STORIUM: A Dataset and Evaluation Platform for Machine-in-the-Loop Story Generation

Nader Akoury[†*] **Shufan Wang**[†] **Josh Whiting**[‡] **Stephen Hood**[‡]
Nanyun Peng[§] **Mohit Iyyer**[†]
[†]University of Massachusetts Amherst [‡]Storium [§]University of California Los Angeles
{nsa,shufanwang,miyyer}@cs.umass.edu
{josh,stephen}@storium.com
violetpeng@cs.ucla.edu

Abstract

Systems for *story generation* are asked to produce plausible and enjoyable stories given an input context. Existing datasets lack rich enough contexts to meaningfully guide models, while crowdsourced and automatic evaluations are unreliable for assessing long-form creative text. To address these issues, we introduce a dataset and evaluation platform built from STORIUM, an online collaborative storytelling community. Our author-generated dataset contains 6K lengthy stories with fine-grained natural language annotations interspersed throughout each narrative, forming a robust source for guiding models. We evaluate models directly on STORIUM, where *real* authors can query for suggested story continuations and then edit them.

1 Introduction

Machine-in-the-loop storytelling [1], in which an author obtains automatically generated sentences or paragraphs when stuck with writer's block, lowers the barrier to entry for creative writing [12]. To spur research in this area, we partner with STORIUM,[1] an online collaborative storytelling platform, to introduce a new dataset and evaluation methodology.

The open-endedness of story writing does not just pose a barrier to humans—it also presents a challenge for building and evaluating computational models. Prior work relies on datasets that are either too artificial to generalize to long-form stories, such as the crowdsourced ROCStories [9] corpus, or too unconstrained, as in the `r/writingprompts` dataset [2], which pairs medium-length stories with short prompts. Furthermore, lack of standardized evaluation makes measuring progress difficult: most prior work evaluates outputs using a combination of simple automatic metrics not designed for long-form creative text generation (e.g., BLEU and ROUGE against a single reference) and crowdsourced ratings [8, 16, 3] that preclude evaluating long-form narratives.

We address these limitations by (1) collecting a dataset of stories (Section 2) containing fine-grained structural annotations written in natural language, and (2) providing a platform for evaluating models in a machine-in-the-loop setting by allowing real STORIUM authors to interact with the generated stories (Section 4). Our dataset contains nearly 6K longform stories (125M tokens) written by STORIUM authors, each of which is broken into discourse-level scene entries annotated with narrative elements, such as character goals or abilities. Conditioning story generation on this information thus imposes loose constraints on what the model should produce, compared to unstructured datasets such as `r/writingprompts`, and also allows future research into modeling narrative planning processes.

[1] `https://storium.com`

Wordplay: When Language Meets Games Workshop @ NeurIPS 2020,
`https://wordplay-workshop.github.io/`

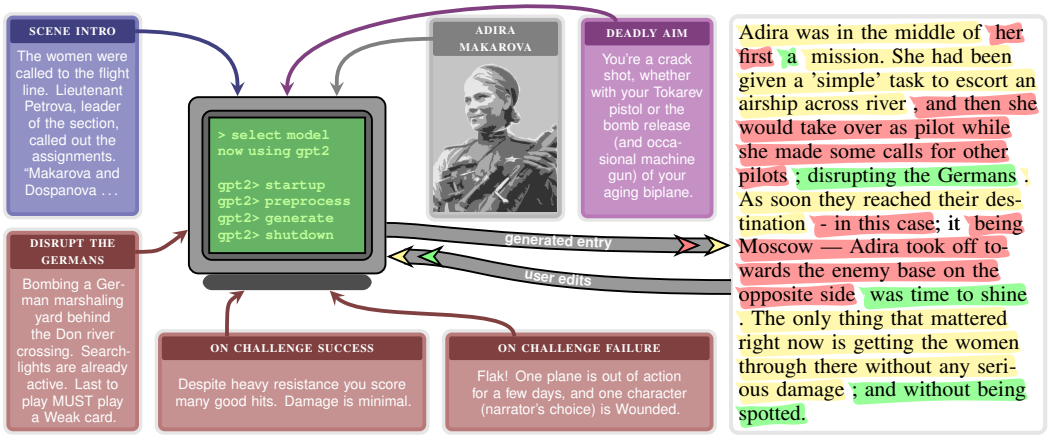

Figure 1: A high-level outline of our dataset and platform. In this example from a real STORIUM game, the character ADIRA MAKAROVA uses the strength card DEADLY AIM to DISRUPT THE GERMANS, a challenge card. Our model conditions on the natural language annotations in the scene intro, challenge card, strength card, and character, along with the text of the previous scene entry (not shown) to generate a suggested story continuation. Players may then edit the model output, by adding or deleting text, before publishing the entry. We collect these edits, using the matched text as the basis of our USER metric. New models can be added to the platform by simply implementing four methods: `startup`, `shutdown`, `preprocess`, and `generate`.

We integrate story generation models with the STORIUM platform, where authors can query a model for the next few sentences in their story and then edit the resulting text to their liking. We devise a metric (inspired by ROUGE) on top of these edits that measures how much of the generated text is preserved in the post-edited version, and discover that this metric correlates with Likert judgments of linguistic properties such as relevance and coherence. Detailed analyses of the edits (Section 5) suggests that generating text *relevant* to the current story context is the most important open problem in this area. We publicly release both the STORIUM dataset and user-facing evaluation platform to facilitate future research on story generation.[2]

2 STORIUM: A Gamified Storytelling Dataset

The STORIUM platform enables a small group of users to collaboratively write a single story by transforming the writing process into a turn-based game. In each game, one player acts as the *narrator*, while other players take on the role of individual *characters* within the story (e.g., ADIRA MAKAROVA in Figure 1). Stories unfold through a series of high-level *scenes* that consist of multiple short *entries*, each of which is written from the perspective of a character (or the narrator). Scenes commonly revolve around *challenges* (e.g., DISRUPT THE GERMANS), that the characters tackle within the text of their entries; to help address these challenges, each character has access to a set of *cards* (e.g., DEADLY AIM, a strength card) that define various properties such as strengths, weaknesses, items, and goals. The narrator moves the story forward by introducing new challenges, locations, and characters, in the form of cards. These are either created from scratch by the narrator or selected from a predefined *world* that contains a common set of story elements. Collectively, the cards played form a set of structural natural language annotations that guide the story being written.

[2] `https://storium.cs.umass.edu`

Authors	Characters	Scenes	Scene Entries	Cards Played	Average Tokens* per Entry	Average Tokens* per Story
30,119	25,955	25,092	448,264	232,596	247	19,278

Table 1: An overview of our dataset, which contains long stories, broken down into scene entries, with structural annotations in the form of cards played to guide the narrative. *We count tokens as contiguous spans of either alphanumeric or non-alphanumeric symbols.

Dataset details: We collect 5,743 publicly available stories written on STORIUM from January 2015 to August 2019. We reserve 569 stories for validation and 570 stories for test — carefully ensuring an 8:1:1 split with respect to both the number of stories and tokens* (126,041,738 total tokens).

Related datasets: Prior story generation papers have frequently focused on the ROCStories [9] and `r/writingprompts` [2] datasets. While STORIUM contains comparatively fewer stories than these datasets, our stories are an order of magnitude longer and contains natural language annotations to guide story generation. Rather than containing a single short prompt to start the story, our stories on average contain 14 narrator prompts[3] per story, with 41 natural language annotations which describe character goals, attributes, and key items useful for conditioning story generation models.[4]

3 Generating Scene Entries

We focus our modeling efforts on generating scene *entries*, which are the smallest units of each story, because we want to evaluate the generated text on the STORIUM platform within a machine-in-the-loop framework. We fine-tune the GPT-2 medium-sized (355M parameters) language model [11] for story generation, as it has been shown to generate coherent long-form prose and has successfully been used as a state-of-the-art model for story generation [7, 4]. To handle the compositional and semi-structured nature of the scenes and cards, we allow each input token to condition on an arbitrary number of *segment embeddings* [15].

During training, a single input instance to our models contains the text of the associated challenge, card metadata, the current character's biography, the scene's introductory text, as well as the the immediately preceding story entry and the current entry (Figure 1). At test time, we provide only the story context and autoregressively sample a scene entry.

4 A Machine-in-the-Loop Evaluation Platform

The inadequacies of existing human and automatic evaluation methods are a major roadblock for story generation research. Automatic evaluations correlate weakly with human judgments [13], and these judgments are obtained from crowd workers who are not invested in the narratives they are assessing.

To evaluate generated stories, we develop a dedicated web service for serving model outputs to the STORIUM platform. STORIUM users simply press a button on the user interface to obtain a generated scene entry conditioned on the story context. Users can then add new text while deleting any of the generated text that they wish (Figure 1). When users publish their edited entry, they are also asked to evaluate the generated text on a 5-point Likert scale with respect to *relevance*, *fluency*, *coherence*, and *likability*. Our framework makes adding a new model using any Python-based deep learning framework very easy, requiring implementation of only four methods: `startup`, `shutdown`, `preprocess`, and `generate`.

A Metric Over User Edits: Intuitively, the amount of generated text that a user preserves in their final published entry clearly indicates the usefulness of the generated text. We quantify this by developing User Story Edit Ratings (USER), inspired by the longest common subsequence (LCS)

[3]We count narrator actions introducing challenges and locations as prompts.

[4]Fan et al. [3] extract internal structure via SRL, but this can be applied to other datasets, including ours.

		Lik	Flu	Coh	USER	Rating
Rel	top-k	0.51	0.28	0.55	0.51	**2.55**
	nucleus	0.53	0.40	0.57	0.39	2.47
Lik	top-k	—	0.28	0.35	0.34	**3.32**
	nucleus	—	0.38	0.55	0.35	3.21
Flu	top-k	—	—	0.54	0.13†	**3.96**
	nucleus	—	—	0.61	0.23	3.76
Coh	top-k	—	—	—	0.25	**3.41**
	nucleus	—	—	—	0.36	2.96
USER	top-k	—	—	—	—	**15.63**
	nucleus	—	—	—	—	9.86

Table 2: Despite low ratings, relevance is clearly important as indicated by the moderately strong Pearson's r correlations (first four columns) with USER and the remaining human judgments. All correlations have $p < 0.01$, except those marked by † ($p > 0.05$).

	Top-k		Nucleus	
1st Run	Rating	κ	Rating	κ
Rel	**3.32**	0.09	3.27	0.13
Lik	**3.27**	0.07	3.22	0.11
Flu	**3.59**	0.17	3.47	0.11
Coh	**3.50**	0.10	3.44	0.20

	Top-k		Nucleus	
2nd Run	Rating	κ	Rating	κ
Lik	**3.28**	0.12	3.06	0.16
Flu	**4.01**	0.46	3.77	0.33
Coh	**3.63**	0.27	3.38	0.23

Table 3: The first crowd sourced judgments have low agreement (κ) and much higher relevance ratings than provided by STORIUM authors. A second run, removes context, thus excluding relevance judgments, but greatly increases agreement for fluency and coherence.

variant of ROUGE [6], applied to user edits. Given a generated entry X and the final published entry Y, we compute $\text{USER}(X, Y) = \frac{|\text{MATCH}(X,Y)|}{|X|}$, where $\text{MATCH}(X, Y)$ considers *contiguous substrings* with at least one non-stopword as matches (see Figure 1 for an example). We do not use ROUGE-L because vanilla LCS typically favors subsequences of unigram matches (often stopwords) over longer contiguous n-gram matches. In our STORIUM setting, users preserving n-grams or full sentences is a clear indication that the generated text was useful.

5 Analysis

In this section, we conduct experiments on our platform and analyze the edits by examining the correlation of USER to Likert scores. We also conduct a crowdsourced evaluation on Amazon Mechanical Turk that demonstrates its unsuitability for assessing relevance in generated stories.

Top-k vs. nucleus sampling: Using our platform (Section 4), we evaluate our model with two different decoding strategies: (1) top-k sampling [2] with $k = 40$, and (2) nucleus sampling [5] with $p = 0.9$. [5]

Interestingly, while Holtzman et al. [5] show that nucleus sampling improves over top-k sampling on measures like repetition, STORIUM users clearly prefer the top-k variant across all categories (last column of Table 2). We collect roughly 200 feedback ratings and 175 edits for each model over a span of three months beginning in late February 2020. We discover that both configurations score best on *fluency* and worst on *relevance*. This is unsurprising as (1) GPT-2 is known to produce fluent text and (2) the complex and lengthy STORIUM data is a challenge for limited-context models. Finally, USER scores are generally low (15.6 for top-k vs. 9.9 for nucleus sampling), indicating that users delete most of the current model's generated text.

Crowdsourced evaluation is unreliable: Thus far, we have argued for our evaluation platform by claiming that crowdsourced methods are unsuitable for evaluating stories with complex and lengthy contexts. Here, we measure fluency, coherence, relevance, and likability of our generated entries with a crowdsourced Amazon Mechanical Turk task, to see if the results correspond to STORIUM user ratings. Designing this crowdsourced task is difficult, as we cannot show crowd workers the entire story context due to its length; we thus decide to show the same inputs that the model receives (Section 3). We collect ratings of 100 examples per model, with three judgments per example.[6]

[5]The sampling parameters, such as the k in top-k sampling, can significantly affect output quality of story generation models [14], so we choose values that worked well in prior work [10].

[6]We limit HITs to crowd workers living in the US and the UK, with over 1000 completed HITs and a 99% approval rating. We pay 50¢ per HIT, by assuming 2 minutes per annotation, for an effective hourly rate of $15.

Table 3 (top) shows that workers have very low agreement (Fleiss' κ) for all properties, even fluency. Crowd workers also rate relevance much higher than the STORIUM authors (Table 2). An analysis of the median task completion time reveals most workers did not actually read the context. We run a second experiment, showing only the generated text (no context), and remove the relevance rating. Table 3 (bottom) shows this improves agreement (Table 3), and that the average ratings align closely with those from STORIUM users. Overall, our struggle to obtain quality judgments from Mechanical Turk further validates our platform: STORIUM provides free expert judgments from people invested in storytelling.

Author Contributions

Dataset Analysis: Akoury, Wang
Generation Model: Akoury, Wang
Evaluation Platform: Akoury, Whiting, Hood
Research Guidance: Iyyer, Peng

Acknowledgements

We thank the wonderful STORIUM users for actively using our story generation models and generously providing their time to be interviewed. We also thank the amazing UMass NLP community for thoughtful insights on our paper and helping to validate whether structural metadata influences story text on STORIUM. Akoury and Iyyer were supported during this project by a research gift from Genpact. Peng was supported in part by the CwC program under Contract W911NF-15-1-0543 with the US Defense Advanced Research Projects Agency (DARPA).

References

[1] Elizabeth Clark, Anne Spencer Ross, Chenhao Tan, Yangfeng Ji, and Noah A. Smith. Creative writing with a machine in the loop: Case studies on slogans and stories. *23rd International Conference on Intelligent User Interfaces*, 2018.

[2] Angela Fan, Mike Lewis, and Yann Dauphin. Hierarchical neural story generation. In *Proceedings of the Association for Computational Linguistics*, 2018.

[3] Angela Fan, Mike Lewis, and Yann Dauphin. Strategies for structuring story generation. In *Proceedings of the Association for Computational Linguistics*, 2019.

[4] Jian Guan, Fei Huang, Zhihao Zhao, Xiaoyan Zhu, and Minlie Huang. A knowledge-enhanced pretraining model for commonsense story generation. *Transactions of the Association for Computational Linguistics*, 8:93–108, 2020. doi: 10.1162/tacl_a_00302. URL https://www.aclweb.org/anthology/2020.tacl-1.7.

[5] Ari Holtzman, Jan Buys, Maxwell Forbes, and Yejin Choi. The curious case of neural text degeneration. *iclr*, 2020.

[6] Chin-Yew Lin. Rouge: A package for automatic evaluation of summaries. In *ACL 2004*, 2004.

[7] Huanru Henry Mao, Bodhisattwa Prasad Majumder, Julian McAuley, and Garrison Cottrell. Improving neural story generation by targeted common sense grounding. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5988–5993, Hong Kong, China, November 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1615. URL https://www.aclweb.org/anthology/D19-1615.

[8] Neil Duncan McIntyre and Mirella Lapata. Learning to tell tales: A data-driven approach to story generation. In *ACL/IJCNLP*, 2009.

[9] Nasrin Mostafazadeh, Nathanael Chambers, Xiaodong He, Devi Parikh, Dhruv Batra, Lucy Vanderwende, Pushmeet Kohli, and James Allen. A corpus and cloze evaluation for deeper

understanding of commonsense stories. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 839–849, San Diego, California, June 2016. Association for Computational Linguistics. doi: 10.18653/v1/N16-1098. URL https://www.aclweb.org/anthology/N16-1098.

[10] Lianhui Qin, Antoine Bosselut, Ari Holtzman, Chandra Bhagavatula, Elizabeth Clark, and Yejin Choi. Counterfactual story reasoning and generation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5043–5053, Hong Kong, China, November 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1509. URL https://www.aclweb.org/anthology/D19-1509.

[11] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. 2019. URL https://openai.com/blog/better-language-models/.

[12] Melissa Roemmele and Andrew S Gordon. Creative help: a story writing assistant. In *International Conference on Interactive Digital Storytelling*, 2015.

[13] Manasvi Sagarkar, John Wieting, Lifu Tu, and Kevin Gimpel. Quality signals in generated stories. In *Proceedings of the Seventh Joint Conference on Lexical and Computational Semantics*, pages 192–202, New Orleans, Louisiana, June 2018. Association for Computational Linguistics. doi: 10.18653/v1/S18-2024. URL https://www.aclweb.org/anthology/S18-2024.

[14] Abigail See, Aneesh Pappu, Rohun Saxena, Akhila Yerukola, and Christopher D Manning. Do massively pretrained language models make better storytellers? In *Conference on Computational Natural Language Learning*, 2019.

[15] Thomas Wolf, Victor Sanh, Julien Chaumond, and Clement Delangue. Transfertransfo: A transfer learning approach for neural network based conversational agents. *ArXiv*, abs/1901.08149, 2019.

[16] Lili Yao, Nanyun Peng, Ralph Weischedel, Kevin Knight, Dongyan Zhao, and Rui Yan. Plan-and-write: Towards better automatic storytelling. In *Association for the Advancement of Artificial Intelligence*, 2019.