Advances in Artificial Intelligence and Machine Learning; Research 3 (3) 1325-1339 Received 01-06-2023; Accepted 12-08-2023; Published 19-08-2023

Sliding-Bert: Striding Towards Conversational Machine Comprehension in Long Contex

Longxiang Zhang lzhang8794@monroecollege.edu
Monroe College, 434 Main St,
New Rochelle, NY 10801
USA

Wenping Wang wenpingw@alumni.cmu.edu
Carnegie Mellon University,
5555 Forbes Ave
USA

Keyi Yu chenlia2@alumni.cmu.edu
Google Inc, 1600 Amphitheatre Parkway
USA

Jingxian Huang jingxianhuang96@gmail.com
Meta Platforms, 1 Hacker Way
USA

Qi Lyu lyuqi1@msu.edu
Michigan State University,
426 Auditorium Road East Lansing,
MI 48824
USA

Haoru Xue hxue@ucsd.edu
University of California,
San Diego, 9500 Gilman Drive La Jolla,
CA 92093
USA

Congrui Hetang congruihetang@gmail.com
Google Inc, 1600 Amphitheatre Parkway
USA

Corresponding Author: Wenping Wang

Abstract

Pre-trained contextual embeddings like BERT have shown substantial improvement across a wide range of natural language processing tasks. We proposed Sliding-BERT, which incorporates BERT with state-of-the-art conversational machine comprehension (MC) model,

Citation: Longxiang Zhang, et al. Sliding-BERT: Striding Towards Conversational Machine Comprehension in Long Context. Advances in Artificial Intelligence and Machine Learning. 2023;3(3):78.

FlowQA, and supersedes its standing performance on the QuAC challenge. We designed a striding filter to overcome the sequence length limit of BERT model in the long conversation context. We also applied various aggregation methods to handle the incompatible tokenization between BERT and FlowQA models. Given the long conversation context, we used gradient accumulation to simulate batched training scenarios without extra memory cost during training. We also found that pretraining our Sliding-BERT on CoQA dataset helps improve its performance on QuAC dataset. Detailed analysis of the model performance considering the types of questions, lengths of questions and other metrics of QA datasets indicates that our Sliding-BERT exceeds FlowQA model in terms of F1, HEQ-Q, and HEQ-D scores by a significant margin.

1. INTRODUCTION

The creation of conversational, contextual question-answering (QA) datasets and design of machine comprehension (MC) systems that can handle long contextual information and flow of reasoning in question-answering dialogs have long been exciting areas of research in the NLP community. Efforts in the former task resulted in many conversational QA datasets such as SQuAD 1.0 [1], SQuAD 2.0 [2], CoQA [3], and more recently QuAC [4]; research in the latter yields strong MC models such as BiDAF [5], and FlowQA [6]. QuAC (**Qu**estion **A**nswering in **C**ontext, hereafter referred to as QuAC) offers several new challenges over traditional QA datasets in that it provides a student-teacher dialog environment, incremental questions for which answers depend on conversation history, and information asymmetry between student and teacher. Traditional MC models that have strong performance on SQuAD and CoQA tend to have much weaker performance on QuAC dataset, falling behind human performance by a large margin. It is argued [6], that QuAC dataset requires capturing not only contextual information, but also the flow of historical dialog for an MC system to achieve decent performance; and FlowQA [6], is one of the first models that capture this idea of dialog and reasoning "flow" in the design of conversational MC system.

The FlowQA model consists of two main components: a base neural network for single-turn MC and a Flow mechanism that passes down the encoded history of the entire conversation, along with the complete context, through every question within a single dialog. The encoded representation of entire conversation history can potentially capture not only related information in the context, but also the hints and reasoning required to answer previous questions, to help the system obtain answer to the next question in a sequential manner. We believe this mechanism is a key component to achieve state-of-the-art result on QuAC dataset, but we consider the use of word embeddings in existing FlowQA to be less than satisfactory. GloVe [7], CoVe [8], and ELMo [9], word embeddings were employed in existing FlowQA, model. Although all of them capture contextual information in their embedding vectors, none of them actively incorporates positional information of words appearing in a context, which should be an important source of information in answering sequential questions as set up in QuAC dataset. We therefore seek to replace the ELMo embeddings with BERT [10], embeddings, which subsume the positional information. We also believe that an MC system designed for QuAC dataset can benefit from pretraining on other datasets, e.g. CoQA, where reasoning and historic context play an important role.

In this study, we present detailed results & analysis from our re-implementation of FlowQA on QuAC challenge, and our improvement and modification to the model, as briefly introduced before, that beat its standing performance. In section 2, we briefly summarize existing literature on QuAC dataset and conversational MC systems; basic statistics on QuAC dataset are presented in section 3; and we include a detailed explanation of FlowQA model and our improved model architecture using pretrained BERT embeddings and effective data augmentation in section 4; The main experimental results based on the FlowQA model and our improved model are included in section 5; in section 7, we conduct detailed error analysis and discussion on all experiment results. We conclude our work in the final section with comments on possible future work.

2. RELATED WORK

Earlier QA datasets, such as NewsQA [11], HotpotQA [12], and SQuAD [1], are largely comprised of a corpus of contextual information (passages) with single question-answer pairs, where reference answers are usually a span of contiguous text within the provided context. These datasets offer a major challenge to the design of machine comprehension systems in that they require the system to have the capacity to incorporate contextual information in answering questions, as well as the ability to handle lexical and syntactical variations and ambiguity. In recent years, there has been a strong trend in the creation of conversational QA datasets, like CoQA, SQuAD 2.0, and QuAC. These datasets usually offer much richer contextual information, multi-turn questions and answers, and have included unanswerable questions (as an example, SQuAD 2.0 has over 50,000 unanswerable questions). Depending on how the dataset was created, different conversational QA datasets emphasize on different aspects of conversational question-answering. [13] offered a comprehensive comparison between CoQA, SQuAD 2.0 and QuAC. The paper discovered that SQuAD 2.0 has a higher variability in the unanswerable questions because it's designed to contain difficult questions; CoQA emphasizes on coherent understanding of long context in that it "drills down for details significantly more frequently and cover more than 60% of sentences in the context material"; QuAC, on the other hand, offers much more frequent topic shifts as the QA dialog based on single context progresses. All these features raise new, complex challenges to designing MC systems that must remember not only the context, but also historical reasoning used to answer previous questions in a conversation or dialog-like environment.

It is therefore crucial for any good performing QA or MC systems to grasp information (contextual or logical) flow in their model architecture [14–16]. The **BiDAF** [5], model is one of the first models in this direction and beat the baseline results on the SQuAD dataset. The concept of "bidirectional attention flow" actively seeks to learn an integrated latent representation of context to assist in answering a series of questions. It has since then become a strong baseline on most other conversational QA datasets and has inspired many other systems that attempt to integrate context and historical reasoning. For example, one of the baselines proposed on the QuAC dataset uses contextualized embeddings like ELMo and augments self-attention to the bi-directional attention layer [17], of the BiDAF model. Another variation, **BiDAF++ w/ k-ctx**, uses dialog history to improve the representations of the context and the query. The question, answer and the dialog turn number from the previous k turns are used to augment to the embedddings to improve the information passed down the layers. **FusionNet** [18], improves upon these models by using the history of word, that incorporates all the relevant information for each word, right from word embeddings to the hidden representations generated from the RNN layers. The higher-level information from the question and context are fused together through a fully-aware attention mechanism on the history of the word. To incorporate information from distant words in the context, **FusionNet** also uses a fully-aware self-attention on the words in the context. **FlowQA**, as explained in detail in section 4, utilizes reasoning from prior context in addition to fully-aware self-attention to answer the current question.

Using pre-trained word embeddings and other syntactic features (such as part-of-speech tags and named entity recognition tags) helps jump start training of MC systems on conversational QA datasets by providing rich representation of words and characters as system input. For example, BERT, the state-of-art transformer model [19], has been employed in many recent MC systems to improve the state-of-the-art results on various QA datasets. The contextualized word embeddings, provided by unsupervised pre-training of BERT, subsume word-level, sequence-level and positional information of input text. BERT employs Masked Language Modeling training objective to learn word embeddings that incorporate contextual information in both directions of the center word across multiple transformer layers. This presents a natural advantage over ELMo model used in the BiDAF++ model, which only learns a shallow bidirectional contextual embedding for each word. The paper [10], also demonstrated that pre-trained embeddings from BERT can be directly used for downstream language modeling tasks, making BERT embeddings a strong candidate for the embedding layer in a BiDAF-like network architecture.

3. DATASET

QuAC dataset contains 14K information-seeking QA dialogs. The dataset shares common features like multi-turn question answering, unanswerable questions, with other well-established QA datasets, e.g. SQuad 1.0, SQuad 2.0 and CoQA. However, QuAC dataset emphasizes the use of context by introducing an imbalanced information setting (only teacher can access the hidden text, student is only provided section title and a short summary at the start of each dialog) and allows "dialog acts" from teachers (listed below) to encourage continuation of dialogs.

- continuation (*follow up*, *maybe follow up*, or *don't follow up*)
- affirmation (*yes*, *no*, or *no answer*)
- answerability (*answerable* or *no answer*)

Two major evaluation metrics are used on QuAC dataset: (1) word-level F1 score that measures the overlap of words between prediction and reference answers; (2) human equivalence score (HEQ) that measures percentage of examples for which model F1 exceeds or matches human F1 score; specifically, two variants of HEQ are used (HEQ-Q, question-level HEQ and HEQ-D, dialog-level HEQ).

4. METHODS

4.1 FlowQA

FlowQA was the state-of-the-art model for 5 months and remained one of the top performing models for QuAC challenge till this day. At a high level, FlowQA shares similar components with conventional single-turn MC (machine comprehension) models: (1) question encoding module, (2) context encoding module, (3) reasoning layers and (4) answer prediction layer. What makes FlowQA stand out as a strong dialog MC model is the introduction of the concept of **Flow**: "**a sequence of latent representations based on context tokens**" [6]. In other words, FlowQA model carries over contextual information not only from earlier question-answer pairs, but also from earlier latent representations of context, which are believed to encapsulate the reasoning process of the system at arriving previous answers.

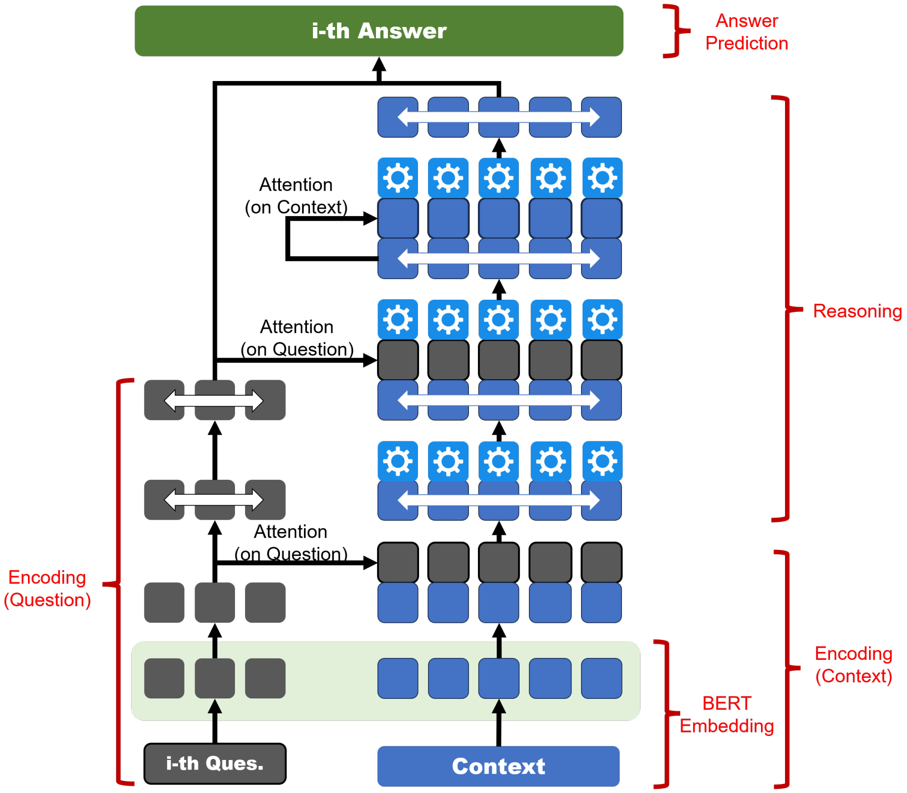

Figure 1: Architecture of FlowQA with BERT embeddings (modified from [6])

The detailed architecture of FlowQA is shown in FIGURE 1. From bottom to top, the major components of the model include:

Question/Context Encoding: For initial encoding of questions and context, FlowQA utilizes pre-trained embeddings both at word level (GloVe and CoVe) and at character level (ELMo). POS and NER tags for each word are also used, and the final concatenated embedding vectors for context and questions are fed as input into separate encoding layers (BiLSTM). The contextual information about each question is encoded into the context by attending on the words in the question. The initial embedding vectors for the question are passed through multiple layers of BiLSTM to obtain a deeper contextual representation.

Integration-Flow Reasoning: The concept of **Flow** in FlowQA is implemented as highly-parallelizable integration-flow (IF) layers: Integration layer integrates contextual information for each question (Eq.(1)),

$$\mathbf{\hat{C}_i^h} = \hat{c}_{i,1}^h, \ldots, \hat{c}_{i,m}^h = BiLSTM(\mathbf{C_i^h}) \tag{1}$$

where $\mathbf{\hat{C}_i^h}$ is the latent representation of context sequence for ith question at hth IF layer. Flow layer passes the integrated contextual representation along the questions within the same dialogue,

$$\mathbf{F}_j^{h+1} = f_{1,j}^{h+1}, \ldots f_{t,j}^{h+1} = GRU(\hat{c}_{1,j}^h, \ldots, \hat{c}_{t,j}^h) \tag{2}$$

Note that in Eq.(2), context word (subscripted by j) is fixed and latent representation is passed down the questions through a forward-directional GRU. The output from IF layer is the concatenation of both outputs, as in Eq.(3),

$$\mathbf{C}_i^{h+1} = [\hat{c}_{i,1}^h; f_{i,1}^{h+1}], \ldots, [\hat{c}_{i,m}^h; f_{i,m}^{h+1}] \tag{3}$$

$\mathbf{\hat{C}_i^h}$ and $\mathbf{F}_j^{h+1}$ together represent the "reasoning" process of the system and is retained from one IF layer to the next in FlowQA model. Furthermore, both intra-attention on question, context and inter-attention between question and context are made use of in the FlowQA model (FIGURE 1).

Answer Prediction. The final representation for question and context are combined to predict the answer span for each question. Since QuAC contains unanswerable questions, answerability probabilities are also calculated.

4.2 Extending FlowQA

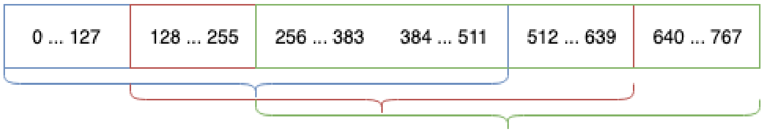

Figure 2: Example of sliding window used in BERT with a stride of 128

To improve the performance of the FlowQA model, we tried the following approaches:

BERT: From the ablation studies we conducted, we realized that ELMo embedding vectors played a crucial role in the performance of the model. Naturally, we replaced the ELMo embeddings with the more efficient BERT [10], embeddings and performed detailed experimental analysis. However, switching from ELMo to BERT embeddings was not straightforward for FlowQA model and the following techniques have been implemented:

- **Tokenization:** BERT uses byte pair encodings (BPE) to tokenize the data, where we get an embedding for each token obtained through BPE. For example, the word "radioactivity" is tokenized as “radio”, “act”, “ivity”. However, the CoVe and GloVe embeddings used in FlowQA work at word level. To get word level embeddings from BERT, we aggregate the embedding vectors of each token of the word. We experiment with two different aggregation methods, *i.e.*, mean pooling and max pooling, across the embeddings vectors of the sub-tokens of the word.

- **Sliding window:** BERT only supports a maximum sequence length of 512 but the context size in FlowQA can be longer than 2000 BPE tokens. In order to preserve information from the context, we use striding to get the token embeddings from BERT and take the mean of different embeddings obtained for each token. An example with stride 128 is shown in FIGURE 2.
- **Gradient accumulation:** Due to memory constraints, we could only use a batch size of 1 dialog with BERT. This resulted in a noisy update and was performing worse than the original FlowQA. To mitigate this, we accumulate the gradients for k steps and then do an update to the network weights, effectively simulating a batch size of k.

Data augmentation/pretraining: Owing to impressive results obtained by pretraining neural models on related datasets, we experimented with pretraining the model on the CoQA dataset [3]. In comparison to QuAC dataset, CoQA has similar format of question and answers in a conversational setting. In addition to the answer, the model also needs to return a rationale for the answer. This makes CoQA an ideal dataset for pretraining. We noticed a big improvement in the initial epochs when using a model pretrained on the CoQA dataset.

Positional embeddings (Pos Emb): One of the main strengths of BERT is the use of positional embeddings. Since it only supports positional embeddings for a maximum of 512 sequence length and we use striding to generate embeddings for sequences beyond that maximum length, the positional information is destroyed. To circumvent this problem, we introduce our own 15 dimensional learnable positional embeddings, which seemed to improve the performance by a good margin.

5. EXPERIMENTS

We performed two different sets of experiments: 1) incorporating BERT into FlowQA and 2) pretraining the model (with/without BERT) on CoQA. In all experiments, we continue fine-tuning the whole model on QuAC dataset before evaluating model performance on QuAC validation set. As described in the previous section, experiments involving BERT required careful tuning of the parameters, specifically the batch size and gradient accumulation steps. In this section, we present and compare results of all such models trained for about 15-20 epochs on V100 instances on GCP. Unless explicitly specified, the BERT model we used refers to the BERT Base Uncased model.

[1] The baseline model utilizes only BERT embeddings without POS tags, NER labels or CoVe embeddings as done in the original FlowQA model. All other models incorporate these additional embeddings.

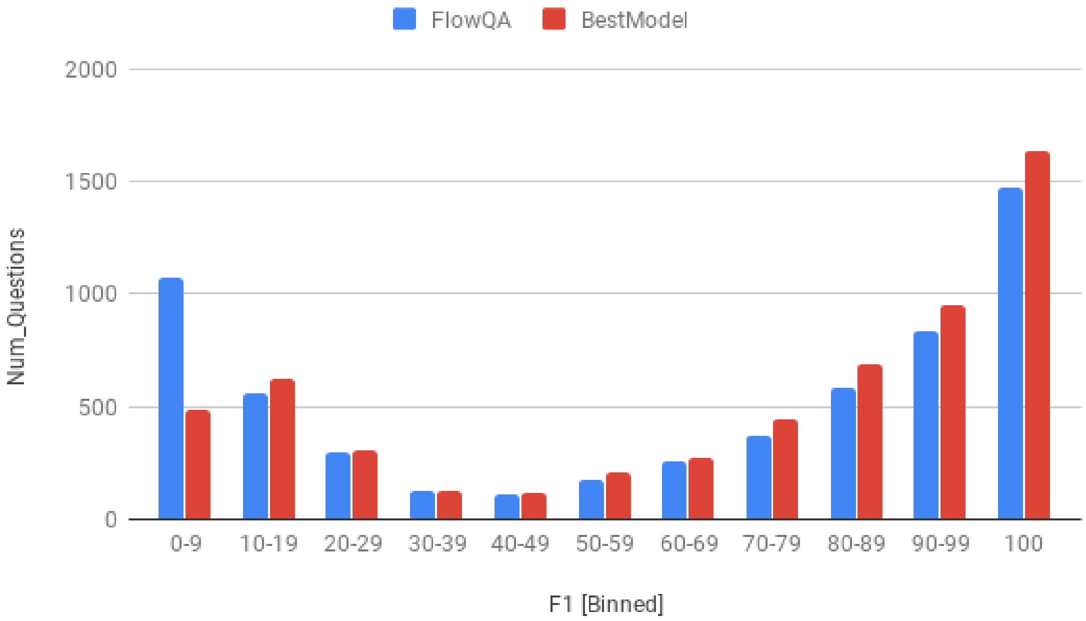

Figure 3: Number of questions over binned F1 scores (only considering answerable questions)

Table 1: F1 scores for different models that we experimented with. "BERT-K" means that the BERT embeddings are summed from the last K layers. "AG-K" stands for K-step aggregated gradient used in gradient accumulation (therefore, AG-1 means no gradient accumulation). "FT-K" stands for fine-tuning the top K layers of BERT along with FlowQA. "max" means taking the elementwise max of BPE embeddings; wherever max is not specified, mean embeddings are used. "CoQA PT" means the model is pretrained first on CoQA dataset. "PosEmb" in our best performing model stands for the 15 dimensional trainable positional embeddings.

Model	**F1**
FlowQA	64.6
FlowQA + CoQA PT	65.687
BERT-1 + AG-1 (baseline[1])	61.741
BERT-4 + AG-1	64.432
BERT-4 + AG-3	64.732
BERT-4 (large) + AG-3	65.197
BERT-1 + FT-12 + AG-1	62.7
BERT-1 + FT-12 + AG-3	62.728
BERT-4 + FT-4 + AG-5	65.465
BERT-4 + FT-4 + AG-5 + max	66.19
BERT-4 + FT-4 + AG-5 + max + PosEmb + CoQA PT (**our best**)	**66.925**

5.1 Main Results

TABLE 1 shows the main results of our experiments and the effect of varying settings of finetuning layers in BERT and number of steps used in gradient accumulation. We saw a big drop from the original FlowQA model in our baseline model, which doesn't include any part-of-speech (POS) tags, named entity recognition (NER) labels, or CoVe embeddings; this suggests these embeddings do encode additional syntactic and contextual information that complements not only the original Elmo embeddings but also the BERT embeddings. Using a larger number of steps for gradient accumulation helps improve model performance, but the improvement is minor compared to using more BERT layers for extracting the word embeddings (compare BERT-4 to BERT-1 models in TABLE 1). Pretraining on CoQA has helped greatly the model performance in F1 score, two out of the top 3 models in TABLE 1 are pretrained on this extra dataset, this suggests continued pretraining on similar data could benefit model performance even on already pretrained model (BERT).

To our surprise, we observed that using element-wise max instead of mean to get word level embeddings from BPE improves model performance; and using more layers of BERT for embedding extraction benefits the model more than fine-tuning the BERT model with more layers (compare FT-12 models with FT-4 models in TABLE 1), this also saves GPU memory consumption of the model training by 25% (from 16 GB to 12 GB). Our best performing model incorporates all these findings, including training additional 15-dimensional positional embeddings, and achieved an F1 score of 66.9, a 3.6% relative improvement over the original FlowQA model.

Table 2: HEQ-Q and HEQ-D for our best models with and without positional embeddings and CoQA pretraining ("PT") compared with FlowQA on the validation dataset

Model	**HEQ-Q**	**HEQ-D**
FlowQA	59.8	5.8
Our best model (No PosEmb + No PT)	62.54	7.0
Our best model	**63.32**	**8.5**

HEQ-Q and HEQ-D: We are able to improve considerably on the HEQ-Q and HEQ-D scores on the validation dataset as shown in TABLE 2. The 2nd row shows the scores for our model without pretraining and positional embeddings. It is obvious that positional embedding and pretraining helps the model performance only to a limited extent; the major advantage in our model performance may come from incorporating BERT model with FlowQA.

6. ERROR ANALYSIS

Since our best model is based upon FlowQA model, it would be instructive to analyze areas in which our model outperforms (underperforms) the original model. We study in details the dependence of our model performance (mainly F1 scores) on various aspects of the QuAC dataset. All the plots in this section are generated on the validation set of QuAC.

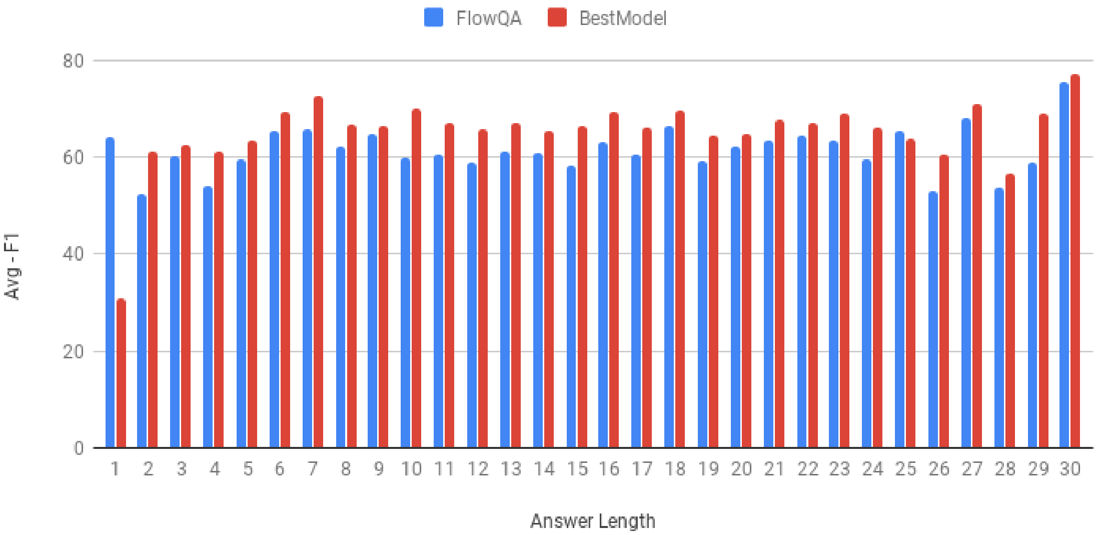

Figure 4: Average F1 score versus answer length(in words). This includes "CANNOTANSWER" as 1 word answers which is the main reason for the difference in performance when answer length is 1.

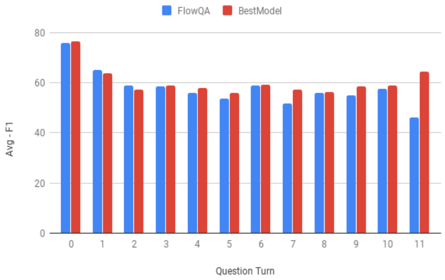

Figure 5: Average F1 score versus question turn (only considering answerable questions)

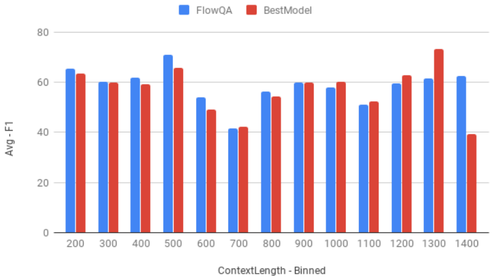

Figure 6: Average F1 score versus context length (only considering answerable questions)

6.1 Performance on Question Answerability

Answerability of questions is an added task in QuAC and unanswerable questions where the model first predicts whether a question can be answered based on the context, before finding the answer span. TABLE 3 shows the true positive, false positive, true negative and false negatives rates for classification of unanswerable questions for both FlowQA and our best model. As can be seen, while our model performs worse in correctly classifying unanswerable questions as unanswerable (true positive rate), it does much better at answering answerable questions (true negative rate) and also has improvements in the false positive rates (i.e. classifying answerable questions as unanswerable). We can see that the bias of the model has shifted more to classifying questions as answerable. Overall,

Table 3: Statistics for unanswerable questions for the base FlowQA model and our best model. Last row shows the number of mis-classified samples. Better results are bolded.

Category	FlowQA	BestModel
TP	**979 (13.3%)**	426 (5.8%)
FP	733 (9.9%)	**100 (1.4%)**
TN	5135 (69.8%)	**5768 (78.4%)**
FN	**507 (6.9%)**	1060 (14.4%)
FP+FN	1240 (16.9%)	**1160 (15.8%)**

any misclassification of such questions leads to an F1 score of 0 and since our model has slightly less misclassifications, we see an improvement in the overall F1 score.

6.2 Performance on Question Answering

Number of questions vs F1 score: FIGURE 3 shows a histogram of number of questions binned by F1 scores. This graph only looks at answerable questions and gives an idea of how good the spans found by our model are as compared to the FlowQA model. We see much a higher count of questions in higher F1 score range and we also see a huge decrease in number of questions getting 0-9 F1 score. This can be attributed to FlowQA 's tendency to mark answerable questions as unanswerable while our model finds some answers for such questions. It's a reasonable deduction that the modifications we made to FlowQA are pushing the questions out from the 0 F1 score category to the other categories, resulting in a higher overall F1 score.

F1 score vs Answer length: FIGURE 4 shows the comparison of average F1 scores for different answer lengths of our best model versus the best results that we got on the base FlowQA model. As we can see the score has consistently improved for answer lengths above 1. This is mainly due to the fact that answer lengths of 1 consists of unanswerable questions as well and as seen previously, the base FlowQA does a better job at such questions however, its higher error rate at marking answerable questions as unanswerable leads to worse performance on other longer answers. It is interesting to see that there is a significant difference even in short answers of length < 5 words.

F1 score vs Question turn: FIGURE 5 shows the average F1 score per question turn. We observe that our model outperforms the base FlowQA model for each question turn. This difference is especially stark at the last question turn. This may be due to the fact that we use explicit learnable positional embeddings.

F1 score vs context length: FIGURE 6 shows the average F1 score for different context lengths. The last bin has only 8 examples so a drop in performance is not significant to be considered a trend. For other longer contexts, we see that our model is comparable to or better than the base FlowQA .

Context:	...The short story collection, La Cosa e altri racconti (The Thing and Other Stories), was dedicated to Carmen Llera, his new companion (forty-five years his junior), whom he married in 1986....
Question:	**Whom did he marry?**
Ground Truth:	Carmen Llera,
Best Model:	Carmen Llera,
FlowQA:	CANNOT ANSWER
Context:	...Kaif has a close relationship with her family, and the lack of a father figure in her life has given her a sense of responsibility towards them. While Kaifś mother is Christian and her father is Muslim, Kaif was brought up to practise all faiths and says she is a ”firm believer in God”...
Question:	**Where is her father?**
Ground Truth:	CANNOT ANSWER
Best Model:	CANNOT ANSWER
FlowQA:	While Kaif’s mother is Christian and her father is Muslim,
Context:	...RoboCop, for Sound Effects Editing, and Total Recall, for Visual Effects, each won an Academy Special Achievement Award. Verhoeven followed those successes with the equally intense and provocative Basic Instinct (1992), an erotic thriller...
Question:	**What was another one of his films he made in the US?**
Ground Truth:	Basic Instinct
Best Model:	Basic Instinct (1992),
FlowQA:	RoboCop, for Sound Effects Editing, and Total Recall,
Context:	...Following the success of the album, Turner embarked on a UK arena tour, including a date at Londonś O2 arena. On 3 January 2014, Turner appeared on the BBCś Celebrity Mastermind answering questions on Iron Maiden, scoring 20 points in total (7 on his specialist subject), and coming first...
Question:	**Anything interesting you can tell me about him?**
Ground Truth:	On 3 January 2014, Turner appeared on the BBC’s Celebrity Mastermind answering questions on Iron Maiden, scoring 20 points in total (7 on his specialist subject), and coming first
Best Model:	On 3 January 2014, Turner appeared on the BBC’s Celebrity Mastermind answering questions on Iron Maiden,
FlowQA:	On 9 January, Frank uploaded a photo to his Instagram page showing the names of all 13 songs that would feature on his new album.

Table 4: Examples of our best model’s prediction compared against FlowQA

6.3 Qualitative Analysis

TABLE 4 shows some sample questions and context from QuAC dataset, where our model performs better than the base FlowQA model. The first example shows a question that FlowQA considers unanswerable while our model finds the correct answer, which falls into the category of true negatives where our model outperforms the FlowQA model.

The second example shows a type of tricky question that our best model outperforms FlowQA . The keyword in the question, ”her father”, does appear in the context, but there is no mention of where “he” is. FlowQA extracts a bogus answer spanning the keyword ”her father”, whereas our model correctly identifies this as an unanswerable question.

The third example asks about the movies following the ones mentioned in prior context. FlowQA fails to “understand” this and extracts the span with lots of movie names while our model correctly identifies the required movie.

The last example is the trickiest as the question is rather open ended. FlowQA answers with a fact about Frank but it is not quite interesting as the one that our model predicts, which also matches with the reference answers. Questions of this kind are largely subjective and there is usually a large variation in the provided 5 reference answers by human. Since ground truth answers in the QuAC dataset are selected from reference answers using majority voting, it’s not a surprise to have neural model confused when humans don’t agree. Therefore it’s actually surprising to see our model was able to capture the slight variation in all answers to such open-ended question and end up with predicting the ”more popular” one.

Overall, we hypothesize that contextual embeddings of BERT, fine-tuned to our dataset has better understanding of the context, thereby yielding better spans to most of the questions.

7. CONCLUSION

For Question and Answering in Context challenge, we incorporated BERT into the FlowQA model and observed a significant improvement in the results both in terms of F1 score and HEQ-Q, HEQ-D scores. During this process, we realized that the performance of the model is quite sensitive to the way we choose to aggregate the byte pair encodings’ embeddings. We experimented with two differnt strategies: 1) Max-Pooling and 2) Mean-Pooling when we aggregate embedding along each dimension. Empirical results showed that the max approach performed better than the mean approach. While we are unsure of why this is the case, it may be an interesting research problem to determine which aggregation method works best for a particular downstream task.

We also show improvement in model performance by using trainable positional embeddings, especially for cases where striding mechanism is required to obtain contextual embeddings for sequences beyond the maximum sequence length (512) of BERT. These are particularly crucial in QuAC dataset and more general conversational Question Answering and beyond [20, 21], since the position of answer frequently depends on the question turn [4]. This may have been one of the main reasons for the excellent performance of BERT for Question Answering tasks. As can be seen from TABLE 1, using positional embeddings improved the F1 score. We used positional embeddings of 15 dimensions based on the heuristic of O(log(max sequence length)). It might be an interesting problem to explore alternative strategies to encode the positional information, which could benefit other tasks as well [22, 23].

Pretraining on CoQA dataset, which is similar to the QuAC dataset has shown to improve results by a significant margin. Specifically, we observed a significant improvement in the F1 score in the first few epochs of fine-tuning, thereby illustrating that pretraining on CoQA helps the model to learn attention patterns that are adapatable to the QuAC dataset. In addition to the performance improvement, it can also speed up the convergence, as we observed empirically. CoQA [3] is a great conversational QA dataset to pretrain on, owing to its answer format that requires reasoning in addition to the answer spans.

The teacher-student setting in QuAC under which the student cannot see the whole context results in challenging unanswerable questions. These questions tend to be genuine and related to the context and thus are more difficult to be recognized as unanswerable. These pose an interesting challenge to the Question answering models that fail to truly understand the context.

Incorporating BERT into FlowQA helped in achieving a significant improvement in our results. We observed that fine-tuning the last 4 layers of BERT yielded better results than just using the pretrained emebeddings. We also noticed that the span of the answers predicted with BERT-FlowQA are better than the FlowQA model. We hypothesize that the contextual embeddings learned by BERT capture the meaning of the underlying context thereby resulting in better inferences required to answer complicated questions. We also believe that our purposed advancement could have better performance, if we can apply some current hardware machine learning optimization techniques [24, 25], etc.

References

[1] Rajpurkar P, Zhang J, Lopyrev K, Liang P. Squad: 100,000+ Questions for Machine Comprehension of Text. 2016. ArXiv preprint: https://arxiv.org/pdf/1606.05250.pdf

[2] Rajpurkar P, Jia R, Liang P. Know What You Don't Know: Unanswerable Questions for Squad. 2018. ArXiv preprint:https://arxiv.org/pdf/1806.03822.pdf

[3] Reddy S, Chen Danqi, Manning CD. Coqa: A conversational question answering challenge. Transactions of the Association for Computational Linguistics. 2018. ArXiv preprint: https://aclanthology.org/Q19-1016.pdf

[4] Choi E, He H, Iyyer M, Yatskar M. Wen-tau Yih, et al. Quac:Question Answering in Context. 2018. ArXiv preprint: https://arxiv.org/pdf/1808.07036.pdf

[5] Seo M, Kembhavi A, Farhadi A, Hajishirzi H. Bidirectional Attention Flow for Machine Comprehension. 2018. ArXiv preprint:https://arxiv.org/pdf/1611.01603.pdf

[6] Huang HY, Choi E, Yih WT. Flowqa: Grasping Flow in History for Conversational Machine Comprehension. 2019. ArXiv preprint: https://arxiv.org/pdf/1810.06683.pdf

[7] Pennington J, Socher R, Manning CD. Glove: Global Vectors for Word Representation. In: Empirical Methods in Natural Language Processing (EMNLP). 2014:1532-1543.

[8] McCann B, Bradbury J, Xiong C, Socher R. Learned in Translation: Contextualized Word Vectors. In Proceedings of the 31st International Conference on Neural Information Processing Systems (NIPS'17). Curran Associates Inc., Red Hook, NY, USA.2017:6297–6308.

[9] Peters ME, Neumann M, Iyyer M, Gardner M, Clark C, et al.. Deep Contextualized Word Representations.2018. ArXiv preprint: https://arxiv.org/pdf/1802.05365.pdf

[10] Devlin J, Chang M-W, Lee K, Toutanova K. Bert: Pretraining of Deep Bidirectional Transformers for Language Understanding. 2019. ArXiv preprint: https://arxiv.org/pdf/1810.04805.pdf

[11] Trischler A, Wang T, Yuan Xingdi, Harris J, Sordoni A, et al. Newsqa: A Machine Comprehension Dataset. In: Proceedings of the 2nd Workshop on Representation Learning for Nlp. Association for Computational Linguistics. 2017:191-200.

[12] Yang Z, Qi P, Zhang S, Bengio Y, Cohen WW, et al. Hotpotqa: A Dataset for Diverse, Explainable Multi-Hop Question Answering. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing. 2018.

[13] Yatskar M. A Qualitative Comparison of Coqa, Squad 2.0 and Quac. 2019. ArXiv preprint: https://arxiv.org/pdf/1809.10735.pdf

[14] Chen T, Wang X, Yue T, Bai X, Le CX, et al.. Enhancing Abstractive Summarization With Extracted Knowledge Graphs and Multisource Transformers. Appl Sci. 2023;13:7753.

[15] Yang Xuanyue. Linguistically Inspired Neural Coreference Resolution. Adv Artif Intell Mach Learn. 2023;3:66.

[16] Keyi Y, Wang Y, Zeng Sihan, Liang C, Bai X, et al. Inkgan: Generative Adversarial Networks for Ink-And-Wash Style Transfer of Photographs. 2023.

[17] Clark C, Gardner M. Simple and Effective Multi-Paragraph Reading Comprehension. 2017. ArXiv preprint: https://arxiv.org/pdf/1710.10723.pdf

[18] Huang HY, Zhu C, Shen Yelong, Chen W. Fusionnet: Fusing via fully aware attention with application to machine comprehension. 2018. ArXiv preprint: https://arxiv.org/pdf/1711.07341.pdf

[19] Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, et al. Attention Is All You Need. Adv Neural Inf Process Syst. 2017;30:5998-6008.

[20] Yue T, Wang Haohan. Deep learning for genomics: A concise overview. ArXiv preprint: https://arxiv.org/pdf/1802.00810.pdf

[21] Wenting Y, Liu X, Yue T, Wang W. A Sparse Graphstructured Lasso Mixed Model for Genetic Association With Confounding Correction. 2023. ArXiv preprint: https://arxiv.org/pdf/1711.04162.pdf

[22] Wang W, Guo Y, Chiyao S, Ding S, Liao G, et al. Integrity and Junkiness Failure Handling for Embedding-Based Retrieval: A Case Study in Social Network Search. 2023. ArXiv preprint: https://arxiv.org/pdf/2304.09287.pdf

[23] Zhang L, Negrinho R, Ghosh A, Jagannathan V, Hassanzadeh HR, et al. Leveraging Pretrained Models for Automatic Summarization of Doctor-Patient Conversations. 2021. ArXiv preprint: https://arxiv.org/pdf/2109.12174.pdf

[24] Zhou Y, Gupta U, Dai S, Zhao R, Srivastava N, et al. Rosetta: A Realistic High-Level Synthesis Benchmark Suite for Software Programmable Fpgas. In: Proceedings of the 2018 Acm/Sigda International Symposium on Field-Programmable Gate Arrays, Fpga'18. NY: Association for Computing Machinery. 2018:269-278.

[25] Zhou Y, Gupta U, Dai S, Zhao R, Srivastava N, et al. Rosetta: A Realistic High-Level Synthesis Benchmark Suite for Software-Programmable Fpgas. Int' Symposium on Field-Programmable Gate Arrays (FPGA). 1018:269-278.