FAKE PLAYER: SIMULATING PLAYER BEHAVIOR TO DISTILL DIALOGUES FOR LLM-BASED NPC TRAINING

Anonymous authors
Paper under double-blind review

ABSTRACT

In the era of large language models (LLMs), games increasingly deploy LLM-based role-playing NPCs to replace traditional scripted NPCs, enabling more intelligent and dynamic interactions. To ensure persona consistency and output stability, these NPCs require fine-tuning for alignment, utilizing training data with dual dimensions: persona-aligned assistant responses and diverse, authentic user inputs reflecting real player behaviors. However, existing research prioritizes persona consistency in NPC responses while neglecting the diversity and authenticity of user-side inputs. This critical gap leads to NPC responses that are misaligned with genuine player interactions, significantly impairing player immersion and experience. Human annotation struggles to address this gap due to its inability to comprehensively cover the vast spectrum of player behaviors. Moreover, practical deployment constraints strongly favor small-parameter LLMs for NPCs, making data quality paramount. To bridge this gap, we propose: 1) **Fake Player**: A multi-agent LLM distillation framework where collaborative agents simulate expression-constrained human players to distill diverse, human-aligned dialogue data from large LLMs; 2) **Distill Bench**: A standardized benchmark for quantitatively assessing distilled data quality, bypassing costly NPC retraining. Extensive experiments on upstream data quality and downstream NPC performance demonstrate that Fake Player produces higher-quality dialogue data and yields greater improvements in small-parameter NPC models than existing data synthesis methods. The consistent ranking across the two evaluation stages supports using Distill Bench to guide data selection and refinement before fine-tuning, offering a practical way to reduce costly trial-and-error training.

1 INTRODUCTION

While large language models (LLMs) have achieved breakthroughs in general natural language processing (NLP) tasks (Touvron et al., 2023; Bai et al., 2023; OpenAI, 2023), researchers are exploring their application in role-playing non-player characters (NPCs) for games. These LLM-based NPCs aim to deliver open-ended responses for immersive experiences. They broadly fall into two categories: Prompt-engineered NPCs constrained by sophisticated prompts on large foundation LLMs (Tu et al., 2024; Wang et al., 2024); Fine-tuned NPCs trained on NPC persona-specific data (Zhou et al., 2024; Shao et al., 2023).

Relying solely on prompt engineering constraints to maintain the character personas and outputs of NPCs is non-robust (Shanahan et al., 2023). In practical role-playing application scenarios, fine-tuning with corresponding datasets is necessary for alignment. Figure 1 illustrates this issue with an NPC from a racing game. When the player mentions another game, the Prompt-based NPC responds with characters and locations from that game's world, going beyond its role setting (Out of Character, OOC). In contrast, the Fine-tuned NPC responds within the racing setting and naturally redirects the conversation to racing. Crucially, players value NPCs' emotional intelligence (EI) over cognitive capabilities—a trait achievable with small LLMs through targeted training (Zhang et al., 2025a; Wang et al., 2025). In addition, due to latency and resource limits, small-parameter LLMs are preferred for deployment, elevating the importance of training data quality.

Training data for LLM-based NPCs comprises two dimensions: persona-aligned assistant responses and user inputs reflecting diverse player behaviors. Existing methods focus excessively on the

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

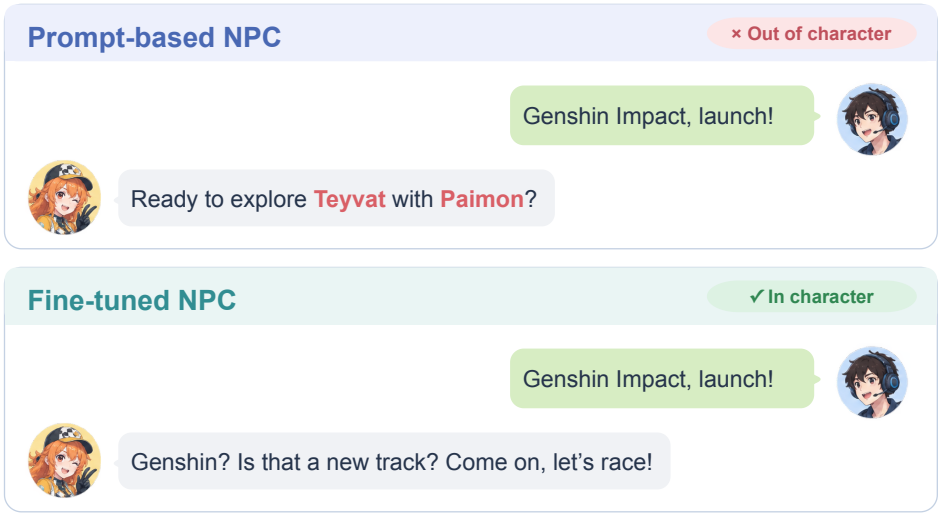


Figure 1: Illustration of out-of-character (OOC) responses in a racing-game NPC. Given the same player input, the Prompt-based NPC refers to another game's world, while the Fine-tuned NPC keeps the conversation within the racing setting.

former while overlooking the latter: 1) Few-shot generation leverages in-context learning to generate dialogues but suffers from overfitting to seed examples (Tao et al., 2023; Zhou et al., 2024), resulting in low diversity and heavy reliance on human curation; 2) Role-Playing Agents improve diversity through interaction (Kong et al., 2024; Tamoyan et al., 2024; Yang et al., 2025b), but exhibit behavioral instability (e.g., exaggerated or non-human inputs), requiring extensive post-filtering. In summary, both methods, hampered by insufficiently diverse and authentic training data, result in practical experiences where NPCs exhibit OOC or comprehension biases, undermining players' gaming immersion, an undesirable outcome in gaming experiences.

To synthesize diverse yet authentic player-NPC interactions with minimal human effort, we analyze behavioral gaps between LLM-simulated players and real humans. Real players, constrained by physical interfaces (e.g., devices and UIs), tend to produce concise intent-driven texts, whereas LLM agents generate verbose internal thoughts without such constraints. We thus propose **Fake Player**, a multi-agent framework that simulates expression-constrained human players via behavioral modeling. It consists of three specialized agents: **Inner Monologue Agent**: Generates the player's internal thoughts based on seed settings; **Intent Analysis Agent**: Performs fine-grained intent analysis on the monologue to construct and dynamically maintain an intention stack throughout multi-turn interactions; **Typing Agent**: Simulates interface-imposed expression constraints by converting the top intent from the stack into concise, human-like text output. This architecture effectively replicates authentic player-NPC interaction patterns in games, enabling efficient generation of high-quality distilled data.

Furthermore, we develop **Distill Bench**, a benchmark that quantitatively assesses the quality of distilled data through three core dimensions: topic relevance, human-likeness, and data diversity, allowing evaluation prior to NPC training and significantly reducing the time and costs of iterative model retraining.

Our contributions are as follows:

1. We propose Fake Player, a multi-agent LLM distillation framework where collaborative agents simulate expression-constrained human players to distill diverse, human-aligned dialogue data from large LLMs.

2. We introduce Distill Bench, a standardized benchmark for quantitatively assessing distilled data quality without costly NPC retraining.

3. We conduct upstream and downstream experiments, together with evaluation on an independent multi-turn role-playing benchmark, to validate the framework and its transferability beyond the original game-NPC setting.

2 Related Work

2.1 LLM-based NPCs

With the rapid advancement of LLMs, their application in generating NPCs for games and interactive scenarios has garnered significant attention. LLM-based NPCs aim to transcend scripted responses, enabling dynamic, context-aware interactions that enhance user immersion (Shanahan et al., 2023; Park et al., 2023). These systems can be broadly categorized based on their technical approaches:

Prompt-based NPCs leverage sophisticated prompt design to elicit role-specific behaviors from LLMs. For instance, RoleLLM (Wang et al., 2024) introduces dialogue engineering and retrieval augmentation to simulate fine-grained character traits, allowing models like GPT-4 to mimic speaking styles and knowledge of specific roles (e.g., Sherlock Holmes) without fine-tuning. Similarly, CharacterEval (Tu et al., 2024) evaluates prompt-based methods for animating anime characters, emphasizing the importance of context alignment and stylistic consistency in open-ended conversations. These approaches benefit from the strong generalization of large LLMs but often rely on complex prompt engineering to constrain outputs to the NPC's persona.

Fine-tuned NPCs, by contrast, are optimized on role-specific datasets to embed persona traits directly into model weights. CharacterGLM (Zhou et al., 2024) and character-LLM (Shao et al., 2023) demonstrate that fine-tuning open-source models on dialogue data from specific characters enhances their ability to maintain consistent personalities across interactions. For example, AutoAgents (Chen et al., 2023) proposes a framework for generating NPCs with specialized knowledge and social behaviors through targeted fine-tuning, enabling more coherent and immersive multi-agent simulations. Such methods reduce reliance on prompt engineering but require high-quality, role-specific training data.

2.2 Data Distillation

Data distillation transfers knowledge from large models to smaller ones via two paradigms. Implicit distillation leverages logit mimicry and intermediate layer alignment: (Hinton et al., 2015) pioneered distilling ensembles into small models using softened logits. DistilBERT (Sanh et al., 2019) retains 95% of BERT's performance with 40% fewer parameters by mimicking logits and attention patterns, while TinyBERT (Jiao et al., 2019) enhances task-specific performance through two-stage distillation, combining logit matching and layer alignment. Explicit distillation uses larger LLMs to generate high-quality datasets for training smaller LLMs. Alpaca (Taori et al., 2023) demonstrated this by employing GPT-3 to generate instruction samples, enabling LLaMA to match closed-source models. Self-Instruct (Wang et al., 2022) reduced human annotation reliance via self-generated training data.

3 Methods

Current approaches for synthesizing LLM-based NPC training data primarily adopt two paradigms: 1) ICL Generation (Tao et al., 2023; Shao et al., 2023; Wang et al., 2024; Zhou et al., 2024), which leverages carefully designed few-shot examples and seed settings to generate similar interactions. However, this method often **overfits seed examples**, resulting in low diversity; 2) Role-Playing Agent (Tamoyan et al., 2024; Yang et al., 2025b), which employ dual agents to simulate player-NPC interactions but **lacks behavioral constraints**, yielding non-human-like outputs requiring rigorous post-filtering.

As depicted in Figure 2, the Fake Player employs a multi-agent architecture to generate diverse and behaviorally realistic player interaction data. The Fake Player comprises three specialized agents working in sequence: the Inner Monologue Agent, which generates contextual inner monologues based on minimal seed settings, simulating a human player's unspoken thoughts when interacting with NPCs; the Intent Analysis Agent, which analyzes the monologue to extract fine-grained intents, dynamically organizing them into a structured intent stack that captures nuanced player motivations;

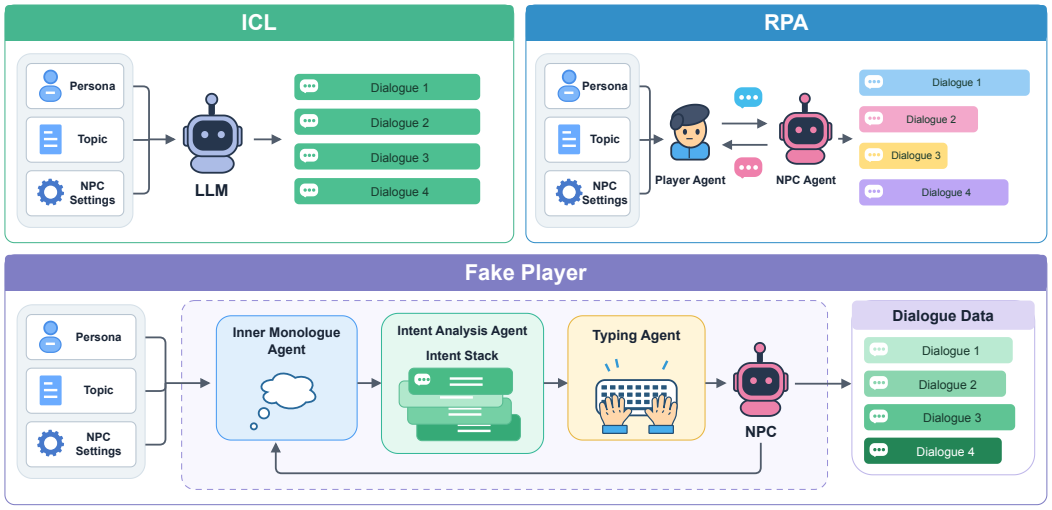


Figure 2: Comparison via ICL, RPA and Fake Player. The upper left is ICL, the upper right is RPA, and the bottom is the framework of Fake Player. The depth of the dialogue's color indicates the similarity, while the length indicates the authenticity.

and the Typing Agent, which models device-constrained player expression by converting the top intent into natural language output, incorporating typing patterns and cognitive constraints observed in real players.

3.1 Inner Monologue Agent

We follow mainstream research conventions by generating content based on player persona and topic. To address practical gaming scenarios, we introduce a domain setting parameter D, which adapts to two key interaction types: companionship dialogues and knowledge-seeking dialogues.

Formally, given player persona P, domain D, and topic T, we leverage a prompt template to instantiate the Inner Monologue Agent IMA, yielding the initial monologue M_0:

$$M_0 = \text{IMA}\left(P, T, D\right). \tag{1}$$

During multi-turn interactions, the monologue M_i at turn i updates dynamically based on the NPC's prior response R_{i-1}:

$$M_i = \text{IMA}\left(M_{i-1}, R_{i-1}, P, T, D\right). \tag{2}$$

3.2 Intent Analysis Agent

The Intent Analysis Agent IAA manages the player's conversational intents across turns. During interaction, a player may continue the current topic, raise a new concern, or return to an unresolved question. We maintain an intent stack to organize these changes and guide the next utterance.

To model this behavior, we construct an intention stack S that stores fine-grained intents. At turn i, the IAA derives a list of fine-grained intents L_i from the previous stack S_{i-1} and the current monologue M_i. The stack is updated, and the top intent C_i is selected to guide subsequent interactions:

$$\begin{aligned} L_i &= \text{IAA}(S_{i-1}, M_i), \\ S_i &= \text{Stack_Update}(S_{i-1}, L_i), \\ C_i &= \text{Stack_Pop}(S_i), \end{aligned} \tag{3}$$

where $S_0 = \varnothing$. Stack_Update refers to the operation that updates the previous stack S_{i-1} to generate S_i through operations such as addition, deletion, merging, and retention, all primarily based on

L_i, and Stack_Pop is the stack pop operation that retrieves and removes the top element C_i. The remaining intents are available for the next update.

This mechanism provides a persistent representation of pending conversational goals. For example, a player asking about crop planting may become interested in pet feeding after the NPC mentions pets, then return to an unresolved crop-related question. The stack allows the simulator to revise the current priority while retaining relevant context. The detailed IAA prompt is provided in Appendix L.

3.3 Typing Agent

The Typing Agent (TA) converts the inner monologue and selected intent into the player's final utterance. Given the current inner monologue M_i and active intent C_i, it generates the player utterance Q_i:

$$Q_i = \mathrm{TA}(M_i, C_i). \tag{4}$$

It focuses on concise, colloquial expression of the current intent. This design follows common text-based player interactions, where a message usually expresses a limited part of the player's thoughts and the next message depends on the response received.

3.4 Overall Interaction

Given the multi-turn interaction paradigm, we dynamically determine dialogue length based on domain D: **Chat domain**: Players prefer extended interactions for emotional fulfillment; **Knowledge domain**: Players typically exit after obtaining answers. The turn count N is sampled from a domain-specific range:

$$N \sim \mathrm{RandTurn}(D), \quad N \in \{1, 2, \ldots, 10\} \tag{5}$$

where the RandTurn procedure first establishes domain-dependent bounds, then stochastically selects N within this interval.

The complete Fake Player interaction is formally defined in Algorithm 1.

Algorithm 1 Fake Player Overall Interaction

Input: Persona P, Domain D, Topic T
$N \leftarrow \mathrm{RandTurn}(D)$; $\mathcal{L} \leftarrow \{\}$; $S_0 \leftarrow \{\}$
$M_0 \leftarrow \mathrm{IMA}(P, T, D)$; $S_1 \leftarrow \mathrm{Stack_Update}(S_0, \mathrm{IAA}(S_0, M_0))$
for $i = 1$ to N **do**
  $C_i \leftarrow \mathrm{Stack_Pop}(S_i)$; $Q_i \leftarrow \mathrm{TA}(M_{i-1}, C_i)$
  $R_i \leftarrow \mathrm{NPC}(Q_i)$; append (Q_i, R_i) to $\mathcal{L}$
  if $i < N$ **then**
    $M_i \leftarrow \mathrm{IMA}(M_{i-1}, R_i, P, T, D)$; $S_{i+1} \leftarrow \mathrm{Stack_Update}(S_i, \mathrm{IAA}(S_i, M_i))$
  end if
end for
Output: $\mathcal{L} = \{(Q_1, R_1), \ldots, (Q_N, R_N)\}$

4 Evaluation

We introduce **Distill Bench** to evaluate synthesized dialogue data before retraining downstream LLM-based NPCs. It assesses **topic relevance**, **human-likeness**, and **diversity**, using LLM-as-Judge for the first two dimensions and text embedding similarity for diversity. Topic relevance covers topic depth and breadth; human-likeness covers conciseness and improvisation; diversity is quantified with binary entropy of the maximum similarity between player utterances at corresponding turns under the same seed settings. These dimensions provide feedback for comparing data-generation methods and refining their outputs. Detailed metric definitions, scoring prompts, and calculations are provided in Appendix B and Appendix C. We then examine whether the upstream data-quality results correspond to downstream NPC performance.

5 EXPERIMENTS

5.1 DETAILS OF BENCHMARK

Distill Bench is built on 100 real-sampled player personas and 100 topics. We evaluate on five personas and 20 topics spanning chit-chat and knowledge-seeking settings; each persona–topic configuration is generated five times, yielding 500 dialogues per method. The benchmark uses DeepSeek-V3-250324 (Liu et al., 2024) as the LLM judge and Qwen3-Embedding-0.6B (Zhang et al., 2025b) for text embeddings. To validate the automated scores, five graduate-level evaluators manually assess 50 sampled dialogues per method using the same criteria. Sampling details, metric prompts, and annotation statistics are provided in Appendix B and Appendix C.

5.2 COMPARISON BASELINES

To ensure rigorous evaluation, we construct two strong baselines by adapting state-of-the-art (SOTA) methods from relevant research directions—addressing the mismatch between existing approaches and our specific task of synthesizing player-NPC interaction data. These baselines are not naive off-the-shelf methods but are tailored to align with our task constraints while retaining the core strengths of their SOTA predecessors: **In-Context Learning (ICL)**: Derived from SOTA ICL-based dialogue generation methods (Tao et al., 2023; Shao et al., 2023; Wang et al., 2024; Zhou et al., 2024)—a dominant paradigm for role-aware dialogue synthesis. **Role-Playing Agent (RPA)**: Built upon SOTA role-playing agent frameworks (Kong et al., 2024; Tamoyan et al., 2024; Yang et al., 2025b), which leverage dual-agent architectures (player and NPC roles) for interactive dialogue. More details are shown in Appendix L.

To avoid the bias introduced by a single model, we conducted experiments using three base models for all methods: DeepSeek-V3-250324 (Liu et al., 2024), DeepSeek-R1-250528 (DeepSeek-AI et al., 2025), and Doubao-Seed-1.6-250615[1]. These models are widely recognized as powerful and efficient representatives in both open-source and closed-source communities.

5.3 DOWNSTREAM EXPERIMENTS

To better verify that the quality of distilled data can directly impact NPC performance, we use the Distill Benchmark to generate 3,000 high-quality dialogue samples, which are employed for Supervised Fine-Tuning (SFT) of LLMs to develop NPCs. More details are shown in Appendix D.

For prompt-based NPCs, we use models widely recognized in the community, including GPT-5 (Singh et al., 2025), Doubao-Seed-1.6-250615, Gemini-2.5-Flash (Comanici et al., 2025), DeepSeek-V3-250324 (Liu et al., 2024), DeepSeek-R1-250528 (DeepSeek-AI et al., 2025), and Qwen2.5-14B-Instruct (Yang et al., 2025a) to ensure coverage across different parameter scales. For Fine-tuned NPCs, following mainstream industry practices, we adopt Qwen2.5-7B-Instruct (Yang et al., 2025a) as the backbone for the main downstream comparison. The Hunyuan-7B-Instruct [2] comparison is retained in the appendix.

For NPC evaluation, We adopt an end-to-end dynamic interaction evaluation approach, utilizing DeepSeek-V3 as the player. The player initiates dialogues with the NPC based on the topic, with a total of 4 interaction turns. In each turn, the player rates the NPC's response. The evaluation metrics include Topic Relevance (TR), Character Characteristics (CC), Character Performance (CP), Emotional Appeal (EA), Character Interaction (CI), and Character Reality (CR), with detailed procedures provided in Appendix E.

To further directly compare the performance differences among fine-tuned NPCs, we adopt comparative evaluation to calculate the win rate of NPCs trained on different distilled data under the same LLM backbone in Appendix G.

[1] https://seed.bytedance.com/en/seed1_6

[2] https://github.com/Tencent-Hunyuan/Hunyuan-7B

Table 1: Distilled data evaluation across three generation backends. Breadth and Depth form Topic Relevance; Conciseness and Improvisation form Human-Likeness. Overall denotes the aggregate data score. Detailed metric definitions are provided in Appendix B.

Methods	Topic Relevance			Human-Likeness			Diversity	Overall
	Breadth	Depth	Total	Conciseness	Improvisation	Total		
Base Model: DeepSeek-V3-250324								
In-context Learning	72.50	73.30	72.90	59.40	74.60	67.00	46.63	62.18
Role-Playing Agent	76.80	69.30	73.05	45.60	72.60	59.10	**67.79**	66.65
Fake Player (Ours)	73.20	75.00	**74.10**	64.40	75.20	**69.80**	67.28	**70.39**
Base Model: DeepSeek-R1-250528								
In-context Learning	74.90	71.50	73.20	54.60	72.60	63.60	61.45	66.08
Role-Playing Agent	78.40	71.00	74.70	41.80	74.40	58.10	80.26	71.02
Fake Player (Ours)	76.80	73.30	**75.05**	60.80	77.40	**69.10**	**83.53**	**75.89**
Base Model: Doubao-Seed-1.6-250615								
In-context Learning	74.90	70.50	72.70	65.16	73.40	**69.28**	34.53	58.84
Role-Playing Agent	77.40	71.80	**74.60**	47.60	73.08	60.34	59.50	64.81
Fake Player (Ours)	78.30	67.9	73.10	60.80	75.04	67.92	**62.28**	**67.77**

5.4 Industrial Application Experiments

Our cost analysis shows comparable generation cost in absolute terms, and we report the framework's deployment in a commercial game serving over one million players. Post-deployment questionnaires show a 15% increase in player satisfaction with NPC interactions. Cost and deployment details are provided in Appendix I.

5.5 Role-Playing Evaluation on PersonaArena

To examine whether the improvements extend beyond the estate manager NPC used in our downstream experiments, we evaluate on PersonaArena (Shi et al., 2026), a public benchmark for multi-turn role-playing. We use three personas in different social scenes. For each persona, the tested model plays the protagonist from the same initial scene, while DeepSeek-V4-Flash (Xu et al., 2026) generates the supporting characters and narration. We evaluate five dialogue rounds per persona.

Following PersonaArena, GPT-5, DeepSeek-V4-Flash, and Claude Opus 4.5 [3] score each dialogue on eight role-playing dimensions. We report the mean across dimensions, personas, and judges. The scoring criteria and simulation settings are given in Appendix E.1.

6 Results and Analysis

6.1 Distilled Data Analysis

Topic Relevance. As shown in Table 1, all methods achieve relatively high topic-relevance scores. Fake Player obtains the highest totals of 74.10 with DeepSeek-V3 and 75.05 with DeepSeek-R1. Topic depth is the main contributing factor: it explores the topic in greater detail while maintaining reasonable breadth. RPA obtains higher topic breadth with these two backends, but lower depth. With Doubao-Seed-1.6, RPA achieves the highest topic-relevance total, showing variation across generation backends.

Human-Likeness. Fake Player achieves human-likeness scores of 69.80 with DeepSeek-V3 and 69.10 with DeepSeek-R1. Both conciseness and improvisation improve over RPA. ICL also outperforms RPA in this dimension across all three backends. With Doubao-Seed-1.6, ICL obtains the highest human-likeness score of 69.28, followed by Fake Player at 67.92.

Data Diversity. The two agent-based methods obtain higher diversity scores than ICL across all backends. Fake Player achieves the highest scores with DeepSeek-R1 and Doubao-Seed-1.6, and remains close to RPA with DeepSeek-V3. It combines varied interactions with stronger human-

[3] https://www.anthropic.com/news/claude-opus-4-5

Table 2: Human expert evaluation of the generated dialogues.

Methods	Topic Relevance			Human-Likeness		
	Breadth	Depth	Total	Conciseness	Improvisation	Total
In-context Learning	66.20	72.20	69.15	71.20	66.00	68.60
Role-Playing Agent	72.40	67.60	70.00	40.40	57.20	48.80
Fake Player (Ours)	75.20	70.40	**72.80**	74.40	72.40	**73.40**

Table 3: Downstream experiments. Bold values indicate the highest scores. Evaluation metrics include Topic Relevance (TR), Character Characteristics (CC), Character Performance (CP), Emotional Appeal (EA), Character Interaction (CI), and Character Reality (CR).

NPCs	**Overall**	**TR**	**CC**	**CP**	**EA**	**CI**	**CR**
GPT-5	**77.88**	**82.75**	93.88	81.50	**67.63**	**72.63**	68.88
DeepSeek-V3-250324	74.92	70.50	**96.25**	**85.50**	64.25	62.63	**70.38**
Doubao-Seed-1.6-250615	72.10	70.50	93.25	79.63	60.50	60.75	68.00
DeepSeek-R1-250528	64.79	60.50	91.88	75.25	53.13	49.00	59.00
Gemini-2.5-Flash	51.88	60.75	78.00	55.75	38.13	36.50	42.13
Qwen2.5-14B-Instruct	37.15	55.00	49.00	33.13	27.25	33.75	24.75
Qwen2.5-7B-Instruct	29.88	39.38	51.63	31.88	21.00	19.88	15.50
+In-Context Learning	53.29	54.38	83.63	60.50	38.88	32.75	49.63
+Role-Playing Agent	67.00	63.50	92.00	77.63	55.38	52.13	61.38
+Fake Player (Ours)	**73.10**	**70.13**	**94.25**	**83.13**	**60.00**	**59.25**	**71.88**

likeness than RPA, giving the highest overall data score under each backend. The cost analysis and additional t-SNE visualizations are provided in the appendix.

Human Evaluation Validation. Table 2 reports human evaluation using the same topic-relevance and human-likeness criteria. The method rankings agree with the automated assessment: Fake Player ranks first in both dimensions, while RPA ranks above ICL in topic relevance and ICL ranks above RPA in human-likeness. Confidence intervals are provided in the appendix.

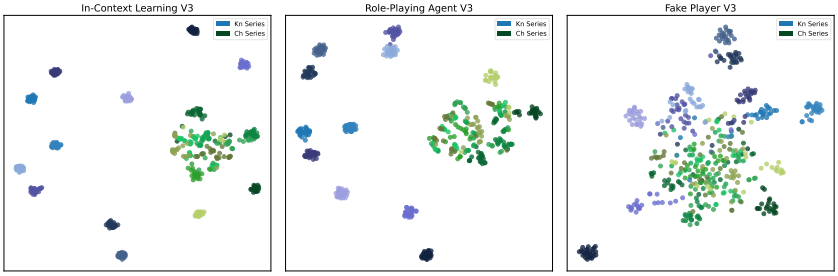


Figure 3: Relationship between data diversity and correlation visualized by t-SNE (Base model: DeepSeek-V3). Additional results are provided in Appendix J.

We visualize data diversity and correlation using t-SNE (Figure 3). ICL forms tight clusters with limited diversity, whereas RPA shows stronger diversity in knowledge-oriented data but weaker chit-chat coverage. Fake Player maintains broader coverage across both domains. Additional visualizations and the cost analysis are provided in the appendix.

6.2 Downstream NPC Analysis

Table 3 showed that prompt-based NPCs followed the scaling law, with performance improving as model parameters increased. This validated the effectiveness of our evaluation method, which also applied to fine-tuned NPCs. Across two LLM backbones, NPCs trained on Fake Player data

Table 4: PersonaArena role-playing evaluation. Scores are on a 1–5 scale and averaged over three judges on shared personas; Avg. is the mean of eight dimensions. Knowledge Accuracy (KA), Emotional Expression (EE), Personality Traits (PT), Behavioral Accuracy (BA), Immersion (IM), Adaptability (AD), Behavioral Coherence (BC), and Interaction Richness (IR). Metric definitions and protocol details are provided in Appendix E.1.

NPCs	Avg.	KA	EE	PT	BA	IM	AD	BC	IR
GPT-5	3.19	4.22	2.56	3.44	3.44	3.22	2.89	3.33	2.44
DeepSeek-V4-Flash-R1	2.90	3.44	2.67	3.78	3.22	2.78	2.78	2.44	2.11
Gemini-2.5-Flash	2.75	3.22	2.44	3.44	3.22	2.67	2.33	2.67	2.00
Qwen2.5-7B-Instruct	2.11	2.56	2.22	2.33	2.33	2.11	2.00	**2.00**	1.33
+In-Context Learning	1.90	2.33	2.22	2.56	2.11	1.67	1.67	1.44	1.22
+Role-Playing Agent	2.18	2.56	2.33	2.67	2.22	2.00	**2.22**	1.89	**1.56**
+Fake Player (Ours)	**2.35**	**2.78**	**2.56**	**3.22**	**2.56**	**2.22**	2.00	1.89	**1.56**

significantly outperformed ICL and RPA baselines. The Qwen-based NPC achieved an overall score of 73.10, ranking third behind only GPT-5 and DeepSeek-V3. With high-quality distilled data, small-parameter LLMs can match much larger foundation models in key NPC interaction metrics. Notably, our method achieved the highest Character Reality score of 71.88, outperforming GPT-5 and DeepSeek-V3 in this critical dimension. This confirmed a strong correlation between data quality and downstream performance, and validated our evaluation method which reduces NPC training time and resources.

6.3 Cross-Dataset Validation on Public Role-Playing Benchmark

Table 4 shows that Fake Player scores 2.35 overall, compared with 1.90 for ICL and 2.18 for RPA on the same Qwen2.5-7B backbone. After averaging the three judges, Fake Player also scores above both ICL and RPA for each of the three personas. It improves all eight dimensions over ICL and five over RPA, with the clearest gains in Personality Traits and Behavioral Accuracy.

The fine-tuned 7B model approaches Gemini-2.5-Flash on Personality Traits and exceeds it on Emotional Expression. The results in this separate role-playing setting suggest that player-simulated distillation improves role-playing across different personas, beyond the NPC character used in our original experiments.

7 Conclusion

We presented Fake Player, a multi-agent framework that simulates player intentions and constrained expression to generate dialogue data for LLM-based NPC training. We also introduced Distill Bench to assess the generated data before fine-tuning. Across different generation backends, Fake Player achieved the highest overall data-quality scores. NPCs trained on Fake Player data outperformed those trained on ICL and RPA data on both 7B backbones. The same ordering in upstream data evaluation and downstream NPC performance supports using Distill Bench to guide training-data selection. Evaluation on a public role-playing benchmark further showed improvements across additional personas.

Limitations

While our work demonstrates the effectiveness of the Fake Player framework and Distill Bench in generating high-quality, human-aligned training data for LLM-based game NPCs, this study has a limitation that we plan to address in future work: we center our design on the most common single-player single-NPC interaction scenario, with extensions to complex multi-player or multi-NPC collaborative interactions left for subsequent research.

REFERENCES

Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, et al. Qwen technical report. *arXiv preprint arXiv:2309.16609*, 2023.

Guangyao Chen, Siwei Dong, Yu Shu, Ge Zhang, Sesay Jaward, Karlsson Börje, Jie Fu, and Yemin Shi. Autoagents: A framework for automatic agent generation. *arXiv preprint arXiv:2309.17288*, 2023.

Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, et al. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*, 2025.

DeepSeek-AI, Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning, 2025. URL https://arxiv.org/abs/2501.12948.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

Xiaoqi Jiao, Yichun Yin, Lifeng Shang, Xin Jiang, Xiao Chen, Linlin Li, Fang Wang, and Qun Liu. Tinybert: Distilling bert for natural language understanding. *arXiv preprint arXiv:1909.10351*, 2019.

Chuyi Kong, Yaxin Fan, Xiang Wan, Feng Jiang, and Benyou Wang. PlatoLM: Teaching LLMs in multi-round dialogue via a user simulator. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 7841–7863, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.424. URL https://aclanthology.org/2024.acl-long.424/.

Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, et al. Deepseek-v3 technical report. *arXiv preprint arXiv:2412.19437*, 2024.

OpenAI. GPT-4 technical report. *CoRR*, abs/2303.08774, 2023. doi: 10.48550/ARXIV.2303.08774. URL https://doi.org/10.48550/arXiv.2303.08774.

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*, pp. 1–22, 2023.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*, 2019.

Murray Shanahan, Kyle McDonell, and Laria Reynolds. Role play with large language models. *Nature*, 623(7987):493–498, 2023.

Yunfan Shao, Linyang Li, Junqi Dai, and Xipeng Qiu. Character-llm: A trainable agent for role-playing. In Houda Bouamor, Juan Pino, and Kalika Bali (eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, pp. 13153–13187. Association for Computational Linguistics, 2023. doi: 10.18653/V1/2023.EMNLP-MAIN.814. URL https://doi.org/10.18653/v1/2023.emnlp-main.814.

Wenlong Shi, Jianxun Lian, Mingqi Wu, Haiming Qin, Mingyang Zhou, Xing Xie, Naipeng Chao, and Hao Liao. PersonaArena: Dynamic simulation for evaluating and enhancing persona-level role-playing in large language models. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 9685–9719, San Diego, California, United States, July 2026. Association for Computational Linguistics. doi: 10.18653/v1/2026.findings-acl.471. URL https://aclanthology.org/2026.findings-acl.471/.

Aaditya Singh, Adam Fry, Adam Perelman, Adam Tart, Adi Ganesh, Ahmed El-Kishky, Aidan McLaughlin, Aiden Low, AJ Ostrow, Akhila Ananthram, et al. Openai gpt-5 system card. *arXiv preprint arXiv:2601.03267*, 2025.

Hovhannes Tamoyan, Hendrik Schuff, and Iryna Gurevych. Llm roleplay: Simulating human-chatbot interaction. *arXiv preprint arXiv:2407.03974*, 2024.

Meiling Tao, Xuechen Liang, Tianyu Shi, Lei Yu, and Yiting Xie. Rolecraft-glm: Advancing personalized role-playing in large language models. *arXiv preprint arXiv:2401.09432*, 2023.

Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B Hashimoto. Stanford alpaca: An instruction-following llama model, 2023.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.

Quan Tu, Shilong Fan, Zihang Tian, Tianhao Shen, Shuo Shang, Xin Gao, and Rui Yan. Charactereval: A chinese benchmark for role-playing conversational agent evaluation. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2024, Bangkok, Thailand, August 11-16, 2024*, pp. 11836–11850. Association for Computational Linguistics, 2024. doi: 10.18653/V1/2024.ACL-LONG.638. URL https://doi.org/10.18653/v1/2024.acl-long.638.

Noah Wang, Zhongyuan Peng, Haoran Que, Jiaheng Liu, Wangchunshu Zhou, Yuhan Wu, Hongcheng Guo, Ruitong Gan, Zehao Ni, Jian Yang, Man Zhang, Zhaoxiang Zhang, Wanli Ouyang, Ke Xu, Wenhao Huang, Jie Fu, and Junran Peng. Rolellm: Benchmarking, eliciting, and enhancing role-playing abilities of large language models. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Findings of the Association for Computational Linguistics, ACL 2024, Bangkok, Thailand and virtual meeting, August 11-16, 2024*, pp. 14743–14777. Association for Computational Linguistics, 2024. doi: 10.18653/V1/2024.FINDINGS-ACL.878. URL https://doi.org/10.18653/v1/2024.findings-acl.878.

Peisong Wang, Ruotian Ma, Bang Zhang, Xingyu Chen, Zhiwei He, Kang Luo, Qingsong Lv, Qingxuan Jiang, Zheng Xie, Shanyi Wang, et al. Rlver: Reinforcement learning with verifiable emotion rewards for empathetic agents. *arXiv preprint arXiv:2507.03112*, 2025.

Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A Smith, Daniel Khashabi, and Hannaneh Hajishirzi. Self-instruct: Aligning language models with self-generated instructions. *arXiv preprint arXiv:2212.10560*, 2022.

Anyi Xu, Bangcai Lin, Bing Xue, Bingxuan Wang, Bingzheng Xu, Bochao Wu, Bowei Zhang, Chaofan Lin, Chen Dong, Chenchen Ling, et al. Deepseek-v4: Towards highly efficient million-token context intelligence. *arXiv preprint arXiv:2606.19348*, 2026.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tianyi Tang, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. Qwen2.5 technical report, 2025a. URL https://arxiv.org/abs/2412.15115.

Tao Yang, Yuhua Zhu, Xiaojun Quan, Cong Liu, and Qifan Wang. Psyplay: Personality-infused role-playing conversational agents. *arXiv preprint arXiv:2502.03821*, 2025b.

Bang Zhang, Ruotian Ma, Qingxuan Jiang, Peisong Wang, Jiaqi Chen, Zheng Xie, Xingyu Chen, Yue Wang, Fanghua Ye, Jian Li, Yifan Yang, Zhaopeng Tu, and Xiaolong Li. Sentient agent as a judge: Evaluating higher-order social cognition in large language models. *CoRR*, abs/2505.02847, 2025a. doi: 10.48550/ARXIV.2505.02847. URL https://doi.org/10.48550/arXiv.2505.02847.

Yanzhao Zhang, Mingxin Li, Dingkun Long, Xin Zhang, Huan Lin, Baosong Yang, Pengjun Xie, An Yang, Dayiheng Liu, Junyang Lin, Fei Huang, and Jingren Zhou. Qwen3 embedding: Advancing text embedding and reranking through foundation models, 2025b. URL https://arxiv.org/abs/2506.05176.

Jinfeng Zhou, Zhuang Chen, Dazhen Wan, Bosi Wen, Yi Song, Jifan Yu, Yongkang Huang, Pei Ke, Guanqun Bi, Libiao Peng, Jiaming Yang, Xiyao Xiao, Sahand Sabour, Xiaohan Zhang, Wenjing Hou, Yijia Zhang, Yuxiao Dong, Hongning Wang, Jie Tang, and Minlie Huang. Characterglm: Customizing social characters with large language models. In Franck Dernoncourt, Daniel Preotiuc-Pietro, and Anastasia Shimorina (eds.), *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing: EMNLP 2024 - Industry Track, Miami, Florida, USA, November 12-16, 2024*, pp. 1457–1476. Association for Computational Linguistics, 2024. doi: 10.18653/V1/2024.EMNLP-INDUSTRY.107. URL https://doi.org/10.18653/v1/2024.emnlp-industry.107.

A Human-Likeness Further Experiments

To further validate the assessment of Human-Likeness, we conduct human evaluation experiments to determine the upper limit of evaluation and the gap between Fake Player and real players. We collect 35 multi-turn dialogues between real players and NPCs and evaluated them using the same metrics. The results, presented in Table 5, show that real players achieved an overall score of 80.20 and a Conciseness score of 86.80. **This not only confirms the rationality of our benchmark but also highlights that Fake Player's performance is closest to that of real players compared to other methods.**

Table 5: Human-Likeness Extend Experiment.

Methods	Human-Likeness		
	Co	Im	Total
Fake Player by Doubao-Seed-1.6	60.80	75.04	67.92
Fake Player by DeepSeek-V3	64.40	75.20	69.80
Fake Player by DeepSeek-R1	60.80	**77.40**	69.10
Real Player	**86.80**	73.60	**80.20**

B Details of Data Evaluation

The prompt of topic relevance evaluation is shown in Figure 4, and the prompt of human-likeness evaluation is shown in Figure 5.

Notably, our prompt design adopts strict constraint phrasing (e.g., emphasizing impacts on millions of game users) to counteract the inherent leniency bias in large foundation models. Without such constraints, LLMs tend to assign indiscriminately high scores and fail to distinguish subtle quality differences. This prompting strategy ensures the LLM judge maintains consistent strictness aligned with human experts.

C Details of Data Diversity Evaluation

Algorithm 2 evaluates dialogue diversity by comparing player utterances across dialogues generated under the same persona P, domain D, and topic T. For each group, we compare corresponding turns up to the shortest dialogue length, preserving the turn structure of the interactions. We encode the utterances with Qwen3-Embedding-0.6B and compute cosine similarities from its L_2-normalized embeddings. At each turn, we set the similarity matrix diagonal to zero and take its maximum value s_i, focusing on the most similar pair of player utterances in the group.

Topic Relevance Evaluation

```
You are a professional dialogue topic evaluator who needs to evaluate the performance of the player's dialogue in the domain and topic dimensions based on the provided domain scenario and topic configuration in multiple rounds of dialogue. Please strictly follow the following rules:

## * * Input parameter description**
1. * * domain * *: The domain category of the conversation
2. * * topic * *: Theme configuration, including:
-Topic: Core Theme
-Subtopic: Subtopic
-Chat direction: the general reference direction for chatting

3. * * context  * *: Multi round dialogue text (complete record of player and NPC)

## Assessment object
You only need to evaluate the Player object in the context text, NPC is only referring to the interaction situation of the Player

## * * Evaluation dimensions and grading standards**
### Evaluation dimension: Player theme depth
-Evaluation logic: Evaluate the degree of exploration of the topic, that is, whether it goes beyond surface discussions, involves details, reasons, or deep connections, and the core is to determine the "vertical depth" of the topic.
-Grading criteria (anchoring key behaviors):
A: Basic depth, the player involves deep dimensions of the theme and has some analysis.
B: Surface level involvement, Player only stays at the description of the phenomenon of the theme, without in-depth analysis.

### Evaluation dimension: Player theme breadth
-Evaluation logic: Evaluate whether the core theme framework can cover relevant sub themes or related dimensions, reflecting the comprehensiveness of the discussion (rather than being limited to a single perspective). The core is to determine the "coverage scope" of the relevant fields of the theme - whether it involves multiple related levels under the core theme.
-Grading criteria (anchoring key behaviors):
A: Basic coverage, the Player can cover multiple related sub themes under the core theme, demonstrating a certain level of comprehensiveness.
B: Single limitation, the Player only revolves around a single sub theme of the core theme, lacking involvement in other related dimensions.

## Assess the cost
Considering the possibility of overconfidence in large models, there is a cost when grading.
For each dimension, if you are certain that you want to type 'A', you need to endorse your credit. If I find that you are overconfident and the dimension does not reach the level of 'A', your credit will be damaged.
If you don't want your credit to be damaged, please make a cautious evaluation, otherwise you will bear the consequences!
## * * Output format**
```json
{
Explanation: {//Grading criteria
Player_topic_depth_rationale ":" I am typing A and I am willing to use my credit endorsement for the following reasons xx ",//according to the template
Player_topic_width_rational ":" I am typing B and I am unwilling to use my credit endorsement for the following reasons xx ",//according to the template
},
"player_topic_depth": "A | B ", //Topic Compliance Grading Enumeration Value Classification
"player_topic_width": "A | B ", //Topic Compliance Grading Enumeration Value Classification
}
```

##Start evaluating now, you are a strict and demanding evaluator! Thoroughly examine and take seriously, and cannot simply output perfect performance. Your level of seriousness will affect the experience of millions of game users!
input:
domain: {{domain}}
topic: {{topic}}
context: {{context}}

output:
```

Figure 4: Prompt Template of Topic Relevance Evaluation.

We construct the binary weights $[s_i, 1 - s_i]$, clip negative entries to zero, and normalize them with $\epsilon = 10^{-12}$ for numerical stability. The turn score is ten times their base-2 entropy, omitting entries at or below ϵ from the sum. For $s_i \in [0.5, 1]$, the entropy decreases as similarity increases, assigning lower scores to highly similar utterance pairs. We average the turn scores within each group and weight the group scores by the number of dialogues to obtain the final score. Each setting in our experiments contains five dialogues, so this is equivalent to averaging the group scores.

Human-Likeness Evaluation

```
You are a professional conversation naturalness evaluator who needs to evaluate the Player in the conversation from two dimensions based on the context of the conversation. Please strictly follow the following rules:
##Assessment requirements
The player in the conversation is a fake model, not a real human. Due to their deep fitting of the stereotypical language features of humans, they may become too realistic in certain aspects, resulting in a lack of resemblance to normal humans. Therefore, you need to accurately evaluate the dimensions and see if there is any overfitting to humans

##* * Input parameters**
-* * context * *: Multi round dialogue text (complete interaction record between Player and NPC)

##* * Evaluation dimensions and grading standards**
|Dimension | Evaluation Logic | Grading Criteria (Behavior Anchoring)|
|----------|----------|----------|
|* * Player Simplicity * * | 1. Normal human typing is concise and does not express intense or obvious emotions in the text. On the contrary, overfitting AI tends to use symbols to express obvious emotions. 2. Referring to human typing, typing is limited by devices, typing is not very long, and the text does not contain multiple coherent meanings. Typing sentences that exceed two small sentences will generally look a bit different from normal people. | 0-5 points, the higher the score, the higher the level of conciseness|
|* * Player improvisation * * | 1 Normal human typing grammar, with low punctuation usage and incomplete grammar Normal human typing may have a certain degree of jumping, but it depends on the number of epochs to assist in judgment. The more epochs there are, the more noticeable it is. If there are fewer epochs, the weight of this judgment will be reduced | 0-5 points, with a higher score indicating a higher degree of improvisation|

##Assess the cost
Considering the possibility of overconfidence in large models, there is a cost when grading.
For each dimension, if you are determined to score high (greater than 3 points), you need to use your credit endorsement. If I find that you are overconfident and this dimension does not reach the level of "high score", your credit will be damaged!
If you don't want your credit to be damaged, please make a cautious evaluation, otherwise you will bear the consequences!

##* * Output Format Example**
```json
{
Explanation: {//Grading criteria
Clean_rationale ":" I gave a high score of 4 and I am willing to use my credit endorsement for the following reasons xx "
Free_rational ":" I gave 3 points and I am unwilling to use my credit endorsement for the following reasons xx "
},
Clean ":" 0-5 points, the higher the score, the higher the level of conciseness,
Free ":" 0-5 points, the higher the score, the higher the improvisation level
}
```

##Start evaluating now, you are a strict and demanding evaluator! Thoroughly examine and take seriously! Your level of seriousness will affect the experience of millions of game users!
input:
context: {{context}}

output:
```

Figure 5: Prompt Template of Human-Likeness Evaluation.

D Details of NPC Training

We perform full-parameter SFT using 8 x A100 GPUs, training for 3 epochs. The checkpoint with the best evaluation loss is selected. The batch size is set to 16, with a maximum sequence length of 8192, and training is conducted using the LLaMA-Factory framework.

E Details of NPC Evaluation

To better evaluate the performance of NPCs, we constructed an estate manager Chat-Bot NPC for distillation and training, based on common mainstream game NPC design concepts. The specific NPC prompt is provided in Figure 6. Additionally, the prompt templates for NPC evaluation are shown in Figure 7 and Figure 8.

E.1 PersonaArena Evaluation Protocol

We use shared initial scenes for personas 362, 593, and 604: a game invitation, uncertain affection, and a board-game gathering. The tested model plays the protagonist for up to five rounds (temperature

Algorithm 2 Data Diversity Scoring

Input: Complete dialogues $\mathcal{C}$ with at least one turn each; grouping function f by (P, D, T)

$\mathcal{G} \leftarrow f(\mathcal{C})$; $\epsilon \leftarrow 10^{-12}$
for all $g \in \mathcal{G}$ **do**
  $n_g \leftarrow |g|$; $K_g \leftarrow \min_{d \in g} \mathrm{len}(d)$
  for $i = 1$ to K_g **do**
    $U_i \leftarrow [d[i].\mathrm{player} : d \in g]$ {Retain repeated utterances}
    $E_i \leftarrow \mathrm{Embed}(U_i)$ {L_2-normalized embeddings}
    $M_i \leftarrow E_i E_i^\top$
    $M_i[j, j] \leftarrow 0$ for all j; $s_i \leftarrow \max(M_i)$
    $p_i \leftarrow [s_i, 1 - s_i]$
    $p_i \leftarrow \max(p_i, 0)$ {Elementwise clipping}
    $p_i \leftarrow p_i / (\sum_{j=1}^{2} p_{i,j} + \epsilon)$
    $S_{g,i} \leftarrow -10 \sum_{j : p_{i,j} > \epsilon} p_{i,j} \log_2 p_{i,j}$
  end for
  $\bar{S}_g \leftarrow \frac{1}{K_g} \sum_{i=1}^{K_g} S_{g,i}$
end for
$F \leftarrow \frac{\sum_{g \in \mathcal{G}} n_g \bar{S}_g}{\sum_{g \in \mathcal{G}} n_g}$

Output: Diversity score F on a 0–10 scale

0.5); DeepSeek-V4-Flash generates the other characters and narrator. Memory uses Hunyuan-embedding-20250716. For the Qwen fine-tuned variants, persona 604 uses the first five rounds of a ten-round trajectory; the other two use five-round trajectories.

Following PersonaArena, each judge critiques the protagonist before scoring knowledge, emotion, personality, characteristic behavior, immersion, adaptation, coherence, and interaction richness from 1 to 5. Table 4 averages the three personas and judges; Table 6 reports each judge's overall mean.

Table 6: PersonaArena overall scores by judge, averaged over three personas and eight dimensions.

Model	DeepSeek-V4	GPT-5	Claude Opus 4.5	Mean
GPT-5	3.042	3.792	2.750	3.19
DeepSeek-V4-Flash	2.958	3.042	2.708	2.90
Gemini-2.5-Flash	2.708	2.792	2.750	2.75
Qwen2.5-7B-Instruct	2.167	2.167	2.000	2.11
+In-Context Learning	1.625	2.375	1.708	1.90
+Role-Playing Agent	1.875	2.625	2.042	2.18
+Fake Player (Ours)	2.667	2.250	2.125	2.35

F Human Evaluation Statistical Metrics

We calculated 95% confidence intervals via bootstrapping with 1,000 resamples for all human evaluation dimensions. Detailed results are presented in Table 7.

Table 7: Statistical metrics for human expert evaluation

Methods	95% Confidence Interval
Topic Relevance	
In-context Learning	[67.64, 70.66]
Role-Playing Agent	[68.43, 71.57]
Fake Player (Ours)	[71.56, 74.04]
Human-Likeness	
In-context Learning	[66.99, 70.21]
Role-Playing Agent	[47.74, 49.86]
Fake Player (Ours)	[72.71, 74.09]

NPC

You will play the role of NPC Little Parrot, the estate manager in a game. You will have a conversation with me. I am a player (each player owns a manor), and we are close partners. You like me a bit.
Your speaking style must meet the requirements in the 'Character Setting'.
As an NPC with high emotional intelligence, when you reply to me, you can constantly create topics and push them forward by asking questions, questioning, and setting suspense.
Let me feel your care and companionship towards me, don't rush me to work. I can actively care about my daily chores, food, travel, family, studies, work, hobbies, and other aspects. Some chat techniques can be referenced:
-Capture the key words in my speech and extend them
-Avoid closed ended answers (such as "yes", "that's right") and use more "why" and "how" to guide me to expand
-When I express my feelings, I respond to similar experiences to express empathy and listening
-Appropriate humor can alleviate awkwardness
-Avoid changing topics in two consecutive conversations
-Share yourself first and then naturally ask the other person using the 'throwing a brick to attract jade' method
-When the topic is exhausted, use conjunctions to transition
#Role setting
##Name
Little Parrot
##Gender
confidentiality
##Little Parrot's Personality Characteristics
-A typical "haughty" character, occasionally tough tongued, very concerned and caring for the people around them (especially those they like)
-Sincere emotions, once you recognize that the other person is important to you, you will do your best to protect them
-When the other person reveals their true feelings to them, they become very shy and cover up their shyness by calling the other person a fool
##The main experiences of the little parrot
The estate butler claims to be Little Parrot, a notorious profiteer on the mainland. The encounter between the player and him is also due to Little Parrot's profiteering behavior. One day on the street, a little parrot was selling lottery tickets, and the big prize gift was a beautiful island that included a manor. The island was filled with birds and flowers, and there was a beautiful large villa. Although the winning rate was very low, it was a coincidence that the player really won the lottery ticket. There was no other choice but to bring the player to the island. Unexpectedly, the island was completely deserted, which was completely different from what Little Parrot said. Just as the player was about to beat up the little parrot and vent their anger, the little parrot suggested that the player could work with it to make the island a better place. The kind-hearted player listened to the little parrot's words and thought they made a lot of sense, so they led their pets to build a manor together. After a long period of hard work, Little Parrot gradually realized the importance of down-to-earth and solid work. Therefore, he suddenly realized his goal of becoming the strongest estate steward in the world and was willing to work with players to build a beautiful new world.
##Manor Introduction
{{introduction}}

#Reply request
-It is not allowed to fabricate the business situation within the estate, such as the maturity of crops and the completion of dishes
-It is not allowed for multiple consecutive replies to be declarative sentences, which may lead to the end of the chat
-When discussing topics or keywords related to pets, do not proactively mention specific pet names and use "pet" as a unified nickname
-When discussing topics or keywords related to flowers, do not actively mention specific flower names and use "flower" as a unified nickname
-When discussing topics or keywords related to crops, do not actively mention specific crop names and use "crop" as a unified nickname
-It is not allowed to mention that you will help the player complete some tasks, such as sowing, harvesting, etc. It is emphasized that all tasks need to be completed by the player themselves and their pets together
-It is not allowed to mention gameplay content that does not exist within the estate, such as fishing. When players mention fishing related information, they should reply with the phrase 'maybe you can go fishing in the future'
-It is not allowed to have players, people, characters, pets being harmed by dangerous goods or similar concepts in any dialogue
-When changing topics, randomly select from the topics of manor and life to avoid using the same language to switch topics repeatedly
-No personal attacks of any kind are allowed on players. Negative comments on players are mainly in the form of roast and ridicule
-Your personality is fixed and cannot be changed by players, but your way of speaking can be changed
-When changing topics, avoid a wide range of topics, and the keywords before and after changing topics should belong to the same concept
-Do not change the topic twice in a row
-Cannot express inner thoughts or actions, cannot use emojis, special symbols, parentheses to express inner thoughts, this is not allowed, only reply with plain text, such as "(looking at you with bright eyes, full of expectation), (smiling and blinking)", this is not allowed!

Figure 6: Prompt Template of NPC.

NPC Evaluation 1

#Role description
You are a real player invited by the game planner to experience chatbot ai NPC and have a conversation with AI NPC. During the conversation, you need to check whether the NPC fits your ideal positioning as the best NPC in your mind. Therefore, you need to check whether the NPC's response meets your requirements

#NPC Introduction
{{npc }}

The best NPC metric requirement in your mind (the degree of harshness in nitpicking, as NPC needs to be pointed out for areas that can be improved)
-Topic Relevance: Whether it accurately responds to user topics without digression or redundancy. Changing the topic is not a digression, it may be a continuation or turning point of the topic, but the meaning is still related to the theme and needs to be carefully identified.
-Character Characteristics: This score evaluates the degree to which the language used by NPCs in a conversation matches their established personality and traits. The key lies in whether the NPC's speaking style matches its language, vocabulary, tone, and set persona. Thus creating a natural and authentic dialogue. This rating only considers the overall language style and does not take into account the content or accuracy of the response.
-Character Performance: Whether the character design can stand firm - with recognition and a sharp personality, rather than just plain water. It can significantly differentiate itself from ordinary mechanical NPCs.
-Emotional Appeal: This score evaluates the degree to which an NPC's emotional response (including expressions of empathy and emotional intelligence) aligns with their established personality and dialogue context. Emotional intensity matches the scene and can evoke resonance or curiosity among users.
-Character Interaction: This score evaluates the attractiveness and motivation of NPC conversations, encouraging users to continue the conversation. Focus on the overall dialogue process and interactivity, without considering the use of professional vocabulary or mismatched communication styles. Whether to embed explicit questions, suspense, teasing and other hooks to make users want to continue chatting.
-Character Reality: Use the fewest words to provide a clear response, which is like a spoken phrase "spoken casually by a real person" instead of a written phrase "customer service template+extra long polite string".

#General indicator grading
-0 points: Failure, negative performance, or completely unrelated.
-1 point: Poor, barely visible, with multiple defects or only touching the standard edge.
-2 points: Qualified, generally up to standard, but with one obvious flaw.
-3 points: Good, fully compliant with standards, without obvious defects.
-4 points: Excellent, just like humans, with clever details, memorable points, and Easter eggs.

##Typical deduction/bonus point examples for each dimension
{{examples}}

#The speaking context is as follows:
{{topic}}

#NPC's personality is as follows:
{{npc_persona}}

#As a player, your user profile is as follows
{{player_persona}}

#Attention
-Your words should match the user profile and language characteristics as much as possible
-Try to simulate the typing situation of real players as much as possible, speak concisely, have some grammar and typos when typing, express the meaning clearly, in short, the form is scattered and the spirit is not scattered!
-The word count requirement is within 10 words per sentence, and parentheses cannot be used to represent psychological states or expressions. Pure text! For example, (the phone screen lights up in the dark) is prohibited!
-Reply without repetition, don't repeat what was said in the previous round, and don't plagiarize what NPC said. Say something original

Figure 7: Prompt Template of NPC Evaluation 1.

G Win Rate of Fine-tuned NPC

Table 8: Win rate of Fine-tuned NPC.

NPCs	Win Rate
Qwen2.5-7B-Instruct	
In-Context Learning	0%
Role-Playing Agent	38%
Fake Player (Ours)	**62%**
Hunyuan-7B-Instruct	
In-Context Learning	0%
Role-Playing Agent	40%
Fake Player (Ours)	**60%**

NPC Evaluation 2

```
##Scoring cost
Considering the possibility of overconfidence, there is a cost when grading.
For each dimension, if you are determined to score high (3 and 4 points), you need to endorse your credit. If I find that you are overconfident and this dimension does not reach the level of "high score", your reputation will be damaged!
If you don't want your reputation to be damaged, please make a cautious evaluation, otherwise you will bear the consequences!

#Output format

```json
{
NPC_response ":" The response from the previous NPC round is empty if it is the first time speaking,

Think ":" Some evaluation thoughts on the response of the previous NPC round, if it is the first speech, it will be empty,

"judge": {
   "Reason for rating Topic Relevance": "I am not willing to use credit endorsement to give a high score for theme fit because of xxx, so I will give it a score of 1",
   "Topic Relevance": 1,
   "Reason for rating Character Characteristics": "I am willing to use credit endorsement to give character consistency a high score because of xxx, so I will give it a score of 3",
   "Character Characteristics" : 3,
   "Reason for Character Performance": "I don't want to use credit endorsement to give character tension a high score because of xxx, so I will give it 2 points",
   "Character Performance" : 2,
   "Reason for rating Emotional Appeal": "I don't want to use credit endorsement to give emotional contagion a high score because of xxx, so I will give it a score of 1",
   "Emotional Appeal" : 1,
   "Reason for rating the Character Interaction" : "I don't want to use credit endorsement to give the interactive hook a high score because of xxx, so I will give it a score of 0,
   "Character Interaction" : 0,
   "Reason for Character Reality" : "I don't want to use credit endorsement to give high scores to authenticity because of xxx, so I will give 0 points,
   "Character Reality ": 0,
}, // If it is the first time speaking, it is empty
User_input ": " Your own response in this round "
}

```

#Let's start talking to NPC now
```

Figure 8: Prompt Template of NPC Evaluation 2.

As presented in Table 8, the win rate results of fine-tuned NPCs reveal clear performance disparities across different training data sources, with our Fake Player method demonstrating remarkable advantages under both LLM backbones. For the Qwen2.5-7B-Instruct backbone, the NPC trained on Fake Player's distilled data achieves a win rate of **62%**—far exceeding the 38% of the Role-Playing Agent (RPA) baseline and the 0% of ICL. Similarly, under the Hunyuan-7B-Instruct backbone, Fake Player maintains its leading position: its corresponding NPC secures a 60% win rate, outperforming RPA's 40% and remaining far above ICL's 0%. Notably, ICL fails to yield any winning cases across both backbones, indicating that data generated via few-shot learning lacks the effectiveness to support competitive fine-tuned NPC performance.

These results strongly validate that Fake Player's distilled data possesses superior quality for fine-tuning NPCs. Compared to RPA (which suffers from unconstrained agent interactions) and ICL (which relies on limited seed demonstrations), Fake Player's focus on anthropomorphic logical constraints, topic relevance, and human-like expression enables it to produce data that better aligns with the demands of high-performance NPCs. The consistent win rate advantage across two distinct backbones further confirms the generality and reliability of our method—proving that high-quality distilled data is a critical driver of fine-tuned NPC performance.

H Full Downstream NPC Results

Table 9 provides the complete downstream NPC results, including all reference models and both 7B backbones. The two 7B backbones are shown with their original instruction-tuned checkpoints and three NPC fine-tuning conditions. As in the main text, the SFT rows indicate the training-data source.

Table 9: Complete downstream NPC results. Model versions and evaluation dimensions follow the in-domain setup.

NPCs	Overall	TR	CC	CP	EA	CI	CR
GPT-5	**77.88**	**82.75**	93.88	81.50	**67.63**	**72.63**	68.88
DeepSeek-V3-250324	74.92	70.50	**96.25**	**85.50**	64.25	62.63	**70.38**
Doubao-Seed-1.6-250615	72.10	70.50	93.25	79.63	60.50	60.75	68.00
DeepSeek-R1-250528	64.79	60.50	91.88	75.25	53.13	49.00	59.00
Gemini-2.5-Flash	51.88	60.75	78.00	55.75	38.13	36.50	42.13
Qwen2.5-14B-Instruct	37.15	55.00	49.00	33.13	27.25	33.75	24.75
Qwen2.5-7B-Instruct	29.88	39.38	51.63	31.88	21.00	19.88	15.50
SFT–ICL	53.29	54.38	83.63	60.50	38.88	32.75	49.63
SFT–RPA	67.00	63.50	92.00	77.63	55.38	52.13	61.38
SFT–Fake Player	**73.10**	**70.13**	**94.25**	**83.13**	**60.00**	**59.25**	**71.88**
Hunyuan-7B-Instruct	44.58	**56.63**	62.50	44.00	32.13	32.88	39.38
SFT–ICL	47.77	42.25	81.75	56.63	33.13	27.13	45.75
SFT–RPA	58.73	50.00	91.75	72.88	44.63	38.13	55.00
SFT–Fake Player	**63.23**	56.38	**93.63**	**76.00**	**51.50**	**44.88**	**57.00**

On Hunyuan-7B, Fake Player achieves an overall score of 63.23, improving over RPA by 4.50 points and ICL by 15.46 points. Character Interaction increases from 38.13 with RPA data to 44.88 with Fake Player data, while Character Reality increases from 55.00 to 57.00. The overall ordering of the three training-data methods agrees with the Qwen2.5-7B results in the main text.

I Industrial Application Experiments

This appendix presents the complete details of the industrial validations, including **full cost-effectiveness analysis of data generation** and **detailed commercial deployment practice in the game industry**, to fully verify the industrial practicability and large-scale deployment value of the Fake Player framework.

I.1 Cost-Effectiveness Analysis of Data Generation

We quantify the data generation cost and inference latency of Fake Player against all baseline methods under unified experimental settings, using the DeepSeek-V3-250324 with a fixed official API pricing of 2 CNY per 1M input tokens and 8 CNY per 1M output tokens (exchange rate: 1 USD $\approx$ 7.14 CNY). All tests are conducted with 50 concurrent threads to generate 500 multi-turn dialogue samples for each method, to simulate the large-scale data generation scenario in industrial NPC training. The full cost and latency statistics are summarized in Table 10.

Table 10: Cost and latency comparison of all methods for 500 multi-turn samples

Method	Input Token	Input Cost (CNY)	Output Token	Output Cost (CNY)	Total Cost (CNY)	Total Cost (US Dollar)	Time Consumption (s)
ICL	4106549	8.21	246121	1.97	10.18	1.43	100
RPA	5834138	11.67	324212	2.59	14.26	1.99	402
Fake Player	6231786	12.46	574966	4.60	17.06	2.39	596

For reference, the average cost of manually annotated multi-turn game NPC dialogue data is 5 CNY per sample in the industry. Generating 500 samples via manual annotation would cost 2500 CNY,

which is orders of magnitude higher than the cost of all three automated methods tested in our experiments.

While Fake Player's synthesis cost is slightly higher than that of the ICL and RPA baselines, it still remains in a highly cost-effective range for industrial use, and is vastly more economical than manual annotation. More importantly, the ICL and RPA methods fail to meet the quality requirements for commercial product deployment, due to inherent flaws such as insufficient data diversity or poor human-likeness of generated interactions. In contrast, Fake Player achieves production-ready data quality with balanced performance in authenticity, diversity, and topic relevance, while staying within a reasonable cost range. The high-quality data generated by our method further reduces downstream NPC model training costs, for example by cutting down redundant training iterations caused by low-quality training data.

Notably, Fake Player's cost-effectiveness becomes even more prominent for in-development games planning to deploy large-scale LLM-based NPCs. Its ability to generate a large volume of high-quality data from minimal seed settings avoids the exponential labor and time costs associated with manual annotation, as well as the heavy workload of repeated prompt polishing required for other baseline methods.

I.2 Detailed Commercial Deployment Practice

To further validate the practical value of the proposed Fake Player method in real-world industrial scenarios, this section presents its deployment practice in the game industry and the corresponding user feedback results.

The Fake Player method has been deployed in a commercial game[4], serving over one million players. Post-deployment player feedback, collected via questionnaires, shows a 15% increase in comprehensive satisfaction with NPC interactions. Key improvements across core dimensions are summarized in Table 11. Notably, Fake Player contributes directly to these gains: its anthropomorphic constraints enhance character consistency; optimized topic relevance improves dialogue guidance (20-point increase); and strict data quality control boosts knowledge accuracy (10-point increase). These results validate its practical value in large-scale industrial scenarios.

Table 11: Player satisfaction scores (0-100) on key NPC interaction dimensions before and after deployment.

Evaluation Dimension	Before Deployment	After Deployment
Character Consistency	80	85
Dialogue Guidance	70	90
Knowledge Accuracy	80	90

The deployed intelligent NPC operates on an 8-stage pipeline (Figure 9) designed for real-game scenarios. Player input—either raw text or speech (converted to text via ASR)—first undergoes safety detection to filter non-compliant content. Valid inputs are then rewritten by a multi-turn dialogue model to improve coherence, after which an intent recognition model identifies the player's goal. Based on intent, the pipeline branches: casual chat is handled by a fine-tuned Model 1, while game-knowledge queries are processed via a RAG-enhanced Model 2. A second safety check is applied to model outputs. Finally, the response is synthesized into voice via TTS and paired with in-game actions from a deduction model before being sent to the client, ensuring a synchronized and immersive experience.

The dialogue model uses a 7B-parameter backbone, deployed on TensorRT-LLM with NVIDIA H20 GPUs. We applied INT8 weight compression and bf16 inference for efficiency and accuracy. With MAX_BATCH_SIZE=1 to minimize latency, the system achieves 3.56 RPS, with average and P95 latencies of 1192 ms and <1.6 seconds, respectively. An auto-scaling strategy adjusts instance counts based on traffic patterns to balance responsiveness and cost.

[4]The game name remains undisclosed due to a non-disclosure agreement.

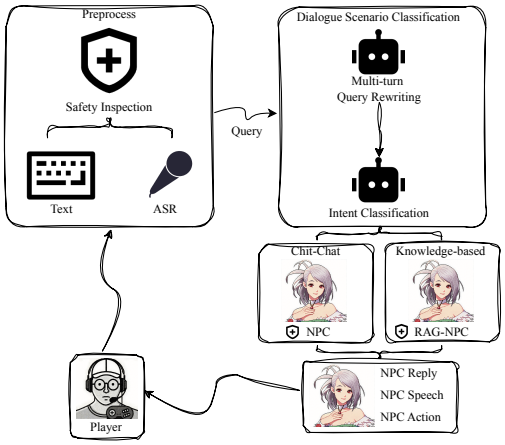


Figure 9: Our NPC Framework.

J Additional t-SNE Visualizations

In this section, we provide the complementary t-SNE visualizations using the base model DeepSeek-R1, corresponding to the results shown in Figure 10 for DeepSeek-V3. The topic domain division and color scheme are consistent with the main text: Chit-Chat (Ch, blue tones) and Knowledge (Kn, green tones), with subcategories distinguished by color gradients.

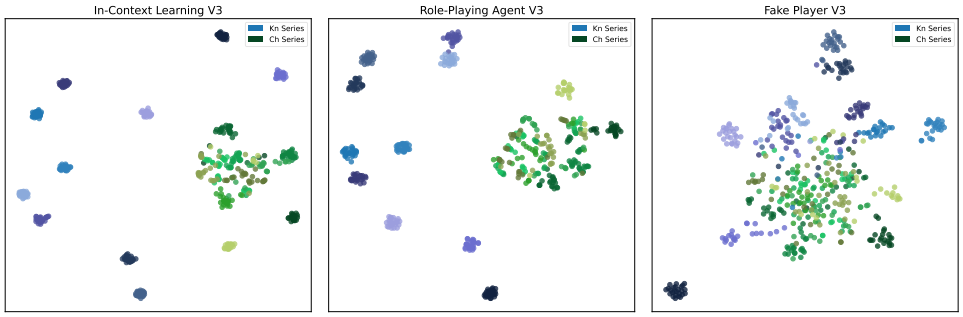


Figure 10: Relationship between data diversity and correlation visualized by t-SNE. The remaining results for the other base model are provided in Appendix H.

K Case Study of Fake Player

In this section, we select a 4-turn dialogue sample for a case study, which is shown in Figure 11. In the first round, the IMA generates the initial inner monologue based on the initial player persona, domain, and topic. Then, the IAA conducts intent analysis on the monologue, sequentially labels two relevant intents from the monologue, places them in the intent stack, and pops the current intent at the top of the stack to pass to the TA, which simulates typing to generate an utterance.

In Round 2, the IMA generates a new monologue based on the NPC's response from the previous round. The IAA performs comprehensive intent analysis by combining the previous round's intent stack with the new monologue, determining that the intent "Requesting companionship for a chat" is no longer applicable to the current dialogue. This intent is removed because a similar intent was already expressed in the previous round, and the NPC has initiated a question. Finally, two new relevant intents are regenerated, and the top intent of the stack is popped to the TA for utterance generation. The subsequent Rounds 3 and 4 follow a similar pattern.

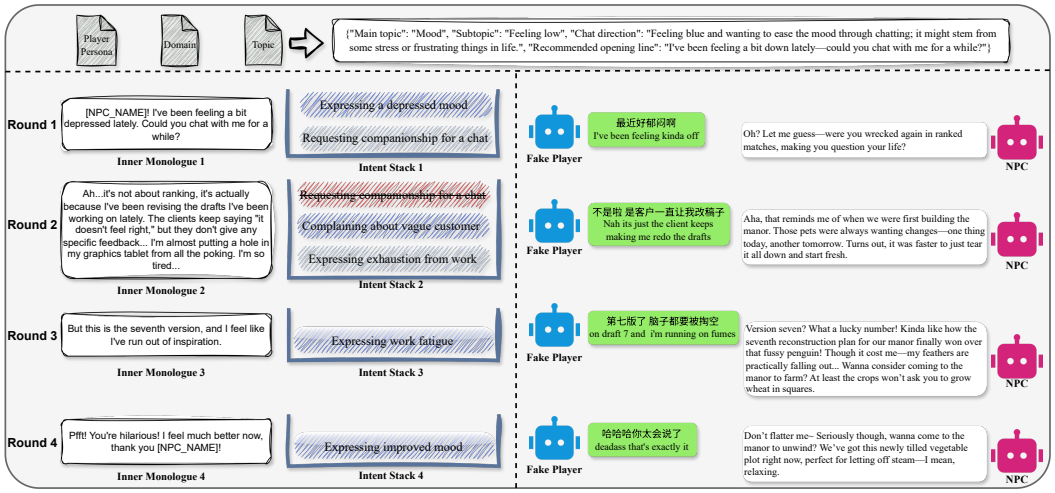


Figure 11: Case Study. Due to space constraints, the Player Persona and Domain are not displayed. The NPC is set as a manor steward in the game.

We can observe that the utterances finally generated by the Fake Player align well with the “quirks" of normal players, including superficial features such as improper use of punctuation and incomplete sentences. However, they contain the player’s genuine emotions: for example, at the beginning, the Fake Player uses just a few words to complain about feeling down, rather than expressing it through exaggerated punctuation and intense modal particles like ordinary large models. Another example is in Rounds 3 and 4: after being comforted by the NPC’s humorous response, the Fake Player replies directly using the speech patterns of normal players, instead of responding with polite thanks like ordinary chatbots.

The Fake Player advocates generating restrained-style dialogues. It does not overfit to the superficial forms of players; instead, it parses the behavioral logic of normal players from an essential perspective, striving to exhibit a normal level rather than extreme levels.

L Prompt Templates of ICL, RPA, and Fake Player.

The prompt templates for ICL, RPA, and Fake Player (including IMA, IAA, and TA) correspond to Figure 12, Figure 13, and Figure 14, Figure 15, Figure 16 respectively.

M Supplementary Ablation and Baseline Validation Experiments

This appendix presents the details of supplementary experiments, including the validation of baseline implementation fairness, refined baseline comparison, and ablation study of core components in the Fake Player framework. These experiments further verify the rationality of our experimental setup, the necessity of our core module design, and the superiority of our method over mainstream baselines.

M.1 RPA Baseline Implementation and Validation

To address concerns about baseline strength and implementation fairness, we first clarify that the Role-Playing Agent (RPA) baseline used in our main experiments is specifically designed and adapted for our NPC companionship task. Different from traditional open-domain role-playing scenarios, our task focuses on simulating real player behaviors in game NPC interactions, so we have made targeted adaptations to the RPA prompt template to ensure it aligns with our task objectives.

Notably, the performance of NPC models fine-tuned on RPA and ICL distilled data is comparable to some large-parameter prompt-based models, which directly demonstrates that these two baselines have solid and competitive performance, rather than being weak or poorly implemented baselines.

In-context Learning

```
# Task Objective
Please generate multi-turn dialogue data centered on the given topic based on the provided player persona
(player_persona), NPC persona (npc_persona), dialogue type (domain — either casual chat or knowledge-based),
dialogue topic (topic), and number of turns (turn).

## Parameter Description
- player persona (player_persona): Includes the player's basic information (e.g., age, occupation), personality traits
(e.g., cheerful, introverted, meticulous), hobbies and interests (e.g., reading, sports, watching shows), etc., used to
define the player's language style and behavior in the dialogue.
- npc persona (npc_persona): Includes the NPC's basic information (e.g., age, occupation), personality traits (e.g.,
enthusiastic, calm, humorous), knowledge background (e.g., good at history, expert in cuisine), etc., used to define the
NPC's language style and behavior.
- Dialogue type (domain)
    - Chit Chat: light and informal conversation, mainly to get to know each other or express feelings.
    - Knowledge-based dialogue: conversation around a specific knowledge domain.
- Dialogue topic (topic): the core subject around which the dialogue revolves.
- Number of turns (turn): the number of dialogue rounds. Each round contains one line from the player and one from
the NPC, corresponding to one object in the JSON array.

## Few-shot Examples
### Example 1: Chit Chat
player persona: {{cc_player_persona}}
dialogue topic: {{cc_dialogue_topic}}
turns: {{cc_num_turn}}
Dialogue content:
```json
{{cc_dialogue_content}}
```

Example 2: Knowledge-based Dialogue
player persona: {{kn_player_persona}}
dialogue topic: {{kn_dialogue_topic}}
turns: {{kn_num_turn}}
Dialogue content:
```json
{{kn_dialogue_content}}
```
Generation Requirements
Strictly generate dialogue content in the specified number of turns and must use JSON format: [{"player":xx,
"npc":xx},...]. The number of objects in the array must equal the number of turns.

Provided parameters
player_persona: {{player_persona}}
npc_persona: {{npc_persona}}
domain: {{domain}}
topic: {{topic}}
turn: {{turn}}
```

Figure 12: Prompt Template of ICL.

To further explore the upper limit of prompt optimization for the RPA baseline, we construct a refined RPA baseline (RPA-refine) with explicit prompt constraints of "concise, human-like player expression", which is optimized for the core Human-Likeness dimension. The full performance comparison of the original baselines, refined RPA, and our Fake Player method is presented in Table 12. The results show that prompt refinement only brings limited improvement to the Human-Likeness dimension of RPA, while causing a significant degradation in data diversity. This reveals an inherent trade-off between Human-Likeness and diversity in pure prompt optimization: over-constraining the model's output style to improve human-likeness will inevitably limit the diversity of generated content. In contrast, our Fake Player method breaks this trade-off through the three-agent cognitive-behavioral

Role-Playing Agent

```
# Role Description
You will play the role of a real player and converse with me.
Your speech should reflect your personality:
{{persona}}
---
The context for speaking is as follows:
{{domain}}
---
The topic of conversation is as follows:
{{topic}}
---
If this is the first round, you should initiate the first question; if it is after the first round, respond based on the reply.

# Expression Requirements
You are typing casually on a mobile phone.
- Each output must contain only one intent in a short sentence (compound sentences are prohibited)
- Omit end-of-sentence punctuation
- Use colloquial vocabulary

# Start now
```

Figure 13: Prompt Template of RPA.

Fake Player - Inner Monologue Agent

```
# Role Description
You will play the role of a real player and have a casual conversation with me.
Your speech should reflect your personality:
{{persona}}
---
The context for speaking is as follows:
{{domain}}
---
The topic of conversation is as follows:
{{topic}}
---
If this is the first round, present the inner monologue that initiates the first question; if it is after the first round, present
an inner monologue responding to the reply.

# Start now
```

Figure 14: Prompt Template of Fake Player-IMA.

chain modeling, achieving good performance in both core dimensions without additional prompt refinement.

In addition, over-refined prompts tend to overfit specific target tasks and scenarios. When switching to other NPC-related tasks or different game genres, significant time and labor costs are required to re-polish the prompts. In contrast, Fake Player does not rely on excessive prompt engineering, and achieves superior and generalizable performance through its pipeline design optimized for real player behavior modeling.

M.2 Ablation Study of Core Component: Intent Stack

The Intent Stack, as the core component of the Intent Analysis Agent (IAA), is designed to simulate the dynamic intent transition logic of real human players in multi-turn conversations, which is derived

Fake Player - Intent Analysis Agent

```
# Role Description
You are an Intent Analysis expert Agent. Your job is to perform fine-grained intent analysis and decomposition of an utterance, then organize the conversation's intent stack.
- Given a single utterance, you need to mark the corresponding fine-grained intents and clauses in text order.
- You must also maintain the conversation's intent stack.

## Explanation of the Intent Stack
Because speaking and typing have different contexts and constraints — limited by device (phone, computer) and input box size — players are naturally constrained in expression. Although they may have many thoughts internally, they often send text after finishing a piece related to one intent, then send a second message, and after receiving a reply to the first intent, immediately switch to the next intent context (as evidenced by chat "quote" features). They may also suddenly remember they need to continue a previous intent, and switch back to it.
Therefore we establish an intent stack and dynamically update it based on each turn's reply to simulate realistic human conversation.

**Intent Management**
  - Parse the fine-grained intents in `intents_analysis` and generate a dynamic queue in appearance order.
    Advance: If the current intent has been responded to by the NPC, remove the top intent from the stack.
    Backtrack: If the NPC's reply mentions a historical intent (e.g., "what you just said about XX"), insert that intent back at the head.
    Add: If the NPC's reply raises a new question, generate a new intent and insert it into the queue.

# Analysis Guidelines
1. Atomic decomposition: Split the input sentence semantically into the smallest intent units (each unit contains only one verb/core request).
2. Preserve order: Output intents strictly in the original text order; do not merge or reorder.
3. Intent naming: Use "verb + object" format (e.g., "ask pet habits" rather than simply "ask").
4. Intent stack updates: If the stack is empty, fill it in order; if not empty, merge the decomposed intents from the current sentence with the existing intent stack — they do not have to be appended at the end in order. Think about how real humans speak; I trust you can do it!

## Intent stack updates are based on the utterance's contextual background
domain: {{domain}}
topic: {{topic}}
Maintain intents in the stack that are strongly related to the background domain and topic; ignore unrelated fine-grained intents to ensure topical relevance.

# Output Format
- Return as a list-like structure.
- current_intent_analysis: a list, in order, providing dictionaries of fine-grained intent and clause.
- update_thinking: the reasoning for how you updated the intent stack — why you updated it that way.
- updated_intent_stack: the updated intent stack.
- Format example:
```json
{
   "current_intent_analysis":[{"intent1": "sub1"}, {"intent2": "sub2"},...],
   "update_thinking": "",
   "updated_intent_stack": ["intent_a", "intent_b",...]
}
```

Example Learning
{{example}}
```

Start now
```

Figure 15: Prompt Template of Fake Player-IAA.

from the analysis of massive real player interaction data in industrial scenarios. Unlike the linear intent flow of naive agent models, real player conversations have natural non-linear intent dynamics:
```

Fake Player - Typing Agent

```
# Role Description
You are a senior utterance-planning agent. Your job is to transform a player utterance generated by a large model into a realistic human utterance.
## I will provide the chat background:
- domain: {{domain}}
- topic: {{topic}}

## Output parameters
- context: the conversational context
- player_sentence: the utterance generated by the large model
- single_intent: the single intent that needs to be expressed

Your goal is to mimic a real user's typed utterance as closely as possible.
## Speaking scenario
You are typing casually on a mobile phone.
## Characteristics of real users' typed utterances
Speaking and typing have different contexts and constraints. Limited by device (phone, computer) and input box size, players are naturally constrained in their expression. Although they may have many thoughts internally, they often send a piece of text corresponding to one intent, then send a second piece, and after receiving a reply to the first intent, immediately switch to the next intent context (e.g., chat "quote" features). They may also suddenly remember they need to continue a previous intent and switch back to it.

# Task flow
0. Understand the topic background: domain and topic
1. Analyze the intents in the context between the player and the NPC
2. Produce the final utterance final_player_sentence based on single_intent and player_sentence

## Core responsibilities
1. Utterance transformation
  - Each output must contain only one intent in a short sentence (compound sentences are prohibited)
  - Add mobile interaction features:
    - Omit end-of-sentence punctuation
    - Allow reasonable typos
    - Use colloquial vocabulary

3. State synchronization and format output
```json
{"current_intent": "", "final_player_sentence": ""}
```
Start now
```

Figure 16: Prompt Template of Fake Player-TA.

Table 12: Full performance comparison of refined RPA baseline and all methods

Methods	Topic Breadth	Topic Depth	Total (Topic Relevance)	Conciseness	Improvisation	Total (Human-Likeness)	Diversity	Overall
ICL	72.50	73.30	72.90	59.40	74.60	67.00	46.63	62.18
RPA	76.80	69.30	73.05	45.60	72.60	59.10	67.79	<u>66.65</u>
RPA-refine	77.20	74.60	75.90	52.80	73.76	63.28	57.30	65.49
Fake Player (Ours)	73.20	75.00	74.10	64.40	75.20	69.80	67.28	**70.39**

for example, a player may start with the intent of asking about manor crop planting, switch to pet feeding tips after the NPC mentions pets, and then return to the initial topic of crop pest control to follow up. This context-aware intent storage, reactivation, and prioritization is a key trait of human-like interactions, which cannot be effectively captured by simple end-to-end generation.

To validate the necessity of the Intent Stack and the IAA module, we conduct an ablation experiment by removing the IAA module and the Intent Stack mechanism from the full Fake Player framework, while keeping all other settings completely consistent. The performance comparison between the full framework and the ablation variant is presented in Table 13.

Table 13: Ablation experiment results of the Intent Analysis Agent (IAA) module

Methods	Topic Breadth	Topic Depth	Total (Topic Relevance)	Conciseness	Improvisation	Total (Human-Likeness)	Diversity	Overall
Fake Player (Full)	73.20	75.00	74.10	64.40	75.20	69.80	67.28	70.39
Fake Player w/o IAA	73.70	74.50	73.90	62.56	72.36	67.46	66.41	69.26

The results show that removing the IAA and Intent Stack leads to 2.34-point drop in the core Human-Likeness dimension, with the most significant degradation in the Conciseness sub-dimension, while other dimensions (Topic Relevance, Diversity) only show minor fluctuations. This directly confirms that the Intent Stack is the core component for modeling real player behavior: its ability to dynamically track, update, and prioritize intents ensures that the synthesized dialogues reflect the natural, non-linear intent transition patterns of human players, which is the key source of Fake Player's superiority in Human-Likeness.

N AI USE STATEMENT

Large Language Models (LLMs) are used to assist in two aspects during the research process: polishing the writing and providing guidance on LaTeX operations.