# Interconnected Question Generation with Coreference Alignment and Conversation Flow Modeling

**Yifan Gao**[1*]     **Piji Li**[2]     **Irwin King**[1]     **Michael R. Lyu**[1]

[1] Department of Computer Science and Engineering,
The Chinese University of Hong Kong, Shatin, N.T., Hong Kong
[2] Tencent AI Lab
[1]{yfgao,king,lyu}@cse.cuhk.edu.hk  [2]pijili@tencent.com

## Abstract

We study the problem of generating interconnected questions in question-answering style conversations. Compared with previous works which generate questions based on a single sentence (or paragraph), this setting is different in two major aspects: (1) Questions are highly conversational. Almost half of them refer back to conversation history using coreferences. (2) In a coherent conversation, questions have smooth transitions between turns. We propose an end-to-end neural model with coreference alignment and conversation flow modeling. The coreference alignment modeling explicitly aligns coreferent mentions in conversation history with corresponding pronominal references in generated questions, which makes generated questions interconnected to conversation history. The conversation flow modeling builds a coherent conversation by starting questioning on the first few sentences in a text passage and smoothly shifting the focus to later parts. Extensive experiments show that our system outperforms several baselines and can generate highly conversational questions. The code implementation is released at https://github.com/Evan-Gao/conversaional-QG.

## 1 Introduction

Question Generation (QG) aims to create human-like questions from a range of inputs, such as natural language text (Heilman and Smith, 2010), knowledge base (Serban et al., 2016) and image (Mostafazadeh et al., 2016). QG is helpful for the knowledge testing in education, i.e., the intelligence tutor system, where an instructor can actively ask questions to students given reading comprehension materials (Heilman and Smith, 2010; Du et al., 2017). Besides, raising good questions

---
* This work was partially done when Yifan Gao was an intern at Tencent AI Lab.

| Passage: Incumbent Democratic President Bill Clinton was ineligible to serve a third term due to term limitations in the 22nd Amendment of the Constitution, and Vice President Gore was able to secure the Democratic nomination with relative ease. Bush was seen as the early favorite for the Republican nomination and, despite a contentious primary battle with Senator John McCain and other candidates, secured the nomination by Super Tuesday. Bush chose ... | | |
|---|---|---|
| $Q_1$: What political party is Clinton a member of? | $A_1$: | Democratic |
| $Q_2$: What was he ineligible to serve? | $A_2$: | third term |
| $Q_3$: Why? | $A_3$: | term limitations |
| $Q_4$: Based on what amendment? | $A_4$: | 22nd |
| $Q_5$: Of what document? | $A_5$: | Constitution |
| $Q_6$: Who was his vice president? | $A_6$: | Gore |
| $Q_7$: Who was the early Republican favorite for the nomination? | $A_7$: | Bush |
| $Q_8$: Who was the primary battle with? | $A_8$: | John McCain |
| $Q_9$: What is his title? | $A_9$: | Senator |
| $Q_{10}$: When did Bush secure the nomination by? | $A_{10}$: | Tuesday |

Table 1: An example for conversational question generation from a conversational question answering dataset CoQA (Reddy et al., 2019). Each turn contains a question $Q_i$ and an answer $A_i$.

in a conversational can enhance the interactiveness and persistence of human-machine interactions (Wang et al., 2018).

Recent works on question generation for knowledge testing are mostly formalized as a standalone interaction (Yuan et al., 2017; Song et al., 2018), while it is a more natural way for human beings to test knowledge or seek information through conversations involving a series of interconnected questions (Reddy et al., 2019). Furthermore, the inability for virtual assistants to ask questions based on previous discussions often leads to unsatisfying user experiences. In this paper, we consider a new setting called **C**onversational **Q**uestion **G**eneration (**CQG**). In this scenario, a system needs to ask a series of interconnected questions grounded in a passage through a question-answering style conversation. Table 1 provides an example under this scenario. In this dialogue, a questioner and an answerer chat about the above passage. Every question after the first turn is dependent on the conversation history.

Considering that the goal of the task is to generate interconnected questions in conversational
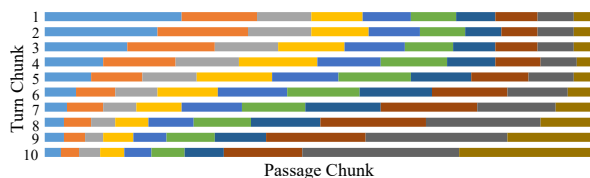
4853

Figure 1: Passage chunks of interest for each turn chunks. Each row contains 10 bands distinguished by different colors. Each band represents a passage chunk. The width of a passage chunk indicates the concentration of conversation in that turn. The $y$-axis indicates turn chunk number. Same passage chunks share the same color across different turn chunks. *(Best viewed in color)*

question answering, CQG is challenging in a few aspects. Firstly, a model should learn to generate conversational interconnected questions depending on the conversation so far. As shown in Table 1, $Q_3$ is a single word 'Why?', which should be 'Why was he ineligible to serve a third term?' in a standalone interaction. Moreover, many questions in this conversation refer back to the conversation history using coreferences (e.g., $Q_2$, $Q_6$, $Q_9$), which is the nature of questions in a human conversation. Secondly, a coherent conversation must have smooth transitions between turns (each turn contains a question-answer pair). We expect the narrative structure of passages can influence the conversation flow of our interconnected questions. We further investigate this point by conducting an analysis on our experiment dataset CoQA (Reddy et al., 2019). We first split passages and turns of QA pairs into 10 uniform chunks and identify passage chunks of interest for each turn chunk. Figure 1 portrays the conversation flow between passage chunks and turn chunks. We see that in Figure 1, a question-answering style conversation usually starts focusing on the first few chunks in the passage and as the conversation advances, the focus shifts to the later passage chunks.

Previous works on question generation employ attentional sequence-to-sequence models on the crowd-sourced machine reading comprehension dataset SQuAD (Rajpurkar et al., 2016). They mainly focus on generating questions based on a single sentence (or paragraph) and an answer phrase (Du et al., 2017; Sun et al., 2018; Zhao et al., 2018), while in our setting, our model needs to not only ask a question on the given passage (paragraph) but also make the questions conversational by considering the conversation his-

tory. Meanwhile, some researchers study question generation in dialogue systems to either achieve the correct answer through interactions (Li et al., 2017) or enhance the interactiveness and persistence of conversations (Wang et al., 2018). Although questions in our setting are conversational, our work is different from these because our conversations are grounded in the given passages rather than open-domain dialogues.

We propose a framework based on the attentional encoder-decoder model (Luong et al., 2015) to address this task. To generate conversational questions (first challenge), we propose a multi-source encoder to jointly encode the passage and the conversation so far. At each decoding timestep, our model can learn to focus more on the passage to generate content words or on the conversation history to make the question succinct. Furthermore, our coreference alignment modeling explicitly aligns coreferent mentions in conversation history (e.g. *Clinton* in $Q_1$ Table 1) with corresponding pronominal references in generated questions (e.g. *he* in $Q_2$), which makes generated questions interconnected to conversation history. The coreference alignment is implemented by adding extra supervision to bias the attention probabilities through a loss function. The loss function explicitly guides our model to resolve to the correct non-pronominal coreferent mentions in the attention distribution and generate the correct pronominal references in target questions. To make the conversations coherent (second challenge), we propose to model the conversation flow to transit focus inside the passage smoothly across turns. The conversation flow modeling achieves this goal via a flow embedding and a flow loss. The flow embedding conveys the correlations between number of turns and narrative structure of passages. The flow loss explicitly encourages our model to focus on sentences contain key information to generate the current turn question and ignore sentences questioned several turns ago.

In evaluations on a conversational question answering dataset CoQA (Reddy et al., 2019), we find that our proposed framework outperforms several baselines in both automatic and human evaluations. Moreover, the coreference alignment can greatly improve the precision and recall of generated pronominal references. The conversation flow modeling can learn the smooth transition of conversation flow across turns.
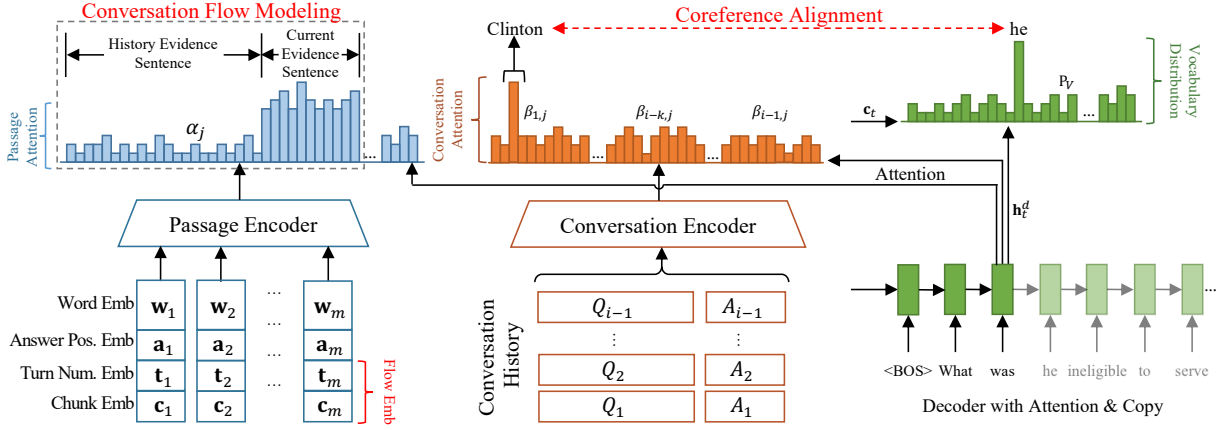
Figure 2: The framework of our proposed model. For clarity, we omit to plot the copy mechanism in the figure. *(Best viewed in color)*

## 2 Problem Setting

In this section, we define the Conversation Question Generation (CQG) task. Given a passage $P$, a conversation history $C_{i-1} = \{(Q_1, A_1), ..., (Q_{i-1}, A_{i-1})\}$ and the aspect to ask (the current answer $A_i$), the task of CQG is to generate a question $\overline{Q_i}$ for the next turn:

$$\overline{Q_i} = \arg\max_{Q_i} \text{Prob}(Q_i | P, A_i, C_{i-1}), \quad (1)$$

in which the generated question should be as conversational as possible.

Note that we formalize this setting as an answer-aware QG problem (Zhao et al., 2018), which assumes answer phrases are given before generating questions. Moreover, answer phrases are shown as text fragments in passages. Similar problems have been addressed in (Du and Cardie, 2018; Zhao et al., 2018; Sun et al., 2018). Our problem setting can also be generalized to the answer-ignorant case. Models can identify which answers to ask first by combining question-worthy phrases extraction methods (Du and Cardie, 2018; Wang et al., 2019).

## 3 Model Description

As shown in Figure 2, our framework consists of four components: (1) multi-source encoder; (2) decoder with copy mechanism; (3) coreference alignment; (4) conversation flow modeling.

### 3.1 Multi-Source Encoder

Since a conversational question is dependent on a certain aspect of the passage $P$ and the conversation context $C_{i-1}$ so far, we jointly encode in-

formation from two sources via a passage encoder and a conversation encoder.

**Passage Encoder.** The passage encoder is a bidirectional-LSTM (bi-LSTM) (Hochreiter and Schmidhuber, 1997), which takes the concatenation of word embeddings $\mathbf{w}$ and answer position embeddings $\mathbf{a}$ as input $\mathbf{x}_i = [\mathbf{w}_i; \mathbf{a}_i]$. We denote the answer span using the typical BIO tagging scheme and map each token in the paragraph into the corresponding answer position embedding (i.e., B_ANS, I_ANS, O). Then the whole passage can be represented using the hidden states of the bi-LSTM encoder, i.e., $(\mathbf{h}_1^p, ..., \mathbf{h}_m^p)$, where $m$ is the sequence length.

**Conversation Encoder.** The conversation history $C_{i-1}$ is a sequence of question-answer pairs $\{(Q_1, A_1), ..., (Q_{i-1}, A_{i-1})\}$. We use segmenters $<q><a>$ to concatenate each question answer pair $(Q, A)$ into a sequence of tokens $(<q>, q_1, ..., q_m; <a>, a_1, ..., a_m)$. We design a hierarchical structure to conduct conversation history modeling. We first employ a token level bi-LSTM to get contextualized representation of question-answer pairs $(\mathbf{h}_{i-k,1}^w, ..., \mathbf{h}_{i-k,m}^w)$, where $i - k$ is the turn number and $k \in [1, i)$. To model the dependencies across turns in the conversation history, we adopt a context level bi-LSTM to learn the contextual dependency $(\mathbf{h}_1^c, ..., \mathbf{h}_{i-1}^c)$ across different turns (denoted in the subscript $1, ..., i-1$) of question-answer pairs.

### 3.2 Decoder with Attention & Copy

The decoder is another LSTM to predict the word probability distribution. At each decoding timestep $t$, it reads the word embedding $\mathbf{w}_t$ and the

hidden state of previous timestep $\mathbf{h}_{t-1}^d$ to generate the current hidden state $\mathbf{h}_t^d = \text{LSTM}(\mathbf{w}_t, \mathbf{h}_{t-1}^d)$.

To generate a conversational question grounded in the passage, the decoder itself should decide to focus more on passage hidden states $\mathbf{h}_j^p$ or the hidden states of conversation history $\mathbf{h}_{i-k,j}^w$ at each decoding timestep. Therefore, we flat token level conversation hidden states $\mathbf{h}_{i,j}^w$ and aggregate the passage hidden states $\mathbf{h}_j^p$ with token level conversation hidden states $\mathbf{h}_{i,j}^w$ into a unified memory: $(\mathbf{h}_1^p, ..., \mathbf{h}_m^p; \ \mathbf{h}_{1,1}^w, ..., \mathbf{h}_{1,m}^w; \ ... ; \ \mathbf{h}_{i-1,1}^w, ..., \mathbf{h}_{i-1,m}^w)$, where $\mathbf{h}_{i,j}^w$ denotes the $j$-th token of the $i$-th turn in token level conversation hidden states. Then we attend the unified memory with the standard attention mechanism (Luong et al., 2015) for the passage attention $(\alpha_1, ..., \alpha_m)$ and the hierarchical attention mechanism for the conversation attention $(\beta_{1,1}, ..., \beta_{1,m}; ...; \beta_{i-1,1}, ..., \beta_{i-1,m})$:

$$e_j^p = \mathbf{h}_j^{p\top} \mathbf{W}_p \mathbf{h}_t^d, \tag{2}$$

$$e_{i-k,j}^w = \mathbf{h}_{i-k,j}^{w\top} \mathbf{W}_w \mathbf{h}_t^d, \tag{3}$$

$$e_{i-k}^c = \mathbf{h}_{i-k}^{c\top} \mathbf{W}_c \mathbf{h}_t^d, \tag{4}$$

$$\alpha_j = \frac{e_j^p}{e_{\text{total}}}, \ \beta_{i-k,j} = \frac{e_{i-k,j}^w * e_{i-k}^c}{e_{\text{total}}}, \tag{5}$$

where $e_{\text{total}} = \Sigma_j e_j^p + \Sigma_{k,j} e_{i-k,j}^w * e_{i-k}^c$ and $\mathbf{W}_p$, $\mathbf{W}_w$, $\mathbf{W}_c$ are learnable weights.

Finally, we derive the context vector $\mathbf{c}_t$ and the final vocabulary distribution $\text{P}_V$:

$$\mathbf{c}_t = \Sigma_j \alpha_j \mathbf{h}_j^p + \Sigma_{j,k} \beta_{i-k,j} \mathbf{h}_{i-k,j}^w,$$

$$\text{P}_V = \text{softmax}(\mathbf{W}_v(\tanh(\mathbf{W}_a[\mathbf{h}_t^d; \mathbf{c}_t]) + \mathbf{b}_v),$$

where $\mathbf{W}_v$, $\mathbf{W}_a$ are learnable weights. Please refer to See et al. (2017) for more details on the copy mechanism.

### 3.3 Coreference Alignment

Using coreferences to refer back is an essential property of conversational questions. Almost half of the questions contains explicit coreference markers such as *he*, *she*, *it* in CoQA (Reddy et al., 2019). Therefore, we propose the coreference alignment to enable our model such ability. Take $Q_2$ in Table 1 as an example, traditional question generation system can only generate question like "What was *Clinton* ineligible to serve?", while our system with coreference alignment can align the name "*Clinton*" to its pronominal reference "*he*" and generate a more conversational question "What was *he* ineligible to serve?".

The coreference alignment modeling tells the decoder to look at the correct non-pronominal coreferent mention in the conversation attention distribution to produce the pronominal reference word. We achieve this via two stages. In the pre-processing stage, given the conversation history $C_{i-1}$ and the question $Q_i$ which has a pronominal reference (e.g., *he* for $Q_2$ in Table 1), we first run a coreference resolution system (Clark and Manning, 2016) to find its coreferent mention $(w_1^c, ...w_m^c)$ (e.g. *Clinton*) in the conversation history $C_{i-1}$, where the superscript $c$ denotes tokens identified as the coreferent mention. During training, we introduce a novel loss function built on the conversation attention of coreferent mentions $\beta_i^c$ and the output word probability of its pronominal reference word $p_{\text{coref}} \in \text{P}_V$. As shown in Figure 2, when our model need to refer back to the coreferent mention, we ask the model focus correctly on the antecedent (e.g. *Clinton*) and maximize the probability of its pronominal reference (e.g. *he*) $p_{\text{coref}}$ in the output vocabulary distribution $\text{P}_V$,

$$\mathcal{L}_{\text{coref}} = -(\lambda_1 \log \frac{\Sigma_j \beta_j^c}{\Sigma_{k,j} \beta_{i-k,j}} + \lambda_2 \log p_{\text{coref}}) * s_c,$$

where $\lambda_1, \lambda_2$ are hyperparameters, $s_c$ is the confidence score between the non-pronominal coreferent mention and the pronoun obtained during the pre-processing stage.

### 3.4 Conversation Flow Modeling

Another key challenge in CQG is that a coherent conversation must have smooth transitions between turns. As illustrated in Figure 1, we find that as the conversations go on, most of the questioners transit their focus from the beginning of passages to the end. Following this direction, we model the conversation flow to learn smooth transitions across turns of the conversation.

**Flow Embedding.** As shown in Figure 2, we feed our model with the current turn number indicator in the conversation and the relative position for each token in the passage, which, intuitively, are useful for modeling the conversation flow. We achieve this goal via two additional embeddings. The turn number embedding is a learned lookup table $[\mathbf{t}_1, ..., \mathbf{t}_n]$ to map the turn number $i$ into its feature embedding space, where $n$ is the maximum turn we consider. For encoding the relative position of each token, we split the passage

into $L$ uniform chunks. Each token in the passage is mapped to its corresponding chunk embedding $[\mathbf{c}_1, ..., \mathbf{c}_L]$. The final input to the passage encoder is the concatenation of word embedding, answer position embedding (introduced in Section 3.1) and these two additional embeddings: $\mathbf{x}_i = [\mathbf{w}_i; \mathbf{a}_i; \mathbf{t}_i; \mathbf{c}_i]$.

We further add a gated self-attention modeling mechanism (Zhao et al., 2018) in the passage encoder. Motivating our use of self-attention we consider two desiderata. One is self-attention with answer position embedding can aggregate answer-relevant information from the whole passage for question generation. Another is we want to learn the latent alignment between the turn number embedding and the chunk embedding for better modeling the conversation flow. We first match the rich-feature enhanced passage representation $\mathbf{H}^p = [\mathbf{h}_1^p; ...; \mathbf{h}_m^p]$ with itself $\mathbf{h}_j^p$ to compute the self-matching representation $\mathbf{u}_j^{p}$, and then combine it with the original representation $\mathbf{h}_j^p$:

$$\mathbf{a}_j^p = \text{softmax}(\mathbf{H}^{p\top}\mathbf{W}_s\mathbf{h}_j^p), \ \mathbf{u}_j^p = \mathbf{H}^p\mathbf{a}_j^p \quad (6)$$

$$\mathbf{f}_j^p = \tanh(\mathbf{W}_f[\mathbf{h}_j^p; \mathbf{u}_j^p]), \quad (7)$$

The final representation $\tilde{\mathbf{h}}_j^p$ is derived via a gated summation through a learnable gate vector $\mathbf{g}_j^p$,

$$\mathbf{g}_t^p = \text{sigmoid}(\mathbf{W}_g[\mathbf{h}_j^p; \mathbf{u}_j^p]) \quad (8)$$

$$\tilde{\mathbf{h}}_j^p = \mathbf{g}_t^p \odot \mathbf{f}_j^p + (1 - \mathbf{g}_t^p) \odot \mathbf{h}_j^p \quad (9)$$

where $\mathbf{W}_s$, $\mathbf{W}_f$, $\mathbf{W}_g$ are learnable weights, $\odot$ is the element-wise multiplication. Self matching enhanced representation $\tilde{\mathbf{h}}_j^p$ takes the place of the passage representation $\mathbf{h}_j^p$ for calculating the passage attention.

**Flow Loss.** In Section 3.1, our answer position embedding can help model the conversation flow by showing the position of answer fragments inside the passage. However, it is still helpful to tell the model explicitly which sentences around the answer are of high informativity to generate the current turn question. The flow loss is designed to help our model to locate the evidence sentences correctly. Firstly, we define two kinds of sentences in the passage. If a sentence is informative to the current question, we call it `Current Evidence Sentence (CES)`. If a sentence is informative to questions in the conversation history and irrelevant to the current question, we call it `History Evidence Sentence (HES)`.

Then our model is taught to focus on current evidence sentences and ignore the history evidence sentences in the passage attention $\alpha_j$ via the following flow loss:

$$\mathcal{L}_{\text{flow}} = -\lambda_3\log\frac{\Sigma_{j:w_j\in\text{CES}}\alpha_j}{\Sigma_j\alpha_j} + \lambda_4\frac{\Sigma_{j:w_j\in\text{HES}}\alpha_j}{\Sigma_j\alpha_j}$$

where $\lambda_3, \lambda_4$ are hyperparameters, and $w_j \in$ CES/HES indicates the token $w_j$ is inside the sentence with a CES/HES label.

### 3.5 Joint Training

Considering all the aforementioned components, we define a joint loss function as:

$$\mathcal{L} = \mathcal{L}_{\text{nll}} + \mathcal{L}_{\text{coref}} + \mathcal{L}_{\text{flow}}, \quad (10)$$

in which $\mathcal{L}_{\text{nll}} = -\log \text{Prob}(Q_i|P, A_i, C_{i-1})$ is the the negative log-likelihood loss in the sequence to sequence learning (Sutskever et al., 2014).

## 4 Experiments

### 4.1 Dataset Preparation

We conduct experiments on the CoQA dataset (Reddy et al., 2019). It is a large-scale conversational question answering dataset for measuring the ability of machines to participate in a question-answering style conversation. The authors employ Amazon Mechanical Turk to collect 8k conversations with 127k QA pairs. Specifically, they pair two crowd-workers: a questioner and an answerer to chat about a passage. The answerers are asked to firstly highlight extractive spans in the passage as rationales and then write the free-form answers. We first extract each data sample as a quadruple of passage, question, answer and conversation history (previous $n$ turns of QA pairs) from CoQA. Then we filter out QA pairs with *yes*, *no* or *unknown* as answers (28.7% of total QA pairs) because there is too little information to generate the question to the point. Finally, we randomly split the dataset into a training set (80%, 66298 samples), a validation set (10%, 8409 samples) and a testing set (10%, 8360 samples). The average passage, question and answer lengths are 332.9, 6.3 and 3.2 tokens respectively.

### 4.2 Implementation Details

**Locating Extractive Answer Spans.** As studied by Yatskar (2018), abstractive answers in CoQA are mostly small modifications to spans occurring in the context. The maximum achievable

4857

performance by a model that predicts spans from the context is 97.8 F1 score. Therefore, we find the extractive spans from the passage which have the maximum F1 score with answers and treat them as answers for our answer position embedding.

**Number of Turns in Conversation History.** Reddy et al. (2019) find that in CoQA dataset, most questions in a conversation have a limited dependency within a bound of two turns. Therefore, we choose the number of history turns as $n = 3$ to ensure the target questions have enough conversation history information to generate and avoid introducing too much noise from all turns of QA pairs.

**Labeling Evidence Sentences.** As mentioned in Section 4.1, the crowd-workers label the extractive spans in the passage as rationales for actual answers. We treat sentences containing the rationale as `Current Evidence Sentence`.

**Model Settings.** We employ the teacher-forcing training, and in the generating stage, we set the maximum length for output sequence as 15 and block unigram repeated token, the beam size $k$ is set to 5. All hyperparameters and models are selected on the validation set and the results are reported on the test set.

### 4.3 Baselines and Ablations

We compare with the state-of-the-art baselines and conduct ablations as follows: **PGNet** is the pointer-generator network (See et al., 2017). We concatenate the passage $P$, the conversation history $C_{i-1}$ and the current answer $A_i$ as a sequence for the input. **NQG** (Du and Cardie, 2018) is similar to the previous one but it takes current answer features concatenated with the word embeddings during encoding. **MSNet** is our base model **M**ulti-**S**ource encoder decoder network (Section 3.1 & 3.2). **CorefNet** is our proposed **Coref**erence alignment model (Section 3.3). **FlowNet** is our proposed conversation **Flow** model (Section 3.4). **CFNet** is the model with both the **C**oreference alignment and the conversation **F**low modeling.

## 5 Results and Analysis

### 5.1 Main Results

Since the average length of questions is 6.3 tokens only, we employ BLEU (1-3) (Papineni et al., 2002) and ROUGE-L (R-L) (Lin, 2004) scores

| | B1 | B2 | B3 | R-L |
|---|---|---|---|---|
| PGNet | 28.84* | 13.74* | 8.16* | 39.18* |
| NQG | 35.56* | 21.14* | 14.84* | 45.58* |
| MSNet | 36.27* | 21.92* | 15.51* | 46.01* |
| CorefNet | 36.89 | 22.28 | 15.77 | 46.53 |
| FlowNet | 36.87 | 22.49 | 15.98 | 46.64 |
| CFNet | **37.38** | **22.81** | **16.25** | **46.90** |

Table 2: Main results of baselines and our models. *t*-test is conducted between our CFNet and baselines/ablations. (underline: *p-value* <0.05, *: *p-value* <0.01).

to evaluate n-gram similarity between the generated questions with the ground truth. We evaluate baselines and our models by predicting the current question given a passage, the current answer, and the ground truth conversation history.

Table 2 shows the main results, and we have the following observations:

- NQG outperforms PGNet by a large margin. The improvement shows that the answer position embedding (Zhou et al., 2017) is helpful for asking questions to the point.
- Our base model MSNet outperforms NQG, which reveals that the hierarchical encoding and the hierarchical attention to conversation history can model the dependency across different turns in conversations.
- Both our CorefNet and FlowNet outperform our base model. We will analyze the effectiveness of our coreference alignment and conversation flow modeling in the following two sections respectively.
- Our CFNet is significantly better than two baselines (PGNet, NQG), our MSNet, and our CorefNet. However, the difference between our CFNet and our FlowNet is not significant. This is because the conversation flow modeling improves all test samples while the coreference alignment contributes only to questions containing pronominal references.

### 5.2 Coreference Alignment Analysis

As we discussed in Section 3.3, it is the nature of conversational questions to use coreferences to refer back. In order to demonstrate the effectiveness of the proposed coreference alignment, we evaluate models on a subset of the test set called coreference set. Each sample in the coreference set requires a pronoun resolution between the conversation history and the current question (e.g., $Q_2$, $Q_6$,

| | B1 | B2 | B3 | R-L | P | R | F |
|---|---|---|---|---|---|---|---|
| PGNet | 27.66* | 13.82* | 8.96* | 38.40* | 26.87* | 25.17* | 25.68* |
| NQG | 34.75* | 21.52* | 15.96* | 45.04* | 34.46* | 32.97* | 33.25* |
| MSNet | 36.31* | 22.92 | 17.07 | 45.97* | 35.34* | 33.80* | 34.07* |
| CorefNet | 37.51 | 24.14 | 18.44 | 47.45 | 42.09 | 40.35 | 40.64 |

Table 3: Evaluation results on the coreference test set. Precision (P), Recall (R) and F-score (F) of predicted pronouns are also reported. Significant tests with *t*-test are conducted between CorefNet and models without the coreference alignment. (underline: *p-value* <0.05, *: *p-value* <0.01).

---

**Passage**: … however , mccain has a very different life story . he grew up in a navy family and was a <u>pilot</u> during the vietnam war in the 1960s …
**Conversation History:**

| <q> | what | war | was | mccain | in | ? |
|---|---|---|---|---|---|---|
| 0.0000 | 0.0001 | 0.0049 | 0.0138 | 0.7710 | 0.0055 | 0.0069 |

| <a> | vietnam | war |
|---|---|---|
| 0.0000 | 0.0140 | 0.0095 |

| <q> | was | he | in | the | army | ? |
|---|---|---|---|---|---|---|
| 0.0000 | 0.0045 | 0.1303 | 0.0005 | 0.0139 | 0.0001 | 0.0250 |

| <a> | no |
|---|---|
| 0.0000 | 0.0000 |

**Question** (Human): what was his job ?
**Question** (Our Model): what was his job ?

**Passage**: … incumbent democratic president bill clinton was ineligible to <u>serve a third term</u> due to term limitations in the 22nd amendment of the constitution …
**Conversation History:**

| <q> | what | political | party | is | clinton | a |
|---|---|---|---|---|---|---|
| 0.0000 | 0.0000 | 0.0002 | 0.0063 | 0.0045 | 0.9260 | 0.0430 |

| member | of | ? | <a> | democratic |
|---|---|---|---|---|
| 0.0008 | 0.0006 | 0.0026 | 0.0000 | 0.0160 |

**Question** (Human): what was he ineligible to serve ?
**Question** (Our Model): what was he ineligible for ?

Figure 3: Examples for the coreference alignment model. We show the attention probability (renormalize to 1) when the CorefNet predicts a pronoun (red color in Question). The current answers are underlined in the passages. *(Best viewed in color)*

$Q_9$ in Table 1). In additional to the BLEU(1-3) and ROUGE-L metrics, we also calculate the Precision (P), Recall (R) and F-score (F) of pronouns in the generated questions with regard to pronouns in the ground truth questions.

The results are depicted in Table 3. With the help of the coreference alignment, CorefNet significantly improves the precision, recall, and f-score of the predicted pronouns. Moreover, the performance on n-gram overlapping metrics is also boosted. To gain more insights into how the coreference alignment model influence the generation process, in Figure 3, we visualize the conversation attention distribution $\beta_j$ at the timestep the model predicts a pronoun. The conversation history distribution $\beta_j$ is renormalized to $\Sigma_j \beta_j = 1$. All two examples show that our model put the highest attention probability on the coreferent

mentions (i.e. *McCain/Clinton*) when it generates the pronominal references (*his/he*). We can conclude that our coreference alignment model can align correct coreferent mentions to generate corresponding pronouns.

## 5.3 Conversation Flow Modeling Analysis

As discussed in Section 3.4, a coherent conversation should have smooth transitions between turns, and we design our model to follow the narrative structure of the passage. Figure 4 shows an example illustrating the transition of passage attention distribution $a_j$ (normalize to 1) during first 11 turns of a conversation. We see that the model transits its focus smoothly across the first 11 turns from the first sentence in the passage to later parts. Sometimes the model drills down with two questions for the same sentence such as turn 2 & 3, 4 & 5 and 10 & 11.

To quantitatively validate the effectiveness of our conversation flow modeling, we study the alignment between passage attention $\alpha_j$ and sentences of interest in the passage. Ideally, a successful model should focus on sentences of interest (i.e., Current Evidence Sentence) and ignore sentences questioned several turns ago (i.e., History Evidence Sentence). We validate this intuition by calculating $\Sigma_{j:w_j \in \text{CES}} \alpha_j$ and $\Sigma_{j:w_j \in \text{HES}} \alpha_j$ for all examples in test set. Results show that $\Sigma_{j:w_j \in \text{CES}} \alpha_j$ and $\Sigma_{j:w_j \in \text{HES}} \alpha_j$ for our model with conversation flow modeling are 0.9966 and 0.0010 on average, which demonstrates that our conversation flow modeling can locate the current evidence sentences precisely and ignore the history evidence sentence. For the model without the flow modeling (CorefNet), $\Sigma_{j:w_j \in \text{CES}} \alpha_j = 0.4093$, $\Sigma_{j:w_j \in \text{HES}} \alpha_j = 0.1778$, which proves our intuition in Section 3.4 that the answer position embedding cannot have comparable effects on the conversation flow modeling.

## 5.4 Human Evaluation

We randomly sample 93 questions with the associated passage and conversation history to conduct human evaluation. We hire 5 workers to evaluate the questions generated by PGNet, MSNet, and our CFNet. All models are evaluated in terms of following 3 metrics: "Grammaticality", "Answerability" and "Interconnectedness". "Grammaticality" measures the grammatical correctness and fluency of the generated questions. "Answerability" evaluates whether the generated question can be

annie s sister , julia , was having a birthday party in the afternoon . annie 's mother was going to bake the cake for the party . mother asked annie to help her bake the cake . they chose to make a chocolate cake with chocolate frosting . annie got the bowls and the ingredients they would need for the cake . she helped measure the flour , the sugar and the cocoa .

Turn number: ▮ : 2nd & 3rd ▮ : 4th & 5th ▮ : 6th ▮ : 7th & 8th ▮ : 9th ▮ : 10th & 11th

Figure 4: The transition of passage attention distribution between turns. Different colors are correspond to different turns. To show attention probability of different turns in one place, we only draw attention probability $\alpha_j > 0.1$ here. If two turns focus on the same sentence, we average the attention probability between them. *(Best viewed in color)*

| | Grammaticality | Answerability | Interconnectedness |
|---|---|---|---|
| PGNet | 2.74 | 1.39 | 1.59 |
| MSNet | 2.85 | 2.39 | 1.74 |
| CFNet | 2.89 | **2.74***  | **2.67*** |

Table 4: Manual evaluation results. All metrics are rated on a 1-3 scale (3 for the best). Two-tailed *t*-test results are shown for our CFNet compared to PGNet/MSNet. * indicates *p-value* <0.01.

answered by the current answer. "Interconnectedness" measures whether the generated questions are *conversational* or not. If a question refers back to the conversation history using coreference or is dependent on the conversation history such as incomplete questions 'Why?', 'Of what?', we define it as a *conversational* question. All metrics are rated on a 1-3 scale (3 for the best).

The results are shown in Table 4. All models achieve high scores on "Grammaticality", owing to the strong language modeling capability of neural models. MSNet and our CFNet perform well on "Answerability" while PGNet does not. This demonstrates our base model MSNet and our CFNet can ask questions to the point. Finally, our CFNet outperforms the other two models in terms of "Interconnectedness" by a large gap, which proves that the proposed coreference alignment and conversation flow modeling can effectively make questions *conversational*.

## 6 Related Work

The task of Question Generation (QG) aims at generating natural questions from given input contexts. Some template-based approaches (Vanderwende, 2007; Heilman and Smith, 2010) were proposed initially, where well-designed rules and heavy human labor are required for declarative-to-interrogative sentence transformation. With the rise of data-driven learning approach and sequence

to sequence (seq2seq) framework (Sutskever et al., 2014), Du et al. (2017) first formulate QG as a seq2seq problem with attention mechanism. They extract sentences and pair them with questions from SQuAD (Rajpurkar et al., 2016), a large-scale reading comprehension dataset. Recent works along this line focus on how to utilize the answer information better to generate questions to the point (Zhou et al., 2017; Gao et al., 2019b; Sun et al., 2018), how to generate questions with specific difficulty levels (Gao et al., 2019a) and how to effectively use the contexts in paragraphs to generate questions that cover context beyond a single sentence (Zhao et al., 2018; Du and Cardie, 2018).

In parallel to question generation for reading comprehension, some researchers recently investigate question generation in dialogue systems. Li et al. (2017) show that asking questions through interactions can receive useful feedbacks to reach the correct answer. Wang et al. (2018) consider asking questions in open-domain conversational systems with typed decoders to enhance the interactiveness and persistence of conversations.

In this paper, we propose a new setting which is related to the above two lines of research. We consider asking questions grounded in a passage via a question-answering style conversation. Since the questions and answers are in the format of a conversation, questions in our setting are highly conversational and interconnected to conversation history. This setting is challenging because we need to jointly model the attention shifting in the passage and the structure of a conversation (Grosz and Sidner, 1986). A limitation of the conversation in our setting is that we can only generate a series of interconnected questions according to predefined answers but in a real dialog the questioner can ask different questions according to the answers' response.

# 7 Conclusion and Future Work

In this paper, we study the problem of question-answering style Conversational Question Generation (CQG), which has never been investigated before. We propose an end-to-end neural model with coreference alignment and conversation flow modeling to solve this problem. The coreference alignment enables our framework to refer back to the conversation history using coreferences. The conversation flow modeling builds a coherent conversation between turns. Experiments show that our proposed framework achieves the best performance in automatic and human evaluations.

There are several future directions for this setting. First, the presented system is still contingent on highlighting answer-like nuggets in the declarative text. Integrating answer span identification into the presented system is a promising direction. Second, in our setting, the roles of the questioner and the answerer are fixed. However, questions can be raised by either part in real scenario.

## Acknowledgments

## References

Kevin Clark and Christopher D. Manning. 2016. Deep reinforcement learning for mention-ranking coreference models. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2256–2262, Austin, Texas. Association for Computational Linguistics.

Xinya Du and Claire Cardie. 2018. Harvesting paragraph-level question-answer pairs from Wikipedia. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1907–1917, Melbourne, Australia. Association for Computational Linguistics.

Xinya Du, Junru Shao, and Claire Cardie. 2017. Learning to ask: Neural question generation for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1342–1352, Vancouver, Canada. Association for Computational Linguistics.

Yifan Gao, Lidong Bing, Wang Chen, Michael R. Lyu, and Irwin King. 2019a. Difficulty controllable generation of reading comprehension questions. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19*. International Joint Conferences on Artificial Intelligence Organization.

Yifan Gao, Lidong Bing, Piji Li, Irwin King, and Michael R. Lyu. 2019b. Generating distractors for reading comprehension questions from real examinations. In *AAAI Conference on Artificial Intelligence*.

Barbara J. Grosz and Candace L. Sidner. 1986. Attention, intentions, and the structure of discourse. *Computational Linguistics*, 12(3):175–204.

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.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural Computation*, 9:1735–1780.

Jiwei Li, Alexander H. Miller, Sumit Chopra, Marc'Aurelio Ranzato, and Jason Weston. 2017. Learning through dialogue interactions by asking questions. In *ICLR*.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out: Proceedings of the ACL-04 Workshop*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Thang Luong, Hieu Pham, and Christopher D. Manning. 2015. Effective approaches to attention-based neural machine translation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1412–1421, Lisbon, Portugal. Association for Computational Linguistics.

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

*40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. 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.

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

Abigail See, Peter J. Liu, and Christopher D. Manning. 2017. Get to the point: Summarization with pointer-generator networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1073–1083, Vancouver, Canada. Association for Computational Linguistics.

Iulian Vlad Serban, Alberto García-Durán, Caglar Gulcehre, Sungjin Ahn, Sarath Chandar, Aaron Courville, and Yoshua Bengio. 2016. Generating factoid questions with recurrent neural networks: The 30M factoid question-answer corpus. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 588–598, Berlin, Germany. Association for Computational Linguistics.

Linfeng Song, Zhiguo Wang, Wael Hamza, Yue Zhang, and Daniel Gildea. 2018. Leveraging context information for natural question generation. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 569–574, New Orleans, Louisiana. Association for Computational Linguistics.

Xingwu Sun, Jing Liu, Yajuan Lyu, Wei He, Yanjun Ma, and Shi Wang. 2018. Answer-focused and position-aware neural question generation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3930–3939, Brussels, Belgium. Association for Computational Linguistics.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. In *Advances in Neural Information Processing Systems 27*, pages 3104–3112. Curran Associates, Inc.

Lucy Vanderwende. 2007. Answering and questioning for machine reading. In *AAAI Spring Symposium: Machine Reading*.

Siyuan Wang, Zhongyu Wei, Zhihao Fan, Yang Liu, and Xuanjing Huang. 2019. A multi-agent communication framework for question-worthy phrase extraction and question generation. In *AAAI Conference on Artificial Intelligence*.

Yansen Wang, Chenyi Liu, Minlie Huang, and Liqiang Nie. 2018. 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.

Mark Yatskar. 2018. A qualitative comparison of coqa, squad 2.0 and quac. *CoRR*, abs/1809.10735.

Xingdi Yuan, Tong Wang, Caglar Gulcehre, Alessandro Sordoni, Philip Bachman, Saizheng Zhang, Sandeep Subramanian, and Adam Trischler. 2017. Machine comprehension by text-to-text neural question generation. In *Proceedings of the 2nd Workshop on Representation Learning for NLP*, pages 15–25, Vancouver, Canada. Association for Computational Linguistics.

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, Brussels, Belgium. Association for Computational Linguistics.

Qingyu Zhou, Nan Yang, Furu Wei, Chuanqi Tan, Hangbo Bao, and Ming Zhou. 2017. Neural question generation from text: A preliminary study. In *Proceedings of the 6th CCF International Conference on Natural Language Processing and Chinese Computing (NLPCC)*, pages 662–671, Dalian, China.