IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 31 May 2023, accepted 17 July 2023, date of publication 24 July 2023, date of current version 1 August 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3298100

RESEARCH ARTICLE

Read-All-in-Once (RAiO): Multi-Layer Contextual Architecture for Long-Text Machine Reading Comprehension

TUAN-ANH PHAN[1], **JASON J. JUNG**[2], **AND KHAC-HOAI NAM BUI**[1]
[1]Viettel Cyberspace Center, Viettel Group, Hanoi 11312, Vietnam
[2]Department of Computer Engineering, Chung-Ang University, Seoul 06974, South Korea
Corresponding author: Khac-Hoai Nam Bui (hoainam.bk2012@gmail.com)

ABSTRACT Machine reading comprehension (MRC) is a cutting-edge technology in natural language processing (NLP), which focuses on teaching machines to read and understand the meaning of texts based on the emergence of large-scale datasets and neural network models. Recently, with the successful development of pre-trained transformer models (e.g., BERT), MRC has advanced significantly, surpassing human parity in several public datasets and being applied in various NLP tasks (e.g., QA systems). Nevertheless, long document MRC is still a remain challenge since the transformer-based models are limited by the input length. For instance, several well-known pre-trained language models such as BERT and RoBERTa are limited by 512 tokens. This study aims to provide a new simple approach for long document MRC. Specifically, recent state-of-the-art models follow the architecture with two crucial stages for reading long texts in order to enable local and global context representations. In this study, we present a new architecture that is able to enrich the global information of the context with one stage by exploiting the interaction of different levels of semantic units of the context (i.e., sentence and word level). Therefore, we name the proposed model as RAiO (Read-All-in-Once) approach. For the experiment, we evaluate RAiO on two benchmark long document MRC datasets such as NewsQA and NLQuAD. Accordingly, the experiment shows promising results of the proposed approach compared with strong baselines in this research field.

INDEX TERMS Natural language processing, long-text machine reading comprehension, question-answering system.

I. INTRODUCTION

The task of machine reading comprehension (MRC) aims to answer the question given by reading and understanding a given document that plays an important role in the question-answering (QA) system. The traditional approach for dealing with this issue is considering it as a downstream task and leveraging the knowledge from the pre-trained language model to fine-tune it. For the last few years, this method has achieved promising results in high-quality data sets such as SQUAD [1], [2]. Nevertheless, the MRC for long texts still remains challenging due to the limitation of input length (e.g., 512 tokens) of the pre-train language model (PrLM) such as BERT [3] and RoBERTa [4].

Intuitively, the natural way to overcome this problem is based on the *sliding window approach*. This method first splits the whole document into a list of smaller chunks. Each chunk is then fed into the reader to predict the local answers. A local answer is a tuple that contains the predicted span text and a corresponding score. These scores are further compared with each other to choose the highest one. This method is straightforward, but it leads the severe problems due to the limitation in reading space in case the model must synthesize the content of multi-part of the document to reason the results [5], [6], [7]. Another stream of research uses the hierarchical networks, which is regarded as *coarse-to-fine approach* [8], [9], [10] with two crucial stages. Specifically,

The associate editor coordinating the review of this manuscript and approving it for publication was Cesar Vargas-Rosales.

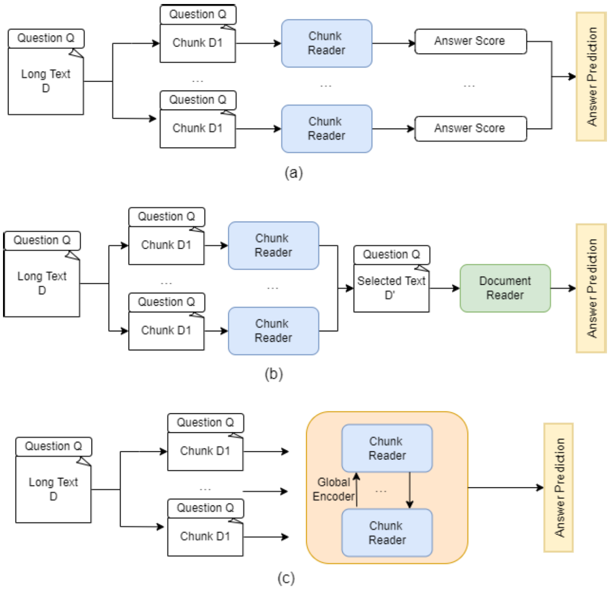

FIGURE 1. Comparative analysis of previous works with our approach: (a) sliding window approach selects the highest scores of the local answers [6], [7]; (b) coarse-to-fine approach first extracts the output of local answers from the chunk readers, the selected version of the long-text is then put into document reader for extract the global answers [8], [9], [10]; (c) Our proposed exploits global information for enriching each local chunk reader in order to directly extract the global answer.

the approach hypothesizes that a few pieces in the document are *sufficient* and *necessary* for fulfilling tasks. Therefore, they first create the condensed version of an original document by selecting several relevant sentences [8], [9] or aggregating the output of the *local answer* from the chunk reader [10]. Consecutively, the compact version of input is fed into a document reader to find the *global answers*. Although the above methods can extend the reading field by reading through the whole document and grabbing some parts from different locations, there are still two remaining drawbacks: i) the error accumulation when discretizing the whole process into two consecutive steps; and ii) lack of contextual information cause of using a small of amount text (chunk-level) instead of the full text (document-level).

In this study, we propose a multi-layer contextual architecture that is able to extract the *global answers* by considering the whole context of the document. Specifically, Figure 1 illustrates the main difference between our approach compared with previous works. Accordingly, the core idea of the proposed approach is to enrich the global information (document level) into each considered chunk (chunk level). In this regard, compared with previous works, the proposed approach is able to enable global answers without re-training the input texts. Specifically, the main contributions of this study are two folds as follows:

- We present a new approach for long-text machine reading comprehension, which is able to enable the global information into each chunk/segment by proposing a multi-layer contextual architecture.
- We execute the proposed model on two public datasets such as NewsQA [5] and NLQuAD [7]. The experimental results show promising results compared with strong baselines in this research field. Our source code is available for further investigations[1].

II. RELATED WORK

Machine reading comprehension for long texts has become an emergent research topic due to the limited length of pre-trained language models (PrLMs). Subsequently, the recent existing models for long-text MRC can be classified into three approaches such as i) *Long Document Modeling*; ii) *Slide Window*; and iii) *Coarse-to-Fine* approaches.

A. LONG DOCUMENT MODELING APPROACH

Several language models such as BERT and RoBERTa have been state-of-the-art for a while. However, the limitation of the input length is that those models can not process with longer sequences. In this regard, more efficient transformer models such as Longformer [11] and Big Bird [12] use sparse self-attention instead of full self-attention to process longer documents (e.g. up to 4,096 tokens). Although using Longformer and Big Bird has increased the performance of long-text MRC compared with baseline models such as BERT and RoBERTa, however, the attention mechanism of these methods is still designed manually, which may lead the insufficient interaction between tokens to model the document representation.

B. SLIDE WINDOW APPROACH

Due to the limitation of input length, a commonly used approach is to split the whole document into equal segments/chunks, as shown in Figure.1 (a). In this regard, each individual chunk is then put into a reader model to predict the answer (local answer). Subsequently, the global answer is then ensemble by local answers from multiple chunks [6], [7]. However, there are two main drawbacks of this approach: i) dividing the long text into equal chunks may result in incomplete answers, and ii) local answers are extracted independently in each chunk causing incomparable answer scores across chunks. In order to enable comparable answer scores across segments, Gong et. al [13] present a recurrent chunking mechanism to allow the information to flow across segments. Nonetheless, lacking the global information for extracting the answers is still an open research issue of this approach.

C. COARSE-TO-FINE APPROACH

Recent works focus on investigating long-text MRC with coarse-to-fine paradigms. The main idea is to expand the reading field from chunk-level to document-level with two stages such as chunk reader and document reader, as shown in Figure 1 (b). Ding et. al [9] first extract important sentences from long text using BERT. Those sentences are then concatenated as the input of another reader for extracting the

[1] https://github.com/tuananhphan97vn/RAiO-LongMRC

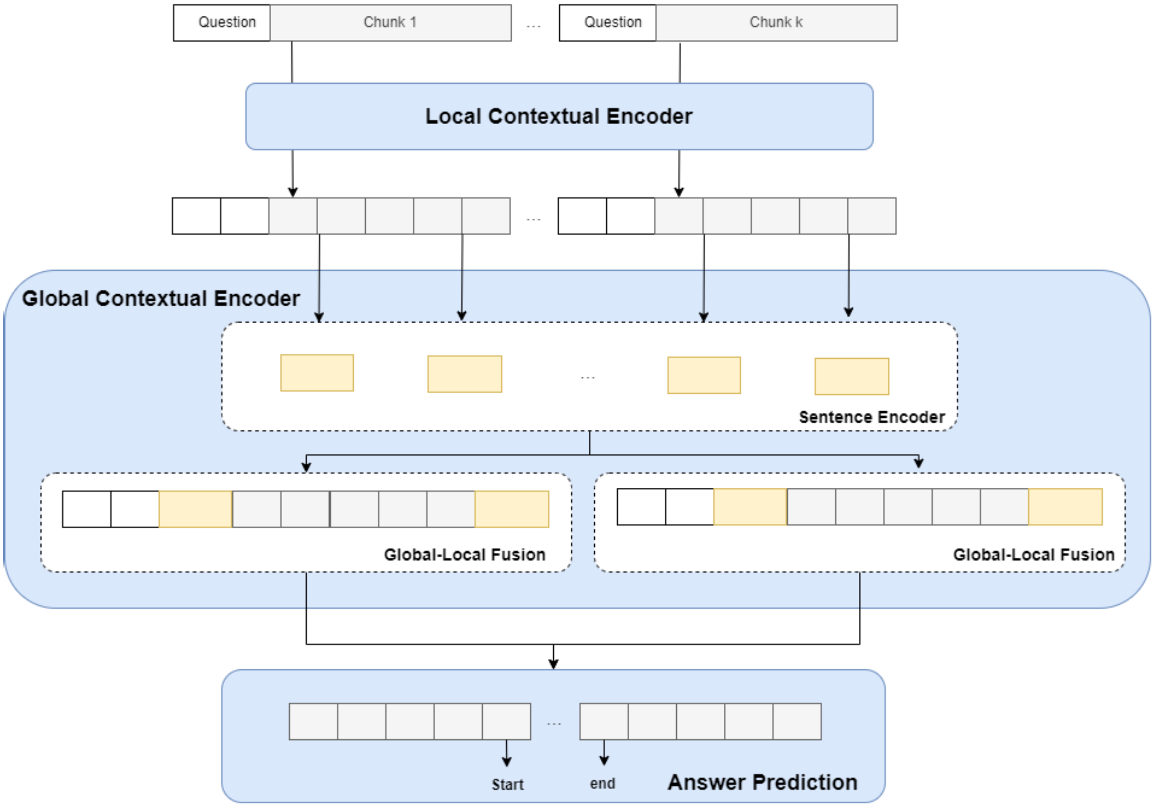

FIGURE 2. The architecture of the proposed Read-All-in-Once pipline.

global answer. On the other hand, Zhao et. al [10] utilized the regional answers of chunk readers to compact into a new document using a minimum span coverage algorithm. The condensed version of the input is further read by a document reader to predict global answers. Observationally, this approach is still not sufficient to fully capture long-range dependencies.

III. METHODOLOGY

We propose RAiO, a *Read-All-in-Once* approach, which is a multi-layer contextual architecture, to exploit the global information for each segment/chunk of the context. Specifically, RAiO comprises three main components, which are Local Contextual Encoder, Global Contextual Encoder, and Answer Prediction. Figure 2 illustrates the architecture of the proposed *Read-All-in-Once* pipeline.

A. LOCAL CONTEXTUAL ENCODER

Given D and Q denote the input long text and the question, respectively. In order to find the local contextual information, we first segment the input D into K smaller chunk:

$$\begin{aligned} D &= \{D_1, D_2, \ldots, D_K\} \\ D_k &= \{D_k^1, D_k^2, \ldots, D_k^n\} \end{aligned} \tag{1}$$

where n denotes the number of subwords/words of the chunk D_k. The goal of the local contextual encoder is to convert the input sequence into a series of contextualized feature representations. Accordingly, the question Q, the chunk D_k, and two special tokens CLS and SEP are concatenated to create the local input. In this regard, the hidden state $H_{D_k}^W$ of the chunk D_k can be acquired using a PrLM (e.g., RoBERTa):

$$H_{D_k} = [H_{CLS}|H_{Q^1}, \ldots, H_{Q^M}|H_{SEP}|H_{D_k^1}, \ldots, H_{D_k^n}] \tag{2}$$

where M denotes the number of tokens of the input question Q.

B. GLOBAL CONTEXTUAL ENCODER

To exploit the global information in each considered chunk, we first represent the whole text at the sentence level. The sentence information is then utilized for enabling the global information for each local information in the subword/word level.

1) SENTENCE ENCODER

For the sentence embedding process, we first split a document into a list of sentences using the NLTK library[2]. Now, each chunk D_k (Equation 1) is re-denoted in the sentence level as follows:

$$D_k = \{D_k^{s1}, D_k^{s2}, \ldots, D_k^{s_m}\} \tag{3}$$

where $\{s_1, \ldots, s_m\}$ denote the total sentences of the chunk D_k. Inspired by the work of [14], we implement the mean

[2] https://www.nltk.org/

pooling strategy for representing the sentence level from the token level. Specifically, the hidden state $H_{D_k}^{s_i}$ of a sentence s_i can be calculated as follows:

$$H_{D_k}^{s_i} = \left(\frac{1}{|D_k^{s_i}|} \sum_{j=0}^{j=|D_k^{s_i}|} H_j \right) W_s \tag{4}$$

where $|D_k^{s_i}|$ and W_s are the total tokens in sentence $D_k^{s_i}$ and the learnable parameter, respectively.

Subsequently, all sentences from all of the segments in the entire document are chained together to form:

$$H_S = \left(H_{D_k^{s_i}} \right) \quad \forall k, s_i \tag{5}$$

For propagating the sentence information throughout the whole document, we implement the full connection attention mechanism as follows:

$$H_S' = Attention(Q, K, V) = softmax\left(\frac{QK^T}{\sqrt{d}}\right) V \tag{6}$$

where $Q = H_S W_Q$, $K = H_S W_K$, and $V = H_S W_V$ are the query matrix, key matrix, and value matrix, respectively. Moreover, the residual connection and one layer norm as in [15] are also implemented for smoothing as follows:

$$H_S'' = LayerNorm(H_S + H_S') \tag{7}$$

In order to deeper exploit, we employ multi-stacked block sentence interaction with the number block as the hyperparameter of the model.

2) GLOBAL-LOCAL FUSION

Via the sentence interaction layer, the information of sentence in different location are transported to other. We continuously construct the sentence fusion layer to utilize them for enriching the tokens. Firstly, we detach the output of the interaction layer to the original list according to each chunk. We use a multi-stacked block, in which each block contains the self-attention layer, residual, and layer norm layer, similar to the sentence interaction block. To fuse the sentence knowledge to tokens level, for each chunk D_k, we create the input of sentence fusion block by concatenating the four objects: question embedding H_Q, list of sentence embedding of all chunk stand in front of the current chunk $\overleftarrow{H_{S_k}}$, all of the token of current chunk H_{D_k}, list of sentence of all chunk stand behind the current chunk $\overrightarrow{H_{S_k}}$:

$$H_{D_k}^{fusion} = H_Q \oplus \overleftarrow{H_{S_k}} \oplus H_{D_k} \oplus \overrightarrow{H_{S_k}} \tag{8}$$

Subsequently, the fusion hidden state of the chunk D_k is then used as the input for the full connection attention mechanism:

$$\hat{H}_{D_k}^{fusion} = Attention(Q, K, V) = softmax\left(\frac{QK^T}{\sqrt{d}}\right) V \tag{9}$$

where $Q = H_{D_k}^{fusion} W_Q$, $K = H_{D_k}^{fusion}$, and $V = H_{D_k}^{fusion} W_V$ are query matrix, key matrix, and value matrix, respectively. Intuitively, the global-local fusion layer has some advantages: i) the information from all tokens that are absent will be replaced by a sentence-level; ii) the number of sentences is extremely less than to a number of tokens, implement the sentence fusion does not violent the limitation of PrLMs. Finally, for spitting out the input for the next reasoning block, we ignore both directional sentences hidden states and only preserve then concatenate the hidden embedding of the question and tokens.

C. ANSWER PREDICTION

For the answer prediction, different from the sliding window approach that only depends on the local representation, we collect the local embedding from all segments and then stick them together to create the global representation. The input is continuously fed into the soft-max layer. The cross-entropy loss is used for the training model to predict both the start and end position of gold answers:

$$\begin{aligned} p^{start} &= softmax(W_{start} T^w) \\ p^{end} &= softmax(W_{end} T^w) \end{aligned} \tag{10}$$

The final loss function is formulated as follows:

$$Loss = -\frac{1}{N}\left(\sum_{i=0}^{i=N} log(p_{y_i}^{start}) + log(p_{y_i}^{end}) \right) \tag{11}$$

D. COMPUTATIONAL COMPLEXITY

For a better explanation of the efficiency of our approach compared and previous approaches (as shown in Figure 1), in this section, we analyze the computational complexity of each approach. Specifically, *sliding window approach* (Figure 1 (a)) takes the output of the pre-trained language models for extracting the local answer as the final output. In particular, given the length of the document $|D|$, the size of each chunk/segment $|C|$, the complexity of this approach is the complexity of the backbone models (e.g., BERT or RoBERTa), which can be formulated as follows:

$$O_{slided\_window} = O(L|D||C|) \tag{12}$$

where L denotes the number of layers of the backbone models (e.g., BERT-large has 24 layers). In order to enable the global answer, *coarse-to-fine approach* (Figure 1 (b)) selects the important information (e.g., key sentence) from all local information to put into another pre-trained language model, which can be simplify formulated as follows:

$$O_{coarse\_to\_fine} = O(L|D||C|) + O(L|C|^2) \tag{13}$$

Note that Equation 13 is the vanilla version, which adopts a simple method for extracting important information. For instance, CogLTX [9], a state-of-the-art model of this approach adopted multi-step reasoning to identify key sentences for the input of the second stage training, in which the computational complexity can be formulated as follows:

$$O_{coarse\_to\_fine}^{CogLTX} = O\left(L|C||D| + \frac{L|C|^3}{|S|} \right) + O(L|C|^2) \tag{14}$$

where $|S|$ denotes the length of sentences.

Regarding the proposed approach of this study, given N_S denotes the total number of sentences in the context D, the computational complexity of our approach can be calculated as follows:

$$O_{RAiO} = O(L|D||C|) + O\left(L'\left(\frac{|D|}{|C|}(|C| + N_S)^2 + N_S^2\right)\right) \tag{15}$$

where $O(L'\left(\frac{|D|}{|C|}(|C| + N_S)^2 + N_S^2\right))$ denotes the complexity of our global contextual block. In particular, due to $L' \ll L$ and $N_S \ll C$, Equation 15 can be approximately re-calculated as follows:

$$O_{RAiO} \approx O(L|D||C|) + O(L'|D||C|)) \tag{16}$$

Accordingly, since we define the number of the proposed global contextual block is varied in range from 1 to 3 ($L' = \{1, 2, 3\}$), which is much smaller than the number of layers in the pre-trained language model (e.g., $L = 24$ for BERT-large), the computational complexity in our approach is significantly lower than state-of-the-art models in this research field.

Intuitively, with the large pre-trained language models (e.g., GPT 3.5), which enable the longer input length (e.g., |C| = 4096 tokens), the complexity is increased significantly and requires a large number of resources for the execution. In this regard, for this study, we only consider the pre-trained language models with limited input length (e.g., 512 tokens) such as BERT or RoBERTa.

IV. EXPERIMENT

A. BENCHMARK DATASETS AND BASELINE MODELS

1) DATASETS

We use two well-known long-text datasets such as NewsQA [5] and NLQuad [7] for the evaluation. Specifically, **NewsQA** is the challenge machine reading comprehension datasets over 100k annotated question-answer pairs that are collected from the set of 10,000 news articles from CNN. The challenging trait of NewsQA derived from i) the arbitrary length of answers span instead of only single words or entities; ii) some question has no answer; and iii) a remarkable percentage of question require inference, paraphrasing, and synthesis across multiple sentences over the document. **NLQuAD** is a non-factoid long question-answering dataset that is collected from BBC news articles. NLQuAD considers the news articles as context documents, the interview sub-headings in the articles as questions, and the body paragraphs related to the sub-headings as the answers. The biggest challenge of NLquAD to others is the long length of both question and answer (even up to 500 tokens), which virtually consist of several contagious sentences. The details of both datasets can be found in table 1.

2) BASELINE MODELS

We compare our proposed method with the recent state-of-the-art models for long-text MRC, which belongs to different approaches such as vanilla sliding window (BERT large [3], RoBERTa large [4], and Retro-Reader [16]), coarse-to-fine approach (CogLTX [9]), and spare attention architecture approach (Longformer [11]) Specifically, the baseline models are sequentially described as follows:

TABLE 1. The statistic of NewsQA and NLQuAD. The $AvgQ_{len}$ and $AvgC_{len}$ are the average question length and text length, respectively.

Dataset	Train	Dev	Test	Avg Q_{len}	Avg C_{len}
NewsQA	97313	5456	5412	7.8	749.2
NLQuAD	24541	3017	3024	7.0	876.8

- BERT-based [3] and RoBERTa-based [4] are the backbone for almost MRC model. Fine-tuning the PrLM for MRC helps the model learn the mutual information between question-context, leading to promising results.
- Retro-reader [16] inspired the human-thinking process to exploit the two-stage MRC model: 1) reading briefly to investigate the interactions of question and context of document 2) reading thoroughly and verifying the unanswered question to obtain the final results.
- Longformer [11] is a variant of transformer architecture that takes into account the spare attention mechanism to deal with the long sequences. Longformer is able to handle the input of up to 4096 tokens (8 times longer than BERT [3]).
- CogLTX [9] is one of the approaches that allow BERT to be applied for long-text tasks. This method first extracts the relevant sentences to the question and then aggregates them as input for the final readers.

B. EXPERIMENT SETTINGS

For the backbone model, we choose the RoBERTa-large [4] with 24 layers, 1024 hidden, and 16 heads as the backbone of our method. For sentence tokenizers, we use the well-known Python library[3]. The number of global contextual encoder blocks is varied in the range from 1 to 3 in the validation set to find the optimal values. The number of sentence interaction blocks and sentence-token interaction blocks is set to 2. Regarding the training configuration, we set the maximum epoch as 5, the learning rate starts by 1e-5 and decays gradually after each epoch, The batch size for NewQA is 32 while that for NLQuAD is 4. The question length is truncated to 30 tokens, the segment length is 478 for both datasets. For the NewsQA, to drive the model to learn the unanswerable question, we set the gold answer as the CLS token. We conducted all the experiments on the NVIDIA A100 40GB VRAM. In terms of inference configurations, we set the maximum answer length for NewsQA and NLQuAD to be 300 and 500, respectively. More specifically, for NewsQA that consists of some unanswerable question, we use the strategy *threshold-based answerable verification* as [16] with the threshold ρ is given and the score is computed for deciding the no-answer

[3] https://www.nltk.org/

score as:

$$score_{diff} = score_{null} - score_{has} \tag{17}$$
$$= log(H_{CLS}) - (log(H_{start}) + log(H_{end})) \tag{18}$$

If $\rho \leq score_{diff}$, the model predicts the *null answer* and predicts the span text in other cases. In our implementations, we vary the value of ρ in $[-2, -1, -0.5, 0, 0.5, 1, 2]$ in the validation set to find the best number and use it for the test set.

C. MAIN RESULTS

1) MEASUREMENT METRICS

Regarding the evaluation metric, we report the result according to two well-known metrics: Exact Match (EM) and F1. EM is a hard metric that scores each sample to receive the binary value: 1 in case the output of the model matches precisely with one of the gold spans of annotators and 0 otherwise. In contrast, F1 is a soft metric that simply computes the overlap between the output of the model with the correct answers.

2) MAIN RESULTS

Table. 2 show the results of our model compared with previous works on the two evaluated datasets. The results are presented in different sections corresponding to different approaches. The first section includes the sliding window approach with different versions (e.g., base and large) of two well-known PrLMs (i.e., BERT and RoBERTa). The second section obtains the result of Longformer, a long document modeling model using the spare-attention mechanism. The third section includes the models of the coarse-to-fine approach. The last section is our model, RAiO, which is based on *read-all-in-once* approach.

TABLE 2. Report results on the test set of two evaluated datasets. The bold texts indicate the best results.

Models	NewsQA		NLQuAD	
	EM	F1	EM	F1
BERT-base [3]	-	-	25.0	64.0
BERT-large [3]	46.5	56.7	30.3	67.9
RoBERTa-base [4]	-	-	29.1	67.2
RoBERTa-large [4]	49.6	66.3	33.4	71.1
Longformer [11]	-	-	50.3	81.4
Retro-Reader [16]	55.9	66.8	-	-
CogLTX [9]	55.2	70.1	-	-
RAiO (Ours)	**64.2**	**72.6**	**55.9**	**83.8**

Accordingly, our proposed model, which is based on the RoBERTa-large version, can improve upon 14.6% EM and 6.3% F1 for the NewsQA and 22.5% EM and 12.7% F1 for NLQuAD dataset compared with the sliding window approach. We hypothesize that there are many gold answers of two benchmark datasets that belong to two different chunks/segments. In this regard, approaches that consider global information such as coarse-to-fine (e.g., Retro-Reader and CogLTX) and the proposed approach in this study can perform better than extracting the local answers in each chunk/segment of the sliding window approach. Furthermore, compared with the coarse-to-fine approaches, our model outperforms CogLTX (+9% EM and +2.5% F1 on NewsQA) and Retro-Reader (+8.3% EM and +5.8% F1 on NewsQA). The main difference between our proposed approach and the coarse-to-fine approach is that the coarse-to-fine approach only selects the relevant information (e.g., key sentences) of each chunk/segment, which might lead to the lack of fully capturing long-range dependencies. In contrast, our approach, by proposing the global contextual encoder block, is able to utilize the whole context for extracting the global answer. The reported results indicate clearly the efficiency of our study. We also evaluate RAiO with the longer modeling approach using the spare-attention mechanism (i.e., Longformer with 4096 input token length). As results, our proposed model using RoBERTa as the backbone model (with 512 input token length) outperforms Longformer (+5.6% EM and 2.4% F1 on NLQuAD). Specifically, the experiment results indicate that our proposed model is able to enable the cross-information between chunks/segments, which proves the capability of our model for handling the long-text MRC task

D. ABLATION STUDIES

To further investigate the role of each component in the global contextual block, we conduct an ablation study, in which are sequentially described as follows:

- **w/o sentence interaction**: We remove the sentence-interaction block (Equation 6), only retaining the sentence encoder and the global-local fusion modules. In this experiment, there is no mechanism for global interaction, the regional tokens information of each segment is enriched simply by the sentence in this segment.
- **w/o global-local interaction**: Instead of using the global-local fusion (Equation 9), we simply aggregate the information of each token by the corresponding sentence that it belongs to. The integrate function is add-operation.

TABLE 3. Our ablation studies.

Models	NewsQA		NLQuAD	
	EM	F1	EM	F1
w/o sentence interaction	63.9	72.3	55.8	82.2
w/o global-local interaction	63.6	72.3	54.2	82.8
Full model	**64.2**	**72.6**	**55.9**	**83.8**

Table 3 shows the results of our ablations in both evaluated datasets. It is clear that the overall model witnesses the highest results in both matrices, which proves the advantage of each component in the proposed global contextual encoder block.

Furthermore, to investigate the influence of the number of global-local encoder blocks on overall results, we conduct extensive experiments with different reasoning blocks ranging from 1 to 3. The detailed results are presented in table 4. Accordingly, the best value for NewsQA is 3 while the number for NLQuAD is 2. This observation indicates that the number of blocks is different and depends on each dataset.

TABLE 4. Our results with different values of n-block.

Block Num.	NewsQA		NLQuAD	
	EM	F1	EM	F1
1	63.6	72.0	55.0	82.7
2	64.0	72.6	**55.9**	**83.8**
3	**64.2**	**72.6**	55.6	83.5

V. CONCLUSION

In this study, we propose a new approach for long-text MRC. Specifically, due to the limitation of the input length of the current PrLMs, long-text MRC is still an open research issue in this research field. A traditional method is to split the input long text into multiple chunks/segments. In this paper, we present a novel method that enriches global information into each local information (chunk reader) by exploiting the sentence level in each document. The experiment on two benchmark datasets such as NewsQA and NLQuAD shows the promising results of the proposed method with strong baseline models in this research field.

REFERENCES

[1] P. Rajpurkar, J. Zhang, K. Lopyrev, and P. Liang, "SQuAD: 100,000+ questions for machine comprehension of text," in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP)*. Austin, TX, USA: The Association for Computational Linguistics, Nov. 2016, pp. 2383–2392.

[2] P. Rajpurkar, R. Jia, and P. Liang, "Know what you don't know: Unanswerable questions for SQuAD," in *Proc. 56th Annu. Meeting Assoc. Comput. Linguistics*, Melbourne, VIC, Australia, Jul. 2018, pp. 784–789.

[3] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics, Human Lang. Technol. (NAACL-HLT)*, J. Burstein, C. Doran, and T. Solorio, Eds. Minneapolis, MN, USA: Association for Computational Linguistics, Jun. 2019, pp. 4171–4186.

[4] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, "RoBERTa: A robustly optimized BERT pretraining approach," 2019, *arXiv:1907.11692*.

[5] A. Trischler, T. Wang, X. Yuan, J. Harris, A. Sordoni, P. Bachman, and K. Suleman, "NewsQA: A machine comprehension dataset," in *Proc. 2nd Workshop Represent. Learn. NLP*, P. Blunsom, A. Bordes, K. Cho, S. B. Cohen, C. Dyer, E. Grefenstette, K. M. Hermann, L. Rimell, J. Weston, and S. Yih, Eds. Vancouver, BC, Canada: Association for Computational Linguistics, Aug. 2017, pp. 191–200.

[6] M. Joshi, E. Choi, D. S. Weld, and L. Zettlemoyer, "TriviaQA: A large scale distantly supervised challenge dataset for reading comprehension," in *Proc. 55th Annu. Meeting Assoc. Comput. Linguistics (ACL)*, R. Barzilay and M.-Y. Kan, Eds. Vancouver, BC, Canada: Association for Computational Linguistics, Jul./Aug. 2017, pp. 1601–1611.

[7] A. Soleimani, C. Monz, and M. Worring, "NLQuAD: A non-factoid long question answering data set," in *Proc. 16th Conf. Eur. Chapter Assoc. Comput. Linguistics*. Vancouver, BC, Canada: Association for Computational Linguistics, Apr. 2021, pp. 1245–1255.

[8] E. Choi, D. Hewlett, J. Uszkoreit, I. Polosukhin, A. Lacoste, and J. Berant, "Coarse-to-fine question answering for long documents," in *Proc. 55th Annu. Meeting Assoc. Comput. Linguistics (ACL)*, R. Barzilay and M.-Y. Kan, Eds. Vancouver, BC, Canada: Association for Computational Linguistics, Jul./Aug. 2017, pp. 209–220.

[9] M. Ding, C. Zhou, H. Yang, and J. Tang, "CogLTX: Applying BERT to long texts," in *Proc. Adv. Neural Inf. Process. Syst., Annu. Conf. Neural Inf. Process. Syst. (NeurIPS)*, H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, Eds., Dec. 2020, pp. 12792–12804.

[10] J. Zhao, J. Bao, Y. Wang, Y. Zhou, Y. Wu, X. He, and B. Zhou, "RoR: Read-over-read for long document machine reading comprehension," in *Proc. Findings Assoc. Comput. Linguistics, (EMNLP)*, M. Moens, X. Huang, L. Specia, and S. Yih, Eds., Nov. 2021, pp. 1862–1872.

[11] I. Beltagy, M. E. Peters, and A. Cohan, "LongFormer: The long-document transformer," 2020, *arXiv:2004.05150*.

[12] M. Zaheer, G. Guruganesh, K. A. Dubey, J. Ainslie, C. Alberti, S. Ontañón, P. Pham, A. Ravula, Q. Wang, L. Yang, and A. Ahmed, "Big Bird: Transformers for longer sequences," in *Proc. Adv. Neural Inf. Process. Syst., Annu. Conf. Neural Inf. Process. Syst. (NeurIPS)*, H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, Eds., Dec. 2020, pp. 17283–17297.

[13] H. Gong, Y. Shen, D. Yu, J. Chen, and D. Yu, "Recurrent chunking mechanisms for long-text machine reading comprehension," in *Proc. 58th Annu. Meeting Assoc. Comput. Linguistics, (ACL)*, D. Jurafsky, J. Chai, N. Schluter, J. R. Tetreault, Eds., Jul. 2020, pp. 6751–6761.

[14] N. Reimers and I. Gurevych, "Sentence-BERT: Sentence embeddings using siamese BERT-networks," in *Proc. Conf. Empirical Methods Natural Lang. Process., 9th Int. Joint Conf. Natural Lang. Process. (EMNLP-IJCNLP)* K. Inui, J. Jiang, V. Ng, and X. Wan, Eds. Hong Kong: Association for Computational Linguistics, Nov. 2019, pp. 3980–3990.

[15] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst., Annu. Conf. Neural Inf. Process. Syst.*, I. Guyon, U. von Luxburg, S. Bengio, H. M. Wallach, R. Fergus, S. V. N. Vishwanathan, and R. Garnett, Eds., Long Beach, CA, USA, 2017, pp. 5998–6008.

[16] Z. Zhang, J. Yang, and H. Zhao, "Retrospective reader for machine reading comprehension," in *Proc. 31th AAAI Conf. Artif. Intell. (AAAI), 33rd Conf. Innov. Appl. Artif. Intell. (IAAI), 11th Symp. Educ. Adv. Artif. Intell., (EAAI)*, Feb. 2021, pp. 14506–14514.

TUAN-ANH PHAN received the B.S. degree in computer science from the Hanoi University of Science and Technology, in August 2020. He is currently an AI Engineer with the Viettel Cyberspace Center, Viettel Group. His current research interests include applying graph neural networks for NLP tasks, such as text summarization and question-answering systems.

JASON J. JUNG received the B.Eng. degree in computer science and mechanical engineering and the M.S. and Ph.D. degrees in computer and information engineering from Inha University, in 1999, 2002, and 2005, respectively.

He was a Visiting Scientist with the Fraunhofer Institute (FIRST), Berlin, Germany, in 2004. He was also a Postdoctoral Researcher with INRIA Rhone-Alpes, France, in 2006. He has been a Full Professor with Chung-Ang University, South Korea, since September 2014. Before joining CAU, he has been an Assistant Professor with Yeungnam University, South Korea, since 2007. Recently, he has been working on intelligent schemes to understand various social dynamics in large-scale social media (e.g., Twitter and Flickr). His research interests include knowledge engineering on social networks by using many types of AI methodologies, e.g., data mining, machine learning, and logical reasoning.

KHAC-HOAI NAM BUI received the M.S. degree in computer science and information engineering from Aletheia University, New Taipei City, Taiwan, in 2014, and the Ph.D. degree in computer science and engineering from Chung-Ang University, Seoul, South Korea, in 2018.

He has been a Researcher with the Viettel Cyberspace Center, Viettel Group, Hanoi, Vietnam, since 2021. Before joining Viettel, he was a Researcher with the Korea Institute of Science and Technology Information (KISTI), Daejeon, South Korea, in September 2019. His research interests include intelligent systems using AI methodologies, such as data mining, machine learning, and logical reasoning. Recently, he has focused on applying deep learning models to NLP tasks.