CAT-BERT: A Context-Aware Transferable BERT Model for Multi-turn Machine Reading Comprehension

Cen Chen[1], Xinjing Huang[2], Feng Ji[3], Chengyu Wang[3], Minghui Qiu[3], Jun Huang[3], and Yin Zhang[2(✉)]

[1] Ant Group, Hangzhou, China
chencen.cc@antfin.com
[2] Zhejiang University, Hangzhou, China
{huangxinjing,zhangyin98}@zju.edu.cn
[3] Alibaba Group, Hangzhou, China
{zhongxiu.jf,chengyu.wcy,minghui.qmh,huangjun.hj}@alibaba-inc.com

Abstract. Machine Reading Comprehension (MRC) is an important NLP task with the goal of extracting answers to user questions from background passages. For conversational applications, modeling the contexts under the multi-turn setting is highly necessary for MRC, which has drawn great attention recently. Past studies on multi-turn MRC usually focus on a single domain, ignoring the fact that knowledge in different MRC tasks are transferable. To address this issue, we present a unified framework to model both single-turn and multi-turn MRC tasks which allows knowledge sharing from different source MRC tasks to help solve the target MRC task. Specifically, the Context-Aware Transferable Bidirectional Encoder Representations from Transformers (CAT-BERT) model is proposed, which jointly learns to solve both single-turn and multi-turn MRC tasks in a single pre-trained language model. In this model, both history questions and answers are encoded into the contexts for the multi-turn setting. To capture the task-level importance of different layer outputs, a task-specific attention layer is further added to the CAT-BERT outputs, reflecting the positions that the model should pay attention to for a specific MRC task. Extensive experimental results and ablation studies show that CAT-BERT achieves competitive results in multi-turn MRC tasks, outperforming strong baselines.

Keywords: Machine reading comprehension · Question answering · Transfer learning · Pre-trained language model

1 Introduction

Conversational search [22,31], a way of seeking information through conversations, has become a heated topic in the filed of Information Retrieval (IR). The

C. Chen and X. Huang—Equal contribution.

C. S. Jensen et al. (Eds.): DASFAA 2021, LNCS 12682, pp. 152–167, 2021.
https://doi.org/10.1007/978-3-030-73197-7_10

core task of conversational search is to answer user questions in a multi-turn scenario. In the literature, such task can modeled as multi-turn Machine Reading Comprehension (MRC) [27], whose goal is to answer user questions based on a given passage, by means of multi-turn interactions between machines and users.

According to previous research, there are mainly two challenges faced by multi-turn MRC [27]. i) Some questions that users raise in the dialogue are unanswerable as the questions may belong to a wrong topic, or the information from which the answers can the extracted is missing in the passage. Thus, the answers to this type of questions can be categorized as "CANNOT ANSWER". ii) As a question raised by users usually depends on previous answers sent by machines (i.e., chatbots), modeling the dialogue history is important and challenging for answering the question in the multi-turn setting. Therefore, many phenomena may occur, such as co-references and omissions. For example, to answer the question "What happened to him?", where "him" is covered in the previous answer "Mr. David found the dog was lost and became very sad at that moment", one has to know that "he" refers to "Mr. David".

To address the above-mentioned challenges, recent studies consider incorporating contextual information into MRC models. Typical methods include prepending previous questions and answers [32], adding history answer markers to the passage [16], or using attention mechanisms to select the dialogue history [17]. There are also studies applying context-aware neural networks such as recurrent neural networks and graph neural networks to convey the information in past turns [2,6,11,28]. However, these methods often ignore the fact that knowledge in many kinds of MRC tasks are transferable. To be more specific. both multi-turn and single-turn MRC tasks share some commonalities, such as unanswerable question recognition and knowledge reasoning. The knowledge learned from one MRC task may benefit the learning of other MRC tasks, especially when the tasks are closely related. Hence, it is crucial to leverage *transfer learning* to capture the shared knowledge from different multi-turn and single-turn MRC tasks for mutual reinforcement of the model performance.

To better leverage the cross-domain, cross-task knowledge, we present a unified framework to solve both single-turn and multi-turn MRC tasks, named Context-Aware Transferable Bidirectional Enoceder Representations from Transformers (CAT-BERT). The overview CAT-BERT framework is shown in Fig. 1. Inspired by the recent success of pre-trained language models, we extend Bidirectional Encoder Representations from Transformers (BERT) [4] to consider both history questions and answers to the model the contextual information. Thus, the learned text representations are more robust across different MRC tasks. Observing the fact that different MRC tasks may possess some unique task-dependent attributes [9], we further augment our model with a task-specific attention layer to capture the task-level importance of different layer outputs.

To the best of our knowledge, our study is the first to present a unified framework for both multi-turn and single-turn MRC tasks. Our framework can also be easily combined with other tasks by multi-task learning.

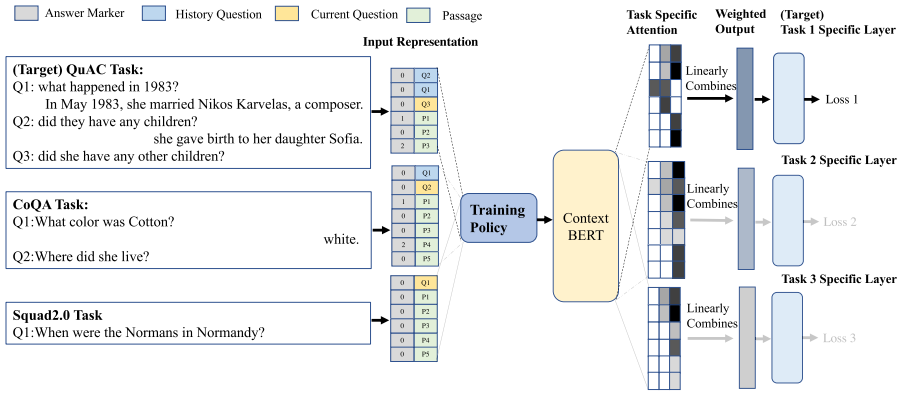

Fig. 1. An overview of the proposed Context-Aware Transferable BERT framework, which unifies three tasks, i.e., two multi-turn MRC tasks (QuAC [3] and CoQA [19]), and one single-turn MRC task (SQuAD 2.0 [18]). In the middle part, the context-aware BERT backbone is employed as the shared encoder, where the index in the input representation is the history answer index. The training policy selects the training data to feed into the context-aware BERT backbone, and then pass the data to task-specific attention and output layers to generate task-specific outputs.

We need to further claim that although multi-task learning has been recently studied for MRC (e.g., MultiQA [21], MT-DNN [10], MT-SAN [25]), CAT-BERT differs from these approaches in the following two perspectives. i) We focus on multi-turn MRC and propose a unified framework that can bridge the gaps between multi-turn and single-turn MRC tasks. ii) We seek to boost the performance of the MRC task in the target domain and better capture the transferable knowledge from other domains by considering task-specific attention.

To summarize, the contributions of this work are three-fold:

- We are the first to propose a unified framework named CAT-BERT for jointly learning multi-turn and single-turn MRC tasks. This sheds the light on how to leverage knowledge from large-scale single-turn MRC datasets to boost the performance of models for multi-turn MRC tasks.
- We propose a task-specific attention mechanism to model the task dependencies on each layer of CAT-BERT. Qualitative experiments show the attention weights learned are insightful and intuitive.
- Our method achieves competitive results in the QuAC leaderboard - a large-scale multi-turn MRC benchmark dataset. Extensive experiments demonstrate our method is effective. The model ablation studies show the importance of different integral parts of our model.

The reminder of this paper is summarized as follows. Section 2 briefly introduces the related work. The techniques of the CAT-BERT model is elaborated in Sect. 3. Experimental results are reported in Sect. 4. Finally, we draw the conclusion and discuss the future work in Sect. 5.

2 Related Work

In this section, we present a brief summarization on the related work of CAT-BERT, including the MRC task and transfer learning.

2.1 Machine Reading Comprehension

Our work is closely related to the MRC task. Unlike the typical question answering task [1,23,24], MRC [29] is a task to understand a given passage and use the passage to answer user questions. Different from single-turn MRC, we specifically focus on the multi-turn setting, where the user and the system interacts multiple times. The main challenging for multi-turn MRC is modeling the rich context of the multi-turns of human-machine interaction. In the literature, SDNet [32] takes the contexts into consideration by appending the history questions and answers to the inputs. HAE [16] adopts the marker to indicate the positions of history answers in the passage. HAM [17] further employs attention mechanisms to select the related history questions. However, these methods may fail when the context dependencies are more complicated.

There are also studies trying to model the contextual information using neural networks such as Recurrent Neural Networks (RNNs) and Graph Neural Networks (GNNs). For example, GraphFlow [2] views the relations between context words in each turn as a graph, and applies GNN to capture the information flow. FlowQA [6] employs RNNs to convey word representations of past turns and incorporates them with the current turn's representations. FlowDelta [28] further extends the FlowQA model to explicitly model the information gains by delta operations. MC^2 [30] adopts convolution neural networks to better capture the flow information in a more fine-grained manner with three perspectives. We notice that these studies only focus on one single domain for the MRC task, while we unify single-turn and multi-turn MRC tasks in different domains.

2.2 Transfer Learning

Moreover, our work is closely related to transfer learning, as we consider the joint learning of multiple MRC tasks in various domains. There are some studies to adopt multi-task and transfer learning to address MRC. The study in [20] transfers models trained on large span-level QA datasets to sentence-level QA datasets. MT-SAN [25] is a multi-task learning framework for MRC. The results of MT-SAN shows that performance on the target task can be improved by knowledge transfer. MT-DNN [10] further extends this idea to natural language understanding by multi-task training of a series of different tasks such as sentiment analysis, text matching and MRC. Li et al. [8] extend a similar method for the task of story ending prediction. Apart from these methods, MultiQA [21] is an empirical investigation of transfer learning in ten single-turn MRC tasks. The paper shows that training on multiple MRC datasets can make the underlying model more general and robust. However, these works do not consider multi-turn MRC tasks yet.

With the rapid development of deep neural networks, knowledge transferred from unsupervised tasks can be used for learning task-specific models. For instance, pre-trained word embeddings such as Word2vec [13] and Glove [14] are the key components for NLP tasks. With deeper models and more data, large-scale pre-trained language models such as ELMO [15], BERT [5], ALBERT [7], RoBERTa [12] and XLNet [26] show their effectiveness on many downstream NLP tasks. Different from the existing studies that address general NLP tasks, our study proposes a unified framework for both single-turn and multi-turn MRC tasks. We further design the CAT-BERT model to leverage information from source MRC tasks to help the learning of the target MRC task.

3 The CAT-BERT Model

In this section, we start with the task description. After that, we introduce the CAT-BERT model and its transfer learning procedure.

3.1 Task Description and Overall Framework

The CAT-BERT model is designed to address the following problem. Let $P = [w_{p_1}, w_{p_2}, \ldots, w_{p_i}]$ be the input passage, where w_{p_i} stands for i-th word in the passage. The history question answer pairs are represented as:

$$history = [(Q_1, A_1), (Q_2, A_2), \ldots, (Q_{n-1}, A_{n-1})] \tag{1}$$

where Q_i and A_i denote the question and the answer in the i-th turn. Given the passage P, the history question answer pairs *history* and the question Q_n in n-th turn, our goal is to predict the correct answer span $\hat{A}_n$ in the passage. Note that *history* is specifically employed to model the multi-turn MRC task. If there is no *history*, the problem setting will become the normal single-turn MRC task.

Figure 1 shows the high-level overview of the framework. It can be referred to as Context-Aware Transferable BERT, CAT-BERT for short. In this model, we design a unified input representations for both single-turn and multi-turn MRC tasks. The context-aware BERT model backbone is employed as the shared encoder for each task, with model modifications to handle both history questions and answers. After that, a token-wise task-specific attention is introduced to model the task dependencies on each layer. Finally, a dynamic training policy is adopted to train the model for multi-task learning of these tasks.

3.2 Context-Aware BERT Encoding

As shown in Fig. 2, our model augments the original BERT model with context modeling and a task-specific attention based transfer learning framework. Details of the model are introduced in the subsequent sections.

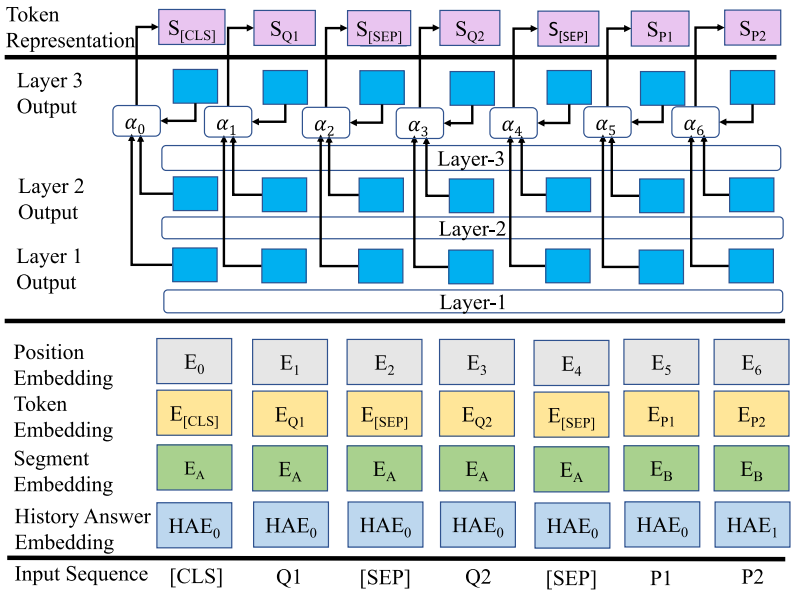

Fig. 2. The details of the context-aware BERT. We showcase an example of a BERT model with 3 layers. α_i is denoted as the token-wise layer attention for i-th token.

Modeling History Answers. Following [16], we introduce the History Answer Embedding (HAE) technique to the BERT model, in order to model history answers. Here, every token in the passage has an embedding index. If the embedding index of a token is non-zero, it means that this token is a part of the answer. For example, if the token "it" belongs to the answer of last third question, its embedding index is set to 3. Then, all embedding vectors of each token, including token embeddings, position embeddings, segment embeddings and history answer embeddings, will be summed together. Then the summed vector sequences serve as the input of the context-aware BERT encoder. For all questions and passage words that have not been used as an answer, the embedding index is 0.

Modeling History Questions. Besides history answers, it is also important to incorporate history questions. We consider a simple strategy to append the latest k history questions to the current n-th question. The history questions are separated by the special symbol `[SEP]`. For example, when k is 2, we append the previous two questions, in the format of followings:

$$\texttt{[CLS]}\, Q_n\, \texttt{[SEP]}\, Q_{n-1}\, \texttt{[SEP]}\, Q_{n-2}\, \texttt{[SEP]}\, P\, \texttt{[SEP]} \tag{2}$$

where Q_n and P refer to the tokens of the current n-th question and the passage.

3.3 Transfer Learning with Task-Specific Attention

We then present the transfer learning component for multi-task learning of different MRC tasks. Briefly speaking, the framework learns multi-turn and single-turn MRC tasks simultaneously, where all tasks share the context-aware BERT but with different task-specific layers and task-specific attention weights.

Task-Specific Attention. To learn the dependencies of tasks on specific layers, we equip the model with task-specific token-wise attention. We denote the i-th token representation in j-th layer as H_{ij}. We employ the soft attention mechanism to adapt the importance of the outputs of different levels in the context-aware BERT encoder. Formally, we define S_i^t, the final representation of i-th token for the task t, as follows:

$$S_i^t = \sum_j \alpha_{ij}^t H_{ij}, \tag{3}$$

where $t \in \{T_1, T_2, \ldots, T_k\}$ (i.e., the MRC task collection). α_{ij}^t is the attention weight corresponding to i-th token at j-th layer for the task t.

The attention weights are then defined as follows:

$$\alpha_{ij}^t = \frac{e^{H_{ij}^t * W_t + b_j^t}}{\sum_j e^{H_{ij}^t * W_t + b_j^t}}, \tag{4}$$

$$\textstyle\sum_j \alpha_{ij}^t = 1. \tag{5}$$

Note that b_j^t in the above formula can be viewed as the layer bias. It is designed for helping the attention module to know the layer depth in the neural network, which plays a similar role to the position embeddings in the original token representations. Meanwhile, the attention weights are task-specific, which are essential for the model to capture the unique characteristics for different tasks.

We further denote the output from the shared context-aware BERT encoder as the matrix $S^t \in \mathbb{R}^{d \times m}$, where d is the dimension of each token's output vector and m is the length of input sequence. We add two output layers on S^t to predict the start position and end position of the answer spans. Formally, we have:

$$P_t^s = Softmax(W_s^t S^t + b_t^s), \tag{6}$$

$$P_t^e = Softmax(W_e^t S^t + b_t^e), \tag{7}$$

where t is the task index. $W_s^t, W_e^t \in R^{1 \times d}$, b_s^t and $b_e^t \in R^{1 \times 1}$ are the corresponding projection matrices and bias terms. s and e stand for the start and end positions of the answer spans, respectively. After we obtain the probabilities P_s^t and P_e^t for each word as the start and end positions of the answer span, during the inference phase, top c words with the highest probabilities are selected to form valid answer candidates.

Algorithm 1. CAT-BERT Training Procedure

Require: Batched context enhanced training examples $B = \{B^1, B^2, \ldots, B^K\}$ from the task set $\{T_1, T_2, \ldots, T_k\}$, where $B^t = \{B_1^t, B_2^t, \ldots, B_p^t\}$
Ensure: The CAT-BERT model M
1: Freeze parameters in task-specific attention and output layers. Set other parameters (w_t) to be trainable.
2: **while** *steps* $< N_1$ **do**
3: Sample a task t from a pre-defined task distribution.
4: Read a batch B_p^t from B^t.
5: Run through the CAT-BERT model to obtain the task-specific loss L_t.
6: Calculate the gradients $\nabla_{w_t} L_t$.
7: Update the parameters $w_t = w_t$ - $\lambda \nabla_{w_t} L_t$ where λ is the learning rate.
8: **end while**
9: Freeze the parameters of the context-aware BERT encoder. Set parameters in task-specific attention and output layers (w_t') to be trainable.
10: **while** *steps* $< N_2$ **do**
11: Sample a task t from a pre-defined task distribution.
12: Read a batch B_p^t from B^t.
13: Run through the CAT-BERT model to obtain the task-specific loss L_t.
14: Calculate the gradients $\nabla_{w_t'} L_t$.
15: Update the parameters $w_t' = w_t' - \lambda \nabla_{w_t'} L_t$.
16: **end while**

Learning Objectives. For a given MRC task, we adopt the negative log likelihood as the loss function. Formally, the sample-wise loss function for the start position is:

$$Loss_s^t = -\log P_{s_i}^t \tag{8}$$

The sample-wise loss for the end position $Loss_e^t$ can be obtained in a similar way. Hence, the total loss of the task t is the sum of two prediction losses, i.e.

$$Loss^t = \frac{Loss_s^t + Loss_e^t}{2}. \tag{9}$$

For simplicity, we omit all the regularization terms in the loss functions.

3.4 Dynamic Training Policy

The training policy is defined as a probability distribution for each MRC task, which can also be viewed as the coefficient weights of different tasks. By utilizing the dynamic training policy, our framework can be more flexible to handle different tasks. For ease of implementation, we adopt a simple strategy in this work, where we sample data from each task with equal probability. We leave the design and analysis of complicated training policies as future work.

Here we explain how to transfer knowledge from source MRC tasks to the target MRC task. The procedure is also shown in Algorithm 1. The whole process has two stages: (1) multi-task training and (2) task-specific fine-tuning:

Multi-task Training. We select a task t according to the training policy, and read the batch from the task t to do a forward pass. Then we make a backward pass and update all the parameters except task-specific attention parameters. This is achieved by simply set the attention weights α as fixed, where we set α_{ij}^{t} as 1 if j is the last layer's index, and 0 otherwise. This helps to train a shared context-aware BERT encoder.

Task-Specific Fine-Tuning. For this stage, we fix the parameters in the context-aware BERT encoder and only update the token-wise task-specific attention and task-specific output layers. This stage seeks to tune task-specific parameters to capture task-specific characters so as to boost the end-task performance.

4 Experiments

In this section, we conduct extensive experiments to examine our model performance. Firstly, we show that the CAT-BERT model is highly effective for multi-turn MRC. Next, we conduct experiments to examine the benefits brought by transfer learning and our context modeling method. Finally, we qualitatively evaluate the learned task-specific attention weights, and discuss the insightfulness of task-specific attention.

4.1 Datasets

In this work, all the experiments are conducted on three public MRC datasets: QuAC [3], SQuAD 2.0 [18] and CoQA [19]. The statistics of these datasets are shown in Table 1. We take SQuAD 2.0 [18] and CoQA [19] as source domain datasets and QuAC [3] as the target domain dataset. Both QuAC and CoQA are famous datasets for multi-turn conversational MRC tasks, thus multi-turn interaction knowledge learned from CoQA can be potentially transferred to QuAC. Below, we briefly introduce the three datasets:

- QuAC: The QuAC dataset aims to simulate the information-seeking scenario in real life. It contains 14k dialogues and 100k question-answers pairs in total. The passages are collected through crowdsourcing from one single domain in Wikipedia.
- CoQA: The CoQA dataset also belongs to conversational question answering, which contains 127k question-answer pairs and 8k conversations. Text passages are selected from seven different domains in Wikipedia. The abstractive answers and the supporting evidence are also provided.
- SQuAD 2.0: The SQuAD 2.0 dataset focuses on single-turn MRC. It augments the version 1.0 of the SQuAD dataset with additional 50k negative question answers.

Table 1. The statistics of QuAC, CoQA and SQuAD 2.0 datasets. Both QuAC and CoQA are multi-turn MRC datasets, while SQuAD is a singe-turn MRC dataset.

	QuAC			CoQA			SQuAD 2.0		
	Train	Dev	Test	Train	Dev	Test	Train	Dev	Test
Questions	83,568	7,354	7,353	108,647	7983	-	130,31	11,873	-
Dialogues	11,567	1,000	1,002	7199	500	-	19,035	1,204	-
Questions/dialogue	7.2	7.4	7.3	15.1	16.0	-	6.8	9.9	-
Tokens/question	6.5	6.5	6.5	5.5	5.5	-	9.9	10.1	-
Tokens/answer	15.1	12.3	12.3	9.3	9.2	-	3.2	3.2	-
Avg. tokens/passage	397	440	446	276	266	-	117	127	-
% Unanswerable	20.2	20.2	20.2	19.0	13.2	-	33.4	50.1	-

4.2 Experimental Setup

We follow the evaluation settings used in QuAC[1] to examine our method and all the baselines. We adopt three metrics to evaluate our model: the word-level F1 score measures the overlap between the prediction and gold answers, HEQQ refers to the percentage of questions in which the model exceeds human, and HEDD measures the percentage of dialogues where the model exceeds human.

In the experiments, we set the learning rate as 3e−5, the batch size as 12, and the max sequence length as 512. The training step is 24k for single task, and we double the training steps if we add another task. For the BERT-WWM model[2] on the three-task setting, the learning rate is set to 2e−5 and the training step is 48k. We sample batches from tasks with equal probability (which is the training policy). The max answer length is set to 50. For QuAC, CoQA and SQuAD 2.0 tasks, we append the token "CANNOT ANSWER" and "UNKNOWN" to the end of the passage. All the models are implemented with TensorFlow and trained with NVIDIA Tesla V100 GPU.

4.3 Overall Results

Table 2 shows the CAT-BERT performance on the QuAC test set[3]. Overall, our model achieves competitive results on the leaderboard, outperforming some strong baselines include BERT-FlowDelta, ConvBERT, BertMT, etc. For the results of history answer embeddings, we suggest readers to refer to the paper [16], as it makes a full comparison with the effects caused by different turns in history answer embeddings.

Note that there are two methods using data augmentation strategies achieve better results on the leaderboard. We will also consider such data augmentation strategies in near future as well. We also note that there is an concurrent study

[1] https://s3.amazonaws.com/my89public/quac/scorer.py.
[2] It refers to the BERT model with whole word masking.
[3] For fair comparison, we omit the ensemble methods and those methods with data augmentation in the QuAC leaderboard.

Table 2. A comparison of the proposed model and methods from the QuAC leaderboard. † means the score is copied from leaderboard. Note that, our final model was originally named as TransBERT in the leaderboard. To avoid confusion with other models, we name it as CAT-BERT in this work.

Methods	F1	HEQQ	HEQD	Total
BiDAF++ †	50.2	43.3	2.2	95.7
BiDAF++ w/2-Context †	60.1	54.8	4.0	118.9
BERT+HAE †	62.4	57.8	5.1	125.3
FlowQA†	64.1	59.6	5.8	129.5
GraphFlow†	64.9	60.3	5.1	130.3
BERT w/2-context†	64.9	60.2	6.1	131.2
HAM†	65.4	61.8	6.7	133.9
zhiboBERT†	67.0	63.5	8.6	139.1
ConvBERT†	68.0	63.5	9.1	140.6
BertMT†	68.9	65.2	8.9	143.0
Context-Aware-BERT †	69.6	65.7	8.1	143.4
BERT-FlowDelta†	67.8	63.6	**12.1**	143.5
CAT-BERT (Our model)	**71.4**	**68.1**	10.0	**149.5**

History-Att-TransBERT that achieves slightly better results, which shows the helpfulness of transfer learning. However, due to the lack of the published paper and the source code of the model, it is difficult to assess their method and compare our method with theirs.

4.4 Comparison of Transfer Policies

In this section, we compare the impacts of different transfer learning policies and the task-specific output layer. In Table 3, we conduct experiments using three types of transfer learning policies with different choices of source-domain tasks. We denote the sequential task learning setting as Seq and the mixed task learning as Mix. Our approach can be viewed as a mixed task training policy augmented with task-specific output layers, denoted as Co. From the results, we can see that our method achieves the best scores among all policies under the same tasks. Comparing with mixed task learning, our method also attains a better performance, due to the design of the task-specific output layer.

From the results, we can also find that sequential learning can obtain a little higher results than the mix training setting in F1 and HEQQ. However, in sequential learning, the training order does matter. The performance drops especially when a different type of task is inserted between two same tasks. Readers can observe the results of Seq (CoQA-SQuAD 2.0-QuAC) v.s. Seq (SQuAD 2.0-CoQA-QuAC). Additionally, a good training order requires prior knowledge. Thus, it might be easier yet beneficial to incorporate task-specific output lay-

Table 3. The experimental results of different transfer policies are presented. The task order in Seq stands for the task learning order. -L and -W refer to results obtained from BERT-Large and BERT-WWM models, respectively.

Policy	Tasks	F1	HEQQ	HEQD
None	QuAC	65.8	61.8	7.2
Seq	CoQA, QuAC	67.6	63.9	8.2
Seq	SQuAD 2.0, QuAC	67.2	63.2	8.6
Seq	CoQA & SQuAD 2.0 & QuAC	68.1	64.3	7.7
Seq	SQuAD 2.0 & CoQA & QuAC	**68.3**	**64.7**	**9.3**
Mix	CoQA & QuAC	67.6	63.4	8.4
Mix	SQuAD 2.0 & QuAC	66.8	62.9	**8.8**
Mix	QuAC & SQuAD 2.0 & CoQA	**68.4**	**64.5**	8.3
Co	CoQA & QuAC	67.9	64.3	8.9
Co	SQuAD 2.0 & QuAC	67.8	64.0	**9.6**
Co	QuAC & SQuAD 2.0 & CoQA	**68.7**	**65.0**	9.4
Co-L	QuAC & SQuAD 2.0 & CoQA	70.2	66.5	9.9
Co-W	QuAC & SQuAD 2.0 & CoQA	**73.1**	**69.9**	**13.3**

Table 4. The effects of task-specific attention mechanism. (w/o attn) means without the using of attention mechanisms.

Model	F1	HEQQ	HEQD	Total
CAT-BERT-12	68.6	64.8	9.2	142.6
CAT-BERT-6	68.7	65.0	9.4	143.1
CAT-BERT-3	**68.7**	**65.1**	**9.6**	**143.4**
CAT-BERT (w/o attn)	68.4	64.6	8.3	141.3
CAT-BERT-WWM-24	73.2	70.0	13.1	149.9
CAT-BERT-WWM-12	73.3	70.1	12.9	156.3
CAT-BERT-WWM-6	**73.3**	**70.1**	**13.4**	**156.8**
CAT-BERT-WWM-3	73.3	70.1	13.0	156.4
CAT-BERT-WWM (w/o attn)	73.1	69.9	13.3	156.3

ers in the mix policy to capture the task differences, so the potential negative transfer brought by the other tasks can be reduced. We also show the improvements made by employing better pre-trained language models. The results show that increasing the model size (see Co-L) and adopting the whole word masking technique for BERT (see Co-W) can improve all metrics greatly.

We notice that (SQuAD 2.0 & QuAC) always achieves better HEQD than (CoQA & QuAC). This means that the single-turn dataset SQuAD helps more than the multi-turn dataset CoQA for the QuAC task, although QuAC belongs to the category of the multi-turn MRC task. Benefiting from our unified frame-

work, we can easily train a model to learn the shared knowledge between single-turn and multi-turn MRC tasks.

4.5 The Benefit from the Attention Mechanism

Table 4 shows the results with respect to the different numbers of layers that employ the attention mechanism. The upper part shows the performance of attention applied to the last 12, 6, 3 layers. The model size of the CAT-BERT backbone is the same as the BERT-Base model. The bottom part shows the performance of attention mechanism, with the backbone changed to the BERT-Large model with whole word masking. We can observe that, with the incorporation of the token-wise attention technique, the performance scores using both BERT-Base (denoted as CAT-BERT) and BERT-Large-WWM (denoted as CAT-BERT-WWM) as backbones are improved. For example, for CAT-BERT-WWM, F1 improves from 73.1 to 73.3, HEQQ from 69.9 to 70.1. This shows it is beneficial to incorporate the attention mechanism to capture the token-wise task-specific information to further improve the model performance.

Furthermore, we visualize the attention scores of the last three layers from our final model CAT-BERT-WWM on a randomly chosen example for both QuAC and SQuAD 2.0 tasks, as shown in Fig. 3. On the QuAC task (left), most of the attention weights are close to 1 on the last two layers; while on the SQuAD 2.0 task (right), the larger attention weights appear only in the last layer. The figure further demonstrates the necessity to introduce the task-specific token level attention mechanism to our framework to deal with the task differences among various MRC tasks.

4.6 Error Analysis

To analyze the shortcomings of our model, we randomly sample 50 wrong answers from predictions. The main errors can be categorized into two types:

- **Logical Error.** A typical error of the model is that the internal semantic changes in the passage are sometimes ignored. For example, the question is "Did Davies recover?", with two descriptions provided: "He subsequently collapsed after a drug overdose and was taken to hospital", and then "Ray recovered from his illness as well as his depression". The model only regards the first description as the answer and ignores the second description. This type of errors contributes mostly to the poor performance.
- **Indirect Description.** Although some answers are contained in the passages, they may be indirectly described, where complicated reasoning may be required for answering those implicit questions. For example, the question is "How profitable was the last album?" and the gold answer should be "the biggest-selling German music act in history". But the model gives a wrong prediction "CANNOT ANSWER". In this case, it is necessary to enhance the reasoning ability of the model, which is a non-trivial task.

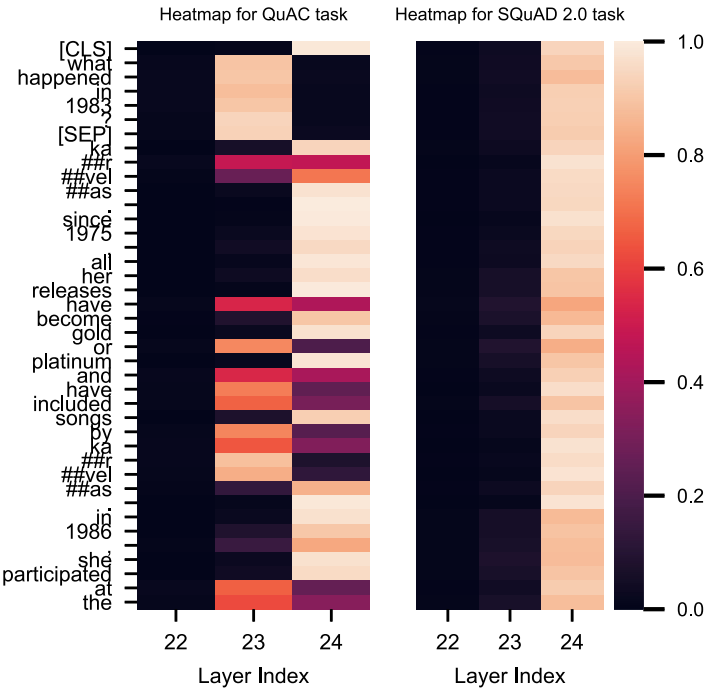

Fig. 3. The visualization of task-specific attention scores in the last three layers from the CAT-BERT-WWM model. The left is from the QuAC task, with the right from the SQuAD 2.0 task.

5 Conclusion and Future Work

In this work, we propose a deep BERT-based transfer learning model named CAT-BERT to unify the learning of multi-turn and single-turn MRC tasks. In this model, a task-specific token-wise attention mechanism is proposed to capture the dependencies on different layers for each task. Extensive evaluation results show thee proposed method is effective and achieves competitive results. Qualitative results also demonstrate that the attention weights learned by the model are insightful.

Acknowledgments. We would like to thank anonymous reviewers for their valuable comments. This work was partially sponsored by the NSFC projects (No. 61402403, No. 62072399, No. U19B2042), Chinese Knowledge Center for Engineering Sciences and Technology, MoE Engineering Research Center of Digital Library, Alibaba-Zhejiang University Joint Institute of Frontier Technologies, and the Fundamental Research Funds for the Central Universities. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect those of the sponsors.

References

1. Chen, H., Liu, X., Yin, D., Tang, J.: A survey on dialogue systems: recent advances and new frontiers. arXiv:1711.01731 [cs] (November 2017)
2. Chen, Y., Wu, L., Zaki, M.J.: GraphFlow: exploiting conversation flow with graph neural networks for conversational machine comprehension, pp. 1230–1236 (2020)
3. Choi, E., et al.: QuAC: question answering in context. In: EMNLP, pp. 2174–2184 (2018)
4. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: BERT: pre-training of deep bidirectional transformers for language understanding. CoRR (2018)
5. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: BERT: pre-training of deep bidirectional transformers for language understanding. NAACL (2018)
6. Huang, H.Y., Choi, E., Yih, W.t.: FlowQA: grasping flow in history for conversational machine comprehension, CoRR abs/1810.06683 (2018)
7. Lan, Z., Chen, M., Goodman, S., Gimpel, K., Sharma, P., Soricut, R.: ALBERT: a lite BERT for self-supervised learning of language representations, CoRR abs/1909.11942 (2019)
8. Li, Z., Ding, X., Liu, T.: Story ending prediction by transferable BERT. In: Proceedings of the 28th International Joint Conference on Artificial Intelligence, IJCAI 2019, Macao, China, 10–16 August 2019, pp. 1800–1806 (2019)
9. Liu, N.F., Gardner, M., Belinkov, Y., Peters, M.E., Smith, N.A.: Linguistic knowledge and transferability of contextual representations. In: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, pp. 1073–1094 (2019)
10. Liu, X., He, P., Chen, W., Gao, J.: Multi-task deep neural networks for natural language understanding. In: Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, 28 July–2 August 2019, Volume 1: Long Papers, pp. 4487–4496 (2019)
11. Liu, X., Shen, Y., Duh, K., Gao, J.: Stochastic answer networks for machine reading comprehension. In: Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), , Melbourne, Australia, pp. 1694–1704. Association for Computational Linguistics (July 2018)
12. Liu, Y.: RoBERTa: a robustly optimized BERT pretraining approach. CoRR (2019)
13. Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J.: Distributed representations of words and phrases and their compositionality. Adv. Neural. Inf. Process. Syst. **26**, 3111–3119 (2013)
14. Pennington, J., Socher, R., Manning, C.D.: GloVe: global vectors for word representation. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, A meeting of SIGDAT, a Special Interest Group of the ACL, Doha, Qatar, 25–29 October 2014, pp. 1532–1543 (2014)
15. Peters, M.E., et al.: Deep contextualized word representations. In: Walker, M.A., Ji, H., Stent, A. (eds.) Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2018, pp. 2227–2237. Association for Computational Linguistics (2018)
16. Qu, C., Yang, L., Qiu, M., Croft, W.B., Zhang, Y., Iyyer, M.: BERT with history answer embedding for conversational question answering. In: SIGIR, pp. 1133–1136 (2019)

17. Qu, C., et al.: Attentive history selection for conversational question answering. In: CIKM, pp. 1391–1400 (2019)
18. Rajpurkar, P., Jia, R., Liang, P.: Know what you don't know: unanswerable questions for SQuAD. In: Gurevych, I., Miyao, Y. (eds.) Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Volume 2: Short Papers, Melbourne, Australia, 15–20 July 2018, pp. 784–789. Association for Computational Linguistics (2018)
19. Reddy, S., Chen, D., Manning, C.D.: CoQA: a conversational question answering challenge, CoRR abs/1808.07042 (2018)
20. Sun, Y., Cheng, G., Qu, Y.: Reading comprehension with graph-based temporal-casual reasoning. In: Proceedings of the 27th International Conference on Computational Linguistics, Santa Fe, New Mexico, USA, pp. 806–817. Association for Computational Linguistics (August 2018)
21. Talmor, A., Berant, J.: MultiQA: an empirical investigation of generalization and transfer in reading comprehension, CoRR abs/1905.13453 (2019)
22. Trippas, J.R., Spina, D., Cavedon, L., Joho, H., Sanderson, M.: Informing the design of spoken conversational search: perspective paper. In: CHIIR (2018)
23. Wang, R., Wang, M., Liu, J., Chen, W., Cochez, M., Decker, S.: Leveraging knowledge graph embeddings for natural language question answering. In: Proceedings of the 24th International Conference on Database Systems for Advanced Applications, DASFAA 2019, pp. 659–675 (January 2019)
24. Wu, H., Tian, Z., Wu, W., Chen, E.: An unsupervised approach for low-quality answer detection in community question-answering. In: Candan, S., Chen, L., Pedersen, T.B., Chang, L., Hua, W. (eds.) DASFAA 2017. LNCS, vol. 10178, pp. 85–101. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-55699-4_6
25. Xu, Y., Liu, X., Shen, Y., Liu, J., Gao, J.: Multi-task learning with sample re-weighting for machine reading comprehension. In: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, pp. 2644–2655 (2019)
26. Yang, Z., Dai, Z., Yang, Y., Carbonell, J.G., Salakhutdinov, R., Le, Q.V.: XLNet: generalized autoregressive pretraining for language understanding, CoRR abs/1906.08237 (2019)
27. Yatskar, M.: A qualitative comparison of CoQA, SQuAD 2.0 and QuAC. In: NAACL-HLT, pp. 2318–2323 (2019)
28. Yeh, Y.T., Chen, Y.N.: FlowDelta: modeling flow information gain in reasoning for conversational machine comprehension, CoRR abs/1908.05117 (2019)
29. Zhang, X., Yang, A., Li, S., Wang, Y.: Machine reading comprehension: a literature review, CoRR abs/1907.01686 (2019)
30. Zhang, X.: MC^2: Multi-perspective convolutional cube for conversational machine reading comprehension. In: ACL, pp. 6185–6190 (2019)
31. Zhang, Y., Chen, X., Ai, Q., Yang, L., Croft, W.B.: Towards conversational search and recommendation: system ask, user respond. In: CIKM (2018)
32. Zhu, C., Zeng, M., Huang, X.: SDNet: contextualized attention-based deep network for conversational question answering, CoRR abs/1812.03593 (2018)