Dungeons and Dragons as a Dialog Challenge for Artificial Intelligence

Chris Callison-Burch∗
University of Pennsylvania
ccb@upenn.edu

Gaurav Singh Tomar∗
Google Research
gtomar@google.com

Lara J. Martin
University of Pennsylvania

Daphne Ippolito
University of Pennsylvania

Suma Bailis
Google Research

David Reitter
Google Research

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 (LM) 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 lead to advances in the field. In many cases, AI game-playing systems have gone on to rival human champions of the game. 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 of 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 natural language processing (NLP) and generation (NLG). In particular, they saw how cooperative story generation between humans already exists in these games and can be used for automated generation. Martin et al. (2018b) 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 in the game.

D&D involves multiple players who roleplay characters in a fantasy setting, guided by a Dungeon Master who sets obstacles and adventures and plays as monsters. In roleplaying games like Dungeons and Dragons, the gameplay happens through language rather than moves on a game board, making it an interesting domain for NLP research. To have an AI successfully play D&D, it would require 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)

Appendix A gives an example of D&D gameplay and the AI challenges presented by it.

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? We argue that now is the perfect time for this challenge, since large scale neural language models like GPT have shown impressive generation results (Brown et al., 2020), and since
incorporating neural LMs into a game setting both exercises their strengths and exposes their weaknesses.

In this paper, we introduce a new dataset of “actual play” game transcripts. Each turn is labeled with game state variables like character information 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 in a multi-player game. We propose our dataset as a challenge for dialogue systems for the following reasons:

- It is naturally occurring conversational dialog that covers a spectrum of task oriented and non-task oriented (e.g. chit chat) dialog.
- It is strongly history dependent – a substantive criticism of recent dialog datasets is their history independence (Mosig et al., 2020).
- 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.

Unlike existing dialog datasets, our data reflects the challenging nature of the D&D game as a multi-party dialogue with creative roleplaying and underlying game states.

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 Section 3) 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. 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 producing 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).

3 Dataset

For this paper, we have created a novel dataset for our dialogue-oriented test of AI’s ability to play Dungeons & Dragons. We scraped Play-By-Post data from a web forum¹ where people play-by-turn posting on the forum to describe their move.

Figure 1 shows an example of part of the gameplay from the play-by-post forums from D&D Beyond. 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.

Table 1 summarizes the amount of play-by-post data that we collected from the D&D Beyond website (with permission from the company).

3.1 Heuristic annotation of game states

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

Table 1: Statistics for our play-by-post corpus

<table>
<thead>
<tr>
<th>Play-By-Post Corpus</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of campaigns</td>
</tr>
<tr>
<td>Average players per campaign</td>
</tr>
<tr>
<td>Average turns per campaign</td>
</tr>
<tr>
<td>Average words per campaign</td>
</tr>
<tr>
<td>Total turns</td>
</tr>
<tr>
<td>Total words</td>
</tr>
<tr>
<td>Average dice rolls per campaign</td>
</tr>
<tr>
<td>Total dice rolls</td>
</tr>
</tbody>
</table>

¹https://www.dndbeyond.com/forums/d-d-beyond-general/play-by-post
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 character 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\(^2\) 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 fantasy races\(^3\) are mentioned. Assign it to character. If not, guess based on the most frequently mentioned race.
- **Pronouns:** Count pronouns mentioned by a player. Assign their character’s pronouns to be the most frequent pronouns used by the player.
- **Inventory:** Use regex to match items occurring after character’s personal pronouns (e.g. her sword).

\(^2\)https://www.dndbeyond.com/classes
\(^3\)https://www.dndbeyond.com/races

• 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 (non-player characters) and monsters.

**Combat**

- We detect the start of combat when there is a roll for initiative, or when there are attack rolls before initiative (from 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.

**Actions**

- 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**,
<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</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</td>
<td>Different physical characteristics (e.g. halflings are small, dragonborn have scales).</td>
</tr>
<tr>
<td>Character Pronouns</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>

Table 2: 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.

*cure, heal, healing, points.*

*Our heuristics resulted in features for around 60% of all conversational turns. We train a convolutional neural network classifier using these conversational turns to predict all of the above control features for each conversational turn in training data. Appendix C estimates the accuracy of the model’s prediction on these state features.*

### 3.2 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 Playground\(^4\), where play happens on two discussion boards – one in-character and one out-of-characters. For example, here is an IC post:

> Kuros pulls the feathered shaft of the arrow back to his cheek winning easily against the resistance of the bowstring. He pulls a lungful of air to keep himself steady, takes aim at the Bandit with the deer, and lets fly.

And here is its corresponding OOC post:

> Surprise round so only 1 standard or move action. Shoot the bow:

\(^4\)https://forums.giantitp.com/forumdisplay.php?3-Play-by-Post-Games

\((1d20+6)[20]\) vs Flat Footed AC at Bandit 1. Damage: \((1d8+2)[10]\)

We train a classifier to predict IC versus OOC text, and then apply it to each paragraph in our D&D Beyond forum data.

### 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 Rudr, 2018), LaMDA can be finetuned to different tasks. The two tasks that we finetune 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 finetuning or additional post-processing. Table 2 describes the control features we have proposed and describes how they could steer generation. Note that we use the terms ‘control features’ and ‘state variables’ interchangeably when referring to our next utterance prediction models.

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. For all models, we used the 7 most recent conversational turns as input, and predict turn 8.

4.2 D&D FineTuning Data

Here is an example of the data used in our versions of LaMDA that are finetuned to on our D&D data.

**TURN 1:**

<table>
<thead>
<tr>
<th>Text</th>
<th>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?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Player ID</td>
<td>0</td>
</tr>
<tr>
<td>Character</td>
<td>Dungeon Master</td>
</tr>
<tr>
<td>Race</td>
<td>N/A</td>
</tr>
<tr>
<td>Class</td>
<td>Dungeon Master</td>
</tr>
<tr>
<td>Pronouns</td>
<td>N/A</td>
</tr>
<tr>
<td>Inventory</td>
<td>N/A</td>
</tr>
<tr>
<td>In combat?</td>
<td>Yes</td>
</tr>
<tr>
<td>In character?</td>
<td>Yes</td>
</tr>
<tr>
<td>Action</td>
<td>Attack</td>
</tr>
</tbody>
</table>

**TURN 2:**

<table>
<thead>
<tr>
<th>Text</th>
<th>I grab my axe and bring it down on the wounded goblin.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Player ID</td>
<td>1</td>
</tr>
<tr>
<td>Character</td>
<td>Magnus</td>
</tr>
<tr>
<td>Race</td>
<td>Human</td>
</tr>
<tr>
<td>Class</td>
<td>Fighter</td>
</tr>
<tr>
<td>Pronouns</td>
<td>he/him</td>
</tr>
<tr>
<td>Inventory</td>
<td>Axe</td>
</tr>
<tr>
<td>In combat?</td>
<td>Yes</td>
</tr>
<tr>
<td>In character?</td>
<td>Yes</td>
</tr>
<tr>
<td>Action</td>
<td>Attack</td>
</tr>
</tbody>
</table>

Figure 2: Negative Log Perplexity of our models after pretraining on generic dialogue data for 600K steps, and then finetuning to our data for a further 60k steps. Colors correspond to the models in Table 3.

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

**LLM-DND-ALL-CTRL:** LLM-Dialog that has been finetuned on Play-by-post D&D gameplay dataset using control features (state variables) for **all** dialog turns including the state variables for the **current turn** the utterance is being predicted for.

**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, not including the current turn.

**LLM-DND-CURRENT-CTRL:** LLM-Dialog that has been finetuned on Play-by-post D&D gameplay dataset using control features (state variables) for **only the current turn** the utterance is being predicted for.

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 2 plots the Negative log perplexity on our development set, and Table 3 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-CURRENT-CTRL and LLM-DND-ALL-CTRL, which used our control features.
Table 3: Perplexity and token accuracy of our models after fine-tuning to our data.

<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-CURR-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>
</tbody>
</table>

Table 4: Average sense, specific and interestingness scores for our systems, and for human-written gold responses

<table>
<thead>
<tr>
<th>Model</th>
<th>Sense</th>
<th>Specific</th>
<th>Interest</th>
</tr>
</thead>
<tbody>
<tr>
<td>LLM-Dialog</td>
<td>0.81</td>
<td>0.85</td>
<td>3.57</td>
</tr>
<tr>
<td>LLM-DND</td>
<td>0.9</td>
<td>0.9</td>
<td>3.91</td>
</tr>
<tr>
<td>LLM-DND-PREV-CTRL</td>
<td>0.86</td>
<td>0.88</td>
<td>3.96</td>
</tr>
<tr>
<td>LLM-DND-CURR-CTRL</td>
<td>0.88</td>
<td>0.9</td>
<td>3.96</td>
</tr>
<tr>
<td>LLM-DND-ALL-CTRL</td>
<td>0.87</td>
<td>0.88</td>
<td>3.92</td>
</tr>
<tr>
<td>GOLD</td>
<td>0.92</td>
<td>0.92</td>
<td>4.17</td>
</tr>
</tbody>
</table>

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

The full annotator instructions and the annotation interface are given in Appendix D.

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 the DM in a game before.

5.2 Inter-Rater Agreement

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

5.3 Analysis

Model Comparison. Table 4 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. On average, the adapted systems make sense 6.75% more often than the baseline, are specific 4% more often, and are 0.37 points more interesting. However, 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. Why then use the control features at all?

Control Features. One reason is that the control features are useful to allow the model to role-play. For example, we can ask the model to play an elf wizard via the control features. Table 5 gives a qualitative example of this. The Table shows outputs from two LLMs without control features, plus 3 outputs from our LLM-DND-CURRENT-CTRL model roleplaying different characters. As input, we gave the models the final seven turns in the game play example from Appendix A – 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 control feature 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 a battle cry by invoking the name of a Dwarf god.

In-Character Turns Are More Interesting. Among our most impactful control features was the one that allowed systems to generate in-character (IC) versus out-of-character (OOC) turns. Table 6 shows that control models’ scores substantially increased on IC turns compared to when their output was generated OOC. The pronounced increase in interestingness makes sense because IC turns are ones where the players describe their characters in...
The fictional world often with evocative language, whereas OOC turns usually discuss rules or mechanics. Our control features allowed the system to intentionally generate IC responses, resulting in substantially improved interestingness scores for those in-character turns.

6 Game State Tracking Model

We conducted an experiment to evaluate whether a LLM could be finetuned to perform game state tracking for D&D using our heuristically annotated game state features. We trained a new model LLM-DND-GST (Game State Tracking). It is a LLM-Dialog that has been finetuned on our Play-by-post D&D gameplay dataset. As input, it takes all previous dialog turns and their state variables, plus the text of the current turn, and then it outputs the corresponding state variables for the current turn.

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 7. 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 LLM-DND-GST is 58%. This suggests that accurately tracking the full game state may require additional machinery beyond a finetuned LLM.

7 Related Work

Previous work has examined AI to play text adventure games (Haroush et al., 2018; Yao et al., 2020; Dambekodi et al., 2020). These games are simpler than D&D because they have a limited vocabulary and more straightforward game states. Creating text adventure games (Ammanabrolu et al., 2020a; Fan et al., 2020) is more challenging than playing them, and is similar to the world-building job of the DM in D&D. There has also been work on persona/character generation in stories (Prabhumoye et al., 2019), and within D&D itself (Louis and Sutton, 2018). Others (Urbanek et al., 2019; Ammanabrolu et al., 2020b) have realized that NPCs are lacking in their abilities to speak and act in text games.

Findings of the automated story generation community are relevant for D&D AI systems. 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 the focus of a lot of current work in neural automated story generation.

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., 2019; Ghazarian et al., 2021). 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 builds 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 textual 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 Beyond. 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), Urbanek et al. (2019)’s crowdsourced LIGHT dataset, and/or Akoury et al. (2020)’s STORIUM dataset for human+AI collaborative story generation.

8 Discussion and Conclusions

We have demonstrated 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 finetuning, suggesting that other models may be required for an AI to play D&D track the full state of the game.

Although our models are unable to play D&D fully autonomously by acting as the Dungeon Master, they could act as an aid for novice DMs. Since our models can generated evocative, in-character text that is appropriate for the context and the game state, DMs could use it as inspiration as they narrate the adventure to the other players.

Here is some model output to inspire your next adventure:

You get a much closer look than the other two... the sarcophagi have the inscriptions of some sort of magic, probably to keep the dead inside, but you can not read them to save your life.

What will you do next? Download our dataset\(^5\) to start your new adventure!

9 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, it therefore 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. 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.

\(^5\)https://www.cis.upenn.edu/~ccb/dnd-data.html
D&D online. Avrae contains many state variables that are missing from our current annotations, such as HIT points and slot-filler values for attacks.

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.

Thank you to Rich Burlew and forum moderator true mane for granting us permission to crawl the Giant in the Playground forum and to build models using the forum posts.

We would like to thank the many Google Researchers who gave valuable input on this project, especially Dipanjan Das and Suma Bailis.

Chris Callison-Burch is grateful for having the opportunity to be a part-time visiting researcher at Google for two years. It was amazing to spend time among such incredibly smart people, and it was eye opening to see large LMs before they became widely available. Thanks!

References


A Example D&D Game Session

Instead of the game being a series of moves on a game board, RPGs D&D are language-based. Players create characters that have a class (wizard, fighter, thief, etc.) that denotes their abilities, and a fantasy race (elf, dwarf, human, etc.). 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 8 provides example dialogue from a game of D&D being played between 3 players – Travis (playing a human fighter named Magnus Burn-sides), Clint (playing Merle Highburgh, 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.

The game session is taken from the podcast The Adventure Zone. In this episode, the hosts are playing an adventure module called Lost Mine of Phadelver, an expert of which is given in Appendix B. In the first episode of the podcast, the hosts explain the rules of D&D.

B Lost Mine of Phadelver 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 verbatim or to paraphrase. 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 Tribor 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 Tribor Trail for about half a day. As you come around a bend, you spot two

---

6https://maximumfun.org/episodes/adventure-zone/ep-1-here-there-be-gerblins-chapter-one/
<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. 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.</td>
<td>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).</td>
</tr>
<tr>
<td>Griffin (DM)</td>
<td>Roll a perception check for me. Perception is a wisdom skill, so be sure to add your wisdom modifier.</td>
<td>Clint has rolled his dice. The number is low so his character fails the check. AI challenges: Multi-party dialogue.</td>
</tr>
<tr>
<td>Clint (out of character)</td>
<td>I got an eight.</td>
<td>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.</td>
</tr>
<tr>
<td>Justin (out of character)</td>
<td>I got a six.</td>
<td>Travis is describing what he is doing using “in character” language. AI challenges: Persona-based chat.</td>
</tr>
<tr>
<td>Travis (out of character)</td>
<td>I rolled a natural twenty plus my wisdom modifier is 23.</td>
<td>The DM describes what happens as a result of the success. AI challenges: Reasoning about consequences of success or failure, descriptive text generation.</td>
</tr>
<tr>
<td>Griffin (DM)</td>
<td>With his eagle eyes, Magnus spots two dead horses lying in the middle of the road about 200 feet ahead of you.</td>
<td>Travis is describing what he is doing using “in character” language. AI challenges: Persona-based chat.</td>
</tr>
<tr>
<td>Travis (in character as Magnus)</td>
<td>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.</td>
<td>The DM describes the start of a battle with several monsters. AI challenges: State tracking (in combat v. out of combat).</td>
</tr>
<tr>
<td>Griffin (DM)</td>
<td>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?</td>
<td>AI challenges: Question answering, state tracking (how many monsters).</td>
</tr>
<tr>
<td>Travis (out of character)</td>
<td>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.</td>
<td>AI challenges: Question answering, Descriptive text generation from game state.</td>
</tr>
<tr>
<td>Griffin (DM)</td>
<td>I will cast sacred flame at the nearest one. If it fails a dexterity saving throw, it takes 6 points of damage.</td>
<td>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>

Table 8: Example dialogue from a game of D&D with explanations of what is happening and comments on potential challenges for AI
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. Since the goblins are hiding, you’ll need 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.

C Estimated accuracy of predicted state variables

In addition to the heuristics that we used to recover state variables for each turn in the game (described in Section 3.1), we used a CNN to fill in state values when our heuristics did not fire. Table 9 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. The CNN classifier only uses current post text as input (no additional context).

D Annotation Guidelines and Annotation Interface

D.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.
- 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 Inter-
Table 9: 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>Pronoun</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>

D.2 Annotation Interface

A mock up of the annotation user interface is given in Figure 3.

D.3 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.
Player 10: Del (Barbarian)
Player 0: Dungeon Master
Player 12: Aramis (Bard)
Player 8: Xavier (Bard)

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.

Player 0 - Dungeon Master: Del - make an investigation check.

Player 10 - Del (Barbarian): 3

Player 0 - Dungeon Master: The barbarian begins tossing things around, overturning tables and tearing crates and sacks open, though his still simmering rage at being locked up prevents him from undertaking a thorough search. He turns up a shortbow, a light crossbow, a few javelins, 2 quivers of arrows, 2 cases of crossbow bolts, a crate full of mismatched leather armour and a few sacks full of preserved fruit and vegetables. There is no sign of the equipment that was taken from him last night.

Player 12 - Aramis (Bard): "anyone seen any keys? or my stuff?" aramis calls out to everyone

Player 8 - Xavier (Bard): "quickly picking up the dead orcs cleaver so a better weapon Xavier turns to the rest of the party "fighting is not over, we are not sure what is to be found up there, raise the store room, we must be ready for more combat" "i turn to del" "would you stand with me to defend the rest while they re-arm?" "i offer the crowbar to del" "then turn back to the party" "and if you find a shield and sword in there, i require a stronger defence and fist"

Response 1
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."

Response 2
del takes the crowbar and tells xavier to find hes ***en axe and ill keep good care of him

Response 3
Anyone see my axe, and leather?

Response 4
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."

---

**Figure 3:** The user interface that our raters used to evaluate the quality of our model’s next utterance prediction.