Complex question generation using discourse-based data augmentation

Kushnur Binte Jahangir
UT3 - IRIT
khushnur@cse.uiu.ac.bd

Philippe Muller
UT3 - IRIT ; ANITI
philippe.muller@irit.fr

Chloé Braud
UT3 - IRIT ; CNRS ; ANITI
chloe.braud@irit.fr

Abstract

Question Generation (QG), the process of generating meaningful questions from a given context, has proven to be useful for several tasks such as question answering or FAQ generation. While most existing QG techniques generate simple, fact-based questions, this research aims to generate questions that can have complex answers (e.g. "why" questions). We propose a data augmentation method that uses discourse relations to create such questions, and experiment on existing English data. Our approach generates questions based solely on the context without answer supervision, in order to enhance question diversity and complexity. We use an encoder-decoder trained on the augmented dataset to generate either one question or multiple questions at a time, and show that the latter improves over the baseline model when doing a human quality evaluation, without degrading performance according to standard automated metrics.

1 Introduction

Question generation is the task of automatically producing varied questions about a document or a set of documents. It is used to facilitate matching real users' questions looking for information contained in those documents, for instance in the context of Customer Relationship Management or producing FAQs (Mass et al., 2020), in dialogue systems to improve interaction with users (Li et al., 2017), to develop interactive learning for educational purposes (Yao et al., 2022; Scharpf et al., 2022; CH and Saha, 2023; Eo et al., 2023) or as auxiliary tasks for e.g. summarization (Pagnoni et al., 2023). More generically, it can help question-answering (QA) systems by augmenting the amount of instances available for training, as in (Duan et al., 2017) where automatically generated questions are integrated within a text-based QA system, or in (Bartolo et al., 2021) where they are used as adversarial data to improve robustness.

As pointed out in e.g. (Sultan et al., 2020; Eo et al., 2023), question diversity is crucial, meaning that a QG system should be able to produce different types of questions, with varied lexical content and associated explicit and implicit answers. However, the majority of the current research techniques in QG have primarily focused on factoid and multiple-choice questions, where the systems are designed to retrieve factual information or require short-span answers. Since they rely more on reasoning, complex questions might help the user to gain deeper and multiple perspectives on a topic. This makes them especially useful in learning environments, complex dialogue systems, and applications that call for a better understanding of text.

On the other hand, generating complex questions is a challenging task, as the system must have a grasp of underlying semantic relationships between different parts of the text. This is where discourse relations can play an important role: discourse, or rhetorical, relations are the semantic-pragmatic links between sentences or clauses within a text, describing e.g. causal, temporal or manner connections. We assume that including discourse relations into the generation process could help the system to produce complicated questions that accurately represent the depth and complexity of the text while also being contextually relevant. For instance, recognizing a "cause-effect" discourse relation can inspire "why" questions that aim to go deeper into the reasons behind a certain occurrence or circumstance addressed in the text.

In this paper, we present an answer-agnostic QG system, based on a Transformer-driven model fine-tuned specifically for question generation. The emphasis of our QG system is on generating complex questions using discourse relations, with a particular focus on causality related questions to enhance contextual understanding. Our approach relies on data augmentation: the system is fine-

Proceedings of the 5th Workshop on Computational Approaches to Discourse (CODI 2024), pages 105–119
March 21, 2024 ©2024 Association for Computational Linguistics

tuned on reference datasets for QA that are reversed to perform the QG task, and augmented with "why" questions that are automatically built from discourse annotated data using simple heuristics. By using gold annotated data for discourse, we ensure the quality of our synthetic data. We use several datasets, the Stanford Question Answering Dataset or SQuAD (Rajpurkar et al., 2016) and Explain Like I'm Five, or ELI5 (Fan et al., 2019) for training a generator, and the Penn Discourse Treebank 2.0 (Prasad et al., 2008) for data augmentation. We evaluate the results using both automatic evaluation metrics comparing generated questions to existing reference questions about the same paragraphs, and a human assessment of the quality of the generated questions, since automated metrics do not account well for the variety of outputs from answer-agnostic models.

2 Related work

Question generation aims at producing relevant questions from documents, that could be a single text or a collection, or other types of inputs such as knowledge bases or images. In this paper, we focus on generating questions from a single document, using datasets in which each source text (i.e. context) is associated to question-answer pairs. In this context, many annotated datasets, primilarly built for QA, have been used for QG with two different settings: answer-aware systems provide the context and the targeted answer to generate the question, while answer-agnostic ones only rely on contexts.

First systems for QG were rule-based: Heilman and Smith (2010) proposed to apply syntactic modifications to generate question from declarative sentences, while Dhole and Manning (2020) refined generating patterns using semantic resources. Interestingly, Agarwal et al. (2011) demonstrated the importance of discourse connections for QG by designing patterns also relying on discourse connectives, i.e. specific expressions that can trigger discourse relations (e.g. *because, but, as a result...*), and that also constrain the type of the question to be generated. We also rely on syntactic templates and discourse information, but we significantly extend this line of work by using gold discourse annotations and by also including implicit relations.

Current approaches rely on neural architectures, either RNNs (Duan et al., 2017; Liu, 2020) or Transformers (Scialom et al., 2019; Lopez et al., 2020; Grover et al., 2021). As in our work, Scialom et al. (2019); Lopez et al. (2020) proposed an answer-agnostic QG system based on a Transformer architecture but only evaluated on SQuAD, where complex questions are almost nonexistent. Within the same setting, Grover et al. (2021) demonstrated the ability of a T5-model to generate relevant and natural questions, but the authors highlighted the challenge of evaluating generated questions using SQuAD: while the answer-agnostic setting encourages diversity, the generated questions could be far from the reference ones, an issue we address through human evaluation (see Section 8).

While these studies successfully applied transformer models such as T5 to QG, they primarily focuses on generic, simple questions, leaving complex questions less explored. Beside (Agarwal et al., 2011), discourse information was also leveraged in Stasaski et al. (2021) where rules are used to extract cause-effect relations in SQuAD: a language model then generates questions on both the cause and effect aspects, and the synthetic questions are evaluated via a QA task. Contrary to this work, we use causal relations that are manually annotated to create synthetic data to augment a generic QG model. In addition, relevant to our work is the approach introduced in (Lal et al., 2021): the authors propose simple transformations based on syntactic templates to create a corpus of "why" questions. Our heuristics to generate questions are inspired by this work, but our evaluation is not done directly on these synthetic, possibly noisy questions, but on a natural, classic benchmark (e.g. SQuAD).

Also using data augmentation, Ashok Kumar et al. (2023) rely on prompting an LLM using context-answer-question triplets to generate a set of new questions, using varied decoding strategies with the aim of increasing diversity. These questions are then ranked, based on perplexity or on a separate model, and the best ranked is added to the training set of a Flan-T5 model fine-tuned on FairytaleQA (Xu et al., 2022a) to generate questions given context-answer pairs. The evaluation demonstrates that the approach allows to generate questions for which the answer is implicit, i.e. no directly present as text span but need to be inferred. Our approach is much simpler, relying on heuristics to generate questions, with a focus on difficult, complex questions while their approach aims at producing generic diversity, with no insight on the

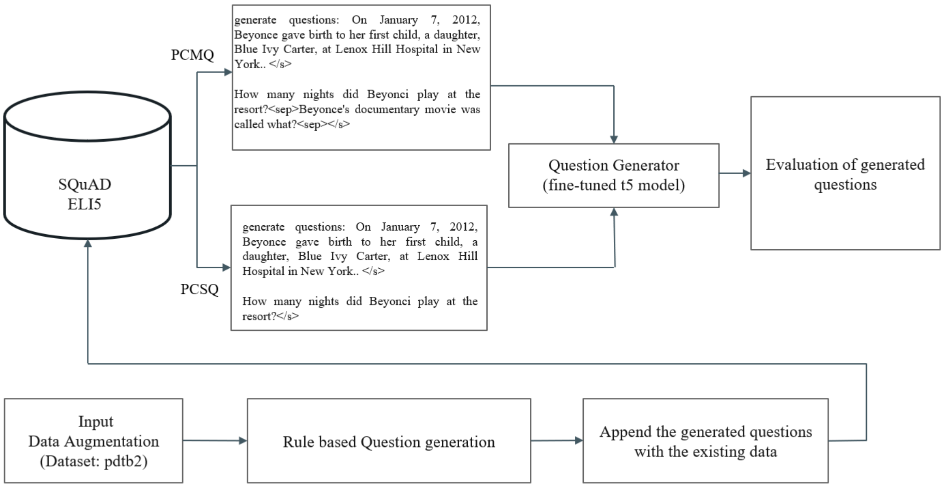
PCMQ
generate questions: On January 7, 2012, Beyonce gave birth to her first child, a daughter, Blue Ivy Carter, at Lenox Hill Hospital in New York.. </s>
How many nights did Beyonci play at the resort?<sep>Beyonce's documentary movie was called what?<sep></s>
SQuAD
ELI5
Question Generator
(fine-tuned t5 model)
Evaluation of generated questions
generate questions: On January 7, 2012, Beyonce gave birth to her first child, a daughter, Blue Ivy Carter, at Lenox Hill Hospital in New York.. </s>
How many nights did Beyonci play at the resort?</s>
PCSQ
Input
Data Augmentation
(Dataset: pdtb2)
Rule based Question generation
Append the generated questions with the existing data

Figure 1: Proposed Pipeline For Complex Question Generation Task.

types of questions generated.

3 Methodology

The pipeline for our question generation task is illustrated in Figure 1 and consists of the following elements:

- Two primary datasets, namely SQuAD, and ELI5, serve as a basis for training a model. Since we want to create an answer-agnostic model we only use the context paragraphs and the associated questions as input (ignoring information about the answer).
- The primary datasets are augmented using discourse annotated data, namely the PDTB2 dataset. We extract sentences with specific relations annotated (causal relations).
- We apply a manual rule-based approach to derive why-questions from these extracted sentences, relying on their syntactic structures, and add them to the primary datasets, with the original sentences from PDTB2 as context paragraphs.
- The augmented dataset is then used as an input for fine-tuning an encoder-decoder model from the T5 family, with two different setups:
  - PCSQ (Per Context Single Question), in which each training instance includes a context paragraph and a corresponding single question associated with it. The context and question together serve as a ’training instance’ for the T5 model during the fine-tuning process. A paragraph can thus appear several times with different questions associated.
  - PCMQ (Per Context Multiple Questions), in which each training instance contains a context and all the questions associated with this context.

PCMQ makes for a more complex decoding, but is supposed to encourage question diversity and avoid redundant generations. This setup is made possible because the reference answer for each question is ignored, and so a given paragraph is associated to several different questions in SQuAD.

Given the scarcity of complex questions in existing datasets, we aim to expand our training examples by integrating more "why" based questions. We thus use the PDTB2 dataset which contains documents annotated with discourse relations, including causality relations. These can be signaled by discourse markers, such as "because", "as", and "since", or be *implicit*, and the annotation consists of a typical marker that could be inserted.

We take the sentences from the PDTB2 dataset for both implicit and explicit relations that represent causal relations and produce questions based on some predefined rule-based templates. The rules operate on the syntactic structure of a sentence to identify the main verb and auxiliary, and transform it to produce a grammatically correct interrogative sentence, in a manner similar to how data was produced in the dataset of (Lal et al., 2021). Table 1 contains some example questions produced by this procedure. More sample of questions generated based on discourse relations is displayed in

Sentence/Arg1	Tense	Question Template	Generated Question
[jaguar was *shocked by mr. ridley's decision]*$_{ARG1}$ because *[...]*$_{ARG2}$	Past	**Why**{aux}*{rest_arg1}*?	**Why** was *jaguar shocked by mr. ridley's decision*?
the beebes' symptoms were not *related to the carpeting*	Past	**Why**{aux}{neg} *{rest_arg1}*?	**Why** were not *the beebes' symptoms related to the carpeting*?
frequently, clients express *interest in paintings but do not end up bidding*	Present	**Why** *do {rest_arg1}*?	***Why** does *frequently, clients express interest in paintings but do not end up bidding* ?

Table 1: Questions generated based on the question templates. Discourse relations link two spans of text ARG1 and ARG2 (explicitly with a marker or implicitly). Except for the first example, we only show the first argument of the causal relation (ARG1) as it is the only part used to create the question. Underlined text in the Sentence/ARG1 column represents verbs, auxiliary verbs, or negation particles extracted from the original sentence. Text in bold in the Question Template column represents fixed elements used in creating the question templates. The generated question column showcases the final questions formed using the respective templates, and incorrect question formations are marked with a star.

Appendix A.

4 Datasets

There are numerous datasets available for question generation tasks, including but not limited to NewsQA (Trischler et al., 2017), MS MARCO (Nguyen et al., 2016), Natural Questions (Kwiatkowski et al., 2019), FairytaleQA (Xu et al., 2022b), SQuAD (Rajpurkar et al., 2016), and ELI5 (Fan et al., 2019). Initially, these datasets were designed for question-answering tasks, yet they are now also broadly used in question generation research. For the present work we rely on two datasets, namely SQuAD and ELI5, to perform question generation from a given text.

SQuAD is chosen for its diverse range of source paragraphs and questions from Wikipedia, it is commonly used as a reliable benchmark for both QA and QG. The dataset was produced by Stanford University academics and contains a sizable number of paragraphs that were taken from Wikipedia articles (Rajpurkar et al., 2016). For our experiment, we use the training and development datasets from SQuAD v2.0, which were created by Rajpurkar et al. (Rajpurkar et al., 2018) in 2018.[1] However, SQuAD focuses mostly on simple factoid questions, so the ELI5 dataset, consisting of more complex questions, is incorporated.

ELI5 which stands for "Explain Like I'm Five", is another popular benchmark dataset used for tasks like QA, QG, and other NLP tasks. It is sourced from the subreddit r/explainlikeimfive. It provides long-form answers and is available from the Hugging Face website.[2] In the ELI5 dataset, each instance consists of a question and user-provided answers on reddit. In our context, we consider the answer as the source paragraph and the questions as our system's input.

PDTB2.0: Additionaly, we use the Penn Discourse Treebank Version 2.0 (PDTB2) (Prasad et al., 2008) that provides discourse annotated texts. The PDTB2 is used here to leverage discourse marker-based annotations and produce additional data to augment the training set. Other corpora exist for discourse annotations, but the PDTB is the largest annotated dataset for English including annotations for discourse relations (e.g. *cause, result, manner*), both explicit – that is triggered by a discourse connective (e.g. *because, as a result, then...*) –, and implicit – no lexical marker. Of particular interest, the PDTB2 has annotations of causal relations that we use to create "why" questions. We use the version provided from the CoNLL 2016 Shared Task (Xue et al., 2016), with level-2 annotations (15 different relation types).

5 Experiments

Our experiments aim at evaluating the influence of the training data composition, the model size, and the generating procedure as outlined in Section 3.

PCSQ *vs* PCMQ: We build the training set differently for the PCSQ and PCMQ setups: for PCSQ – i.e. one question per paragraph –, we select paragraphs and all questions about them to generate one instance per paragraph-question pair ; for PCMQ,

[1] Retrieved from the GitHub page https://rajpurkar.github.io/SQuAD-explorer/.

[2] https://huggingface.co/datasets/eli5.

Input: Many locals and tourists frequent the southern California coast for its popular beaches, and the desert city of Palm Springs is popular for its resort feel and nearby open spaces.
Reference Question: Other than the desert city why do many locals and tourists frequent southern California?
Baseline: How many locals and tourists frequent the southern California coast? **SQuAD+ELI5:** What city has a beach? **+ELI5+PDTB (Exp):** Why do many locals and tourists frequent the southern California coast? **+ELI5+PDTB (Exp+Imp):** Why do many locals and tourists frequent the southern California coast?

Table 2: Example of generated questions by different models in PCMQ approach for SQuAD test data.

all questions about a paragraph are concatenated in the same instance.

Training data composition: For the training data, we use SQuAD data as a baseline, and vary the training set by adding either (i) ELI5 data only, or (ii) ELI5 and the generated questions from the explicit examples of the PDTB, or (iii) ELI5 and the generated questions from both the implicit and explicit examples of the PDTB.

For the baseline dataset (SQuAD) we keep approximately 50k instances for the PCSQ setup, and compare to similarly-sized datasets, by having 20k instances from SQuAD and 30k instances from ELI5. The additional augmentation from the PDTB is much smaller, with about $1,600$ instances generated from explicit relations, and $1,550$ from implicit relations.

For the PCMQ setup we cannot hold the number of instances constant without restraining SQuAD too much (there are only 19k paragraphs in total), so we chose to start from a baseline including all of SQuAD + 30k ELI5 instances (note that there are much less questions per paragraph in ELI5). We kept the SQuAD-only setup for comprehensiveness, but the PCMQ setup is not entirely fair to this dataset compared to the others.

While the training and development sets of SQuAD are publicly available, the test set is not accessible to the public. So we divided the development set evenly, allocating 50% for validation and the remaining 50% for testing.

Models The experiments are conducted using the T5-base model, which is available in the Hugging Face transformers library.[3] The code, written in Python, uses the PyTorch library for fine-tuning the model. This experiment was conducted in a Google Colab Pro environment. The T5 base model has 220 million parameters. The T5 tokenizer handled data preprocessing, limiting input sequences to 512 tokens and target sequences to 64 tokens. The training involves a batch size of 4, a gradient accumulation size of 32, and 3 epochs, employing the Adam optimizer with a learning rate of 1e-4.

Decoding and post-processing To ensure diversity and comprehensiveness in questions, we keep a generation beam of four results for each test paragraph. In the case of PCSQ this ensures we have more than one question to match to the several references in SQuAD. In the PCMQ approach, the model independently generates varying lengths of questions per set, offering a greater variety compared to PCSQ. We need some post-processing to remove duplicate questions and some not well-formed ones, lacking a ’?’ mark (incomplete generations), and this impacts the final count of questions obtained for each input text. Examples of questions generated by different models are shown in Table 2. In Table 9 in Appendix C, we provide a more complete example of a question generated by the "+ELI5+PDTB (Exp+Imp)" model in the PCMQ setup.

6 Automated evaluation

We generated questions in both approaches on the SQuAD left-out paragraphs and evaluated against the corresponding reference questions using automated evaluation metrics: BLEU (Bilingual Evaluation Understudy) (Papineni et al., 2002), METEOR (Metric for Evaluation of Translation with Explicit ORdering) (Lavie and Denkowski, 2009), and ROUGE-L (Recall-Oriented Understudy for Gisting Evaluation-longest common subsequences or LCS) (Lin, 2004). These metrics are widely adopted in the literature for evaluating question generation. The evaluation tasks involved the use of the following library packages: the Natural Language Toolkit (NLTK), ROUGE, and METEOR

[3] https://huggingface.co/docs/transformers/index

Approach	Training	BLEU-1	BLEU-2	BLEU-3	BLEU-4	ROUGE-L	METEOR
PCSQ	SQuAD Baseline	**37.51**	**25.25**	**18.54**	**13.81**	**45.57**	**46.13**
	+ELI5	36.77	24.41	17.75	12.99	45.07	45.24
	+ELI5+P-E	37.10	24.74	18.09	13.40	45.18	45.18
	+ELI5+P-(E+I)	**37.51**	25.11	18.36	13.56	45.53	45.53
PCMQ	SQuAD alone	33.48	21.94	15.86	11.60	41.27	40.34
	SQuAD+ELI5	33.55	22.14	16.12	11.86	41.52	40.36
	+ELI5+P-E	33.78	22.39	16.32	12.03	**41.63**	40.77
	+ELI5+P-(E+I)	**33.89**	**22.45**	**16.34**	**12.07**	**41.63**	**40.91**

Table 3: Comparative performance of PCSQ and PCMQ approaches with different models. The scores are given in percentages. The highest scores in each metric and approach are highlighted in bold. Here, the baseline model is trained on the SQuAD dataset only. The results are presented for t5-base models. P-E and I stand for PDTB explicit and implicit relations respectively. Note that PCMQ/SQuAD alone is here for reference but not comparable to the other PCMQ setups.

in calculating BLEU (1 to 4), Rouge-L, and METEOR scores. Note that those measures, relying on common ngrams or subsequences between reference and system outputs, are moderately appropriate to our setup, where we try to generate more diverse questions than are present in the reference, without a target answer. We address this problem with a human evaluation in Section 8.

A total of 500 paragraphs were chosen from the SQuAD test dataset to assess the question-generation capability of our model. These paragraphs consist of multiple reference questions, and correspondingly, our model generates multiple questions for each paragraph. To accommodate the presence of multiple references and generated questions per paragraph in the SQuAD dataset, we implemented a mapping approach to find out which reference and generated question pairs are more relevant to each other for the evaluation, especially focusing on one-to-one match between reference and generated questions. For both PCSQ and PCMQ approaches, we combined questions generated for each context's four outputs from the beam. Using automatic evaluation metrics such as BLEU, ROUGE-L, or METEOR, we then calculated scores for each pair of matched generated and reference questions. This filtering resulted in a one-to-one matching between generated and reference questions, ensuring meaningful evaluation of our model's question-generation accuracy. Given the decoding procedure, the average number of non-duplicate generated questions was about 3.9 for PCSQ, and 9.5 for PCQM (with small variations depending on the training data).

7 Results

The results presented in Table 3 provide insights into the impact of data augmentation and the effectiveness of PCSQ and PCMQ approaches. For PCSQ, the model trained solely on SQuAD slightly outperforms augmented models in all mentioned evaluation metrics, highlighting the effectiveness of focused training on a single dataset. On the other hand, PCMQ, when using everything from SQuAD, ELI5, and the PDTB2 augmentation, outperforms slightly the baseline in BLEU (1 to 4), ROUGE-L, and METEOR.

When train with PDTB derived instances, the number of "why" question is higher (+38% when using explicit and implicit with PCMQ wrt the baseline, +24% with only explicit). In PCSQ, the increase in "why" questions is limited (going from 0 for the baseline to 10 for the full training data), reflecting its lower effectiveness in generating this question type. Questions in "how" do not seem positively affected (each system generates almost the same amount), but we did not distinguish simple "how" questions (asking for quantities, i.e "how much/many") and more complex ones. We just observed that some generated "how" questions were causal in nature, but more manual analysis is needed to evaluate this precisely.

Thus, aligned with our objective, our augmentation techniques effectively increased the number of generated "why" questions, particularly within the PCMQ models, without detrimentally affecting the quality of the questions generated as a whole, at least according to the automated metrics.

This is notable since our models are not trained on example answers, meaning they can generate questions about any aspect of the chosen paragraph, for which it is likely the reference does not include any question-answer pair.

This is why it is important to have a separate, more fine-grained evaluation of the quality of the generated answers, and this is the subject of the following section.

Model	how	why
SQuAD alone	**866**	63
SQuAD+ELI5	747	64
+ELI5+P-E	781	78
+ELI5+P-(E+I)	772	**87**

Table 4: The table presents the number of "why" and "how" questions generated by various models in PCMQ approach. The results are presented for T5-base models. Here, the baseline model is trained on the SQuAD dataset only. P-E and I stand for PDTB explicit and implicit relations respectively.

8 Human evaluation

Model	Bad	≈ ok	Good
Baseline	39.29	10.71	50.00
All+P-E	40.43	2.13	57.45
All+P-(E+I)	26.15	3.08	**70.77**

Table 5: Human quality assessment of generated questions in % according to the data that was used to train the generation model (PCMQ setup). Baseline means T5 was only fine-tuned on SQuAD.

We conducted a human evaluation to assess the quality of questions generated in PCMQ approach by three models: Baseline model, +ELI5+PDTB (Exp), and +ELI5+PDTB (Exp+Imp), all fine-tuned from the T5-base model. Two of the authors annotated a subset of randomly selected questions and their context from the SQuAD test dataset using a set of 7 predetermined categories that included subcategories for incorrect questions (more details are provided in Appendix 12); the selection was done by a third author, who kept hidden the system that produced each question. There were 137 annotated questions, some generated by more than one system. Adjudications of annotations were done by the two annotators. It turned out some of the error subcategories were quite similar, and the final categories were restricted to three cases: (1) the generated question is good: fluent, and can be answered from the source paragraph, (2) the generated question is almost good: minor disfluency and the answer is in the paragraph, (3) the question is either impossible to understand or too vague, or the paragraph does not contain an answer to the question.

Cohen's kappa (κ) was 0.48 on the 7 original categories, indicating a moderate level of inter-annotator agreement, but was 0.74 when only distinguishing between good questions and all the rest.

Table 5 presents the model-wise percentage distribution of the final adjudicated categories, providing insights into the quality assessment of generated questions. The +ELI5+PDTB (Exp+Imp) model exhibits fewer "bad" questions and a substantial increase in "good" questions compared to the baseline, presenting improved question quality with explicit and implicit relation augmentations.

Moreover, from the annotated questions, we determined the distribution of good, almost okay, and bad questions for each question type (e.g., what, why, etc.), see Table 6. We can see for instance that implicit examples help generating more why questions (32), but with a cost on the average quality of the questions (61% of good questions), while using only explicit examples has a much higher quality (79% of good questions, vs 55% for the baseline) with less why questions generated (12). This is done on a small sample of "why questions" so must be taken with a grain of salt.

9 Conclusion

We presented an approach based on discourse relation annotations to augment a question generation training set, in the case of a general answer-agnostic question generation system, and with a focus on causal questions. Our experiments show that with a small set of additional instances we can make the system generate more causal questions with a good quality, as evaluated by human annotators, and with almost no difference with respect to classic automated metrics for question generation. This is only preliminary, as the results would need to be tested on different base question-answer corpora, and more human evaluation would be precious to better separate the roles of the different factors at play here. It would also be interesting to investigate the impact of including other discourse relation types to generate different kinds of questions (e.g. "how" questions with relations of the type "goal" or "manner").

		% correct	nb
Type	Model		
How	ELI5+Exp	58.33	12
	ELI5+Exp+Imp	80.95	21
	Baseline	50.00	18
Others	ELI5+Exp	0.00	1
What	ELI5+Exp	43.75	16
	ELI5+Exp+Imp	70.00	5
	Baseline	61.36	22
When	ELI5+Exp	75.00	4
	ELI5+Exp+Imp	100.00	4
	Baseline	33.33	3
Where	ELI5+Exp	0.00	1
	Baseline	75.00	2
Who	ELI5+Exp	100.00	1
	ELI5+Exp+Imp	100.00	3
Why	ELI5+Exp	79.17	12
	ELI5+Exp+Imp	60.94	32
	Baseline	54.55	11

Table 6: Breakdown of the number of questions in the human evaluation by type, with the % of correct questions and the number of generated questions.

10 Limitations

The proposed approach augments existing datasets and thus depends on the quality and diversity of this basis. We are also reliant on existing annotated discourse data, which is costly to produce, and exist only in various quantities for some languages. As mentioned in the conclusion, the results would need to be tested on different base question-answer corpora and other languages, and more human evaluation is needed to better separate the roles of the different factors at play here. A limitation of our evaluation is the use of automated metrics, which are already known not to be very adequate to compare semantically equivalent questions if they have lexical differences, but are even more inappropriate with the goal to produce diverse questions not tied to existing answers.

Acknowledgements

This work is supported by the Quantum project (ANR-19-CE23-0022). Our work has benefited from the AI Interdisciplinary Institute ANITI. ANITI is funded by the French "Investing for the Future – PIA3" program under the Grant agreement n°ANR-19-PI3A-0004. This work is also partially supported by the AnDiaMO project (ANR-21-CE23-0020). Chloé Braud and Philippe Muller are part of the program DesCartes and are also supported by the National Research Foundation, Prime Minister's Office, Singapore under its Campus for Research Excellence and Technological Enterprise (CREATE) program.

References

Manish Agarwal, Rakshit Shah, and Prashanth Mannem. 2011. Automatic question generation using discourse cues. In *Proceedings of the Sixth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 1–9.

Nischal Ashok Kumar, Nigel Fernandez, Zichao Wang, and Andrew Lan. 2023. Improving reading comprehension question generation with data augmentation and overgenerate-and-rank. In *Proceedings of the 18th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2023)*, pages 247–259, Toronto, Canada. Association for Computational Linguistics.

Max Bartolo, Tristan Thrush, Robin Jia, Sebastian Riedel, Pontus Stenetorp, and Douwe Kiela. 2021. Improving question answering model robustness with synthetic adversarial data generation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 8830–8848, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Dhawaleswar Rao CH and Sujan Kumar Saha. 2023. Generation of multiple-choice questions from textbook contents of school-level subjects. volume 16, pages 40–52.

Kaustubh D Dhole and Christopher D Manning. 2020. Syn-qg: Syntactic and shallow semantic rules for question generation. *arXiv preprint arXiv:2004.08694*.

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, Copenhagen, Denmark. Association for Computational Linguistics.

Sugyeong Eo, Hyeonseok Moon, Jinsung Kim, Yuna Hur, Jeongwook Kim, SongEun Lee, Changwoo Chun, Sungsoo Park, and Heuiseok Lim. 2023. Towards diverse and effective question-answer pair generation from children storybooks. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 6100–6115, Toronto, Canada. Association for Computational Linguistics.

Angela Fan, Yacine Jernite, Ethan Perez, David Grangier, Jason Weston, and Michael Auli. 2019. ELI5: Long form question answering. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3558–3567, Florence, Italy. Association for Computational Linguistics.

Khushnuma Grover, Katinder Kaur, Kartikey Tiwari, Rupali, and Parteek Kumar. 2021. Deep learning based question generation using t5 transformer. In *Advanced Computing: 10th International Conference, IACC 2020, Panaji, Goa, India, December 5–6, 2020, Revised Selected Papers, Part I 10*, pages 243–255. Springer.

Michael Heilman and Noah A. Smith. 2010. Good question! statistical ranking for question generation. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 609–617, Los Angeles, California. Association for Computational Linguistics.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, et al. 2019. Natural questions: a benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:453–466.

Yash Kumar Lal, Nathanael Chambers, Raymond Mooney, and Niranjan Balasubramanian. 2021. TellMeWhy: A dataset for answering why-questions in narratives. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 596–610, Online. Association for Computational Linguistics.

Alon Lavie and Michael J Denkowski. 2009. The meteor metric for automatic evaluation of machine translation. *Machine translation*, 23:105–115.

Jiwei Li, Alexander H. Miller, Sumit Chopra, Marc'Aurelio Ranzato, and Jason Weston. 2017. Learning through dialogue interactions by asking questions. Publisher Copyright: © ICLR 2019 - Conference Track Proceedings. All rights reserved.; 5th International Conference on Learning Representations, ICLR 2017 ; Conference date: 24-04-2017 Through 26-04-2017.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Bingran Liu. 2020. Neural question generation based on seq2seq. In *Proceedings of the 2020 5th International Conference on Mathematics and Artificial Intelligence*, pages 119–123.

Luis Enrico Lopez, Diane Kathryn Cruz, Jan Christian Blaise Cruz, and Charibeth Cheng. 2020. Transformer-based end-to-end question generation. *arXiv preprint arXiv:2005.01107*, 4.

Yosi Mass, Boaz Carmeli, Haggai Roitman, and David Konopnicki. 2020. Unsupervised FAQ retrieval with question generation and BERT. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 807–812, Online. Association for Computational Linguistics.

Tri Nguyen, Mir Rosenberg, Xia Song, Jianfeng Gao, Saurabh Tiwary, Rangan Majumder, and Li Deng. 2016. MS MARCO: A human generated machine reading comprehension dataset. In *Proceedings of the Workshop on Cognitive Computation: Integrating neural and symbolic approaches 2016 co-located with the 30th Annual Conference on Neural Information Processing Systems (NIPS 2016), Barcelona, Spain, December 9, 2016*, volume 1773 of *CEUR Workshop Proceedings*. CEUR-WS.org.

Artidoro Pagnoni, Alex Fabbri, Wojciech Kryscinski, and Chien-Sheng Wu. 2023. Socratic pretraining: Question-driven pretraining for controllable summarization. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12737–12755, Toronto, Canada. Association for Computational Linguistics.

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, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Rashmi Prasad, Nikhil Dinesh, Alan Lee, Eleni Miltsakaki, Livio Robaldo, Aravind Joshi, and Bonnie Webber. 2008. The Penn Discourse TreeBank 2.0. In *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)*, Marrakech, Morocco. European Language Resources Association (ELRA).

Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know what you don't know: Unanswerable questions for SQuAD. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 784–789, Melbourne, Australia. Association for Computational Linguistics.

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, Austin, Texas. Association for Computational Linguistics.

Philipp Scharpf, Moritz Schubotz, Andreas Spitz, André Greiner-Petter, and Bela Gipp. 2022. Collaborative and ai-aided exam question generation using wikidata in education. In *Workshop Proceedings*, page 18568.

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, Florence, Italy. Association for Computational Linguistics.

Katherine Stasaski, Manav Rathod, Tony Tu, Yunfang Xiao, and Marti A. Hearst. 2021. Automatically generating cause-and-effect questions from passages. In *Proceedings of the 16th Workshop on Innovative Use of NLP for Building Educational Applications*, pages 158–170, Online. Association for Computational Linguistics.

Md Arafat Sultan, Shubham Chandel, Ramón Fernandez Astudillo, and Vittorio Castelli. 2020. On the importance of diversity in question generation for QA. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5651–5656, Online. Association for Computational Linguistics.

Adam Trischler, Tong Wang, Xingdi Yuan, Justin Harris, Alessandro Sordoni, Philip Bachman, and Kaheer Suleman. 2017. NewsQA: A machine comprehension dataset. In *Proceedings of the 2nd Workshop on Representation Learning for NLP*, pages 191–200, Vancouver, Canada. Association for Computational Linguistics.

Ying Xu, Dakuo Wang, Mo Yu, Daniel Ritchie, Bingsheng Yao, Tongshuang Wu, Zheng Zhang, Toby Li, Nora Bradford, Branda Sun, Tran Hoang, Yisi Sang, Yufang Hou, Xiaojuan Ma, Diyi Yang, Nanyun Peng, Zhou Yu, and Mark Warschauer. 2022a. 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.

Ying Xu, Dakuo Wang, Mo Yu, Daniel Ritchie, Bingsheng Yao, Tongshuang Wu, Zheng Zhang, Toby Jia-Jun Li, Nora Bradford, Branda Sun, Tran Bao Hoang, Yisi Sang, Yufang Hou, Xiaojuan Ma, Diyi Yang, Nanyun Peng, Zhou Yu, and Mark Warschauer. 2022b. Fantastic questions and where to find them: FairytaleQA – an authentic dataset for narrative comprehension. Association for Computational Linguistics.

Nianwen Xue, Hwee Tou Ng, Sameer Pradhan, Attapol Rutherford, Bonnie Webber, Chuan Wang, and Hongmin Wang. 2016. CoNLL 2016 shared task on multilingual shallow discourse parsing. In *Proceedings of the CoNLL-16 shared task*, pages 1–19, Berlin, Germany. Association for Computational Linguistics.

Bingsheng Yao, Dakuo Wang, Tongshuang Wu, Zheng Zhang, Toby Li, Mo Yu, and Ying Xu. 2022. It is AI's turn to ask humans a question: Question-answer pair generation for children's story books. In *Proceedings of the 60th Annual Meeting of the Association for Computational*

Linguistics (Volume 1: Long Papers), pages 731–744, Dublin, Ireland. Association for Computational Linguistics.

Paragraph	Generated Questions
Due to the heavy rain, the soccer match was canceled[1], and as a result, the players were disappointed[3]. Since the field was waterlogged, it was unsafe to play[6]. The organizers made the decision to cancel the match[7], and consequently, the players had to wait for another opportunity to showcase their skills[4]. Additionally, the spectators were also disappointed[5] because they were eagerly looking forward to the game. The cancellation of the match, due to the inclement weather, not only affected the players' morale but also dampened the overall excitement surrounding the event.	1. Why was the soccer match canceled? 2. Why was the soccer match canceled due to heavy rain? **(Incorrect Question)** 3. What caused the players to be disappointed? 4. What caused players to wait for another opportunity to showcase their skills? 5. Why were spectators disappointed? 6. Why was it unsafe to play? 7. Who made the decision to cancel the match?

Table 7: Example of generation from one paragraph. The table presents a text passage along with a set of generated questions intended to reflect cause-effect relationships described within the text. Corresponding answers within the text passage are color-coded to match their respective questions, and annotated with superscripts denoting question numbers for clear cross-referencing. The question is generated by +ELI5+PDTB (Exp) model in PCMQ approach.

A Sample of Generated Questions from Data Augmentation

The questions generated from the PDTB2 dataset, along with the corresponding discourse relation and discourse connective used in their formulation, are presented in Table 8.

Sentence: jaguar was shocked by mr. ridley's decision because management had believed the government wouldn't lift the golden share without consulting the company first. **(Explicit Relation)** **Connective** : because **Arg1:** jaguar was shocked by mr. ridley's decision **Question:** Why was jaguar shocked by mr. ridley's decision?
Sentence: jeastern airlines' creditors have begun exploring alternative approaches to a chapter 11 reorganization , they are unhappy with the carrier's latest proposal. **(Implicit Relation)** **Connective** : None **Arg1:** jeastern airlines' creditors have begun exploring alternative approaches to a chapter 11 reorganization **Question:** Why have eastern airlines' creditors begun exploring alternative approaches to a chapter 11 reorganization?

Table 8: Examples of generated questions for both explicit and implicit relation from PDTB2 dataset.

B Sample of Generated Questions

We have included some samples of generated questions from the fine-tuned models on test dataset in Table 9 and 10. In Table 9, for a given input paragraph, the augmented model in PCMQ approaches generated 4 set of questions where each question set may contain various length questions.

C Human Annotation

C.1 Sample of Annotated Questions

We have provided some sample questions that have been annotated by the annotators. The annotated questions can be found in Table 11.

Input: The Normans (Norman: Nourmands; French: Normands; Latin: Normanni) were the people who in the 10th and 11th centuries gave their name to Normandy, a region in France. They were descended from Norse ("Norman" comes from "Norseman") raiders and pirates from Denmark, Iceland and Norway who, under their leader Rollo, agreed to swear fealty to King Charles III of West Francia. Through generations of assimilation and mixing with the native Frankish and Roman-Gaulish populations, their descendants would gradually merge with the Carolingian-based cultures of West Francia. The distinct cultural and ethnic identity of the Normans emerged initially in the first half of the 10th century, and it continued to evolve over the succeeding centuries.

Generated Questions:

1. Who gave their name to Normandy? What was the name of the region in France? Who did the Normans swear fealty to? When did the distinct cultural and ethnic identity of the Norman people emerge?
2. What was the name of the Normans in the 10th and 11th centuries? What was Normandy? Who did the Norman people swear fealty to? When did the distinct cultural and ethnic identity of the people emerge?
3. What was the name of the Normans in the 10th and 11th centuries? What was Normandy? Who did the Norman people swear fealty to?When did the distinct cultural and ethnic identity of the people emerge? How did the people of Normand
4. Who gave their name to Normandy? Who did the Normans swear fealty to?What did the people of Normands mix with? When did the distinct cultural and ethnic identity of the Norman people emerge?

Table 9: Example of input text and generated questions in PCMQ format for SQuAD test set. The question is generated by "+ELI5+PDTB2(Exp+Imp)". The incorrect questions are red-colored and the yellow-highlighted text present an incomplete question.

Input: Price controls exacerbated the crisis in the US. The system limited the price of "old oil" (that which had already been discovered) while allowing newly discovered oil to be sold at a higher price to encourage investment. Predictably, old oil was withdrawn from the market, creating greater scarcity. The rule also discouraged development of alternative energies. The rule had been intended to promote oil exploration. Scarcity was addressed by rationing (as in many countries). Motorists faced long lines at gas stations beginning in summer 1972 and increasing by summer 1973.
Reference Question: Why was old oil withdrawn from the market?
Baseline: What was withdrawn from the market? **SQuAD+ELI5:** What did the price control limit? **+ELI5+PDTB (Exp):** Old oil was withdrawn from the market creating what? **+ELI5+PDTB (Exp+Imp):** Why was old oil withdrawn from the market?

Table 10: Example of generated questions by different models in PCMQ approach for SQuAD test data.

Input Text	**Generated Questions**	**Category**
Price controls exacerbated the crisis in the US. The system limited the price of "old oil" (that which had already been discovered) while allowing newly discovered oil to be sold at a higher price to encourage investment. Predictably, old oil was withdrawn from the market, creating greater scarcity. The rule also discouraged development of alternative energies. The rule had been intended to promote oil exploration. Scarcity was addressed by rationing (as in many countries). Motorists faced long lines at gas stations beginning in summer 1972 and increasing by summer 1973.	Why was old oil withdrawn from the market?	Good question. Answer is present in the text.
Highly concentrated sources of oxygen promote rapid combustion. Fire and explosion hazards exist when concentrated oxidants and fuels are brought into close proximity; an ignition event, such as heat or a spark, is needed to trigger combustion. Oxygen is the oxidant, not the fuel, but nevertheless the source of most of the chemical energy released in combustion. Combustion hazards also apply to compounds of oxygen with a high oxidative potential, such as peroxides, chlorates, nitrates, perchlorates, and dichromates because they can donate oxygen to a fire.	How do compounds with oxidation potential contribute oxygen to?	Incorrect question but with relevant words from the input.
As indigenous territories continue to be destroyed by deforestation and ecocide, such as in the Peruvian Amazon indigenous peoples' rainforest communities continue to disappear, while others, like the Urarina continue to struggle to fight for their cultural survival and the fate of their forested territories. Meanwhile, the relationship between non-human primates in the subsistence and symbolism of indigenous lowland South American peoples has gained increased attention, as have ethno-biology and community-based conservation efforts.	Why do indigenous territories continue to be destroyed by deforestation and ecocide?	Grammatically correct but the answer doesn't exist.

Table 11: Sample of Annotated Questions by the Annotators. Red-colored text represents the answer texts for the question within the paragraph. The input paragraph is from SQuAD test dataset.

Question Category	Description
Good question. Answer is present in the text.	The answer to the generated question exists in the given sentence/paragraph.
Incorrect question but with relevant words from the input.	The generated question contains some words/phrases from the input, but the question is not grammatically correct and/or does not make sense.
Question and answer are mixed	The generated question contains some part of the answer.
Grammatical mistake	The generated question is grammatically incorrect.
Grammatically correct but the answer doesn't exist	The generated question is grammatically correct, but the answer to the question does not exist in the input context.
Completely vague	The generated question is not meaningful, too vague.
Two valid questions are mixed	The generated question contains two questions from different parts of the input.

Table 12: Description of different category set for question evaluation

C.2 Question Category for Annotations

The annotators assigned each question to one of the seven predetermined categories. Details of each category are provided in Table 12.