

ペルソナトークンの生成とこれを用いた対話型ゲームの難易度制御

角田 悠之介[†] 中園 康聖[†] 馬 強[†]

[†] 京都工芸繊維大学 〒606-0951 京都府京都市左京区松ヶ崎橋上町
E-mail: †{m5622034,m4622034}@edu.kit.ac.jp, ††qiang@kit.ac.jp

あらまし LLM エージェントを利用した対話型ゲームが多数存在する。これらのゲームの面白さを向上させるには、プレイヤーのレベルに応じて LLM エージェントの能力を制御しゲームの難易度を動的に調整することが有効である。そこで本研究ではプレイヤーを模倣するライバルエージェントを導入して、対話型ゲームの難易度を調整する手法を提案する。プレイヤーの行動を阻害するライバルエージェントは、会話履歴から生成される (a) ペルソナトークンの生成による人格模倣及び (b) 動的に更新される知識グラフによる知識レベルの模倣を行うことでプレイヤーと同等の能力を獲得し、適切な競合相手としての振る舞いを実現する。これにより、プレイヤーに合わせた柔軟な難易度調整を行う。

キーワード LLM, 対話型ゲーム, 対話エージェント, ライバルエージェント, ペルソナトークン, 知識グラフ

1 はじめに

近年、人工知能技術の進化に伴い、自然言語処理技術の向上が著しく進展しており、その中でも大規模言語モデル (Large Language Model, LLM) の発展がゲーム業界に与える影響は計り知れないものとなっている。特に、様々なゲームにおいて LLM が導入されるケースが増加しており、その活用範囲は広がり続けている [1] [2] [3] [4] [5]., 対話を主体とするゲームジャンルでは、LLM を適用させキャラクターを演じさせることで、これまでにない自然なコミュニケーションが可能となる [6] [7] [8] [9].

従来のゲームにおける会話システムは、主にスクリプト型の手法が主流であり、事前に用意された選択肢や固定的な応答を提示する形が一般的であった。このような従来の会話システムでは、あらかじめ開発者が設定した分岐シナリオに基づいて対話が進行するため、プレイヤーの発言の自由度は限定的であり、ゲームのリプレイ性にも一定の制約があった。しかし、LLM を導入することにより、ゲームキャラクターはプレイヤーの自由な発言や選択に応じて適切な応答を即座に生成できるようになり、より動的かつ多様なインタラクションが実現されている [10] [11]. この技術革新により、プレイヤーはゲーム内でよりリアルな会話を体験し、自らの選択や行動が物語の進行やキャラクターの反応に直接影響を及ぼすようになった結果、より強い没入感を得ることが可能となった。

一方で、このような自由度の高さは、難易度の調整を困難にしている。プレイヤーのスキルや知識量は千差万別であり、すべてのプレイヤーに対して適切な「挑戦」と「達成感」を提供するためには、動的な難易度調整が不可欠である。我々は、プレイヤーの進行を妨害するライバルエージェントを導入し [12], エージェントの知識量に差をつけることで難易度調整を試みた。しかし、単なる知識量の制限だけでは LLM 本来の会話能力 (推論能力や言語流暢性) の影響を排除できず、適切な難易

度制御には至らなかった。そこで本研究では、対話型ゲームにおける新たな難易度調整手法として、ライバルエージェントにプレイヤーを模倣する機構を提案する。このライバルエージェントは、プレイヤー自身の会話スタイルや知識を模倣することで、プレイヤーにとって「ちょうど良い」競合相手となり、ゲームの動的な難易度制御を実現する。

本研究で提案する手法の特徴は、ゲームの進行管理と知識制御のために「状態遷移図」と「知識グラフ」を導入し、これらに基づいてライバルエージェントの行動生成に必要な「ペルソナトークン」を生成する点にある。本システムにおいて、状態遷移図は対話の結果によるプレイヤーとエージェントの関係性の変化や、会話のフェーズ進行を管理するために用いられる。これにより、自由な対話の中にもゲームとしての明確な「勝敗」や「進行度」を定義することが可能となる。また、知識グラフは対話エージェントの情報およびプレイヤーが対話を通じて獲得した知識を構造化して保持する役割を担う。ここでいうペルソナトークンとは、プレイヤーの全会話履歴、対話エージェントに関する知識 (ライバルエージェントの知識グラフへのアクセス権)、およびプレイヤー固有の会話スタイルを統合した情報である。具体的には、図 1 の手順で難易度調整を行う。プレイヤーはメッセージを作成し、対話エージェントと会話を行う。対話エージェントはプレイヤーとの会話に必要な情報としてデータベースから対話に必要な情報を抽出する。この情報の中に事前に定義された知識グラフがあるが、プレイヤーとの対話を通じて新たな事実や設定が生まれた場合、対話エージェントがデータベースの知識グラフを更新する形でその情報を追加する。これにより、データベースの知識グラフは「プレイヤーが現在知っていること」を詳細に反映した状態となる。また、ライバルエージェントはペルソナトークンを用いてプレイヤーを模したメッセージを作成する。この際、ペルソナトークン内の対話エージェントに関する知識を、プレイヤーの知識レベルを基準としてわずかに増減させる形で設定する。これにより、ライバルは常にプレイヤーと拮抗した知識量を持ちつつ、プレ

イヤーの口調や思考パターンを模倣した発言を行うことが可能となる。

本研究の貢献は以下のとおりである。

- 状態遷移図および知識グラフを用いて、プレイヤーに合わせた難易度の動的な変化を実現する。
- ペルソナトークンを用いて、プレイヤーの振る舞いを模倣するシステムを構築する。

2 関連研究

LLM にペルソナを割り当てシミュレートするエージェント (Role-Playing Language Agents: RPLA) の研究は幅広く行われている。[13] 職業、性格などの共通の特徴を共有するグループに焦点を当てた人口統計学的ペルソナ、有名人などの広く認知されたキャラクターのペルソナを模倣するキャラクターペルソナ、個々のユーザーデータに基づいて LLM を構築する個別化ペルソナが存在する。またゲームの難易度を AI を用いて動的に調整する手法である Dynamic Difficulty Adjustment (DDA) も多く研究されている。

2.1 RPLA

2.1.1 人口統計学的ペルソナ (Demographic Persona)

Sirui ら [14] は各エージェントに対し、詳細な人口統計学的属性 (年齢・性別等) ではなく、名前・職種・目標・制約からなる「職務上のプロファイル (Role)」を定義し、実社会の組織構造に基づいた専門的な役割分担を行なった。

2.1.2 キャラクターペルソナ (Character Persona)

Yufan Shao ら [7] は定の人物の「経験」や「記憶」を詳細に再構築して学習させ、さらに時代背景に矛盾する知識をあえて忘却 (Protective Experience) させることで、単なる口調模倣を超えた高忠実度なペルソナを実現する「Character-LLM」手法を提案した。

2.1.3 個別化ペルソナ (Individualized Persona)

Liu ら (2025) [15] は、軽量な「プラグイン・ユーザー・エンベッダー」モジュールを用いて、各ユーザーの過去の履歴からユーザー固有の埋め込み (ペルソナ埋め込み) を生成する PPlug を提案した。この手法は、LLM のパラメータを調整することなく、埋め込みをタスク入力に付加するだけで、ユーザーの習慣や好みを捉えたパーソナライズされた出力を可能にする。Roy ら (2023) [16] は、少数の例示対話のみでターゲットのスタイルに変換する「Few-Shot 会話スタイル変換」を提案した。彼らは、スタイルを含む対話を一度「スタイル・フリー」な形式に還元し、それをターゲットスタイルで書き直す 2 段階のイン・コンテキスト学習を採用している。この手法により、会話の文脈 (マルチターン) を維持したまま、自然なスタイル変換が可能となる。

2.2 DDA

チクセントミハイのフロー理論 [17] によれば、個人のスキルと課題の難易度が釣り合った状態こそ没入感や楽しさを生む条

件であり、これはゲームにも適応可能である。[18] Ronja Fuchs ら [19] は、プレイヤーの行動をコピーする「模倣学習エージェント」とそれを倒すために訓練された「強化学習エージェント」を組み合わせることで、プレイヤー個人のスキルに適応した対戦相手を生成するパーソナライズ動的難易度調整 (PDDA) フレームワークを提案した。Robin Hunicke ら [20] は、在庫理論と確率モデルを用いてプレイヤーのリソース (ヘルスや弾薬) が枯渇する「苦戦状態」を予測し、ゲーム内の物資の供給と需要を動的に制御することで難易度を調整するシステム「Hamlet」を開発した。

2.3 本研究について

既存研究 [12] ではライバルエージェントとプレイヤーの知識に差をつけることで難易度調整を行っていた。しかしこの研究では LLM の会話能力が難易度調整に大きく影響を及ぼしていた。本研究では知識に差をつけず、ライバルエージェントにプレイヤーを模倣させることによって難易度調整を行う。また個別化ペルソナにおける LLM のパーソナライズ技術と会話スタイル変換の研究を基盤とし、これらをゲーム内エージェントの模倣に応用する。

3 難易度の定義と用語説明

本節では、対話型ゲームと難易度の定義および本論文で用いる用語の説明を行う。

3.1 対話型ゲームと難易度の定義

本研究において「対話型ゲーム」とは、プレイヤーとエージェント間の自然言語による対話を主要な入力とし、その意味内容に基づいて状態遷移図上でゲームの状態 S が遷移するシステムと定義する。この状態遷移図は、対話を通じて変化するプレイヤーとエージェントの関係性やゲームの進行状況を表現しており、特定のノードへの到達がゲームの終了を意味する。

従来のアクションゲーム等における難易度は、敵の HP や攻撃頻度といった数値パラメータによって定量的に定義可能であった。しかし、対話型ゲームにおいては会話の難易度を定量化することが困難である。エージェントの知識量がプレイヤーを圧倒すればゲームは成立せず、逆に貧弱すぎればプレイヤーは退屈を感じる。そこで本研究では、エージェントが保持する情報を知識グラフとして構造化し、その参照可能な範囲 (アクセスレベル) によって知識量を定義する。その上で、対話型ゲームにおける難易度 D を、プレイヤーの入力 u_p と競合するライバルエージェントの入力の「質的・知識の均衡度」として再定義する。理想的なゲームバランスとは、知識グラフに基づくプレイヤーの知識 K_p と会話能力 C_p に対し、ライバルが同等の能力 ($K_r \approx K_p, C_r \approx C_p$) を持って競い合う状態であるとする。この均衡を保つために、システムはプレイヤーの履歴から動的にペルソナを生成し、ライバルの振る舞いを制御する必要がある。

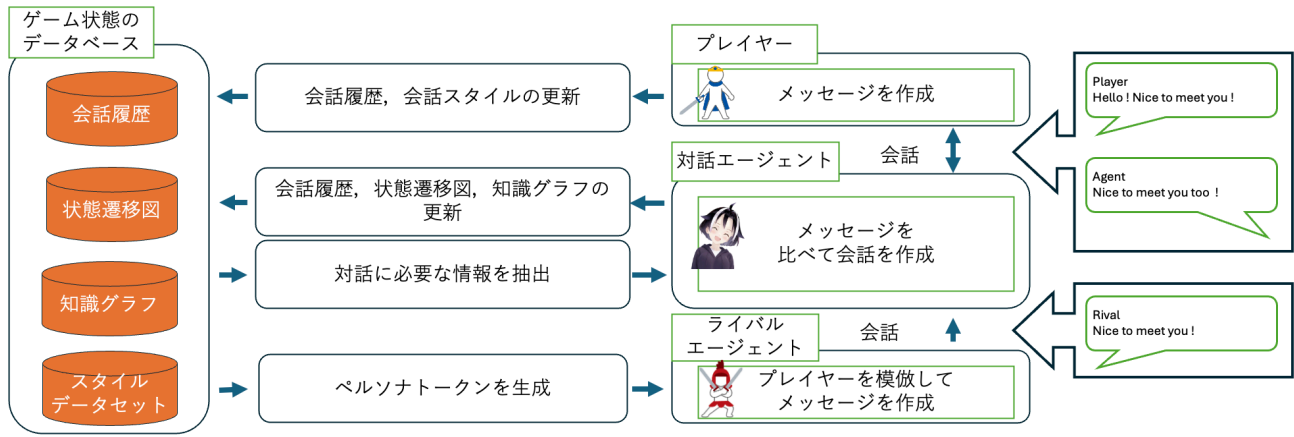


図 1 提案手法の概略図

3.2 用語説明

本論文で用いる用語と記号を定義し、説明を行う。

対話エージェント プレイヤーと対話を行う LLM のエージェント。

ライバルエージェント プレイヤーのゲーム進行の妨害を行う LLM エージェント。

状態遷移図 プレイヤーと対話エージェントとの関係性をあらわした有向グラフ。ノードは対話エージェントとプレイヤーとの関係性を表している。エッジは次のターンの遷移方向を表している。

ターン プレイヤーの状態遷移図における遷移回数。

最大ターン数 ゲームが強制終了するまでのターン数。

知識グラフ 対話エージェントについての情報を保持した無向グラフ。データノード N_{data} 、関連ノード N_{rel} 、エッジ E で構成され、 $G = (N_{data}, N_{rel}, E)$ と表される。エッジはノード同士の関係性を表し、アクセス可能な経路を形成する。初期状態 G_0 に対し、対話を通じて新たなノードやエッジが動的に追加される。

アクセスレベル 知識グラフ G 全体のうち、エージェントが参照可能な情報の割合または範囲。プレイヤーのアクセスレベルを V_p 、ライバルのアクセスレベルを V_r とする。

プレイヤーアクセスレベル ターン T におけるプレイヤーのアクセスレベル。

ライバルアクセスレベル ターン T におけるライバルエージェントのアクセスレベル。

ペルソナトークン ライバルエージェントの人格と能力を決定づける入力情報の総体。本研究では、プレイヤーの全会話履歴 H 、ライバルがアクセス可能な知識グラフの部分集合 G_{rival} 、およびプレイヤーの会話スタイル S_{player} の組み合わせとして定義する。

用語と記号の対応関係を表 1 に示す。

表 1 用語と記号

用語	記号
ターン	T
最大ターン数	T_{max}
知識グラフ	G
知識グラフのデータノード	N_{data}
知識グラフの関連ノード	N_{rel}
知識グラフのエッジ	E
プレイヤーアクセスレベル	V_p, T
ライバルアクセスレベル	V_r, T
ペルソナトークン	P
プレイヤーの全会話履歴	H
ライバルがアクセス可能な知識グラフの部分集合	G_{rival}
プレイヤーの会話スタイル	S_{player}

4 提案手法

4.1 モデル概要

本手法の概略図を図 1 に示す。本研究で提案する対話型ゲームの難易度調整システムは、プレイヤーと対話を行う「対話エージェント」と、プレイヤーを模倣して競合する「ライバルエージェント」の 2 種類の LLM エージェントによって構成される。システムは主に以下のプロセスで動作する。

- 知識の抽出と更新:** プレイヤーと対話エージェントの会話から、プレイヤーが獲得した新しい知識や設定を抽出し、知識グラフを動的に更新する。
- ペルソナトークンの生成:** プレイヤーの「全会話履歴」、更新された「知識グラフ」、および抽出された「会話スタイル」を統合し、ライバルエージェントの人格を決定づける「ペルソナトークン」を生成する。
- ライバルの行動生成:** ペルソナトークンを入力されたライバルエージェントは、プレイヤーと同等の知識とスタイルで発話を行い、ゲームの難易度を維持する。

このアーキテクチャにより、事前に固定された難易度パラメー

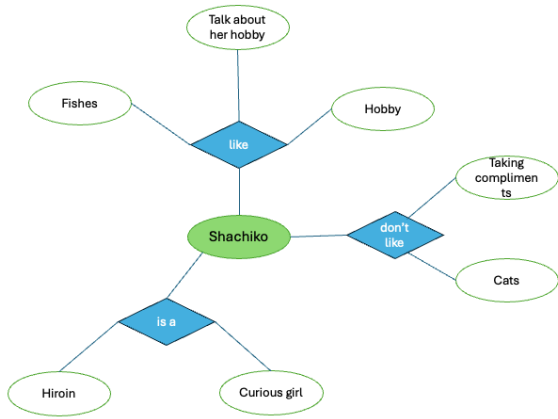


図 2 知識グラフ例

たではなく、プレイヤーの実際の行動履歴に基づいた動的な難易度制御が可能となる。

4.2 会話手順とゲーム進行

本システムにおけるゲームは、プレイヤー、対話エージェント、およびライバルエージェントの三者によって行われる。1ターンにおける会話手順と処理フローは以下の通りである。

Step 1 **プレイヤー入力 (Input):** ターン t において、プレイヤーは対話エージェントに対するメッセージ $u_{p,t}$ を入力する。

Step 2 **ライバルエージェントの生成 (Rival Generation):** ライバルエージェントは、ペルソナトークン P を用いて、プレイヤーの発話に対抗または割り込む形の発話 $u_{r,t}$ を生成する。

$$u_{r,t} = \text{LLM}(P) \quad (1)$$

Step 3 **判定と状態遷移 (Evaluation & Transition):** 対話エージェントはデータベースから対話に必要な情報 (状態遷移図, 知識グラフ, 会話履歴) を受け取り, $u_{p,t}$ と $u_{r,t}$ を比較し, どちらの発話が文脈に適しているか, あるいは好ましいかを判定する。

Step 4 **応答生成 (Response):** 対話エージェントは判定結果に基づき, 次の展開となる応答 $u_{a,t}$ を生成し, プレイヤーに提示する。

Step 5 **知識グラフと状態遷移図の更新 (Knowledge Update):** 対話エージェントはプレイヤーの発話 $u_{p,t}$ と応答 $u_{a,t}$ を解析し, 新しい事実や設定が含まれる場合, 知識グラフ G_t を更新する。また勝者 $w \in \{p, r\}$ に基づき, 状態遷移図 S_t を更新する。知識グラフの例を図 2, 状態遷移図の例を図 3 に示す。

$$G_{t+1} = G_t \cup \text{Extract}(u_{p,t}, u_{a,t}) \quad (2)$$

$$S_{t+1} = \text{Transition}(S_t, w) \quad (3)$$

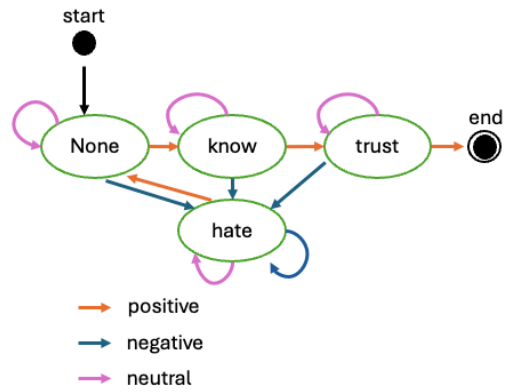


図 3 状態遷移図の例, プレイヤーが勝てば positive, ライバルエージェントが勝てば negative, どちらも同等であれば neutral に遷移する。

Step 6 **スタイルデータセットの更新 (Knowledge Update):**

プレイヤーの会話履歴からスタイルデータセットの更新を行う。

4.3 ペルソナトークン

4.3.1 ペルソナトークンの構成

本研究におけるライバルエージェントの核となる「ペルソナトークン」 P は, 既存の LLM パーソナライズ手法を対話型ゲームの敵対キャラクター生成に応用するために, 以下の 3 要素によって独自に定義した情報の総体である。

$$P = \{H, G_{rival}, S_{player}\} \quad (4)$$

ここで, H はプレイヤーが行った過去の全会話履歴, G_{rival} は知識グラフ G のうちライバルエージェントに開示された部分グラフ (知識レベルの制約), S_{player} はプレイヤーの発話から抽出・蓄積された会話スタイルデータセットである。ペルソナトークンを用いたライバルエージェントの制御は, (1) 思考の模倣と, (2) 会話スタイルの模倣の 2 段階で構成される。

4.3.2 思考の模倣

ライバルエージェントがプレイヤーと同等の文脈理解力を持つために, プレイヤーの会話履歴から抽出された特徴を用いる。具体的には, 直近の会話履歴および過去の全履歴を入力とした LLM (Extractor) に対し, 「このプレイヤーの性格, 話し方, 思考パターンを分析し, ライバルエージェントが模倣するためのプロンプトを作成せよ」という指示を与える。この出力結果をライバルエージェントのシステムプロンプトに動的に組み込むことで, ライバルエージェントは「プレイヤーのような行動」で発話を行うことが可能となる。

4.3.3 会話スタイルの模倣 (Style Transfer の適用)

行動内容だけでなく「口調」も模倣するために, Roy らの Conversation Style Transfer (CST) 手法 [16] を応用する。CST は Few-shot 学習により任意のスタイルへ変換を行う手法であるが, 本研究では変換ターゲットを「リアルタイムに更新されるプレイヤーのスタイル」に設定している。

a) スタイルデータセットの動的構築

CSTにおける「スタイル除去 (Style Reduction)」の概念を用い、プレイヤーの発話 u_p からスタイルを取り除いた中立文 u_n を LLM により生成する。本システムでは、ターンが進行するたびにペア (u_n, u_p) をデータセット S_{player} に追加し続けることで、プレイヤーの口調の変化に追従する学習データセットを動的に構築する。

b) 検索ベースのスタイル変換

ライバルエージェントが発話意図 (ドラフト) を生成した際、CST の Dynamic Prompt Selection の手法に基づき、 S_{player} の中から意味的に近い事例を検索する。検索された「中立文とプレイヤー発話のペア」を In-Context Learning の例示 (Demonstrations) として LLM に提示することで、ライバルの発話をプレイヤー特有の語彙や言い回しに変換して出力する。

4.4 プレイヤー知識の模倣

本手法では、知識グラフを単なるデータベースとしてではなく、「プレイヤーが獲得した知識の表現」として利用し、ライバルエージェントの知識レベルを制御する。

4.4.1 知識グラフの動的更新

対話エージェントは初期状態の知識グラフ G_0 を保持している。ターン t におけるプレイヤーとの対話 D_t において、新しい事実が発生した場合、その情報は新しいノード n_{new} またはエッジ e_{new} として知識グラフに追加される。

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

これにより、知識グラフは常に「その時点までにプレイヤーが知り得た情報」を網羅的に表現する状態となる。

4.4.2 アクセスレベルの決定と模倣

プレイヤーのアクセスレベル V_p を、現在の知識グラフ G_t においてプレイヤーがアクセス済み (既知) である情報の範囲と定義する。これは会話履歴 H を解析することで特定される。ライバルエージェントのアクセスレベル V_r (すなわち G_{rival} の範囲) は、この V_p を基準に決定される。ライバルはプレイヤーと同等の知識を持つことで競合相手として機能するが、ゲームとしての揺らぎを持たせるため、 V_p に対して微小な変動を与えた範囲を設定する。

$$V_r \approx V_p \pm \epsilon \quad (6)$$

ここで ϵ は状況に応じた変動幅を表す。具体的には、プレイヤーが知っている情報の集合に加え、それに関連する未知の情報をわずかに含めるか、あるいは一部を欠落させた状態で G_{rival} を構築する。これにより、ライバルは「プレイヤーとほぼ同じことを知っているが、少しだけ違う視点を持つ」存在となる。

5 実験と結果

本提案手法の有効性を検証するため、被験者を用いた評価実験を行った。実験に使用した環境および設定パラメータを表 2 に示す。



図 4 実験に使用したゲーム画面

5.1 ゲーム概要とルール

実験では、ヒロイン「しゃちこ」と会話を行い、関係性を深めていく恋愛シミュレーションゲームを使用した。ゲームの流れは以下の通りである。

1. プレイヤーはヒロインに対するメッセージを入力する。
2. プレイヤーと同時にライバルエージェントもメッセージを入力する。
3. ヒロインは、プレイヤーとライバルの2つのメッセージに対し、どちらが文脈に適しているか、あるいは好ましいかを判断する。
4. 判定結果に基づき状態遷移図の遷移方向を [very positive, positive, neutral, negative, very negative] の5段階で決定する。
5. 状態遷移図の遷移方向によってプレイヤーとヒロインとの関係性を変化させる。遷移方向が positive または negative の場合は1段階、very positive または very negative の場合は2段階関係性が変化する。neutral の場合は関係性が変化しない。

実際に使用したゲーム画面を図 4 に示す。

5.2 実験項目 (比較システム)

本実験では、以下の3つの条件 (システム) を用意し、比較を行った。

System 1 (Normal) 提案手法。ライバルエージェントがプレイヤーの会話履歴から生成されたペルソナトークンを用いて、プレイヤーの思考や口調を模倣する。

System 2 (No Persona) 比較手法 1。ライバルエージェントは存在するが、プレイヤーの模倣を行わず、一般的なライバルとして振る舞う。

System 3 (No Rival) 比較手法 2。ライバルエージェントが存在しない。ヒロイン役の LLM がプレイヤーの発話のみに基づいて遷移を判断する。

表 2 実験環境およびパラメータ設定

項目	設定値・内容
ライバルの LLM	gemma3:27b
ヒロインの LLM	qwen3:30b-a3b-instruct-2507-fp16
初期の知識グラフのノード数	70
状態遷移図のノード数	12
プレイ時間	15 min

5.3 実験手順

実験は以下の手順で行われた。

- ゲームの説明:** 被験者に対し、ゲームの目的（ヒロインとの関係深化）および操作方法を説明する。
- ゲームプレイ:** 被験者は、2種類の比較システムと本研究のシステムをそれぞれ 15 分間プレイする。
 - 各システムのプレイ順序は、学習効果や順序効果を排除するため、被験者ごとにランダムに決定した。
 - ヒロインの性格は 3 種類用意し、プレイヤーごとにランダムに 3 つのシステムへ割り振った。
- アンケート:** 全システムのプレイ終了後、各システムに対する評価アンケートを実施した。

5.4 取得項目

本実験では、各システムの効果を検証するために以下のデータを取得した。

- 勝率 (Win Rate):** 各システムにおけるプレイヤーの勝利回数の割合。
- Talk Score:** プレイヤーの会話能力を測定するための指標。Gemini3 Pro を用いて、プレイヤーの会話履歴からその能力を 10 段階で評価した。
- アンケート回答:** 実験後のアンケートにより、各システムにおける「楽しさ (Enjoy)」と「難しさ (Difficulty)」を 5 段階のリッカート尺度で評価した結果。

5.5 結果

実験により得られた結果について述べる。

5.5.1 アンケート結果と勝率

各被験者のゲームプレイにおける勝率および Talk Score を表 3 に、プレイ後のアンケート結果（楽しさと難しさ）を表 4 に示す。アンケート結果について、Normal と No Persona における「楽しさ (Enjoy)」の平均スコアを比較すると、それぞれ 3.6 および 3.3 であり、両者の間に大きな差は見られず、統計的に有意な変化は確認されなかった。

5.5.2 統計的分析

各手法の勝率について、対応のある t 検定を行った結果を表 5 に示す。表のはみ出しを防ぐため、各システム名はそれぞれ、System 1 (Normal) は Normal, System 2 (No Persona) は No Persona, System 3 (No Rival) は No Rival と略記する。

表 3 各手法における勝率 (%) と Talk Score

User	Normal	No Persona	No Rival	Talk Score
Player1	22.2	10.0	100.0	4.5
Player2	27.3	7.7	90.9	2.0
Player3	20.0	46.2	83.3	7.5
Player4	44.4	10.0	90.9	6.0
Player5	57.1	11.1	100.0	4.5
Player6	22.2	30.0	77.8	8.5
Player7	27.3	30.8	100.0	9.5
Player8	25.0	33.3	72.7	10.0
Player9	8.3	36.4	75.0	7.0
Average	28.2	23.9	87.9	-

表 4 アンケート結果 (5 段階評価: 5 が高い)

User	Normal		No Persona		No Rival	
	Enjoy	Diff	Enjoy	Diff	Enjoy	Diff
Player1	4	3	2	5	4	2
Player2	3	3	4	3	3	2
Player3	3	3	3	3	3	2
Player4	3	2	3	5	3	1
Player5	4	4	3	4	4	1
Player6	5	5	5	2	5	1
Player7	4	5	2	4	5	1
Player8	1	1	3	2	1	1
Player9	5	5	5	2	5	1
Average	3.6	3.4	3.3	3.3	3.7	1.3

表 5 統計分析結果 (勝率の対応のある t 検定)

比較ペア	t 統計量	p 値
Normal vs No Persona	0.4991	0.63113
Normal vs No Rival	-14.6506	0.00000
No Persona vs No Rival	-8.3592	0.00003

表 5 に示すように、有意水準 5% ($p < 0.05$) において、以下の結果が得られた。

- Normal vs No Persona:** $p \geq 0.05$ であり、有意差は認められなかった。
- Normal vs No Rival:** $p < 0.05$ であり、有意差が認められた。
- No Persona vs No Rival:** $p < 0.05$ であり、有意差が認められた。

以上の結果より、ライバルエージェントが存在する条件 (Normal および No Persona) は、ライバルが存在しない条件 (No Rival) と比較して、統計的に有意にプレイヤーの勝率が低下することが確認された。これは、ライバルエージェントがプレイヤーの妨害役として機能し、ゲームの難易度を有意に上昇させていることを示している。一方で、提案手法である Normal と既存の No Persona の間には、全体平均として統計的に有意な差は見られなかった。これは、次項で述べるように、プレイヤーのスキルレベルに応じて勝率の変化方向が異なる（高スキル者には難しく、低スキル者には易くなる）ため、全体としての

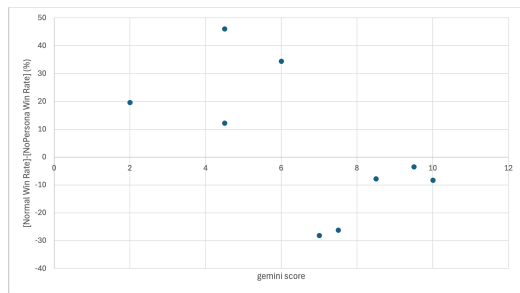


図5 Gemini Score と勝率差 (Normal - NoPersona Win Rate) の関係

平均値の差が相殺された結果であると言える。

図5に Talk Score (Gemini Score) と勝率差 (Normal - NoPersona) の関係を示す。

被験者を Talk Score の中央値を基準に「高スコア群 (Score ≥ 7.0)」と「低スコア群 (Score < 7.0)」に分類して分析すると、以下の傾向が確認された。

1. **高スコア群:** Normal の勝率 (平均 20.6%) は No Persona の勝率 (平均 35.3%) よりも低くなった。
2. **低スコア群:** Normal の勝率 (平均 37.8%) は No Persona の勝率 (平均 9.7%) よりも高くなった。

これは、高スコアのプレイヤーに対してはライバルも高度な模倣を行うため強力になり (難易度上昇)、低スコアのプレイヤーに対してはライバルも同様に低いパフォーマンスとなるため (難易度低下)、結果として勝率が逆転したと考えられる。

6 考察

6.1 難易度調整の実現

実験結果より、提案手法 (Normal) はプレイヤーのスキルレベルに応じて動的に難易度 (勝率) を変化させる機能を持つことが示された。従来の No Persona 手法では、ライバルの強さが一定であるため、プレイヤーのスキルによって勝率に偏りが生じる。一方、提案手法では、ライバルエージェントがプレイヤーを模倣することで、強いプレイヤーには強く、弱いプレイヤーには弱く振る舞うようになり、常に拮抗した勝負を提供する調整機能が働いていると言える。t 検定において Normal と No Persona の全体平均に差がなかったのは、この「高スキル者への難化」と「低スキル者への易化」が相殺し合った結果であると解釈できる。これにより、ライバルエージェントへのペルソナ導入による難易度制御の有効性が示唆された。

6.2 楽しさと難易度調整の関連性

本研究の主目的は、プレイヤーのスキルに応じた適切な難易度調整を行うことで、ゲーム体験における「楽しさ」を増加させることにある。アンケート結果においては、Normal と No Persona の間の楽しさに統計的な有意差は確認されなかった。しかし、これは提案手法が楽しさを損なったことを意味するものではない。7.1 節で述べたように、Normal 手法はプレイヤー

のスキルレベルに応じて勝率 (難易度) を拮抗させることに成功している。チクセントミハイのフロー理論 [17] によれば、個人のスキルと課題の難易度が釣り合った状態こそが没入や楽しさを生む条件である。本実験では、ランダムに割り当てられたヒロインの性格への好みが楽しさの評価に影響を与えた可能性があるが、難易度制御という楽しさを支える基盤機能に関しては、提案手法が最も適切に動作していたことが確認された。No Rival (ライバル不在) の楽しさの平均値が高い傾向にあったが、これは対話そのものを楽しむ層の影響と考えられ、競争要素を含むゲームデザインにおいては、Normal 手法のようにプレイヤーの実力に合わせて「勝てるか負けるか分からない」緊張感を維持する仕組みが、長期的なプレイ意欲やより深い楽しさの創出に寄与すると考えられる。

7 まとめ

本研究では、LLM を利用した対話型ゲームにおいて、ライバルエージェントにプレイヤーを模倣させることで、動的に難易度を調整する手法を提案した。提案手法は、プレイヤーの会話履歴、知識グラフ、会話スタイルに基づいたペルソナトークンを生成し、ライバルエージェントがプレイヤーと同等の知識と振る舞いを持つように制御するものである。

被験者実験の結果、ライバルエージェントが存在することでゲームの難易度が有意に上昇することが確認された。さらに、提案手法を用いることで、会話能力の高いプレイヤーに対しては難易度が上昇し、低いプレイヤーに対しては低下するという、プレイヤーのスキルに応じた適切な難易度調整が実現されていることが示された。これは、固定的な強さを持つライバルエージェント (No Persona) と比較して、より柔軟で適応的な対戦環境を提供できることを意味する。

アンケート結果における「楽しさ」の評価では統計的な有意差は見られなかったものの、難易度とスキルを拮抗させる本手法は、フロー理論に基づけばプレイヤーの没入感を高めるための重要な基盤となり得る。今後は、より多様なゲームシナリオや長期的なプレイにおける適応精度の検証、および難易度調整がプレイヤーの感情や継続意欲に与える影響の更なる分析が課題である。

8 謝辞

本研究の一部は科研費基盤 B (23K28094) による。

文 献

- [1] Marvin Zammit Sam Earle Antonios Liapis Julian Togelius Roberto Gallotta, Graham Todd and Georgios N. Yannakakis. Large language models and games: A survey and roadmap. *IEEE Transactions on Games*, 2024.
- [2] Christian Guckelsberger Susanna Va itinen, Perttu Ha mäläinen. Generating role-playing game quests with gpt language models. *IEEE Transactions on Games*, 2022.
- [3] Carrie J. Cai Meredith Ringel Morris Percy Liang Michael S. Bernstein Joon Sung Park, Joseph C. O'Brien. Generative agents: Interactive simulacra of human behavior. 2023.

- [4] Guanzhi Wang, Yuqi Xie, Yunfan Jiang, Ajay Mandilekar, Chaowei Xiao, Yuke Zhu, Linxi Fan, and Anima Anandkumar. Voyager: An open-ended embodied agent with large language models. *Transactions on Machine Learning Research*, 2024.
- [5] Carlos Souza, Saulo Oliveira, Luciana Berretta, and Sérgio Carvalho. Large language models and dynamic difficulty adjustment: An integration perspective. In *Anais Estendidos do XXIII Simpósio Brasileiro de Jogos e Entretenimento Digital*, pages 31–36, Porto Alegre, RS, Brasil, 2024. SBC.
- [6] 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, editors, *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 13153–13187, Singapore, December 2023. Association for Computational Linguistics.
- [7] Junqi Dai Xipeng Qiu Yunfan Shao, Linyang Li. Character-llm: A trainable agent for role-playing. *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, 2023.
- [8] Zekun Moore 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, Stephen W. Huang, Jie Fu, and Junran Peng. Rolellm: Benchmarking, eliciting, and enhancing role-playing abilities of large language models, 2024.
- [9] Cheng Li, Ziang Leng, Chenxi Yan, Junyi Shen, Hao Wang, Weishi MI, Yaying Fei, Xiaoyang Feng, Song Yan, HaoSheng Wang, Linkang Zhan, Yaokai Jia, Pingyu Wu, and Haozhen Sun. Chatharuhi: Reviving anime character in reality via large language model, 2023.
- [10] Ali Emami Qi chen Gao. The turing quest: Can transformers make good npcs? *IEEE Transactions on Games*, 2022.
- [11] Jakub Myśliwiec Judith van Stegeren. Fine-tuning gpt-2 on annotated rpg quests for npc dialogue generation. *FDG '21*, 2021.
- [12] 馬強 角田 悠之介, 中園 康聖. 恋愛シミュレーションゲームにおける会話難易度調整に関する研究. *DEIM 2025*, 2025.
- [13] Jiangjie Chen, Xintao Wang, Rui Xu, Siyu Yuan, Yikai Zhang, Wei Shi, Jian Xie, Shuang Li, Ruihan Yang, Tinghui Zhu, Aili Chen, Nianqi Li, Lida Chen, Caiyu Hu, Siye Wu, Scott Ren, Ziquan Fu, and Yanghua Xiao. From persona to personalization: A survey on role-playing language agents. *Transactions on Machine Learning Research*, 2024. Survey Certification.
- [14] Sirui Hong, Mingchen Zhuge, Jonathan Chen, Xiawu Zheng, Yuheng Cheng, Jinlin Wang, Ceyao Zhang, Zili Wang, Steven Ka Shing Yau, Zijuan Lin, Liyang Zhou, Chenyu Ran, Lingfeng Xiao, Chenglin Wu, and Jürgen Schmidhuber. MetaGPT: Meta programming for a multi-agent collaborative framework. 2024.
- [15] Shuting Wang Xiaochi Wei Erxue Min Yu Lu Shuaiqiang Wang Dawei Yin Zhicheng Dou Jiongnan Liu, Yutao Zhu. Llms + persona-plugin = personalized llms. *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics*, 2025.
- [16] Nikolaos Pappas Elman Mansimov Yi Zhang Saab Mansour Dan Roth Shamik Roy, Raphael Shu. 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*, 2023.
- [17] M. Csikszentmihalyi. Flow: The psychology of optimal experience. ,*Harper Row, New York, NY,USA*, 2009.
- [18] Mohammad Zohaib. Dynamic difficulty adjustment (DDA) in computer games: A review. *Adv. Hum. Comput. Interact.*, 2018:5681652:1–5681652:12, 2018.
- [19] Ronja Fuchs, Robin Gieseke, and Alexander Dockhorn. Personalized dynamic difficulty adjustment imitation learning meets reinforcement learning. In *IEEE Conference on Games, CoG 2024, Milan, Italy, August 5-8, 2024*, pages 1–2. IEEE, 2024.
- [20] Robin Hunicke and Vernell Chapman. Ai for dynamic difficulty adjustment in games. *Challenges in game artificial intelligence AAAI workshop*, 2, 01 2004.