Query-to-Session Matching: Do NOT Forget History and Future during Response Selection for Multi-Turn Dialogue Systems

Zhenxin Fu*
Wangxuan Institute of Computer Technology, Peking University
fuzhenxin95@gmail.com

Shaobo Cui, Feng Ji, Ji Zhang, Haiqing Chen
DAMO Academy, Alibaba Group
{yuanchun.csb,zhongxiu.jf,zj122146,haiqing.chenhq}@alibaba-inc.com

Dongyan Zhao
Wangxuan Institute of Computer Technology, Peking University
zhaodongyan@pku.edu.cn

Rui Yan†
[1] Wangxuan Institute of Computer Technology, Peking University
[2] Young Fellow in Beijing Academy of Artificial Intelligence
ruiyan@pku.edu.cn

ABSTRACT

Given a user query, traditional multi-turn retrieval-based dialogue systems first retrieve a set of candidate responses from the historical dialogue sessions. Then the response selection models select the most appropriate response to the given query. However, previous work only considers the matching between the query and the response but ignores the informative dialogue session in which the response is located. Nevertheless, this session, composed of the response, the response's history and the response's future, always contains valuable contextual information which can help the response selection task. More specifically, if the current query and a response's history both refer to the same question, we can conclude that this response is quite likely to answer this query. As for the response's future, it can always provide contextual hints and supplementary information that might be omitted in the response. Inspired by such motivation, we propose a *query-to-session* matching (QSM) framework to make full use of the session information: matching the query with the candidate session instead of the response only. Different from the previous work which ranks response directly, the response in the session with the highest *query-to-session* matching score will be selected as the desired response. In our proposed framework, the query, history, and future are all sequences of utterances, which makes it necessary to model the relationships among the utterances. So we propose a novel dialogue flow aware *query-to-session* matching (DF-QSM) model. The dialogue flows model the relationships among the utterances through a memory network. To our best knowledge, our paper is the first work to utilize both the response's history and future in the response selection task. The experimental results on three multi-turn response selection benchmarks show that our proposed model outperforms existing state-of-the-art methods by a large margin.

CCS CONCEPTS

• **Computing methodologies** → **Discourse, dialogue and pragmatics**; • **Information systems** → **Question answering**.

KEYWORDS

Response selection, Query-to-session matching, Dialogue flow

ACM Reference Format:
Zhenxin Fu, Shaobo Cui, Feng Ji, Ji Zhang, Haiqing Chen, Dongyan Zhao, and Rui Yan. 2020. Query-to-Session Matching: Do NOT Forget History and Future during Response Selection for Multi-Turn Dialogue Systems. In *Proceedings of the 29th ACM International Conference on Information and Knowledge Management (CIKM '20), October 19–23, 2020, Virtual Event, Ireland*. ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3340531.3411938

1 INTRODUCTION

Human-machine interaction through conversations has attracted increasing attention in recent years due to its wide applications. Some chatbots have been developed to serve human and make life intelligent: Microsoft XiaoIce [38] for social chats, Alibaba AliMe [17] for customer service, etc. There are mainly two types of chatbots: the generation-based and the retrieval-based. The former one generates response through language generation methods [23, 28], while the latter one retrieves the response from a large candidate set [25, 30]. In this paper, we focus on multi-turn retrieval-based dialogue systems which can produce informative and fluent responses.

A multi-turn retrieval-based dialogue system takes the query, which consists of the previous utterances and the current question, as input to retrieve the most appropriate response from a candidate set. The mainstream methods for retrieval-based dialogue systems comprise two steps: (1) constructing a rough candidate set from the whole corpus; (2) selecting the best response from the candidate set. Concretely, the first step is a coarse-grained relevance searching process between the query and the candidate conversations, which is usually accomplished by TF-IDF for efficiency [32]. The second

*This work was done when Zhenxin Fu was an intern at Alibaba Group.
†Corresponding author: Rui Yan (ruiyan@pku.edu.cn)

CIKM '20, October 19–23, 2020, Virtual Event, Ireland

ACM ISBN 978-1-4503-6859-9/20/10…$15.00
https://doi.org/10.1145/3340531.3411938

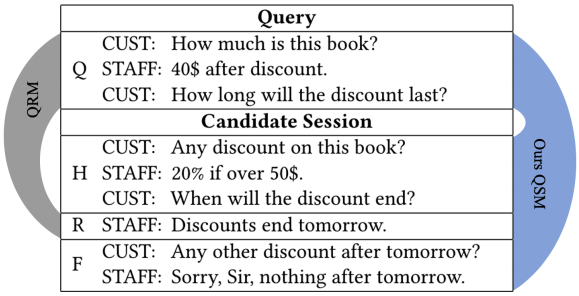

Figure 1: Example for the query context and the corresponding candidate session which is composed of the response, response's history, and response's future. CUST stands for customer. STAFF represents the customer service staff.

step is a fine-grained reranking process by computing the matching degree between the query and the candidate responses [9, 19, 39].

The aforementioned systems produce the **response** in the **response** level, in which they retrieve rough candidate **responses** in the first step and conducts the **query-to-response** matching in the second step. However, we should keep in mind that the candidate response never exists in isolation. In particular, the candidate response is located in a dialogue session as shown in Figure 1. In addition to the response, there are response's history (utterances before the response) and response's future (utterances after the response) in the candidate session. We can observe that the response's history and future are all helpful information for the response selection task. To be specific: (1). For the response's history, if the query Q and response's history H both imply the same question, we will have more confidence to conclude that this response R can answer the current query well [6]. From the example in Figure 1, we can see that both the query and the response's history are about the duration of the discount. And the response after the history, which answers the question about express choice, fits the query quite precisely; (2). As for the response's future F, from the subtle hints provided by F ("other discount after tomorrow"), we can safely back-infer that the response may talk about the discount time information, which matches the query well. In particular, this type of supplementary information provided by the response's future can also help us in the response selection task. Inspired by such motivation, we propose the *query-to-session* matching (QSM) framework which considers the whole candidate session instead of the single response.

As shown in Figure 1, the conventional query-to-response matching (QRM) framework only copes with the matching between one sequence of utterances and one response. Our *query-to-session* matching framework, on the other hand, involves the matching between several sequences of utterances. The query, the history, and the future are all sequences of utterances. Thus, there are two key problems in this setting: (1). different utterances in the session contribute differently in the *query-to-session* matching task; (2). not all information in the utterances help the matching task. To solve such problems, we propose a novel memory network named dialogue flow to gracefully extract useful information from the session's utterances. More specifically, for each utterance, dialogue flow dynamically decides {how much information, which aspect of the information} should be written into the memory.

We conduct experiments on three benchmarks for the response selection task: Ubuntu Dialogue Corpus, Douban Conversation Corpus, and E-commerce Dialogue Corpus. The results show that our proposed dialogue flow aware *query-to-session* matching model achieves state-of-the-art results. The model ablation results and the session ablation results indicate the effectiveness of each proposed component and verify the usefulness of the response's history and future. The code is released[1]. The contributions of our paper are:

(1) To our best knowledge, our paper is the first work to propose the *query-to-session* matching framework which provides a new perspective for the response selection task.
(2) We propose to use a memory-based network named dialogue flow to cope with the knotty matching problem between the sequences of utterances, which can precisely extract useful and related information from the utterances in the dialogue session.
(3) Detailed experiments show our proposed dialogue flow aware *query-to-session* matching model significantly outperforms the conventional query-to-response matching approaches and our proposed other strong *query-to-session* matching baselines.
(4) We conduct empirical experiments to quantitatively evaluate the role of the history and future in the dialogue session and the influence of the session size.

2 RELATED WORK

2.1 Retrieval-based Dialogue Systems

This paper explores a novel *query-to-session* framework for the response selection task in multi-turn retrieval-based dialogue systems. We first introduce some literature work about the retrieval-based dialogue systems. Most of the existing multi-turn retrieval-based dialogue systems focus on the query-to-response matching that matches the query and the response directly. Wu et al. [32] propose the Sequential Matching Network (SMN) which models the relationship between each utterance in the query and the response by a cross-attention matrix. Zhou et al. [40] propose the Deep Attention Matching (DAM) model that encodes the query and the response through self-attention. Cross-attention and a 3-D convolution are also applied to predict the matching degree in DAM. Tao et al. [26] propose the Interaction over Interaction (IoI) model to make utterance-response interaction go deep by stacking multiple interaction blocks. Most of the aforementioned models are composed of the representation layer, interaction layer, and interaction aggregation layer. The representation layer encodes the utterances, which is usually accomplished by recurrent neural network (RNN) or self-attention. The interaction layer takes technologies like cross-attention to obtain the interaction representation between the utterances in the query and the response. The interaction aggregation layer aggregates the interaction representations through RNN or convolutional neural network (CNN).

Different from these methods that only consider the query-to-response matching but ignore the informative history and future of the response, we investigate the *query-to-session* matching problem where the history and future are considered. Fu et al. [6] propose the

[1] https://github.com/fuzhenxin/Query-Session-Matching-CIKM2020

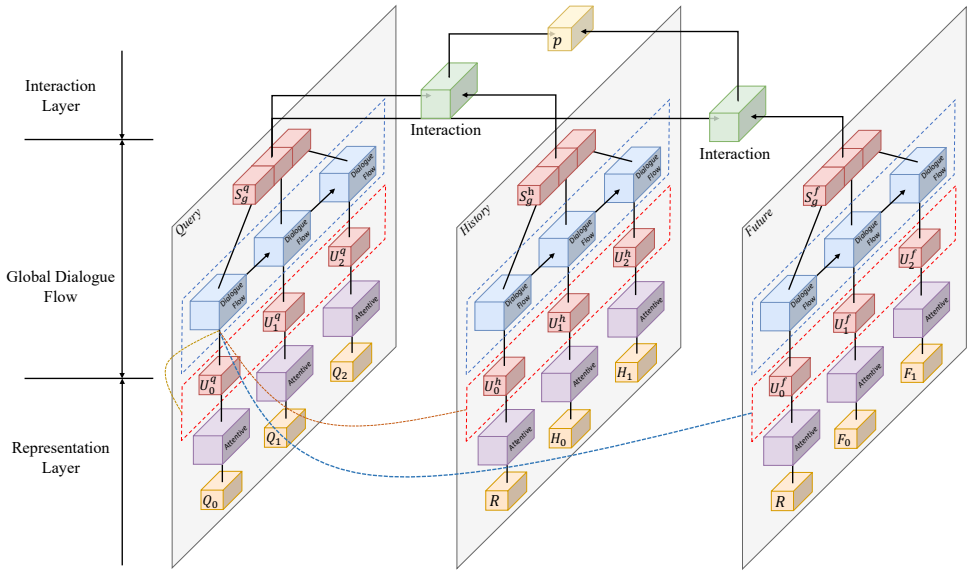

Figure 2: Model Overview. For simplicity, the query contains three turns, while the history and future contain only two turns. We only show the global dialogue flow and omit the local dialogue flow for a clear demonstration. The representation of the query, the history, the future are fed into the global dialogue flow (see these three dashed curves). We only show the whole global dialogue flow's updating of the first step in the query as an example.

context-to-session matching which utilizes the response's history only but ignores the informative response's future in information-seeking dialogue systems. Besides, our proposed model is verified in three corpora including information-seeking conversations and open-domain dialogues.

Another line of research considers external knowledge for the response selection task. Young et al. [34] propose to promote response selection with a commonsense knowledge base. Yang et al. [33] propose to help the response selection with external knowledge through pseudo-relevance feedback and QA correspondence knowledge distillation. Qu et al. [22] propose to understand and characterize how people interact in information-seeking conversation through user intent. Aliannejadi et al. [1] explore how to use the utterance relevance in the query through human-annotated labels. Different from these work which need external information and additional human annotations, our proposed *query-to-session* matching framework only utilizes the built-in history and future utterances in the dialogue sessions.

2.2 Text Matching

Since our framework is located in the text matching domain, we also introduce the mainstream text matching tasks and models. Text matching technologies have been applied to lots of areas including Natural Language Inference [3], Paraphrase Identification [10], and Information Retrieval [14]. The mainstream text matching models transform the sentences into sentence representations first and then model the relationship among sentences using technologies like cross-attention [7]. Most of the existing work investigates the matching between two sentences. Our proposed *query-to-session* framework, nevertheless, involves matching between sequence of utterances. More specifically, both the query and the candidate session consist of multiple utterances. The relationships among the utterances in the query or in the candidate session become quite important in our setting. To cope with this knotty problem, we propose an insightful dialogue flow strategy to capture the tangled relationships among them. See more details in §3.5. And in the future, we will explore to adapt the proposed dialogue flow to related domains like conversational search [4].

3 QUERY-TO-SESSION MATCHING

In this part, we first introduce the task formulation which also contains the notations. Then, the model overview is introduced to summarize the model. After that, we will introduce the basic attentive module and our proposed dialogue flow aware *query-to-session* matching model in detail.

3.1 Task Formulation

Our dialogue flow aware *query-to-session* matching model is designed to perform the *query-to-session* matching task. Each sample in the training set is denoted as $\{Q, S, l\}$, where Q is the query, S is the candidate session, and $l \in \{0, 1\}$ is the label which indicates whether the response R in session S is an appropriate response to the query Q. The session $S = \{H, R, F\}$ consists of the candidate response R and its corresponding history H and future F. The query Q, history H, and future F are all sequences of utterances which can be formulated as $Q = \{Q_0, \cdots, Q_{T_q-1}\}$, $H = \{H_0, \cdots, H_{T_h-1}\}$, and $F = \{F_0, \cdots, F_{T_f-1}\}$, where Q_j, H_j, F_j are utterances and T_q, T_h, T_f are the max turn numbers for the query, history, and future respectively. The response R is a single utterance. Given the query Q and candidate session S, our goal is to predict the label l correctly.

3.2 Model Overview

Our model contains three layers: the representation layer, the dialogue flow layer, and the interaction layer. The representation layer uses an attentive module to encode the utterances. The dialogue flow layer models the dialogue flow through local and global memory networks and explores how much information of utterances should be written to the dialogue flow. The interaction layer utilizes an attentive module and cross-attention mechanism to obtain the interaction matching representation between the query and candidate session. Finally, the interaction representations are used to predict the *query-to-session* matching score. The entire model is shown in Figure 2 for better understanding.

3.3 Background: Attentive Module

Inspired by the success of Transformer [27], we adopt the attentive module following Yuan et al. [36], Zhou et al. [40] to learn the utterance representations. The attentive module is a variant of the encoder of the Transformer with single-head attention. The attentive layer is composed of a single-head self-attention sub-layer and a position-wise fully connected feed-forward sub-layer. A residual connection [11] is employed around each of the two sub-layers, followed by layer normalization [16]. It is abstracted as $f_{\text{att}}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) \in \mathbb{R}^{t \times d_k}$, where $\mathbf{Q} \in \mathbb{R}^{t \times d_k}$, $\mathbf{K} \in \mathbb{R}^{t \times d_k}$ and $\mathbf{V} \in \mathbb{R}^{t \times d_k}$ are matrices representing the query input, the key input, and the value input respectively. t is the sentence length and d_k is the dimension of the word embedding.

3.4 Representation Layer

Given the query and the candidate session which consists of the response, the response's history, and the response's future, the representation layer transforms them into the corresponding utterance representations. Specifically, they are first transformed into word embedding representations. Then the aforementioned attentive module is applied to encode the word embedding representations into utterance representations.

To better illustrate this process, we take the i-th utterance in the query, which is represented as Q_i, as an example. The utterance Q_i is transformed into word representation by looking up the word embedding table, obtaining the word representation $\mathbf{E}_i^q \in \mathbb{R}^{t \times d_k}$, where t is the sentence length and d_k is the dimension of word embedding. Then the word representation is fed into an attentive module to get the utterance representation $\mathbf{U}_i^q \in \mathbb{R}^{t \times d_k}$:

$$\mathbf{U}_i^q = f_{\text{att}}(\mathbf{E}_i^q, \mathbf{E}_i^q, \mathbf{E}_i^q) \tag{1}$$

The history, the future, and the response can be encoded using the same approach. The i-th utterance of the history and future can be represented as $\mathbf{U}_i^h$ and $\mathbf{U}_i^f$. The response can be represented as $\mathbf{U}^r$.

The reasons why we encode the utterances using the attentive module are two-fold: firstly, the attentive module not only shows great success in machine translation [27], but also achieves superior performance in pre-training [2, 5] and response selection [26, 40]. Additionally, the attentive module is much faster than its RNN-based counterparts, which tremendously improves the training and inference efficiency.

3.5 Dialogue Flow Layer

3.5.1 Why Dialogue Flow is Necessary? There are multiple utterances in the query, history, and future, so modeling the relationships among the utterances becomes important. Generally, we have the following two intuitions:

(1) Not all information of utterances contribute to matching accuracy. Namely, the utterances always contain noise and unrelated information.
(2) The farther the utterance is from the response, the less it contributes to the matching task.[2] The utterances in different positions play different roles. Not all the utterances of the query, history, and future contribute equally to the *query-to-session* matching.

The aforementioned intuitions naturally lead to one key problem: *how to extract useful information (rather than the noise, see in question 1) from the utterances with different usefulness (see in question 2).* To solve this problem, we propose a framework named **dialogue flow** to extract useful information. Dialogue flow can be viewed as a generalized memory network. It aims to appropriately combine the related information that can help the matching task from the utterances (history, future, etc.). In other words, it only keeps useful information in its memory. Along with the dialogue flow, the useful information of each utterance is updated to the dialogue flow memory. The memory stores the dialogue information from the start-point to the current checkpoint. Though with similar names, the dialogue flow strategy involved in our work is quite different from that of FLOWQA [13]. FLOWQA explores the dialogue flow of document representation for the machine reading comprehension task, which is inappropriate and hard to be applied in the response selection task.

3.5.2 Dialogue Flow Direction. Before introducing the dialogue flow model, we need to define the dialogue flow direction for the query, history, and future. Response is the key utterance in the candidate session, so we put the response after the last utterance in history and before the first utterance in the future. For simplicity, in the following, history represents the response-integrated history (the dialogue turns before response and the response itself). Similarly, future represents the response-integrated future (response itself and the dialogue turns after the response). As we have introduced, the closer certain utterance is to the response, the more important role it plays in the matching task. To capture such motivation, the dialogue flow of the query, the history, and the future all take the utterance nearest to the response as the start-point, and flow to the farthest utterance from the response.

3.5.3 Dialogue Flow Strategies. Two kinds of dialogue flow strategies are proposed: local dialogue flow and global dialogue flow. What difference lies between the local and global dialogue flow? These two dialogue flow strategies focus on different aspects. The local dialogue flow focuses its attention on its own utterances, i.e., the query dialogue flow only focuses on the query utterances while the future dialogue flow only concentrates on the future utterances, etc. The global dialogue flow, however, takes the whole query-session

[2] For history, the latter utterances are always more important than the former ones in the history utterances. As for the future, the former, nevertheless, are more related to the response than the latter.

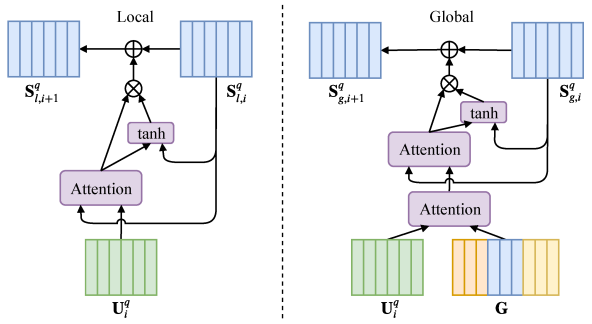

Figure 3: Local (left) and global (right) dialogue flow updating. tanh denotes the whole operation in Equation 3 and Equation 8 of local and global dialogue flow respectively. The crucial difference between the local dialogue flow and the global dialogue flow is that: the utterance representation in the global dialogue flow first attends to the global representation G to obtain an intermediate representation.

pair into consideration when updating the dialogue flow memory. In other words, the global dialogue flow considers not only its own utterances but also the global view. We provide an analogy for better understanding. The local dialogue flow is an *expert* versed in a single aspect. However, the global dialogue flow is more like a *polymath* who is knowledgeable in multiple aspects but may not be as good as the expert in certain aspects. Through our experimental results, we find the combination of these two strategies can further boost the matching performance since they capture the dialogue flow from two different perspectives: (1). the pure and specialized local flow. (2) the interrelated and encyclopedic global-aware flow.

(1). Local dialogue flow: We take the dialogue flow in the future as an example. The dialogue flow in the query and history can be modeled in the same way. Given the future representations $\{\mathbf{U}^f_0, \cdots, \mathbf{U}^f_{T_f}\}$. The dialogue flow is from the first utterance in the future to the last utterance. We model the dialogue flow using memory network [8, 24]. The i-th local memory $\mathbf{S}^f_{l,i} \in \mathbb{R}^{t \times d_k}$ represents the dialogue flow up to now. The memory updating adds and deletes the message of the i-th utterance $\mathbf{U}^f_i$ to the memory $\mathbf{S}^f_{l,i}$, forming the next memory $\mathbf{S}^f_{l,i+1}$.

In detail, we first calculate which information of the current utterance is related to the dialogue flow (question 1). We take the attention mechanism to extract useful information:

$$\mathbf{S}^f_{u,i} = \text{Softmax}(\frac{\mathbf{S}^f_i \mathbf{U}^{f\,\mathsf{T}}_i}{\sqrt{d_k}})\mathbf{U}^f_i \tag{2}$$

where $\mathbf{S}^f_{u,i} \in \mathbb{R}^{t \times d_k}$ indicates the weighted sum of rows in current utterance representation $\mathbf{U}^f_i$ and the weights are controlled by $\mathbf{S}^f_i$. $\mathbf{S}^f_{u,i}$ decides which information in $\mathbf{U}^f_i$ will be updated to the dialogue flow memory $\mathbf{S}^f_i$.

Next, we answer question 2 by assigning a weight to the updating information $S^f_{u,i}$. The weight is calculated by:

$$\alpha_i = \tanh(\text{MLP}([\mathbf{S}^f_{l,i}, \mathbf{S}^f_{u,i}])) \tag{3}$$

where $[,]$ means concatenation and MLP represents the multi-layer perceptron. $\alpha_i \in (-1, 1)$ is the updating weight which is from -1 to 1. α_i also controls whether the information should be added or deleted to $\mathbf{S}^f_i$. negative α_i represents deletion and positive α_i means addition.

Finally the utterance updating information $\mathbf{S}^f_{u,i}$ is updated to the local dialogue flow memory $\mathbf{S}^f_{l,i}$ with weight α_i, forming the next memory $\mathbf{S}^f_{l,i+1}$:

$$\mathbf{S}^f_{l,i+1} = \mathbf{S}^f_{l,i} + \alpha_i \mathbf{S}^f_{u,i} \tag{4}$$

$\mathbf{S}^f_0$ is initialized by the first utterance in the future: $\mathbf{S}^f_0 = \mathbf{U}^f_0$. Different from other memory networks [8, 20] which model the addition and deletion operation separately, we integrate them into one operation as in Equation 4. α_i can dynamically decide whether to add or delete information of the current utterance into the dialogue flow.

(2). Global dialogue flow: The local memory only considers the local dialogue flow in the query, history, and future. Here we have a whole dialogue flow at hand, to what extent the utterances in the future are useful should also be influenced by the query and history, and vice versa. So we propose the global dialogue flow of which both the query and the candidate session are considered when updating the memory. We first calculate the global query-session pair representation:

$$\mathbf{G} = [\mathbf{U}^q, \mathbf{U}^h, \mathbf{U}^f] \tag{5}$$

where $\mathbf{U}^f \in \mathbb{R}^{T_f t \times d_k}$ is the concatenation of $\{\mathbf{U}^f_0, \cdots, \mathbf{U}^f_{T_f}\}$. $\mathbf{U}^q$ and $\mathbf{U}^h$ are constructed in the same way.

Different from the local dialogue flow, current utterance first attends to the global representation $\mathbf{G}$ forming the global-aware utterance representation $\mathbf{U}^f_{g,i} \in \mathbb{R}^{t \times d_k}$:

$$\mathbf{U}^f_{g,i} = \text{Softmax}(\frac{\mathbf{U}^f_i \mathbf{G}^\mathsf{T}}{\sqrt{d_k}})\mathbf{G} \tag{6}$$

Then the global-aware utterance representation $\mathbf{U}^f_{g,i}$ replaces the utterance representation $\mathbf{U}^f_i$ in Equation 2 forming the global-aware dialogue flow updating (other operations are the same as the local dialogue flow.):

$$S^f_{u,i} = \text{Softmax}(\frac{\mathbf{S}^f_i \mathbf{U}^{f\,\mathsf{T}}_{g,i}}{\sqrt{d_k}})\mathbf{U}^f_{g,i} \tag{7}$$

$$\alpha_i = \tanh(\text{MLP}([\mathbf{S}^f_{g,i}, \mathbf{S}^f_{u,i}])) \tag{8}$$

$$\mathbf{S}^f_{g,i+1} = \mathbf{S}^f_{g,i} + \alpha_i \mathbf{S}^f_{u,i} \tag{9}$$

Finally, the local and global dialogue flow memories are concatenated into the final dialogue flow representations $\mathbf{S}^f_l \in \mathbb{R}^{T_f t \times d_k}$

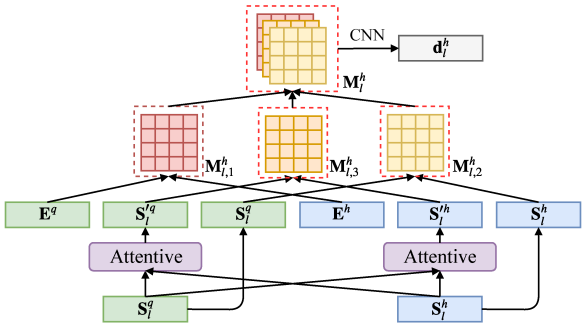

Figure 4: The interaction block. We take the query-history matching in the local dialogue flow as an example

and $\mathbf{S}_g^f \in \mathbb{R}^{T_f t \times d_k}$ respectively:

$$\begin{cases} \mathbf{S}_l^f = [\mathbf{S}_{l,0}^f, \cdots, \mathbf{S}_{l,T_f}^f] & \text{local dialogue flow} \\ \mathbf{S}_g^f = [\mathbf{S}_{g,0}^f, \cdots, \mathbf{S}_{g,T_f}^f] & \text{global dialogue flow} \end{cases} \tag{10}$$

The local and global dialogue flow of query $\{\mathbf{S}_l^q, \mathbf{S}_g^q\}$ and history $\{\mathbf{S}_l^h, \mathbf{S}_g^h\}$ can be obtained in the same way.

3.6 Interaction Layer

After obtaining the dialogue flow memories, we need to make interaction over the query and the history to get the matching relationship between them. The interaction over the query and future is also needed. In recent years, the researchers adopt the cross-attention mechanism [32, 35] to obtain the interaction matching representations, which has shown great success in a lot of tasks [3, 10]. Here we adopt the cross-attention mechanism to learn the interaction representation.

The inputs of the interaction layer can be the utterance representations $\{\mathbf{U}^q, \mathbf{U}^h, \mathbf{U}^f\}$, or the local dialogue flow memory $\{\mathbf{S}_l^q, \mathbf{S}_l^h, \mathbf{S}_l^f\}$, or the global dialogue flow memory $\{\mathbf{S}_g^q, \mathbf{S}_g^h, \mathbf{S}_g^f\}$ to learn interaction representations in different levels, where $\mathbf{U}^* \in \mathbb{R}^{T_* t \times d_k}$, $\mathbf{S}_l^* \in \mathbb{R}^{T_* t \times d_k}$ and $\mathbf{S}_g^* \in \mathbb{R}^{T_* t \times d_k}$. We take the interaction between the query and the history in the local dialogue flow level as an example. Other interaction representations can be obtained in the same way. The interaction layer is presented in Figure 4.

(1) **Two-level cross-attention:** We first calculate the two-level cross-attention matrix between query and response's history: the word representation level cross-attention matrix $\mathbf{M}_1^h \in \mathbb{R}^{T_h t \times T_h t}$ and the dialogue flow aware representation level cross-attention matrix $\mathbf{M}_2^h \in \mathbb{R}^{T_h t \times T_h t}$. Each element of the cross-attention matrix $\mathbf{M}_1^h$ and $\mathbf{M}_2^h$ is represented as:

$$\begin{cases} \mathbf{M}_{l,1,a,b}^h = \{\mathbf{E}^q\}_a^\top \cdot \{\mathbf{E}^h\}_b & \text{word-level} \\ \mathbf{M}_{l,2,a,b}^h = \{\mathbf{S}_l^q\}_a^\top \cdot \{\mathbf{S}_l^h\}_b & \text{dialogue-flow-level} \end{cases} \tag{11}$$

in which $\{\mathbf{E}^q\}_a$ is the a-th row of $\mathbf{E}^q$ and $\{\mathbf{S}_l^h\}_a$ is the a-th row of $\mathbf{S}_l^h$.

(2) **Attentive cross-attention:** The self-attentive module is designed to learn the self-attentive representation and it can also learn the history-aware query representation and query-aware history representation which makes more interaction between them. The attentive query representation $\mathbf{S}_l'^q$ and the attentive history representation $\mathbf{S}_l'^h$ are represented as:

$$\mathbf{S}_l'^q = f_{\text{att}}(\mathbf{S}_l^q, \mathbf{S}_l^h, \mathbf{S}_l^h) \tag{12}$$

$$\mathbf{S}_l'^h = f_{\text{att}}(\mathbf{S}_l^h, \mathbf{S}_l^q, \mathbf{S}_l^q) \tag{13}$$

Then, a new cross-attention matrix can be calculated:

$$\mathbf{M}_{l,3,a,b}^h = \{\mathbf{S}_l'^q\}_a^\top \cdot \{\mathbf{S}_l'^h\}_b \tag{14}$$

(3) **Projection sublayer:** The three cross-attention matrices are then stacked into one matrix $\mathbf{M}_l^h \in \mathbb{R}^{3 \times T_h t \times T_h t}$. Then a 2-layer 2-D CNN is adopted to project the attention matrix $\mathbf{M}_l^h$ to matching features. The output of the CNN are flattened and mapped into low-dimension vector representation $\mathbf{d}_l^h$:

$$\mathbf{M}_l^h = f_{\text{stack}}(\mathbf{M}_{l,1}^h, \mathbf{M}_{l,2}^h, \mathbf{M}_{l,3}^h) \tag{15}$$

$$\mathbf{d}_l^h = f_{\text{flat}}(f_{\text{CNN}}(\mathbf{M}_l^h)) \tag{16}$$

After the interaction layer, we obtain the query-history interaction representation $\mathbf{d}_l^h$. The interaction layer can be abstracted as:

$$\mathbf{d}_l^h = f_{\text{inter}}(\mathbf{S}_l^q, \mathbf{S}_l^h, \mathbf{E}^q, \mathbf{E}^h) \tag{17}$$

In the same way, we can obtain the utterance representation level interaction representation for history and future $(\mathbf{d}_u^h, \mathbf{d}_u^f)$, the local dialogue flow representation level $(\mathbf{d}_l^h, \mathbf{d}_l^f)$, and the global dialogue flow representation level $(\mathbf{d}_g^h, \mathbf{d}_g^f)$, by feeding different inputs to the interaction layer as introduced in the second paragraph of the section "interaction layer".

3.7 Model Training

The six interaction representations $\{\mathbf{d}_u^h, \mathbf{d}_u^f, \mathbf{d}_l^h, \mathbf{d}_l^f, \mathbf{d}_g^h, \mathbf{d}_g^f\}$ are concatenated and fed into an MLP to predict the *query-to-session* matching prediction score p:

$$p = \text{MLP}([\mathbf{d}_u^h, \mathbf{d}_u^f, \mathbf{d}_l^h, \mathbf{d}_l^f, \mathbf{d}_g^h, \mathbf{d}_g^f]). \tag{18}$$

And the cross-entropy loss is calculated by (D represent all the training samples):

$$\mathcal{L} = -\frac{1}{|D|} \sum_{(Q,S,l) \in D} l \log p + (1-l) \log(1-p) \tag{19}$$

Inspired by Tao et al. [26], we fed each of the six interaction representations into an MLP respectively to obtain six matching prediction scores $\{p_u^h, p_u^f, p_l^h, p_l^f, p_g^h, p_g^f\}$. The cross-entropy is also calculated for the six scores. The average of the cross-entropy loss for the six scores is added to the cross-entropy $\mathcal{L}$ to train the models. The final ranking score p_r is calculated by:

$$p_r = p + \frac{\beta p_u^h + \beta p_u^f + \gamma p_l^h + \gamma p_l^f + \gamma p_g^h + \gamma p_g^f}{2\beta + 4\gamma}, \tag{20}$$

where β controls the contribution of utterance level matching scores, and γ controls the dialogue flow level contributions. In our model, β is set to 2 and γ is set to 1 to balance the utterance level matching scores and dialogue flow level matching scores.

Table 1: The number of {query, session} pairs in training, validation and testing set for the three corpora. The vocabulary size is also shown in the Table.

Dataset	# Training	# Validation	# Testing	# Vocab
Ubuntu	1,013,172	102,000	102,210	438,565
Douban	956,918	104,220	104,560	156,159
E-commerce	494,916	120,740	118,020	36,130

4 EXPERIMENT SETUP

In this section, we introduce the dataset construction, the evaluation metrics, the model settings, and the baselines.

4.1 Dataset Construction

In this section, we first introduce the original datasets involved in our experiments and then describe how we modify these original datasets into the corpora we need.

4.1.1 Original Datasets. We compose the query-session pairs based on the conversations from three widely used corpora. The Ubuntu Dialogue Corpus [18] contains multi-turn technical support conversations from the Ubuntu-related chat rooms on the Freenode Internet Relay Chat network[3]. The Douban Conversation Corpus [32] contains open-domain conversations from the Chinese social network Douban[4]. The conversations in E-commerce Dialogue Corpus [37] are between the customers and customer service staffs from Taobao[5] which is the largest e-commerce platform in China.

4.1.2 Dataset Construction. The key question we face is how to construct the whole query-history-response-future (QHRF) pairs from the initial corpora which consist of only query-response pairs. The whole process is described as follows:

(1) Suppose we have a conversation C_a. We randomly select an sentence in C_a as the response r_a. The sentences before the response in C_a are the query q_a. The sentences after the response become the future f_a:
$$C_a = q_a + r_a + f_a$$
(2) However, the aforementioned conversation C_a lacks the history. Because the sentences before the response have been the query. If the utterances before the response also become history, the history and query are the same and the model will only learn to judge if the query and history are the same.
To solve such problem, another kind of QHRF pairs are constructed. Assume that we have another conversation C_b which also contains the response r_a ($r_b = r_a$). The response r_b in C_b is an appropriate response for q_a, so C_b is an appropriate candidate session for q_a. The sentences before r_b in C_b are the response's history h_b while the sentences after r_b in C_b become the future f_b:
$$C_b = h_b + r_b + f_b$$
If there is no conversation C_b which contains the same response with r_a, the case will not be added to the second kind of QHRF pair in Equation 21. The responses which occur once are dropped and the common responses are reserved. To avoid this bias and keep the diversity of the responses, the first kind of the QHRF pair in Equation 21 is reserved for all the cases.
(3) For the $\{q_a, r_a\}$, we have two kinds of query-history-response-future pairs in the above settings:
$$\text{QHRF pairs} = \begin{cases} [q_a, unk, r_a, f_a] & C_a \\ [q_a, h_b, r_b, f_b] & \text{query} + C_b \end{cases} \quad (21)$$
For the first kind of QHRF pairs which lack response's history, we use *unk* to represent the history to train the models.

For each query, we randomly sample another session as the negative sample. As for the validation and testing set, we sample 9 negative samples instead of 1 following previous work [18]. The statistics of the three datasets are shown in Table 1. All datasets will be released for future research.

4.2 Evaluation Metrics

Following Tao et al. [26], Wu et al. [32], we evaluate the models in terms of MRR (Mean Reciprocal Rank) [29], $R_{10}@1$, $R_{10}@2$, $R_{10}@5$, and $R_2@1$, which are all frequently-used metrics in response selection tasks. $R_n@k$ calculates the recall of the true positive responses among the k selected candidates from n available candidates. The MRR first calculates the reciprocal rank in which the reciprocal rank of each query is the multiplicative inverse of the rank of the first correct response. Then the average of reciprocal rank over the whole testing set becomes the MRR score.

4.3 Model Setup

We use Adam [15] optimizer with a learning rate of 0.0001 and a batch size of 100 to optimize the parameters. The exponential decay of 0.9 on the learning rate is applied every 5000 iterations. The word embedding dimension is 200, and we pre-train the word embeddings through GloVe [21] for the corpora separately. The embeddings are tuned during the model training to get better performance. In the future, we will explore to use the contextualized word embeddings to obtain better performance. The vocabulary sizes are shown in Table 1 following the previous work [26, 32]. The max turn numbers of the query, history, and future are all 5 and the max utterance length is 20, which is sufficient to cover most of the turns and words in the corpora. We use padding to handle the various lengths of the text. The best performing checkpoint on the validation set is selected according to $R_{10}@1$ for testing.

In §3.6, we adopt two convolution layers to encode the cross-attention matrix. The stride sizes of the convolution and max-pooling layer are (1,1) and (3,3) respectively. The filter sizes are all (3,3) for the convolution and max-pooling layers. The output channels of the two convolution layers are 32 and 16 respectively.

4.4 Baselines and Models

In this paper, we have the following hypotheses:

Hypothesis I: Our proposed *query-to-session* matching approach generally outperforms conventional query-to-response approaches.

Hypothesis II: Our proposed DF-QSM model is superior to other models in utilizing the candidate session

[3]https://freenode.net/
[4]https://www.douban.com/group/
[5]https://www.taobao.com/

Table 2: Evaluation results. The inference time for each case is also shown in the table to analysis efficiency. The results of DF-QSM are significant with p-value < 0.01 measured by the Student's paired t-test over the QRM models.

Models	Ubuntu Dialogue Corpus					Douban Conversation Corpus					E-commerce Dialogue Corpus					Time
	MRR	$R_{10}@1$	$R_{10}@2$	$R_{10}@5$	$R_2@1$	MRR	$R_{10}@1$	$R_{10}@2$	$R_{10}@5$	$R_2@1$	MRR	$R_{10}@1$	$R_{10}@2$	$R_{10}@5$	$R_2@1$	(ms)
Comparison between our DF-QSM model and baselines																
SMN-QRM	0.6530	0.5572	0.7196	0.9204	0.8646	0.7275	0.6536	0.7963	0.9530	0.9101	0.7543	0.6872	0.8607	0.9776	0.9327	41
DAM-QRM	0.7003	0.6196	0.7688	0.9376	0.8921	0.7459	0.6776	0.8103	0.9564	0.9155	0.7662	0.7023	0.8654	0.9786	0.9381	77
IoI-QRM	0.7275	0.6530	0.7957	0.9500	0.9070	0.7529	0.6860	0.8199	0.9626	0.9204	0.8088	0.7566	0.9003	0.9858	0.9518	100
DF-QSM (ours)	**0.8105**	**0.7588**	**0.8770**	**0.9705**	**0.9365**	**0.8396**	**0.7954**	**0.9033**	**0.9819**	**0.9540**	**0.8192**	**0.7695**	**0.9050**	**0.9875**	**0.9545**	103
Baselines that applied with **OUR QSM** framework																
BiMPM-QSM	0.7810	0.7213	0.8286	0.9378	0.9098	0.8213	0.7717	0.8843	0.9768	0.9446	0.7730	0.7102	0.8521	0.9685	0.9265	138
DAM-QSM	0.7932	0.7370	0.8346	0.9357	0.9097	0.8272	0.7796	0.8821	0.9704	0.9430	0.7759	0.7156	0.8599	0.9630	0.9269	272
IoI-QSM	0.8064	0.7537	0.8499	0.9392	0.9165	0.8249	0.7765	0.8771	0.9680	0.9430	0.7918	0.7348	0.8677	0.9670	0.9341	581
Ablation study: session ablation																
DF-QSM Base	0.6916	0.6075	0.7611	0.9356	0.8869	0.7523	0.6843	0.8175	0.9608	0.9167	0.7865	0.7284	0.8807	0.9827	0.9438	24
DF-QSM w/o F	0.7097	0.6312	0.7797	0.9381	0.8920	0.7617	0.6967	0.8297	0.9594	0.9220	0.7889	0.7309	0.8824	0.9820	0.9426	40
DF-QSM w/o H	0.7884	0.7299	0.8462	0.9584	0.9263	0.8194	0.7694	0.8836	0.9792	0.9475	0.8059	0.7531	0.8954	0.9842	0.9503	39
Ablation study: dialogue flow ablation																
DF-QSM w/o DF	0.8024	0.7480	0.8686	0.9668	0.9351	0.8217	0.7739	0.8908	0.9793	0.9491	0.8100	0.7589	0.9001	0.9852	0.9532	37
DF-QSM w/o GDF	0.8087	0.7560	0.8695	0.9674	0.9349	0.8320	0.7863	0.9010	0.9811	0.9527	0.8145	0.7652	0.8995	0.9874	0.9544	64
DF-QSM w/o LDF	0.8025	0.7488	0.8732	0.9671	0.9347	0.8336	0.7879	0.8972	0.9845	0.9530	0.8152	0.7641	0.9033	0.9866	0.9532	82

information in our proposed *query-to-session* approach.

Hypothesis III: The components (local dialogue flow, global dialogue flow) in our proposed DF-QSM model all contribute to the QSM's performance.

Hypothesis IV: Both the history and the future help the response selection task.

To verify the aforementioned hypotheses, we consider five types of baselines and models:

(a) *The SOTA models in the query-to-response matching approach.* SMN [32] is designed for the response selection task. DAM [40] is a strong baseline for the QRM task. IoI [26] is the state-of-the-art model of the QRM task. We adopt them as the query-to-response matching baselines. They are represented as SMN-QRM, DAM-QRM, and IoI-QRM. To make the results reliable, we use the official code released by the authors.

(b) *The strong text-matching models but applied with our query-to-session matching approach.* DAM-QSM and IoI-QSM concatenate the history, response, and future into one sentence as a fake response. Then the query and the fake response are fed into DAM or IoI to predict the matching score. Different from IoI-QSM which only concatenates the session into one utterance, BiMPM-QSM also concatenates the whole query into one utterance. In this way, the sentence-to-sentence matching models can be applied to the two utterances. Here, we feed the two concatenated utterances into bilateral multi-perspective matching (BiMPM) [31] model to predict the matching score. BiMPM has shown great success in sentence-to-sentence text matching tasks. It mainly utilizes Bidirectional Long Short Term Memory (BiLSTM) [12] to learn the sentence representations and aggregate the cross matching representations.

(c) *Our full-version proposed dialogue flow aware query-to-session matching model, represented as DF-QSM.*

(d) *Our ablated-version DF-QSM model to analyze the Dialogue Flow component (model ablation).* "DF-QSM w/o LDF" and "DF-QSM w/o GDF" omit the local and global dialogue flow respectively. "DF-QSM w/o DF" means the whole dialogue flow is omitted.

(e) *Our ablated-version DF-QSM model without history or future, denoted as "DF-QSM w/o H" and "DF-QSM w/o F" (session ablation).* Besides, "DF-QSM Base" works as the base component where both the future and history are omitted and only the response is considered in the candidate session.

The comparisons between (a) and (c) are to contrast the query-to-response approach against the *query-to-session* approach (**Hypothesis I**). The comparison results between (b) and (c) are to check **Hypothesis II**. The ablations (d) and (e) are designed to check the **Hypothesis III** and **Hypothesis IV** respectively.

5 RESULTS AND ANALYSIS

In this section, we first introduce the model results and ablation studies to demonstrate the performance of our proposed model in §5.1. To evaluate the efficiency of our proposed QSM framework and manifest the industrial prospect of our DF-QSM models, we present the efficiency experiment results in §5.2. After that, we introduce the analysis of the dialogue flow updating weight (§5.3) and the size of the candidate session (§5.4) for better comprehension of the proposed QSM framework and DF-QSM model.

5.1 Results and Ablation Studies

We conduct experiments on three datasets and the results are shown in Table 2. Generally, models applied with our *query-to-session* strategies significantly outperform their query-to-response counterparts. Additionally, the ablation study (including model ablation and session ablation) also verifies the effectiveness and necessity of each component in our model. We present the detailed analyses in the order of the hypotheses in §4.4.

*5.1.1 Comparison between QSM and QRM: **Hypothesis I**.* Compared to the QRM baselines, the models applied with our QSM strategy (BiMPM-QSM, DAM-QSM, IoI-QSM, DF-QSM) show great

improvement over the strong QRM model IoI, especially on the Ubuntu and Douban corpora. The DF-QSM also gains 0.01 improvement of $R_{10}@1$ on the E-commerce dataset. The results are strong evidence to prove the superiority of *query-to-session* matching strategy over the conventional query-to-response strategy.

5.1.2 Comparison between DF-QSM and Existing Models Adapted in Our QSM Framework: ***Hypothesis II****.* Compared with the baselines adapted into our QSM framework (DAM-QSM, IoI-QSM, BiMPM-QSM), which are strong baselines for the QSM framework, our DF-QSM model achieves SOTA on all of the metrics. It shows that our proposed dialogue flow aware representations can handle the intractable matching problem between sequences of utterances well and be helpful for the *query-to-session* matching.

5.1.3 Model Ablation: ***Hypothesis III****.* We conduct the model ablation to investigate the effect of the local dialogue flow strategy and the global dialogue flow strategy. The comparisons among {DF-QSM, QSM w/o GLF, QSM w/o LDF, QSM w/o DF} show the enhancement brought by our dialogue flow strategies. Furthermore, it shows that the local and global dialogue flow strategies work together to obtain the best performance.

5.1.4 Session Ablation: ***Hypothesis IV****.* Except for the comparisons between QRM and QSM, which verifies the effectiveness of the whole session, we also conduct the session ablation to study the effectiveness of history and future. DF-QSM considers all the history, response, and future. "DF-QSM Base" only considers the response. "QSM w/o H" and "QSM w/o F" omit the history and future respectively. The performance of DF-QSM drops significantly when omitting the history or future. The results on the four models show that the history and future are all helpful for the response selection. It also indicates the way to construct the query-history-response-future makes sense. Besides, we can observe that the future is more useful than the history, the reason lies in that not all the QHRF pairs contain history, some of them are replaced by *unk* as in §4.1.

5.2 Model Efficiency

To evaluate the efficiency of our proposed methods, we present the average inference time of each case for all the involved models in Table 2. The efficiency experiment is conducted on the same Nvidia P100 GPU for fairness. From the experimental results, we have the following observations: (1) Compared with QRM-based baselines, our proposed model DF-QSM gains great improvement measured by different evaluation metrics. Additionally, the inference time of DF-QSM is comparable with the strong QRM baseline IoI-QRM. Although the inference time of SMN-QRM is the lowest among the QRM models, its performance is unsatisfactory. (2) Compared with other baselines applied with our QSM framework (BiMPM-QSM, DAM-QSM, and IoI-QSM), our proposed DF-QSM not only achieves the best performance, but also uses the shortest time. The efficiency improvement could be attributed to the attentive block and memory network in our DF-QSM model, which are efficient against the RNN-based models.

5.3 Memory Updating Weight Analysis

In the dialogue flow layer, the memory updating weight α (see in Equation 8) controls how much information will be added into or deleted from the dialogue flow memory. This weight reflects the influence of the current utterance to the *query-to-session* matching task. The larger α is, the more important role this utterance plays in the *query-to-session* matching task. We compute the average of absolute memory updating weights across the testing set for each turn in the query, history, and future respectively. The results are shown in Figure 5. When modeling the dialogue flow, the response is added to the history and future as in §3.5.2, so the history and future contain 6 turns while the query contains 5 turns. The following findings can be summarized from the Figure: (1) Generally, the farther the utterance to the response, the less this utterance contributes to the QSM matching task. (2) Both the history and query indicate what has been asked. However, the future represents what will happen. So the tendency of context and history are quite similar, especially for the Ubuntu and Douban corpora. (3) For the Douban corpus, the weights become flattening after the third turn. We assume the reason is that the Douban corpus contains open-domain conversations. The open-domain conversations are usually more casual and contain multiple topics in a dialogue session. So the utterances which are far from the response contribute much less than the near ones.

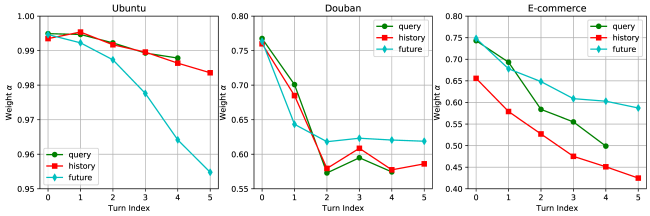

Figure 5: Average of memory updating weights α for each turn. For each utterance, the farther the utterance is from the response, the bigger its turn index is.

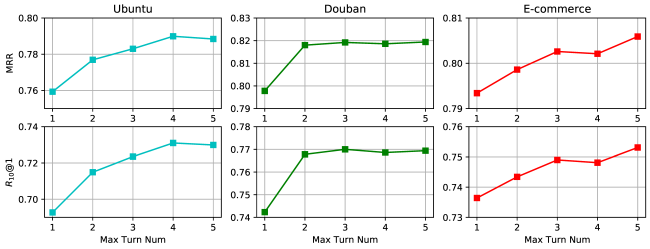

Figure 6: The performance with different session sizes. Without loss of generality, we use the future size to illustrate the effects of session size.

5.4 Session Size Analysis

As shown in the previous sections, we have proven that the history and the future do benefit to the response selection task. Next, another key question arises: *How large a session do we need?* In general, a large session may introduce not only useful information but also noise. Without loss of generality, we only analyze the results under different future size to explore how large a session do we need. The results are shown in Figure 6. We can find that the performance

increases as the turn number increases, especially for the Ubuntu corpus and E-commerce corpus. As for the Douban corpus, it becomes almost flattening after the second turn. The reason may be that the Douban corpus contains more open-domain conversation in which the topics change a lot with the dialogue going on. We can see from the results that a too large session will not introduce more performance improvement.

6 CONCLUSION

In this paper, we presented the *query-to-session* matching approach for response selection in multi-turn retrieval-based dialogue systems, which transcends the *query-to-response* matching counterparts significantly. We were surprised by the extent of the improvement brought by the history and the future of response in the session. Furthermore, our proposed local and global dialogue flow strategies provide graceful and efficient strategies to precisely integrate the utterance information into the memory network. The experimental results verify the superiority brought by our proposed DF-QSM models.

7 ACKNOWLEDGMENTS

We would like to thank the anonymous reviewers for their constructive comments. This work was supported by the National Key R&D Program of China (No. 2017YFC0804001), the National Natural Science Foundation of China (NSFC No. 61876196 and NSFC No. 61672058). Rui Yan is partially supported as a Young Fellow of Beijing Institute of Artificial Intelligence (BAAI). This work was supported by Alibaba Group through Alibaba Research Fellowship Program.

REFERENCES

[1] Mohammad Aliannejadi, Manajit Chakraborty, Esteban Andrés Ríssola, and Fabio Crestani. 2020. Harnessing evolution of multi-turn conversations for effective answer retrieval. In *CHIIR*. 33–42.
[2] Mohammad Aliannejadi, Hamed Zamani, Fabio Crestani, and W Bruce Croft. 2019. Asking clarifying questions in open-domain information-seeking conversations. In *SIGIR*. 475–484.
[3] Qian Chen, Xiaodan Zhu, Zhen-Hua Ling, Si Wei, Hui Jiang, and Diana Inkpen. 2017. Enhanced LSTM for Natural Language Inference. In *ACL*. 1657–1668.
[4] Jeffrey Dalton, Chenyan Xiong, and Jamie Callan. 2019. Cast 2019: The conversational assistance track overview. In *TREC*. 13–15.
[5] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *NAACL*. Association for Computational Linguistics, Minneapolis, Minnesota, 4171–4186.
[6] Zhenxin Fu, Shaobo Cui, Mingyue Shang, Feng Ji, Dongyan Zhao, Haiqing Chen, and Rui Yan. 2020. Context-to-Session Matching: Utilizing Whole Session for Response Selection in Information-Seeking Dialogue Systems. In *KDD*.
[7] Yichen Gong, Heng Luo, and Jian Zhang. 2017. Natural language inference over interaction space. *arXiv preprint arXiv:1709.04348* (2017).
[8] Alex Graves, Greg Wayne, and Ivo Danihelka. 2014. Neural turing machines. *arXiv preprint arXiv:1410.5401* (2014).
[9] Jia-Chen Gu, Zhen-Hua Ling, and Quan Liu. 2019. Interactive Matching Network for Multi-Turn Response Selection in Retrieval-Based Chatbots. In *CIKM*. ACM, 2321–2324.
[10] Hua He, Kevin Gimpel, and Jimmy Lin. 2015. Multi-perspective sentence similarity modeling with convolutional neural networks. In *EMNLP*. 1576–1586.
[11] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *CVPR*. 770–778.
[12] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long Short-term Memory. *Neural computation* 9 (12 1997), 1735–80.
[13] Hsin-Yuan Huang, Eunsol Choi, and Wen-tau Yih. 2018. Flowqa: Grasping flow in history for conversational machine comprehension. *arXiv preprint arXiv:1810.06683* (2018).
[14] Po-Sen Huang, Xiaodong He, Jianfeng Gao, Li Deng, Alex Acero, and Larry Heck. 2013. Learning deep structured semantic models for web search using clickthrough data. In *CIKM*. ACM, 2333–2338.
[15] Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).
[16] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. 2016. Layer normalization. *arXiv preprint arXiv:1607.06450* (2016).
[17] Feng-Lin Li, Minghui Qiu, Haiqing Chen, Xiongwei Wang, Xing Gao, Jun Huang, Juwei Ren, Zhongzhou Zhao, Weipeng Zhao, Lei Wang, et al. 2017. AliMe assist: an intelligent assistant for creating an innovative e-commerce experience. In *CIKM*. ACM, 2495–2498.
[18] Ryan Lowe, Nissan Pow, Iulian Serban, and Joelle Pineau. 2015. The Ubuntu Dialogue Corpus: A Large Dataset for Research in Unstructured Multi-Turn Dialogue Systems. In *SIGDIAL*. 285–294.
[19] Wentao Ma, Yiming Cui, Nan Shao, Su He, Wei-Nan Zhang, Ting Liu, Shijin Wang, and Guoping Hu. 2019. TripleNet: Triple Attention Network for Multi-Turn Response Selection in Retrieval-Based Chatbots. In *CoNLL*. Association for Computational Linguistics, Hong Kong, China, 737–746.
[20] Fandong Meng, Zhengdong Lu, Hang Li, and Qun Liu. 2016. Interactive Attention for Neural Machine Translation. In *Proceedings of COLING 2016*. 2174–2185.
[21] Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *EMNLP*. 1532–1543.
[22] Chen Qu, Liu Yang, W Bruce Croft, Johanne R Trippas, Yongfeng Zhang, and Minghui Qiu. 2018. Analyzing and characterizing user intent in information-seeking conversations. In *SIGIR*. ACM, 989–992.
[23] Lifeng Shang, Zhengdong Lu, and Hang Li. 2015. Neural Responding Machine for Short-Text Conversation. In *ACL-IJCNLP*. ACL, Beijing, China, 1577–1586.
[24] Sainbayar Sukhbaatar, Jason Weston, Rob Fergus, et al. 2015. End-to-end memory networks. In *Advances in neural information processing systems*. 2440–2448.
[25] Chongyang Tao, Wei Wu, Can Xu, Wenpeng Hu, Dongyan Zhao, and Rui Yan. 2019. Multi-Representation Fusion Network for Multi-Turn Response Selection in Retrieval-Based Chatbots. In *WSDM*. ACM, 267–275.
[26] Chongyang Tao, Wei Wu, Can Xu, Wenpeng Hu, Dongyan Zhao, and Rui Yan. 2019. One Time of Interaction May Not Be Enough: Go Deep with an Interaction-over-Interaction Network for Response Selection in Dialogues. In *ACL*. Association for Computational Linguistics, Florence, Italy, 1–11.
[27] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*. 5998–6008.
[28] Oriol Vinyals and Quoc Le. 2015. A neural conversational model. *arXiv preprint arXiv:1506.05869* (2015).
[29] Ellen M Voorhees et al. 1999. The TREC-8 question answering track report. In *Trec*, Vol. 99. 77–82.
[30] Heyuan Wang, Ziyi Wu, and Junyu Chen. 2019. Multi-Turn Response Selection in Retrieval-Based Chatbots with Iterated Attentive Convolution Matching Network. In *CIKM*. ACM, 1081–1090.
[31] Zhiguo Wang, Wael Hamza, and Radu Florian. 2017. Bilateral multi-perspective matching for natural language sentences. In *IJCAI*. AAAI Press, 4144–4150.
[32] Yu Wu, Wei Wu, Chen Xing, Ming Zhou, and Zhoujun Li. 2017. Sequential Matching Network: A New Architecture for Multi-turn Response Selection in Retrieval-Based Chatbots. In *ACL*. 496–505.
[33] Liu Yang, Minghui Qiu, Chen Qu, Jiafeng Guo, Yongfeng Zhang, W Bruce Croft, Jun Huang, and Haiqing Chen. 2018. Response ranking with deep matching networks and external knowledge in information-seeking conversation systems. In *SIGIR*. ACM, 245–254.
[34] Tom Young, Erik Cambria, Iti Chaturvedi, Hao Zhou, Subham Biswas, and Minlie Huang. 2018. Augmenting end-to-end dialogue systems with commonsense knowledge. In *Thirty-Second AAAI Conference on Artificial Intelligence*.
[35] Jianfei Yu, Minghui Qiu, Jing Jiang, Jun Huang, Shuangyong Song, Wei Chu, and Haiqing Chen. 2018. Modelling domain relationships for transfer learning on retrieval-based question answering systems in e-commerce. In *WSDM*. ACM, 682–690.
[36] Chunyuan Yuan, Wei Zhou, Mingming Li, Shangwen Lv, Fuqing Zhu, Jizhong Han, and Songlin Hu. 2019. Multi-hop Selector Network for Multi-turn Response Selection in Retrieval-based Chatbots. In *EMNLP-IJCNLP*. Association for Computational Linguistics, Hong Kong, China, 111–120.
[37] Zhuosheng Zhang, Jiangtong Li, Pengfei Zhu, Hai Zhao, and Gongshen Liu. 2018. Modeling Multi-turn Conversation with Deep Utterance Aggregation. In *COLING*. ACL, Santa Fe, New Mexico, USA, 3740–3752.
[38] Li Zhou, Jianfeng Gao, Di Li, and Heung-Yeung Shum. 2018. The design and implementation of XiaoIce, an empathetic social chatbot. *arXiv preprint arXiv:1812.08989* (2018).
[39] Xiangyang Zhou, Daxiang Dong, Hua Wu, Shiqi Zhao, Dianhai Yu, Hao Tian, Xuan Liu, and Rui Yan. 2016. Multi-view response selection for human-computer conversation. In *EMNLP*. 372–381.
[40] Xiangyang Zhou, Lu Li, Daxiang Dong, Yi Liu, Ying Chen, Wayne Xin Zhao, Dianhai Yu, and Hua Wu. 2018. Multi-turn response selection for chatbots with deep attention matching network. In *ACL*. 1118–1127.