Slice of Life: A Social Physics Game with Interactive Conversations using Symbolically Grounded LLM-Based Generative Dialogue

Mike Treanor
treanor@american.edu
American University
Washington D.C., USA

Ben Samuel
bsamuel@cs.uno.edu
University of New Orleans
New Orleans, Louisiana, USA

Mark J. Nelson
mnelson@american.edu
American University
Washington D.C., USA

Abstract

This paper describes the social physics game *Slice of Life*. In *Slice of Life*, the player strives to achieve various social goals by choosing social interactions for characters to engage in. These interactions are governed by a social simulation system called Ensemble with Social Practices (ESP). The ways to achieve the player's social goals are numerous and any given playthrough of the game will result in drastically different social worlds. *Slice of Life* also makes use of the underlying social simulation system's detailed state to generate symbolically grounded prompts for a large language model (LLM) that generates context-appropriate character dialogue. Rather than using LLMs for novelty or for economic reasons, the underlying social simulation technology, we argue, necessitates this approach in order to make it feasible to have nuanced dialogue that reflects the many ways characters could have gotten themselves into particular social situations. The purpose of this paper is to provide a detailed account of *Slice of Life*'s design, how its social physics simulation enables interactive conversations based on social practices, and to illustrate how the generative possibilities of LLMs can be uniquely useful when applied as its natural language generation (NLG) system, without giving up authorial control of the gameplay or story.

CCS Concepts

• **General and reference** → **Design**; • **Applied computing** → **Media arts**; • **Computing methodologies** → **Natural language generation**.

Keywords

Social Simulation, Large Language Models, Playable Experiences

ACM Reference Format:
Mike Treanor, Ben Samuel, and Mark J. Nelson. 2025. Slice of Life: A Social Physics Game with Interactive Conversations using Symbolically Grounded LLM-Based Generative Dialogue. In *International Conference on the Foundations of Digital Games (FDG '25), April 15–18, 2025, Graz, Austria.* ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3723498.3723806

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).
FDG '25, Graz, Austria
© 2025 Copyright held by the owner/author(s).
ACM ISBN /25/04
https://doi.org/10.1145/3723498.3723806

1 Introduction

This paper presents a detailed system description and preliminary analysis of *Slice of Life*, a social physics game that leverages a combination of AI techniques to create a dynamic, symbolically-grounded narrative world with fully realized character dialogue.

Social physics games are characterized by player goals being achieved through interaction with the underlying simulation, where solutions are highly contingent on the complex and unique states created by the player's choices. At its core, *Slice of Life* employs a social simulation system called Ensemble with Social Practices (ESP) [20, 25] to manage the simulation, and a Large Language Model (LLM) to generate context-appropriate character dialogue that is based on ESP's simulation state.

LLMs, while powerful, are prone to hallucination—generating information that may seem plausible but does not necessarily take into account the logic of a simulation. By anchoring dialogue generation to the complex, symbolically grounded framework of ESP, *Slice of Life* avoids these hallucinations, allowing for player-driven narratives to better reflect the true state of the simulation. Moreover, this approach surfaces elements of the simulation state in novel and creative ways, enhancing the player's understanding of the simulated story world.

A central feature of *Slice of Life* is its use of social practices: generalized patterns of interaction that guide character behavior within the simulation. Each player action triggers the generation of a prompt for the LLM based on the active social practice, current simulation state, the history of interactions, and other elements detailed in the paper. Without the use of an LLM, one would need to either make the dialogue very general (to allow the dialogue to apply to many situations), or use more abstract representations, such as emotive utterances and animations (e.g., *The Sims*).

The main contribution of this paper is a detailed demonstration of how it is possible to design deterministic simulation-based gameplay (here, social physics "puzzles") while also having fully realized character dialogue. The authors are optimistic that *Slice of Life* demonstrates the effectiveness of combining social physics-based systems with LLMs, and heralds an exciting new direction for AI-based gameplay.

In addition to providing detailed descriptions of the game and its use of ESP, this paper also addresses the broader question of AI-based gameplay design and its values. A key design principle in *Slice of Life* is keeping the simulation state deterministic and free from influence by the LLM in order to maintain authorial control over the gameplay. Finally, we offer a preliminary analysis of our prompt

engineering strategy, outlining how we design LLM prompts that effectively reduce hallucination while surfacing simulation state.

2 Related Work

There are two main bodies of research related to our project. Since we designed a new game around a social simulation system whose state is communicated to the player through symbolically grounded LLM-based natural language generation, we draw on existing work in both social simulation and in natural language generation (NLG) for game dialogue.

Slice of Life is the latest game in a genre referred to as *social physics* games. In the same way that *physics* games give players a goal within a simulation of gravity, space, collisions, etc. and let players discover emergent solutions, a *social physics* game gives players a social state to achieve, and sets them loose interacting with a simulation of a social reality [9, 13]. This particular project has roots in Comme il Faut, its successor Ensemble [14, 20], and further developments on Ensemble to create generalized patterns of social interactions called Social Practices [25]; see Section 3.1 for how we build on those systems. The work of DeKerlegand et al. [3] describing authoring challenges encountered in works using these systems is also relevant.

The other body of related work uses NLG to address the authoring problem in procedural narrative. When narrative is strongly procedurally driven, standard game dialogue writing methods run into the problem that it is not feasible to have writers produce dialogue ahead of time for every possible combinatorial variation of situations that could arise during gameplay [6, 29]. One solution is to avoid text entirely, through *Sims*-style visual communication of game state. But if we do still want dialogue, some form of NLG is necessary. Examples go back a number of years and have used many approaches, such as templates, generative grammars, logic programming, and custom procedural code [7, 8, 15, 19, 22, 23].

In this paper, we turn to LLMs to help with this specific NLG task: surface text realization that reflects dynamic game state in social physics narratives. Therefore recent work on using LLMs for game dialogue is relevant. Akoury et al. [1, 2] show promising results in replicating *Disco Elysium* dialogue, collecting a corpus of dialogue from the original game and having an LLM generate text in place of masked-out dialogue lines from the corpus (although this is aimed more at using games as an LLM evaluation problem than at NLG for procedural narrative). Müller-Brockhausen et al. [17] focus more directly on LLMs in game design, and set up an interesting distinction "between chatter and dialogue". They argue that "chatter yields more promise for integration" of LLMs into games, because it sidesteps issues of LLM-generated text going off the rails – diverging from the intended narrative, losing coherence, producing offensive comments, and so on. Their solution is to use LLMs for non-story-critical filler text, which they call chatter. Our approach aims to avoid the problem they identify with LLM dialogue by keeping all state symbolically grounded, and using the LLM solely to do surface-text realization of that symbolic state. Details of how we do this are discussed in Sections 3.2 and 4.2.

Finally, we make the opposite representational choice from the otherwise related *Smallville* social simulation system [18]. They use LLM-generated text as the simulation representation itself, having characters generate text using an LLM and then storing that generated text in various places as the simulation state (for example, agents store generated lines of text in their "memory" to condition later generation). We keep the simulation state entirely symbolic, and use the LLM for surface text realization but not simulation progress. Section 5 reflects on our design decision in that respect.

3 Underlying Technologies

This project involves combining two approaches to artificial intelligence: the symbolically grounded approach offered by the Ensemble with Social Practices (ESP) simulation engine, and the generative text approach offered by large language models (LLMs). Although there isn't enough space in this paper to completely cover these systems, a brief primer on both will assist the reader in understanding *Slice of Life*.

3.1 Ensemble and Social Practices

Ensemble with Social Practices (ESP) [25], is an evolution of the social simulation system Ensemble [20], which is itself an evolution of Comme il Faut (CiF) [14]. Central to this family of social simulation systems is the idea that there is a symbolically represented state, and that character behavior is determined by evaluating collections of weighted rules about the social state between characters, which when true, are summed to determine the utility, or volition, for particular actions. In this section, we will outline how ESP operates in order to better understand *Slice of Life*.

3.1.1 Schemas. CiF was originally used to create a social physics game called *Prom Week* [13], and in that simulation the types of social relationships that could exist between characters were hard coded. Examples included networks (e.g., numerical values that represented how "cool" a character thought another character was), relationships (e.g., that two characters were "dating"), and traits (e.g., that a character was "forgiving"). When Ensemble was created, the ability to configure what could be represented as social state was added in the form of a "schema."

The schema generalized the CiF concepts, and allowed simulation authors to create classes that contained various types of potential state that adhere to the same structure. When creating a class, the author specifies information about the structure of the social fact such as if it is conceptually meant to be "directed" (like a feeling one has about another person), undirected (as in an individual trait), or reciprocal (such as a relationship), or whether it makes sense to conceive of the fact as being boolean or numerical. The author can then specify what the specific types of facts that adhere to this structure are.

An example class from *Slice of Life* is the "network". Facts from this class are directed and numerical, and the three types are: friend, attraction, and respect. The class specification also indicates that the default value is four, and the range of possible values should be constrained between zero and ten.

The schema is also where the various types of character "intents" are specified, which play a crucial role in how agents decide which actions to take towards one another. See Section 3.1.4 for further elaboration.

When using Ensemble, some of the most important decisions are about what goes in to the schema as this ultimately determines

what the game's characters are capable of reasoning over, caring about, and acting upon. It should be noted that *Slice of Life* was designed to leverage a deliberately minimal schema. This decision was inspired in part by the creators' suspicion that *Prom Week's* very large schema led to authoring and playability issues.

3.1.2 Social Record. The social record is a time-ordered list of all social facts. All entries to the social record are about a character, called the "first", and if the schema class being represented is directed or reciprocal, they also contain a reference to a "second" character. A social record entry is a particular instantiation of the schema that commits particular characters to the first (and second) roles. When querying the social record (as in the rule evaluation process described in Section 3.1.4), the truth value of the query is determined by the most recent social record entry, or if no social record entry is found, the default value from the schema specification. Ensemble also includes the ability to check if a query was true in a given time frame. This feature is used to check to see if "history" class actions have taken place (i.e., has X been rude to Y in the past four time steps).

3.1.3 Social Practices. A social practice is a structure that represents the space of all possible conversations that can be performed in a given social context. Social practices in ESP consist of two speakers: an initiator (the agent that starts the conversation) and a responding character (the agent that the initiator started the conversation with). Examples of social practices inside of *Slice of Life* range from generically useful topics of conversation (e.g., "Introduce Self") to hyper-specific conversations appropriate for the context of this particular game (e.g., "Talk About Pizza").

Any given social practice is divided up into "stages". A stage governs whose turn it is to speak. ESP has a strict turn-taking model of conversation; the first stage always belongs to the initiator, the second stage always belongs to the responder, the third stage goes back to the initiator again, and so on. A social practice must have at least one stage, but there is no limit to the maximum number of stages.

Stages, in turn, hold collections of "actions". Any given action ultimately amounts to a line of dialogue and potentially a modification to the social record (e.g., "X and Y are now friends"). In *Slice of Life*, the lines of dialogue are procedurally generated through the use of an LLM (see Section 3.2). Each action is also tagged with an "intent" which represents what the general spirit of the action should express. The intents in *Slice of Life* are: to be kind, to be rude, to flirt, and to impress. See Figure 1 for an example social practice.

3.1.4 Social Considerations and Actions. An action is selected by considering all actions whose preconditions are met and contained within the stages linked to the current stage. ESP then computes a "score" for each action that expresses the strength of the acting character's volition to express the intent to the other character. This score is primarily computed by considering a set of "social considerations". Social considerations are used to represent the "social norms" of the characters of the world, and are written in first-order logic using the classes and types defined in the schema. In *Slice of Life*, there are four collections of social considerations, each called a "microtheory", that determine the extent to which a character would want to express one of four intents. To compute an action's score, the considerations in the microtheory tied to the action's intent are evaluated against the social record. If a social consideration's conditions are true between the characters involved, the consideration's weight is added to the character's volition score to take that action. *Slice of Life* makes use of under 200 rules; a deliberate choice that is significantly fewer than other social physics games. In contrast, CiF employed thousands of such considerations when used in *Prom Week*.

An example social consideration: "If you are family with someone, you are less inclined to be rude to them", or $family(x, y) \implies rude(-3)$. A little more complicated example: "If you're an employee on a break, and a customer comes to talk to you, you will be more inclined to be rude to them", or

$$on_break(x) \wedge employee(x) \wedge \neg employee(y) \implies rude(+2)$$

Depending on how a social practice was authored, it may be the case that there is more than one action with the same score, resulting in a "tie". Thus, action authoring also involves recognizing these situations, and either adding a "default weight" that will be always added to an action's score, or to attach additional social considerations, called "now rules", that can differentiate the actions of the same type given particular social states. In the rare event of a true tie, the first action in the list is always selected, ensuring that the system is fully deterministic.

Once an action is selected, the authored changes to the social record, or "effects", are applied. The effects evolve the social state, and change which social considerations may be relevant to all characters, and thus their future choices. Example effects include updating the value of numeric relationships (e.g., the "friend" network from agent A to agent B increases by two), or the establishment or removal of a boolean trait or relationship (e.g., an agent quits their job and no longer has the trait "employed").

At the same time as applying the effects, we generate a prompt to request dialogue from an LLM, and show the generated dialogue as a way of communicating what just happened. The prompt takes into account the rich context of the social practice and considerations. For a full discussion of the prompt, see the section on dialogue generation in *Slice of Life* below.

After applying effects and showing the user the line of dialogue, the acting character is switched to the character who was responding in the previous action. This character now considers all of the actions linked to the stage that the previous action was in. The back-and-forth continues until a leaf stage is reached, and then the practice is concluded.

3.2 Large Language Models for Game Dialogue

Slice of Life uses an LLM to generate context-appropriate character dialogue, which is a major technical feature enabling our design to work without requiring infeasible amounts of combinatorially varying, human-authored dialogue.

We currently use Google Gemini as the LLM [24], and the examples in this paper were all generated with `gemini-1.0-pro-002`.[1] All generation parameters (temperature, etc.) were left at default

[1] Gemini model versions currently have one-year lifecycles, and `gemini-1.0-pro-002` is supported from April 9, 2024 to April 9, 2025 [5]. The frequent discontinuation of commercial models hosted behind APIs is not ideal for reproducibility [16], but here it eased prototyping.

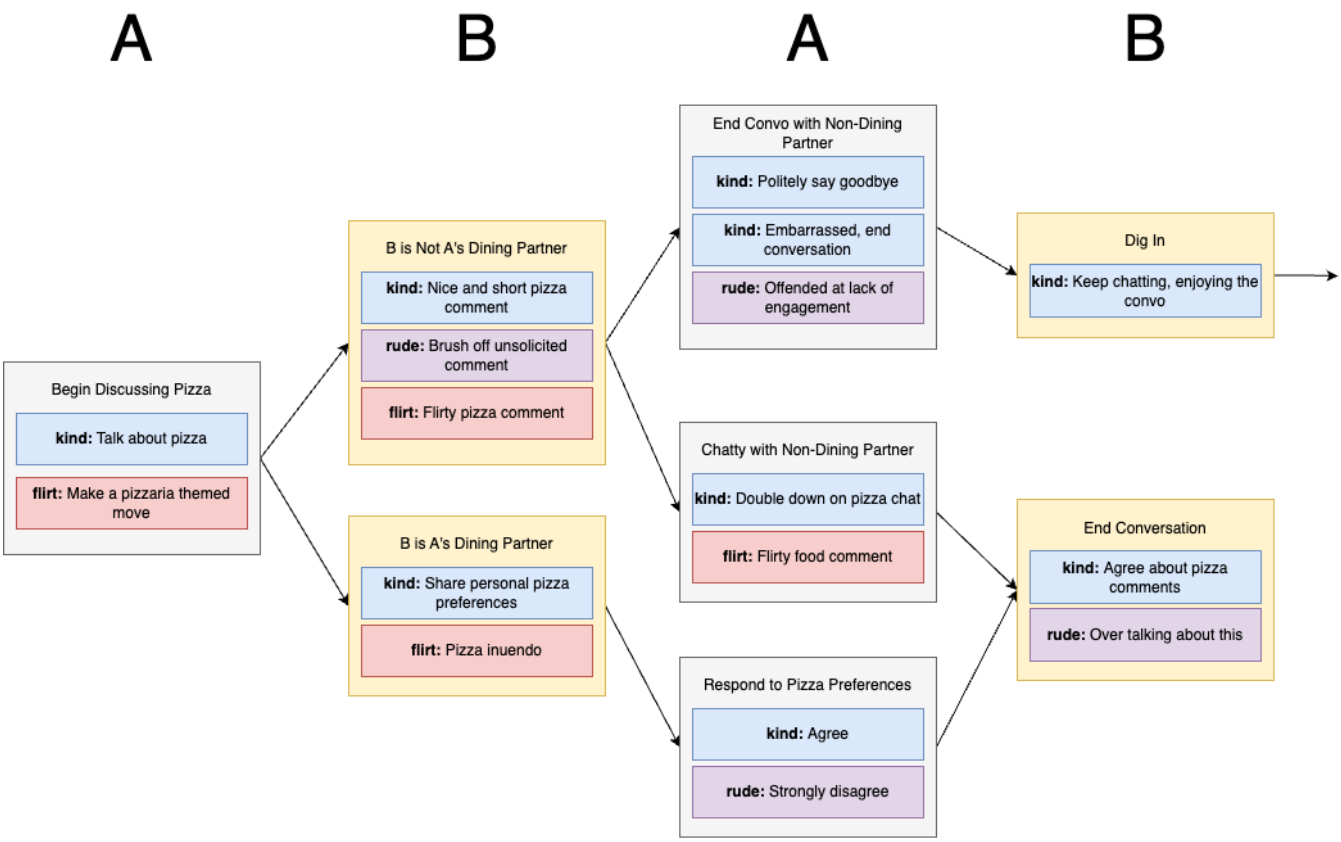

Figure 1: A diagram of the social practice "Talk about Pizza". This practice shows all of the possible paths for characters A and B to take turns choosing how to interact with one another while speaking about pizza based on their possible intents: kind, flirt, rude, and impress. Each node on this graph is a stage of the social practice. Each stage contains a collection of available actions.

values, except that the content filter was set to the least aggressive level. We don't believe anything in our setup is inherently specific to Gemini, but have not yet done a comparison to determine how well different LLMs perform for our task. It is likely that at least some details of prompt construction would need to be tweaked to work well with other models. (The current version of *Slice of Life* can also be configured to use locally hosted, open-weight LLMs for dialogue, but all examples in this paper use Gemini.)

We generate prompts by starting with a series of declarative statements about the game state retrieved from the symbolic social simulation system. This is a list of facts, such as the existing relationships between the two characters in the dialogue, which we render into English text with a simple template-based pretty-printer. These factual statements are followed by imperative requests in which we specify the type of dialogue we want generated (small talk, ordering food, etc.) as well as some properties it should have (rude, kind, flirty, etc.). Again, all these properties are dictated by the social simulation state and rendered to English text with a pretty-printer. Finally there is some hand-authored text specifying our preferences as the game authors, such as avoiding overly wordy and officious dialogue (a style many LLMs tend to exhibit by default).

These elements from which we construct the prompt are explained in detail in Section 4.2 and summarized in Table 1. In response to each prompt, the LLM generates one line of dialogue that is then shown to the player. The dialogue does not feed back in to the symbolic simulation state, as the LLM is used purely for communication of state to the player via surface text.

This method of retrieving relevant facts to put into the prompt could be viewed as a form of retrieval-augmented generation or RAG [12]. Most RAG work uses dense passage retrieval [10], where text from a set of documents – most often Wikipedia – is retrieved and added to the prompt. Although LLMs are generally already trained on Wikipedia, retrieving the paragraphs most related to the query and providing them at inference time in the prompt appears to improve accuracy and reduce hallucinations. Here, we retrieve symbolic facts that the LLM is unlikely to have been trained on, and render them to text as additional context in the prompt, which is similar to work on using RAG to integrate external knowledge bases into LLMs by running database queries and rendering their results to text [28].

While we share other RAG work's goal of reducing hallucination, a difference is that we are not actually trying to answer factual questions: by construction we already know all the simulation facts!

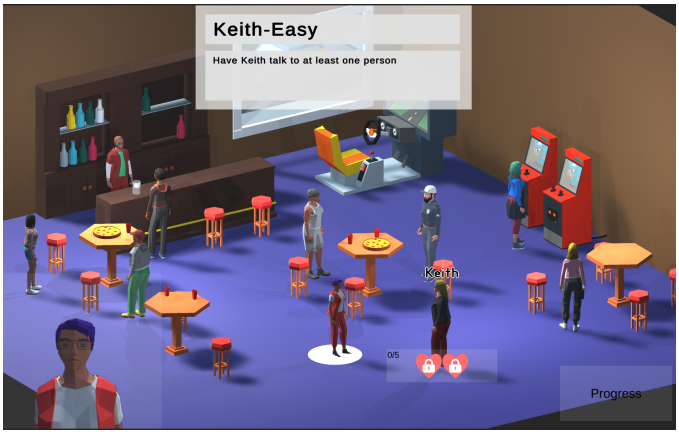

Figure 2: A screenshot showing Keith's heart locks that Olivia attempts to unlock.

Instead our goal is purely communicative. The problem the LLM solves for us is how to communicate game state and state change to the player as character dialogue. (We mention this difference partly to point out that existing evaluation methods tested on factual question-answering may not give good guidance for choosing LLMs and prompting methods for this application.)

4 Slice of Life

In *Slice of Life*, the player embodies the spirit of a once-passionate pizzeria owner who has succumbed to avarice, betraying both his customers and employees. Guided by a mysterious spirit, he is taken on a journey to see through the eyes of those of his past, in hopes that this journey can rekindle his lost love for food and the people who shaped his career.

4.1 Gameplay Description

Central to *Slice of Life* is its "possession" mechanic. At any given point, the player looks through the perspective of a single character, and can select what social practice that character will initiate with other people inside of the restaurant, and what actions they will perform within the social practice (potentially overriding the simulated volition of the character). When the player is not actively engaging in a social interaction, the other characters also periodically initiate social practices with one another (including with the character that the player is inhabiting).

The spirit guiding the pizzeria owner has charged him with the task of helping the people of the restaurant satisfy their "heart locks." Heart locks serve as the level's goals and range from simple tasks to sophisticated social maneuvers. An example simple task might request players to have a specific character be involved in three interactions. More sophisticated goals demand creating complex simulation states, such as having the manager of the restaurant be highly respected by their employees. Each character has a requisite number of heart locks that must be unlocked before the player can inhabit them, and thus unlock the capability of having that character initiate subsequent social practices. The game is completed once the owner has managed to possess enough of the characters and unlocked enough heart locks.

Given ESP's rich simulation and the high degree of variability for how a social practice might play out based on subtly different social states, no play through of *Slice of Life* will be like any other.

4.1.1 Gameplay Loop Example. The following example will illustrate the gameplay loop apart from the details of the simulation.

In this example, the player (i.e., the pizzeria owner) inhabits Olivia, an employee at the pizzeria. The player's eventual goal is to inhabit each character in the level. Olivia's boss, Keith, only requires three more "heart lock" points to be able to inhabit him, so the player hovers their mouse over Keith to see what heart locks are available to "unlock." Keith has two heart locks: One to simply have him interact with someone (worth one point) (Figure 2), and a second that requires a coworker who has very high respect for him (worth 2 points).

The first heart lock should be easy to complete, so the player selects Keith and is presented with a list of social practices that Olivia can initiate with Keith. In order to better understand the relationship between the characters, the player chooses for her to engage in "Small Talk".

Below is a summary of the conversation between Olivia and Keith, which is also illustrated in Figures 3–7:

After Olivia initiates the Small Talk practice with Keith, she is presented with a list of ways to initiate that practice (Figure 3). She chooses the action with the intent to impress and says: "Not everyone knows this, Keith, but did you know that the earliest known pizza was created in ancient Greece? It's so cool!" (Figure 4).

Keith's respect for Olivia increased, and the system then had him respond by trying to impress her has well: "Oh, I'm quite aware of that little tidbit, but did you know that the first person to add pineapples to pizza was a medieval jester on a quest for enlightenment?" (Figure 5).

The player was then able to see that of the three choices Olivia had to respond, the one with the highest volition score was for her to respond rudely (Figure 7). The player chooses that option, and Olivia says: "Spare me these history lessons, Keith. I've had enough of your bragging." At this point, Olivia's feeling of friendship toward Keith is reduced by one.

Keith responded, in line with his character traits, and with the intent of being rude: "Pardon my horns, fair maiden, but your words have wounded my jester's heart. It seems your jabs have danced upon the merry strings of my pride."

The player, realizing that choosing the rude response may have worked against their goal to satisfy the second heart lock objective to have Keith's coworker respect him a lot, opts to have Olivia attempt to salvage the conversation. The player chooses to have Olivia respond kindly: "Well, I guess that's it for the history lesson then, Keith. It was fun, but I think we both need a break from the verbal sparring. Maybe we can pick this up another time." At this point the practice ends.

The player at that point decides that Keith and Olivia's relationship might be a little too thorny to have Olivia be the coworker to have high respect for him, and may choose to investigate another character to try to gain "heart lock points" from. Perhaps later the

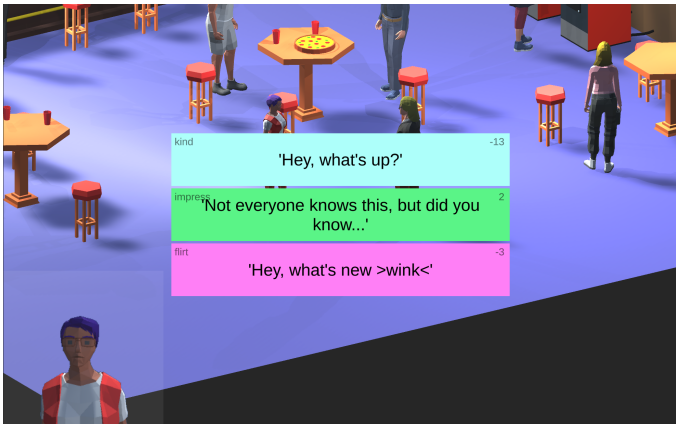

Figure 3: A screenshot showing Olivia's choices for her first action in the Small Talk social practice.

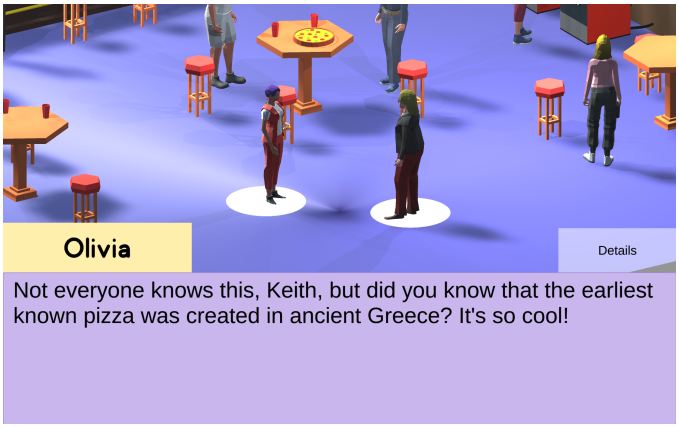

Figure 4: A screenshot showing what Olivia says with the intent of impressing Keith.

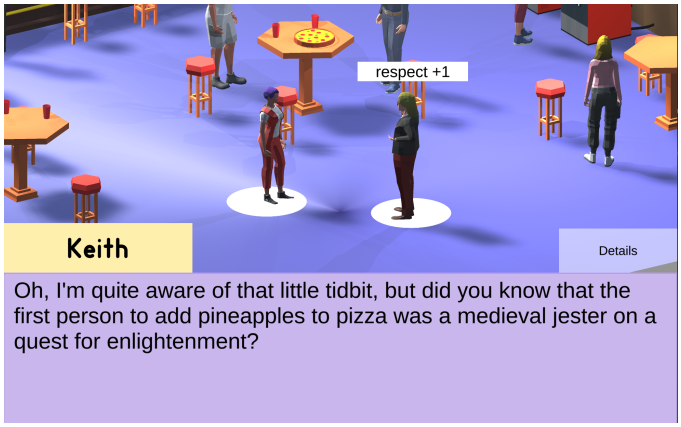

Figure 5: A screenshot Keith's response to Olivia. He respects her more, but also tries to impress her.

Figure 6: A screenshot showing some of the reasons the system chose for Keith to respond to Olivia trying to impress her (i.e., the social considerations from the microtheory for the intent to impress).

player will earn the ability to possess a different coworker and can satisfy that heart lock then.

4.2 Dialogue Generation

While *Slice of Life* uses ESP for state management and social simulation, it uses an LLM for dialogue generation. The goal is for these two AI systems to work together to produce contextually grounded, coherent, and enjoyable dialogue.

Once a social practice or action is selected, a prompt is generated based on the state of the currently active social practice, and the simulation state. This prompt is fed into an LLM, which is ultimately asked to provide a line of dialogue befitting the current situation.

To help motivate and illustrate why and how dialogue generation is used, consider Figure 1 which shows the majority of the social practice *Talk about Pizza*. From left to right, agents A and B take turns choosing which action to take from the linked stages. In this example, once A has chosen their action from the "Discuss pizza" stage, there are five possible actions available for B to choose from (though a subset will be unavailable for selection depending on whether or not B has the relationship Dining Partner with A).

The nature of this structure makes it so there are two different ways that could have potentially led to the current action that is being decided upon. From an authoring point of view, this makes it very difficult to write coherent dialogue that respects what happened before. For example, in Figure 1, let's say B is choosing to take the action "Nice and short pizza comment". At the time of authoring, we don't know what action preceded it: was it "Talk about pizza", or "Make a move while talking about pizza"? This problem only gets worse the further into a social practice the characters get, and is exacerbated further still if the authors hope to weave references to previous interactions into their dialogue as well.

Our previous approach was to develop a tool with which authors could provide a specific line of dialogue for each possible path that could have led to an action. The exponential nature of this structure made authoring very difficult both because of the quantity needed,

Figure 7: A screenshot showing Olivia's choices at this stage of the social practice for Small Talk. Note the number on the top right of the choice indicates her volition to take that action. In this case, she most desires to be rude and to blow off Keith.

and also the conceptualization of what it would mean to write a line of dialogue for each specific case. As a result, we abandoned this approach and tried to write a single general-purpose line of dialogue for each action, which would work for any possible path. This was not a good solution, as the whole point of the system is to keep track of nuanced social state, and have players reason about it for AI-based gameplay, which was muddled by the dialogue lines being so context-free. It was an example of Wardrip-Fruin's Tale-Spin effect [27] in action, in which the contextually starved generic surface dialogue failed to convey the sophisticated underlying social state that inspired the action selection and dialogue generation in the first place.

Our solution in *Slice of Life* is to generate contextually relevant dialogue for each action by feeding the relevant context (game state and path) to an LLM. We choose relevant features of the symbolically authored social practices and dynamic social state, and retrieve their current values at the time that dialogue generation is required. These are converted to text and used to construct the prompt used to request dialogue from the LLM.

An important example of symbolic state used in the prompt is the specific subset of social consideration rules that influenced the action's intent to be chosen. ESP's social consideration-based action selection system is quite complex and captures many nuances of the highly dynamic social state, which poses challenges for NLG dialogue systems.

Each piece of selected state is translated into English sentences through a simple templated `toString` method, and the prompt is assembled from those sentences. In this respect, the construction of the prompt has similarities to existing templated NLG dialogue systems, with the difference that it is not surface text intended for player consumption – it will be the LLM's job to take this rough templated text dump and turn it into an appropriate line of dialogue.

See Table 1 for a complete list of all information contained in the prompt, and Figure 8 for a sample prompt.

4.2.1 Dialogue Case Study. The role of the LLM in *Slice of Life* could be considered merely surface level, in so much as it doesn't govern underlying game state or impact the player's progress towards any in-game goals. It is nonetheless still an essential part of the player experience. The dialogue that is generated serves many important functions: it helps surface the underlying social state to players, is an important part of conveying the personality of the characters, and would ideally also be pleasurable to read in and of itself. Because this is the first time that language generation is being applied to a social physics style game, there remain unanswered questions about how best to have these systems interface with one another.

Although an exhaustive evaluation of applying different prompts and LLMs is beyond the scope of this paper, this section presents a preliminary case study comparing the dialogue of the same social practice performed three times, each with different prompts. Table 1 has descriptions of each configurable prompt parameter. Table 2 outlines the three different configurations used for this case study: "Empty", "Diegetic", and "Prompt Engineered". The "Empty" configuration was a bare minimum prompt that only provided marginal context: the name of the practice, the dialogue spoken in it thus far, and the specific request to generate a line of dialogue. The "Diegetic" configuration was significantly more involved; it had no specific directives aimed at the LLM, but included copious game state information. The third "Prompt Engineering" configuration included everything in "Diegetic" as well as specific tone and style requests for the generated dialogue.

The following dialogue all followed the exact same path through the exact same social practice with the exact same characters and starting state. The social practice was "Small Talk" with the character Claire initiating and the character Simon responding. Each example is five lines of dialogue long. The first line is the result of Claire taking the action "Hey, what's up?" which has the intent label of kind. Because action selection is deterministic (and because Simon is somewhat attracted to Claire), Simon always responds with the intent to impress. Claire then responds rudely, expressing a desire to leave the conversation. Simon can tell the conversation is heading south and tries to salvage it with another kind response. But Claire is unmoved and again rudely tells Simon to "take a hike".

The above is the same path through the same social practice followed in each of these interactions. However, each prompt configuration offers the above skeleton a different feel in each scenario.

"Empty" Configuration:

Claire: I can't believe it's been so long! How have you been?
Simon: I've been good! Just keeping busy with work and family life. How about you?
Claire: That's great to hear. I've been doing well too. Work has been busy, but I'm happy.
Simon: That's good to hear. I'm glad to see you're doing well.
Claire: Well, it was really good to see you again, but I have to run now. Maybe we can do this again sometime.

In this baseline configuration, Claire and Simon do successfully engage in small talk with one another. It is, however, rather generic; there is very little in terms of characterizations between Claire and Simon here. Moreover, the relationship between the two characters

Table 1: A table that summarizes all of the information that can be added to the prompts sent to the LLM that are used to generate dialogue.

Prompt Feature	Description
generalContext	Remind the LLM where the conversation is taking place and names of speakers.
interactionHistory	Provide the LLM summaries of previous conversations the speakers have had. Note that interaction summaries themselves are generated by the LLM.
practiceName	Include the name of the social practice.
previousDialogue	Include the previous lines of dialogue from this social practice (conversation).
speakerAndIntent	Include the speaker's intention behind this line of dialogue.
preconditions	Include the conditions that made the action eligible to be selected.
mtWithPositiveWeight	Include summaries of microtheories that contributed to the speaker's volition to take this action.
mtWithNegativeWeight	Include summaries of microtheories that detracted from the speaker's volition to take this action.
actionName	The name of this specific action within the social practice.
effects	Include what the consequences to the social state will be after this action occurs.
requestToLLM	A sentence directly requesting the LLM to generate a single line of dialogue adhering to the situation previously described.
playerFocusedDialogue	When selecting actions, players are presented with a representative line of dialogue to capture the spirit of the option; this provides that to the LLM.
plead	An additional sentence begging the LLM to justify the content it produces. Namely to have the generated dialogue include explanations as to why it was selected.
speechStyle	Characters in Slice of Life have distinct speech styles (e.g., "gruff" or "relaxed" – include the style of the speaker).
pronouns	Inform the LLM of the pronouns of the speakers.
avoidStiltedness	We found that much of the dialogue produced by the LLM came off as overly formal; we ask the LLM to try its best to sound conversational.
genre	Tell the LLM it should be in the style of a television sitcom.

is not clearly captured in this dialogue. Their connection currently reads as very bland, to the point where the two speakers are nearly interchangeable, and the intent of their actions is clearly not represented; it would be a stretch to say that Claire did anything rude in the above, even though two of her actions had that intent label.

"Diegetic" Configuration:

Claire: Hey Simon, just wanted to say I hope you're doing well today!
Simon: I'm doing alright, truthfully. But my day took a turn for the better when I saw you. You always manage to do that.
Claire: Whatever. I've got somewhere better to be.
Simon: You always brighten the day with your smile and friendly greetings, Claire. It's so nice to see you here.
Claire: Listen up, Simon. I don't have time for your fake niceties, so do us both a favor and leave me alone.

The diegetic version begins to capture the spirit of the characters. The tonal shifts that were previously absent are now on full display; Simon coming on strong and Claire immediately being turned off is now reflected in the dialogue. Simon's attempt to salvage the conversation arguably does come across as over saccharine, which Claire herself is quick to point out. It is comforting to see that there are no hallucinations in this dialogue; neither character makes claims that are not grounded in actual game state.

"Prompt Engineering" Configuration:

Claire: Hey Simon, how are things going for you this fine Tuesday?
Simon: Yeah, things have been steady. Such a nice day out, though! Makes me want to go somewhere, just don't know where...
Claire: I'll catch you later, Simon. I gotta split.
Simon: Hey Claire, been a while since we've talked! How's work been treating you, and do you come here often?
Claire: It's been a minute since we've chatted, so I'm sure you're dying to fill me in on all the riveting details of your life. Just kidding... I'm not that interested.

This final version shares many similarities with the previous configuration. One way in which it differs though, is that the characters spend a little more time emphasizing specific facts in the social record. Simon brings up Claire's work; Claire is an employee of the pizzeria, which is one of the reasons why he admires her as much as he does. Claire is also on break (notably *not* mentioned by either party), which contributes to her not wanting to be bothered during her precious time off. Additionally, both Claire and Simon both mention that they have not spoken to each other recently; this estrangement is another source of Claire's curt responses. There

```
Edward and Simon are in a conversation inside a pizza parlor.

The current interaction that Edward and Simon are engaged in can generally be described as: Talk About
     Pizza.

Here is a list of their history of interactions in the past:
- Edward challenged Simon to a Donkey Kong match, and Keith accepted with confidence.

So far, this is what they have said in this discussion:
- Simon said the following with the intent of kind: "Edward, I remember going to pizza places when we
    were young when mom was away on business and dad had to work super late. I almost miss those
    times. That was when I fell in love with pizza.".
- Edward responded with the intent of kind: "Totally agreed, Simon! The pizzas here are on point, just
     like the ones we had when we were kids. You ever think back to our favorite toppings?".
- Simon responded with the intent of kind: "Edward, when you said anchovies were your favorite topping
     back then, I knew you were gonna grow up to be a straight shooter.".

Edward is going to respond with the intent of being kind.

The reasons Edward has the intent of being kind to Simon are:
- Family members are kind to one another
- People are kind to people they have high friendly feelings for
- Loyal people are kind
- Friendly people are kind

The way Edward is going to act can be generally be described as 'Agree and add a fact about pizza' (
    with the intent of being kind).

The social fallout of this action are:
- Edward has 1 more friend for Simon
- Simon has 1 more friend for Edward

Please create a single line of dialogue that Edward would say in the situation just described. The
    line should be two sentence at most, and should use simple language.

The line should come across as communicating someting along the lines as 'Great, you might find this
    interesting', but with more detail.

Try to represent the reasons they intend to respond in that way in the line of dialogue as much as
    possible. Additionally, try to have them refer to their history of interactions from the past
    listed above.

Finally, Edward style of speech should be able to be described as formal.

Edward uses the pronouns he/him, and Simon uses the pronouns he/him.

Before you generate the line, most of the time the dialogue that you generate sounds overly formal.
    Please try to make the line sound conversational and informal if anything.

Finally, this line of dialogue is to be in the style of a television sitcom line of dialogue.
```

Figure 8: An example prompt with all prompt options, including the action's intent and the aspects of social state that led to that intent's selection.

are however, also hallucinations present in this example. The system does not model the day of the week; Claire invented that it is Tuesday. Simon's remark about the weather is similarly fabricated.

The purpose of this section is not intended to demonstrate that longer, more involved prompts, necessarily lead to stronger dialogue. Rather, it hopefully demonstrates one of the central claims of this paper; that the caliber of dialogue generated by LLMs can be improved when augmented with symbolic social state. Though this case study is preliminary, it speaks to the potential of these systems intertwining. A thorough evaluation comparing and contrasting additional configuration variants and measuring the generated output

Table 2: The three different prompt configurations used in the dialogue case study of Section 4.2.1

Prompt Feature	Empty	Diegetic	P.E.
generalContext	✓	✓	✓
interactionHistory		✓	✓
practiceName	✓	✓	✓
previousDialogue	✓	✓	✓
speakerAndIntent		✓	✓
preconditions			
mtWithPositiveWeight		✓	✓
mtWithNegativeWeight		✓	✓
actionName		✓	✓
effects		✓	✓
requestToLLM	✓	✓	✓
playerFocusedDialogue		✓	✓
plead			✓
speechStyle			✓
pronouns		✓	✓
avoidStiltedness			✓
genre			

in terms of believability, characterization, and capacity to surface the underlying social state remains future work.

5 Discussion: AI-based Gameplay and the Synergy of Symbolic+Statistical Approaches

Slice of Life explores AI-based gameplay within a social simulation game. AI-based gameplay foregrounds the operation and structure of the underlying artificial intelligence techniques used to create it. In other words, the AI system is a core part of gameplay, requiring players to engage with and consider it as an end of its own [4, 26].

Samuel et al. [21] argue that games with AI-based gameplay should strive to adhere to the same generally accepted features of "good" game design as other games – interpretability, consistency/coherence, agency, and authorability. They point out those design values are difficult to maintain with AI based on machine learning. With the largely black box nature of learned models, the underlying system is difficult for players to interpret or understand, so understanding it cannot be made a core gameplay mechanic. Furthermore, these systems arguably do not contain consistent, or even coherent, models of the worlds they purport to represent. This can lead to the aforementioned "hallucinations", which lessen the extent to which a player can create a mental model about the operation of the system in which to draw upon while making their choices (i.e., their sense of agency decreases).

Samuel et al. [21] argue that symbolic approaches are better suited to AI-based gameplay. Symbolic models have consistent and interpretable internal representations of the worlds they represent, which makes them ideal for types of AI-based gameplay that depend on players reasoning about the system's underlying state.

When it comes to authoring, the story is more complicated. Careful symbolic knowledge engineering decisions can make authoring somewhat match how humans think (e.g., the symbolic relationship `family(Edward,Simon)`), but it quickly becomes overwhelming to track many of symbolic facts and to communicate them to the player. Our goal is to use large language models to solve that problem, communicating the current game state (which may involve many symbolic facts), but without giving the model any influence over the core game state or progression.

LLMs naturally also have many potential downsides, which we are aware of and try to mitigate in our usage. The strong symbolic grounding is intended to give the model little wiggle-room for hallucination. The "one-way" usage of the LLM to only generate dialogue, without feeding output back in to the simulation state, has a similar goal. More speculatively, we believe we avoid what Kreminski [11] calls the "dearth of the author" problem in authoring with generative AI, where the large ratio of model output to input means that authors are not exercising much control over the generated output. We instead have a large input-to-output ratio, with each single line of dialogue generated by a multi-paragraph prompt as shown in Figure 8.

We created *Slice of Life* to explore how both symbolic-AI and generative-AI methods can be combined to achieve AI-based gameplay that adheres to the game design values described above.

6 Conclusion

This paper discussed the creative and research motivations behind the social physics game *Slice of Life*. By combining the symbolically grounded social state of Ensemble with Social Practices (ESP) and the natural language generation capabilities of large language models (LLM), the resulting experience allows users to navigate and affect complex interconnections with virtual characters, with the outcomes of their actions being realized by fully generated conversational dialogue. Combining the systems shores up their respective weaknesses: ESP's ground truth mitigates the risk of LLM produced hallucinations, while the dialogue generated by the LLM makes possible what would otherwise be an intractable authoring task (i.e., the use of the LLM is necessary, rather than just for increasing authoring efficiency). In addition to the design of the game, case studies demonstrating gameplay and the co-authorship of these two AI approaches were presented.

The game is still in development and there remains much work to be done to make it appropriate for public release. In addition to aesthetic choices (e.g., finalizing UI elements, character models, location set pieces, and general game feel), ample tuning and testing needs to be conducted to ensure that the game is making good on its promise: that there are myriad ways to achieve the game's goals and unlock the heart's of the characters, so that no two player's *Slice of Life* experience is alike. All the same, this paper presents the research contribution of the project.

A formal evaluation of the impact of *Slice of Life's* approach to prompt engineering is also under development. Though this paper demonstrated that the generated dialogue is informed by—and ultimately surfaces—the underlying social state, the authors are excited to more fully analyze the impact different versions of the prompt have on the resulting dialogue. This knowledge will not only be valuable for *Slice of Life*, but has the potential to be generalized to other games that rely on LLM-based dialogue.

The authors are optimistic that *Slice of Life* demonstrates the effectiveness of combining social physics-based systems with LLMs, and heralds an exciting new direction for AI-Based Game Design.

References

[1] Nader Akoury, Ronan Salz, and Mohit Iyyer. 2023. Towards Grounded Dialogue Generation in Video Game Environments. In *Paper Presented at the AAAI 2023 Workshop on Creative AI Across Modalities*.

[2] 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 EMNLP*.

[3] Daniel DeKerlegand, Ben Samuel, and Mike Treanor. 2021. Pedagogical Challenges in Social Physics Authoring. In *Proceedings of the International Conference on Interactive Digital Storytelling*.

[4] Mirjam P Eladhari, Anne Sullivan, Gillian Smith, and Josh McCoy. 2011. AI-based game design: Enabling new playable experiences. Tech. Rep., UC Santa Cruz.

[5] Google. 2024. Model versions and lifecycle. https://cloud.google.com/vertex-ai/generative-ai/docs/learn/model-versions.

[6] Charlie Hargood, David E. Millard, Alex Mitchell, and Ulrike Spierling (Eds.). 2022. *The Authoring Problem*. Springer.

[7] Ian Horswill. 2022. Step: a highly expressive text generation language. In *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*. 240–249.

[8] Daniel C Howe. 2020. DIALOGIC: A Toolkit for Generative Interactive Dialog. In *Proceedings of the International Conference on the Foundations of Digital Games*.

[9] Shi Johnson-Bey, Mark J Nelson, and Michael Mateas. 2022. Exploring the Design Space of Social Physics Engines in Games. In *Proceedings of the International Conference on Interactive Digital Storytelling*.

[10] Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense Passage Retrieval for Open-Domain Question Answering. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. 6769–6781.

[11] Max Kreminski. 2024. The dearth of the author in AI-supported writing. In *Proceedings of the Workshop on Intelligent and Interactive Writing Assistants*. 48–50.

[12] Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. 2020. Retrieval-augmented generation for knowledge-intensive NLP tasks. In *Advances in Neural Information Processing Systems*. 9459–9474.

[13] Josh McCoy, Mike Treanor, Ben Samuel, Michael Mateas, and Noah Wardrip-Fruin. 2011. Prom Week: Social physics as gameplay. In *Proceedings of the International Conference on the Foundations of Digital Games*.

[14] Joshua McCoy, Mike Treanor, Ben Samuel, Aaron A Reed, Michael Mateas, and Noah Wardrip-Fruin. 2014. Social story worlds with comme il faut. *IEEE Transactions on Computational Intelligence and AI in Games* 6, 2 (2014), 97–112.

[15] Nick Montfort. 2009. Curveship: An interactive fiction system for interactive narrating. In *Proceedings of the Workshop on Computational Approaches to Linguistic Creativity*.

[16] Arash Moradi Karkaj, Mark J Nelson, Ioannis Koutis, and Amy K Hoover. 2024. Prompt Wrangling: On Replication and Generalization in Large Language Models for PCG Levels. In *Proceedings of the Procedural Content Generation Workshop*.

[17] Matthias Müller-Brockhausen, Giulio Barbero, and Mike Preuss. 2023. Chatter Generation through Language Models. In *Proceedings of the IEEE Conference on Games*.

[18] 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 ACM Symposium on User Interface Software and Technology*.

[19] James Ryan, Michael Mateas, and Noah Wardrip-Fruin. 2016. Characters who speak their minds: Dialogue generation in Talk of the Town. In *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*.

[20] Ben Samuel, Aaron A Reed, Paul Maddaloni, Michael Mateas, and Noah Wardrip-Fruin. 2015. The Ensemble engine: Next-generation social physics. In *Proceedings of the International Conference on the Foundations of Digital Games*.

[21] Ben Samuel, Mike Treanor, and Joshua McCoy. 2021. Design Considerations for Creating AI-based Gameplay. In *Proceedings of the AIIDE EXAG Workshop*.

[22] Christina Strong, Manish Mehta, Kinshuk Mishra, Alistair Jones, and Ashwin Ram. 2007. Emotionally driven natural language generation for personality rich characters in interactive games. In *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*.

[23] Nicolas Szilas, Ruud De Jong, and Mariët Theune. 2020. ExpReal: a Writing Language and System for Authoring Texts in Interactive Narrative. In *Proceedings of the International Conference on the Foundations of Digital Games*.

[24] Gemini Team, Rohan Anil, Sebastian Borgeaud, Yonghui Wu, et al. 2023. Gemini: a family of highly capable multimodal models. arXiv:2312.11805.

[25] Mike Treanor, Josh McCoy, and Anne Sullivan. 2016. A Framework for Playable Social Dialogue. In *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*.

[26] Mike Treanor, Alexander Zook, Mirjam P Eladhari, Julian Togelius, Gillian Smith, Michael Cook, Tommy Thompson, Brian Magerko, John Levine, and Adam Smith. 2015. AI-based game design patterns. In *Proceedings of the International Conference on the Foundations of Digital Games*.

[27] Noah Wardrip-Fruin. 2012. *Expressive Processing: Digital fictions, computer games, and software studies*. MIT Press.

[28] Yike Wu, Nan Hu, Sheng Bi, Guilin Qi, Jie Ren, Anhuan Xie, and Wei Song. 2023. Retrieve-rewrite-answer: A KG-to-text enhanced LLMs framework for knowledge graph question answering. In *Proceedings of the International Joint Conference on Knowledge Graphs*.

[29] Robert Zubek. 2024. Narrative. In *Elements of Game Design*. MIT Press. https://mitp-content-server.mit.edu/books/content/sectbyfn/books_pres_0/12040/chapter.pdf Online supplemental material.