Simple and Effective Curriculum Pointer-Generator Networks for Reading Comprehension over Long Narratives

[1]Yi Tay, [2]Shuohang Wang, [3]Luu Anh Tuan, [4]Jie Fu, [5]Minh C. Phan
[6]Xingdi Yuan, [7]Jinfeng Rao,* [8]Siu Cheung Hui, [9]Aston Zhang
[1,5,8]Nanyang Technological University [2]Singapore Management University [3]MIT CSAIL
[4]Mila, Polytechnic Montréal [6]Microsoft Research Montréal [7]Facebook AI [9]Amazon AI
[1]ytay017@e.ntu.edu.sg

Abstract

This paper tackles the problem of reading comprehension over long narratives where documents easily span over thousands of tokens. We propose a curriculum learning (CL) based Pointer-Generator framework for reading/sampling over large documents, enabling diverse training of the neural model based on the notion of alternating contextual difficulty. This can be interpreted as a form of domain randomization and/or generative pretraining during training. To this end, the usage of the Pointer-Generator softens the requirement of having the answer within the context, enabling us to construct diverse training samples for learning. Additionally, we propose a new Introspective Alignment Layer (IAL), which reasons over decomposed alignments using block-based self-attention. We evaluate our proposed method on the NarrativeQA reading comprehension benchmark, achieving state-of-the-art performance, improving existing baselines by 51% relative improvement on BLEU-4 and 17% relative improvement on Rouge-L. Extensive ablations confirm the effectiveness of our proposed IAL and CL components.

1 Introduction

Teaching machines to read and comprehend is a fundamentally interesting and challenging problem in AI research (Hermann et al., 2015; Trischler et al., 2016; Rajpurkar et al., 2016). While there have been considerable and broad improvements in reading and understanding textual snippets, the ability for machines to read/understand complete stories and novels is still in infancy (Kočiskỳ et al., 2018). The challenge becomes insurmountable in lieu of not only the large context but also the intrinsic challenges of narrative text which arguably requires a larger extent of reasoning. As such, this motivates the inception of relevant, interesting benchmarks such as the NarrativeQA Reading Comprehension challenge[1] (Kočiskỳ et al., 2018).

The challenges of having a long context have been traditionally mitigated by a two-step approach - retrieval first and then reading second (Chen et al., 2017; Wang et al., 2018; Lin et al., 2018). This difficulty mirrors the same challenges of open domain question answering, albeit introducing additional difficulties due to the nature of narrative text (stories and retrieved excerpts need to be coherent). While some recent works have proposed going around by training retrieval and reading components end-to-end, this paper follows the traditional paradigm with a slight twist. We train our models to be robust regardless of whatever is retrieved. This is in similar spirit to domain randomization (Tobin et al., 2017).

In order to do so, we propose a diverse curriculum learning scheme (Bengio et al., 2009) based on two concepts of difficulty. The first, depends on whether the answer exists in the context (*answerability*), aims to bridge the gap between training time and inference time retrieval. On the other hand, and the second, depends on the size of retrieved documents (coherence and *understandability*). While conceptually simple, we found that these heuristics help improve performance of the QA model. To the best of our knowledge, we are the first to incorporate these notions of difficulty in QA reading models.

All in all, our model tries to learn to generate the answer even if the correct answer does not appear as evidence which acts as a form of *generative pretraining during training*. As such, this is akin to learning to guess, largely motivated by how

*Work done while at University of Maryland.

[1] We tackle the full story setting instead of the summary setting which, inherently, is a much harder task.

Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pages 4922–4931
Florence, Italy, July 28 - August 2, 2019.

humans are able to extrapolate/guess even when given access to a small fragment of a film/story. In this case, we train our model to generate answers, making do with whatever context it was given. To this end, a curriculum learning scheme controls the extent of difficulty of the context given to the model.

At this juncture, it would be easy to realize that standard pointer-based reading comprehension models would not adapt well to this scheme, as they fundamentally require the golden label to exist within the context (Wang and Jiang, 2016b; Seo et al., 2016). As such, our overall framework adopts a pointer-generator framework (See et al., 2017) that learns to point and generate, conditioned on not only the context but also the question. This relaxes this condition, enabling us to train our models with diverse views of the same story which is inspired by domain randomization (Tobin et al., 2017). For our particular task at hand, the key idea is that, even if the answer is not found in the context, we learn to generate the answer despite the noisy context.

Finally, our method also incorporates a novel Introspective Alignment Layer (IAL). The key idea of the IAL mechanism is to introspect over decomposed alignments using block-style local self-attention. This not only imbues our model with additional reasoning capabilities but enables a finer-grained (and local-globally aware) comparison between soft-aligned representations. All in all, our IAL mechanism can be interpreted as learning a matching over matches.

Our Contributions All in all, the prime contributions of this work is summarized as follows:

- We propose a curriculum learning based Pointer-Generator model for reading comprehension over narratives (long stories). For the first time, we propose two different notions of difficulty for constructing diverse views of long stories for training. We show that this approach achieves better results than existing models adapted for open-domain question answering.

- Our proposed model incorporates an Introspective Alignment Layer (IAL) which uses block-based self-attentive reasoning over decomposed alignments. Ablative experiments show improvements of our IAL layer over the standard usage of vanilla self-attention.

- Our proposed framework (IAL-CPG) achieves state-of-the-art performance on the NarrativeQA reading comprehension challenge. On metrics such as BLEU-4 and Rouge-L, we achieve a 17$\%$ relative improvement over prior state-of-the-art and a **10** times improvement in terms of BLEU-4 score over BiDAF, a strong span prediction based model.

- We share two additional contributions. Firstly, we share negative results on using Reinforcement Learning to improve the quality of generated answers (Paulus et al., 2017; Bahdanau et al., 2016). Secondly, we show that the evaluation scheme in NarrativeQA is flawed and models can occasionally generate satisfactory (correct) answers but score zero points during evaluation.

2 Our Proposed Framework

This section outlines the components of our proposed architecture. Since our problem is mainly dealing with extremely long sequences, we employ an initial retrieval[2] phrase by either using the answer or question as a cue (query for retrieving relevant chunks/excerpts). The retrieval stage is controlled by our curriculum learning process in which the details are deferred to subsequent sections. The overall illustration of this framework is depicted in Figure 1.

2.1 Introspective Alignment Reader

This section introduces our proposed Introspective Alignment Reader (IAL-Reader).

Input and Context Encoding Our model accepts two inputs, (context C and question Q). Each input is a sequence of words. We pass each sequence into a shared Bidirectional LSTM layer.

$$H^c = \text{BiLSTM}(C) \quad , \quad H^q = \text{BiLSTM}(Q)$$

where $H^c \in \mathbb{R}^{\ell_c \times d}$ and $H^q \in \mathbb{R}^{\ell_q \times d}$ are the hidden representations for C and Q respectively.

Introspective Alignment Next, we pass H^c, H^q into an alignment layer. Firstly, we compute a soft attention affinity matrix between H^c and H^q as follows:

$$E_{ij} = F(h_i^c)^\top \, F(h_j^q) \qquad (1)$$

[2] This is unavoidable since supporting up to 20K-30K words in computational graphs is still not manageable even with top-grade GPUs.

三个模块十课程学习

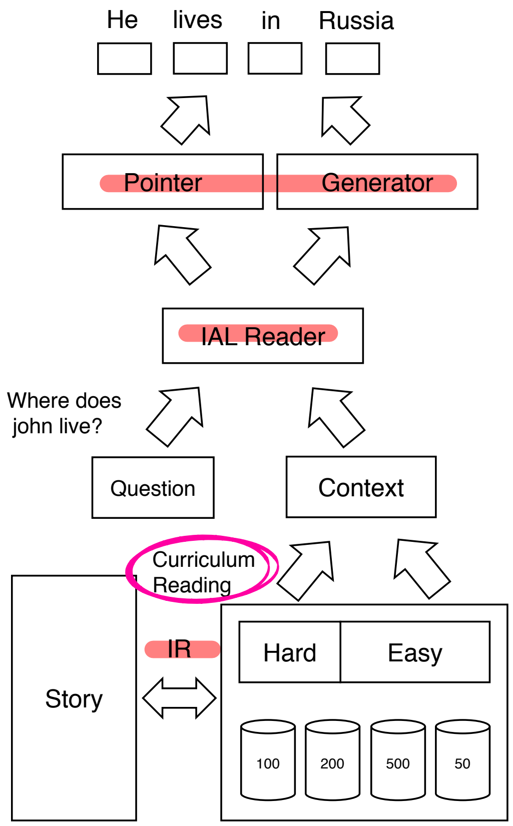

Figure 1: Illustration of our proposed IAL-CPG framework.

where h_i^c is the i-th word in the context and h_j^q is the j-th word in the question. $F(\cdot)$ is a standard nonlinear transformation function (i.e., $F(x) = \sigma(Wx + b)$, where σ indicates non-linearity function), and is shared between context and question. $E \in \mathbb{R}^{\ell_c \times \ell_q}$ is the soft matching matrix. To learn alignments between context and question, we compute:

$$A = \text{Softmax}(E)\, H^q$$

where $A \in \mathbb{R}^{\ell_c \times d}$ is the aligned representation of H^c.

Reasoning over Alignments Next, to reason over alignments, we compute a self-attentive reasoning over decomposed alignments:

$$G_{ij} = F_s([A_i; H_i^c; A_i - H_i^c, A_i \odot H_i^c])^\top \cdot F_s([A_j; H_j^c; A_j - H_j^c, A_j \odot H_j^c]) \quad (2)$$

where square brackets $[\cdot;\cdot]$ denote vector concatenation, $F_s(\cdot)$ is another nonlinear transformation layer which projects onto $4d$ dimensions. i is the positional index of each word token. Intuitively, A_i comprises of softly aligned question representations with respect to the context. The usage of the *Hadamard* and *Subtraction* operators helps to enhance the degree of comparison/matching. Hence, by including an additional local reasoning over these enhanced alignment vectors, our model can be interpreted as introspecting over alignment matches.

Local Block-based Self-Attention Since ℓ_c is large in our case (easily ≥ 2000), computing the above Equation (2) may become computationally prohibitive. As such, we compute the scoring function for all cases where $|i - j| \leq b$, in which, b is a predefined hyperparameter and also the block size. Intuitively, the initial alignment layer (i.e., Equation 1) already considers a global view. As such, this self-attention layer can be considered as a local-view perspective, confining the affinity matrix computation to a local window of b. Finally, to compute the introspective alignment representation, we compute:

$$B = \text{Softmax}(G)\, [A; H^c; A - H^c; A \odot H^c]$$

where $B^{\ell_c \times 4d}$ is the introspective aligned representation of A. Finally, we use another d dimensional BiLSTM layer to aggregate the aligned representations:

$$Y = \text{BiLSTM}([B; A; H^c; A - H^c; A \odot H^c]) \quad (3)$$

where $Y \in \mathbb{R}^{\ell_c \times 2d}$ is the final contextual representation of context C.

2.2 Pointer-Generator Decoder

Motivated by recent, seminal work in neural summarization, our model adopts a pointer-generator architecture (See et al., 2017). Given Y (the question infused contextual representation), we learn to either generate a word from vocabulary, or point to a word from the context. The decision to generate or point is controlled by an additive blend of several components such as the previous decoder state and/or question representation.

The pointer-generator decoder in our framework uses an LSTM decoder[3] with a cell state $c_t \in \mathbb{R}^n$ and hidden state vector $h_t \in \mathbb{R}^n$. At

[3] To initialize the LSTM, we use an additional projection layer over the mean pooled representation of Y similar to (Xu et al., 2015).

each decoding time step t, we compute an attention over Y as follows:

$$g_i = \tanh(\underbrace{\overset{\text{非线性变换}}{\boxed{F_a}}(y_i)}_{R^n} + \underbrace{\boxed{F_h}(h_{t-1})}_{R^n} + \underset{\text{attentive pooling}}{\boxed{F_q}}(H^q)), \tag{4}$$

$$a_i = g_i^\top \underset{\text{参数}}{\boxed{w_a}}\ , \quad \underset{\text{attention 向量}}{y_t} = \sum_{i=0}^{\ell_c} a_i \cdot y_i \tag{5}$$

where $F_a(\cdot)$ and $F_h(\cdot)$ are nonlinear transformations projecting to n dimensions. i is the position index of the input sequence. $F_q(\cdot)$ is an additional attentive pooling operator over the question representation H_q (after the context encoding layer). The semantics of the question may be lost after the alignment based encoding. As such, this enables us to revisit the question representation to control the decoder. $y_t \in \mathbb{R}^n$ is the context representation at decoding time step t and $a \in \mathbb{R}^{\ell_c}$ is an attention distribution over the context words which is analogous to the final probability distributions that exist in typical span prediction models. Next, we compute the next hidden state via:

$$h_t, c_t = \text{LSTM}([\underset{\text{attention 向量}}{y_t}; \overset{\text{上一时刻的输出}}{w_{t-1}}], h_{t-1}, c_{t-1})$$

where w_{t-1} is the $(t-1)_{th}$ token in the ground truth answer (teacher forcing). To learn to generate, we compute:

$$\overset{\text{生成的 word 概率分布}}{\boxed{v_t}} = W_v(h_t) + b_v \tag{6}$$

where $v_t \in \mathbb{R}^{|V_g|}$, V_g is the global vocabulary size. The goal of the pointer-generator decoder is to choose between the abstractive distribution v_t over the vocabulary (see Equation 6) and the extractive distribution a_t (see Equation 5) over the context text tokens. To this end, we learn a scalar switch $p_t \in \mathbb{R}$:

$$\overset{\text{gate}}{\boxed{p_t}} = \text{sigmoid}(\overset{\text{无bias的线性变换}}{\boxed{F_{pc}}(c_t) + \boxed{F_{ph}}(h_t) + \boxed{F_{py}}}(y_t))$$

where $F_{pc}(\cdot), F_{ph}(\cdot), F_{py}(\cdot)$ are linear transformation layers (without bias) which project c_t, h_t and y_t into scalar values. To control the blend between the attention context and the generated words, we use a linear interpolation between a_t and v_t. The predicted word w_t at time step t is therefore:

$$\overset{\text{t时刻最终的预测词}}{\boxed{w_t}} = \text{argmax}(p_t \cdot a_t + (1 - p_t)v_t)$$

Note that we scale (append and prepend) a_t and v_t with zeros to make them the same length (i.e., $\ell_c + |V_g|$). The LSTM decoder runs for a predefined fix answer length. During inference, we simply use greedy decoding to generate the output answer.

2.3 Curriculum Reading

A key advantage of the pointer-generator is that it allows us to generate answers even if the answers do not exist in the context. This also enables us to explore multiple (diverse) views of contexts to train our model. However, to this end, we must be able to identify effectively the most useful retrieved context evidences for the training. For that purpose, we propose to use a diverse curriculum learning scheme which is based on two intuitive notions of difficulty:

Answerability - It is regarded as common practice to retrieve excerpts based by using the correct answer as a cue (during training). This establishes an additional gap between training and inference since during inference, correct answers are not available. This measure aims to bridge the gap between question and answer (as a query prompt for passage retrieval). In this case, we consider the set of documents retrieved based on questions as the *hard* setting, H. Conversely, the set of retrieved documents using answers is regarded as the *easy* setting, E.

Understandability - This aspect controls how understandable the overall retrieved documents are as a whole. The key idea of this setting is to control the paragraph/chunk size. Intuitively, a small paragraph/chunk size would enable more relevant components to be retrieved from the document. However, its understandability might be affected if paragraph/chunk size is too small. Conversely, a larger chunk size would be easier to be understood. To control the level of understandability, we pre-define several options of chunk sizes (e.g., $\{50, 100, 200, 500\}$) which will be swapped and determined during training.

To combine the two measures described above, we comprise an easy-hard set pair for each chunk size, i.e., $\{E_k, H_k\}$, where:

$$\begin{aligned} k &\in \{50, 100, 200, 500\}, \\ E_n &\leftarrow F(\text{corpus}, \text{answer}, n), \\ H_n &\leftarrow F(\text{corpus}, \text{question}, n) \end{aligned} \tag{7}$$

$F(.)$ is an arbitrary ranking function which may or may not be parameterized, and n is the size of each retrieved chunk.

Two-layer Curriculum Reading Algorithm. As our model utilizes two above measures of difficulty, there lies a question on which whether we

Algorithm 1 Curriculum Reading

```
1: $chunk\_list \leftarrow \{50, 100, 200, 500\}$
2: $n \leftarrow sample\ i\ in\ chunk\_list$
3: $chunk\_list \leftarrow chunk\_list \setminus \{n\}$
4: $E_n \leftarrow F(Corpus, Answers, n)$
5: $H_n \leftarrow F(Corpus, Questions, n)$
6: $D \leftarrow E_n$    ▷ initial training set
7: $count \leftarrow 0$    ▷ number of swaps within a chunk size
8: for $i \leftarrow 1$ to $numEpochs$ do
9:     $Train(D)$
10:    $score \leftarrow Evaluate(Dev\_set)$
11:    if $score < bestDev$ then
12:        if $count <= 1/\delta$ then
13:            $D \leftarrow Swap(D, E_n, H_n, \delta)$    ▷ Swap $\delta$ percent of easy set in $D$ with the hard set
14:            $count \leftarrow count + 1$
15:        else
16:            $Repeat\ step\ 3\ to\ 8$    ▷ Replace training set with new easy set of another chunk size
17:    else
18:        $bestDev = score$
```

should swap one measure at a time or swap both whenever the model meets the failure criterion. In our case, we find that prioritizing answerability over understandability is a better choice. More concretely, at the beginning of the training, we start with an easy set E_k of a random chunk size k. When the failure criterion is met (e.g. the model score does not improve on the validation set), we randomly swap a small percent δ (e.g., 5% in our experiments[4]) of the easy set E_k with the hard set H_k within its own chunk size group k to improve the $answerability$. In this case, after $\frac{1}{\delta}$ failures, the model runs out of easy set E_k and is completely based on the hard set H_k. At this junction, we swap the model for *understandability*, replacing the training set with a completely new easy set E_l of another chunk size l, and repeat the above process. The formal description of our proposed curriculum reading is introduced in Algorithm 1.

3 Experiments

We conduct our experiments on the NarrativeQA reading comprehension challenge.

3.1 Experimental Setup

This section introduces our experimental setups.

Model Hyperparameters We implement our model in Tensorflow. Our model is trained with Adadelta (Zeiler, 2012). The initial learning rate is tuned amongst $\{0.1, 0.2, 0.5\}$. The L2 regularization is tuned amongst $\{10^{-8}, 10^{-6}, 10^{-5}\}$. The size of the LSTM at the encoder layer is set to 128 and the decoder size is set to 256. The block size b for the Introspective Alignment Layer is set to 200. We initialize our word embeddings with pre-trained GloVe vectors (Pennington et al., 2014) which are not updated[5] during training.

Implementation Details Text is lowercased and tokenized with NLTK[6]. For retrieval of paragraphs, we use the cosine similarity between TF-IDF vector representations. TF-IDF representations are vectorized by Scikit-Learn using an N-gram range of $[1, 3]$ with stopword filtering. The maximum context size is tuned amongst $\{2000, 4000\}$ and reported accordingly. The paragraph/chunk size is dynamic and configured amongst $\{50, 100, 200, 500\}$. The retrieved excerpts are retrieved based on similarity match between context chunks and answer **or** question depending on the curriculum learning scheme. We tune the maximum answer length amongst $\{6, 8, 12\}$ and the maximum question length is set to 30. Since two answers are provided for each question, we train on both sets of answers. During construction of the golden labels, first perform an n-gram search of the answer in the context. The largest n-gram match is allocated indices belonging to the context (i.e., $[1,\ell_c]$). For the remainder words, stopwords are automatically allocated indices in the global vocabulary and non-stopwords are assigned context indices. If an answer word is not found, it is ignored. To construct the global vocabulary for the pointer generator decoder and avoid story-specific words, we use words that appear in at least 10 stories.

Evaluation During evaluation, we (1) remove the full stop at the end of answers and (2) lowercase both answers. We use the BLEU, Rouge and METEOR scorers provided at `https://github.com/tylin/coco-caption`.

Baselines As baselines, we compare the proposed model with reported results in (Kočiskỳ et al., 2018).. Additionally, we include several baselines which we implement by ourselves. This is in the spirit of providing better (and fairer) com-

[4] In early experiments, we found that $5\% - 10\%$ works best.

[5] In our early experiments, we also masked entities following the original work (Kočiskỳ et al., 2018), however, we did not observe obvious difference in performance. This is probably because we do not update word embeddings during training.

[6] `https://www.nltk.org/`

		Dev Set				Test Set			
Model	ℓ	BLEU-1	BLEU-4	Meteor	Rouge	BLEU-1	BLEU-4	Meteor	Rouge
IR (BLEU)	-	6.73	0.30	3.58	6.73	6.52	0.34	3.35	6.45
IR (ROUGE)	-	5.78	0.25	3.71	6.36	5.69	0.32	3.64	6.26
IR (Cosine)	-	6.40	0.28	3.54	6.50	6.33	0.29	3.28	6.43
BiDAF	-	5.82	0.22	3.84	6.33	5.68	0.25	3.72	6.22
ASR	200	16.95	1.26	3.84	1.12	16.08	1.08	3.56	11.94
ASR	400	18.54	0.00	4.2	13.5	17.76	1.10	4.01	12.83
ASR	1K	18.91	1.37	4.48	14.47	18.36	1.64	4.24	13.4
ASR	2K	20.00	2.23	4.45	14.47	19.09	1.81	4.29	14.03
ASR	4K	19.79	1.79	4.60	14.86	19.06	2.11	4.37	14.02
ASR (Ours)	4K	12.03	1.06	3.10	8.87	11.26	0.65	2.66	8.68
R^3	-	16.40	0.50	3.52	11.40	15.70	0.49	3.47	11.90
RNET-PG	4K	17.74	0.00	3.95	14.56	16.89	0.00	3.84	14.35
RNET-CPG	4K	19.71	2.05	4.91	15.05	19.27	1.45	4.87	15.50
IAL-CPG	4K	**23.31**	**2.70**	**5.68**	**17.33**	**22.92**	**2.47**	**5.59**	**17.67**
Rel. Gain	-	+31%	+51%	+23%	+17%	+20%	+17%	+28%	+26%

Table 1: Results on NarrativeQA reading comprehension dataset (Full story setting). Results are reported from (Kočiskỳ et al., 2018) .The numbers besides the model name denote the total context size. Rel. Gain reports the relative improvement of our model and the best baseline reported in (Kočiskỳ et al., 2018) on a specific context size setting.

parisons. The compared baselines are listed below:

- **Attention Sum Reader (ASR)** (Kadlec et al., 2016) is a simple baseline for reading comprehension. Aside from our the results on (Kočiskỳ et al., 2018), we report our own implementation of the ASR model. Our implementation follows (Kočiskỳ et al., 2018) closely.

- **Reinforced Reader Ranker (R^3)** (Wang et al., 2018) is a state-of-the-art model for open domain question answering, utilizing reinforcement learning to select relevant passages to train the reading comprehension model. Our objective is to get a sense of how well do open-domain models work on understanding narratives.

- **RNET + PG / CPG** (Wang et al., 2017b) is a strong, competitive model for paragraph level reading comprehension. We replace the span[7] prediction layer in RNET with a pointer generator (PG) model with the exact setup as our model. We also investigate equipping RNET + PG with our curriculum learning mechanism (curriculum pointer generator).

3.2 Experimental Results

Table 1 reports the results of our approach on the NarrativeQA benchmark. Our approach achieves state-of-the-art results as compared to prior work (Kočiskỳ et al., 2018). When compared to the best ASR model in (Kočiskỳ et al., 2018), the relative improvement across all metrics are generally high, ranging from $+17\%$ to 51%. The absolute improvements range from approximately $+1\%$ to $+3\%$.

Pertaining to the models benchmarked by us, we found that our re-implementation of ASR (Ours) leaves a lot to be desired. Consequently, our proposed IAL-CPG model almost doubles the score on all metrics compared to ASR (Ours). The R^3 model, which was proposed primarily for open-domain question answering does better than ASR (Ours) but still fall shorts. Our RNET-PG model performs slightly better than R^3 but fails to get a score on BLEU-4. Finally, RNET-CPG matches the state-of-the-art performance of (Kočiskỳ et al., 2018). However, we note that there might be distinct implementation differences[8] with the primary retrieval mechanism

[7] The performance of the RNET + span predictor is similar to the BiDAF model reported in (Kočiskỳ et al., 2018).

[8] This is made clear from how our ASR model performs

and environment/preprocessing setup. A good fair comparison to observe the effect of our curricum reading is the improvement between RNET-PG and RNET-CPG.

3.3 Ablation Study

In this section, we provide an extensive ablation study on all the major components and features of our proposed model. Table 2 reports results of our ablation study.

Attention ablation In ablations (1-3), we investigate the effectiveness of the self-attention layer. In (1), we remove the entire IAL layer, piping the context-query layer directly to the subsequent layer. In (2), we replace block-based self-attention with the regular self-attention. Note that the batch size is kept extremely small (e.g., 2), to cope with the memory requirements. In (3), we remove the multiplicative and subtractive features in the IAL layer. Results show that replacing the block-based self-attention with regular self-attention hurts performance the most. However, this may be due to the requirement of reducing the batch size significantly. Removing the IAL layer only sees a considerable drop while removing the enhancement also reduces performance considerably.

Curriculum ablation In ablations (4-8), we investigate various settings pertaining to curriculum learning. In (4), we remove the pointer generator (PG) completely. Consequently, there is also no curriculum reading in this setting. Performance drops significantly in this setting and demonstrates that the pointer generator is completely essential to good performance. In (5-6), we remove one component from our curriculum reading mechanism. Results show that the answerabiity heuristic is more important than the understandability heuristic. In (7-8), we focus on non curriculum approaches training on the easy or hard set **only**. It is surprising that training on the hard set alone gives considerablely decent performance which is comparable to the easy set. However, varying them in a curriculum setting has significant benefits.

RL ablation In ablation (9), we investigated techniques that pass the BLEU-score back as a reward for the model and train the model jointly using Reinforcement learning. We follow the setting of (Paulus et al., 2017), using the mixed training objective and setting λ to 0.05. We investigated using BLEU-1,BLEU-4 and Rouge-L (and combinations of these) as a reward for our model along with varying λ rates. Results in Table 2 reports the best result we obtained. We found that while RL does not significantly harm the performance of the model, there seem to be no significant benefit in using RL for generating answers, as opposed to other sequence transduction problems (Bahdanau et al., 2016; Paulus et al., 2017).

Understandability ablation From ablations (10-16), we study the effect of understandability and alternating paragraph sizes. We find that generally starting from a smaller paragraph and moving upwards performs better and moving the reverse direction may have adverse effects on performance. This is made evident by ablations (10-11). We also note that a curriculum approach beats a static approach often.

3.4 Qualitative Error Analysis

Table 3 provides some examples of the output of our best model. First, we discuss some unfortunate problems with the evaluation in generation based QA. In examples (1), the model predicts a semantically correct answer but gets no credit due to a different form. In (2), no credit is given for word-level evaluation. In (3), the annotators provide a more general answer and therefore, a highly specific answer (e.g., moscow) do not get any credit.

Second, we observe that our model is occasionally able to get the correct (exact match) answer. This is shown in example (4) and (7). However, there are frequent inability to generate phrases that make sense, even though it seems like the model is trudging along the right direction (e.g., "to wants to be a love of john" versus "because he wants her to have the baby" and "in the york school" versus "east harlem in new york"). In (9), we also note a partially correct anwer, even though it fails to realize that the question is about a male and generates "she is a naval".

4 Related Work

The existing work on open domain QA (Chen et al., 2017) has distinct similarities with our problem, largely owing to the overwhelming large corpus that a machine reader has to reason over. In recent years, a multitude of techniques have been developed. (Wang et al., 2018) proposed reinforce-

much worse than (Kočiskỳ et al., 2018). We spend a good amount of time trying to reproduce the results of ASR on the original paper.

Ablation	BLEU-1	BLEU-4	Meteor	Rouge
Original Full Setting	**23.31**	**2.70**	**5.68**	**17.33**
(1) Remove IAL layer	18.93	1.94	4.52	14.51
(2) Replace regular Self-Attention	19.61	0.96	4.38	15.24
(3) Remove Enhancement	20.25	1.76	4.92	15.14
(4) Remove PG + CR	15.30	0.91	3.85	11.36
(5) Remove CR (understandability)	20.13	2.30	4.94	16.96
(6) Remove CR (answerability)	20.13	1.82	4.92	15.77
(7) Train Easy Only	20.75	1.52	4.65	15.42
(8) Train Hard Only	19.18	1.49	4.60	14.19
(9) Add RL	21.85	2.70	5.31	16.73
(10) 50 → 100 → 200	23.31	2.70	5.68	17.33
(11) 50 → 100 → 200 → 500	21.07	2.86	5.33	16.78
(12) 100 → 200 → 500 → 50	20.18	2.60	5.50	18.14
(13) 500 → 50 → 100 → 200	20.95	2.51	5.41	17.05
(14) 500 → 200 → 100 → 50	17.13	2.38	4.60	15.56
(15) 50 (static)	20.91	2.57	5.35	18.78
(16) 500 (static)	19.36	2.45	4.94	16.00

Table 2: Ablation results on NarrativeQA development set. (1-3) are architectural ablations. (4-8) are curriculum reading based ablations. (9) investigates RL-based generation. (10-16) explores the understandability/paragraph size heuristic. Note that (10) was the optimal scheme reported in the original setting. Moreover, more permutations were tested but only representative example are reported due to lack of space.

Question	Model Answer	Ground Truth
(1) how many phases did the court compliment competition have?	two	2
(2) who suffers from a crack addiction?	dick	dicky
(3) where did john and sophia go to from the airport?	moscow	russia
(4) what country did nadia's cousin and friend visit her from?	russia	russia
(5) why is nadia kidnapped by alexei?	to wants be a love of john	because he now wants her to have the baby
(6) who does mary marry?	charles who is her	charles
(7) what instrument does roberta guaspari play?	violin	violin
(8) where is the school located where roberta takes a position as a substitute violin teacher?	in the york school	east harlem in new york city
(9) what is the profession of roberta's husband?	she is a naval	he is in the us navy

Table 3: Qualitative analysis on NarrativeQA development set.

ment learning to select passages using the reader as the reward. (Min et al., 2018) proposed ranking the minimal context required to answer the question. (Clark and Gardner, 2017) proposed shared norm method for predicting spans in the multi-paragraph reading comprehension setting. (Lin et al., 2018) proposed ranking and de-noising techniques. (Wang et al., 2017a) proposed evidence aggregation based answer re-ranking. Most techniques focused on constructing a conducive and less noisy context for the neural reader. Our work provides the first evidence of diverse sampling for training neural reading comprehension models.

Our work draws inspiration from curriculum learning (CL) (Bengio et al., 2009). One key difficulty in CL is to determine which samples are easy or hard. Self-paced learning (Jiang et al., 2015) is a recently popular form of curriculum learning that treats this issue as an optimization problem. To this end, (Sachan and Xing, 2016) applies self-paced learning for neural question answering. Automatic curriculum learning (Graves et al., 2017), similarly, extracts signals from the learning process to infer progress.

State-of-the-art neural question answering models are mainly based on cross-sentence attention (Seo et al., 2016; Wang and Jiang, 2016b; Xiong et al., 2016; Tay et al., 2018c). Self-attention (Vaswani et al., 2017; Wang et al., 2017b) has also been popular for reading comprehension (Wang et al., 2018; Clark and Gardner, 2017). However, its memory complexity makes it a challenge for reading long context. Notably, the truncated/summary setting of the NarrativeQA benchmark have been attempted recently (Tay et al., 2018c,b; Hu et al., 2018; Tay et al., 2018a). However, this summary setting bypasses the difficulties of long context reading comprehension, reverting to the more familiar RC setup.

While most of the prior work in this area has mainly focused on span prediction models (Wang and Jiang, 2016b) and/or multiple choice QA models (Wang and Jiang, 2016a), there have been recent interest in generation based QA (Tan et al., 2017). S-NET (Tan et al., 2017) proposed a two-stage retrieve then generate framework.

Flexible neural mechanisms that learn to point and/or generate have been also popular across many NLP tasks. Our model incorporates Pointer-Generator networks (See et al., 2017) which learns to copy or generate new words within the context of neural summarization. Prior to Pointer Generators, CopyNet (Gu et al., 2016) incorporates a copy mechanism for sequence to sequence learning. Pointer generators have also been recently adopted for learning a universal multi-task architecture for NLP (McCann et al., 2018).

5 Conclusion

We proposed curriculum learning based Pointer-generator networks for reading long narratives. Our proposed IAL-CPG model achieves state-of-the-art performance on the challenging NarrativeQA benchmark. We show that sub-sampling diverse views of a story and training them with a curriculum scheme is potentially more effective than techniques designed for open-domain question answering. We conduct extensive ablation studies and qualitative analysis, shedding light on the task at hand.

6 Acknowledgements

The authors would like to thank the anonymous reviewers of ACL 2019 for their comments and time to review our paper.

References

Dzmitry Bahdanau, Philemon Brakel, Kelvin Xu, Anirudh Goyal, Ryan Lowe, Joelle Pineau, Aaron Courville, and Yoshua Bengio. 2016. An actor-critic algorithm for sequence prediction. *arXiv preprint arXiv:1607.07086*.

Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. 2009. Curriculum learning. In *Proceedings of the 26th annual international conference on machine learning*, pages 41–48. ACM.

Danqi Chen, Adam Fisch, Jason Weston, and Antoine Bordes. 2017. Reading wikipedia to answer open-domain questions. *arXiv preprint arXiv:1704.00051*.

Christopher Clark and Matt Gardner. 2017. Simple and effective multi-paragraph reading comprehension. *arXiv preprint arXiv:1710.10723*.

Alex Graves, Marc G Bellemare, Jacob Menick, Remi Munos, and Koray Kavukcuoglu. 2017. Automated curriculum learning for neural networks. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 1311–1320. JMLR. org.

Jiatao Gu, Zhengdong Lu, Hang Li, and Victor OK Li. 2016. Incorporating copying mechanism in sequence-to-sequence learning. *arXiv preprint arXiv:1603.06393*.

Karl Moritz Hermann, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching machines to read and comprehend. In *Advances in Neural Information Processing Systems*, pages 1693–1701.

Minghao Hu, Yuxing Peng, Furu Wei, Zhen Huang, Dongsheng Li, Nan Yang, and Ming Zhou. 2018. Attention-guided answer distillation for machine reading comprehension. *arXiv preprint arXiv:1808.07644*.

Lu Jiang, Deyu Meng, Qian Zhao, Shiguang Shan, and Alexander G Hauptmann. 2015. Self-paced curriculum learning. In *Twenty-Ninth AAAI Conference on Artificial Intelligence*.

Rudolf Kadlec, Martin Schmid, Ondrej Bajgar, and Jan Kleindienst. 2016. Text understanding with the attention sum reader network. *arXiv preprint arXiv:1603.01547*.

Tomáš Kočiskỳ, Jonathan Schwarz, Phil Blunsom, Chris Dyer, Karl Moritz Hermann, Gáabor Melis, and Edward Grefenstette. 2018. The narrativeqa reading comprehension challenge. *Transactions of the Association of Computational Linguistics*, 6:317–328.

Yankai Lin, Haozhe Ji, Zhiyuan Liu, and Maosong Sun. 2018. Denoising distantly supervised open-domain question answering. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 1736–1745.

Bryan McCann, Nitish Shirish Keskar, Caiming Xiong, and Richard Socher. 2018. The natural language decathlon: Multitask learning as question answering. *arXiv preprint arXiv:1806.08730*.

Sewon Min, Victor Zhong, Richard Socher, and Caiming Xiong. 2018. Efficient and robust question answering from minimal context over documents. *arXiv preprint arXiv:1805.08092*.

Romain Paulus, Caiming Xiong, and Richard Socher. 2017. A deep reinforced model for abstractive summarization. *arXiv preprint arXiv:1705.04304*.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. Squad: 100,000+ questions for machine comprehension of text. *arXiv preprint arXiv:1606.05250*.

Mrinmaya Sachan and Eric Xing. 2016. Easy questions first? a case study on curriculum learning for question answering. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 453–463.

Abigail See, Peter J Liu, and Christopher D Manning. 2017. Get to the point: Summarization with pointer-generator networks. *arXiv preprint arXiv:1704.04368*.

Minjoon Seo, Aniruddha Kembhavi, Ali Farhadi, and Hannaneh Hajishirzi. 2016. Bidirectional attention flow for machine comprehension. *arXiv preprint arXiv:1611.01603*.

Chuanqi Tan, Furu Wei, Nan Yang, Bowen Du, Weifeng Lv, and Ming Zhou. 2017. S-net: From answer extraction to answer generation for machine reading comprehension. *arXiv preprint arXiv:1706.04815*.

Yi Tay, Anh Tuan Luu, and Siu Cheung Hui. 2018a. Multi-granular sequence encoding via dilated compositional units for reading comprehension. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2141–2151, Brussels, Belgium. Association for Computational Linguistics.

Yi Tay, Anh Tuan Luu, and Siu Cheung Hui. 2018b. Recurrently controlled recurrent networks. In *Advances in Neural Information Processing Systems*, pages 4731–4743.

Yi Tay, Anh Tuan Luu, Siu Cheung Hui, and Jian Su. 2018c. Densely connected attention propagation for reading comprehension. In *Advances in Neural Information Processing Systems*, pages 4906–4917.

Josh Tobin, Rachel Fong, Alex Ray, Jonas Schneider, Wojciech Zaremba, and Pieter Abbeel. 2017. Domain randomization for transferring deep neural networks from simulation to the real world. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 23–30. IEEE.

Adam Trischler, Tong Wang, Xingdi Yuan, Justin Harris, Alessandro Sordoni, Philip Bachman, and Kaheer Suleman. 2016. Newsqa: A machine comprehension dataset. *arXiv preprint arXiv:1611.09830*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, pages 5998–6008.

Shuohang Wang and Jing Jiang. 2016a. A compare-aggregate model for matching text sequences. *arXiv preprint arXiv:1611.01747*.

Shuohang Wang and Jing Jiang. 2016b. Machine comprehension using match-lstm and answer pointer. *arXiv preprint arXiv:1608.07905*.

Shuohang Wang, Mo Yu, Xiaoxiao Guo, Zhiguo Wang, Tim Klinger, Wei Zhang, Shiyu Chang, Gerry Tesauro, Bowen Zhou, and Jing Jiang. 2018. R 3: Reinforced ranker-reader for open-domain question answering. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

Shuohang Wang, Mo Yu, Jing Jiang, Wei Zhang, Xiaoxiao Guo, Shiyu Chang, Zhiguo Wang, Tim Klinger, Gerald Tesauro, and Murray Campbell. 2017a. Evidence aggregation for answer re-ranking in open-domain question answering. *arXiv preprint arXiv:1711.05116*.

Wenhui Wang, Nan Yang, Furu Wei, Baobao Chang, and Ming Zhou. 2017b. Gated self-matching networks for reading comprehension and question answering. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 189–198.

Caiming Xiong, Victor Zhong, and Richard Socher. 2016. Dynamic coattention networks for question answering. *arXiv preprint arXiv:1611.01604*.

Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhudinov, Rich Zemel, and Yoshua Bengio. 2015. Show, attend and tell: Neural image caption generation with visual attention. In *International conference on machine learning*, pages 2048–2057.

Matthew D Zeiler. 2012. Adadelta: an adaptive learning rate method. *arXiv preprint arXiv:1212.5701*.