Exploring Question-Specific Rewards for Generating Deep Questions

Yuxi Xie[1] **Liangming Pan**[2,4*] **Dongzhe Wang**[3]
Min-Yen Kan[4] **Yansong Feng**[1]
[1]Wangxuan Institute of Computer Technology, Peking University
[2]NUS Graduate School for Integrative Sciences and Engineering
[3]ZhuiYi Technology, Singapore
[4]School of Computing, National University of Singapore, Singapore
`xieyuxi@pku.edu.cn, e0272310@u.nus.edu, wdzethan2010@gmail.com`
`kanmy@comp.nus.edu.sg, fengyansong@pku.edu.cn`

Abstract

Recent question generation (QG) approaches often utilize the sequence-to-sequence framework (Seq2Seq) to optimize the log likelihood of ground-truth questions using teacher forcing. However, this training objective is inconsistent with actual question quality, which is often reflected by certain global properties such as whether the question can be answered by the document. As such, we directly optimize for QG-specific objectives via reinforcement learning to improve question quality. We design three different rewards that target to improve the fluency, relevance, and answerability of generated questions. We conduct both automatic and human evaluations in addition to thorough analysis to explore the effect of each QG-specific reward. We find that optimizing on question-specific rewards generally leads to better performance in automatic evaluation metrics. However, only the rewards that correlate well with human judgement (e.g., relevance) lead to real improvement in question quality. Optimizing for the others, especially answerability, introduces incorrect bias to the model, resulting in poorer question quality. The code is publicly available at https://github.com/YuxiXie/RL-for-Question-Generation.

1 Introduction

Question Generation (QG) aims to endow machines with the ability to ask relevant and to-the-point questions about a document. QG has important practical applications, such as generating assessments for course materials in education (Heilman and Smith, 2010; Lindberg et al., 2013), prompting user interaction in dialog systems (Shukla et al., 2019), enabling machines to ask clarification questions such as FAQs (Saeidi et al., 2018; Krishna and Iyyer, 2019), and automatically building large-scale QA datasets for the research community (Du et al., 2017; Zhao et al., 2018).

Recent QG approaches (Du et al., 2017; Zhao et al., 2018; Liu et al., 2019) have used Seq2Seq models with attention (Bahdanau et al., 2015), which feeds the input document into an encoder, and generates a question about the document through a decoder. The training objective is to maximize the log likelihood of the ground-truth question paired with each input document using teacher forcing (Williams and Zipser, 1989). However, as the ground-truth questions are insufficient to account for the many equivalent ways of asking a question, this likelihood-based training suffers from the problem of exposure bias (Ranzato et al., 2016); *i.e.*, the model does not learn how to distribute probability mass over sequences that are valid but different from the ground truth. To address this issue, previous QG works proposed to optimize the model directly on *question-specific rewards* via Reinforcement Learning (RL). This process decouples the training procedure from the ground truth data, so that the space of possible questions can be better explored. Moreover, it allows the training to target on specific properties we want the question to exhibit, such as relevant to a specific topic or answerable by the document. Although various rewards have been employed for QG — such as BLEU (Kumar et al., 2019), the answerability reward (Zhang and Bansal, 2019), and the word movers distance (Chen et al., 2020) — optimizing the reward scores does not always lead to higher question quality in practice, as observed by Hosking and Riedel (2019). How to define robust and effective QG-specific rewards still requires further investigation.

[*] Corresponding author.

We aim to analyze the effectiveness of question-specific rewards in QG. Instead of using general NLG metrics such as BLEU, we target three QG-related metrics that are commonly cited in human evaluations of question quality: (1) **Fluency** indicates whether the question follows the grammar and accords with the correct logic; (2) **Relevance** indicates whether the question is relevant to the document; and (3) **Answerability** indicates whether the question is answerable given the document. We design a specific RL reward for each metric: a language model based reward for fluency, a discriminator-based reward for relevance, and an QA-based reward for answerability. After optimizing each reward via RL, we conduct comprehensive analysis, including automatic and human evaluation, to arrive at the following conclusions: (1) both individual and joint optimization of these rewards can lead to performance gain in automated metrics such as BLEU, but this does not guarantee an improvement in the real question quality; (2) the reward for relevance substantially helps to improve the question quality, while the reward for answerability reduces the quality due to the bias brought by the QA model; and (3) a reward is more likely to improve the question quality if the reward score correlates well with human judgement.

2 Related Work

Early QG studies focused on using manually-designed rules or templates to transform a piece of given text to questions (Heilman, 2011; Chali and Hasan, 2012), with low generalizability and scalability. To address this, recent neural-based question generation (NQG) take advantage of the Seq2Seq with attention. These models are trained in an end-to-end manner, requiring far less labor and enabling better language flexibility. Many improvements have been made to the first Seq2Seq-based NQG model (Du et al., 2017), such as encoding answer information (Zhou et al., 2017; Sun et al., 2018) and incorporating linguistic features (Liu et al., 2019). A comprehensive survey of QG can be found in (Pan et al., 2019).

Among these attempts, utilizing RL to optimize QG-specific rewards has been adopted by recent works to address the exposure bias problem. To find a good proxy for question quality, various rewards have been proposed. One common type of reward is the similarity between the generated question and the reference question written by human. Kumar *et al.* (2019) adopted BLEU, ROUGE, and METEOR as rewards. Followup works employed more semantic-relevant metrics, such as the word movers distance (Chen et al., 2020; Yu et al., 2020) and the paraphrasing probability (Zhang and Bansal, 2019). To generate more passage-relevant questions, Kumar *et al.* (2019) designed a reward to measure the relevance between the input passage and the generated question based on their degree of overlapping. The *answerability* reward measures whether the generated question can be answered by the input passage. It is designed as either the confidence score that a pre-trained QA model can correctly answer the generated question (Zhang and Bansal, 2019), or the overlapping degree between the target answer and the answer predicted by the QA model (Yu et al., 2020). Other types of rewards include Yao *et al.* (2018), which train a discriminator to measure the *naturalness*, *i.e.*, the question is human-written or generated.

Most question-specific rewards are empirically successful since they achieve performance gain in automatic evaluation metrics after RL training. However, this brings several followup questions that existing works have failed to answer: (1) does optimizing RL rewards really improve the question quality from the human standard, (2) which reward is more effective in improving the question quality, and (3) how the rewards interfere with each other when jointly optimized. This paper aims to bridge this gap through human evaluation and analytic experiments, aiming to provide a better understanding of how different rewards affect the question generation process.

3 Methodology

Given a document $\mathcal{D}$ as input, the objective is to generate a relevant question $\hat{\mathcal{Y}}$ which can be answered by the document $\mathcal{D}$. This is formulated as maximizing the conditional probability $p(\mathcal{Y}|\mathcal{D})$:

$$\hat{\mathcal{Y}} = \arg\max_{\mathcal{Y}} P(\mathcal{Y}|\mathcal{D}) = \arg\max_{\mathcal{Y}} \prod_{t=1}^{T} P(y_t|\mathcal{D}, \mathcal{Y}_{<t}) \tag{1}$$

where y_t is the t-th token of the generated question $\mathcal{Y}$, and $\mathcal{Y}_{<t}$ represents the previous decoded tokens, *i.e.*, $y_1, \cdots, y_{t-1}$. The general framework of our model is shown in Figure 1, consisting of two parts: the

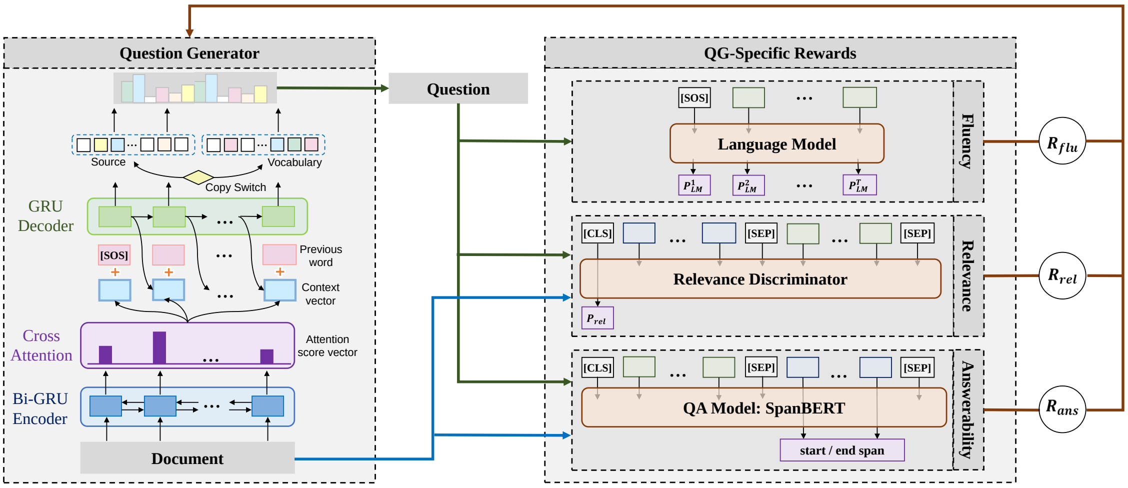


Figure 1: The framework of our model, consisting of the basic question generator (on the left) and the discriminators for QG-specific rewards (on the right). The blue sequence on the right represents the input document, and the green sequence is the generated question.

Question Generator and the *QG-specific Rewards*. The **Question Generator** uses the Seq2Seq framework with attention (Bahdanau et al., 2015), copying (Gu et al., 2016; See et al., 2017), and coverage mechanisms (Tu et al., 2016), following most existing NQG works. The model is trained by maximizing the likelihood of ground-truth questions. As discussed in the introduction, this basic question generator suffers from the exposure bias problem. Therefore, we design three **QG-Specific Rewards** aiming at evaluating the fluency, relevance, and answerablity of the question generated by the basic model. We then fine-tune the model by optimizing these rewards following the RL framework with a baseline (Rennie et al., 2017). In the following, we describe the design of the three QG-specific rewards in detail.

3.1 LM-based Reward for Fluency

The perplexity of a sentence under a well-trained Language Model (LM) usually serves as a good indicator of its fluency (Yang et al., 2018b). Therefore, we introduce a LM-based reward to improve the fluency of the generated question. We first pre-train a language model P_{LM} and then define the **fluency reward** $\mathcal{R}_{flu}$ of a generated question $\mathcal{Y}$ as its negative perplexity evaluated by P_{LM}, formulated as:

$$\mathcal{R}_{flu}(\mathcal{Y}) = -\exp(-\frac{1}{T}\sum_{t=1}^{T}\log P_{LM}(y_t|\mathcal{Y}_{<t})) \tag{2}$$

To optimize the fluency reward in training, we define the following loss function $\mathcal{L}_{flu}$.

$$\mathcal{L}_{flu} = -(\mathcal{R}_{flu}(\hat{\mathcal{Y}}) - \alpha_{flu})\frac{1}{T}\sum_{t=1}^{T}\log P_{QG}(\hat{y}_t|\mathcal{D}, \hat{\mathcal{Y}}_{<t}) \tag{3}$$

where $\hat{y}_t$ is the t-th token in the predicted question $\hat{\mathcal{Y}}$, which is sampled from the vocabulary distribution $P_{QG}(y_t|\mathcal{D}, Y_{<t})$ specified by the RNN decoder of the question generator. α_{flu} is a pre-defined negative perplexity, which is used as the baseline reward in the RL algorithm to stabilize the training process.

3.2 Discriminator-Based Reward for Relevance

We then design a classifier-based discriminator to judge whether the generated question is relevant to the input document. As shown in Figure 1, the discriminator is a binary classifier based on the pre-trained BERT (Devlin et al., 2019), which takes both the input document $\mathcal{D}$ and the generated question $\mathcal{Y}$ as inputs and outputs the probability that $\mathcal{Y}$ is relevant to the $\mathcal{D}$. To train the relevance discriminator, we use the human-written ground-truth questions $\mathcal{Y}_G$ for each document as the positive training data. For a document-question pair $(\mathcal{D}, \mathcal{Y}_G)$, we create the negative sample $\mathcal{Y}_N$ for $\mathcal{D}$ in the following three ways.

- **Question Swap.** We randomly select a ground-truth question from another document $\mathcal{D}'$ as the negative sample for the document $\mathcal{D}$.

- **Inter-Doc Entity Swap.** We create the negative sample $\mathcal{Y}_N$ by replacing the entity in the ground-truth question $\mathcal{Y}_G$ with another entity of the same type but does not occur in the document $\mathcal{D}$. This helps the discriminator to learn whether the question involves entities not mentioned in the document.

- **Intra-Doc Entity Swap.** We also replace the entity in the ground-truth question with a different entity from the same document. This often creates logical errors in the question, *e.g.*, *William Shakespeare is written by the book*, which is more challenging for the discriminator to differentiate.

Following the above process, we create three negative samples for each ground-truth question. To address the unbalance between positive and negative training data, we adopt the α-balanced focal loss (Lin et al., 2017) to train the relevance discriminator, given as follows.

$$\mathcal{L}_F(P_t) = -\alpha_t(1-P_t)^\lambda \log P_t \tag{4}$$

where P_t is the predicted probability for class t. $(1-P_t)^\lambda$ is a modulating factor with a tunable focusing parameter $\lambda \geq 0$ that smoothly adjusts the rate at which easy examples are down-weighted.

After training the relevance discriminator, we use it to obtain the relevance reward and then fine-tune the question generator by maximizing the relevance reward in RL training. Given a document $\mathcal{D}$ and a question $\mathcal{Y}$, the **relevance reward** $\mathcal{R}_{rel}(\mathcal{D},\mathcal{Y})$ is defined as a scaling of the relevance probability $P_{rel}(\mathcal{D},\mathcal{Y})$ output by the relevance discriminator as:

$$\mathcal{R}_{rel}(\mathcal{D},\mathcal{Y}) = -\log\left(1 - P_{rel}(\mathcal{D},\mathcal{Y}) + \epsilon\right) \tag{5}$$

where ϵ is a positive factor close to zero to avoid calculating $\log 0$. We scale the relevance probability in this way to augment the reward value for positive samples, *i.e.,* those samples whose rewards are greater than the baseline, because the QG model generally samples more negative samples during training. To optimize the relevance reward in RL training, we define the loss function $\mathcal{L}_{rel}$ as follows.

$$\mathcal{L}_{rel} = -(\mathcal{R}_{rel}(\mathcal{D},\hat{\mathcal{Y}}) - \alpha_{rel})\frac{1}{T}\sum_{t=1}^{T}\log P_{QG}(\hat{y}_t|\mathcal{D},\hat{\mathcal{Y}}_{<t}) \tag{6}$$

3.3 QA-Based Reward for Answerability

Answerability indicates whether the question is answerable by the document without the need of external information. We design the **answerability reward** based on the SpanBERT (Joshi et al., 2020), a state-of-the-art model for extractive QA. Given a document $\mathcal{D}$ and a question $\mathcal{Y}$ as inputs, SpanBERT predicts the start and end spans of the potential answer in the document $\mathcal{D}$. Formally, it outputs two probability distributions over the tokens in the document: P^s_{ans} and P^e_{ans}, where $P^s_{ans}(i)/P^e_{ans}(i)$ is the probability that the i-th token is the start/end span of the answer. Based on the pre-trained SpanBERT model, we first fine-tune it with the HotpotQA dataset (Yang et al., 2018a) and then use it to obtain the answerability reward for the generated question $\mathcal{Y}$. Intuitively, when the question is answerable, the model should be quite confident about the start/end span of the answer, so the distribution should be peak for both P^s_{ans} and P^e_{ans}, *i.e.*, the value of $\max_i P^s_{ans}(i)$ and $\max_j P^e_{ans}(j)$ are both large. Therefore, we use the geometric average of these two values to indicate the answerability, formulated as follows.

$$\mathcal{R}_{ans}(\mathcal{D},\mathcal{Y}) = -\log\big(1 - \max_{1\leq i\leq j\leq T,\ j-i\leq l}\sqrt{P^s_{ans}(i|\mathcal{D},\mathcal{Y})\cdot P^e_{ans}(j|\mathcal{D},\mathcal{Y})} + \epsilon\big) \tag{7}$$

where l represents the maximum allowed length of the answer. Similar to Equation 5, we also scale the probability to balance positive and negative samples during training. Similar to previous sections, to optimize the answerability reward in RL training, we define a loss function $\mathcal{L}_{ans}$:

$$\mathcal{L}_{ans} = -(\mathcal{R}_{ans}(\mathcal{D},\hat{\mathcal{Y}}) - \alpha_{ans})\frac{1}{T}\sum_{t=1}^{T}\log P_{QG}(\hat{y}_t|\mathcal{D},\hat{\mathcal{Y}}_{<t}) \tag{8}$$

3.4 Model Training

We train the whole model following the pre-training and fine-tuning paradigm, following (Hosking and Riedel, 2019). We first pre-train the question generation model by minimizing the cross-entropy loss together with the copying loss and the coverage loss, which can be written together as $\mathcal{L}_{base}$:

$$\mathcal{L}_{base} = \frac{1}{T}\sum_{t=1}^{T}(-\log P_{QG}(\hat{y}_t|\mathcal{D},\mathcal{Y}_{<t}) + \gamma_{cov}\sum_i \min(a_i^t, c_i^t)) \tag{9}$$

where the copy mechanism is involved in the question generator P_{QG}, and a_i^t is the i^{th} element of the attention score vector over the document at time stamp t. We then fine-tune the basic QG model trained with $\mathcal{L}_{base}$ to maximize the previously defined QG-specific rewards. This is achieved by linearly combining $\mathcal{L}_{base}$ with the RL-based losses $\mathcal{L}_{flu}, \mathcal{L}_{rel}, \mathcal{L}_{ans}$, as follows.

$$\mathcal{L} = \mathcal{L}_{base} + \gamma_{flu}\mathcal{L}_{flu} + \gamma_{rel}\mathcal{L}_{rel} + \gamma_{ans}\mathcal{L}_{ans} \tag{10}$$

where the hyper-parameters γ_{flu}, γ_{rel} and γ_{ans} specify the trade-off between different kinds of rewards. Note that we empirically set baseline rewards α_{flu}, α_{rel}, and α_{ans} to reduce the variance of gradient estimation during RL training, as reflected in Equations 3, 6, and 8.

4 Experiments

We provide ancillary material about supplementary experiments on hyper-parameter sensitivity and result analysis at https://github.com/YuxiXie/RL-for-Question-Generation/doc.

We conduct experiments on HotpotQA (Yang et al., 2018a), containing ∼100K crowd-sourced questions paired with Wikipedia articles. Generating a fluent, relevant, and answerable question in HotpotQA is a non-trivial task as it requires reasoning over different pieces of information in the input document. We follow the data split of Pan *et al.* (2020) to get 90,440 / 6,072 examples for training and testing, respectively. We further hold out 6,072 examples from the training data as the development set.

The basic question generator is a Seq2Seq framework with copying (Gu et al., 2016), coverage (See et al., 2017), and attention (Hou et al., 2019) mechanisms. We employ a 1-layer bi-directional GRU as the encoder and a 1-layer GRU as the decoder. We use the cased WordPiece tokenizer for the question generator following Joshi *et al.* (2020). The hidden size of the Seq2Seq model and the maximal input sequence length are set as 512 and 256, respectively.

To train the language model used for evaluating the fluency reward, we fine-tune the pre-trained BERT model (Devlin et al., 2019) on our target dataset, resulting in an LM with a perplexity of 8.85 on the dev set. Our relevance discriminator, which is also fine-tuned from the pre-trained BERT model, achieves a 91.16 F_1 score. The answerability discriminator based on SpanBERT-large obtains a 70.60 Exact Match (EM) score and an 83.44 F1 score on the HotpotQA development set. In RL training, we empirically set the baseline rewards α_{flu}, α_{rel} and α_{ans} as -10, $\log(2)$, and $\log(2)$, respectively. When jointly training all the rewards, the trade-off parameters γ_{cov}, γ_{flu}, γ_{rel} and γ_{ans} are tuned on the dev set and set to 0.25, 0.2, 1, 1, respectively. Other settings for training follow the standard best practice[1].

4.1 Automatic Evaluation

To investigate the effect of different QG-specific rewards, we first report the performance of automatic evaluation metrics when optimizing different rewards. The metrics include: a) Perplexity (PPL); b) BLEU 1 and 4 (Papineni et al., 2002), METEOR (Lavie and Agarwal, 2007), and ROUGE-L (Lin, 2004), which are based on the n-gram similarity between the generated questions and the ground truth; and c) gain on reward scores (the fluency, relevance, answerability rewards) after RL training. Table **??** summarizes the performance, where B1 is the basic question generator without RL training. The other models are fine-tuned based on B1 by optimizing either a single (S1–S3) or multiple rewards together

[1] All models are trained using Adam (Kingma and Ba, 2015) with mini-batch size 64. The learning rate is initially set to 10^{-3}, and adaptive learning rate decay applied. We also adopt early stopping and use gradient clipping (Pascanu et al., 2013) with clip norm of 5.

Model	Rewards			Metrics							
	F	R	A	PPL	BLEU1	BLEU4	Meteor	Rouge-L	R-FLU	R-REL	R-ANS
B1. Baseline				12.14	33.68	13.46	**21.39**	35.06	–	–	–
S1. F	√			11.91	37.59	15.22*	19.49	35.08	+1.48	+0.49	+0.03
S2. R		√		**11.63**	36.33	14.83*	20.63	**35.58**	+1.06	**+0.61**	+0.04
S3. A			√	12.07	36.40	13.95*	18.73	34.07	+1.30	+0.18	+0.21
E1. F + R	√	√		11.84	37.82	15.30*	19.95	35.48	+1.30	+0.60	+0.03
E2. R + A		√	√	12.43	35.77	14.46	20.53	35.26	+0.78	+0.49	+0.36
E3. F + A	√		√	12.30	**38.30**	14.99	18.02	34.50	**+1.71**	+0.40	**+0.51**
E4. F + R + A	√	√	√	12.35	37.97	**15.41***	19.61	35.12	+1.57	**+0.61**	+0.49

Table 1: The QG performance evaluated by automatic metrics when separately or jointly optimizing for various rewards. The last three columns show the change of reward scores compared with B1, where **R-FLU** is the fluency, **R-REL** the relevance, and **R-ANS** the answerability rewards. * denotes that the improvement on BLEU4 of the current model over B1 is statistically significant for $p < 0.01$.

(E1–E4). F, R, and A represents the fluency, relevance reward, and answerability rewards, respectively. We make four major observations:

1. Optimizing a single reward alone (F, R, A) can lead to an improvement on the BLEU score and also its corresponding reward score (F→R-FLU, R→R-REL, and A→R-ANS). When optimizing one reward, the scores for the other two also slightly increase, showing that the three rewards are correlated. This is in line with our intuition; *e.g.*, a question answerable by the passage is also likely to be fluent.

2. Jointly training multiple rewards in general leads to better performance. For example, the best improvement of R-REL, R-FLU and R-ANS are achieved by E3 and E4. This shows that different rewards can mutually enhance each other in joint training, which provides a prospective future direction on RL reward integration.

3. In general, the increase in rewards do not correlate well with improvement on automatic metrics. For example, E3 has the largest reward gain in fluency and answerability, but achieves relatively worse results in BLEU4, METEOR and ROUGE-L. This shows that the RL rewards focus on different parts of the question quality other than the n-gram based similarity with the ground truth. We further investigate how each reward affects the question quality later in Section 4.3.

4. We find that our B1 baseline tends to generate longer questions (the average question length is 1.44 times that of the ground truth, compared with 1.13 for E4). The RL rewards thus encourage shortening to lengths which are closer to the ground truth. This explains why the improvements brought by RL rewards are especially significant on BLEU.

4.2 Comparison with Baselines

We then compare our best performing model (E4. F + R + A) against several strong baselines in QG. The technologies employed by each model as well as the performance results are summarized in Table 2. Without using the answer information and any external linguistic knowledge, our model achieves a comparative BLEU4 with the state-of-the-art QG model (B7) in HotpotQA. This demonstrates the effectiveness of optimizing QG-specific rewards via RL. Surprisingly, the CGC-QG (B6) model exhibits an unusual pattern, achieving the best METEOR and ROUGE-L, but worst BLEU1 among all baselines. Our analysis finds that CGC-QG tends to generate irrelevant word during word-level content selection, leading to lengthy questions that are unanswerable or which contain semantic errors (Pan et al., 2020).

4.3 Human Evaluation

To further investigate whether optimizing QG-specific rewards leads to real improvement in question quality, we conduct human evaluation on the generated questions for 200 randomly-sampled testing documents. We ask 6 workers to rate the questions generated by 5 different models: the basic question generator (B1), the models fine-tuned with a single reward (S1, S2, S3), and the model with all three rewards (E4). Raters were blinded from the identity of the models. We designed the scale differently for each metric to ease human rating effort. For each question, we ask three workers to give ratings on four

Model	Features						Metrics			
	AE	LF	CP	CV	SA	RL	BLEU1	BLEU4	Meteor	Rouge-L
B2. Bahdanau et al. (2015)							32.97	11.81	18.19	33.48
B3. NQG++ (Zhou et al., 2017)		●	●				35.31	11.50	16.96	32.01
B4. Zhao et al. (2018)			●		●		35.36	11.85	17.63	33.02
B5. Zhao et al. (2018) + ans, cov	●		●	●	●		38.74	13.48	18.39	34.51
B6. CGC-QG (Liu et al., 2019)	●	●	●				31.18	14.36	**25.20**	**40.94**
B7. SG-DQG (Pan et al., 2020)	●	●	●	●			**40.55**	**15.53**	20.15	36.94
E4. Ours (F + R + A)			●	●		●	37.97	15.41	19.61	35.12

Table 2: Performance comparison. For all baselines, we use the reported performance from Pan *et al.* (2020). Legend: **AE**: answer encoding, **LF**: linguistic features, **CP**: copying mechanism, **CV**: coverage mechanism, **SA**: gated self-attention, **RL**: reinforcement learning.

Model	Flu. (1-5)	Rel. (1-3)	Ans. (0-1)	Cpx. (1-3)
B1. Baseline	3.98	2.77	0.67	**1.59**
S1. F	4.07	2.78	0.61	1.50
S2. R	**4.24**	**2.83**	**0.70**	1.51
S3. A	3.82	2.63	0.46	1.55
E4. F+R+A	4.10	2.72	0.53	1.52

Table 3: Human evaluation results for different methods. **Flu.**, **Rel.**, **Ans.**, and **Cpx.** denote the *Fluency*, *Relevance*, *Answerability*, and *Complexity*, respectively.

Question with Options

Q1. Whether this is a readable / understandable question?
○ yes ○ no

Q2. Which of the following errors occur in the question?
□ correct □ repetition □ incomplete □ ambiguous reference
□ incoherent □ wrong semantic collocation □ others

Q3. Whether this question is answerable by the passage?
○ yes ○ no

Q4. Why the question is not answerable?
□ invalid question □ ghost entity
□ no ghost entity, but information insufficient □ others

Q5. Whether this question require reasoning to answer?
○ yes, and very hard ○ yes, but simple reasoning ○ no

Table 4: Questionnaire designed for human evaluation, where ○ and □ indicate single-item and multiple-item selection, respectively.

criteria: *Fluency* (on a scale of 1–5), *Relevance* (scaled 1–3), *Answerability* (0 for unanswerable and 1 for answerable), and *Complexity* (scaled 1–3). To reduce the subjectivity of human rating, we obtain the rating score according to the annotator's answers to our designed questionnaire shown in Table 4. For more accurate evaluation, we give an unreadable question labeled by Q1 the lowest fluency rating and do not consider its relevance, answerability, and complexity ratings, as it is infeasible to judge them when the question is unreadable. The proportion of the unreadable questions generated by B1, S2, S2, S3 and E4 are 11.8%, 10.7%, 4.4%, 11.0% and 10.0%, respectively. For a readable question, the fluency rating is determined by the number of grammar errors it makes (the answer to Q2). The answerability and complexity ratings are given by the answers of Q3 and Q5, respectively. The relevance score depends on both Q3 and Q4. We average the scores from raters on each question, reporting the performance in Table 3. We discuss four major findings:

1. Human ratings do not correlate well with automatic evaluation metrics (BLEU4, Meteor, ROUGE-L), showing that the n-gram based metric is not a good reflection of actual question quality. Similar observations also exist in other language generation tasks (Callison-Burch et al., 2006; Liu et al., 2016; Novikova et al., 2017) for fluency, adequacy and coherence, validating our findings.

2. Optimizing the relevance reward (S2) alone leads to a substantial improvement of the human ratings for fluency, relevance, and answerability. Our further analysis in Section 4.5 shows that optimizing the relevance reward reduces ghost entity errors, a major source of error in previous QG models.

3. In contrast, optimizing for answerability (S3) has a surprising negative effect: reducing scores against all three human ratings, compared against the baseline (B1). We believe this is due to the immature of the current QA model in answering deep questions; *i.e.*, when used as a discriminator, the QA model we used cannot accurately predict whether a question is answerable or not, especially when the question involves reasoning (the case in HotpotQA). We analyse this in more depth in Section 4.4. We also show in Section 4.5 that the model tends to learn spurious correlations for answerability (*e.g.*, a *what*

year question is more likely to be answerable), which also accounts for its poor performance.

4. The average length of the generated questions are 1.44, 1.14, 1.25, 1.17, and 1.13 times compared with the ground-truth questions for B1, S1, S2, S3 and E4, respectively. This correlates with the human ratings for complexity (1.59, 1.50, 1.51, 1.55, and 1.52 for B1, S1, S2, S3 and E4 respectively). Longer questions tend to get higher complexity scores, which we feel validates intuition.

4.4 Consistency between Rewards and Human Judgement

To figure out why certain rewards improve the real question quality while others do not, we plot violins to show the distribution of reward scores on each level of human rating, shown in Figure 2.

We observe that the relevance reward has the highest consistency with human judgement; *i.e.*, both the median and the maximal rewards improves when the human rating gets higher. This provides an explanation of why optimizing the relevance reward leads to the best question quality.

The answerability reward predicted by the QA model, however, exhibits a poor correlation with the true answerability judged by humans. The median answerability reward is low for both answerable and unanswerable questions labeled by humans. This lends evidence for our claim in Section 4.3 that the innate capability of the QA model is the bottleneck for this reward. We expect the answerability reward to become more effective as deep QA improves, and could become a key component in future work.

The correlation between the fluency reward is also unsatisfactory: the increase of the fluency reward score is not obvious when the human rating for fluency increases. This makes the performance of S1 similar to the baseline model in Table 3; *i.e.*, the effect of optimizing the fluency reward is not obvious.

Based on the above observations, we conclude that the rewards that correlate well with human judgements tend to achieve real improvement in question quality. Therefore, to design an effective QG-specific reward, testing its performance on n-gram based metrics such as BLEU may not faithfully reflect its effectiveness. Instead, running an initial test of how well the reward score correlates with human judgement seems more viable.

We further provide a full view of how human ratings correlate to each other and how they correlate to the reward scores in Figure 3. We find that the relevance rating has strong correlations with both the fluency rating (0.79) and the answerability rating (0.67), indicating that a question is more relevant to the document when it is fluent and answerable. However, a relatively weak correlation exists between the fluency and answerability, meaning a fluent question is not necessarily answerable. In Figure 3(b), we further find that the relevance reward has a strong correlation with not only the relevance rating, but also fluency and answerability ratings. This explains why optimizing the relevance reward alone (S2) leads to improvements on fluency and answerability as well. In contrast, R-ANS has poor correlation with Flu. and Rel., explaining why it decreases the fluency and relevance ratings.

4.5 Mesoscopic Analysis of Generated Questions

To further understand why the fluency and the answerability reward fail to produce a consistent judgement with humans, we conduct a mesoscopic analysis on our E4 model by comparing the generated questions receiving high rewards with those with low rewards. We detail our observations for each reward type, guided by the results in Table 5.

• **Fluency.** From Table 5 Row F, we observe that sometimes the fluency reward is consistent with the human judgement on fluency; *e.g.*, the incomplete question [FL-1] receives a low reward. However, there is often inconsistency between the fluency judged by the language model and that of human-judged fluency. For example, [FH-2] has a repetition error but is assigned a high reward, while [FL-2] with a similar repetition error receives a low reward. This is caused by the statistical bias in the language model; *i.e.*, the LM tends to assign low rewards to the questions with rare or unseen entities (*e.g., Kenji Mizoguchi*). The lack of commonsense knowledge is another problem of the LM: *e.g.*, in [FH-1] the model fails to replace the word *born* with *founded* to make the question logically correct.

• **Relevance.** Table 5 Row R shows that the relevance discriminator judges the document–question relevance largely based on two aspects: 1) whether the question contains an entity that does not appear in the passage (ghost entity), *e.g.*, *Granly* in [RL-1], and 2) whether the question has a logical inconsistency

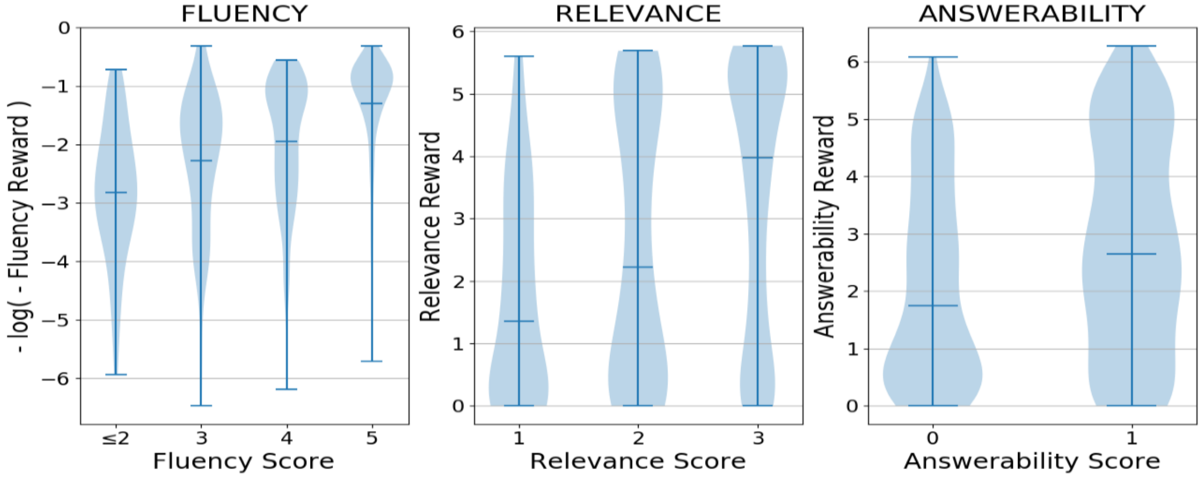
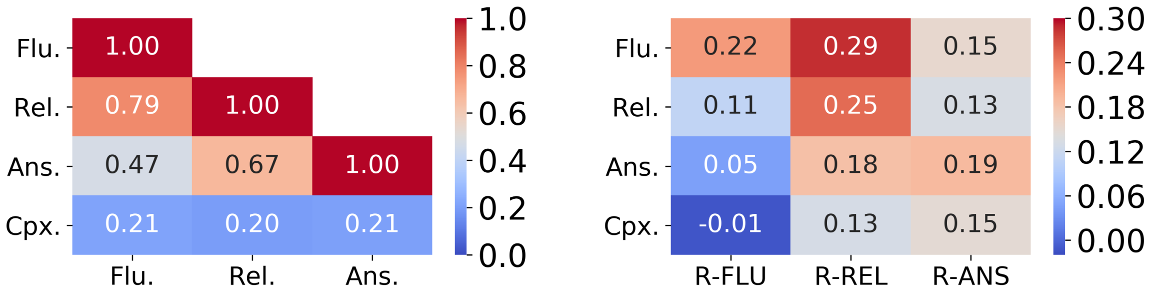


Figure 2: Correlation between reward scores and human ratings. Each sub-figure shows the distribution of reward scores on each level of human rating, where the hyphens in each column represent the minimal, median, and maximal values of the reward score.



(a) Correlation between human ratings

(b) Correlation between reward scores and human ratings

Figure 3: Heatmaps of the Pearson correlation coefficient matrices between human ratings and rewards. Flu., Rel., Ans. and Cpx. denote fluency, relevance, answerability, and complexity ratings in human evaluation, respectively. R-FLU, R-REL and R-ANS represent fluency, relevance and answerability rewards, respectively.

with the document, *e.g.*, [RL-2]. These two targets are quite consistent with the human judgement on relevance, which explains its good correlation in Figure 2. However, when the question is asking about an unmentioned aspect of an entity in the document, it is difficult for the model to assign an appropriate relevance score as in [RH-1]. A potential solution is to factor in the judgement of a good answerability discriminator (a challenge itself).

• **Answerability.** We observe in Row A that the answerability reward follows quite different criteria for whether a question can be answered compared against humans. First, most of the questions with high rewards are asking *what year* (the text highlighted in pink). We find that 45.0% questions generated by S3 are *what year* questions, compared with 11.2% for the baseline model B1. This may be caused by the data bias of the training set. Since a large portion of questions in HotpotQA are asking about date or time, this leads the QA model to learn a spurious correlation that a *what year* question is more likely to be answered and hence should receive high rewards. Second, when the question becomes complex, *i.e.* requiring the QA model to conduct reasoning such as comparison (Questions [AL-1] and [AL-2]) and to utilize world knowledge (*e.g.* *United States* is a *country*), the QA model tends to give a low answerability reward. This can be explained by the insufficient ability of current QA model in answering deep questions. To improve the answerability via a QA-based reward, we believe it is crucial to address the QA model's bias in prediction and improve its reasoning ability. Otherwise, optimizing an immature

		Samples with High / Low Rewards
F.	H.	**[FH-1] Que.** Eleven : A Music Company was born in what year ?
		[FH-2] Que. Dan Smith was born in what year ? Dan Smith
	L.	**[FL-1] Que.** Park Seo - joon starred in a South Korean television series that premiered on May 22 , 2017 every Monday and Tuesday where ?
		[FL-2] Que. Kenji Mizoguchi was born in what year ? Kenji Mizoguchi
R.	H.	**[RH-1] Doc.** " Sk8er Boi " is a song by the singer Avril Lavigne , released as the second single from her debut album , " Let Go " (2002) . **Que.** " Sk8er Boi " is a song written by a singer born in what year ?
		[RH-2] Doc. Roy Holder then appeared in " The Taming of the Shrew " (1967) , " Here We Go Round the Mulberry Bush " (1967) , " Romeo and Juliet " (1968) . . . The Taming of the Shrew is a 1967 film based on the play of the same name by William Shakespeare about a courtship between two strong - willed people . **Que.** Roy Holder appeared in a 1967 film based on the play of the same name by William Shakespeare about what?
	L.	**[RL-1] Doc.** Beitun , Xinjiang is a county - level city under the direct administration of the regional government . Wafangdian is one of the two northern county - level cities , the other being Zhuanghe , under the administration of Dalian , located in the south of Liaoning province , China . **Que.** Are both Granly and Wafangdian located in the same country ?
		[RL-2] Doc. In physics and engineering , the Fourier number or Fourier modulus , named after Joseph Fourier , is a dimensionless number that characterizes transient heat conduction . **Que.** Joseph Fourier was named after a man born in what year ?
A.	H.	**[AH-1] Doc.** The Worst Journey in the World was written and published in 1922 by a member of the expedition , Apsley Cherry – Garrard . **Que.** The Worst Journey in The Worst Journey was born in what year ?
		[AH-2] Doc. Weber ' s Store , at 510 Main St . in Thompson Falls in Sanders County (founded in 1905) , Montana was listed on the National Register of Historic Places in 1986 . **Que.** In what year was the county founded in which Terry ' s Store was listed on the National Register of Historic Places ?
	L.	**[AL-1] Doc.** John Stoltenberg is the former managing editor of " AARP The Magazine " , a bimonthly publication of the United States - based advocacy group AARP , a position John Stoltenberg held from 2004 until 2012 . AARP The Magazine is an American bi - monthly magazine , published by the American Association of Retired People , AARP , which focuses on aging issues . **Que.** John Stoltenberg is the former managing editor of a magazine published by which organization?
		[AL-2] Doc. 8 Spruce Street is a 76 - story skyscraper designed by architect Frank Gehry in the New York City . The original World Trade Center was a large complex of seven buildings in Lower Manhattan , New York City , United States . **Que.** 8 Spruce Street and the original World Trade Center , are located in which country ?

Table 5: Generated questions of E4, classified by high (H.) / low (L.) rewards. Colors indicate error types or patterns: red: grammatical and logical errors, blue: repetition, green: corresponding parts between documents and questions, purple: ghost entities and unmentioned parts, pink: patterns asking about *year*.

QA-based reward may introduce an incorrect bias, which in turn harms the question quality.

5 Conclusion

In this paper, we optimize three question-specific rewards via reinforcement learning on a Seq2Seq based question generator, aiming to improve the fluency, relevance and answerability of the generated question. Through comprehensive analytic experiments, including automatic and human evaluation, consistency validation, and meso analysis, we show that the effectiveness of a reward is poorly reflected by automatic evaluation metrics such as BLEU. Instead, we find a reward that correlates well with the human judgement generally has better effects on improving the question quality. In future works, we believe these observations can help to guide the design of other QG-specific rewards that target on unexplored aspects of question generation, such as the informativeness and the utility of questions.

Acknowledgements

This research is supported by the National Research Foundation, Singapore under its International Research Centres in Singapore Funding Initiative. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not reflect the views of National

Research Foundation, Singapore.

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. *CoRR*, abs/1409.0473.

Chris Callison-Burch, Miles Osborne, and Philipp Koehn. 2006. Re-evaluation the role of bleu in machine translation research. In *Conference of the European Chapter of the Association for Computational Linguistics (EACL)*.

Yllias Chali and Sadid A. Hasan. 2012. Towards automatic topical question generation. In *International Conference on Computational Linguistics (COLING)*, pages 475–492.

Yu Chen, Lingfei Wu, and Mohammed J. Zaki. 2020. Reinforcement learning based graph-to-sequence model for natural question generation. In *8th International Conference on Learning Representations, ICLR 2020*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: pre-training of deep bidirectional transformers for language understanding. In *Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL-HLT)*, pages 4171–4186.

Xinya Du, Junru Shao, and Claire Cardie. 2017. Learning to ask: Neural question generation for reading comprehension. In *Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 1342–1352.

Jiatao Gu, Zhengdong Lu, Hang Li, and Victor O.K. Li. 2016. Incorporating copying mechanism in sequence-to-sequence learning. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1631–1640.

Michael Heilman and Noah A. Smith. 2010. Good question! statistical ranking for question generation. In *Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL-HLT)*, pages 609–617.

Michael Heilman. 2011. Automatic factual question generation from text. *Language Technologies Institute School of Computer Science Carnegie Mellon University*, 195.

Tom Hosking and Sebastian Riedel. 2019. Evaluating rewards for question generation models. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2278–2283.

Ruibing Hou, Hong Chang, Bingpeng Ma, Shiguang Shan, and Xilin Chen. 2019. Cross attention network for few-shot classification. In Hanna M. Wallach, Hugo Larochelle, Alina Beygelzimer, Florence d'Alché-Buc, Emily B. Fox, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, 8-14 December 2019, Vancouver, BC, Canada*, pages 4005–4016.

Mandar Joshi, Danqi Chen, Yinhan Liu, Daniel S. Weld, Luke Zettlemoyer, and Omer Levy. 2020. Spanbert: Improving pre-training by representing and predicting spans. *Transactions of the Association for Computational Linguistics (TACL)*, 8:64–77.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In Yoshua Bengio and Yann LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015*.

Kalpesh Krishna and Mohit Iyyer. 2019. Generating question-answer hierarchies. In *Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 2321–2334.

Vishwajeet Kumar, Ganesh Ramakrishnan, and Yuan-Fang Li. 2019. Putting the horse before the cart: A generator-evaluator framework for question generation from text. In Mohit Bansal and Aline Villavicencio, editors, *Proceedings of the 23rd Conference on Computational Natural Language Learning, CoNLL 2019*, pages 812–821.

Alon Lavie and Abhaya Agarwal. 2007. METEOR: an automatic metric for MT evaluation with high levels of correlation with human judgments. In *Proceedings of the Second Workshop on Statistical Machine Translation (WMT@ACL)*, pages 228–231.

Tsung-Yi Lin, Priya Goyal, Ross B. Girshick, Kaiming He, and Piotr Dollár. 2017. Focal loss for dense object detection. In *IEEE International Conference on Computer Vision (ICCV)*, pages 2999–3007.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*.

David Lindberg, Fred Popowich, John C. Nesbit, and Philip H. Winne. 2013. Generating natural language questions to support learning on-line. In *European Workshop on Natural Language Generation (ENLG)*, pages 105–114.

Chia-Wei Liu, Ryan Lowe, Iulian Serban, Michael Noseworthy, Laurent Charlin, and Joelle Pineau. 2016. How NOT to evaluate your dialogue system: An empirical study of unsupervised evaluation metrics for dialogue response generation. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2122–2132.

Bang Liu, Mingjun Zhao, Di Niu, Kunfeng Lai, Yancheng He, Haojie Wei, and Yu Xu. 2019. Learning to generate questions by learning what not to generate. In *International World Wide Web Conference (WWW)*, pages 1106–1118.

Jekaterina Novikova, Ondrej Dusek, Amanda Cercas Curry, and Verena Rieser. 2017. Why we need new evaluation metrics for NLG. In Martha Palmer, Rebecca Hwa, and Sebastian Riedel, editors, *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, EMNLP 2017, Copenhagen, Denmark, September 9-11, 2017*, pages 2241–2252.

Liangming Pan, Wenqiang Lei, Tat-Seng Chua, and Min-Yen Kan. 2019. Recent advances in neural question generation. *CoRR*, abs/1905.08949.

Liangming Pan, Yuxi Xie, Yansong Feng, Tat-Seng Chua, and Min-Yen Kan. 2020. Semantic graphs for generating deep questions. In *Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 1463–1475.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 311–318.

Razvan Pascanu, Tomas Mikolov, and Yoshua Bengio. 2013. On the difficulty of training recurrent neural networks. In *Proceedings of the 30th International Conference on Machine Learning, ICML 2013*, volume 28 of *JMLR Workshop and Conference Proceedings*, pages 1310–1318.

Marc'Aurelio Ranzato, Sumit Chopra, Michael Auli, and Wojciech Zaremba. 2016. Sequence level training with recurrent neural networks. In Yoshua Bengio and Yann LeCun, editors, *4th International Conference on Learning Representations, ICLR 2016*.

Steven J. Rennie, Etienne Marcheret, Youssef Mroueh, Jerret Ross, and Vaibhava Goel. 2017. Self-critical sequence training for image captioning. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1179–1195.

Marzieh Saeidi, Max Bartolo, Patrick S. H. Lewis, Sameer Singh, Tim Rocktäschel, Mike Sheldon, Guillaume Bouchard, and Sebastian Riedel. 2018. Interpretation of natural language rules in conversational machine reading. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2087–2097.

Abigail See, Peter J. Liu, and Christopher D. Manning. 2017. Get to the point: Summarization with pointer-generator networks. In Regina Barzilay and Min-Yen Kan, editors, *Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 1073–1083.

Pushkar Shukla, Carlos E. L. Elmadjian, Richika Sharan, Vivek Kulkarni, Matthew Turk, and William Yang Wang. 2019. What should I ask? using conversationally informative rewards for goal-oriented visual dialog. In *Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 6442–6451.

Xingwu Sun, Jing Liu, Yajuan Lyu, Wei He, Yanjun Ma, and Shi Wang. 2018. Answer-focused and position-aware neural question generation. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3930–3939.

Zhaopeng Tu, Zhengdong Lu, Yang Liu, Xiaohua Liu, and Hang Li. 2016. Modeling coverage for neural machine translation. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 76–85.

Ronald J. Williams and David Zipser. 1989. A learning algorithm for continually running fully recurrent neural networks. *Neural Computation*, 1(2):270–280.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William W. Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. 2018a. Hotpotqa: A dataset for diverse, explainable multi-hop question answering. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2369–2380.

Zichao Yang, Zhiting Hu, Chris Dyer, Eric P. Xing, and Taylor Berg-Kirkpatrick. 2018b. Unsupervised text style transfer using language models as discriminators. In *Annual Conference on Neural Information Processing Systems (NeurIPS)*, pages 7298–7309.

Kaichun Yao, Libo Zhang, Tiejian Luo, Lili Tao, and Yanjun Wu. 2018. Teaching machines to ask questions. In Jérôme Lang, editor, *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI 2018, July 13-19, 2018, Stockholm, Sweden*, pages 4546–4552.

Jianxing Yu, Wei Liu, Shuang Qiu, Qinliang Su, Kai Wang, Xiaojun Quan, and Jian Yin. 2020. Low-resource generation of multi-hop reasoning questions. In *Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 6729–6739.

Shiyue Zhang and Mohit Bansal. 2019. Addressing semantic drift in question generation for semi-supervised question answering. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Yao Zhao, Xiaochuan Ni, Yuanyuan Ding, and Qifa Ke. 2018. Paragraph-level neural question generation with maxout pointer and gated self-attention networks. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 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 *CCF International Conference of Natural Language Processing and Chinese Computing (NLPCC)*, pages 662–671.