Modeling Multi-turn Conversation with Deep Utterance Aggregation

Zhuosheng Zhang[1,2,*], **Jiangtong Li**[1,2,3,*], **Pengfei Zhu**[1,2,5], **Hai Zhao**[1,2,†], **Gongshen Liu**[4]

[1]Department of Computer Science and Engineering, Shanghai Jiao Tong University
[2]Key Laboratory of Shanghai Education Commission for Intelligent Interaction
and Cognitive Engineering, Shanghai Jiao Tong University, Shanghai, 200240, China
[3]College of Zhiyuan, Shanghai Jiao Tong University, China
[4]School of Cyber Security, Shanghai Jiao Tong University, China
[5]School of Computer Science and Software Engineering, East China Normal University, China
{zhangzs, keep_moving-lee}@sjtu.edu.cn, 10152510190@stu.ecnu.edu.cn,
zhaohai@cs.sjtu.edu.cn, lgsheng@sjtu.edu.cn

Abstract

Multi-turn conversation understanding is a major challenge for building intelligent dialogue systems. This work focuses on retrieval-based response matching for multi-turn conversation whose related work simply concatenates the conversation utterances, ignoring the interactions among previous utterances for context modeling. In this paper, we formulate previous utterances into context using a proposed deep utterance aggregation model to form a fine-grained context representation. In detail, a self-matching attention is first introduced to route the vital information in each utterance. Then the model matches a response with each refined utterance and the final matching score is obtained after attentive turns aggregation. Experimental results show our model outperforms the state-of-the-art methods on three multi-turn conversation benchmarks, including a newly introduced e-commerce dialogue corpus.

1 Introduction

Human-computer interactive systems are booming due to their promising potentials and alluring commercial values (Qiu et al., 2017; Cui et al., 2017; Yan et al., 2017; Huang et al., 2018; Jia and Zhao, 2014). With the development of neural models (Zhang et al., 2018c; He et al., 2018; Li et al., 2018; Cai et al., 2018; Zhang and Zhao, 2018), building an intelligent dialogue system as our personal assistant or chat companion, is no longer a fantasy, among which multi-turn natural language understanding still keeps extremely challenging, requiring the system to comprehend the conversation context and reply in an informative and coincident manner.

Multi-turn conversation modeling plays a key role in dialogue systems, either for generation-based (Serban et al., 2017b; Serban et al., 2017a; Zhou et al., 2017; Wu et al., 2018) or retrieval-based ones (Wu et al., 2017; Zhou et al., 2016) in which the latter is the focus of this paper. A natural approach for multi-turn modeling is simply concatenating the context utterances (Lowe et al., 2015; Yan et al., 2016). However, this will introduce much noise since previous utterances as the context is lengthy and redundant. The gist is to identify pertinent information in previous utterances and properly model the utterance relationships to ensure conversation consistency. To avoid unnecessary information loss, (Wu et al., 2017) matches a response with each utterance in the context, paying little attention on distrinct importance of each utterance and also failing to touch internal semantics inside utterances.

In fact, the relevance of each utterance to the supposed response usually varies. As shown in Figure 1, the last utterance in a conversation empirically conveys the user intention while the other utterances depict the conversation in different aspects [1]. Thus, instead of considering all the conversation turns

---

[1]For a multi-turn conversation, we define the latest user utterance (or called current message) as the *last utterance*, which is waiting for a response.
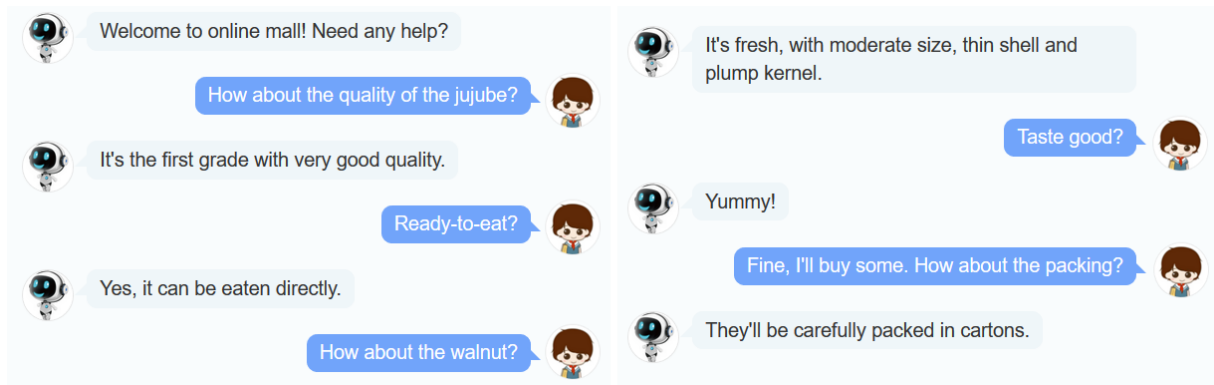
turns-awareaggregation design

Figure 1: An example of E-commerce Dialogue Corpus.

equally, we have to weigh previous conversations in a more sophisticated way. With a turns-aware aggregation design, our model alleviates the drawback of previous work.

In addition, words in an utterance also hold different importance to the whole utterance representation. Our solution is to employ attention-based recurrent networks on each utterance against utterance itself, aggregating the vital pieces of the contextual utterances.

Finally, in conjunction with this paper, we release an E-commerce Dialogue Corpus (ECD) to facilitate the related studies. To our best knowledge, this is the first public e-commerce dataset for dialogue system development that is extracted from real human conversations. Different from previous datasets that only focus on a single type of dialogue like chitchat, this dataset is more comprehensive due to diverse types of conversations (e.g. *commodity consultation, logistics express, recommendation, negotiation* and *chitchat*) concerning various commodities. Our improved retrieval-based multi-turn dialogue response matching model is evaluated on three benchmark datasets, including our newly released one, giving state-of-the-art performance.

The rest of this paper is organized as follows. The next section reviews related work. Our proposed model is introduced in Section 3, then the experiments and analysis are reported in Section 4, followed by the conclusion in Section 5.

2 Related Work

With the impressive success of various referential natural language processing studies (Zhang et al., 2016; Cai and Zhao, 2017; Zhang et al., 2018d; Qin et al., 2017; Zhang et al., 2018b; Bai and Zhao, 2018), developing an intelligent dialogue system becomes realizable, which means training machines to converse with human in natural languages (Williams et al., 2017; He et al., 2017; Dhingra et al., 2017; Zhang et al., 2018a). Towards this end, a number of data-driven dialogue systems are designed (Lowe et al., 2015; Wu et al., 2017; Wen et al., 2017; Mei et al., 2017; Young et al., 2018; Lipton et al., 2018), in which modeling multi-turn conversation has drawn more and more attention. To acquire a contextual response, previous utterances are taken as input. Lowe et al. (2015) concatenated all previous utterances and last utterance as the context representation and then computed the matching degree score based on the context representation to encode candidate response. Yan et al. (2016) selected the previous utterances in different strategies and combined them with last utterance to form a reformulated context. Zhou et al. (2016) performed context-response matching with a multi-view model on both word level and utterance level. Wu et al. (2017) improved the leveraging of utterances relationship and contextual information by matching a response with each utterance in the context based on a convolutional neural network.

Different from previous studies, our model for the first time discriminates the importance of previous conversations and also accumulates substantial parts from each utterance according to each word in the utterance itself in a multi-turn scenario.
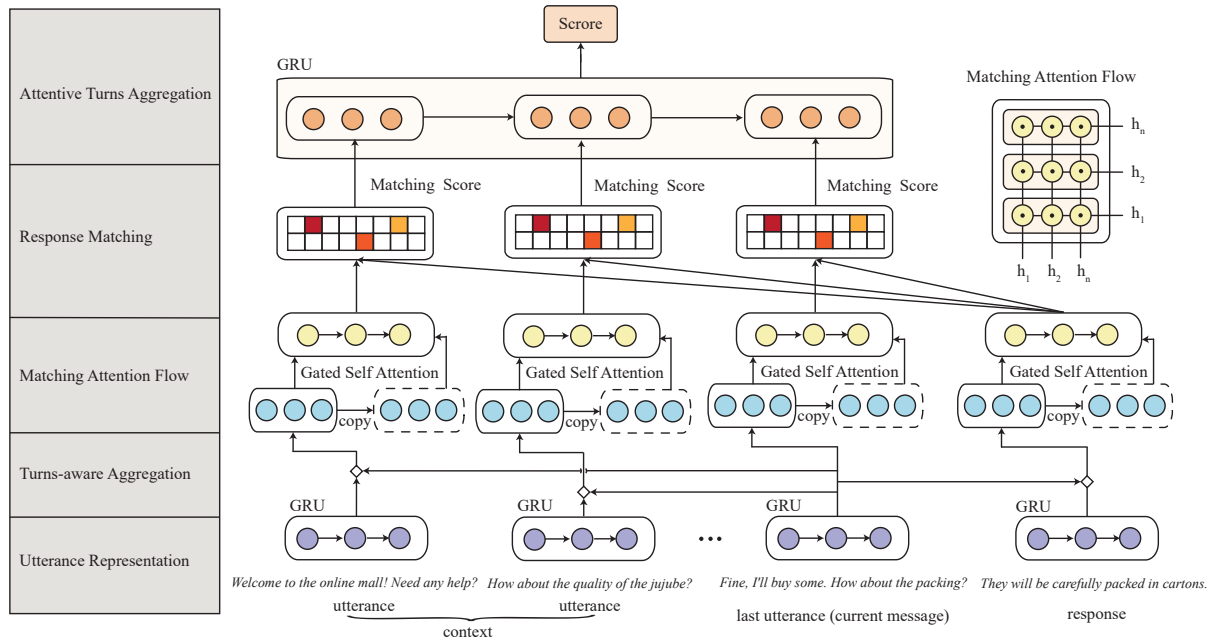
3741

Figure 2: Structure overview of the proposed dialogue system.

3 Deep Utterance Aggregating Strategy

Each conversation in the concerned multi-turn response retrieval task can be described as a triple $< C, R, Y >$. $C = \{U_1, ..., U_t\}$ is the conversation context where $\{U_k\}$ denotes the k-th utterance. R is a response of the conversation and Y belongs to $\{0, 1\}$, where $Y_i = 1$ means the response is proper, otherwise $Y_i = 0$. The aim is to build a discriminator $\mathcal{F}(\cdot, \cdot)$ on $< C, R, Y >$. For each context-response pair $\{C, R\}$, $\mathcal{F}(C, R)$ measures the matching score of the pair.

In this section, we will introduce our Deep Utterance Aggregation (DUA) model for the multi-turn conversation task. Figure 2 shows the architecture. DUA formulizes utterances into the context and mines the key information from utterances and response. Then DUA conducts semantic matching between each utterance and the response candidate to obtain a matching score. Specifically, there are five modules within DUA. Each utterance or response is fed to the first module to form an utterance or response embedding. The second module combines the last utterance with the preceding utterances. Then, the third module filters the redundant information and mines the salient feature within the utterances and response. The fourth module matches the response and each utterance at both word and utterance levels to feed a Convolutional Neural Network (CNN) for encoding into matching vectors. In the last module, the matching vectors are delivered to a gated recurrent unit (GRU) (Cho et al., 2014) in chronological order of the utterances in the context and the final matching score of $\{U, R\}$ is obtained.

DUA is superior to existing models in the following ways. First, the last utterance which is the most important in dialogue is especially fused within preceding utterances, thus the key guideline information from the last utterance can be handled in a more semantically pertinent way. Second, in each utterance, the salient information can be highlighted and those redundant pieces will be neglected to some extent, both of which can effectively guide the later response matching. Third, after attentive turns aggregation, the connections in the conversation are accumulated again to calculate the matching scores.

3.1 Utterance Representation

To use deep neural networks, symbolic data needs to be transformed into distributed representations, namely, word embedding (Bengio et al., 2003; Mikolov et al., 2013). Given a context-response pair, $\{C, R\}$ whose context are split into utterances, $C = \{U_1, ..., U_t\}$, a lookup table is used to map each word into a low-dimensional vector. Let n_u and n_r denote the length of the k-th utterance and response, U_k and R can be represented as $U_k = [u_1, ..., u_{n_u}]$ and $R = [r_1, ..., r_{n_r}]$, where u_i, r_i are the i-th word

in the utterance and response respectively.

To encode each utterance and response, we employ a GRU to propagate information along the word sequence of U_k and R. Suppose $H_k = [h_1, ..., h_n]$ is the hidden states of the input sequence, the structure of GRU is described as follows.

$$
\begin{aligned}
z_i &= \sigma(W_z u_i + V_z h_{i-1}) \\
r_i &= \sigma(W_r u_i + V_r h_{i-1}) \\
\tilde{h}_i &= tanh(W_h u_i + V_h(r_i \odot h_{i-1})) \\
h_i &= z_i \odot \tilde{h}_i + (1 - z_i) \odot h_{i-1}
\end{aligned}
\tag{1}
$$

where $\sigma(\cdot)$ is the sigmoid function, z_i and r_i are the update and reset gates respectively, $\odot$ denotes the element-wise multiplication, and $W_z, W_r, W_h, V_z, V_r, V_h$ are parameters. We fed each utterance and response sequence to the GRUs and obtain the utterance representation S_k and response representation S_r, respectively.

3.2 Turns-aware Aggregation

Encoding the utterance sequence and response in the above way, there comes a drawback that all the utterances in the conversation are fairly dealt with, which fails to mine the connections between the last utterance and the rest preceding utterances. Thus, a first-stage turns-aware aggregation mechanism is proposed to address this problem.

Let $S = [S_1, ..., S_t, S_r]$ denote the representation of the utterances and response. Suppose $F = [F_1, ..., F_t, F_r]$ is the fusion of each $S_j \in S$ with the last utterance S_t, for each $\forall j \in \{1, ..., r\}$, $F_j \in F$, we define the fusion of the utterance as

$$
F_j = S_j \diamond S_t
\tag{2}
$$

where $\diamond$ denotes the aggregation operation. In this work, we adopt a simple concatenation strategy[2]. So far, the turns-aware representation F is obtained via aggregation.

3.3 Matching Attention Flow

After turns-aware aggregation, the representations of the preceding utterances and response have been refined by the last utterance. However, the sequences are quite lengthy and redundant, which makes it hard to distill the pivotal information. In order to address this problem, we adopt a self-matching attention mechanism to directly match the fused representation against itself, which is similar as that adopted in (Wang et al., 2017). It dynamically collects information from the input sequence and filters the redundant information. Suppose $\hat{F} = [f_1, ..., f_n] \in F$ is the input and $P = [p_1, ..., p_n]$ is the output of the self-matching attention on response, then $\forall t$, $p_t \in P$ is defined as

$$
p_t = GRU(p_{t-1}, [f_t, c_t]) \qquad \text{self-attention} \qquad \text{GRU}
\tag{3}
$$

where $GRU(\cdot, \cdot)$ denotes the same calculation as (2), $[\cdot, \cdot]$ is the concatenation of two vectors and $c_t = att(\hat{F}, f_t)$ is the result of the self-matching attention. $\forall t$, $f_t \in \hat{F}$, c_t is defined as

$$
\begin{aligned}
s_j^t &= v^T tanh(W_v f_j + W_{\tilde{v}} f_t + b_r) \quad \text{f\_t} \\
&\qquad\qquad\qquad\qquad\qquad\quad \text{self-attention} \quad \text{c\_t} \qquad\qquad\qquad \text{GRU} \\
&\qquad\qquad\qquad\qquad\qquad\quad \text{p\_t} \quad \text{GRU} \\
a_i^t &= exp(s_i^t) / \sum_{j=1}^{n} exp(s_j^t) \\
c_t &= \sum_{i=1}^{n} a_i^t f_i
\end{aligned}
\tag{4}
$$

---

[2]We empirically investigated *concatenation*, *element-wise summation*, *element-wise multiplication* in this work and *concatenation* strategy shows the best performance.

where W_r, $W_{\tilde{v}}$, b_r are the parameters and v^T is a context matrix which is randomly initialized and jointly trained.

Self-matching attention pinpoints important parts from the utterance according to the current word and the whole utterance representation through fusing each previous utterance and the last utterance.

3.4 Response Matching

Following (Wu et al., 2017), we use word-level and utterance-level representations to build two matching matrices and employ CNN to obtain salient matching information from the matrices. Suppose we have matching matrices M_1 and M_2 in word-level and utterance-level for each utterance-response pair. Then, $\forall k$, $U_k \in U$ and $\forall (i,j)$, the (i,j)-th element of M_1 and M_2 is defined respectively

$$e_{1,i,j} = u_i^T r_j \tag{5}$$

$$e_{2,i,j} = P_{u_i}^T A P_{r_j} \qquad \text{self match attention} \tag{6}$$

where P_{u_i} and P_{r_j} denote the outputs of the utterance and response after *Matching Attention Flow* respectively. $A \in \mathbb{R}^{c \times c}$ is a linear transforming matrix.

A convolutional operation followed by a max-pooling operation will be applied to M_1 and M_2 for each utterance. The convolutional layer is used to extract and combine local features from adjacent words and the following max-pooling layer forms the representations for the current word. For the convolutional operation, a group of filter matrices K with variable sizes $l*l$ and bias b are utilized. The filter transforms the word matrices M_1 and M_2 to another two matrices M_{1c} and M_{2c}. $\forall i \ k \in (1,2)$, the transformed matrices M_{kc} is define as:

$$M_{kc,[i][j]} = ReLU \left(\sum_{i}^{i+l-1} \sum_{j}^{j+l-1} K \cdot M_{k,[i:i+l-1][j:j+l-1]} + b \right) \tag{7}$$

where i and j index the row i-th and column j-th element, respectively. Next, a max-pooling operation is adopted and the representation m_p for p-th utterance in a conversation is obtained through flattening and concatenating the two matrices after pooling as follows:

$$\hat{m}_{k,[i][j]} = \max(M_{kc,[i:i+l-1][j:j+l-1]}) \tag{8}$$

$$m_p = [flatten(\hat{m}_1) \oplus flatten(\hat{m}_2)] \tag{9}$$

where $flatten()$ is the flatten operation and $\oplus$ is the concatenation operation.

3.5 Attentive Turns Aggregation

To aggregate the matching information of the attentive turns in the last stage, The outputs of CNN, $M = [m_1, ..., m_n]$ are fed to GRU to obtain $H_m = [h_{m_1}, ..., h_{m_n}]$. $\forall i$, $h_{m,i} \in H_m$ is defined as

$$h_{m,i} = GRU(h_{m_{i-1}}, m_i) \tag{10}$$

where $GRU(\cdot, \cdot)$ denotes the same calculation and parameterization as Eq.(2). Suppose $v_f = L(H_m)$ is the attention operation which is defined as:

$$t_i = v^T \tanh(W_t P_{u_i} + V_t h_{m_i} + b) \qquad \text{P\_u} \qquad \text{h\_m}$$

$$\alpha_i = \exp(t_i) / \sum_{j=1}^{n} \exp(t_j) \tag{11}$$

$$L(H_m) = \sum_{i=1}^{n} \alpha_i h_{m_i}$$

where W_t, V_t and b are parameters. With v_f, we define $\mathcal{F}(U, R)$ as:

$$\mathcal{F}(U, R) = softmax(W_s v_f) \tag{12}$$

3744

	Ubuntu			Douban			ECD		
	Train	Valid	Test	Train	Valid	Test	Train	Valid	Test
# context-response pairs	1M	500K	500K	1M	50K	10K	1M	10K	10K
# candidates per context	2	10	10	2	2	10	2	2	10
Avg # turns per context	10.13	10.11	10.11	6.69	6.75	6.45	5.51	5.48	5.64
Avg # words per utterance	11.35	11.34	11.37	18.56	18.50	20.74	7.02	6.99	7.11

Table 1: Data statistics template for latter use.

where W_s is the parameter. During the training phase, model parameters are updated according to a cross-entropy loss.

Note that *Turns-aware Aggregation* and *Attentive Turns Aggregation* can be seen as two stages of inter-action across the utterances (we call all these two process as "*Context Fusion*" henceforth). Specifically, the former is simply a combination after the *Utterance Representation* for richer turns-aware information while the latter is to aggregate matching states of previous turns after attention learning against each utterance itself and the response.

4 Experiment

4.1 Dataset

We evaluate our model on three multi-turn conversation datasets, the Ubuntu Dialogue Corpus (Ubuntu) (Lowe et al., 2015), the Douban Conversation Corpus (Douban) (Wu et al., 2017) and our released E-commerce Dialogue Corpus (ECD) [3]. Data statistics are in Table 1.

Ubuntu Dialogue Corpus *Ubuntu Dialogue Corpus* consists of multi-turn human-computer conversations constructed from Ubuntu IRC chat logs. The training set contains 1 million label-context-response triples where the original context and corresponding response are labeled as positive and negative response are selected randomly on the dataset. On both validation and test sets, each context contains one positive response and 9 negative responses.

Douban Conversation Corpus *Douban conversation corpus* is an open domain dataset constructed from Douban group which is a popular social networking service in China. Response candidates on the test set are collected by a standard search engine *Apache Lucene*[4], other than negative sampling without human judgment on Ubuntu Dialogue Corpus. That is, the last turn of each Douban dialogue with additional keywords extracted from the context on the test set is used as query to retrieve 10 response candidates from the Lucene index set.

E-commerce Dialogue Corpus In this part, we will introduce our E-commerce Dialogue Corpus. Though previously described public datasets have served in solid studies, there is no comprehensive e-commerce dataset available for research. We collect real-world conversations between customers and customer service staff from our E-commerce partners in Taobao [5], which is the largest e-commerce platform in China [6]. It contains over 5 types of conversations (e.g. commodity consultation, logistics express, recommendation, negotiation and chitchat) based on over 20 commodities. As word segmentation treatment is the primary step in Chinese language processing tasks (Zhao et al., 2017; Cai et al., 2017; Cai and Zhao, 2016), we adopt *BaseSeg* (Zhao et al., 2006) to tokenize the texts. For a discriminative learning, we add negative responses by ranking the response corpus based on the last utterance along with the top-5 key words in the context using *Apache Lucene*. The ratio of the positive and the negative is 1:1 in training and validation, and 1:9 in testing.

---

[3] Our released dataset along with source code can be accessed via `https://github.com/cooelf/DeepUtteranceAggregation`.

[4] `http://lucene.apache.org/`

[5] `https://www.taobao.com`

[6] All the data have been carefully desensitized and anonymized with the consent of our partners and avoid privacy issues.

Model	Ubuntu Dialogue Corpus			Douban Conversation Corpus					
	$R_{10}@1$	$R_{10}@2$	$R_{10}@5$	MAP	MRR	P@1	$R_{10}@1$	$R_{10}@2$	$R_{10}@5$
TF-IDF	0.410	0.545	0.708	0.331	0.359	0.180	0.096	0.172	0.405
RNN	0.403	0.547	0.819	0.390	0.422	0.208	0.118	0.223	0.589
CNN	0.549	0.684	0.896	0.417	0.440	0.226	0.121	0.252	0.647
LSTM	0.638	0.784	0.949	0.485	0.537	0.320	0.187	0.343	0.720
BiLSTM	0.630	0.780	0.944	0.479	0.514	0.313	0.184	0.330	0.716
Multi-View	0.662	0.801	0.951	0.505	0.543	0.342	0.202	0.350	0.729
DL2R	0.626	0.783	0.944	0.488	0.527	0.330	0.193	0.342	0.705
MV-LSTM	0.653	0.804	0.946	0.498	0.538	0.348	0.202	0.351	0.710
Match-LSTM	0.653	0.799	0.944	0.500	0.537	0.345	0.202	0.348	0.720
Attentive-LSTM	0.633	0.789	0.943	0.495	0.523	0.331	0.192	0.328	0.718
Multi-Channel	0.656	0.809	0.942	0.506	0.543	0.349	0.203	0.351	0.709
Multi-Channel$_{exp}$	0.368	0.497	0.745	0.476	0.515	0.317	0.179	0.335	0.691
SMN	0.726	0.847	0.961	0.529	0.569	0.397	0.233	0.396	0.724
DUA	**0.752**	**0.868**	**0.962**	**0.551**	**0.599**	**0.421**	**0.243**	**0.421**	**0.780**

Table 2: Comparison of different models on Ubuntu Dialogue Corpus and Douban Conversation Corpus. All the results except ours are from (Wu et al., 2017).

4.2 Settings

Our evaluation is based on the following information retrieval metrics: Mean Average Precision (MAP), Mean Reciprocal Rank (MRR), Precision at 1 (P@1) and Recall at position k in n candidates ($Rn@k$) , which are widely used for relevance evaluation (Wu et al., 2017; Lowe et al., 2015). For the sake of computational efficiency, the maximum number of utterances is specialized as 10 and each utterance contains at most 50 words. We apply truncating and zero-padding when necessary. Word embedding is trained by Word2Vector (Mikolov et al., 2013) on the training data and the dimension is 200. Our model is implemented using the Theano [7]. We use stochastic gradient descent with ADAM (Kingma and Ba, 2014) updates for optimization. The batch size is 200 and the initial learning rate is 0.001. The window size of convolution and pooling is (3, 3) and the number of hidden units for the character GRU is set to 200. All of our models are run on a single GPU (GeForce GTX 1080 Ti). We run all the models up to 5 epochs and select the model that achieves the best result in validation.

Our baselines include:

• **Single-turn matching models**: Basic models in (Kadlec et al., 2015; Lowe et al., 2015), including TF-IDF, CNN, RNN, LSTM and biLSTM ; We also explore other advanced single-turn matching models, MV-LSTM (Wan et al., 2016), Match-LSTM (Wang and Jiang, 2015), Attentive-LSTM (Tan et al., 2015), Multi-Channels (Wu et al., 2017); These models concatenate the context utterances together to match a response.

• **Advanced multi-turn matching models**: Multi-view model of (Zhou et al., 2016) that models utterance relationships from word sequence view and utterance sequence view; Deep Learning-to-Respond (DL2R) model of (Yan et al., 2016) which reformulates the last utterance (query) with other utterances in the context via neural model; Sequential Matching Network (SMN) (Wu et al., 2017) that matches a response with each utterance in the context.

The results of baseline models on Ubuntu and Douban are from (Wu et al., 2017). For evaluation on our ECD dataset, we reproduce the models following their same settings.

4.3 Experimental Results

Table 2-3 show the results on the three corpora. Our model outperforms all other models greatly in terms of most of the metrics. Single matching models which concatenate the previous utterances, perform much worse than our model, showing the importance of utterance relationships and simply concatenating

---

[7]https://github.com/Theano/Theano

Model	$R_{10}@1$	$R_{10}@2$	$R_{10}@5$
TF-IDF	0.159	0.256	0.477
RNN	0.325	0.463	0.775
CNN	0.328	0.515	0.792
LSTM	0.365	0.536	0.828
BiLSTM	0.355	0.525	0.825
Multi-View	0.421	0.601	0.861
DL2R	0.399	0.571	0.842
MV-LSTM	0.412	0.591	0.857
Match-LSTM	0.410	0.590	0.858
Attentive-LSTM	0.401	0.581	0.849
Multi-Channel	0.422	0.609	0.871
Multi-Channel$_{exp}$	0.352	0.556	0.827
SMN	0.453	0.654	0.886
DUA	**0.501**	**0.700**	**0.921**

Table 3: Comparison of different models on E-commerce Dialogue Corpus.

utterances together is not an appropriate solution for multi-turn conversation modeling. Our model also achieves a great improvement (4.8% $R_{10}@1$ on ECD corpus) over the state-of-the-art multi-turn response matching model, SMN, which matches each utterance and response without turns-aware aggregation and matching attention flow. This comparison indicates the effectiveness of our context composing approach. The advantage on ECD dataset further indicates our model can well imitate the conversations of real customer service instead of merely being good at chitchat.

4.4 Discussion

Conversation Type Analysis   To evaluate the model performance on different types of conversations, we manually separate our ECD test set into 5 categories.

- **Consultation**: consultations about commodity's property, usage, packaging, etc.
- **Logistics**: questions about logistics partners, delivery progress.
- **Recommendation**: commodity comparisons and recommendations.
- **Negotiation**: customer complaints and negotiations.
- **Chitchat**: greetings, non task-oriented conversations and chitchats.

Table 4 shows the statistics and the model results. As we see, the types of chitchat and logistics tend to be easily handled. Recommendations, consultation and negotiations are relatively harder to respond since they often involve with various topics (e.g. the concerned commodities) and intentions, which makes our corpus more challenging than previous chitchat or question answering based corpora.

Visualization   To analyze the effectiveness of the attention mechanism of our model, we draw the self-matching distributions after *matching attention flow*. From the validation set of our ECD data, Figure 3 shows the word weights of a momentous utterance (with high weights in the response matching component) and the response respectively. We see the model could accurately distill the linchpin from the utterance, {*Next consumption, reissue, a bag of almond, send you, some nuts, cashback*} and from the response {*too many orders before, really sorry, don't be angry, your gift*}. When a user complained about the missing gift and slow delivery, our model could distinguish the user's intention after self-matching and seek out the suitable response substantially according to the crux of the presented utterance. This shows our model is effective at selecting the vital points after *Matching Attention Flow*, guiding the *Response Matching* layer to collect more relevant pieces.

Ablation Study   To have an insight of the effectiveness of each component in DUA, we remove one each time. The steepest reduction (6.9% $R_{10}@1$) is observed when we remove *Matching Attention Flow* which shows it quite vital to draw the linchpins of each utterance. The performance also drops substan-

(a) Highlighted utterance　　　　　　　　(b) Response
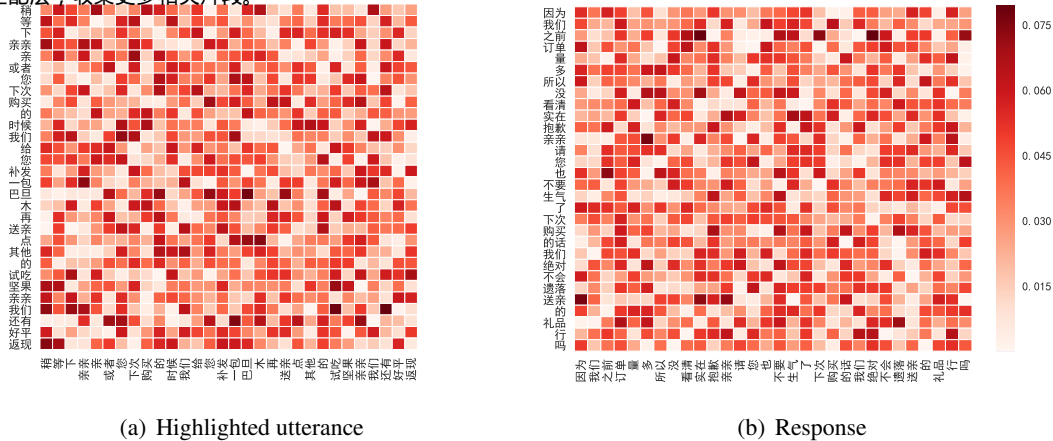
Last utterance (user): How can you miss my gift! And the delivery is so slow. You are spoiled !!!
Highlighted utterance (bot): please wait a moment dear. For compensation, we'll reissue you a bag of almonds at your next consumption. Besides, we will also send you some nuts to taste. If you give us five-star rating and comments, you'll also receive some cashback.
Response: Because we had too many orders before, we unfortunately misread your order. We are really sorry for the mistake. Please don't be angry, we will never forget your gift again.

Figure 3: Pair-wise attention visualization on utterance and response after matching attention flow.

	$R_{10}@1$	$R_{10}@2$	$R_{10}@5$
Consultation (36.1%)	0.474	0.696	0.900
Logistics (7.3%)	0.510	0.707	0.916
Recommendation (4.4%)	0.487	0.590	0.897
Negotiation (5.9%)	0.385	0.462	0.846
Chitchat (26.3%)	0.573	0.762	0.931
Overall(100%)	0.501	0.700	0.921

Table 4: Results on different types of conversations.

tially (4.8% $R_{10}@1$) when removing *Context Fusion* including the first turns-aware aggregation (first-stage aggregation) and replacing the last GRU (last-stage aggregation) for matching accumulation with a multi-layer perceptron. This indicates that utterance relationships are indeed important. Without *Context Fusion* and *Matching Attention Flow* mechanisms, the model performs the worst which verifies our proposed mechanism indeed improves the context representation essentially.

4.5　Error Analysis

After carefully analyzing the predicted responses, we find the error cases could be classified into the following categories for later further improvement.

Multiple intentions　In E-commerce conversations users extremely likely express various intentions in a single message, which is another big difference from previous multi-turn conversation corpus besides diverse types of conversations among various commodities. For example, {*User: How about the packaging of skin care products. By the way, which delivery company will be responsible for shipping and how long can I receive the goods?*}. This would seriously confuse the model where the given response might be preferential to one or another aspect.

Topic errors　Our model retrieves response according to semantic similarity with the context, with no special attention on the conversation topic, such as the currently discussed commodities. In most cases, the concerned commodity would be picked out from the context with high attention weights and guide the model to select responses. However, when the conversation involves several goods, for example, {*User: How about nuts? Bot: Nuts is good. User: Ok then, how about zongzi?*}, the model might give the response about *nuts* instead of *zongzi*. This indicates there exists much potential for improvements

	$R_{10}@1$	$R_{10}@2$	$R_{10}@5$
DUA	0.501	0.700	0.921
-CF	0.453	0.642	0.890
-MAF	0.432	0.625	0.883
-CF -MAF	0.413	0.613	0.867

Table 5: Ablation study on ECD dataset. CF and MAF denote the *Context Fusion* and *Matching Attention Flow*. The bracket means the context fusion approach adopted by the model.

by considering extra topic recognition.

Multiple suitable responses   In our ECD dataset, we assume there is only one correct response for each conversation which is the same setting as Ubuntu Dialogue Corpus. However, the model sometimes gives responses having similar meaning with the ground-truth one, but they would be regarded as wrong during evaluation, especially for fairly long conversations. This could make the task rather challenging with the strict restriction of exact match. This might be alleviated by involving expert labeling like (Wu et al., 2017). However, this is quite labour-intensive and subjective. In the future, we would explore more automatic solutions.

5   Conclusion

In this paper, we propose a deep utterance aggregation approach to form a fine-grained context representation. We also release the first e-commerce dialogue corpus to research communities. Experiments on three datasets show the model can yield new state-of-the-art results. Various analyses are conducted to evaluate the model and the released dataset. In the future, we may study how to improve modeling of contextual semantics and design a better neural network for multi-turn conversations in terms of various intentions and topics.

References

Hongxiao Bai and Hai Zhao. 2018. Deep enhanced representation for implicit discourse relation recognition. In *Proceedings of the 27th International Conference on Computational Linguistics (COLING 2018)*.

Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Jauvin. 2003. A neural probabilistic language model. *Journal of machine learning research*, pages 1137–1155.

Deng Cai and Hai Zhao. 2016. Neural word segmentation learning for Chinese. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (ACL 2016)*, pages 409–420.

Deng Cai and Hai Zhao. 2017. *Pair-Aware Neural Sentence Modeling for Implicit Discourse Relation Classification*. IEA/AIE 2017, Part II, LNAI 10351.

Deng Cai, Hai Zhao, Zhisong Zhang, Yuan Xin, Yongjian Wu, and Feiyue Huang. 2017. Fast and accurate neural word segmentation for Chinese. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (ACL 2017)*, pages 608–615.

Jiaxun Cai, Shexia He, Zuchao Li, and Hai Zhao. 2018. A full end-to-end semantic role labeler, syntactic-agnostic or syntactic-aware? In *Proceedings of the 27th International Conference on Computational Linguistics (COLING 2018)*.

Kyunghyun Cho, Bart Van Merrienboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using rnn encoder-decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP 2014)*, pages 1724–1734.

Lei Cui, Shaohan Huang, Furu Wei, Chuanqi Tan, Chaoqun Duan, and Ming Zhou. 2017. Superagent: A customer service chatbot for e-commerce websites. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, System Demonstrations (ACL 2017)*, pages 97–102.

Bhuwan Dhingra, Lihong Li, Xiujun Li, Jianfeng Gao, Yun-Nung Chen, Faisal Ahmed, and Li Deng. 2017. Towards end-to-end reinforcement learning of dialogue agents for information access. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (ACL 2017)*, pages 484–495.

He He, Anusha Balakrishnan, Mihail Eric, and Percy Liang. 2017. Learning symmetric collaborative dialogue agents with dynamic knowledge graph embeddings. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (ACL2017)*, pages 1766–1776.

Shexia He, Zuchao Li, Hai Zhao, Hongxiao Bai, and Gongshen Liu. 2018. Syntax for semantic role labeling, to be, or not to be. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (ACL 2018)*.

Yafang Huang, Zuchao Li, Zhuosheng Zhang, and Hai Zhao. 2018. Moon IME: neural-based chinese pinyin aided input method with customizable association. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (ACL 2018), System Demonstration*.

Zhongye Jia and Hai Zhao. 2014. A joint graph model for Pinyin-to-Chinese conversion with typo correction. In *Proceedings of the 52th Annual Meeting of the Association for Computational Linguistics (ACL 2014)*, pages 1512–1523.

Rudolf Kadlec, Martin Schmid, and Jan Kleindienst. 2015. Improved deep learning baselines for ubuntu corpus dialogs. *arXiv preprint arXiv:1510.03753*.

Diederik Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Zuchao Li, Jiaxun Cai, Shexia He, and Hai Zhao. 2018. Seq2seq dependency parsing. In *Proceedings of the 27th International Conference on Computational Linguistics (COLING 2018)*.

Zachary C Lipton, Xiujun Li, Jianfeng Gao, Lihong Li, Faisal Ahmed, and Li Deng. 2018. Bbq-networks: Efficient exploration in deep reinforcement learning for task-oriented dialogue systems. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence (AAAI-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 *Proceedings of the SIGDIAL 2015 Conference (SIGDIAL 2015)*, pages 285–294.

Hongyuan Mei, Mohit Bansal, and Matthew R Walter. 2017. Coherent dialogue with attention-based language models. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence (AAAI 2017)*, pages 3252–3259.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Lianhui Qin, Zhisong Zhang, Hai Zhao, Zhiting Hu, and Eric P. Xing. 2017. Adversarial connective-exploiting networks for implicit discourse relation classification. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (ACL 2017)*, pages 1006–1017.

Minghui Qiu, Feng Lin Li, Siyu Wang, Xing Gao, Yan Chen, Weipeng Zhao, Haiqing Chen, Jun Huang, and Wei Chu. 2017. Alime chat: A sequence to sequence and rerank based chatbot engine. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (ACL 2017)*, pages 498–503.

Iulian Vlad Serban, Tim Klinger, Gerald Tesauro, Kartik Talamadupula, Bowen Zhou, Yoshua Bengio, and Aaron Courville. 2017a. Multiresolution recurrent neural networks: An application to dialogue response generation. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence (AAAI 2017)*, pages 3288–3295.

Iulian Vlad Serban, Alessandro Sordoni, Ryan Lowe, Laurent Charlin, Joelle Pineau, Aaron Courville, and Yoshua Bengio. 2017b. A hierarchical latent variable encoder-decoder model for generating dialogues. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence (AAAI 2017)*, pages 3295–3302.

Ming Tan, Cicero Dos Santos, Bing Xiang, and Bowen Zhou. 2015. LSTM-based deep learning models for non-factoid answer selection. In *Proceedings of the International Conference on Learning Representations (ICLR 2016)*.

Shengxian Wan, Yanyan Lan, Jun Xu, Jiafeng Guo, Liang Pang, and Xueqi Cheng. 2016. Match-srnn: Modeling the recursive matching structure with spatial rnn. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence (IJCAI 2016)*, pages 2922–2928.

Shuohang Wang and Jing Jiang. 2015. Learning natural language inference with LSTM. In *Proceedings of NAACL-HLT 2016 (NAACL 2016)*, pages 1442–1451.

Wenhui Wang, Nan Yang, Furu Wei, Baobao Chang, and Ming Zhou. 2017. Gated self-matching networks for reading comprehension and question answering. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (ACL 2017)*, pages 189–198.

Tsung-Hsien Wen, David Vandyke, Nikola Mrkšić, Milica Gasic, Lina M. Rojas Barahona, Pei-Hao Su, Stefan Ultes, and Steve Young. 2017. A network-based end-to-end trainable task-oriented dialogue system. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics (EACL 2017)*, pages 438–449.

Jason D Williams, Kavosh Asadi, and Geoffrey Zweig. 2017. Hybrid code networks: practical and efficient end-to-end dialog control with supervised and reinforcement learning. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (ACL 2017)*, pages 665–677.

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 *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (ACL 2017)*, pages 496–505.

Yu Wu, Wei Wu, Dejian Yang, Can Xu, Zhoujun Li, and Ming Zhou. 2018. Neural response generation with dynamic vocabularies. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence (AAAI 2018)*.

Rui Yan, Yiping Song, and Hua Wu. 2016. Learning to respond with deep neural networks for retrieval-based human-computer conversation system. In *International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 55–64.

Zhao Yan, Nan Duan, Peng Chen, Ming Zhou, Jianshe Zhou, and Zhoujun Li. 2017. Building task-oriented dialogue systems for online shopping. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence (AAAI 2017)*, pages 4618–4627.

Tom Young, Erik Cambria, Iti Chaturvedi, Minlie Huang, Hao Zhou, and Subham Biswas. 2018. Augmenting end-to-end dialog systems with commonsense knowledge. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence (AAAI-18)*.

Zhuosheng Zhang and Hai Zhao. 2018. One-shot learning for question-answering in gaokao history challenge. In *Proceedings of the 27th International Conference on Computational Linguistics (COLING 2018)*.

Zhisong Zhang, Hai Zhao, and Lianhui Qin. 2016. Probabilistic graph-based dependency parsing with convolutional neural network. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (ACL 2016)*, pages 1382–1392.

Saizheng Zhang, Emily Dinan, Jack Urbanek, Arthur Szlam, Douwe Kiela, and Jason Weston. 2018a. Personalizing dialogue agents: I have a dog, do you have pets too? In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (ACL 2018)*.

Zhuosheng Zhang, Yafang Huang, and Hai Zhao. 2018b. Subword-augmented embedding for cloze reading comprehension. In *Proceedings of the 27th International Conference on Computational Linguistics (COLING 2018)*.

Zhuosheng Zhang, Jiangtong Li, Hai Zhao, and Bingjie Tang. 2018c. Sjtu-nlp at semeval-2018 task 9: Neural hypernym discovery with term embeddings. In *Proceedings of the 12th International Workshop on Semantic Evaluation (SemEval 2018), Workshop of NAACL-HLT 2018*.

Zhuosheng Zhang, Jiangtong Li, Hai Zhao, and Bingjie Tang. 2018d. Sjtu-nlp at semeval-2018 task 9: Neural hypernym discovery with term embeddings. In *Proceedings of the 12th International Workshop on Semantic Evaluation (SemEval 2018), Workshop of NAACL-HLT 2018*.

Hai Zhao, Chang-Ning Huang, Mu Li, and Taku Kudo. 2006. An improved Chinese word segmentation system with conditional random field. *Proceedings of the Fifth Sighan Workshop on Chinese Language Processing*, pages 162–165.

Hai Zhao, Deng Cai, Changning Huang, and Chunyu Kit. 2017. *Chinese Word Segmentation, a decade review (2007-2017)*. China Social Sciences Press, Beijing, China, July.

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 *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing (EMNLP 2016)*, pages 372–381.

Ganbin Zhou, Ping Luo, Rongyu Cao, Fen Lin, Bo Chen, and Qing He. 2017. Mechanism-aware neural machine for dialogue response generation. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence (AAAI 2017)*, pages 3400–3408.