2022 The 22st International Conference on Control, Automation and Systems (ICCAS 2022)
BEXCO, Busan, Korea, Nov. 27∼Dec. 01, 2022

Leveraging Extended Chat History through Sentence Embedding in Multi-turn Dialogue toward Increasing User Engagement

Zeyu Ding*, Armagan Elibol, and Nak Young Chong

School of Information Science, Japan Advanced Institute of Science and Technology,
Ishikawa, 923-1292, Japan ({s2010132, aelibol,nakyoung}@jaist.ac.jp) * Corresponding author

Abstract: Multi-turn dialogue is the major manifestation of a conversation. Compared with single-turn dialogue, response selection is more complex as the context varies. We stress the importance of dialogue history and apply the pre-trained model BERT to assign proper weight to each utterance of a dialogue. Previous works take all the dialogue history as context to measure the matching degree of a context-response pair, causing the quadratic computational cost and truncation of longer sequences exceeding the length limitation of BERT. We propose a sentence-based method to deal with the aforementioned problems, obtaining the sentence embedding of a single unit utterance of dialogue and forming a classification token of a context-response pair. We discuss how to obtain a sentence embedding with high quality and to design the input representations in response selection. The results show that the average of the first-last layer output exhibits the best performance for obtaining a sentence representation. The proposed method, concatenating the sentence embeddings of context with the token embeddings of response candidates, is nearly on a par with the token embedding based SOTA method. Notably, the processable length of dialogue history is enlarged about ten times with a low computational cost, potentially reducing chatbot response time and inspiring user engagement.

Keywords: Multi-turn Dialogue, Chatbox, Human-Robot Interaction.

1. INTRODUCTION

Customer service chatbots are being widely used in e-commerce [4] and the nursing robot provides elderly companion services through conversation [1], [2]. The two major types of chatbots are retrieval-based systems and generative systems. The former requires a large number of predefined responses and knowledge base, which makes it reliable yet less flexible. The latter can generate new dialogues, but is still under research. Retrieval-based multi-turn chatbots use pairwise text scoring. Leveraging BERT [5], there are two common methods for evaluating the quality of pairwise text: bi-encoder and cross-encoder [23]. Bi-encoder, for example, calculates the cosine semantic similarity of given context and the context of prepared context-response pair. On the other hand, the cross-encoder measures the matching score of the given context and candidate response pair. The context and response candidate are concatenated as the input. Then the output of high dimensional classification token embedding is fed to a logistic regression model to calculate the matching score. Cross-encoders achieve higher performance, while their computational cost is higher than their counterparts.

We hypothesized that the longer the dialogue history, the more accurate the response selection. It is reported that a chatbot can benefit from a previous chat history learning an individual user's preference and background [13]. Specifically, the self-attention mechanism helps increase the impact of related dialogue utterances and decrease the impact of unrelated ones. This led to two consequent problems. One is the high computational cost due to the interaction between arbitrary two tokens with a self-attention-based model. The other is that as the chat history goes longer, it will eventually exceed the input length limitation of BERT. Managing the trade-off between better performance and reasonable computational cost, we use sentence embedding instead of a series of token embeddings. This allows for accommodating longer chat history and lowering the computational cost. We discuss strategies to obtain high-quality sentence embeddings for response selection. We shed light on the input token representation of context-response pair and sentence embedding representation of a single dialogue utterance of context. Additional small-scale experiments are conducted on dialogue length. We train and test our method on the Ubuntu Dialogue Corpus (UDC). BERT takes an input of a sequence within a maximum of 512 tokens. The average number of words of an utterance in UDC is 10.34. We therefore take about 49.2 utterances into consideration with a token embedding-based method. In contrast, the number of utterances increases to about 500 with our method which is more than ten times the token-embedding method. We can estimate the number by subtracting the length of response (assumed average utterance length 10.34) as well as three special tokens (one [CLS] and two [SEP] tokens) from the maximum length of input. Moreover, our sentence embedding-based method reduces the computational complexity compared with the token embedding-based method. When a chatbot provides a response to the user's utterance in a dialogue, given l_i as the length of the i-th utterance of the dialogue, the computational complexity decreases from $O((\sum_{n=0}^{i} l_i)^2)$ to $\sum_{n=0}^{i} O(l_i^2)$. The final result of our sentence embedding-based method (86.62%) is comparable to the token embedding-based SOTA model (90.82%).

2. RELATED WORK

2.1 Multi-turn Dialogue

Context plays an important role in multi-turn dialogues since it may affect response selection. The main challenges as well as the latest advances for multi-turn dialogue have been presented in different types of surveys [18]. For retrieval-based model, RNN-based architecture had been a common choice until the Pre-trained Language Model (PLM) was widely used, such as a sequential matching network [26] and a fine-grained context representation by a deep utterance aggregation model [31]. In recent years, PLM is extensively preferred due to the low burden of fine-tuning as its extra training makes the model adapt to given tasks like post-training [25] and fine-tuning with domain-specific dataset [10]. Combination with other learning strategies similar to fine-grained contrastive learning [14] and external auxiliary self-supervised learning have made it overcome the problem of incoherence and inconsistency in multi-turn dialogues [27]. Efficient utilization of context has proven to be effective as in Multi-hop Selector [29] that filters out the noise inside the context. On the other hand, the generative model is a typical sequence-to-sequence model. Kim *et al.* [11] proposed an RNN model that emphasizes the importance of dialogue history assigning different weights learned by an attention mechanism to context utterances. Zhang *et al.* [30] investigated the topical relevance. Different from models that mainly concentrate on word and sentence level information, this model verified the usefulness of topic level information in multi-turn dialogue generation. Data manipulation [3] provided another perspective by increasing the proportion of effective data through data augmentation. There are studies that combine generative model and retrieval-based model together. In [21], it was shown that an ensemble model could outperform each single model by a large margin.

Strategies for response selection include topic-aware modeling, speaker-aware modeling, and knowledge-grounded modeling. Dialogue comprehension attempts to understand the dialogue contents from a different level, such as utterance-level dialogue understanding [6], topic-aware modeling [28], and hierarchical residual matching [24]. Earlier strategies for personalized chatbots utilized explicit user profiles from external resources, whereas recent approaches learn the implicit information like personalized language style and preferences [17]. Speaker-aware chatbots can filter context according to the speaker's information and select a subset of utterances considered as important ones [8]. External knowledge improves the performance of a chatbot. The knowledge base can be created from chat history, document or web page [20], and saved as structured data. It is possible to increase the accuracy when picking or generating a response, applying attention mechanism to balance context and knowledge base [12] or manage the dialogue flow [7].

2.2 BERT Related Models

The pre-trained BERT is considered a standard model in NLP tasks [5]. Input token ids are converted into input token embeddings and sequentially fed to the transformer layers and transformed into a context-aware embedding. Self-attention allows BERT to learn longer range dependencies of sentences than RNN-based models.

Our work aims to find a proper way of getting sentence embeddings. BERT returns dynamic token embeddings with a different context. The easiest way to obtain the sentence embedding is to use the [CLS] token embedding based on the dynamic token embeddings. Sentence BERT [19] used a siamese network to get sentence embeddings which greatly decreases the time consumption for semantic similarity searching. The anisotropy of token embedding distribution was reported harmful for [CLS] sentence embedding. Data whitening was proposed to eliminate the adverse impact of anisotropy by normalizing the data distribution [22].

The purpose of [CLS] token is to learn the correlation of a sentence pair, which determines its suitability for classification related tasks rather than semantic similarity comparison. Since our proposed cross-encoder-method is based on matching-degree with [CLS] token rather than cosine similarity, these calibration strategies are not suitable to our situation. Another sentence embedding method with higher performance applies different pooling methods with different layer combinations. The pooling method includes average pooling and max pooling. For the objective layer, the last layer is the most common, and combining multiple layers, such as the last two layers or the first and last layers, is an alternative strategy. In this work, we investigate the proportion of the two outputs that provides the best result.

2.3 Basis of this Research

This research builds on [9] that provided a fine-grained post-training for learning the utterance-level knowledge. For post-training, this work uses a Masked Language Model (MLM) variant and utterance relevance classification (URC) as training objectives different from the standard BERT. The MLM variant proposed in RoBERTa [15] learns a more contextual representation by randomly masking a token rather than masking a predetermined token. The URC task generates short sub-dialogues consisting of a short context and a response. There are three possible sources for the response: the correct response, a random utterance in the same dialogue, or a random response from another dialogue. With this multi-classification task, the model can distinguish the positive and negative cases better in subsequent fine-tuning. The loss of the post-training is twofold: the cross-entropy of the MLM task and URC task formulated as Eq. 1, and the loss of the URC as Eq. 2.

$$L_{FP} = L_{MLM} + L_{URC} \tag{1}$$

$$L_{URC} = -\sum \sum_{i}^{3} y_i \log \left(g_{urc} \left(sc, u_t \right)_i \right), \qquad (2)$$

where sc is the short context of a sub-dialogue, u_t is the candidate response, y_i is the ground truth of a sample, and the subscript i denotes one of the three classes in classification task for post-training, respectively. $g_{urc}(sc, u_t)$ is the final score calculated via feeding [CLS] through a single layer perceptron for the multi-classification. g_{urc} measures the relevance between the short context and target utterance, which is used to calculate the cross-entropy loss for back-propagation. The outer sum adds up the cross-entropy for each utterance in dialogue, and the inner sum does the discrete probability distributions of the three classes.

3. PROPOSED MODEL

We define technical terms used. **Utterance** is the contents a single speaker speaks in one dialogue turn. An utterance may contain several **sentences**, but generally only one. In a multi-turn dialogue, **context** means all the utterances except the last utterance, which is named as the candidate **response**.

3.1 Outline of the Proposed Model

Our model as depicted in Fig 1 deals with context and response differently. The input to BERT is the id of each input token. Then the ids are converted into static token embeddings. We use a single sentence embedding for each context utterance to represent a series of token ids. The token ids of the response candidate are converted into embeddings and concatenated with the previous sentence embeddings. After obtaining the context-response pair that consists of sentence embedding of context utterances and token embeddings of response candidate, the pair is fed into BERT as the input. Then the final output of [CLS] token is used for classification to determine whether the given pair is proper or not. The 768-dimension [CLS] token are transformed into a scalar by a single-layer perceptron. The output scalar is processed by a sigmoid function to calculate the normalized score. Comparing with the ground truth labels, the loss is calculated with a cross-entropy function and parameters are updated via back-propagation.

3.2 Input Token Representation

We consider the input token representation and output embedding representation to get high-quality sentence embeddings of the context utterances. The output sentence embeddings are concatenated with response token embeddings and become the input of the subsequent process. Likewise, the input embedding representation needs to be designed for better performance.

3.2.1 Inner-utterance Input Representation

Inner-utterance input token representation is the input for obtaining the context-related token embeddings of context utterances. The left side of Fig. 2 shows the four types of strategies used.

- Method 1 is a standard BERT input representation with special tokens [CLS] and [SEP] on both ends.
- Method 2 removes all special tokens. We hypothesize that the knowledge learned by special tokens may become noise while representing a sentence. Input without special tokens can represent the original utterance better.
- Method 3 uses [EOU] token instead of [SEP], which is used as the segmentation mark between context utterances in previous studies.
- Method 4 only uses [EOU] token at the last of a context utterance, which simply splits the original token embedding based input with the [EOU] token.

3.2.2 Sentence Embedding Representation

We propose a sentence embedding method to represent a sequence of context utterances instead of token embeddings. This method should represent a sentence without too much information loss and with high efficiency of restoring the information carried by a certain token in the original input. [CLS] token is one simple representation, but not good as sentence embedding [22]. As mentioned in related work, applying pooling on different layers shows a better performance. We test different layer combinations with average pooling.

3.2.3 Inter-utterance Input Representation

The inter-utterance input embedding representation determines how to combine the sentence embeddings together. The bottom right corner of Fig 2 shows the input token representation of BERT (top) and the work in [9] (bottom). [SEP] is used to segment a pair of sentences with a standard BERT model. In the post-training, [SEP] is used to segment the context utterances and the response. Meanwhile, an additional special token [EOU] is used to segment utterances inside the context. We aim to find out the difference when applying different inter-utterance input embedding representation.

4. EXPERIMENTS

An extensive set of experiments were conducted with different strategies on UDC [16], designing our proposed base model under the following conditions.

- The max length of a context utterance is set to 50.
- The max length of a context utterance embedding and response token embedding pair is set to 128.
- The average of the first-last layer output is used as the context utterance embeddings.
- The input token representation of inner-utterance utterances is set without any special tokens.
- The input token representation of inter-utterance utterances is set without any special tokens.

Two versions of the SOTA model and our model are:

- BERT with fine-tuning based on token embeddings (a variant of SOTA model)
- BERT with fine-grained post-training and fine-tuning based on token embeddings (SOTA model)

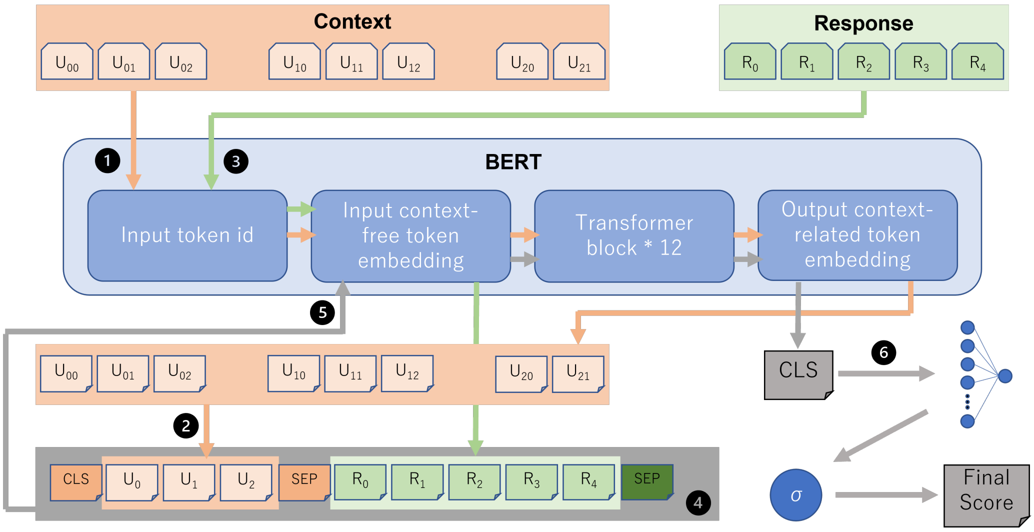
Context
U00 U01 U02 U10 U11 U12 U20 U21
Response
R0 R1 R2 R3 R4
BERT
1
3
Input token id
Input context-free token embedding
Transformer block * 12
Output context-related token embedding
5
U00 U01 U02 U10 U11 U12 U20 U21
2
CLS U0 U1 U2 SEP R0 R1 R2 R3 R4 SEP
4
CLS
6
σ
Final Score

Fig. 1. Proposed model architecture: Context utterances are fed into BERT to obtain context-related embeddings ①, which are subsequently processed to obtain sentence embeddings ②. Candidate responses are fed to BERT to get the context-free token embeddings before further processed by the transformer blocks ③. The sentence embedding of context utterances and token embedding of response are concatenated to get a text pair ④, and fed into BERT ⑤. The [CLS] token embedding of output will be given to a one-layer perceptron ⑥ and the normalized result is used as the final score for assessment.

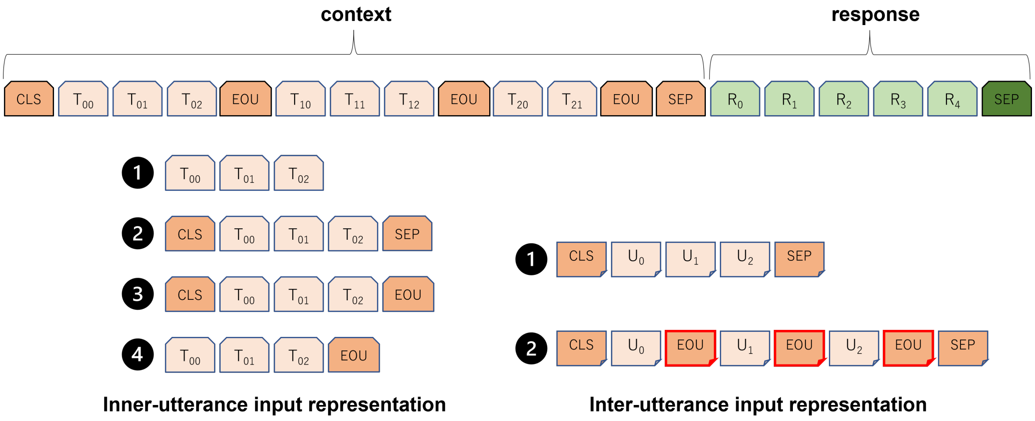
context
response
CLS T00 T01 T02 EOU T10 T11 T12 EOU T20 T21 EOU SEP R0 R1 R2 R3 R4 SEP
1 T00 T01 T02
2 CLS T00 T01 T02 SEP
3 CLS T00 T01 T02 EOU
4 T00 T01 T02 EOU
Inner-utterance input representation
1 CLS U0 U1 U2 SEP
2 CLS U0 EOU U1 EOU U2 EOU SEP
Inter-utterance input representation

Fig. 2. We fine-tune our model after fine-grained post-training [9]. The input token design of post-training is on the top of the figure. To use a single sentence embedding instead of a series of token embeddings with ideal performance, we discuss various inner-utterance as well as inter-utterance input representations. The former concentrate on the input design of an utterance, whereas the latter concentrate on the relationship between utterances.

- BERT with fine-tuning based on sentence embeddings (a variant of our model)
- BERT with fine-grained post-training based on token embeddings and fine-tuning based on sentence embeddings (our model)

For input representation, different inner-utterance and inter-utterance input representations are tested. The tokens of a context utterance are fed into BERT with or without different special token combinations in order to investigate how these special tokens affect the quality of generating a sentence embedding. For inter-utterance representation, we aim to find out whether keeping correspondence with the structure of token embedding-based post-training is necessary or not. This is tested by with or without a special token [EOU] inserted between the context utterances embedding.

Sentence embedding representations were tested with the following three combinations: [CLS] token, the average of last hidden layer output, and the average of first-last hidden layer outputs. We observed that the last method outperformed the others. Further experiments were carried out to find the optimal mixing ratio of Layer 1 and Layer 12.

The chat history was evaluated using dialogues containing more than 512 tokens and those lasting more than 15 turns, respectively. For the former, we compare the difference between an entire sequence of tokens and 512 tokens maximum. For the latter, we verify the importance of topical coherence in utterances.

5. COMPUTATIONAL RESULTS AND ANALYSIS

We compared our proposed method with the SOTA model. Table 1 shows the recall rate of top 1, top 2, and top 5 based on the token embedding method with different training methods. FP and FT stand for fine-grained post-training and fine-tune, respectively. The suffix 't' and 's' denote that training methods (FP and FT) are based on the token embedding method or sentence embedding method.

Comparing BERT_FTt with BERT_FPt_FTt, post-training is highly effective since the recall rate at top 1 increased 10.01% to 90.82%. BERT_FTs uses our sentence embedding-based fine-tuning method. The recall rate at top 1 is 11.32% lower than the token embedding-based fine tuning BERT_FTt. Sentence embedding-based fine-tuning applied to a token embedding-based pre-trained model does not reach the same performance level as token embedding-based fine-tuning. Applying sentence embedding-based fine-tuning to a token embedding-based post-trained model, and testing with sentence embedding-based method, the recall rate at top 1 increased 13.89% to 83.39%.

Table 1. Results of our model with different embedding methods

model	R@1	R@2	R@5
BERT_FTt	80.81	89.67	97.52
BERT_FPt_FTt	90.82	95.97	99.39
BERT_FTs	69.50	82.45	95.43
BERT_FPt_FTs	83.39	92.34	98.67

Fig. 3 shows the fine-tuning convergence curve of 2 epochs with BERT_FPt_FTt model. Leveraging post-training, the curve converges rapidly within the first 10 batches. The average losses of epoch 1 and epoch 2 are 0.156 and 0.093, respectively. As shown in Fig. 4, the converging speed of BERT_FPt_FTs is slower than the token embedding-based SOTA model. Also, the fluctuation and average loss are greater than the SOTA model with an average loss of 0.278 at epoch 1 and 0.211 at epoch 2.

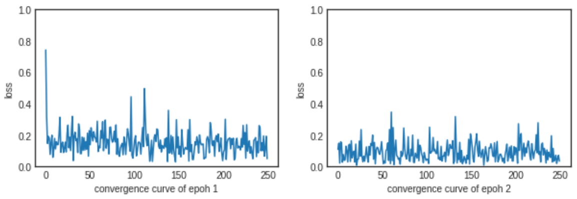

Fig. 3. The convergence curve of SOTA model [9]

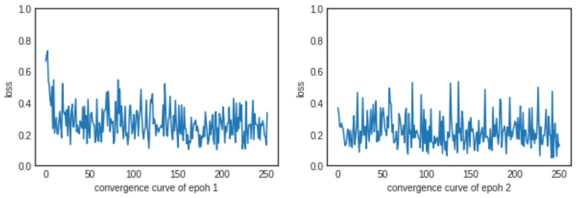

Fig. 4. The convergence curve of our model

It should be noted that there is still a gap between our proposed sentence embedding method and the token embedding-based method. However, delicately designing the input representation and sentence embedding, we can still get an acceptable result that narrows the gap to about 4.5%. This provides an evidence that token embedding-based post-training mainly learns utterance level knowledge with token level input. However, our sentence embedding-based method may not efficiently leverage this knowledge. Thus, we conjecture that a sentence embedding-based post-training may lead to better performance which is a promising direction of future research.

5.1 Effect of Different Input Representations

We present the experimental results obtained using different input token representation methods.

5.1.1 Inner-utterance Input Representation

From the results presented in Table 2, it can be concluded that special tokens are necessary when applying a sentence embedding instead of a sequence of token embeddings to represent a sentence. As the initial hypothesis, we consider that special tokens are not helpful for a good sentence embedding representation. Since these tokens are designed for specific tasks like MLM and NSP, the information carried by these special tokens is only useful for specific purposes and they may become noise otherwise. Therefore, we considered input token representation of a context utterance without any special tokens to better represent a sentence. However, the result indicates that with the special tokens, the performance increased 2.5%. This means that the standard BERT input token representation pattern with special tokens is the best way for gaining the sentence embedding. Even if other customized special tokens are used in post-train or fine-tuning, [CLS] and [SEP] can help with a better representation.

Table 2. Models comparison with different **inner-utterance** input token representations

model	loss	R@1	R@2	R@5
with [CLS] & [SEP]	0.170	86.17	93.96	99.00
with [CLS] & [EOU]	0.201	84.25	92.94	98.77
with [EOU]	0.209	83.67	92.57	98.74
without special tokens	0.211	83.39	92.34	98.67

5.1.2 Inter-utterance Input Representation

A simple test was performed on the inter-utterance utterances input token representation. As evidenced by the result in Table 3, placing a special token [EOU] between context utterances brings about a slight performance improvement. The recall rate of top 1 increased slightly 0.30%. This indicates that the correspondence with the inter-utterance input token representation pattern of post-training is beneficial at least to some extent. Differences between the structure of the token embedding-based fine-tuning method and the sentence embedding-based fine-tuning method are narrowed.

Table 3. Models comparison with different **inter-utterance** input token representations

model	loss	R@1	R@2	R@5
with [EOU]	0.265	83.69	92.58	98.74
without [EOU]	0.211	83.39	92.34	98.67

5.2 Effect of Different Sentence Embedding Representations

Table 4 summarizes the experimental results obtained with different sentence embedding methods. It can be seen that using [CLS] as a sentence embedding achieved the lowest accuracy among the proposed methods. The average of the last layer as sentence embedding outperformed [CLS] but is still not satisfactory. The average of the first layer and average of the first-last layer (the weight of the first and the last layer is 0.5 respectively) showed almost the same results, which are better than the other methods. In consideration of the favorable performance of using the average of multi-layer outputs, we conducted a series of experiments to see if there is an optimal balance between the first and the last layer.

Table 4. Models comparison for sentence embedding methods

model	loss	R@1	R@2	R@5
[CLS]	0.433	52.98	66.62	86.55
L12	0.314	79.01	91.20	98.46
L1	0.199	82.83	92.02	98.60
L1+L12	0.211	83.39	92.34	98.67

5.3 Effect of Different Mixing Ratio of L1 and L12

We tested different weights for Layer 1 and Layer 12 as summarized in Table 5. The first two columns denote the weight of Layer 1 and Layer 12, while the following columns are for average loss and recall rate at top 1 of epoch 1 and epoch 2. For the sake of convenience, the normalized weights of Layer 1 and Layer 12 are denoted by two numbers in parentheses separated by a comma.

Table 5. Comparison of different weights in Layer1 and Layer12 with our baseline model

L1	L12	epoch1		epoch2	
		avg loss	R@1	avg loss	R@2
1.00	0.00	0.270	82.54	0.199	82.83
0.90	0.10	0.268	82.83	0.197	83.11
0.80	0.20	0.268	82.96	0.197	83.55
0.75	0.25	0.266	83.04	0.197	83.60
0.70	0.30	0.267	82.94	0.198	83.58
0.60	0.40	0.268	83.17	0.198	83.49
0.50	0.50	0.276	82.40	0.211	83.39
0.40	0.60	0.285	82.22	0.218	82.90
0.25	0.75	0.300	81.15	0.233	81.50
0.20	0.80	0.308	80.77	0.237	81.32
0.00	1.00	0.326	78.24	0.314	79.01

First, we set several pairs of weights for the coarse scale investigation of the relationship between the weights and recall rate. We observed an apparent decrease in recall rate when the weight of Layer 1 is lower than 0.5. Therefore, additional experiments were carried out with finer interval in the range of (0.5, 0.5) to (1.0, 0.0). Fig. 5 shows the recall rate with different weights in epoch 1 and epoch 2. The optimal range of the first-last layer ratio was found between (0.6, 0.4) and (0.8, 0.2). Either only using Layer 1 or Layer 12 exhibited substandard performance. Compared with the Layer 1 only case, the Layer 12 only case performed worse.

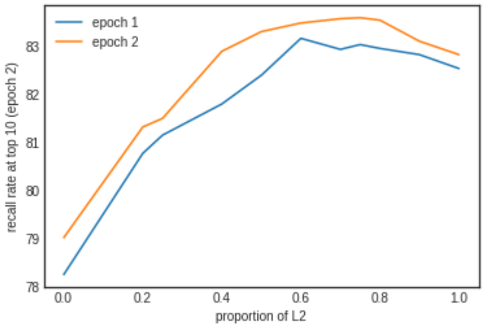

Fig. 5. Recall rate at top 1 with different mixing ratios

In consideration of the performance of inner-utterance input representation with special tokens, Table 6 shows an adverse outcome in mixing ratio compared with the previous one. A high proportion of high-level information performed better. It needs further study to determine whether the high-level information carried by special tokens has a great impact on sentence embedding representation.

Table 6. Comparison of different weights in Layer1 and Layer12 with special token attached inner-utterance input representation

L1	L12	epoch1		epoch2	
		avg loss	R@1	avg loss	R@2
0.90	0.10	0.263	83.92	0.190	84.21
0.75	0.25	0.260	85.36	0.183	85.36
0.60	0.40	0.243	85.79	0.173	86.29
0.50	0.50	0.239	86.14	0.171	86.51
0.40	0.60	0.237	86.37	0.169	86.62
0.25	0.75	0.240	86.32	0.171	86.57
0.10	0.90	0.239	86.05	0.173	86.32

5.4 Chat history

We picked up $6,070$ dialogues from the UDC dataset that contained more than 512 tokens. Table 7 shows that our sentence-embedding method with an entire sequence of tokens (Method 1) outperformed that with the maximum of 512 tokens (Method 2). Since the part beyond the limit of 512 tokens may contain the topical information, the recall rate at top 1 increased 0.71%.

Table 8 shows the importance of topically coherent utterances. We chose $77,240$ dialogues lasting more than 15 turns. We compared the effect of selective removal of utterances on recall rate. Method 3 did not remove any utterances and yielded $R@1$ of 85.14%. Method 4 removed the 1st and the 2nd utterances, whereas Method 5 removed the 3rd and the 4th utterances. Method 4 and Method 5 decreased 0.77% and 0.52%, respectively, which means topical coherence was of importance.

Table 7. Different strategies for dealing with long dialogues

model	R@1	R@2	R@5
Method 1	83.25	95.41	99.36
Method 2	82.54	95.32	99.53

Table 8. Comparison of utterance removals

model	R@1	R@2	R@5
Method 3	85.14	93.68	99.11
Method 4	84.37	93.45	98.94
Method 5	84.62	93.31	98.95

6. CONCLUSIONS AND FUTURE WORK

We proposed a BERT-based sentence embedding method that has proven to be promising in multi-turn dialog, which was assessed from four main aspects: inner-utterance input representation, inter-utterance input representation, sentence embedding representation, and chat history.

For inner-utterance input representation, special tokens [CLS] and [SEP] of a standard BERT model were desired for better performance. Although the special tokens were designed for specific purposes, the information carried by them was necessary for better sentence representation. For inter-utterance input token representation, the performance would increase slightly if the input representation pattern corresponded with the pre-training or post-training. In this work, an [EOU] token was used for segmenting context utterances. With the [EOU] token inserted in the sentence embeddings of each context utterance, the performance improved 0.30% in recall rate at the top 1. The comparison of inner-utterance and inter-utterance input representation indicated that the former has a more obvious positive impact. For sentence embedding representation, the average of the first-last layer output was a good option. Inner-utterance input representation without any special tokens preferred higher proportion of the first layer, which is the low level information source. However, Inner-utterance input representation with [CLS] and [SEP] gave a contrary result that higher proportion of high level information source was preferred. We will verify in the future that adding high level information of special tokens with low level information of word tokens may bring about better performance. Chat history also matters in multi-turn dialogue selection. For long dialogues, there exist topically coherent utterances carrying information more important than others. Our model was made better at retaining such important utterances and helpful to guide response selection.

There is still room for further improvements in the following three aspects. First, sentence embedding-based post-training may lead to better understanding on sentence embeddings of context utterances. Secondly, a new type of embeddings is needed to distinguish a sentence embedding of context and a token embedding of response. Thirdly, different mixing ratios for special tokens and word tokens can be applied for better sentence embedding representation.

REFERENCES

[1] Barbara Bruno, Carmine Tommaso Recchiuto, Irena Papadopoulos, Alessandro Saffiotti, Christina Koulouglioti, Roberto Menicatti, Fulvio Mastrogiovanni, Renato Zaccaria, and Antonio Sgorbissa. Knowledge representation for culturally competent personal robots: requirements, design principles, implementation, and assessment. *International Journal of Social Robotics*, 11(3):515–538, 2019.

[2] Ha-Duong Bui and Nak Young Chong. Autonomous speech volume control for social robots in a noisy environment using deep reinforcement learning. In *IEEE International Conference on Robotics and Biomimetics*, pages 1263–1268, 2019.

[3] Hengyi Cai, Hongshen Chen, Yonghao Song, Cheng Zhang, Xiaofang Zhao, and Dawei Yin. Data manipulation: Towards effective instance learning for neural dialogue generation via learning to augment and reweight. *arXiv preprint arXiv:2004.02594*, 2020.

[4] Lei Cui, Shaohan Huang, Furu Wei, Chuanqi Tan, Chaoqun Duan, and Ming Zhou. Superagent: A customer service chatbot for e-commerce websites. In *Proceedings of ACL 2017, system demonstrations*, pages 97–102, 2017.

[5] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[6] Deepanway Ghosal, Navonil Majumder, Rada Mihalcea, and Soujanya Poria. Utterance-level dialogue understanding: An empirical study. *arXiv preprint arXiv:2009.13902*, 2020.

[7] Lucrezia Grassi, Carmine Tommaso Recchiuto, and Antonio Sgorbissa. Knowledge-grounded dialogue flow management for social robots and conversational agents. *arXiv preprint arXiv:2108.02174*, 2021.

[8] Jia-Chen Gu, Tianda Li, Quan Liu, Zhen-Hua Ling, Zhiming Su, Si Wei, and Xiaodan Zhu. Speaker-aware bert for multi-turn response selection in retrieval-based chatbots. In *Proc. ACM International Conference on Information & Knowledge Management*, pages 2041–2044, 2020.

[9] Janghoon Han, Taesuk Hong, Byoungjae Kim, Youngjoong Ko, and Jungyun Seo. Fine-grained post-training for improving retrieval-based dialogue systems. In *Proc. Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1549–1558, 2021.

[10] Matthew Henderson, Ivan Vulić, Daniela Gerz, Iñigo Casanueva, Paweł Budzianowski, Sam Coope, Georgios Spithourakis, Tsung-Hsien Wen, Nikola Mrkšić, and Pei-Hao Su. Training neural response selection for task-oriented dialogue systems. *arXiv preprint arXiv:1906.01543*, 2019.

[11] Jintae Kim, Shinhyeok Oh, Oh-Woog Kwon, and Harksoo Kim. Multi-turn chatbot based on query-context attentions and dual wasserstein generative adversarial networks. *Applied Sciences*, 9(18):3908, 2019.

[12] Sihyung Kim, Oh-Woog Kwon, and Harksoo Kim. Knowledge-grounded chatbot based on dual wasserstein generative adversarial networks with effective attention mechanisms. *Applied Sciences*, 10(9):3335, 2020.

[13] Juntao Li, Chang Liu, Chongyang Tao, Zhangming Chan, Dongyan Zhao, Min Zhang, and Rui Yan. Dialogue history matters! personalized response selectionin multi-turn retrieval-based chatbots. *arXiv preprint arXiv:2103.09534*, 2021.

[14] Yuntao Li, Can Xu, Huang Hu, Lei Sha, Yan Zhang, and Daxin Jiang. Small changes make big differences: Improving multi-turn response selection in dialogue systems via fine-grained contrastive learning. *arXiv preprint arXiv:2111.10154*, 2021.

[15] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019.

[16] Ryan Lowe, Nissan Pow, Iulian Serban, and Joelle Pineau. The ubuntu dialogue corpus: A large dataset for research in unstructured multi-turn dialogue systems. *arXiv preprint arXiv:1506.08909*, 2015.

[17] Hongjin Qian, Zhicheng Dou, Yutao Zhu, Yueyuan Ma, and Ji-Rong Wen. Learning implicit user profile for personalized retrieval-based chatbot. In *Proc. ACM International Conference on Information & Knowledge Management*, pages 1467–1477, 2021.

[18] Xipeng Qiu, Tianxiang Sun, Yige Xu, Yunfan Shao, Ning Dai, and Xuanjing Huang. Pre-trained models for natural language processing: A survey. *Science China Technological Sciences*, pages 1–26, 2020.

[19] Nils Reimers and Iryna Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084*, 2019.

[20] Tara Safavi and Danai Koutra. Relational world knowledge representation in contextual language models: A review. *arXiv preprint arXiv:2104.05837*, 2021.

[21] Yiping Song, Cheng-Te Li, Jian-Yun Nie, Ming Zhang, Dongyan Zhao, and Rui Yan. An ensemble of retrieval-based and generation-based human-computer conversation systems. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, IJCAI'18, page 4382–4388. AAAI Press, 2018.

[22] Jianlin Su, Jiarun Cao, Weijie Liu, and Yangyiwen Ou. Whitening sentence representations for better semantics and faster retrieval. *arXiv preprint arXiv:2103.15316*, 2021.

[23] Nandan Thakur, Nils Reimers, Johannes Daxenberger, and Iryna Gurevych. Augmented sbert: Data augmentation method for improving bi-encoders for pairwise sentence scoring tasks. *arXiv preprint arXiv:2010.08240*, 2020.

[24] Heyuan Wang, Ziyi Wu, and Junyu Chen. Multi-turn response selection in retrieval-based chatbots with iterated attentive convolution matching network. In *Proc. ACM International Conference on Information and Knowledge Management*, pages 1081–1090, 2019.

[25] Taesun Whang, Dongyub Lee, Chanhee Lee, Kisu Yang, Dongsuk Oh, and Heuiseok Lim. An effective domain adaptive post-training method for bert in response selection. In *INTERSPEECH*, pages 1585–1589, 2020.

[26] Yu Wu, Wei Wu, Chen Xing, Ming Zhou, and Zhoujun Li. Sequential matching network: A new architecture for multi-turn response selection in retrieval-based chatbots. *arXiv preprint arXiv:1612.01627*, 2016.

[27] Ruijian Xu, Chongyang Tao, Daxin Jiang, Xueliang Zhao, Dongyan Zhao, and Rui Yan. Learning an effective context-response matching model with self-supervised tasks for retrieval-based dialogues. *arXiv preprint arXiv:2009.06265*, 2020.

[28] Yi Xu, Hai Zhao, and Zhuosheng Zhang. Topic-aware multi-turn dialogue modeling. In *AAAI Conference on Artificial Intelligence*, 2021.

[29] Chunyuan Yuan, Wei Zhou, Mingming Li, Shangwen Lv, Fuqing Zhu, Jizhong Han, and Songlin Hu. Multi-hop selector network for multi-turn response selection in retrieval-based chatbots. In *Proc. Conference on Empirical Methods in Natural Language Processing and International Joint Conference on Natural Language Processing*, pages 111–120, 2019.

[30] Hainan Zhang, Yanyan Lan, Liang Pang, Hongshen Chen, Zhuoye Ding, and Dawei Yin. Modeling topical relevance for multi-turn dialogue generation. *arXiv preprint arXiv:2009.12735*, 2020.

[31] Zhuosheng Zhang, Jiangtong Li, Pengfei Zhu, Hai Zhao, and Gongshen Liu. Modeling multi-turn conversation with deep utterance aggregation. *arXiv preprint arXiv:1806.09102*, 2018.