Question Generation for Reading Comprehension Assessment by Modeling How and What to Ask

Anonymous ACL submission

Abstract

Reading is integral to everyday life, and yet learning to read is a struggle for many young learners. During lessons, teachers can use comprehension questions to increase engagement, test reading skills, and to improve retention. Historically such questions were written by skilled teachers, but recently language models have been used to generate comprehension questions. However, many existing Question Generation (QG) systems focus on generating *extractive* questions from the text, and and have no way to control the type of the generated question. In this paper, we study QG for reading comprehension where *inferential* questions are critical and extractive techniques cannot be used. We propose a two-step model (HTA-WTA) that takes advantage of previous datasets, and can generate questions for a specific targeted comprehension skill. We propose a new reading comprehension dataset that contains questions annotated with story-based reading comprehension skills (SBRCS), allowing for a more complete reader assessment. Across several experiments, our results show that HTA-WTA outperforms multiple strong baselines on this new dataset. We show that the HTA-WTA model tests for strong SCRS by asking deep inferential questions.

1 Introduction

Reading is an invaluable skill, and is core to communicating in our digital age. Reading also supports other forms of development; when children read, it sharpens their memory, and improves social skills (Halliday, 1973; Mason, 2017). Yet, statistics show that one out of five children in the U.S. face learning difficulties (Shaywitz, 2005), especially in reading (Cornoldi and Oakhill, 2013). The coronavirus pandemic beginning in 2020 had a huge impact on the early reading skills of many children, and threatens to leave a lasting impact on a whole generation of young readers (Gupta and Jawanda, 2020).

The pandemic forced many children to learn online, putting in sharp relief the need for effective online education platforms. In particular, reading games have become popular, and can help fill the gap when teachers cannot read in person with students. These platforms present students with short passages and associated comprehension questions. These questions are key to assessing a reader's comprehension of a passage, and can also enhance learning (Chua et al., 2017). But, writing diverse and engaging comprehension questions is no trivial task.

Teachers need to generate new comprehension questions whenever they incorporate new text into a curriculum. New text helps to keep material fresh and topical, and can allow teachers to customize lessons to the interests of a particular student cohort. After finding such custom reading material, teachers must write new comprehension questions to evaluate several reading aspects of comprehension (e.g. understanding complex words, recalling events, etc.).

Thus, to improve the educational process, and lighten the load on teachers, we need tools to automate Question Generation (QG): the task of writing questions for a given passage. Generated questions can be either inferential or extractive questions. Extractive questions can be answered using only information stated in the text, whereas inferential questions require additional information or reasoning. Previous work focused on this aspect of the questions in reading comprehension and discarded the comprehension skills (e.g. close reading, predicting, figurative language, etc.).

We take inspiration from continual learning (Parisi et al., 2019), which orders a set of learning tasks to improve model performance. We begin by training a model on the general task of QG (How to ask: HTA), and follow with our task of interest: generating a targeted question of a particular type (What to ask: WTA).

This paper focuses on the generation of questions for story-based reading comprehension skills (SBRCS), which are varied and cover many aspects of reading comprehension. We create a QG dataset for SBRCS[1]. Although our aim in creating this dataset is to enrich educational applications, this dataset can be considered as a source for general QG and question answering (QA) systems in NLP.

Our focus here is to build a question generator without answer supervision as the case in a real-life application, where a story only will be given as input. This is a challenging task, as many different questions can be generated from a story when there is no answer supervision. QG with answer supervision is another prevalent research line in the literature (Zhao et al., 2018; Ma et al., 2020; Wang et al., 2020; Chen and Xu, 2021).

The contributions in this work are as follows:

- We build a novel QG dataset for SBRCS. The dataset contains advanced reading comprehension skills extracted from stories.
- We propose a two-steps method to generate skill-related questions from a given story. The method takes advantage of previous datasets to improve generalizability, and then, teaches a model how to ask predefined styles of questions.
- We demonstrate the efficiency of the proposed method after extensive experiments, and we investigate its performance in a few-shot learning setting.

The rest of the paper is structured as follows. In the next section, we present an overview of the literature work. In Section 3, we describe how we built our dataset. Section 4 describes the proposed methodology. The experimental setting is presented in Section 5. The results and the analysis are presented in Section 6. Finally, we draw some conclusions and possible future work for this study.

2 Related Works

QG has progressed rapidly due to new datasets and model improvements. Many different QG models have been proposed, starting for simple vanilla Sequence to Sequence Neural Networks models (seq2seq) (Du et al., 2017; Zhou et al., 2017; Yuan et al., 2017) to the more recent transformer-based models (Dong et al., 2019; Chan and Fan, 2019; Varanasi et al., 2020; Narayan et al., 2020; Bao et al., 2020). Some QG systems use manual linguistic features in their models (Harrison and Walker, 2018; Khullar et al., 2018; Liu et al., 2019a; Dhole and Manning, 2020), some consider how to select question-worthy content (Du and Cardie, 2017; Li et al., 2019; Scialom et al., 2019; Liu et al., 2020), and some systems explicitly model question types (Duan et al., 2017; Sun et al., 2018; Kang et al., 2019; Zhou et al., 2019). The last group focused only on generating questions that start with specific interrogative words (what, how, etc.).

QG has been used to solve many real-life problems. For example, QG in conversational dialogue (Gu et al., 2021; Shen et al., 2021; Liu et al., 2021b) where models were taught to ask a series of coherent questions grounded in a QA style, QG based on visual input (Mostafazadeh et al., 2016; Shin et al., 2018; Shukla et al., 2019), and QG for deep questions such as mathematical, curiosity-driven, clinical, and examination-type questions (Liyanage and Ranathunga, 2019; Scialom and Staiano, 2020; Yue et al., 2020; Jia et al., 2021).

3 Data

Despite the recent efforts for building reading comprehension QA datasets, to the best of our knowledge, none of the available datasets explored SBRCS. Questions in previous datasets ask only either inferential or extractive questions from a given passage/story. Rogers et al. (2020), developed questions with general reasoning types based on text from news and blogs. We believe that those texts sources are not rich enough to examine reasoning skills. Advanced reasoning skills (e.g. Figurative Language) are usually used in stories to assess comprehension skills. In the following, we will show how we built our dataset. Table 5 gives an overview of the dataset. In Appendix A, Table A.1, we provide further dataset statistics.

3.1 Dataset Design

3.1.1 Stories Collection

Our stories (passages) are multi-genre, self-contained narratives. This content variety leads annotators towards asking non-localized questions

[1] We are working with our industrial partner to publish the dataset once it is completed as we are still working on incorporating more SBRCS. The dataset will be published only for research purposes.

that test for more advanced reading comprehension skills. The stories are generated using several resources: 1) acquired from free public domain content (Gutenberg Project[2]), 2) partnerships with a publishing house (Blue Moon Publishers[3]) and an educational curriculum development foundation (The Reimagined Classroom[4]), and 3) authored by two professional writers, (the majority of the stories are from this last category). To provide good lexical coverage and diverse stories, we choose to write and collect stories that come from a varied set of genres (e.g. science, social studies, fantasy, fairy tale, historical fiction, horror, mystery, adventure, etc.). In total, we collect 726 multi-domain stories. The stories' lengths range from a single sentence to 113 sentences.

3.1.2 Questions and Comprehension Skills

Previous comprehension question datasets focused on either inferential or extractive (literal) questions. Although these questions assess comprehension skills, they do not provide fine-grained evaluation of the reader comprehension. Thus, to build a more comprehensive list of question types, we started by reviewing curriculum documents available from Columbia University Teacher's College Readers[5] and Writers Workshop Program[6]. Then, we compiled a list of SBRCS, which we then expanded to include additional skills based on school teachers' recommendations. Our final list contains the following skills:

1. **Basic Story Elements**: Can the reader identify the story's main characters and setting?

2. **Character Traits**: Can the reader identify the traits attributable to certain characters in the story (e.g. character feelings, physical attributes)?

3. **Close Reading**: Can the reader extract the text span in a story where the author best describes or explains a key point?

4. **Figurative Language**: Is the reader able to recognize the implied meaning of a sentence?

5. **Inferring**: Can the reader infer what happened in between scenes if the time in-between is not explicitly described?

6. **Predicting**: Can the reader find textual clues and use them to guess what would happen next?

7. **Summarizing**: Is the reader able to recognize the main literary elements of the characters, the events, the problem, and the solutions?

8. **Visualizing**: Can the reader visualize scenes in her/his head to fully comprehend the story?

9. **Vocabulary**: Can the reader identify the right meaning of a word within a context when the word has multiple possible definitions?

Note that some of these SBRCS are prerequisites for others. For instance, the predicting skill may depend on the reader's ability to identifying character attributes and to summarize story elements. In Section A.2, we present further details for each skill type.

With our list of SBRCS as a guide, we wrote question-answer pairs for each story. Given the difficulty of the task, we needed a large number of trained content writers to build the required questions. Each written question should fall into one of the mentioned skills, and obviously, should meet the educational goal. For that, a total of 25 professionals contributed to the writing process (18 teachers, 7 graduate students). We chose not to use crowdworkers (e.g. Amazon Mechanical Turk) to ensure high-quality and educationally-appropriate questions. To verify the quality of the generated content, a second team member reviews each question-answer pair before adding them to the dataset. In addition to annotating questions with a skills label, our content writers annotate each question as either *Literal* or *Inferential* question types. This information is important to measure the comprehension performance of the reader on each question type. Overall, we generate 4K question-answer pairs, with an average of 5.5 pairs per story.

3.2 Additional Data

In addition to the collected dataset, we use two well-known datasets, SQuAD and CosmosQA. We choose these two datasets because of their large size, and their focus on literal or inferential questions.

[2]https://www.gutenberg.org/
[3]https://bluemoonpublishers.com/
[4]https://www.reimaginedclassroom.com/
[5]https://www.tc.columbia.edu/curriculum-and-teaching/literacy-specialist/the-reading–writing-project/
[6]https://readingandwritingproject.org/

	Basic Sto..	Character Tr..	Close Rea..	Figurative La..	Inferring	Predicting	Summarizing	Visualizing	Vocabulary
# Stories	269	280	448	219	449	152	360	153	403
# Question–answer pairs	390	415	719	292	695	162	560	163	604
# Literal Questions	274	120	606	108	16	11	464	36	168
# Inferential Questions	115	295	113	148	679	151	96	127	436

Table 1: Collected dataset's statistics. There are 726 stories, which can have questions from multiple skill types (described in Section 3.1).

SQuAD A reading comprehension dataset, consists of questions created by crowdworkers on a set of Wikipedia articles that cover a large set of topics (from musical celebrities to abstract concepts), where the answer to every question is a span from the corresponding reading passage (Rajpurkar et al., 2016). This dataset can be considered as an extractive QA dataset. It is one of the largest QA datasets in the literature. In this work, we use SQuAD 2.0 version with discarding the questions that has no answers. The size of the dataset is 100K paragraph/question/answer triplets.

CosmosQA It is another reading comprehension dataset consisting of 35.6K paragraph/question pairs that require commonsense-based reading comprehension. It is a collection of people's everyday narratives, and it asks questions about the likely causes of events that require reasoning (Huang et al., 2019). We discard questions that have no answers in this dataset, resulting in 28K paragraph/question/answer triplets.

4 Methodology

Given the fact that including more data in a reading comprehension system is important for generalization (Chung et al., 2018; Talmor and Berant, 2019), and given that our created dataset has the SBRCS which are missed in previous datasets, we propose a two-steps method to generate skill-related questions from a given story: HTA followed by WTA. HTA teaches the model the typical format for comprehension questions using large previously released datasets. These previous datasets are not annotated with the question types outlined in Section 3.1, but the HTA phase allows us to take advantage of those datasets. WTA guides the model to generate questions to test the specific comprehension skills enumerated in Section 3.1. Thus, in HTA, we train (fine-tune) a model on large QG datasets, and then, we further train the model to teach the model what to ask (WTA). For the generation model, we use the pre-trained Text-to-Text Transfer Transformer T5 (Raffel et al., 2020), which closely follows the encoder-decoder architecture of the Transformer model (Vaswani et al., 2017). T5 is a SOTA model on multiple tasks, including QA.

4.1 How to Ask (HTA)

Previous works showed that incorporating more data when training a reading comprehension model improves performance and generalizability (Chung et al., 2018; Talmor and Berant, 2019). However, we cannot incorporate previously released datasets with our new one, as they do not include compatible question skills information. However, they do contain many well-formed and topical questions. Thus, we train a T5 model on SQuAD and CosmosQA datasets to teach the model *how* to ask questions.

Previous neural question generation models take the passage as input, along with the answer. However, encoders can pass all of the information in the input to the decoder, occasionally causing the generated question to contain the target answer. Since the majority of the questions in our created dataset are inferential questions, the answers are not explicitly given in the passages (unlike extractive datasets). Thus, we feed the stories to the encoder, but withhold the answers. Unlike previous systems, we then train the model to generate the questions *and answers*. We propose this setting to generate fewer extractive questions. During our experiments, we evaluated the effect of excluding the answers, and we found them useful to the system.

In Figure 1 we show the input-output format of the model. The encoder input is structured as *<STORY_TEXT> </s>*, where *</s>* is the end-of-sentence token. The decoder generates multiple question-answer pairs as *<QUESTION_TOKENS>*$_1$ *<as>* *<ANSWER_TOKENS>*$_1$ *<sp>* ... *<QUESTION_TOKENS>*$_n$

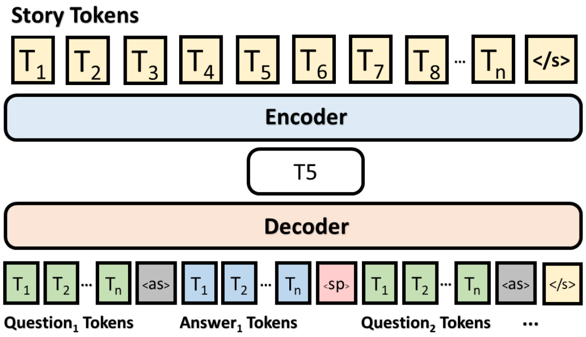

Figure 1: Input and output format of the **How to Ask** (**HTA**) model.

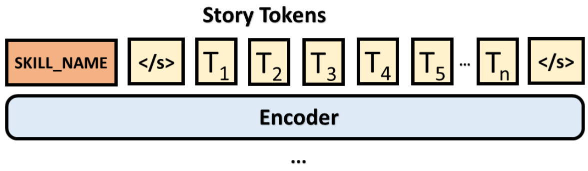

Figure 2: Input format of the **What to Ask** (**WTA**) model. The output format is the same as in HTA model (see Figure 1).

<as> *<ANSWER_TOKENS>$_n$* *</s>*, where *<as>* separates a question from its answer, and *<sp>* separates a question-answer pair from another. The model can generate more than one question-answer pair. We prepare the data to include all of a passage's question-answer pairs in the decoder. Some passages include single question-answer pair, and some passages have up to fifteen pairs.

4.2 What to Ask (WTA)

QG models take a passage/story as input and generate a question. The type of generated question is not controlled and is left for the system to decide it. Thus, the generated question is usually undesired question. Thus, in order to control the style of the generated question, the system needs an indication about the skill that the system is expected to generate a question for. Liu et al. (2020) proposed a way to control the style of the generated questions (e.g. what, how, etc.). The authors built a rule-based information extractor to sample meaningful inputs from a given text, and then learn a joint distribution of <answer, clue, question style> before asking the GPT2 model (Radford et al., 2019) to generate questions. However, this distribution can only be learned using an extractive dataset (e.g. SQuAD); the model cannot learn to generate inferential questions.

To control the skill of the generated question, we use a specific prompt per skill, by defining a special token *<SKILL_NAME>* corresponding to the desired target skill. This helps us to control what to extract from the pretrained model. Thus, the encoder takes as input *<SKILL_NAME>* and *<STORY_TEXT>*, where *<SKILL_NAME>* indicates to the model for which skill the question should be generated (see Figure 2). The data format in the decoder is similar to the one in the HTA step, but here the model generates a single question-answer pair. As a result, the encoding of the *<STORY_TEXT>* will be based on the given *<SKILL_NAME>*. In this way, the model encodes the same story in a different representation when a different *<SKILL_NAME>* is given. A similar technique was used in the literature to include persona profiles in dialogue agents to produce more coherent and meaningful conversations (Scialom et al., 2020).

5 Experiments

5.1 Decoding Method

Decoding strategies are crucial and directly impact output quality. In general, Beam Search (Reddy, 1977) is the most common algorithm, in addition to some other sampling techniques such as Nucleus sampling (Top-p) (Holtzman et al., 2019). In Beam Search, the output of a model is found by maximizing the model probability. On the other hand, Nucleus sampling selects the smallest possible set of tokens whose cumulative probability exceeds the probability p. Experimentally, we found that using the top-p (p=0.9) algorithm yields the best results in terms of the used scoring metrics, thus we use it in all of our experiments.

5.2 Evaluation Metrics

QG often uses standard evaluation metrics from text summarization and machine translation (BLEU (Papineni et al., 2002), ROUGE, METEOR, etc.). However, such metrics do not provide an accurate evaluation for QG task (Novikova et al., 2017), especially when the input passage is long (and many acceptable questions that differ from the gold question can be generated). Thus, to alleviate shortcomings associated with n-gram based similarity metrics, we use BLEURT (Sellam et al., 2020), which is state-of-the-art evaluation metric in WMT Metrics shared task[7]. BLEURT is a BERT-based

[7]BLEURT trained on WMT data, so the output of the metric is in between **-2 to 1**, from worst to the best. The

model that uses multi-task learning to evaluate a generated text by giving it a value in between **-2 to 1**. In our experiments, we consider BLEURT as the main metric for the evaluation. We also report standard MT metric BLEU (1-4 ngrams), and perform an additional manual evaluation.

Manual evaluation is required in our collected dataset, because teachers wrote a single question per skill for a given story, where the model might generate other possible questions for the same skill.

5.3 Implementation Details

We fine-tune the T5 model (base) using the Adam optimizer with a batch size of 8 and a learning rate of $1e-4$. We use a maximum sequence length of 512 for the encoder, and 128 for the decoder[8]. We tested the T5-large model, but we did not notice any improvements considering BLEURT metric. We train all models for a maximum of ten epochs with an early stopping value of 1 (patience) based on the validation loss. We use a single NVIDIA TITAN RTX with 24G RAM.

For HTA, we validate on a combined version of the validation sets from both datasets (SQuAD and CosmosQA). Regarding the collected dataset validation set, we use stratified sampling: we took a random 10% of stories from each skill since the dataset is unbalanced. We apply the same strategy with the test set but with a value of 20%.

5.4 Baselines

To evaluate the performance of our model, we use a set of models that showed state-of-the-art results on several datasets. We obtain the results of those models by running their published GitHub code on our collected dataset. For all of the following baselines, we use SQuAD, CosmosQA, and the collected dataset for training and we test on the test part of the collected dataset:

- Vanilla Seq2seq (Sutskever et al., 2014): a basic encoder-decoder sequence learning system for machine translation. This model takes the story as input and generates a question.

- NQG-Seq (Du et al., 2017): another Seq2seq that implements an attention layer on top of a bidirectional-LSTM encoder. The authors use two encoders, one to encode the sentence that has the answer, and another to encode the whole document. The model then is trained to generate questions.

- NQG-Max (Zhao et al., 2018)[9]: a QG system with a maxout pointer mechanism and gated self-attention LSTM-based encoder to address the challenges of processing long text input. This model takes a passage and an answer as input and generate a question. The answer must be a sub span of the passage.

- CGC-QG (Liu et al., 2019a): a Clue Guided Copy network for Question Generation, which is a sequence-to-sequence generative model with a copying mechanism that takes a passage and an answer (as a span in the text) and generate the question. The text representation in the encoder (GRU network) is represented using a variety of features such as GloVe vectors, POS information, answer position, clue word, etc.

- AnswerQuest (Roemmele et al., 2021): a pipeline model that uses as a first step a previous model (Yang et al., 2019) to retrieve the relevant sentence that has the answer from a document. And then, the sentence is fed to a transformer-based sequence-to-sequence model that is enhanced with a copy mechanism.

- One-Step: a baseline that uses T5 model trained with all data in one step instead of having separate HTA and WTA steps. Because there is only a single step, the skill name is not included in the encoder's input.

- T5-WTA: the WTA model trained using T5 model as a seed model. The HTA training step is not used here. We use this baseline to evaluate the effect of training WTA using HTA.

For all of the previous baselines that require the answer to be a sub-span in the passage, we use the semantic text similarity method that was proposed in (Ghanem et al., 2019) to retrieve the most similar span in the passage. The method extracts several ngrams features from a claim and text spans, and then compute cosine similarity to get the most similar span. In this work, we replace the ngrams features of a text with embeddings extracted from

authors advised not to scale those values to be in percentage.

[8] We were restricted to this length due to memory shortage.

[9] We used the unofficial implementation in this GitHub repo: https://github.com/seanie12/neural-question-generation

RoBERTa model (Liu et al., 2019b). This process has been done on the inferential questions as their answers are not clearly given in the text.

6 Results and Analysis

Table 2 presents the results of the proposed *HTA-WTA* method with the baselines. We can see that out of the baselines, *T5-WTA* performs best in terms of BLEURT score (-1.17), followed by *QG-Max* with a value of -1.18. Given its high BLEURT score, it is surprising that *T5-WTA* model has low BLEU-4. This implies that the generated questions use rich vocabulary, making them different from the gold in terms of overlapping ngrams, but semantically similar leading to higher BLEURT score. As shown in the table, *HTA-WTA*'s BLEURT score outperforms all of the previous QG models by a noticeable margin, showing that including the skill name information plays an important role in generating the intended questions. Also, training on more QG datasets improves the performance.

Regarding the generated questions type, in Table 3 we show the performance of the T5-based models per question type (inferential and literal). Though *One-Step* and *HTA-WTA* models were trained on the same amount of data, the results show that *HTA-WTA* model clearly performs better than the *One-Step* model, especially on inferential questions. We see a similar scenario when comparing *One-Step* and *T5-WTA* models, yet, the gap is smaller. In general, we can notice that the performance gaps for the inferential questions are larger than the literal ones. Thus, we can conclude that *HTA-WTA* is generating more correct inferential questions, which is challenging. This experiment concludes that transformers-based models are capable of asking questions beyond the literal meaning of the text. This confirms what was shown by Liu et al. (2021a) regarding the skills that language models can acquire. Additionally, as some training questions directly quote text from the given story. The T5 model was able to learn how to quote the proper segment of the passage when generating questions.

The *One-Step* model performs similarly to the baselines, although it has been trained using the T5 model and on all three datasets. This may be due to the fact that we did not include the skill name in the encoder, which guides the model to generate skill related questions. To better understand the differences between the outputs of *One-Step* and *HTA-WTA* models, we used human evaluation. This evaluation is to assess the quality of the generated question in terms of *1) Answerability (Ay), 2) Fluency (Fy), and 3) Grammaticality (Gy)* categories, following Harrison and Walker (2018); Azevedo et al. (2020). We include these three criteria as questions may have high *Fluency* and *Grammaticality* scores, but not be answerable.

We select a sample of 110 story-question pairs from the test dataset, for both models. Then, we perform a human evaluation using crowdworkers on Amazon Mechanical Turk. We use a "master" qualification criteria to restrict the participation of workers in our evaluation study to those who have a high historical HIT accuracy, and workers are required to be located in an English speaking country. Workers received $0.41 USD for completing each HIT. Each HIT was answered by three workers. Each worker needs reads the story, and provides ratings (1-5, low to high) for the generated questions, and the three criteria. Table 4 shows the average rating assigned by the workers for the 3 criteria.

Originally, we hypothesized that adding the skill name to the input would force the model to formulate a specific SBRCS question, even if it is not applicable to the current passage. Omitting the skill name may allow the model score high values as it has been left to decide the question. The results show that both models are similar in terms of the given categories, except that *HTA-WTA* performs slightly better in all of the three categories. However, these results refute our claim and show that adding the skill information makes the model generates slightly better questions in terms of quality.

6.1 Impact of Skill Name Token

In order to quantify the impact of skill name in the input, we do another human manual evaluation to measure how accurate both models are when it comes to the generated question skill. Thus, we ask two professional persons who were involved in the annotation process to assign skill names to the generated questions of both *One-Step* and *HTA-WTA* models. We use the same question sample that was used in the previous human evaluation experiment. Few annotation conflicts were found and were solved after a discussion. We evaluate the results using accuracy (see Table 4). The result for *One-Step* model is 0.16, and 0.8 for *HTA-WTA* model. We can clearly see a large gap in accuracy between both models, and this becomes clear with

Model	BLEU-1	BLEU-2	BLEU-3	BLEU-4	BLEURT
Vanilla Seq2seq	17.16	7.78	4.28	2.37	−1.42
NQG-Seq	18.85	8.31	4.37	2.49	−1.37
QG-Max	19.27	7.17	4.12	2.77	−1.18
CGC-QG	**23.93**	12.01	7.82	5.68	−1.29
AnswerQuest	20.44	9.08	4.53	4.71	−1.31
One-Step	15.19	8.05	4.76	2.94	−1.26
T5-WTA	18.53	9.98	6.06	3.92	−1.17
HTA-WTA	22.15	**14.29**	**10.19**	**7.67**	**-1.1**

Table 2: Models' performances on the collected dataset. For all scores, higher is better.

Model	Inferential	Literal
One-Step	-1.28	-1.24
T5-WTA	-1.15	-1.19
HTA-WTA	-1.06	-1.16

Table 3: T5-based models' performances on each question type using BLEURT metric.

Model	Ay	Fy	Gy	Skills Accuracy
One-Step	3.82	4.28	4.37	0.16
HTA-WTA	3.89	4.29	4.45	0.8

Table 4: Human evaluation ratings for our 3 criteria, on a scale 1-5.

the skills that have a low number of instances in the dataset (e.g. Figurative Language, Precision, etc.). Table 6 in Appendix A.3 presents the F1 scores per skill name. We also notice that *HTA-WTA* model performed perfectly on the given sample of *Predicting* and *Figurative Language* (F1 is 1.0 for each skill). This is an interesting result given that the type of the questions for both skills is inferential, which is harder to generate compared to the extractive questions.

6.2 Few-Shot Generation

The process of manually writing questions to assess humans SBRCS is difficult. In some stories, professional writers find obstacles in writing questions for some skills as those skills require high attention and advanced reasoning skills to be written. We can see that in our own dataset, as some skills have fewer questions (e.g. Predicting, Visualizing, etc.). Thus, in this experiment, we evaluate the performance of *HTA-WTA* model when we inject a low percentage of the skills' instances into the training set. This experiment will simulate the case when training a model on a dataset that contains few skills' instances. We use the stratified sampling technique when sampling fewer instances from the collected dataset. Figure 3 shows that injecting only 10% of the data led to a boost in performance of 0.22 (BLEURT). The result at 10% (-1.17) exceeds the

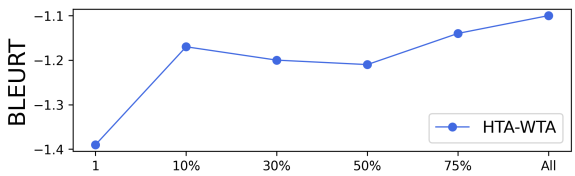

Figure 3: Few-shot performance in BLEURT of the *HTA-WTA* model over a percentage of added few-shot samples. 1 means single instance per skill (9 instances).

results of most of the baselines and is similar to *T5-WTA* and *QG-MAX* models when trained on all the datasets (see Table 2). In Table A.4 in the appendix, we present the results considering other models and metrics. In most cases, the performance gradually improves as data grows. We notice a small drop when we move from 10% to 30%. This behaviour was previously reported by Stappen et al. (2020). Further research is needed to investigate the causes of this behaviour.

7 Conclusion and Future Work

In this paper, we presented a new reading comprehension dataset to assess reading skills using stories. Unlike previous datasets that focused on either inferential or extractive questions, our dataset has nine different SBRCS, each contains inferential and extractive questions. In addition to that, we proposed *HTA-WTA* model which uses two-steps fine-tuning processes to take advantage of previous datasets which have different question formats, and to learn how to ask skill-related questions. We evaluated the model on the collected dataset and compared it to several strong baselines. Our extensive experiments showed the effectiveness of the model. Additionally, *HTA-WTA* is able to generate high quality questions when only 10% of the dataset is used (∼240 instances). In future work, we plan to extend our dataset with additional skills, and to investigate how our model can be integrated into online educational platforms.

References

Pedro Azevedo, Bernardo Leite, Henrique Lopes Cardoso, Daniel Castro Silva, and Luís Paulo Reis. 2020. Exploring NLP and Information Extraction to Jointly Address Question Generation and Answering. In *IFIP International Conference on Artificial Intelligence Applications and Innovations*, pages 396–407. Springer.

Hangbo Bao, Li Dong, Furu Wei, Wenhui Wang, Nan Yang, Xiaodong Liu, Yu Wang, Jianfeng Gao, Songhao Piao, Ming Zhou, et al. 2020. Unilmv2: Pseudo-Masked Language Models for Unified Language Model Pre-training. In *International Conference on Machine Learning*, pages 642–652. PMLR.

Ying-Hong Chan and Yao-Chung Fan. 2019. A Recurrent BERT-based Model for Question Generation. In *Proceedings of the 2nd Workshop on Machine Reading for Question Answering*, pages 154–162.

Xu Chen and Jungang Xu. 2021. An Answer Driven Model For Paragraph-level Question Generation. In *2021 International Joint Conference on Neural Networks (IJCNN)*, pages 1–7. IEEE.

Bee Leng Chua, Oon-Seng Tan, and Paulina Sock Wah Chng. 2017. Mediated Learning Experience: Questions to Enhance Cognitive Development of Young Children. *Journal of Cognitive Education and Psychology*, 16(2):178–192.

Yu-An Chung, Hung-Yi Lee, and James Glass. 2018. Supervised and Unsupervised Transfer Learning for Question Answering. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1585–1594.

Cesare Cornoldi and Jane V Oakhill. 2013. *Reading Comprehension Difficulties: Processes and Intervention*. Routledge.

Kaustubh Dhole and Christopher D Manning. 2020. Syn-QG: Syntactic and Shallow Semantic Rules for Question Generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 752–765.

Li Dong, Nan Yang, Wenhui Wang, Furu Wei, Xiaodong Liu, Yu Wang, Jianfeng Gao, Ming Zhou, and Hsiao-Wuen Hon. 2019. Unified Language Model Pre-training for Natural Language Understanding and Generation. *Advances in Neural Information Processing Systems*, 32:13063–13075.

Xinya Du and Claire Cardie. 2017. Identifying Where to Focus in Reading Comprehension for Neural Question Generation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2067–2073.

Xinya Du, Junru Shao, and Claire Cardie. 2017. Learning to Ask: Neural Question Generation for Reading Comprehension. In *Association for Computational Linguistics*.

Nan Duan, Duyu Tang, Peng Chen, and Ming Zhou. 2017. Question Generation for Question Answering. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 866–874.

Bilal Ghanem, Goran Glavaš, Anastasia Giachanou, Simone Paolo Ponzetto, Paolo Rosso, and Francisco Rangel. 2019. UPV-UMA at CheckThat! Lab: Verifying Arabic Claims using a Cross Lingual Approach. In *CEUR Workshop Proceedings*, volume 2380, pages 1–10.

Jing Gu, Mostafa Mirshekari, Zhou Yu, and Aaron Sisto. 2021. ChainCQG: Flow-Aware Conversational Question Generation. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 2061–2070.

Sonia Gupta and Manveen Kaur Jawanda. 2020. The Impacts of COVID-19 on Children. *Acta Paediatr*, 109(11):2181–2183.

Michael Alexander Kirkwood Halliday. 1973. Explorations in the Functions of Language. *Canadian Journal of Linguistics*.

Vrindavan Harrison and Marilyn Walker. 2018. Neural Generation of Diverse Questions using Answer Focus, Contextual and Linguistic Features. In *Proceedings of the 11th International Conference on Natural Language Generation*, pages 296–306.

Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. 2019. The Curious Case of Neural Text Degeneration. In *International Conference on Learning Representations*.

Lifu Huang, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2019. Cosmos QA: Machine Reading Comprehension with Contextual Commonsense Reasoning. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, pages 2391–2401.

Xin Jia, Wenjie Zhou, Xu Sun, and Yunfang Wu. 2021. EQG-RACE: Examination-Type Question Generation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 14, pages 13143–13151.

Junmo Kang, Haritz Puerto San Roman, and Sung-Hyon Myaeng. 2019. Let Me Know What to Ask: Interrogative-Word-Aware Question Generation. In *Proceedings of the 2nd Workshop on Machine Reading for Question Answering*, pages 163–171.

Payal Khullar, Konigari Rachna, Mukul Hase, and Manish Shrivastava. 2018. Automatic Question Generation using Relative Pronouns and Adverbs. In *Proceedings of ACL 2018, Student Research Workshop*, pages 153–158.

Jingjing Li, Yifan Gao, Lidong Bing, Irwin King, and Michael R Lyu. 2019. Improving Question Generation With to the Point Context. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, pages 3216–3226.

Bang Liu, Haojie Wei, Di Niu, Haolan Chen, and Yancheng He. 2020. Asking Questions the Human Way: Scalable Question-Answer Generation from Text Corpus. In *Proceedings of The Web Conference 2020*, pages 2032–2043.

Bang Liu, Mingjun Zhao, Di Niu, Kunfeng Lai, Yancheng He, Haojie Wei, and Yu Xu. 2019a. Learning to Generate Questions by Learning What not to Generate. In *The World Wide Web Conference*, pages 1106–1118.

Leo Z Liu, Yizhong Wang, Jungo Kasai, Hannaneh Hajishirzi, and Noah A Smith. 2021a. Probing Across Time: What Does RoBERTa Know and When? *arXiv preprint arXiv:2104.07885*.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019b. RoBERTa: A Robustly Optimized BERT Pretraining Approach. *CoRR*, abs/1907.11692.

Zhongkun Liu, Pengjie Ren, Zhumin Chen, Zhaochun Ren, Maarten de Rijke, and Ming Zhou. 2021b. Learning to Ask Conversational Questions by Optimizing Levenshtein Distance. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 5638–5650.

Vijini Liyanage and Surangika Ranathunga. 2019. A Multi-Language Platform for Generating Algebraic Mathematical Word Problems. In *2019 14th Conference on Industrial and Information Systems (ICIIS)*, pages 332–337. IEEE.

Xiyao Ma, Qile Zhu, Yanlin Zhou, and Xiaolin Li. 2020. Improving Question Generation with Sentence-Level Semantic Matching and Answer Position Inferring. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 05, pages 8464–8471.

Jana M Mason. 2017. *Reading Stories to Preliterate Children: A Proposed Connection to Reading*. Routledge.

Nasrin Mostafazadeh, Ishan Misra, Jacob Devlin, Margaret Mitchell, Xiaodong He, and Lucy Vanderwende. 2016. Generating Natural Questions About an Image. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1802–1813.

Shashi Narayan, Gonçalo Simoes, Ji Ma, Hannah Craighead, and Ryan Mcdonald. 2020. QURIOUS: Question Generation Pretraining for Text Generation. *arXiv preprint arXiv:2004.11026*.

Jekaterina Novikova, Ondřej Dušek, Amanda Cercas Curry, and Verena Rieser. 2017. Why We Need New Evaluation Metrics for NLG. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2241–2252.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a Method for Automatic Evaluation of Machine Translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318.

German I Parisi, Ronald Kemker, Jose L Part, Christopher Kanan, and Stefan Wermter. 2019. Continual Lifelong Learning with Neural Networks: A Review. *Neural Networks*, 113:54–71.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language Models are Unsupervised Multitask Learners. *OpenAI blog*, 1(8):9.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. *Journal of Machine Learning Research*, 21:1–67.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ Questions for Machine Comprehension of Text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392.

Raj Reddy. 1977. Speech Understanding Systems: Summary of Results of the Five-Year Research Effort at Carnegie Mellon University.

Melissa Roemmele, Deep Sidhpura, Steve DeNeefe, and Ling Tsou. 2021. AnswerQuest: A System for Generating Question-Answer Items from Multi-Paragraph Documents. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations*, pages 40–52.

Anna Rogers, Olga Kovaleva, Matthew Downey, and Anna Rumshisky. 2020. Getting Closer to AI Complete Question Answering: A Set of Prerequisite Real Tasks. In *Proceedings of the AAAI conference on artificial intelligence*, 05, pages 8722–8731.

Thomas Scialom, Benjamin Piwowarski, and Jacopo Staiano. 2019. Self-Attention Architectures for Answer-Agnostic Neural Question Generation. In *Proceedings of the 57th annual meeting of the Association for Computational Linguistics*, pages 6027–6032.

Thomas Scialom and Jacopo Staiano. 2020. Ask to Learn: A Study on Curiosity-driven Question Generation. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 2224–2235.

Thomas Scialom, Serra Sinem Tekiroğlu, Jacopo Staiano, and Marco Guerini. 2020. Toward Stance-based Personas for Opinionated Dialogues. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings*, pages 2625–2635.

Thibault Sellam, Dipanjan Das, and Ankur Parikh. 2020. BLEURT: Learning Robust Metrics for Text Generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7881–7892.

Sally Shaywitz. 2005. Overcoming Dyslexia: A New and Complete Science-Based Program for Reading Problems at Any Level. *Education Review*.

Lei Shen, Fandong Meng, Jinchao Zhang, Yang Feng, and Jie Zhou. 2021. GTM: A Generative Triple-wise Model for Conversational Question Generation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 3495–3506, Online. Association for Computational Linguistics.

Andrew Shin, Yoshitaka Ushiku, and Tatsuya Harada. 2018. Customized Image Narrative Generation via Interactive Visual Question Generation and Answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8925–8933.

Pushkar Shukla, Carlos Elmadjian, Richika Sharan, Vivek Kulkarni, Matthew Turk, and William Yang Wang. 2019. What Should I Ask? Using Conversationally Informative Rewards for Goal-oriented Visual Dialog. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6442–6451.

Lukas Stappen, Fabian Brunn, and Björn Schuller. 2020. Cross-Lingual Zero-and Few-Shot Hate Speech Detection Utilising Frozen Transformer Language Models and AXEL. *arXiv preprint arXiv:2004.13850*.

Xingwu Sun, Jing Liu, Yajuan Lyu, Wei He, Yanjun Ma, and Shi Wang. 2018. Answer-Focused and Position-Aware Neural Question Generation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3930–3939.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to Sequence Learning with Neural Networks. In *Advances in neural information processing systems*, pages 3104–3112.

Alon Talmor and Jonathan Berant. 2019. MultiQA: An empirical investigation of generalization and transfer in reading comprehension. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4911–4921, Florence, Italy. Association for Computational Linguistics.

Stalin Varanasi, Saadullah Amin, and Günter Neumann. 2020. CopyBERT: A Unified Approach to Question Generation with Self-Attention. In *Proceedings of the 2nd Workshop on Natural Language Processing for Conversational AI*, pages 25–31.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Liuyin Wang, Zihan Xu, Zibo Lin, Haitao Zheng, and Ying Shen. 2020. Answer-driven Deep Question Generation based on Reinforcement Learning. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 5159–5170.

Wei Yang, Yuqing Xie, Aileen Lin, Xingyu Li, Luchen Tan, Kun Xiong, Ming Li, and Jimmy Lin. 2019. End-to-End Open-Domain Question Answering with BERTserini. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, pages 72–77, Minneapolis, Minnesota. Association for Computational Linguistics.

Xingdi Yuan, Tong Wang, Caglar Gulcehre, Alessandro Sordoni, Philip Bachman, Saizheng Zhang, Sandeep Subramanian, and Adam Trischler. 2017. Machine Comprehension by Text-to-Text Neural Question Generation. In *Proceedings of the 2nd Workshop on Representation Learning for NLP*, pages 15–25.

Xiang Yue, Xinliang Frederick Zhang, Ziyu Yao, Simon Lin, and Huan Sun. 2020. CliniQG4QA: Generating Diverse Questions for Domain Adaptation of Clinical Question Answering. *arXiv preprint arXiv:2010.16021*.

Yao Zhao, Xiaochuan Ni, Yuanyuan Ding, and Qifa Ke. 2018. Paragraph-Level Neural Question Generation with Maxout Pointer and Gated Self-Attention Networks. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3901–3910.

Qingyu Zhou, Nan Yang, Furu Wei, Chuanqi Tan, Hangbo Bao, and Ming Zhou. 2017. Neural Question Generation from Text: A Preliminary Study. In *National CCF Conference on Natural Language Processing and Chinese Computing*, pages 662–671. Springer.

Wenjie Zhou, Minghua Zhang, and Yunfang Wu. 2019. Question-Type Driven Question Generation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, pages 6032–6037.

	Basic Sto..	Character Tr..	Close Rea..	Figurative La..	Inferring	Predicting	Summarizing	Visualizing	Vocabulary
# Stories	269	280	448	219	449	152	360	153	403
# Question–answer pairs	390	415	719	292	695	162	560	163	604
Avg. #tok. in stories	168.98	189.62	133.44	137.86	133.63	145.09	192.8	118.61	143.21
Max. #tok. in stories	1159	1159	1159	935	1159	1132	1132	935	1040
Avg. #tok. in questions	9.14	11.82	11.12	16.38	13.21	12.92	9.88	12.98	15.96
Max. #tok. in questions	24	58	55	70	52	76	43	39	49
Avg. #tok. in answers	4.17	3.81	4.49	4.7	6.16	6.48	5.91	5.10	3.46
Max. #tok. in answers	29	34	73	30	29	21	46	40	22
# Literal Questions	274	120	606	108	16	11	464	36	168
# Inferential Questions	115	295	113	148	679	151	96	127	436

Table 5: Collected dataset's statistics. There are 726 stories, which can have questions from multiple skill types.

A Appendix

A.1 Full Dataset Details

A.2 Further Details on Skills

1. **Basic Story Elements**: Determining what are the main story elements is one of the comprehension skills to assess the reader understanding. Using this skill, we can understand whether the reader is able to identify the main characters and environment settings of the stories.

2. **Character Traits**: Identifying permanent traits that can be assigned to characters or describe character development. For instance, knowing what most likely *X* character felt during the story, recognizing facts about *X*, identifying main adjectives that *X* has, etc.

3. **Close Reading**: Identifying the place in a story where the author best describes or explains a key point. Also, it includes questions to identify the purpose of a quote or a sentence. This skill requires advanced reading comprehension ability from the reader since its answers cannot be extracted directly from the story text, where inferential skills are needed.

4. **Figurative Language**: Figurative language is common in stories as it makes ideas and concepts easier to visualize by the reader. Also, it is an effective way of conveying an idea that is not easily understood. With this skill, we examine the reader ability of recognizing the implicated meaning of a sentence or a type of figurative language.

5. **Inferring**: Writers sometimes jump into the action or skip forward in their stories. Good readers must infer what happened in between scenes if the time in-between is not explicitly detailed. In addition, readers must infer their characters' emotions if their characters do not share those aloud.

6. **Predicting**: Predicting involves guessing what will happen next. It is different from inferring; inferring is guessing what is happening now or what happened before. Good readers do not let books passively happen to them, they work to "solve" the story before it reaches its end by finding clues and using them to guess what will happen next or to guess how the conflict will be resolved.

7. **Summarizing**: Consolidating a text into a precise synopsis of only the most key information. Summarizing skill contains the main literary elements of the characters, the problem, and the solutions. Key events from the beginning, middle, and end are included in a summary.

8. **Visualizing**: This skill requires readers to visualize scenes in their heads to fully comprehend the story. It can assess readers ability of imagining specific events or elements in the stories. An example of visualizing questions is: *What do you imagine when reading this sentence "quote"?*

9. **Vocabulary**: Identifying the meaning of unfamiliar words in the text is a key skill for readers to fully comprehend the story. In this skill, the reader should identify the right meaning of a word within a context when the word has multiple possible definitions. Additionally,

the reader should be able to identify vocabulary based questions related to identifying synonyms, antonyms, homophones, compound words, and word types (e.g. noun, verb, etc.).

A.3 Manual Evaluation Results of Questions' Skills

In Table 6, we show the fined-grained results per skill name after the manual labeling experiment for the generated questions from both *One-Step* and *HTA-WTA* models.

A.4 Few-Shot Question Generation Results

In Table 7, we show the few-shot experiment's results considering both scoring metrics (BLEU, and BLUERT). We do not experiment with *One-Step* model as we need to sample SQuAD and CosmosQA datasets when we sample the collected data; it is hard to set up a fair comparison here as, for instance, sampling 10% of SQuAD dataset is larger than the whole collected dataset.

A.5 Samples of the Generated Questions

In this section, we list some random examples from *HTA-WTA* model for inferential questions:

- Story: "The Line 1 Toronto train was a subway like many others you've seen. He rocketed down Yonge Street, around the Union loop, and rattled off towards Vaughn. At Vaughn he'd let out a loud, hissing sigh and a clanking sort of grunt, then reverse and do the whole thing backwards all over again. He liked his transit union job well enough, but he couldn't help thinking about the lights at the end of his tunnels. No matter how long he'd been running, or how much he wished for anything else, that little hopeful point of light always turned out to be just one more dirty subway platform."

  Generated *Figurative Language* question: "Reread this sentence: "He rocketed down Yonge Street, around the Union loop, and rattled off towards Vaughn." Which figurative language technique is being used here?"

  Generated answer: "Alliteration".

- Story: ""The map says left", said Bri. "But my heart says right!" cried Rob. "Is your heart full of hidden treasure?" asked Bri. "Yes." Rob replied. "At least, that's what my mom says.""

  Generated *Inferring* question: "Why do you think Bri's heart says "But my heart says right!"?"

  Generated answer: "Because she thinks she has found something".

- Story: "Mary looked at it, not really knowing why the hole was there, and as she looked she saw something almost buried in the newly-turned soil. It was something like a ring of rusty iron or brass and when the robin flew up into a tree nearby she put out her hand and picked the ring up. It was more than a ring, however; it was an old key which looked as if it had been buried a long time. Mistress Mary stood up and looked at it with an almost frightened face as it hung from her finger. "Perhaps it has been buried for ten years," she said in a whisper. "Perhaps it is the key to the garden!""

  Generated *Vocabulary* question: "Reread this sentence: "Perhaps it has been buried for ten years" What is the correct definition of the word "frightened" as it is used here?"

  Generated answer: "Scared".

Model	Basic Sto..	Character Tr..	Close Rea..	Figurative La..	Inferring	Predicting	Summarizing	Visualizing	Vocabulary
#instances	12	8	23	7	14	6	14	10	16
One-Step	0.13	0.00	0.31	0.00	0.19	0.00	0.07	0.00	0.18
HTA-WTA	0.88	0.93	0.68	1.00	0.69	1.00	0.81	0.18	1.00

Table 6: F1 score results per skill name.

Instances Ratio	Model	BLEU-1	BLEU-2	BLEU-3	BLEU-4	BLEURT
1	T5-WTA	8.61	3.38	1.71	1.04	-1.41
1	HTA-WTA	10.2	4.74	2.85	1.96	-1.39
0.1	T5-WTA	14.8	6.68	3.63	2.22	-1.34
0.1	HTA-WTA	16.55	9.54	6.28	4.37	-1.17
0.3	T5-WTA	16.02	8.3	5.07	3.45	-1.3
0.3	HTA-WTA	16.14	9.7	6.64	4.82	-1.20
0.5	T5-WTA	16.32	8.25	4.77	3.00	-1.24
0.5	HTA-WTA	15.48	9.25	6.34	4.61	-1.21
0.75	T5-WTA	18.9	10.12	6.24	4.19	-1.17
0.75	HTA-WTA	18.69	11.53	7.97	5.74	-1.14
All	T5-WTA	18.53	9.99	6.07	3.93	-1.17
All	HTA-WTA	22.15	14.3	10.2	7.67	-1.10

Table 7: Few-shot performance of the *HTA-WTA* and *T5-WTA* models over a percentage of added few-shot samples. 1 means single instance per skill (9 instances).