DialCSP: A Two-stage Attention-based Model for Customer Satisfaction Prediction in E-commerce Customer Service

Zhenhe Wu[1,2,3], Liangqing Wu[3], Shuangyong Song[3], Jiahao Ji[1], Bo Zou[3], Zhoujun Li✉[1,2], and Xiaodong He[3]

[1] School of Computer Science and Engineering, Beihang University, Beijing, China
{wuzhenhe,jiahaoji}@buaa.edu.cn
[2] State Key Lab of Software Development Environment, Beihang University, Beijing, China lizj@buaa.edu.cn
[3] JD AI Research, Beijing, China
{wuliangqing,songshuangyong,cdzoubo,hexiaodong}@jd.com

Abstract. Nowadays, customer satisfaction prediction (CSP) on e-commerce platforms has become a hot research topic for intelligent customer service. CSP aims to discover customer satisfaction according to the dialogue content of customer and intelligent customer service, for the purpose of improving service quality and customer experience. Previous works have made some progress in many aspects, but they mostly ignore the huge expressional differences between customer questions and customer service answers, and fail to adequately consider the internal relations of these two kinds of personalized expressions. In this paper, we propose a two-stage dialogue-level classification model containing an intra-stage and an inter-stage, to emphasize the importance of modeling customer part (content of customer questions) and service part (content of customer service answers) separately. In the intra-stage, we model customer part and service part separately by using attention mechanism combined with personalized context to obtain a *customer state* and a *service state*. Then we interact these two states with each other in the inter-stage to obtain the final satisfaction representation of the whole dialogue. Experiment results demonstrate that our model achieves better performance than several competitive baselines on our in-house dataset and four public datasets.

Keywords: customer satisfaction prediction · intelligent customer service · attention-based model.

1 Introduction

With the development of e-commerce platforms in recent years, a large number of companies use customer service chatbots, for the reasons that they could answer to customers' questions quickly and save labor cost. Customer satisfaction prediction (CSP) for the dialogues in customer service chatbots has become

an important problem in industry. For one thing, customers' satisfaction is a crucial indicator to evaluate the quality of service, which can help improve the ability of chatbots. For another, predicting customers' satisfaction in real time helps platforms handle problematic dialogues by transferring customer service chatbots to staffs timely, which can improve the customers' experience.

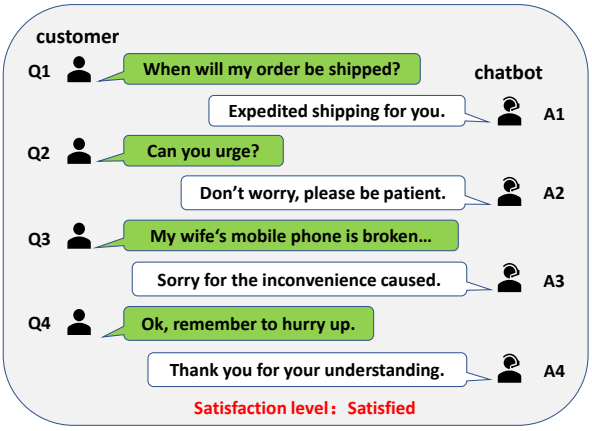

Fig. 1. A dialogue of customer and chatbot on e-commerce platform.

CSP is a multi-class classification task. Existing researches on CSP is mainly divided into two directions, one is the turn-level CSP, the other is the dialogue-level CSP. The former direction concerns satisfaction prediction in every customer-service turn [19, 20, 24], while the latter one predicts satisfaction level of the whole dialogue [21, 22, 9, 13, 14]. Turn-level CSP can only capture temporary user's satisfaction results which may have certain contingency, while dialogue-level CSP is the key point to evaluate the quality of the service and whether the customer's problem has been solved. Thus, in this study, we concentrate on dialogue-level CSP with five satisfaction levels (*strongly satisfied*, *satisfied*, *neutral*, *dissatisfied*, or *strongly dissatisfied*). As shown in figure 1, the customer expresses his anxiety and displeasure at the beginning, then turns into satisfied after the good answers of the chatbot.

To address the dialogue-level CSP task, many approaches extracted features from dialogue content and built models to fully utilize the interaction between customer questions and customer service answers. Some earlier studies used manual features to represent conversational context [21, 22], while recent studies concerned more on how well the questions and answers match each other [13, 14]. Although these works have made great progress in CSP task, two issues still remain: 1) existing studies ignore the huge expressional differences between customers and customer service chatbots, in terms of the emotion intensity, speaking habits, language richness, sentence length and etc. 2) most prior studies fail to

adequately consider the internal relations of personalized expressions for customers and staffs/chatbots respectively.

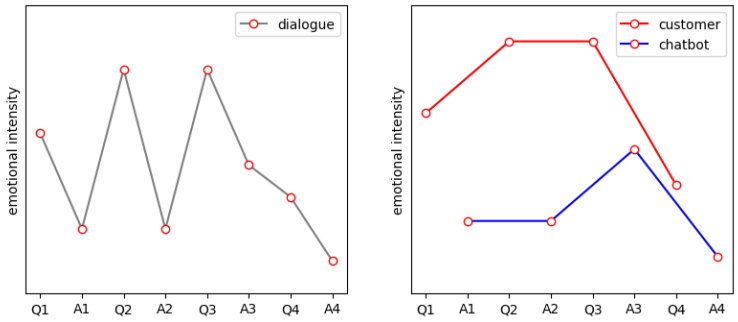

Fig. 2. The emotional intensity trends are obviously after split.

According to the above analysis, we figure that besides handling the interaction of customer questions and customer service answers, modeling customer part and service part separately should also be taken into consideration due to their expressional differences in many aspects. For example, customers' questioning emotion is volatile and the emotional intensity is usually high, while the answering emotion of service is relatively stable and the emotional intensity is low. Figure 2 shows the emotional intensity trend of the case in Figure 1, in which the customer's emotional intensity is higher with greater fluctuation, while the chatbot is the opposite. After splitting the dialogue into customer part and service part, we are able to catch their emotional intensity trends intuitively. For the similar reason, other aspects of expressional differences also matter.

Thus, we propose a classification model for CSP in E-commerce customer service dialogues which is called DialCSP. Besides an encoder and an decoder, our model contains an intra-stage and an inter-stage as core structures. Firstly, we adopt an encoding module to extract features of the dialogue content. Next comes the intra-stage, which consists of a customer part and a service part. We split customer questions and customer service answers as two independent sequences and send them into these two parts separately. Specifically, each part is designed to fully extract the internal relations of the sequence. In the end of the intra-stage, we get *customer state* and *service state* as the results of the customer part and service part. Then, the inter-stage apply an interactive attention mechanism to capture satisfaction representations of the whole dialogue from *customer state* and *service state*. In the end, a decoder module is used to predict the final satisfaction level.

To summarize, our contributions are as follows:

– We propose a dialogue-level classification model DialCSP, for CSP in E-commerce customer service chatbots.

- By bringing forward a two-stage architecture, we split the dialogue content into customer part and service part to model them separately. With the results of *customer state* and *service state*, we construct interaction to capture final satisfaction representation. This architecture handles the above-mentioned two issues well.
- Experimental results indicate that our proposed model outperforms several baselines on our in-house dataset and four public datasets.

2 Related Work

In recent years, researchers paid much more attention to CSP and similar tasks. Some earlier works aimed to predict sentiment levels for subjective texts in different granularities, such as words [15], sentences [16], short texts [17] and documents [18]. More recently, mainstream research direction concentrated on turn-level and dialogue-level CSP.

Some researchers explored the turn-level structure, such as modeling dialogues via a hierarchical RNN [19], keeping track of satisfaction states of dialogue participants [20], exploring contrastive learning [24] and so on. But, due to the labels of turn-level satisfaction is difficult to obtain and dialogue-level CSP appears to reflect service quality more realistically, we focus on dialogue-level CSP in this paper.

To study the dialogue-level CSP, earlier methods used manual features [21, 22], while recent studies preferred deep neural networks and attention mechanism to explore how questions and answers interact with each other. Some researchers adopted a Bi-directional LSTM network to capture the contextual information of conversational services and use the hidden vector of the last utterance for satisfaction prediction [9], some researchers used each question to capture information from all answers to model customer-service interaction [13], while another study focusing on dialogue-level CSP used LSTM networks to capture contextual features and computed the semantic similarity scores between customer questions and customer service answers across different turns to model customer-service interaction [14]. However, these works didn't consider the expressional differences between customer and customer service. Morever, they failed to excavate the internal relations of personalized expression sequences. In this work, we work on addressing the two existing issues above, thus proposing DialCSP model for dialogue-level CSP.

3 Methodology

3.1 Problem Definition

In the real scenario, a customer asks questions and the chatbot provides the corresponding answers in turn, so a customer service dialogue is defined as a sequence of utterances $C = \{q_1, a_1, q_2, a_2, ..., q_n, a_n\}$. Each question q_i is followed by an answer a_i, and the length of dialogue is $2n$. The goal of our task is to predict

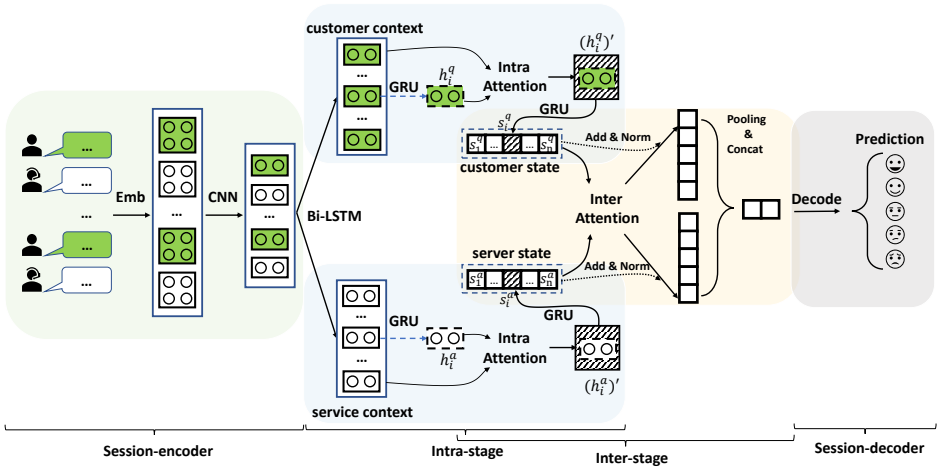

Fig. 3. Framework of the DialCSP model.

the satisfaction level y based on the dialogue content C, while the satisfaction level is divided into five classes: *strongly satisfied, satisfied, neutral, dissatisfied, strongly dissatisfied*.

3.2 Proposed Model

As shown in Figure 3, we propose DialCSP, a two-stage classification model for dialogue-level CSP. Besides a session-encoder and a session-decoder, the core part of our model contains an intra-stage and an inter-stage. The session-encoder is a dialogue encoding module to process the raw conversation content. Intra-stage is comprised of a customer part and a service part, which helps extract sufficient internal features of question sequences and answer sequences separately. For both parts in inter-stage, we utilize attention mechanisms to adequately discover the sentence characteristics at each time step from their personalized context, served as *customer state* and *service state*. Next, inter-stage applies an interactive attention mechanism to fully capture satisfaction representations of the whole dialogue from *customer state* and *service state*. Finally, the session-decoder contributes to predict the final satisfaction level. In the following sections, we will introduce the details of the model structure in order.

3.3 Session-encoder

Session-encoder aims to encode natural language dialogues into semantic representations. Our input is the whole dialogue text, in which words are separately transformed into 300 dimensional vectors by using pre-trained GloVe model [23]:

$$E = \text{GloVe}\,(C) \tag{1}$$

Then, inspired by previous study [1], we leverage a CNN layer with max-pooling to extract context independent features of each utterance. Concretely, we apply three filters of size 1,2,3 with 50 feature maps each, and employ ReLU activation [2] and max-pooling to deal with these feature maps:

$$fm_{1,2,3} = \text{ReLU}\left(\text{CNN}_{1,2,3}\left(E\right)\right) \tag{2}$$

$$fm'_{1,2,3} = \text{max-pooling}\left(fm_{1,2,3}\right) \tag{3}$$

Then, we concatenate these features and send them into a fully connected layer, which produces the context representations cr as follow:

$$fm' = \text{concat}\left(fm'_{1,2,3}\right) \tag{4}$$

$$cr = \text{ReLU}\left(W_0 fm' + b_0\right) \tag{5}$$

3.4 Intra-stage

Intra-stage is a core module of our two-stage model, which consists of the customer part and service part. We can alternately divide cr into question representations $qr = \{qr_1, qr_2, ..., qr_n\}$ and answer representations $ar = \{ar_1, ar_2, ..., ar_n\}$ as the input of customer part and service part. In the following, we will illustrate how these two parts of intra-stage adequately exploit the inside relations of their own utterance sequences.

customer part LSTM has a special unit called memory cell, which is similar to an accumulator or a gated neuron. We adapt a Bi-directional LSTM to capture long-term dependencies of qr:

$$m_i^q = \text{BiLSTM}^{\text{q}}\left(m_{i\pm1}^q, qr_i\right) \tag{6}$$

where $i = 1, 2, ..., n$. m_i is the output of Bi-directional LSTM at time step i, the whole context representation of question sequence is $m^q = \{m_1^q, m_2^q, ..., m_n^q\}$.

To better explore the internal relations of question sequence, we capture the satisfaction representation of each time step iteratively by adequately interacting current features with context information. Firstly, an GRU encoder is used to process the sequence:

$$h_i^q = \text{GRU}^{\text{q}}_{\text{encode}}\left(m_i^q, h_{i-1}^q\right) \tag{7}$$

where $h^q = \{h_1^q, h_2^q, ..., h_n^q\}$, h^q is the hidden state of GRU. Secondly, we use an attention mechanism to match h_i^q with the masked personalized context:

$$masked_i\left(m^q\right) = \begin{cases} m_j^q, & j \in \{1, 2, ..., i\} \\ 0, & \text{Otherwise} \end{cases} \tag{8}$$

$$q, k, v = h_i^q, masked_i\left(m^q\right), masked_i\left(m^q\right) \tag{9}$$

$$h_i^{q\prime} = \mathrm{IntraAtt}^{\mathrm{q}}\left(q, k, v\right) \tag{10}$$

where $h^{q\prime} = \{h_1^{q\prime}, h_2^{q\prime}, ..., h_n^{q\prime}\}$, $h^{q\prime}$ is the result of this attention layer.

Up to now, we have adequately obtained the internal relations of question sequence. Then, a GRU is used to decode the result from the intra attention layer:

$$s_i^q = \mathrm{GRU}_{\mathrm{decode}}^{\mathrm{q}}\left(h_i^{q\prime}, s_{i-1}^q\right) \tag{11}$$

$s^q = \{s_1^q, s_2^q, ..., s_n^q\}$, where s^q is *customer state* after the complete process of customer part.

service part Service part is the other part in intra-stage, which contributes to the satisfaction state of service. The whole structure of service part is similiar to customer part:

$$m_i^a = \mathrm{BiLSTM}^{\mathrm{a}}\left(m_{i\pm1}^a, ar_i\right) \tag{12}$$

$$h_i^a = \mathrm{GRU}_{\mathrm{encode}}^{\mathrm{a}}\left(m_i^a, h_{i-1}^a\right) \tag{13}$$

$$masked_i\left(m^a\right) = \begin{cases} m_j^a, & j \in \{1, 2, ..., i\} \\ 0, & \text{Otherwise} \end{cases} \tag{14}$$

$$q, k, v = h_i^a, masked_i\left(m^a\right), masked_i\left(m^a\right) \tag{15}$$

$$h_i^{a\prime} = \mathrm{IntraAtt}^{\mathrm{a}}\left(q, k, v\right) \tag{16}$$

$$s_i^a = \mathrm{GRU}_{\mathrm{decode}}^{\mathrm{a}}\left(h_i^{a\prime}, s_{i-1}^a\right) \tag{17}$$

where s^a is the *service state*.

3.5 Inter-stage

Inter-stage aims to fully interact s^q with s^a. Some researchers utilize attention mechanisms to capture the most relevant information and construct interaction between two sequences [3]. Inspired by those works, we use an attention mechanism to interact s^q with s^a:

$$\tilde{s}^q = \mathrm{InterAtt}^{\mathrm{q}}\left(s^q, s^a, s^a\right) \tag{18}$$

$$\tilde{s}^a = \text{InterAtt}^{\text{a}}\left(s^a, s^q, s^q\right) \tag{19}$$

In order to make the learning process smoother, we adopt a layer of add & normalization [4]:

$$\tilde{s}^{q'} = \text{Normalization}\left(\text{Add}\left(\tilde{s}^q, s^q\right)\right) \tag{20}$$

$$\tilde{s}^{a'} = \text{Normalization}\left(\text{Add}\left(\tilde{s}^a, s^a\right)\right) \tag{21}$$

In the end of the inter-stage, by using average pooling, we transform $\tilde{s}^{q'}$ and $\tilde{s}^{a'}$ into vectors and concatenate them together as follow:

$$s = \text{concat}\left(\text{pooling}\left(\tilde{s}^{q'}\right), \text{pooling}\left(\tilde{s}^{a'}\right)\right) \tag{22}$$

where s is the final satisfaction representation of the whole dialogue.

3.6 Session-decoder

The session-decoder module is used to decode the satisfaction representation s to predict the customer satisfaction. We use two layers of fully connected network with ReLU activation and softmax, then get the probability distribution of classification P. $\hat{y}$ is the final prediction of satisfaction level:

$$H = \text{ReLU}\left(W_1 s + b_1\right) \tag{23}$$

$$P = \text{softmax}\left(W_2 H + b_2\right) \tag{24}$$

$$\hat{y} = \underset{k}{\text{argmax}}\left(P[k]\right) \tag{25}$$

As for the loss function, we choose cross-entropy:

$$\mathcal{L}(\theta) = -\sum_{v \in y_{\mathcal{V}}} \sum_{z=1}^{Z} Y_{vz} \ln P_{vz} \tag{26}$$

where $y_{\mathcal{V}}$ is the set of dialogues that have real labels. Y is the label indicator matrix, and θ is the collection of trainable parameters in DialCSP.

4 Experimental Settings

This section mainly introduces datasets, hyper parameters and baselines used in our experiments.

Datasets	Train	Val	Test	Avg-turns
CECSP	22576	2822	2801	3.67
Clothes	8000	1000	1000	8.14
Makeup	2832	354	354	8.01
MELD	1037	113	279	3.19
EmoryNLP	685	88	78	3.86

Table 1. The statistics of the five datasets. While **CECSP** is our constructed Chinese E-commerce CSP dataset, **Clothes** and **Makeup** are two released corpora in different domains. **MELD** and **EmoryNLP** are two CER datasets.

4.1 Datasets

We evaluate DialCSP on our in-house dataset (*a Five-classification task*) and four released public datasets (*Three-classification tasks*).

- **CECSP** This is our in-house Chinese E-commerce CSP dataset collected from one of the largest E-commerce platforms. We use real customer feedback as the dialogue-level satisfaction labels which include *strongly satisfied*, *satisfied*, *neutral*, *dissatisfied* and *strongly dissatisfied*.
- **Clothes & Makeup** These are two CSP datasets in clothes and makeup domain collected from a top E-commerce platform [13]. Each dialogue is annotated as one of the three satisfaction classes: *satisfied*, *neutral* and *dissatisfied*.
- **MELD** This is a multi-party conversation corpus collected from the TV show Friends [5]. Each utterance is annotated as one of the three sentiment classes: *negative*, *neutral* and *positive*. While *negative* and *positive* are considered as *dissatisfied* and *satisfied* respectively, *neutral* is kept unchanged.
- **EmoryNLP** This is also a multi-party conversation corpus collected from Friends, but varies from MELD in the choice of scenes and emotion labels [6]. The emotion labels include *neutral*, *joyful*, *peaceful*, *powerful*, *scared*, *mad* and *sad*. To create three satisfaction classes: *joyful*, *peaceful* and *powerful* are positive emotion so we group them together to form the *satisfied* class; *scared*, *mad* and *sad* are negative emotion so we group them together to form the *dissatisfied* class; and *neutral* is kept unchanged.
- **Transforming rules for MELD & EmoryNLP** Original MELD and EmoryNLP are two released conversational emotion recognition (CER) datasets. We transform them into the conversational service scenario following four rules: (1) We consider the first speaker of a dialogue as the customer (all other speakers as the customer service) and map all the emotion labels into turn-level satisfaction labels; (2) We concatenate consecutive utterances from the same person as a long utterance; (3) If a dialogue is ended by the first speaker, we use utterance "NULL" as the answer of the last turn; (4) We set the dialogue-level satisfaction as the average of turn-level satisfaction.

4.2 Baselines

We compared DialCSP with the following baselines in our experiments:

- **LSTMCSP** [9]:This model uses a Bi-directional LSTM network to capture the user's intent and identify user's satisfaction.
- **CMN** [10]:It is an end-to-end memory network which updates contextual memories in a multi-hop fashion for conversational emotion recognition.
- **DialogueGCN** [11]: It is a graph-based approach which leverages inter-speakers' dependency of the interlocutors to model conversational context for emotion recognition.
- **CAMIL** [13]: This Context-Assisted Multiple Instance Learning model predicts the sentiments of all the customer utterances and then aggregates those sentiments into service satisfaction polarity.
- **LSTM-Cross** [14]: This model uses LSTM networks to capture contextual features .Then, these features are concatenated with the cross matching scores to predict the satisfaction.
- **DialogueDAG** [12]: This model uses directed graphs to collect nearby and distant historical informative cues. We aggregate the node representations to capture dialogue-level representations for CSP.
- **BERT** [25] **& Dialog-BERT** [26]: We use pre-trained language models and linear layers with softmax on CSP problem. For each dialogue, we use [sep] to concat utterances as input of the models. We use pre-trained *bert-base-chinese* and *dialog-bert-chinese* on CECSP, and pre-trained *bert-base-uncased* and *dialog-bert-english* on MELD & EmoryNLP (In Clothes & Makeup datasets, words are replaced with ids for the data-safety, so we can not use pre-trained model on them).

4.3 Hyper parameters

We reproduce all baselines with their original experimental settings. In our two-stage model, the batch sizes are set to be 64 for CECSP, Clothes, Makeup, MELD and EmoryNLP. We adopt Adam [8] as the optimizer with initial learning rates of 1e-3 and L2 weight decay rates of 1e-4, respectively. The dropout is set to be 0.5 [7]. We train all models for a maximum of 200 epochs and stop training if the validation loss does not decrease for 30 consecutive epochs. The total number of parameters in this model is 59.84 million. We use a piece of Tesla P40 24GB. Each epoch of these experiments costs around 400 seconds.

5 Results and Analysis

5.1 Overall Results

The overall results of all the models on five datasets are shown in Table 2. We can learn from the results that DialCSP achieves better performance than all the baselines on five datasets.

Model	CECSP		Clothes		Makeup		MELD		EmoryNLP	
	Acc.	F1	Acc.	F1	Acc.	F1	Acc.	F1	Acc.	F1
LSTMCSP	51.55	50.10	75.59	75.78	76.31	76.56	42.29	43.08	50.01	47.56
CMN	52.09	50.32	78.5	78.1	81.07	80.88	45.52	44.08	52.56	48.52
DialogueGCN	52.69	50.25	76.89	76.82	77.72	77.78	46.39	44.99	52.72	48.78
CAMIL	55.43	52.92	78.30#	78.40	78.50#	78.64	44.44	39.02	55.13	49.52
LSTM-Cross	55.51	53.11	78.91	79.33	79.88	79.58	46.70	45.41	55.28	51.00
DialogueDAG	55.12	51.97	75.4	75.04	73.73	73.73	48.03	47.28	59.26	54.82
BERT	55.57	52.86	-	-	-	-	50.18	49.79	58.97	57.31
Dialog-BERT	56.44	51.72	-	-	-	-	47.31	46.42	**64.10**	**60.35**
DialCSP	**57.34**	**54.69**	**81.2**	**80.71**	**82.2**	**82.07**	**50.9**	**50.35**	61.54	57.88

Table 2. Overall performance on the five datasets. We use the accuracy and the weighted F1 score to evaluate each model. Scores marked by "#" are reported results in authors' paper, while others are based on our re-implementation.

LSTMCSP, CMN, and DialogueGCN achieve similar performance on CECSP, MELD and EmoryNLP. CMN is capable of capturing the emotional cues in context, thus achieving better F1 scores than LSTMCSP on Clothes and Makeup. However, chatbot answers are always neutral in conversational service, which narrow the gap between CMN and LSTMCSP on CECSP. In our scene, customer questions and chatbot answers are alternating, so the related positions between them cannot provide additional information. Thus, the position method in DialogueGCN does not have better performance here.

CAMIL takes turn-level sentiment information into account and outperforms previous strategies on four datasets except MELD. Due to the customer-service interaction modeling method, LSTM-Cross has made further improvement on all datasets, which implies the importance of interactions in single turn. DiologueDAG uses graphical structure to effectively collect nearby and distant information, so it performs well on datasets with shorter average turns, such as MELD and EmoryNLP. But when the average turns become longer, it doesn't work well.

BERT is one of the strongest baslines in multiple NLP tasks. We use pre-trained *bert-base-chinese* on CECSP and *bert-base-uncased* on MELD & EmoryNLP. Dialog-BERT is further designed to focus on dialogue tasks, we use pre-trained *dialog-bert-chinese* on CECSP and *dialog-bert-english* on MELD & EmoryNLP. The superiority of pre-training makes BERT and Dialog-BERT achieve better weighted F1 score over other baselines on small datasets MELD & EmoryNLP, but the performance is mediocre on big dataset CECSP.

DialCSP reaches the new state of the art on four datasets except EmoryNLP. On the one hand, intra-stage extracts internal correlation features of question sequence and answer sequence in customer part and service part separately. Using attention mechanism with the personalized context of both sequences makes feature extraction sufficient at each time step. On the other hand, we think each customer question is not only associated with the answer behind, but

Method	Weighted F1 score	
	CECSP	Clothes
(1) Two-stage model	**54.69**	**80.71**
(2) - Inter-stage	53.68(↓ 1.01)	78.64(↓ 2.07)
(3) - Intra-stage	54.07(↓ 0.62)	78.23(↓ 2.48)
(4) - Intra-stage & Inter-stage	53.53(↓ 1.16)	78.44(↓ 2.27)
(5) + context part	53.61(↓ 1.08)	79.37(↓ 1.34)
(6) + context part & attention	54.31(↓ 0.38)	79.16(↓ 1.55)

Table 3. Results of ablation study on the two representative datasets.

also the answers in other turns, so inter-stage conducts fully interaction between *customer state* and *service state*, which is different from the turn-level approaches in earlier researches. As the results, our proposed model has improved by at least 1%∼3% on F1 score over five datasets, compared with non-pretrained baseline models. It also performs better than BERT and Dialog-BERT on CECSP and MELD, meanwhile, taking the advantage of light and fast. Only on the smallest dataset EmoryNLP, Dial-BERT performs better than DialCSP. However, the test set of EmoryNLP contains only 78 samples, so the results may have certain contingency.

5.2 Ablation Study

To study the impact of the modules in our two-stage model, we evaluate it by removing 1) inter-stage 2) intra-stage 3) intra-stage and inter-stage together. Removing the inter-stage means the we only retain the intra-stage (The output of the Bi-LSTM is taken as the input of the intra-stage). Removing the intra-stage means only the inter-stage remains (The output of the inter-stage is taken as the input of session-decoder after pooling & concatenate). Removing both intra-stage & inter-stage means we no longer separate the dialogue into two parts and only retain the customer part of the intra-stage (The output of session-encoder is taken as the input of customer part). We use CECSP and Clothes as the representatives in this study because they are larger datasets with short and long average turns. The results are shown in Table 3.

Here are two sets of comparative experiments. Firstly, let's pay attention to the comparison of rows(1)(2)(3). Without inter-stage, the weighted F1 score drops by 1.01% on CECSP and 2.07% on Clothes. Without intra-stage, the weighted F1 score drops by 0.62% on CECSP and 2.48% on Clothes. The results imply the importance of both two stages, none of them can be removed. Secondly, experiments on rows(2)(4) illustrate the advantage of intra attention. Both of them don't have inter-stage, and the only difference between them is whether to split the dialogue into question sequence and answer sequence. As shown in the table, the weighted F1 score drops by 0.15% and 0.20% if we don't apply

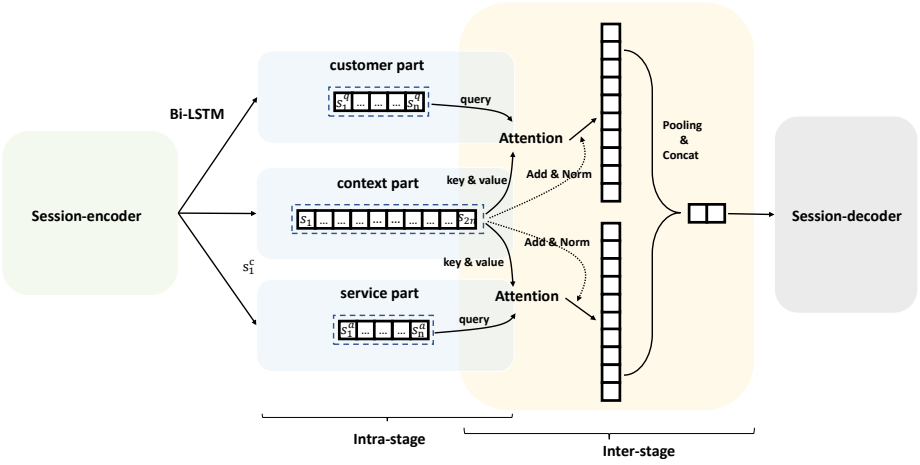

Fig. 4. Framework of DialCSP model with extra context part in the intra-stage.

intra method. Thus we can draw a conclusion, the intra method helps extract the internal correlation of customer context and service context respectively, and indeed improves the performance of our model.

Furthermore, to verify if full context features may improve model performance, we conduct two groups of experiments. 1) We add context part in the intra-stage (The full output of session-encoder is taken as the input of context part, the structure of context part is the same as customer part and service part) and concatenate its output with $\tilde{s}^q$ and $\tilde{s}^a$ after pooling. 2) We add context part and get context state s as the output, then we use attention mechanism to interact s with s^q and s^a separately, as shown in Figure 4. The results are shown in Table 3.

In the experiments on row(5), the only difference between this model and DialCSP is an extra context part. As shown in the Table 3, the weighted F1 score drops by 1.08% and 1.34%, which proves that simply increasing extra fully context information would not improve the performance of DialCSP. In the experiments on row(6), we conduct interaction of context part with customer part and service part by using attention mechanism, as shown in Figure 4. The weighted F1 score drops by 0.38% and 1.55%, which shows the effect of interaction compared to row(5), while it still can't improve DialCSP model.

In conclusion, the ablation study proves that both intra-stage and inter-stage play important roles. In particular, the intra method of separating context representations into questions and answers contributes to the improvement of our model. Furthermore, we find extra fully context features extraction can't improve the performance of DialCSP model, which signifies the completeness and rationality of our model.

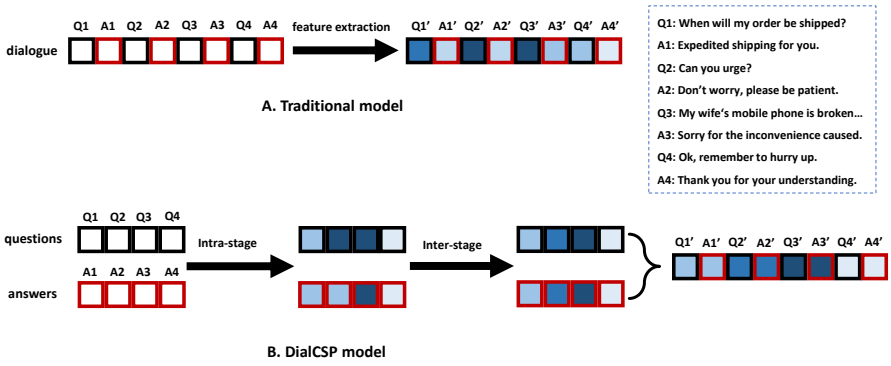

Fig. 5. Results of case analysis. Part A represents the feature extraction process of traditional model, while Part B represents our DialCSP model. The colors of heatmap show the values of attention weights.

5.3 Case Analysis

In order to better understand the advantages of the DialCSP model, we analyse the case in Figure 1. The result shows in Figure 5. The heatmap is used to represent the values of attention weights, where darker colors mean larger weights.

Part A illustrates the feature extraction process in traditional models, in which the dialogue would not be separated into questions and answers. As customer's expression contains richer information ("when", "urge", "broken") of his problem and emotion, the model will pay more attention to Q1, Q2 and Q3. So, it is likely to ignore the importance of answers, which are critical to deciding whether these questions are solved, thus affecting customer's satisfaction deeply too.

By contrast, Part B illustrates our DialCSP model. The dialogue is split into customer questions and chatbot answers, so the model can better learn the inside relations of the two sequences separately in the intra-stage, which ensures the expressions of customers would not attract much more attention than chatbots. In this case, the customer expresses his anxiety and tells the mobile phone is broken in Q2 and Q3, so the weights of those two are larger in the question sequence. Similarly, A3 have larger weights in the answer sequence due to its obvious comforting expression. Then, the inter-stage conducts the interaction to adjust the attention weights of the two parts. In the end, we concatenate two parts and find Q2,A2,Q3,A3 are important utterances of this dialogue. In this dialogue situation, although the customer mainly shows his bad emotion and unsolved problem in Q2 and Q3, the chatbot comforts him in A2 and A3, which leads to a satisfied result. The result of part B appears to be more reasonable.

By comparing the two results, we find the intra-stage of our DialCSP model can balance the expressional differences of customer questions and chatbot answers, while the traditional model pays more attention to customer questions. What's more, the inter-stage interacts *customer state* with *service state* to adjust the weights of attention, which can help capture the characteristics of dialogue more smoothly.

6 Conclusion

In this paper, we propose a two-stage model for dialogue-level CSP task. We first introduce an intra-stage to discover the relations inside customer part and service part respectively, in which an attention mechanism with masked personalized context is used to fully capture the *customer state* and *service state*. Then, we use an inter attention mechanism to combine those two states in inter-stage and predict the customer satisfaction of the whole dialogue. Experimental results on our in-house dataset and four public datasets indicate our model outperforms all the baseline models on the dialogue-level CSP task.

In the future work, we will further improve our two-stage model by constructing more specific structures. For example, we can make differentiated design on customer part and service part in intra-stage. Moreover, we will try DialCSP or its variants on other interesting tasks in customer service dialogues, such as good dialogue mining or dialogue-level use intent detection.

7 Acknowledgments

This work was supported in part by the National Natural Science Foundation of China (Grant Nos.U1636211, 61672081,61370126), and the Fund of the State Key Laboratory of Software Development Environment (Grant No. SKLSDE-2021ZX-18).

References

1. Kim, Y.: Convolutional neural networks for sentence classification. In: EMNLP 2014
2. Nair, V., Hinton, G.E.: Rectified linear units improve restricted boltzmann machines. In: ICML 2010
3. Zhou, X., Li, L., Dong, D., Liu, Y., Chen, Y., Zhao, W.X., Yu, D., Wu, H.: Multi-turn response selection for chatbots with deep attention matching network. In:ACL 2018
4. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I.: Attention is all you need. In: NIPS 2017
5. Poria, S., Hazarika, D., Majumder, N., Naik, G., Cambria, E., Mihalcea, R.: MELD: A multimodal multi-party dataset for emotion recognition in conversations. In: ACL 2019
6. Zahiri, S.M., Choi, J.D.: Emotion detection on TV show transcripts with sequence-based convolutional neural networks. In: The Workshops of AAAI 2018

7. Srivastava, N., Hinton, G.E., Krizhevsky, A., Sutskever, I., Salakhutdinov, R.: Dropout: a simple way to prevent neural networks from overfitting. J. Mach. Learn. Res., 15(1):1929–1958.(2014)
8. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. In: ICLR 2015
9. Hashemi, S.H., Williams, K., Kholy, A.E., Zitouni, I., Crook, P.A.: Measuring user satisfaction on smart speaker intelligent assistants using intent sensitive query embeddings. In: CIKM 2018
10. Hazarika, D., Poria, S., Zadeh, A., Cambria, E., Morency, L., Zimmermann, R.: Conversational memory network for emotion recognition in dyadic dialogue videos. In: NAACL-HLT 2018
11. Ghosal, D., Majumder, N., Poria, S., Chhaya, N., Gelbukh, A.F.: Dialoguegcn: A graph convolutional neural network for emotion recognition in conversation. In: EMNLP-IJCNLP 2019
12. Shen, W., Wu, S., Yang, Y., Quan, X.: Directed acyclic graph network for conversational emotion recognition. In: ACL/IJCNLP 2021
13. Song, K., Bing, L., Gao, W., Lin, J., Zhao, L., Wang, J., Sun, C., Liu, X., Zhang, Q.: Using customer service dialogues for satisfaction analysis with context-assisted multiple instance learning. In: EMNLP-IJCNLP 2019
14. Yao, R., Song, S., Li, Q., Wang, C., Chen, H., Chen, H., Zeng, D.D.: Session-level user satisfaction prediction for customer service chatbot in e-commerce (student abstract). In: AAAI 2020
15. Song, K., Gao, W., Chen, L., Feng, S., Wang, D., Zhang, C.: Build emotion lexicon from the mood of crowd via topic-assisted joint non-negative matrix factorization. In: SIGIR 2016
16. Ma, D., Li, S., Zhang, X., Wang, H.: Interactive attention networks for aspect-level sentiment classification. In: IJCAI 2017
17. Song, K., Feng, S., Gao, W., Wang, D., Yu, G., Wong, K.: Personalized sentiment classification based on latent individuality of microblog users. In: IJCAI 2015
18. Yang, J., Yang, R., Wang, C., Xie, J.: Multi-entity aspect-based sentiment analysis with context, entity and aspect memory. In: AAAI 2018
19. Cerisara, C., Jafaritazehjani, S., Oluokun, A., Le, H.T.: Multi-task dialog act and sentiment recognition on mastodon. In: COLING 2018
20. Majumder, N., Poria, S., Hazarika, D., Mihalcea, R., Gelbukh, A.F., Cambria, E.: Dialoguernn: An attentive RNN for emotion detection in conversations. In: EAAI 2019
21. Yang, Z., Li, B., Zhu, Y., King, I., Levow, G., and Meng, H.: Collaborative filtering model for user satisfaction prediction in spoken dialog system evaluation. In: SLT 2010
22. Jiang, J., Hassan Awadallah, A., Jones, R., Ozertem, U., Zitouni, I., Gurunath Kulkarni, R., and Khan, O. Z.: Automatic online evaluation of intelligent assistants. In: WWW 2015
23. Pennington, J., Socher, R., Manning, C.D.: Glove: Global vectors for word representation. In: EMNLP 2014
24. Kachuee, M., Yuan, H., Kim, Y.B., Lee, S.: Self-supervised contrastive learning for efficient user satisfaction prediction in conversational agents. In: NAACL-HLT 2021
25. Devlin, J., Chang, M., Lee, K., Toutanova, K.: BERT: pre-training of deep bidirectional transformers for language understanding. In: NAACL-HLT 2019
26. Xu, Y., Zhao, H.: Dialogue-oriented pre-training. In: Findings of the Association for Computational Linguistics: ACL/IJCNLP 2021