Curriculum Learning for Natural Answer Generation

Cao Liu[1,2*], **Shizhu He**[1*], **Kang Liu**[1,2], **Jun Zhao**[1,2]
[1] National Laboratory of Pattern Recognition, Institute of Automation,
Chinese Academy of Sciences, Beijing, 100190, China
[2] University of Chinese Academy of Sciences, Beijing, 100049, China
{cao.liu, shizhu.he, kliu, jzhao}@nlpr.ia.ac.cn

Abstract

By reason of being able to obtain natural language responses, natural answers are more favored in real-world Question Answering (QA) systems. Generative models learn to automatically generate natural answers from large-scale question answer pairs (QA-pairs). However, they are suffering from the uncontrollable and uneven quality of QA-pairs crawled from the Internet. To address this problem, we propose a curriculum learning based framework for natural answer generation (CL-NAG), which is able to take full advantage of the valuable learning data from a noisy and uneven-quality corpora. Specifically, we employ two practical measures to automatically measure the quality (complexity) of QA-pairs. Based on the measurements, CL-NAG firstly utilizes simple and low-quality QA-pairs to learn a basic model, and then gradually learns to produce better answers with richer contents and more complete syntaxes based on more complex and higher-quality QA-pairs. In this way, all valuable information in the noisy and uneven-quality corpora could be fully exploited. Experiments demonstrate that CL-NAG outperforms the state-of-the-art, which increases 6.8% and 8.7% in the accuracy for simple and complex questions, respectively.

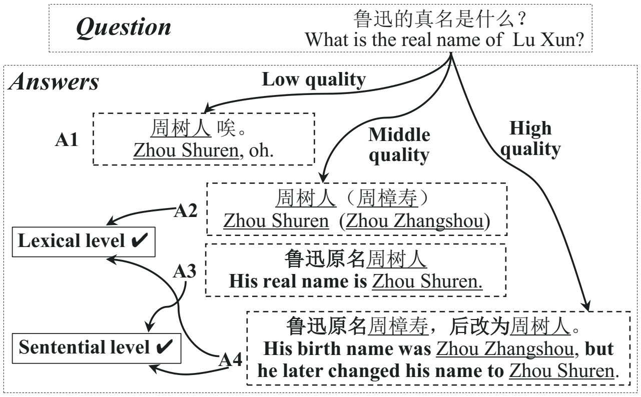
Question
鲁迅的真名是什么？
What is the real name of Lu Xun?
Answers
Low quality
Middle quality
High quality
A1
周树人 唉。
Zhou Shuren, oh.
A2
周树人（周樟寿）
Zhou Shuren (Zhou Zhangshou)
A3
鲁迅原名周树人
His real name is Zhou Shuren.
A4
鲁迅原名周樟寿，后改为周树人。
His birth name was Zhou Zhangshou, but he later changed his name to Zhou Shuren.
Lexical level ✔
Sentential level ✔

Figure 1: An example of different-quality answers (QA-pairs) on the real-world data.

1 Introduction

Natural Answer Generation (NAG, or natural question answering), which is able to generate natural answers in the form of natural language sentences, has received much attention in recent years [Yin *et al.*, 2016; He *et al.*, 2017]. Compared with the typical question answering (QA) systems which merely obtain exact answers in the form of entities or phrases [Unger *et al.*, 2014; Hao *et al.*, 2017], NAG could provide more natural responses, which is able to interact with ordinary users more friendly.

Recently, with the development of deep learning, more and more approaches utilize end-to-end models for text generation [Yin *et al.*, 2016; Gu *et al.*, 2016]. Most of them adopt the sequence-to-sequence (Seq2Seq) learning framework [Sutskever *et al.*, 2014], which takes word sequences as input, and generates output sequences word by word. In order to generate natural answers, copy mechanism [Gu *et al.*, 2016] and knowledge retrieval [Eric and Manning, 2017] are usually incorporated in Seq2Seq models. As a data-driven approach, the quality of generated natural answers in NAG heavily depends on the training data (QA-pairs). In fact, in order to train a robust and universal model, most work obtains large-scale QA-pairs by crawling human-generated questions and answers from Community Question Answering (CQA) websites such as *Yahoo! Answers*[1] and *Baidu Zhidao*[2].

However, due to the subjectivity, randomness, and one-sidedness of questions and answers written by the ordinary Internet users, the quality of learning QA-pairs is uncontrollable and uneven. Figure 1 demonstrates different-quality answers sharing the question "*What is the real name of Lu Xun*[3]*?*". In fact, 84.2% QA-pairs have duplicated questions within a real-world CQA dataset[2]. Although all answers in Figure 1 contain the correct entity (marked as underline), only A4 (*His birth name was Zhou Zhangshou, but he later*

[*] Equal contribution.

[1] https://answers.yahoo.com/

[2] https://zhidao.baidu.com/

[3] All examples are translated from Chinese in this paper. Here, *Lu Xun* is a leading figure of modern Chinese literature.

changed his name to Zhou Shuren.) is totally correct as well as having rich contents (marked as **bold**). Particularly, it explains that *Lu Xun*'s real name was changed to "*Zhou Shuren*" from "*Zhou Zhangshou*". In contrast, the first answer (A1) is partly noisy, which contains a correct entity as well as some noise; A2 is the solitary entity (not a natural sentence); A3 is correct and fluent. However, it is one-sided because of omitting the other correct answer entity "*Zhou Zhangshou*".

It is hard to learn a good model for NAG from such a noisy and uneven-quality corpora. On the one hand, it is an intractable task to select QA-pairs with high-quality answers. In fact, the quality of answers is hard to evaluate and calculate. Novikova *et al.* [2017] have demonstrated that the state-of-the-art automatic metrics are poorly related to human evaluation in natural language generation tasks. Moreover, human evaluation is impractical for scalable machine learning models. On the other hand, the size of learning data will be reduced sharply if we only take high-quality QA-pairs into consideration. In fact, even partly noisy, solitary and one-sided answers still contain some useful information for generating natural answers. For instance, though A1 in Figure 1 contains partial noise, the entity "*Zhou Shuren*" in A1 also contributes to learning the interaction with the knowledge base (KB), which should not be directly and completely removed from the learning corpora.

Therefore, in order to take full use of the learning data, as well as to robustly deal with the noisy and uneven-quality QA-pairs, inspired by Sachan and Xing [2016], we propose a natural answer generation framework based on curriculum learning [Bengio *et al.*, 2009] (**CL-NAG**). In this framework, answer selectors are used to automatically measure the different complexities and qualities of QA-pairs. Specifically, we employ two answer selectors including a term frequency (T-F) based selector and a grammar (GM) based selector. On the lexical level, TF selector is able to measure whether answers are rich in contents or not (e.g. A2 and A4 in Figure 1 which contain the low-frequency term "*Zhou Zhangshou*" are regarded as more complex and higher-quality answers). On the sentential level, GM selector can estimate answers with good grammar (e.g. fluent answers with complete syntax are supposed to be more complex and higher-quality ones (A3, A4) in Figure 1). Thereafter, curriculum learning is employed for training NAG model, which is able to firstly learn a basic QA model with simple and low-quality QA-pairs (such as distilling correct entities from the low-quality and short answers to interact with the KB), and then gradually learn to produce better answers with complex and high-quality QA-pairs (generating natural responses). Experiments on an open real-world CQA dataset demonstrate the effectiveness of CL-NAG on automatic and manual evaluations. Compared to the state-of-the-art, CL-NAG increases 6.8% and 8.7% in the accuracy for simple and complex questions, respectively. Furthermore, our model is able to deliver answers with richer contents and more complete syntax.

In brief, our main contributions are as follows.

- We propose a curriculum learning based Natural Answer Generation framework (CL-NAG), which is able to make full use of all valuable information on a noisy and uneven-quality corpora.
- We employ two practical measures to automatically measure the complexity and quality of QA-pairs. Based on these measures, we adopt curriculum learning for NAG. It is able to firstly learn a basic QA model with simple and low-quality QA-pairs, and then gradually learn to produce better answers with richer contents and more complete syntaxes.
- Experiments on an open CQA dataset demonstrate that CL-NAG outperforms the state-of-the-art on automatic and manual evaluations. Especially, it increases 6.8% and 8.7% in the accuracy for simple and complex questions, respectively.

2 Background

2.1 Task Description

Natural Answer Generation can be regarded as a fusion task of knowledge base question answering (KBQA) and chatbot / (one-turn) dialog. The NAG system takes a sequence of words as the input question sentence, and then produces another sequence of words as the answer sentence. Meanwhile, the system needs to interact with KB for obtaining correct answer entities, which retrieves a set of candidate facts and generates correct answers using corresponding facts [Yin *et al.*, 2016]. In other words, answering words consist of both common words (from vocabulary) and KB-words (from retrieved facts).

COREQA [He *et al.*, 2017] is a typical NAG model, which incorporates the copy and retrieval mechanisms in sequence to sequence learning [Cho *et al.*, 2014]. In this model, a knowledge retrieval model is firstly utilized to retrieve related facts from the KB. Then the encoder transforms all the inputs (such as words, entities as well as their structural information) into numerical representations. Finally, the decoder generates natural answers with the encoded questions and knowledge.

2.2 Curriculum Learning

Curriculum learning [Bengio *et al.*, 2009] is a learning strategy in machine learning, which starts from easy instances and then gradually handles harder ones. Curriculum learning has been used to question answering by Sachan and Xing [2016]. They utilize training loss to represent the complexity of instances and propose several heuristic strategies, which achieve high performance. In contrast, in our natural answer generation task, one question may correspond to multiple correct answers. So the complexity of instances (QA-pairs) could be hardly reflected from the training loss because of the difficult evaluation of the generated natural answers [Novikova *et al.*, 2017]. Furthermore, the noisy and uneven-quality train data increases the difficulty of curriculum learning.

3 Methodology

Traditional methods, based on the Seq2Seq learning framework, are difficult to generate natural answers from a noisy and uneven-quality corpora. Curriculum learning is able to learn a basic model (e.g. how to interact with KB) from low-quality and short (simple) QA-pairs. Then it gradually generates correct and natural answers from training instances

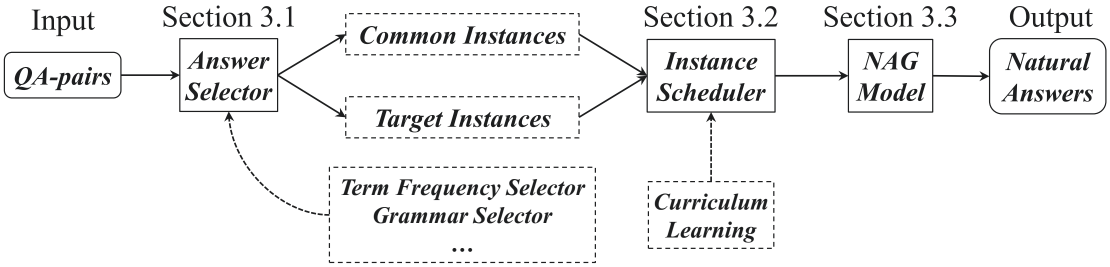

Figure 2: The overall diagram of natural answer generation framework based on curriculum learning.

whose answers are richer in contents and more complete in syntax. Nevertheless, the most challenging issue is how to measure the complexity of QA-pairs. Moreover, the complexity needs to be related to the quality of QA-pairs.

To address this problem, we propose curriculum learning based framework for NAG as shown in Figure 2. In this framework, answer selectors (see Section 3.1) are firstly used to measure the complexity and quality of QA-pairs. In more details, a term frequency (TF) selector (lexical level) and a grammar (GM) selector (sentential level) are used to select QA-pairs with richer contents and more complete syntaxes. The instances selected by answer selectors are regarded as **target instances**, while other instances are **common instances**. Thereafter, instance scheduler (see Section 3.2) based on curriculum learning is designed by two monotonous functions, which contribute to defining the learning progress from common instances to target instances. Finally, natural answers are generated by the NAG model (see Section 3.3). The details are shown as follows.

3.1 Answer Selector

Answer selectors are designed to select complex (high-quality) QA-pairs, which is under the assumption that the complexity (quality) of QA-pairs is determined by the complexity (quality) of their answers. Although it is very challenging to calculate the complexity (quality) of language totally, it is feasible to measure it on some aspects. We compute it from the view of term frequency and grammar.

Term Frequency Selector

Term frequency (e.g. TF-IDF [Salton and McGill, 1986]) is a significant feature to estimate the importance of terms. TF selector is able to select QA-pairs whose answers contain the low-frequency term, where the term frequency is obtained from statistics on the training data. On the one hand, low-frequency terms are richer in content compared to common words, so answers selected by TF selector are more meaningful than most common sentences. On the other hand, considering that very low-frequency terms bring noises with high probability, a minimum threshold for the low term frequency (e.g. 10) could be used to filter such noise. Consequently, TF selector is able to select sentences with richer contents and correct answers. As in the example illustrated in Figure 1, though both "*Zhou Shuren*" and "*Zhou Zhangshou*" are *Lu Xun's* real name, most people only know "*Zhou Shuren*". TF selector is able to choose A2 and A4 which contain the low-frequency term "*Zhou Zhangshou*". It is able to provide more meaningful contents.

Grammar Selector

Grammar is an important feature for evaluating the quality of natural language sentences. We utilize the Stanford Parser score as the metric of our grammar selector [Levy and Manning, 2003; Novikova *et al.*, 2017]. The Stanford Parser is not designed for measuring grammaticality. Fortunately, sentences with good grammar usually obtain higher scores than bad ones (e.g. with grammatical errors). However, the short answer or solitary entity obtains a high score in the parser, which limits the expressive power and naturalness of answers. To address this problem, a proportion (e.g. 0.5) of short and long answers is set to choose fewer short answers, which is beneficial to obtain meaningful expression of answers. Eventually, GM selector chooses A3 and A4 in Figure 1, which are more complex and higher-quality with good grammar.

3.2 Instance Scheduler

After obtaining the target instances and common instances, the next step is to determine the distribution of training instances (called instance scheduler) based on curriculum learning. Answers from common instances are usually in a short length and with a simple structure. Some of them contain noise. Initially, curriculum learning is able to learn a basic QA model with common instances (e.g. distilling correct entities from the low-quality and short answers to interact with the KB), and then gradually learn to produce better answers with richer contents and more complete syntaxes based on target instances (generating natural responses).

Similar to Sachan and Xing [2016], we formalize the idea as follows. Let $w \in [0,1]^{|Q|}$ represent the probability of sampling in each QA-pair, where $|Q|$ is the size of QA-pairs, and w is related to the complexity of QA-pairs and progress (such as the number of current training epoch). Common instances and target instances are marked as Q_c and Q_t, respectively. At first, the model tends to select common instances, so $w_{Q_c} \gg w_{Q_t}$. Subsequently, w_{Q_c} decreases and w_{Q_t} increases monotonically. Eventually, $w_{Q_c} \ll w_{Q_t}$, which means that the model is favor of target instances. The probabilities on the target and common instances are as follows.

$$w_{Q_t} = \left(\frac{epoch_t}{|epoch|}\right)^2 \tag{1}$$

$$w_{Q_c} = 1 - w_{Q_t} \tag{2}$$

where, $epoch_t$ and $|epoch|$ are the number of current epoch and entire epochs in training, respectively. Moreover, the probability of sampling is normalized by the accumulated probabilities on all samples.

3.3 Natural Answer Generation Model

We employ COREQA [He *et al.*, 2017] as the NAG model, which incorporates the copy and retrieval mechanisms in Seq2Seq learning. The details are as follows.

Retrieving Knowledges

NAG focuses on the knowledge requiring questions, and the number of topic entity (related to knowledge retrieval) is uncertain. The gold topic entities are utilized for simplifying the model. Based on the topic entity, the corresponding fact is retrieved from the KB, which is under the assumption that *subject* and *object* match the question and answer for the SPO fact <*subject, relation, object*>, respectively.

Encoder

Encoder transforms all inputs into numerical representations, which includes question encoder and knowledge encoder.

Bi-LSTM [Hochreiter and Schmidhuber, 1997] is utilized to encode the questions. Given a question (words sequence) $X = [x_1, ..., x_{L_X}]$, the concatenated representation for each word of hidden states in both directions ($\mathbf{h}_t = [\overrightarrow{\mathbf{h}}_t, \overleftarrow{\mathbf{h}}_{L_X-t+1}]$) is considered as the short-term memory of question ($\mathbf{M}_Q = \{\mathbf{h}_t\}$). And the last word in the forward can represent the whole question.

For the knowledge encoder, each part of the triple <*subject, relation, object*> has its own embedding, indicated as **s**, **r** and **o**, respectively. Each fact is represented by the concatenation of **s**, **r** and **o**. The list of facts $\{\mathbf{f}\} = \{\mathbf{f}_1, ..., \mathbf{f}_{L_F}\}$ is the short memory of retrieved knowledge, marked as $\mathbf{M}_{KB}$, where L_F is the maximun number of facts for answering each question.

Decoder

The decoder generates answers based on the short-memory of question and knowledge ($\mathbf{M}_Q$ and $\mathbf{M}_{KB}$), which contains three modes: 1) **predict-mode**, the answer word is generated from the vocabulary list; 2) **copy-mode**, the output word is copied from the source question; 3) **retrieve-mode**, answering word is obtained from the objects of the retrieved knowledge facts.

Attention mechanism [Bahdanau *et al.*, 2015] is also utilized in generating output words. At each step, it selectively reads the context vector $\mathbf{c}_{q_t}$ and fact vector $\mathbf{c}_{kb_t}$ from the short-term memory of question and fact ($\mathbf{M}_Q$ and $\mathbf{M}_{KB}$), and the accumulated attentions on $\mathbf{M}_Q$ and $\mathbf{M}_{KB}$ are kept to record the history information in decoding.

4 Experiments

Experimental data is an open real-world CQA dataset, which is from COREQA [He *et al.*, 2017]. In this dataset, raw QA-pairs are automatically "grounded" with KB by an integer linear programming (ILP) based method. However, only 44% QA-pairs are high-quality. It shows that curriculum learning is necessary. The dataset is divided into simple-QA and complex-QA according to the number of matched knowledge facts, in which simple-QA only matches one "grounded" fact, and the complex-QA contains multiple "grounded" facts.

Methods	Accuracy	WBM	
		BLEU	ROUGE
COREQA [2017]	52.7	20.5	23.9
TF-SOLE	**49.0**	21.0 ↑	24.9 ↑
TF-FP	48.9	22.9 ↑	25.5 ↑
TF-CL	47.8	**23.2** ↑	**26.7** ↑
GM-SOLE	55.5 ↑	23.6 ↑	26.2 ↑
GM-FP	58.4 ↑	**26.8** ↑	35.3 ↑
GM-CL	**59.5** ↑	21.8 ↑	**40.6** ↑

Table 1: Performances for automatic evaluation (%) on real-world simple-QA.

For purpose of comparison, we design experimental settings as follows.

- **COREQA** [He *et al.*, 2017]: As the basic model for the following methods, COREQA performs better than CopyNet [Gu *et al.*, 2016] and genQA [Yin *et al.*, 2016], and it achieves the state-of-the-art in this dataset.
- **SOLE**: Target instances are the only learning data.
- **FP**: Common and target instances are combined by a fixed proportion (we set it to 0.5).
- **CL**: Common and target instances are dynamically selected by curriculum learning.

Target instances on SOLE, FP and CL need to be selected by answer selectors (TF or GM selector), so the final methods are marked as TF/GM-SOLE/FP/CL (e.g. TF-CL means that TF selector is used to curriculum leaning). Specifically, the target instances of TF selector and GM selector are occupied 13.1% and 18.9%, respectively.

4.1 Automatic Evaluation (AE)

Similar to GenQA [Yin *et al.*, 2016], the accuracy is used to evaluate the correctness. Moreover, we utilize some word-based metrics (WBMs) to analyze the co-occurrences of n-gram between the references and generated answers, include BLEU and ROUGE [Sharma *et al.*, 2017][4].

In order to study the different effects of curriculum learning for the simple and complex question, we evaluate our model on the simple-QA and complex-QA, respectively.

AE on the simple-QA

Performance of automatic evaluation on the simple-QA is shown in Table 1. We can clearly obtain the following observations.

[4] WBMs are implemented in https://github.com/Maluuba/nlg-eval. In this paper, WBMs are based on Chinese characters, and the BLEU metric is BLEU2. To improve the quality of evaluating on WBMs, we label 100 high-quality QA-pairs for simple-QA and complex-QA.

Methods	**Accuracy**	**WBM**	
		BLEU	**ROUGE**
COREQA [2017]	47.4	12.5	25.9
TF-SOLE	15.1	1.8	5.9
TF-FP	3.3	1.3	3.5
TF-CL	**56.1** ↑	**24.8** ↑	**39.4** ↑
GM-SOLE	32.5	2.4	19.5
GM-FP	5.8	0.3	2.4
GM-CL	**51.6** ↑	**24.3** ↑	**30.9**

Table 2: Performances for automatic evaluation (%) on real-world complex-QA.

- TF-$*$ ($*$ includes SOLE, FP, and CL) and GM-$*$ achieve higher performance than COREQA on the most AEs. It indicates that TF and GM selectors contribute to generating better answers.
- The performance advantages of TF-CL and GM-CL are obvious. Especially, GM-CL increases 6.8% compared to the baseline in the accuracy[5]. The results demonstrate that curriculum learning is able to make full use of the noisy and uneven-quality data.
- TF-SOLE (GM-SOLE) is slightly better than COREQA. Due to the small-size training data obtained by TF (GM) selector only, the improvement is not obvious, and the performance is even declined.
- Both TF-FP and GM-FP work well, which is remarkably better than the SOLE method. It proves that the common instances contribute to producing better answers, too.

AE on the complex-QA

Performance of automatic evaluation on the complex-QA is shown in Table 2. It supports the following statements.

- Both TF-CL and GM-CL outperform the baseline on all metrics. Especially, the accuracy in TF-CL increases 8.7% compared to COREQA. Because the gold answers of complex-QA are longer than the ones in simple-QA, long and diversified answers selected by TF selector are more likely to cover the knowledge entities on the complex-QA.
- The SOLE method is worse than the baseline. It indicates that complex questions require more data to learn.
- TF-FP and GM-FP perform the worst. Although the TF and GM selectors are effective, the inappropriate fixed proportion of common and target instances is able to bring in a great deal of noise. Therefore, the dynamic sampling on different instances based on curriculum learning is necessary.

Other AE (Unreferenced Metric)

Methods	**Simple-QA**		**Complex-QA**	
	Length	**#Noun**	**Length**	**#Noun**
COREQA [2017]	2.92	1.24	2.89	1.23
TF-CL	**3.70** ↑	**1.32** ↑	**5.14** ↑	**3.09** ↑
GM-CL	1.81	0.79	4.50 ↑	1.84 ↑

Table 3: Unreferenced metrics on the simple-QA and complex-QA.

Models	**Correctness**	**Fluency**	**Coherence**
COREQA [2017]	11.5	18.5	11.3
TF-CL	**49.5** ↑*	38.5 ↑	**37.0** ↑*
GM-CL	20.5 ↑	**41.8** ↑*	21.0 ↑

Table 4: Manual evaluations on the complex-QA[7].

Except for the accuracy and WBMs, the length [Mou *et al.*, 2016] and noun amount of answers are used to be the "intrinsic" (no reference) evaluation. The length of answers is an objective and surfaced metric reflects the substance of answers. The number of nouns in the answer is another objective metric, which shows the meaningful context of answers. And we employ the *jieba*[6] toolkit for part-of-speech tagging. Table 3 illustrates that CL-NAG is able to generate better answers with a longer length and more nouns.

4.2 Manual Evaluation (ME)

Apart from automatic evaluation, we additionally utilize manual evaluation (ME). ME takes three aspects into consideration to evaluate the quality of generated answers (referred to [He *et al.*, 2017]).

(1) Correctness: measuring the correctness of answers. Richer and more diversified contents are considered to be more correct.

(2) Fluency: measuring generated answers are natural or good in grammars. Too short answers are considered as lacking fluency.

(3) Coherence: measuring whether answers are coherent to the source question or not.

Two annotators rate (win, failure or tie) the three aspects for COREQA, TF-CL, and GM-CL on complex-QA. We sample 100 questions and C_3^2 pair-wise comparisons for each question, and compute the percentage of winning times on the three models.

The manual evaluation is shown in Table 4. The Cohen Kappa statistics[8] between two annotators are 0.681, 0.443 and 0.484 in the correctness, fluency, and coherence, respectively, which is consistent with the intuition that fluency and coherence are more subjective than the correctness. Results in Table 4 support the following statements. Firstly, CL models (TF-CL and GM-CL) are better than COREQA on all evaluative aspects. It indicates that CL-NAG is able to deliver correct, fluent and coherent answers. Secondly, TF-CL is superior to any other models in the correctness and coherence except the fluency. In the contrary, GM-CL produces fluent answers with good grammar (syntax).

[5] The accuracy of COREQA in this paper is different from the results of COREQA [He *et al.*, 2017], while it is on the same tendency. It may be caused by the difference in experimental parameters and environments.

[6] https://github.com/fxsjy/jieba

[7] Mean scores ± 95% confidence intervals are verified for each highest performance, and * means that the score is significantly better than other models at 95% confidence.

[8] https://github.com/jorgearanda/kappa-stats

ID	Question	COREQA [He *et al.*, 2017]	CL-NAG
1	西游记的作者是谁 What is the author of *Journey to the West*	吴承恩 Wu Cheng’en	西游记的作者是吴承恩 Wu Chengen is the author of Journey to the West
2	三国演义的作者是谁 Who’s the author of *Romance of the Three Kingdoms*	罗贯中 Luo Guanzhong	**元末明初**的罗贯中 Luo Guanzhong in the **Late Yuan and Early Ming**
3	希特勒是哪国人？ Which country does *Hitler* comes from?	奥地利 Austria	奥地利**裔**德国**籍** Austrian **descent**, German **nationality**
4	msn是什么呀？ What is *MSN*?	微软的聊天工具，qq Chating tool of Microsoft, QQ	微软公司推出的即时通讯工具 Instant messaging tool launched by Microsoft
5	孔子是那国家的？ Which country is *Confucius*?	春秋时期鲁国人 Lu’s people in the Spring and Autumn Period	春秋时期鲁国人 Lu’s people in the Spring and Autumn Period
6	《红楼梦》的真正作者？ What is the author of *Dream of Red Mansions*	曹雪芹 Cao Xueqin	曹雪芹 Cao Xueqin
7	《情不自禁》的主演是谁啊 What is the Starring of *Take Me*	无答案 no answer	邬玉君和邬玉君 Wu Yujun and Wu Yujun
8	深瞳一共多少集? How many episodes of *ignorant*?	2424深瞳 24,24, ignorant	我知道是深瞳 I know is ignorant

Figure 3: Examples of the generated natural answers by COREQA and curriculum learning.

4.3 Case Study

Figure 3 lists some examples of generated natural answers by the basic model COREQA and the CL-based method. It is clear to see that our model generates more knowledgeable (marked as underline), richer contents (marked as **bold**) and more complete-syntax answers compared to COREQA. However, our model is still flawed. For example, 1) the system produces repetitive words with the purpose of generating complete answers (ID 7); 2) answers only contain the solitary entity (ID 6 and ID7); 3) Answers do not match the proper facts. The system delivers fluent answers without any knowledge (ID 8).

5 Related Work

Our generative model is based on Seq2Seq learning [Cho *et al.*, 2014], which is able to generate sequences such as natural language sentences. Recently, many works extend the Seq2Seq framework. GenQA [Yin *et al.*, 2016] is the first model to combine KB retrieval and produce fluent answers with Seq2Seq learning. However, it deals with the simple questions which contain only one fact. Furthermore, it can not copy words from source questions and it is liable to cause the out-of-vocabulary (OOV) problem. CopyNet [Gu *et al.*, 2016] utilizes the copy mechanism in sequence learning, which is failed to interact with KB and generate answers lacking facts. In order to reduce generic responses, Li *et al.* [2016] construct a set of dull responses and give a punishment to them. Eric and Manning [2017] incorporate KB by a key-value retrieval network on the task-oriented dialog with a small-size KB. Moreover, COREQA [He *et al.*, 2017] incorporates the copy mechanism and KB retrieval in the Seq2Seq framework for knowledge-requiring CQA questions, where different words in answers are generated by predicting from vocabularies, copying from the source questions and retrieving from the KB. All of these approaches are the foundations of our work.

Furthermore, our work is inspired by curriculum learning too. The strategy of curriculum learning is starting from easier instances and gradually handling harder ones, which stems from cognitive psychology [Skinner, 1958]. Curriculum learning has been applied in some NLP tasks, such as language model [Bengio *et al.*, 2009] and question answering [Sachan and Xing, 2016]. Sachan and Xing [2016] propose some heuristically strategies over SPL [Kumar *et al.*, 2010] for question answering. In order to avoid the exposure bias problem, Mixer [Ranzato *et al.*, 2015] also utilizes curriculum learning for training text generation model. For a sequence with the length of T, the first L tokens are cross-entropy loss while the last $T - L$ adopts reinforcement learning, and L is gradually reduced to zero from T.

Moreover, our work is related to KBQA. Berant *et al.* [2013] and Yih *et al.* [2016] utilize the semantic parsing based method for KBQA. [Yao and Van Durme, 2014; Bordes *et al.*, 2014a] adopt the information retrieval based method. Neural network-based method is used by [Bordes *et al.*, 2014b; 2015; Hao *et al.*, 2017]. KBQA devotes to obtaining correct answers in the form of entities while NAG task aims at generating correct answer entity as well as a natural expression.

6 Conclusion and Future Work

In this paper, we propose a curriculum learning based natural answer generation framework (CL-NAG). Under this framework, all valuable information in the noisy and uneven-quality corpora could be fully exploited. In particular, we employ two practical methods to automatically measure the complexity and quality of QA-pairs on the lexical level and sentential level, respectively. Based on these measures, CL-NAG is able to firstly learn a basic QA model with simple and low-quality QA-pairs (distilling correct entities from the low-quality and short answers to interact with the KB), and then gradually learn to produce better answers with complex and high-quality QA-pairs (generating natural responses). Experiments on a large-scale open dataset demonstrate the effectiveness of our model. Compared with the state-of-the-art, CL-NAG increases 6.8% and 8.7% in the accuracy for simple and complex questions, respectively. Moreover, it is able to deliver answers with richer contents and better syntax. In the future, we are planning to expand current work as follows. 1) Exploring more practical metrics to evaluate NAG and adopting these metrics to select high-quality QA-pairs; 2) Incorporating more kinds of data based on curriculum learning.

Acknowledgments

The work was supported by the Natural Science Foundation of China (No.61533018 and No.61702512). This work was also supported by Alibaba Group through Alibaba Innovative Research (AIR) Program and Huawei Tech. Ltm through Huawei Innovation Research Program.

References

[Bahdanau *et al.*, 2015] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. In *Proceedings of ICLR*, 2015.

[Bengio *et al.*, 2009] Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. Curriculum learning. In *Proceedings of ICML*, pages 41–48. ACM, 2009.

[Berant *et al.*, 2013] Jonathan Berant, Andrew Chou, Roy Frostig, and Percy Liang. Semantic parsing on Freebase from question-answer pairs. In *Proceedings of EMNLP*, pages 1533–1544, 2013.

[Bordes *et al.*, 2014a] Antoine Bordes, Sumit Chopra, and Jason Weston. Question answering with subgraph embeddings. In *Proceedings of EMNLP*, pages 615–620, 2014.

[Bordes *et al.*, 2014b] Antoine Bordes, Jason Weston, and Nicolas Usunier. Open question answering with weakly supervised embedding models. In *Proceedings of ECML-PKDD*, pages 165–180, 2014.

[Bordes *et al.*, 2015] Antoine Bordes, Nicolas Usunier, Sumit Chopra, and Jason Weston. Large-scale simple question answering with memory networks. *CoRR*, abs/1506.02075, 2015.

[Cho *et al.*, 2014] Kyunghyun Cho, Bart van Merrienboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using rnn encoder–decoder for statistical machine translation. In *Proceedings of EMNLP*, pages 1724–1734, 2014.

[Eric and Manning, 2017] Mihail Eric and Christopher D Manning. Key-value retrieval networks for task-oriented dialogue. In *Proceedings of SIGDIAL*, 2017.

[Gu *et al.*, 2016] Jiatao Gu, Zhengdong Lu, Hang Li, and Victor O.K. Li. Incorporating copying mechanism in sequence-to-sequence learning. In *Proceedings of ACL*, pages 1631–1640, 2016.

[Hao *et al.*, 2017] Yanchao Hao, Yuanzhe Zhang, Kang Liu, Shizhu He, Zhanyi Liu, Hua Wu, and Jun Zhao. An end-to-end model for question answering over knowledge base with cross-attention combining global knowledge. In *Proceedings of ACL*, pages 221–231, 2017.

[He *et al.*, 2017] Shizhu He, Cao Liu, Kang Liu, and Jun Zhao. Generating natural answers by incorporating copying and retrieving mechanisms in sequence-to-sequence learning. In *Proceedings of ACL*, pages 199–208, 2017.

[Hochreiter and Schmidhuber, 1997] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[Kumar *et al.*, 2010] M Pawan Kumar, Benjamin Packer, and Daphne Koller. Self-paced learning for latent variable models. In *Proceedings of NIPS*, pages 1189–1197, 2010.

[Levy and Manning, 2003] Roger Levy and Christopher D. Manning. Is it harder to parse chinese, or the chinese treebank? In *Proceedings of ACL*, pages 439–446, 2003.

[Li *et al.*, 2016] Jiwei Li, Will Monroe, Alan Ritter, Dan Jurafsky, Michel Galley, and Jianfeng Gao. Deep reinforcement learning for dialogue generation. In *Proceedings of EMNLP*, pages 1192–1202, 2016.

[Mou *et al.*, 2016] Lili Mou, Yiping Song, Rui Yan, Ge Li, Lu Zhang, and Zhi Jin. Sequence to backward and forward sequences: A content-introducing approach to generative short-text conversation. In *Proceedings of COLING*, pages 3349–3358, 2016.

[Novikova *et al.*, 2017] Jekaterina Novikova, Ondej Duek, Amanda Cercas Curry, and Verena Rieser. Why we need new evaluation metrics for nlg. In *Proceedings of EMNLP*, pages 2241–2252, 2017.

[Ranzato *et al.*, 2015] Marc'Aurelio Ranzato, Sumit Chopra, Michael Auli, and Wojciech Zaremba. Sequence level training with recurrent neural networks. *Proceedings of ICLR*, 2015.

[Sachan and Xing, 2016] Mrinmaya Sachan and Eric Xing. Easy questions first? a case study on curriculum learning for question answering. In *Proceedings of ACL*, pages 453–463, 2016.

[Salton and McGill, 1986] Gerard Salton and Michael J McGill. Introduction to modern information retrieval. 1986.

[Sharma *et al.*, 2017] Shikhar Sharma, Layla El Asri, Hannes Schulz, and Jeremie Zumer. Relevance of unsupervised metrics in task-oriented dialogue for evaluating natural language generation. *CoRR*, abs/1706.09799, 2017.

[Skinner, 1958] Burrhus F Skinner. Reinforcement today. *American Psychologist*, 13(3):94, 1958.

[Sutskever *et al.*, 2014] Ilya Sutskever, Oriol Vinyals, and Quoc V Le. Sequence to sequence learning with neural networks. In *Proceedings of NIPS*, pages 3104–3112, 2014.

[Unger *et al.*, 2014] Christina Unger, André Freitas, and Philipp Cimiano. An introduction to question answering over linked data. In *Reasoning Web. Reasoning on the Web in the Big Data Era*, pages 100–140. Springer, 2014.

[Yao and Van Durme, 2014] Xuchen Yao and Benjamin Van Durme. Information extraction over structured data: Question answering with freebase. In *Proceedings of ACL*, pages 956–966, 2014.

[Yih *et al.*, 2016] Wen-tau Yih, Matthew Richardson, Chris Meek, Ming-Wei Chang, and Jina Suh. The value of semantic parse labeling for knowledge base question answering. In *Proceedings of ACL*, pages 201–206, 2016.

[Yin *et al.*, 2016] Jun Yin, Xin Jiang, Zhengdong Lu, Lifeng Shang, Hang Li, and Xiaoming Li. Neural generative question answering. In *Proceedings of IJCAI*, 2016.