

Tell Me A Story Like I’m Five: Story Generation via Question Answering

Louis Castricato *, Spencer Frazier , Nitya Tarakad , Jonathan Balloch and Mark O. Riedl

Georgia Institute of Technology

{lcastric, sf, ntarakad3, balloch}@gatech.edu, riedl@cc.gatech.edu

Abstract

Neural language model-based approaches to automated story generation suffer from two important limitations. First, language model-based story generators generally do not work toward a given goal or ending. Second, they often lose coherence as the story gets longer. We propose a novel approach to automated story generation that treats the problem as one of generative question-answering. Our proposed story generation system starts with sentences encapsulating the final event of the story. The system then iteratively (1) analyzes the text describing the most recent event, (2) generates a question about “why” a character is doing the thing they are doing in the event, and then (3) attempts to generate another, preceding event by answering this question. We show that the coherency of a story can be measured as the relative entropy over the distribution of responses to claims about said story’s events. Using a within-subjects human evaluation we measure this coherency entropy over the responses to sets of True-False statements for multiple stories generated by our model and each baseline. The evaluation shows that our system generates stories that are on average 15.9% more coherent than those generated by the BART [Lewis *et al.*, 2019] language model fine-tuned on a story corpus to generate sentences in reversed order to more closely match our process.

1 Introduction

Consider a story, at the ending of which a princess is reunited with her lover thought to be lost at sea, a swordsman has enacted revenge on the man who killed his father, and a giant becomes a pirate. One might reasonably wonder how this situation came to pass. Aristotle writes in *Poetics* that the events of the story serve the plot and the end. Under this interpretation, storytelling is explanation—every event answers the question of “how did the next event come to pass?”. In this paper, we propose an automated story generation system using the principles of question-answering and show how it can

improve automated story generation capabilities.

Automated Story Generation is the challenge of designing an artificial intelligence system that can generate a story from a minimal number of inputs—often just a prompt and some storytelling knowledge. Symbolic story and plot generation systems have traditionally relied on planning or case-based reasoning (see Gervás [2009] for an overview of symbolic story generation systems). Some of these systems start with an end state—the state the fictional world should be in at the end of the story—and work backward, determining what must have happened to transform an initial world state into the goal. These systems often generate coherent stories guaranteed to end in a given state. Their drawback is that they require significant hand-authored domain knowledge.

Machine learning-based story generation systems acquire or learn story domain knowledge from data, often corpora of human-authored stories. Most machine learning-based story generation systems have relied on neural network-based language models. Auto-regressive neural language models trained on a corpus of stories learn a probability distribution over tokens $p(t_n|t_{n-1}, t_{n-2}, \dots, t_{n-k})$ based on the tokens that occur in the training corpus. This distribution can then be sampled to create new texts that emulate the training corpus. Training a neural language model on story corpora results in a generative model that produces texts that look like stories [Roemmele, 2016; Khalifa *et al.*, 2017; Martin *et al.*, 2018]. However, language model based approaches are unable to bring stories to a particular conclusion or goal state. Stories generated by language models also tend to lose coherence over time as they rely on probabilistic sampling and do not learn a richer model of the story world.

We consider how neural story generation systems can be induced to generate more coherent narratives that also end in a pre-determined, desirable way. Narratives are perceived to be coherent when events are related to each other in a way that is comprehensible by the reader [Trabasso and Van Den Broek, 1985; Graesser *et al.*, 1991]. There are many relations between events which fit this need, the most important are: (1) causal relations—one event cannot happen if another event had not happened prior to it—and (2) character goal hierarchies—an action is in service of a goal or another action that is in service of a goal.

Our insight is that if each event in the story is generated to explicitly answer the question of “why” the next event in the

*Contact Author

story happens, then readers will perceive the story as more coherent. To generate a story that will be perceived as a coherent and build up to a pre-determined ending, we propose to generate the story backward. This is achieved by starting from a textual description of the final event; each event added best answering the question of what must have preceded it. Our system, EDGAR, repeats this process for a specified number of iterations. Questions are generated using a commonsense inference model, Para-COMET [Gabriel *et al.*, 2020], to predict what readers are likely to believe about a story event; the inferences are transformed into questions using templates. EDGAR then attempts to answer each question using a generative question-answering model.

We evaluate our system against a baseline neural transformer-based language model approach that is fine-tuned to generate story events backward, matching the backward process of EDGAR. We measure story coherence with two human-participant studies. In the first, perceived coherence is measured as the entropy in participant responses to true/false questions about the story; a story that is more comprehensible results in less random guessing by human readers. We find that EDGAR generates more coherent stories than the baseline as evidenced by the entropy of answers about stories generated by EDGAR had 15.9% lower entropy than those of the baseline. The second evaluation is subjective—we qualitatively measure coherency via subjective questionnaire about coherence. Participants consider stories written by EDGAR twice as coherent as those written by the baseline.

2 Related Work

Gervás [2009] overviews early symbolic story generation systems. Story generation systems that use symbolic story planners utilize logic-like domain representations that provide knowledge about available actions, their preconditions, and their effects. A search process—such as that by Riedl and Young [2010]—selects a goal condition or a precondition of an action in the plan and attempts to find another, preceding action that has an effect that establishes the condition. This process iterates, creating chains of preconditions and effects until everything is grounded in the initial world state. However, the chaining can be done forward from the initial state to the goal as well [Ware and Young, 2010].

Neural networks have the potential to generate a greater range of stories by learning model for how to tell stories from a corpus of exemplar stories. Neural language models learn the probability that one or more tokens will occur given a history of one or more prior tokens, $P_{\theta}(t_{n+1}, \dots, t_{n+m} | t_{n-k}, \dots, t_{n-1}, t_n)$, according to token occurrence patterns in a corpus. Neural language models can be induced to generate text that can be read as a story by sampling from the learned distribution over tokens and appending them to a prompt. Some neural language model based story generation techniques include [Roemmele, 2016; Martin *et al.*, 2018; Khalifa *et al.*, 2017]. However, a neural language model alone is incapable of achieving a specific end state or event. Sampling from a distribution over tokens only considers the most likely successive tokens given a win-

dow of prior tokens. Neural language models also tend to lose story coherence over time. This is due to the fact that a language model only models a distribution over tokens in the training set. Additionally, the hidden parameters of current neural networks are unlikely to encode the state of a fictional world, as human readers would understand.

Tambwekar *et al.* [2018] attempt to train a neural language model to generate toward a given goal. They fine-tune a neural language model with a policy-gradient reinforcement learning technique that rewards the language model for generating events progressively closer to the goal event. This has the benefit of improving readers’ perceptions of coherence, but—being based on a language model—does not ensure that any transition from one event to the next will always be perceived as related.

Other neural language model approaches to story generation using neural networks use *hierarchical conditioning*, in which a high-level guidance specification is given either periodically or per sentence in the story [Fan *et al.*, 2018; Yao *et al.*, 2019; Rashkin *et al.*, 2020; Ammanabrolu *et al.*, 2020b]. These high-level guidance specifications turn the generation problem into a supervised learning problem. We do not consider these approaches further in this paper because we do not assume the existence of a guidance specification.

One approach to automated story generation that uses neural networks that are not based on language modeling is C2PO [Ammanabrolu *et al.*, 2020a], which uses the COMET [Bosselut *et al.*, 2019] commonsense inference engine to generate successor and predecessor events, performing a bi-directional search from a given start event and a given end event. It is relevant to our work in that it does partially chain backward from a given end event, and also uses a commonsense inference engine. However, C2PO generates plots made up of short statements of character intentions, whereas our system generates stories that have more descriptive detail.

3 The EDGAR System

The *Explanatory Drama Generation And Recall* (EDGAR) system constructs a story backwards from a given sentence describing the end of the story. The system contains three major components. The first component is a question generator. Given a story context—the sequence of text describing the earliest event in the ending context—a set of questions about the event is generated. Second, a question answering component attempts to generate text describing one or more events that answer that question. A number of candidate answers are generated for each question. Finally, the answers are iteratively pre-pended to the context and a ranker chooses the best sequence. The best sequence is added to the story and the process iterates. See the pipeline in Figure 1.

3.1 Question Generation

We use Para-COMET [Gabriel *et al.*, 2020] to generate questions. Para-COMET is a commonsense inference model trained to generate potential commonsense explanations about one or more sentences of input text. Inferences have types. `xIntent` in particular, explains what a character in the sentences might have intended in performing any

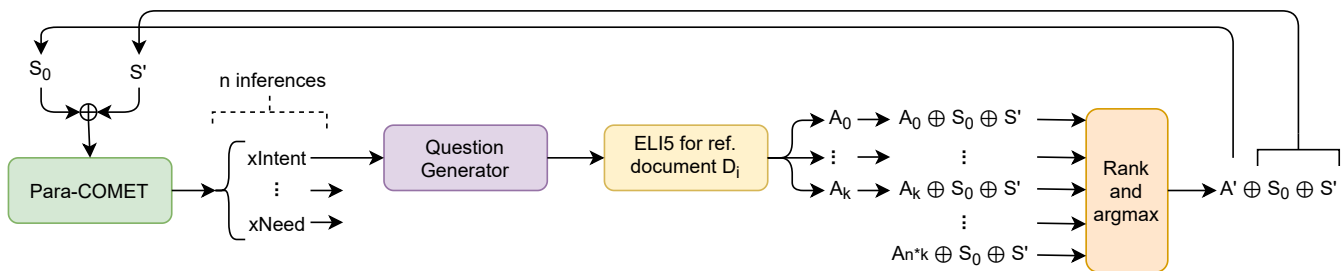


Figure 1: EDGAR generates stories backward. Given the end of the story where S_0 is the earliest event sequence and S' is the remainder, Para-COMET generates a set of n inferences. Each inference is converted a question and the ELI5 QA model generates $k + 1$ answers. The answers are concatenated to the beginning of the story and the ranker selects the best scoring story. This process is repeated.

actions in the sentences. These correspond to goal relations in reader comprehension [Graesser *et al.*, 1991]. $xNeed$ explains what a character might have needed to perform any actions in the sentences. These provide precondition-like inferences, corresponding to causal relations in reader comprehension [Trabasso and Van Den Broek, 1985]. We discard all other relation types.

Because Para-COMET works on multi-sentence sequences, we extract a rolling window of the last 5 $xIntent$ and $xNeed$ inferences. However, Para-COMET does not identify which character is associated with each $xIntent$ and $xNeed$, which is problematic for stories with more than one character. To associate the $xNeed$ and $xIntent$ clauses with a character, we generate the following templates:

- “Who needs to $xIntent$ ”
- “Who needs $xNeed$ ”

filling in the details of the inferences. These filled templates are provided as input to RoBERTa [Liu *et al.*, 2019], a question-answering model. The outputs are the names of the characters most likely to have had these needs and intents.

Finally, we use a second set of templates to assemble the final set of questions:

- “Why does *character* do $xIntent$?”
- “What does *character* do to need $xNeed$?”

This process generates a total of 8 questions.

3.2 Question Answering

Once we have a set of questions, EDGAR generates candidate answers, such that each candidate can be added to the beginning of the story context so far. To generate sentences describing the preceding event that answers the questions generated, we feed the questions into the ELI5 QA model [Fan *et al.*, 2019]. The ELI5 QA model is a long-form, question-answering model trained on the *Explain Like I’m Five* Reddit corpus,¹ in which people give long, yet easily comprehensible answers to open-ended questions as one might give to a five-year old. ELI5 QA requires a reference document from which to abstract answers. The reference document is the source material—in this case a story—that ELI5 QA uses to generate an answer. Because EDGAR is an unsupervised

technique designed to generate novel stories, there is no one reference document that should be used; using a single reference document would run the risk of accidentally recreating a human-written story. For every iteration we randomly select a reference document from the *Flash Fiction Online* repository.² The question templates above were constructed to induce relatively short answers from ELI5, which has a tendency to generate very long explanations.

We use beam search to generate 15 candidate answers for each question. As another measure to prevent ELI5 from providing overly verbose explanations, we have accumulated a list of over 700 unique banned phrases, which occur when ELI5 commentators point out “facts” or likening a character’s action to mental disability. This blocked phrases list was accumulated iteratively, by rerunning the model repeatedly and adding any toxic phrases to this excluded list. The result is $n \times k$ story continuations where n is the number of questions, k is the number of beams per question on ELI5.

3.3 Ranking

Once EDGAR has generated a set of candidates, the final step in the process is to select the best candidate for prepending to the context (the end of the story). We prepend each answer to the context and rate each resulting text sequence using GPT-2 [Radford *et al.*, 2019] to assess the probability of the sequence. GPT2 was fine-tuned on the science fiction summary corpus [Ammanabrolu *et al.*, 2020b] dataset, which consists of 2,276 high-quality plot summaries from science fiction TV and movie wikis. We fine-tune on the science fiction summary corpus because wiki plots do not include descriptive details or dialogue; our ranker thus prefers more plot-like narrative content. Candidates are ranked by perplexity of the GPT2 model. The normalized perplexity distribution over the beams outputted by ELI5 refers to the $1 - probability$ distribution of a body of text existing within the distribution of science fiction summaries.

Ranking is an important step because of the numerous processes involved; as a consequence the ranking of ELI5 beam distribution does not necessarily correlate with the final ranking, which roughly measures fluency. The best scoring candidate is added to the overall story. The process repeats with the new, longer story, attempting to determine what happened

¹<https://www.reddit.com/r/explainlikeimfive/>

²<https://www.flashfictiononline.com>

just before the new context.

4 Objective Evaluation

We hypothesize that EDGAR, by virtue of question-answering, can generate more coherent stories than a pure language modeling technique. We define coherence as any perceivable relationship between events in a story. Research on reading comprehension [Trabasso and Van Den Broek, 1985; Graesser *et al.*, 1991] suggest that causal and goal relationships are particularly important.

Common automated evaluation metrics for story generation such as perplexity and BLEU are insufficient as they only measure whether a generator can recreate the ground truth corpus. A story may deviate from the ground truth and be considered a good story—indeed this is a desirable property of an automated story generator. Furthermore, systems such as ours may be unsupervised and have many components that intentionally push a language model away from any one corpus, thus making perplexity less meaningful. For these reasons, story generation research often relies on human participant studies with subjective questions.

We assert human participant studies are the best way to assess the coherence of generated stories. Question-answering protocols, wherein questions are asked about a story, have been proposed as a means to make human-participant evaluations more objective [Riedl and Young, 2010; Cardona-Rivera *et al.*, 2016]. We conduct a human-participant evaluation using a new metric based on question-answering protocols, *Entropy Index*, which is an objective measure of story coherence based on human question-answering.

4.1 Baselines

The BART [Lewis *et al.*, 2019] neural language model was used as a baseline, but fine-tuned to generate events backward to conform to EDGAR and guarantee the presence of a given end event. The dataset used to fine-tune consisted of 2276 narratives from a science fiction summary corpus [Amanabrolu *et al.*, 2020b]. The narratives are preprocessed to create our dataset. From every narrative, $2 + 2k$ sequential sentences are obtained, where k is a random integer less than 5. The $2 + 2k$ sentences are split apart into 2 sentences and $2k$ sentences, creating the source and target of the dataset respectively. The 2 sentences generated in the $2 + 2k$ sentence chunk always precede the $2k$ sentences, establishing a relationship between sequential sentences. We preprocess this data to this format because an attribute found in most narrative summaries within our dataset is that preceding sentences to any given sentences gives some notion of causality. BART utilizes seq2seq as its translation architecture. As a consequence of the input data format, our fine-tuned Backward-BART—which we refer to as bBART—can generate narratives backwards by assessing the causality between sequentially sentences.

Human-written stories from the ROCStories corpus [Mostafazadeh *et al.*, 2016] were also included in our evaluation as a point of comparison. These stories have a definitive causality between sequential sentences.

4.2 Method

To evaluate the objective coherence of stories, we turn to cognitive psychology. Cognitive psychology research suggests that recall is strongly correlated with narrative causal coherence [Trabasso and Van Den Broek, 1985]. The cognitive load of inferring entailments about a story is strongly correlated with how well the story conveys information about its *fabula*³ [Carney, 2019]. We devise a new evaluation methodology wherein we ask participants to read stories and then answer true/false questions about how the events of the story relate to each other. We measure the amount of agreement between readers’ answers in terms of *entropy*. If the story is coherent, readers will come to the same conclusions about the truth or falseness of the questions, and entropy will be low. If the story is incoherent, readers—forced to choose between true and false—will choose more randomly, resulting in higher entropy. We do not require a ground truth “correct” answer to each question in order to compute the entropy; this is a desirable property of our methodology given (1) there are no algorithmically produced ground truth answers to the true/false questions and (2) obtaining a ground truth answer from humans can be noisy. Our index method is inspired by the evaluation used in Li *et al.* [2012] where human participants were asked to choose event orderings and participant agreement was assessed as entropy.

We generated 11 stories using EDGAR, 11 stories using backward-BART, and randomly selected 11 stories from the ROCStories corpus. Stories were generated by running the respective systems 3 iterations. Stories ranged from 5 sentences to 20 sentences in length. See Table 1 for examples from EDGAR and bBART. The Appendix gives the entire set of stories used in the evaluation. For the 33 stories, we produced 7 true/false questions for each story using the technique described in Section 4.3. To avoid bias, a non-computer-science graduate student not affiliated with the research project was paid to write the questions using the above template.

We recruited 180 human-subject participants from Mechanical Turk. Participants were recruited from countries that speak English as the national language. Each participant was asked to read 3 stories and answer the 7 true/false questions after each story. Participants were paid \$7 for a 15 minute task. The first story is a “screener” story, an uncommon fable that is easy to understand. If a participant did not answer the questions how we expect, we eliminated the participant from the pool. Participants were also eliminated from the pool if they resorted to marking all questions true or false or marked questions in some otherwise visually obvious repeating pattern; we eliminated 26 participants.

4.3 Computing the Entropy Index

We define the entropy index of each story as follows. For a given story generation system, we randomly selected 11 generated stories. For each story, we then produced 7 entailment questions about each story. Entailment questions are of the form of implications. By asking the reader to answer true or

³A story’s *fabula* denotes the chronological sequence of events in a narrative.

Stories generated by EDGAR

He needs to find a way out of the house. If he wants to go to the kitchen, he must first find a way to escape from the house. This means that he must either run away or fight his way past the lock. He goes to the bathroom. The only way to get to the kitchen is through a locked door, which Hansel has never been able to do before. However, when he opens the door, it doesn't take him very long at all. **Hansel's hand still trembles as he pushes open the twice-cooked door. The last time he saw the house he was glancing back over his shoulder as he and his sister fled into the trees.**

If it does not have a pulse then it is dead. The heart rate of an animal tells them if it is alive or dead. There is no such thing as absolute death. He is able to tell the difference between something being alive and something being dead, so when he looks at the house, he feels like he's seeing things that aren't there. This makes him feel uncomfortable because he doesn't want to be in that situation. It's similar to how people can see ghosts or monsters from inside their head but they don't know what those things are. **Hansel's hand still trembles as he pushes open the twice-cooked door. The last time he saw the house he was glancing back over his shoulder as he and his sister fled into the trees.**

Stories generated by bBART

A young boy named Hansel has been walking through town recently looking at some strange things that are going on outside their walls. In fact, they were just playing cards before anyone noticed them. A young boy called Hansel has been walking through town lately looking at some odd things that are gone missing outside their walls – maybe even someone noticing them. In reality, they were probably playing cards before everyone noticed them. He wants to learn more about it though, but can always tell for sure. He wants to know more about it, but can never tell for sure. After all, this is a city of humans who have no interest in living here. He wants to know what happened to him, but can only tell for sure if this is a City of Humans who have little interest in living there. Hansel's hand still feels like an old man with no interest whatsoever. His mind wanders frantically around trying to figure out how to live here. **Hansel's hand still trembles as he pushes open the twice-cooked door. The last time he saw the house he was glancing back over his shoulder as he and his sister fled into the trees.**

Table 1: Example stories generated by EDGARnd backwards BART (bBART). Bold text indicates the given ending prompt.

false we are asking the reader to prove or disprove the statement within the realm of what has been presented about the story world.

In order to ensure our questions were not biased, we provide annotators the following templates, two of which are given as examples here:

- E_i depends on E_j
- E_i could be removed and the story would still make sense.

$i < j$ and E refers to an event within the story. The full set of templates can be found in the appendix. The questions themselves were manually written to ensure grammatically correctness and readability.

The answers to the entailment questions give us a measure of entropy. When participants disagree, it can be determined how ambiguous their model of the story world is, such that they must rely heavily on external bias.

Consider that we have some story, S , composed of an event chain $E = \{E_i\}_n$. An event chain being a sequence of events discussed in a story, one path in a fabula. Generate two events, one that could be inserted into E and preserve coherence and its negation. We'll refer to these events as A and B . Refer to their insertions as E^A and E^B . Assume that we had some function $f(\cdot)$ that could take either E^A or E^B and rank all of the explanations for A and B respectively by mental load induced on the reader. Then, if E^B is coherent, consider what mental leaps are required by the reader for justification. Let $D(A)$ and $D(B)$ refer to these normalized distributions respectively. Measure the following:

$$KL_A = KL(D(A)||U) \text{ and } KL_B = KL(D(B)||U) \quad (1)$$

Where KL is Kullback–Leibler divergence and U is a uniform distribution. Inductively if A and B are in direct contradiction

of each other, we can collapse the above statement to

$$KL_{A,B} = KL\left(D\left(\sum f(E^A), \sum f(E^B)\right)||U\right) \quad (2)$$

In this case, since U is of dimension two, simplify the above to entropy. We can conclude that measuring the coherence of such an insertion is equivalent to measuring the entropy over the answers to a similarly constructed T/F statement about a causal relationship within a story. Over a large number of questions and stories per model, the above serves as a sound proxy for coherence. Consider a coherent story and a set of T/F questions concerning this story. It is often easier to disprove a statement about a coherent story than it is to prove a statement about an incoherent story [O'Brien and Albrecht, 1992; Albrecht and O'Brien, 1993]. By utilizing the format of T/F questions, the above will tend to converge to zero on a coherent story as there will always be one option that is disprovable. To get a large enough sample, we used 77 questions per model over 11 stories.

4.4 Results

The evaluation results are plotted in Figure 2. The evaluation shows that EDGAR scores a median of 0.427 on the entropy index, compared to bBART's median of 0.508. Human written stories from the ROCStories corpus scored a median entropy of 0.26.

From these results we can draw a number of conclusions. First, the median entropy of human-authored stories is over 95% better than bBART and over 63% better than EDGAR. This implies that human-authored stories are much more coherent than computer-generated stories according to our Entropy Index metric. This is the expected result and shows that our Entropy Index metric is operating as expected. The human story entropy index is a lower bound. Importantly, the median entropy EDGAR is 15.9% lower than that of

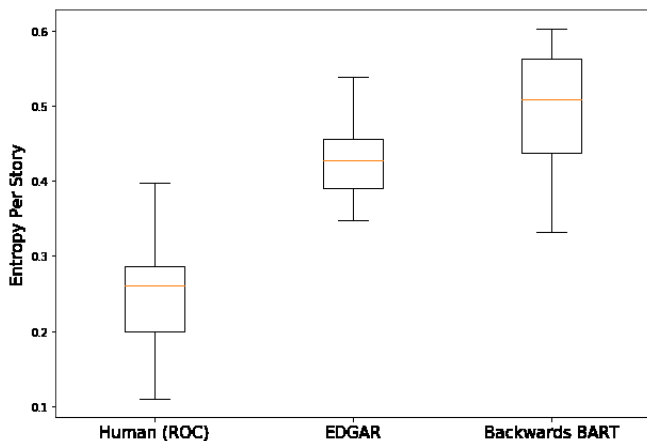


Figure 2: The entropy indices for human-written stories, EDGAR, and backwards BART (bBART). Lower is better.

the bBART baseline, indicating that our technique has improved the coherence of generated stories when generating backwards in order to ensure a given ending.

5 Subjective Evaluation

We conducted a second human-participant evaluation in which participants read stories and answered subjective questions about the coherence of the stories. We would expect the results of this experiment to concur with the results of the previous experiment.

Purdy *et al.* [2018] proposes a number of questions to be used to evaluate story generation systems. They have been used in a number of story generation system evaluations (cf. [Tambwekar *et al.*, 2018; Ammanabrolu *et al.*, 2020b; Ammanabrolu *et al.*, 2020a]). We use a subset of the questions and adapt them to rank-order choice between stories from two systems:

- Which story’s events occur in a more PLAUSIBLE ORDER?
- Which story’s sentences MAKE MORE SENSE given sentences before and after them?
- Which story better follows a SINGLE PLOT?
- Which story is of HIGHER QUALITY?
- Which story is more ENJOYABLE?

The first three questions ask about different aspects of perceived story coherence.

5.1 Method

We used the same stories from the first evaluation and the same baselines. Participants read two stories from two different sources back-to-back. Then for that pair of stories, the participant was asked to answer the subjective questions above, picking between the two stories.

We recruited 48 human-subject participants from Mechanical Turk. Participants were recruited from countries that speak English as the national language. Each participant was asked to read 4 stories, presented in pairs of two, and answer

Question	EDGAR	bBART	<i>p</i> -value
Plausible	31	15	0.013
Single plot	32	14	0.005
Makes sense	29	17	0.052
Quality	31	15	0.013
Enjoyable	31	15	0.013

Table 2: Total counts of times per question in the subjective evaluation that participants selected a story generated by each system. *P*-tests were determined to ensure that the chance of EDGAR winning a pairing was greater than 50/50.

the 5 questions after each story. Participants were paid \$5 for a 10 minute task. We screened participants by asking them similar questions about human written stories but inserted the answers to the questions in the directions, to determine their attentiveness. Participants that were considered inattentive where disqualified.

5.2 Results

The results are summarized in Table 2, which shows the number of times, per question, a participant selected the story from each system. When forced to pick between stories generated by EDGAR and stories generated by Backward-BART, participants chose stories generated by EDGAR twice as often for every question asked. A one-tailed binomial *p*-test for the results of each question determines EDGAR was significantly preferred above the baseline for every dimension at $p \leq 0.013$ except the “Makes sense” dimension, which was significant at $p = 0.052$. These results suggest that EDGAR generates more coherent and overall better quality stories than Backward-BART. These results are consistent with the Entropy Index metric, confirming that the metric is also measuring coherence.

6 Conclusions

We propose a new approach to neural story generation that treats story generation as question-answering problem—given an ending, the story must answer the question of how the ending comes about. Our proposed EDGAR system generates backward from the ending event to ensure the presence of the desired ending. It decomposes the generation process into distinct processes for using human commonsense to produce questions and then to answer them. These processes are grounded in reader narrative comprehension. We show that stories generated by EDGAR are more coherent than stories generated in a more conventional language modeling approach based on subjective and objective measures of perceived coherence. The EDGAR technique is a significant departure from techniques that sample from a language model that opens up new avenues for improving neural story generation in ways that are inspired by the comprehension needs of the human reader.

References

[Albrecht and O’Brien, 1993] Jason E Albrecht and Edward J O’Brien. Updating a mental model: Maintaining both local and global coherence. *Journal of exper-*

- imental psychology: Learning, memory, and cognition*, 19(5):1061, 1993.
- [Ammanabrolu *et al.*, 2020a] Prithviraj Ammanabrolu, Wesley Cheung, William Broniec, and Mark O Riedl. Automated storytelling via causal, commonsense plot ordering. *arXiv preprint arXiv:2009.00829*, 2020.
- [Ammanabrolu *et al.*, 2020b] Prithviraj Ammanabrolu, Ethan Tien, Wesley Cheung, Zhaochen Luo, William Ma, Lara J. Martin, and Mark O. Riedl. Story Realization: Expanding Plot Events into Sentences. In *AAAI Conference on Artificial Intelligence*, page 8, 2020.
- [Bosselut *et al.*, 2019] Antoine Bosselut, Hannah Rashkin, Maarten Sap, Chaitanya Malaviya, Asli Celikyilmaz, and Yejin Choi. Comet: Commonsense transformers for automatic knowledge graph construction. *arXiv preprint arXiv:1906.05317*, 2019.
- [Cardona-Rivera *et al.*, 2016] Rogelio E. Cardona-Rivera, Thomas W. Price, David R. Winer, and R. Michael Young. Question answering in the context of stories generated by computers. *Advances in Cognitive Systems*, 4:227–245, 2016.
- [Carney, 2019] James Carney. *Necessary Fictions: Supernormal Cues, Complex Cognition, and the Nature of Fictional Narrative*, pages 391–413. University of Nebraska Press, 2019.
- [Fan *et al.*, 2018] Angela Fan, Mike Lewis, and Yann Dauphin. Hierarchical neural story generation. *arXiv preprint arXiv:1805.04833*, 2018.
- [Fan *et al.*, 2019] Angela Fan, Yacine Jernite, Ethan Perez, David Grangier, Jason Weston, and Michael Auli. Eli5: Long form question answering. *arXiv preprint arXiv:1907.09190*, 2019.
- [Gabriel *et al.*, 2020] Saadia Gabriel, Chandra Bhagavatula, Vered Shwartz, Ronan Le Bras, Maxwell Forbes, and Yejin Choi. Paragraph-level commonsense transformers with recurrent memory. *arXiv preprint arXiv:2010.01486*, 2020.
- [Gervás, 2009] P. Gervás. Computational approaches to storytelling and creativity. *AI Mag.*, 30:49–62, 2009.
- [Graesser *et al.*, 1991] Art Graesser, Kathy L. Lang, and Richard M. Roberts. Question answering in the context of stories. *Journal of Experimental Psychology: General*, 120(3):254–277, 1991.
- [Khalifa *et al.*, 2017] Ahmed Khalifa, Gabriella AB Barros, and Julian Togelius. Deeptingle. *arXiv preprint arXiv:1705.03557*, 2017.
- [Lewis *et al.*, 2019] Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv preprint arXiv:1910.13461*, 2019.
- [Li *et al.*, 2012] Boyang Li, Stephen Lee-Urban, D. Scott Appling, and Mark O. Riedl. Crowdsourcing narrative intelligence. *Advances in Cognitive Systems*, 2:25–42, December 2012.
- [Liu *et al.*, 2019] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019.
- [Martin *et al.*, 2018] Lara Martin, Prithviraj Ammanabrolu, Xinyu Wang, William Hancock, Shruti Singh, Brent Harrison, and Mark Riedl. Event representations for automated story generation with deep neural nets. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018.
- [Mostafazadeh *et al.*, 2016] Nasrin Mostafazadeh, Nathanael Chambers, Xiaodong He, Devi Parikh, Dhruv Batra, Lucy Vanderwende, Pushmeet Kohli, and James F. Allen. A corpus and evaluation framework for deeper understanding of commonsense stories. *CoRR*, abs/1604.01696, 2016.
- [O’Brien and Albrecht, 1992] Edward J O’Brien and Jason E Albrecht. Comprehension strategies in the development of a mental model. *Journal of experimental psychology: learning, memory, and cognition*, 18(4):777, 1992.
- [Purdy *et al.*, 2018] Christopher Purdy, Xinyu Wang, Larry He, and Mark Riedl. Predicting generated story quality with quantitative measures. In *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*, volume 14, 2018.
- [Radford *et al.*, 2019] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.
- [Rashkin *et al.*, 2020] Hannah Rashkin, Asli Celikyilmaz, Yejin Choi, and Jianfeng Gao. Plotmachines: Outline-conditioned generation with dynamic plot state tracking. *arXiv preprint arXiv:2004.14967*, 2020.
- [Riedl and Young, 2010] Mark O Riedl and Robert Michael Young. Narrative planning: Balancing plot and character. *Journal of Artificial Intelligence Research*, 39:217–268, 2010.
- [Roemmele, 2016] Melissa Roemmele. Writing stories with help from recurrent neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2016.
- [Tambwekar *et al.*, 2018] Pradyumna Tambwekar, Murtaza Dhuliawala, Lara J Martin, Animesh Mehta, Brent Harrison, and Mark O Riedl. Controllable neural story plot generation via reinforcement learning. *arXiv preprint arXiv:1809.10736*, 2018.
- [Trabasso and Van Den Broek, 1985] Tom Trabasso and Paul Van Den Broek. Causal thinking and the representation of narrative events. *Journal of memory and language*, 24(5):612–630, 1985.
- [Ware and Young, 2010] Stephen G Ware and Robert Michael Young. Modeling narrative conflict to generate interesting stories. In *AIIDE*, 2010.

[Yao *et al.*, 2019] Lili Yao, Nanyun Peng, Ralph Weischedel, Kevin Knight, Dongyan Zhao, and Rui Yan. Plan-and-write: Towards better automatic storytelling. In *Proceedings of the AAAI Conference on AI*, 2019.