ECAI 2020
G.D. Giacomo et al. (Eds.)

doi:10.3233/FAIA200330

Semantics-Reinforced Networks for Question Generation

Zhuang Liu [1] and **Kaiyu Huang** [2] and **Degen Huang** [3] and **Jun Zhao** [4]

Abstract. Question Generation (QG) is the task of generating questions from a given document. Its aims to generate relevant and natural questions, answered by a given answer. However, existing approaches for QG usually fail to utilize the rich text structure that could complement the simple word sequence. Meantime, Cross-entropy based training has notorious limitations, such as exposure bias and inconsistency between train and test measurement. To address the issues, we propose a novel Reinforcement Learning (RL) based Semantics-Reinforced architecture, named **SiriQG**, for QG task. In SiriQG, we propose a hierarchical attention fusion network, for better modeling of both answer information and passage information by integrating explicit syntactic constraints into attention mechanism, and for better understanding the internal structure of the passage and the connection between answer, which makes it better to fuse different levels of granularity (i.e., passages and questions). Last, we also introduce a hybrid evaluator with using a mixed objective that combines both RL loss and cross-entropy loss to ensure the generation of semantically and syntactically question text. To evaluate the performance, we test our SiriQG model on well-known dataset for QG. Extensive experimental results demonstrated that proposed SiriQG can obtained a significant increase in accuracy comparing existing models based on public dataset, and it consistently outperformed all tested baseline models including the state-of-the-arts (SOTA) techniques.

1 INTRODUCTION

Question Generation (QG) is a very important yet challenging problem in NLP. The task is to syntactically generate correct, semantically sound and appropriate questions from various input formats, such as a structured database, text, or a knowledge base. More recently, neural network based techniques such as sequence-to-sequence (Seq2Seq) learning have achieved great success on various NLP tasks, including Question Generation. Recently, Learning to ask [4] proposes a Seq2Seq model with attention for question generation from text. [25] (in an approach referred to as QG) encoded ground-truth answers and employed bi-directional RNN in a Seq2Seq setting. Besides, they use the context matching and copy mechanism [23] to capture interactions between its context within the passage and the given ground-truth.

These models, however, do not make use of the rich text structure, such syntactic knowledge, which can help to complement question generation. Cross-entropy sequence training has infamous drawbacks including sensitivity bias and inconsistency between train and test measurement [14, 21, 29]. In order to overcome these problems, several recent QG models seek to optimize the evaluation metrics with using RL approach. However, the jointly mixed objective functions with both semantic and syntactic constraints to guide question generation are generally not considered. When generating a question, previous neural QG approaches did not take the answer information into account. Recent works [2, 11], have discussed different ways to use the answers to make the problems more important. However, the potential semantine relationships between the answer and passage are neglected, and the global interactions between them thus fail explicitly to model.

To address all of the aforementioned problems, we propose an RL based Semantics-Reinforced architecture, named **SiriQG**, for Question Generation task. The main contributions of this paper are three fold.

- First, we propose introduce an effective Hierarchical Attention Fusion network, for better modeling of both answer information and passage information by integrating explicit syntactic constraints into attention mechanism, and for better understanding the internal structure of the passage and the connection between answer, which makes it better able to fuse different levels of granularity;
- Second, we output a question using an LSTM Question Decoder. Also our hybrid evaluator is trained by optimizing a mixed objective function combining both RL loss and cross-entropy loss;
- Last, we conduct extensive experiments on well-known datasets for QG. Our proposal is end-to-end trainable, and outperforms previous state-of-the-art methods by a great margin on both SQuAD and MS MARCO datasets.

2 OUR APPROACH

We will describe in details the architecture of our SiriQG. As show in Figure 1, our proposed SiriQG contains four major components as follows:

- Hierarchical Attention Fusion Network
- LSTM Question Decoder
- Hybrid Evaluator

2.1 Hierarchical Attention Fusion Network

Answer information is essential to creating high-quality, relevant questions from a Passage. However, most existing attention models that without explicit constraint attend to all words, often neglect potential semantic relations, which leads to an inaccurate focus on some dispensable words. Thus we propose a Hierarchical Attention Fusion Network (HAF), for better modeling of both answer information and passage information by integrating explicit syntactic constraints into attention mechanism, and for effectively incorporating the answer information into the passage information by performing attention at both contextualized hidden state level and word-level level.

[1] Dalian University of Technology, email: zhuangliu@mail.dlut.edu.cn
[2] Dalian University of Technology, email: Huangkaiyu@mail.dlut.edu.cn
[3] Dalian University of Technology, email: huangdg@dlut.edu.cn
[4] Union Mobile Fintech Technology Co.LTD, email: zhaojun1978@126.com

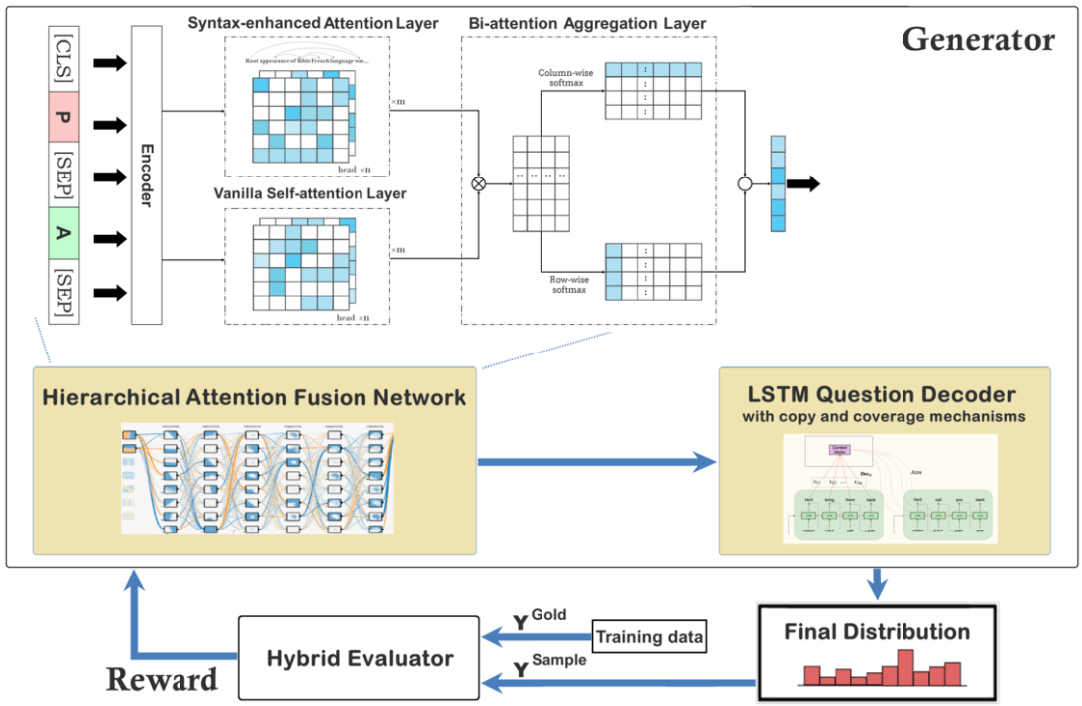

Figure 1. An illustration of the architecture for our SiriQG.

We first present proposed HAF architecture that is a hierarchical multi-stage attention fusion process, consisting of three layers. We will discuss each component in detail.

2.1.1 *Encoder*

To encode input tokens into representations, we take pre-trained LMs, such as BERT, as the encoder. In SiriQG, we take $\text{BERT}_{\text{LARGE}}$ as encoder. To get global contextualized representation, for each different candidate answer, we concatenate its corresponding passage and question with it to form one sequence and then feed it into the encoder. Let $P = [p_1, p_2, ..., p_m]$, $A = [a_1, a_2, ..., a_k]$ respectively denote the sequences of passage and answer, where p_i, a_i are tokens. The adopted encoder with encoding function $Encoder(\cdot)$ takes the concatenation of P and A as input,

$$E = Encode(P \oplus A) \tag{1}$$

In detail, following the implementation of pre-trained LMs, we organize the input X for pre-trained LMs, as the following sequence: [CLS]P[SEP]A[SEP], where, the first token is the special token [CLS] and the sequences are separated by the [SEP] token. The output E will then be fed to both vanilla self-attention layer and proposed syntax-enhanced attention layer for obtaining the syntax-guided representation.

2.1.2 *Syntax-enhanced Attention Layer*

First, we train a syntactic (dependency) parser to build each sentence's dependency structures that are then supplied to the SiriQG, as a token-aware attention guidance. Specifically, we only restrict attention between the word and all the ancestor head words to use the relationship between headword and the dependent words that are given by the syntax dependency tree of the sentence. That word we want only to deal with syntactic words in a sentence, ancestor head words in the child's word view. In detail, Let input sequence $X = \{x_1, x_2, ..., x_n\}$, in which n denotes the token sequence length, first we construct a dependency tree with syntactic parser, and for each word x_i, we select the ancestor node set Ri from the dependency tree. Then, we learn a sequence of DOI mask $\mathcal{D}$, which is organized as a matrix $(n \times n)$, and the elements in each row represent the row-index word dependency mask. If token x_i is the token s_j's ancestor node, $\mathcal{D}[i, j]$=1, 0 otherwise.

Then, to take advantage of the syntax structure information, we propose a mask-self-attention, seen as an extension of self-attention. Formally, the mask-self-attention is,

$$\begin{aligned} \text{head}_i &= \text{Attention}(hW_i^Q, hW_i^K, hW_i^V) \\ \text{Attention}(Q, K, V) &= \text{softmax}(\frac{\mathcal{D} \cdot (QK^T)}{\sqrt{d}})V \end{aligned} \tag{2}$$

where we project the representation E directly from the last layer of the BERT encoder into the distinct query Q, key K and value V, respectively. W_i^Q, W_i^K, HW_i^V are the weights to learn. d is the hidden size of head_i for scaling attention weights. Each layer is constructed by multi-head self-attention:

$$\text{MultiHead(H)} = f(\text{Concat}(\text{head}_1, ..., \text{head}_\text{I})) \tag{3}$$

where $\text{MultiHead}(\cdot)$ denotes multi-head attention. H is the hidden representation from last layer. $f(\cdot)$ is a non-linear transformation, and the function $\text{Concat}(\cdot, \cdot)$ is to concatenate all the head_i. Finally, the output H' denoted as $H' = \{h_1, h_2, ..., h_n\}$.

2.1.3 *Bi-attention Aggregation Layer*

Finally, we integrate two contextual vectors for answer prediction: i) a syntax-enhanced context vector from proposed syntax-enhanced attention layer; ii) a vanilla BERT context vector directly from the last

layer of the BERT encoder. The final model output $\bar{H}$ is computed by:

$$\begin{aligned}\tilde{H} &= \text{Self}-\text{attention}(H)\\ \tilde{H} &= \text{Bi}-\text{attention}(\tilde{H}, H')\end{aligned} \tag{4}$$

2.2 GRU Question Decoder

The decoder model takes the context-rich embedding that is previously computed. Following pointer-generator network [23], our decoding framework is a Bi-LSTM, in detail, we adopt an attention-based [15] Bi-LSTM decoder with coverage mechanisms [6, 27] and copy mechanism [7, 26, 28]. At each decoding time-step t, a mechanism of attention learns to attend to the most relevant input sequence words, and computes the context representation h_t^* that is based on the attention memory, the current coverage vector c^t, and the current decoding state s_t. Besides, the probability of generation p_{gen} is calculated from the decoder input y_{t-1}, the decoder state s_t, and the context representation h_t^*. We then use p_{gen} as a switch to choose between copying a word from the input or generating a word from the vocabulary. We keep a decoder vocabulary that is an extended vocabulary dynamically, and is an union of all words that appear in a batch of source passages ore answers, and the usual vocabulary. Also, to encourage the decoder to make use of the diverse components of the input text, we finally use a coverage vector c^t, the sum of attention distributions over prior decoder time steps. Meantime, a coverage loss is measured to penalize attending to the same locations of the input text repeatedly. In addition, to allow the decoder to use the different input components, we also employ the coverage mechanism, which acknowledges other contextually important (and possibly rare) passage words that the answer needs to conform to, while not repeating words redundantly. At each step, we keep a coverage vector c^t that is the sum of the attention distributions during prior decoder steps. Also, a coverage loss is calculated to penalize attending to the same locations repeatedly.

2.3 Hybrid Evaluator

To solve the loss limit using cross-entropy that is based sequence training, existing QG models [6, 11, 26] often optimize evaluation metrics with reinforcement directly. However, they fail to generate syntactically and semantically valid text. To address potential problems, we propose a hybrid evaluator that has a mixed objective, which combines both RL losses and cross-entropy to ensure the generation of syntactically coherent and semantically meaningful text.

For the reinforcement learning part, inspired by [25, 32], we introduce a pretrained question paraphrasing classification approach, to evaluate to paraphrasing probability and provide accurate rewards. Because paraphrasing is about semantic similarity rather than superficial phrase matching or word matching, it more fairly treats question paraphrases. Specifically, first we train a question paraphrasing model using Quora Question Pairs dataset[5]. We then adopt it as an environment and during training the question generation model will interact, and finally we take the paraphrasing probability of question generated and the ground-truth as a reward. To apply this reward, we use the RL algorithm learning a generation policy that is defined by the question generation model parameters. Specifically, we use the self-critical sequence training (SCST) algorithm [22] for optimizing evaluation metrics directly. During training, at each iteration, the model will generate 2 output sequences: the baseline output $\hat{Q}$ that is obtained by greedy search and the sampled output Q^s that is produced by sampling. We define $r(Q)$ as a reward of the output Q, which is calculated with certain reward metrics in accordance with the corresponding ground-truth sequence Q^*. Formally, the loss function is defined as

$$\nabla_\theta \mathcal{L}_{RL} = -(r(Q^s) - r(\hat{Q}))\nabla_\theta log P_\theta(Q^s) \tag{5}$$

We train the model in 2 stages. During training phase, we first use a regular cross-entropy loss training model, defined as

$$\mathcal{L}_{ML} = \sum_t (-logP(y_t^*|X, y_{<t}^*)) + \lambda_c \mathcal{L}_{cov} \tag{6}$$

where y_t^* is the word for the t-th position element of the output sequence of the ground-truth. λ_c is the coverage hyper-parameter and the coverage loss $\mathcal{L}_{cov}$ is defined as:

$$\mathcal{L}_{cov} = \sum_t (min(a_i^t, wcv_i^t)) \tag{7}$$

where a_i^t is the word for the t-th position element of the attention on input sequence at the time step t. For words already predicted we maintain a word coverage vector wcv as the sum of all attention distributions which range over time-step from 0 to t-1, and at the time-step t, $wcv^t = \sum_{t'=0}^{t-1} a^{t'}$. We use scheduled teacher forcing [1] to alleviate the exposure bias problem. Next, we fine-tune the model through optimizing a hybrid objective that combines both RL loss and cross-entropy loss. Formally, loss function defined as,

$$\mathcal{L} = \gamma \mathcal{L}_{RL} + (1-\gamma)\mathcal{L}_{ML} \tag{8}$$

where ratio factor γ controls the trade-off between RL loss and cross-entropy loss. In text summarization and machine translation [18, 30, 32], a similar mixed objective function approach has also been adopted. We adopt beam search to generate final predictions during prediction.

3 EXPERIMENTS

3.1 Dataset

We use two question answering datasets, SQuAD[6] [20] and MS Marco[7] [17] as the target datasets. These constitute a comprehensive set of data used to evaluate QG task.

SQuAD is a large-scale question answering dataset that contains 107,785 questions with 536 passages, which is posed by human crowd-workers on a variety of Wikipedia articles, in which answer span is in a Wikipedia passage. We construct a dataset for our QG task based on the dev dataset and training dataset of the accessible SQuAD.

- Split1: We manintain the SQuAD training set and split the SQuAD development set into our development and test set with the ratio 1:1, randomly. Split1 is based on sentence level, which is similar to [25, 33];
- Split2: We split the SQuAD training set into development set and training set with the ratio of 1:9, randomly. Also, we use the SQuAD development set as the test set. Split2 is based on article level, which is similar to [4, 34];

[5] https://tinyurl.com/y2y8u5ed

[6] https://rajpurkar.github.io/SQuAD-explorer/

[7] http://www.msmarco.org/leaders.aspx

Table 1. Automatic evaluation results on the SQuAD.

	Split1						Split2					
	Bleu1	**Bleu2**	**Bleu3**	**Blue4**	**Meteor**	**Rouge-L**	**Bleu1**	**Bleu2**	**Bleu3**	**Blue4**	**Meteor**	**Rouge-L**
NQG++ [34]	42.36	26.33	18.46	13.51	18.18	41.60	-	-	-	-	-	-
seq2seq+GAN[31]	44.42	26.03	17.60	13.36	17.70	40.42	-	-	-	-	-	-
MPQG [24]	-	-	-	13.91	-	-	-	-	-	13.98	18.77	42.72
ASs2s [8]	-	-	-	16.17	-	-	-	-	-	16.20	19.92	43.96
L2A [4]	-	-	-	-	-	-	43.09	25.96	17.50	12.28	16.62	39.75
CGC-QG [12]	-	-	-	-	-	-	40.45	23.52	15.68	11.06	17.43	43.16
s2sa-amg[33]	45.69	30.25	22.16	16.85	20.62	44.99	45.07	29.58	21.60	16.38	20.25	44.48
G2S [2]	-	-	-	17.94	21.76	46.02	-	-	-	18.30	-	-
SiriQG (ours)	**47.68**	**32.51**	**24.39**	**19.30**	**22.91**	**47.52**	**48.10**	**32.8**	**24.81**	**20.04**	**23.42**	**47.67**

Notes: Results in question generation on SQuAD split1 and split2, respectively.

MS MARCO is the human generated reading comprehension dataset that is from a million search queries from Bing. In MS MARCO, each query is linked to passages from several documents form Bing, and the dataset mentions from these paragraphs a list of the basic truth answers. Similar to [33], we further extract a subset of MS MARCO in which the answers are sub spans of passages, and divide the original training set randomly into sets of training set and development set.

We run Stanford CoreNLP [16] for pre-processing. We first lowercase all the data, and extract a sentence that contains an answer phrase, and we use it as the input passage. If the answer spans several phrases, we extract those phrases and use them as the input passage to concatenate. Stanford CoreNLP is used for extracting sentences from the questions. If CoreNLP extracts multiple words as question sentences, we use the entire word as the question expression. Table 2 shows the details of the both datasets that are used for training, development, and test, respectively.

Table 2. Statistics of the evaluation datasets.

	SQuAD-1	**SQuAD-2**	**MS MARCO**
# Train	87,488	77,739	51,000
# Dev	5,267	9,749	6,000
# Test	5,272	10,540	7,000
# passages	126	127	60
# questions	11	11	6
# answers	3	3	15

Notes: The statistics of both datasets used in our experiments. The # passages, # questions and # answers denote the size of the average length of passages, questions and answers of corresponding dataset respectively.

3.2 Baseline Models

As benchmarks, we compare proposed SiriQG against several previous models on QG task. The baseline methods in our experiments include: NQG++, Seq2seq+GAN, QG+QA, MPQG, ASs2s, L2A, CGC-QG, s2sa-amg, and G2S.

- **NQG++** [34] is an Seq2Seq model based on attention, which is equipped with a feature-rich encoder and copy mechanism to encode answer position, NER, and POS information;
- **Seq2seq+GAN**: [31] proposed a generation model based on GAN, which can better learn representations, and capture the diversity with the observed variables;
- **QG+QA** [5] is a Seq2Seq model, combining supervised learning and reinforcement learning for QG task.
- **MPQG** [24] is a Seq2Seq model, matching the answer with the document for QA task.
- **ASs2s** is an answer-separated Seq2Seq model that is proposed by [8], which separately treats the passage and the answer.
- **L2A** [4] is a seminal question generation model.
- **CGC-QG** [12] is a QG model using multi-task learning framework, learning the accurate boundaries between generation and copying.
- **s2sa-amg** [33] proposed a Seq2Seq model containing a maxout pointer decoder and a self attention encoder to encode the context of question.
- **G2S** [2] proposed a novel graph-to-sequence (Graph2Seq) model based on reinforcement learning (RL) for QG task.

3.3 Evaluation Metrics

Following most of previous QG works, we use six automatic metrics as our evaluation metrics: Bleu1, Bleu2, Bleu3, Bleu4, METEOR, and ROUGE-L. In order to have an empirical comparison, we also use Bleu, METEOR and ROUGE to evaluate the QG models:

- **Bleu:** It is used to evaluate the average n-gram accuracy on a series of reference sentences;
- **METEOR:** It is a recall-oriented metric to measure the similarity of references and generations;
- **ROUGE-L:** It measures the longest common sub-sequence recall form the sentences generated compared with references;

On these metrics, a question that is syntactically and structurally similar to the human level score high, which indicates relevance to the passage and answer. We also perform human evaluation between benchmarks and our SiriQG on SQuAD split-2, besides automated evaluation.

3.4 Implementation Details

For the syntactic parser, we adopt the biaffine attention dependency parser from [3], and use Penn Treebank to annotate our task datasets. By joint learning of constituent parsing [9], we re-train the dependency parser using BERT as sole input, achieving high accuracy on PTB. For model implementation, in order to avoid extra influence and focus on the intrinsic performance of proposed SiriQA, we follow the same fine tuning procedure as BERT. At decoding stage, we utilized

beam search (beam size: 10), until each stack beam generation the <EOS> token.

3.5 Experimental Results

We show and compare results with evaluation metrics in Table 1 and Table 3. As can be seen, our SiriQG model outperforms the two baseline datasets, SQuAD and MS MARCO, on all evaluation metrics, and yields the state-of-the-art overall accuracy.

SQuAD Results. We report the overall performance of proposed SiriQG along with the benchmark approaches on SQuAD split1 and split2, respectively. The detailed results on GLUE are depicted in Table 1. As illustrated in Table 1, our SiriQG achieves much better results and outperforms previous best score on all of the 12 automatic evaluation metrics, all achieving new state-of-the-art (SOTA) results, in details, on Split1, improving from 17.94 to 19.30 in Bleu-4, 21.76 to 22.91 in Meteor, 46.02 to 47.52 in Rouge-L; on Split2, improving from 18.30 to 20.04 in Bleu-4, 20.25 to 23.42 in Meteor, 44.48 to 47.67 in Rouge-L.

MS MARCO Results. Table 3 shows the performances on three evaluation metrics, Blue4, METEOR and ROUGE-L, on MS MACRO dataset. From the data in Table 3, it is apparent that proposed SiriQG consistently significantly outperforms prior works on all of three evaluation metrics. Compared to the best previous reported result, we obtain an absolute improvement of 5.08 in Bleu4, 4.73 in Meteor and 4.92 in Rouge-L, and achieve the SOTA over all 3 accuracy. To better comparison, we also re-implemented MPQG model based on its released source code. As expected, SiriQG are much better than our re-implemented MPQG, which is a Seq2Seq model, just matching the answer with the passage for QA task. The experimental results indicate that our SiriQG is more consistently effective, and demonstrate that our proposed semantics-reinforced method is highly effective.

Table 3. Automatic evaluation results on the MS MARCO.

	Blue4	Meteor	Rouge-L
L2A [4]	10.46	-	-
QG+QA [5]	11.46	-	-
s2sa-amg [33]	17.24	-	-
MPQG†	15.03	20.10	43.98
SiriQG (ours)	**20.11**	**24.83**	**48.90**

Notes: † indicates our reimplemented model using released source code. We reimplemented the MPQG model in our experiments on MS MARCO.

4 ABLATION STUDY AND ANALYSES

In this section, we perform ablation experiments on the SQuAD dataset to investigate key features of our proposed SiriQG. We further perform the comprehensive ablation analyses to systematically assess the impact of different model components (e.g., syntax-enhanced attention, evaluator) for our proposed full SiriQG model. Specifically, in ablation, we examine how model performance is affected when : i) we remove the syntax-enhanced attention mechanism in the encoder, just passing the vanilla self-attention representation to the decoder; ii) we use just one loss evaluator for optimizing evaluation. Ablation experimental results confirmed that components make the contribution to the overall performance in our proposed model. As shown in Table 4, without using syntax-enhanced attention mechanism, the performance drops significantly, suffering a more than drop of 5.8 points, showing syntax-attention effectiveness. By integrating explicit syntactic constraints into attention mechanism, our SiriQG model can better model both answer information and passage information, which helps effectively incorporates answer information into the passage information, and makes it better able to fuse different levels of granularity. Besides, we can see the advantages of using hybrid evaluator. The Bleu4 obviously drops from 20.04% to 17.99% when cross-entropy loss evaluator, to 18.31% when reducing question paraphrasing RL loss evaluator. Experimental results showed the benefits of using hybrid loss evaluator. The hybrid evaluator has a considerable impact on the performance.

Table 4. Ablation study on the SQuAD split-2 test set.

	Bleu-4
SiriQG	20.04
SiriQG w/o syntax-enhanced attention	14.18 (-5.86)
SiriQG w/o question paraphrasing evaluator	17.99 (-2.05)
SiriQG w/o cross-entropy loss evaluator	18.31 (-1.63)

5 HUMAN EVALUATION

In addition, we perform a human evaluation to measure the quality of the questions produced by proposed SiriQG. We performed a small human assessment on the SQuAD split-1 training data, 80 random samples per model. We then asked 8 English speakers to evaluate the quality of generated questions from QG models. In each sample, given a triple that contains a Passage, an Answer and an output generated by baseline models. By answering the following three questions, they were asked to rate the output of the model:

1) Is the generated question correct syntactically?
2) Is the generated question correct semantically?
3) Is the question generated relevant to the source passage?

The evaluation question obtained is evaluated on a 1-5 scale by all eight human evaluators, where a higher score is a better quality (i.e., 1 for Bad, 2 for Marginal, 3 for Acceptable, 4 for Better, 5 for Excellence). We report the average from all evaluators as the final score.

First, following [4], we use Naturalness and Difficulty as our human evaluation metrics: i) Naturalness, measuring the grammaticality and fluency; ii) Difficulty, indicating the reasoning and the syntactic divergence that are needed to answer the question. As shown in Table 5, we compare proposed SiriQG to the baseline L2A, and find that SiriQG significantly outperforms L2A [4].

Table 5. Human evaluation results on Naturalness and Difficulty evaluation metrics (5 for the best).

	Naturalness	Difficulty
Human Performance	4.00	2.87
L2A [4]	3.36	3.03
MPQG †	3.42	3.08
SiriQG (ours)	3.71	3.36

Notes: † indicates our reimplemented model using its released source code. we reimplemented the MPQG in our experiments on SQuAD split1. The majority of previous models, except for LTA [4] and our reimplemented MPQG, does not perform human evaluations, and we do not have the code to replicate outputs for most concurrent methods.

Furthermore, we conduct a further human comparison to rate the generated questions quality, evaluation metrics: relevance, semantics

and syntax. As shown in Table 6, we can see that SiriQG consistently outperforms the strong benchmark MPQG. More remarkably, our proposal is obtained great results even compared to the ground-truth.

Table 6. Human evaluation results on Syntax, Semantics and Relevance evaluation metrics (5 for the best).

	Syn.	Sem.	Rel.
Human Performance	4.78	4.80	4.34
MPQG †	4.29	4.15	3.32
SiriQG (ours)	4.75	4.71	4.20

Notes: The Syn., Sem. and Rel. denote syntactically correct, semantically correct and relevant score, respectively. † indicates our reimplemented model using its released source code. we reimplemented the MPQG in our experiments on SQuAD split1.

6 CASE STUDY

In Table 7, we also show an example that illustrates the quality of the text generated by different ablated systems. As shown in Table 7, integrating explicit syntactic constraints into attention mechanism helps SiriQG to better model both answer information and passage information, and thus makes it better able to fuse different levels of granularity and makes the generated a question that is syntactically and structurally similar to the human level.

Table 7. An example of generated question and ground-truth question in SQuAD.

Passage:
... The appearance of the Bible in French language was important to the important to the spread of the protestant movement and development of the reformed church in France. The country had a long history of struggles with the papacy by the time the protestant reformation finally arrived. Around 1294, a French version of the scriptures was prepared by the Roman Catholic priest, guards de Moulin...
Gold:
When did the first French language Bible appear?
SiriQG (full):
When was the appearance of the Bible in French language?
SiriQG w/o syntax-enhanced attention:
When did a French version of the scriptures was prepared ?

Notes: Answers are underlined. The cyan sentence indicates additional background that is used by a human for generating the Question. The olive sentences contain answers.

7 RELATED WORK

Neural network-based models represent the SOTA in question generation (QG) from a given document. Inspired by neural MT, [5] suggested a Seq2Seq model, combining supervised learning and reinforcement learning for QG task. [31] proposed to add linguistic features to each word and to encode the most appropriate answer to the document when generating questions. Given in the training data, [8] encoded ground-truth answers, using the copy mechanism to employ matching context to capture the interactions between its context within the document and the answer. [10] proposed a cross lingual training method that generating questions from text in low resource languages automatically. [2] proposed a novel graph-to-sequence (Graph2Seq) model based on reinforcement learning (RL) for QG task. [12] is a QG model using multi-task learning framework, learning the accurate boundaries between generation and copying. [34] is a Seq2Seq model based on attention, which is equipped with a feature-rich encoder and copy mechanism to encode answer position, NER, and POS information;

Deep Reinforcement Learning, very recently, has successfully been applied to NLP tasks, such as Text Summarization, [19, 32]. On the other hand, question generation also involves deciding query type, i.e. when, what, etc., being selective on which keywords to copy from the passage into the question text, and leave remaining keywords from the answer. It is required to develop a specific generative probabilistic model. [12] is a QG model using multi-task learning framework, learning the accurate boundaries between generation and copying. [13, 33] proposed a Seq2Seq model containing a maxout pointer decoder and a self-attention encoder to encode the context of question. In comparison, we create a semantics-reinforced network model to predict the main text for generating questions, which helps to better understand the internal structure of Answer and Passage.

8 CONCLUSION

We proposed a novel RL based semantics-reinforced networks to address the challenging Question Generation (QG) task. Our proposed method, named SiriQG, in which the answer information is utilized by an effective hierarchical fusion attention, which can better model of both answer information and passage information by integrating explicit syntactic constraints into attention mechanism, and makes it better to fuse different levels of granularity. Also we introduce a hybrid evaluator with a mixed objective that combines both RL loss and cross-entropy loss to ensure the generation of semantically and syntactically question text. Future studies on QG will focus on the following aspects: i) We further consider automatic selection of appropriate interrogative phrases such that the answerers can reach the target answer easily; ii) To reduce that risk, we will use textual entailment to verify whether the generated questions are consistent with the source passages.

ACKNOWLEDGEMENTS

We would like to thank the reviewers for their helpful comments and suggestions to improve the quality of the paper. The authors gratefully acknowledge the financial support provided by the National Natural Science Foundation of China under (No.61672127, U1916109).

REFERENCES

[1] Samy Bengio, Oriol Vinyals, Navdeep Jaitly, and Noam Shazeer, 'Scheduled sampling for sequence prediction with recurrent neural networks', in *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada*, pp. 1171–1179, (2015).

[2] Yu Chen, Lingfei Wu, and Mohammed J. Zaki, 'Natural question generation with reinforcement learning based graph-to-sequence model', *CoRR*, **abs/1910.08832**, (2019).

[3] Timothy Dozat and Christopher D. Manning, 'Deep biaffine attention for neural dependency parsing', in *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*, (2017).

[4] Xinya Du, Junru Shao, and Claire Cardie, 'Learning to ask: Neural question generation for reading comprehension', in *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 1: Long Papers*, pp. 1342–1352, (2017).

[5] Nan Duan, Duyu Tang, Peng Chen, and Ming Zhou, 'Question generation for question answering', in *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, EMNLP 2017, Copenhagen, Denmark, September 9-11, 2017*, pp. 866–874, (2017).

[6] Yifan Gao, Piji Li, Irwin King, and Michael R. Lyu, 'Interconnected question generation with coreference alignment and conversation flow modeling', in *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pp. 4853–4862, (2019).

[7] Jiatao Gu, Zhengdong Lu, Hang Li, and Victor O. K. Li, 'Incorporating copying mechanism in sequence-to-sequence learning', in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1: Long Papers*, (2016).

[8] Yanghoon Kim, Hwanhee Lee, Joongbo Shin, and Kyomin Jung, 'Improving neural question generation using answer separation', in *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019*, pp. 6602–6609, (2019).

[9] Nikita Kitaev and Dan Klein, 'Constituency parsing with a self-attentive encoder', in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers*, pp. 2676–2686, (2018).

[10] Vishwajeet Kumar, Kireeti Boorla, Yogesh Meena, Ganesh Ramakrishnan, and Yuan-Fang Li, 'Automating reading comprehension by generating question and answer pairs', in *Advances in Knowledge Discovery and Data Mining - 22nd Pacific-Asia Conference, PAKDD 2018, Melbourne, VIC, Australia, June 3-6, 2018, Proceedings, Part III*, pp. 335–348, (2018).

[11] Vishwajeet Kumar, Ganesh Ramakrishnan, and Yuan-Fang Li, 'A framework for automatic question generation from text using deep reinforcement learning', *CoRR*, **abs/1808.04961**, (2018).

[12] Bang Liu, Mingjun Zhao, Di Niu, Kunfeng Lai, Yancheng He, Haojie Wei, and Yu Xu, 'Learning to generate questions by learning what not to generate', *CoRR*, **abs/1902.10418**, (2019).

[13] Zhuang Liu, Degen Huang, Kaiyu Huang, and Jing Zhang, 'DIM reader: Dual interaction model for machine comprehension', in *Chinese Computational Linguistics and Natural Language Processing Based on Naturally Annotated Big Data - 16th China National Conference, CCL 2017, - and - 5th International Symposium, NLP-NABD 2017, Nanjing, China, October 13-15, 2017, Proceedings*, pp. 387–397, (2017).

[14] Zhuang Liu, Keli Xiao, Bo Jin, Kaiyu Huang, Degen Huang, and Yunxia Zhang, 'Unified generative adversarial networks for multiple-choice oriented machine comprehension', *ACM TIST*, **1**(1), 19:1–19:23, (2019).

[15] Thang Luong, Hieu Pham, and Christopher D. Manning, 'Effective approaches to attention-based neural machine translation', in *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP 2015, Lisbon, Portugal, September 17-21, 2015*, pp. 1412–1421, (2015).

[16] Christopher D. Manning, Mihai Surdeanu, John Bauer, Jenny Rose Finkel, Steven Bethard, and David McClosky, 'The stanford corenlp natural language processing toolkit', in *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, ACL 2014, Baltimore, MD, USA, System Demonstrations*, (2014).

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

[18] Romain Paulus, Caiming Xiong, and Richard Socher, 'A deep reinforced model for abstractive summarization', in *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*, (2018).

[19] Romain Paulus, Caiming Xiong, and Richard Socher, 'A deep reinforced model for abstractive summarization', in *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, Conference Track Proceedings*, (2018).

[20] Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang, 'Squad: 100, 000+ questions for machine comprehension of text', in *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016, Austin, Texas, USA, November 1-4, 2016*, pp. 2383–2392, (2016).

[21] Marc'Aurelio Ranzato, Sumit Chopra, Michael Auli, and Wojciech Zaremba, 'Sequence level training with recurrent neural networks', in *4th International Conference on Learning Representations, ICLR 2016, San Juan, Puerto Rico, May 2-4, 2016, Conference Track Proceedings*, (2016).

[22] Steven J. Rennie, Etienne Marcheret, Youssef Mroueh, Jerret Ross, and Vaibhava Goel, 'Self-critical sequence training for image captioning', in *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, pp. 1179–1195, (2017).

[23] Abigail See, Peter J. Liu, and Christopher D. Manning, 'Get to the point: Summarization with pointer-generator networks', in *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 1: Long Papers*, pp. 1073–1083, (2017).

[24] Linfeng Song, Zhiguo Wang, and Wael Hamza, 'A unified query-based generative model for question generation and question answering', *CoRR*, **abs/1709.01058**, (2017).

[25] Linfeng Song, Zhiguo Wang, Wael Hamza, Yue Zhang, and Daniel Gildea, '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, NAACL-HLT, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 2 (Short Papers)*, pp. 569–574, (2018).

[26] Sandeep Subramanian, Tong Wang, Xingdi Yuan, Saizheng Zhang, Adam Trischler, and Yoshua Bengio, 'Neural models for key phrase extraction and question generation', in *Proceedings of the Workshop on Machine Reading for Question Answering@ACL 2018, Melbourne, Australia, July 19, 2018*, pp. 78–88, (2018).

[27] Zhaopeng Tu, Zhengdong Lu, Yang Liu, Xiaohua Liu, and Hang Li, 'Modeling coverage for neural machine translation', in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1: Long Papers*, (2016).

[28] Oriol Vinyals, Meire Fortunato, and Navdeep Jaitly, 'Pointer networks', in *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada*, pp. 2692–2700, (2015).

[29] Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V. Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, and Jeff Klingner, 'Google's neural machine translation system: Bridging the gap between human and machine translation', *CoRR*, **abs/1609.08144**, (2016).

[30] Min Yang, Qiang Qu, Wenting Tu, Ying Shen, Zhou Zhao, and Xiaojun Chen, 'Exploring human-like reading strategy for abstractive text summarization', in *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019*, pp. 7362–7369, (2019).

[31] Kaichun Yao, Libo Zhang, Tiejian Luo, Lili Tao, and Yanjun Wu, 'Teaching machines to ask questions', in *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI 2018, July 13-19, 2018, Stockholm, Sweden*, pp. 4546–4552, (2018).

[32] Shiyue Zhang and Mohit Bansal, 'Addressing semantic drift in question generation for semi-supervised question answering', in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pp. 2495–2509, (2019).

[33] Yao Zhao, Xiaochuan Ni, Yuanyuan Ding, and Qifa Ke, '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, Brussels, Belgium, October 31 - November 4, 2018*, pp. 3901–3910, (2018).

[34] Qingyu Zhou, Nan Yang, Furu Wei, Chuanqi Tan, Hangbo Bao, and Ming Zhou, 'Neural question generation from text: A preliminary study', in *Natural Language Processing and Chinese Computing - 6th CCF International Conference, NLPCC 2017, Dalian, China, November 8-12, 2017, Proceedings*, pp. 662–671, (2017).