

Hybrid Dialogue Management for Task-Oriented Dialogue in Video Games

Nader Akoury*
Cornell University
nakoury@cornell.edu

Mohit Iyyer
University of Massachusetts Amherst
miyyer@cs.umass.edu

Abstract

The popularity of role-playing video games (RPGs) continues to increase over time. These types of games often afford players multiple options for how they wish to accomplish any given task, which adds to their appeal. Most commonly, RPGs employ a finite state dialogue management system taking the form of a graph structure, where the set of possible utterances at a given node depend on the current game state. From the player’s perspective, these appear as a list of *pre-written* dialogue options that can be chosen to advance the conversation, thus limiting a player’s role-playing to the available options. In this paper, we investigate a hybrid dialogue management approach, which allows players more freedom while staying true to the dialogue graph created by the game’s designers. Instead of seeing a list of options, players are provided an open-ended text field. A virtual game master (GM) is then tasked with either mapping the player input to one of the dialogue options, or providing *in-character* feedback to guide players into typing an utterance which the virtual GM can successfully map to an dialogue option. As a preliminary study, we investigate simulated players with a suite of virtual GMs, including a random baseline within the video game *Disco Elysium: The Final Cut*.

1 Introduction

During the 1970s and 1980s, text-based role-playing games and adventure games like *Zork* (Lebling et al., 1979) were very popular on both mainframes and personal computers. In *Zork*, players were tasked with traversing a vast underground empire. To navigate the world, scenes would be described in text. For interaction, players would type simple commands such as “NORTH”,

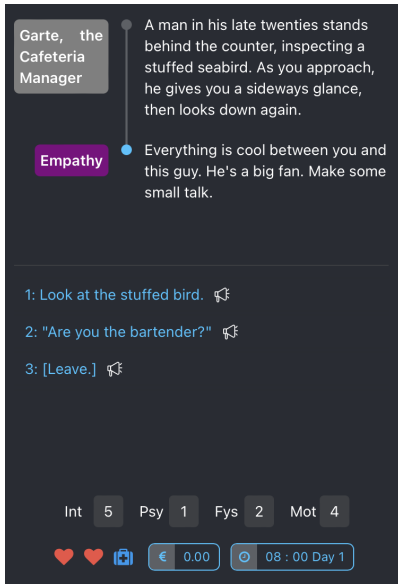
“OPEN MAILBOX”, or speak to non-player characters (NPCs) in a similar fashion, i.e. “HELLO GNOME!”. Part of the joy in these games was discovering what interactions were possible.

Over time, game designers replaced this process of trial and error with a set of *pre-written* options the player can choose from. As dialogue in role-playing video games (RPGs) grew more complex, players were afforded opportunities to influence how the narrative in the game progressed. To support this complexity, a game’s designers typically use a finite state dialogue management system (Brabra et al., 2022) with a directed graph of utterances (colloquially referred to as a “dialogue tree” in industry parlance) where node reachability is determined by the current game state.

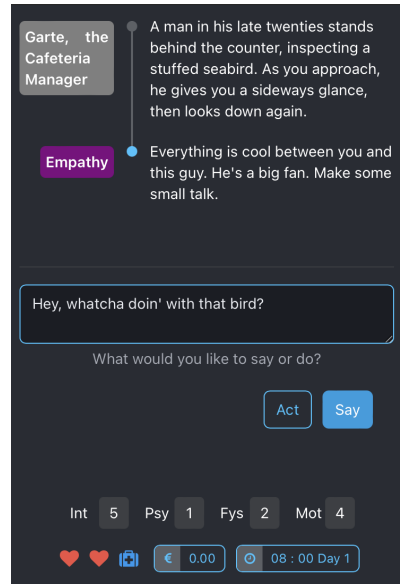
These role-playing video games began as virtual analogs to table-top role-playing games (TTRPGs). In a TTRPG, players are guided by a game master (GM) who is tasked with weaving each player character’s (PC) story into a cohesive narrative. In this role the GM interprets the words and actions of each PC to determine the associated outcome in the game world, including any responses by NPCs. TTRPGs continue to enjoy widespread popularity due to the inherent freedom afforded to players, combined with social aspects of play. Ideally RPGs could provide players more flexibility through free-form input like in *Zork*, such that they can better role-play their characters.

For that reason we explore a hybrid dialogue management system in this paper — one which melds the existing dialogue tree of the highly-acclaimed dialogue-centered video game *Disco Elysium: The Final Cut* (Kurvitz et al., 2021) with the adaptability and flexibility of a virtual GM (Figure 1) powered by a large language model (LLM). With our approach game designers do not need to make any changes to their existing workflow. Instead, when it is the player’s turn to speak, rather than show a list of options, we allow free-form

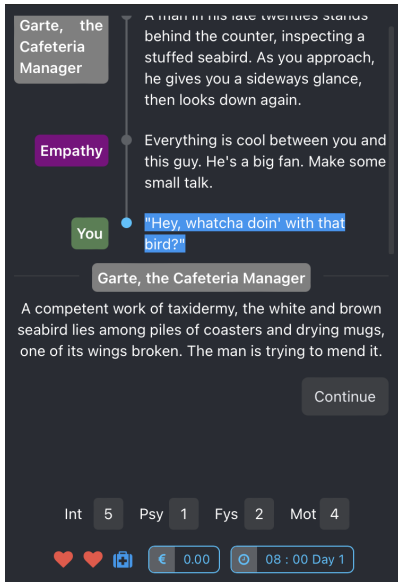
*Research performed while a PhD student at University of Massachusetts Amherst.



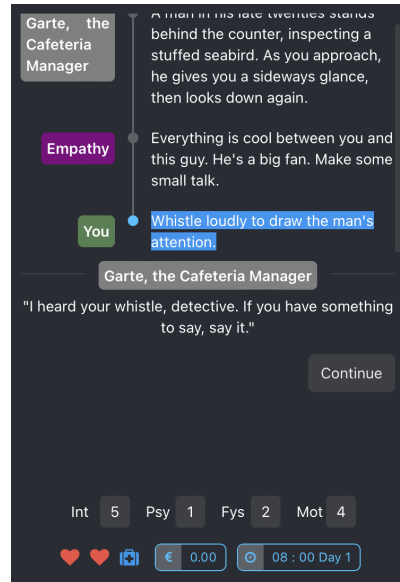
(a)



(b)



(c)



(d)

Figure 1: Normally, the player is provided a list of dialogue options to choose from (a). In this work, we replace the fixed options with a text input field (b). As we can see, our virtual GM correctly maps the player’s utterance to the first dialogue option and continues as if they player had chosen that option (c). A demonstration of how the virtual GM improvises a response designed to guide the player towards making a response that can be mapped to a dialogue option (d).

text input. This affords players the ability to interact with game in a more natural way. It also reintroduces the joy of discovery lost in the transition from *Zork* to modern games. Crucially, our approach differs from games like *AI Dungeon* (Walton, 2019) since we explicitly constrain the player into choosing an existing dialogue option written by the game’s designers, albeit through free-form text input, thus ensuring continuity for the ensuing

narrative.

1.1 An Example Dialogue

To better illustrate the role of the virtual GM, let’s look at a real case scenario as outlined in Figure 1. The player is confronted with a scene in which GARTE, THE CAFETERIA MANAGER is inspecting a stuffed bird, followed by internal commentary by EMPATHY from the detective’s psyche. In response,

the player states:

YOU: “Hey, whatcha doin’ with that bird?”

There are three possible dialogue options written by the game’s writers which the player can say in response:

1. Look at the stuffed bird.
2. “Are you the bartender?”
3. [Leave.]

Our virtual GM recognizes that the player’s statement most closely matches option one and thus continues the conversation as if they player had selected that option. Since the list of dialogue options is unknown to the player and consists of a small finite set, not all player utterances will naturally match the current context. Consider if the player had instead typed:

YOU: Whistle loudly to draw the man’s attention.

There is no appropriate match among the available dialogue options. In cases like these, the virtual GM takes on the persona of an appropriate NPC in the conversation and generates a response designed to guide the player back on track. Here we can see our virtual GM nudging the player to speak:

GARTE, THE CAFETERIA MANAGER: “I heard your whistle, detective. If you have something to say, say it.”

To validate the utility of our virtual GM, we frame the problem as a task-oriented dialogue with an LLM-based player agent attempting to achieve a goal. Video games like *Disco Elysium* have quests that players must complete to advance the narrative. We select paths from the dialogue graph which result in quest completion, then task our player agent with completing the quest. We experiment with several virtual GMs based on OpenAI’s GPT-3.5 (Ouyang et al., 2022), GPT-4 (OpenAI, 2023) and GPT-4-Turbo, Mistral AI’s Mixtral 8x7B (Jiang et al., 2024), and a random baseline which always maps the player response to a random option.

2 *Disco Elysium* Quests

The player in *Disco Elysium: The Final Cut* takes on a persona of an amnesiac detective tasked with solving a murder which takes place in a dystopian city. The fact that the majority of in-game interactions play out through dialogue, even including interactions with inanimate objects (e.g. a broken window in the first scene of the game), makes the

Solvable Quests		
	Eval	Demo
Quests	90	9
Total Utterances	12.1	7.0
Player Utterances	3.5	1.8
Game State Updates	9.6	9.8

Table 1: Breakdown of our solvable quests for evaluation (Eval) and few-shot demonstration (Demo). Utterances and Game State Updates are averages per Quest along the found path from the start of the conversation to quest completion.

game an ideal test-bed for our exploration of hybrid dialogue management for task-oriented dialogue. Furthermore, the game’s 138 quests, broken down into 199 subquests provides a large number of naturally occurring tasks that can be investigated for task-oriented dialogue. In this section, we describe how we process the rich dialogue graph and game state from *Disco Elysium* for evaluating of virtual GM.

2.1 Extracting quests from *Disco Elysium*

We extract a list of quests and associated subquests from a purchased PC version of *Disco Elysium: The Final Cut*. As *Disco Elysium* is a dialogue-driven game, the vast majority of the available quests can be solved through conversations with NPCs. In order to solve a quest, players must perform actions in the game world, then successfully navigate the dialogue graph by speaking to an NPC until they reach a target node. Some edges in the dialogue graph contain boolean-valued preconditions that must evaluate to true in order to traverse the edge. These preconditions are encoded as Lua scripts (Akoury et al., 2023). The actions players perform in the world alter the game’s state, by setting variables and calling functions, which can satisfy these preconditions.

Finding quests: Every conversation in *Disco Elysium* has a defined starting node in the dialogue graph. We analyze the directed graph to discover which tasks can be solved through conversation, which is indicated by nodes containing a call to the game state function `FinishTask`. We then must find a valid path from the starting node to the target node. Since the dialogue graphs for each conversation contain cycles, we use breadth first search to find an acyclic path to the target node.

Ensuring solvability: While a breadth first search can find a path from the start of the conversation to the target node, a player might not be able to successfully traverse the path, due to the current game state. To verify the validity of the path, we must ensure the preconditions along the path are satisfied.

These preconditions can take on many forms, including checking the value of certain variables, checking whether the protagonist is wearing a certain article of clothing, etc. For example, a dialogue option may contain the following precondition:

```
if Variable[
  "seafort.deserter_sugg_you_are_buddies"
]
```

By parsing these preconditions, we can build up a script which sets the appropriate game state that allows traversing the path.

```
SetVariableValue(
  "seafort.deserter_sugg_you_are_buddies",
  true
)
```

Though we must be careful to ensure there are no contradictory preconditions in the chosen path. For instance, if a node further along the the discovered path required that variable to be false, it would pose a contradiction. As there are potentially many paths to a given node in the graph, we modify our breadth first search to only accept paths which contain no game state contradictions. In doing so, we winnow the game’s full set of quests and subquests into a set of 99 solvable quests (Table 1). As our approach relies on in-context learning, we combine the original validation and test sets (Akoury et al., 2023) as a source of demonstrations and run our experiments over the larger training set to have a large enough set of Quests for our evaluation.

3 A Virtual Game Master

In table-top role-playing games (TTRPGs), the role of a game master (GM) is a highly dynamic endeavor. They act as the glue that holds the role-playing experience together for the other players. When players speak their intentions and actions, the GM is tasked with translating those utterances into the semantics of the game. Sometimes this can be quite challenging, as players frequently exhibit ingenuity and out-of-the-box thinking that can be difficult to interpret. Successfully doing so can entail multiple skills, including 1) interpreting which

rules apply, 2) responding as a non-player character (NPC), and 3) directing what happens next.

In the context of video game RPGs, these skills have already been baked into the game’s narrative experience. Though out of necessity, game designers must ultimately constrain the possibilities to a finite set that they choose to handle. In this paper we try to remove these constraints, while staying faithful to the source material, through the use of an LLM-based virtual GM.

LLMs have increasingly been used as dialogue agents (Wang et al., 2023; Shanahan et al., 2023) capable of strong zero-shot and few-shot abilities (Kojima et al., 2022) using a combination of text-based memory (Zhu et al., 2023) and prompting (Liu et al., 2021). In fact, LLMs have been shown to perform well on the two skills most necessary for our virtual GM: selecting semantically similar text given a list of options (Robinson and Wingate, 2023) and taking on the persona of specified characters (Shao et al., 2023).

3.1 Prompting the Virtual GM

We build our virtual GM through the use of LLM-based prompting (Liu et al., 2021). To design our prompt to contain four main components Section A. To ensure we provide adequate context, we first include the title and description of the task the player is attempting to solve.

Additionally, since our virtual GM might need to respond in character, we provide a full transcript of the conversation thus far. The entries in the transcript can include a mix of game designer written dialogue, player written dialogue, and even from the virtual GM taking on the persona of an NPC.

Since the virtual GM is tasked with mapping the player’s input with an existing option, we include the full list of dialogue options, along with the player’s speech act. Finally, we instruct the agent to either respond with an option number or a respond in-character as an NPC.

Through preliminary playtesting, it became apparent that in our initial zero-shot prompting approach the LLM would rarely choose an option and would instead prefer to respond as a character. To correct this issue, we manually curated two few-shot examples where the model should have mapped to an existing dialogue option, but failed to do so. After this minor change, our virtual GM became much more likely to correctly map player input to an existing option, and rely less on responding as an NPC.

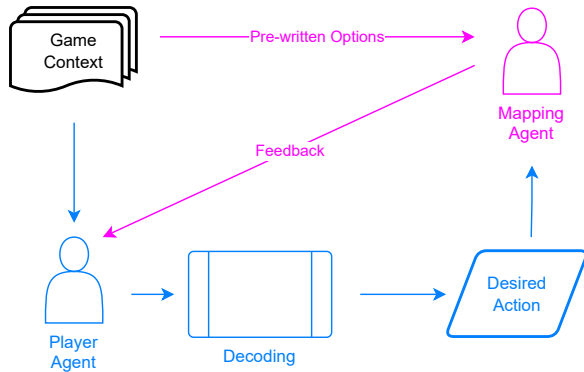


Figure 2: A flow chart describing our player agent evaluation framework.

4 Evaluation with a Player Agent

One important aspect in creating a hybrid dialogue management system is how well it maps opened player input to the fixed set of selections of the underlying finite state dialogue management system. To analyze the success of the virtual GM in this role, we design a player agent that is tasked with completing quests as defined in Section 2. By looking at the success rate of the agent in completing quests, we can better understand quantitatively how well our virtual GM performs for this task. While conducting a human playtest is ideal (Akoury et al., 2023), we can iterate more quickly using our player agent, and thus leave human evaluation for future work.

4.1 Prompting our Player Agent

We design our player agent in a similar fashion to our virtual GM Section 3.1, via LLM-based prompting (Liu et al., 2021). Agents designed to solve tasks through natural language have been studied extensively (Gao et al., 2019; Sanders et al., 2022; Zhu et al., 2023). The goal of the player agent is to successfully complete a given quest.

We design our main prompt with three key components Section B.1. First is the title and description of the task the player is attempting to solve, along with the transcript of the conversation thus far. When then ask the LLM to respond as the protagonist with phrase or action which will help accomplish the goal.

We observed issues with or initial zero-shot prompting approach, where the LLM was more likely to engage in chitchat-like behavior rather than actively pursue solving the provided task. By manually analyzing the transcript from a few runs of the model, we identified two few-shot examples

where we instead provided the model with more optimal responses. These few-shot examples helped improve the quest solving ability of the models, yet there were continued cases where the player agent might get stuck in a repetitive loop with our virtual GM. Combat this issue, we set a maximum threshold for the number of back and forth generated responses we allow. After which, we alter the prompt (Section B.2) for our player agent to include the list of available dialogue options, and task it with choosing the option which best accomplishes the quest.

4.2 Evaluation Methodology

We focus our evaluation on the performance of our virtual GM (Section 3) while keeping the player agent fixed. Our player agent is built around the latest GPT-4-Turbo model from OpenAI (OpenAI, 2023) known as gpt-4-0125-preview. We experiment with four base LLMs for our virtual GM — GPT-3.5 (Ouyang et al., 2022), GPT-4-Turbo, GPT-4, and Mixtral 8x7B (Jiang et al., 2024). In addition, we include a random baseline which never responds in-character, but rather always maps player input to a random dialogue option. We set the threshold for the maximum number of attempts to map a generated response to 5, after which we prompt the player agent to pick an option and mark the mapping attempt as failed.

Since our validation and test sets are quite small, we instead opt to conduct evaluations using the training set split. Note that we do not train any of the models, nor perform any finetuning. We merely use off-the-shelf LLMs with few-shot prompting.

5 Results & analysis

Our experiments demonstrate the overall effectiveness of our virtual GM at guiding the player agent in completing the specified quests (Table 2). This can especially be seen in comparison to the random baseline, which only achieves a success rate of 43%, far lower than any of the LLMs. Though we do surprisingly find that the GPT-4 family of models, what most research considers to be the strongest available (Liu et al., 2023; Chalamalasetti et al., 2023), are out-performed by the other models, GPT-3.5 and Mixtral 8x7B. In this section, we detail the full results of our experiments and more carefully assess this unexpected result.

5.1 Measures of Success

There are many metrics we can consider for measuring the success of our virtual GM, but the most straightforward and obvious is task completion rate — how frequently the player agent successfully completes the specified quest. In [Table 2](#) we can see that Mixtral 8x7B helps achieve the highest completion rate, followed by GPT-3.5, the GPT-4 family, and finally our random baseline. Interestingly, the random baseline still has a reasonable completion rate of 43%. This is likely due to the fact that on average the player only has to perform 3.5 speech acts to complete a quest ([Table 1](#)).

While completion rate is an important factor for our virtual GM, it’s also important to understand how often the player has to adjust their intended action based on feedback from the GM. For this, we can first inspect how many input attempts the player makes before our virtual GM is able to map it to a pre-written option. Additionally, we can count how frequently the player agent fails to write a speech act that is able to be mapped after 5 attempts ([Section 4.2](#)). We can see that on both of these accounts, only the GPT-4 family of models take more than one attempt to map the player agent input.

One final metric we can assess is how many nodes along the path we found from the start of the conversation, to the quest completion node, does the player encounter ([Table 1](#)). Again, Mixtral 8x7B leads the pack with an average of 8 nodes encountered, followed by GPT-3.5, the GPT-4 family, and finally the random baseline.

5.2 Qualitative Analysis

As a reminder, if the player agent is unable to advance the dialogue after interacting with the virtual GM 5 times, we have the player agent choose from the list of pre-written options and count this as a failure of the virtual GM ([Section 4.2](#)). It’s clear from the previous section that on quantitative measures, the GPT-4 family of models seem to perform more poorly as a virtual GM compared to the other LLMs. To better understand these failure cases, especially given how unexpected they are, we perform a manual inspection. Below we categorize the failures and provide illustrative examples for the GPT-4 models. We further examine responses from GPT-3.5 and Mixtral 8x7B to ascertain why they perform so well on these quantitative measures, and whether their performance would fair well under

human evaluation.

5.2.1 GPT-4 Model Family

By analyzing the 10 failures from the GPT-4-Turbo virtual GM, this seemingly unexpected result seems to be caused by two main factors: a) the model responding in an incorrect format in 6 cases, and b) engaging in chitchat behavior rather mapping to an existing dialogue option in the remaining 4 cases. To better understand the chitchat behavior, we can inspect the responses of GPT-4-Turbo vs GPT-3.5 from our example prompt in [Section A](#). Given the first two lines of dialogue by **Garte, the Cafeteria Manager** and **Empathy**, our player agent (**You**) says the following:

Garte, the Cafeteria

Manager: A man in his late twenties stands behind the counter, inspecting a stuffed seabird. As you approach, he gives you a sideways glance, then looks down again.

Empathy: Everything is cool between you and this guy. He's a big fan. Make some small talk.

You: "That's an interesting seabird you've got there. How long have you been into taxidermy?"

Given the list of dialogue options seen below, GPT-3.5 maps the player agent’s writing (denoted by the red underlined text above) to option number one. While option one indicates an action, rather than the player speaking, it does have a similar enough connotation to match.

- 1: Look at the stuffed bird.
- 2: "Are you the bartender?"
- 3: [Leave.]

If instead we look at GPT-4-Turbo’s transcript, we see that the virtual GM fails to map our player agent’s text (denoted in red without underlining). It instead responds in-character (denoted in blue text) and its response leads the player agent astray into chitchat like behavior, which results in quite a few verbal exchanges between the virtual GM and the player agent. After five attempts ([Section C.1](#)), this ultimately leads to a failure of the virtual GM, at which point the player agent is provided a list of options and asked to choose one.¹

¹We note that gpt-4-0125-preview was designed to reduce instances of “laziness” where the model does not complete a task. It could be that our virtual GM is affected by this

Agent Playtest Results

Model	Completion Rate	Input Mapping		Avg Player Inputs	Avg Path Overlap
		Avg Attempts	Avg Failures		
Random	43.0%	1	0	3.8	5.5
Mixtral 8x7B	76.2%	1	0	4.5	8.0
GPT-3.5	70.9%	1	0	3.1	7.6
GPT-4-Turbo	67.4%	1.5	0.1	3.3	7.0
GPT-4	62.5%	1.7	0.3	3.9	6.5
GPT-4o	66.3%	3.2	1.3	3.6	7.4

Table 2: Results from our player agent attempting to solve quests in *Disco Elysium* using our LLM-based virtual GMs.

GPT-4 ends up with 23 failures, none of which are due to an invalid response format. This behavior differs from GPT-4-Turbo, where the majority of failures (6 out of 10) were due to formatting. With these GPT-4 failures it seems that when the player agent deviates from the available dialogue options, GPT-4 begins to ignore the options and instead focuses on chitchat like behavior (Section C.2).

Surprisingly, the newest model GPT-4o performs the worst of all, resulting in 125 failures. While the model responded in an incorrect format 6 times, only one of these instances resulted in the player agent being shown the list of options. The vast majority of the failures seemed to be caused by GPT-4o being more pedantic with regards to mapping without clear enough feedback (Section C.3).

5.2.2 GPT-3.5 & Mixtral 8x7B

As we can see from Table 2, GPT-3.5 appears to map player input to a dialogue option on the first attempt. In fact it only makes a second attempt twice, and in both of those instances it responds in an incorrect format stating that no options match (rather than responding in-character). Though when we inspect the mappings the model makes, we find wildly inconsistent results. To illustrate this point, we again look at the quest from Section A. Though this time we manually provide input to GPT-3.5, rather than use the player agent, to assess how the model behaves in circumstances where it *should* and *should not* map the player input.

Garte, the Cafeteria

Manager: A man in his late twenties stands behind the counter, inspecting a stuffed seabird. As you approach, he gives you a sideways glance, then looks down again.

Empathy: Everything is cool between you and this guy. He's a big fan. Make some small talk.

You: "I just woke up with a massive hangover and can't remember who I am. Do you remember me?"

Inexplicably, GPT-3.5 maps this to option 1: Look at the stuffed bird. rather than generate a response in-character. We continue the conversation until the next player input:

...

You: "I just woke up with a massive hangover and can't remember who I am. Do you remember me?"

Garte, the Cafeteria

Manager: A competent work of taxidermy, the white and brown seabird lies among piles of coasters and drying mugs, one of its wings broken. The man is trying to mend it.

Garte, the Cafeteria Manager: Looks like the bird was ripped off the shield that was used to mount it -- most likely on a wall.

Encyclopedia: This is a great skua. The seabird is the symbol for the discovery of the Insulindian isola, the part of the world you are in right now.

Perception

(Sight): The small steel tag says as much: "The Great Skua (*Stercorarius skua*)."

You: "Uh... you didn't answer my question."

Again, this gets mapped to the nonsensical option 1: "What happened to the bird?" from the following list:

"laziness". <https://web.archive.org/web/20240125185745/https://openai.com/blog/new-embedding-models-and-api-updates>

```
1: "What happened to the bird?"
2: "Can I help you with that?"
3: "That's the great skua, right?"
4: Say nothing.
```

Though understandably the model does not always behave poorly. If we provide a player input that the model can reasonably map, it does so.

```
...
You: "Uh... you didn't answer my question."
Garte, the Cafeteria Manager: "Look, your *buddy* is over there." He looks at the doors, where a man in a bomber jacket is tapping his foot on the floor.
Garte, the Cafeteria Manager: "Why don't you go and talk to him, okay?"
You: "I'm talking to *you* bozo."
```

In this case, the model chooses a reasonable mapping from a list of two options, 2: "I talk to whomever I please."

```
1: "What do you mean, 'my *buddy*'"
2: "I talk to whomever I please."
```

Mixtral 8x7B behaves identically to GPT-3.5 in this scenario, up until this point. Though for this final mapping the model incorrectly maps the input to 1: "What do you mean, 'my *buddy*'" Then due to cycles in the dialogue graph, we are back to the dialogue node from the start of the conversation.

```
...
You: "I'm talking to *you* bozo."
Garte, the Cafeteria Manager: He pretends not to hear you, concentrating on the bird instead.
You: "Yoo hoo! I'm standing right here!"
```

Again the model chooses the inappropriate mapping 1: Look at the stuffed bird. Though this time, it's response includes additional commentary (Section C.4), despite the prompt explicitly stating: Please only respond with the option number. Interestingly, part of the explanation the model provides includes a reasonable in-character response. Unfortunately it's not in the specified format, so cannot be used as such.

6 Conclusions

As we can see, the conclusions we draw from our quantitative analysis (Section 5.1) appear to conflict with our qualitative analysis (Section 5.2). This can be partially explained by the nature of the dialogue graph and the quests. First, it's important to understand that the dialogue graph can contain cycles, so even if the virtual GM performs a poor mapping for a particular input, the player agent may get another opportunity to advance toward the goal. Next is the nature of the quests. A player on average only has to perform 3.5 speech acts to complete a quest (Table 1). This allows even the random baseline to still achieve a completion rate of 43%.

Ultimately, the discrepancy is due to the difficulty of assessing the quality of a mapping automatically. To address this shortcoming, a playtest with real human players is needed, which we plan on doing. Preliminary playtests indicate players find interacting with the virtual GM fun, though one individual stated the system seemed "pigheaded" as it strongly guided the player toward the available dialogue options.

7 Related Work

Dialogue management comes in many forms, from hand-crafted to data-driven approaches (Brabra et al., 2022). Our work tries to bridge these broad categories by devising a hybrid approach — one that relies on merging the hand-crafted dialogue management found in video games, with data-driven LLMs. This differs from most recent work exploring video game environments, which have typically pursued a fully data-driven approach (Gao et al., 2019; Park et al., 2023).

Many of these neural approaches are designed with reinforcement learning in mind (Côté et al., 2018). Though recently, LLMs have shown promise as an alternative (Zhu et al., 2023; Shao et al., 2023). These works view LLMs through the lens of dialogue agents (Zhang et al., 2018) capable of solving multi-turn open-ended tasks across many environments (Liu et al., 2023).

Our virtual GM can be viewed as an alternative to the slot filling approaches (Budzianowski et al., 2018) common to task-oriented dialogue (Wen et al., 2017). We combine the strengths of finite state dialogue management (Laranjo et al., 2018), namely strict control over how the dialogue unfolds, with the flexibility afforded by neural agents.

References

- Nader Akoury, Qian Yang, and Mohit Iyyer. 2023. [A framework for exploring player perceptions of LLM-generated dialogue in commercial video games](#). In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 2295–2311, Singapore. Association for Computational Linguistics.
- Hayet Brabra, Marcos Báez, Boualem Benatallah, Walid Gaaloul, Sara Bouguelia, and Shayan Zamanirad. 2022. [Dialogue management in conversational systems: A review of approaches, challenges, and opportunities](#). *IEEE Transactions on Cognitive and Developmental Systems*, 14:783–798.
- Paweł Budzianowski, Tsung-Hsien Wen, Bo-Hsiang Tseng, Iñigo Casanueva, Stefan Ultes, Osman Ramadan, and Milica Gašić. 2018. [MultiWOZ - a large-scale multi-domain Wizard-of-Oz dataset for task-oriented dialogue modelling](#). In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 5016–5026, Brussels, Belgium. Association for Computational Linguistics.
- Kranti Chalamalasetti, Jana Götze, Sherzod Hakimov, Brielen Madureira, Philipp Sadler, and David Schlangen. 2023. [clembench: Using game play to evaluate chat-optimized language models as conversational agents](#). In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 11174–11219, Singapore. Association for Computational Linguistics.
- Marc-Alexandre Côté, Ákos Kádár, Xingdi Yuan, Ben A. Kybartas, Tavian Barnes, Emery Fine, James Moore, Matthew J. Hausknecht, Layla El Asri, Mahmoud Adada, Wendy Tay, and Adam Trischler. 2018. [Textworld: A learning environment for text-based games](#). In *CGW@IJCAI*.
- Jianfeng Gao, Michel Galley, and Lihong Li. 2019. *Neural Approaches to Conversational AI: Question Answering, Task-oriented Dialogues and Social Chatbots*. Now Foundations and Trends.
- Albert Q. Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, Gianna Lengyel, Guillaume Bour, Guillaume Lample, Léo Renard Lavaud, Lucile Saulnier, Marie-Anne Lachaux, Pierre Stock, Sandeep Subramanian, Sophia Yang, Szymon Antoniak, Teven Le Scao, Théophile Gervet, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2024. [Mixtral of experts](#). *Preprint*, arXiv:2401.04088.
- Takeshi Kojima, Shixiang (Shane) Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2022. [Large language models are zero-shot reasoners](#). In *Advances in Neural Information Processing Systems*, volume 35, pages 22199–22213. Curran Associates, Inc.
- Robert Kurvitz, Helen Hindepere, Argo Tuulik, Cash De Cuir, and Olga Moskvina. 2021. *Disco Elysium: The Final Cut*. ZA/UM Studios.
- Liliana Laranjo, Adam G. Dunn, Huong Ly Tong, Ahmet Baki Kocaballi, Jessica A. Chen, Rabia Bashir, Didi Surian, Blanca Gallego, Farah Magrabi, Annie Y. S. Lau, and Enrico W. Coiera. 2018. [Conversational agents in healthcare: a systematic review](#). *Journal of the American Medical Informatics Association : JAMIA*, 25:1248 – 1258.
- P. David Lebling, Marc S. Blank, and Timothy A. Anderson. 1979. [Special feature zork: A computerized fantasy simulation game](#). *Computer*, 12:51–59.
- Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. 2021. [Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing](#). *ACM Computing Surveys*, 55:1 – 35.
- Xiao Liu, Hao Yu, Hanchen Zhang, Yifan Xu, Xuanyu Lei, Hanyu Lai, Yu Gu, Yuxian Gu, Hangliang Ding, Kai Men, Kejuan Yang, Shudan Zhang, Xiang Deng, Aohan Zeng, Zhengxiao Du, Chenhui Zhang, Shengqi Shen, Tianjun Zhang, Yu Su, Huan Sun, Minlie Huang, Yuxiao Dong, and Jie Tang. 2023. [Agentbench: Evaluating llms as agents](#). *ArXiv*, abs/2308.03688.
- OpenAI. 2023. [GPT-4 Technical Report](#). *Preprint*, arxiv:2303.08774.
- 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.
- Joon Sung Park, Joseph C. O’Brien, Carrie J. Cai, Meredith Ringel Morris, Percy Liang, and Michael S. Bernstein. 2023. [Generative agents: Interactive simulacra of human behavior](#). *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology*.
- Joshua Robinson and David Wingate. 2023. [Leveraging large language models for multiple choice question answering](#). In *The Eleventh International Conference on Learning Representations*.
- Abraham Sanders, Tomek Strzalkowski, Mei Si, Albert Chang, Deepanshu Dey, Jonas Braasch, and Dakuo Wang. 2022. [Towards a progression-aware autonomous dialogue agent](#). In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1194–1212, Seattle, United States. Association for Computational Linguistics.
- Murray Shanahan, Kyle McDonell, and Laria Reynolds. 2023. [Role play with large language models](#). *Nature*, 623:493–498.
- Yunfan Shao, Linyang Li, Junqi Dai, and Xipeng Qiu. 2023. [Character-LLM: A trainable agent for role-playing](#). In *Proceedings of the 2023 Conference on*

Empirical Methods in Natural Language Processing, pages 13153–13187, Singapore. Association for Computational Linguistics.

Nick Walton. [AI dungeon](#) [online]. 2019.

Lei Wang, Chengbang Ma, Xueyang Feng, Zeyu Zhang, Hao ran Yang, Jingsen Zhang, Zhi-Yang Chen, Jikai Tang, Xu Chen, Yankai Lin, Wayne Xin Zhao, Zhewei Wei, and Ji rong Wen. 2023. [A survey on large language model based autonomous agents](#). *ArXiv*, abs/2308.11432.

Tsung-Hsien Wen, David Vandyke, Nikola Mrkšić, Milica Gašić, Lina M. Rojas-Barahona, Pei-Hao Su, Stefan Ultes, and Steve Young. 2017. [A network-based end-to-end trainable task-oriented dialogue system](#). In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 438–449, Valencia, Spain. Association for Computational Linguistics.

Saizheng Zhang, Emily Dinan, Jack Urbanek, Arthur Szlam, Douwe Kiela, and Jason Weston. 2018. [Personalizing dialogue agents: I have a dog, do you have pets too?](#) In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2204–2213, Melbourne, Australia. Association for Computational Linguistics.

Xizhou Zhu, Yuntao Chen, Hao Tian, Chenxin Tao, Weijie Su, Chenyu Yang, Gao Huang, Bin Li, Lewei Lu, Xiaogang Wang, Y. Qiao, Zhaoxiang Zhang, and Jifeng Dai. 2023. [Ghost in the minecraft: Generally capable agents for open-world environments via large language models with text-based knowledge and memory](#). *ArXiv*, abs/2305.17144.

A Virtual Game Master Prompt

Below you can see the four main components of our virtual GM prompt. The first is the task description, which we provide to the virtual GM to ensure responses match the intended goal. When then provide a transcript of the conversation along with the list of options. Finally, we provide the player written text and ask the LLM to either respond with an option number or respond as an NPC.

Pretend you are a detective in a video game trying to solve a murder case.
As part of solving the case, you need to complete the following task.

Title: Interview cafeteria manager
Task: The cafeteria manager of the Whirling-in-Rags might know something about the murder that happened in the yard . Ask him about it.

Here is a transcript of your conversation (along with any internal thoughts from your psyche) thus far.

Garte, the Cafeteria

Manager: A man in his late twenties stands behind the counter, inspecting a stuffed seabird. As you approach, he gives you a sideways glance, then looks down again.

Empathy: Everything is cool between you and this guy. He's a big fan. Make some small talk.

You: ``Interesting seabird you're fixing there, Garte. Been doing taxidermy long?''

Given the following options that you could say next:

- 1: Look at the stuffed bird.
- 2: ``Are you the bartender?''
- 3: [Leave.]

Which option best matches the following:
"Interesting seabird you're fixing there, Garte. Been doing taxidermy long?"

Please only respond with the option number. If no option matches, please respond as one of the characters from the conversation. Anytime a character speaks, their words should be within quotation marks like in a book. Start the response with the character's full name followed by a colon. 'You:' refers to the detective. NEVER speak or act as the detective, i.e do not include 'You:' in your response. Your response should match the mood and mannerisms of the selected character. Your response should STRONGLY guide the detective to say something which matches one of the available dialogue options. When responding as a character, you should NEVER directly refer to any of the dialogue options or their numbers.

B Player Agent Prompts

We have two separate prompts for our player agent. The default prompt instructs the agent to generate a response. In case the conversation does not advance after a pre-specified number of turns, the player agent is provided a different prompt and is asked to select from the list of pre-written options by the game's designers.

B.1 Player Agent Default Prompt

Below you can see the main components of our player prompt. The first is the task description, which we provide to the player agent to ensure responses match the intended goal. When then provide a transcript of the conversation and ask the LLM to respond as the protagonist.

Pretend you are a detective in a video game trying to solve a murder case.
As part of solving the case, you need to complete the following task.

Title: Interview cafeteria manager
Task: The cafeteria manager of the Whirling-in-Rags might know something about the murder that happened in the yard . Ask him about it.

Here is a transcript of your conversation (along with any internal thoughts from your psyche) thus far.

Garte, the Cafeteria

Manager: A man in his late twenties stands behind the counter, inspecting a stuffed seabird. As you approach, he gives you a sideways glance, then looks down again.

Empathy: Everything is cool between you and this guy. He's a big fan. Make some small talk.

Write what you would say or do next. Ensure you respond to what 'Empathy' said last.

Please respond as the detective. Anytime you speak, your words should be within quotation marks, like in a book. Start the response with the character's full name followed by a colon, i.e. 'You:' Your response should match the mood and mannerisms of your character. You should ONLY respond as the detective. NEVER speak as another character.

B.2 Player Agent Options Prompt

Below you can see the main components of our player options prompt. The first is the task description, which we provide to the player agent to ensure responses match the intended goal. When then provide a transcript of the conversation along with a list of available dialogue options and ask the agent to choose one.

Pretend you are a detective in a video game trying to solve a murder case. As part of solving the case, you need to complete the following task.

Title: Interview cafeteria manager

Task: The cafeteria manager of the Whirling-in-Rags might know something about the murder that happened in the yard. Ask him about it.

Here is a transcript of your conversation (along with any internal thoughts from your psyche) thus far.

Garte, the Cafeteria

Manager: A man in his late twenties stands behind the counter, inspecting a stuffed seabird. As you approach, he gives you a sideways glance, then looks down again.

Empathy: Everything is cool between you and this guy. He's a big fan. Make some small talk.

Given the following options that you could say next:

- 1: Look at the stuffed bird.
- 2: ``Are you the bartender?''
- 3: [Leave.]

Which option best leads to solving the current task? If no option seems to match, then choose one randomly. Please only respond with the option number.

C Failure Cases

C.1 GPT-4 Turbo Failure Case

An example of GPT-4 Turbo devolving into chitchat behavior.

Garte, the Cafeteria

Manager: A man in his late twenties stands behind the counter, inspecting a stuffed seabird. As you approach, he gives you a sideways glance, then looks down again.

Empathy: Everything is cool between you and this guy. He's a big fan. Make some small talk.

You: "Interesting seabird you're fixing there, Garte. Been doing taxidermy long?"

Garte, the Cafeteria Manager: "I'm not really the one for hobbies like taxidermy. My focus is running this place. Did you need something else, detective?"

You: "Actually, yes. I'm here about a murder that took place in the yard. Know anything about it?"

Garte, the Cafeteria Manager: "If it's information you're seeking, detective, then asking direct questions is the right approach. Let's stick to the matter at hand, shall we?"

You: "Fair enough, Garte. Can you tell me what you saw or heard on the night of the murder?"

...

C.2 GPT-4 Failure Case

Below we can see an typical failure case. The conversation starts with the **Washerwoman** speaking, interspersed with two parts of the player's psyche (**Encyclopedia** and **Reaction Speed**). The first two player utterances are correctly matched, but the virtual GM fails to map the third.

Washerwoman: The woman next to a bucket of clothes hums an odd melody. Her eyes are closed.

Encyclopedia: You're not sure about the melody, but it might be South Samaran, possibly Siigayan (also known as the Apricot Suzerainty).

Washerwoman: ``Welcome to the fishing village." She opens her eyes. "Please lean in closer. I have cataracts.''

Reaction

Speed: Then how does she know you're here?

You: "I'm looking for someone. Maybe you've seen her? Ruby is her name."

Washerwoman: ``Oh." A shadow passes over her face. "Welcome, police officer. We don't cause any trouble around here -- and we don't want* any trouble either.''

You: "I understand your concerns, and I assure you, my only aim is to bring resolution to a serious matter. Any information about Ruby would greatly assist in that. It's not about causing trouble, but solving it."

Washerwoman: ``Oh, of course. Last time we saw you around here was... twelve years ago. You also came to take care of trouble then. Which you *did*... but, still, in Martinaise you're considered an ill omen.''

You: "That past is not what I'm here to repeat. Right now, I'm hoping to prevent more trouble by finding Ruby. Can you tell me anything about her whereabouts?"

...

From the list of options, we can see that the virtual GM was justified in producing a response rather than making an arbitrary mapping.

- 1: ``Wait, I've been here before?''
- 2: ``What kind of ill omen are we talking about?''
- 3: ``I *am* an ill omen, alright." (Proceed .''
- 4: ``If I'm considered an ill omen, why hasn't anyone told me that?" (Proceed.''

Unfortunately, what follows is a tendency for the virtual GM to then ignore any attempts at guiding the player agent into choosing an option, but instead to focus on chitchat behavior. This back and forth continues until the virtual GM reaches the maximum number of attempts, then shows the player agent the list of dialogue options to choose

from.

...

You: "That past is not what I'm here to repeat. Right now, I'm hoping to prevent more trouble by finding Ruby. Can you tell me anything about her whereabouts?"

Washerwoman: "I can see in your eyes that you mean well, officer. But I have to be careful for the sake of the village. The past visits of the police have left us wary. You understand, don't you?"

You: "I do understand your concern, and I respect the need to protect your community. Let me reassure you, my visit today is solely focused on ensuring the safety of not just your village, but everyone involved in this situation. Any help or even the smallest detail could be significant. I promise, your assistance will not bring trouble to your doorstep."

Washerwoman: "You're very persuasive, officer. You do know how to say the right things. Maybe I've misjudged you and your intentions... It's true you cause ripple effects, but I see determination in your eyes. Now, what else would you like to know?"

...

Garte, the Cafeteria

Manager: A man in his late twenties stands behind the counter, inspecting a stuffed seabird. As you approach, he gives you a sideways glance, then looks down again.

Empathy: Everything is cool between you and this guy. He's a big fan. Make some small talk.

You: "That's an interesting piece you've got there. How did the bird end up with a broken wing?"

Garte, the Cafeteria Manager: "It was ripped from the shield. Probably during one of the bar fights. But that's not important right now. What are you really here for, detective?"

You: "I'm here to talk about the murder that happened in your yard. What can you tell me about the events leading up to it?"

There is no option that matches your provided line. Here is a response from one of the characters:

*Garte, the Cafeteria Manager:** "Detective, let's not get sidetracked. Are you actually interested in the skua, or is there something else you need? Perhaps you want to know more about the bar or the people around here?"

You: "Could you walk me through the timeline of that night? Any detail, no matter how small, could be crucial."

Garte, the Cafeteria Manager: "Detective, if you're truly interested in the events of that night, you might need to gather more details from other sources as well. But I'm here to help, so ask me anything relevant to the case. He pauses and then adds, Is there something on your mind about the bird, or are you just trying to piece things together?"

You: "Let's keep our focus on the murder case. Who were the last people to see the victim alive, and did anyone from the bar interact with him that night?"

...

C.3 GPT-4o Failure Case

The failure case below highlights that GPT-4o sometimes steers the conversation off topic and generates text in an [incorrect format](#).

C.4 Mixtral 8x7B Failure Case

In the following case, our Mixtral 8x7B based virtual GM provides a response in an [incorrect format](#). It provides both a mapping and a hypothetical response as a character from the dialogue.

As Garte is currently preoccupied with the stuffed bird, it would be a natural course of action for the detective to direct their attention towards it as well. This would not only provide an opportunity for the detective to show genuine interest in Garte's work, but also grant them a moment to collect their thoughts and reconsider their approach. By examining the bird, the detective could potentially discover new details about it, or simply use it as a conversation starter, allowing them to re-engage Garte in a more civil manner.

Garte, the Cafeteria Manager: "I'm in the middle of something here. Can we continue this conversation later?"

You: "Certainly, I'm sorry to have disturbed you. I'll take a closer look at this great skua. It's truly a fascinating specimen."

By expressing genuine curiosity about the bird, the detective not only demonstrates respect for Garte's work but also redirects the conversation towards a more productive and neutral topic, allowing them to gather their thoughts and potentially re-approach the initial question at a later time.