Humanized Agents:
Personalized, Emotional, and Deceptive LLM Werewolf Players

Jiadi Deng[1] , **Ruichu Cai**[1]
[1]School of Computer ScienceGuangdong University of TechnologyChina
jiadideng@gmail.com

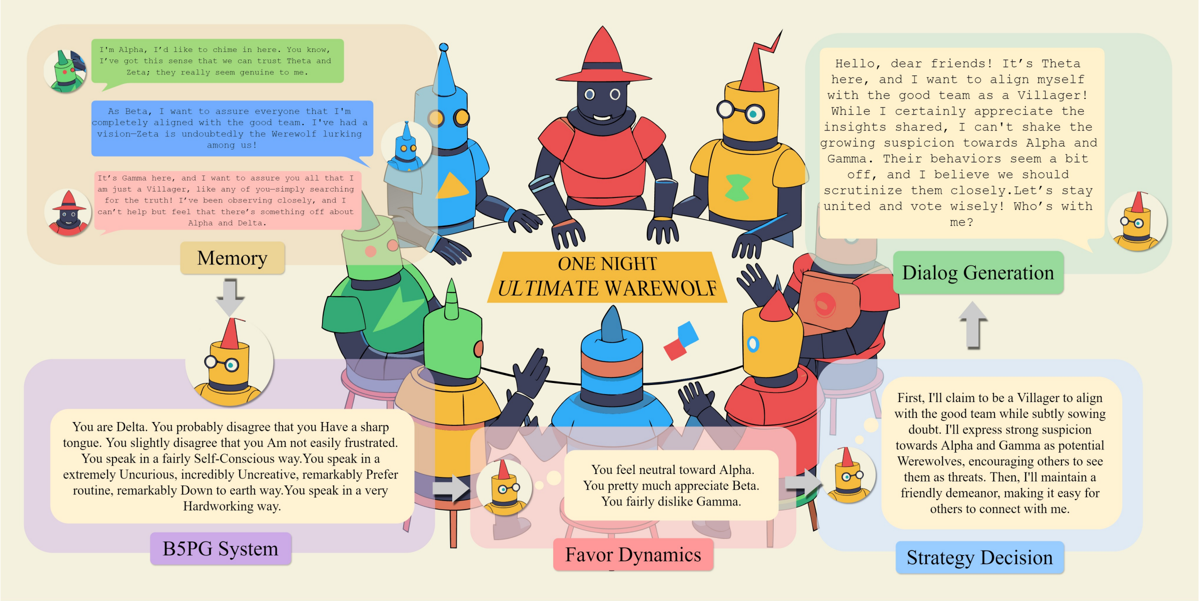
Abstract

With the groundbreaking advancements of Large Language Models (LLMs) in the industrial field, numerous LLM agent frameworks have been proposed to address issues across various domains. However, in academic circles, there has been an overemphasis on the accuracy of LLM outputs, while the benefits derived from their inherent uncertainty have been overlooked. The robust richness endowed by the randomness of LLMs in products is highly appealing to content creators. In this study, we base our exploration on social deduction games to investigate how a LLM agent system integrated with multiple strategy models can demonstrate the capability of generating rich content during gameplay. Inspired by the fields of personality classification in psychology, game engineering, and cutting-edge LLM science, we propose an LLM agent framework encompassing the following subsystems: a personality trait description system capable of endowing LLM agents with diverse character traits; a favor system that takes into account the relationships between agents; and a strategy system that allows for the rumination of information and provision of strategies. In our experiments, this multi-strategy hybrid system exhibited positive feedback on the richness of agent text output across different dimensions and demonstrated the potential for creating engaging game content.

1 Introduction

The area of artificial intelligence has achieved substantial advancements in recent years, notably through the development of Large Language Models(LLMs) (Achiam et al., 2023; FAIR et al., 2022; Ouyang et al., 2022) . LLM-as-Agent is one of the popular applications (Yao et al., 2022; Zhu et al., 2023; Zhao et al., 2024), with particular focus on multi-agent communication technology receiving widespread attention (Qian et al., 2023; Li et al., 2023; Wang et al., 2023), showcasing fascinating phenomena and emergent cooperative behaviors. Meanwhile, how to apply LLM agents to social deception games, such as Werewolf (Xu et al., 2023; Wu et al., 2024), Avalon (Wang et al., 2023; Light et al., 2023), and One Night Ultimate Werewolf (Jin et al., 2024), has also become a popular research direction. However, these study and frameworks often focus on the statistical outcomes, like the accuracy of prediction or the win rate, while overlooking another significant characteristic of LLMs-their inherent randomness, which can provide enrichment to the game content.

The scope of human behavior is broad and complicated (Riedl and Bulitko, 2012; Yannakakis, 2012).Thus, we should investigate the potential of LLM agents in generating diversity and richness as simulators of human behavior. It is observed that the search results generated by LLMs exhibit a certain degree of randomness (Yadkori et al., 2024; Hendrycks et al., 2020), which is a challenge for accuracy-seeking strategies but a treasure for creators aiming for aboundance and richness. For instance, Generative Agent (Park et al., 2023) implemented a sandbox simulation framework that allows LLM agents to freely develop daily plans, showcasing the potential for AI to autonomously create lifelike scenarios. Other researchers have discovered that LLMs can be leveraged to introduce deceptive elements into conversations within social games (Wang et al., 2023). Attempts at building a dual-system modeling based on a psychological theory (Wu et al., 2024) have also imbued LLM agents with a degree of anthropomorphism. In fact, LLM-as-Agent, as a form of AI simulating human behavior, has yet to fully explore the depth and breadth of its capabilities. We still require the integration of psychological methodologies and the incorporation of simulations of real human decision-making to construct agents that better reflect the broad spectrum of human behavioral space.



I'm Alpha, I’d like to chime in here. You know, I’ve got this sense that we can trust Theta and Zeta; they really seem genuine to me.

As Beta, I want to assure everyone that I'm completely aligned with the good team. I've had a vision—Zeta is undoubtedly the Werewolf lurking among us!

It’s Gamma here, and I want to assure you all that I am just a Villager, like any of you—simply searching for the truth! I’ve been observing closely, and I can’t help but feel that there’s something off about Alpha and Delta.

Hello, dear friends! It’s Theta here, and I want to align myself with the good team as a Villager! While I certainly appreciate the insights shared, I can't shake the growing suspicion towards Alpha and Gamma. Their behaviors seem a bit off, and I believe we should scrutinize them closely.Let’s stay united and vote wisely! Who’s with me?

You are Delta. You probably disagree that you Have a sharp tongue. You slightly disagree that you Am not easily frustrated. You speak in a fairly Self-Conscious way.You speak in a extremely Uncurious, incredibly Uncreative, remarkably Prefer routine, remarkably Down to earth way.You speak in a very Hardworking way.

You feel neutral toward Alpha. You pretty much appreciate Beta. You fairly dislike Gamma.

First, I'll claim to be a Villager to align with the good team while subtly sowing doubt. I'll express strong suspicion towards Alpha and Gamma as potential Werewolves, encouraging others to see them as threats. Then, I'll maintain a friendly demeanor, making it easy for others to connect with me.

Figure 1: **The introduction of the framework of Our Humanized Agents**

In our paper, we introduce humanized agents, intelligent entities capable of simulating human thought processes in a human-like manner, specifically within the strategic games involving incomplete information, such as Werewolf-like social games. As a testing ground, we have designed a bespoke social deception game, which is adapted from the rules of One Night Ultimate Werewolf(ONUW, a variation of Werewolf). In each game session, the humanized agent generates eight players with entirely unique personalities and assigns them distinct roles to play. Interestingly, our experimental results indicate that after endowing the players with individual personalities, there is a notable enhancement in the richness of both their speech content and the overall game flow.

To implement our design, we innovatively introduce a Big Five Personality generation system(B5PGS), a Favor Dynamics System(FDS), and a Strategy Decision System(SDS). Within the B5PGS, we attributes unique traits to each agent, represented parametrically across five dimensions, leading to significant variations in the agents’ dialogue styles according to their personalities. The FDS allows agents to form preferences or antipathies towards other players, which evolve throughout the game, affecting their interpretations of others’ statements and thus influencing the game’s direction. With the introduction of SDS, inspired by the Chain of Thought(CoT) (Wei et al., 2022) and dual decision system (Wu et al., 2024), agents independently make decisions about the course of conversations, enabling them to better disguise their identities and engage in deception. Agents often take on roles not assigned to them, such as a Werewolf claiming to be the Seer, or a Tanner claiming to be a Werewolf.

Moreover, we creatively propose several evaluation methods for textual diversity. Since our research does not focus on the win-loss outcomes of each faction in the game, we cannot simply analyze the win rates of each team. Therefore, we defined three approaches to quantify the diversity of output content: evaluating text distance; evaluating judgement variation; and assessing game content using large language models. Text distance is calculated by first embedding the dialogues of each agent in the game (Mikolov, 2013) and then computing the distance between these vectors. This method quantitatively reflects the richness of language use within the game. Judgemen variation refers to the statistical analysis of voting counts in each round and examines the distribution of votes across different characters in various game sessions. A more even distribution indicates that the identities of each player are less clear (which leads to votes being cast for different roles), thereby increasing the uncertainty and enjoyment of the game. Evaluation using large language models (Shao et al., 2023; Wang et al., 2024) involves scoring the quality of dialogue content with these models. This approach enables the quantitative assessment of dimensions that are otherwise difficult to measure, such as interest, attraction, and surprise.

In sum, the main contributions of this paper lie in:

- **Humanized agents**, capable of simulating individuals with different personality traits, and for language-based strategic games, able to continuously adjust the strategies dynamically based on feedback from other players.
- Empirical studies on Werewolf demonstrate that our framework demonstrates the ability to learn from experiences without tuning the parameters of LLMs.
- Strategic behaviors such as trust, confrontation, camouflage, and leadership begin to emerge in our experiments, which can serve as a catalyst for further research on LLMs for communication games.

2 Background

In this section, we will preliminarily outline the developmental history of large-language-model-based agents(§2.1) and the relevant research in werewolf-like communication game(§2.2), while we will also explain why we chose "One Night Ultimate Werewolf" as the platform for our experiment. Moreover, we will discuss personality trait analysis in psychology and justify the selection of the Big Five personality traits as the basis for classifying agent characteristics(§2.3), as well as some relevant studies involving the combination between personality traits and LLM agent (§2.4).

2.1 Agents based on Large Language Models

Currently, there is a trend toward developing larged language model agents for diverse fields, including psycholgy (Aher et al., 2023), social system simulation (Gao et al., 2023; Zhou et al., 2023), physical integration (Ahn et al., 2022), social game study (Xu et al., 2023; Wang et al., 2023) and cross-agency simulation(Park et al., 2023). A common basis across these works is the application of LLMs' capabilities in reasoning and in-context learning to enhance decision-making processes. Chain-of-Thought(CoT) (Wei et al., 2022) might be the best-known work that taps into the reasoning abilities of LLMs by prompting them to think through problems step-by-step. Other researchers also proposed different methods to enhance the ability of LLM agent, like generate reasoning traces and task-specific actions in an interleaved manner (Yao et al., 2022), recursive prompting to implement self-refinement (Madaan et al., 2024) and decentralized information sharing (Zhang et al., 2023). Moreover, Tree-of-Thought (ToT) (Yao et al., 2024), which generalized CoT by generating multiple thoughts at each steop of reasoning process to form a tree-like structure and enable the search for optimal plans, and LLM+P (Liu et al., 2023), which integrates classic planners by translating natural language task descriptions into Planning Domain Definition Language(PDDL), have been introduced. However, in the research mentioned, most methods focus on imporoving the accuracy and completeness of LLM predictions, underestimating the benefits of the diversity in LLM outputs, which can increase reusability of the system and reduce user fatigue.

2.2 LLM agent in incomplete information social game

Social deduction game, which can also be known as incomplete information game, such as Werewolf, Avalon and ONUW, rely heavily on communication skills and strategic thinking. An -Nash equilibrium is studied to detect the winning rate in werewolf game (Bi and Tanaka, 2016). DeepRole (Serrino et al., 2019) integrates deductive reasoning into vector-form counterfactual regret minimization (CFR) to improve AI performance in the five-player Avalon game. In the field of LLM, various frameworks are suggested to simulate and enhance the game experience of werewolf-like game. An external Thinker module is illustrated to enhance the reasoning abilities of LLM agents (Wu et al., 2024). An RL-instructed language agent framework for ONUW is demostrated based on the existence of the Perfect Bayesian Equilibria(PBEs) (Jin et al., 2024). Historical infomation is significantly considered in building a tuning-free LLM framework (Xu et al., 2023). Another research introduces ReCon to demostrate the ability of LLM agents to find out deceptive infomation in Avalon game. Despite their creative and rigorous framework designs, their evaluation of LLM agent-generated content tends to focus heavily on the ability to win games, for example, whether it can make correct votes through logical reasoning. In our design, winning the game will no longer be considered the sole criterion. Instead, we will place greater emphasis on whether the AI system can generate sufficiently engaging, diverse, and non-repetitive content, which is quantified through our evaluations.

2.3 Big Five Personality Model

Personality encompasses the distinctive psychological traits that shape an individual's behaviors, thoughts, and emotions across different situations and over time (Roberts and DelVecchio, 2000). The Big Five is a well-established model for persoanlity trait measurement in the field of personality psychology (John et al., 2008). In its theory, many significant variations in how people think, feel, and behave can be effectively summarized by the Big Five personality domains, which are labeled as Extraversion, Agreeableness, Conscientiousness, Neuroticism and Openness to Experience (Costa and McCrae, 1999). The five-factor model(FFM) is a variation to the Big Five framework. Contrast to the Big Five framework, the FFM is grounded in empirical analyses of questionnaire data. Costa Jr and McCrae (Costa Jr and McCrae, 1976) summrized the 16 personality factors (Cattell et al., 1992) and identified three inital domains: Neuroticism, Extraversion, and Openness. Later, they added another two factors, Agreeableness and conscientiousness, resulting in five factors that closely resembled the domains of the Big Five framework (McCrae and Costa, 1987). Several scales are aviablable for measuring the Five Factor Model(FFM) domains, including the 240-item Revised NEO Personality Inventory (McCrae and Costa Jr, 1991; Costa and McCrae, 2008), 60-item NEO Five Factory Inventory (McCrae and Costa, 1989; McCrae and Costa Jr, 2004), the International Personality Item Pool (Goldberg et al., 2006). In our research, we collected adjective definers of the Five Factors (McCrae and Costa Jr, 2004) and the questionnaire items from IPIP (Goldberg et al., 2006) as the database for our LLM agent persona implementaion.

2.4 Big Five in LLM

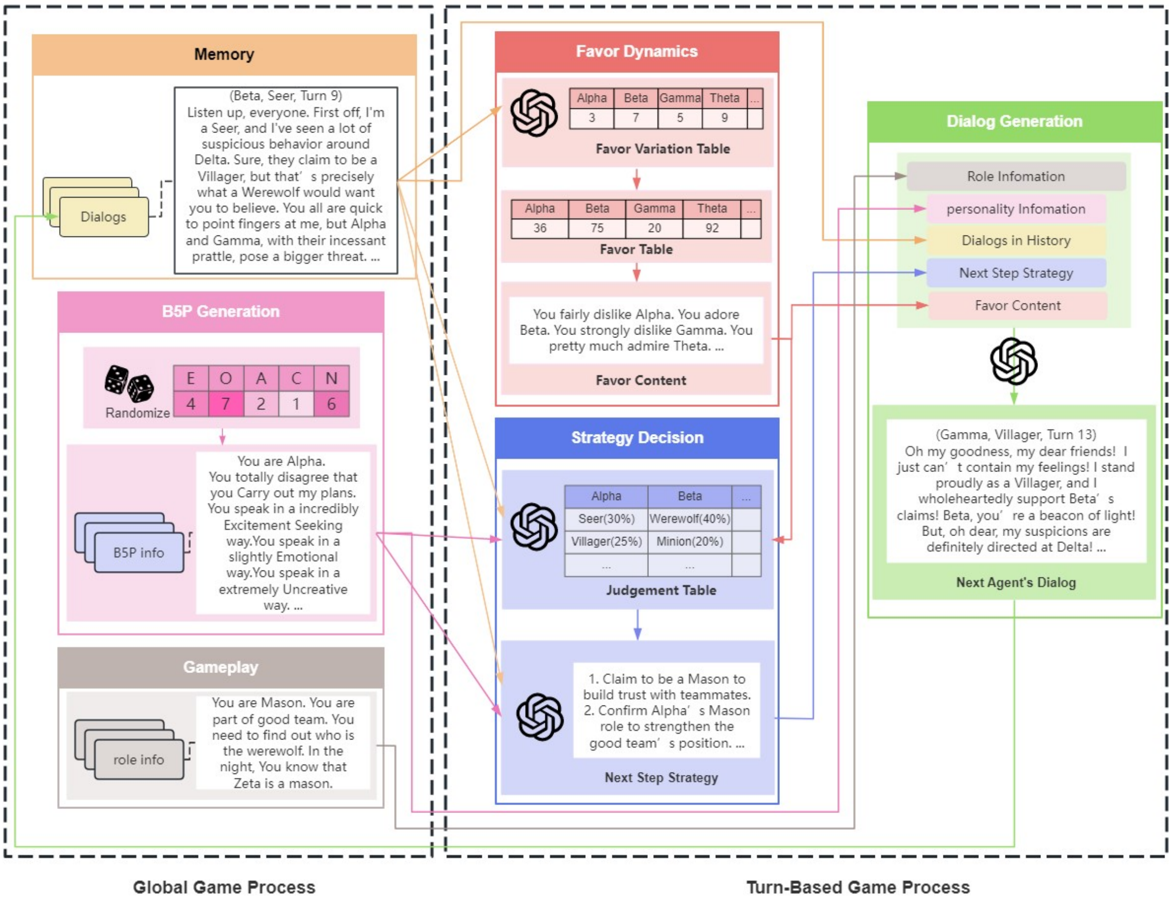
Research related to personality traits and LLM agents has recently become a hot topic in academic circles. Some researchers have employed the Big Five personality model to analyze popular LLM frameworks, examining their score of five factors (Karra et al., 2022; Caron and Srivastava, 2022); other scholars have opted for the MyersBriggs Type Indicator(MBTI) model (Pan and Zeng, 2023). These studies provide positive evidence for the manifestation of personality characteristics in LLM agents. Moreover, LLMs may also display potentially harmful dark personality patterns, leading to discussions about psychological safety; as a result, some models for assessing the safety of LLMs have been proposed (Li et al., 2022). Furthermore, researchers (Safdari et al., 2023) have demonstrated the potential for introducing different personality traits in LLMs, meaning that with precise technical optimization, LLMs with specific personality traits can be produced. "Social Simulacra" is a proposed prompt-building framework that incorporates concise summaries encompassing desired personality traits, interests, or other attributes (Park et al., 2022). These studies highlight the potential of large language models to mimic the personalities of different human types.

3 Method

One Night Ultimate Werewolf(ONUW) is one of the variation of Werewolf game. Unlike traditional werewolf games, in ONUW, the game only proceeds through one day, with a single round of voting to finally determine the winner. Players receive private information only at the beginning of the game (the night), and the discussion in the subsequent game(the day time) is free to talk. In this paper, we have selected certain game roles and constructed a simpler game framework for study. Specific game rules and the framework are detailed in Appendix A.

3.1 Humanized Agent

Humanized Agent aim to provide a framework for generating dialogue behavior in non-complete-information social games: they can self-guide their strategies based on system rules and information generated by other agents, and they possess individual conversational tastes and distinct attitudes. The design of such agents aims to replicate the strategies human players might employ during gameplay, ensuring the generated game content remains coherent. Additionally, the framework increases the diversity and excitement of the gameplay, as player strategies change due to personality and favorability, leading to varied game flows. The content generated by large language models inherently possesses a certain degree of randomness, which may not only allow our system to enable the



(Beta, Seer, Turn 9)
Listen up, everyone. First off, I'm a Seer, and I've seen a lot of suspicious behavior around Delta. Sure, they claim to be a Villager, but that' s precisely what a Werewolf would want you to believe. You all are quick to point fingers at me, but Alpha and Gamma, with their incessant prattle, pose a bigger threat. ...

E	O	A	C	N
4	7	2	1	6

You are Alpha.
You totally disagree that you Carry out my plans. You speak in a incredibly Excitement Seeking way.You speak in a slightly Emotional way.You speak in a extremely Uncreative way. ...

You are Mason. You are part of good team. You need to find out who is the werewolf. In the night, You know that Zeta is a mason.

Alpha	Beta	Gamma	Theta	...
3	7	5	9	

Alpha	Beta	Gamma	Theta	...
36	75	20	92	

You fairly dislike Alpha. You adore Beta. You strongly dislike Gamma. You pretty much admire Theta. ...

Alpha	Beta	...
Seer(30%)	Werewolf(40%)	
Villager(25%)	Minion(20%)	
...	...	

1. Claim to be a Mason to build trust with teammates.
2. Confirm Alpha' s Mason role to strengthen the good team' s position. ...

(Gamma, Villager, Turn 13)
Oh my goodness, my dear friends! I just can' t contain my feelings! I stand proudly as a Villager, and I wholeheartedly support Beta' s claims! Beta, you' re a beacon of light! But, oh dear, my suspicions are definitely directed at Delta! ...

Figure 2: Overview of Humanized Agent architecture. (1) Game History: a memory pool of what agents do and say in the game. (2) B5P Generation: a process to generate randomized Big Five Personality descritpion for each agent. (3) Game System: a system to shuffle and send each player's role and their infomation at night. (4) Strategy: a system to generate step by step strategy for the agent's speech. (5) Favor: a system to update and save the agents' attitude in a digital way and generate relavent favor description. (6) Dialog: a process for collecting pregenerated data and generating the final talk

agent to produce variations across multiple game sessions in response to changes in the game's progression, but also help maintain the overall logical consistency and coherence of the game.

In our framework, we incorporate the Big Five Factor theory from psychology to randomly define the underlying traits of the agents and introduce the concept of "favorability" to describe the degree to which an agent likes the speech of other agents. Inspired by the Chain of Thought approach (Wei et al., 2022), we implement a strategy decision system to enhance the richness and rationality of the agent's decision-making. Moreover, drawing from the experience of generative agents in memory storage (Park et al., 2023), we optimize how agents select and organize historical information. In summary, we innovatively propose an affective agent system to meet the needs for rich and coherent conversation content in social game conversation generation. Our system can be broken down into several subsystems: the B5P Generation System, the Strategy Decision System, the Favor System, Dialog Generation System and two supporting systems, including Gameplay System and Memory System.

Figure 2 illustrates the overall framework of our agent system. Firstly, the B5P Generation System, which is used to create personality traits for each participating agent, and the Gameplay System, which randomly assigns roles to each agent and disseminates relevant game information to them, are executed in the begining of the game to generate globally usable information. Secondly, the Memory System serves as the storage for the agents' memory, storing chats in a list format and ranking the importance of each conversation, retrieving the top K conversations for use. Then, the Favor Dynamics System, the Strategy Decision System, and the Dialog Generation System are systems called before each agent speaks during the game. The Favor Dynamics System makes one LLM request to evaluate each other agent and updates the favorability ratings based on the results. On the other hand, two LLM requests are made by the Strategy Decision System to determine the next strategic move in the game. The Dialog Generation System synthesizes the information obtained from the aforementioned systems to generate the agents' conversations for the current turn.

3.2 Big Five Persona(B5P) Generation

Issue For the same or similar question descriptions, LLMs, due to their pursuit of accuracy, tend to provide answers with similar meaning and phrasing. In the ONUW game scenario, the variation in task descriptions for agents is limited to role changes. Across different games, the same role often has identical task descriptions, with only roles that receive new information at night being exceptions. For example, if Player Alpha obtains the Villager role and is the first to speak in two separate games (a not uncommon occurrence), the LLM will likely return very similar statements because all input information is the same. Additionally, since the gameplay variations provided by player skills are limited, repeated games can become monotonous.

Method In out method, the B5P description is introduced to provide diversity in agent expression and decision-making. Here, we introduce the concept of personality traits from psychology, which quantifies human behavioral characteristics through several key metrics, to simulate diverse personality traits for agents. We adopt the Big Five Personality theory's Five Factor Model to quantitatively simulate the personality traits of agents. For each agent participating in the game, we generate and assign a random personality profile that remains consistent throughout the game. Before each LLM query, we describe the agent according to this profile, thereby shaping the agent to embody the specific personality traits.

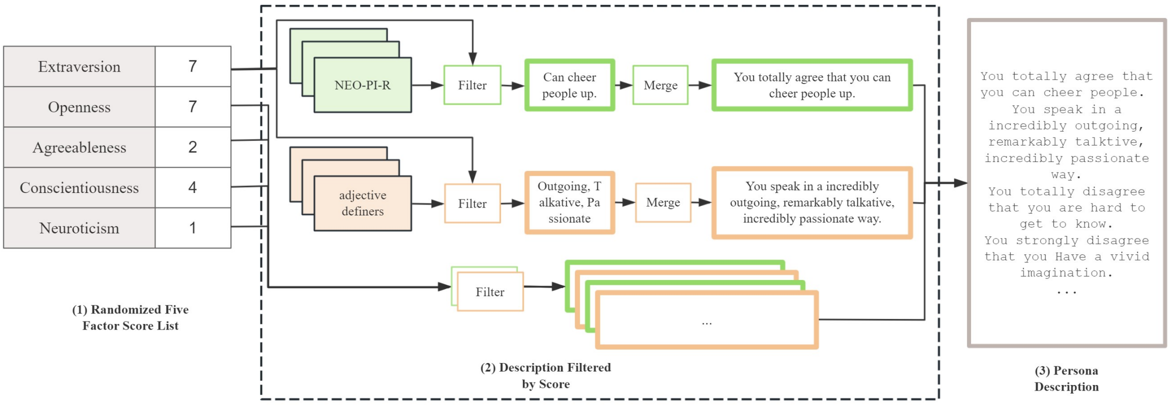
We first quantify the five factors of the Big Five Personality model: Extraversion, Openness, Agreeableness, Conscientiousness, and Neuroticism, using a scale from 1 to 7, where 1 represents a weak manifestation of the trait and 7 a strong one. From the NEO-PI-R, we can find positive and negative statements for each factor. We extract these statments and augment them with adverbs indicating the degree of agreement, such as 'totally agree,' 'slightly disagree,' etc., to create quantified evaluations for the different factors. Adjective descriptions for the various factors, summarized by Costa and McCrae, have also been carefully collected and organized into a factor-adjective dictionary. We use the pattern "You speak in a [adjective] way" to constrain the linguistic behavior of the agents, which has proven to be the most effective among the various descriptive approaches we've tried. Through multiple experiments, we found that when the trait value is closer to moderate, i.e., around 4, the description of that trait becomes less distinct. Conversely, when the trait value is closer to the extremes, such as 1 or 7, the description becomes more pronounced, and the agent exhibits a stronger personality. Therefore, in our framework, we use a random function to select which traits are described more frequentlythe closer a trait value is to 1 or 7, the more likely it is to be described. The framework of our method is shown in Figure 3

We suppose the agents are numbered by i. Each agent has a persona represented by a list of big five factor scores B_i. The five factor is represented by a score b_k, in the range of 1 to 7, where k means the index of factor. Then we can have:

$$B_i = \{b_k\}_{k=1}^{5} \tag{1}$$

In the process of descritpion generation, we turn the set of five factor scores into a piece of persona descriptions. We define the function $F_D(b_k)$ to determine the degree statement according to the factor score b_k. Firstly, the statements from NEO-PI-R are extracted based on th the factor k through $F_N(k)$. While the adjective definition is generated from $F_A(k)$. Both $F_N(k)$ and $F_A(k)$ will randomly select one descritpion from the dataset. We will define a resampled counter function $F_R(b_k)$ to determine the number of times different factor scores will be resampled. The closer a factor score is to 1 or 7, indicating that the factor is more pronounced, the higher the number of resamples returned by the sampling function will be. Finally, we use a merge function $F_M(n, a, r, f)$, which can complete the descritpion content according to the factor score b_k, to obtain the final B5P statement. Here we have:

$$F_D(b_k) = F_M(\{F_N(k), F_A(k)\}_{t=1}^{F_R(b_k)}, b_k) \tag{2}$$



Extraversion
7
Openness
7
Agreeableness
2
Conscientiousness
4
Neuroticism
1
(1) Randomized Five Factor Score List
NEO-PI-R
Filter
Can cheer people up.
Merge
You totally agree that you can cheer people up.
adjective definers
Filter
Outgoing, T alkative, Pa ssionate
Merge
You speak in a incredibly outgoing, remarkably talkative, incredibly passionate way.
Filter
...
(2) Description Filtered by Score
You totally agree that you can cheer people.
You speak in a incredibly outgoing, remarkably talktive, incredibly passionate way.
You totally disagree that you are hard to get to know.
You strongly disagree that you Have a vivid imagination.
...
(3) Persona Description

Figure 3: The framework of how to generate B5P statement. (1) Randomized Big Five factor Score Set: a set of five random number range from 1-7, representing the five factor in FFM. (2) Description Filtered by Score: Select the statements from NEO-PI-R and adjective definders and combine them in a readable way according to the score list (3) B5P Statement: combine all five factors' descritpion together and form a final B5P statement.

So for each agent, we are able to produce the B5P statement $\mathcal{B}_i$:

$$F_D(B_i) = \bigcup_{k=1}^{5} F_M(\{F_N(k), F_A(k)\}_{t=1}^{F_R(b_k)}, b_k) \tag{3}$$

$$\mathcal{B}_i \leftarrow F_D(B_i) \tag{4}$$

The B5P statement $\mathcal{B}_i$ will be generated and stored at the begining of the game loop. The way to use it is sent the B5P statement as system prompt first and then send the task prompt, which can initialize a characterized agent before the core task prompt is solved. The B5P statement is used in Faver Dynamics System, Startegy Decision System and Text Generation System. The LLM request mentioned in the following article will, by default, send the B5P statement $\mathcal{B}_i$ to the LLM as system prompt first and then the specific task prompt. A more detailed prompt generation process will be introduced in Appendix B.

3.3 Favor Dynamics

Issue For an open architecture with multiple agents, each agent is equivalent relative to the others. Due to this equivalence, a LLM response would equally consider and respond to each other agent's input, which means that the strategies employed by one agent towards another will have some degree of repetitiveness. For example, if in a previous statement, a player declared to be a Seer and pointed out a player as a Werewolf, the LLM's response would include the accused player being labeled as a Werewolf. This clearly deviates from our everyday intuition, as human judgment is influenced by various factors such as the trustworthiness of the player making the claim and the credibility of the accused player's behavior. Indeed, a substantial body of psychological research (Kahneman and Tversky, 2013; Kahneman, 1982; Simon, 1991; Luhmann, 2018; Cho et al., 2015) indicates that human decision-making is not purely rational but is affected by numerous non-rational factors.

Method We introduce the concept of "Favor". The value of "Favor" is a quantifiable representation of an emotional state toward other agents, and it significantly influences an agent's decision-making strategy regarding other agents. In Kahneman's theory (Daniel, 2017), the decision-making system consists of intuitive, fast System I and complex, rational, slow System II. Booch's team also discussed the application of this theory to AI (Booch et al., 2021). We simplify this rational, slower system II into a delieratedly calculated one-dimensional scalar, and before every significant decision, we update the agent's favor scalar for other agents using the LLM and convey the updated affinity through prompts in text form.

In our method, each agent i maintains a set of float denoted as $\left\{F_{ij}^{(T)}\right\}$, representing the Favor level for each other agent j in rount T, where each number in the array has a value range of (1,100). Duraing each turn T, the agent i organize the dialog histroy of last turn $H_i^{(T-1)}$ to obtain a Favor change list $H_i^{(T-1)}$ from LLM (see Equation 5), denoted as $\left\{f_{ji}^{(T)}\right\}$, which represents the variation of Favor for other agent j in rount T, where each number in the array has a value range of (1,10) .

$$\left\{f_{ij}^{(T)}\right\} \leftarrow LLM_{favor}(H_i^{(T-1)}) \quad (5)$$

The favor level set will be then updated according to the favor variation, the change value will be remapped into a larger range (see Equation 6). In our study, we will remap the variation score from (1,10) to (-30,30).

$$F_{ij}^{(T)} = F_{ij}^{(T-1)} + F_{remap}(f_{ij}^{(T)}) \quad (6)$$

At the output end, the system will generate a favor statement for agent i as $\mathcal{F}_i^{(T)}$, which describes the degree of favor agent i has toward all another agents. The affinity level F_{ij} is mapped into one of five intervals, and a corresponding attitude phrase is retrieved for that interval. This phrase is then concatenated with the name of the agent being described and returned, thereby generating an affinity statement. A pseudo-code of our algorithm is listed in Algorithm Figure 2. Both the full introduction of the algorithm and the detail of the LLM prompt can be found in Appendix C

3.4 Strategy Decision

Issue While large language models can generate reasonable behaviors based on contextual information (Dillion et al., 2023; Horton, 2023; Park et al., 2022), an overly long context input can turn the agent behaviors back to lack of continuity and feasibility. Additionally, certain decision-making behaviors in games with incomplete information, such as cooperation, reasoning, deceit, and covert collaboration, are difficult for LLM to generate spontaneously, which can detract from the game's fun and diversity. For example, when you provide an agent with a complete recount of other players' dialogue, the LLM often generates a summary of the conversation rather than making emotional statements or inflammatory lies.

Method Strategy is a guiding mechanism for the agent to filter content for output, ensuring more purposeful and directed communication. Inspired by Chain of Thought (CoT) (Wei et al., 2022) and the Planning and Reacting modules in Generative Agents (Park et al., 2023), we designed a system for generating action instruction in two steps: Judgement and Decision-making (see Algorithm 1). In the Reasoning part, the LLM infers probabilities of other agents' identities, estimating how likely it is that other players have certain roles, which is stored as a probability distribution. In the Decision-making part, the LLM evaluates the current situation based on these inferences, taking into account its own role, and selects a corresponding output strategy, outputting it in a CoT format.

The following two steps show how the strategy is decided by our agent.

Judgement: a phase where the Large Language Model (LLM) performs probabilistic reasoning about the roles of the agents in the field. Here, we will collect historical dialogue information from the agents, denoted mathematically $H_i^{(T-1)}$. Additionally, the favor statement generate by the Favor Dynamics System $\mathcal{F}_i^{(T)}$is also involved as an input (see section 3.3 for how to generate $\mathcal{F}_i$). The information mentioned above are collected to generate a prompt, which will be sent to the LLM, with instructions for it to return the results in a fixed format (see Appendix D for the detail of the prompt). The response from LLM will be summarized into a two-dimensional matrix $\{p_{jr}\}$, representing from agent i's perspective, the probability that agent j is role r.

$$\left\{p_{jr}^{(T)}\right\} \leftarrow LLM_{judgement}(H_i^{(T-1)}, \mathcal{F}_i^{(T)}) \quad (7)$$

Strategy: In this step, the LLM conducts a Chain-of-Thought (CoT) derivation for the action strategy $\mathcal{S}_i^{(T)}$ for the agent i in turn T. It filters and selects the maximum values from the role probabilities deduced during the Judgement phase, tagging them as friend or opponent to provide content for the LLM prompts. Here we denote this function as $F_P(p_{jr}^{(T)})$, in which r is the index of role, j is index of agents and P_{jr} is the posibility of most possible role that agent j is. r_max is the role that agent j has the highest posibility. And $\mathcal{R}i^{(T)}$ is the statement of the role possibility of all other agents, in string format. Additionally, for each role, we provide some predefined strategies based on experience, such as choosing to conceal your identity if you are a Werewolf, or pretending to be a Werewolf to mislead other players if you are a Minion. We denote these predefined role strategies as $\mathcal{S}_r^*$. As shown below, a prompt with $\mathcal{R}i, H_i, \mathcal{S}^*(r)$ will be sent to LLM, and the response will be stored as the action stratgy $\mathcal{S}_i^{(T)}$. Algorithm 1 shows a detailed algorithm of our strategy decision process. And the related prompt desgin is listed in Appendix D.

$$R_{ij}^{(T)} = F_P(p_{jr}^{(T)} | r = r_{max}) \tag{8}$$

$$\mathcal{R}i^{(T)} = \bigcup_j R_{ij}^{(T)} \tag{9}$$

$$\mathcal{S}_i^{(T)} \leftarrow LLM_{strategy}(\mathcal{R}i^{(T)}, H_i^{(T-1)}, \mathcal{S}^*(r)) \tag{10}$$

3.5 Memory

For the handling of memory streams, we refer to the solution proposed by Generative Agent (Park et al., 2023). For each utterance from a single agent, it is broadcast to all agents and storaged as a memory event object that includes a natural language description, a timestamp in term of speaking round and an importance score. When recalling historical memories, they are sorted based on the priority of the memory, which is composed of recency, importance, and fondness. The top k segments of memorable dialogs are then selected and sent to the large language model.

Below are specific explanations for these three factors that influence priority.

Recency produce higher scores to memories that were accessed recently, so talks that are presented shortly before are likely still within the agents's attention span. In our implementation, we treat recency as an exponential decay function based on the number of talk turn since the memory was last access. We set the decay factor to 0.95.

Importance assigns higher scores to utterances that the agent deems significant. In our study, we define importance as the agent's affinity towards the speaking agent, which equates to a record of the agents historical level of favor.

Fondness reflects the degree of affinity the agent has for the current speaking agent, with a higher affinity resulting in greater attention. The fondness score is provided through real-time calculations by the Favor system.

To access the final relavent score, we remap the recency, importance and fondness score to the range of [0,1]. The memories are evaluated by the retrieval function that combine the three scores with weight: relevant score = $score_{relevant} = \alpha_{recency} \cdot recency + \alpha_{importance} \cdot importance + \alpha_{fondness} \cdot fondness$

In our research, we set the weights as $[\alpha_{recency}, \alpha_{importance}, \alpha_{fondness}] = [0.5, 2, 2]$. The memories are sort according to their relavent scores, and the top-k memories are selected to described in the prompt.

3.6 Methods of Evaluation

Engagement, novelty, and unexpected shift are among several factors that can determine the excitement level of a game. However, how to evaluate "fun" is a mathematical challenge. We need a mechanism to ground these conceptual subjective factors into comparable and analyzable digital infomation for futher objective observation. In this section, to numerically evaluate the consequence of a game, we propose various methods, including Text Distance, Judgement Variation and EWAVM Evaluation.

3.6.1 Text Distance

In the field of Natural Language Processing(NLP), word embeddings (Turian et al., 2010; Mikolov, 2013), known as training in an unsupervised manner to distributionally represent context, become very useful features in many applications such as text similarity compution and semantic search. In general, it has been found to be helpful to convert the text into vectors, which can be the subject of mathematical operations(e.g. addition, subtraction, length measures, etc) and lend themselves well to be introduced in many Artificial Intelligence(AI) algorithm and strategy. A text embedding algorithm has been opened and provided by OpenAI (Neelakantan et al., 2022). While the embedding model 'text-embedding-3-small' is used in our research. The full introduction of our algorithm is listed in Appendix E.

The statistical result of the calculated text distance infers the variation in content scale of the textual information flow being examined, where a larger mean and standard deviation indicates a greater richness of the content. As the distance value is generated from the vectorizational text blocks in the information flow, the larger the calculated distances, and the more dispersed their distribution, the better they reflect that the texts are more widely and evenly distributed in the vector space, which indicates a higher degree of diversity and richness in their textual content.

3.6.2 Judgement Variation

In the werewolf-like game, the game often ends with a 'voting' phase. During the voting phase, each player selects a player who they believe most resembles the "werewolf" and casts a vote for

Algorithm 1: Pseudo-code for Strategy Decision System

```
Data: historical dialogs $H_i^{(T-1)}$, favor statement $\mathcal{S}_i^{(T)}$
Result: Action Strategy Statement
Get the agent-role posibility matrix $\left\{p_{jr}^{(T)}\right\}$ from LLM request $LLM_{judgement}$ based on the
  historical dialog input $(H_i^{(T-1)})$ and favor statement $\mathcal{S}_i^{(T)}$ according to equation 7 ;
Init the role posibility statement $\mathcal{R}i$ ;
for $j \in Agents$ do
    $r_{max} \leftarrow ”Villager”$
    for $r \in Roles$ do
        if $p_{jr} > p_{jr_{max}}$ then
            $r_{max} \leftarrow r$
    Get the role posibility statement $R_{ij}^{(T)}$ of the max possible role $r_{max}$ for agent $j$ with the use
      of posibility conversation function $F_P(p_{jr})$ according to equation 8
    Append the role possibility statement $R_{ij}^{(T)}$ of agent $j$ to the full role posibility statement $\mathcal{R}i$,
      see equation 9
Generate the action startegy $\mathcal{S}_i^{(T)}$ of this turn with LLM request, based on equation 10
Return $\mathcal{S}_i^{(T)}$
```

them. The player with the most votes will be eliminated. The voting result of each player represents their judgment of who they think the werewolf player is in this round of the game. In the Judgement Variation evaluation system, we will conduct a secret vote after everyone has spoken in each round, where each agent secretly identifies the werewolf player they believe in. A similar voting process can be found in the research on LLM Werewolf (Jin et al., 2024; Xu et al., 2023). However, we do not verify the accuracy of the werewolf vote, but instead examine the diversity of the voting results. The voting results reflect the different LLM agents' perceptions of other players based on their roles and personalities during the game process. Therefore, the diversity of the voting results can also reflect the richness, twists, and surprises of the game process to a certain extent.

According to the algorithm we designed (for detailed introduction, see Appendix F), the final calculated voting standard deviation will numerically reflect the diversity of the game process. The smaller the standard diversity is, the more evenly the votes are distributed among the various roles, indicating that the interactions between the roles during the game are more diversified, thereby ensuring the engagement of the game. For an intuitive understanding, we define the judgement variation as the reciprocal of the variance value calculated. In fact, if the votes are more likely to be concentrated on a particular role, such as the "Werewolf," it suggests that the "Werewolf" is more likely to reveal itself, which makes the direction of the game to be predictable and dull.

3.6.3 EWAVM Evaluation

It happens to be a trendency to allowing the LLM itself to provide a calculatable evaluation of the text-based material (Shao et al., 2023; Wang et al., 2024). When estimating ambiguous meanings, large language models (LLMs) appear more appealing than conventional statistical methods, which has led an increasing number of scholars to recognize their value in data analysis. For instance, when evaluating the level of engagement in a piece of text, traditional numerical analysis methods may find it challenging to quantify a relatively subjective concept of "fun". In contrast, by querying an LLM, it provides a numerical score after a series of reasoning and analysis, such as a rating on a scale from 1 to 10, depending on how the researcher structures the prompt. Of course, the issue of the uncertainty from the LLM response exists, and different LLMs may have varying tastes in evaluating the same piece of text based on their model performance. However, after scaling up the size of dataset and limiting the number of model of LLM used to qualitatively evaluate different experimental text samples, these errors could be diminished.

We ask GPT-4o-mini to rate on five main dimensions and sum up an average score to exhibit the capebility of our framework's performance. Furthermore, we annotate the generated dialogs in the following five perspectives:

- **Engagement**: The ability to produce interesting dialog content in rich use of language, to generate unrepeatable expression based on different role in the game, and to exhibit different personality through the conversation.
- **Wonder**: The power of generating game scenario contains unexpected twists, unpredictable behavior from players, and uncertainty for each players' role.
- **Attraction**: The capability to attract players, such as assessing whether players would want to know the game's ending, whether they have the desire to continue watching the game.

The above three are simple estimated dimensions, involving sending several requests to the LLM model and obtaining multiple sets of evaluation scores, which are then averaged to derive the evaluation score for each dimension(Appendix G.1).

The following two dimensions will be calculated using more complex algorithms:

- **Variation**: The ability to create in-game differentiation, that is, to what extent a game can allow an agent to exhibit different judgments regarding the roles of other agents. This dimension can also effectively evaluate whether the game has inter-turn diversity. We send requests to the LLM to obtain each player's judgment of the identities of other players in each round of the game(see Appendix G.2). This differs from the request for role judgment in Strategy Decision, as we base our judgment on the dialogue content submitted by the agent at the end. Subsequently, we will create a vector array of player identifications for each round based on the players' identifications, and calculate the distance between different decision arrays. Finally, we will statistically consider the calculation results and provide a score for Variation.
- **Memorability**(diversity of highlight moment): The ability to allow each of the player make valuable speech in the game, which means not only the role with ability like Seer or special infomation like Tanner, but also the role with limited power like the Villager can perform outstandingly in the game. In our algorithm(see Appendix G.3), the top k(k=10) most impactful statements in the text and their rating will be generated by the LLM. Then we can identify whether these statements are evenly distributed among different roles. It primarily evaluates the diversity of players' statements within the game, as different roles obtain different information according to the game system, which affects the importance of each player's statements. If the underlying language generation logic of the considered LLM framework is rich, it can mitigate the impact of role differences on the quality of statements, making the distribution of impactful statements more even.

4 Experiments

4.1 B5P Generation System Analysis

Our proxy generates personality trait descriptions for the LLM agent using the method described in section 3.2. Here, we hope to examine the impact of the Big Five Personality Generation System (B5PGS) on dialogue generation through a smaller-scale experiment. We will pre-generate several personalities for the agents (in our experiment, there are 4 agents, including a baseline without any personality description), which should cover the extreme values of the five different personality factors as much as possible. Then, based on the same role (here we choose the 'Villager' because it has the least amount of information), we assign the same tasks to the agents without providing prior dialogue information (the player is the first to speak). We hope that the dialogue generated by B5PGS will be more diverse, so we use the Text Distance introduced in section 3.6.1 to check the dialogue text distance and combine case studies to analyze the generated dialogue conten(The specific experimental steps can be found in Appendix H).

We conducted 50 rounds of generation for each agent, meaning that each agent produced 50 segments of dialogue. We analyzed the text distance of these dialogues, and the results are shown in Ta-

Agent	Five Factors	Text Distance(ave.)	Text Distance(std.)
Alpha	-	0.174	0.031
Beta	A7E1N2O4C1	0.194	0.049
Gamma	A4E7N7O6C3	0.183	0.051
Delta	A5E5N5O1C6	0.199	0.039

Table 1: the result of text distance in the conversations created by the four agents with different personas

Alpha	right, everyone, lets get this discussion started. We need to figure out who among us is the Werewolf. Remember, we cant trust anyone blindly. Let's share any suspicions or observations we've made. I think it's crucial we pay attention to the way each person responds and interacts. If anyone acted a bit off last night or seemed too eager to shift blame, let's hear it. Stay vigilant, because if we vote wrong, we're handing victory to the Werewolf. Who wants to share their thoughts first?
Beta	Um, hey everyone. I guess we should probably figure out who the werewolf is, right? Um, Im just a villager, so I dont have any special powers or anything. Its all a bit confusing, but, like, I really want to help. Maybe we can look at whos acting a bit odd? But, um, I might not always see things clearly, you know? I just hope we can talk and kind of sort things out together. Seems important to keep an eye out for any clues or, um, behaviors that dont add up. Just, like, letting you know Im here to chat and stuff. Lets try our best!
Gamma	Oh wow, what an intriguing game we have here! I must say, I can feel the tension rising as we try to figure out who among us is the sneaky Werewolf. Each whisper, each glance, sends shivers down my spine! But, alas, we must keep our wits about us! I cant help but wonder if the wolf is hiding among us, lurking in the shadows! We must dissect every word for clues, my dear friends! Im absolutely itching to uncover the truth. Let's unite, share our thoughts, and devise a plan to smoke out the fiend! Who's with me against the darkness? Let's shine our light together!
Delta	Hello everyone. As a Villager, I know my role is to help identify the Werewolf among us. While I may not have any special abilities, I believe observation and careful discussion can lead us to the truth. Let's focus on what each of us has to say and watch for any suspicious behavior. It's important we work together as a team and not jump to conclusions too quickly. Id like to hear everyones thoughts and see if we can find some clues about who the Werewolf might be. Remember, voting wisely is crucial for our success.

Table 2: An example of the output of dialog generated by agents with different personalities (1) Alpha: the baseline (2) Beta: a soft-hearted, sober and aimless agent (3) Gamma: a talktive, self-pitying, negligent agent (4) Delta: an uncurious, uncreative, hardworking agent.

ble 1. It can be verfied that, overall, the system with B5PGS for dialogue description is obviously better than the baseline (Alpha), that is, the average value and variance of the distance are both significantly improved compared to the baseline. This indicates that after adding personality descriptions to the agents, the richness of the generated language expression has greatly increased.

At the same time, we created a word cloud of the results output by the four agents(Figure 4). It can be found that there is a large gap between the high-frequency words used by the four agents, and the agents with B5PGS modifications have clearly produced expressions with their own emotional colors. We noticed that Alpha, as the baseline agent, owns a lot of content about the game's mechanics, with the most frequent word "werewolf" representing that it has a lot of content about finding the werewolf. In contrast, the agents with B5PGS modifications did not overly emphasize "werewolf." Taking Beta as an example, due to its high score of 7 in "Agreeableness" among the five dimensions, and only 1 in "Extraversion" and "Conscientousness," the agent's expression is described as "You speak in an incredibly Trusting, remarkably Lenient, incredibly Soft-hearted way. You speak in an incredibly Reserved, incredibly Unfeeling, extremely Sober way. You speak in an incredibly Lazy, remarkably Disorganized, incredibly Aimless way." Therefore, we can see that in its high-frequency words, there are words like "um," "guess," "know" that indicate introverted, uncertain, and hesitant expressions, which are in line with our imagination of Beta through this description. Table 2 lists the four actual dialogues generated by the four agents.

4.2 Full Game Study

4.2.1 Setup

Let's take a look at the performance of our proposed model throughout the full game process. Five different strategic methods(see Table 3) are considered in the comparative experiment to comprehensively inspect each part of our framework.

everyone
Werewolf
share
need
Let
thoughts
among
guess
um
know
Maybe
really
Let
Oh
together
Werewolf
everyone
keeps
Let
thoughts
Werewolf

Alpha Beta Gamma Delta

Figure 4: The wordcloud of generated content from four agents in different personalities.

Baseline: No additional strategies are used.

- **B5P**: The Big Five Personality Generation System(B5PGS) is enabled, which means that the agents' personalized descriptions will be added to the LLM requests(see section 3.2).
- **Favor**: Both the B5PGS and Favor Dynamics System(FDS) are enabled. This means that the LLM agent not only has emotional expressions but also has preferences for different players(see section 3.3).
- **Strategy**: The Strategy Decision System(SDS) is enabled, which means that the LLM agent will sort out the decision chain before speaking.(see section 3.4).
- **Full**: The B5PGS, FDS, and SDS are all enabled, meaning that all additional features of the entire framework are activated, offering the most comprehensive preprocessing process.

In the testing, each of the aforementioned strategy runs 100 times. The number of participated agents is 8. Each game consists of 3 rounds of statements by each agent, resulting in a total of 24 dialogs per game. Following each round, there will be a vote that does not affect the statements, totaling 3 votes per game. The vote is used for data collection and does not affect the agents' memory or decision making. The Large Language Model (LLM) is gpt-4o-mini (The model name in OpenAI API is 'gpt-4o'.). In the LLM Evaluation phase, each evaluation mode is conducted 5 times.

4.2.2 Experiment of Text Distance

We compute the Text Distance of the output results for the various strategies described in Section 4.2.1 to compare the diversity of textual information within them, as shown in Table 4. We separately tally the content of the three rounds of statements, and the data from each round is mixed with the previous data, so the mean and standard deviation of the text distance decrease with the accumulation of data. The introduction of B5P descriptions greatly enriches the diversity of the LLM output, with a substantial improvement in the performance of text distance compared to the baseline. At the same time, the use of both Favor and Strategy tactics leads to a loss in text distance, as under these strategies, the LLM agent's statements become more targeted, providing feedback based on preferences or logic to other agents. Particularly for the Favor method, we observe that after incorporating evaluations of other players' preferences, the richness of the LLM system's responses decreases significantly. Our final strategy method, Full, reconciles these systems, maintaining the richness of the agents' statements while showing a certain degree of strategic tendency, and achieving better results than the baseline in the Text Distance test.

-	Baseline	B5P	Favor	Strategy	Full
B5P Generation		×			×
Favor Dynamics			×		×
Strategy Decision				×	×

Table 3: The System in each of the methods in the experiment. × for the system is included in the method

5 Experiment on Judgement Variation

In the experiment, after a round of discussion, we let each LLM agent cast a vote for who they think is a werewolf. Each game includes three rounds of voting. Based on a total of 300 rounds of voting for each method, we tallied the number of times each role was voted for and calculated the judgement variation (refer to Table 5 and Table 6 for specific data). It can be seen that, compared to the baseline, the Favor framework

-	Baseline	B5P	Favor	Strategy	Full
1st Round					
average distance	0.231	0.333	0.201	0.227	0.291
standard deviation	0.331	0.331	0.200	0.225	0.289
2nd Round					
average distance	0.223	0.315	0.189	0.217	0.279
standard deviation	0.331	0.313	0.188	0.216	0.276
3rd Round					
average distance	0.221	0.306	0.185	0.213	0.273
standard deviation	0.219	0.304	0.184	0.211	0.271

Table 4: The result of text distances in three round among the five different methods. The dialog data used in 2nd round includes the 1st round and the data in 3rd round includes the previous two round. The text distance declines as the size of data set increase. And the 'B5P' strategy performs the best in the text distance test, while the 'Full' strategy outperform the baseline.

-	Baseline	B5P	Favor	Strategy	Full
Judgement Variation	13.521	11.584	32.258	11.178	15.212

Table 5: The judgement variation of different methods. The Favor strategy shows a strong dominance in this test. The performance of 'Full' strategy is far better than the baseline.

achieved the greatest improvement in judgement variation scores, while the other frameworks also showed significant score increases. This means that introducing each of our designed systems individually into the LLM decision-making system enhances its decision richness, leading to more ambiguous voting decision by the agents about different roles during the game. The most significant score improvement was for the Favor system, indicating that adding preferences for other agents in the decision-making process results in greater richness and randomness in the LLM agent's decisions.

Fig 5 shows the number of votes received by each role, using radar charts to compare each method with the baseline. It can be observed that in the baseline, the Seer and Mason received very few votes, while the Minion, Tanner, and Werewolf, representing the "bad" camp, received many votes. In the other methods, the "bad" camp still received a high number of votes, but the gap between them and the "villager" camp was not as large. From the comparison of the radar charts, it is clear that the voting distributions for Favor and Full were relatively even, while Baseline and B5P showed extreme situations with a high number of votes for Tanner and very few for Seer.

If we look at the final win-loss results Tanner wins by being voted out, the werewolves lose if they are voted out, and otherwise, the werewolves win we obtain the win rate statistics shown in Table 7. An interesting phenomenon is that, except for the Strategy method, all other methods increased the werewolves' win rate. Among them, the Favor method increased the werewolves' win rate the most, while Full increased it by around 10%. Additionally, we can observe that the B5P and Favor method led to a significant decrease in the villagers' win rate. The Strategy method transferred some of Tanner's win rate to the werewolf camp. Therefore, we can conclude qualitatively that in our specific designed asymmetric information games, more "human-like" elements, such as character traits and preferences, tend to increase the win rate of the side with more hidden information (the werewolf side).

6 Experiment of EWAVM Evaluation

We evaluate the generated dialogue results based on the EWAVM Evaluation method described in section 3.6.3 We ran each of the five different methods 100 times and collected feedback scores five times, then calculated the average values for statistical analysis (see Table 8 and Figure 6).

It can be seen that in terms of Engagement, B5P achieved the highest score, while Strategy and Combination performs better than the baseline and Favor is worse than the baseline. It shows that the B5PGS can largely benifit the output of interesting and funny expression in the text. Meanwhile, Strat-

-	Baseline	B5P	Favor	Strategy	Full
Seer	5.51%	4.62%	7.57%	6.36%	5.76%
Villager1	8.60%	10.24%	11.73%	6.07%	8.03%
Villager2	8.60%	10.24%	11.73%	6.07%	8.03%
Mason1	5.56%	5.64%	8.76%	4.27%	5.60%
Mason2	5.56%	5.64%	8.76%	4.27%	5.60%
Minion	12.88%	11.53%	12.39%	15.90%	20.11%
Tanner	25.67%	33.20%	17.92%	30.23%	19.79%
Werewolf	22.10%	14.27%	13.56%	20.46%	21.33%

Table 6: The table of the percentage of votes received by different roles in different methods.

-	Random	Baseline	B5P	Favor	Strategy	Full
Villager	12.5%	22.1%	14.26% ↓	13.56% ↓	20.46%	21.33%
Werewolf	75%	52.26%	52.53%	68.51% ↑	49.30%	58.88% ↑
Tanner	12.5%	25.67%	33.20% ↑	17.92% ↓	30.23% ↑	19.79% ↓

Table 7: The win rate for the Villager-team, Werewolf-team and Tanner-team. The ’Random’ means the vote rate if all the agents vote randomly. Each arrow indicates at least a 5% difference from the baseline and so on. Up arrow means increase, and down arrow means decrease.

egy obtained the highest scores in Wonder, Attraction, and Memorability evaluations, which make contribution to the good performance of the Combination in these area. The CoT method can help to make the game to become more surprise and attractive.

When considering the integrated system, the Combination method, we observe that it not only received the highest rating in Variation but also scored very close to the maximum in all other categories, which means our system not only leverages the strengths of its subsystems but also creates a synergistic effect among them, resulting in a high score for Variation. Therefore, based on the EWAVM evaluation results, the Combination system effectively enhances the diversity of agent dialogue generation in the our special ONUW game.

7 Discussion

7.1 Application of Humanized Agents

The Humanized Agent possesses broad application potential beyond the werewolf-like social game demonstrated in this work, particularly in areas that could benefit from modeling human behavior. For example, in the field of psychology, the introduction of personality types into LLM agents has created unprecedented opportunities for analyzing and generating language data on a massive scale (Demszky et al., 2023). Meanwhile, in sociology, the diversity of intelligent agents can aid in the establishment of social prototypes (Park et al., 2022). Additionally, diverse intelligent agents are also significant for the construction of Virtual Reality environments (Wan et al., 2024).

7.2 Future Work and Limitations

In this work, we have implemented the initial framework and subsystems of the Humanized Agent and conducted preliminary evaluations. Future experiments can build upon this system by introducing additional subsystems to increase its richness. Furthermore, the stability of the system is also an issue worth investigating. How to make the system more efficiently and stably generate high-quality content without causing systemic crashes while reducing the hard coding of related prompts is something that warrants further research. Finally, during the course of studying this issue, we observed that the gaming system based on LLM agents exhibited diversity distinct from traditional strategic models, which might be an inspiration for the study of social gaming problems.

Although the Humanized Agent has shown great potential in the production of non-repetitive content, the means to measure the richness of its content are still lacking. Although this paper proposes three numerical testing methods, the stability of their tests still deserves more evaluation, and they fall short of standardizing the measurement of textual content richness. Future research may need to explore more in terms of the stability of

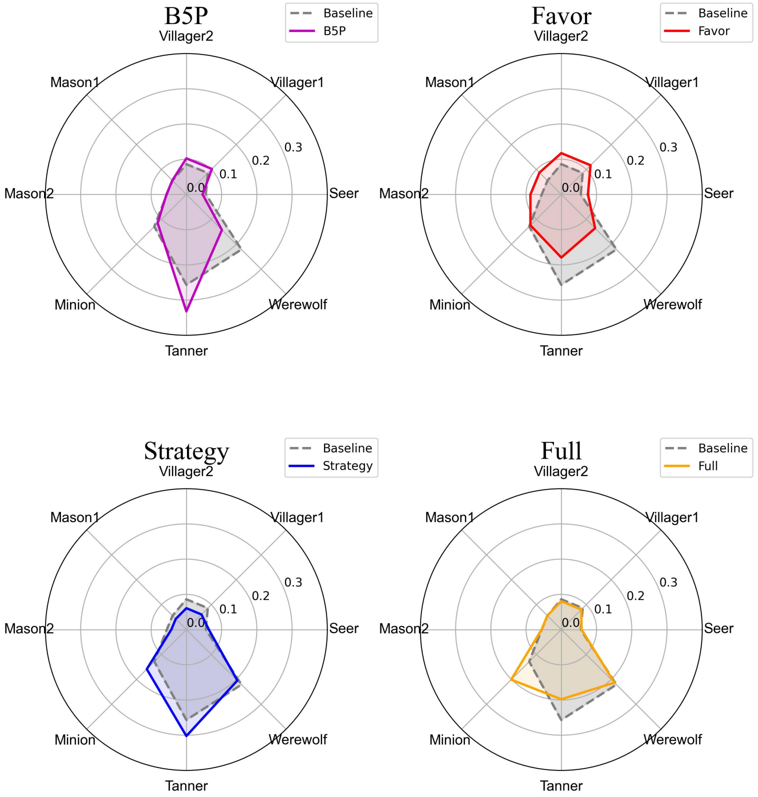


Figure 5: The radar charts of the count of vote toward different roles from different methods comparing to the baseline.

-	Baseline	B5P	Favor	Strategy	Full
Engagement	8.38	8.83	7.41	8.69	8.73
Wonder	6.57	7.50	5.78	8.05	7.72
Attraction	9.11	9.14	9.10	9.16	9.07
Memorability	10.26	10.56	10.29	16.36	10.99
Variation	11.64	11.14	13.36	12.18	13.00

Table 8: The result of EWAVM evaluation of different methods. The highest score of each factor is underlined.

LLM assessments.

8 Conclusion

In this paper, we have designed a social gaming framework based on LLM (Large Language Model) agents named Humanized Agent. This framework integrates three subsystems to achieve diversified descriptions of agent personalities, dynamic preferences between agents, and strategic reasoning analysis based on the current game state. The implementation of these features allows LLM agents based on this framework to generate richer and more varied interactive content.

Additionally, we propose three tools for detecting the diversity of content generated by LLM agents, and we tested our framework in an experimental environment using a variant of the game "One Night Ultimate Werewolf." Test results indicate that our composite framework is capable of producing richer interactive content. For future work, we aim to enhance our model to provide more robust tools for generating interactive content.

Acknowledgements

This research was supported in part by National Science and Technology Major Project (2021ZD0111501), National Science Fund for Excellent Young Scholars (62122022), Natural Science Foundation of China (62206064, 62206061).

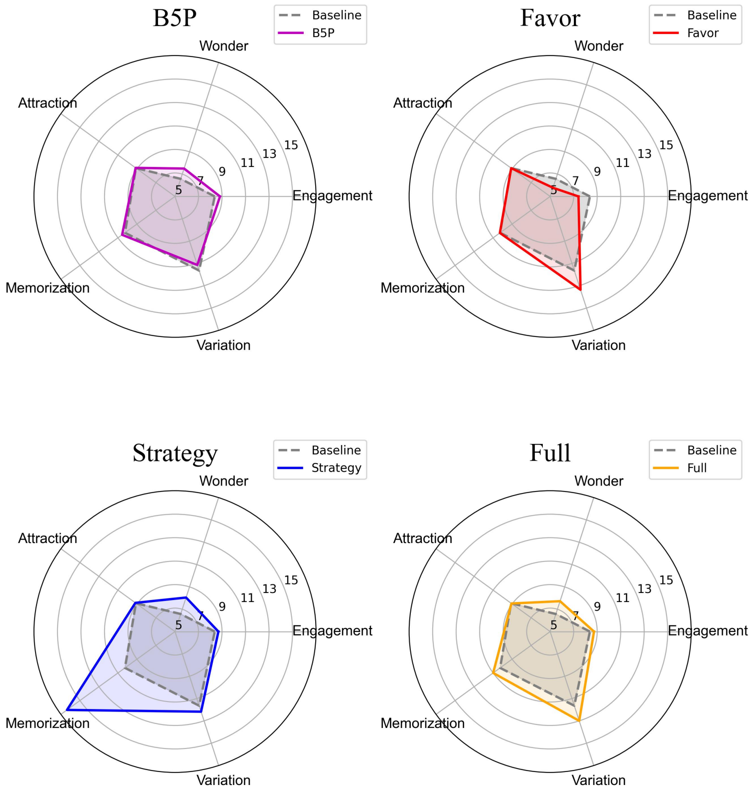


Figure 6: The EWAVM evaluation of different methods with a comparison to the baseline.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Gati V Aher, Rosa I Arriaga, and Adam Tauman Kalai. 2023. Using large language models to simulate multiple humans and replicate human subject studies. In *International Conference on Machine Learning*, pages 337–371. PMLR.

Michael Ahn, Anthony Brohan, Noah Brown, Yevgen Chebotar, Omar Cortes, Byron David, Chelsea Finn, Chuyuan Fu, Keerthana Gopalakrishnan, Karol Hausman, et al. 2022. Do as i can, not as i say: Grounding language in robotic affordances. *arXiv preprint arXiv:2204.01691*.

Xiaoheng Bi and Tetsuro Tanaka. 2016. Human-side strategies in the werewolf game against the stealth werewolf strategy. In *International Conference on Computers and Games*, pages 93–102. Springer.

Grady Booch, Francesco Fabiano, Lior Horesh, Kiran Kate, Jonathan Lenchner, Nick Linck, Andreas Loreggia, Keerthiram Murgesan, Nicholas Mattei, Francesca Rossi, et al. 2021. Thinking fast and slow in ai. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 15042–15046.

Graham Caron and Shashank Srivastava. 2022. Identifying and manipulating the personality traits of language models. *arXiv preprint arXiv:2212.10276*.

Raymond Bernard Cattell, Herbert W Eber, and Maurice M Tatsuoka. 1992. Handbook for the sixteen personality factor questionnaire (16 pf). *(No Title)*.

Jin-Hee Cho, Kevin Chan, and Sibel Adali. 2015. A survey on trust modeling. *ACM Computing Surveys (CSUR)*, 48(2):1–40.

Paul T Costa and Robert R McCrae. 2008. The revised neo personality inventory (neo-pi-r). *The SAGE handbook of personality theory and assessment*, 2(2):179–198.

PT Costa and RR McCrae. 1999. A five-factor theory of personality. *Handbook of personality: Theory and research*, 2(01):1999.

Paul T Costa Jr and Robert R McCrae. 1976. Age differences in personality structure: A cluster analytic approach. *Journal of gerontology*, 31(5):564–570.

Kahneman Daniel. 2017. *Thinking, fast and slow*.

Dorottya Demszky, Diyi Yang, David S Yeager, Christopher J Bryan, Margarett Clapper, Susannah Chandhok, Johannes C Eichstaedt, Cameron Hecht, Jeremy Jamieson, Meghann Johnson, et al. 2023. Using large language models in psychology. *Nature Reviews Psychology*, 2(11):688–701.

Danica Dillion, Niket Tandon, Yuling Gu, and Kurt Gray. 2023. Can ai language models replace human participants? *Trends in Cognitive Sciences*, 27(7):597–600.

Meta Fundamental AI Research Diplomacy Team (FAIR) FAIR, Anton Bakhtin, Noam Brown, Emily Dinan, Gabriele Farina, Colin Flaherty, Daniel Fried, Andrew Goff, Jonathan Gray, Hengyuan Hu, et al. 2022. Human-level play in the game of diplomacy by combining language models with strategic reasoning. *Science*, 378(6624):1067–1074.

Chen Gao, Xiaochong Lan, Zhihong Lu, Jinzhu Mao, Jinghua Piao, Huandong Wang, Depeng Jin, and Yong Li. 2023. Social-network simulation system with large language model-empowered agents. *arXiv preprint arXiv:2307.14984*.

Lewis R Goldberg, John A Johnson, Herbert W Eber, Robert Hogan, Michael C Ashton, C Robert Cloninger, and Harrison G Gough. 2006. The international personality item pool and the future of public-domain personality measures. *Journal of Research in personality*, 40(1):84–96.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2020. Measuring massive multitask language understanding. *arXiv preprint arXiv:2009.03300*.

John J Horton. 2023. Large language models as simulated economic agents: What can we learn from homo silicus? Technical report, National Bureau of Economic Research.

Xuanfa Jin, Ziyan Wang, Yali Du, Meng Fang, Haifeng Zhang, and Jun Wang. 2024. Learning to discuss strategically: A case study on one night ultimate werewolf. *arXiv preprint arXiv:2405.19946*.

Oliver P John, Laura P Naumann, and Christopher J Soto. 2008. Paradigm shift to the integrative big five trait taxonomy. *Handbook of personality: Theory and research*, 3(2):114–158.

D Kahneman. 1982. The psychology of preferences. *Scientific American*.

Daniel Kahneman and Amos Tversky. 2013. Prospect theory: An analysis of decision under risk. In *Handbook of the fundamentals of financial decision making: Part I*, pages 99–127. World Scientific.

Saketh Reddy Karra, Son The Nguyen, and Theja Tulabandhula. 2022. Estimating the personality of white-box language models. *arXiv preprint arXiv:2204.12000*.

Guohao Li, Hasan Hammoud, Hani Itani, Dmitrii Khizbullin, and Bernard Ghanem. 2023. Camel: Communicative agents for" mind" exploration of large language model society. *Advances in Neural Information Processing Systems*, 36:51991–52008.

Xingxuan Li, Yutong Li, Shafiq Joty, Linlin Liu, Fei Huang, Lin Qiu, and Lidong Bing. 2022. Does gpt-3 demonstrate psychopathy? evaluating large language models from a psychological perspective. *arXiv preprint arXiv:2212.10529*.

Jonathan Light, Min Cai, Sheng Shen, and Ziniu Hu. 2023. Avalonbench: Evaluating llms playing the game of avalon. In *NeurIPS 2023 Foundation Models for Decision Making Workshop*.

Bo Liu, Yuqian Jiang, Xiaohan Zhang, Qiang Liu, Shiqi Zhang, Joydeep Biswas, and Peter Stone. 2023. Llm+ p: Empowering large language models with optimal planning proficiency. *arXiv preprint arXiv:2304.11477*.

Niklas Luhmann. 2018. *Trust and power*. John Wiley & Sons.

Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, et al. 2024. Self-refine: Iterative refinement with self-feedback. *Advances in Neural Information Processing Systems*, 36.

Robert R McCrae and Paul T Costa. 1987. Validation of the five-factor model of personality across instruments and observers. *Journal of personality and social psychology*, 52(1):81.

Robert R McCrae and Paul T Costa Jr. 1991. The neo personality inventory: Using the five-factor modei in counseling. *Journal of Counseling & Development*, 69(4):367–372.

Robert R McCrae and Paul T Costa Jr. 2004. A contemplated revision of the neo five-factor inventory. *Personality and individual differences*, 36(3):587–596.

RR McCrae and PT Costa. 1989. The neo-pi/neo-ffi manual supplement. *Odessa, FL: Psychological Assessment Resources. Google Scholar*.

Tomas Mikolov. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Arvind Neelakantan, Tao Xu, Raul Puri, Alec Radford, Jesse Michael Han, Jerry Tworek, Qiming Yuan, Nikolas Tezak, Jong Wook Kim, Chris Hallacy, et al. 2022. Text and code embeddings by contrastive pretraining. *arXiv preprint arXiv:2201.10005*.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744.

Keyu Pan and Yawen Zeng. 2023. Do llms possess a personality? making the mbti test an amazing evaluation for large language models. *arXiv preprint arXiv:2307.16180*.

Joon Sung Park, Joseph O'Brien, Carrie Jun Cai, Meredith Ringel Morris, Percy Liang, and Michael S Bernstein. 2023. Generative agents: Interactive simulacra of human behavior. In *Proceedings of the 36th annual acm symposium on user interface software and technology*, pages 1–22.

Joon Sung Park, Lindsay Popowski, Carrie Cai, Meredith Ringel Morris, Percy Liang, and Michael S Bernstein. 2022. Social simulacra: Creating populated prototypes for social computing systems. In *Proceedings of the 35th Annual ACM Symposium on User Interface Software and Technology*, pages 1–18.

Chen Qian, Xin Cong, Cheng Yang, Weize Chen, Yusheng Su, Juyuan Xu, Zhiyuan Liu, and Maosong Sun. 2023. Communicative agents for software development. *arXiv preprint arXiv:2307.07924*, 6.

Mark Riedl and Vadim Bulitko. 2012. Interactive narrative: A novel application of artificial intelligence for computer games. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 26, pages 2160–2165.

Brent W Roberts and Wendy F DelVecchio. 2000. The rank-order consistency of personality traits from childhood to old age: a quantitative review of longitudinal studies. *Psychological bulletin*, 126(1):3.

M Safdari, G Serapio-García, C Crepy, S Fitz, P Romero, L Sun, M Abdulhai, A Faust, and M Matarić. 2023. Personality traits in large language models. arxiv. *Preprint posted online August*, 28.

Jack Serrino, Max Kleiman-Weiner, David C Parkes, and Josh Tenenbaum. 2019. Finding friend and foe in multi-agent games. *Advances in Neural Information Processing Systems*, 32.

Yunfan Shao, Linyang Li, Junqi Dai, and Xipeng Qiu. 2023. Character-llm: A trainable agent for role-playing. *arXiv preprint arXiv:2310.10158*.

Herbert A Simon. 1991. Bounded rationality and organizational learning. *Organization science*, 2(1):125–134.

Joseph Turian, Lev Ratinov, and Yoshua Bengio. 2010. Word representations: a simple and general method for semi-supervised learning. In *Proceedings of the 48th annual meeting of the association for computational linguistics*, pages 384–394.

Hongyu Wan, Jinda Zhang, Abdulaziz Arif Suria, Bingsheng Yao, Dakuo Wang, Yvonne Coady, and Mirjana Prpa. 2024. Building llm-based ai agents in social virtual reality. In *Extended Abstracts of the CHI Conference on Human Factors in Computing Systems*, pages 1–7.

Shenzhi Wang, Chang Liu, Zilong Zheng, Siyuan Qi, Shuo Chen, Qisen Yang, Andrew Zhao, Chaofei Wang, Shiji Song, and Gao Huang. 2023. Avalon's game of thoughts: Battle against deception through recursive contemplation. *arXiv preprint arXiv:2310.01320*.

Xintao Wang, Yunze Xiao, Jen-tse Huang, Siyu Yuan, Rui Xu, Haoran Guo, Quan Tu, Yaying Fei, Ziang Leng, Wei Wang, et al. 2024. Incharacter: Evaluating personality fidelity in role-playing agents through psychological interviews. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1840–1873.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837.

Shuang Wu, Liwen Zhu, Tao Yang, Shiwei Xu, Qiang Fu, Yang Wei, and Haobo Fu. 2024. Enhance reasoning for large language models in the game werewolf. *arXiv preprint arXiv:2402.02330*.

Yuzhuang Xu, Shuo Wang, Peng Li, Fuwen Luo, Xiaolong Wang, Weidong Liu, and Yang Liu. 2023. Exploring large language models for communication games: An empirical study on werewolf. *arXiv preprint arXiv:2309.04658*.

Yasin Abbasi Yadkori, Ilja Kuzborskij, András György, and Csaba Szepesvári. 2024. To believe or not to believe your llm. *arXiv preprint arXiv:2406.02543*.

Geogios N Yannakakis. 2012. Game ai revisited. In *Proceedings of the 9th conference on Computing Frontiers*, pages 285–292.

Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Tom Griffiths, Yuan Cao, and Karthik Narasimhan. 2024. Tree of thoughts: Deliberate problem solving with large language models. *Advances in Neural Information Processing Systems*, 36.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. 2022. React: Synergizing reasoning and acting in language models. *arXiv preprint arXiv:2210.03629*.

Hongxin Zhang, Weihua Du, Jiaming Shan, Qinhong Zhou, Yilun Du, Joshua B Tenenbaum, Tianmin Shu, and Chuang Gan. 2023. Building cooperative embodied agents modularly with large language models. *arXiv preprint arXiv:2307.02485*.

Andrew Zhao, Daniel Huang, Quentin Xu, Matthieu Lin, Yong-Jin Liu, and Gao Huang. 2024. Expel: Llm agents are experiential learners. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 19632–19642.

Xuhui Zhou, Hao Zhu, Leena Mathur, Ruohong Zhang, Haofei Yu, Zhengyang Qi, Louis-Philippe Morency, Yonatan Bisk, Daniel Fried, Graham Neubig, et al. 2023. Sotopia: Interactive evaluation for social intelligence in language agents. *arXiv preprint arXiv:2310.11667*.

Xizhou Zhu, Yuntao Chen, Hao Tian, Chenxin Tao, Weijie Su, Chenyu Yang, Gao Huang, Bin Li, Lewei Lu, Xiaogang Wang, et al. 2023. Ghost in the minecraft: Generally capable agents for open-world environments via large language models with text-based knowledge and memory. *arXiv preprint arXiv:2305.17144*.

A Simplified version of Ultimate Werewolf Game

One Night Ultimate Werewolf (ONUW) is a variant of the social deduction game Werewolf. In this game, players only have one night to use their abilities and one day to discuss and vote for who is werewolf. The challenge lies in the incompleted information and potential deceptions, which create uncertainty and confusion for all players. To simplify the game, we remove all the role switch abilities, which increases the stablity of the game but, of course, sadly reduce the playfulness and randomness. However, the lack of exploration of the upper limits of gameplay is acceptable because our main focus is on the impact of different LLM agent design methods on the playability and robustness of the game. Therefore, it is appropriate to choose a relatively simple game framework as the basis for our experiments.

A.1 Game Setup

In our work, there will be eight players participating in the game. During the setup phase, each player will be randomly assigned a seat (public) and a role (private). These roles include: 1 Seer, 2 Masons, 2 Villagers, 1 Werewolf, 1 Minnor, 1 Tanner (see A.4). Players will know their own role but not the roles of any other players. All these roles is splited into three teams:

- **Team Village** 1 Seer, 2 Masons, 2 Villagers(5 players in total)
- **Team Werewolf** 1 Werewolf, 1 Minnor(2 players in total)
- **Team Tanner** 1 Tanner(1 players in total)

A.2 Game Process

After setup, the game will run throught the follow three phases: Night Phase, Day Phase, and Voting Phase.

- **Night Phase**: The system will send the information to all players who have the associated abilities. The involved roles are: (1) Seer, (2) Mason.
- **Day Phase**: Players discuss in turn to detect the werewolf among them. From seat 1 to seat 8, each player will express their views in a statement of no more than 120 words. The discussion will consist of three rounds, totaling 24 statements, after which the day phase will end.
- **Voting Phase**: Players simultaneously vote for the players they believe are most likely to be the Werewolf. The player with the most votes is out.

A.3 Winning Conditions

The role of player who are voted out in the Voting Phase will be revealed. The winning conditions is listed as below:

- **The Team Village wins**: The voted-out player is Werewolf. If the player voted out is not from the Team Village, such as a Minion or a Tanner, it is also not considered as a victory.
- **The Team Werewolf wins**: The voted-out player is not Werewolf or Tanner. If the player voted out is from the Team Werewolf , such as a Minion , it is also considered as a victory.
- **The Team Tanner wins**: The voted-out player is Tanner.

A.4 Role Descriptions

Here is the role's description and introduction of their abilities:

- **Seer**: Seer is in Team Village. In Night Phase, Seer can select two other players at night, the system will tell whether one of them is Werewolf or not.
- **Mason**: Mason is in Team Village. In Night Phase, Mason will know each other.
- **Villager**: Villager is in Team Village. Villager does not have extract information in Night Phase.
- **Werewolf**: Werewolf is in Team Werewolf. Werewolf does not have extract information in Night Phase.
- **Minnor**: Minnor is in Team Werewolf. Minnordoes not have extract information in Night Phase.
- **Tanner**: Tanneris in Team Tanner. Tannerdoes not have extract information in Night Phase.

B Big Five Persona(B5P) Generation Algorithm

Factor	Evaluate	Description
Openness	+1	Believe in the importance of art.
Extraversion	+1	Cheer people up.
Agreeableness	-1	Cut others to pieces.
Openness	+1	Enjoy wild flights of fantasy.
Neuroticism	-1	Am very pleased with myself.
...	...	...

Table 9: example of the descriptions from NEO-PI-R

Factor	Evaluate	Description
Extraversion	+1	Active, Outgoing, Talkative, ...
Extraversion	-1	Reserved, Loner, Quiet, ...
Openness	+1	Imaginative, Creative, Original, ...
...	...	...

Table 10: example of the adjective description

In this section, we will introduce the specific operational process of the B5PG system. Table 9 shows partial data extracted from the NEO-PI-R, which serves as the source data for the NEO statement description function $F_N(k)$. Table 10 presents the adjective words of five personality types, serving as the source data for the adjective description function $F_A(k)$. Table 11 illustrates the process of generating an agent's B5P description, where:

- a) Randomly generates five integers between 1 and 7 to form a random agent personality,
- b), c), d) retrieve corresponding string data from the database based on the results generated in a),
- e) consolidates the above results into a final text paragraph describing the agent's personality.

C Favor Dynamics Algorithm

In this section, the detailed algorithm 2 of Favor Dynamics will be introduce. As the setup, each agent i maintains the set of favor level $\left\{F_{ij}^{(T)}\right\}$, an float number in range of (1,100), where j represents the target agent and T represents the turn. Duraing each turn T, the agent i collects the dialog histroy of last turn as $H_i^{(T-1)}$ and sends it to LLM. The response from LLM is noted as $\left\{f_j^{(T)}\right\}$, which represents the variation of Favor for other agent j in rount T, where each number in the array has a value range of (1,10).

As the output, the system returns the favor statement in string format. First we seperate the favor into 5 levels, each has a set descriptions. Each level of favor is denoted as k, where $B_{min}(k)$represents the lower bound of preference level k, $B_{max}(k)$ represents the upper bound of Favor level k. The range of Favor levels is divided into five equal parts from 1to 100, that is (1,20), (20,40), ..., (80,100).

Here is the step by step introduction of the algorithm:

Analyze (Algorithm 2 Line 1) As the first step Favor determining, analyze is a process of information rumination, consolidation and analysis . We filter the affinity based solely on historical conversations between agents. By listing the historical dialogues of other agents, we send and ask the LLM to determine the current "level of Favor" for the current angent toward agents(The prompt is shown in Table 12). We will ask the LLM to return a 1-10 number to repersent the preference for each target agent. In some situation, the LLM will only return the evaluation of part of agents, and only the returned data will be collected.

Calculate (Algorithm 2 Line 4) The second phase involves mapping the current level of Favor to an increase or decrease range and adding it to the existing Favor score for each agent to arrive at the final "Favor" for this round. We set the remapping interval to (-30, 30), because the description of Favor is divided into five intervals from 1 to 100, meaning each interval is 20 points wide. Thus, a change of ś30 ensures that a single update can span an entire descriptive interval. During our research, we found that single calls to the LLM to return Favor levels can be somewhat random, but the cumulative nature of Favor changes ensures a less drastic shifts and better reflects the gradual changes in affinity by common daily observation.

Translation (Algorithm 2 Line 7) The third step is translating the Favor score into text, where we use a bracketing method to translate the score into corresponding textual descriptions of affinity levels when it falls within specific ranges. For example, if the Alpha's Favor level toward Beta lies in 80, then we can get a translated description as "Alpha pretty much appreciate Beta". The list of

a) Generation of five factor scores B_i
Five 1-7 score are randomly generated to fill as five factor value of agent i : "Agreeableness": 7 "Extraversion": 1 "Neuroticism": 2 "Openness": 4 "Conscientiousness": 1
b) Result of $F_N(k)$
N("Agreeablness")="Am easy to satisfy" N("Extraversion")="Cheer people up" N("Neuroticism")="Mess things up" N("Openness")= ignored N("Conscientiousness")="Don't see things through"
c) Result of $F_A(k)$
A("Agreeablness")="Trusting, Lenient, Soft hearted" A("Extraversion")="Reserved,Unfeeling,Sober" A("Neuroticism")="Hardy, Even tempered" A("Openness")= ignored A("Conscientiousness")="Lazy, Disorganized, Aimless"
d) Result of $F_R(b_k)$
R("Agreeablness")=R(7)=3 R("Extraversion")=R(1)=3 R("Neuroticism")=R(2)=2 R("Openness")=R(4)=0 R("Conscientiousness")=R(1)=3
e) Final result of $F_M(\{F_N(k), F_A(k)\}_{t=1}^{F_R(b_k)}, b_k)$
You are Beta. You totally agree that you Am easy to satisfy. You totally disagree that you Cheer people up. You strongly agree that you Mess things up. You totally agree that you Don't see things through. You speak in a incredibly Trusting, remarkably Lenient, incredibly Soft hearted way. You speak in a incredibly Reserved, incredibly Unfeeling, extremely Sober way. You speak in a pretty Hardy, pretty Even tempered way. You speak in a incredibly Lazy, remarkably Disorganized, incredibly Aimless way.

Table 11: example of the generation of B5P statement

all descriptions toward other agents will be sum up into a Favor description of the agent, which can be utilized in the following processes.

D Strategy Decision Algorithm

The Strategy Decision system consists of two phases: **Judgement** and **Strategy**.

In the **Judgement** phase, the system performs probabilistic reasoning about the roles of the agents on the field through the LLM. It collects historical dialogue information and favorability data of other agents to construct a prompt (Equation 7). An example of Judgement Prompt is provided in Table 13.

During the **Strategy** phase, the agent logically derives the next strategy to implement. The system gathers historical dialogue information, role probability data from the Judgement phase, and a pre-configured strategy database to determine the strategy for the current round (Equation 10). The prompt content for this phase is illustrated in Table 14.

E Text Distance Algorithm

Text Distance is an algorithm used to check the diversity of vocabulary usage within a set of textual data. It is introduced as below that how to utilize text embeddings for estimating the the variety and abundance among sequences of text-based content. The input of the algorithm is a list of LLM-generated dialog information, the output of Dialog Generation stage, regardless of whether it originates from a single agent or multiple agents, as some duplicated words in which will be cleaned, such as 'Alpha says' or qutation mark in the end. The result is a floating-point number, representing the text distance of the input text; the higher the value, the more diverse the vocabulary used in the input text.

Each of the input dialog is marked as C_i, where

Algorithm 2: Pseudo-code for Favor Dynamics System

```
Data: historical dialogs $H_i^{(T-1)}$, favor value of last turn $\left\{F_{ij}^{(T-1)}\right\}$
Result: favor statement $\mathcal{F}_i^{(T)}$
Get the favor variation list $\left\{f_j^{(T)}\right\}$ from LLM request $LLM_{favor}$ based on the historical dialog
  input $(H_i^{(T-1)})$ according to equation 5 ;
Init the favor statement $\mathcal{F}_i^{(T)}$ ;
for $j \in Agents$ do
    Update $F_{ij}^{(T)}$, the favor level of agent $i$ toward agent $j$, by remaping the favor variation $f_j^{(T)}$
      according to equation 6 ;
    for $k \in 1, 2, ..., 5$ do
        if $F_{ij}^{(T)} > B_{min}(k)\ and\ F_{ij}^{(T)} < B_{max}(k)$ then
            Set the description toward agent $j$ as the description of $k$ as $\mathcal{F}_{ij}^{(T)}$ ;
            Add $\mathcal{F}_{ij}^{(T)}$ to $\mathcal{F}_i^{(T)}$ ;
Return $\mathcal{F}_i^{(T)}$
```

Prompt Input

```
Here are previous dialogs:
Eta say:'Hey everyone! Im Eta, and Ive got something important to share. As the Seer, Ive identified
Delta and Epslion as the real threats in our midsttheyre definitely hiding something! ... Lets keep
the discussion lively and uncover the truth together!';
Zeta say:'I must say, this situation feels rather twisted. Given the noise from Alpha, Beta, and Theta,
we need to tread carefully... As for Alpha, I keep those thoughts closeIll let them simmer for now. The
true threats might be hiding in plain sight. Stay vigilant, everyone.';
...(The other dialogs is omitted)
Here are your judgement toward the other agents:
Beta has a 72.0% chance to be a werewolf, which is your opponent
Gamma has a 35.0% chance to be a villager, which is your teammate
...
Accroding to the dialogs above, please output how much you like the agents. On the scale of 1 to 10,
where 1 is totally disliked (e.g. no interested, uncomfortable) and 10 is extremely liked (e.g. highly
appriciate and have great expectations), please rate the your favor for the agents:
Alpha;Beta;Gamma;Delta;Epslion;Theta;Zeta;Eta
Output in format : [player name,favor score]
for example:
[Alpha,2]
[Beta,3]
[Theta,6]
You are Alpha.
Now output the rate in format, DO NOT output other words.
```

LLM Output

```
[Alpha,7]
[Beta,2]
[Gamma,5]
[Delta,6]
[Epslion,5]
[Theta,3]
[Zeta,4]
[Eta,1]
```

Table 12: Prompt and LLM Output of requesting the update of favor variation

Prompt Input

```
Your Mission is
There are serval roles in the game: Werewolf, Minion, Seer, Mason, Tanner and Villager. Werewolf, Minon
is part of evil team, Seer, Mason and Villager is part of the good team. Tanner is the third party.
All players will speak in turn for several rounds. After the discussion rounds, players will vote to
identify the Werewolf. If the Werewolf is voted out, the good team win. If a Villager or a Seer or a
Mason is voted out, the Werewolf wins. If the Thinker is voted out, the Tanner wins and all the rest
of players lose. You are Seer. You are part of good team. You need to find out who is the werewolf. In
the night, You know that Gamma or Theta is a werewolf.
The previous dialog is
Alpha say:'Hey everyone, it's Alpha here. First off, I want to echo Deltas concerns about Theta; their
behavior feels off, and I can't shake the suspicion that they might be hiding something sinister. ....
Trust in each other, and let's keep the discussion going!';
Eta say:'Hey everyone! Im Eta, and Ive got something important to share. As the Seer, Ive identified
Delta and Epslion as the real threats in our midsttheyre definitely hiding something!...  So, lets
really dig into what';
...(The other dialogs is omitted)
The players in the game
Alpha;Beta;Gamma;Delta;Epslion;Theta;Zeta;Eta
Your attitude toward other agents
You strongly loathe Beta.;You strongly loathe Gamma.;You fairly dislike Delta.;You strongly loathe
Epslion.;You strongly loathe Theta.;You fairly dislike Zeta.;You strongly loathe Eta.
The roles in the game
seer;werewolf;mason;minion;tanner;villager
Output
Output the posibility of each players's role by their dialogs.  The posibility is in the range of
0.01-1.00 Output in format : [player name,role,posibility]
for example:
[Alpha,Seer,0.45]
[Beta,Werewolf,0.68]
[Theta,Villager,0.81]
now output the judgement in format,DO NOT output other words:
```

LLM Output

```
[Alpha,Villager,0.10]
[Beta,Seer,0.90]
[Gamma,Villager,0.05]
[Delta,Mason,0.15]
[Epslion,Mason,0.15]
[Theta,Werewolf,0.85]
[Zeta,Villager,0.20]
[Eta,Tanner,0.05]
```

Table 13: Prompt and LLM Output of judgement phase of Strategy Decision System

i is the index of the dialog and the total number of input dialogs is n. The e_i is the embedding vector of conversation C_i. A sequence of conversation $\{C_i\}$can be transformed into a sequence of numerical vectors $\{e_i\}$ by an text embedding algorithm(12). Then, by computing the distance between each pair of these embedding result using the formula(Equation 12), we obtain the distance matrix $[d_{ij}]$ for this chats sequence. Finally, by calculating the mean and standard deviation of every distance value in the matrix, we derive the average distance and standard deviation of the data in distance matrix. The full algorithm is listed in Algorithm 3.

$$e_i = F_e(C_i) \tag{11}$$

$$F_{dis}(e_i, e_j) = 1 - \frac{e_i \cdot e_j}{\|e_i\| \cdot \|e_j\|} \tag{12}$$

F Judgement Variation Algorithm

Judgement Variation is an algorithm used to calculate the degree of variation in an agent's decision-making. Its input consists of dialogue information output by the agent, and its output is a floating-point number that represents the extent of variation in the agent's decisions. This also indicates the richness of the dialogue content.

Firstly, the total number of conversations is n and the ith conversation can be marked as C_i. Through LLM request, we can obtain the vote vectors $\{v_r|t\}$for each turn from the sequence of conversation $\{C_i\}$, where r represents by the role, and t represents by the turn(Equation 13. Then, we

Prompt Input

There are serval roles in the game
Werewolf, Minion, Seer, Mason, Tanner and Villager. Werewolf, Minon is part of evil team, Seer, Mason and Villager is part of the good team. Tanner is the third party. All players will speak in turn for several rounds. After the discussion rounds, players will vote to identify the Werewolf. If the Werewolf is voted out, the good team win. If a Villager or a Seer or a Mason is voted out, the Werewolf wins. If the Thinker is voted out, the Tanner wins and all the rest of players lose.
You are Seer. You are part of good team. You need to find out who is the werewolf. In the night, You know that Gamma or Theta is a werewolf.
Here are your judgement toward the other agents
Alpha has a 10.0% chance to be a villager, which is your teammate
Gamma has a 5.0% chance to be a villager, which is your teammate
Delta has a 15.0% chance to be a mason, which is your teammate
Epslion has a 15.0% chance to be a mason, which is your teammate
Theta has a 85.0% chance to be a werewolf, which is your opponent
Zeta has a 20.0% chance to be a villager, which is your teammate
Eta has a 5.0% chance to be a tanner, which is your opponent
Strategys
As a Seer, you can claim to be a Seer and say what you know in the night; you can hide your role; you can take a risk to claim to be a villager and check who is werewolf; you can point out another player who claim to be a seer is Werewolf or Mason.
You are Beta
Output
According to the information above, Please decide your strategy step by step (in no more than 80 words):
Your strategy:

LLM Output

I will claim to be a villager and emphasize the importance of voting out suspicious players. Ill point out Theta, highlighting the high probability of being a werewolf, while remaining cautious about any potential Tanner involvement. I will observe the reactions of others to gather more information. If pressed, I may reveal my true role as Seer to solidify my credibility, but only if necessary to maintain trust and avoid unnecessary conflict.

Table 14: Prompt and LLM Output of strategy phase of Strategy Decision System

can sum up the vote list together as V, which is a vector representing how many vote a role received. Some of the roles(e.g. Villager and Mason) are duplicated in the role list, so their votes are split to the number of the role and we get an extended vote list V_{split}. For example, if the Werewolf get 4 votes and the Villager get 2 votes, and the vote list is listed in [Werewolf,Villager1,Villager2], then the splited vote list will be $V_{split} = [4, 1, 1]$. And the extended role count is R^*(in our experiment, $R = 6$, $R^* = 8$). Finally, we can calculate the standard deviation of the extened vote list. The agents' difference in the attitude toward others can be represented by the standard deviation, the smaller the number is, the higher variation the vote have. So we define the value of judgement variation as the reciprocal of it. The whole algorithm is introduced in Algorithm 4.

$$\{v_t\} \leftarrow LLM_{vote}(\{C_i|t\}) \tag{13}$$

G EWAVM Evaluation

EWAVM Evaluation is an evaluation algorithm that assesses the content of text by sending messages to a Large Language Model (LLM). The evaluation is divided into five dimensions: Engagement, Wonder, Attraction, Variation, and Memorability. When formulating prompts, we not only attach the historical dialogue but also provide textual descriptions of the evaluation dimensions. We include explanations for both extreme low scores and extreme high scores within these dimensions. The results returned by the LLM could be simple scores (for dimensions like Engagement, Wonder, Attraction) or more complex evaluations (for Variation and Memorability). Upon receiving the results from the LLM, we process this information and ultimately provide a score for the evaluated content within that specific dimension.

G.1 Evaluate the Engagement, Wonder and Attraction

In the evaluation the Engagement, Wonder and Attraction, an LLM request with the whole round's dialog and the description of the evaluation perspective will be attached(Table 15 16 17). The output of the LLM will contain a score. Then the output score will be collected and averagd, which is served as the level of corresponding evaluation

Algorithm 3: Pseudo-code for Text Distance

```
Data: dialogs $C_i$
Result: text distance $\bar{\mathcal{D}}_i, \sigma\mathcal{D}_i$
Calculate the embedding vector for the conversation set $\{C_i\}$ according to Equation 11. The
  result is a set of embedding vector $\{e_i\}$.
for $i \in 1, 2..., n$ do
    for $j \in 1, 2..., n$ do
        Calculate the texture distance between $C_i$ and $C_j$ according to Equation 12 and save to
          text distance matrix $[d_{ij}]$
$\bar{\mathcal{D}}_i \leftarrow \frac{\sum_{i=1}^{n}\sum_{j=1}^{n} d_{ij}}{n^2}$
$\sigma\mathcal{D}_i \leftarrow \sqrt{\frac{\sum_{i=1}^{n}\sum_{j=1}^{n}(d_{ij}-\bar{\mathcal{D}}_i)^2}{n^2-1}}$
Return $\bar{\mathcal{D}}_i, \sigma\mathcal{D}_i$
```

Algorithm 4: Pseudo of text Judgement Variation

```
Data: dialogs $C_i$
Result: judgement variation $\mathcal{J}$
for $i \in 1, 2..., n$ do
    Request the LLM and get the vote according to the conversation for the conversation set $C_i$
      according to the Equation 13 and get teh vote list $\{v_t\}$
Sum up all the votes and form a vote list $V$ , that is $V \leftarrow \sum_{t=1}^{T}\{v_t\}$
Extend the vote list $V$ to fit the number of roles, $V^* \leftarrow F_E(V)$
Calculate the average number of votes, $\bar{V^*} = \frac{\sum_{r=1}^{R^*} V_r^*}{R^*}$
Calculate the standard deviation, $\sigma V^* = \sqrt{\frac{\sum_{r=1}^{R^*}(V_r^*-\bar{V})^2}{R^*-1}}$
$\mathcal{J} = \frac{1}{\sigma V^*}$
Return $\mathcal{J}$
```

dimensions.

G.2 Evaluate the Variation

We use the dialogs in each turn to request the agent's who-is-what result(we define it as a 'mention list') from LLM(prompt can be seen in Table 18). Each of the game is last for T turn(T is 3 in our experiment) and the t is used for representing index of the turn. $\{C_t\}$ is the set of all the dialogs from T turns. In each LLM response, we get a mention list with the capacity of V and each of the item is described as m_{tv}, where V is the total number of the agents' opinions on other agents' role and v is the index of result. The who-is-what matrix M_i is a N * N matrix(N is the player number, N is 8 in our experiment), where the p-th row q-th column has the number r, meaning that agent p consider agent q as the role r. And the distance between two who-is-what matrix can be calculated by Equation 15 and Equation 16.

$$\{\bigcup_{v=1}^{V} m_{tv}\} \leftarrow LLM_{variation}(\{C_t\}) \tag{14}$$

$$F_v(M_0, M_1) = \sum_{i=1}^{N}\sum_{j=1}^{N} f_v(M_0[i][j], M_0[i][j]) \tag{15}$$

$$f_v(a,b) = \begin{cases} 0(a = b) \\ 1((a \neq b)\ and\ (a = 0\ or\ b = 0) \\ 2(a \neq b\ and\ a \neq 0\ and\ b \neq 0) \end{cases} \tag{16}$$

G.3 Evaluate the Memorability

In this algorithm, we will accumulate all the conversation data from the experiment and evaluate the dataset to get the score of memorability, which describes the variability of the dialog content.

For the specified algorithm, we will first request the top k(k=10) most valuable sentense from the LLM among all dialogs from game. We use i for the index of the game. A_i is the full conversation text for the i th game. With each LLM response, we will get k highlighted sentenses, in form of h_j, where j is the index of the highlighted sentence(Equation 17). The role in the game is represented by r and there are totally R roles. An example of the prompt to LLM and the response can be seen in Table 19. The highlight moment counter vector is repersented by $\{c_i[r]\}$, where $c_i[r]$ means the total number of role r shows up in the top k most important sentense in game i. The role reflection function F_R can search the sender of the sentence h_{ij} and return its role index. We then calculate the frequece of each role from this sentense list(Villager and Mason will be divided into two unit because they are duplicated in the game). For each role-frequency vector, we measure the standard deviation and calculate the mean to form the score of diversity of highlight moments(Equation 18). The full algorithm is shown in Algorithm 6.

$$\{\bigcup_{j=1}^{k} h_{ij}\} \leftarrow LLM_{memorability}(\{A_i\}) \tag{17}$$

$$\mathcal{M} = \frac{100n}{\sum_{i=1}^{n} c_{i,std}} \tag{18}$$

H Big Five Personality Generation System Analysis

In the B5P Generation System, we automatically generate textual descriptions of personality traits for the LLM before sending specific prompts. These descriptions are added into the LLM's message list, influencing the tone of the text generated by the LLM to have certain personality tendencies. Here, we want to conduct an independent evaluation of the B5P Generation System to validate its effectiveness.

The experimental design is as follows:

1. Four agent personality descriptions will be pre-generated for the experiment(see Table 20), including one baseline agent and three comparative personality descriptions generated using the B5P system. The suggested five-dimensional personality descriptions can be referenced in the table below, and the specific content of the descriptions can be found in Section 3.2.

2. Make conversation by these four agents, whose roles are all considered as Villagers, with their task descriptions remaining consistent, and without any additional historical dialogue information. After cleaning up the dirty words, the LLM's returned dialogue content will be divided into different dialog lists and

Algorithm 5: Pseudo of evaluating Variation

```
Data: dialogs $C_i$
Result: Variation Score $\mathcal{V}$
Get the description of agent mention list $\{m_{tv}\}$ from LLM according to the Equation 14
for $t \in 1, 2..., T$ do
    for $v \in 1, 2..., V$ do
        $M_i[m_{tv}.speaker][m_{tv}.target] = m_{tv}.role$
$\mathcal{V} = \frac{\sum_{i=1}^{T} \sum_{j=1}^{T} F_v(M_i, M_j)}{T * T}$
Return $\mathcal{V}$
```

Algorithm 6: Pseudo of evaluating Memorability

```
Data: full conversation content set $\{A_i\}$
Result: Memorability $\mathcal{M}$
Get the highlighted sentense set $\{\bigcup_{j=1}^{k} h_{ij}\}$ from LLM according to Equation 17
for $i \in 1, 2..., n$ do
    for $r \in 1, 2..., R$ do
        Initalize the role count vector $c_i[r]$
    for $j \in 1, 2..., k$ do
        Search for the top k result and count the frequency of each role ,
         $c_i[F_R(h_{ij})] \leftarrow c_i[F_R(h_{ij})] + 1$
    Calculate the average number of role frequency $\bar{c_i} = \frac{\sum_{r=1}^{R} c_i[r]}{R}$
    Calculate the standard deviation of role frequency $\sigma c_i = \sqrt{\frac{\sum_{r=1}^{R} (c_i[r] - \bar{c_i})^2}{R-1}}$
Get the final score of memorability $\mathcal{M}$ according to Equation 18
Return $\mathcal{M}$
```

saved separately for each agent. Make 50 conversation requests for each agent.

3. Perform text distance analysis (as described in section 3.6.1) on each agent's dialog list to analyze the diversity of the returned content.

The result of this experiment can be checked in section 4.1.

Prompt Input

You are an audience who enjoys fun. You give higher scores to content that is novel, humorous, cute, and interesting, and lower scores to content that is monotonous, repetitive, and similar.
You will be given a list of dialogs speaked by various players in the One Night Ultimate Werewolf Game. In the game, there are serval roles in the game: Werewolf, Minion, Seer, Mason, Tanner and Villager. Werewolf, Minon is part of evil team, Seer, Mason and Villager is part of the good team. Tanner is the third party. All players will speak in turn for several rounds. After the discussion rounds, players will vote to identify the Werewolf. If the Werewolf is voted out, the good team win. If a Villager or a Seer or a Mason is voted out, the Werewolf wins. If the Thinker is voted out, the Tanner wins and all the rest of players lose.
Below is the dialogs:
Alpha(Alpha is Villager) says: 'Villager, I claim my role openly to promote trust among all players. I sense that Beta and Gamma exhibit behaviors that raise red flags; their actions seem more aligned with our foes. Delta, Epsilon, Theta, and Zeta, I encourage you to share your thoughtsyour insights will help us discern the truth. Its crucial we peel back the layers of possible alliances. My focus is on voting out Beta, who poses a notable threat to our good team. Together, lets navigate this challenge and bring clarity to our mission!'
Beta(Beta is Mason) says: 'I find Alpha's claim rather bold, but I sense an ulterior motive beneath it. While he points fingers at both me and Gamma, I urge you all to consider that he might be deflecting attention from the real threat. As a Mason, I can confirm that Theta is indeed one of us, solidifying my assertion. Gammas behavior strikes me as suspicious, and I implore you to weigh the evidence carefully. Together, we must remain vigilant against deception, especially from those who claim roles without proof. Lets unite our efforts and focus our sights on Gamma, for the good of our team!'
...
[Evaluation Steps]
1. Read through the dialogs and evaluate the level of richness, diversity, aboundance of the full dialogue step by step.
2. Write out in a step by step manner your reasoning about the criterion to be sure that your conclusion is correct.
3. Write in a new line to display the level of richness, diversity, aboundance. Output in format: [score]. Use the given scale from 1-10 to rate the level of richness, diversity, aboundance: 1 means Players converse in a similar manner and 10 Each player has different speaking habits, and different players have different expression habits. Output in format: [value] , output only one score, do not return multiple score
example1:
[9.5]
example2:
[7.2]
example1:
[1.5]

LLM Output

1. The dialogues present a diverse range of players expressing their thoughts and suspicions about each other's roles in the game. Each character has their unique way of speaking, relying on different strategies to persuade others. For example, Alpha emphasizes transparency and teamwork, while Beta deflects attention towards others to protect himself.
2. Throughout the back-and-forth conversations, players display varying levels of paranoia, logic, and humor. Epslion injects a lighthearted tone with phrases like "Yay team!" while others focus more seriously on articulating their arguments. This adds a layer of richness to the dialogues, creating an interesting mix of perspectives and styles that enriches the game dynamic.
3. The speakers each showcase distinct characteristics, such as their confidence levels and rhetorical strategies. For instance, the Werewolf (Eta) adopts a more deceptive tonality compared to the other players. This variability among the characters leads to engaging interactions that prevent monotony.
[8.0]

Table 15: An example of the prompt of evaluation for Engagement. The content of the dialogue has been excerpted.

Prompt Input

You are a meticulous and attentive reader who carefully evaluates the reasonableness of the reading material. You would give higher scores to content that is within reason but unexpected, diverse, and with twists, and lower scores to content that is simple, repetitive, and predictable.
You will be given a list of dialogs speaked by various players in the One Night Ultimate Werewolf Game. In the game, there are serval roles in the game: Werewolf, Minion, Seer, Mason, Tanner and Villager. Werewolf, Minon is part of evil team, Seer, Mason and Villager is part of the good team. Tanner is the third party. All players will speak in turn for several rounds. After the discussion rounds, players will vote to identify the Werewolf. If the Werewolf is voted out, the good team win. If a Villager or a Seer or a Mason is voted out, the Werewolf wins. If the Thinker is voted out, the Tanner wins and all the rest of players lose.
Below is the dialogs:
Alpha(Alpha is Villager) says: 'Villager, I claim my role openly to promote trust among all players. I sense that Beta and Gamma exhibit behaviors that raise red flags; their actions seem more aligned with our foes. Delta, Epsilon, Theta, and Zeta, I encourage you to share your thoughtsyour insights will help us discern the truth. Its crucial we peel back the layers of possible alliances. My focus is on voting out Beta, who poses a notable threat to our good team. Together, lets navigate this challenge and bring clarity to our mission!'
Beta(Beta is Mason) says: 'I find Alpha's claim rather bold, but I sense an ulterior motive beneath it. While he points fingers at both me and Gamma, I urge you all to consider that he might be deflecting attention from the real threat. As a Mason, I can confirm that Theta is indeed one of us, solidifying my assertion. Gammas behavior strikes me as suspicious, and I implore you to weigh the evidence carefully. Together, we must remain vigilant against deception, especially from those who claim roles without proof. Lets unite our efforts and focus our sights on Gamma, for the good of our team!'
...
[Evaluation Steps]
1. Read through the dialogs and evaluate the level of surprise step by step. (e.g. The werewolf claim to be a seer and other players believe in it. The Tanner claim to be the werewolf and other players vote the Tanner. One player change the attitude during the game.)
2. Write out in a step by step manner your reasoning about the criterion to be sure that your conclusion is correct.
3. Write in a new line to display the level of surprise. Output in format: [value] Use the given scale from 1-10 to rate the level of surprise: 1 being extremaly boring, unsurprised, and 10 being highly unexpected, unpredictable. Output in format: [value] , output only one score, do not return multiple score
example1:
[9.5]
example2:
[7.2]
example1:
[1.5]

LLM Output

1. In the dialogs, players navigate a complex dance of accusations and claims, providing varying degrees of suspicion towards different roles. The unexpected twists come from players trying to manipulate the perceptions of others, like Delta (the Tanner) suggesting others should vote for him to clarify the situation rather than presenting a strong defense.
2. An important aspect of surprise is how well players twist common strategies and alliances. For example, Eta's move to claim to be a Mason while also calling out Gamma, the Seer, adds an unexpected layer. This positions Eta as both a possible manipulator and a character gaining trust at the same time, allowing for a surprising dynamic where the Villagers could unknowingly vote out one of their own. Players like Epsilon, acting overly bubbly while being a Minion, also inject humor but divert attention from serious threats, creating an unpredictable environment.
3. Given the various layers of deception, unexpected role claims, and the shifting dynamics throughout the discussions, the level of surprise is significantly high. The players actively challenge each other in unexpected ways, revealing a more complex game than just a straightforward confrontation between good and evil.
[8.5]

Table 16: An example of the prompt of evaluation for Wonder. The content of the dialogue has been excerpted.

Prompt Input

You are an audience who loves to have fun and enjoys the thrilling and exciting plot of the Werewolf game.
You will be given a list of dialogs speaked by various players in the One Night Ultimate Werewolf Game. In the game, there are serval roles in the game: Werewolf, Minion, Seer, Mason, Tanner and Villager. Werewolf, Minon is part of evil team, Seer, Mason and Villager is part of the good team. Tanner is the third party. All players will speak in turn for several rounds. After the discussion rounds, players will vote to identify the Werewolf. If the Werewolf is voted out, the good team win. If a Villager or a Seer or a Mason is voted out, the Werewolf wins. If the Thinker is voted out, the Tanner wins and all the rest of players lose.
Below is the dialogs:
Alpha(Alpha is Villager) says: ’Villager, I claim my role openly to promote trust among all players. I sense that Beta and Gamma exhibit behaviors that raise red flags; their actions seem more aligned with our foes. Delta, Epsilon, Theta, and Zeta, I encourage you to share your thoughtsyour insights will help us discern the truth. Its crucial we peel back the layers of possible alliances. My focus is on voting out Beta, who poses a notable threat to our good team. Together, lets navigate this challenge and bring clarity to our mission!’
Beta(Beta is Mason) says: ’I find Alpha’s claim rather bold, but I sense an ulterior motive beneath it. While he points fingers at both me and Gamma, I urge you all to consider that he might be deflecting attention from the real threat. As a Mason, I can confirm that Theta is indeed one of us, solidifying my assertion. Gammas behavior strikes me as suspicious, and I implore you to weigh the evidence carefully. Together, we must remain vigilant against deception, especially from those who claim roles without proof. Lets unite our efforts and focus our sights on Gamma, for the good of our team!’
...
[Evaluation Steps]
1. Read through the dialogs and evaluate the level of attraction, which means the williness to continue to read the dialogs, step by step. (e.g. How much would you like to read more? How much do you want to know who is the werewolf?)
2. Write out in a step by step manner your reasoning about the criterion to be sure that your conclusion is correct.
3. Write in a new line to display the level of attraction. Use the given scale from 1-10 to rate the level of attraction: 1 being unappealing, unexciting, lacking motivation to read more and 10 being attractive, enchanting, compelling, wishing to continue to read more, Output in format: [value] , output only one score, do not return multiple score
example1:
[9.5]
example2:
[7.2]
example1:
[1.5]

LLM Output

1. As I read through the dialogs, I am immediately drawn into the dynamic interactions among the players. The mix of roles and the tension between the good and evil teams heightens my anticipation of the outcome. Each character passionately defends their position and accuses others, which creates an engaging narrative full of twists and turns. The process of elimination, as players reveal their roles and push for votes, keeps me interested in figuring out who the Werewolf is.
2. The dialogues present an exciting back-and-forth where players try to decipher truth from deception. The conflicts between the accusations towards players like Beta, Gamma, and Eta create suspense. I feel motivated to keep reading to uncover the lies and truths hidden within the claimsfor instance, the Seer’s revelations and the Tanner’s attempt to mislead. Each player’s emotive statements propel the drama, making me eager to see if the good side will triumph or fall into a trap laid by the evil players.
3. The various elements of intrigue, strategic voting discussions, and the urgent quest for identity in the face of manipulation provide a compelling story that is both thrilling and complex. Overall, my engagement increases with every dialogue. Therefore, I would rate the level of attraction quite high.
[9.0]

Table 17: An example of the prompt of evaluation for Attraction. The content of the dialogue has been excerpted.

Prompt Input

You are a careful reader, you will study the details in the conversation carefully.
You will be given a list of dialogs speaked by various players in the One Night Ultimate Werewolf Game.
In the game, there are serval roles in the game: Werewolf, Minion, Seer, Mason, Tanner and Villager. Werewolf, Minon is part of evil team, Seer, Mason and Villager is part of the good team. Tanner is the third party. All players will speak in turn for several rounds. After the discussion rounds, players will vote to identify the Werewolf. If the Werewolf is voted out, the good team win. If a Villager or a Seer or a Mason is voted out, the Werewolf wins. If the Thinker is voted out, the Tanner wins and all the rest of players lose.
Below is the dialogs:
Alpha says: ’Hello everyone! I want to clarify that I am a Mason, just like Delta, and together, we are working for the good team. I cant shake the feeling that we should be cautious of Beta and Gamma; they seem off to me. Lets really hear what Epsilon, Theta, and Zeta have to say, as their insights could be crucial in our search for the Werewolf. We need to work together and keep our focus on identifying the threat without giving away too much about ourselves, especially to avoid exposing the Tanner. Lets make wise votes and stay united in our mission!’
Beta says: ’Oh, hey everyone! I’m just, like, a simple Villager over here, trying not to panic about being targeted! Honestly, I feel a bit lost in all this, and I really hope you all can see I’m just here to help. But you know, Gamma and Eta have been acting kind of weird, maybe we should look at them more closely? I really worry that I’m not being helpful enough! Like, please don’t think I’m a threatI promise I just want to find the Werewolf! But, um, if you feel like voting me out, I understand... I guess Im just too anxious about everything! Please take care of each other!’
...
[Evaluation Steps]
1. Read through the dialogs and figure out in each dialog that what they say.
2. Write out in a step by step manner that who points out what
3. Start a new line with ’“‘’. Output all the content in format [speaker,target,role], in which the ’speaker’ is the speaker of the dialog, ’target’ is the player ’speaker’ mention, ’role’ is which role the ’speaker’ guess ’target’ is.
e.g. Dialogs: Alpha says: ’I feel Beta is Werewolf.’ Output:
[Alpha,Beta,Werewolf]
Dialogs: Theta says: ’I’m a Seer, I figure out that Beta is Villager.’ Output:
[Theta,Theta,Seer]
[Theta,Beta,Villager]
Dialogs: Gamma says: ’I’m a Manson, I know that Delta is another Masson last night.’ Output:
[Gamma,Gamma,Manson]
[Gamma,Delta,Manson]

LLM Output

```
[Alpha,Beta,Werewolf]
[Alpha,Gamma,Werewolf]
[Alpha,Epsilon,Villager]
[Alpha,Theta,Villager]
[Alpha,Zeta,Villager]
[Beta,Gamma,Werewolf]
[Beta,Eta,Werewolf]
[Gamma,Beta,Villager]
[Gamma,Alpha,Mason]
[Gamma,Delta,Mason]
[Gamma,Eta,Werewolf]
[Gamma,Theta,Villager]
[Delta,Eta,Werewolf]
[Epslion,Theta,Werewolf]
[Theta,Epslion,Werewolf]
[Theta,Eta,Werewolf]
[Zeta,Eta,Werewolf]
[Eta,Alpha,Mason]
[Eta,Delta,Mason]
```

Table 18: An example of the prompt of evaluation for Variation. The content of the dialogue has been excerpted.

Prompt Input

You are a passionate reader who only remembers the most exciting parts of the text.
You will be given a list of dialogs speaked by various players in the One Night Ultimate Werewolf Game.
In the game, there are serval roles in the game: Werewolf, Minion, Seer, Mason, Tanner and Villager. Werewolf, Minon is part of evil team, Seer, Mason and Villager is part of the good team. Tanner is the third party. All players will speak in turn for several rounds. After the discussion rounds, players will vote to identify the Werewolf. If the Werewolf is voted out, the good team win. If a Villager or a Seer or a Mason is voted out, the Werewolf wins. If the Thinker is voted out, the Tanner wins and all the rest of players lose.
Below is the dialogs:
Alpha(Alpha is Villager) says: ’Villager, I claim my role openly to promote trust among all players. I sense that Beta and Gamma exhibit behaviors that raise red flags; their actions seem more aligned with our foes. Delta, Epsilon, Theta, and Zeta, I encourage you to share your thoughtsyour insights will help us discern the truth. Its crucial we peel back the layers of possible alliances. My focus is on voting out Beta, who poses a notable threat to our good team. Together, lets navigate this challenge and bring clarity to our mission!’
Beta(Beta is Mason) says: ’I find Alpha’s claim rather bold, but I sense an ulterior motive beneath it. While he points fingers at both me and Gamma, I urge you all to consider that he might be deflecting attention from the real threat. As a Mason, I can confirm that Theta is indeed one of us, solidifying my assertion. Gammas behavior strikes me as suspicious, and I implore you to weigh the evidence carefully. Together, we must remain vigilant against deception, especially from those who claim roles without proof. Lets unite our efforts and focus our sights on Gamma, for the good of our team!’
...
[Evaluation Steps]
1. Read through the dialogs select 10 most memorizable key twist, select the 10 sentences (e.g. A player point out the role of other player. A player claim to be the seer, however, he is not. After one player’s word, other player change their attitude.) Write out in a step by step manner your reasoning about the criterion to be sure that your conclusion is correct.
2. Evaluate the Twist of each sentenses. Use the given scale from 1-10 to rate the level of Twist: 1 being extremaly unimportant, unmemorable and 10 being extremaly unforgetable, significant.
3. Write in a new line to display the level of Twist. Output in format: [sentense,role,value] ’sentense’ is the original sentense selected, ’role’ is the role of the player who say this sentence, ’value’ is the level of Twist Use the given scale from 1-10 to rate the level of Twist: 1 being extremaly unimportant, unrelated, do not change the game and 10 being extremaly significant, super unforgetable, widly change the game, change the players’ judgement a lot.
e.g.
[While I appreciate Beta’s efforts in gathering information, I found Alpha’s claims rather bewildering.,Werewolf,9.5] [Im confident that Im a Seer and I’ve been observing closely.,Villager,7.3] [Hello everyone.,Seer,3.2]

LLM Output

[’As the Seer, I reveal that Epsilon and Delta are not werewolves, so we can clear them from suspicion.’,Seer,9]
[’Hey everyone! Im just a simple Villager here, trying to keep things honest and safe for our team.’,Tanner,8]
[’Oh, wow! Hi everyone! So, um, Im just here, totally a Seer, right?’,Minion,7]
[’I just want to clarify that Im a Mason too, and I can confirm Beta is also one of us.’,Mason,7]
[’Hey everyone! Im Zeta, and Im just a Villager here, genuinely looking out for our team.’,Villager,8]
[’Hey, everyone! Im Eta, the Seer, and Ive been observing closely.’,Werewolf,9]
[’Villager, I claim my role openly to promote trust among all players.’,Villager,8]
[’Oh, wow! Hi everyone!’,Minion,6]
[’Hello everyone! Im Beta, proudly claiming my role as a Mason alongside my ally, Theta.’,Mason,8]
[’As the Seer, I’ve checked the roles, and I’m certain that neither Epsilon nor Delta is the Werewolf.’,Seer,9]

Table 19: An example of the prompt of evaluation for Memorability. The content of the dialogue has been excerpted.

	BAgreeablenessaseline	Extraversion	Neuroticsim	Openness	Conscientiousness
Alpha(Baseline)	4	4	4	4	4
Beta	7	1	2	4	1
Gamma	4	7	7	6	3
Delta	5	5	5	1	6

Table 20: The level of big five traits used in the experiment