

On Few-Shot Prompting for Controllable Question-Answer Generation in Narrative Comprehension

Bernardo Leite^{1,2} ^a and Henrique Lopes Cardoso^{1,2} ^b

¹Faculty of Engineering of the University of Porto (FEUP), Portugal

²Artificial Intelligence and Computer Science Laboratory (LIACC), Portugal

Keywords: Controllable Question-Answer Generation, Few-Shot Prompting.

Abstract: Question Generation aims to automatically generate questions based on a given input provided as context. A controllable question generation scheme focuses on generating questions with specific attributes, allowing better control. In this study, we propose a few-shot prompting strategy for controlling the generation of question-answer pairs from children’s narrative texts. We aim to control two attributes: the question’s explicitness and underlying narrative elements. With empirical evaluation, we show the effectiveness of controlling the generation process by employing few-shot prompting side by side with a reference model. Our experiments highlight instances where the few-shot strategy surpasses the reference model, particularly in scenarios such as semantic closeness evaluation and the diversity and coherency of question-answer pairs. However, these improvements are not always statistically significant. The code is publicly available at github.com/bernardoleite/few-shot-prompting-qg-control.

1 INTRODUCTION

The task of Question Generation (QG) involves the automatic generation of well-structured and meaningful questions from diverse data sources, such as free text or knowledge bases (Rus et al., 2008). Controllable Question Generation (CQG) holds significant importance in the educational field (Kurdi et al., 2020), as it boosts the creation of customized questions tailored to student’s specific needs and learning objectives.

From a methodological perspective, some prior works on QG have focused on fine-tuning large pre-trained language models (PLM) (Zhang et al., 2021) for generating questions (output) given a source text and possibly a target answer (input). This is also the case for CQG, with the addition of incorporating controllability labels into the input to serve as guidance attributes during the generation process (Zhao et al., 2022; Ghanem et al., 2022). After undergoing fine-tuning, the models have demonstrated good performance (Ushio et al., 2022). However, utilizing these models demands a custom model design (e.g., architecture, choice of hyperparameters) and substantial

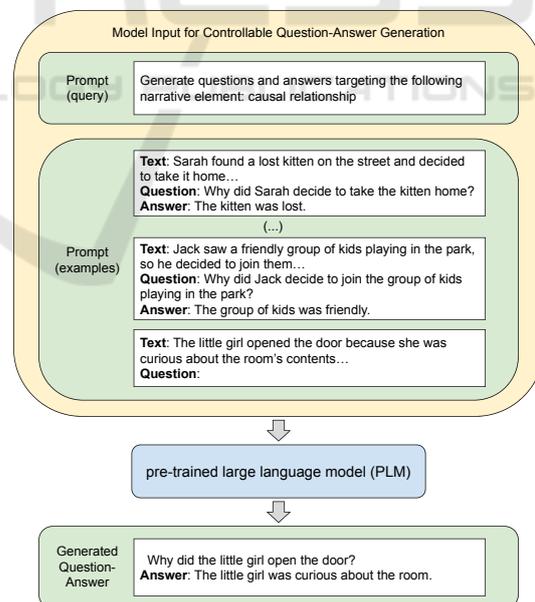


Figure 1: A simplistic example of few-shot prompting for controllable question-answer generation.

computational resources for training, which may deter practitioners who prefer a convenient “plug-and-play” AI-assisted approach, enabling them to effortlessly interact with a QG system without the need

^a  <https://orcid.org/0000-0002-9054-9501>

^b  <https://orcid.org/0000-0003-1252-7515>

to undergo model design and training (Wang et al., 2022).

Motivated by this, we explore a few-shot prompting strategy to address the CQG task in the context of narrative comprehension. We aim to control the generation of question-answer pairs conditioned by their underlying narrative elements (e.g., character, setting, acting) and question explicitness (explicit or implicit). We explore the prompting paradigm, where a prompt specifies the desired generation task. Figure 1 shows an illustrative example of the few-shot prompting strategy for CQG. Prompting offers a straightforward interface and a significant level of control to interact with PLMs and tailor them according to various generation requirements. Its practicality and simplicity have contributed to its popularity as a means of customizing PLMs for diverse tasks (Wang et al., 2022; Liu et al., 2023).

We assess the few-shot strategy's performance alongside a smaller yet reference model, where fine-tuning is applied. To this end, we empirically evaluate the question-answer pairs generated by the two methods using similarity and quality metrics.

Nonetheless, we stress out that our primary goal is not to determine whether the few-shot strategy is better or worse than fine-tuning for question generation control. Such an assessment would necessitate a separate, comprehensive study focused on an in-depth model comparison. Instead, our primary aim is to delve into the potential of a few-shot strategy for controlled question-answer generation. We indeed build and incorporate, for reference, the results obtained by a fine-tuned model based on previous findings (Zhao et al., 2022; Ghanem et al., 2022), known for achieving commendable results in CQG.

In summary, our primary contribution is a few-shot strategy, based on the prompting paradigm, for controlling the generation of question-answer pairs based on their narrative elements, explicitness, or both. While using prompts via few-shot prompting has been explored in previous research, the novelty of our study lies in its focused application on *narrative comprehension* through the controlled generation of *both* questions and answers. As a result, our analysis contributes to the ongoing discourse by providing valuable and unique insights into this unexplored area.

2 BACKGROUND AND RELATED WORK

Few-Shot: This study presents a strategy for CQG based on few-shot prompting. To ensure clarity in

terminology, we provide definitions for few-shot and fine-tuning as elucidated by Brown et al. (2020). Few-Shot pertains to approaches where the model receives limited task demonstrations as conditioning context during inference (Radford et al., 2019), without allowing weight updates. Fine-Tuning refers to the process of adjusting the weights of a pre-trained model by training it on a dataset tailored to a specific task. This involves utilizing a significant number of labeled examples.

Controllable Question Generation (CQG): Prior research has explored CQG for education. Ghanem et al. (2022) employed fine-tuning with the T5 model (Raffel et al., 2020) to control the reading comprehension skills necessary for formulating questions, such as understanding figurative language and summarization. Similarly, Zhao et al. (2022) aimed to control the generated questions' narrative aspects. Still, through fine-tuning, Leite and Lopes Cardoso (2023) propose to control question explicitness using the T5 model. Finally, via few-shot, Elkins et al. (2023) propose to address the task of CQG by controlling three difficulty levels and Bloom's question taxonomy (Krathwohl, 2002) for the domains of machine learning and biology.

In our study, we make use of the FairytaleQA (Xu et al., 2022) dataset, which consists of question-answer pairs extracted from stories suitable for children. This dataset has been investigated by two studies above mentioned (Zhao et al., 2022; Leite and Lopes Cardoso, 2023) for CQG via fine-tuning.

To the best of our knowledge, this is the first study addressing few-shot prompting for controlling the generation of both questions and answers in the narrative comprehension domain.

3 PURPOSE OF QG CONTROL ELEMENTS

We chose FairytaleQA dataset (Xu et al., 2022) because its texts and the corresponding question-answer pairs align with the goal of *supporting narrative comprehension*. As highlighted by Xu et al. (2022), narrative comprehension is a high-level skill that strongly correlates with reading success (Lynch et al., 2008). Additionally, narrative stories possess a well-defined structure comprising distinct elements and their relationships. This dataset stands out since education experts have annotated each question, following evidence-based narrative comprehension frameworks (Paris and Paris, 2003; Alonzo et al., 2009), and addressing two key attributes: *narrative elements* and *explicitness*. Narrative elements we aim to control are

as follows:

- **Character:** These require test takers to identify or describe the characteristics of story characters.
- **Setting:** These inquire about the place or time where story events occur and typically start with “Where” or “When”.
- **Action:** These focus on the behaviors of characters or seek information about their actions.
- **Feeling:** These explore the emotional status or reactions of characters, often framed as “How did/does/do...feel.”
- **Causal Relationship:** These examine the cause-and-effect relationships between two events, often starting with “Why” or “What made/makes”.
- **Outcome Resolution:** These ask for the events that result from prior actions in the story, typically phrased as “What happened/happens/has happened...after...”.
- **Prediction:** These request predictions about the unknown outcome of a particular event based on existing information in the text.

Question explicitness is defined as follows:

- **Explicit:** The answers are directly present in the text and can be located within specific passages.
- **Implicit:** Answers cannot be directly pinpointed in the text, requiring the ability to summarize and make inferences based on implicit information.

4 METHOD

4.1 Few-Shot Prompting for CQG

Let E be an example set containing K text passages and their corresponding question-answer pairs, denoted as $E = \{(x_i, y_i)\}_{i=1}^K$, where x_i represents the text passage and y_i represents the associated question-answer pair.

The few-shot prompting process can be represented as follows: Given a *query*, the example set E and a new text passage x_{new} , the aim is to generate a question-answer pair (q_{new}, a_{new}) . This can be formulated as:

$$(q_{new}, a_{new}) = \text{PLM}(query, E, x_{new}), \quad (1)$$

where PLM represents the pre-trained language model that generates the question-answer pair (q_{new}, a_{new}) based on a *query*, the example set E , and the new text passage x_{new} . *query* is the textual instruction designed to control the generation of question-answer pairs conditioned to the desired attributes. In this study, it can assume four formats¹:

1. No Control (baseline): “Generate questions and answers from text:”
2. Narrative Control: “Generate questions and answers targeting the following narrative element: $\langle \text{NAR} \rangle$.”
3. Explicitness Control: “Generate $\langle \text{EX} \rangle$ questions and answers:”
4. Narrative + Explicitness Control: “Generate $\langle \text{EX} \rangle$ questions and answers targeting the following narrative element: $\langle \text{NAR} \rangle$.”

The concrete textual form of the query has been obtained from preliminary and empirical experimentation.

4.2 Reference Model for CQG

In the case of the reference model, we frame the CQG task as an encoder-decoder model, where the encoder receives the input text and encodes it into a fixed-length representation known as a context vector. The decoder takes the context vector and generates the output text. Here, the control attributes $\langle \text{NAR} \rangle$ or $\langle \text{EX} \rangle$ are added at the start of the input, preceding the section text. Then, the decoder is equipped with labels that serve to differentiate the $\langle \text{QUESTION} \rangle$ and $\langle \text{ANSWER} \rangle$ sections of the output. The idea is to guide the model to generate a question-answer pair of the intended type. Figure 2 illustrates the reference model setup for generating a question-answer pair targeting the *action* narrative element. This technique is based on recent studies (Zhao et al., 2022; Ghanem et al., 2022) aiming at controlling QG conditioned on specific attributes.

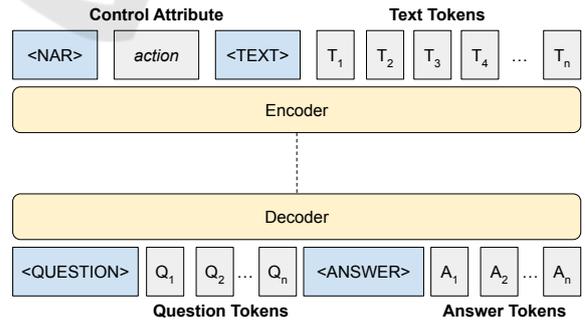


Figure 2: Reference model setup for performing controllable question-answer generation.

¹See Figures 7 and 8 for concrete examples.

5 EXPERIMENTAL SETUP

5.1 FairytaleQA

We make use of FairytaleQA (Xu et al., 2022), a dataset composed of 10,580 question-answer pairs manually created by educational experts based on 278 stories. Every question is associated with one of the following narrative elements: *character*, *setting*, *action*, *feeling*, *causal relationship*, *outcome resolution*, or *prediction*. Additionally, each question is accompanied by an explicitness attribute, denoting whether it is explicit or implicit (recall Section 3 for details). Each story consists of approximately 15 sections, each with an average of 3 questions. We use the original train/validation/test splits², comprising 8,548/1,025/1,007 question-answer pairs, respectively.

5.2 Data Preparation

From the original dataset, we have prepared different data setups³:

- section \rightarrow question + answer: This setup only contains the section text as input, so it serves as a baseline to compare with the subsequent setups, which consider control attributes.
- $\langle \text{EX} \rangle$ + section \rightarrow question + answer: This setup considers *explicitness* as a control attribute in the input.
- $\langle \text{NAR} \rangle$ + section \rightarrow question + answer: This setup considers *narrative* as a control attribute in the input.
- $\langle \text{NAR} \rangle$ + $\langle \text{EX} \rangle$ + section \rightarrow question + answer: This setup considers both the explicitness and narrative attributes.

Fair Comparison: To ensure a fair comparison in our evaluation results (Section 6.2) between these setups, we guarantee that each section text is utilized in a single instance⁴ within the test set. This eliminates the creation of redundant instances, such as the repeated use of the same section text as input.

Selection of Ground Truth Pairs: We ensure that all ground truth question-answer pairs within each instance refer to a single explicitness type and narrative

²https://github.com/WorkInTheDark/FairytaleQA_Dataset/tree/main/FairytaleQA_Dataset/split_for_training

³The arrow separates the input (left) and output (right) information. On the left part, the + symbol illustrates whether the method incorporates control attributes.

⁴A dataset instance consists of a text and corresponding ground truth question-answer pairs.

element. This step is crucial in supporting the rationale behind Hypothesis 1 within Section 6.1 (Evaluation Procedure).

5.3 Implementation Details

For few-shot prompting, we use the *text-davinci-003* model (GPT-3.5) from OpenAI⁵ with 128 as the maximum token output, 0.7 for temperature and 1.0 for nucleus sampling. Following previous recommendations (Wang et al., 2022; Elkins et al., 2023), we choose the 5-shot setting: beyond the query, 5 examples (each composed of text, question and answer) are incorporated into the prompt. The 5 examples have been randomly extracted from the train set based on the following criterion: the selected examples are consistent with the target attribute (either narrative element or explicitness) one aims to control in the generation process. So, this deliberate selection ensures a focus on a specific narrative element or explicitness (which is the goal). In the data setup where the input is just the section text, the selected examples target varied attributes.

For the reference model, we use the pre-trained T5 encoder-decoder model (Raffel et al., 2020). Firstly, T5 was trained with task-specific instructions in the form of prefixes, aligning with our methodology. Secondly, it has remarkable performance in text generation tasks, particularly in Question Generation (Ushio et al., 2022) and CQG (Ghanem et al., 2022; Leite and Lopes Cardoso, 2023). Hence, we designate T5 as a smaller yet established reference model for QG and CQG. We use the *t5-large* version available at Hugging Face⁶. Our maximum token input is set to 512, while the maximum token output is set to 128. During training, the models undergo a maximum of 10 epochs and incorporate early stopping with a patience of 2. Additionally, a batch size of 8 is employed. During inference, we utilize beam search with a beam width of 5.

6 EVALUATION

6.1 Evaluation Procedure

For CQG, our evaluation protocol is based on recent work (Zhao et al., 2022; Leite and Lopes Cardoso, 2023) that performed CQG via fine-tuning. We enrich the evaluation process with metrics related to linguistic quality (Wang et al., 2022; Elkins et al., 2023).

⁵<https://platform.openai.com/docs/models/gpt-3-5>

⁶<https://huggingface.co/t5-large>

Narrative Control: For assessing narrative elements control, we employ the traditional evaluation procedure in QG: directly compare the generated questions (via reference and few-shot) with the ground truth questions. Hypothesis (1) is that *generated questions will be closer to the ground truth when control attributes are incorporated*. This hypothesis gains support from the observation that an increased closeness implies that the generated questions, prompted to match a particular narrative element, exhibit a close alignment with the ground truth questions of the same narrative element. In Figure 6, we present examples of both ground truth and generated questions that motivated this evaluation procedure, noting mainly that the beginnings of the questions are very close. To measure this closeness, we use n -gram similarity BLEU-4 (Papineni et al., 2002) and ROUGE_L-F1 (Lin, 2004). Also, for semantic similarity, we use BLEURT (Sellam et al., 2020). An enhancement in these metrics means that the generated questions, with a specified target narrative type, closely approximate the ground truth questions that share the same narrative type, indicating better controllability.

Explicitness Control: For assessing explicitness control, we resort to creating a question-answering system (QAsys) trained along FairytaleQA⁷. The goal is to put QAsys answering questions that were generated (via reference and few-shot) and then compare QAsys answers with the answers generated (via reference and few-shot). Hypothesis (2) is that *QAsys will perform significantly better on explicit than implicit generated questions*, as previously supported by FairytaleQA’s authors. We also provide this evidence in Appendix A. To measure QAsys performance, we use ROUGE_L-F1 and EXACT MATCH, a stringent scoring approach that considers a perfect match as the only acceptable outcome when comparing two strings.

Linguistic Quality: To evaluate the linguistic quality of the generated questions and answers, we report perplexity, grammatical error, and diversity metrics. For perplexity, our motivation is that previous studies (Wang et al., 2022) claim there is a relation between perplexity and coherence, in a way that perplexity is inversely related to the coherence of the generated text: the lower the perplexity score, the higher the coherence. For the sake of computational efficiency, we use GPT-2 (Radford et al., 2019) to compute perplexity. For diversity, we use Distinct-3 score (Li et al., 2016), which counts the average num-

⁷We created QAsys by fine-tuning a T5 model on FairytaleQA for generating answers given the question and section text.

ber of distinct 3-grams in the generated text. Finally, we use Python Language Tool⁸ to count the number of grammatical errors averaged over all generated questions and answers.

6.2 Results and Discussion

Narrative Control: Table 1 presents the results using traditional QG evaluation, which directly compares the generated questions with the ground truth. Both via reference model and few-shot prompting, a significant growth in closeness to ground truth questions is observed when incorporating narrative control attributes (this happens for all metrics). Thus, we conclude that the few-shot prompting strategy has been successfully applied to control the questions’ underlying narrative elements⁹. Based on the results obtained from different metrics, we believe that the few-shot prompting might be more convenient for generating questions semantically closer to the ground truth, as reflected by the higher performance on the BLEURT metric (improves from .438 to .445). However, this improvement is not statistically significant¹⁰. Also, the relatively lower performance on the BLEU (worsens from .201 to .168) and ROUGE_L-F1 (worsens from .429 to .409) metrics suggests that it may struggle with capturing certain n -gram patterns or surface-level similarities. These findings highlight the importance of considering multiple evaluation metrics to comprehensively understand the strengths and weaknesses of different approaches in CQG (and natural language generation in general).

Explicitness Control: When incorporating explicitness control attributes, as opposed to only providing section text without a specific explicitness attribute, both the reference model and few-shot prompting also result in questions closer to the ground truth¹¹, although less significantly than when considering the narrative attributes. To further evaluate question explicitness control, we need to analyze the question-answering results obtained by the QAsys model (as motivated earlier in Section 6.1). Table 2 presents the question-answering scores of QAsys when attempting to answer generated questions. For the reference model and few-shot prompting, QAsys performs significantly better on explicit than implicit generated questions, considering ROUGE_L-F1 and EXACT MATCH. Therefore, these results show that

⁸https://github.com/jxmorris12/language_tool_python

⁹We provide the control results by narrative element in the Appendix B.

¹⁰We perform student’s t-test and find that $p_1 > .05$.

¹¹This is not true for the reference model, with ROUGE_L-F1.

Table 1: Closeness between generated and ground truth questions on the test set. All scores are 0-1.

	Data Setups	ROUGEL-F1 ↑	BLEU-4 ↑	BLEURT ↑
Reference Model	section → question + answer	0.335	0.137	0.394
	ex + section → question + answer	0.333	0.138	0.398
	nar + section → question + answer	0.429	0.201	0.438
	nar + ex + section → question + answer	0.442	0.198	0.442
Few-Shot Prompting	section → question + answer	0.339	0.108	0.397
	ex + section → question + answer	0.358	0.123	0.411
	nar + section → question + answer	0.409	0.168	0.445
	nar + ex + section → question + answer	0.402	0.177	0.441

Table 2: QAsys performance by question explicitness on the test set. All scores are 0-1.

	Data Setups	ROUGEL-F1 ↑			EXACT-MATCH ↑		
		Overall	Explicit	Implicit	Overall	Explicit	Implicit
Reference Model	ex + section → question + answer	0.661	0.716	0.513	0.371	0.413	0.259
	nar + ex + section → question + answer	0.628	0.681	0.487	0.383	0.434	0.250
Few-Shot Prompting	ex + section → question + answer	0.481	0.531	0.351	0.119	0.143	0.056
	nar + ex + section → question + answer	0.490	0.556	0.315	0.155	0.185	0.074

Table 3: Linguistic quality of generated questions and answers on the test set. Except for perplexity (PPL), all scores are 0-1.

	Data Setups	Generated Questions			Generated Answers		
		PPL ↓	Dist-3 ↑	Gram. ↓	PPL ↓	Dist-3 ↑	Gram. ↓
Reference Model	section → question + answer	197.192	0.776	0.013	303.331	0.668	0.033
	ex + section → question + answer	175.717	0.789	0.005	336.649	0.662	0.028
	nar + section → question + answer	168.303	0.782	0.018	343.050	0.597	0.020
	nar + ex + section → question + answer	183.665	0.789	0.013	352.672	0.560	0.025
Few-Shot Prompting	section → question + answer	166.160	0.787	0.005	248.966	0.725	0.038
	ex + section → question + answer	143.270	0.791	0.013	240.593	0.734	0.036
	nar + section → question + answer	155.761	0.797	0.008	224.536	0.679	0.020
	nar + ex + section → question + answer	153.056	0.790	0.010	260.307	0.671	0.033

controlling question explicitness is possible through the few-shot strategy. It should be noted that QAsys scores are lower when answering questions generated via few-shot prompting. We strongly believe this happens because QAsys is a question-answering model trained along FairytaleQA using the T5 model, just like the reference fine-tuned models to control generation. Thus, QAsys has greater ease in answering questions generated by the reference fine-tuned models. The situation is reversed if we employ the GPT-3.5 model for QAsys (see Appendix C).

Narrative + Explicitness Control: By looking at the scores in Tables 1 and 2, when incorporating both narrative and explicitness attributes, we verify the same trend as when incorporating the attributes individually. This is true for the reference model and the few-shot prompting strategy. Therefore, using the proposed scheme, it is possible to control the generation process in few-shot prompting. While there are cases where incorporating both attributes improves the results, there are also cases where that is not the case. So, we do not find clear evidence that using multiple control attributes helps or worsens the process of controlling the generation. In Appendix D, we show the impact of varying the number of prompt examples

on the performance of few-shot prompting.

Linguistic Quality: Table 3 reports results for perplexity, diversity, and grammatical error to provide insight into the linguistic quality of generated questions and answers. Regarding perplexity, the few-shot strategy presents questions and answers with a lower perplexity value, indicating higher coherence. Regarding diversity (Dist-3), the few-shot strategy presents questions with a higher diversity value than the reference model. Again, we find that the difference is not consistently statistically significant. For the answers, in contrast, we confirm that the difference is indeed consistently statistically significant. Finally, both methods yield an average value of zero for grammatical errors in questions and answers¹². Overall, the linguistic results indicate that besides showing competence for CQG in narrative comprehension, the few-shot strategy can deliver coherent and diverse questions and answers, which motivates their use in an educational context.

¹²We did not consider MORFOLOGIK_RULE error from Python Language Tool, which suggests possible spelling mistakes in uncommon nouns, such as Ahtola.

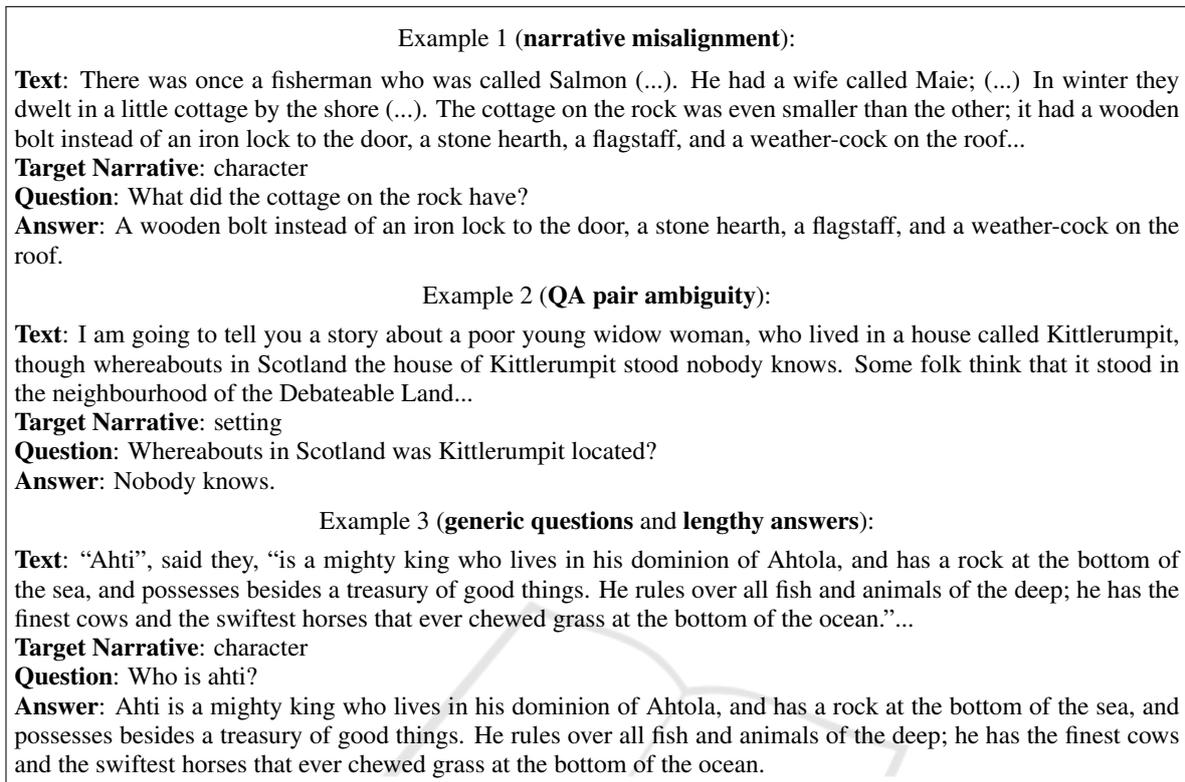


Figure 3: Examples of problematic generated question-answer pairs (error analysis) via few-shot prompting.

6.3 Error Analysis

We randomly selected 105 QA pairs¹³ generated from the FairytaleQA test set and analyzed potential problems. We identify 4 types of issues, which are exemplified in Figure 3. The issues found are (a) *narrative misalignment* (19 in 105), (b) *QA pair ambiguity* (1 in 105), (c) *generic questions* (1 in 105) and (d) *lengthy answers* (4 in 105).

In the first example, the generated question is not aligned with the specified "Character" attribute in the prompt; instead, it focuses on details about the cottage on the rock. The generated answer describes the features of the cottage but does not address the specified character (Salmon/Maie). While we are uncertain about the reasons for this model failure, it reveals the importance of verifying alignment between questions and attributes. An approach could involve implementing a post-model solution to assess alignment.

In the second example, the generated question accurately captures the attribute "Setting" inquiring about the location of Kittlerumpit. While the generated answer appropriately reflects the uncertainty mentioned in the text, it could be perceived as ambiguous due to the general statement ("Nobody

knows"). An improvement could involve instructing the model to formulate more comprehensive questions encouraging answers that explore alternative interpretations or speculations mentioned in the text, such as the belief that Kittlerumpit stood near the Debateable Land.

In the third example, the generated question appears generic and could benefit from increased specificity. Encouraging the model to formulate questions that require detailed information, such as Ahti's dominion, possessions, and influence over the sea, may enhance the precision of the generated answers. Related to this, we find the generated answer overly lengthy. Providing guidance for more concise responses could help retain key details and shorten the answers.

7 CONCLUSION

This work introduces a few-shot prompting strategy to address CQG for narrative comprehension, using the FairytaleQA dataset. Through experimental analysis, we observed that the generated questions, tailored to specific attributes, closely approximate the ground truth questions of the same type. This sug-

¹³ 15 QA pairs for each of the 7 target narrative elements.

gests promising indicators of controllability for narrative elements and explicitness. However, our error analysis revealed instances where control did not occur, underscoring the need for further investigation when employing few-shot prompting with a state-of-the-art model like GPT-3.5. Additionally, our findings demonstrate that the few-shot strategy can outperform the reference model in certain scenarios, however, these improvements are not consistently statistically significant.

Considering our results, which align with those of a smaller yet well-established reference model for QG and CQG tasks, we find it worthwhile to employ the few-shot strategy for CQG, especially when (1) data availability is limited or (2) one favors for a “plug-and-play” AI-assisted approach. For future work, we consider it important the application of a post-model solution to ensure that the QA pairs align with the attributes to be controlled, thereby excluding misaligned QA pairs.

LIMITATIONS

The effectiveness of controlling narrative elements and explicitness may vary across different datasets and tasks due to the unique characteristics of each context. While we have established that our study is focused on a specific domain and data, we recognize this limitation. Also, the lack of human evaluation is a limitation of this work. Although we believe the current evaluation process is solid for assessing the method’s performance in CQG, an assessment with domain experts may help better understand the potential of CQG for educational purposes.

ACKNOWLEDGEMENTS

This work was financially supported by Base Funding - UIDB/00027/2020 of the Artificial Intelligence and Computer Science Laboratory (LIACC) funded by national funds through FCT/MCTES (PIDDAC). Bernardo Leite is supported by a PhD studentship (reference 2021.05432.BD), funded by Fundação para a Ciência e a Tecnologia (FCT).

REFERENCES

Alonzo, J., Basaraba, D., Tindal, G., and Carriveau, R. S. (2009). They read, but how well do they understand? an empirical look at the nuances of measuring read-

ing comprehension. *Assessment for Effective Intervention*, 35(1):34–44.

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D., Wu, J., Winter, C., Hesse, C., Chen, M., Sigler, E., Litwin, M., Gray, S., Chess, B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I., and Amodei, D. (2020). Language models are few-shot learners. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., and Lin, H., editors, *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.

Elkins, S., Kochmar, E., Serban, I., and Cheung, J. C. K. (2023). How useful are educational questions generated by large language models? In Wang, N., Rebolledo-Mendez, G., Dimitrova, V., Matsuda, N., and Santos, O. C., editors, *Artificial Intelligence in Education. Posters and Late Breaking Results, Workshops and Tutorials, Industry and Innovation Tracks, Practitioners, Doctoral Consortium and Blue Sky*, pages 536–542, Cham. Springer Nature Switzerland.

Ghanem, B., Lutz Coleman, L., Rivard Dexter, J., von der Ohe, S., and Fyshe, A. (2022). Question generation for reading comprehension assessment by modeling how and what to ask. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 2131–2146, Dublin, Ireland. Association for Computational Linguistics.

Krathwohl, D. R. (2002). A revision of bloom’s taxonomy: An overview. *Theory Into Practice*, 41(4):212–218.

Kurdi, G., Leo, J., Parsia, B., Sattler, U., and Al-Emari, S. (2020). A systematic review of automatic question generation for educational purposes. *International Journal of Artificial Intelligence in Education*, 30(1):121–204.

Leite, B. and Lopes Cardoso, H. (2023). Towards enriched controllability for educational question generation. In Wang, N., Rebolledo-Mendez, G., Matsuda, N., Santos, O. C., and Dimitrova, V., editors, *Artificial Intelligence in Education*, pages 786–791, Cham. Springer Nature Switzerland.

Li, J., Galley, M., Brockett, C., Gao, J., and Dolan, B. (2016). A diversity-promoting objective function for neural conversation models. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 110–119, San Diego, California. Association for Computational Linguistics.

Lin, C.-Y. (2004). ROUGE: A Package for Automatic Evaluation of Summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. ACL.

Liu, P., Yuan, W., Fu, J., Jiang, Z., Hayashi, H., and Neubig, G. (2023). Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM Comput. Surv.*, 55(9).

Lynch, J. S., Van Den Broek, P., Kremer, K. E., Kendeou, P., White, M. J., and Lorch, E. P. (2008). The development of narrative comprehension and its relation

- to other early reading skills. *Reading Psychology*, 29(4):327–365.
- Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. (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, Philadelphia, Pennsylvania, USA. ACL.
- Paris, A. H. and Paris, S. G. (2003). Assessing narrative comprehension in young children. *Reading Research Quarterly*, 38(1):36–76.
- Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., Sutskever, I., et al. (2019). Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.
- Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., and Liu, P. J. (2020). Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.
- Rus, V., Cai, Z., and Graesser, A. (2008). Question generation: Example of a multi-year evaluation campaign. *Proc WS on the Question Generation Shared Task and Evaluation Challenge*.
- Sellam, T., Das, D., and Parikh, A. (2020). BLEURT: Learning robust metrics for text generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7881–7892, Online. Association for Computational Linguistics.
- Ushio, A., Alva-Manchego, F., and Camacho-Collados, J. (2022). Generative language models for paragraph-level question generation. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 670–688, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.
- Wang, Z., Valdez, J., Basu Mallick, D., and Baraniuk, R. G. (2022). Towards human-like educational question generation with large language models. In Rodrigo, M. M., Matsuda, N., Cristea, A. I., and Dimitrova, V., editors, *Artificial Intelligence in Education*, pages 153–166, Cham. Springer International Publishing.
- Xu, Y., Wang, D., Yu, M., Ritchie, D., Yao, B., Wu, T., Zhang, Z., Li, T., Bradford, N., Sun, B., Hoang, T., Sang, Y., Hou, Y., Ma, X., Yang, D., Peng, N., Yu, Z., and Warschauer, M. (2022). Fantastic questions and where to find them: FairytaleQA – an authentic dataset for narrative comprehension. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 447–460, Dublin, Ireland. Association for Computational Linguistics.
- Zhang, R., Guo, J., Chen, L., Fan, Y., and Cheng, X. (2021). A review on question generation from natural language text. *ACM Trans. Inf. Syst.*, 40(1).
- Zhao, Z., Hou, Y., Wang, D., Yu, M., Liu, C., and Ma, X. (2022). Educational question generation of children storybooks via question type distribution learning and event-centric summarization. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5073–5085, Dublin, Ireland. Association for Computational Linguistics.

APPENDIX

A Analyzing QAsys Performance in Ground Truth Questions

In Section 6.1, we hypothesize that QAsys, a question-answering system, will perform significantly better on explicit questions compared to implicit ones. Supporting this hypothesis is important for validating our evaluation procedure regarding question explicitness control. So, in Table 4, we present QAsys’s results on ground truth questions (test set). As shown in the results, QAsys performs significantly better on explicit questions, providing support for the hypothesis.

B Controllability by Nar. Element

To gain a more comprehensive understanding of narrative element control, Table 5 provides a breakdown of results for each narrative element. In the case of few-shot prompting, we observe that transitioning from using the section text solely as input to incorporating <NAR> yields the most substantial improvement (0.227) in the “Feeling” narrative element. On the other, the reference model demonstrates the best enhancement of 0.271 in the “Setting” narrative element, with the second best improvement in the “Feeling” element as well (0.217).

C Analyzing QAsys Performance via Few-Shot Prompting

In Section 6.2, where we evaluate explicitness control, we notice that QAsys performance improves when answering questions generated by the reference models. This observation led us to speculate that this advantage stems from the fact that QAsys, like the reference models to control question generation, is also trained using the same T5 model. To further analyse this, we conducted an additional experiment where we replaced the T5 model with GPT-3.5 to answer the generated questions.

Table 6 shows the QAsys scores, obtained through the few-shot strategy with GPT-3.5, when attempting to respond to generated questions. We now observe a reversal in the situation, with QAsys achieving

higher scores when addressing questions generated via few-shot prompting and lower scores for those generated from the reference model. This supports our previous speculation, suggesting that a question-answering model yields improved results when tasked with answering questions generated by a model with the same architecture. Most importantly, these results reinforce our main conclusion: regardless of the model, QAsys performs better when dealing with explicit as opposed to implicit generated questions, demonstrating that both a few-shot strategy and reference model (via fine-tuning) effectively enable the control of question explicitness.

D Varying the Number of Prompt Examples

While our current approach uses 5 examples, aligned with previous recommendations (Wang et al., 2022; Elkins et al., 2023), we explore alternative numbers of examples. Figure 4 shows the impact on closeness results (for assessing narrative elements control) when using 1, 3, 5 and 7 prompt examples. Our conclusions are as follows:

- In line with our primary results (using 5 prompt examples), we consistently observe a significant increase in closeness when incorporating narrative control attributes, regardless of the number of prompt examples.
- With the inclusion of control attributes (see green bars), increasing the number of prompt examples leads to improved narrative closeness results.
- In the absence of control attributes (see blue bars), increasing the number of prompt examples does not yield a consistent improvement in closeness results. This is expected, as the goal in this scenario is not to provide specific prompt examples for controlling a particular type of question. Instead, the goal is to generate questions that do not target specific attributes, relying on prompt examples that address various narrative and explicitness attributes.

Figure 5 presents the question-answering scores of QAsys when attempting to answer generated questions (for assessing explicitness control), which were generated by experimenting with different numbers of prompt examples. We conclude the following:

- Consistent with our main results (where we use only 5 prompt examples), QAsys consistently outperforms on explicit questions compared to implicit ones, regardless of the number of prompt examples.

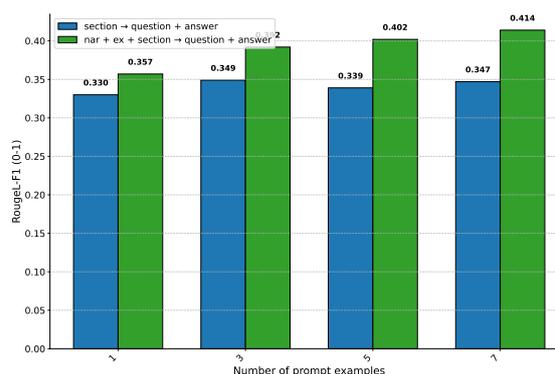


Figure 4: Results of varying the number of examples in few-shot prompting for question narrative control.

- As the number of prompt examples increases, QAsys consistently improves for explicit questions and consistently underperforms for implicit questions. We posit that this can be related to the larger set of examples “helping” the model in refining both explicit and implicit question generation.
- As the number of prompt examples increases, the gap in QAsys results widens between explicit and implicit questions. This indicates the advantage of increasing the number of prompt examples for better explicitness control.

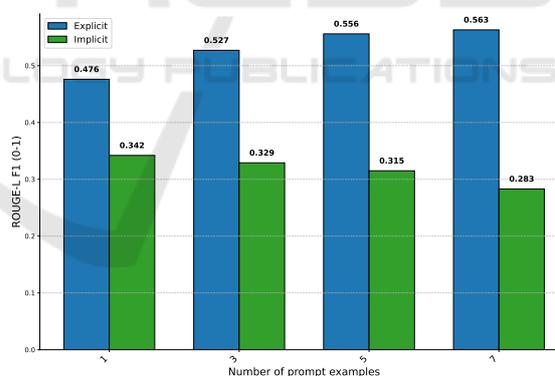


Figure 5: Results of varying the number of examples in few-shot prompting for question explicitness control.

Table 4: QAsys results in ground truth explicit and implicit questions on the test set. All scores are 0-1.

ROUGEL-F1			EXACT-MATCH		
Overall	Explicit	Implicit	Overall	Explicit	Implicit
0.597	0.732	0.194	0.315	0.403	0.051

Table 5: Closeness (ROUGE_L-F1 ↑) between generated and ground truth questions on the test set by narrative element. All scores are 0-1.

Reference Model	Data Setups	Narrative Elements						
		Chara.	Setting	Action	Feeling	Causal	Out.	Pred.
Reference Model	section → question + answer	0.320	0.279	0.372	0.300	0.381	0.273	0.240
	nar + section → question + answer	0.360	0.550	0.461	0.517	0.409	0.374	0.379
	nar + ex + section → question + answer	0.350	0.615	0.461	0.568	0.419	0.447	0.450
Few-Shot Prompting	section → question + answer	0.254	0.307	0.449	0.305	0.303	0.324	0.300
	nar + section → question + answer	0.277	0.380	0.496	0.532	0.377	0.387	0.335
	nar + ex + section → question + answer	0.296	0.365	0.498	0.516	0.367	0.337	0.327

Table 6: QAsys performance (via few-shot) by question explicitness on the test set. All scores are 0-1.

Reference Model	Data Setups	ROUGEL-F1 ↑			EXACT-MATCH ↑		
		Overall	Explicit	Implicit	Overall	Explicit	Implicit
Reference Model	ex + section → question + answer	0.517	0.580	0.352	0.160	0.185	0.093
	nar + ex + section → question + answer	0.479	0.522	0.365	0.193	0.224	0.111
Few-Shot Prompting	ex + section → question + answer	0.754	0.785	0.673	0.325	0.360	0.231
	nar + ex + section → question + answer	0.674	0.727	0.532	0.256	0.294	0.157

<p>Question-Answer Pairs (character):</p> <p>GT: Who wished the princesses into the earth? Three trolls.</p> <p>GEN: Who told the youth he could have whichever one of them he wanted? The third princess.</p> <p>GT: Who was a tall, handsome man, with dark hair, and eyes like sloes? Paul.</p> <p>GEN: Who was Snorro’s bosom friend and companion? An enormous grey-headed Raven.</p> <p style="text-align: center;">Question-Answer Pairs (setting):</p> <p>GT: Where did the bannock escape to after running away from the farmhouse? A little cottage standing among the heather.</p> <p>GEN: Where was the little cottage located? Among the heather.</p> <p>GT: Where was the small village? The foot of some high mountains.</p> <p>GEN: Where was the village located? At the foot of some high mountains.</p> <p style="text-align: center;">Question-Answer Pairs (action):</p> <p>GT: What did Matte and Maie do on Saturdays? Sailed to the nearest town, sold their fish, and went to church on Sunday.</p> <p>GEN: What did Maie and Matte do to provide for themselves? They fished for salmon in spring, for herring in summer, and for cod in winter.</p> <p>GT: What did the cow do in the autumn? Went to sea.</p> <p>GEN: What did the cow do to find food? She went out and cropped the seaweed which grew in great abundance near the shore.</p> <p style="text-align: center;">Question-Answer Pairs (feeling):</p> <p>GT: How did the princess feel in her new home? Happy.</p> <p>GEN: How did the princess feel when her mother visited her in her new home? Welcomed and happy.</p> <p>GT: How did Ryn Jin feel when the Happy Hunter asked to leave? Sorrow.</p> <p>GEN: How did King Ryn Jin feel when he had to part with Mikoto? Overcome with sorrow.</p>

Figure 6: Examples of both ground truth (GT) and generated (GEN) questions, highlighting the close similarity, mainly in the beginnings of the questions. This observation has inspired the procedure used in the narrative control evaluation (Section 6.1).

Generate questions and answers targeting the following narrative element: outcome resolution

Text: One day Isaac had put out a few miles to sea to fish, when suddenly a dark fog fell. In a flash such a tremendous storm broke, that he had to throw all his fish overboard in order to lighten ship and save his life...

Question: What happened because of the tremendous storm?

Answer: Isaac had to throw all his fish overboard in order to lighten ship and save his life.

Text: When he rode toward the pasture, such a fire burned in the herdsman's hut that it lit up every road, and then he knew that the mountain folk were inside...

Question: What happened because such a fire burned in the herdsman's hut?

Answer: It lit up every road.

(...+3 prompt examples...)

Text: But the second son spoke most sensibly too, and said: 'Whatever I give to you I deprive myself of. Just go your own way, will you?' Not long after his punishment overtook him, for no sooner had he struck a couple of blows on a tree with his axe, than he cut his leg so badly that he had to be carried home.

Question: *What happened to the second son?*

Answer: *He cut his leg so badly that he had to be carried home.*

Figure 7: Example of CQG targeting the following narrative element: outcome resolution. Generated text is shown in italics and blue. Texts are from the FairytaleQA dataset.

Generate implicit questions and answers

Text: The roaring of the torrents of water rushing along a narrow bed so disturbed the Emperor's rest day and night, that a serious nervous disorder was the result... The waters ceased their roaring, and the river was quiet in direct answer to her prayer. After this the Emperor soon recovered his health...

Question: How did the ceasing of the water roar allow the Emperor to recover in his health?

Answer: He could now sleep soundly.

Text: That one thing was that there was one room in the Castle—a room which stood at the end of a passage by itself—which she could never enter, as her husband always carried the key... But one day the Prince chanced to leave the door unlocked. As he had never told her not to do so, she went in. There she saw Princess Gold-Tree lying on the silken couch, looking as if she were asleep...

Question: Why wasn't the second wife allowed to enter one room in the Castle?

Answer: Gold-Tree was in the room.

(...+3 prompt examples...)

Text: 'Oh, only the words of an old rhyme that keeps running in my head', answered the old woman; and she raised her voice and went on: Oh, Ahti, with the long, long beard, Who dwellest in the deep blue sea, A thousand cows are in thy herd, I pray thee give one onto me. 'That's a stupid sort of song' said Matte... Then they returned to the island, and soon after went to bed. But neither Matte nor Maie could sleep a wink; the one thought of how he had profaned Sunday, and the other of Ahti's cow.

Question: *What did Maie keep singing during their journey?*

Answer: *An old rhyme about Ahti and his thousand cows.*

Figure 8: Example of CQG targeting implicit questions. Generated text is shown in italics and blue. Texts are from the FairytaleQA dataset.