

OPEN-THEATRE: An Open-Source Toolkit for LLM-based Interactive Drama

Tianyang Xu^{◇1}, Hongqiu Wu^{◇2,3,4}, Weiqi Wu^{2,3,4}, Hai Zhao^{†2,3,4}

¹ UM-SJTU Joint Institute, Shanghai Jiao Tong University

² AGI Institute, School of Computer Science, Shanghai Jiao Tong University

³ Key Laboratory of Shanghai Education Commission for Intelligent Interaction and Cognitive Engineering, Shanghai Jiao Tong University

⁴ Shanghai Key Laboratory of Trusted Data Circulation and Governance in Web3
{johnnie.walker, wuhongqiu}@sjtu.edu.cn

Abstract

LLM-based Interactive Drama introduces a novel dialogue scenario in which the player immerses into a character and engages in a dramatic story by interacting with LLM agents. Despite the fact that this emerging area holds significant promise, it remains largely under-explored due to the lack of a well-designed playground to develop a complete drama. This makes a significant barrier for researchers to replicate, extend, and study such systems. Hence, we present OPEN-THEATRE, the first open-source toolkit for experiencing and customizing LLM-based interactive drama. It refines prior work with an efficient multi-agent architecture and a hierarchical retrieval-based memory system, designed to enhance narrative coherence and realistic long-term behavior in complex interactions. In addition, we provide a highly configurable pipeline, making it easy for researchers to develop and optimize new approaches.

1 Introduction

Interactive drama (Ryan, 1997; Dow et al., 2007) offers a novel participatory storytelling paradigm where users engage as an in-story character. In LLM-based interactive drama, the in-story characters (known as NPCs) are simulated by large language models (LLMs) (OpenAI, 2023; Jiang et al., 2023; Dubey et al., 2024), and the storytelling is collaboratively generated through dialogues between users and LLM agents.

The incorporation of LLMs have significantly enhanced the realism and adaptability of the interactive experiences. Despite growing interest in LLM-based interactive drama (Laurel, 1993; Wu et al., 2024; Han et al., 2024), a lack of accessible,

open-source tools persists, hindering researchers from easily creating, modifying, and experimenting with such interactive experiences.¹

To bridge this gap, we introduce **Open-Theatre**, an open-source toolkit designed for creating and experiencing configurable LLM-based interactive drama.

Current interactive drama frameworks vary in their focus, with some prioritizing user freedom while others emphasize stronger narrative coherence. Open-Theatre integrates these different architectures, such as one-for-all, director-actor, and hybrid structures, allowing users to flexibly choose according to their narrative preferences. A key challenge, however, remains in balancing freedom and coherence, since excessive user influence may disrupt plot progression. To address this, Open-Theatre introduces the **Director-Global Actor** framework, an extension of the traditional Director-Actor setup, where a global actor agent enhances character decision-making and narrative coherence. Furthermore, Open-Theatre enhances long-term coherence through a novel **hierarchical memory management system**, enabling intelligent information recall and prioritization.

Through experimental evaluation across various scripts and architectures, this paper demonstrates how Open-Theatre effectively bridges users with the crafted narrative world.

We outline the novel features of Open-Theatre:

- It is the first open-source toolkit that provides a highly configurable platform for creating and interacting with LLM-based interactive drama, enabling easy scripting, modification, and experimentation with dynamic narratives.
- It integrates and advances multiple architectures, utilizing sophisticated prompts and demonstrating the efficacy of the Director-Global Actor frame-

[◇] The authors contributed equally. [†] Corresponding author. OPEN-THEATRE is available at <https://github.com/johnnie193/Open-Theatre>, with a demo video for setup.

¹A newest interactive drama: <https://github.com/gingasan/interactive-drama>

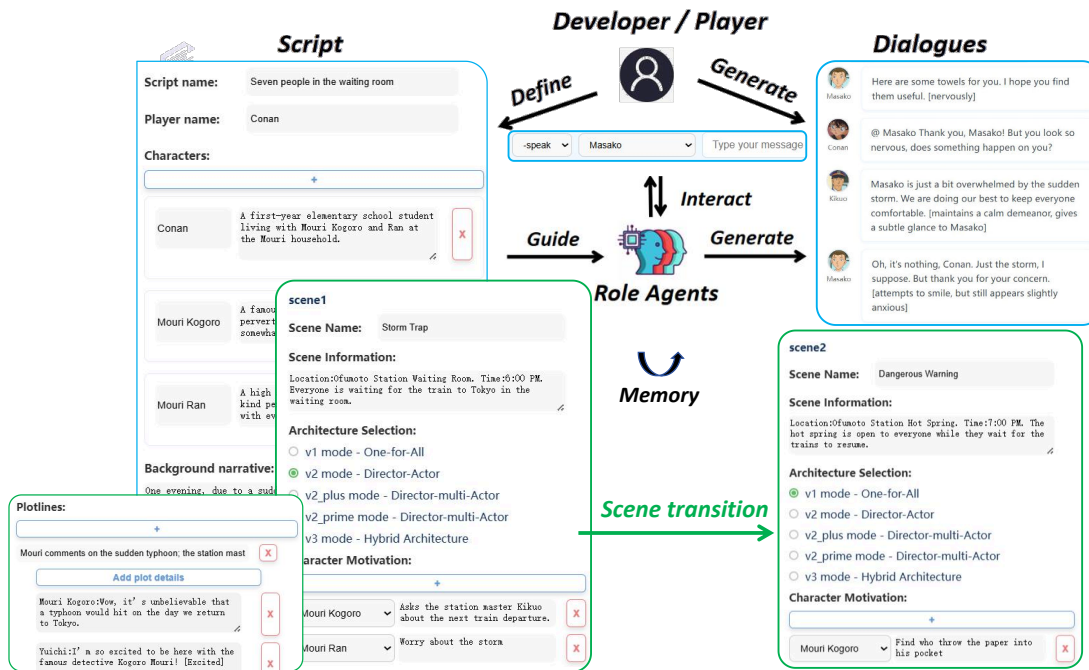


Figure 1: A demonstration of LLM-based interactive drama using the interface of the Open-Theatre. The script is adapted from the popular anime *Detective Conan*.

work.

- It features an innovative hierarchical memory management system for role agents, promoting human-like and consistent character responses.
- It allows secondary development for researchers with highly stable performance, enabling the integration of new architectures and customization of drama scripts and prompts in a lightweight manner.

2 Related work

Interactive Drama Computer-driven interactive drama has a long history of exploration (Laurel, 1993). More recently, Wu et al. (2024) introduces the concept of LLM-based interactive drama, outlining its six key components and proposing a fine-tuning strategy for a global drama LLM. Building on this line of research, Wu et al. (2025a) propose a plot-based reflection mechanism that periodically reviews the evolving storyline and adjusts narrative elements to facilitate player-centered story curves.

Simulate Dramatic Characters by LLM Agents

LLM-based role-playing has seen significant advancements in generating dynamic and consistent character behavior (Chen et al., 2024, 2023; Liu et al., 2024; Zhou et al., 2023; Wu et al., 2025b). Memory has been recognized as a core component of LLM-based agents, fundamentally shaping their ability to maintain coherence and adapt to long-

term interactions. While prior work (Park et al., 2023) has explored character memory over interactions, these systems often lack dynamic, context-aware memory management crucial for evolving dramatic narratives. General memory systems such as Mem0 (Chikara et al., 2025) and MemBank (Zhong et al., 2023) offer structured storage but struggle with customization based on plot progression. Open-Theatre addresses these limitations by incorporating a novel hierarchical memory system.

Multi-Agent Framework

Multi-agent systems (Tao et al., 2024; Islam et al., 2024) such as BabyAGI (Nakajima, 2023), AutoGen (Wu et al., 2023) and Camel-AI (Li et al., 2023), aim to achieve complex and goal-oriented behavior. In the context of interactive narratives, multi-agent frameworks such as director-actor (Han et al., 2024; Wu et al., 2025a) and ego-superego (Magee et al., 2024) coordinate agents to enable natural language interactions and strategic decision-making. However, maintaining narrative coherence in open-ended interactions remains a key challenge, as character autonomy can lead to inconsistent or fragmented storylines. Our Open-Theatre consolidates and extends these existing architectures through the introduction of a Director-Global Actor framework.

3 LLM-based Interactive Drama

We introduce the background of *LLM-based Interactive Drama*. It is a novel dialogue scenario where players experience an unfolding dramatic story by having dialogues with LLM-based characters.

As illustrated in Figure 1, a **drama script** guides the entire dialogues, which is in an episodic structure of scenes. Each scene details its background, character motivations, and plotlines, dictating character behaviours and plot objectives. For instance, Scene 1 (*Storm Trap*) depicts a scene where characters are trapped in the station waiting room due to a sudden typhoon. Completing Scene 1’s plotlines transitions to Scene 2 (*Dangerous Warning*).

The drama script forms the foundation, guiding **role agents** (LLM agents simulating characters) to follow a specific architecture to collaborate together. Players could interact with any character in the ongoing scene, and characters would respond based on their motivations and plotlines defined in the script, adapting to player influence. For instance, in Figure 1, a player (as *Conan*) interacts with *Masako*, who, despite being scripted to distribute towels, adaptively responds when questioned about her nervousness, balancing plotline adherence with player-driven interaction.

4 Open-Theatre

This section introduces our OPEN-THEATRE toolkit designed for a highly configurable and easy-to-use LLM-based interactive drama. The core components of the Open-Theatre interface includes developer console, player console, and monitor.

4.1 Developer Console

The Developer Console serves as the backbone of Open-Theatre, providing creators with tools for pre-configuring narratives and refining storytelling during interaction.

Script Management This module enables comprehensive customization of the drama script, from foundational background settings to individual scene design. The **Background** module defines the narrative context, including setting, themes, and detailed character profiles. **Scenes** are broken down into discrete components that shape story progression: **Scene Information** defines location, time, and background; **Character Motivation** outlines scene-specific roles and motivations, ensuring agent alignment with narrative demands; **Plot-**

lines organize the scene story into manageable segments, ensuring logical progression and coherence and allowing predefinition of dialogues or events. The system supports real-time modifications to the script, enhancing dynamic narrative control.

Prompt Management While default prompts are pre-designed for general use, users can refine them to suit specific narrative needs or subsequent development.

Save and Load Management This function enables persistent storage of the current script state and chat history, allowing revisiting specific narrative points and ensuring continuity across sessions. For researchers, it provides robust support for debugging, scenario testing, interaction replication, and iterative storytelling and replayability.

4.2 Player Console

The Player Console serves as the primary interface for players to engage in an interactive drama, enabling them to execute specific commands, trigger scene transitions, and influence the story’s direction. Players, as characters, can select other characters for **interaction** or choose **autonomous drama progression**. Advanced features include **withdrawing** to experiment with responses and **navigating directly to the next or previous scene**. All character dialogues, user inputs, and system feedback are presented in real-time, offering greater flexibility and dynamic story exploration.

4.3 Monitor

The Monitor provides players and developers with essential real-time contextual details of the Open-Theatre system’s processes, facilitating understanding and guiding adjustments to the script and prompts. It comprises several key components:

- **Current Script:** Offers a real-time view of the ongoing script’s state and progression, allowing users to track drama development.
- **Characters:** Enables users to explore details about characters in the current scene, including their profile, memory, motivation and agent reaction if applicable.
- **System Feedback:** Displays the system’s generated responses and real-time prompts, offering transparency into underlying processes and insights into system decisions for refinement.
- **Record:** Captures the complete state of the drama, including all major events, actions, and interactions throughout the narrative.

4.4 Architectural Integration and Innovation

The Open-Theatre toolkit enhances flexible narrative experiences by integrating existing architectures and introducing its own novel designs.

One-for-All This architecture uses a single global agent to control all characters and the progression of drama. It is computationally efficient and suitable for scenes that prioritize narrative flow over player agency, such as expository or background-setting scenes.

Director-Actor In this architecture, each character is represented by an independent actor agent, while a higher-level director agent coordinates and guides the narrative. It allows for complex decision-making, making it ideal for scenes requiring interactive depth. However, it often incurs significant computational cost and latency due to requiring individual LLM requests for each actor’s decision-making and profile adherence.

Hybrid Architecture This architecture switches between the director-actor and one-for-all architectures based on the scene’s characteristics, ensuring a balance between narrative immersion and interactive depth.

Director-Global Actor We extend the traditional Director-Actor architecture, employing a single, centralized global actor agent that makes decisions for all characters. This global actor agent serves as a high-level consciousness for the entire dramatic ensemble. Guided by the director agent, it synthesizes comprehensive, real-time knowledge of all characters’ profiles, memories, and states, providing a unifying context individual actor agents would otherwise lack. This capability not only ensures more strategically aligned and cohesive character actions, preventing contradictions and maintaining consistent character arcs, but also drastically improves efficiency, reducing redundant LLM API calls and latency for individual actor agents.

5 Hierarchical Memory System

Open-Theatre introduces a novel hierarchical memory architecture to provide narrative agents with a persistent, contextually-aware memory for extended, interactive narratives. While previous works (Park et al., 2023) emphasized recency, importance, and relevance in memory recall, our model extends this by introducing a partitioned

memory space and a dynamic scoring model tailored for dramatic agency. This adaptive memory lifecycle ensures computational tractability and psychological plausibility throughout extended interactions.

5.1 Architectural Design

The architecture comprises four specialized memory stores, structured to model the varying cognitive accessibility of different information types within a partitioned memory space. The fundamental unit is a memory entry, a natural language description with metadata including semantic type, scene identifier, and a dynamically updated importance score.

These stores include: a **Global Store** for foundational, static knowledge like core identity attributes; an **Event Store** capturing the chronological stream of episodic experiences including dialogues and actions; a **Summary Store** housing condensed, abstractive summaries of events produced by memory consolidation; and an **Archive Store** serving as a long-term repository for consolidated, less frequently accessed records, preserving historical context for potential future retrieval.

5.2 Retriever

To manage the cognitive load of long dialogues, a retriever module computes relevant information from the memory stores in response to new dialogue turns. For a given new dialogue query q , the final retrieval score for each candidate memory entry c is computed based on three factors: relevance, importance, and narrative recency, defined as:

$$S_{final}(c) = P_{recency} \cdot (S_{relevance} + S_{importance}) \quad (1)$$

5.2.1 Base score

The base retrieval score is an additive combination of relevance and importance. The **Relevance Score** ($S_{relevance}$) is computed using a standard hybrid method, fusing lexical signals from BM25Okapi with semantic similarity from FAISS-indexed sentence embeddings. For sentence embeddings, we utilize the all-MiniLM-L6-v2 model. The system processes the narrative at the character utterance level; each dialogue turn is treated as a single memory entry. These entries are aggregated into larger chunks for efficient storage and retrieval. Chunking is performed with a configurable size (`chunk_max_pieces = 5`) and an

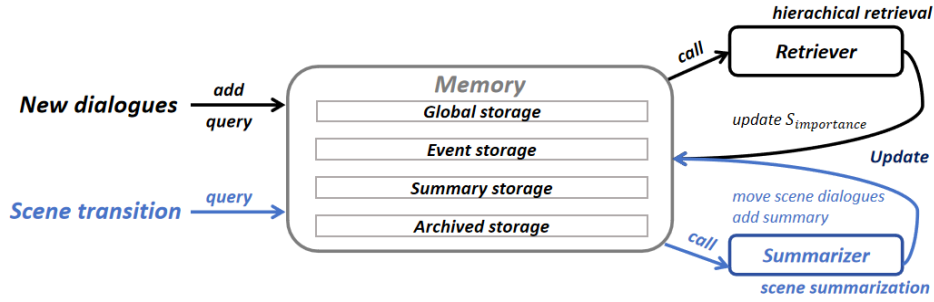


Figure 2: **Hierarchical Memory Architecture.** New dialogues are ingested into the Event Store and call retriever to query all stores. The scene transition signal calls the summarizer module to create new entries in the Summary Store and move original records to the Archive Store, completing the memory lifecycle.

overlap ($\text{chunk_overlap_pieces} = 1$), allowing the system to preserve local coherence while reducing fragmentation. The **Importance Score** ($S_{importance}$) is an evolving score unique to each memory entry. It increases with each retrieval event, allowing the system to learn a memory’s prominence from its usage history. The relative weighting between relevance and importance is determined by the `ADDITION_WEIGHT` (default set to 0.05), ensuring that the retrieval is primarily driven by immediate semantic and lexical relevance, while the learned importance serves as a subtle long-term signal to capture memory salience. This design ensures frequently recalled memories can influence agent behavior more, consistent with cognitive theories of memory salience.

5.2.2 Recency Punishment

The base score is modulated by a dynamic factor ($P_{recency} \in (0, 1]$) that penalizes temporally distant memories. Our model implements a two-tiered decay logic to differentiate narrative recency types:

Inter-Scene Penalty To model the decay of relevance *across* contextual shifts, for memory entries from a different scene ($c_{scene} \neq S_{current}$), a penalty based on scene distance is applied:

$$P_{recency} = (1 + \alpha \cdot |S_{current} - c_{scene}|)^{-1} \quad (2)$$

where α is a decay constant, empirically tuned to control the rate of forgetting across scene transitions, set to 0.25 as default.

Intra-Scene Dialogue Penalty For records *within* the current scene, a penalty based on turn order captures the immediate conversational context:

$$P_{recency} = (1 + \beta \cdot \text{TurnsAgo})^{-1} \quad (3)$$

where β is a per-turn decay constant, and *TurnsAgo* represents number of dialogue turns since creation.

We set β to 0.005, ensuring that recent turns are slightly favored but older relevant memories within the scene are not entirely excluded. For other memory types within the current scene, $P_{recency}$ is 1.0, incurring no penalty.

5.3 Summarizer

To manage the cognitive load inherent in long narratives, the system implements a summarizer module for memory consolidation. It processes records from the Event Store for completed scenes, leveraging an LLM to generate abstractive summaries that are added to the Summary Store. This process enriches the agent’s memory base with high-level, semantic knowledge from raw episodic experiences, concurrently reducing the pressure to retain all original records, which are then moved to the Archive Store for long-term preservation.

6 Evaluation

We conduct a rigorous evaluation to assess the performance of distinct agent architectures with and without memory system. Our analysis is performed across three diverse scripts to ensure robust, genre-agnostic findings: a detective story (adapted from *Detective Conan*), an adventure tale (adapted from *Harry Potter*), and a classical drama (adapted from *Romeo and Juliet*).

6.1 Experimental Setup

To ensure a comprehensive and objective assessment, we employed an automated evaluation pipeline. We utilized a suite of AI agents to act as players, designed with 10 distinct personas (ranging from cooperative to aggressive) to test system robustness and adaptability of each system configuration against a wide spectrum of behaviors. Following each simulated playthrough, we use an

Table 1: A multi-dimensional assessment of different architectures with and without memory configurations. Qualitative scores are averaged across three diverse scripts and rated on a 1-5 scale by our AI Judge.

ARCHITECTURE	CONFIG	MEMORY			ARCHITECTURE			EFFICIENCY	
		RA	RP	NC	CC	MAC	PA	Latency (s)	LLM Calls
<i>One-for-All</i>	w/o RAG	/	3.8	3.8	3.5	4.4	4.3	12.1	1.0
	w/ RAG	4.6	4.4	4.2	3.4	4.4	4.6	16.8	1.0
<i>Director-Actor</i>	w/o mem	/	4.0	3.7	4.6	3.3	3.7	34.5	3.4
	w/ mem	4.5	4.7	4.0	4.6	3.4	4.0	40.6	3.4
<i>Director-Global Actor</i>	w/o mem	/	3.9	4.0	4.1	4.5	4.0	20.5	2.0
	w/ mem	4.5	4.4	4.6	4.3	4.5	4.3	25.0	2.0

impartial Judge Agent (powered by GPT-4o) to score the complete interaction log, which provides scalable, consistent, and reproducible ratings based on defined metrics, effectively minimizing human bias and effort.

6.2 Evaluation Metrics

6.2.1 Memory System Performance

We evaluate our hierarchical memory system through three complementary dimensions:

- **Retrieval Accuracy (RA):** Success rate of retrieving relevant historical information.
- **Response Plausibility (RP):** Degree to which memory-based responses align well with the character and the current dialogue context.
- **Narrative Coherence (NC):** Long-term story consistency through memory integration.

6.2.2 Architectural Performance

We assess the interactive experience under different architectures through:

- **Character Consistency (CC):** Maintenance of distinct character personalities.
- **Multi-Agent Coordination (MAC):** Quality of inter-character strategic alignment.
- **Plot Adherence (PA):** Effectiveness in narrative progression and engagement.

6.2.3 System Efficiency

We measure computational performance through:

- **Response Latency:** Average response time per interaction turn (seconds/turn).
- **Computational Cost:** Number of LLM API calls required per turn.

6.3 Results

Table 1 presents our evaluation results, comparing different architectures with and without our hierar-

chical memory system.

6.3.1 Memory System Validation

Across all architectures, integrating the hierarchical memory system yields consistent improvements in response quality and long-term coherence. Specifically, **Retrieval Accuracy (RA)** exceeds **4.5** out of 5, confirming the system’s ability to retrieve relevant past information reliably. When compared to the baseline without memory system, **Response Plausibility** saw a substantial improvement (e.g., from **3.8** to **4.4** in One-for-all), indicating that memory-based responses are much better aligned with character and current dialogue context. These gains translate into higher **Narrative Coherence** (e.g., up to **4.6** in Director-Global Actor), showing improved consistency over long interactions. Even in simpler settings like One-for-All, memory boosts NC from **3.8** to **4.2**, further supporting the general utility of our memory system.

6.3.2 Architecture Performance

In terms of architectural design, the proposed **Director-Global Actor** architecture offers a strong balance between multi-agent coordination and narrative coherence, while maintaining competitive efficiency. Unlike Director-Actor, which excels at **Character Consistency (CC=4.6)** but suffers from lower **Multi-Agent Coordination (MAC=3.3)** and **Plot Adherence (PA=3.7)**, the Director-Global Actor achieves strong performance across almost all metrics when paired with our memory system. It scores highest on **NC (4.6)** and maintains high values for **MAC (4.5)** and **PA (4.3)**, suggesting that centralized reasoning leads to better plot alignment and action coordination among characters.

6.3.3 System Efficiency

In terms of efficiency, the One-for-All architecture requires only 1.0 LLM call and achieves the lowest latency. However, this simplicity comes at the cost of limited multi-character reasoning, resulting in lower coordination and consistency scores. By contrast, while the Director-Actor architecture incurs higher overhead, requiring 3.4 LLM calls per turn due to decentralized decision-making, the Director-Global Actor strikes a more favorable balance. It reduces the number of calls to 2.0 while maintaining high-quality outputs, confirming its computational viability for complex narrative generation. The competitive latency offers a favorable trade-off between performance and runtime cost.

7 Conclusion

This paper proposed Open-Theatre, a toolkit designed to facilitate the creation and interaction for LLM-based interactive drama. It assembles and advances different architectures, notably proposing a Director-Global Actor architecture and integrating a novel hierarchical memory system. Our experimental evaluation demonstrates that Open-Theatre provides a flexible and configurable platform for users to experience in a lightweight manner and facilitates secondary development. We hope this toolkit contributes to the broader field of interactive storytelling and inspires future research in AI-assisted narrative development.

8 Limitation

While an AI judge offers a highly scalable and standardized method for evaluating user experience, it might not fully capture the nuanced and subjective elements of human perception. Certain subtleties that only direct human feedback can reveal could be overlooked.

Also, our validation relies mostly on GPT-4o (accessed on August 6, 2024), limiting insights into framework generalizability across diverse LLMs. Future work will prioritize hybrid evaluation strategies and cross-model benchmarking to address these gaps.

References

Hongzhan Chen, Hehong Chen, Ming Yan, Wenshen Xu, Xing Gao, Weizhou Shen, Xiaojun Quan, Chenliang Li, Ji Zhang, Fei Huang, and Jingren Zhou. 2024. [Socialbench: Sociality evaluation](#)

[of role-playing conversational agents](#). *Preprint*, arXiv:2403.13679.

Nuo Chen, Yan Wang, Haiyun Jiang, Deng Cai, Yuhan Li, Ziyang Chen, Longyue Wang, and Jia Li. 2023. [Large language models meet harry potter: A dataset for aligning dialogue agents with characters](#). In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 8506–8520, Singapore. Association for Computational Linguistics.

Prateek Chhikara, Dev Khant, Saket Aryan, Taranjeet Singh, and Deshraj Yadav. 2025. [Mem0: Building production-ready ai agents with scalable long-term memory](#). *arXiv preprint arXiv:2504.19413*.

Steven Dow, Manish Mehta, Ellie Harmon, Blair MacIntyre, and Michael Mateas. 2007. [Presence and engagement in an interactive drama](#). In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '07*, page 1475–1484, New York, NY, USA. Association for Computing Machinery.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, Arun Rao, Aston Zhang, Aurélien Rodriguez, Austen Gregerson, Ava Spataru, Baptiste Rozière, Bethany Biron, Binh Tang, Bobbie Chern, Charlotte Caucheteux, Chaya Nayak, Chloe Bi, Chris Marra, Chris McConnell, Christian Keller, Christophe Touret, Chunyang Wu, Corinne Wong, Cristian Canton Ferrer, Cyrus Nikolaidis, Damien Al-lonsius, Daniel Song, Danielle Pintz, Danny Livshits, David Esiobu, Dhruv Choudhary, Dhruv Mahajan, Diego Garcia-Olano, Diego Perino, Dieuwke Hupkes, Egor Lakomkin, Ehab AlBadawy, Elina Lobanova, Emily Dinan, Eric Michael Smith, Filip Radenovic, Frank Zhang, Gabriel Synnaeve, Gabrielle Lee, Georgia Lewis Anderson, Graeme Nail, Grégoire Mialon, Guan Pang, Guillem Cucurell, Hailey Nguyen, Hannah Korevaar, Hu Xu, Hugo Touvron, Iliyan Zarov, Imanol Arrieta Ibarra, Isabel M. Kloumann, Ishan Misra, Ivan Evtimov, Jade Copet, Jaewon Lee, Jan Geffert, Jana Vranes, Jason Park, Jay Mahadeokar, Jeet Shah, Jelmer van der Linde, Jennifer Billock, Jenny Hong, Jenya Lee, Jeremy Fu, Jianfeng Chi, Jianyu Huang, Jiawen Liu, Jie Wang, Jiecao Yu, Joanna Bitton, Joe Spisak, Jongsoo Park, Joseph Rocca, Joshua Johnstun, Joshua Saxe, Junteng Jia, Kalyan Vasuden Alwala, Kartikeya Upasani, Kate Plawiak, Ke Li, Kenneth Heafield, Kevin Stone, and et al. 2024. [The llama 3 herd of models](#). *CoRR*, abs/2407.21783.

Senyu Han, Lu Chen, Li-Min Lin, Zhengshan Xu, and Kai Yu. 2024. [IBSEN: director-actor agent collaboration for controllable and interactive drama script generation](#). In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, ACL 2024, Bangkok, Thailand,

- August 11-16, 2024, pages 1607–1619. Association for Computational Linguistics.
- Md. Ashraful Islam, Mohammed Eunos Ali, and Md Rizwan Parvez. 2024. [Mapcoder: Multi-agent code generation for competitive problem solving](#). *Preprint*, arXiv:2405.11403.
- Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de Las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, L lio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timoth e Lacroix, and William El Sayed. 2023. [Mistral 7b](#). *CoRR*, abs/2310.06825.
- Brenda Laurel. 1993. *Computers as theatre*. Addison-Wesley.
- Guohao Li, Hasan Abed Al Kader Hammoud, Hani Itani, Dmitrii Khizbullin, and Bernard Ghanem. 2023. [Camel: Communicative agents for "mind" exploration of large language model society](#). *Preprint*, arXiv:2303.17760.
- Jiaheng Liu, Zehao Ni, Haoran Que, Tao Sun, Zekun Wang, Jian Yang, Jiakai Wang, Hongcheng Guo, Zhongyuan Peng, Ge Zhang, Jiayi Tian, Xingyuan Bu, Ke Xu, Wenge Rong, Junran Peng, and Zhaoxiang Zhang. 2024. [Roleagent: Building, interacting, and benchmarking high-quality role-playing agents from scripts](#). In *Advances in Neural Information Processing Systems*, volume 37, pages 49403–49428. Curran Associates, Inc.
- Liam Magee, Vanicka Arora, Gus Gollings, and Norma Lam-Saw. 2024. [The drama machine: Simulating character development with llm agents](#). *Preprint*, arXiv:2408.01725.
- Yohei Nakajima. 2023. [Babyagi](#). *GitHub repository*.
- OpenAI. 2023. [GPT-4 technical report](#). *CoRR*, abs/2303.08774.
- Joon Sung Park, Joseph C. O’Brien, Carrie J. Cai, Meredith Ringel Morris, Percy Liang, and Michael S. Bernstein. 2023. [Generative agents: Interactive simulacra of human behavior](#). *Preprint*, arXiv:2304.03442.
- Marie-Laure Ryan. 1997. Interactive drama: Narrativity in a highly interactive environment. *MFS Modern Fiction Studies*, 43(3):677–707.
- Wei Tao, Yucheng Zhou, Yanlin Wang, Wenqiang Zhang, Hongyu Zhang, and Yu Cheng. 2024. [Magis: Llm-based multi-agent framework for github issue resolution](#). *Advances in Neural Information Processing Systems*, 37:51963–51993.
- Hongqiu Wu, Weiqi Wu, Tianyang Xu, Jiameng Zhang, and Hai Zhao. 2025a. [Towards enhanced immersion and agency for llm-based interactive drama](#). In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, ACL 2025, Vienna, Austria, July 27 - August 1, 2025, pages 11166–11182. Association for Computational Linguistics.
- 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.
- Weiqi Wu, Hongqiu Wu, Lai Jiang, Xingyuan Liu, Hai Zhao, and Min Zhang. 2024. [From role-play to drama-interaction: An LLM solution](#). In *Findings of the Association for Computational Linguistics, ACL 2024, Bangkok, Thailand and virtual meeting, August 11-16, 2024*, pages 3271–3290. Association for Computational Linguistics.
- Weiqi Wu, Hongqiu Wu, and Hai Zhao. 2025b. [X-TURING: towards an enhanced and efficient turing test for long-term dialogue agents](#). In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, ACL 2025, Vienna, Austria, July 27 - August 1, 2025, pages 5874–5889. Association for Computational Linguistics.
- Wanjun Zhong, Lianghong Guo, Qiqi Gao, He Ye, and Yanlin Wang. 2023. [Memorybank: Enhancing large language models with long-term memory](#). *Preprint*, arXiv:2305.10250.
- Jinfeng Zhou, Zhuang Chen, Dazhen Wan, Bosi Wen, Yi Song, Jifan Yu, Yongkang Huang, Libiao Peng, Jiaming Yang, Xiyao Xiao, Sahand Sabour, Xiaohan Zhang, Wenjing Hou, Yijia Zhang, Yuxiao Dong, Jie Tang, and Minlie Huang. 2023. [Characterglm: Customizing chinese conversational ai characters with large language models](#). *ArXiv*, abs/2311.16832.