BERT with History Answer Embedding for Conversational Question Answering

Chen Qu[1] Liu Yang[1] Minghui Qiu[2] W. Bruce Croft[1] Yongfeng Zhang[3] Mohit Iyyer[1]
[1] University of Massachusetts Amherst [2] Alibaba Group [3] Rutgers University
{chenqu,lyang,croft,miyyer}@cs.umass.edu,minghui.qmh@alibaba-inc.com,yongfeng.zhang@rutgers.edu

ABSTRACT

Conversational search is an emerging topic in the information retrieval community. One of the major challenges to multi-turn conversational search is to model the conversation history to answer the current question. Existing methods either prepend history turns to the current question or use complicated attention mechanisms to model the history. We propose a conceptually simple yet highly effective approach referred to as *history answer embedding*. It enables seamless integration of conversation history into a conversational question answering (ConvQA) model built on BERT (Bidirectional Encoder Representations from Transformers). We first explain our view that ConvQA is a simplified but concrete setting of conversational search, and then we provide a general framework to solve ConvQA. We further demonstrate the effectiveness of our approach under this framework. Finally, we analyze the impact of different numbers of history turns under different settings to provide new insights into conversation history modeling in ConvQA.

ACM Reference Format:
Chen Qu, Liu Yang, Minghui Qiu, W. Bruce Croft, Yongfeng Zhang, and Mohit Iyyer. 2019. BERT with History Answer Embedding for Conversational Question Answering. In *Proceedings of the 42nd Int'l ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'19), July 21–25, 2019, Paris, France.* ACM, NY, NY, USA, 4 pages. https://doi.org/10.1145/3331184.3331341

1 INTRODUCTION

A long-term goal of the information retrieval (IR) community has been to design a search system that can retrieve information iteratively and interactively [1, 4, 8]. The emerging field of conversational AI has impacted this goal, leading to a direction referred to as conversational search. Conversational AI consists of three branches, namely, task-oriented bots, social bots, and question answering (QA) bots [6]. The first two have attracted extensive research efforts in the recent years, resulting in a wide range of personal assistants, such as Siri and Cortana. These systems, however, are not capable of handling complicated information-seeking conversations that require multiple turns of information exchange. Much work remains to empower common users to conduct conversational search.

SIGIR '19, July 21–25, 2019, Paris, France

ACM ISBN 978-1-4503-6172-9/19/07…$15.00
https://doi.org/10.1145/3331184.3331341

It is natural for people to seek information through conversations. In the setting of conversational search, a user initiates a conversation with a specific information need. The search system conducts multiple turns of interaction with the user via a "System Ask, User Respond" paradigm [19] to better understand this information need. The system then tries to fulfill this need by retrieving answers iteratively based on the user's feedback or clarifying questions. The user sometimes asks follow-up questions with a related but new information need and thus enters the next "cycle" of the conversational search process. In order to understand the user's latest information need, the system should be capable to handle the conversation history. In our view, conversational question answering (ConvQA) is a simplified setting of conversational search since ConvQA systems do not focus on asking proactively. However, ConvQA is concrete enough for IR researchers to work on modeling the change of information needs between cycles. Therefore, we focus on handling conversation history in a ConvQA setting.

ConvQA can be formalized as the relatively well-studied machine comprehension (MC) problem [3, 13, 20]. This is achieved by incorporating the conversation history into an MC model. There are two aspects to handle this. The first is history selection, which selects a subset of the history turns that are more helpful than others. The second is history modeling, which models the selected history turns in an MC model. Thus, we define a general framework to describe these two aspects and lay the groundwork for future efforts with ConvQA. We focus on the history modeling aspect in this work and adopt a rule-based method for history selection.

History modeling is essential for ConvQA. Table 1 shows a part of a dialog from a ConvQA dataset (QuAC [3]). When the user issues the query Q_2, we expect the agent to refer to A_1 so that it can understand the meaning of "that way". In such cases, previous history turns play an essential role in understanding the user's current information need.

Table 1: A part of an information-seeking dialog from QuAC. "R", "U", and "A" denotes role, user, and agent respectively.

Topic: Augusto Pinochet: Intellectual life			
#	ID	R	Utterance
1	Q_1	U	Was he known for being intelligent
	A_1	A	No, Pinochet was publicly known as a man with a lack of culture.
2	Q_2	U	Why did people feel that way?
	A_2	A	reinforced by the fact that he also portrayed himself as a common man

Some existing methods simply prepend history turns [13, 20] or mark answers in the passage [3]. These methods cannot handle a long conversation history. Another existing method [7] uses complicated attention mechanisms to model history and thus generates relatively large system overhead. We propose a *history answer*

embedding method to model conversation history. Our method is conceptually simple, robust, effective, and has better training efficiency compared to previous approaches. Moreover, our method is specifically tailored for BERT-based architectures to leverage this latest breakthrough in large scale pre-trained language modeling.

We summarize our contribution as follows. (1) We introduce a general framework to handle the conversation history in ConvQA, laying the groundwork for future efforts with this task. (2) Our proposed history modeling method is one of the first attempts to model conversation history in a BERT-based model for information-seeking conversations.[1] We conduct extensive experiments on QuAC, a large open benchmark, to show the effectiveness of our method. Our methods achieved an F1 score of 62.4 on the QuAC leaderboard[2] with a significantly shorter training time compared with the state-of-the-art method. Our code is open sourced.[3] (3) We perform an in-depth analysis to show the impact of different amounts of conversation history. We show that history prepending methods degrade dramatically with long history while our method is robust and shows advantages under such a situation, which provides new insights into conversation history modeling in ConvQA.

2 RELATED WORK

ConvQA is closely related to machine comprehension. High quality datasets [9, 12] have boosted research progress, resulting in a wide range of MC models [2, 14]. A major difference between ConvQA and MC is that questions in ConvQA are organized in conversations. Thus, we need to model conversation history to understand the current question. Compared to existing methods that prepend history turns [13, 20] to the current question or mark history answers in the passage [3], our method can handle longer conversation history and thus is more robust and effective. In addition, our method is conceptually simple and more efficient than FlowQA [7] that uses complicated recurrent structures. Our method is specifically tailored for BERT [5], which pre-trains language representations with bidirectional encoder representations from transformers [16].

CoQA [13] and QuAC [3] are ConvQA datasets with very different properties. Questions in CoQA are often factoid with simple entity-based answers while QuAC consists of mostly non-factoid QAs. More importantly, information-seekers in QuAC have access to the title of the passage only, simulating an information need. The information-seeking setting in QuAC is more in line with our interest as IR researchers. Thus, we focus on QuAC in this work.

In addition to ConvQA, there are other related works focused on conversational search. For example, neural approaches are widely adopted to train a model to ask questions proactively [19], predict user intent [11], predict next question [17], and incorporate external knowledge in response ranking [18]. In addition, several observational studies are also conducted [10, 15] to inform the design of conversational search systems. We focus on dealing with conversation history in this work, which is an integral part in the joint effort of building functional conversational search systems.

[1] Most existing models are tested on CoQA, which is not information-seeking (see Section 2). Moreover, descriptions of these models are not available at the time of our paper submission. [2] http://quac.ai/ [3] https://github.com/prdwb/bert_hae

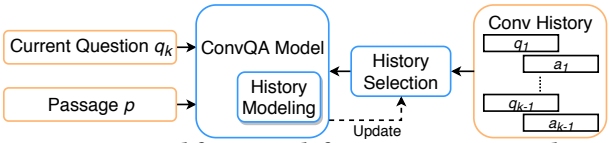


Figure 1: A general framework for ConvQA. Orange denotes model input and blue denotes model components.

3 OUR APPROACH

3.1 Task Definition

The ConvQA task is defined as follows [3, 13]. Given a passage p, the k-th question q_k in a dialog, and the conversation history $\mathbf{H}_k$ preceding q_k, the task is to answer q_k by predicting an answer span a_k within p. $\mathbf{H}_k$ has $k-1$ turns, where the i-th turn is $\mathbf{H}_k^i = (q_i, a_i)$. q_i and a_i denote the question and the ground truth answer.

3.2 A ConvQA Framework

We present an abstract framework for ConvQA with modularized design in Figure 1. It consists of three major components, a ConvQA model, a history selection module, and a history modeling module. In practice, the history modeling module can be a mechanism inside the ConvQA model. Given a training instance $(p, q_k, \mathbf{H}_k, a_k)$, the history selection module chooses a subset of the history turns $\mathbf{H}'_k$ that are expected to be more helpful than others. The history modeling module then incorporates $\mathbf{H}'_k$ into the ConvQA model. If the history selection module is a learned policy, the ConvQA model can generate a signal to guide its update. In this work, we employ a simple rule as the history selection module that always chooses the immediate j previous turn(s). This is based on the intuition that closer history turns are typically more relevant to the current question. We introduce our implementations for the ConvQA model and the history modeling module in the following sections.

3.3 BERT with History Answer Embedding

Our implementation for the ConvQA model can be considered as an MC model integrated with a history modeling mechanism.

3.3.1 ***Machine Comprehension.*** Our model is adapted from the BERT-based MC model by Devlin et al. [5]. The input is a question and a passage, and the output is the probability of passage tokens being the start/end token of the answer span. We illustrate the model architecture in Figure 2. First, the question and the passage are packed into a sequence. Then BERT generates a representation for each token based on the embeddings for tokens, segments, and positions. After that, a start/end vector is learned to compute the probability of a token being the start/end token of the answer span. Specifically, let $\mathbf{T}_i$ be the BERT representation of the i-th token and $\mathbf{S} \in \mathbb{R}^h$ be the start vector, where h is the token representation size. The probability of this token being the start token is $P_i = \frac{e^{\mathbf{S} \cdot \mathbf{T}_i}}{\sum_j e^{\mathbf{S} \cdot \mathbf{T}_j}}$. The probability of a token being the end token is computed likewise. The loss is the average of the cross entropy loss for the start and end positions. Invalid predictions are discarded at testing time.

3.3.2 ***History Answer Embedding.*** One important difference of MC and ConvQA lies in handling conversation history. Suppose we are given a subset of the conversation history chosen by the

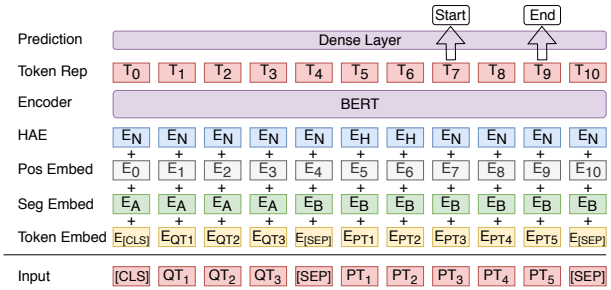


Figure 2: Architecture of the ConvQA model with HAE. E_H/E_N in HAE denote the token is in/not in history answers.

history selection module for the current question. There are various ways to model the selected history turns. The most intuitive way is to prepend the conversation history to the current question [13, 20]. In this work, we propose a different approach to model the conversation history by giving tokens extra embedding information. As shown in Figure 2, a *history answer embedding* (HAE) layer is included in addition to other embeddings. We learn two unique history answer embeddings that denote whether a token is part of history answers or not. This introduces the conversation history to BERT in a natural way. HAE modifies the embedding information for a token and thus has influence on the token representation generated by BERT, not only for this token but also for other tokens since BERT considers contextual information. This process also improves the prediction of the answer span as shown in the experiments. By representing conversation history with HAE, we turn an MC model into a ConvQA model.

3.3.3 ***Model Training.*** Given a training instance $(p, q_k, \mathbf{H}_k, a_k)$, we first transform it to a list of variations, where each variation $(p, q_k, \mathbf{H}_k^i, a_k)$ contains only one turn of the conversation history. A history selection module then selects immediate previous j turns. After that, we merge the selected variations to form a new instance $(p, q_k, \mathbf{H}_k', a_k)$. It is then used to generate input for the ConvQA model, where $\mathbf{H}_k'$ is used for HAE. We use a sliding window approach to split long passages following Devlin et al. [5].

4 EXPERIMENTS

4.1 Data Description

We experiment with QuAC (Question Answering in Context) [3]. This dataset contains interactive dialogs between information-seekers and -providers. The seeker tries to learn about a *hidden* Wikipedia passage by asking questions. He/she has access to the heading of the passage only. The provider answers the questions by giving a short span of the passage. Many questions have co-references with conversation history. The training/validation sets have over 11K/1K dialogs with 83K/7K questions. All dialogs are within 12 turns, meaning that a question can have at most 11 history turns.

4.2 Experimental Setup

4.2.1 ***Competing Methods.*** We consider all the methods on the QuAC leaderboard as baselines. The competing methods are:

- **BiDAF++** [3]: BiDAF++ augments BiDAF [14] with self-attention and contextualized embeddings.
- **BiDAF++ w/ 2-Context** [3]: It incorporates 2 history turns in BiDAF++ by encoding the dialog turn # in question embeddings and concatenating marker embeddings to passage embeddings.
- **FlowQA** [7]: It considers conversation history by integrating intermediate representation generated when answering previous questions and thus can grasp the latent semantics of the history.
- **BERT**: We implement a ConvQA model with BERT as described in Section 3.3.1. This version is without any history modeling.
- **BERT + Prepend History Turns**: On top of BERT, we consider conversation history by prepending history turn(s) to the current question. **BERT + PHQA** prepends both history questions and answers; **BERT + PHA** prepends history answers only.
- **BERT + History Answer Embedding (HAE)**: A BERT-based ConvQA model with our history answer embedding method.

4.2.2 ***Evaluation Metrics.*** We use the word-level F1 to evaluate the overlap of the prediction and the ground truth answer and HEQ (human equivalence score) to measure the percentage of examples where system F1 exceeds/matches human F1. HEQ is computed on a question level (HEQ-Q) and a dialog level (HEQ-D).

4.2.3 ***Implementation Details.*** Models are implemented with TensorFlow.[4] We use the BERT-Base (Uncased) model[5] with the max sequence length set to 384. The batch size is set to 12. The number of history turns to incorporate is tuned as presented in Section 4.4. We train the ConvQA model with an Adam weight decay optimizer with an initial learning rate of 3e-5. We set the stride in the sliding window for passages to 128, the max question length to 64, and the max answer length to 30. We save checkpoints every 1,000 steps and test on the validation set. We use QuAC v0.2.

4.3 Main Evaluation Results

Experiment results are shown in Table 2. Our best model was evaluated officially and the result is displayed on the leaderboard[6].

Table 2: Evaluation results. Each cell displays val/test scores. Test results are from the QuAC leaderboard on 02/17/2019. ‡ means statistically significant improvement over other methods (except FlowQA) with $p < 0.05$ tested by the Student's paired t-test. We can only do significance test on F1.

Models	F1	HEQ-Q	HEQ-D	Train Time (h)
BiDAF++	51.8 / 50.2	45.3 / 43.3	2.0 / 2.2	-
BiDAF++ w/ 2-Context	60.6 / 60.1	55.7 / 54.8	5.3 / 4.0	-
FlowQA	**64.6** / **64.1**	– / **59.6**	– / **5.8**	56.8
BERT	54.4 / –	48.9 / –	2.9 / –	6.8
BERT + PHQA	62.0 / –	57.5 / –	5.4 / –	7.9
BERT + PHA	61.8 / –	57.5 / –	4.7 / –	7.2
BERT + HAE	**63.1**‡ / **62.4**	**58.6** / **57.8**	**6.0** / **5.1**	10.1

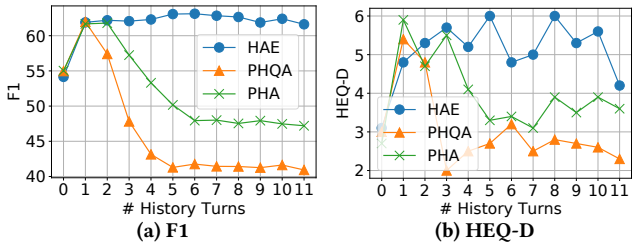
We summarize our observations as follows. (1) Incorporating conversation history significantly boosts the performance in ConvQA. This is true for both BiDAF++ and BERT-based models. This not only suggests the importance of conversation history, but also shows the effectiveness of our history modeling approaches. (2) Our BERT-based ConvQA model outperforms BiDAF++. Furthermore, BERT with any of the history modeling methods outperform BiDAF++

[4] https://www.tensorflow.org/ [5] http://goo.gl/language/bert [6] http://quac.ai/

w/ 2-Context. This shows the advantage of using BERT for ConvQA. (3) Prepending history turns with PHQA and PHA are both effective. The fact that they achieve similar performance suggests that history questions contribute little to the performance. This verifies our observation of the data that most follow-up questions are relevant to history answers. (4) Our HAE approach achieves better performance than simply prepending history turns. This indicates that HAE is more effective in modeling conversation history. (5) HAE manages to perform reasonably well with a relatively simple history modeling approach compared with the state-of-the-art method FlowQA. (6) In addition to the model performance, we also compare the training efficiency. We observe our models are at least 5 times faster than FlowQA in training.[7] Prepending history has little impact on training efficiency. Compared to PHQA, PHA is slightly faster because it only considers history answers. HAE is slightly slower than PH(Q)A but achieves considerable improvements. Compared to FlowQA, our HAE method achieves comparable performance with much higher training efficiency.

4.4 Impact of Conversation History

We then give an in-depth analysis on the impact of different amounts of conversation history with different history modeling approaches.



(a) F1 (b) HEQ-D

Figure 3: Impact of different amounts of conversation history with different history modeling methods with BERT.

As presented in Figure 3, the most important observation is that our history answer embedding method can handle more conversation histories than simply prepending history turns, which shows the robustness of HAE. More importantly, the ability to model more history turns indeed gives some gains. This is based on the fact that HAE with 5 or 6 history turns gives the best performance. In addition, Choi et al. [3] also show that their answer marking method in BiDAF++ saturates at two turns. This further verifies the capability of our method to handle long conversation history.

Another interesting observation is that both PHQA and PHA show dramatic degradation as the number of history turns grows. The best performance of PH(Q)A occurs at considering only 1 or 2 history turns. This is consistent with the results by Reddy et al. [13], who use the same history modeling approach on Seq2Seq and DrQA [2]. The low performance when prepending a large amount of history suggests that a BERT-based ConvQA model is especially vulnerable to long prepended questions. This can be explained by the mechanism of constructing the input sequence as explained in Section 3.3.1. A long prepended question shrinks the passage part in the input sequence and affect the answer prediction performance.

[7] We use the code at https://github.com/momohuang/FlowQA, which was released by original authors. We set the batch size to 1 *dialog* per batch to avoid memory issues.

This also explains the observation that PHQA drops faster as it also prepends history questions in addition to answers. These results show that history answer embedding is a better history modeling approach in a BERT-based ConvQA model. This is reasonable as HAE can be seamlessly integrated into BERT as shown in Figure 2.

5 CONCLUSIONS AND FUTURE WORK

In this work, we introduce a general framework for ConvQA to illustrate the two aspects of handling conversation history. We then propose a history answer embedding method to model conversation history in ConvQA. Extensive experiments show the effectiveness of our method. Finally, we perform an in-depth analysis to show the impact of different amounts of conversation history under different settings. Future work will consider to integrate our history modeling method with a learned history selection strategy for ConvQA.

ACKNOWLEDGMENTS

This work was supported in part by the Center for Intelligent Information Retrieval and in part by NSF IIS-1715095. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect those of the sponsor.

REFERENCES

[1] N. J. Belkin, C. Cool, A. S., and U. Thiel. Cases , Scripts , and Information-Seeking Strategies : On the Design of Interactive Information Retrieval Systems. 1994.
[2] D. Chen, A. Fisch, J. Weston, and A. Bordes. Reading Wikipedia to Answer Open-Domain Questions. In *ACL*, 2017.
[3] E. Choi, H. He, M. Iyyer, M. Yatskar, W. Yih, Y. Choi, P. Liang, and L. S. Zettlemoyer. QuAC: Question Answering in Context. In *EMNLP*, 2018.
[4] W. B. Croft and R. H. Thompson. I3R: A new approach to the design of document retrieval systems. *JASIS*, 38:389–404, 1987.
[5] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *CoRR*, 2018.
[6] J. Gao, M. Galley, and L. Li. Neural Approaches to Conversational AI. In *SIGIR*, 2018.
[7] H.-Y. Huang, E. Choi, and W. Yih. FlowQA: Grasping Flow in History for Conversational Machine Comprehension. *CoRR*, 2018.
[8] Alexander Kotov and ChengXiang Zhai. Towards natural question guided search. In *WWW*, 2010.
[9] T. Nguyen, M. Rosenberg, X. Song, J. Gao, S. Tiwary, R. Majumder, and L. Deng. MS MARCO: A Human Generated MAchine Reading COmprehension Dataset. *CoRR*, abs/1611.09268, 2016.
[10] C. Qu, L. Yang, W. B. Croft, J. R. Trippas, Y. Zhang, and M. Qiu. Analyzing and Characterizing User Intent in Information-seeking Conversations. In *SIGIR*, 2018.
[11] C. Qu, L. Yang, W. B. Croft, Y. Zhang, J. R. Trippas, and M. Qiu. User Intent Prediction in Information-seeking Conversations. *CoRR*, 2019.
[12] P. Rajpurkar, J. Zhang, K. Lopyrev, and P. Liang. SQuAD: 100, 000+ Questions for Machine Comprehension of Text. In *EMNLP*, 2016.
[13] S. Reddy, D. Chen, and C. D. Manning. CoQA: A Conversational Question Answering Challenge. *CoRR*, abs/1808.07042, 2018.
[14] M. J. Seo, A. Kembhavi, A. Farhadi, and H. Hajishirzi. Bidirectional Attention Flow for Machine Comprehension. *CoRR*, abs/1611.01603, 2016.
[15] J. R. Trippas, D. Spina, L. Cavedon, H. Joho, and M. Sanderson. Informing the Design of Spoken Conversational Search: Perspective Paper. In *CHIIR*, 2018.
[16] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin. Attention Is All You Need. In *NIPS*, 2017.
[17] L. Yang, H. Zamani, Y. Zhang, J. Guo, and W. B. Croft. Neural Matching Models for Question Retrieval and Next Question Prediction in Conversation. *CoRR*, 2017.
[18] L. Yang, M. Qiu, C. Qu, J. Guo, Y. Zhang, W. B. Croft, J. Huang, and H. Chen. Response Ranking with Deep Matching Networks and External Knowledge in Information-seeking Conversation Systems. In *SIGIR*, 2018.
[19] Y. Zhang, X. Chen, Q. Ai, L. Yang, and W. B. Croft. Towards Conversational Search and Recommendation: System Ask, User Respond. In *CIKM*, 2018.
[20] C. Zhu, M. Zeng, and X. Huang. SDNet: Contextualized Attention-based Deep Network for Conversational Question Answering. *CoRR*, 2018.