ChainCQG: Flow-Aware Conversational Question Generation

Jing Gu
SearchableAI

Mostafa Mirshekari
SearchableAI

Zhou Yu
UC Davis/SearchableAI

Aaron Sisto
SearchableAI

Abstract

Conversational systems enable numerous valuable applications, and question-answering is an important component underlying many of these. However, conversational question-answering remains challenging due to the lack of realistic, domain-specific training data. Inspired by this bottleneck, we focus on conversational question generation as a means to generate synthetic conversations for training and evaluation purposes. Here, we present *ChainCQG*, a neural Conversational Question Generation (CGQ) model and evaluate its performance on both automatic metrics and human metrics. Our work proposes a number of novel strategies to improve conversational flow and accommodate varying question types and overall fluidity. Specifically, we design *ChainCQG* as a two-stage architecture that learns question-answer representations across multiple dialogue turns using a flow propagation training strategy. *ChainCQG* significantly outperforms both answer-aware and answer-unaware SOTA baselines (e.g., up to 48% BLEU-1 improvement). Additionally, our model is able to generate different question types, with improved fluidity and coreference alignment.

1 Introduction

Conversational systems are important in many real-world applications, including personal assistants, educational tutors (Winkler et al., 2020), customer service (Asri et al., 2017; Budzianowski et al., 2018), and increasingly, entertainment. A key component of these systems is the ability to retrieve information from across different sources as naturally and efficiently as possible. In analogous human interactions, such a search generally occurs through conversation. In this context, a conversation consists of a sequence of dialogue turns during which the search objective becomes clearer over time. The applications mentioned above could benefit greatly from this type of multi-turn interaction, enabling conversational agents to accurately predict intent, request additional information, and better understand ambiguous followup questions and comments. In an applied setting, meaningful and natural conversations are important features of virtual entities as a means to establish trust and improve usability.

Here, we are motivated by the challenging task of conversational question answering (CQA). Current approaches make use of open-source datasets such as CoQA (Reddy et al., 2019) and QuAC (Choi et al., 2018). However, these datasets have limited applicability in practical settings, because 1) they are created from generic source material, and 2) they do not necessarily consider the full diversity of question types and vernacular that may be encountered in natural dialogue. However, creating realistic, domain-specific datasets to train CQA models is notoriously costly and time-consuming. As such, we focus on the related task of question generation as a means to generate synthetic user queries on out-of-domain textual data and subsequently, create new datasets or complement existing ones. This will ultimately enable training CQA models in closed-loop, simulation environments, as well as allow machines to initiate dialogue and engage in information-seeking behavior.

Intuitively, an end-to-end CQA simulation consists of question generation (QG), and question answering (QA) modules. The QA module aims to predict the answer given the passage, question and dialogue history, whereas the QG module aims to predict a question given the passage, dialogue history, and possibly, the target answer and rationale. QA models are heavily studied in the literature (e.g., (Zhu et al., 2018; Huang et al., 2018; Yeh and Chen, 2019; Chen et al., 2019; Ju et al., 2019; Ohsugi et al., 2019)); however, the QG module,

which is the focus of this paper, has received less attention. Most QG-related literature has focused on single-turn question generation from question-answer datasets such as SQuAD (Rajpurkar et al., 2016), and other textual sources like Wikipedia articles (Du and Cardie, 2018).

Conversational Question Generation (CQG) proves more challenging than single-turn QG as the questions are often highly ambiguous on their own, forcing the model to learn a deeper understanding of the context surrounding the passage text and dialogue history (Pan et al., 2019). Most conversational QG studies have generated questions using only the passage and dialogue history as inputs (i.e., answer-unaware) (Pan et al., 2019; Qi et al., 2020; Nakanishi et al., 2019; Wang et al., 2018a). Answer-aware CQG models, on the other hand, generate questions based on the target answer, as well as dialogue history and passage. Although answer-aware CQG models seek to improve the generated conversation flow, current answer-aware QG models lack the necessary performance and robustness in real-life applications, suffering from issues including coreference alignment and inability to generate many different types of questions.

In this paper, we introduce *ChainCQG*, a Conversational QG model that achieves improved performance and robustness by jointly learning the representations of questions and answers sequentially, across multiple dialogue turns. To this end, we outline a two-stage architecture, inspired by the approach discussed in (Wu et al., 2019), where two language models are used to simulate user and system in a response generation task. Our *ChainCQG* model is trained end-to-end, resulting in high-quality questions, while reducing computational cost by using shared parameters across both GPTs. Using an answer-aware strategy grounds each turn of QG by jointly encoding the passage with the target answer rationale, increasing accuracy of the generated question types and further aligning coreferences between dialogue turns. We evaluate our approach using the inverted CoQA dataset (Reddy et al., 2019), which is a large-scale conversational question answering dataset that we re-purposed for question generation. Our model outperforms existing baselines (i.e., CFNet (Gao et al., 2019a) and ReDR (Pan et al., 2019)) by a large margin on automatic evaluation metrics, and shows improved results on human evaluation metrics as well. More information about the baselines will be discussed in Section 2.

In summary, the main contributions of this paper are threefold:

- The *ChainCQG* two-stage architecture is introduced with answer-aware input encoding, and it is an end-to-end model which is able to generate different types of questions, such as summary, factoid and Yes/No questions.
- *ChainCQG* sets the new SOTA results on answer-aware CQG task with robust human evaluation results.
- We demonstrate a flow propagation-based training method to learn question-answer representations across multiple dialogue spans.

The remainder of the paper is structured as follows: First, we discuss the related works and how our work is distinguished from them (in Section 2). Then, we discuss the *ChainCQG* framework and the preprocessing steps (in Section 3). Next, we describe our experiments, datasets and metrics, and evaluation results (in Section 4). Finally, we discuss future work and the ethical issues, and conclusions (in Sections 5 and 6).

2 Related Work

Conversationally generating questions is conceptually similar to tasks such as single-turn question generation and conversational question answering. In this section, we first explore previous approaches for question generation and then discuss outstanding research challenges.

2.1 Single-turn Question Generation

Single-turn question generation has been the focus of extensive research. Two of the main categories in QG are answer-agnostic/open-domain and answer-aware. The former category generates the question without knowledge of the answer and solely based on the passage; whereas, the latter takes both passage and answer as inputs for generating questions. Traditional approaches for answer-agnostic QG include two main steps: content selection and question construction (Du and Cardie, 2017; Subramanian et al., 2017). Some of the more recent approaches utilize sequence-to-sequence (seq2seq) models for end-to-end question generation using Transformer-based architectures (Scialom et al., 2019). Various techniques has been used for improving the generated questions, including contextualized word embeddings (Scialom et al., 2019),

question type usage and copying mechanism (Wu et al., 2020), and typed decoders (Wang et al., 2018b).

To enable answer-aware question generation, the input passage is combined with information describing the answer. For example, the passage can be concatenated with the answer positions and lexical features (e.g., part-of-speech (POS) and named entity (NER)) to form the encoder input of a seq2seq model (Zhou et al., 2017). Jointly modelling the unstructured passage and the structured answer-relevant relation has been suggested for improving the question generation using a seq2seq model (Li et al., 2019). Additional techniques have been proposed to solve various answer-aware QG challenges, including poor performance on long passages (Zhao et al., 2018) and the bias of repeating the terms in the target answer (Kim et al., 2019).

2.2 Conversational Question Generation

Compared to single-turn QG, conversational (i.e., multi-turn) QG is less frequently explored in the literature. Further, it is more difficult as it requires a deeper understanding of the context and the dialogue history, and will be the focus of this paper. Previous work is mostly focused on answer-agnostic/open-domain question generation (Pan et al., 2019; Qi et al., 2020; Nakanishi et al., 2019; Wang et al., 2018a). Specifically, Pan et al. (2019) proposed an encoder-decoder framework, ReDR, for answer-agnostic CQG, which is fine-tuned using reinforcement from an auxiliary question-answer model. However, in this setting, maintaining conversational flow and consistency between dialogue turns is a primary challenge.

In this paper, we focus on answer-aware question generation. By grounding the generated question with the target answer rationale in each turn, this approach seeks to improve conversational flow and question-answer consistency. Within answer-aware CQG, (Gao et al., 2019b) introduced CFNet, which combined an auxiliary coreference alignment module with a copy mechanism and dialogue flow embedding, which outperformed answer-unaware baselines on CQG tasks. As will be described in the experimental setting section, we compare our model to answer-aware, CFNet (Gao et al., 2019b), and answer-unaware, ReDR (Pan et al., 2019), SOTA baselines. As a practical note, in real-world applications, answer-aware QG systems may be augmented with an Answer Generation (AG) model to form an answer-unaware model that predicts the next answer before generating a consistent question. Discussing this AG model is out of the scope of this paper and will be the focus of future work.

3 The *ChainCQG* Framework

ChainCQG learns the question-answer representations jointly through two modules: Answer Encoding (AE) and Question Generation (QG). Encoding the answer based on the passage and dialogue history improves the answer understanding within the QG module, which in turn improves the generated questions. In the rest of this section, we provide a description of the input pre-processing steps, the general CQG problem formulation, and the AE and QG modules.

3.1 Task Definition

The conversational QG task in this paper aims to predict the next question given the passage (P), target answer (A_n) and history of the dialogue preceding the n$^{\text{th}}$ turn, (H_n). We also consider the answer rationale in each turn, and annotate the passage with the target answer rationale span, which we denote as P_{HL_n}. Mathematically speaking, given the annotated passage (P_{HL_n}), target answer (A_n), and dialogue history ($H_n = ((Q_1, A_1), (Q_2, A_2), ..., (Q_{n-1}, A_{n-1}))$), the QG task predicts the next question Q_n. This task can be defined as to generate a question $\hat{Q}$ that:

$$\hat{Q} = \underset{Q_n}{argmax} Prob(Q_n | P_{HL_n}, A_n, H_n). \quad (1)$$

3.2 Input Preprocessing

In this Section, we briefly describe the processing steps necessary to prepare the input data. Specifically, we take the following steps:

1. we first create n sub-dialogs based on the dialogue, with i-th sub-dialog starting from the first turn and finishing with i-th turn, i.e., $SD_1 = \{\{Q_1, A_1\}\}$, $SD_2 = \{\{Q_1, A_1\}, \{Q_2, A_2\}\}$, ..., $SD_n = \{\{Q_1, A_1\}, \{Q_2, A_2\}, ..., \{Q_n, A_n\}\}$.

2. For i-th sub-dialog, we use a highlight token, [HL], to denote the answer rationale in the passage corresponding to the answer in i-th turn, which serves as additional context for

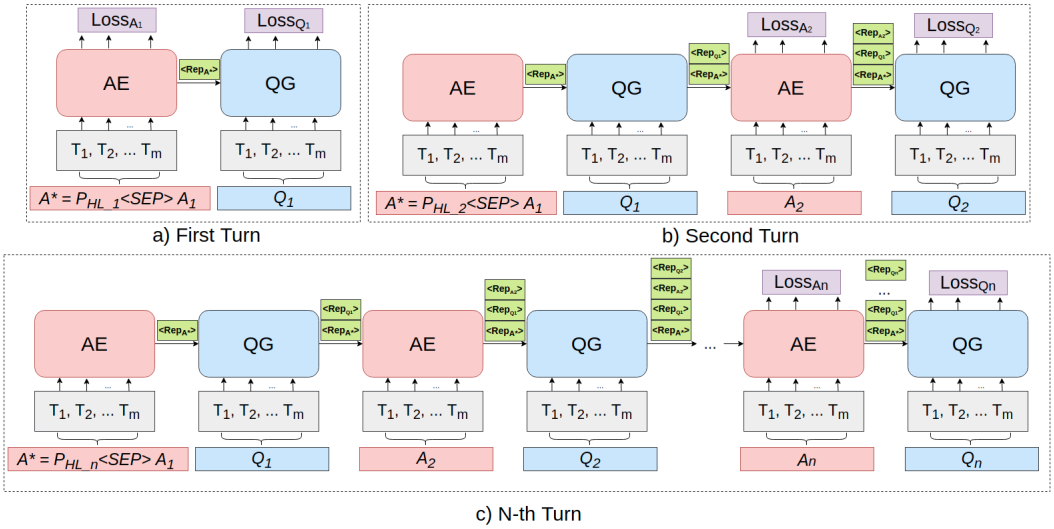

Figure 1: Main structure of *ChainCQG*. Each QA turn in the dialogue span is trained with a separate conversational flow that contains all previous dialogue turns. Answer Encoder and Question Generator modules iteratively generate and share answer and question representations across multiple dialogue turns.

the target answer. The passage (with the highlighted token corresponding to i-th turn) will be concatenated to the first answer (i.e., A_1) by a SEP token. We denote the concatenation as A^*.

3. We then reverse the order of the answers and questions in the sub-dialogues (e.g., {A1, Q1} instead of {Q1, A1}). The reason behind this step is to align the input sequence with the natural order of the QG task where the questions come after the answers. We examine the effects of this ordering scheme in later ablation studies.

4. We use a highlight token, [HL], to denote the answer rationale in the passage, which serves as additional context for the target answer.

3.3 Answer Encoding and Question Generation Modules

Our goal is to generate questions for the answer-aware CQG task. We employ two separate modules (AE and QG) to learn a better representation of the dialog history, context, and target answer. In this paper, we use GPT to represent both of these modules. Specifically, the AE model is used to get the representation of the passage and the representation of the answer in each turn, and the QG model is used to get the representation of questions in the dialogue history and generate the next question in the conversation. The AE and QG models communicate via the hidden states, which are the K and V values when using GPT. K and V values together form a contexual representation for conversation history.

3.4 Flow Propagation-Based Training

To improve the conversational flow of the CQG, we introduce a sequential training process. Figure 1 shows the main intuition behind this process. To make it more concrete, let us consider a dialogue of n turns ({$\{Q_1, A_1\}$, $\{Q_2, A_2\}$, ..., $\{Q_n, A_n\}$ }). The forward propagation process iterates through all previous turns and finally estimates the loss values for A_n and Q_n. In this process, we pass the GPT-based (K, V) representation forward to the next module, which accumulates the representations of each previous turn, with the original highlighted passage as reference. For each sub-dialog, we only update the loss from the answer and question in the last turn since the highlighted span specifies the information for the last turn. For a sub-dialog of n turns, the loss is calculated as

$$Loss = Loss_{A_n} + Loss_{Q_n} \quad (2)$$

where

$$Loss_{A_n} = CE(A_n, P_{A_n}) \quad (3)$$

and

$$Loss_{Q_n} = CE(Q_n, P_{Q_n}) \quad (4)$$

CE means the cross-entropy loss from a target sentence. The parameters of the model are updated by backpropagating the aggregated loss values. By considering the encoding of the previous turns for

estimating the loss and increasingly considering various subdialogues, the flow propagation-based training improves the conversational flow of the CQG.

4 Experiments

4.1 Dataset

We conduct experiments on the CoQA dataset (Reddy et al., 2019), which is a large-scale conversational question answering dataset composed of 8k conversations with 127k question-answer pairs collected via Amazon Mechanical Turk. Each dialogue turn also contains the supporting rational for each answer. A number of different question types are present, as documented in the original paper, including Yes/No, explanation (i.e. How?, Why?), and factoid (i.e. What? When? Where? How much?). Yes/No and explanation questions, alongside the overall conversational language, make this an exceedingly challenging dataset for CQG. Since the private test set is not available, we conduct experiments on the training set and the dev set. We randomly sample 10% from the original training set to form the test set, and keep the original dev set unchanged. We conduct our experiments with a training set with 97783 examples, dev set with 7983 examples and test set with 10846 examples. We report the performance on the test set.

4.2 Implementation Details

We use both $\text{GPT}_{\text{small}}$ and $\text{GPT}_{\text{medium}}$ in all the experiments. For baselines, we consider ReDR, the SOTA method in answer-unaware CQG, and CFNet, the SOTA method in answer-aware CQG. We also implemented two SOTA pre-training generation model, T5 and BART. We also use T5_{large} (770M) and $\text{BART}_{\text{large}}$ (400M), which are comparable with *ChainCQG*-M in terms of parameter size.

We initialize *ChainCQG* with the open sourced GPT-2 parameters (Radford et al., 2019). We apply AdamW optimizer (Loshchilov and Hutter, 2019), and the number of warmup ratio is set to be 0.1. Learning rate is tuned between 2e-5 5e-5. The dropout ratio is set to be 0.1. We decode questions by nucleus sampling (Holtzman et al., 2020) with top-p as 0.2, top-k as 400, and temperature as 0.7.

4.3 Evaluation Metric

Our main objectives when evaluating our model are quality of the generated questions and performance on our task goal (e.g., asking conversational questions that are consistent with the target answers). To this end, we first examine a set of automated metrics. Then, to ensure robustness, we evaluate and discuss a set of human-based metrics.

4.3.1 Automated Metrics

To evaluate our question generation approach, we aim to show that it is 1) grammatically and semantically correct and 2) able to achieve the task objectives. To achieve the first goal, we compute automatic metrics with respect to the ground truth questions. We report multiple commonly used metrics, including BLEU (Papineni et al., 2002), ROUGE (Lin, 2004), METEOR (Banerjee and Lavie, 2005), and perplexity (Clarkson and Robinson, 1999). BertScore (Zhang et al., 2019) and MoverScore (Zhao et al., 2019) are recently proposed metrics that utilize a large pre-trained model to evaluate the generation quality in semantic level. We use both of them to evaluate the semantical similarity between the generated question and the reference question. Since two questions could express similar meaning but represent in different wording, these two semantic-level metric could also show important information about the question quality.

4.3.2 Human Evaluation Metrics and Procedure

In this section, we discuss the metrics used for human evaluation. Human evaluation provides additional support for the approach and the robustness of automatic evaluation. Specifically, we use answerability, and fluency to measure the quality of the generated questions in relation to the context. We have used Mechanical Turk for this evaluation.

In the context of answer-aware question generation, **Answer Consistency** answers the question of whether the generated questions result in the right answers (Celikyilmaz et al., 2020a). To measure this metric, we provide the passage and the answer and ask the evaluators whether the generated question is consistent with the answer (i.e., 1 for consistent and 0 for inconsistent). **Fluency** measures the quality of the generated questions and accounts for criteria such as grammar, spelling, choice of words, and style (Du et al., 2017). To measure this metric, we provide the generated question and ask the human evaluator whether the language in the generated question fluent. We assign 2 for cases with no grammar, style, spelling, missing entity names or mismatched pronouns, 0 for cases with one or more

grammar/spelling mistakes, one or more missing entity names, one or more mismatched pronouns, and 1 for cases with maximum of one mistake in any category. We scale the fluency score to (0,1) by maximal possible evaluation scores.

4.4 Main Results

The *ChainCQG* model architecture is evaluated alongside two SOTA baselines (ReDR, CFNet) on a number of automatic metrics, including BLEU (1-4), METEOR, ROUGE-L, MoverScore, and BERTScore. More information about these metrics and baselines was presented in Section 4.3. These scores seek to evaluate the lexical overlap, and to some degree, the semantic similarity, between generated and ground truth questions within each dialogue span. We also train two QG models based on pretrained Transformer sequence-to-sequence architectures (BART-large (Lewis et al., 2020), T5-large (Raffel et al., 2020)) using all elements of the *ChainCQG* methodology except the question-answer representation sharing mechanism used in the final 2-GPT *ChainCQG* model.

4.4.1 Automated Metrics Results

Results of all models and baselines are shown in Table 1. In the top row of this Table, P is the perplexity, B1-4 are BLEU 1 through BLEU 4, M is METEOR, RL is ROUGE-L, BS is the BERTScore, and MS is MoverScore. In the first column, ChainCQG-M and ChainCQG-S refer to two version of our approach using medium and small GPT-2. Major observations are listed below:

The top performing *ChainCQG* model, composed of two GPT-2 Medium modules, improves upon all baselines, across all metrics considered here, by a considerable margin. Besides, it also outperform the T5-large by a large margin, which has more parameters. This suggests that our *ChainCQG* could get a better representation by the two GPT structure with less parameters. We improve upon the current answer-aware CQG SOTA, CFNet, on each metric as well. Note that with our methods, T5-large and BART-large also outperforms the SOTA methods. T5-large is the next best performing model, trailing *ChainCQG* by at least two points on all metrics except BERTScore, which shows a narrower margin of improvement.

4.4.2 Human Evaluation Results

It is well-known that automatic evaluation metrics do not always correlate with human judgement in conversational generation tasks Celikyilmaz et al. (2020b). Especially in the context of CQG, there is an enormous range of possible questions that could satisfy a target answer and passage, and token-based metrics are inherently unable to measure the similarity between sequences with low degrees of lexical overlap. As a recourse, we also assess our model performance on a number of human evaluation metrics described in a previous section: Answer Consistency and Fluidity. These metrics cover an important cross-section of human judgement, which is not represented in the automatic metrics. Specifically, we seek to quantify the naturalness and consistency of *ChainCQG* results within each dialogue span. Table 2 shows the performance of our models and baselines on these metrics.

Overall, both standalone QG models using BART and T5, as well as the 2-GPT Medium *ChainCQG* outperformed the SOTA baseline, CFNet, on both metrics, while the *ChainCQG* model achieved the best performance of all models on both metrics. The Answer Consistency roughly indicates that the question types were better aligned with the target answer and dialogue history, than the baseline, while the Fluency metric points to improvements in factors like grammar, coreference alignment, and dialogue flow. Together with the Automatic Metrics, these results support our finding that the *ChainCQG* model is able to learn to produce conversational dialogue that is well aligned with the CoQA dataset, both lexically and semantically, and more robust in general.

4.5 Results Discussion

These results indicate that the *ChainCQG* architecture is able to more accurately reproduce many of the features of conversational QA in the dataset, including coreference, varying question types, ambiguity of followup questions and other types of conversational noise. More specifically, the improvement over single CQG sequence-to-sequence models like T5 demonstrates the success of learning joint question-answer representations and using the encoding of the latter to inform the QG module. We present a more complete analysis of error and ablation studies in the following sections. The answer-aware QG strategy is also validated here, as shown by the significant improvement of every answer-aware model over the answer-unaware (ReDR) baseline. Finally, the comparison of *ChainCQG* to the current answer-aware

Baseline Models	B1	B2	B3	B4	M	RL	BS	MS
ReDR	27.58	7.81	2.83	1.35	12.15	34.05	87.14	7.62
CFNet	38.24	22.60	16.11	12.23	25.75	43.25	91.25	25.92
Our Models	B1	B2	B3	B4	M	RL	BS	MS
BART-large	49.41	30.57	19.40	12.34	35.78	46.88	92.55	31.89
T5-large	50.83	32.64	20.81	13.84	37.08	48.67	92.86	33.91
ChainCQG-M	**53.15**	**35.31**	**23.31**	**15.78**	**40.15**	**50.98**	**93.14**	**36.40**
ChainCQG-S	49.26	31.06	20.24	12.11	33.26	46.23	92.53	32.82

Table 1: Automated metrics evaluation Results.

Model	Consistency	Fluency
CFNET	0.710	0.439
BART	0.792	0.482
T5	0.757	0.462
ChainCQG-M	**0.817**	**0.548**

Table 2: Human evaluation results.

QG SOTA, CFNet, demonstrates the importance of the question-answer representation and encoding scheme, in our model. While CFNet incorporates tactical model components to improve the quality of CQG along specific dimensions, such as coreference and dialogue flow, our model is able to flexibly learn the progression of questions without the need for architectural components that target specific linguistic features. Another important point is that CFNet excluded all Yes/No questions from their analysis, as they proved difficult to reliably generate. Our model not only achieves SOTA performance, but is also able to natively generate every question type present in the dataset. Moreover, we noticed that our models shows a great coreference alignment ability when generating question with complex and entangled dialog history.

4.6 Ablation Study

Our ablation study aims to understand the effectiveness of various design choices of the *ChainCQG* approach outlined in Section 4. In all ablation experiments, the base model is the 2-GPT *ChainCQG* model, combining the AE and QG modules. Results of this analysis are shown in Table 3. We have applied the following ablations:

4.6.1 Removing the dialogue history

Here, we evaluate the effect of the flow propagation training scheme. Table 3 shows that removing the dialogue history, and concsequently, any notion of dialogue flow, reduces performance across all the metrics (e.g., approximately 14% for both small and medium versions). These results match the intuition that dialogue history provides essential context to correctly handle coreferences and natural transitions.

4.6.2 Removing the answer rationale highlight tokens

To evaluate the effect of grounding the generated questions in the relevant passage text, we remove the answer rationale highlight tokens from the input passage. The results in Table 3 show that this ablation decreases performance in all the metrics. For example, removing the highlight reduces BLEU-1 for the medium GPT from 53.15 to 47.07 (approximately 11% reduction). We conclude that the highlighted tokens ground the model in the relevant passage information, providing essential context while focusing the scope of the question.

4.6.3 Changing the order of the questions and answers

As discussed in Section 3.2, we have used the AQ order instead of QA. Here, we evaluate the effect of such ordering. As Table 3 shows, reversing the order of the question and answers results in a performance reduction of approximately 5% in the BLEU-1 score for the medium GPT model. This shows that the AQ order is a more natural structure for dialogue flow propagation, since answers proceed the generated question in each turn.

4.6.4 Removing the AE module

As discussed in Sections 1 and 3.3, by including the AE module, we aim to address the challenge of expressing the representations of questions and answers over multiple dialogue turns. Here, we remove the AE module to validate the effect of this modelling choice. The results in Table 3 show that removing the AE module reduces the performance of the model by 3% and 8% in the BLEU-

Model	P	B1	B2	B3	B4	M	RL	BS	MS
ChainCQG-M	7.04	53.15	35.31	23.31	15.78	40.15	50.98	93.14	36.40
ChainCQG-S	9.55	49.26	31.06	20.24	12.11	33.26	46.23	92.53	32.82
M w/o history	9.13	45.53	28.35	18.31	11.27	30.35	40.45	92.54	27.59
S w/o history	11.1	42.54	25.63	14.91	8.01	27.73	39.23	91.03	24.30
M w/o highlight	7.83	47.07	30.63	20.54	12.91	32.56	43.63	92.36	31.50
S w/o highlight	11.09	43.63	25.74	17.14	10.54	27.73	40.43	91.23	24.82
M w/o AQ order	7.43	50.23	33.65	21.43	14.08	37.35	48.88	92.94	34.73
S w/o AQ order	9.90	47.65	30.51	19.04	10.80	31.58	42.82	92.21	29.74
M w/o AE module	8.05	51.64	33.26	21.26	13.86	37.23	47.23	92.64	32.23
S w/o AE module	15.21	45.23	28.19	18.01	10.73	29.43	44.12	92.13	27.28

Table 3: Ablation Study Results.

1 score for the medium and small GPT models, respectively. This indicates that propagating the question-answer representations across dialogue turns produces rich temporal representations that improve the fidelity of dialogue flow.

4.7 Error Analysis

In order to better understand the performance differentiation between our model and baselines considered here, we inspected samples of poor quality of questions. While the SOTA baseline, CFNet, neglected all Yes/No questions completely, our model is overall, very successful at generating this type, alongside others such as factoid and explanation. However, Yes/No questions can still be problematic when the answer context includes many potential targets, each of which could be satisfied by a consistent Yes/No question. We also find that in minority cases, *ChainCQG* cannot handle questions requiring complex logic or reasoning to arrive at the target answer. A more powerful pre-training model might alleviate this issue. Our *ChainCQG* model sometimes includes additional details related to the answer, not contained in the gold question, which results in slightly more verbose, though consistent, questions.

5 Discussion and Ethical Issues

The results presented here demonstrate the efficacy of modern Transformer-based architectures, and specifically *ChainCQG*, in producing conversational questions on a challenging dataset. While answer-aware QG is our focus here, we plan to expand this in future work to include answer-unaware and open-ended QG, multi-task NLG involving QA, and domain-specific dialogue simulation. The flexibility of the *ChainCQG* architecture lends itself well to each of these problems, as representations from different inputs and tasks can be shared easily between modules.

As for the practical implications of our QG work, a number of applications mentioned in previous sections could immediately take advantage of QG features, either for training QA models or generating user-facing questions. In the former setting, NLG models such as this risk polluting the training dataset with examples that are noisy or inconsistent with the target answer, which can cause deleterious effects at inference time. In the latter setting, NLG models may suffer from bias based on the training data generation process, which leads to misrepresentation of application domains and individual users. Additional work is required to understand the extent of these issues on real-world applications, and identify corrective measures to ensure model robustness and diversified training distributions.

6 Conclusion

In this paper, we introduce *ChainCQG*, an answer-aware Conversational Question Generation model that outperforms all baselines on both automatic and human evaluation metrics on the inverted CoQA dataset (e.g., BLEU-1 improvement of 48% and 28% with GPT medium compared to ReDR and CFNet, respectively). We have designed a two-stage GPT-2-based architecture that jointly learns passage and dialogue history representations via a flow propagation training method. *ChainCQG* produces high-quality questions in multi-turn dialogue, addressing previous SOTA issues such as question type fidelity, question-answer inconsistency and coreference misalignment. Finally, we have performed and presented extensive ablation studies for various aspects of our approach.

References

Layla El Asri, Hannes Schulz, Shikhar Sharma, Jeremie Zumer, Justin Harris, Emery Fine, Rahul Mehrotra, and Kaheer Suleman. 2017. Frames: A corpus for adding memory to goal-oriented dialogue systems. *arXiv preprint arXiv:1704.00057*.

Satanjeev Banerjee and Alon Lavie. 2005. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, pages 65–72.

Paweł Budzianowski, Tsung-Hsien Wen, Bo-Hsiang Tseng, Inigo Casanueva, Stefan Ultes, Osman Ramadan, and Milica Gašić. 2018. Multiwoz-a large-scale multi-domain wizard-of-oz dataset for task-oriented dialogue modelling. *arXiv preprint arXiv:1810.00278*.

Asli Celikyilmaz, Elizabeth Clark, and Jianfeng Gao. 2020a. Evaluation of text generation: A survey. *arXiv preprint arXiv:2006.14799*.

Asli Celikyilmaz, Elizabeth Clark, and Jianfeng Gao. 2020b. Evaluation of text generation: A survey.

Yu Chen, Lingfei Wu, and Mohammed J Zaki. 2019. Graphflow: Exploiting conversation flow with graph neural networks for conversational machine comprehension. *arXiv preprint arXiv:1908.00059*.

Eunsol Choi, He He, Mohit Iyyer, Mark Yatskar, Wentau Yih, Yejin Choi, Percy Liang, and Luke Zettlemoyer. 2018. Quac: Question answering in context. *arXiv preprint arXiv:1808.07036*.

Philip Clarkson and Tony Robinson. 1999. Towards improved language model evaluation measures. In *Sixth European Conference on Speech Communication and Technology*.

Xinya Du and Claire Cardie. 2017. Identifying where to focus in reading comprehension for neural question generation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2067–2073.

Xinya Du and Claire Cardie. 2018. Harvesting paragraph-level question-answer pairs from wikipedia. *arXiv preprint arXiv:1805.05942*.

Xinya Du, Junru Shao, and Claire Cardie. 2017. Learning to ask: Neural question generation for reading comprehension. *arXiv preprint arXiv:1705.00106*.

Yifan Gao, Piji Li, Irwin King, and Michael R. Lyu. 2019a. Interconnected question generation with coreference alignment and conversation flow modeling. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4853–4862, Florence, Italy. Association for Computational Linguistics.

Yifan Gao, Piji Li, Irwin King, and Michael R Lyu. 2019b. Interconnected question generation with coreference alignment and conversation flow modeling. *arXiv preprint arXiv:1906.06893*.

Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. 2020. The curious case of neural text degeneration. In *International Conference on Learning Representations*.

Hsin-Yuan Huang, Eunsol Choi, and Wen-tau Yih. 2018. Flowqa: Grasping flow in history for conversational machine comprehension. *arXiv preprint arXiv:1810.06683*.

Ying Ju, Fubang Zhao, Shijie Chen, Bowen Zheng, Xuefeng Yang, and Yunfeng Liu. 2019. Technical report on conversational question answering. *arXiv preprint arXiv:1909.10772*.

Yanghoon Kim, Hwanhee Lee, Joongbo Shin, and Kyomin Jung. 2019. Improving neural question generation using answer separation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 6602–6609.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

Jingjing Li, Yifan Gao, Lidong Bing, Irwin King, and Michael R. Lyu. 2019. Improving question generation with to the point context. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3216–3226, Hong Kong, China. Association for Computational Linguistics.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *International Conference on Learning Representations*.

Mao Nakanishi, Tetsunori Kobayashi, and Yoshihiko Hayashi. 2019. Towards answer-unaware conversational question generation. In *Proceedings of the 2nd Workshop on Machine Reading for Question Answering*, pages 63–71.

Yasuhito Ohsugi, Itsumi Saito, Kyosuke Nishida, Hisako Asano, and Junji Tomita. 2019. A simple but effective method to incorporate multi-turn context with bert for conversational machine comprehension. *arXiv preprint arXiv:1905.12848*.

Boyuan Pan, Hao Li, Ziyu Yao, Deng Cai, and Huan Sun. 2019. Reinforced dynamic reasoning for conversational question generation. In *Proceedings of the 57th Conference of the Association for Computational Linguistics*, pages 2114–2124.

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.

Peng Qi, Yuhao Zhang, and Christopher D. Manning. 2020. Stay hungry, stay focused: Generating informative and specific questions in information-seeking conversations.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. Squad: 100,000+ questions for machine comprehension of text. *arXiv preprint arXiv:1606.05250*.

Siva Reddy, Danqi Chen, and Christopher D Manning. 2019. Coqa: A conversational question answering challenge. *Transactions of the Association for Computational Linguistics*, 7:249–266.

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.

Sandeep Subramanian, Tong Wang, Xingdi Yuan, Saizheng Zhang, Yoshua Bengio, and Adam Trischler. 2017. Neural models for key phrase detection and question generation. *arXiv preprint arXiv:1706.04560*.

Yansen Wang, Chenyi Liu, Minlie Huang, and Liqiang Nie. 2018a. Learning to ask questions in open-domain conversational systems with typed decoders. *arXiv preprint arXiv:1805.04843*.

Yansen Wang, Chenyi Liu, Minlie Huang, and Liqiang Nie. 2018b. Learning to ask questions in open-domain conversational systems with typed decoders. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2193–2203, Melbourne, Australia. Association for Computational Linguistics.

Rainer Winkler, Sebastian Hobert, Antti Salovaara, Matthias Söllner, and Jan Marco Leimeister. 2020. Sara, the lecturer: Improving learning in online education with a scaffolding-based conversational agent. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, pages 1–14.

Qingyang Wu, Yichi Zhang, Yu Li, and Zhou Yu. 2019. Alternating roles dialog model with large-scale pre-trained language models. *arXiv*, pages arXiv–1910.

Xiuyu Wu, Nan Jiang, and Yunfang Wu. 2020. A question type driven and copy loss enhanced framework for answer-agnostic neural question generation. *arXiv preprint arXiv:2005.11665*.

Yi-Ting Yeh and Yun-Nung Chen. 2019. Flowdelta: Modeling flow information gain in reasoning for conversational machine comprehension. *arXiv preprint arXiv:1908.05117*.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019. Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675*.

Wei Zhao, Maxime Peyrard, Fei Liu, Yang Gao, Christian M Meyer, and Steffen Eger. 2019. Moverscore: Text generation evaluating with contextualized embeddings and earth mover distance. *arXiv preprint arXiv:1909.02622*.

Yao Zhao, Xiaochuan Ni, Yuanyuan Ding, and Qifa Ke. 2018. Paragraph-level neural question generation with maxout pointer and gated self-attention networks. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3901–3910.

Qingyu Zhou, Nan Yang, Furu Wei, Chuanqi Tan, Hangbo Bao, and Ming Zhou. 2017. Neural question generation from text: A preliminary study. In *National CCF Conference on Natural Language Processing and Chinese Computing*, pages 662–671. Springer.

Chenguang Zhu, Michael Zeng, and Xuedong Huang. 2018. Sdnet: Contextualized attention-based deep network for conversational question answering. *arXiv preprint arXiv:1812.03593*.