# Personalized Quest and Dialogue Generation in Role-Playing Games: A Knowledge Graph- and Language Model-based Approach

Trevor Ashby
trevor.clark.ashby@gmail.com
Brigham Young University
Provo, Utah, USA

Braden Webb*
bkwebb23@byu.edu
Brigham Young University
Provo, Utah, USA

Gregory Knapp*
gregory.c.knapp@gmail.com
Brigham Young University
Provo, Utah, USA

Jackson Searle*
jacksonjaysearle@gmail.com
Brigham Young University
Provo, Utah, USA

Nancy Fulda
nfulda@byu.edu
Brigham Young University
Provo, Utah, USA

## ABSTRACT

Procedural content generation (PCG) in video games offers unprecedented opportunities for customization and user engagement. Working within the specialized context of role-playing games (RPGs), we introduce a novel framework for quest and dialogue generation that places the player at the core of the generative process. Drawing on a hand-crafted knowledge base, our method grounds generated content with in-game context while simultaneously employing a large-scale language model to create fluent, unique, accompanying dialogue. Through human evaluation, we confirm that quests generated using this method can approach the performance of hand-crafted quests in terms of fluency, coherence, novelty, and creativity; demonstrate the enhancement to the player experience provided by greater dynamism; and provide a novel, automated metric for the relevance between quest and dialogue. We view our contribution as a critical step toward dynamic, co-creative narrative frameworks in which humans and AI systems jointly collaborate to create unique and user-specific playable experiences.

## CCS CONCEPTS

• **Computing methodologies → Natural language generation**; • **Software and its engineering** → *Interactive games*; • **Human-centered computing** → Text input; • **Applied computing** → *Computer games*; • **Information systems → Massively multi-player online games**.

## KEYWORDS

computational creativity, human-AI co-creativity, human-computer interaction, narrative, GPT-2, large-scale language models, language model, transformers, knowledge graph, World of Warcraft, English, NPC dialogue, procedural content generation, text generation, video games, natural language processing, RPG, MMORPG, quest, quests, dynamic quest generation, knowledge-grounded text generation

**ACM Reference Format:**
Trevor Ashby, Braden Webb, Gregory Knapp, Jackson Searle, and Nancy Fulda. 2023. Personalized Quest and Dialogue Generation in Role-Playing Games: A Knowledge Graph- and Language Model-based Approach. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems (CHI '23), April 23–28, 2023, Hamburg, Germany*. ACM, New York, NY, USA, 20 pages. https://doi.org/10.1145/3544548.3581441

*These authors contributed equally to this research.


*CHI '23, April 23–28, 2023, Hamburg, Germany*

ACM ISBN 978-1-4503-9421-5/23/04…$15.00
https://doi.org/10.1145/3544548.3581441

## 1 INTRODUCTION

Procedural Content Generation (PCG) is an active area of research in video game design that focuses on producing game content as users play through the game [9, 22, 23, 33, 48, 59]. Applications of PCG have included content generation in diverse domains including map and dungeon generation in early games [23] such as *Rogue* [53], infinite expanding world- and asset- generation in *Minecraft* [49], and open world text-based adventure game generation via large-scale language models for *AI Dungeon*[1]. Historically, procedurally generated content failed to adapt effectively to changing game constraints such as game state, world lore, story line, input, and specifically player diversity [23]. This weakness has been addressed with great success in recent years [6, 8, 45]. Our work aims to complement and add to this line of research. Most implementations of PCG utilize deterministic processes constrained by predefined rules, and are not equipped to incorporate player input into the generated content. In contrast, stochastic content generation based on neural networks or other machine-learned processes suffer from the inverse problem: an ability to incorporate player input and transcend pre-defined game constraints in ways that produce erratic, illogical, or incoherent content [10]. Both of these limitations can lead to a lack of immersion and waning interest for players [29].

Our work seeks to bridge this divide by creating procedurally generated content that is constrained by world state and genre conventions while still being responsive to player input, a paradigm very much in line with the mixed-initiative co-creativity framework first proposed by Yannakakis et al. [59]. This is accomplished via a

[1]https://play.aidungeon.io

content generation framework that integrates key facts and entities from the game world with NPC dialogue generated by a large-scale language model. Although our framework is applicable in a wide array of domains and game genres, we have chosen to probe its capabilities initially within the context of quests and quest-related dialogue in fantasy-based role playing games (RPGs). This choice is motivated both by the rich tradition of roguelike and role-playing games as test beds for novel procedural content generation techniques, as well as by the widespread familiarity of potential playtesters with the genre. Perhaps more importantly, quests and quest generation are critical elements of co-creative narrative play in RPG frameworks, as they directly impact gameplay and can have long-term effects on the state of the game world. Thus, player-mediated quests can have integral and psychologically profound impacts on game and player outcomes.

A high-level overview of our method is shown in Figure 1. Input from the player acts as the impetus for quest generation; an embedded representation of the player's free-form text is compared via cosine similarity to each of several possible knowledge graph paths that are traversable from a given NPC node. These traversible paths represent the needs and wants of the NPC, and the path with the highest similarity to the player's statement is selected as a quest seed. The system then expands this seed by (a) identifying which of three foundational quest types best aligns with the player's desires, (b) mapping knowledge graph elements from the quest seed onto an appropriate quest template via a context-free grammar, and (c) using a large-scale language model to generate accompanying NPC dialogue that retains key knowledge graph elements from the initial traversal.

A key advantage of our method is the centrality of the player as a key instigator and driving impetus behind quest generation. Insofar as the game world allows, quests are aligned with both the topic and style of the player's text input. Critically, this is accomplished without sacrificing in-game integrity. Our quest generation framework uses templated text generation and a structured coherence mechanism to align the final NPC dialogue with the game world state as encoded within the knowledge graph. Thus, the player is empowered to influence the type and style of quests they are offered as well as to receive unique, non-repetitive accompanying narratives. While we acknowledge that our experiments are generative in nature, we envision them as part of a larger co-creative framework akin to the framework presented in [31]. Within this larger co-creative framework, a player's conversational input has firsthand influence to generate novel content within the game world.

Through human evaluation, we determine that dialogue generation achieves quality comparable to that of hand-written quests while remaining constrained by the world state and responsive to user input. The relevance of aforementioned dialogue to its generated counterparts, title and quest, is likewise equivalent to hand-written examples. Our framework shows potential in providing an enhanced user experience due to the adaptability provided by dynamic models. This is specifically the case when player-entered input matches knowledge graph nodes. Though not perfect, our system aims to deepen immersion for players, allow another degree of agency, and enable elevated interaction between players and NPCs.

The remainder of this paper is structured as follows: We begin with an overview of related work in the fields of procedural quest generation (Section 2.1), NPC dialogue generation (Section 2.2), and large-scale language models (Section 2.3). We then introduce our core methodology, which combines a relational database with a large-scale language model to generate quest descriptions and accompanying NPC dialogue that are uniquely aligned to player preferences (Sections 3, 4, and 5). We measure the effectiveness of our contribution with an adaptation of the evaluation criteria used by van Stegeren and Myśliwiec [56], measuring generated quests based on Fluency, Coherence, Novelty, and Creativity (Section 6.1), and show that our framework is capable of generating text that can nearly match the quality of hand-crafted quests. To determine the degree of alignment between the generated NPC dialogue and the quest description used to generate it, we present a novel evaluation metric based on normalized co-occurrence scores of word lemmas between the two strings (Section 6.2). Lastly, we inspect responses from playtesters to determine the overall impact of our framework on player satisfaction, and confirm that generated quests from our system are also capable of reflecting the player's input in generation (Section 6.3) before our discussion in Section 7.

## 2 RELATED WORK

The video game content generation space is highly multi- dimensional with respect to both content type and generative timelines, and this is reflected in the literature. In this section, we look at various approaches to content generation by considering *what* type of content is produced (quests, dialogue, narratives, or other assets) and *when* creativity is intended to be introduced (design-time or run-time). We discuss some details of *how* these approaches were implemented, with particular emphasis on the influence that the player has on the generated content, throughout this section.

In contrast to the prevailing approaches to content creation in video games in which either (1) humans took the initiative in designing game elements while using computers as a tool to bring their visions to life or (2) computers took that initiative in PCG, Yannakakis et al. [59] first suggested a third approach in which computers could foster those visions through proactively suggesting alternate possibilities during the design process. They termed this approach *mixed-initiative co-creativity (MI-CC)*. We find this three-fold distinction between human-initiated, mixed-initiative, and procedurally generated content creation to be a useful categorization, and we will use the terms in this discussion.

### 2.1 Quest Generation

The study of quests in video games evolved out of the study of quests in both table-top role-playing games and literary analysis [25, 51]. For this reason, the term "quest" has been attached to a variety of different definitions throughout the literature [1, 3, 9, 18, 25, 32, 54, 61]. Yu et al. [60] discuss some of these interpretations and propose a specific, formalized definition of a quest as a set of tasks to be completed in ordinal order, with an associated distribution of in-game rewards. We find this definition valuable and widely applicable. However, because the consideration of rewards as integral quest elements falls beyond the scope of our current research, in this

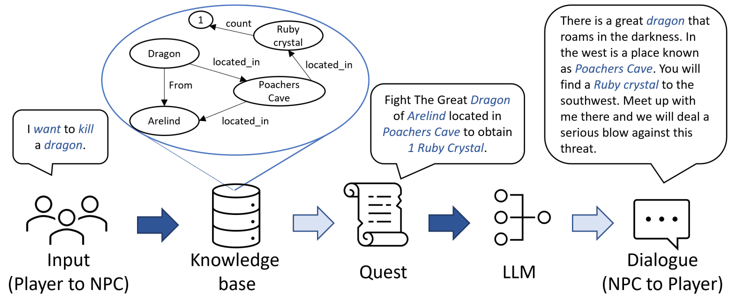



**Figure 1: Player-mediated framework for the creation of RPG quests and quest-related dialogue. Text from the player is matched via cosine similarity to knowledge graph nodes, which are then used to generate a quest description that is coherently aligned with the current state of the game world. A fine-tuned GPT-2 language model accepts this quest description as input in order to generate an appropriate quest title and accompanying NPC dialogue that introduces the quest.**

work we follow the lead of Breault et al. [9] in defining quests as "multistep tasks to achieve some goal".

Some of the first significant steps in the study of RPG quests were taken by Doran and Parberry [18], who laid the framework for quest generation by analyzing quest structure in systematic ways — ultimately identifying abstract forms for quests that could be concretely instantiated by the assignment of real, in-game entities. Along with [43], they also identified the potential motivating power that procedurally generated quests would have on player engagement and replayability as opposed to static quests, which do not fundamentally change the game state. The same authors later provided a prototype implementation which utilized a generative grammar to produce quests by using simple, atomic player actions as terminal symbols [19].

*2.1.1 Design-time.* Building off Doran and Parberry's foundational work [18], the mixed-initiative approach was taken up by Alvarez et al. [2, 3], who implemented a grammar-based tool in the Evolutionary Dungeon Designer [2], called *Questgram*, for aiding manual quest writing. The goal was to assist human designers in generating content that "[adhered] to game design patterns."

Also working in the domain of quest generation, Sullivan [50] described a method for creating *environment puzzle* quests, in contrast to the combat quests prevalent in many games. Similar to our knowledge graph-grounded approach, Sullivan proposed incorporating a common-sense database to identify logical relationships between world elements. While they recognized the system's potential for eventual automatic quest generation, the project seems to have been intended to aid authors as a brainstorming tool, as evidenced by subsequent work [51].

Similarly, Dormans and Bakkes [20] write of the potential for quest-adjacent procedural content generation, but ultimately provide a system for aiding the game design process. They analyze grammar-based approaches to space- and mission-generation in games and focus on graph grammars for mission generation due to their superior controllability.

Martin [38] trained a unique event-to-event language model through reinforcement learning called ASTER-XT. ASTER-XT generates events (later turned into sentences) in order to separate semantic event generation from sentence generation. ASTER-XT generates compelling, coherent plots that stick to a single story line. Martin's novel insight into neuro-symbolic architecture shows areas where neuro-symbolic language models outshine traditional language models.

*2.1.2 Run-time.* In one of the earliest attempts at fully-automated quest generation, Calvin and Michael [12] implemented a context-aware system for situating lock-and-key quests spatially in the game *Charbitat*. While they used a graph-based representation of the game world, the type of quests the system could create were highly constrained in nature, and the dynamic expression of quest tasks in language was not yet attempted.

Wang [58] provides a structural analysis of quests in RPGs and of the environments in which they occur, proposes 5 different metrics by which to evaluate procedurally generated quests, and supplies an abstract model for a quest generator.

Neural language models were used as a tool for cooking-themed quest generation by Ammanabrolu et al. [4], who focused on the semantically-convenient organization of online recipes for training data. They compared quest-generating models based on Markov chains to those based on both an LSTM (to generate a sequence of ingredients) and GPT-2 (to generate recipe instructions), with the goal of developing a model that could be used for text-based adventure games. They did not, however, apply their results outside of "home"-themed contexts or to games with visual and spacial components.

More recently, Breault et al. [9] introduced the *CONAN* engine, which generates quests through a planning algorithm that can then be delivered to players through NPCs. The proposed approach can generate quests associated to every quest motivation described by Doran and Parberry [18]'s foundational ontology. While their planning approach shows promise, they have not yet implemented a vehicle for delivering the generated quests through NPC dialogue.

Previous work has been done to inject global and local knowledge for intentional narrative planning [52] to allow custom narrative generation. Our work lends itself naturally to integrate with narrative planners in that such a planner's intended use is to generate a plot with meaningful events in sequence, whereas our work serves as a steward and connection between discrete entities (places, objects, and people) that inform the narrative. Our system provides a way for not only events, but quests to be included into narrative planning in a dynamic and natural way, while responding to a player's textual input. Similarly, Si et al. [46] showed that incorporating models of character relationships in dialogue-based narratives can improve performance on story continuation [46].

## 2.2 Dialogue Generation

Plausible story lines, natural dialogue, and coherent narratives are essential elements of human interaction within Role-Playing Games (RPGs) [8, 16]. Conversations with non-player characters (NPCs) are particularly essential because they are the vehicle through which human users interact with the digital inhabitants of the game world and receive the clearly defined errands and objectives known as quests. In earlymost game designs, NPC conversations are both conceived and carefully scripted by game designers. This hard-coded approach has many advantages, including thematic and causal coherence within the context of the game world. However, pre-written dialogue can detract from the illusion of reality and diminish a game's replayability [19, 22]. Additionally, human authorship of such content is also costly and time-intensive [9, 23, 35].

With respect to NPC dialogue, Tychsen et al. [55] provided an early emphasis on the importance of personalizing the player experience in large-scale RPGs, pointing out how bland a game world can feel when NPCs don't recognize players with whom they have already spoken and how easily surface-level, pre-scripted dialogue breaks the illusion of reality. They compare *massively multiplayer online role-playing games* (MMORPGs) to *pen-and-paper (PnP)* RPGs, and recommend incorporating several specific aspects of PnP character components into online RPGs. In later years, a variety of methods for procedural dialogue generation followed [13, 16, 17, 26, 28, 29, 34, 42, 44, 56], of which we outline a few of the most impactful below.

Large-scale language models, as described extensively in Section 2.3, are powerful tools for creating dynamic dialogue [36]. The most prominent neural architecture for such language models at present is the transformer [57] introduced by researchers at Google Brain in 2017. Transformers have been shown to generate text that is largely indistinguishable from human-generated content, and are able to produce text completions with astonishing fluency and uniqueness [10]. However, they struggle in the realm of narrative consistency. It is not uncommon for transformer-based models to generate text that is inconsistent with the larger context of the generative domain, or to produce text that is self-contradictory [10]. They also have known predispositions to produce text that is biased in undesirable ways, a topic discussed in greater detail in Section 7.3.

In part due to these limitations, and also as a consequence of the prohibitive computational cost incurred by such models, other methods of procedural dialogue generation are also common, including dialogue generated via grammars [13, 44] or via smaller language models such as LSTMs and RNNs [16, 26].

*2.2.1 Design-time.* In a focus on procedurally generated narratives, Reed [42] used a rule-based, mixed-initiative approach to produce "satellite sentences" to hand-authored content. The system relied on the author to compose several types of of short grammars, which could then be used to generate to describe setting or to provide dialogue pacing.

More recently, van Stegeren and Myśliwiec [56] fine-tuned the pre-trained transformer-based neural language model GPT-2 to learn the structure of *World of Warcraft* [21] quest titles, objectives, and dialogues. They specifically investigated the role of the temperature parameter in determining the novelty of generated content. While their model apparently did learn quest structures in addition to dialogue, their work was intended for game designers as an authoring aid, and they did not propose the use of their work for run-time content generation.

Kalbiyev [28] also took a neural approach, fine-tuning GPT-2 on lines of dialogue from *Fallout 4*. Their goal was to simulate the human-written dialogue in the context of the game with a model that could target specific affects or emotions. While the resulting fine-tuned model was able to capture the "fit" of the world setting, it struggled across most other metrics, including coherence, human-likeness, and the expression of the desired affect.

*2.2.2 Run-time.* Even prior to the RPG-focused work done by [55], Cavazza and Charles [13] drew on linguistically-informed semantic and syntactic structures to propose a framework for dialogue generation in interactive storytelling. While not implemented at any significant scale, this approach used meticulously hand-written rules and templates to generate structures that could be populated from a hierarchical task network. They focused on dialogue for narrative generation in the sitcom genre.

Dai [16] approached the task of NPC dialogue generation from the standpoint of neural text generation, and found that encoder-decoder architectures featuring a convolutional encoder and a long short-term memory decoder performed better than vanilla LSTM language models. However, the lack of data used to train these models prohibited analysis of their full potential, and no comparison was made to transfer learning from pre-trained LLMs.

On the other hand, Davies et al. [17] demonstrated the effectiveness of fine-tuned versions of GPT-2, accessible through a Web API, on generating human-like dialogue responses for NPCs in video games. Rather than focusing on quest generation and delivery, however, they instead focused on trade scenes wherein players make in-game transactions with NPCs.

Ryan et al. [44] created a tool for authoring context-free grammars for dialogue generation. Unlike many other grammar-based systems, which tend to focus on expanding top-level symbols to terminal tokens of English, their approach targets middle-out expansion, where non-terminals are replaced by both lower and higher symbols in the grammatical hierarchy. While unconventional, this allows systems to better adapt to current game state by placing conditions on when production rules are utilized in the grammar, and also significantly reduces the authorial burden for authoring these grammar rules (meaning that this paper could also be classified as a design-time contribution). Lessard et al. [34] built off this work by implementing the methodology in the game *Hammurabi* and observing how game designers appropriated the tool. They found that the current state of the technology was unlikely to be desirable for lay authors and game designers (i.e., those who aren't specialists in natural language processing) to work with, due to the learning curve of authoring grammar rules, predicting run-time behavior, and identifying bugs. They also found, however, that one of the most useful and promising aspects of the approach was not simply the *generative* potential to magnify authorial output, but also the *modularity* of that output, by prompting writers to think on a lower syntactic level to increase diversity of expression.

Finally, Hämäläinen and Alnajjar [26] presented a method from the standpoint of computational creativity for adapting video game dialogue to changes in game state. Their goal was not to generate new text from scratch, but to paraphrase existing dialogue to provide greater variety. Like [28], they also worked with dialogue data from *Fallout 4*, using a syntactic parser to determine the grammatical structure of each utterance. They trained a bidirectional recurrent neural network (RNN) on parallel corpus data wherein the source and target lines had the same semantic meaning but different syntactic structure, and then used word embeddings of the original *Fallout* dialogue to determine how to paraphrase the text with the new syntactic structure. While the authors were successful in deploying a mod for integrating their model into the game, they found that additional work was needed for preserving semantic content in the paraphrased result.

## 2.3 Large-scale Language Models

Our work relies heavily on the GPT-2 language model [41], which is built on the transformer neural network architecture [57] with approximately 1.5 billion trainable parameters. At the time our research commenced, this was the largest available language model with publicly released network weights suitable for custom fine-tuning. With the release of more recent language models and the introduction of for-pay fine-tuning services, we anticipate that our content generation framework could be easily applied to other language models including GPT-3 [10], GPT-XL [41], and GPT-Neo [7].

Auto-regressive language models such as those listed above can be viewed as conditional probability distributions over sequences of words, such that they are able to predict likely continuations of an initial text sequence, often called a *conditioning prompt*, or *narrative context*. Given a vocabulary $V$ of words and context window $(w_0, \ldots, w_{t-1})$ of $t$ previous words, the model will calculate the resulting $P(w_t|w_0 w_1 \ldots w_{t-1})$ for each word in the vocabulary. Novel text can be generated by sampling stochastically from among the words with the highest contextualized probabilities.

Such language models are aligned with the probability distributions of a text corpus by providing excerpts from the corpus as conditioning prompts and requiring the model to predict likely continuations of the sequence. Errors in the predicted probabilities are used to gradually adjust the network’s parameters to induce more accurate responses, a process commonly referred to as *training*. Over time, and given a massive enough training corpus, the language model learns to identify the patterns, nuances, and “meaning” of words and sentences within the text’s language. For example, GPT-2 was trained on 8 million web pages. Many language models have generated text that is largely indistinguishable to human written text [10], but training such a model from scratch, as previously mentioned, requires expansive resources, both financially and computationally. For this reason, a *fine-tuning* process is often applied. In this process, a pre-trained language model is adapted to a new task (usually a new training corpus) such that the model begins to give higher preference to text continuations that match those in the newly introduced training data. Because the model still retains all of the implicit language-related knowledge acquired during its initial training, the fine-tuning process is able to proceed rapidly and with relatively few computational resources. Fine-tuning can be effectively accomplished using only a very small number of additional training examples.

## 2.4 Our Work in Context

A number of authors [3, 12, 25, 27] have emphasized the role that quests play as "a bridge between games and narratives" [25]. Our work seeks to extend that bridge to NPC dialogue. Where prior research has sought to procedurally generate either quests or NPC dialog in isolation (i.e. independent both of each other and of the player), we propose an integrated framework which leverages user input for personalized quest generation, then uses the resulting quests as an instrument for dialogue generation. While this pipeline is not yet flawless (see Sections 6.4 and 7.3), it represents a major stride in the direction of integrated, co-creative player experiences according to the vision of [20, 55, 58, 59].

Our work strives for a multi-party co-creative narrative that is jointly constructed by the player and the game engine. This is accomplished by allowing the player’s input to influence the rule-based creation of quests, which are grounded in the codified constraints of the game world and then expounded upon via a large-scale neural language model to produce NPC dialogue. By involving the player as an active co-creator of the game diegesis, we hope to increase player engagement [43], facilitate emotional exploration [15], and improve replayability [47].

Long-term, we envision potential extensions to this work in which game elements invoked by the language model, as well as game state changes implied by the language model, are used to update the knowledge graph and so bring the world into alignment with generated content, just as the generated content seeks to maintain alignment with the state of the world.

# 3 METHOD

Our content generation framework[2] relies on three core elements: (1) An initial impetus provided by the player, in the form of an open-ended comment directed toward an NPC, (2) A relational database, also called a knowledge graph, that encodes key elements of the game world, including the locations and properties of NPCs, enemies, treasures, and related entities, and (3) A large-scale language model that has been fine-tuned to create quest-related narratives that reference the same objects and entities supplied in the quest description. We describe each of these core elements in detail below, followed by a precise description of our methods for creating quests based on player input (Section 4) and for creating quest-related NPC dialogue (Section 5).

## 3.1 Player Interactions

Player agency and desires should be at the forefront of an immersive game, influencing the outcome. By personalizing the experience

[2] https://github.com/DRAGNLabs/DRAGN-Town-Quests

to the player, game play can lead to a more satisfying experience [15, 43].

In our framework, players initiate quest generation via free-form text entered into the terminal. Conceptually, this is comparable to an in-game experience where the player is communicating with an NPC via voice or keyboard, and expresses a desire to achieve some goal or execute some action. We currently interpret all such player-entered text as a trigger event for quest generation; however, in the larger context of a full-scale RPG, one would expect a filtering mechanism to be used in order to prevent too many quests from being delivered in too short a time span.

Text entered by the player is evaluated as described in Section 4 and is instrumental in determining both the *content* of the quest, or the subset of knowledge graph nodes which will be used in quest generation, as well as the *style* or "flavor" of the quest to be generated. By engaging with the player's text both thematically and stylistically, we seek to leverage multiple modalities of satisfaction.

## 3.2 Knowledge Graph

A knowledge graph [14] is a relational database used to encode information about objects, entities, or ideas and the relationships between them. Such knowledge is often encoded as triples that include a subject, a predicate, and an object. When depicted graphically, knowledge graphs are usually shown as a series of nodes with labeled, directed edges between them.

The knowledge graph in our quest generation framework was implemented using Neo4j[3] . This knowledge graph is intended to serve as a prototype of a graph capable of producing quests representative of any desired quest class. As such, our graph is currently quite small, with all entries hand-crafted by members of the research team; however, we envision that improving aspects such as graph size and connectivity will lead to improved quest generation performance. In practice, most work that currently goes in to hand-writing individual quests and NPC dialogue utterances could instead be replaced by fleshing out a complete, comprehensive knowledge base of the initial game state.

Although our usage of a knowledge graph differs in purpose from story sifters [30], a unification of the two would be natural. Our knowledge base represents granular world ideas such as places, objects, people, and their relationships, whereas story sifters effectively generate and parse overarching plot structures.

## 3.3 Language Model

Our work uses the GPT-2 language model [41], a transformer-based neural network architecture that learns to predict likely continuations of a conditioning prompt or narrative context. Beginning with pre-trained network weights released by OpenAI, we aligned the language model with our intended use case by fine-tuning it on *World of Warcraft* [21] quest data containing 24,981 training examples extracted from the WOWHead[4] website, as compiled and annotated by [56]. This data was re-ordered to present each training example as a sequential group consisting of a quest, a quest title, and accompanying NPC dialogue. This modification enables quests, formed from knowledge graph elements as described in Section 4, to be used as language model prompts for context-specific title and dialogue generation.

# 4 QUEST GENERATION

We follow the example of Breault et al. [9] in defining a quest as a call to action, or more specifically a call to the player to change the state of the game. In our framework, a quest is defined first and foremost by its *description*, which is a concise phrase describing the goal to be achieved. Additionally, each quest has a corresponding *title*, or identifying tagline, of about 1 to 8 words, and an accompanying *NPC dialogue* that embeds the quest description within a narrative structure inspired by the current state of the game.

In the game world, quest generation begins when the player converses with an NPC. In the knowledge graph, the node representing the selected NPC becomes the starting point for quest generation. To narrow down the possible nodes to use, nodes connected to the starting node with relationships indicating a need or want of the NPC are identified and deterministically composed into simple English-language statements. For example, the node in Figure 2 for the "Great Dragon of Arelind" is connected to the NPC node "Trevor" by the relationship "wants-killed". This would be verbalized in our framework as "NPC (a.k.a "Trevor") wants killed The Great Dragon of Arelind." Although our graph uses relationships containing the words "want" or "need" as criteria for selection during the quest generation process, the exact definition of acceptable relations are arbitrary and can be determined by the knowledge graph creator.

Once all acceptable relations to the NPC nodes have been identified, their English-language representations are converted to a vector representation and compared via cosine similarity to the player's original input. The relation with the highest cosine score is selected as the first edge in a knowledge graph traversal to extract information with which to construct a quest. The graph is traversed in the direction of the highest cosine score node and the values of each visited node are stored to assemble into the final quest. Nodes are traversed until a traversal depth of 2 away from the selected node corresponding to the highest cosine similarity is achieved, although this depth can also vary by implementation.

Once all nodes are saved, the type of the final quest is determined by classifying the user input. In our work, quest types were limited to 3 possibilities: combat, gathering, and exploration quests. A zero-shot spaCy[5] pipeline trained on hand-written examples by one of the authors was used to classify the user input into one of these three categories. These hand-written examples can be seen in Table 1. Depending on the needs of the game developers, these quest categories could be easily expanded.

A simple grammar and the rules for each possible quest type is shown in Table 2. Values to fill the grammar are determined by the relationship connecting each node and the node value saved during traversal. The grammar is then filled using these pre-saved values; if there are two or more possible values for the field, one is randomly selected to be used. If there are no matching values for the template field, then the field is skipped altogether. The final filled template then becomes the final output quest and is passed to the language model to be processed for dialogue generation.

[3]https://neo4j.com/
[4]https://www.wowhead.com/
[5]https://spacy.io/

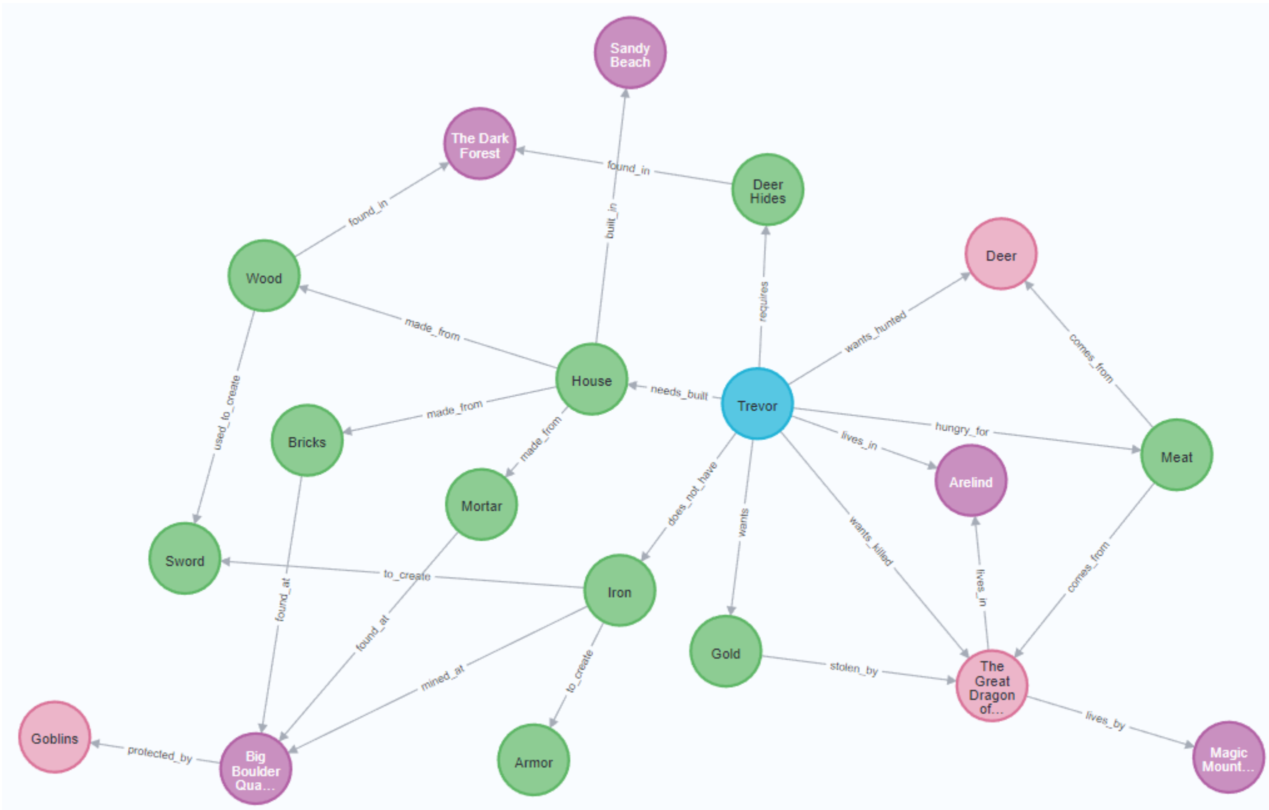



**Figure 2: Sample knowledge graph of the type used in our experiments. In-game objects and entities are represented as nodes, with relations between them indicated by labeled arrows. Nodes include annotations within the knowledge graph that describe whether they are an NPC ("Trevor"), a resource ("House", "Iron", "Mortar", etc.), a place ("Arelind", "Sandy Beach"), or a living creature ("Goblins", "Deer").**

Table 3 demonstrates the development of a quest from the knowledge graph depicted in Figure 2. Given an initial player input of "Do you know where I can get some gold?", the system will test the cosine similarity of all acceptable initial edge traversals from the initial NPC node. The traversal with the highest cosine similarity is "<NPC> likes to have Gold". From there, the system next classifies the user input as being most similar to a *Gathering* quest template, and subsequent knowledge graph edges are traversed as the template demands, resulting in the final quest, "Bring back 100 Gold which is protected by a Great Dragon of Arelind to create Gold."

## 5 NPC DIALOGUE

Although grammars can produce (accurate, world grounded) quests, these quests can lack detail, creativity, and an immersive nature. To provide each of these qualities, we leverage a large-scale language model to supplement the generated quest with a corresponding NPC dialogue. The non-deterministic sampling strategy applied by the language model allows it to generate multiple novel titles and NPC dialogues for any given quest description, thus increasing the game's replayability.

A core challenge with such language models is narrative consistency. The generated content must be not only novel and interesting, but also relevant to events and characters in the game world that are known to the player. This tends not to be the case for most language models, particularly when the conditioning prompt is short. We overcome this drawback by creating a fine-tuned language model specifically for the purpose of generating titles and NPC dialogue from quest descriptions (see Section 5. When given quest descriptions generated from the intersection between a player's expressed desires and the items, characters, and locations represented in the game knowledge graph (see Section 4), this model is able to produce NPC dialogue that remains largely in alignment with the state of the game world.

To explore the impact of different types and sizes of language models, as well as different fine-tuning regimens, we utilize four language models in the experiments detailed in Section 6. For comparison, some experiments also include a set of hand-coded World of Warcraft quests [21]. A brief overview of our experimental variants is presented as follows:

We selected GPT-2 as the base for our fine-tuning processes because of its impressive performance, as well as its resource-efficiency when fine-tuning. VS-MYS was optimized to produce a dialogue given title and quest, but to best implement the knowledge graph and retain the relations contained therein, the DRG-L and

**Table 1: Prompts used to train the zero-shot classifier for identifying quest type.**

| Type | Quests |
|---|---|
| Exploration | "Where should I go for an adventure?"<br>"Any recommendations of places to visit?"<br>"I am new here and looking for somewhere to stay the night."<br>"What are some famous locations in this area?"<br>"I love traveling to foreign countries!"<br>"I hope I can visit that city someday." |
| Combat | "Do you need help cleaning things up around here?"<br>"I'm just itching to fight some bad guys!"<br>"Anybody causing you trouble recently?"<br>"Looks like you have a pest problem on your hand."<br>"Anything that needs killing around here?" |
| Gathering | "Do you need me to get something?"<br>"Is there something I can get you?"<br>"Anything you are missing?"<br>"Looks like you need some supplies."<br>"I should get you some iron to make a sword"<br>"What else do you need me to get?" |

**Table 2: Quest Template Types**

| **Quest Type** | **Quest Template** |
|---|---|
| Gathering | <start> <number-to-collect><quest-target> <location> protected by <enemy> |
| Exploration | <start> <quest-target> and (meet <person-to-visit>) \| (obtain <item-to-retrieve>) |
| Combat | <start> <number-to-defeat> <quest-target> located in <location> and retrieve <item-to-retrieve> |

DRG-M models produce title and dialogue given quest. In Section [6] we validate these models' quality and evaluate their generated text against hand-crafted quests [21], quests generated by VS-MYS, and quests generated by our baseline model DRG-T.

Leveraging the creative abilities of sampling from a language model allows the dialogue through which quests are delivered to be unique and creative each time a user speaks to an NPC. However, due to the deterministic nature of our template-based approach to quest generation, the underlying quest will be consistently determined with the same user input. This process allows for more immersive PCG. For example, if the player approached an NPC and said, "I need some iron.", the NPC could respond, "That's it, Player. You're almost there, but there's one place where you can't go. One of the iron ore fields on the right side of the valley is guarded by sand golems. Go there and steal the ore!" Following the dialogue, the player would receive a quest "Obtain 26 Iron Ore located in The Ore Fields which is protected by 6 Sand Golems", titled "What's that noise?". If the player were to approach the same NPC without a change of game state, and say, "I need some iron.", an alternate dialogue would be generated. See Table 7 for examples.

**Table 3: Sample Quest Creation**

| # | Step | Example |
|---|---|---|
| 1. | Player input | Do you know where I can get some gold? |
| 2. | Initial knowledge graph traversal | <NPC> likes to have Gold |
| 3. | *Gathering* quest template | <start> <number-to-defeat> <quest-target> located in <location> and retrieve <item-to-retrieve> |
| 4. | Final Quest,<br>Title, and Dialogue | **Quest**: Bring back 100 Gold which is protected by a Great Dragon of Arelind to create Gold<br>**Title**: A Little Gold<br>**Dialogue**: This is interesting. The Great Dragon appears to not have a protective shield. Perhaps the gold they found was not too much... |

# 6 EVALUATION AND RESULTS

We evaluated our generated NPC dialogues across two dimensions: the intrinsic quality of the dialogue, and the relevance of that dialogue to the quest with which it is associated. We further evaluate the impact of player initiative on overall satisfaction, and confirm that when quests are directly related to free-form text entered by the player, the player's experience is enhanced.

To measure both intrinsic quality and overall satisfaction we conducted an online survey and accompanying user study. Survey and study participants were recruited from two populations: (a) students enrolled at Brigham Young University in Provo, Utah, and (b) members of online communities about games and gaming. The voluntarily collected demographics of age and familiarity with role-playing games (either digital or table-top) can be seen in Table 5.

## 6.1 Dialogue Quality

For evaluation of the intrinsic dialogue quality, we distributed an online survey to collect human feedback on the quality of the generated dialogue. This approach was chosen for consistency in comparison with the results of van Stegeren and Myśliwiec, although we ultimately made several changes to their survey design, as described below.

Survey respondents were recruited from students enrolled in computer science courses at Brigham Young University, and were asked to evaluate a random sample of 16 title-dialogue pairs taken from three of the four models (also randomized). Initially, we attempted to collect survey feedback along the same categories as van Stegeren and Myśliwiec: English, Content, Novelty, Surprise, and Creativity [56]. However, our early survey respondents indicated

**Table 4: Language model variants**

| Model | Description |
|---|---|
| DRG-L | The largest version (774M parameters) of OpenAI's GPT-2 language model, fine-tuned using the HuggingFace transformers library[6] on the dataset described in Section 3.3. The primary difference between this model and VS-MYS is in it's training. This model is trained on entire sequences of (Quest, Title, Description) tuples, in that order to support our framework of generation to produce titles and descriptions provided quests. |
| DRG-M | A medium-sized (355M parameters) implementation of GPT-2 fine-tuned on a cleaner subset of 25% (8638/34450) of the dataset from Section 3.3. This model explores the ability of a smaller and more resource-effective language model to produce effective NPC dialogue. |
| DRG-T | A distilled variant of the (82M parameters) GPT-2 language model trained to generate quests whose locations, organizations, and people are replaced by back-filling from respective node values. This model tests the feasibility of learning generalized dialogue structures rather than fully developed NPC dialogue. |
| VS-MYS | A GPT-2 language model fine-tuned by van Stegeren and Myśliwiec [56] on a corpus of video game quests. This model produces NPC dialogue that most closely matches the qualities of hand-crafted quests. However, due to differences in training, this model is incompatible with our content generation framework and cannot receive raw quests as input to dialogue generation. |
| WoW | (Not a language model.) Hand-coded quests from the World of Warcraft massively multiplayer online video game [21]. These quests of necessity reflect neither the intent of the user nor the state of the game knowledge graph, but they are a useful comparison when evaluating the fluency and believability of quests generated via our framework. |

that they found it difficult to semantically differentiate between categories. We therefore chose to eliminate Surprise as an evaluation category, change English to Fluency, and Content to Coherence. Surprise was eliminated because respondents had difficulty discerning "surprise" of a prompt in which no foundational context was provided (i.e., should the presence of a dragon be surprising in the story?). We chose only to include titles and dialogue passages (but not quest descriptions) so that respondents would focus on the raw output of the language models rather than on the conditioning prompts. This also allowed us to compare the results to DRG-T, a model which had not been fine-tuned on quests. The following

[6]https://huggingface.co/docs/transformers/index

**Table 5: Age and Familiarity with role-playing games demographics of participants**

| Age Range | Number of participants |
|---|---|
| **18-24** | **200** |
| 25-34 | 4 |
| 35-44 | 1 |
| 45-54 | 1 |
| 55-74 | 1 |
| 75+ | 1 |
| Familiarity Level | Number of participants |
| Not familiar at all | 23 |
| Slightly familiar | 41 |
| **Moderately familiar** | **63** |
| Very familiar | 51 |
| Extremely familiar | 30 |

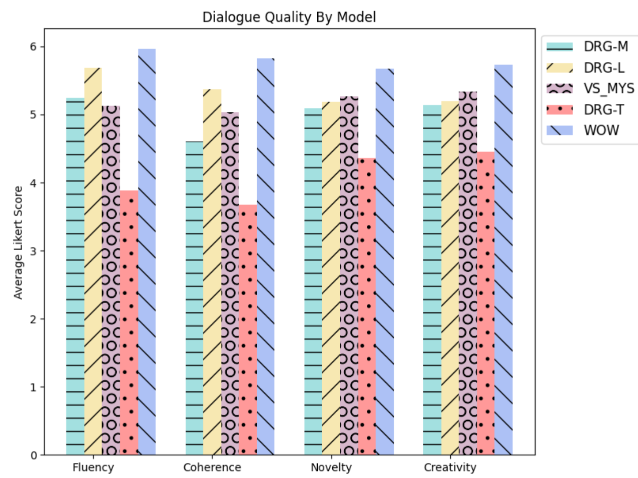



**Figure 3: NPC dialogue quality as evaluated by survey participants. Those who took the survey were instructed to consider fluency to measure the English grammaticality of the text, coherence to measure the internal consistency, novelty to measure how original the text seems, and creativity to measure how interesting and engaging the text seems.**

definitions were provided in the instructions so that participants were able to evaluate using standardized definitions: "Fluency: The dialogue makes use of correct English", "Coherence: The goal is clear from the dialogue", "Novelty: The dialogue is written in a novel way", and "Creativity: The dialogue is creative".

In total, 208 people responded to our survey. Results are displayed in Figure 3. The model from van Stegeren and Myśliwiec performed the best in each category. However, both DRG-M and

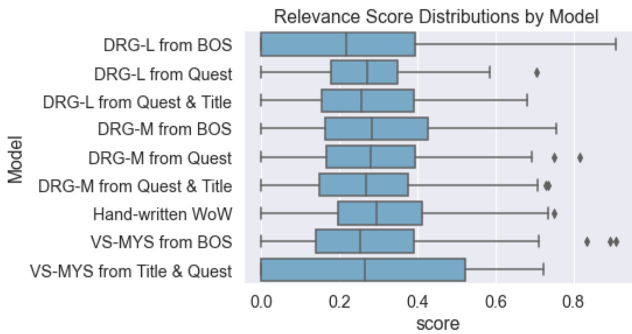



**Figure 4: Comparison of relevance distributions.**

DRG-L performed comparably, and we believe that the quality of their output could be improved with additional training. Importantly, three of the four language models were able to achieve Likert scores similar to those received by human-crafted quests from the World of Warcraft game.

## 6.2 Dialogue Relevance

In addition to the survey, we also propose a method for measuring the relevance of a given dialogue to its associated quest. There are two elements to this score; in part (1), we measure how important individual lemmas from the quest are to the dialogue by counting how often each non-stopword from the quest appears in the dialogue. In part (2), we count how many unique non-stopword lemmas from the quest appear in the dialogue. We normalize (1) by the length of the dialogue and (2) by the number of non-stopwords in our initial quest. We use the Python package spaCy for tokenization and lemmatization [24]. Formally, let Q denote the set of non-stopword lemmas in the quest, J the set of non-stopword lemmas in the text, J' the sequence (allowing for duplicates) of the words in the text, and $\alpha$ a real scalar between 0 and 1. Then our overall metric R can be expressed as

$$R = \alpha \overbrace{\frac{\sum_{q \in Q, j \in J'} [q = j]}{|J'|}}^{(1)} + (1 - \alpha) \overbrace{\frac{|Q \cap J|}{|Q|}}^{(2)}, \text{ where } [P] = \begin{cases} 1 & \text{if } P \text{ is true} \\ 0 & \text{otherwise} \end{cases}$$

In our calculations, we used $\alpha = 0.5$ to weigh the two sub-scores equally. It's worth noting that this metric should not be maximized to optimize quest quality; indeed, it returns a score of 1 if the model regurgitates the given quest verbatim and a score of 0 if none of the key words appear in any form in the generated output. Rather, as shown in Figure 3, hand-written quests and dialogue from World of Warcraft [21] average relevance scores of around 0.31, with a standard deviation of about 0.159, so we would anticipate that a high-quality quest- and dialogue-generating system would produce a similar distribution.

We averaged the relevance score over random samples of n > 100 quests generated from each model shown in Figure 3 (but not for DRG-T, as quests were not used to fine-tune that model). Full distributions are shown in figure 4. Notably, despite the relatively small size of DRG-M, its relevance scores mirror those of hand-written quests more than any other model, and its performance in the human-evaluated categories almost matches that of VS-MYS.

## 6.3 Player Satisfaction

In an effort to evaluate both the quality of our knowledge graph-verbalized quests and the extent to which our framework impacts player experience and satisfaction, we conducted a user study from a subset of the online survey participants.

Each of our (N=124) users was presented with our system and received some general instructions on how it worked. In the instructions, we stated that each interaction should be treated as if they (the player) were walking up to a new NPC for the first time with the intention of obtaining a quest. We then invited them to initiate a text-based conversation with an NPC. Because we have not yet implemented any additional conversational abilities in the system, their initial introductory phrase was sufficient (regardless of its semantic content) to trigger the internal quest generation process.

For the purposes of this playtest, we generated different tuples (Quest, Title, Dialogue) for each user prompt using three different methods: an option from our normal, knowledge graph-based model, a randomly-selected hand-written option from the *World of Warcraft* training data, and an option drawn from a naive, baseline 4-gram language model. In principle, $n$-gram language models work the same way as LLMs by autoregressively predicting $P(w_t|w_{t-1} \dots w_1)$ for any context window of length $t-1$. However, they do so by approximating
$P(w_t|w_{t-1} \dots w_1) \approx P(w_t|w_{t-1} \dots w_{t-n+1})$, where the latter probability is estimated empirically from raw counts in training data. We used the *World of Warcraft* data to train the simple 4-gram model as a comparative baseline.

Each generated tuple was then blindly shown to the testers, who were asked three questions:

- "Q1: Which prompt did you feel was most responsive to your input?"
- "Q2: Which prompt did you feel was most exciting/creative (i.e. Which would be the most fun to do in a game)?"
- "Q3: Write 2-3 sentences explaining your answers to Q1 & Q2."

In Table 8 we display the accumulated results of this play test, displaying which prompts participants felt were most satisfactory and responsive. Similarly, examples of player input, the generated quests relating to their response, and their responses can be seen in Table 9. Remarkably, the quests generated not only by our system, but even by the $n$-gram model, were able to respond to the user's input in a noticeable way.

To better understand why participants often preferred prompts retrieved from the WoW dataset (which were selected randomly) to prompts generated by either of the responsive models, we conducted qualitative analysis of the user responses. We found that prompts generated by our pipeline an the n-gram model often seemed disassociated with the input in semantically incongruent ways, whereas the participant felt that the human-written example from the WoW dataset, while less directly related to the user's

**Table 6: Average relevance of generated dialogue and title to corresponding input over samples of $n > 100$. For the column labeled "[BOS] token", a quest, title, and dialogue were generated from each model from the beginning-of-string token used in the fine-tuning of each model. Blank entries occur in the last row because only one score could be meaningfully generated from the *World of Warcraft* data. As the VS-MYS model was trained to produce NPC dialogue given a title followed by a quest, prompting the model with only a quest produced an incomplete triple—so that entry is also blank.**

| Average Relevance Scores | [BOS] token | Quest & Title | Quest |
|---|---|---|---|
| VS-MYS | .2675 | .2572 | - |
| DRG-M (ours) | **.3060** | .2720 | **.2902** |
| DRG-L (ours) | .2438 | .2794 | .2704 |
| Hand-written WoW | - | **.3073** | - |

| | |
|---|---|
| VS-MYS | The GPT-2 (774M parameters) model produced by van Stegeren and Myśliwiec. |
| DRG-M | Our GPT-2 (medium) model trained on a smaller subset of the data set. |
| DRG-L | Our GPT-2 (large) model trained on the entire data set. |

**Table 7: Examples of generated output by models**

| Model Name | Generated Output | Relevance Score |
|---|---|---|
| DRG-M | **Quest**: Obtain 26 Iron Ore located in The Ore Fields which is protected by 6 Sand Golems<br>**Title**: What's that noise?<br>**Dialogue**: <Taelia peers at you curiously.> That's it, Warrior. You're almost there, but there's one place where you can't go. One of the iron ore fields on the right side of the valley is guarded by sand golems. Go there and steal the ore! | 0.41 |
| | **Quest**: Obtain 26 Iron Ore located in The Ore Fields which is protected by 6 Sand Golems<br>**Title**: The Ore Fields<br>**Dialogue**: I've got no interest in mining the ore itself... but instead, I want you to retrieve what we need from the sand golems. If this little mining task doesn't kill something, then it's going to leave behind a lot of rubble. | 0.32 |
| DRG-L | **Quest**: Search for Torg Earkeeper, Meka the Face Chewer, and Chag the Noseless in Razorbloom.<br>**Title**: MossyFate.<br>**Dialogue**: Greetings. I have some news for you and the Horde.Torg is going to find us. He's been doing that for some time now, so be on your guard!You should start by searching the nearby Razorbloom caves. We're heading up there. See if anything interesting comes up. | 0.17 |
| DRG-T | **Title**: Escaping the Tomb<br>**Dialogue**: My boss said the last one was in a big cave. Then he went back to his cave when he died. It was over there you found him. I guess I will go fetch his grave. I will go back there, but he never comes back. Lost! I've seen their spirits before. He's a hunter, but he's been killing people. I've no idea why. He's just a hunter's spirit and he doesn't really deserve to have a second chance at life. | This output was not associated with a quest. |

input, contained enough foundational information for them to justify that it was the most "responsive" (i.e., "because it gave me a specific task to do"[9]). That hand-written quests outperformed machine-generated quests in overall satisfaction is unsurprising. However, our framework shows the potential for an enhanced user experience due to the adaptability provided by dynamic models. In particular, an expansion of our framework to larger and more fluent language models has potential to closely match the semantic coherence of human-crafted quests while maintaining consistency with the world knowledge graph as well as the user's input.

Additionally, we observed that many participants initiated NPC conversations with a generic phrase such as "How can I help you" or "Do you have any quests?" In such cases, the content of the quest had no impact on relevance, and it appears that participants used fluency and coherence as a tie-breaker to decide between quests that seemed equally responsive to their input. We view this phenomenon as being largely an artifact of the user study's natural constraints. Without the larger context of a game world and ongoing story line, the initial text entered by study participants lacked the specificity needed to create truly personalized quests. We anticipate that a more embedded study involving ongoing gameplay and repeated NPC interactions over time would elicit more specific quest solicitations such as "How can I kill the evil overlord?" or "I need to craft a fire blade", which in turn would produce quests with a high degree of customization for each user.

## 6.4 Failure Analysis

Here, we will outline the most common errors present in our content generation framework.

- Two out of every one hundred quests do not generate an accompanying dialogue. This is due to training examples that had quests, but no dialogue. (Unfortunately, this oversight in our pre-processing algorithms was discovered only after our models had been fully trained and our experiments completed.)
- While grammar and spelling hold up well even after fine-tuning, it should be noted that some dialogue includes nonsensical speech patterns and accents depending on the character it is taking on. For example, if the model is attempting to impersonate a pirate, vocabulary such as "yarr", "ahoi", and "matey" might be present. Additionally, dialogue may contain odd contractions, alternate spellings, and alternate ways to say a word depending on the character's background and accent (i.e., "I saw da man ova dere!"). These artifacts reflect the personalized dialogue present in the training set examples. It could be common for users to see the unique vocabulary and dialogue to be seen as an incorrect generation of our model.
- While the language model succeeds in most cases at incorporating entities and relations from the quest description into the NPC dialogue, it also tends to invent additional details not present in the knowledge graph. This could become disorienting to players who try to take action based on these additions.
- Our framework's performance is heavily impacted by the quality of the knowledge base provided. During playtesting, the most common cause of player dissatisfaction with our framework was the system's inability to find knowledge graph nodes that were similar to the player-entered text. A more expansive knowledge graph might help to resolve this issue.
- The zero-shot attributes of our quest triggering might not properly function if the user has irregular speech patterns or vocabulary. (i.e., "I'd desire the dragon vanquished." instead of "I want to kill a dragon.")
- Because of the possible ambiguity presented in graph edges upon knowledge graph creation, our model will occasionally mistake entities such as enemies with objects. An example of this could result in the statement, "Kill 40 Gold..." This confusion has only been seen between enemies and objects using the defined knowledge graph as seen in Figure 2.

# 7 DISCUSSION

## 7.1 Large-scale Language Models

The impressive performance of models such as OpenAI's GPT-3 and ChatGPT [40] have led to an increased popularity of generative text frameworks. We hypothesize that due to this recent performance, the use of LLMs within games and human-computer interaction will drastically increase in the coming years. The popularity of ChatGPT has brought the awareness of large-scale language models not only to researchers, but to the general public as well, which will likely lead to increased desire for more features and integration that leverages large language models. Because our framework is not model specific, it allows for plug-and-play integration with any generative model, including ChatGPT. Due to this design, improvements made to language models will naturally integrate with our work. Additionally, the unique grounding opportunities provided by our framework can enable a level specificity and customization that could not be achieved via a language model alone. We look forward to the improved human-computer interactions that may emerge from the enhanced storytelling and creativity of newer, larger models within our HCI centered framework.

## 7.2 Co-Creativity

We envision our framework as a step toward a fully co-creative system, and anticipate making additional contributions to the framework in pursuit of this goal. Possible expansions might include a chatbot-style technology, wherein the process of "receiving a quest" is only a small part of the human-NPC interaction. This would allow for users not only to receive personalized quests based upon the conversational requests, but also develop a personalized and ongoing relationship with the NPCs who provide the quests.

The current research uses a knowledge graph that is static and unchanging in nature. As an additional expansion in future work, this constraint could be relaxed to allow bidirectional information flow. For example, Ammanabrolu and Riedl [5] have recently taken large steps to advance the learning of world models from text environments by using underlying graph structures. Incorporating such research into a system that allows both dynamic NPC dialogue and dynamic knowledge graph evolution could be a productive way forward. Similarly, future work to provide dynamic game asset generation could ensure that the additions to the knowledge graph could be seen and experienced within the game.

In terms of quest generation, some previous work on the subject was not designed for PCG, but rather as aids for human game designers [20, 51]. With some tweaking, many of these approaches (including various generative grammars), could satisfy the quest generating-component of our pipeline, improving the quality of the quest provided to the LLM. Likewise, to improve the quest generation further, research could be conducted to better understand the efficacy of alternative and deeper graph traversal methods. Just as [30] provided a wide variety of sifting patterns, alternative graph traversal methods could be implemented to allow for variety in extraction. Although our work differs in purpose, such variety would be beneficial to our framework. Lastly, the exploration of more structured knowledge bases such as tabular relational databases would be beneficial as well.

The co-creativity of this completed framework would resemble that discussed in [59], the "co-creative process for the generation of

**Table 8: The collected votes from all playtest participants based on their responses to the two questions: "Which quest is most satisfying to you?" and "Which quest was most responsive to your input?"**

| Model | Satisfaction | Input-Responsive | Satisfaction (when input contained 'dragon') | Input-Responsive (when input contained 'dragon') |
|---|---|---|---|---|
| WoW | 197 | 159 | 14 | 5 |
| Ours | 125 | 169 | 14 | 21 |
| $N$-Gram | 58 | 52 | 0 | 2 |
| Total | 380 | 380 | 28 | 28 |

outcomes, solutions, or items". This completed framework would operate as such; an initial human game designer would develop a foundational world (i.e., knowledge graph, world assets, NPCs, etc.) on which the game would be built upon. During gameplay, the human player would interact with the NPCs within this world through chatbot-style conversations, learning about not only the world, but about the NPCs and their needs. Within this system, the world begins to learn about the player based upon the information that the player chooses to disclose. While conversing with the player, the computer would expand and update the world based upon the current conversation. While conversing, the computer would determine when to administer the quest by responding with the accompanying quest dialogue. Additional information within the dialogue generated by the computer would need to be added into the knowledge graph and world. This system would allow for co-creative world development to occur completely based upon the human-computer interaction between the player and the NPCs, built upon the foundation presented by the game designer.

While this somewhat lofty vision is still far from being realized, the current work suggests that it may someday be attainable, and lays important groundwork for future researchers. In particular, we hope that the principle of player-motivated quest generation will be more expansively investigated in coming years, not only within the context of quest generation, but in other aspects of procedural content generation, as well.

## 7.3 Limitations

One of the largest limitations of this project is the known tendency of large language models to absorb social biases and undesired language patterns from the text corpora on which they were initially trained [39]. While there has been a strong movement to reduce toxic and antisocial outputs in recently released language models, there is still always the chance that a player prompt may trigger offensive text. While this drawback is largely mitigated by our content generation pipeline, which mediates the player's input via the knowledge graph rather than passing it directly to the language model, there remains the possibility of potentially harmful dialogue generation. Any attempt to use our framework in a production setting should carefully consider and investigate this risk, and put appropriate safeguards in place. Specifically, it is recommended that known bias mitigation approaches upon large-scale language models be applied during deployment and testing [37]. Care should also be taken to ensure that the hand-crafted knowledge that informs the language model is free from harmful stereotypes and offensive patterns of bias.

In terms of data, the current study has been limited to an exploration of only one genre: fantasy. In fact, we are specifically only representing *World of Warcraft* fantasy. This causes all of our generation to reflect themes and patterns that may only exist in *World of Warcraft*. While we believe that most of our results will be equally observable in other game genres, it is possible that some genres are more naturally compatible than others with this type of automated content generation.

Along with this, because we leverage a pre-trained GPT-2 with a vast array of pre-trained knowledge, the language model will occasionally reference locations or objects that do not exist within the knowledge graph. In the current implementation this can be seen as an inaccuracy, but as discussed in Section 8, it's possible that these out-of-context generations could eventually become nodes themselves to be added into the graph.

Some areas of our framework are functionally lacking. For example, our framework only provides a single dialogue instance, rather than a continuous conversation. Potential talking points within the conversation are not being fully represented without a more functional chatbot-styled experience.

Finally, our dialogue generator relied on prompting a fine-tuned language model with a quest. While quests are necessary game elements at a very convenient level of abstraction for prompting language models, many other approaches could be taken. For example, our system provides no direct method for incorporating different personalities or socio-linguistic patterns for individual NPCs, but such attributes could be incorporated by associating classes of NPCs with neural programming interfaces [11] or other controllable text generation techniques [62].

We note that the limitations mentioned aren't due to concrete attributes of the pipeline, and could be further reduced/mitigated through additional development. We likewise acknowledge that our methods may not be absolute. Similar to the knowledge base, many of our extraction, verbalization, and storage methods have been designed to be plug-and-play, allowing for continual improvements to be made upon the framework as expanded on in the conclusion and future works section of this paper.

# 8 CONCLUSION

Recent developments in neural networks and large-scale language models are creating new and intriguing opportunities for procedural text generation, particularly as the fluency, coherence, and context windows of such models continue to expand. A critical challenge remains, however. Language models used during interactive gameplay must be able to adapt fluidly to an ever-changing world state,

**Table 9: Examples of playtest inputs, outputs, and tester responses.**

| Input | Generation(s) Referenced in Response | Tester Response |
|---|---|---|
| "How can I help you?" | **WoW**<br>**Quest**: Speack with The Bone Witch within the cave at Njorndar Village<br>**Title**: The Bone Witch<br>**Dialogue**: You remember Lady Nightswood , don't you ? She certainly remembers you . She's the forlorn and very angry banshee that you 'saved' when you also broke Vile and myself out from under the Lich King's dominion . The lady has gone into Jotunheim and possessed their bone witch . She intends to wreak havoc upon the vrykul from within . She has asked that you seek her out . You'll find her in the cave on the south side of Njorndar Village to the southwest within Jotunheim .<br><br>**Ours**<br>**Quest**: Eliminate Bear located in Forest to obtain Leather obtained from Bear<br>**Title**: A Visit to a Bear<br>**Dialogue**: The spirit world is full of animals , Player . Not everyone is interested in the wildlife . But I do ! If it weren't for a giant bear living in the woods west of town , my workshop would have been in utter ruins today . The bear must be the key to saving the town ! A visit to a bear would be so much more than just a simple tanning kit ! Please head into the woods to the west and kill a bear , bring back leather obtained from one of the bears , and I'll give you an extra trip . | **Satisfaction**: WoW<br>**Input Responsive**: WoW<br>**Experience**: "My input was based around asking a bartender about local gossip, and I feel like the first prompt was most responsive to that. The second prompt was almost incomprehensible, and the third prompt–while better than the second–felt more like a carpenter or smith pleading for help instead of a bartender giving me gossip. The tone of voice and setting of the first prompt was much more like a bartender giving local gossip. I also felt like the first prompt was the most creative and exciting. It referenced several characters, a tangible conflict, and pointed me to a specific location. The names were also very creative–they felt like something straight out of Skyrim. Again, I wouldn't be excited about the second prompt because it's almost incomprehensible, and the third one was too generic and frankly didn't make too much sense." |
| "I want to slay a dragon." | **WoW**<br>**Quest**: Speak to Krasus at the top of Wyrmrest Temple .<br>**Title**: What The Dragons Know<br>**Dialogue**: There's no mistaking it - this hilt comes from a powerful , ancient weapon , but which one ? I do not recall seeing or hearing of a sword with these markings , but that doesn't mean that all is lost . Korialstrasz , better known as Krasus , has long been a friend to the quel'dorei and the Kirin Tor . With his vast knowledge and resources , he's certain to know something . Travel to the top of Wyrmrest Temple and seek his counsel , then report to Arcanist Tybalin inside the Silver Enclave in Dalaran .<br><br>**Ours**<br>**Quest**: Kill Dragon located in Mountain to obtain Gems possesed by Dragon<br>**Title**: Gems of Destiny<br>**Dialogue**: I'm not sure how to start this , but I think the crystals I gathered back on the Isle of Spirits should be a start . The crystals you acquire are stored in the Dragonmaw Crater , and I'm not sure which ones I should bring to your fight , but I'd be happy to trade for a few for a few more . How about some for me ? | **Satisfaction**: Ours<br>**Input Responsive**: Ours<br>**Experience**: Prompt 3 was more responsive because I inputted that I wanted to slay a dragon and that prompt talked about slaying dragons for gems. That prompt also seemed more exciting because it was what I wanted to do in the game and it was more exciting than the others |

remaining consistent with the characters, events, and items known to the player. Additionally, player engagement and satisfaction can be further improved by anchoring procedural content generation to the self-expressed goals and desires of each player, allowing a level of customization that is not available in traditional design paradigms.

Our empirical results suggest that player engagement and overall satisfaction can be improved by (a) allowing the player to participate in the co-creative process of quest design, and (b) using world-consistent quest descriptions as the impetus for neural NPC dialogue generation. Long-term, we envision potential extensions of our framework in which all NPC speech, rather than just quest-related dialog, is generated via automated means. We also envision fully co-creative game environments in which dialogue generated by the language model becomes the impetus for updates to the world state. For example, if an NPC asserts that a specific item is located "Under that hill there", the in-game knowledge graph could be updated to include a "location" relationship between the item and a nearby hill, and ensuing gameplay would proceed accordingly.

The core contributions of this work include (a) the demonstration that large-scale language models can be constrained to conform with a world knowledge graph and used for quest generation along with a corresponding dialogue presentation, (b) the presentation of a dynamic quest generation pipeline that is responsive to free-form text entered by the user, allowing quests to become in-game instantiations of user goals and desires, and (c) an exploration of user responses to the presented pipeline which can be used to inform and improve the deployment of our system in expanded contexts. A key advantage of our approach is its natural compatibility with tools such as story sifters and narrative planners, and to expand upon them to not only involve players on a narrative level, but also on a personal level with every defined NPC. This system intends to deepen immersion and provide higher satisfaction to players, allowing them to exert more agency within the game narrative and allowing them to interact with NPCs at an elevated level. By providing opportunity for players to make a deeper connection to the digital residents along their journey, this enables the possibility of increased emotional resonance between the player and prominent NPCs.

In this paper, we have proposed a modular framework for procedurally generating quests and NPC dialogue from user input and game state, an approach with particular relevance to co-creative paradigms of game play. As a proof-of-concept, we implement and test this framework using a variety of possible language models, and show that procedurally generated quests and dialogue are in many ways comparable to hand-crafted *World of Warcraft* quests. We also demonstrate that, when the system successfully matches knowledge graph nodes to player-entered input, players are likely to prefer the generated quests over hand-crafted (but unrelated) ones [8].

We are hopeful that this work will be found useful to other researchers in framing the tasks of quest generation and NPC dialogue generation, and that our relevance metric will also be of value in evaluating such future work.

## REFERENCES


[1] Espen Aarseth. 2005. From Hunt the Wumpus to EverQuest: Introduction to Quest Theory. In *Entertainment Computing - ICEC 2005 (Lecture Notes in Computer Science)*, Fumio Kishino, Yoshifumi Kitamura, Hirokazu Kato, and Noriko Nagata (Eds.). Springer, Berlin, Heidelberg, 496–506. https://doi.org/10.1007/11558651_48

[2] Alberto Alvarez, Steve Dahlskog, Jose Font, Johan Holmberg, Chelsi Nolasco, and Axel Österman. 2018. Fostering creativity in the mixed-initiative evolutionary dungeon designer. In *Proceedings of the 13th International Conference on the Foundations of Digital Games (FDG '18)*. Association for Computing Machinery, New York, NY, USA, 1–8. https://doi.org/10.1145/3235765.3235815

[3] Alberto Alvarez, Eric Grevillius, Elin Olsson, and Jose Font. 2021. Questgram [Qg]: Toward a Mixed-Initiative Quest Generation Tool. In *The 16th International Conference on the Foundations of Digital Games (FDG '21)*. Association for Computing Machinery, New York, NY, USA, 1–10. https://doi.org/10.1145/3472538.3472544

[4] Prithviraj Ammanabrolu, William Broniec, Alex Mueller, Jeremy Paul, and Mark Riedl. 2019. Toward Automated Quest Generation in Text-Adventure Games. In *Proceedings of the 4th Workshop on Computational Creativity in Language Generation*. Association for Computational Linguistics, Tokyo, Japan, 1–12. https://aclanthology.org/2019.ccnlg-1.1

[5] Prithviraj Ammanabrolu and Mark O. Riedl. 2021. Learning Knowledge Graph-based World Models of Textual Environments. https://doi.org/10.48550/ARXIV.2106.09608

[6] Alexander Baldwin, Steve Dahlskog, Jose M. Font, and Johan Holmberg. 2017. Towards pattern-based mixed-initiative dungeon generation. In *Proceedings of the 12th International Conference on the Foundations of Digital Games (FDG '17)*. Association for Computing Machinery, New York, NY, USA, 1–10. https://doi.org/10.1145/3102071.3110572

[7] Sid Black, Leo Gao, Phil Wang, Connor Leahy, and Stella Biderman. 2021. *GPT-Neo: Large Scale Autoregressive Language Modeling with Mesh-Tensorflow*. EleutherAI. https://doi.org/10.5281/zenodo.5297715

[8] Barbaros Bostan (Ed.). 2022. *Games and Narrative: Theory and Practice* (1 ed.). Springer Cham, Berlin, Germany. https://link.springer.com/book/10.1007/978-3-030-81538-7

[9] Vincent Breault, Sébastien Ouellet, and Jim Davies. 2021. Let CONAN tell you a story: Procedural quest generation. *Entertainment Computing* 38 (May 2021), 100422. https://doi.org/10.1016/j.entcom.2021.100422

[10] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language Models are Few-Shot Learners. In *Advances in Neural Information Processing Systems*, Vol. 33. Curran Associates, Inc., Vancouver, Canada, 1877–1901. https://proceedings.neurips.cc/paper/2020/hash/1457c0d6bfcb4967418bfb8ac142f64a-Abstract.html

[11] Zachary Brown, Nathaniel Robinson, David Wingate, and Nancy Fulda. 2020. Towards Neural Programming Interfaces. In *Advances in Neural Information Processing Systems*, Vol. 33. Curran Associates, Inc., Vancouver, Canada, 17416–17428. https://proceedings.neurips.cc/paper/2020/hash/c9f06bc7b46d0247a91c8fc665c13d0e-Abstract.html

[12] Ashmore Calvin and Nitsche Michael. 2007. The Quest in a Generated World. In *DiGRA 3907 - Proceedings of the 2007 DiGRA International Conference: Situated Play*. The University of Tokyo, Tokyo, Japan, 503–509. http://www.digra.org/wp-content/uploads/digital-library/07311.20228.pdf

[13] Marc Cavazza and Fred Charles. 2005. Dialogue generation in character-based interactive storytelling. In *Proceedings of the First AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment (AIIDE'05)*. AAAI Press, Marina del Rey, California, 21–26.

[14] Xiaojun Chen, Shengbin Jia, and Yang Xiang. 2020. A review: Knowledge reasoning over knowledge graph. *Expert Systems with Applications* 141 (2020), 112948. https://doi.org/10.1016/j.eswa.2019.112948

[15] Tom Cole and Marco Gillies. 2022. Emotional Exploration and the Eudaimonic Gameplay Experience: A Grounded Theory. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems (CHI '22)*. Association for Computing Machinery, New York, NY, USA, 1–16. https://doi.org/10.1145/3491102.3502002

[16] Kevin Dai. 2020. *Multi-Context Dependent Natural Text Generation for More Robust NPC Dialogue*. Ph. D. Dissertation. Harvard College. https://nrs.harvard.edu/URN-3:HUL.INSTREPOS:37364681

[17] Richard Davies, Nathan Dewell, and Carlo Harvey. 2021. A framework for interactive, autonomous and semantic dialogue generation in games. In *Proceedings of the 1st Games Technology Summit*. 1st Games Technology Summit, Cologne, Germany, 16–28. https://opus.bibliothek.uni-wuerzburg.de/frontdoor/index/index/docId/24602

[18] Jonathon Doran and Ian Parberry. 2010. Towards procedural quest generation: A structural analysis of RPG quests. *Dept. Comput. Sci. Eng., Univ. North Texas, Tech. Rep. LARC-2010* 2 (2010), 42 pages.

[19] Jonathon Doran and Ian Parberry. 2011. A prototype quest generator based on a structural analysis of quests from four MMORPGs. In *Proceedings of the 2nd*

*International Workshop on Procedural Content Generation in Games (PCGames '11)*. Association for Computing Machinery, New York, NY, USA, 1–8. https://doi.org/10.1145/2000919.2000920
[20] Joris Dormans and Sander Bakkes. 2011. Generating Missions and Spaces for Adaptable Play Experiences. *IEEE Transactions on Computational Intelligence and AI in Games* 3, 3 (Sept. 2011), 216–228. https://doi.org/10.1109/TCIAIG.2011.2149523
[21] Blizzard Entertainment. 2004. *World of Warcraft.* Game [PC]. Blizzard Entertainment, Irvine, California, US.
[22] Ken Hartsook, Alexander Zook, Sauvik Das, and Mark O. Riedl. 2011. Toward supporting stories with procedurally generated game worlds. In *2011 IEEE Conference on Computational Intelligence and Games (CIG'11)*. IEEE, Seoul, Korea (South), 297–304. https://doi.org/10.1109/CIG.2011.6032020
[23] Mark Hendrikx, Sebastiaan Meijer, Joeri Van Der Velden, and Alexandru Iosup. 2013. Procedural content generation for games: A survey. *ACM Transactions on Multimedia Computing, Communications, and Applications* 9, 1 (Feb. 2013), 1:1–1:22. https://doi.org/10.1145/2422956.2422957
[24] Matthew Honnibal and Ines Montani. 2017. spaCy 2: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing. (2017).
[25] Jeff Howard. 2022. *Quests: Design, theory, and history in games and narratives.* CRC Press, Boca Raton, Florida.
[26] Mika Hämäläinen and Khalid Alnajjar. 2019. Creative contextual dialog adaptation in an open world RPG. In *Proceedings of the 14th International Conference on the Foundations of Digital Games (FDG '19)*. Association for Computing Machinery, New York, NY, USA, 1–7. https://doi.org/10.1145/3337722.3341865
[27] Jesper Juul. 2011. *Half-real: Video games between real rules and fictional worlds.* MIT press, Cambridge, Massachusetts.
[28] A. Kalbiyev. 2022. *Affective dialogue generation for video games.* Master's thesis. University of Twente. http://essay.utwente.nl/89325/
[29] Christopher Kerr and Duane Szafron. 2009. Supporting Dialogue Generation for Story-Based Games. *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment* 5, 1 (Oct. 2009), 154–160. https://ojs.aaai.org/index.php/AIIDE/article/view/12371
[30] Max Kreminski, Melanie Dickinson, and Michael Mateas. 2021. Winnow: A Domain-Specific Language for Incremental Story Sifting. In *Artificial Intelligence and Interactive Digital Entertainment Conference.* AAAI, Palo Alto, California, 156–163. https://doi.org/10.1609/aiide.v17i1.18903
[31] Max Kreminski, Melanie Dickinson, Michael Mateas, and Noah Wardrip-Fruin. 2020. Why Are We Like This?: The AI Architecture of a Co-Creative Storytelling Game. In *Proceedings of the 15th International Conference on the Foundations of Digital Games* (Bugibba, Malta) *(FDG '20)*. Association for Computing Machinery, New York, NY, USA, Article 13, 4 pages. https://doi.org/10.1145/3402942.3402953
[32] K Yu Kristen, Nathan R Sturtevant, and Matthew Guzdial. 2020. What is a Quest?. In *AIIDE Workshops.* AIIDE, Worcester, MA, 8.
[33] Young-Seol Lee and Sung-Bae Cho. 2012. Dynamic quest plot generation using Petri net planning. In *Proceedings of the Workshop at SIGGRAPH Asia (WASA '12)*. Association for Computing Machinery, New York, NY, USA, 47–52. https://doi.org/10.1145/2425296.2425304
[34] Jonathan Lessard, Etienne Brunelle-Leclerc, Timothy Gottschalk, Marc-Antoine Jetté-Léger, Odile Prouveur, and Christopher Tan. 2017. Striving for author-friendly procedural dialogue generation. In *Proceedings of the 12th International Conference on the Foundations of Digital Games (FDG '17)*. Association for Computing Machinery, New York, NY, USA, 1–6. https://doi.org/10.1145/3102071.3116219
[35] Boyang Li and Mark Riedl. 2010. An Offline Planning Approach to Game Plotline Adaptation. *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment* 6, 1 (Oct. 2010), 45–50. https://ojs.aaai.org/index.php/AIIDE/article/view/12394
[36] Junyi Li, Tianyi Tang, Wayne Xin Zhao, and Ji-Rong Wen. 2021. Pretrained Language Models for Text Generation: A Survey. https://doi.org/10.48550/arXiv.2105.10311
[37] Paul Pu Liang, Chiyu Wu, Louis-Philippe Morency, and Ruslan Salakhutdinov. 2021. Towards Understanding and Mitigating Social Biases in Language Models. https://doi.org/10.48550/ARXIV.2106.13219
[38] Lara Jean Martin. 2021. *Neurosymbolic automated story generation.* Ph. D. Dissertation. Georgia Institute of Technology. https://smartech.gatech.edu/handle/1853/64643
[39] Moin Nadeem, Anna Bethke, and Siva Reddy. 2020. StereoSet: Measuring stereotypical bias in pretrained language models. https://doi.org/10.48550/arXiv.2004.09456
[40] Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback. https://doi.org/10.48550/ARXIV.2203.02155
[41] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog* 1, 8 (2019), 9. https://d4mucfpksywv.cloudfront.net/better-language-models/language-models.pdf
[42] Aaron A. Reed. 2012. Sharing Authoring with Algorithms: Procedural Generation of Satellite Sentences in Text-Based Interactive Stories. In *Proceedings of the The Third Workshop on Procedural Content Generation in Games* (Raleigh, NC, USA) *(PCG'12)*. Association for Computing Machinery, New York, NY, USA, 1–4. https://doi.org/10.1145/2538528.2538540
[43] Simão Reis, Luís Paulo Reis, and Nuno Lau. 2019. Player Engagement Enhancement with Video Games. In *New Knowledge in Information Systems and Technologies (Advances in Intelligent Systems and Computing)*, Álvaro Rocha, Hojjat Adeli, Luís Paulo Reis, and Sandra Costanzo (Eds.). Springer International Publishing, Cham, 263–272. https://doi.org/10.1007/978-3-030-16184-2_26
[44] James Ryan, Michael Mateas, and Noah Wardrip-Fruin. 2016. Characters Who Speak Their Minds: Dialogue Generation in Talk of the Town. *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment* 12, 1 (2016), 204–210. https://ojs.aaai.org/index.php/AIIDE/article/view/12877
[45] Noor Shaker, Julian Togelius, and Mark J. Nelson. 2016. *Procedural Content Generation in Games.* Springer International Publishing, Midtown Manhattan, New York City. https://doi.org/10.1007/978-3-319-42716-4
[46] Wai Man Si, Prithviraj Ammanabrolu, and Mark O. Riedl. 2021. Telling Stories through Multi-User Dialogue by Modeling Character Relations. In *Proceedings of the 22nd Annual Meeting of the Special Interest Group on Discourse and Dialogue.* Association for Computational Linguistics, Singapore and Online, 269–275. https://aclanthology.org/2021.sigdial-1.30
[47] Gillian Smith. 2014. Understanding procedural content generation: a design-centric analysis of the role of PCG in games. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '14)*. Association for Computing Machinery, New York, NY, USA, 917–926. https://doi.org/10.1145/2556288.2557341
[48] Edirlei Soares de Lima, Bruno Feijó, and Antonio L. Furtado. 2019. Procedural Generation of Quests for Games Using Genetic Algorithms and Automated Planning. In *2019 18th Brazilian Symposium on Computer Games and Digital Entertainment (SBGames)*. IEEE, Rio de Janeiro, Brazil, 144–153. https://doi.org/10.1109/SBGames.2019.00028
[49] Mojang Studios. 2011. Minecraft. Game [PC]. Mojang Studios, Stockholm, Sweden.
[50] Anne Sullivan. 2009. Gender-inclusive quest design in massively multiplayer online role-playing games. In *Proceedings of the 4th International Conference on Foundations of Digital Games (FDG '09)*. Association for Computing Machinery, New York, NY, USA, 354–356. https://doi.org/10.1145/1536513.1536581
[51] Anne Sullivan, Michael Mateas, and Noah Wardrip–Fruin. 2009. Questbrowser: Making quests playable with computer-assisted design. *UC Irvine: Digital Arts and Culture* 2009 (2009), 8. https://eis.ucsc.edu/papers/0011_sullivan.pdf
[52] Jonathan Teutenberg and Julie Porteous. 2015. Incorporating Global and Local Knowledge in Intentional Narrative Planning. In *14th International Conference on Autonomous Agents and Multiagent Systems.* AAMAS, Istanbul, Turkey, 8. https://www.ifaamas.org/Proceedings/aamas2015/aamas/p1539.pdf
[53] Michael Toy, Glenn Wichman, Ken Arnold, and Jon Lane. 1980. *Rogue.*
[54] Marcus Trenton, Duane Szafron, Josh Friesen, and Curtis Onuczko. 2010. Quest Patterns for Story-Based Computer Games. *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment* 6, 1 (Oct. 2010), 204–209. https://ojs.aaai.org/index.php/AIIDE/article/view/12408
[55] Anders Tychsen, Susana Tosca, and Thea Brolund. 2006. Personalizing the Player Experience in MMORPGs. In *Technologies for Interactive Digital Storytelling and Entertainment (Lecture Notes in Computer Science)*, Stefan Göbel, Rainer Malkewitz, and Ido Iurgel (Eds.). Springer, Berlin, Heidelberg, 253–264. https://doi.org/10.1007/11944577_26
[56] Judith van Stegeren and Jakub Myśliwiec. 2021. Fine-tuning GPT-2 on annotated RPG quests for NPC dialogue generation. In *The 16th International Conference on the Foundations of Digital Games (FDG) 2021 (FDG'21)*. Association for Computing Machinery, New York, NY, USA, 1–8. https://doi.org/10.1145/3472538.3472595
[57] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is All you Need. In *Advances in Neural Information Processing Systems*, I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (Eds.), Vol. 30. Curran Associates, Inc., Long Beach, California, 11. https://proceedings.neurips.cc/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf
[58] Hao Wang. 2014. Automatic Narratives in MMORPGs. In *Handbook of Digital Games.* John Wiley & Sons, Ltd, Hoboken, New Jersey. https://doi.org/10.1002/9781118796443
[59] Georgios N Yannakakis, Antonios Liapis, and Constantine Alexopoulos. 2014. Mixed-initiative co-creativity. In *Proceedings of the 9th Conference on the Foundations of Digital Games.* FDG, Liberty of the Seas, Caribbean, 8. http://www.fdg2014.org/papers/fdg2014_paper_37.pdf
[60] Kristen K. Yu, Nathan R. Sturtevant, and Matthew Guzdial. 2021. Towards Disambiguating Quests as a Technical Term. In *The 16th International Conference on the Foundations of Digital Games (FDG) 2021 (FDG'21)*. Association for Computing Machinery, New York, NY, USA, 1–11. https://doi.org/10.1145/3472538.3472543

[61] Wenhao Yu, Chenguang Zhu, Zaitang Li, Zhiting Hu, Qingyun Wang, Heng Ji, and Meng Jiang. 2022. A Survey of Knowledge-Enhanced Text Generation. *ACM Comput. Surv.* 54, 11s, Article 227 (nov 2022), 38 pages. https://doi.org/10.1145/3512467

[62] Hanqing Zhang, Haolin Song, Shaoyu Li, Ming Zhou, and Dawei Song. 2022. A Survey of Controllable Text Generation using Transformer-based Pre-trained Language Models. https://doi.org/10.48550/arXiv.2201.05337

**Table 10: Examples of generated output by DRG-M**

| Generated Output |
|---|
| **Quest**: Go collect 15 gold nuggets.<br>**Title**: To Catch a Treasure<br>**Dialogue**: <Dodger hands you a small gold nugget.>These nuggets are the highest grade metal in a cave near the east side of the lava pools. The best you can hope for is to find them deep within the lava pools and carry them to the surface surface, and some time later, return to me. If you hurry...<Gadger's eyes glisten a bit.>you should make the journey to the northeast soon. You can even use the tunnel to reach a place you can collect nuggets from. |
| **Quest**: Go collect 15 gold nuggets.<br>**Title**: The Gold Nuggets<br>**Dialogue**: To collect the nuggets, go south across the border into Taurajo. Head through the bridge and cross the valley. Follow the road east until you hit the gold nugget mines. We must not let this happen again.<Vindicator Kalec sits in his throne.> |
| **Quest**: Go collect 15 gold nuggets.<br>**Title**: WANTED: Gold Nuggets<br>**Dialogue**: I've heard of a rogue barrow-wumpf named George Korson. Apparently he keeps a large cache of gold near his camp. I've never seen him before, but he's always at the bottom of that tree to my south. I'll bet he's been living off it for a while. Do ya know George? |
| **Quest**: Go collect 15 gold nuggets.<br>**Title**: Gold nuggets<br>**Dialogue**: <A goblin scrawl is scrawled on the ground. It reads with strange, scrawled language: “”Your next target is a few yards out on the ridge west of here. If you bring me back some of the gold that you collect on your travels, I will pay you a fair price.“”> |
| **Quest**: Inspect the Tarkman aircraft until you find a faulty one, then help rebuild it.<br>**Title**: Building a Better One<br>**Dialogue**: Well, there you are! I took some inspiration from my work in Zereth Mortis and created a blueprint based on the knowledge I gained there. Unfortunately, after many prototypes and adjustments, it is still not very accurate. The pilot, Fergus McEvoy, has been fixin' those propeller blades for the last couple of weeks and has already restored several of them. You'd better check on Fergus and his work, then take a look at the other propeller blades. |
| **Quest**: Inspect the Tarkman aircraft until you find a faulty one, then help rebuild it.<br>**Title**: Eating, Hunting, and Fighting All Around<br>**Dialogue**: Hey, Mechanic! Look at all the planes out here! Even the smallest little bit of damage might've lead to the whole mess taking a turn for the worse. Can you spare some time and help me out a bit? We need to ensure we keep the plane running properly, so we're going to have to upgrade some of the weapons. Our first target is the engine, isn't it? Go down there and find a failed one, and help me get it back to working order. |

**Table 11: Examples of generated output by DRG-M (Continued)**

| Generated Output |
|---|
| **Quest**: Inspect the Tarkman aircraft until you find a faulty one, then help fix it.<br>**Title**: Wrong Place at the Wrong Time<br>**Dialogue**: Just in case you didn’t notice all the noise coming from the north wing end of the bomber, there’s a very odd maintenance deficiency. The problem appears to stem from the faulty riveting mechanism on the wings, but it was recently restored. You’ll see it on the far side of the ramp when you enter the valley. |
| **Quest**: Inspect the Tarkman aircraft until you find a faulty one, then help fix it.<br>**Title**: I Never Wanted to Leave Nazmir<br>**Dialogue**: I’ve built up an incredible amount of trust with the locals in Nazmir. I’ve noticed that they have not been so welcoming to newcomers. I was able to borrow one of their flight simulators with the aid of the Alliance, and was able to fly in their skies for just two-tenths of a second. I was impressed by what I saw, and I’d like to give them a hand. As a matter of fact, I’ll pay you to take my flying machine and inspect the nearby airship until you find a faulty one, and then I’ll fix it. |
| **Quest**: Inspect the Tarkman aircraft until you find a faulty one, then help fix it.<br>**Title**: Tarkman Service<br>**Dialogue**: Your speed is great, Mechanic. If you see a Tarkman flight manual, you’ll know that it contains information on repairing damaged aircraft. This means you should be able to locate one of these aircrafts at any of our hangars. Unfortunately, the wings of the aircraft are bent and loose in the water, making them difficult to inspect. Could you check the wings for an item of the manual? |
| **Quest**: Inspect the Tarkman aircraft until you find a faulty one, then help fix it.<br>**Title**: Flight Quality<br>**Dialogue**: The Tarkman’s second-gen airframe should be more reliable than the first one. As you can see from the flying time in the schematic, this one has been damaged. It will be very difficult to fly safely if it breaks down or if the engine is broken, but if you can get this thing flying again, it should be a breeze. |

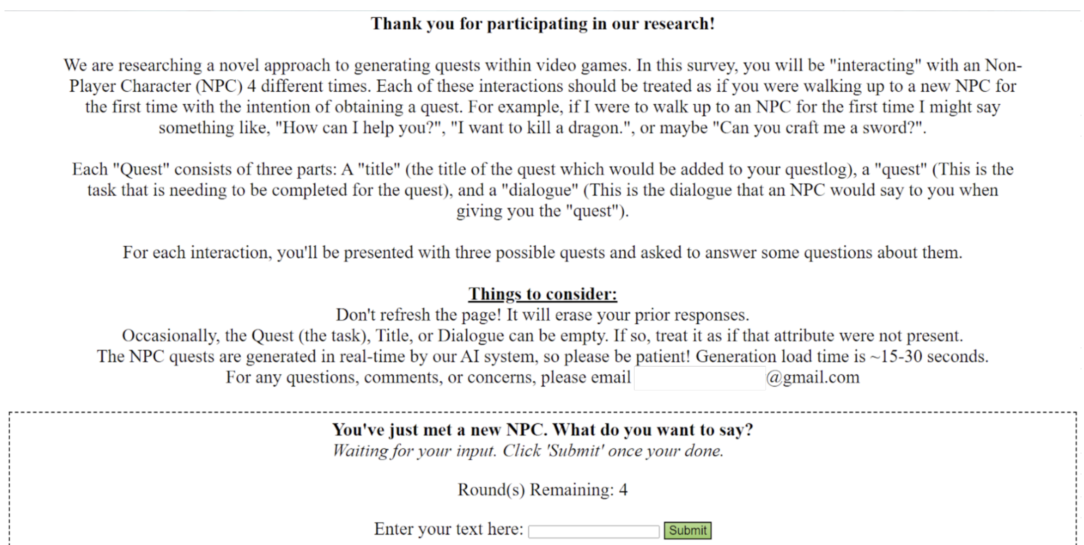
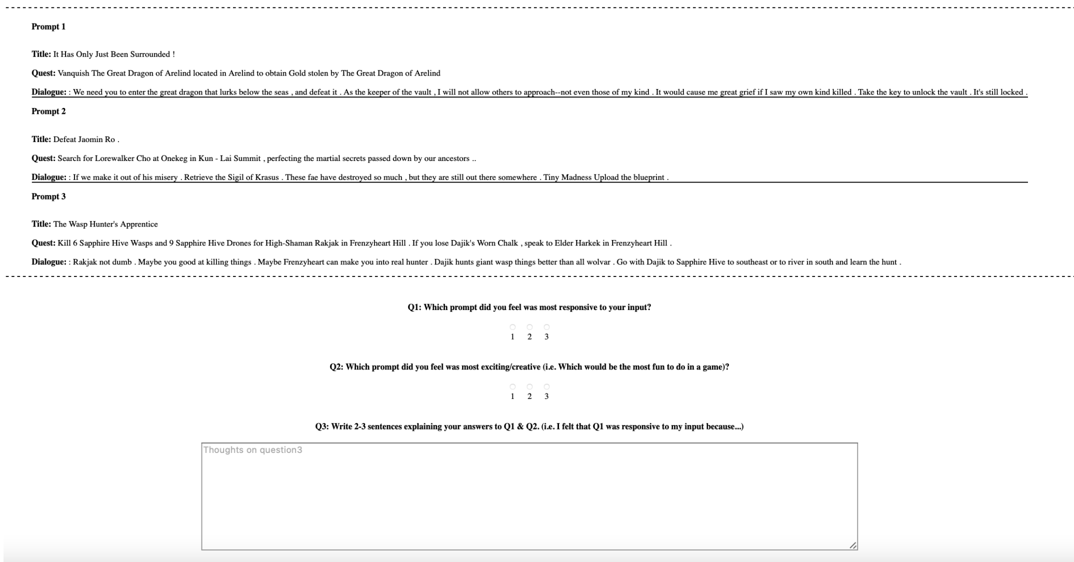



**Figure 5: Survey Website**



**Prompt 1**

**Title:** It Has Only Just Been Surrounded !

**Quest:** Vanquish The Great Dragon of Arelind located in Arelind to obtain Gold stolen by The Great Dragon of Arelind

**Dialogue:** : We need you to enter the great dragon that lurks below the seas , and defeat it . As the keeper of the vault , I will not allow others to approach--not even those of my kind . It would cause me great grief if I saw my own kind killed . Take the key to unlock the vault . It's still locked .

**Prompt 2**

**Title:** Defeat Jaomin Ro .

**Quest:** Search for Lorewalker Cho at Onekeg in Kun - Lai Summit , perfecting the martial secrets passed down by our ancestors ..

**Dialogue:** : If we make it out of his misery . Retrieve the Sigil of Krasus . These fae have destroyed so much , but they are still out there somewhere . Tiny Madness Upload the blueprint .

**Prompt 3**

**Title:** The Wasp Hunter's Apprentice

**Quest:** Kill 6 Sapphire Hive Wasps and 9 Sapphire Hive Drones for High-Shaman Rakjak in Frenzyheart Hill . If you lose Dajik's Worn Chalk , speak to Elder Harkek in Frenzyheart Hill .

**Dialogue:** : Rakjak not dumb . Maybe you good at killing things . Maybe Frenzyheart can make you into real hunter . Dajik hunts giant wasp things better than all wolvar . Go with Dajik to Sapphire Hive to southeast or to river in south and learn the hunt .

**Q1: Which prompt did you feel was most responsive to your input?**

1 2 3

**Q2: Which prompt did you feel was most exciting/creative (i.e. Which would be the most fun to do in a game)?**

1 2 3

**Q3: Write 2-3 sentences explaining your answers to Q1 & Q2. (i.e. I felt that Q1 was responsive to my input because...)**

Thoughts on question3

**Figure 6: Survey Website 2**