Attentive History Selection for Conversational Question Answering

Chen Qu[1] Liu Yang[1*] Minghui Qiu[2] Yongfeng Zhang[3] Cen Chen[4]
W. Bruce Croft[1] Mohit Iyyer[1]
[1] University of Massachusetts Amherst [2] Alibaba Group [3] Rutgers University [4] Ant Financial Services Group
{chenqu,lyang,croft,miyyer}@cs.umass.edu,minghui.qmh@alibaba-inc.com
yongfeng.zhang@rutgers.edu,chencen.cc@antfin.com

ABSTRACT

Conversational question answering (ConvQA) is a simplified but concrete setting of conversational search [24]. One of its major challenges is to leverage the conversation history to understand and answer the current question. In this work, we propose a novel solution for ConvQA that involves three aspects. First, we propose a *positional history answer embedding* method to encode conversation history with position information using BERT [6] in a natural way. BERT is a powerful technique for text representation. Second, we design a *history attention mechanism* (HAM) to conduct a "soft selection" for conversation histories. This method attends to history turns with different weights based on how helpful they are on answering the current question. Third, in addition to handling conversation history, we take advantage of *multi-task learning* (MTL) to do answer prediction along with another essential conversation task (dialog act prediction) using a uniform model architecture. MTL is able to learn more expressive and generic representations to improve the performance of ConvQA. We demonstrate the effectiveness of our model with extensive experimental evaluations on QuAC, a large-scale ConvQA dataset. We show that position information plays an important role in conversation history modeling. We also visualize the history attention and provide new insights into conversation history understanding.

KEYWORDS

Conversational Question Answering; Multi-turn Question Answering; Conversation History; Attention

ACM Reference Format:
Chen Qu, Liu Yang, Minghui Qiu, Yongfeng Zhang, Cen Chen, W. Bruce Croft, and Mohit Iyyer. 2019. Attentive History Selection for Conversational Question Answering. In *The 28th ACM International Conference on Information and Knowledge Management (CIKM '19), November 3–7, 2019, Beijing, China.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3357384.3357905

[*] Liu Yang is at Google currently.

CIKM '19, November 3–7, 2019, Beijing, China

ACM ISBN 978-1-4503-6976-3/19/11…$15.00
https://doi.org/10.1145/3357384.3357905

1 INTRODUCTION

It has been a longstanding goal in the information retrieval (IR) community to design a search system that can retrieve information in an interactive and iterative manner [1, 5, 13, 19]. With the rapid development of artificial intelligence and conversational AI [7], IR researchers have begun to explore a concrete implementation of this research goal, referred to as conversational search. Contributions from both industry and academia have greatly boosted the research progress in conversational AI, resulting in a wide range of personal assistant products. Typical examples include Apple Siri, Google Assistant, Amazon Alexa, and Alibaba AliMe [15]. An increasing number of users are relying on these systems to finish everyday tasks, such as setting a timer or placing an order. Some users also interact with them for entertainment or even as an emotional companion. Although current personal assistant systems are capable of completing tasks and even conducting smalltalk, they cannot handle information-seeking conversations with complicated information needs that require multiple turns of interaction. Conversational personal assistant systems serve as an appropriate media for interactive information retrieval, but much work needs to be done to enable functional conversational search via such systems.

A typical conversational search process involves multiple "cycles" [24]. In each cycle, a user first specifies an information need and then an agent (a system) retrieves answers iteratively either based on the user's feedback or by asking for missing information proactively [38]. The user could ask a follow-up question and shift to a new but related information need, entering the next cycle of conversational search. Previous work [24] argues that conversational question answering (ConvQA) is a simplified but concrete setting of conversational search. Although the current ConvQA setting does not involve asking spontaneously, it is a tangible task for researchers to work on modeling the change of information needs across cycles. Meanwhile, conversation history plays an important role in understanding the latest information need and thus is beneficial for answering the current question. For example, we show that coreferences are common across conversation history in Table 1. Therefore, one of the major focuses of this work is handling conversation history in a ConvQA setting.

In two recent ConvQA datasets, QuAC [2] and CoQA [27], ConvQA is formalized as an answer span prediction problem similar in SQuAD [25, 26]. Specifically, given a question, a passage, and the conversation history preceding the question, the task is to predict a span in the passage that answers the question. In contrast to typical machine comprehension (MC) models, it is essential to handle conversation history in this task. Previous work [24] introduced a

Table 1: An example of an information-seeking dialog from QuAC. "R", "U", and "A" denote role, user, and agent. Co-references and related terms are marked in the same color across history turns. Q_2, Q_4, Q_5 and Q_6 are closely related to their immediate previous turn(s) while Q_7 is related to a remote question Q_1. Also, Q_3 does not follow up on Q_2 but shifts to a new topic. This table is best viewed in color.

Topic: Lorrie Morgan's music career			
#	ID	R	Utterance
1	Q_1	U	What is relevant about Lorrie's musical career?
	A_1	A	... her first album on that label, Leave the Light On, was released in 1989.
2	Q_2	U	What songs are included in the album?
	A_2	A	CANNOTANSWER
3	Q_3	U	Are there any other interesting aspects about this article?
	A_3	A	made her first appearance on the Grand Ole Opry at age 13,
4	Q_4	U	What did she do after her first appearance?
	A_4	A	... she took over his band at age 16 and began leading the group ...
5	Q_5	U	What important work did she do with the band?
	A_5	A	leading the group through various club gigs.
6	Q_6	U	What songs did she played with the group?
	A_6	A	CANNOTANSWER
7	Q_7	U	What are other interesting aspects of her musical career?
	A_6	A	*To be predicted ...*

general framework to deal with conversation history in ConvQA, where a history selection module first selects helpful history turns and a history modeling module then incorporates the selected turns. In this work, we extend the same concepts of history selection and modeling with a fundamentally different model architecture.

On the aspect of history selection, existing models [2, 27] select conversation history with a simple heuristic that assumes immediate previous turns are more helpful than others. This assumption, however, is not necessarily true. Yatskar [36] conducted a qualitative analysis on QuAC by observing 50 randomly sampled passages and their corresponding 302 questions. He showed that 35.4% and 5.6% of questions have the dialog behaviors of *topic shift* and *topic return* respectively. A topic shift suggests that the current question shifts to a new topic, such as the Q_3 in Table 1. While topic return means that the current question is about a topic that has previously been shifted away from. For example, Q_7 returns to the same topic in Q_1 in Table 1. In both cases, the current question is not directly relevant to immediate previous turns. It could be unhelpful or even harmful to always incorporate immediate previous turns. Although we expect this heuristic to work well in many cases where the current question is *drilling down* on the topic being discussed, it might not work for topic shift or topic return. There is no published work that focuses on *learning* to select or re-weight conversation history turns. To address this issue, we propose a *history attention mechanism* (HAM) that learns to attend to all available history turns with different weights. This method increases the scope of candidate histories to include remote yet potentially helpful history turns. Meanwhile, it promotes useful history turns with large attention weights and demotes unhelpful ones with small weights. More importantly, the history attention weights provide explainable interpretations to understand the model results and thus can provide new insights in this task.

In addition, on the aspect of history modeling, some existing methods either simply prepend the selected history turns to the current question [27, 39] or use complicated recurrent structures to model the conversation history [11], generating relatively large system overhead. Another work [24] introduces a history answer embedding (HAE) method to incorporate the conversation history to BERT in a natural way. However, they fail to consider the position of a history utterance in the dialog. Since the utility of a history utterance could be related to its position, we propose to consider the position information in HAE, resulting in a *positional history answer embedding* (PosHAE) method. We show that position information plays an important role in conversation history modeling.

Furthermore, we introduce a new angle to tackle the problem of ConvQA. We take advantage of *multi-task learning* (MTL) to do answer span prediction along with another essential conversation task (dialog act prediction) using a uniform model architecture. Dialog act prediction is necessary in ConvQA systems because dialog acts can reveal crucial information about user intents and thus help the system provide better answers. More importantly, by applying this multi-task learning scheme, the model learns to produce more generic and expressive representations [17], due to additional supervising signals and the regularization effect when optimizing for multiple tasks. We show that these benefits have contributions to the model performance for the dialog action prediction task.

In this work, we propose a novel solution to tackle ConvQA. We boost the performance from three different angles, i.e., history selection, history modeling, and multi-task learning. Our contributions can be summarized as follows:

(1) To better conduct history selection, we introduce a history attention mechanism to conduct a "soft selection" for conversation histories. This method attends to history turns with different weights based on how helpful they are on answering the current question. This method enjoys good explainability and can provide new insights to the ConvQA task.
(2) To enhance history modeling, we incorporate the history position information into history answer embedding [24], resulting in a positional history answer embedding method. Inspired by the latest breakthrough in language modeling, we leverage BERT to jointly model the given question, passage and conversation history, where BERT is adapted to a conversation setting.
(3) To further improve the performance of ConvQA, we jointly learn answer span prediction and dialog act prediction in a multi-task learning setting. We take advantage of MTL to learn more generalizable representations.
(4) We conduct extensive experimental evaluations to demonstrate the effectiveness of our model and to provide new insights for the ConvQA task. The implementation of our model has been open-sourced to the research community.[1]

2 RELATED WORK

Our work is closely related to several research areas, including machine comprehension, conversational question answering, conversational search, and multi-task learning.

Machine Comprehension. Machine reading comprehension is one of the most popular tasks in natural language processing.

[1] https://github.com/prdwb/attentive_history_selection

Many high-quality challenges and datasets [12, 14, 18, 25, 26] have greatly boosted the research progress in this field, resulting in a wide range of model architectures [4, 9, 10, 28, 32]. One of the most influential datasets in this field is SQuAD (The Stanford Question Answering Dataset) [25, 26]. The reading comprehension task in SQuAD is conducted in a single-turn QA manner. The system is given a passage and a question. The goal is to answer the question by predicting an answer span in the passage. Extractive answers in this task enable easy and fair evaluations compared with other datasets that have abstractive answers generated by human. The recently proposed BERT [6] model pre-trains language representations with bidirectional encoder representations from transformers and achieves exceptional results on this task. BERT has been one of the most popular base models and testbeds for IR and NLP tasks including machine comprehension.

Conversational Question Answering. CoQA [27] and QuAC [2] are two large-scale ConvQA datasets. The ConvQA task in these datasets is very similar to the MC task in SQuAD. A major difference is that the questions in ConvQA are organized in conversations. Although both datasets feature ConvQA in context, they come 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. QuAC also comes with dialog acts, which is an essential component in this interactive information retrieval process. The dialog acts provide an opportunity to study the multi-task learning of answer span prediction and dialog act prediction. Overall, the information-seeking setting in QuAC is more in line with our interest since we are working towards the goal of conversational search. Thus, we focus on QuAC in this work. Although leaderboards of CoQA[2] and QuAC[3] show more than two dozen submissions, these models are mostly work done in parallel with ours and rarely have descriptions, papers, or codes.

Previous work [24] proposed a "history selection - history modeling" framework to handle conversation history in ConvQA. In terms of history selection, existing works[2, 11, 24, 27, 39] adopt a simple heuristic of selecting immediate previous turns. This heuristic, however, does not work for complicated dialog behaviors. There is no published work that focuses on *learning* to select or re-weight conversation history turns. To address this issue, we propose a history attention mechanism, which is a learned strategy to attend to history turns with different weights according to how helpful they are on answering the current question. In terms of history modeling, existing methods simply prepend history turns to the current question [27, 39] or use a recurrent structure to model the representations of history turns [11], which has a lower training efficiency [24]. Recently, a history answer embedding method [24] was proposed to learn two unique embeddings to denote whether a passage token is in history answers. However, this method fails to consider the position information of history turns. We propose to enhance this method by incorporating the position information into the history answer embeddings.

Conversational Search. Conversational search is an emerging topic in the IR community, however, the concept of it dates back to several early works [1, 5, 19]. Conversational search poses unique challenges as answers are retrieved in an iterative and interactive manner. Much effort is being made towards the goal of conversational search. The emerging of neural networks has made it possible to train conversation models in an end-to-end manner. Neural approaches are widely used in various conversation tasks, such as conversational recommendation [38], user intent prediction [23], next question prediction [34], and response ranking [8, 35]. In addition, researchers also conduct observational studies [3, 21, 22, 29, 30] to inform the design of conversational search systems. In this work, we focus on handling conversation history and using a multi-task learning setting to jointly learn dialog act prediction and answer span prediction. These are essential steps towards the goal of building functional conversational search systems.

Multi-task Learning. Multi-tasking learning has been a widely used technique to learn more powerful representations with deep neural networks [37]. A common paradigm is to employ separate task-specific layers on top of a shared encoder [16, 17, 33]. The encoder is able to learn representations that are more expressive, generic and transferable. Our model also adopts this paradigm. Not only can we enjoy the advantages of MTL, but also handle two essential tasks in ConvQA, answer span prediction and dialog act prediction, with a uniform model architecture.

3 OUR APPROACH

3.1 Task Definition

The ConvQA task is defined as follows [2, 27]. Given a passage p, the k-th question q_k in a conversation, 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 the passage p. The conversation history $\mathbf{H}_k$ contains $k-1$ turns, where the i-th turn $\mathbf{H}_k^i$ contains a question q_i and its groundtruth answer a_i. Formally, $\mathbf{H}_k = \{(q_i, a_i)\}_{i=1}^{k-1}$. One of the unique challenges of ConvQA is to leverage the conversation history to understand and answer the current question.

Additionally, an important task relevant to conversation modeling is dialog act prediction. QuAC [2] provides two dialog acts, namely, *affirmation* (Yes/No) and *continuation* (Follow up). The affirmation dialog act v^a consists of three possible labels: {`yes, no, neither`}. The continuation dialog act v^c also consists of three possible labels: {`follow up, maybe follow up, don't follow up`}. Each question is labeled with both dialog acts. The labels for each dialog act are mutually exclusive. This dialog act prediction task is essentially two sentence classification tasks. Therefore, a complete training instance is composed of the model input $(q_k, p, \mathbf{H}_k)$ and its ground truth labels (a_k, v_k^a, v_k^c), where a_k and v_k^a, v_k^c are labels for answer span prediction and dialog act prediction respectively.

3.2 Model Overview

In the following sections, we present our model that tackles the two tasks described in Section 3.1 together. A summary of key notations is presented in Table 2.

Our proposed model consists of four components: an encoder, a history attention module, an answer span predictor, and a dialog act predictor. The encoder is a BERT model that encodes the question q_k, the passage p, and conversation histories $\mathbf{H}_k$ into contextualized

[2] https://stanfordnlp.github.io/coqa/ [3] http://quac.ai/

Table 2: A summary of key notations used in this paper.

Notation	Description
q_k, p	The k-th (current) question in a dialog and the given passage
$\mathbf{H}_k, \mathbf{H}_k^i$	The conversation history for q_k and the i-th history turn
a_k, a_i	The ground truth answer for q_k and a history answer for q_i
v_k^a, v_k^c	The ground truth affirmation and continuation dialog acts for q_k
$\|V_a\|, \|V_c\|$	The number of classes for affirmation and continuation dialog acts
n	The number of "sub-passages" after applying a sliding window to p
V_{PosHAE}	The vocabulary for PosHAE
$\mathbf{ET}$	The embedding look up table for PosHAE
h	The hidden size for PosHAE, $\mathbf{B}$, $\mathbf{E}$, and $\mathbf{D}$
$\mathbf{T}_k^i, \mathcal{T}_k$	One and a batch of contextualized token-level representation(s)
$\mathbf{s}_k^i, \mathbf{S}_k$,	One and a batch of contextualized sequence-level representation(s)
I	The max # history turns, which is the first dimension for $\mathcal{T}_k$ and $\mathbf{S}_k$
$F(\cdot)$	The encoder is a transformation function that $\mathbf{T}_k^i, \mathbf{s}_k^i = F(q_k, p, \mathbf{H}_k^i)$
$\mathbf{D}$	The attention vector in the history attention module
$\mathbf{w}, w_i$	History attention weights and one of the weights
$\hat{\mathbf{T}}_k, \hat{\mathbf{s}}_k$	Aggregated token- and sequence-level representations for $\mathcal{T}_k$ and $\mathbf{S}_k$
$\mathbf{t}_k^i(m)$	The token representation for the m-th token in $\mathbf{T}_k^i$
$\mathbf{t}_k(m)$	All token representations in $\mathcal{T}_k$ for the m-th token
$\hat{\mathbf{t}}_k(m)$	The aggregated token rep computed by applying $\mathbf{w}$ to $\{\mathbf{t}_k^i(m)\}_{i=1}^I$
M	The sequence length, which means $\mathbf{T}_k^i$ consists of M tokens
$\mathbf{B}, \mathbf{E}$	The begin and end vectors in answer span prediction
p_m^B, p_m^E	The probabilities of the m-th token in $\hat{\mathbf{T}}_k$ being the begin/end tokens
$\mathcal{L}_B, \mathcal{L}_E$	The begin and end losses
$\mathbf{A}, \mathbf{C}$	Parameters for the affirmation and continuation dialog act predictions
$\mathcal{L}_A, \mathcal{L}_C$	Losses for two dialog act predictions
$\mathcal{L}_{ans}, \mathcal{L}$	The loss for answer span prediction and the total loss
λ, μ	Factors to combine $\mathcal{L}_{ans}$, $\mathcal{L}_A$, $\mathcal{L}_C$ to generate $\mathcal{L}$

representations. Then the history attention module learns to attend to history turns with different weights and computes aggregated representations for $(q_k, p, \mathbf{H}_k)$ on a token level and a sequence level. Finally, the two prediction modules make predictions based on the aggregated representations with a multi-task learning setting.

In our architecture, history modeling is enabled in the BERT encoder, where we model one history turn at a time. History selection is performed in the history attention module in the form of "soft selection". Figure 1 gives an overview of our model. We illustrate each component in detail in the following sections.

3.3 Encoder

3.3.1 ***BERT Encoder.*** The encoder is a BERT model that encodes the question q_k, the passage p, and conversation histories $\mathbf{H}_k$ into contextualized representations. BERT is a pre-trained language model that is designed to learn deep bidirectional representations using transformers [31]. Figure 2 gives an illustration of the encoder. It zooms in to the encoder component in Figure 1. It reveals the encoding process from an input sequence (the yellow-green row to the left of the encoder in Figure 1) to a contextualized representation (the pink-purple row to the right of the encoder in Figure 1).

Given a training instance $(q_k, p, \mathbf{H}_k)$, we first generate $k-1$ variations of this instance, where each variation contains the same question and passage, with only one turn of conversation history. Formally, the i-th variation is denoted as $(q_k, p, \mathbf{H}_k^i)$, where $\mathbf{H}_k^i = (q_i, a_i)$. We follow the previous work [6] and use a sliding window approach to split long passages, and thus construct multiple input sequences for a given instance variation. Suppose the passage is split into n pieces,[4] the training instance $(q_k, p, \mathbf{H}_k)$ would generate $n(k-1)$ input sequences. We take the $k-1$ input sequences corresponding to the first piece of the passage (still denoted as p here for simplicity) for illustration here. As shown in Figure 2, we pack the question q_k and the passage p into one sequence. The input sequences are fed into BERT and BERT generates contextualized token-level representations for each sequence based on the embeddings for tokens, segments, positions, and a special positional history answer embedding (PosHAE). PosHAE embeds the history answer a_i into the passage p since a_i is essentially a span of p. This technique enhances the previous work [24] by integrating history position signals. We describe this method in the next section.

The encoder can be formulated as a transformation function $F(\cdot)$ that takes in a training instance variation and produces a hidden representation for it on a token level, i.e., $\mathbf{T}_k^i = F(q_k, p, \mathbf{H}_k^i)$, where $\mathbf{T}_k^i \in \mathbb{R}^{M \times h}$ is the token-level representation for this instance variation. M is the sequence length, and h is the hidden size of the token representation. $\mathbf{T}_k^i$ can also be represented as $\{\mathbf{t}_k^i(m)\}_{m=1}^M$, where $\mathbf{t}_k^i(m) \in \mathbb{R}^h$ refers to the representation of the m-th token in $\mathbf{T}_k^i$. Instead of using separate encoders for questions, passages, and histories in previous work [11, 39], we take advantage of BERT and PosHAE to model these different input types jointly.

In addition, we also obtain a sequence-level representation $\mathbf{s}_k^i \in \mathbb{R}^h$ for each sequence. We take the representation of the `[CLS]` token, which is the first token of the sequence, and pass it through a fully-connected layer that has h hidden units [6]. That is, $\mathbf{s}_k^i = \tanh(\mathbf{t}_k^i(1) \cdot \mathbf{W}_{\text{CLS}})$, where $\mathbf{W}_{\text{CLS}} \in \mathbb{R}^{h \times h}$ is the weight matrix for this dense layer. The bias term in this equation and following equations are omitted for simplicity. This is a standard technique to obtain a sequence-level representation in BERT. It is essentially a pooling method to remove the dimension of sequence length. We also conduct experiments with average pooling and max pooling on this dimension to achieve the same purpose.

3.3.2 ***Positional History Answer Embedding.*** One of the key functions of the encoder is to model the given history turn along with the question and the passage. Previous work [24] introduces a history answer embedding (HAE) method to incorporate the conversation history into BERT in a natural way. They learn two unique history answer embeddings that denote whether a token is part of history answers or not. This method gives tokens extra embedding information and thus impacts the token-level contextual representations generated by BERT. However, this method fails to consider the position of a history utterance in the dialog. A commonly used history selection method is to select immediate previous turns. The intuition is that the utility of a history utterance could be related to its position. Therefore, we propose to consider the position information in HAE, resulting in a *positional history answer embedding* (PosHAE) method. The "position" refers to the relative position of a history turn in terms of the current question. Our method only considers history answers since previous works [2, 24] show that history questions contribute little to the performance.

Specifically, we first define a vocabulary of size $I+1$ for PosHAE, denoted as $V_{PosHAE} = \{0, 1, \ldots, I\}$, where I is the max number of history turns.[5] Given the current question q_k and a history turn H_k^i, we compute the relative position of H_k^i in terms of q_k as $k-i$. This relative position corresponds to a vocabulary ID in V_{PosHAE}. We use the vocabulary ID 0 for the tokens that are not in the given

[4] $n = 2$ in Figure 1

[5] In QuAC, $I = 11$, which means a dialog has at most 11 history turns.

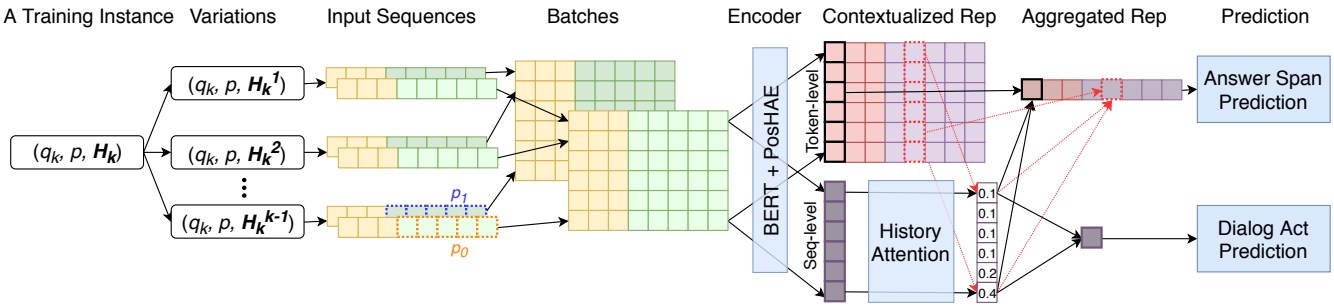


Figure 1: Our model consists of an encoder, a history attention module, an answer span predictor, and a dialog act predictor. Given a training instance, we first generate variations of this instance, where each variation contains the same question and passage, with only one turn of conversation history. We use a sliding window approach to split a long passage into "sub-passages" (p_0 and p_1) and use p_0 for illustration. The BERT encoder encodes the variations to contextualized representations on both token level and sequence level. The sequence-level representations are used to compute history attention weights. Alternatively, we propose a fine-grained history attention approach as marked in red-dotted lines. Finally, answer span prediction and dialog act predictions are conducted on the aggregated representations generated by the history attention module.

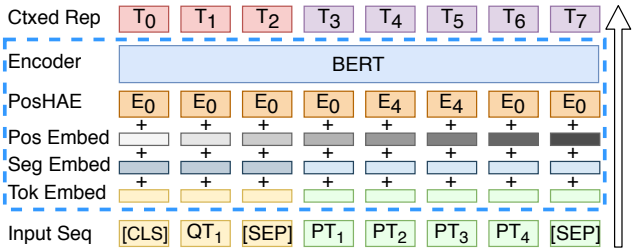


Figure 2: The encoder with PosHAE. It zooms in to the encoder in Fig. 1. It reveals the encoding process (marked by the blue-dotted lines) from an input sequence (the yellow-green row to the left of the encoder in Fig. 1) to contextualized representations (the pink-purple row to the right of the encoder in Fig. 1). QT_i/PT_i denote question/passage tokens. Suppose we are encoding (q_6, p, H_6^2), E_4 and E_0 are the history embeddings for tokens that are in and not in H_6^2).

history. We then use a truncated normal distribution to initialize an embedding look up table $\mathbf{ET} \in \mathbb{R}^{|I+1|\times h}$. We use V_{PosHAE} to map each token to a history answer embedding in **ET**. The history answer embeddings are learned. An example is illustrated in Figure 2. In addition to introducing conversation history, PosHAE enhances HAE by incorporating position information of history turns. This enables the ConvQA model to capture the spatial patterns of history answers in context.

3.4 History Attention Module

The core of the history attention module is a history attention mechanism (HAM). The inputs of this module are the token-level and sequence-level representations for all variations that are generated by the same training instance. The token-level representation is denoted as $\mathcal{T}_k = \{\mathbf{T}_k^i\}_{i=1}^I$, where $\mathcal{T}_k \in \mathbb{R}^{I\times M\times h}$. Similarly, the sequence-level representation is denoted as $\mathbf{S}_k = \{\mathbf{s}_k^i\}_{i=1}^I$, where $\mathbf{S}_k \in \mathbb{R}^{I\times h}$. The first dimension of $\mathcal{T}_k$ and $\mathbf{S}_k$ are both I because they are always padded to the max number of history turns. The padded parts are masked out. $\mathcal{T}_k$ and $\mathbf{S}_k$ are illustrated in Figure 1 as the "Token-level" and "Seq-level Contextualized Rep" respectively.

The history attention network is a single-layer feed-forward network. We learn an attention vector $\mathbf{D} \in \mathbb{R}^h$ to map a sentence representation $\mathbf{s}_k^i$ to a logit and use the softmax function to compute probabilities across all sequences generated by the same instance. Formally, the history attention weights are computed as follows.

$$w_i = \frac{e^{\mathbf{D}\cdot\mathbf{s}_k^i}}{\sum_{i'=1}^I e^{\mathbf{D}\cdot\mathbf{s}_k^{i'}}} \tag{1}$$

where w_i is the history attention weight for $\mathbf{s}_k^i$. Let $\mathbf{w} = \{w_i\}_{i=1}^I$. We compute aggregated representations for $\mathcal{T}_k$ and $\mathbf{S}_k$ with $\mathbf{w}$:

$$\hat{\mathbf{T}}_k = \sum_{i=1}^I \mathbf{T}_k^i \cdot w_i \quad , \qquad \hat{\mathbf{s}}_k = \sum_{i=1}^I \mathbf{s}_k^i \cdot w_i \tag{2}$$

where $\hat{\mathbf{T}}_k \in \mathbb{R}^{M\times h}$ and $\hat{\mathbf{s}}_k \in \mathbb{R}^h$ are aggregated token-level and sequence-level representations respectively. The attention weights $\{w_i\}_{i=1}^I$ are computed on a sequence-level and thus the tokens in the same sequence share the same weight. Intuitively, the history attention network attends to the variation representations with different weights and then each variation representation contributes to the aggregated representation according to the utility of the history turn in this variation.

Alternatively, we develop a *fine-grained history attention* approach to compute the attention weights. Instead of using sequence-level representations $\mathbf{S}_k$ as the input for the attention network, we use the token-level ones. The token-level attention input for the m-th token in the sequence is denoted as $\mathbf{t}_k(m) = \{\mathbf{t}_k^i(m)\}_{i=1}^I$, where $\mathbf{t}_k(m) \in \mathbb{R}^{I\times h}$. This is marked as a column with red-dotted lines in Figure 1. Then these attention weights are applied to $\mathbf{t}_k(m)$ itself:

$$\begin{aligned} w_i &= \frac{e^{\mathbf{D}\cdot\mathbf{t}_k^i(m)}}{\sum_{i'=1}^I e^{\mathbf{D}\cdot\mathbf{t}_k^{i'}(m)}} \\ \hat{\mathbf{t}}_k(m) &= \sum_{i=1}^I \mathbf{t}_k^i(m) \cdot w_i \end{aligned} \tag{3}$$

where $\hat{\mathbf{t}}_k(m) \in \mathbb{R}^h$ is the aggregated token representation for the m-th token in this sequence. Therefore, the aggregated token-level

representation $\hat{\mathbf{T}}_k$ for this sequence is $\{\hat{\mathbf{t}}_k(m)\}_{m=1}^{M}$. We show the process of computing the aggregated token representation for one token, but the actual process is vectorized and paralleled for all tokens in this sequence. Intuitively, this approach computes the attention weights given different token representations for the same token but embedded with different history information. These attention weights are on a token level and thus are more fine-grained than those from the sequence-level representations.

In both granularity levels of history attention, we show the process of computing attention weights for a single instance, but the actual process is vectorized for multiple instances. Also, if the given question does not have history turns (i.e., the first question of a conversation), it should bypass the history attention module. In practice, this is equivalent to pass it though the history attention network since all the attention weights will be applied to itself.

3.5 Answer Span Prediction

Given the aggregated token-level representation $\hat{\mathbf{T}}_k$ produced by the history attention network, we predict answer span by computing the probability of each token being the begin token and the end token. Specifically, we learn two sets of parameters, a begin vector and an end vector, to map a token representation to a logit. Then we use the softmax function to compute probabilities across all tokens in this sequence. Formally, let $\mathbf{B} \in \mathbb{R}^h$ and $\mathbf{E} \in \mathbb{R}^h$ be the begin vector and the end vector respectively. The probabilities of this token being the begin token p_m^B and end token p_m^E are:

$$p_m^B = \frac{e^{\mathbf{B}\cdot\hat{\mathbf{t}}_k(m)}}{\sum_{m'=1}^{M} e^{\mathbf{B}\cdot\hat{\mathbf{t}}_k(m')}} \quad , \qquad p_m^E = \frac{e^{\mathbf{E}\cdot\hat{\mathbf{t}}_k(m)}}{\sum_{m'=1}^{M} e^{\mathbf{E}\cdot\hat{\mathbf{t}}_k(m')}} \tag{4}$$

We then compute the cross-entropy loss for answer span prediction:

$$\mathcal{L}_B = -\sum_{M} \mathbb{1}\{m = m_B\} \log p_m^B \quad , \quad \mathcal{L}_E = -\sum_{M} \mathbb{1}\{m = m_E\} \log p_m^E$$
$$\mathcal{L}_{ans} = \frac{1}{2}(\mathcal{L}_B + \mathcal{L}_E) \tag{5}$$

where tokens at positions of m_B and m_E are the ground truth begin token and end token respectively, and $\mathbb{1}\{\cdot\}$ is an indicator function. $\mathcal{L}_B$ and $\mathcal{L}_E$ are the losses for the begin token and end token respectively and $\mathcal{L}_{ans}$ is the loss for answer span prediction. For unanswerable questions, a "CANNOTANSWER" token is appended to each passage in QuAC. The model learns to predict an answer span of this exact token if it believes the question is unanswerable.

Invalid predictions, including the cases where the predicted span overlaps with the question part of the sequence, or the end token comes before the begin token, are discarded at testing time.

3.6 Dialog Act Prediction

Given the aggregated sequence-level representation $\hat{\mathbf{s}}_k$ for a training instance, we learn two sets of parameters $\mathbf{A} \in \mathbb{R}^{|V_a| \times h}$ and $\mathbf{C} \in \mathbb{R}^{|V_c| \times h}$ to predict the dialog act of affirmation and continuation respectively, where $|V_a|$ and $|V_c|$ denote the number of classes.[6] Formally, the loss for dialog act prediction for affirmation is:

$$p(v|\hat{\mathbf{s}}_k) = \frac{e^{\mathbf{A}_v \cdot \hat{\mathbf{s}}_k}}{\sum_{v'=1}^{|V_a|} e^{\mathbf{A}_{v'} \cdot \hat{\mathbf{s}}_k}}$$
$$\mathcal{L}_A = -\sum_{v} \mathbb{1}\{v = v_k^a\} \log p(v|\hat{\mathbf{s}}_k) \tag{6}$$

[6] $|V_a| = 3$ and $|V_c| = 3$ in QuAC.

where $\mathbb{1}\{\cdot\}$ is an indicator function to show whether the predicted label v is the ground truth label v_k^a, and $\mathbf{A}_v \in \mathbb{R}^h$ is the vector in $\mathbf{A}$ corresponding to v. The loss $\mathcal{L}_C$ for predicting the continuation dialog act v_k^c is computed in the same way. We make dialog act predictions independently based on the information of each single training instance $(q_k, p, \mathbf{H}_k)$. We do not model history dialog acts in the encoder for this task.

3.7 Model Training

3.7.1 ***Batching***. We implement an *instance-aware batching* approach to construct the batches for BERT. This method guarantees that the variations generated by the same training instance are always included in the same batch, so that the history attention module operates on all available histories. In practice, a passage in a training instance can produce multiple "sub-passages" (e.g., p_0 and p_1 in Figure 1) after applying the sliding window approach [6]. This results in multiple "sub-instances" (e.g. $(q_k, p_0, \mathbf{H}_k^i)$ and $(q_k, p_1, \mathbf{H}_k^i)$), which are modeled separately and potentially in different batches. This is because the "sub-passages" have overlaps to make sure that every passage token has sufficient context so that they can be considered as different passages.

3.7.2 ***Training Loss and Multi-task Learning***. We adopt the multi-task learning idea to jointly learn the answer span prediction task and the dialog act prediction task. All parameters are learned in an end-to-end manner. We use hyper-parameters λ and μ to combine the losses for different tasks. That is,

$$\mathcal{L} = \mu\mathcal{L}_{ans} + \lambda\mathcal{L}_A + \lambda\mathcal{L}_C \tag{7}$$

where $\mathcal{L}$ is the total training loss.

Multi-task learning has been shown to be effective for representation learning [16, 17, 33]. There are two reasons behind this. 1) Our two tasks provide more supervising signals to fine-tune the encoder. 2) Representation learning benefits from a regularization effect by optimizing for multiple tasks. Although BERT serves as a universal encoder by pre-training with a large amount of unlabeled data, MTL is a complementing technology [17] that makes such representations more generic and transferable. More importantly, we can handle two essential tasks in ConvQA, answer span prediction and dialog act prediction, with a uniform model architecture.

4 EXPERIMENTS

4.1 Data Description

We experiment with the QuAC (Question Answering in Context) dataset [2]. It is a large-scale dataset designed for modeling and understanding information-seeking conversations. It contains interactive dialogs between an information-seeker and an information-provider. The information-seeker tries to learn about a *hidden* Wikipedia passage by asking a sequence of freeform questions. She/he only has access to the heading of the passage, simulating an information need. The information-provider answers each question by providing a short span of the given passage. One of the unique properties that distinguish QuAC from other dialog data is that it comes with dialog acts. The information-provider uses dialog acts to provide the seeker with feedback (e.g., "ask a follow up question"), which makes the dialogs more productive [2]. This

dataset poses unique challenges because its questions are more open-ended, unanswerable, or only meaningful within the dialog context. More importantly, many questions have coreferences and interactions with conversation history, making this dataset suitable for our task. We present some statistics of the dataset in Table 3.

Table 3: Data Statistics. We can only access the training and validation data.

Items	Train	Validation
# Dialogs	11,567	1,000
# Questions	83,568	7,354
# Average Tokens Per Passage	396.8	440.0
# Average Tokens Per Question	6.5	6.5
# Average Tokens Per Answer	15.1	12.3
# Average Questions Per Dialog	7.2	7.4
# Min/Avg/Med/Max History Turns Per Question	0/3.4/3/11	0/3.5/3/11

4.2 Experimental Setup

4.2.1 ***Competing Methods.*** We consider all methods with published papers on the QuAC leaderboard as baselines.[7] In addition, we also include a "BERT + PosHAE" model that replaces HAE in Qu et al. [24] with PosHAE to demonstrate the impact of the PosHAE. To be specific, the competing methods are:

- **BiDAF++** [2, 20]: BiDAF [28] is a top-performing SQuAD model. It uses bi-directional attention flow mechanism to obtain a query-aware context representation. BiDAF++ makes further augmentations with self-attention [4] and contextualized embeddings.
- **BiDAF++ w/ 2-Context** [2]: This model incorporates conversation history by modifying the passage and question embedding processes. Specifically, it encodes the dialog turn number with the question embedding and concatenates answer marker embeddings to the word embedding.
- **FlowQA** [11]: This model incorporates conversation history by integrating intermediate representation generated when answering the previous question. Thus it is able to grasp the latent semantics of the conversation history compared to shallow approaches that concatenate history turns.
- **BERT + HAE** [24]: This model is adapted from the SQuAD model in the BERT paper.[8] It uses history answer embedding to enable a seamless integration of conversation history into BERT.
- **BERT + PosHAE**: We enhance the BERT + HAE model with the PosHAE that we proposed. This method considers the position information of history turns and serves as a stronger baseline. We set the max number of history turns as 6 since it gives the best performance under this setting.
- **HAM** (History Attention Mechanism): This is the solution we proposed in Section 3. It employs PosHAE for history modeling, the history attention mechanism for history selection, and the MTL scheme to optimize for both answer span prediction and dialog act prediction tasks. We use the fine-grained history attention in Equation 3. We use "HAM" as the model name since the attentive history selection is the most important and effective component that essentially defines the model architecture.
- **HAM (BERT-Large)**: Due to the competing nature of the QuAC challenge, we apply BERT-Large to HAM for a more informative evaluation. This is more resource intensive. Other HAM models in this paper are constructed with BERT-Base for two reasons: 1) To alleviate the memory and training efficiency issues caused by BERT-Large and thus speed up the experiments for the research purpose. 2) To keep the settings consistent with existing and published work [24] for fair and easy comparison.

4.2.2 ***Evaluation Metrics.*** The QuAC challenge provides two evaluation metrics, the word-level F1, and the human equivalence score (HEQ) [2]. The word-level F1 evaluates the overlap of the prediction and the ground truth answer span. It is a classic metric used in MC and ConvQA tasks [2, 25, 27]. HEQ measures the percentage of examples for which system F1 exceeds or matches human F1. Intuitively, this metric judges whether a system can provide answers as good as an average human. This metric is computed on the question level (HEQ-Q) and the dialog level (HEQ-D). In addition, the dialog act prediction task is evaluated by accuracy.

4.2.3 ***Hyper-parameter Settings and Implementation Details.*** Models are implemented with TensorFlow[9]. The version of the QuAC data we use is v0.2. We use the BERT-Base Uncased model[10] with the max sequence length set to 384. The batch size is set to 24. We train the ConvQA model with a Adam weight decay optimizer with an initial learning rate of 3e-5. The warming up portion for learning rate is 10%. We set the stride in the sliding window for passages to 128, the max question length to 64, and the max answer length to 40. The total training steps is set to 30,000. Experiments are conducted on a single NVIDIA TESLA M40 GPU. λ and μ for multi-task learning is set to 0.1 and 0.8 respectively for HAM.

4.3 Main Evaluation Results

We report the results on the validation and test sets in Table 4. Our best model was evaluated officially by the QuAC challenge and the result is displayed on the leaderboard[11] with proper anonymization. Since dialog act prediction is not the main task of this dataset, most of the baseline methods do not perform this task.

We summarize our observations of the results as follows.

(1) BERT + PosHAE brings a significant improvement compared with BERT + HAE, achieving the best results among baselines. This suggests that the position information plays an important role in conversation history modeling with history answer embedding. In addition, previous work reported that BERT + HAE enjoys a much better training efficiency compared to FlowQA but suffers from a poorer performance. However, after enhancing HAE with the history position information, it manages to achieve a slightly higher performance than FlowQA when maintaining the efficiency advantage. This shows the effectiveness of this conceptually simple idea of modeling conversation history in BERT with PosHAE.

(2) Our model HAM obtains statistically significant improvements over the strongest baseline (BERT + PosHAE) with $p < 0.05$

[7] The methods without published papers or descriptions are essentially done in parallel with ours and may not be suitable for comparison since their model details are unknown. Besides, these work could be using generic performance boosters, such as BERT-large, data augmentation, transfer learning, or better training infrastructures.

[8] We notice the hyper-parameter of "max answer length" is set to 30 in BERT + HAE [24], which is sub-optimal. We set it to 40 to be consistent with our settings and updated their validation results.

[9] https://www.tensorflow.org/

[10] https://github.com/google-research/bert

[11] http://quac.ai/

Table 4: Evaluation results on QuAC. Models in a bold font are our implementations. Each cell displays val/test scores. Val result of BiDAF++, FlowQA are from [2], [11]. Test results are from the QuAC leaderboard at the time of the CIKM deadline. ‡ means statistically significant improvement over the strongest baseline with $p < 0.05$ tested by the Student's paired t-test. We can only do significance test on F1 on the validation set. "–" means a result is not available and "N/A" means a result is not applicable for this model.

Models	F1	HEQ-Q	HEQ-D	Yes/No	Follow up
BiDAF++	51.8 / 50.2	45.3 / 43.3	2.0 / 2.2	86.4 / 85.4	59.7 / 59.0
BiDAF++ w/ 2-C	60.6 / 60.1	55.7 / 54.8	5.3 / 4.0	86.6 / 85.7	61.6 / 61.3
BERT + HAE	63.9 / 62.4	59.7 / 57.8	5.9 / 5.1	N/A	N/A
FlowQA	64.6 / 64.1	– / 59.6	– / 5.8	N/A	N/A
BERT + PosHAE	64.7 / –	60.7 / –	6.0 / –	N/A	N/A
HAM	65.7‡ / 64.4	62.1 / 60.2	7.3 / 6.1	**88.3** / **88.4**	62.3 / **61.7**
HAM (BERT-Large)	**66.7**‡ / **65.4**	**63.3** / **61.8**	**9.5** / **6.7**	88.2 / 88.2	**62.4** / 61.0

tested by the Student's paired t-test. These results demonstrate the effectiveness of our method.

(3) Our model HAM also achieves a substantially higher performance on dialog act prediction compared to baseline methods, showing the strength of our model on both tasks. We can only do significance test on F1. We are unable to do a significance test on dialog act prediction because the prediction results of BiDAF++ is not available. In addition, the sequence-level representations of HAM are obtained with max pooling. We see no major differences when using different pooling methods.

(4) Applying BERT-Large to HAM brings a substantial improvement to answer span prediction, suggesting that a more powerful encoder can boost the performance.

4.4 Ablation Analysis

Section 4.3 shows the effectiveness of our model. This performance is closely related to several design choices. So we conduct an ablation analysis to investigate the contributions of each design choice by removing or replacing the corresponding component in the complete HAM model. Specifically, we have four settings as follows.

- **HAM w/o Fine-grained (F-g) History attention**. We use the sequence-level history attention (Equation 1 and 2) instead of the fine-grained history attention (Equation 3).
- **HAM w/o History Attention**. We do not learn any form of history attention. Instead, we modify the history attention module and make it always produce equal weights. Note that this is not equivalent to "BERT + PosHAE". "BERT + PosHAE" incorporates the selected history turns in a single input sequence and relies on the encoder to work out the importance of these history turns. The architecture we illustrated in Figure 1 models each history turn separately and capture their importance by the history attention mechanism explicitly, which is a more direct and explainable way. Therefore, even when we disable the history attention module, it is not equivalent to "BERT + PosHAE".
- **HAM w/o PosHAE**. We use HAE [24] instead of the PosHAE we proposed in Section 3.3.2.
- **HAM w/o MTL**. Our multi-task learning scheme consists of two tasks, an answer span prediction task and a dialog act prediction task. Therefore, to evaluate the contribution of MTL, we further design two settings: (1) In **HAM w/o Dialog Act Prediction**, we set $\mu = 1$ and $\lambda = 0$ in Equation 7 to block the parameter updates from dialog act prediction. (2) In **HAM w/o Answer Span Prediction**, we set $\mu = 0$ in Equation 7 and thus block the updates caused by answer span prediction. We tune λ in (0.2, 0.4, 0.6, 0.8) in Equation 7 and try different pooling methods to obtain the sequence-level representations. We finally adopt $\lambda = 0.2$ and average pooling since they give the best performance. We consider these two ablation settings to fully control the factors in our experiments and thus precisely capture the differences in the representation learning caused by different tasks.

The ablation results on the validation set are presented in Table 6. The following are our observations.

(1) By replacing the fine-grained history attention with sequence-level history attention, we observe a performance drop. This shows the effectiveness of computing history attention weights on a token level. This is intuitive because these weights are specifically tailored for the given token and thus can better capture the history information embedded in the token representations.

(2) When we disable the history attention module, we notice the performance drops dramatically for 4.6% and 3.8% compared with HAM and "HAM w/o F-g History Attention" respectively. This indicates that the history attention mechanism, regardless of granularity, can attend to conversation histories according to their importance. Disabling history attention also hurts the performance for dialog act prediction.

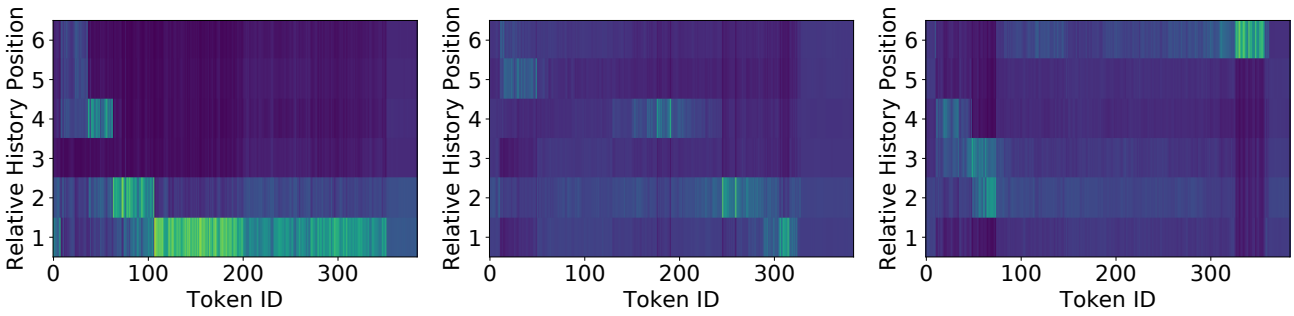
(3) Replacing PosHAE with HAE also witnesses a major drop in model performance. This again shows the importance of history position information in modeling conversation history.

(4) When we remove the dialog act prediction task, we observe that the performance for answer span prediction has a slight and insignificant increase. This suggests that dialog act prediction does not contribute to the representation learning for answer span prediction. Since dialog act prediction is a secondary task in our setting, its loss is scaled down and thus could have a limited impact on the optimization for the encoder. Although the performance for our main model is slightly lower on answer span prediction, it can handle both answer span prediction and dialog prediction tasks in a uniform way.

(5) On the contrary, when we remove the answer span prediction task, we observe a relatively large performance drop for dialog act prediction. This indicates that the additional supervising signals from answer span prediction can indeed help the encoder to produce a more generic representation that benefits the dialog act prediction task. In addition, the encoder could also benefit from a regularization effect because it is optimized for two different tasks and thus alleviates overfitting. Although the multi-task learning scheme does not contribute to answer span prediction, we show that it is beneficial to dialog act prediction.

4.5 Case Study and Attention Visualization

One of the major advantages of our model is its explainability of history attention. In this section, we present a case study that visualizes the history attention weights predicted by our model.



(a) Drill down (b) Topic shift (c) Topic return

Figure 3: Attention visualization for different dialog behaviors. Brighter spots mean higher attention weights. Token ID refers to the token position in an input sequence. A sequence contains 384 tokens. Relative history position refers to the difference of the current turn # with a history turn #. The selected examples are all in the 7th turn. These figures are best viewed in color.

Table 5: QuAC dialogs that correspond to the dialog behaviors in Fig. 3. The examples are all in the 7th turn. "#" refers to the relative history position, which means "0" is the current turn and "6" is the most remote turn from the current turn. Each turn has a question and an answer, with the answer in italic. Co-references and related terms are marked in the same color.

(a) Drill down

#	Utterance
6	When did Ride leave NASA? *In 1987, Ride left ... to work at the Stanford ...*
5	What did she do at the Stanford Center? *International Security and Arms Control.*
4	How long was she there? *In 1989, she became a professor of physics at ...*
3	Was she successful as a professor? *CANNOTANSWER*
2	Did she have any other professions? *Ride led two public-outreach programs for NASA ...*
1	What was involved in the programs? *The programs allowed middle school students to ...*
0	What did she do after this? *To be predicted ...*

(b) Topic shift

#	Utterance
6	When did the Greatest Hits come out *beginning of 2004*
5	What songs were on the album *cover of Nick Kamen's "I Promised Myself" ...*
4	Was the album popular *The single became another top-two hit for the band ...*
3	Did it win any awards *CANNOTANSWER*
2	Why did they release this *... was just released in selected European countries ...*
1	Did they tour with this album? *the band finished their tour*
0	Are there other interesting aspects about this article? *To be predicted ...*

(c) Topic return

#	Utterance
6	What is relevant about Lorrie's musical career? *... she signed with RCA Records ... her first album ...*
5	What songs are included in the album? *CANNOTANSWER*
4	Are there any other interesting aspects about this article? *made her first appearance on the Grand Ole Opry at age 13,*
3	What did she do after her first appearance? *... she took over ... and began leading the group ...*
2	What important work did she do with the band? *leading the group through various club gigs.*
1	What songs did she played with the group? *CANNOTANSWER*
0	What are other interesting aspects of her musical career? *To be predicted ...*

Table 6: Results for ablation analysis. These results are obtained on the validation set since the test set is hidden for official evaluation only. "w/o" means to remove or replace the corresponding component. † means statistically significant performance *decrease* compared to the complete HAM model with $p < 0.05$ tested by the Student's paired t-test. We can only do significance test on F1 and dialog act accuracy.

Models	F1	HEQ-Q	HEQ-D	Yes/No	Follow up
HAM	65.7	62.1	7.3	88.3	62.3
w/o F-g History Attention	64.9†	61.0	7.1	88.4	62.1
w/o History Attention	61.1†	57.2	6.4	87.9	60.5†
w/o PosHAE	64.2†	60.0	7.3	88.6	62.1
w/o Dialog Act Prediction	65.9	62.2	8.2	N/A	N/A
w/o Answer Span Prediction	N/A	N/A	N/A	86.2†	59.7†

Qu et al. [21] observed that *follow up questions* is one of the most important user intents in information-seeking conversations. Yatskar [36] further described three history-related dialog behaviors that can be considered as a fine-grained taxonomy of follow up questions. We use these definitions to interpret the attention weights. These dialog behaviors are as follow.

- **Drill down**: the current question is a request for more information about a topic being discussed.
- **Topic shift**: the current question is not immediately relevant to something previously discussed.
- **Topic return**: the current question is asking about a topic again after it had previously been shifted away from.

We keep records of the attention weights generated at testing time on the validation data. We use a sliding window approach to split long passages as mentioned in Section 3.3.1. However, we specifically choose short passages that can be put in a single input sequence for easier visualization. The attention weights obtained from our fine-grained history attention model are visualized in Figure 3 and the corresponding dialogs are presented in Table 5.

Our history attention weights are computed on the token level. We observe that salient tokens are typically in the corresponding history answer in the passage. This suggests that our model learns to attend to tokens that carry history information. These tokens also bring some attention weights to other tokens that are not in the

history answer since the token representations are contextualized. Although each history turn has an answer, the weights vary to reflect the importance of the history information.

We further interpret the attention weights with examples for different dialog behaviors. First, Table 5a shows that the current question is drilling down on more relevant information on the topic being discussed. In this case, the current question is closely related to its immediate previous turns. We observe in Figure 3a that our model can attend to these turns properly with greater weights assigned to the most immediate previous turn. Second, in the topic shift scenario presented in Table 5b and Figure 3b, the current question is not immediately relevant to its preceding history turns. Therefore, the attention weights are distributed relatively evenly across history turns. Third, as shown in Table 5c and Figure 3c, the first turn talks about the topic of musical career while the following turns shift away from this topic. The information-seeker returns to musical career in the current turn. In this case, the most important history turn to consider is the most remote one from the current question. Our model learns to attend to certain tokens the first turn with larger weights, suggesting that the model could capture the topic return phenomenon. Moreover, we observe that the model does not attend to the passage token of "CANNOTANSWER", further indicating that it can identify useful history answers.

5 CONCLUSIONS AND FUTURE WORK

In this work, we propose a novel model for ConvQA. We introduce a history attention mechanism to conduct a "soft selection" for conversation histories. We show that our model can capture the utility of history turns. In addition, we enhance the history answer embedding method by incorporating the position information for history turns. We show that history position information plays an important role in conversation history modeling. Finally, we propose to jointly learn answer span prediction and dialog act prediction with a uniform model architecture in a multi-task learning setting. We conduct extensive experimental evaluations to demonstrate the effectiveness of our model. For future work, we would like to consider to apply our history attention method to other conversational retrieval tasks. In addition, we will further analyze the relationship between attention patterns and different user intents or dialog acts.

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] 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.
[3] A. Chuklin, A. Severyn, J. R. Trippas, E. Alfonseca, H. Silén, and D. Spina. Prosody Modifications for Question-Answering in Voice-Only Settings. *CoRR*, 2018.
[4] C. Clark and M. Gardner. Simple and Effective Multi-Paragraph Reading Comprehension. In *ACL*, 2018.
[5] W. B. Croft and R. H. Thompson. I3R: A new approach to the design of document retrieval systems. *JASIS*, 38:389–404, 1987.
[6] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *CoRR*, 2018.
[7] J. Gao, M. Galley, and L. Li. Neural Approaches to Conversational AI. In *SIGIR*, 2018.
[8] J. Guo, Y. Fan, L. Pang, L. Yang, Q. Ai, H. Zamani, C. Wu, W. B. Croft, and X. Cheng. A Deep Look into Neural Ranking Models for Information Retrieval. *CoRR*, abs/1903.06902, 2019.
[9] M. Hu, Y. Peng, Z. Huang, X. Qiu, F. Wei, and M. Zhou. Reinforced Mnemonic Reader for Machine Reading Comprehension. In *IJCAI*, 2018.
[10] H.-Y. Huang, C. Zhu, Y. Shen, and W. Chen. FusionNet: Fusing via Fully-Aware Attention with Application to Machine Comprehension. *CoRR*, abs/1711.07341, 2017.
[11] H.-Y. Huang, E. Choi, and W. Yih. FlowQA: Grasping Flow in History for Conversational Machine Comprehension. *CoRR*, 2018.
[12] M. S. Joshi, E. Choi, D. S. Weld, and L. S. Zettlemoyer. TriviaQA: A Large Scale Distantly Supervised Challenge Dataset for Reading Comprehension. In *ACL*, 2017.
[13] A. Kotov and C. Zhai. Towards natural question guided search. In *WWW*, 2010.
[14] T. Kwiatkowski, J. Palomaki, O. Redfield, M. Collins, A. Parikh, C. Alberti, D. Epstein, I. Polosukhin, M. Kelcey, J. Devlin, K. Lee, K. N. Toutanova, L. Jones, M.-W. Chang, A. Dai, J. Uszkoreit, Q. Le, and S. Petrov. Natural Questions: a Benchmark for Question Answering Research. *Transactions of the Association of Computational Linguistics*, 2019.
[15] F.-L. Li, M. Qiu, H. Chen, X. Wang, X. Gao, J. Huang, J. Ren, Z. Zhao, W. Zhao, L. Wang, G. Jin, and W. Chu. AliMe Assist : An Intelligent Assistant for Creating an Innovative E-commerce Experience. In *CIKM*, 2017.
[16] X. Liu, J. Gao, X. He, L. Deng, K. Duh, and Y.-Y. Wang. Representation Learning Using Multi-Task Deep Neural Networks for Semantic Classification and Information Retrieval. In *HLT-NAACL*, 2015.
[17] X. Liu, P. He, W. Chen, and J. Gao. Multi-Task Deep Neural Networks for Natural Language Understanding. *CoRR*, abs/1901.11504, 2019.
[18] 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.
[19] R. N. Oddy. *Information Retrieval through Man-Machine Dialogue.* 1977.
[20] M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. S. Zettlemoyer. Deep contextualized word representations. In *NAACL-HLT*, 2018.
[21] 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.
[22] C. Qu, L. Yang, W. B. Croft, F. Scholer, and Y. Zhang. Answer Interaction in Non-factoid Question Answering Systems. In *CHIIR*, 2019.
[23] C. Qu, L. Yang, W. B. Croft, Y. Zhang, J. R. Trippas, and M. Qiu. User Intent Prediction in Information-seeking Conversations. In *CHIIR*, 2019.
[24] C. Qu, L. Yang, M. Qiu, W. B. Croft, Y. Zhang, and M. Iyyer. BERT with History Answer Embedding for Conversational Question Answering. *CoRR*, abs/1905.05412, 2019.
[25] P. Rajpurkar, J. Zhang, K. Lopyrev, and P. Liang. SQuAD: 100, 000+ Questions for Machine Comprehension of Text. In *EMNLP*, 2016.
[26] P. Rajpurkar, R. Jia, and P. Liang. Know What You Don't Know: Unanswerable Questions for SQuAD. In *ACL*, 2018.
[27] S. Reddy, D. Chen, and C. D. Manning. CoQA: A Conversational Question Answering Challenge. *CoRR*, abs/1808.07042, 2018.
[28] M. J. Seo, A. Kembhavi, A. Farhadi, and H. Hajishirzi. Bidirectional Attention Flow for Machine Comprehension. *CoRR*, abs/1611.01603, 2016.
[29] P. Thomas, D. McDuff, M. Czerwinski, and N. Craswell. MISC: A data set of information-seeking conversations. In *SIGIR (CAIR'17)*, 2017.
[30] J. R. Trippas, D. Spina, L. Cavedon, H. Joho, and M. Sanderson. Informing the Design of Spoken Conversational Search: Perspective Paper. In *CHIIR*, 2018.
[31] 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.
[32] W. Wang, N. Yang, F. Wei, B. Chang, and M. Zhou. Gated Self-Matching Networks for Reading Comprehension and Question Answering. In *ACL*, 2017.
[33] Y. Xu, X. Liu, Y. Shen, J. Liu, and J. Gao. Multi-Task Learning for Machine Reading Comprehension. *CoRR*, abs/1809.06963, 2018.
[34] 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.
[35] 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.
[36] M. Yatskar. A Qualitative Comparison of CoQA, SQuAD 2.0 and QuAC. *CoRR*, abs/1809.10735, 2018.
[37] Y. Zhang and Q. Yang. A Survey on MultiTask Learning. 2018.
[38] Y. Zhang, X. Chen, Q. Ai, L. Yang, and W. B. Croft. Towards Conversational Search and Recommendation: System Ask, User Respond. In *CIKM*, 2018.
[39] C. Zhu, M. Zeng, and X. Huang. SDNet: Contextualized Attention-based Deep Network for Conversational Question Answering. *CoRR*, 2018.