Dungeons and Dragons as a Dialog Challenge for Artificial Intelligence

Anonymous ACL submission

Abstract

AI researchers have posited Dungeons and Dragons (D&D) as a challenge problem to test systems on various language-related capabilities. In this paper, we frame D&D specifically as a dialogue system challenge, where the tasks are to both generate the next conversational turn in the game and predict the state of the game given the dialogue history. We create a gameplay dataset consisting of nearly 900 games, with a total of 7,000 players, 800,000 dialogue turns, 500,000 dice rolls, and 58 million words. We automatically annotate the data with partial state information about the game play. We train a large language model to generate the next game turn, conditioning it on different information. The LM can respond as a particular character or as the player who runs the game—i.e., the Dungeon Master (DM). It is trained to produce dialogue that is either in-character (roleplaying in the fictional world) or out-of-character (discussing rules or strategy). We perform a human evaluation to determine what factors make the generated output plausible and interesting. We further perform an automatic evaluation to determine how well the model can predict the game state given the history and examine how well tracking the game state improves its ability to produce plausible conversational output.

1 Introduction

Artificial Intelligence has a long and rich history of using games as challenge problems that led to advances in the field. In many cases, AI game-playing systems have gone on to rival human champions of the game.

Tackling board games like checkers, backgammon, and chess resulted in AI search algorithms like MiniMax and alpha-beta pruning as well as representations of the search space like game trees, and heuristic evaluation on non-terminal game state. Video games like StarCraft or classic Atari games have provided test beds for reinforcement learning where systems must learn game-playing policies by interacting with dynamic worlds where the game state is not as easily repressed as board games. Games like Jeopardy or Quiz Bowl have presented language-related challenges that advanced question-answering and information extraction (Ferrucci et al., 2010; Iyyer et al., 2014).

1.1 Challenge Problem for AI: Role Playing Games

Dungeons and Dragons has been identified as an appropriate challenge for the next stage of artificial intelligence (Ellis and Hendler, 2017; Louis and Sutton, 2018; Martin et al., 2018b). Ellis and Hendler (2017) proposed open-ended creative games like D&D as the next challenge for AI after the human-level successes by AI at Chess and Go, which are zero-sum, deterministic, sequential two-player games with perfect information. Louis and Sutton...
(2018) understood the importance of narrative in NLP/NLG and saw how cooperative story generation between humans already exists in these games and can be used for automated generation. Martin et al. (2018b) took this a step further and outlined some of the specific challenges D&D presents to the NLP community; such as a state of the game world distributed across the Dungeon Master (DM) and other players or dealing with the intrinsic rewards players get from taking certain actions that would not necessarily provide them with points or experience within the game.

Role playing games like Dungeons and Dragons are an interesting challenge problem for AI. Gameplay happens through language rather than moves on a game board. D&D involves multiple players who roleplay characters in a fantasy setting, guided by a Dungeon Master who sets obstacles and adventures and plays monsters. To have an AI successfully play D&D requires abilities like

- Language generation (multi-party dialog, generating descriptions of the world/actions, storytelling)
- Language understanding (knowledge acquisition and representation, state tracking, automated reasoning)
- Planning / strategic play during battles (similar to chess or go)

Is it possible to design an AI system that is capable of playing a game of D&D either as a character in the game or as the Dungeon Master using current AI technology?

1.2 Why is this the right time for this challenge?

Large scale neural language models like GPT have shown impressive generation results. Incorporating Neural LMs into a game exercises their strengths and exposes their weaknesses. D&D-style role playing games are a mix of language generation, language understanding, state-tracking and rule-following that make them a good research challenge that could advance AI research.

1.3 Research Challenges Presented by Role Playing Games

Instead of the game being a series of moves on a game board, Role Playing Games (RPGs) like Dungeons and Dragons (D&D) are language-based. Players create characters that have a class (wizard, fighter, thief) that denotes their abilities, and a fantasy race (like elf, dwarf, human). Players describe what they want their character to do, and roll dice to determine if they are successful. The dungeon master (DM) acts as the narrator who shapes the overall story. The DM describes scenarios and locations, and takes on the role of non-player characters (NPCs), and monsters.

A common element to the game play is an encounter with monsters. Battles are governed by rules, and unfold in a turn-based fashion where the DM controls the monsters and each player controls their character. Each player and monster has a health meter (called their HIT points), an armor class (which indicates the threshold of the dice roll needed to damage them), and a set of possible attack or move actions.

Table 2 provides example dialogue from a game of D&D being played between 3 players – Travis (playing a human fighter named Magnus Burnside), Clint (playing Merle Highchurch, a dwarf cleric), Justin (playing Taako an elf wizard), and DM Griffin. We add comments about each dialogue turn to describe what is happening in the game, and to highlight the challenges that would need to be addressed if an AI system were to play the game either as a player or as the DM.

1.4 Our Contributions

In this paper, we introduce a new dataset of “actual play” game transcripts. Each turn is labeled with state variables, character information, partial game state, and whether the conversational turn was in-character or out-of-character. Our data is a novel, large scale, real-world conversational dataset. It is unique in that the dialog turns are generated entirely through player collaboration and written interaction. Unlike existing dialog datasets, our data is modeling the Dungeons Dragons roleplaying game as a multi-party dialogue. We also train a large language model to perform response generation and game state tracking. Our dataset is interesting as a challenge for dialogue systems for the following reasons:

- It is naturally occurring dialog that is purely conversational.
- It is strongly history dependent – a substantive criticism of recent dialog datasets is their history independence (Mosig et al., 2020).
Griffin (DM) | A dwarf named Gundren Rockseeker has hired you to transport a wagonload of provisions to the rough-and-tumble settlement of Phandalin, which is a couple days’ travel to the southeast. A day and a half after leaving, you turn off the high road that connects the major cities on the coast onto a smaller trail that will lead you to Phandalin. This trail is not as well maintained, and bandits and outlaws have been known to lurk along the trail.

Griffin (DM) | Roll a perception check for me. Perception is a wisdom skill, so be sure to add your wisdom modifier.

Clint (out of character) | I got an eight.

Justin (out of character) | I got a six.

Travis (out of character) | I rolled a natural twenty plus my wisdom modifier is 23.

Griffin (DM) | With his eagle eyes, Magnus spots two dead horses lying in the middle of the road about 200 feet ahead of you.

Travis (in-character as Magnus) | I stop the wagon and motion silently to get the attention of Merle and Taako, and kinda pull them up towards the front of the wagon.

Griffin (DM) | As you warn them that shit has gone south, you notice a few goblins crouching in a part of the shaded woods off to the side of the road. Two of the goblins begin charging your wagon. How many goblins are there?

Travis (out of character) | There are three goblins; two of them are rushing the group, one is pretty heavily obscured by the brush, probably about 40 feet out, sort of between you and the dead horses laying in the middle of the road.

Clint (Merle) | I will cast sacred flame at the nearest one. If it fails a dexterity saving throw, it takes 6 points of damage.

Griffin (DM) | You attack. You launch some fire onto the goblin closest to the wagon. And with that, he looks like he is on death’s door. And the other goblin that you can see, the one that’s not in the brush somewhere, just sort of stops in his tracks. What do you do next?

<table>
<thead>
<tr>
<th>Player (character)</th>
<th>Game Dialogue</th>
<th>D&amp;D Game Description and AI challenges</th>
</tr>
</thead>
<tbody>
<tr>
<td>Griffin (DM)</td>
<td>A dwarf named Gundren Rockseeker has hired you to transport a wagonload of provisions to the rough-and-tumble settlement of Phandalin, which is a couple days’ travel to the southeast. A day and a half after leaving, you turn off the high road that connects the major cities on the coast onto a smaller trail that will lead you to Phandalin. This trail is not as well maintained, and bandits and outlaws have been known to lurk along the trail. Roll a perception check for me. Perception is a wisdom skill, so be sure to add your wisdom modifier.</td>
<td>This game is based on the D&amp;D starter adventure called “Lost Mine of Phadelver”. The adventure book is a mixture of rules and “boxed text” which is descriptive text for the DM to read aloud or paraphrase. See the appendix for the text that the DM is consulting. AI challenges: Generation of stories and descriptive text The previous text was descriptive text. Here the DM is asking the players to perform a game mechanic and referencing a game rule. This is called “out of character” dialogue. AI challenges: Knowledge base population (extraction of rules from a rulebook) Clint has rolled his dice. The number is low so his character fails the check. AI challenges: Multi-party dialogue Justin also fails. Neither character sees the thing that requires the perception check. Travis rolls high number and succeeds on the check. AI challenges: Understanding rules, determining success or failure The DM describes what happens as a result of the success. AI challenges: Reasoning about consequences of success or failure, descriptive text generation Travis is describing what he is doing using “in character” language. AI challenges: Persona-based chat The DM describes the start of a battle with several monsters. AI challenges: State tracking (in combat v. out of combat). Clint chooses an action based on what is allowed for his character class. He describes the rule that governs the spell in an out-of-character fashion. AI challenges: Intent detection (perform attack action against a particular goblin) The DM rolls for the monster, updates the state of its health meter, and describes the result of Merle’s attack. AI challenges: Reasoning about rules, state tracking (monster’s HIT points), descriptive text generation.</td>
</tr>
</tbody>
</table>
• It covers a spectrum of task oriented and non-task oriented (e.g. chit chat) dialog.
• It has many participants in the conversation, since there are several players in the game.
• It conveys narrative elements including descriptions of events, that denote changes in the state of the game.

We evaluate a state-of-the-art language model, to understand how conditioning generation on different elements improves the quality of the generated text.

2 Tasks

We trained a large language model (LLM) to perform two tasks: Next Utterance Prediction and Game State Tracking.

Next Utterance Prediction We trained our language model on a corpus of human conversations (see Datasets section) to predict the next utterance. We varied the conditioning information to examine the effects on the quality of predicted next utterance. In all variations, we included the conversational history as input. We used most recent 7 conversational turns as input (in our example in the Table above, that would be from Griffin saying “With his eagle eyes...” up until “What do you do next?”). Given the conversational input (and other input in the variant models), the LLM must generate the next utterance, such that it is both interesting and a plausible next turn in the D&D game.

Game State Tracking In this task, rather than outing the next utterance, we had the model predict the game state for a given dialogue turn in the conversation. We have kept the state definition similar to task-oriented dialogue state tracking (DST). In DST, the dialogue state is a collection of slot-value pairs. In our case, each slot is a state variable feature related to D&D games. Our target slot values do not need to appear as a word in the dialogue context. We track several game states aspects including some that remain relatively static throughout the game (character attributes like their pronouns, class, fantasy race, and their inventory), some that change periodically (like being in combat or out of combat), and some that change from turn to turn (like what action the player is taking).

In this paper we attempt to track the following game state variables:

• Character specific state variables (remain constant throughout the game). These include Player ID, Character’s name, pronouns, class and fantasy race, Items in character’s inventory.
• Combat related state variables (change periodically) These include Whether the players are engaged in combat, Monsters in combat
• Dialog turn level state variables. These are actions related to dice rolls .

3 Datasets

For this paper, we have created a novel dataset for our dialogue-oriented test of AI’s ability to play Dungeons Dragons. We investigated two sources of data of people playing the game. The first source of data was “Actual Play” podcasts, where people record themselves actually playing RPGs. The second Play-By-Post data scraped from a web forum where people play by taking turns posting on the forum to describe their move. Ultimately, we focused primarily on the Play-By-Post data rather than the Actual Play Podcast data, since our automatic transcripts of the podcast audio had some speaker diarization errors and misrecognized words. Furthermore, using data from podcasts is not straightforward due to the inclusion of extraneous content, such as advertisements (Reddy et al., 2021).

3.1 Actual Play Podcasts

Actual Play podcasts are a genre of podcasts where people record themselves playing RPGs. We collected lists of popular actual play podcasts from the web, and expanded the list by searching for additional shows with relevant keywords including Actual Play, Role Playing Game, D&D 5e. Extracted all podcast episodes for these candidate podcast shows, and transcribed all podcast episodes using Google Cloud Speech-to-Text API (discarding about 15% of the episodes because of download errors or transcription failures). Table 3 shows a fragment of one of the podcast transcript. Table 4 summarizes the amount of Actual Play podcast data that we collected.

Dealing with speech data presents several challenges, including large audio files (the mean episode length was 1 hour and 17 minutes, with a max of 7 hours and 23 minutes), automatic speech recognition errors, and speaker diarization errors
Speaker 1 Now, let’s meet the cast of DPR live who are often referred to as the tide 5. Brachii, the Goliath with arms Made of Stone his Barbarian rage will cut to the Bone. Ale and the Angelic Sorcerer of Helm he’ll kill you with charm. It is known through the realm. Snatch is a halfling, you know, like a hobbit from Lord of the Rings and apple munching klepto. He’ll steal all of your things. Pandora’s a war priest who fights with goodness and piety. Unfortunately, he also fights with a crippling case of anxiety. O’ree Keyes. That’s me. Wait, that’s me

Speaker 4 a real devil with a

Speaker 1 lute with courage and Valor and an eye patch to boot. Alrighty, let’s do this. It’s time to roll dice. Let’s all get Brody. It stays people who

Speaker 4 live. Nice

Speaker 3 nice. Hello everyone. Welcome. This is Dice paper roll live. Thank you all for coming. My name is Emil and I play brachii the Goliath Barbarian.

Speaker 1 Hi everybody. My name is Greg. I’m playing a land ay-ay-ron a on an AC Mars Sorcerer of Helm. Hey folks, my name is Ben and I play snatch of the halfling Rogue Uh, I’m Jack I play a re Keys song and I’m a tiefling Bard. I’m real nervous. There’s a lot of people here today.

Speaker 2 And I’m Dan I play thund all the human fighter and cleric of Tempest. But I am also the dungeon master

Table 3: Example of an automatic transcript of the “Dice Paper Role” actual play podcast

<table>
<thead>
<tr>
<th>Actual Play Podcast Corpus</th>
<th>Play-By-Post Corpus</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Shows</td>
<td>555</td>
</tr>
<tr>
<td>Number of Episodes</td>
<td>38,713</td>
</tr>
<tr>
<td>Number of Episodes (transcribed)</td>
<td>32,602</td>
</tr>
<tr>
<td>Average no. of speakers per ep.</td>
<td>2.23</td>
</tr>
<tr>
<td>Average no. of turns per episode</td>
<td>597</td>
</tr>
<tr>
<td>Average no. of words per episode</td>
<td>11,374</td>
</tr>
<tr>
<td>Total no. of turns</td>
<td>19,478,136</td>
</tr>
<tr>
<td>Total no. of words</td>
<td>370,824,073</td>
</tr>
<tr>
<td>Number of campaigns</td>
<td>896</td>
</tr>
<tr>
<td>Average players per campaign</td>
<td>8</td>
</tr>
<tr>
<td>Average turns per campaign</td>
<td>910</td>
</tr>
<tr>
<td>Average words per campaign</td>
<td>64,941</td>
</tr>
<tr>
<td>Total turns</td>
<td>815,106</td>
</tr>
<tr>
<td>Total words</td>
<td>58,187,526</td>
</tr>
<tr>
<td>Average dice rolls per campaign</td>
<td>594</td>
</tr>
<tr>
<td>Average annotations per campaign</td>
<td>94</td>
</tr>
<tr>
<td>Total dice rolls</td>
<td>532,270</td>
</tr>
<tr>
<td>Total annotations</td>
<td>84,447</td>
</tr>
</tbody>
</table>

Table 4: Statistics for the actual play podcast corpus that we constructed

(2.2 speakers per episode is certainly an underestimate given that most RPGs have a DM and multiple players). Although podcast data is an interesting and plentiful source of game-play data, we ultimately decided to use text-based game play data instead.

3.2 Play-by-Post Data

Another source of game-play data are internet discussion forums where users engage in play-by-post of D&D and other roleplaying games. Figure 2 shows an example of part of the gameplay from the “play-by-post” forums from D&D Beyond. These forums contain conversations similar to actual play podcasts, but players take turns writing a forum post describing their play instead of recording audio. Diarization and transcription errors are not a concern in play-by-post data, since it starts as text instead of audio.

The play-by-post data also has partial annotations. D&D Beyond provides a mechanism in its forum to roll dice using a “roll” tag. Their dice roller allows players to conduct the rolls that are used for D&D ability checks and in combat. We are able to locate the span of each post that corresponds to a dice roll, and extract information about the roll like what kind of die was used and what the total was.

Table 5 below summarizes the amount of play-by-post data that we collected from the D&D Beyond website (with permission from D&D Beyond).

We designed a set of rule-based heuristics to extract game state information from the play by post. These were implemented using regular expressions and NLP tools like named entity recognizers (Gardner et al., 2018). Although this heuristically extracted information is not perfect, it provides a reasonable approximation of the game state. It is useful for testing whether large language models can benefit from inclusion of complex state information for next utterance prediction and whether LLMs can be used for state tracking. We designed rules to extract state information relating to charac-
ter properties, combat and player actions.

**Character properties**

- **Name**: perform NER on all the player’s turns in a campaign. The character’s name is assigned to be the player’s most frequently mentioned name, on the assumption that they tend to describe their own character’s actions.

- **Class**: count how many times each D&D class is mentioned by each player. Most frequently mentioned class is their character’s class.

- **Race**: On a player’s first turn, check whether any of the D&D races are mentioned. Assign it to character. If not, apply other heuristics to guess it.

- **Pronouns**: Count pronouns mentioned by a player. Assign their character’s pronouns to be the most frequent pronouns used by the player.

- **Inventory**: Regex that matches items occurring after character’s personal pronouns (e.g. her sword).

- ** Spells known**: Regex that matches cast followed by a spell name

The DM is assumed to be the player who has the first post in the game. The DM’s entries in the dataset are scrubbed of other character properties, since they play multiple NPCs and monsters.

**Combat**

- **We detect the start of combat when there is a roll for initiative, or when there are attack rolls before initiative (which happens in surprise attacks).**

- **Combat continues while there are attack rolls happening.**

- **Combat concludes after there are no rolls for a number of turns.**

- **In a combat span, we extract a list of monsters mentioned, and heuristically guess the number of each kind of monster.**

**Combat**

- **Dice rolls are marked in D&D Beyond posts. We detect the associated actions based on the kind of die used (D20 = a check, other dice are used for calculating damage if an attack check is successful)**

- **We use a regex to match the nearest pattern, which includes attack or a list of abilities like acrobatics, animal handling, arcana, athletics, etc.**
• Damage rolls are matched with damage, dmg, cure, heal, healing, points.

These heuristics help to obtain at least one of the control features for around 58% of all conversational turns. We train a convolutional neural network (CNN) classifier using these conversational turns to predict all of the above control features for each conversational turn in training data.

The CNN classifier only uses current post text as input (no context). Table 6 estimates gives an estimate of the CNN’s performance on filling in the state variables where the rule-based heuristic did not extract a value.

3.3 In-Character Versus Out-Of-Character Text

In addition to labeling the game states in our Play by Post data, we also labeled the text of each turn as being either spoken in-character (IC) or out-of-character (OOC). To do so, we crawled another Play by Post forum hosted at Giant in the Play- ground\(^2\), where play happens on two discussion boards – one in-character and one out-of-characters. In the IC board, they also mask OOC actions with spoiler tags. We train a classifier to predict IC versus OOC text, and then apply it to all text in our D&D Beyond datasets.

4 Models

For our large language model, we use a 64B parameter version of Google’s LaMDA language model (Thoppilan et al., 2022), which was trained on conversations. LaMDA is similar to other Transformer-based pre-trained language models like GPT-3. As with other pre-trained language models (Howard and Ruder, 2018), LaMDA can be fine-tuned to different tasks. The two tasks that we fine-tune LaMDA to perform are game state tracking and response generation. In both cases, the LLM can be thought of as a function that maps inputs onto an output. For instance, game state tracking is a language understanding task where the function takes in inputs like \( f(\text{current utterance, previous state, history}) \rightarrow \text{new state} \), and response generation is a language generation task where \( f(\text{current state, history}) \rightarrow \text{next utterance} \). The LLM functions are trained via the fine tuning process.

In our experiments we try a variety of different inputs to our LLM functions to see how they enable better learning of the tasks. We train our LLMs on the conversation history (which is typical in dialog modeling) and we also augment the conversations by conditioning other explicit signals. These conditioning signals can be thought of as sophisticated “control features”, inspired by the CTRL language model (Keskar et al., 2019). During training, the model learns a relationship between the control features and appropriate responses. In turn, during inference, one can explicitly influence dimensions of the conversation – enabling more compelling dialogue – by setting the values of control features. These control features can be set dynamically, without necessitating fine-tuning or additional post-processing. Table 7 describes the control features we have proposed and describes how they could steer generation.

4.1 Baseline Pre-Training Data

LaMDA is trained on turn-based conversational data. For a conversation of length \( n \), LaMDA takes as input the first \( n - 1 \) turns, and the \( n \)th turn as the target. The example below illustrates the input and target for a three-turn conversation.

**INPUT:**

- TURN 1: I like the ocean
- TURN 2: Why?

**TARGET:**

- TURN 3: It’s beautiful, vast, and mysterious.

**TASK:**

- TURN\(_1\), TURN\(_2\), ..., TURN\(_n\) \rightarrow TURN\(_{n+1}\)

4.2 D&D Fine-Tuning Data

Here is an example of the data used in our versions of LaMDA that are fine-tuned to our D&D data.

**TURN 1:**

Text: You attack. You launch some fire onto the goblin closest to the wagon. And with that, he looks like he is on death’s door. And the other goblin that you can see, the one that’s not in the brush somewhere, just sort of stops in his tracks. What do you do next?

**Player ID:** 0
**Character:** Dungeon Master
**Race:** N/A
**Class:** Dungeon Master
**Pronouns:** N/A
**Inventory:** N/A
**In combat?** Yes
**Action:** Unknown
Table 6: The estimated performance of our CNN classifier on predicting state values for turns where our rule-based heuristics did not predict a value.

<table>
<thead>
<tr>
<th>State variable</th>
<th>Model</th>
<th>Type</th>
<th>Multi-valued</th>
<th>Availability</th>
<th>Evaluation metric</th>
<th>Performance</th>
</tr>
</thead>
<tbody>
<tr>
<td>Character</td>
<td>Span labeller</td>
<td>Text</td>
<td>No</td>
<td>42%</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Race</td>
<td>Classifier</td>
<td>Text</td>
<td>No</td>
<td>&gt;58%</td>
<td>Macro AUC</td>
<td>0.45</td>
</tr>
<tr>
<td>Class</td>
<td>Classifier</td>
<td>Text</td>
<td>No</td>
<td>&gt;75%</td>
<td>Macro AUC</td>
<td>0.71</td>
</tr>
<tr>
<td>Gender</td>
<td>Classifier</td>
<td>Text</td>
<td>No</td>
<td>42%</td>
<td>Macro AUC</td>
<td>0.92</td>
</tr>
<tr>
<td>Inventory</td>
<td>Span labeller</td>
<td>Text</td>
<td>Yes</td>
<td>11%</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>In combat?</td>
<td>Classifier</td>
<td>Score</td>
<td>No</td>
<td>100%</td>
<td>Accuracy</td>
<td>0.91</td>
</tr>
<tr>
<td>Action</td>
<td>Classifier</td>
<td>Text</td>
<td>Yes</td>
<td>20%</td>
<td>Macro AUC</td>
<td>0.92</td>
</tr>
</tbody>
</table>

Figure 2: An example post from the Giant in the Playground forums where the text is segmented into in-character (IC) and out-of-character (OOC) portions.

Table 7: Our LLMs are conditioned on a variety of control features that allow the models to better learn what kind of text to generate for the next utterance prediction task.

<table>
<thead>
<tr>
<th>Control Feature</th>
<th>Description</th>
<th>Expected Impact on Model’s Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>Player ID</td>
<td>Player writing a given dialog turn</td>
<td>Connects the current turn to the player’s previous turns, which is important in multi-party conversations.</td>
</tr>
<tr>
<td>IC versus OOC</td>
<td>Whether a player is in-character or out-of-character for a given dialog turn</td>
<td>Changes whether the generated text is more like descriptive text found in a novel, or more like a discussion of rules and strategies.</td>
</tr>
<tr>
<td>Character Name</td>
<td>Name of the character being played by the player of a given dialog turn</td>
<td>IC descriptions use the character’s name.</td>
</tr>
<tr>
<td>Character Class</td>
<td>D&amp;D classes(^3)</td>
<td>Character classes perform different actions (e.g. wizards cast spells, thieves pick locks).</td>
</tr>
<tr>
<td>Character Race</td>
<td>D&amp;D fantasy races(^4)</td>
<td>Different physical characteristics (e.g. halflings are small, dragonborn have scales).</td>
</tr>
<tr>
<td>Character Pronoun</td>
<td>The character’s pronouns</td>
<td>Uses the correct pronouns when describing the character.</td>
</tr>
<tr>
<td>Character Actions</td>
<td>List of actions taken by the character in the current turn</td>
<td>Allows a description to be generated for a given action. The action can be thought of as a goal for the description.</td>
</tr>
<tr>
<td>Combat</td>
<td>Whether the players are currently engaged in combat or not during a given dialog turn</td>
<td>Affects the likelihood of actions (e.g. attacks are more likely during combat and investigations checks are more likely outside of combat).</td>
</tr>
</tbody>
</table>
4.3 Next Utterance Prediction Models

**LLM-DIALOG** We call our baseline model LLM-DIALOG. It is a LaMDA dialogue model that does not use not use any D&D data.

**LLM-DND** LLM-DIALOG that has been finetuned on Play-by-post D&D gameplay dataset using *no* control features

**TASK**: TURN$_1$ SEP TURN$_2$ SEP … SEP TURN$_{N-1}$ → TURN$_N$

where TURN$_i$ denotes text along with player id for $i$th turn and SEP is the separator. Note that the last turn TURN$_N$ is the target and previous conversation history is the input for this task.

**LLM-DND-ALL-CTRL** LLM-DIALOG that has been finetuned on Play-by-post D&D gameplay dataset using control features for *all* dialog turns up to the last or target turn.

**TASK**: TURN$_1$ STATE$_1$ SEP TURN$_2$ STATE$_2$ SEP … SEP TURN$_{N-1}$ STATE$_{N-1}$ SEP STATE$_N$ → TURN$_N$

where STATE$_i$ denotes the set of control features for $i$th dialog turn, e.g., in-combat : 0 | class : fighter | character : magnus | race : human | pronouns : he/him | inventory : axe | actions : attack

**LLM-DND-PREV-CTRL** LLM-DIALOG that has been finetuned on Play-by-post D&D gameplay dataset using control features for all *previous* dialog turns to the last or target turn.

**TASK**: TURN$_1$ STATE$_1$ SEP TURN$_2$ STATE$_2$ SEP … SEP TURN$_{N-1}$ STATE$_{N-1}$ → TURN$_N$

**LLM-DND-RECENT-CTRL** LLM-DIALOG that has been finetuned on Play-by-post D&D gameplay dataset using control features for most recent dialog turn before the last or target turn.

<table>
<thead>
<tr>
<th>Model</th>
<th>Perplexity</th>
<th>Token accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>LLM-DIALOG</td>
<td>2.65</td>
<td>44.61</td>
</tr>
<tr>
<td>LLM-DND</td>
<td>2.50</td>
<td>46.92</td>
</tr>
<tr>
<td>LLM-DND-PREV-CTRL</td>
<td>2.51</td>
<td>46.84</td>
</tr>
<tr>
<td>LLM-DND-RECENT-CTRL</td>
<td>2.34</td>
<td>49.67</td>
</tr>
<tr>
<td>LLM-DND-ALL-CTRL</td>
<td>2.37</td>
<td>49.02</td>
</tr>
<tr>
<td>LLM-DND-Podcast</td>
<td>2.53</td>
<td>46.64</td>
</tr>
<tr>
<td>LLM-Podcast</td>
<td>2.69</td>
<td>44.6</td>
</tr>
</tbody>
</table>

Table 8: Perplexity and token accuracy of our models after finetuning to our data

**TASK**: TURN$_1$ STATE$_1$ SEP TURN$_2$ STATE$_2$ SEP … SEP TURN$_{N-1}$ STATE$_{N-1}$ → TURN$_N$

**LLM-Podcast** LLM-DIALOG that has been finetuned on (transcribed) Dungeons Dragons podcasts using no control features

**LLM-DND-Podcast** LLM-DIALOG that has been finetuned on both (transcribed) Dungeons Dragons podcasts and using Play-by-post D&D gameplay dataset no control features

4.4 Dev set perplexity during training

Each of our models starts from a pretrained LaMDA model trained for 600K steps and then is finetuned for a further 60K steps. Figure 8 plots the Negative log perplexity on our development set, and Table 7 shows the final perplexity and token accuracies on the dev set. At the end of finetuning, the models with the best perplexity scores and the best token accuracy scores were LLM-DND-RECENT-CTRL and LLM-DND-ALL-CTRL.
5 Manual Evaluation

To evaluate the quality of our models for the task of next utterance prediction in D&D, we perform a human evaluation. We recruited professional raters to perform a manual evaluation. They read a version of the content that was provided to the models – the seven turns of conversational history plus a list of players and the names/classes of the characters that they played. Then they were shown several model outputs for the context (or the “gold”, which was the actual next turn in the game). The annotators asked to rate each output along the three dimensions, following the evaluation procedure used for the Meena LM (Adiwardana et al., 2020):

- Does the response make sense? (yes/no) the response specific? (yes/no)
- How interesting is the response? (10 point scale)

The full annotator instructions are given in the Appendix. A mock up of the user interface is given in Figure 4.

5.1 Raters

Because of the specialized nature of the D&D domain, we recruited 6 professional raters rather than crowd workers to perform the task. The raters were selected based on their professed interest in the fantasy genre, and on their background with D&D. All raters were fantasy fans, and 5 of the 6 had played D&D. 3 raters had been DM in a game before.

5.2 Agreement

Our raters annotated 500 system outputs with 3-way redundancy on each output. For the binary sense and specific scores, the pairwise annotator agreement was 0.8, with a chance-adjusted Radolph Kappa score of 0.6. For the scalar interestingness scores, the Kendall’s Tau correlation was 0.46.

5.3 Model comparison

Table 9 shows the average sense, specific and interestingness scores for the systems, and for the human-written gold response. All of the D&D adapted systems outperform the vanilla dialogue system. The added control features do not seem to differ substantially from the LLM that is adapted to the D&D data without any control features.

We further analyzed the systems to see if the in-character versus out-of-character turns made a difference. Table 10 shows that interestingness substantially increased on in-character turns than when the output was generated out-of-character. Because our CTRL models allowed the system to intentionally generate in-character responses, it was able to intentionally produce in-character responses, resulting in substantially improved interestingness scores for in-character turns.

Table 11 shows a head-to-head comparison of systems based on what fraction of time one system’s interestingness score was higher than another system’s. In this analysis we found that the systems with the highest average head-to-head comparison were all of the systems that used the CTRL variables.

5.4 Qualitative Example

Table 12 shows example outputs from different versions of our LLMs. We gave the models the final seven turns in the game play example –from “With his eagle eyes, Magnus spots two dead horses…” to “... What do you do next?”, and then generate what each model thinks the next utterance ought to be. The vanilla dialogue system with no D&D data generates an irrelevant reply about Discord. The dialogue system fine tuned on D&D data without state generates a relevant response. The dialogue system that adds conditioning control features allows the model to generate specific replies for each of the different characters. These replies are relevant to the character classes – the wizard casts a spell, and the Dwarf cleric shouts the name of a Dwarf god.

6 Game State Tracking Model

We conducted one final pilot experiment to try to evaluate if a LLM could be fine-tuned to perform game state tracking for D&D using our heuristically annotated partial game state features. We trained LLM-DND-GST (Game State Tracking). It is a LLM-D十里 that has been finetuned on Play-by-post D&D gameplay dataset using control features or state variables for all previous dialog turns to the last turn (and all dialog turns including the last turn) to predict the dialog state of the last turn.

TASK: TURN1 STATE1 SEP TURN2 STATE2 SEP ... SEP TURNN-1 STATEN-1 TURN)N → STATEN

5 10
Figure 4: The user interface that our raters used to evaluate the quality of our models’ next utterance prediction.

Table 9: Average sense, specific and interestingness scores for our systems, and for human-written gold responses.

<table>
<thead>
<tr>
<th>Context</th>
<th>Player 10 - Del</th>
<th>Player 12 - Aramis (Bard)</th>
<th>Player 8 - Xavier (Bard)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Player 10 - Del (Barbarian)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Player 0 - Dungeon Master</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Player 12 - Aramis (Bard)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Player 8 - Xavier (Bard)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Player 10 - Del (Barbarian): Del gives a final kick to the dead orc that he just killed then proceeds to hunt through the main room then the store room, looking for his axe and other gear.</td>
<td>Does this response make sense?</td>
<td>Makes sense</td>
<td>Does not make sense</td>
</tr>
<tr>
<td></td>
<td>Response 1</td>
<td>Del quickly searches the store room, grabbing whatever weapons he find useful. He then heads back to the others “I found another crossbow and arrows, and a dagger. I have a short bow, but I’ll leave it here if someone else wants it. I also found one of my javelins that I was missing.”</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Response 2</td>
<td>Del takes the crossbow and tells Xavier to find his axe and keep good care of him</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Response 3</td>
<td>Anyone see my axe, and leather?</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Response 4</td>
<td>Noticing Del is searching the same room. Del calls out to him “X. I just killed an orc you could use its stuff if you like.”</td>
<td></td>
</tr>
</tbody>
</table>

Table 10: Analysis of in-character (IC) versus out-of-character (OOC) turns. Numbers are IC/OOC scores.

<table>
<thead>
<tr>
<th>Context</th>
<th>Player 10 - Del</th>
<th>Player 12 - Aramis (Bard)</th>
<th>Player 8 - Xavier (Bard)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Player 10 - Del (Barbarian)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Player 0 - Dungeon Master</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Player 12 - Aramis (Bard)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Player 8 - Xavier (Bard)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Player 10 - Del (Barbarian): Del gives a final kick to the dead orc that he just killed then proceeds to hunt through the main room then the store room, looking for his axe and other gear.</td>
<td>Does this response make sense?</td>
<td>Makes sense</td>
<td>Does not make sense</td>
</tr>
<tr>
<td></td>
<td>Response 1</td>
<td>Del quickly searches the store room, grabbing whatever weapons he find useful. He then heads back to the others “I found another crossbow and arrows, and a dagger. I have a short bow, but I’ll leave it here if someone else wants it. I also found one of my javelins that I was missing.”</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Response 2</td>
<td>Del takes the crossbow and tells Xavier to find his axe and keep good care of him</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Response 3</td>
<td>Anyone see my axe, and leather?</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Response 4</td>
<td>Noticing Del is searching the same room. Del calls out to him “X. I just killed an orc you could use its stuff if you like.”</td>
<td></td>
</tr>
</tbody>
</table>

Table 9: Average sense, specific and interestingness scores for our systems, and for human-written gold responses.

<table>
<thead>
<tr>
<th>System</th>
<th>Sense</th>
<th>Specific</th>
<th>Avg. Interesting</th>
<th>Avg. Interesting Rank</th>
</tr>
</thead>
<tbody>
<tr>
<td>LLM-Dialog</td>
<td>0.81</td>
<td>0.85</td>
<td>3.57</td>
<td>3.79</td>
</tr>
<tr>
<td>LLM-DND</td>
<td>0.9</td>
<td>0.9</td>
<td>3.91</td>
<td>3.46</td>
</tr>
<tr>
<td>LLM-DND-ALL-CTRL</td>
<td>0.87</td>
<td>0.88</td>
<td>3.92</td>
<td>3.43</td>
</tr>
<tr>
<td>LLM-DND-PREV-CTRL</td>
<td>0.86</td>
<td>0.88</td>
<td>3.96</td>
<td>3.35</td>
</tr>
<tr>
<td>LLM-DND-RECENT-CTRL</td>
<td>0.88</td>
<td>0.9</td>
<td>3.96</td>
<td>3.46</td>
</tr>
<tr>
<td>GOLD</td>
<td>0.92</td>
<td>0.92</td>
<td>4.17</td>
<td>3.11</td>
</tr>
</tbody>
</table>

Table 10: Analysis of in-character (IC) versus out-of-character (OOC) turns. Numbers are IC/OOC scores.

<table>
<thead>
<tr>
<th>System</th>
<th>Sense</th>
<th>Specific</th>
<th>Avg. Interesting</th>
<th>Avg. Interesting Rank</th>
</tr>
</thead>
<tbody>
<tr>
<td>LLM-Dialog</td>
<td>0.80/0.81</td>
<td>0.84/0.85</td>
<td>3.60/3.54</td>
<td>4.10/3.41</td>
</tr>
<tr>
<td>LLM-DND</td>
<td>0.89/0.91</td>
<td>0.92/0.89</td>
<td>4.09/3.69</td>
<td>3.59/3.29</td>
</tr>
<tr>
<td>LLM-DND-ALL-CTRL</td>
<td>0.90/0.83</td>
<td>0.91/0.85</td>
<td>4.28/3.47</td>
<td>3.28/3.62</td>
</tr>
<tr>
<td>LLM-DND-PREV-CTRL</td>
<td>0.87/0.85</td>
<td>0.89/0.87</td>
<td>4.23/3.63</td>
<td>3.33/3.37</td>
</tr>
<tr>
<td>LLM-DND-RECENT-CTRL</td>
<td>0.91/0.85</td>
<td>0.93/0.87</td>
<td>4.38/3.45</td>
<td>3.10/3.63</td>
</tr>
<tr>
<td>GOLD</td>
<td>0.95/0.88</td>
<td>0.94/0.89</td>
<td>4.63/3.61</td>
<td>2.84/3.44</td>
</tr>
<tr>
<td>Test</td>
<td>Dialog</td>
<td>DND</td>
<td>DND-ALL-CTRL</td>
<td>DND-PREV-CTRL</td>
</tr>
<tr>
<td>---------------</td>
<td>--------</td>
<td>---------</td>
<td>--------------</td>
<td>---------------</td>
</tr>
<tr>
<td>Dialog</td>
<td>-</td>
<td>0.37</td>
<td>0.37</td>
<td>0.37</td>
</tr>
<tr>
<td>DND</td>
<td>0.63</td>
<td>-</td>
<td>0.46</td>
<td>0.47</td>
</tr>
<tr>
<td>DND-ALL-CTRL</td>
<td>0.63</td>
<td>0.54</td>
<td>-</td>
<td>0.47</td>
</tr>
<tr>
<td>DND-PREV-CTRL</td>
<td>0.63</td>
<td>0.53</td>
<td>0.53</td>
<td>-</td>
</tr>
<tr>
<td>DND-RECENT-CTRL</td>
<td>0.63</td>
<td>0.53</td>
<td>0.53</td>
<td>0.50</td>
</tr>
<tr>
<td>GOLD</td>
<td>0.69</td>
<td>0.60</td>
<td>0.60</td>
<td>0.56</td>
</tr>
</tbody>
</table>

Table 11: Head-to-head comparison of systems of how often one system’s interestingness score was higher than another's.

<table>
<thead>
<tr>
<th>Model</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>LLM-Dialog</td>
<td>moon_monk you can just join our discord conversation. it's free and easier to deal with</td>
</tr>
<tr>
<td>LLM-DND</td>
<td>I cast Ice Knife (I’m assuming this is not one of the ones I can’t do?)</td>
</tr>
<tr>
<td>LLM-DND-PREV-CTRL</td>
<td>Responding as Magnus (human fighter, inventory: axe) In combat I turn from the goblin and go to help Merle and Taako with getting this wagon moving again. Magnus can’t see them anymore with such a dense forest. They’re likely in a battle right now... I have to find them as fast as I can!</td>
</tr>
<tr>
<td>LLM-DND-PREV-CTRL</td>
<td>Responding as Taako (elf wizard, inventory: spellbook) In combat If the goblins still don’t look like they’re getting closer, Taako will yell, “Alright you little monsters! Let’s see what ya got!” And he’ll cast Chill Touch on one of the goblins that isn’t dead/on its doorstep. Attack: 22 Damage: 6</td>
</tr>
<tr>
<td>LLM-DND-PREV-CTRL</td>
<td>Responding as Mearle (dwarf cleric, inventory: holy symbol) In combat Merle steps away from the wagon. He draws his sword and takes aim at the hidden goblin! Attack: 27 Damage: 8 He puts his free hand on the pommel of his sword and shouts his battle cry, &quot;For Clangeddadin!&quot; His eyes glow brightly red as he strikes!</td>
</tr>
</tbody>
</table>

Table 12: Example outputs our models for the next utterance predicted, given the sample gameplay given in Table 2 as input.
Table 13: Average accuracy for our Game State Tracking LLM on the slot-filling our state variables, compared to a majority class baseline.

<table>
<thead>
<tr>
<th>State variable</th>
<th>Majority</th>
<th>LLM-DND-GST</th>
</tr>
</thead>
<tbody>
<tr>
<td>All</td>
<td>72.45</td>
<td>81.48</td>
</tr>
<tr>
<td>Combat</td>
<td>88.46</td>
<td>82.41</td>
</tr>
<tr>
<td>Character Class</td>
<td>58.24</td>
<td>76.37</td>
</tr>
<tr>
<td>Character Name</td>
<td>58.24</td>
<td>76.92</td>
</tr>
<tr>
<td>Character Race</td>
<td>74.73</td>
<td>78.57</td>
</tr>
<tr>
<td>Character Pronouns</td>
<td>58.24</td>
<td>86.81</td>
</tr>
<tr>
<td>Character Actions</td>
<td>80.22</td>
<td>84.61</td>
</tr>
</tbody>
</table>

We analyzed the accuracy of the LLM-DND-GST model its ability to do slot-filling for each of the individual game states, and compared its performance to a simple baseline that always output the the majority class. The results are shown in Table 13. The average accuracy of the dialogue state tracker is better than the majority class baseline, but likely falls short of being useful when it comes to joint accuracy. The joint accuracy for the Majority class baseline is 58.24 and the joint accuracy for LLM-DND-GST is 58.24. This suggests that accurately tracking the state of the game may require additional machinery beyond a fine-tuned LLM.

7 Limitations

One limitation of our human evaluation is that it is a static evaluation. The raters are simply reading the outputs of the model, and there is no interactive evaluation wherein they engage in gameplay with the system. An interactive user-study would be required before any claims could be made about how well AI is able to play D&D alongside human players.

Because our state information was created heuristically, and therefore it potentially contains errors. It is also incomplete. There are several kinds of state tracking variables that would be useful to include, but were not possible to heuristically extract from our data. These include:

- Current HIT points totals for each character and monster. This is perhaps the most relevant state that we’re missing.
- Slot-filler values for attacks
  1. Attacker
  2. Target
  3. Weapon

4. Damage amount

To address this problem in the future, we have begun a collaboration with the developer of Avrae, which is a Discord bot for playing D&D online. Avrae contains many of the state variables that are missing from our current annotations.

8 Related Work

Although there has been work done on text-based game playing (Haroush et al., 2018; Yao et al., 2020; Dambekodi et al., 2020), these games still provide points for taking correct actions and generally have a limited vocabulary to work with. Creating text games (Ammanabrolu et al., 2020a; Fan et al., 2020) is more challenging but closer to the type of world-creating job the DM has in D&D. There has also been work on persona/character generation (Louis and Sutton, 2018; Prabhumoye et al., 2019) in stories, sometimes within D&D itself (Louis and Sutton, 2018). Others (Urbanek et al., 2019; Ammanabrolu et al., 2020b) have realized that non-player characters (NPCs) are lacking in their abilities to speak and act in text games.

Thankfully, we do not have to start from scratch when working toward the vast challenge of creating a D&D player and can rely upon the findings of the automated story generation community. Neural language models have become increasingly more popular for story generation (Roemmele, 2018; Martin et al., 2018a; Mathewson et al., 2019; Hou et al., 2019). We have also started to see storytelling with transformers (See et al., 2019; Peng et al., 2021; Branch et al., 2021). Transformer-based storytelling systems have even been introduced to the general public thanks to the popularity of AI Dungeon (Walton, 2019). Although neural networks possess a lot of power in terms of what text they generate, they are still limited in their ability to produce longer spans of coherent text. Many (Fan et al., 2018; Yao et al., 2019; Ippolito et al., 2019; Tambwekar et al., 2019; Ammanabrolu et al., 2020b; Rashkin et al., 2020) have improved the coherence of neural storytellers by splitting the
generation into two steps: ideation of the story plot, followed by the realization of sentences. This controllable story generation is where we are currently seeing a lot of work within neural automated story generation since it enables the neural network(s) to focus on a single task at a time.

Due to the conversational nature of D&D, we decided to use a dialog-based system. Deep neural networks have been used for dialog agents for a while (Serban et al., 2016), with a shift toward using transformers in recent years (Zhang et al., 2018; Smith et al., 2020). Like in automated story generation and other neural text generation tasks, we are also seeing controllability being an important factor being integrated into systems. This includes using deep reinforcement learning techniques to guide the dialog toward a goal (Li et al., 2016; Saleh et al., 2020) or controlling for style (Zhang et al., 2018; Smith et al., 2020).

In this paper, we use LaMDA, a transformer-based open-domain dialogue system that building on the Meena model (Adiwardana et al., 2020). The original Meena model was an end-to-end model trained on public conversations found on social media. Controllable text generation with transformers has been seen before with CTRL (Keskar et al., 2019), a language model that is conditioned on a given "control code" in addition to the Bayesian history. This work takes a similar approach. We integrate contextual information such as character descriptions, actions, and in- and out-of-character classifications.

We have finetuned our LaMDA models on data crawled from D&D Beyond6. This data contains both in-character and out-of-character dialog and can be used in conjunction with Rameshkumar and Bailey (2020)’s dataset from Critical Role (a D&D podcast), Louis and Sutton (2018)’s dataset from roleplayerguild.com (a D&D forum), and/or Urbanek et al. (2019)’s crowdsourced LIGHT dataset. For the purposes of this work, we will only be working with our D&D Beyond dataset.

9 Discussion and Conclusions

We find that training on D&D data results in much higher quality outputs than a vanilla dialogue system (as expected), that controlling the model to generate in-character responses results in substantially more interesting output, and that conditioning on game state information qualitatively results in responses that are appropriate to the character class. Our preliminary experiments with using the large language models to perform game state tracking show low performance even after fine-tuning, suggesting that other models may be required for an AI to play D&D track the full state of the game.

10 Acknowledgments

We would like to thank Antony Pegg of Fandom for granting us permission to use D&D Beyond’s forum data for research purposes.

References


6https://www.dndbeyond.com/


A Lost Mine of Phandelver Adventure

Here is an excerpt from the adventure book that the Dungeon Master was using in our example game play. The adventure book provides boxed text, which is descriptive text to be read aloud to paraphrased. It also gives details about the combat that is about to ensure, and links to relevant game rules (like stealth checks, and statistics about the monsters that the characters will be in combat with).

The adventure begins as the player characters are escorting a wagon full of provisions and supplies from Neverwinter to Phandalin. The journey takes them south along the High Road to the Tribor Trail, which heads east (as shown on the overland map). When they’re a half-day’s march from Phandalin, they run into trouble with goblin raiders from the Cragmaw tribe.

Read the boxed text when you’re ready to start. If you create a different adventure hook, skip to the second paragraph and adjust the details as necessary, ignoring the information about driving the wagon.

In the city of Neverwinter, a dwarf named Gundren Rockseeker asked you to bring a wagonload of provisions to the rough-and-tumble settlement of Phandalin, a couple of days’ travel southeast of the city. Gundren was clearly excited and more than a little secretive about his reasons for the trip, saying only that he and his brothers had found “something big,” and that he’d pay you ten...
gold pieces each for escorting his supplies safely to Barthen’s Provisions, a trading post in Phandalin. He then set out ahead of you on horse, along with a warrior escort named Sildar Hallwinter, claiming he needed to arrive early to “take care of business.”

You’ve spent the last few days following the High Road south from Neverwinter, and you’ve just recently veered east along the Triboar Trail. You’ve encountered no trouble so far, but this territory can be dangerous. Bandits and outlaws have been known to lurk along the trail.

You’ve been on the Triboar Trail for about half a day. As you come around a bend, you spot two dead horses sprawled about fifty feet ahead of you, blocking the path. Each has several black-feathered arrows sticking out of it. The woods press close to the trail here, with a steep embankment and dense thickets on either side.

Four goblins are hiding in the woods, two on each side of the road. They wait until someone approaches the bodies and then attack. This will likely be the first of many combat encounters in the adventure. Here are the steps you should follow to run it effectively:

- Review the goblin stat block in appendix B. Since the goblins are hiding, you’ll need to know their Stealth skill modifier: +6.
- Check to see who, if anyone, is surprised. The party can’t surprise the goblins, but the goblins might surprise some or all of the characters. Make a Dexterity (Stealth) check for the goblins: roll one d20 for all of them, add their Stealth skill modifier (+6) to the roll, and compare the total to the characters’ passive Wisdom (Perception) scores. A character whose score is lower than the goblins’ check total is surprised and therefore can’t do anything on his or her first turn in the combat (see “Surprise” in the Basic Rules).
- Use the initiative rules in the Basic Rules to determine who acts first, second, third, and so on. Keep track of everyone’s initiative count on a piece of paper.
- When the time comes for the goblins to act, two of them rush forward and make melee attacks while two goblins stand 30 feet away from the party and make ranged attacks. The goblins’ stat block contains the information you need to resolve these attacks. For more information on what the goblins can do on their turn, see chapter 9, “Combat,” in the Basic Rules. When three goblins are defeated, the last goblin attempts to flee, heading for the goblin trail.

B Annotation Guidelines

B.1 Annotation task

In this task, you will see part of a conversation between a few people playing D&D. The players and their characters are listed at the beginning of the conversation. The conversations that are shown as context are real conversations from players. Your job is to read the context and then rate different responses for a player/character given conversational context. Please note that the context you are given represents only a part of the players’ past conversations/interactions with one another during the game.

For each response, you would be asked the following questions.

- Does the response make sense?
  - Use your common sense here. Is the response completely reasonable in terms of the rules of D&D?
  - The response “makes sense” if it is cohesive as a standalone statement, consistent with the rules of the game, and the elements/entities mentioned are plausible, given the prior context.
  - If anything seems off—not fluent, confusing, illogical, out of context, or wrong according to the rules of D&D —then rate it as Does not make sense. If in doubt, choose Does not make sense.
- Is the response specific?
  - You may be asked to assess whether the response is specific to a given context. In other words, do you think that the response represents a good thing for the character to do now?
- The response is "specific" if it flows logically from the narrative established by the prior context.
  * Note: It is possible for a response to "make sense" (due to being cohesive, consistent and plausible in and of itself), but be marked "not specific" when it is not a logical next step in the overall game progression.
  * Note: "Specific" for the purposes of this task does not have to do with how detailed the response is per se; a response can be fairly general in its language, but still qualify as "specific" when it is a logical next step in the overall game progression.

- How interesting is the response?
  - You may be asked to score the response for its interestingness on a scale of 10. Choose a high score for “Interesting” if the response would likely catch someone’s attention or arouse curiosity in the game; or it is insightful, creative, or witty with respect to the game. If the response is monotonous and predictable, or if you’re unsure, then it is Less Interesting.

B.2 Survey of Raters

We recruited raters who had a background in role playing games and an understanding of the fantasy genre. We surveyed our raters, asking them the following questions:

1. Have you ever played Dungeons and Dragons or another role playing game before?

2. If so,
   - roughly how many times have you played
   - were you a player or a game master or both

3. If not,
   - what kind of exposure do you have to Dungeons and Dragons? (For example, have you seen it referred to in TV or movies)

4. Are you a fan of the fantasy genre (like Lord of the Rings)?

Our 6 raters responded to the survey as follows: 5 out of the 6 have played D&D or another role playing game before. All 5 of those who have played D&D /other role playing games before have played more than 6 times. Of the 5 who have played D&D /other role playing games before, 3 played as both Game Master and Player. For the one who had not played D&D /other role playing games, they indicated they had not had much exposure to D&D through TV or other channels. All 6 answered that they were fans of the fantasy genre.