CS-7650 Team 5 : Final Project Report
Improving Contextual Answer Generation

Shalini Chaudhuri
shalini.chaudhuri@gatech.edu

Sushmita Singh
sushmita.singh@gatech.edu

Rohit Mujumdar
rohitmujumdar@gatech.edu

1 Abstract

The goal of this project is to develop a model that can provide answers and give information to an agent that has no prior knowledge of a subject. By engaging with our model, the agent is able to extract details about the said topic through a series of questions and answers. The answering agent we developed keeps track of the context and uses that to answer questions of the agent from the scope of the document. Using deep learning architectures that capture flow of information and attention mechanisms, we have developed and enhanced a few models for our task. We enhanced FlowQA to predict not only answers but also other meta-information such as ”does this question require a follow-up?” and ”is this a yes/no question?”. To make improvements to the performance, we also added attention to FLowQA. In addition, we experimented with BiDAF using Bert+Glove embeddings to examine the changes embeddings can bring to the model. The code for our project is hosted at https://github.gatech.edu/schaudhuri34/ContextQA/and the video is hosted at https://bluejeans.com/s/qScWr. From our experiments our models have not been able to beat the human F1 score but perform better than the baseline models provided on the QuAC leaderboard. Adding attention and Bert embeddings helped us improve the performance of the enhanced FlowQA model. However, there is a significant gap between humans and the model's performance.

2 Introduction and Related Work

Information-seeking QA dialogue systems are an important application for natural language query systems (1), when users have limited or no access to an underlying corpus (2). In such systems, there is a Questioning-agent (Q-bot) that has access to the corpus that contains relevant and important information, and an Answering-agent (A-bot) which seeks to know more about a particular topic or entity. Unlike regular QA systems, contextual dialogue QA pairs are often meaningful only within the context of the dialog with questions often being abstract, open-ended or unanswerable. In our project, we aim to build an A-bot capable of participating in a human-like, information-seeking conversation (3) about a particular subject. In this setting, while the A-bot has access to documents related to the subject and uses them to answer the questions posed to it, the Q-bot is only introduced to a seed topic - the headline of a news article or the subject of a wiki-article.

While our initial goal was to generate both questions and answers, we changed it during the midterm report due to time constraints and the effort required to understand, set up and test each new architecture. Since then, our focus has been on training and improving the answering agent. Our primary motivation behind this goal is that the A-bot plays a very important role as it promotes knowledge distribution and a fruitful dialogue with the questioning agent. In a nutshell, our goal is to address the following research question: ”Using an answering agent in an information-seeking QA dialogue, how can we improve the state-of-the-art in contextual answering, especially by applying concepts like attention and deeply bidirectional encodings?”

Question Answering systems have gained significant popularity in recent years. Plenty of systems have achieved promising results with the advancement in neural networks for Natural language processing and new techniques like attention. Neural attention mechanism enables the system to focus on a targeted area within a context paragraph (for Machine Comprehension) or within an image (for Visual QA), that is most relevant to answering the question (4). A major contribution has also been the availability of datasets. Early datasets like MCTest (5) were too small for end-to-end training of neural models. More recently, datasets like SQuAD (6), QuAC (7) and CoQA (8), have made possible the research of deep neural networks on Machine comprehension. For our project, we chose the QuAC dataset as it incorporates within it the conversational nature of question-answering. It is designed to be context-crucial in answering, that is, the location of answer within the text is influenced by the number of questions asked previously. Moreover, apart from guessing the correct answer span, QuAC also requires the user to guess three Dialog acts - Yes/No (affirmation), Follow up (continuation) and answerability (answerable or no answer).

The baseline model (7) is an improvement on the Bi-directional Attention flow network model. The original model by Seo et al. (9) builds on the granular representation of context and the input questions. It includes character-level, word-level, contextual embeddings and uses bi-directional attention flow to obtain a query-aware context representation. This model was further enhanced in 2018 by Peters et al. (10) by adding self-attention. A major complication with these models is that they focus on a single-turn setting, much like Machine comprehension tasks. On the other hand, humans generally seek answers in a conversational fashion. Recently proposed FlowQA (Huang et al.) (11) addresses this by modelling dialog context in order to improve the performance for conversational QA.

3 Methods

1. **Data**

   We used the QuAC dataset (7) for our problem statement. It is diverse and is conditioned on a problem statement where the student has access to only the title while the teacher has access to the relevant document. We chose to use QuAC because this dataset exposes the context corpus only to the answering agent. QuAC includes multiple dialog instances and each instance includes a ”dialog history” of questions and answers asked in the dialog prior to the given question, along with some additional meta-data. The dataset was sourced from 8,854 unique sections of 3,611 unique Wikipedia articles.

Unique Wikipedia Sections	8,854
No. of Dialogs	13,594
QA pairs in each Dialog	4 to 12
Total QA pairs	98,407
Dataset size	74 MB
Train-Val-Test Split	85-7.5-7.5
tokens per section	401.0
tokens per question	6.5
tokens per answer	14.6
questions per dialog	7.2
percent yes/no	25.8

Table 1: Dataset Statistics

The data split is further delineated below.

	Train	Dev	Test
Unique Sections	6,843	1,000	1,002
Dialogs	11,567	1,000	1,002
Questions	83,568	7,354	7,353
tokens per section	396.8	440.0	445.8
tokens per question	6.5	6.5	6.5
tokens per answer	15.1	12.3	12.3
questions per dialog	7.2	7.4	7.3
percent yes/no	26.4	22.1	23.4
percent unanswerable	20.2	20.2	20.1

Table 2: Statistics summarizing each split of the dataset

The data consists of QA pairs, where the answers contain additional information such as ”Should a follow-up question be asked?” and ”Is this a yes or no answer question?”. There is added flexibility in this dataset as there can be questions that ”cannot be answered” in the given context.

Sample data

For a context (a section from the Wikipedia article) say, *‘In May 1983, she married Nikos Karvelas.....music hall at the time’* (20 lines), we have 8 Question instances.

A sample instance looks like this :
followup: y, *[continuation (follow up, maybe follow up, or don’t follow up)]*
yesno: n, *[affirmation (yes, no, or neither) for yes/no questions]*
question: did he win the lawsuit?,
answers : [.....]
[several reference answers]

orig_answer: { text: 'His lawsuit was unsuccessful,'...} *[ground truth]*

Data cleaning was not required other than dropping the metadata of data instances such as the first paragraph of Wikipedia articles, article titles and section titles which are not needed to train the model.

2. **Baseline Models**

   We have tried to improve two baseline models, namely FlowQA (11) and BiDAF++ (9).

   (a) FlowQA: FlowQA is a model designed specifically for conversational machine comprehension. It consists of two main components: a base neural model for single-turn MC and a FLOW mechanism that encodes conversation history which is crucial to grasping the flow of the conversation. The model integrates both previous question/answer pairs and FLOW : the intermediate context representation from conversation history. The F1 score on the QuAC dataset of this model is 64.1.

   (b) BiDAF++: BiDAF++ uses a bidirectional attention to capture contextual representation of questions. These representations are computed for each QA round independently and flown to the modelling layers. Each round depends only on the question and context of the current round and is independent of previous rounds. Since there is no summarizing involved, information loss is reduced. Attention is computed in both directions: context to query and vice versa, thus helping in answering questions that are paraphrased or do not use the terms exactly as in the document. The modeling layer encodes the context in a query aware representation, which aids in answering the questions posed by the agent. The F1 score on the QuAC dataset of this model is 50.2.

3. **Models and Analysis**

   Extensions to the existing architecture:

   (a) FlowQA with meta information: FlowQA was enhanced to predict meta information about the question itself. We think this was important as it gave us an insight into what the model has learnt and helped enhance interpretability. If the answering agent is able to figure out if the question in the given context can be answered with a yes or a no, it exhibits that the model is able to understand the scope of the question itself. If the model can find that the question cannot be answered in the given context (lack of information, unrelated query), it means it has been able to assimilate information from the document effectively. Another enhancement made to the architecture was to determine if the Questioning agent should ask follow up questions after seeing this answer. If our model is able to guess this correctly, it means the model is encouraging a dialogue exchange given the current state of the conversation. It is also able to predict whether the document has been fully explored in the conversation and if the answer has enough context to encourage a new question.

   (b) FlowQA with Attention: We also added attention to our FlowQA model because it is not necessary for all preceding questions to be equally important for generating the current context and answer. Using attention would enable us to gain context from only useful questions and would make answer-generation more precise. This is true of a natural human conversation itself. Some questions might be related to questions from several rounds ago but not to any other questions in the current conversation. Therefore, attention helps us learn useful representations and skip the unrelated ones.

   (c) BiDAF++ w/ BERT: Using BiDAF++ from QuAC as our baseline model, we wanted to examine the possibility of a performance improvement if we used state-of-the-art word embeddings in our model. We enhanced the original embeddings in the BiDAF++ architecture with BERT embeddings. We concatenated the BERT vector both at the token

Metric	FlowQA with meta	FlowQA with attention	BiDAF++ with BERT
Epochs	19	19	14
Learning Rate	0.1	0.1	0.1
Optimizer	adamax	adamax	adamax
Batch size	3	3	3
Elmo Batch Size	12	12	12
Train-Val-Test Split	85-7.5-7.5	85-7.5-7.5	85-7.5-7.5

Table 3: Experimental Set-Up for our models

level and at the contextual embeddings. Even though BiDAF++ is the model with the lowest performance metrics on the leaderboard, enhancing the architecture to use BERT embeddings made its performance surpass those of FlowQA(with meta information) and FlowQA(with attention).

4 Results

1. **Experimental Setup**

   We trained our models on Google Cloud using a NVIDIA Tesla V100 GPU supported by n1-highmem-8 (8 vCPUs, 52 GB memory). The following tables delineate the configurations to train each model. Since we had access to faster computing resources this time, we could train our models on the complete training dataset which wasn't the case in our midterm implementation.

   We saw that our models converged well before our predefined 30 epochs.

2. **Evaluation Metrics Used**

   Our core evaluation metric is the F1 score, which along with other metrics is outlined below.

   (a) The overall F1-score measures the average overlap between the prediction and the ground truth answer. We treat the prediction and ground truth as bags of tokens, compute their F1 after removing stop-words and determine the maximum F1 over all of the ground-truth answers for a given question. That is then averaged over all of the questions.

   (b) Accuracy On ”Cannot answer” measures how many of the total number of unanswerable questions (questions with the ’CANNOTANSWER’ tag) were predicted correctly. Note that CANNOTANSWER questions are handled separately and they do not account for the F1-score mentioned above.

   HEQ measures the proportion of dialogs for which the F1 score of the model is comparable to human performance.

   (a) HEQ-Q represents the number of questions for which the above statement is true, and

   (b) HEQ-D represents the proportion of dialogs for which every question of the dialog has performance comparable to human performance.

   Since QuAC is a standard dataset, we calculate and report our results on the standard test split. The following tables contain the results of our models - the metric used and the performance obtained on the dev set.

3. **Difference in behaviour of models**

   (a) Loss: The loss function for FlowQA (with meta information) has an additional term for the loss of yes/no and follow-up meta information too. These are computed as an L2 norm cost function. Therefore, the loss values of this architecture is significantly higher than the other models(FlowQA with attention and BiDAF enhanced with Bert).

   (b) In our midterm report, the models fall behind in performance because they were trained on a subset of the training data. However, on training them on the complete dataset and with sufficient computing resources, our baseline models improved in performance even though they trail the human F1 score.

Metric	FlowQA with meta	FlowQA with attention	BiDAF++ with BERT	Baseline: BiDAF(no context)	Baseline: FlowQA
Overall F1	60.96	59.1	63.9	50.2	64.1
Unfiltered F1	59.43	57.9	61.1	51.8	-
Yes/No Accuracy	44.01	-	86.9	85.4	-
Followup Accuracy	22.74	-	62.5	59.0	-
Accuracy On ”Cannot answer”	32.97	20.7	62.1	-	-
Human F1	80.8	80.8	80.8	80.8	80.8
HEQ-Q	56.351	54.4	59.5	43.3	59.6
HEQ-D	4.9	3.8	6.5	2.2	5.8

Table 4: Statistics summarizing the performance of all models over evaluation metrics. The last two columns represent the baseline models used for comparison.

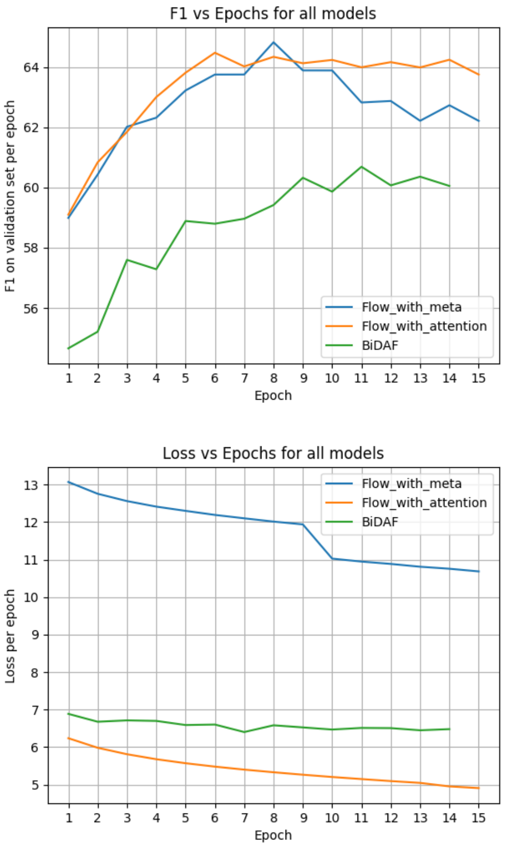

Figure 1: a) Training F1 of the models across training epochs b) Training loss of the models across training epochs

(c) We observe that amongst all our improvements, the BiDaf++ w/ Bert has the highest F1 score as well as the highest proportion of dialogs comparable to human performance. It also showed superior results in predicting the meta information as compared to the FlowQA model.

4. **Result Comparison**

(a) F1 score: We see that all our models have a significant improvement over the baseline BiDAF model and the performance of BiDAF boosted with BERT is close to the one achieved by the baseline FlowQA. Our augmented architectures of FlowQA do not perform as well as the baseline which is due to a different hyperparamater setting than the one used in the baseline. Even so, our model is able to reach performance metrics close to the SOTA without using BERT embeddings.

(b) Accuracy on Cannot Answer: BiDAF enhanced with BERT performs surprisingly well on this metric, given that the other two models perform worse than a random predictor(one which gives ”Cannot answer” 50% of the time). We believe it could be given the fact that some questions are vague enough to elicit a response from the document even when it is probably out of the scope of the document. The model also tries to predict an answer in cases where the question has phrases that pertain to the general idea of the document but is not entirely related to the paragraph. In these cases, the probability value of ”CANNOTANSWER” is not high enough for it to be considered an ”unanswerable question”.

(c) HEQ-Q: All our models beat BiDAF(baseline) and are fairly close to the baseline FlowQA model. Our models are close to 55%, which means that the model is able to simulate human-like conversations 55% of the times. This is a good indicator that our answering agent has learned effective representations of the document and the question in the context of the document.

(d) HEQ-D: It is important to note that BiDAF enhanced with BERT was able to beat all the baseline models and has an HEQ-D of 6.5. Our other models beat the performance of BiDAF(baseline) in this metric.

(e) Follow-up accuracy and Yes/No Questions accuracy: In both these metrics, BiDAF enhanced was able to outperform the baseline, although our FlowQA model performed significantly worse. We believe this means the representations learned by FlowQA are not descriptive enough to be able to answer meta information about the question. It encapsulates enough information to generate answers but not enough information about the question to be able to predict the meta information about the question.

5. Error Analysis

For analysing the performance and errors in our models, we used the following metrics:

(a) F1 score over QA turns: Using this score, we are able to analyse and evaluate the main objective of our project, which was to build an agent that not only answers the Q-bot but also encourages dialog. This helps us determine how viable a conversation with the model is and how much the performance degrades over turns. The models degrade in performance as the number of turns increase with a stark decrease for Flow with attention model (Our hypothesis for why this occurs is explained further ahead).

(b) F1 score over number of tokens in answer: This is used to see if the model starts to perform worse over questions that need more explanation or over questions that are not targeted. When questions are specific in nature, it is often easier to answer them even without having much contextual information. If the model is able to answer questions that require more elaborate answers, it implies that our model is able to tackle indirect/vague questions. It is also able to impart knowledge to the Q-bot about the document by giving it more details. As the number of tokens in the predicted answer increase, the performance of BiDAF enhanced with BERT increases too. We think it likely that this is due to the superior word embeddings used in this architecture. It makes answering paraphrased questions and indirect questions a lot easier.

Turn#	FlowQA with meta	FlowQA with attention	BiDAF++ with BERT
1	74.37	74.47	73.36
4	56.21	54.0	56.9
7	57.69	54.94	58.96
10	60.58	56.51	62.96

Table 5: F1 score of models over Turn#

Token#	FlowQA with meta	FlowQA with attention	BiDAF++ with BERT
1	76.27	80.25	55.32
6	57.13	57.72	60.64
11	61.43	57.49	65.67
16	58.47	58.17	63.98
21	57.15	59.36	63.69
26	52.28	50.86	64.21

Table 6: F1 score of models over Token length

From Figure 2, it is evident that BiDAF++ enhanced with BERT is able to outperform the other models in both turn-based and token-based metrics. This is because BERT embeddings are more diverse and representative of natural language. It holds more useful representation in a QA context where the Q-agent can generate questions that are paraphrased or are not completely related to the document.

It is also pertinent to recognize that our hypothesis of using attention over the flow model was not entirely successful. Our initial assumption being that attention would help improve the performance of FlowQA has been refuted in our experiments. We suspected not all questions are important in the context over multiple rounds and employing attention would let us skip certain questions. We think this hypothesis has been disproved primarily due to two reasons:

i. The number of rounds in the conversation on an average are 7. In such short dialogue instances, attention will not really help improve the performance since most of the context is already encoded. It is not necessary for the model to forget/skip any question while generating answers.

ii. The experiment setting is such that the answer of a previous question gives rise to a new question, much

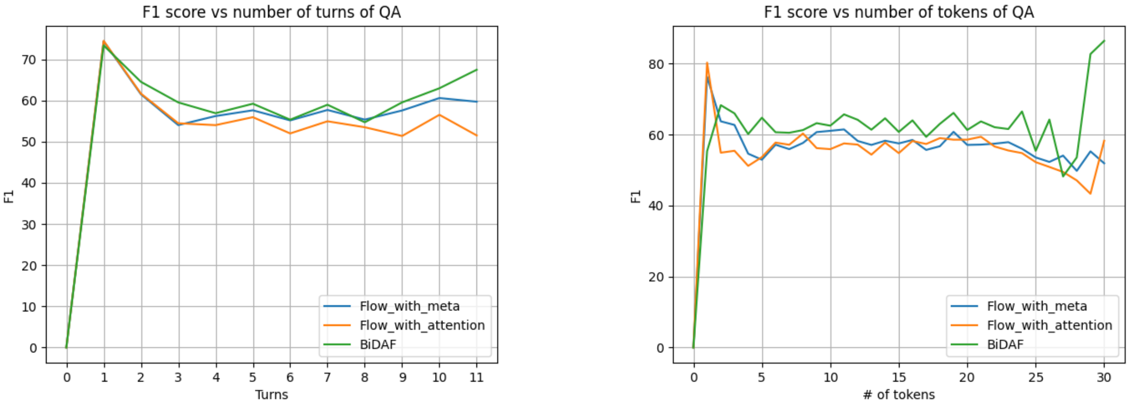

Figure 2: a) F1 values across different turns of QA b) F1 values of the models across different answer token lengths

like a customer service chat-bot. The lack of jumps in conversation arise mainly due to the fact that the Q-agent does not have access to the information (document). Therefore, all the questions asked are dependent on the answers given by the A-bot. Thus, in this case, attention is not necessary because the questions are generated serially (there are no jumps in the conversation).

(c) CANNOTANSWER Inaccuracy : We observed that the A-bots in both FlowQA models sometimes give out answers when the expected answer is CANNOTANSWER. However, we noticed that often, while the answer may be incorrect in the context of the conversation and the provided articles, the question-answer pair looks fairly meaningful in its absolute sense when its taken out of context.

(Article 666) Question : *was the band upset by this?*
Original Answer : CANNOTANSWER
FlowQA + Attention Answer : *As the group began recording their new album without him, he started working on his first solo album.*

(Article 505) Question : *When did he become president?*
Original Answer : CANNOTANSWER
FlowQA + Attention Answer : *Niyazov became president at the transition of Turkmenistan from a Soviet republic to an independent state.*

These happened when either the questions were not fact-based or when the predicted answer had contextual overlap with the question. The former case is plausible because, in these QA systems, we are not paraphrasing the answer but merely highlighting a span in the context document. As a result, the span doesn't directly answer the question but has implicit references to it. The latter case points to contextual ambiguity. Our models provide an implicit mapping of the question to the context and vice versa. Therefore, it assigns a higher probability to a span of text which corresponds to a contextually fitting answer (even when the question is not answerable).

(d) Shorter, precise model answers : We also observed that many answers given by the enhanced models were more precise than the original answers, i.e., the original answers could have done without some of the additional information.

(Article 787) Question : *What other movies did he star in?*
Original Answer : *1949 Disney animated film The Adventures of Ichabod and Mr. Toad, Crosby provided the narration and song vocals*
FlowQA+Meta Answer : *In the 1949 Disney animated film The Adventures of*

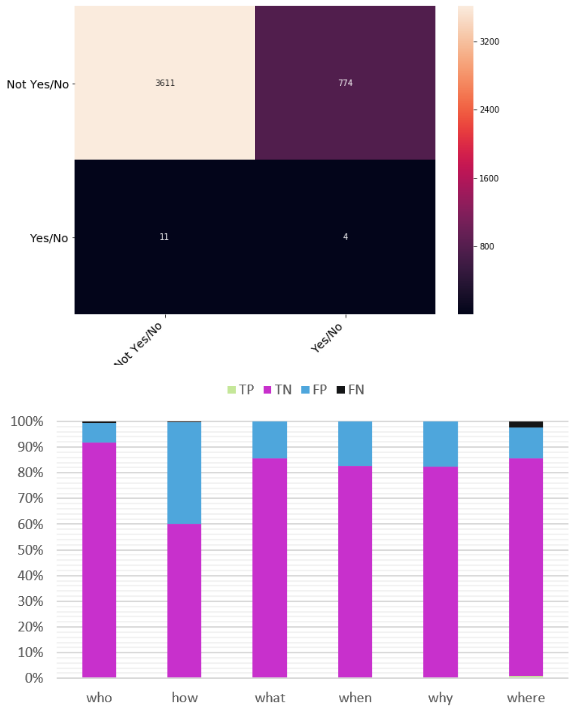

Figure 3: a) Confusion matrix for predicting Yes/No label for 5W1H questions b) Categorical distribution of Yes/No Label predictions for every type in 5W1H questions

Ichabod and Mr. Toad,

(Article 765) Question : *who took his place?*
Original Answer : *Toru took his place as lead guitar, and the band re-arranged their songs to be played for one guitar.*
FlowQA + Attention Answer : *Toru took his place as lead guitar,*

(e) Analysis of 5W1H questions : This error analysis is based on the hypothesis that affirmative questions (can be answered with a Yes/No) generally begin with verbs like *was, are, do* or modal auxiliaries like *can, should, will* and rarely with the 5W1H question-words *who, what, when, where, why, and how*. Using this, we wanted to analyse how well our model encapsulates language cues in questions that help in predicting meta information about the question itself.

6. **Work Division**

1. FlowQA with meta information preprocessing and training: Shalini
2. FlowQA with Attention preprocessing and training: Rohit
3. BiDAF++ preprocessing and training: Sushmita
4. Evaluation of predictions: Rohit and Sushmita
5. Comparison and Error Analysis of Models: Shalini
6. Final Report: Rohit, Shalini and Sushmita
7. Video : Rohit

5 Conclusion

In this project, we attempted to improve two of the existing models for Question-Answering in a contextual setting. We modified FlowQA to predict meta information related to the dialog - yes/no (affirmation) and follow-up (continuity). In another experiment, we modified the FlowQA model by adding attention in the ”flow” operation, assuming that the current question may depend on a subset of historical questions and not all of them. Finally, we also enhanced the baseline model in the QuAC paper by appending state-of-the-art BERT embeddings to get better performance. Our experiments demonstrated that using state of the art word embeddings definitely have a performance boost on the model and achieve a closer HEQ index. We also established that in short conversation rounds where most questions are derived from previous answers, attention over flow does not add any significant improvement and may even lead to a worse performance. We also recognized that as the number of turns of QA increases, the quality of predicted answers decreases. As the number of tokens in the answer increase, the model performance degrades indicating that the model works well for direct/specific questions which have localized answers. From our initial findings, future directions for the project could include implementing ensemble models or techniques like coreference resolution to enhance the baselines. Output from a coreference model (such as the one proposed by Lee et. al. (12)) fed to FlowQA such that the internal representations of Named-Entity tokens and their coreferenced POS can be mapped to further boost the performance of the ContextQA system. Another potential direction that can be explored is a pair of Q-bot and A-bot that is trained using reinforcement learning and is rewarded dependent upon the portion of the document it explores and the number of innovative questions the Q-bot asks.

References

[1] C. Xiong, V. Zhong, and R. Socher, “Dynamic coattention networks for question answering,” *CoRR*, vol. abs/1611.01604, 2016.

[2] D. Chen, J. Bolton, and C. D. Manning, “A thorough examination of the CNN/daily mail reading comprehension task,” in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, (Berlin, Germany), pp. 2358–2367, Association for Computational Linguistics, Aug. 2016.

[3] M. Stede and D. Schlangen, “Information-seeking chat: Dialogues driven by topic-structure,” 2004.

[4] C. Xiong, S. Merity, and R. Socher, “Dynamic memory networks for visual and textual question answering,” *CoRR*, vol. abs/1603.01417, 2016.

[5] M. Richardson, C. J. C. Burges, and E. Renshaw, “Mctest: A challenge dataset for the open-domain machine comprehension of text,” in *In Proceedings of EMNLP*, 2013.

[6] P. Rajpurkar, J. Zhang, K. Lopyrev, and P. Liang, “Squad: 100, 000+ questions for machine comprehension of text,” *CoRR*, vol. abs/1606.05250, 2016.

[7] E. Choi, H. He, M. Iyyer, M. Yatskar, W. Yih, Y. Choi, P. Liang, and L. Zettlemoyer, “Quac : Question answering in context,” *CoRR*, vol. abs/1808.07036, 2018.

[8] S. Reddy, D. Chen, and C. D. Manning, “Coqa: A conversational question answering challenge,” *CoRR*, vol. abs/1808.07042, 2018.

[9] M. Seo, A. Kembhavi, A. Farhadi, and H. Hajishirzi, “Bidirectional attention flow for machine comprehension,” 2016.

[10] M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer, “Deep contextualized word representations,” *CoRR*, vol. abs/1802.05365, 2018.

[11] H. Huang, E. Choi, and W. Yih, “Flowqa: Grasping flow in history for conversational machine comprehension,” *CoRR*, vol. abs/1810.06683, 2018.

[12] K. Lee, L. He, M. Lewis, and L. Zettlemoyer, “End-to-end neural coreference resolution,” *CoRR*, vol. abs/1707.07045, 2017.