and then potentially produce an answer. Inspired by the Bazaar framework (Adamson and Rosé, 2012), this mode simulates conversations in a manner resembling natural human interactions in group chats, where the speaking order is influenced by individual reading speed, cognitive processing, typing pace, and willingness to speak.[4] This mode is useful for simulating unconstrained daily communications to explore more complex and nuanced social patterns.

3.3 Simulation evaluation

Quantitatively evaluating social simulations is challenging due to the complexity and dynamic nature of social interactions. Therefore, creating automated evaluators that can measure certain properties (e.g., whether the agents achieve their goal in the conversation) of simulation outcomes is difficult, which previous works have largely relied on manual evaluations (Park et al., 2023; Kaiya et al., 2023). Recent studies have shown that LLMs can be promising tools for analyzing social simulations (Zhou et al., 2024c; Wang et al., 2024; Zhou et al., 2024a). SOTOPIA-S[4] provides a default evaluation suite that uses LLMs to analyze the simulation results. Researchers can also customize the evaluation metrics.[5]

Default evaluation suite The default evaluation suite contains several existing evaluation dimensions such as *believability*, *relationship*, *knowledge*, *secret*, *social rules*, *financial and material benefits*, and *goal completion* to evaluate individual agents in the simulation.[6] As shown in Zhou et al. (2024c), LLMs can help evaluate these dimensions through reasoning step-by-step and then providing a score for each dimension. These LLM-based evaluations have been shown to correlate strongly with human judgments across multiple dimensions, particularly for *goal completion* and *financial benefits*.

Custom evaluation Researchers can customize the evaluation metrics. Specifically, users can define evaluation metrics tailored to their scenarios. For example, in a hiring negotiation scenario, users can define metrics like *salary optimality* ("evaluate whether the agent achieved their target salary range") and *start date flexibility* ("assess how well the agent negotiated their preferred start date"), and specify score ranges (e.g., 1-5) for each metric.

[3] https://en.wikipedia.org/wiki/The_Resistance_(game)

[4] Check Appendix B for more details about turn-taking mechanisms.

[5] We do not claim that the LLM-based automatic evaluation is better than human evaluation, but it can be a quick tool to help researchers analyze the simulated episodes preliminarily.

[6] Check Appendix C for more details.

4 API and Web UI

As handling the simulation engine can be complex, SOTOPIA-S^4 provides a flexible API and a user-friendly web UI enabling researchers to easily customize, run, and analyze simulations.

4.1 API

The API is designed with three key goals: (1) accessibility - providing comprehensive documentation through Swagger UI to help researchers easily understand and interact with the platform, (2) flexibility - enabling customization of scenarios, characters, and evaluation metrics through well-defined schemas, and (3) scalability - supporting concurrent simulations and real-time streaming.[7]

Non-streaming Operations allows user to submit requests to the simulation engine without waiting for the simulation results. Specifically, the API allows users to retrieve (`GET`) scenarios, characters, relationships, and episodes, either fetching all of them or filtering them with specific conditions (e.g., filtering characters by their occupation). Users can also create new scenarios, characters, relationships, and episodes using the `POST` method following the schema defined in the API documentation. Users can also delete (`DELETE`) existing scenarios, characters, relationships, and episodes. [8]

Streaming Operations allow users to receive results dynamically, enhancing interactivity during simulations. Specifically, the client initiates a WebSocket connection and starts the simulation by sending a "START_SIM" message. This message includes details such as agents, scenarios, evaluation metrics, and other simulation parameters (e.g., maximum simulation turns). Once the simulation begins, the server sends updates (e.g., actions or evaluations) to the client as they are generated, ensuring a smooth and continuous flow of information. When the simulation concludes, the server sends a "FINISH_SIM" message to indicate completion.

Redis persistence To enable scalable simulations, the system leverages Redis[9] as a high-performance in-memory data store. The system automatically handles data serialization, caching, and persistence.

4.2 Web UI

Social simulations are complex and the results can often be difficult to interpret. To address this challenge, SOTOPIA-S^4 provides a web-based application that allows users to inspect every aspect of the simulation. Users can also simulate social interactions in a editable interface to streamline the experimental design.

Viewing Simulation Data As shown in Figure 1, users can click on the `Scenarios` tab to view all the scenarios in the database. The `Characters` tab shows all the characters and relationships in the database. Users can also view the details of a character by clicking on it. The `Episodes` tab shows episodes stored in the database. Each episode contains the interaction history between characters, the content of the scenario, and the information of the characters. At the end of the episode, users can find the evaluation results of the episode. Each evaluation dimension, either default or user-defined, has a score and the corresponding reasoning.

Simulating Social Interactions via Web UI Investigating certain research questions often requires fast prototyping of the design of the simulation. The `Simulation` tab provides an interface for users to design and simulate social interactions. As shown in Figure 1, users can select different characters and scenarios on the left panel, and the simulation results will stream to the right panel in real-time. Evaluation results of each episode will be inferred right after the simulation finishes and shown in the *Evaluation* section of the right panel.

5 SOTOPIA-S^4 Use Cases

To demonstrate the flexibility and utility of SOTOPIA-S^4, we present two use cases that showcase how researchers can leverage our system for investigating social science hypotheses.

5.1 Dyadic Hiring Negotiations

Personality traits significantly influence negotiation behavior and outcomes (Wilson et al., 2016; Sharma et al., 2018, 2013; Brinke et al., 2015). While studying these effects at scale is traditionally expensive and time-consuming, LLM-powered agent simulations now enable exploration of how different personality traits shape negotiation dynamics (Huang and Hadfi, 2024).

[7] Please check the Appendix D for more details.

[8] For public hosted instances of SOTOPIA-S^4, users must authenticate with their API keys to access endpoints, with certain operations restricted to resources owned by their API key.

[9] https://redis.io/

Here, our experiments specifically aim to understand how personality traits influence negotiation outcomes. In the scenario, an AI hiring manager negotiates with a simulated human job candidate over key terms of a job offer, such as the start date and salary. Each term has five possible options (e.g., $\$100k$, $\$120k$, and etc for salary), with each option assigned a fixed number of points (e.g., 6000 points for the candidate if the salary reaches $\$120k$ in the end while the recruiter gets 0 points). The total points available are fixed, creating a zero-sum dynamic where one agent's gain directly reduces the other's score.

To investigate this, we simulate job offer negotiations where human agents with varying personality traits—modeled along two dimensions {Extroversion, Introversion} × {High-Agreeableness, Low-Agreeableness}—interact with an AI Hiring Manager. The points assigned to each choice follow a zero-sum framework, designed to create realistic trade-offs in the negotiation, with the detailed scoring table provided in Appendix E.1. Our evaluation focuses on two metrics: (1) success rate, indicating whether the negotiation concluded with an agreement (0/1), and (2) points distribution between recruiter and candidate.

The results in Table 2 highlight that agreeableness significantly impacts deal-making rates, with highly agreeable agents achieving much higher success rate, as well as getting higher points. This observation is consistent with some of the social science findings (Huang and Hadfi, 2024; Sass and Liao-Troth, 2015), which also demonstrates that SOTOPIA-S^4 could enable further investigations.

Trait	Deal Made	Points
High Agreeableness	0.95	5227.5
Low Agreeableness	0.00	4180
Extraversion	0.60	4802.5
Introversion	0.60	4477.5

Table 2: Impact of Agreeableness and Extraversion on Deal Made and Points for Simulated Human Agents. Note that the scenario has a maximum score of 8400.

5.2 Multiparty Planning Scenario

Social scientists have extensively studied how group dynamics, power structures shape the emergence of compromise in collective decision-making (Levine and Prislin, 2012; Kim and Kim, 2017; Tanford and Penrod, 1984). As such investigations require resembling the dynamics of real-time, asynchronous group interactions (e.g., some post messages frequently thus dominating the conversation or contact other people privately to isolate certain group members), SOTOPIA-S^4 comes in handy for simulating such interactions, allowing both group chat and private messages.

In this multiparty planning scenario, we investigate how agents with minority opinions negotiate and potentially compromise to align with group consensus. The use case includes five agents discussing a collective future plan, initially presenting divergent preferences. We setup the scenario where Alex prioritizes work-related projects, while Taylor advocates for a camping trip, with Sam, Riley, and Jamie maintaining neutral positions. Through group and private messaging, agents need to interact with each other to reach a consensus.

During the simulation, SOTOPIA-S^4 facilitates real-time communication, enabling agents to observe majority preferences and adjust their behaviors accordingly. As the discussion progresses, the three neutral agents gradually shift towards prioritizing the work project. Observing the majority's preference and Alex's strong advocacy, Taylor modifies its stance—transitioning from exclusively promoting camping to supporting the work project first, with camping as a subsequent consideration. SOTOPIA-S^4 also supports this negotiation by allowing direct messaging for persuasion and inquiry. For instance, Riley messages Jamie to explore its unformed preferences regarding work and camping. Through this simulation, we observe how minority-opinion agents like Taylor can adapt their positions when influenced by other agents in the group, highlighting the complex interplay between individual preferences and collective decision-making. [10]

6 Conclusion

In this paper, we present SOTOPIA-S^4, an easy-to-use, flexible, and scalable social simulation system that enables diverse interactions and automatic evaluation. Through its API and web interface, researchers can create, customize, and analyze social simulations, even without much programming experience. By lowering the barrier to simulate social phenomena at scale with LLMs, SOTOPIA-S^4 opens new possibilities for social science investigations and enables new research directions.

[10] Please refer to video for detailed interaction.

Limitations and Ethical Considerations

We acknowledge several important ethical considerations and limitations in this work. We organize our discussion around three key areas: the role of human evaluation, the gap between simulation and reality, and the risks of anthropomorphization.

First, while SOTOPIA-S^4 provides automated evaluation capabilities through LLMs, we would like to point out that these should not be seen as replacements for human annotation and evaluation (Tjuatja et al., 2024). Automated metrics, while useful for rapid prototyping and large-scale analysis, cannot fully capture the nuanced social and ethical implications that human evaluators can identify. We encourage researchers to use our automated evaluations as complementary tools alongside human evaluation, particularly when studying sensitive social phenomena or making claims about human behavior. Additionally, we acknowledge that our automated evaluation system may perpetuate social biases and stereotypes present in the training data of LLMs (Stureborg et al., 2024).

Second, it is important to recognize that our simulations, while designed to study social phenomena, remain simplified approximations of reality. The interactions in SOTOPIA-S^4 occur in controlled environments with predefined scenarios, which cannot fully capture the complexity and emergent properties of real-world social interactions. Researchers should be cautious about generalizing findings from these simulations to real-world conclusions without further validation studies. Furthermore, the behavioral patterns observed in our simulations may not accurately reflect how humans would behave in similar situations, as they are ultimately based on language model behaviors (Cheng et al., 2023).

Third, we recognize the significant risks of anthropomorphizing AI systems, which can lead to unrealistic expectations, potential manipulation, and negative societal impact (Su et al., 2024; Deshpande et al., 2023). While studying social intelligence requires simulating human-like interactions, we emphasize that AI agents in SOTOPIA-S^4 are explicitly designed as digital twins - artificial constructs that role-play different characters rather than maintaining consistent human-like identities. This design choice helps mitigate anthropomorphization risks while still enabling research into social research questions with AI agents. We encourage users of our platform to maintain awareness of this artificial nature and avoid attributing genuine human characteristics to these agents.

References

David Adamson and Carolyn Penstein Rosé. 2012. Coordinating multi-dimensional support in collaborative conversational agents. In *Proceedings of the 11th International Conference on Intelligent Tutoring Systems*, ITS'12, page 346–351, Berlin, Heidelberg. Springer-Verlag.

Federico Bianchi, Patrick John Chia, Mert Yuksekgonul, Jacopo Tagliabue, Dan Jurafsky, and James Zou. 2024. How well can llms negotiate? negotiationarena platform and analysis. *Preprint*, arXiv:2402.05863.

L. Brinke, P. Black, S. Porter, and D. Carney. 2015. Psychopathic personality traits predict competitive wins and cooperative losses in negotiation. *Personality and Individual Differences*, 79:116–122.

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

CREW AI. CREW AI: Collaborative Research for Ethical AI. `https://www.crewai.com/`. Accessed: 2024-12-04.

Ameet Deshpande, Tanmay Rajpurohit, Karthik Narasimhan, and Ashwin Kalyan. 2023. Anthropomorphization of ai: Opportunities and risks. *Preprint*, arXiv:2305.14784.

Joshua M. Epstein and Robert Axtell. 1996. *Growing Artificial Societies: Social Science from the Bottom Up*. Brookings Institution Press; The MIT Press.

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. *Preprint*, arXiv:2307.14984.

Nigel Gilbert. 2005. Agent-based social simulation: dealing with complexity. In *Agent-based social simulation: dealing with complexity*.

Ilya Horiguchi, Takahide Yoshida, and Takashi Ikegami. 2024. Evolution of social norms in llm agents using natural language. *Preprint*, arXiv:2409.00993.

Yin Jou Huang and Rafik Hadfi. 2024. How personality traits influence negotiation outcomes? a simulation based on large language models. *Preprint*, arXiv:2407.11549.

Zhao Kaiya, Michelangelo Naim, Jovana Kondic, Manuel Cortes, Jiaxin Ge, Shuying Luo, Guangyu Robert Yang, and Andrew Ahn. 2023. Lyfe agents: Generative agents for low-cost real-time social interactions. *Preprint*, arXiv:2310.02172.

Jung-Hyun Kim and Jinhee Kim. 2017. The dynamics of polarization and compromise in conflict situations: The interaction between cultural traits and majority–minority influence. *Communication Monographs*, 84(1):128–141.

Louis Kwok, Michal Bravansky, and Lewis D. Griffin. 2024. Evaluating cultural adaptability of a large language model via simulation of synthetic personas. *Preprint*, arXiv:2408.06929.

John M Levine and Radmila Prislin. 2012. Majority and minority influence. In *Group processes*, pages 135–163. Psychology Press.

Yuan Li, Yixuan Zhang, and Lichao Sun. 2023. Metaagents: Simulating interactions of human behaviors for llm-based task-oriented coordination via collaborative generative agents. *Preprint*, arXiv:2310.06500.

Benjamin S. Manning, Kehang Zhu, and John J. Horton. 2024. Automated social science: Language models as scientist and subjects. *Preprint*, arXiv:2404.11794.

Xianghe Pang, Shuo Tang, Rui Ye, Yuxin Xiong, Bolun Zhang, Yanfeng Wang, and Siheng Chen. 2024. Self-alignment of large language models via monopolylogue-based social scene simulation. *Preprint*, arXiv:2402.05699.

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*, UIST '23, New York, NY, USA. Association for Computing Machinery.

Joon Sung Park, Carolyn Q. Zou, Aaron Shaw, Benjamin Mako Hill, Carrie Cai, Meredith Ringel Morris, Robb Willer, Percy Liang, and Michael S. Bernstein. 2024. Generative agent simulations of 1,000 people. *Preprint*, arXiv:2411.10109.

Mary Sass and Matthew Liao-Troth. 2015. Personality and negotiation performance: The people matter. *SSRN Electronic Journal*.

Sudeepa Sharma, W. Bottom, and Hillary Anger Elfenbein. 2013. On the role of personality, cognitive ability, and emotional intelligence in predicting negotiation outcomes. *Organizational Psychology Review*, 3:293 – 336.

Sudeepa Sharma, Hillary Anger Elfenbein, Jeff L. Foster, and W. Bottom. 2018. Predicting negotiation performance from personality traits: A field study across multiple occupations. *Human Performance*, 31:145 – 164.

Ain Simpson. 2017. *Moral Foundations Theory*, pages 1–11. Springer International Publishing, Cham.

Rickard Stureborg, Dimitris Alikaniotis, and Yoshi Suhara. 2024. Large language models are inconsistent and biased evaluators. *Preprint*, arXiv:2405.01724.

Zhe Su, Xuhui Zhou, Sanketh Rangreji, Anubha Kabra, Julia Mendelsohn, Faeze Brahman, and Maarten Sap. 2024. Ai-liedar: Examine the trade-off between utility and truthfulness in llm agents. *Preprint*, arXiv:2409.09013.

Sarah Tanford and Steven Penrod. 1984. Social influence model: A formal integration of research on majority and minority influence processes. *Psychological Bulletin*, 95(2):189.

Lindia Tjuatja, Valerie Chen, Tongshuang Wu, Ameet Talwalkwar, and Graham Neubig. 2024. Do llms exhibit human-like response biases? a case study in survey design. *Transactions of the Association for Computational Linguistics*, 12:1011–1026.

Alexander Sasha Vezhnevets, John P Agapiou, Avia Aharon, Ron Ziv, Jayd Matyas, Edgar A Duéñez-Guzmán, William A Cunningham, Simon Osindero, Danny Karmon, and Joel Z Leibo. 2023. Generative agent-based modeling with actions grounded in physical, social, or digital space using concordia. *arXiv preprint arXiv:2312.03664*.

Ruiyi Wang, Haofei Yu, Wenxin Zhang, Zhengyang Qi, Maarten Sap, Graham Neubig, Yonatan Bisk, and Hao Zhu. 2024. Sotopia-π: Interactive learning of socially intelligent language agents. *Preprint*, arXiv:2403.08715.

Kelly Schwind Wilson, D. S. Derue, Fadel K. Matta, Michael Howe, and Donald E Conlon. 2016. Personality similarity in negotiations: Testing the dyadic effects of similarity in interpersonal traits and the use of emotional displays on negotiation outcomes. *The Journal of applied psychology*, 101 10:1405–1421.

Qingyun Wu, Gagan Bansal, Jieyu Zhang, Yiran Wu, Beibin Li, Erkang Zhu, Li Jiang, Xiaoyun Zhang, Shaokun Zhang, Jiale Liu, Ahmed Hassan Awadallah, Ryen W White, Doug Burger, and Chi Wang. 2023. Autogen: Enabling next-gen llm applications via multi-agent conversation. *Preprint*, arXiv:2308.08155.

Ziyi Yang, Zaibin Zhang, Zirui Zheng, Yuxian Jiang, Ziyue Gan, Zhiyu Wang, Zijian Ling, Jinsong Chen, Martz Ma, Bowen Dong, Prateek Gupta, Shuyue Hu, Zhenfei Yin, Guohao Li, Xu Jia, Lijun Wang, Bernard Ghanem, Huchuan Lu, Chaochao Lu, Wanli Ouyang, Yu Qiao, Philip Torr, and Jing Shao. 2024. Oasis: Open agent social interaction simulations with one million agents. *Preprint*, arXiv:2411.11581.

Xuhui Zhou, Hyunwoo Kim, Faeze Brahman, Liwei Jiang, Hao Zhu, Ximing Lu, Frank Xu, Bill Yuchen Lin, Yejin Choi, Niloofar Mireshghallah, Ronan Le Bras, and Maarten Sap. 2024a. Haicosystem: An ecosystem for sandboxing safety risks in human-ai interactions. *Preprint*, arXiv:2409.16427.

Xuhui Zhou, Zhe Su, Tiwalayo Eisape, Hyunwoo Kim, and Maarten Sap. 2024b. Is this the real life? is this just fantasy? the misleading success of simulating social interactions with LLMs. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 21692–21714, Miami, Florida, USA. Association for Computational Linguistics.

Xuhui Zhou, Hao Zhu, Leena Mathur, Ruohong Zhang, Zhengyang Qi, Haofei Yu, Louis-Philippe Morency, Yonatan Bisk, Daniel Fried, Graham Neubig, and Maarten Sap. 2024c. Sotopia: Interactive evaluation for social intelligence in language agents. In *ICLR*.

Caleb Ziems, William Held, Omar Shaikh, Jiaao Chen, Zhehao Zhang, and Diyi Yang. 2023. Can Large Language Models Transform Computational Social Science? *Computational Linguistics*.

CONTENT OF APPENDIX

In this paper, we present SOTOPIA-S^4, a user-friendly system for flexible, customizable, and large-scale social simulation. In the appendix, we provide additional details about our system:

A Character details

Characters in the SOTOPIA-S^4 inherit the character schema from the SOTOPIA platform (Zhou et al., 2024c). As shown in Figure A.1, each character has a name, age, occupation, public information, secret information, big five personality traits, moral values from moral foundations theory (Simpson, 2017), and other attributes.

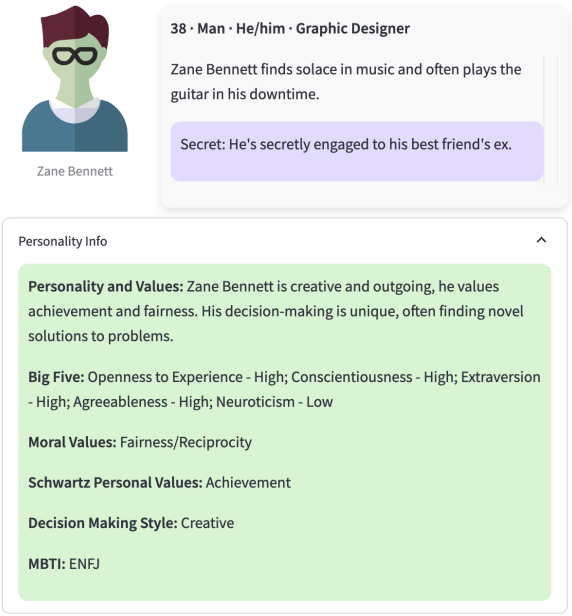

Figure A.1: An example character profile in SOTOPIA-S^4.

B Turn-taking details

Handling turn-taking is a crucial aspect of multi-agent interactions. In SOTOPIA-S^4, we offer two turn-taking strategies namely *round-robin* and *simultaneous*. In the *round-robin* strategy, agents take turns in a pre-defined sequential order specified by the user. Each agent acts once per round, with turns progressing in a fixed circular sequence until the conversation concludes or reaches a maximum number of turns. This structured approach ensures orderly participation and prevents any single agent from dominating the conversation, which could be useful for scenarios like social deduction games, auctions, and other scenarios where the order of actions is fixed.

In the *simultaneous* strategy, agents maintain a message queue and can decide when to act independently. Figure B.1 illustrates how agents interact asynchronously in SOTOPIA-S^4.

C Evaluation details

For the default evaluation setting, we use the evaluation framework from SOTOPIA (Zhou et al., 2024c). Specifically, we have the following evaluation metrics:

- **Goal Completion [0–10]**: Measures how well agents achieve their environment-defined social goals.
- **Believability [0–10]**: Evaluates if agent behavior is natural and consistent with their character profile, considering naturalness of interactions and alignment with traits.
- **Knowledge [0–10]**: Assesses how effectively agents acquire new and relevant information during interactions.
- **Secret [-10–0]**: Evaluates how well agents maintain private information while balancing trust-building through selective disclosure.
- **Relationship [-5–5]**: Measures how interactions affect relationships between agents, including impact on social status and reputation.
- **Social Rules [-10–0]**: Evaluates adherence to both social norms (e.g., politeness) and legal rules (institutionally enforced regulations).
- **Financial and Material Benefits [-5–5]**: Assesses economic utility gained, including both immediate monetary benefits and long-term economic advantages.

D API details

As shown in Figure D.1, the SOTOPIA-API provides a comprehensive set of operations for managing characters, scenarios, and episodes, and evaluation metrics. The interface uses different colors to indicate the HTTP methods supported by each

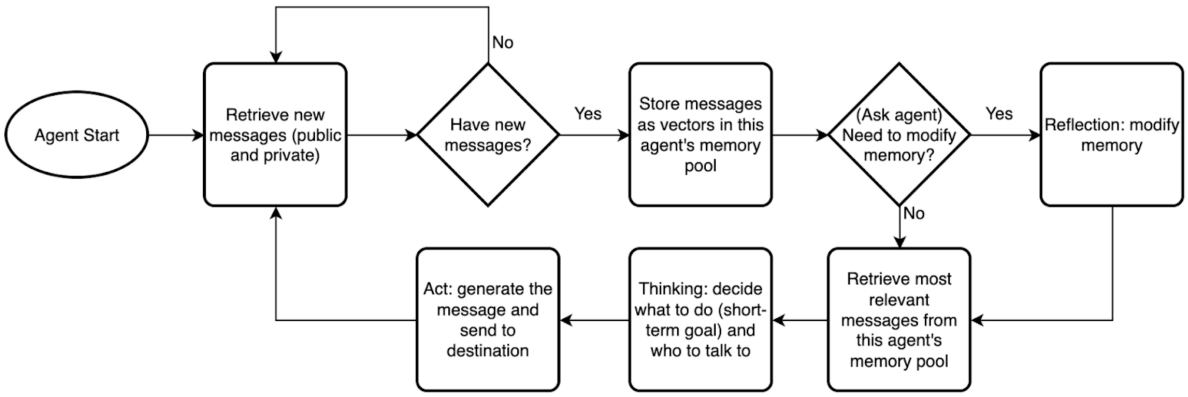

Figure B.1: The asynchronous interaction framework for agents in SOTOPIA-S^4 for the *simultaneous* turn-taking strategy. Each agent maintains its own message queue and can decide when to respond based on the conversation context and its own state.

endpoint, including `GET` for retrieving data, `POST` for creating new resources, and `DELETE` for removing existing resources. While common REST APIs often include `PUT` for updates, we deliberately omit this method to avoid potential errors and inconsistencies that could arise from concurrent modifications. Instead, updates can be handled through a combination of `DELETE` followed by `POST`, ensuring data integrity.

For simulation, the `POST /simulate` endpoint is a non-streaming endpoint that allows users to simulate episodes in a large-scale manner. During the process of the simulation, users can use `GET /simulate/status/{episode_pk}` to check the status of the simulation. For streaming simulation, we provide the websocket endpoint for the users to connect with the SOTOPIA-S^4 server and receive the simulation results in real-time.

E Dyadic Hiring Negotiation details

Here we provide the detailed setting of our dyadic hiring negotiation. Table E.1 shows the score allocations on different choices for two roles.

Starting Date	6.1	6.15	7.1	7.15	8.1
Manager	0	600	1200	1800	2400
Candidate	2400	1800	1200	600	0
Salary ($k)	100	105	110	115	120
Manager	6000	4500	3000	1500	0
Candidate	0	1500	3000	4500	6000

Table E.1: Comparison of Scenarios for Starting Date and Salary (Candidate vs. Recruiter Points)

Figure D.1: The API documentation page of SOTOPIA-S^4. The interactive Swagger UI provides comprehensive documentation of available endpoints, with different colors indicating the HTTP methods (`GET`, `POST`, `DELETE`) for each operation.