IADIS International Journal on WWW/Internet
Vol. 24, No. 2, pp. 93-110
ISSN: 1645-7641

PLAYER-IMITATING RIVAL AGENTS FOR DYNAMIC DIFFICULTY ADJUSTMENT IN LLM-BASED CONVERSATIONAL GAMES

Yunosuke Tsunoda, Kosei Nakazono and Qiang Ma
Kyoto Institute of Technology, Matsugasaki, Sakyo-ku, Kyoto 606-8585, Japan

ABSTRACT

Conversational games built on large language model (LLM) agents give players an unprecedented freedom of expression, and that freedom makes *difficulty* hard to author in advance: a single fixed *difficulty* risks being trivial for a fluent player and impenetrable for a hesitant one. Our approach makes the obstacle the player faces a rival agent that imitates the player. The rival is driven by a *persona token* fusing the player's conversation history, a dynamically maintained knowledge graph scoped to the player's access level with a small perturbation, and a conversational-style dataset that grows as the player speaks. Because the rival's capability is defined relative to the player rather than by fixed parameters, the challenge is intended to track the player instead of the designer's assumptions. In this paper, we formalize dynamic difficulty adjustment (DDA) in a conversational game as a parity condition between the player and the rival agent, and propose a mechanism that maintains it. We also develop a prototype system, a Japanese dating simulation. We conduct a user study (N = 9, three within-subject conditions) and analyze it turn by turn rather than by session aggregate, showing the imitation to be measurable: the imitating rival's rate of question asking stays on average within 0.07 of the player's own, against 0.31 for a knowledge-matched rival without persona imitation ($p = .002$), and the two rivals' trajectories over a session diverge, the imitating one converging on the player while the knowledge-matched one drifts away ($p = .039$). We also show that imitation changes the character of difficulty, not only its level: the rival withholds progress instead of punishing the player, making relationship setbacks more than four times rarer, although that difference falls short of significance ($p = .063$), and players spend 41% longer composing each message against it than against the knowledge-matched rival. Difficulty also tracks the individual player, although with nine participants that last result is indicative rather than established.

KEYWORDS

LLM Agents, Conversational Games, Dynamic Difficulty Adjustment, Rival Agent, Persona Token, Knowledge Graph

1. INTRODUCTION

Large language models have changed what game characters, agents and content can be (Gallotta et al., 2024; Park et al., 2023; Värtinen et al., 2024; Wang, G. et al., 2024). In dialogue-centered genres, the change is especially sharp: a character driven by an LLM can respond to an utterance no designer anticipated, in a register no designer wrote (Gao and Emami, 2023; Li et al., 2023; Shanahan et al., 2023; Shao et al., 2023; Wang, N. et al., 2024).

Script-based dialogue severely bounds expression, but it buys something in return: the designer knows exactly how hard the game is, having written every path through it. Generative dialogue (e.g., van Stegeren and Myśliwiec, 2021) gives that up. Once a player may say anything, the *difficulty* of saying the *right* thing depends on the player's own fluency, vocabulary, and social intuition, which vary enormously between individuals. Dynamic difficulty adjustment (DDA) is therefore arguably less a refinement here than a precondition for the genre to work.

Our earlier work attempted difficulty control by giving a rival agent less knowledge than the player (Tsunoda et al., 2025). It did not work well. Restricting what an LLM knows does not restrict how well it argues, and the model's fluency appeared to dominate the outcome whatever knowledge level we set. The present work takes the opposite approach. Rather than handicapping the rival, we make it an imitation of the player. A rival that reasons as the player reasons, knows what the player knows and speaks as the player speaks should, by construction, be an evenly matched opponent — and an evenly matched opponent is what flow theory identifies as appropriate difficulty (Chen, 2007; Csikszentmihalyi, 2008). Preliminary reports of this method appear in Tsunoda et al. (2026a, 2026b); Section 2.3 states what the present paper extends.

The paper makes four contributions.

- We formalize difficulty control in a conversational game as a parity condition between the player and the rival, and specify a mechanism that maintains that condition without any difficulty parameter (Sections 3 and 4).
- We document a complete implementation at the level of prompts, data schemas and turn scheduling, so that the mechanism can be reproduced inside a working system (Section 5).
- We establish that the imitation is real and measurable by comparing every rival utterance with the player utterance it competed against, and find it partial in a way that shows which parts of a persona transfer (Section 6.3).
- We show that imitation changes the character of the difficulty produced, not only its level (Section 6.4), and extend the conference version's finding that the difficulty tracks the individual player with correlation and within-participant regression analyses (Section 6.5). The evidence comes from a within-subject user study in which nine participants each played three variants of the same game.

2. RELATED WORK

2.1 Role-Playing Language Agents

Among role-playing language agents (Chen et al., 2024; Tseng et al., 2024), which range from agents given a compact professional profile (Hong et al., 2024) to reconstructions of specific characters (Shao et al., 2023), individualized personas — an agent conditioned on one person's history — are the closest relatives of our rival. The line runs from PersonaChat's explicit persona sentences (Zhang et al., 2018) through benchmark-driven personalization (Salemi et al., 2024) to lightweight conditioning: Liu et al. (2025) generate a user-specific embedding from past behavior, personalizing outputs without touching model parameters, and Kirk et al. (2024) set out a taxonomy of what such personalization buys and costs. Style is treated separately: Roy et al. (2023) reduce a conversation to a style-free form and rewrite it in a target style from few-shot demonstrations, and we adopt their conversation style transfer (CST) for the rival's register.

2.2 Dynamic Difficulty Adjustment

DDA rests on flow theory — engagement is highest when challenge and skill are matched (Chen, 2007; Csikszentmihalyi, 2008; Sweetser and Wyeth, 2005) — and Zohaib (2018) reviews how games pursue that match. The classical mechanism is resource manipulation: Hunicke and Chapman (2004) and Hunicke (2005) predict when a player is about to struggle and adjust the supply of health and ammunition and the strength of enemies, and Vicencio-Moreira et al. (2015) balance players of different skill in a first-person shooter mainly through dynamic aim assistance. Yannakakis and Togelius (2011) generalize adaptation to content generation.

Closest to us is opponent-side adaptation. Fuchs et al. (2024) pair an imitation-learning agent that copies the player with a reinforcement-learning agent trained to defeat that copy, producing an opponent tailored to the individual player's behavior. The parallel is direct — copy the player, then compete — but the medium changes what copying means. In an action game, the imitated quantity is a policy over a small action space, and difficulty is a scalar the designer can turn; in a conversation, the action space is unbounded, and there is no scalar to turn. Souza et al. (2024), in an early study of LLMs for DDA, let an LLM tune the numeric parameters of a shooter; adapting a conversational opponent remains open.

LLM game research has meanwhile concentrated on social-deduction settings such as Werewolf (Xu et al., 2023) and Avalon (Light et al., 2023), and on benchmarking social competence (Zhou et al., 2024). These evaluate how well agents play; they do not treat agent capability as a dial for the human player's experience.

2.3 Positioning of this Work

This paper is an extension of our earlier reports of this method (Tsunoda et al., 2026a, 2026b). The conference version (Tsunoda et al., 2026a) introduced the imitating rival, the persona token and the turn procedure, and reported the same user study at the session level: per-participant win rates and talk scores, enjoyment and difficulty ratings compared with pairwise tests, and the

observation that the imitating rival lowered the win rates of high-scoring players and raised those of low-scoring ones. The present paper adds an explicit formulation of difficulty as a parity condition (Section 3), a specification of the mechanism at the level of prompts, data schemas and turn scheduling (Section 5), turn-level analyses of the interaction logs that measure imitation directly and characterize the difficulty it produces (Sections 6.3 and 6.4), further questionnaire items with omnibus tests (Section 6.2), and correlation and within-participant regression analyses of the skill-group contrast reported in the conference version (Section 6.5). It also corrects the description of the comparison condition, which the conference version called a rival without player imitation: that rival shares the player's knowledge, and we call it the knowledge-matched rival (Section 6.1). Finally, and unlike the wider literature, we do not create a capability gap: earlier work, our own included, makes the opponent weaker or stronger on some measured axis, whereas we make the opponent *equal* and let the player's own capability set the level.

3. DEFINITIONS AND PROBLEM FORMULATION

We define a *conversational game* as a system in which natural-language dialogue between a player and an agent is the primary input, and the game state advances over a state transition diagram according to the semantic content of that dialogue. The diagram represents the evolving relationship between player and agent; reaching a designated node marks the goal, although what counts as the goal, and whether play ends there, depends on the game. This gives free-form conversation a clear notion of winning, losing, and progress.

Difficulty has no comparable handle. In an action game, it is quantitative by construction, but if a conversational agent's knowledge overwhelms the player, the game is unplayable; if it is too thin, the player is bored. We proceed in two steps. Operationally, difficulty is defined through the player's *win rate*: a high win rate means low difficulty. Structurally, the *target* of control is a *parity condition*. Writing K_p and C_p for the player's knowledge and conversational competence and K_r and C_r for the rival's, we take the ideal state to be

$$K_r \approx K_p , \quad C_r \approx C_p \quad (1)$$

and difficulty is the signed departure from (1): a rival above the player on either axis makes the game harder; a rival below it easier. This reformulation makes the problem tractable. A designer cannot specify an appropriate win rate for a player they have never met, whereas a system can continuously estimate what the player knows and how the player speaks, and hold the rival at parity.

4. PROPOSED METHOD

4.1 Overview

The system comprises two LLM agents. The *dialogue agent* plays the character the player talks to and also acts as judge, comparing the player's utterance with the rival's each turn and deciding which better suits the context. The *rival agent* never interacts with the player directly; it produces a competing utterance that the dialogue agent evaluates against the player's, and this

competition — not explicit obstruction — generates difficulty. Figure 1 shows how the two agents, the player and the five stores of game state, are wired together.

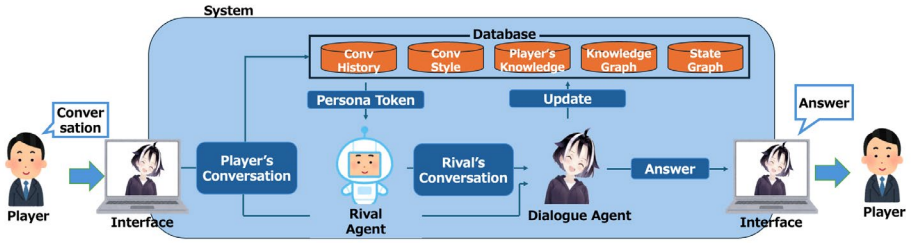

Figure 1. Overview of the difficulty adjustment system

4.2 The Persona Token

The persona token is the total input determining the rival's personality and capability:

$$P_t = \{H_t, G_{rival,t}, S_{player,t}\} \quad (2)$$

where H_t is everything the player has said and heard, $G_{rival,t}$ constrains what the rival may know, and $S_{player,t}$ carries how the player speaks.

4.2.1 Imitation of Thought

Asking an LLM in a single prompt to act like the player tended, in our development, to produce caricature; we use a three-stage pipeline instead. In *persona extraction*, an extractor LLM receives the player's recent utterances and summarizes the player's personality, current thought process and emotional state. The summary fills the player-profile section of the rival's prompt, which instructs the rival to imitate the described personality, thought process and emotional state as well as the player's verbosity and conversational skill, so the rival's instructions are rewritten from evidence each turn rather than fixed in advance. In *content generation*, the rival drafts $u'_{r,t}$ — what this player would plausibly say next, given what this player plausibly knows. In *style transfer*, that draft is first rewritten in polite Japanese (teineigo), which serves as the neutral register, and then in the player's voice. We did not evaluate the single-prompt and three-stage variants against each other, so this remains a design choice rather than a measured improvement.

4.2.2 Imitation of Conversational Style

We adapt the CST method of Roy et al. (2023), re-estimating the transfer target in real time rather than fixing it. Adapting CST's style reduction step to single utterances, we convert each player utterance u_p into a neutral sentence u_n, using polite Japanese as the neutral register, and append the pair to the dataset, which therefore grows by one demonstration per turn and tracks drift in the player's register:

$$S_{player,t+1} = S_{player,t} \cup \{(u_{n,t}, u_{p,t})\} \quad (3)$$

When the rival has a draft, CST's dynamic prompt selection retrieves the semantically nearest pairs from $S_{player,t}$ as in-context demonstrations, so the rival's final utterance is

$$u_{r,t} = CST(u'_{r,t}, S_{player,t}) , \quad u'_{r,t} = LLM(P_t) \quad (4)$$

Retrieval is what distinguishes this from generic style prompting: the model is shown the player's own sentences on semantically similar matters, not a description of how the player writes. Because the dataset does not yet hold enough of the player's utterances at the start of a session, the rival's first two utterances in each session are used as drafted, without style transfer.

4.2.3 Imitation of the Player's Knowledge

The dialogue agent starts from an initial knowledge graph G_0 describing the character and the world (see Pan et al., 2024, for a roadmap of combining LLMs with knowledge graphs). When a conversation establishes something not in the graph — a new fact, preference, or relation — it is added as a node or edge:

$$G_{t+1} = G_t \cup Extract(u_{p,t}, u_{a,t}) = G_t \cup \{n_{new}, e_{new}\} \quad (5)$$

The system also generates new elements of its own accord, subject to consistency with the existing setting, so the world grows richer as it is explored rather than merely being uncovered. The graph thus represents everything the player could have come to know.

The player's access level $V_{p,t}$ is the region of G_t the player has demonstrably reached, identified by analyzing H_t. The rival's access level is set as follows:

$$V_{r,t} = V_{p,t} + \varepsilon \quad (6)$$

$G_{rival,t}$ is the subgraph within that range. The perturbation ε keeps the game unpredictable: the rival knows almost exactly what the player knows, sometimes a little more and sometimes a little less. Without ε, the rival would be a mirror of the player; with too much ε, the parity condition (1) would fail, and we would be back to a fixed handicap. We did not vary ε systematically, so its effect on the experience is untested. In our implementation, ε is a signed random offset in the number of knowledge elements disclosed, bounded in magnitude by roughly one tenth of the player's own count.

4.3 Turn Procedure

A turn involves three parties and proceeds in eight steps.

- **Step 1, input:** the player sends $u_{p,t}$ to the dialogue agent.
- **Step 2, rival generation:** the rival drafts from the persona token and applies style transfer, yielding $u_{r,t}$ as in (4).
- **Step 3, evaluation:** the dialogue agent retrieves the context it needs — current state, relevant knowledge, history — and compares $u_{p,t}$ with $u_{r,t}$.
- **Step 4, response:** it generates its reply $u_{a,t}$ in the light of that judgment; the player's and the agent's utterances are appended to the history.
- **Step 5, knowledge and state update:** new knowledge is written per (5) and the relationship node S_t advances according to the transition condition c_t that the dialogue agent selects on the basis of its judgment (Section 5.1),
    - $S_{t+1} = Transition(S_t, c_t)$ (7)
- **Step 6, style update:** the player's latest utterance is reduced to neutral (polite) form and the pair appended per (3).

- **Step 7, rival knowledge update:** the elements the player has just learned are identified, $V_{p,t}$ is revised, $V_{r,t}$ recomputed per (6), and $G_{rival,t+1}$ extracted.
- **Step 8, persona token update:** assemble P_{t+1} from Steps 5 to 7 for the next turn.

Two properties of this schedule matter later. The rival's utterance for turn t + 1 is generated from the state at the end of turn t, so the rival is always one step behind the player's newest revelation — a deliberate lag that also makes pre-generation possible (Section 5.4). And nothing in the loop stores a difficulty value; the parity condition is re-derived each turn.

5. APPLICATION: A DATING-SIMULATION GAME

5.1 Game Design and Relationship States

The player converses with a heroine named Shachiko, a graduate student in a research laboratory, with the goal of deepening the relationship. She is the dialogue agent introduced in Section 4; hereafter, we call her the heroine. She is rendered as an animated character with synthesized Japanese speech, and the interface is an ordinary chat window with free-text entry and no menu of options (Figure 2).



Figure 2. The game screen: animated heroine, chat history, current relationship state and free-text input (the interface is designed in Japanese)

Each turn, the player's and the rival's messages arrive at the heroine simultaneously. She judges which better suits the context on a five-point scale and, based on that judgment, selects one of five transition conditions: very positive, positive, neutral, negative, very negative. The player therefore experiences the rival only indirectly, through the heroine's responsiveness, and is never told a rival exists or shown its utterances. The indirection is intended to keep the fiction intact, but it also means difficulty can be inferred only from the heroine's warmth and the relationship label shown on screen — which are precisely the signals the method manipulates.

The state transition diagram is a single ordered chain of twelve nodes from strong dislike to romance: Hate3, Hate2, Hate1, Stranger, Aware, Acquaintance, Friend1, Friend2, GoodFriend, Trust, Crush, Love. Play begins at Stranger, and the goal is Love. Positive and negative conditions move the relationship one step; very positive and very negative move it two steps; neutral moves it none; the endpoints absorb overshoot. The chain is symmetric in its

consequences, so a player who alternates good and bad turns makes no progress; whether losing a comparison stalls the relationship or sets it back depends on the transition condition the heroine selects.

5.2 Character and Knowledge Authoring

Character knowledge is authored as two JSON files per personality. *Entities* carry an identifier, a *condition* field marking its role (heroine, player, object, attribute, or runtime-generated item), and a Japanese property string, which is what the LLM actually sees. *Relationships* carry a source entity, a list of target entities, and a property string describing the relation; each target expands into one edge at load time. The initial graph of each personality contained 68 knowledge elements: 35 entities and 33 edges.

Three personalities were authored, differing in temperament, dialect, greeting, and topical knowledge. All three share the same transition file and the same start and goal nodes, so personality can be counterbalanced across conditions without changing the game's structure. The name and gender the player supplies are also written into the graph, so the heroine can use the player's name and the rival can refer to the player's attributes.

5.3 Agent Roles and Prompt Contracts

Up to eleven distinct LLM calls implement one turn (Table 1). Every call returning structured data must emit a JSON code block against a fixed schema, and everyone has a defined fallback, because with free-form generation malformed output is a certainty rather than an exception.

Table 1. LLM calls implementing one turn, with output contracts and fallbacks

Call	Purpose	Output	Fallback
Persona extractor	Summarize the player's personality and affect	Free text	"A new player."
Rival generator	Predict the player's next message	Free text	Empty utterance
Style reduction	Rewrite an utterance in polite (neutral) form	Free text	Unchanged
Style transfer	Rewrite the draft in the player's register	Free text	Draft unchanged
Knowledge selector	Select context-relevant knowledge	JSON list	Empty list
Comparator	Score player against rival, 1–5	JSON score	Score 3 (tie)
Emotion selector	Pick a transition condition	JSON label	neutral
Reply generator	Produce the heroine's reply	JSON message	Placeholder
Knowledge generator	Invent consistent new world knowledge	JSON dictionary	No addition
Knowledge tracker	Identify the knowledge the player has acquired	JSON list	Empty list
History summarizer	Compress a long history	Free text	Unchanged

Four details of these contracts affect the experiment directly.

- First, the comparator receives the two candidates in randomized order, and the score is normalized afterward, which should prevent the judge from systematically favoring either side due to position bias. The comparator is also told that both candidates are messages the player is considering sending, so it never learns that one of them comes from a rival.
- Second, the emotion selector cannot choose freely: the comparator's score restricts the candidate set, so a decisive win cannot yield a negative transition — but the mapping is non-deterministic within those bounds, which keeps identical scores from always producing identical progress.
- Third, the knowledge selector is instructed to include two or three *irrelevant* items alongside the relevant ones; in development, replies without such distractors tended to converge on the same well-worn facts.
- Finally, the routine deciding which knowledge the player has acquired is told to ignore anything derivable from the rival's utterances, since the player never sees them; without that instruction, the player's access level would inflate from information never shown, and the parity condition (1) would be computed against a fiction.

5.4 System Architecture and Latency

The application is a three-tier system. A React and TypeScript front end renders the chat, the animated character and its lip-synchronized speech, and measures how long the player spends reading and composing. A Python back end orchestrates turns, holds session state, and owns the state and knowledge graphs. A model tier serves the LLMs and the speech synthesizer locally; the models used in our experiment are named in Section 6.1. The same interfaces support cloud back ends, which we did not use; serving every in-game model locally kept inference conditions fixed throughout the experiment.

Latency is a central engineering problem. One turn involves up to eleven kinds of model call (Table 1), and heroine-side generation alone took a median of 28.1 seconds. We therefore exploit the one-step lag of Section 4.3: as soon as a turn completes, the knowledge update and the *next* rival utterance are computed in the background while the player reads and composes. Because composition time is substantial — a median above 20 seconds — the rival's utterance was usually ready before it was needed, and its measured contribution to perceived latency was effectively zero.

6. EXPERIMENT

6.1 Design and Measures

We compared three conditions within subjects.

- **PS** — *proposed system* — is the system with no rival: the heroine judges the player's utterance alone.
- **PS-R** — *proposed system with a rival* — adds the rival agent, which predicts the player's next message from the conversation history and its knowledge subgraph, but without the

extracted player profile and without style transfer; we therefore call it the knowledge-matched rival.

- **PS-IR** — *proposed system with an imitating rival* — is the full method: the rival imitates the player's thought and register via the persona token.

Nine undergraduate and graduate students (seven male, two female) each played all three conditions for fifteen minutes per condition, with prior consent and without collection of personally identifying information. Reaching Love did not end a session; play continued until the fifteen minutes had elapsed. Order of conditions and assignment of the three personalities were randomized per participant. The personality assignment happened to form a complete within-participant rotation — every participant met each personality exactly once, although the three were not balanced evenly across conditions — and personality had no detectable main effect (Kruskal–Wallis over the rival sessions: win rate $p = .60$, enjoyment $p = .22$, difficulty $p = .19$). The rival agent and the extractor ran on gemma3:27b, and the heroine on qwen3:30b-a3b-instruct-2507-fp16, both served locally; the initial knowledge graph held 68 elements (35 entities and 33 edges) and the state diagram twelve nodes with sixty transitions.

Win rate is the proportion of turns in which the player's utterance was judged better than the rival's (a comparator score of 1 or 2 after order normalization); under PS, where no competing utterance exists, the heroine instead rates the player's utterance alone on a five-point scale (1 = very good, 5 = very bad), and win rate is the proportion of turns rated 1 or 2. *Talk score* rates each player's conversational competence from 1 to 10: we gave Gemini 3 Pro the transcripts of all sessions in a single prompt and asked it to rate each participant so that the ratings spread across the scale, which makes talk score a relative measure within this sample (the model returned half-point ratings for some participants). A post-session questionnaire used five-point items for enjoyment, perceived difficulty, perceived consistency, and perceived variety, and asked participants to rank the conditions by difficulty. The consistency item asked whether the heroine's utterances contradicted the preceding or following conversation, scored from 1 (there were contradictions) to 5 (there were none), so that higher values mean a more consistent conversation. We added log-derived measures for this paper: per-turn transition conditions and state trajectories, growth of the player's knowledge, the interval between the player starting to type and the message being sent, which we call composition time, and every player utterance paired with the rival utterance that competed against it.

6.2 Outcomes and Subjective Ratings

Table 2 shows the questionnaire means. For the ordinal questionnaire items, we report the omnibus Friedman test across the three conditions rather than uncorrected pairwise comparisons. The per-participant data appear in Tsunoda et al. (2026a), where the enjoyment and difficulty ratings of P5 under PS-R were transposed; the means here use the corrected values. The consistency and variety items and the difficulty ranking are reported here for the first time.

Table 2. Questionnaire means (5-point scale) and difficulty ranking (1 = hardest)

Item	PS-IR	PS-R	PS	Test
Enjoyment	3.6	3.3	3.7	Friedman p = .78
Perceived difficulty	3.4	3.3	1.3	Friedman p = .001
Perceived consistency	3.3	3.8	4.1	Friedman p = .44
Perceived variety	3.7	2.9	3.3	Friedman p = .65
Mean difficulty rank	1.67	1.44	2.89	Friedman p = .004

Three findings are clear. The presence of a rival raises difficulty: win rates under both rival conditions were far below those under PS (87.9%; paired t, both $p < .001$), perceived difficulty differed across the three conditions (Friedman $p = .001$), with PS rated 1.3 against 3.4 and 3.3, and eight of nine participants ranked PS the easiest of the three (Friedman $p = .004$ on the ranks). The two rival conditions are close to each other on every difficulty measure: win rate 28.2% against 23.9% (paired t, $p = .63$), perceived difficulty 3.4 against 3.3, mean rank 1.67 against 1.44. And we detected no difference across conditions in enjoyment (Friedman $p = .78$), perceived consistency ($p = .44$) or perceived variety ($p = .65$).

PS, however, is not the neutral baseline we intended: without a rival, the comparator has nothing to compare against, and the player wins almost by default — two-thirds of PS turns drew the strongest available transition condition (Table 4) and all nine participants reached the goal state. Section 7.3 draws out what this implies for evaluating conversational difficulty. Progress differs sharply too: the mean index of the final node on the twelve-node chain was 5.7 under PS-IR and 5.3 under PS-R against 11.0 — that is, Love — under PS, while knowledge accumulated at similar rates throughout (0.67 to 0.81 new entities and 0.85 to 1.10 new relations per turn). Players learned at similar rates in all conditions and converted that learning into progress at very different rates.

6.3 Does the Rival Actually Imitate the Player?

We measure imitation directly, comparing each rival utterance with the player utterance it competed against. Six measures are used: the proportion of utterances containing a question mark, mean utterance length in characters, the ratio of rival to player length, character 3-gram similarity, the proportion containing a polite-form marker, and the proportion containing an exclamation mark. Except for the similarity measure and the length ratio, we report divergence — the absolute difference between the player's value and the rival's, computed within each session and then averaged over participants (Table 3).

Table 3. Player–rival divergence by measure (lower = closer imitation, except for 3-gram similarity, where higher is closer)

Property	PS-IR	PS-R	Paired t	Wilcoxon
Question rate	0.071	0.310	p = .002	p = .008
Utterance length (characters)	12.90	11.99	p = .76	p = .73
Length ratio (rival / player)	1.61	1.76	p = .54	p = .73
Character 3-gram similarity (higher = closer)	0.178	0.143	p = .22	p = .20
Polite-form markers	0.184	0.114	p = .32	p = .38
Exclamatory markers	0.133	0.136	p = .96	p = .91

The pattern is sharp. On the *pragmatic* property — whether an utterance asks a question — imitation works decisively: the divergence is 0.07 under PS-IR against 0.31 under PS-R. The difference also remains significant after Bonferroni correction across the six measures of Table 3 ($\alpha = .05/6 \approx .008$; paired t, p = .002). On every *surface* property — length, register markers, lexical overlap — imitation makes no measurable difference, and the rival remains 61% longer than the player under PS-IR and 76% longer under PS-R.

We also asked whether imitation improves as evidence accumulates, as a history-driven mechanism should. When each session is split at its midpoint, question-rate divergence under PS-IR falls from 0.165 to 0.104, a change that is not significant on its own (paired t, p = .33), while under PS-R it *rises* from 0.213 to 0.443 (p = .018). The data show a significant difference between the two trajectories (Wilcoxon p = .039). The imitating rival converges on the player as the persona token fills up; the knowledge-matched rival drifts away as the conversation moves somewhere it cannot follow. Figure 3 shows both the per-measure divergences and these two trajectories. The half-session figures are on average larger than the whole-session divergence in Table 3 because a divergence computed over a whole session lets opposite-signed differences in its two halves cancel.

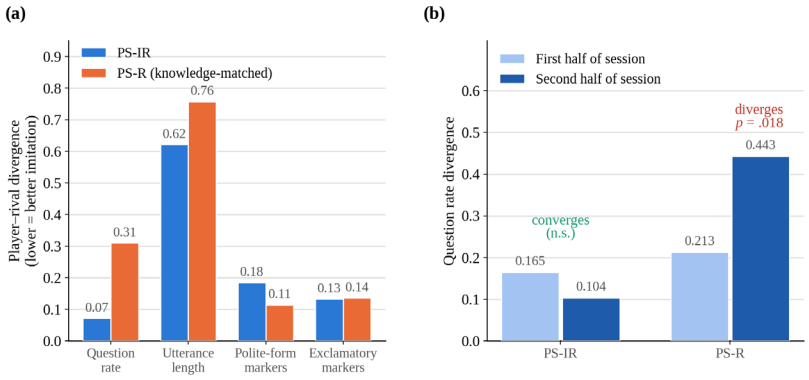

Figure 3. Imitation fidelity by measure (the length divergence is divided by the player's mean utterance length in each session), and the within-session trajectories of the question-rate divergence

One counter-current deserves recording. Under PS-IR, the mean per-turn difference in length between rival and player grows over a session, from 12.9 to 20.2 characters (paired t, $p = .0498$): the rival becomes progressively more verbose as history accumulates, even as its question asking converges on the player's. This is the clearest defect the mechanism analysis exposed, and we read it as a consequence of feeding an ever-longer history into a generator with no length constraint.

6.4 The character of difficulty, and player behavior

The two rival conditions were indistinguishable in aggregate; their composition is not (Table 4).

Table 4. Distribution of per-turn transition conditions (% of all turns) and relationship setbacks per turn (mean over participants)

Condition	v. positive	positive	neutral	negative	v. negative	Setbacks
PS-IR	2.2	24.4	71.1	2.2	0.0	0.021
PS-R	5.9	18.8	63.4	11.9	0.0	0.093
PS	66.3	22.1	10.5	1.1	0.0	0.012

The imitating rival makes the game hard by *withholding progress*: 71% of its turns are neutral and only 2.2% negative. The knowledge-matched rival makes it hard by *taking progress away*: negative transition conditions are five times as frequent and the relationship moves backward more than four times as often per turn (Wilcoxon $p = .063$). Given the symmetric chain of Section 5.1, these are qualitatively different experiences of the same nominal difficulty: a stalemate against an equal, or a series of setbacks inflicted by something the player cannot see.

The logs also record how long players took between starting to type and sending their message. Taking each participant's median and averaging across participants, that interval was 28.8 seconds under PS-IR against 20.4 under PS-R and 22.4 under PS — 41% longer against the imitating rival (Wilcoxon $p = .020$) — and players completed slightly fewer turns in the fixed fifteen minutes (10.0 against 11.2, $p = .039$). We read this as engagement rather than friction, because the effect is specific to the imitating condition rather than to the presence of a rival. Composition time is nevertheless ambiguous — deliberation and difficulty of expression are not separable within it.

6.5 Does the Difficulty Track the Individual Player?

Finally, we ask whether the difficulty the method produces tracks the individual player. Splitting participants at a talk score of 7 (high group TS ≥ 7, $n = 5$; low group $n = 4$), PS-IR win rates were 20.6% for the high-score group against 35.3% under PS-R, and 37.8% for the low-score group against 9.7% (paired t, $p = .045$ and $p = .034$; Cohen's $d_z = -1.29$ and $+1.86$, against $d_z = 0.17$ overall). A difference-in-differences comparison of the groups is significant (Welch $t = -4.69$, $p = .004$; Mann–Whitney $p = .016$).

A stronger version emerges from the correlations. Under PS-R, win rate is strongly predicted by talk score (Pearson $r = .76$, $p = .018$; Spearman $\rho = .71$, $p = .032$; slope +4.15 points of win rate per talk score point, $R^2 = .58$). Under PS-IR it is not detectable ($r = -.31$, $p = .41$; slope -1.71, $R^2 = .10$), which with nine participants is an absence of evidence rather than evidence of absence. Regressing each participant's win-rate difference between the two rival conditions

(PS-IR − PS-R) on talk score, which respects the pairing of each participant's two sessions, gives a slope of −5.86 points per talk-score point (p = .090, R^2 = .36): the difference in favor of the imitating condition shrinks as skill rises and reverses at a talk score of about 7.3. Stated this way the claim is not that the method *balances* win rates — it does not, and mean absolute deviation from a 50% win rate is statistically indistinguishable between the conditions, 23.4 against 26.1 points, p = .74 — but that it *decouples* the outcome from the player's conversational skill, which is what a difficulty control ought to do. Figure 4 plots both relationships.

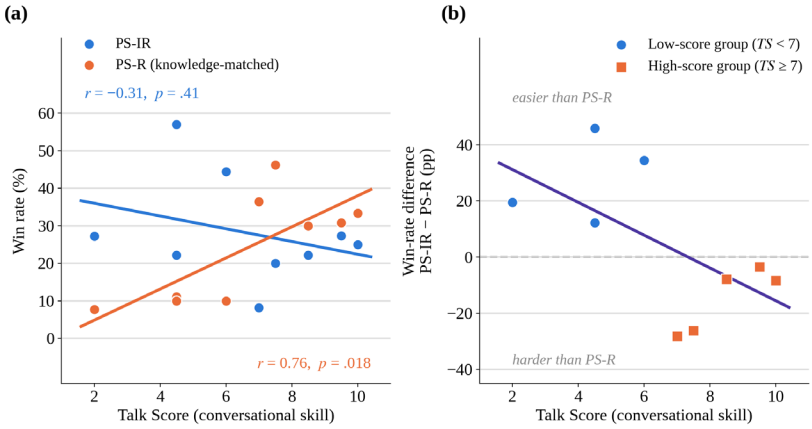

Figure 4. Win rate against talk score by condition, and the win-rate difference between conditions

7. DISCUSSION

7.1 What Imitation Actually Imitates

The mechanism analysis draws an unanticipated line between what the persona token transfers and what it does not. The rival reproduces the player's question asking — the one dialogue act we measured — with high fidelity while its surface style stays largely its own. That the pragmatic layer transfers is unsurprising in hindsight, since the extractor summarizes the player's personality, thought process and emotional state; that the stylistic layer resists transfer is more interesting, since style transfer is precisely the component included to handle it.

Three explanations are plausible, and our data cannot separate them. Early in a session, when the style dataset holds only a handful of pairs, the retrieved demonstrations may be too few to override the generator's default register. Alternatively, the three-stage pipeline may be leaking: a draft generated at length is hard to compress into a terse player's register without discarding content, and the transfer step, given the choice, preserves content. A third is simpler still: the extractor almost never describes how much the player says (2 of 90 profiles do), so the rival's instruction to imitate verbosity has little evidence to act on. If the second or third explanation holds, the fix is not more demonstrations but a length target estimated from the player's own utterances and carried into the draft stage.

7.2 Difficulty has a texture, not just a level

The result we find most interesting is one a win rate cannot express. Two systems that no aggregate measure could tell apart were hard in different ways.

The methodological implication is that DDA research in conversational settings should not evaluate difficulty by outcome alone: two systems can be indistinguishable on outcome and opposite in the composition of that outcome, and here the outcome measures failed to discriminate.

7.3 Evaluating Conversational Difficulty and Enjoyment

Our no-rival condition was uninformative in a way worth reporting. Removing the rival did not produce a game without competitive pressure; it produced a game in which the judge had no second candidate and so approved almost everything. In a comparison-based design such as ours, the comparison is the mechanic, and ablating it did not yield a baseline on the same scale. A useful control would supply a second candidate that is neither an imitation of the player nor adapted to the player at all: a fixed corpus utterance, or one drawn from another player's session.

We also found no difference in enjoyment across the three conditions. We read this not as evidence that difficulty control is irrelevant to enjoyment but as a limitation of a single five-point item administered after fifteen minutes of play. A validated multidimensional instrument (Vanden Abeele et al., 2020) applied over longer play would be a more promising way to make the question answerable.

8. LIMITATIONS AND FUTURE WORK

8.1 Sample Size

The sample of nine participants is the binding constraint on everything reported here. It limits power and makes the group comparisons in Section 6.5 rest on four and five observations. The skill-adaptation result of Section 6.5 in particular should be read as indicative until it is replicated with more participants.

8.2 Construct Validity and Implementation

The evaluation covers one genre, one heroine and one language. Whether a player-imitating rival is a useful difficulty control in a negotiation game, a mystery, or a cooperative setting is open, and the answer may depend on whether the game's judging mechanism resembles the comparison used here.

We define difficulty as the win rate over judgments produced by an LLM. That judge has its own biases: we randomize candidate order to counter position bias, but we have not validated its judgments against human raters, and a systematic preference for longer utterances would propagate directly into our difficulty measure. The rival's persistent verbosity makes this a live concern, not a hypothetical one. The same caution applies to the skill measure: talk score is

produced by a second LLM from the transcript and has not been checked against human ratings, yet the skill-adaptation result of Section 6.5 is stated entirely in its terms.

8.3 Future Work

The mechanism analysis suggests concrete work. Carrying an explicit length target from persona extraction into draft generation should address the verbosity drift. Extending the imitation measures beyond question rate to a fuller inventory of dialogue acts would test whether the pragmatic transfer generalizes. And instrumenting the game so that the parity condition (1) can be evaluated directly, rather than inferred from win rates, would close the control loop on the quantity the method targets.

9. CONCLUSION

This paper developed and evaluated a difficulty-control mechanism for LLM-based conversational games in which the player's opponent is a rival agent built to imitate the player. We documented a complete implementation as a Japanese dating simulation and evaluated it against a knowledge-matched rival and against a no-rival condition that turned out to be degenerate rather than neutral.

Turn-level analysis shows the imitation to be real and partial: the rival's rate of question asking stays on average within 0.07 of the player's own, where the knowledge-matched rival differs by 0.31, and over a session the imitating rival converges on the player while the knowledge-matched one diverges, whereas surface style transfers hardly at all. The imitating rival also appears to change the character of difficulty rather than only its level, withholding progress rather than taking it away — a difference at the margin of significance — and players spent 41% longer composing each message against it. Difficulty also tracks the individual player: with the knowledge-matched rival, win rate is strongly predicted by conversational skill, whereas with an imitating rival that association is no longer detectable — a decoupling that a study of this size can indicate but not yet establish (Section 8.1).

The results support a shift in how difficulty is specified for conversational games. Rather than tuning an opponent's strength against an imagined player, a designer can define the opponent relative to the actual one, and let the game's difficulty become a property of the conversation taking place.

ACKNOWLEDGEMENT

This work was partly supported by JSPS KAKENHI (No. 23K28094 and No. 26K23812).

REFERENCES

Chen, J., 2007. Flow in games (and everything else). *Communications of the ACM*, Vol. 50, No. 4, pp. 31-34.

Chen, J. et al., 2024. From persona to personalization: A survey on role-playing language agents. *Transactions on Machine Learning Research*.

Csikszentmihalyi, M., 2008. *Flow: The psychology of optimal experience*. Harper Perennial Modern Classics, New York, USA.

Fuchs, R. et al., 2024. Personalized dynamic difficulty adjustment: Imitation learning meets reinforcement learning. *Proceedings of the IEEE Conference on Games*. Milan, Italy, pp. 1-2.

Gallotta, R. et al., 2024. Large language models and games: A survey and roadmap. *IEEE Transactions on Games*, pp. 1-18.

Gao, Q. C. and Emami, A., 2023. The Turing quest: Can transformers make good NPCs? *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 4: Student Research Workshop)*. Toronto, Canada, pp. 93-103.

Hong, S. et al., 2024. MetaGPT: Meta programming for a multi-agent collaborative framework. *Proceedings of the International Conference on Learning Representations*. Vienna, Austria.

Hunicke, R., 2005. The case for dynamic difficulty adjustment in games. *Proceedings of the ACM SIGCHI International Conference on Advances in Computer Entertainment Technology*. Valencia, Spain, pp. 429-433.

Hunicke, R. and Chapman, V., 2004. AI for dynamic difficulty adjustment in games. *Challenges in Game Artificial Intelligence: AAAI Workshop*. San Jose, USA, pp. 91-96.

Kirk, H. R. et al., 2024. The benefits, risks and bounds of personalizing the alignment of large language models to individuals. *Nature Machine Intelligence*, Vol. 6, No. 4, pp. 383-392.

Li, C. et al., 2023. ChatHaruhi: Reviving anime character in reality via large language model. *arXiv preprint* arXiv:2308.09597.

Light, J. et al., 2023. AvalonBench: Evaluating LLMs playing the game of Avalon. *arXiv preprint* arXiv:2310.05036.

Liu, J. et al., 2025. LLMs + persona-plug = personalized LLMs. *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics*, Vol. 1. Vienna, Austria, pp. 9373-9385.

Pan, S. et al., 2024. Unifying large language models and knowledge graphs: A roadmap. *IEEE Transactions on Knowledge and Data Engineering*, Vol. 36, No. 7, pp. 3580-3599.

Park, J. S. et al., 2023. Generative agents: Interactive simulacra of human behavior. *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology*. San Francisco, USA, pp. 1-22.

Roy, S. et al., 2023. Conversation style transfer using few-shot learning. *Proceedings of the 13th International Joint Conference on Natural Language Processing and the 3rd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics*, Vol. 1. Nusa Dua, Bali, Indonesia, pp. 119-143.

Salemi, A. et al., 2024. LaMP: When large language models meet personalization. *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics*, Vol. 1. Bangkok, Thailand, pp. 7370-7392.

Shanahan, M. et al., 2023. Role play with large language models. *Nature*, Vol. 623, No. 7987, pp. 493-498.

Shao, Y. et al., 2023. Character-LLM: A trainable agent for role-playing. *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*. Singapore, pp. 13153-13187.

Souza, C. H. R. et al., 2024. Large language models and dynamic difficulty adjustment: An integration perspective. *Anais Estendidos do XXIII Simpósio Brasileiro de Jogos e Entretenimento Digital*. Manaus, Brazil, pp. 31-36.

Sweetser, P. and Wyeth, P., 2005. GameFlow: A model for evaluating player enjoyment in games. *ACM Computers in Entertainment*, Vol. 3, No. 3, Article 3A.

Tseng, Y.-M. et al., 2024. Two tales of persona in LLMs: A survey of role-playing and personalization. *Findings of the Association for Computational Linguistics: EMNLP 2024*. Miami, USA, pp. 16612-16631.

Tsunoda, Y. et al., 2025. A study on conversational difficulty adjustment in dating simulation games. *Proceedings of the 17th Forum on Data Engineering and Information Management (DEIM 2025)*. Japan (in Japanese).

Tsunoda, Y. et al., 2026a. Player imitation for difficulty control in LLM-agent-based conversational games. *Proceedings of the International Conference on Game and Entertainment Technologies (GET 2026)*.

Tsunoda, Y. et al., 2026b. Generation of persona tokens and difficulty control of conversational games using them. *Proceedings of the 18th Forum on Data Engineering and Information Management (DEIM 2026)*. Japan (in Japanese).

Vanden Abeele, V. et al., 2020. Development and validation of the player experience inventory: A scale to measure player experiences at the level of functional and psychosocial consequences. *International Journal of Human-Computer Studies*, Vol. 135, Article 102370.

van Stegeren, J. and Myśliwiec, J., 2021. Fine-tuning GPT-2 on annotated RPG quests for NPC dialogue generation. *Proceedings of the 16th International Conference on the Foundations of Digital Games*. Montreal, Canada, pp. 1-8.

Värtinen, S. et al., 2024. Generating role-playing game quests with GPT language models. *IEEE Transactions on Games*, Vol. 16, No. 1, pp. 127-139.

Vicencio-Moreira, R. et al., 2015. Now you can compete with anyone: Balancing players of different skill levels in a first-person shooter game. *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*. Seoul, Korea, pp. 2255-2264.

Wang, G. et al., 2024. Voyager: An open-ended embodied agent with large language models. *Transactions on Machine Learning Research*.

Wang, N. et al., 2024. RoleLLM: Benchmarking, eliciting, and enhancing role-playing abilities of large language models. *Findings of the Association for Computational Linguistics: ACL 2024*. Bangkok, Thailand, pp. 14743-14777.

Xu, Y. et al., 2023. Exploring large language models for communication games: An empirical study on Werewolf. *arXiv preprint* arXiv:2309.04658.

Yannakakis, G. N. and Togelius, J., 2011. Experience-driven procedural content generation. *IEEE Transactions on Affective Computing*, Vol. 2, No. 3, pp. 147-161.

Zhang, S. et al., 2018. Personalizing dialogue agents: I have a dog, do you have pets too? *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, Vol. 1. Melbourne, Australia, pp. 2204-2213.

Zhou, X. et al., 2024. SOTOPIA: Interactive evaluation for social intelligence in language agents. *Proceedings of the International Conference on Learning Representations*. Vienna, Austria.

Zohaib, M., 2018. Dynamic difficulty adjustment (DDA) in computer games: A review. *Advances in Human-Computer Interaction*, Vol. 2018, Article ID 5681652, pp. 1-12.