Collaborative Storytelling with Large-scale Neural Language Models

Eric Nichols*
Honda Research Institute Japan
e.nichols@jp.honda-ri.com

Leo Gao*
OrientExpress Technologies Inc.
leogao31@gmail.com

Randy Gomez
Honda Research Institute Japan
r.gomez@jp.honda-ri.com

ABSTRACT

Storytelling plays a central role in human socializing and entertainment. However, much of the research on automatic storytelling generation assumes that stories will be generated by an agent without any human interaction. In this paper, we introduce the task of *collaborative storytelling*, where an artificial intelligence agent and a person collaborate to create a unique story by taking turns adding to it. We present a collaborative storytelling system which works with a human storyteller to create a story by generating new utterances based on the story so far. We constructed the storytelling system by tuning a publicly-available large scale language model on a dataset of writing prompts and their accompanying fictional works. We identify generating sufficiently human-like utterances to be an important technical issue and propose a sample-and-rank approach to improve utterance quality. Quantitative evaluation shows that our approach outperforms a baseline, and we present qualitative evaluation of our system's capabilities.

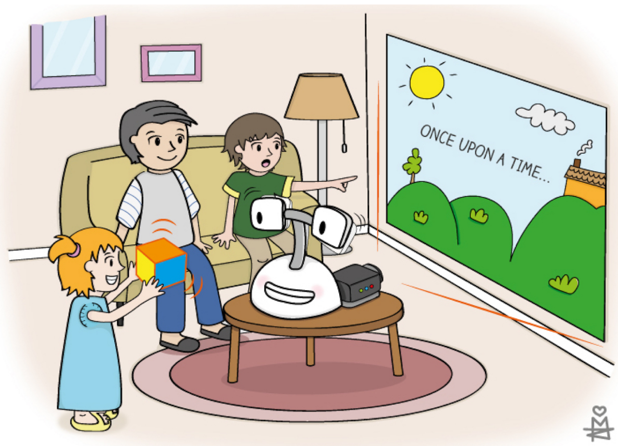

Figure 1: Collaborative storytelling with an AI agent.

CCS CONCEPTS

• **Human-centered computing** → **Collaborative interaction**; **Natural language interfaces**; **Collaborative and social computing devices**; • **Computer systems organization** → **Neural networks**.

KEYWORDS

storytelling, interactivity, language models, AI agents

ACM Reference Format:
Eric Nichols, Leo Gao, and Randy Gomez. 2020. Collaborative Storytelling with Large-scale Neural Language Models . In *Motion, Interaction and Games (MIG '20), October 16–18, 2020, Virtual Event, SC, USA.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3424636.3426903

1 INTRODUCTION

Storytelling is a central part of human socialization and entertainment. Many of the popular forms of storytelling throughout history –such as novels, plays, television, and movies– have passive audience experiences. However, gaming is an interesting medium because interactivity is a large part of the entertainment experience, and interactivity and storytelling can often be in conflict: too much player freedom means a storyline may never be explored, while on the other hand, too many restrictions on player freedom risks reducing gaming to a passive medium. Thus, interactivity in storytelling has been an important challenge for gaming, with much design effort put into striking a balance between entertaining gameplay and compelling storytelling.

As gaming technology advances, new opportunities for interactive storytelling present themselves. Better storage technology made telling longer, more intricate stories possible, and better graphical capabilities helped foster more immersive gaming experiences. Advances in artificial intelligence have lead to more challenging opponents, more realistic NPC behavior, and other benefits. Better procedural content generation algorithms help ensure unique gameplay experiences that stay fresh for longer. Finally, recent breakthroughs in language modeling present a new opportunity: language, and thus stories, can potentially be generated on demand.

In this paper, we introduce a novel game of *collaborative storytelling*, where a human player and an artificial intelligence agent construct a story together. The game starts with the AI agent reciting one of a curated set of *story starters* –opening sentences meant to kick-start participants' storytelling creativity– and the human player responds by adding a line, which we refer to from here on out as a *story continuation*, to the story. The AI agent and human player then take turns adding continuations to the story until the human player concludes the story. The game is designed to have a few restrictions as possible and contrasts with traditional storytelling settings where the narrative is fixed in advance.

*Equal contribution.

MIG '20, October 16–18, 2020, Virtual Event, SC, USA

ACM ISBN 978-1-4503-8171-0/20/10.
https://doi.org/10.1145/3424636.3426903

Collaborative storytelling builds on a rich tradition of collaboration in storytelling that includes Dungeons and Dragons, improvisational comedy, and theater. It could be a useful tool for encouraging creativity and overcoming writer's block, as well as being an entertaining game in its own right.

Our end goal is to make it possible for intelligent agents, such as robot companions and avatars [Gomez et al. 2020; Park et al. 2019], to play the collaborative storytelling game, as shown in Figure 1.

Our primary contributions are as follows:

- We introduce a novel task of *collaborative storytelling*, where humans and AI agents work together to create a story.
- We present a collaborative storytelling system that is constructed by tuning a large-scale neural language model on a writing prompts story dataset.
- We develop a method for ranking language model output to obtain more human-like story continuations.
- We conduct quantitative and qualitative analysis of the storytelling capabilities of our system through collaborative storytelling with human participants.

2 RELATED RESEARCH

In this section, we summarize relevant research in story generation, interactive language generation, and language modeling.

2.1 Story Generation

In recent years, the task of automatic story generation has gained a lot of attention. [Fan et al. 2018] construct a corpus of stories and propose a hierarchical story generation model. [Yao et al. 2019] approach the task by first generating a plot outline and then filling in the language. [Gupta et al. 2019] generate story endings by incorporating keywords and context into a sequence-to-sequence model. [Luo et al. 2019] incorporate sentiment analysis into story ending generation. [See et al. 2019] conduct an in-depth analysis of the storytelling capabilities of large-scale neural language models. However, the primary assumption of these works is that story generation is conducted without any interaction from humans.

2.2 Interactive Language Generation

While research dedicated to interactive language generation games is still sparse, there are a few notable recent developments.

AI Dungeon[1] is a text adventure game that is generated by a GPT-2 language model [Radford et al. 2019] tuned on a collection of text adventure play-throughs. In the game, players assume the first person and interact with the world by inputting commands or actions. The language model is used to generate the world's reaction to the player's actions. Our *collaborative storytelling* task and approach are similar to AI Dungeon, but our task is not constrained to the genre of first-person adventures, and we rank model output.

[Cho and May 2020] build an improvisational theater chatbot by identifying and collecting instances of improvisational dialogue on the Web and using it to tune and evaluate public domain dialogue systems. Our collaborative storytelling task is similar to improv, but stories are linguistically different enough from improv that it would be impractical to apply their dataset to our task. In addition, our approach employs sampling and ranking to improve the likelihood that language model utterances are in the desired storytelling domain, while [Cho and May 2020] use the model's output as-is.

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

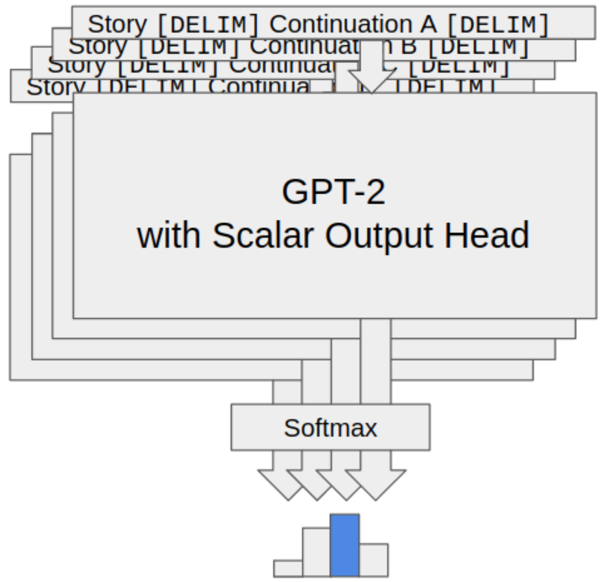

Figure 2: The ranking system architecture.

2.3 Language Models

In order for an AI agent to participate in collaborative storytelling, it must be able to generate story continuations. A language model (LM) is a mathematical model that assigns likelihoods to sequences of words where sequences that are more likely in a target language are given higher scores. Such a model can be used to generate text.

More precisely, a language model provides a probability distribution $P(x)$ over all sequences of tokens x. Sampling from the LM distribution is equivalent to generating text, motivating the approximation of the value of $P(x)$ using a universal function approximator, like an artificial neural network. Specifically, autoregressive LMs predict the next token given all previous tokens; this is equivalent to factoring the probability $P(x)$ as as a product of conditional probabilities of each token x_i given previous tokens $x_{<i}$:

$$P(x) = \prod_{i=0}^{n} P(x_i | x_{<i})$$

The autoregressive formulation facilitates text generation as previous tokens are used to predict a distribution over potential next tokens, from which a token can be sampled and appended to the sequence. In other words, at each timestep an autoregressive LM predicts the next token given the sequence of previous tokens.

Early language models estimated token sequence likelihood based on token sequence counts taken from large collections of text together with various smoothing methods to handle novel token sequences [Ney et al. 1994]. Later, RNNs and other sequential neural networks models became popular due to their ability to apply distributed word representations [Bengio et al. 2003; Mikolov et al.

Collaborative Storytelling Story 1	
Prompt:	*A man sits alone at a table, listening to the low and pleasant hum of a bassist pluck something smooth and jazzy nearby.*
Human:	*He lets out a big sigh and dreamily looks at the bassist, sipping on his whisky.*
System:	*The music is not too loud, just enough to drown out his worries, but still enough to cause him to feel like he has heard it before.*
Human:	*The man tries his best to just immerse himself into the music until it ended.*
System:	*But then the phone rang, making the man jump out of his chair.*
Human:	*"Darn it", says the man as he frantically searches his pockets for the phone.*
System:	*As the man was getting ready to answer, a beautiful lady, with long flowing blonde hair, greeted him.*
Human:	*The man drowned out the constant ringing on the phone and silenced it, saying hello back to the lady.*
System:	*She turns to the guy and smiles while staring him straight in the eyes.*
Human:	*The man replied "My name is Jacob... how about you"?*
System:	*After a few moments of silence, she smiled and explained that the man had already lived with her for years.*
Human:	*"Huh... what do you mean I already live with you?" Jacob said in utter confusion.*
System:	*She explained that she was the same age as he, and is the love of his life.*
Human:	*"No, you got to be kidding me I'm single and I come to this bar every day", Jacob said with a quiver in his voice.*
System:	*"You know what's funny?"*
Human:	*"Please explain, and are you just a stranger playing a joke on me?"*
System:	*She handed him a picture of her and a girl.*
Human:	*Her thumb was covering another person, and she slowly uncovered it as Jacob looked at the picture.*
System:	*"Look at me, " said the girl.*
Human:	*"This... picture shows me, you, and a child... but why?" Jacob started to tear up.*
System:	*"Just a moment, it will all make sense soon enough."*

Collaborative Storytelling Story 2	
Prompt:	*The garage door lurched open and my face turned from joy to agonizing surprise.*
Human:	*A small squirrel leaped from behind my car, likely as shocked to see me as I was him.*
System:	*"I'm so sorry, I didn't mean to scare you."*
Human:	*He tilted his head to the side, almost indicating he understood my words.*
System:	*"You don't scare me."*
Human:	*Completely floored that this squirrel can speak English, I fumbled myself into asking "Are you real? This can't be real!"*
System:	*"Yes, this is real, and I have been waiting for you for quite some time."*
Human:	*I asked him "What have you been waiting for? Some peanuts from the kitchen?!"*
System:	*"No, I can answer all your questions with just one sentence!"*
Human:	*He leaned against the tire of my car with a sly look on his face, "I'm madly in love with Penny, your cat, and I'd like to take her to dinner tonight!"*
System:	*I sat there confused for a good thirty seconds before I snapped out of it.*
Human:	*"Have you been spying on my house cat!?"*
System:	*The squirrel cocked his head to the side and shrugged, as if he had no response to what I had just said.*
Human:	*"Well I'll have you know that I am not okay with you romantically pursuing my poor Penny!"*
System:	*This time, the squirrel cracked his knuckles and the look in his eyes made me question everything I was hearing.*
Human:	*"Look Mr, all I want to do here is show her a good time and maybe be a part of the family!"*
System:	*I was dumbfounded.*
Human:	*I felt a slight lump in my throat realizing that maybe this squirrel was lonely and just wanted a good home and asked "Well, I guess if you insist, where did you want to take her to dinner?"*
System:	*"To my favorite fishing spot!"*
Human:	*The squirrel chuckled, pulled out a teeny tiny fishing rod, slapped on a little fishing hat and started making his way inside to ask my soon to be very confused cat out on a date.*
System:	*Thank you for reading!*

Table 1: Example stories generated by the `tuned` system with a human through the collaborative storytelling annotation task.

2011; Sutskever et al. 2011], but RNNs have issues with vanishing gradients and modelling long-term dependencies found in text.

The recent transformer architecture [Vaswani et al. 2017] uses attention layers to model long-term dependencies by greatly increasing the model's visible context. Transformers have been shown to perform well in a variety of tasks, including machine translation [Vaswani et al. 2017] and a variety of language understanding [Radford et al. 2019] and language generation tasks [Zhang et al. 2019]. A notable transformer model is BERT [Devlin et al. 2018]. However, as it is a bidirectional model, BERT and its variants are rarely used for text generation, due to the necessity for computationally-expensive Gibbs sampling [Wang and Cho 2019].

The model we use as a basis for our system, GPT-2 [Radford et al. 2019], is a large-scale neural network using the transformer architecture, and it has been applied successfully in a variety of language generation tasks ranging from news article generation to dialog. GPT-2 is a general purpose auto-regressive LM trained on a large corpus of internet text and its pretraining has been shown to be effective for transfer learning to novel domains.

3 APPROACH

Our approach to collaborative storytelling is simple: a *Generator model* that is a large-scale neural language model tuned on storytelling data to generate story continuation candidates is combined with a *Ranking model* that is trained on human storyteller preferences to score them and select the highest quality continuation.

3.1 Generation

The Generator is a unidirectional autoregressive language model which is sampled from multiple times to generate candidate story continuations. We used the publicly-available pretrained 774M parameter GPT-2-large model[2] tuned on our WritingPrompts dataset.

One issue with using an LM for generation is the output may be ill-formed or lacking in logical coherence. The main solutions for this issue are the use of larger models, the use of different sampling methods, and the use of various methods of traversing the search space of possible sentences. However, larger models are at greater risk of over-fitting and result in large increases in memory usage for modest gains in quality, which makes them impractical to use. As such, we focused on sampling and searching through ranking.

3.2 Sampling

The most popular approaches for sampling from autoregressive models have predominantly focused on techniques for truncating the low-quality tail of the model distribution, like top-k and nucleus sampling [Holtzman et al. 2019]. Sampling is used in most GPT-2 based text generation systems, superseding greedy or untruncated sampling. In all experiments, we use nucleus sampling with $p = 0.9$.

3.3 Ranking

The Ranker model scores each story continuation candidate and selects the highest scoring one. It is a standard GPT-2-large model with a final classification head consisting of a linear layer outputting a single scalar for each token. The input format to the model is: `(context)<|endoftext|>(choice)<|endoftext|>`.

The `<|endoftext|>` token is used because it is guaranteed not to occur elsewhere in the input. As GPT-2 is unidirectional, the embedding of the final token integrates information from the entire input context window; this is similar to the use of the `[CLS]` token in BERT. Thus we execute the Ranker model once for each choice, keep only the outputs from the last token of the final layer for each choice as the logit score of each choice, and compute a softmax over them. The Ranking model architecture is shown in Figure 2.

We chose a neural network-based Ranker model to select the best story completion from the Generator output because it offers us control over the trade-off between text generation quality and computational demand, while avoiding the significantly increased memory footprint and inflexibility in computational cost of using a larger language model. The amount of computational resources used is easily adjustable by changing the number of rollouts considered by the Ranker. This serves as a middle ground between the intractable extreme of searching the entire space of all $\text{vocab}^{\text{length}}$ possible sentences, and the computation-efficient but suboptimal solution of sampling without any branching or backtracking.

One popular alternative search solution making a similar trade-off is beam search, which keeps a dynamic list of generation candidates. Beam search has been applied in many language generation tasks, including machine translation [Tillmann and Ney 2003]. However, sampling from an LM using beam search can lead to degenerate text (which is typically repetitive and uninteresting), in an open-ended task such as storytelling. [Holtzman et al. 2019] These issues are avoided using a neural network-based Ranker model because it has richer text representations, it scores full text utterances rather than incomplete text fragments, and it can incorporate additional information about the storytelling domain from its training data.

3.4 Datasets

In this section we describe our datasets: (i) a collaborative storytelling dataset constructed by crowdsourcing workers interacting with our collaborative storytelling system that are used to train the Ranker model and for evaluation, and (ii) a writing prompts dataset comprised of short stories written in response to writing prompts posted to a Web forum that are used to train the Generator model.

3.4.1 Collaborative Storytelling Dataset. We collected collaborative stories using Mechanical Turk, each consisting of 20 interactions in response to a provided story starter (which is sampled from the initial sentences of stories in the WritingPrompts dataset described in Section 3.4.2). The interactions in the story alternate between *choice* type interactions, in which a human participant chooses from 10 story continuations that are generated by out collaborative storytelling system, and *freeform* type interactions, in which the human participant is able to provide a complete sentence response. The Web interface for this task in shown in Figure 3.

In order to ensure data quality, one of the continuations in the *choice* type interaction is a *distractor* which is made by concatenating randomly sampled words. The distractors are also filtered through Mechanical Turk beforehand by asking workers whether

[2] https://github.com/openai/gpt-2

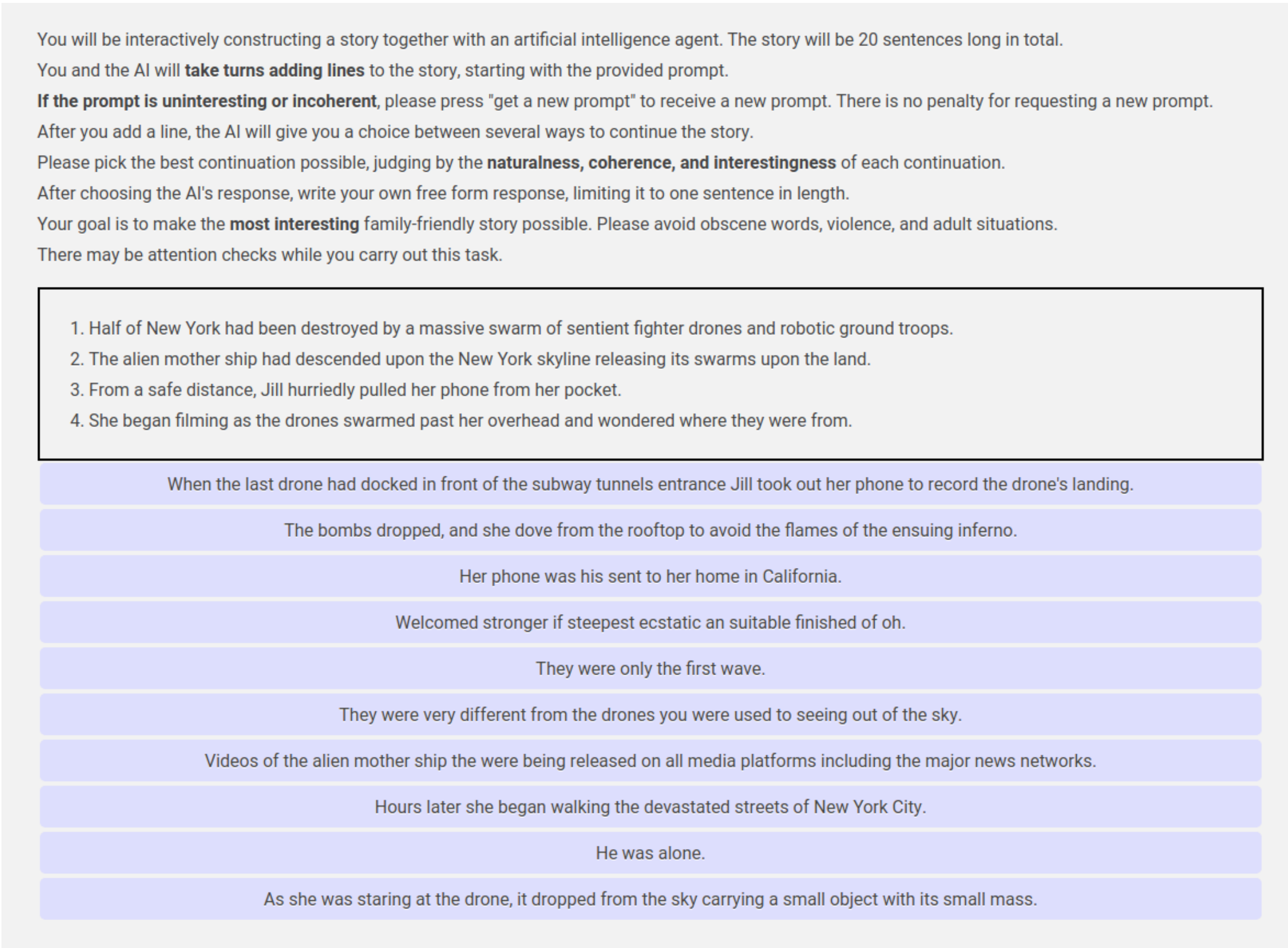

Figure 3: Web interface for collaborative storytelling annotation task. Participants select from amongst ten possible story continuations generated by the system before adding their own line to the story.

System	Dataset	Accuracy	
`tuned+ranked`	validation	22.9%	(229 / 1000)
`tuned+ranked`	test	23.3%	(233 / 1000)
random baseline	-	10.0%	

Table 2: Accuracy of the `tuned+ranked` model at predicting the story continuation that was selected by the Mechanical Turker who constructed the story. Note that a random baseline would pick the correct continuation 1 out of 10 times.

System	Acceptability	
`untuned`	33.9%	(305 / 900)
`tuned`	39.8%	(358 / 900)
`tuned+ranker`	**62%**	(62 / 100)

Table 3: Mean acceptability of story continuations in the test set. To evaluate `untuned` and `tuned`, acceptability is calculated over all 9 continuations from each system, while `tuned+ranked` uses the Ranker to consider only the best one.

the sentences are coherent or not, and only the ones labelled incoherent by workers are used. As a quality check, if a worker selects a distractor during a *choice* type interaction, the story is discarded.

We collected a total of 2,200 stories, which we randomly partitioned into a training split of 2,000 stories, and validation and test splits of 100 stories each. Some example stories generated by human participants together with our system are shown in Table 1.

3.4.2 Writing Prompts Dataset. We constructed a dataset of stories from the `r/WritingPrompts` subreddit[3], consisting of all posts with score greater than 3 made before 2019-11-24, amounting to 140k stories in total. Some heuristics were used to clean the stories[4]. This data was used to train the Generator model.

[3]https://www.reddit.com/r/WritingPrompts/
[4]We removed smart quotes, links and user/subreddit mentions, and all HTML entities and markdown formatting.

To train the Ranker model, stories with less than 100 characters or 35 sentences were also removed. This data is then used to generate synthetic collaborative storytelling data. The first sentence of the story is used as the story starter, and the next 20 sentences are all used as the preferred story continuations of *choice* type interactions, where the other 9 incorrect choices are sampled from the 25th and subsequent sentences of the story.

We chose to collect our own WritingPrompts dataset instead of using the FAIR WritingPrompts dataset [Fan et al. 2018], because it gave us the flexibility to filter stories by custom score thresholds, as well as to perform the different preprocessing necessary for GPT-2. Our dataset also contains more than an additional year's worth of data compared to the FAIR dataset.

3.5 Story Continuation Sampling and Ranking

To generate story continuations from our system, sentences are generated from the Generator model and filtered using a set of cleanliness heuristics until the desired number of samples is achieved. Our heuristic rejected sentences with less than 60% alphabetic characters, unbalanced quotations, select profanity, or words like "chapter" that are not typically part of the story.

For systems using ranking, the Ranker model computes a score for each story continuation and selects the highest scoring one.

3.6 Training

The Generator model is trained with a maximum likelihood estimation loss function using Adafactor [Shazeer and Stern 2018] with a learning rate of 5e-5 on a weighted mixture of the WritingPrompts and BookCorpus [Zhu et al. 2015] datasets. The addition of BookCorpus helps reduce the risk of over-fitting on the comparatively smaller WritingPrompts dataset.

The Ranking model is trained using Adam [Kingma and Ba 2014] with a maximum learning rate of 1e-5. The entire model is trained; no layers are frozen. The checkpoint is resumed from a GPT-2 text generation model that was tuned on the BookCorpus and WritingPrompts datasets in the same way as the Generator model.

The Ranking model is trained on the WritingPrompts dataset and 8 copies of the training split of the Collaborative Storytelling dataset, shuffled at the story level. Each batch for the Ranking model consists of 20 sentences taken from a single story. To ensure that the model fits in memory, only the sentences that fit within 400 tokens are used, resulting in some batches with less than 20 sentences. The majority of stories do not have to be truncated.

4 EVALUATION

We evaluate our collaborative storytelling system through a combination of qualitative and quantitative metrics. To understand how well our system replicates human preferences, we measure story continuation ranking accuracy and story continuation acceptability. To gain insights into the characteristics that people feel our system has, we adapt the Acute-eval chatbot evaluation metric [Li et al. 2019] to collaborative storytelling evaluation.

The three systems we evaluate are (i) `untuned` (pretrained GPT-2) as a baseline, (ii) `tuned` (GPT-2 tuned on storytelling data), and (iii) `tuned+ranker` (GPT-2 tuned on storytelling data with a single story continuation selected by the Ranker model).

4.1 Story Continuation Prediction Accuracy

Story continuation prediction accuracy measures the accuracy of the Ranker model at predicting the continuation chosen by the Mechanical Turk worker that interacted with the model to produce the story. This metric is a proxy for how often the `tuned+ranked` picks the best continuation of the story, but its usefulness is diminished by variance in human annotators and the possibility of multiple equally good continuations. The results are summarized in Table 2. Nonetheless, we find that our Ranker model outperforms chance by a factor of over two, providing evidence that it is able to capture the preferences of human annotators to an extent.

4.2 Story Continuation Acceptability

As an additional measure of our systems' capacity to generate story continuations that match human preferences, we formulate the story continuation acceptability task. In this task, each story continuation generated by a system is classified as either *acceptable* or *unacceptable*, and we compare their mean acceptability precision.

We annotated the acceptability of candidate story continuations by asking Mechanical Turk workers to classify each continuation given the context of the story generated so far. To ensure annotation quality, we have 3 workers evaluate each *choice* interaction per story from both the validation and test sets and take the majority vote across the three labels as the final label[5]. These *choice* interactions consist of 9 story continuations generated by the system and 1 incoherent distractor. If a worker labels a distractor acceptable, their annotations are discarded. We use this method to evaluate how often each model produces outputs that are an acceptable continuation of the story, rather than the best continuation.

Since the `tuned` and `tuned+ranked` systems use the same language model samples, we use the test set to evaluate their performance, considering the mean acceptability of all of the sampled continuations from `tuned` and the acceptability of the single continuation selected by `tuned+ranked` for each *choice* interaction in the datasets. To evaluate the `untuned` system, we gather and evaluate 100 *choice* interactions by having Mechanical Turkers construct stories with the `untuned` system.

The results are summarized in Table 3. As we can see, the `tuned` system outperforms the `untuned` system, showing that tuning the language model on storytelling data is important in improving generation quality. We also find that `tuned+ranked` greatly outperforms the other two systems, providing supporting evidence that our Ranking model is effective at helping our language model produce story continuations that are likely to be preferred by humans.

4.3 Human Annotator Story Preferences

Conducting qualitative evaluation of collaborative storytelling is challenging because the highly interactive nature of the task means that the influence of human participants makes it difficult to isolate the performance of the system. Ideally we would like to conduct subjective evaluation of participants' collaborative storytelling experience with an intelligent agent, but this is left for future work.

Instead, since collaborative storytelling involves language exchange between entities with turn taking, we take inspiration from dialogue system evaluation methodology. Faced with the challenge

[5]The workers reached unanimous agreement 41.9% of the time on the test data.

Figure 4: Web interface for storytelling system preference evaluation.

Characteristic	Question
Engagingness	Who would you prefer to collaborate with for a long story?
Interestingness	If you had to say one of these storytellers is interesting and one is boring, who would you say is more interesting?
Humanness	Which storyteller sounds more human?
Story Preference	Which of these stories do you like better?

Table 4: Questions asked to human evaluators of collaborative storytelling systems. Characteristics and questions are based on the *PersonaChat* evaluation metric of [Li et al. 2019], with minor changes to wording to reflect the task's storytelling nature.

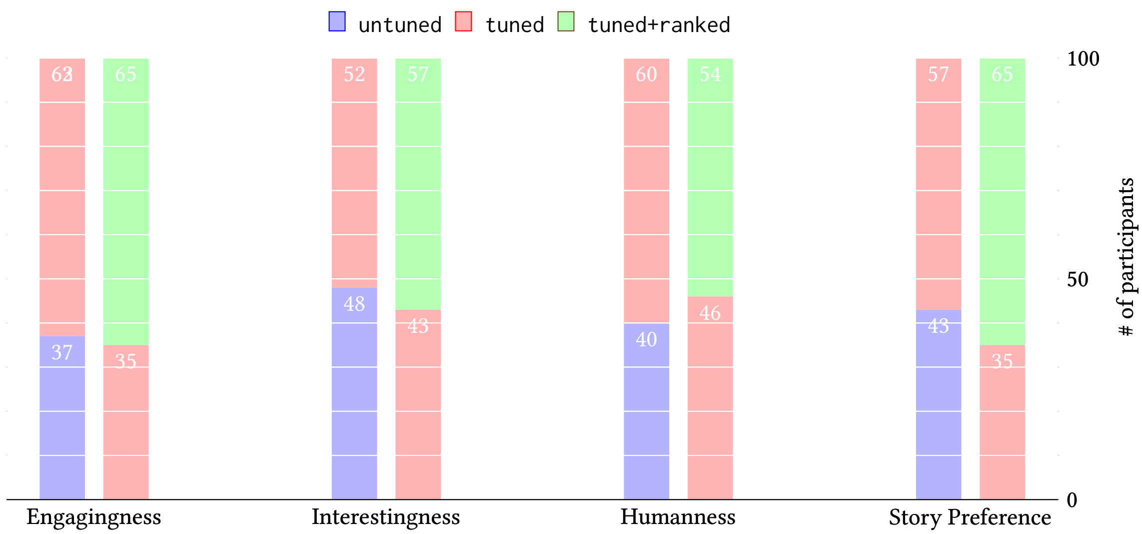

Figure 5: Human evaluation of collaborative storytelling systems. We compare the pairs (`untuned`, `tuned`) and (`tuned`, `tuned+ranking`). Each bar graph shows a comparison of two different systems generating stories through self chat. A larger portion of the bar indicates that system was preferred by evaluators.

of comparing multiple dialogue systems, [Li et al. 2019] developed a method of comparing conversation pairs that instructs evaluators to only pay attention to the contributions of a single specified speaker in the conversation. In addition, their evaluation method, known as ACUTE-EVAL, allowed them to evaluate the contributions of a given dialogue system in terms of characteristics, such as engagingness, interestingness, humanness, and knowledgeability. Finally, to evaluate different dialogue systems without requiring a human to chat with them, they apply the self-chat technique of [Ghandeharioun et al. 2019] and generate conversations for evaluation by having dialogue systems talk to themselves.

We create our own evaluation metric based on the characteristics targeted by the *PersonaChat* metric of ACUTE-EVAL[6]. For each target characteristic, we take the question that [Li et al. 2019] identified as most likely to differentiate between the evaluation of two systems and reword it to fit the collaborative storytelling setting. Finally, we add a question to measure overall story preference. The resulting evaluation metric is shown in Table 4.

We created a Mechanical Turk task to determine relative pairwise user preferences using our evaluation metric. To eliminate variance from human storytellers, we use the self-chat setting of [Li et al. 2019], where each model converses with itself. Some example stories are shown in Table 5. We compare the `untuned` and `tuned+ranked` models against the `tuned` model. For each pair of models, we collect 100 comparisons per question, and we instruct workers to provide short justifications for their decisions. The Web interface shown to workers is given in Figure 4.

The results of the evaluation are summarized in Figure 5. For each characteristic evaluated, the pairs of models are shown as stacked bar graphs, where a larger portion represents a stronger preference for that system. As can be seen, `tuned` is preferred over `untuned`, and `tuned+ranked` is preferred over `tuned` for all characteristics and overall story preferences, providing evidence that tuning the language model on storytelling data and ranking the generated story continuations make complementary contributions to our collaborative storytelling system's performance.

5 DISCUSSION

In this section, we discuss the advantages and limitations of our approach to collaborative storytelling.

5.1 Advantages

The advantages of our approach are that our storytelling system can produce well-formed story contributions that display creativity and

[6] We exclude the *Wizard of Wikipedia* metric because knowledgeability is not directly relevant to our collaborative storytelling setting.

	Self-chat tuned+ranked Story 1
A:	*The clock finally struck 10am and the massive archway gates were slowly pulled open with the sound of a grinding mechanism.*
B:	*I stood before the immense bronze doors, arms crossed, awaiting the fate of the two evil Dictators.*
A:	*"Ladies and gentlemen of the land" a soft yet authoritative voice boomed through the doors.*
B:	*"We are here for your arrival" a commanding and officious voice echoed through the halls.*
A:	*A slow breath tickled my lips as I was finally allowed inside.*
B:	*The archway was brimming with people, many in red, with metal cufflinks, and most in ornate black garb.*
A:	*I stepped forward into the main audience hall, pausing to take in my surroundings.*
B:	*The ceiling was made of rough jagged glass panels; out in the back stood eight shiny and elaborately painted metal ladders.*
A:	*The stately mahogany wooden podium was suspended from the ceiling, very tall and proud.*
B:	*The four different Dictators circled the podium with their various groups in the room.*
A:	*The oldest of the Dictators stood, hands together in front of him, on his magnificent golden podium.*
B:	*He seemed more imposing than the rest, holding a bony silver sword by the hilt in his shining gold hand.*
A:	*"Hallowed be the name of his majesty, King Gautir" the King boomed through the open archway.*
B:	*The King was a corpulent, middle aged man with his long white hair growing out of his crown, nearly the size of his small face.*
A:	*I could see that his eyes were piercing like the flaming sun.*
B:	*The King's long, white robes seemed long like a black robe worn by someone with a wide girth.*
A:	*His sneering smile was reflected in his large, deep blue eyes.*
B:	*"Today, we greet you both, members of the committee, from the Holy Empire of All Flame and Sound."*
A:	*The King's voice boomed as the rest of the Dictators turned and stepped towards the podium, stopping on their footsteps.*
B:	*"Welcome to this evening's meeting" the King announced to us.*
A:	*"You two can either choose to remain in your chambers or come out and face the Committee" The King said in a commanding tone.*

	Self-chat tuned+ranked Story 2
A:	*Have you ever wondered what exactly goes into making our favorite foods?*
B:	*You know, the big baked potato with oil and salt, the slushy that's savory enough to eat for lunch every day?*
A:	*The answer is simple: the food industry.*
B:	*It's a fairly big industry, you see.*
A:	*It's sort of estimated that five hundred different corporations control 70% of the world's food supply.*
B:	*If you were to think of it in simple terms, that's right: almost everyone on the planet eats food produced by a food corporation.*
A:	*As you might imagine, the world is full of giant corporations and a huge amount of money being made through the food industry.*
B:	*We've all heard the expression "money talks, money does."*
A:	*What do you think happens when that money goes through the food industry?*
B:	*It goes into the mouths of politicians and it goes into the mouths of stockholders.*
A:	*This whole industry is very complicated, although people never seem to come up with any clear-cut answers for it.*
B:	*But, really, the only thing that people seem to have a handle on about the food industry is that it's a big financial mess.*
A:	*When you think about it, this seems pretty clear.*
B:	*Let me explain.*
A:	*In order to make money, companies have to produce enough food to feed a population.*
B:	*When a company creates more food than it can eat, it has to buy more food, so that it can keep producing more food.*
A:	*More food means more food prices, so that the corporation can make more money.*
B:	*So how does the food industry achieve this?*
A:	*The answer to this is pretty simple.*
B:	*In the world of food production, companies come up with marketing schemes that manipulate people's tastes.*
A:	*Using a carrot for example, a company might work to improve a persons' reaction to carrots.*

Table 5: Example stories generated by self-chat with the `tuned+ranked` system.

react to the contributions made by human storytellers. In Collaborative Storytelling Story 1 from Table 1, we see an example of that creativity, when our system introduces the plot twist that the man and women not only know each other but have been living together for year. In Story 2 from the same table, we see our system's ability to play along with a human storyteller when the system accepts its collaborator's assertion that the squirrel can speak English and starts crafting dialogue for it.

5.2 Limitations

The limitations of our approach are that our storytelling system has a very shallow model of the world, which can lead to incoherent output. This is illustrated by the self-chat Story 2 in Figure 4: the narrative makes jarring shifts in setting and lacks overall cohesion. Such problems in cohesion are often amplified in self-chat settings, as the model lacks human input to reign it in.

In addition, because the storytelling model lacks explicit story structure, it can be hard to steer toward desired output, such as a human-preferred genre or mood, or generation of story endings on demand. We plan to address these issues in future work by adding more structure to the data used to train our models.

Finally, evaluation of this task is challenging: because interaction with human players introduces variance into the output, it is difficult to directly compare generated stories, but at the same time, evaluation limited to self-chat is not fully reflective of our desired task setting. Once our system has been implemented in a suitable agent, we plan to carry out detailed subjective evaluation of the collaborative storytelling experience of volunteers to gain further insights about our task and approach.

6 CONCLUSION

In this paper, we introduced the novel task of *collaborative storytelling*, where humans and AI agents work together to make stories. We presented a collaborative storytelling system that tunes a large-scale neural LM on storytelling data and uses a sampling-and-ranking approach to select more human-preferred story continuations. Quantitative evaluation of our system found that tuning and ranking both greatly contribute to its capability to generate story continuations that human evaluators prefer and consider acceptable. Qualitative evaluation of human evaluator preferences showed that humans found `tuned+ranked` more preferable than `tuned` and `tuned` more preferable than `untuned` in terms of engagingness, interestingness, and humanness metrics, as well as overall story quality preferences. Finally, we identified areas for potential future work, including evaluation of stories produced by humans and our system, integration of our system into intelligent agents such as robots and avatars, and improvement of generated story continuation quality by allowing genres or moods to be targeted.

REFERENCES

Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Jauvin. 2003. A neural probabilistic language model. *Journal of machine learning research* 3, Feb (2003), 1137–1155.

Hyundong Cho and Jonathan May. 2020. Grounding Conversations with Improvised Dialogues. *arXiv preprint arXiv:2004.09544* (2020).

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805* (2018).

Angela Fan, Mike Lewis, and Yann Dauphin. 2018. Hierarchical neural story generation. *arXiv preprint arXiv:1805.04833* (2018).

Asma Ghandeharioun, Judy Hanwen Shen, Natasha Jaques, Craig Ferguson, Noah Jones, Agata Lapedriza, and Rosalind Picard. 2019. Approximating interactive human evaluation with self-play for open-domain dialog systems. In *Advances in Neural Information Processing Systems*. 13658–13669.

Randy Gomez, Keisuke Nakamura, Deborah Szapiro, and Luis Merino. 2020. A Holistic Approach in Designing Tabletop Robot's Expressivity. In *Proceedings of the International Conference on Robotics and Automation*.

Prakhar Gupta, Vinayshekhar Bannihatti Kumar, Mukul Bhutani, and Alan W Black. 2019. WriterForcing: Generating more interesting story endings. In *Proceedings of the Second Workshop on Storytelling*. Association for Computational Linguistics, Florence, Italy, 117–126. https://doi.org/10.18653/v1/W19-3413

Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. 2019. The curious case of neural text degeneration. *arXiv preprint arXiv:1904.09751* (2019).

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).

Margaret Li, Jason Weston, and Stephen Roller. 2019. Acute-eval: Improved dialogue evaluation with optimized questions and multi-turn comparisons. *arXiv preprint arXiv:1909.03087* (2019).

Fuli Luo, Damai Dai, Pengcheng Yang, Tianyu Liu, Baobao Chang, Zhifang Sui, and Xu Sun. 2019. Learning to control the fine-grained sentiment for story ending generation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. 6020–6026.

Tomáš Mikolov, Stefan Kombrink, Lukáš Burget, Jan Černockỳ, and Sanjeev Khudanpur. 2011. Extensions of recurrent neural network language model. In *2011 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 5528–5531.

Hermann Ney, Ute Essen, and Reinhard Kneser. 1994. On structuring probabilistic dependences in stochastic language modelling. *Computer Speech and Language* 8, 1 (1994), 1–38.

Hae Won Park, Ishaan Grover, Samuel Spaulding, Louis Gomez, and Cynthia Breazeal. 2019. A model-free affective reinforcement learning approach to personalization of an autonomous social robot companion for early literacy education. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 33. 687–694.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language Models are Unsupervised Multitask Learners. (2019).

Abigail See, Aneesh Pappu, Rohun Saxena, Akhila Yerukola, and Christopher D Manning. 2019. Do Massively Pretrained Language Models Make Better Storytellers?. In *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*. 843–861.

Noam Shazeer and Mitchell Stern. 2018. Adafactor: Adaptive learning rates with sublinear memory cost. *arXiv preprint arXiv:1804.04235* (2018).

Ilya Sutskever, James Martens, and Geoffrey E Hinton. 2011. Generating text with recurrent neural networks. In *ICML*.

Christoph Tillmann and Hermann Ney. 2003. Word reordering and a dynamic programming beam search algorithm for statistical machine translation. *Computational linguistics* 29, 1 (2003), 97–133.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*. 5998–6008.

Alex Wang and Kyunghyun Cho. 2019. Bert has a mouth, and it must speak: Bert as a markov random field language model. *arXiv preprint arXiv:1902.04094* (2019).

Lili Yao, Nanyun Peng, Ralph Weischedel, Kevin Knight, Dongyan Zhao, and Rui Yan. 2019. Plan-and-write: Towards better automatic storytelling. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 33. 7378–7385.

Yizhe Zhang, Siqi Sun, Michel Galley, Yen-Chun Chen, Chris Brockett, Xiang Gao, Jianfeng Gao, Jingjing Liu, and Bill Dolan. 2019. Dialogpt: Large-scale generative pre-training for conversational response generation. *arXiv preprint arXiv:1911.00536* (2019).

Yukun Zhu, Ryan Kiros, Rich Zemel, Ruslan Salakhutdinov, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. 2015. Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. In *Proceedings of the IEEE international conference on computer vision*. 19–27.