

A Supplemental Material

Data Statistics

	SubD1	SubD2	SubD3	SubD4	SubD5
# of Scenarios per Dialogue	5.2	5.2	4.6	3.7	3.9
# of Utterances per Dialogue	11.7	11.7	10.6	9.8	12.4
# of Tokens per Utterance	3.8	4.0	4.1	4.3	5.1
# of Paraphrases per Query	8.9	7.0	6.5	6.9	6.9
Vocab Size after Preprocessing	194	253	303	430	620
# of Products	50				
Training Dialogues	20000				
Validation Dialogues	5000				
Test Dialogues	5000				

Table 1: Data statistics of each sub-dataset.

SubD1	query product information, query payment methods, query express information
SubD2	<i>scenarios of T1</i> , verify product information
SubD3	<i>scenarios of T2</i> , compare two products
SubD4	<i>scenarios of T3</i> , ask for an invoice, consult system error, consult nfc error, consult network error, return goods, exchange goods, query logistics
SubD5	<i>scenarios of T4</i> , express positive emotion, express negative emotion

Table 2: The user needs covered in each sub-datasets.

Product Attributes	price, discount, discount URL, operating system, thickness, color, size, weight, material, network, nfc, generation
Other Attributes	payment methods, delivery time, express company

Table 3: All attributes in the KB.

Dialogue examples

Our original corpus is in Chinese and we translate them into English. We perform the following data preprocessing. First, we use special tokens (e.g., \$orderNumber\$ and \$address\$) to anonymize all identification information. Second, to reduce the size of vocabulary and candidates, all entities are replaced with the orders in which they appear in the dialogue. Third, we replace detailed KB results with special tokens (e.g., “The price of \$entity_order_1\$ is 1,000 CNY now.” is replaced with “\$sapi call price entity_order_1\$”). The special tokens \$sapi call\$ mean calling external services. Detailed examples are shown as follows.

Table 4: Sample dialogues of SubD5. Each dialogue consists of multiple user needs (e.g., the third sample dialogue consists of two user needs: “consult system error” and “exchange goods”). Different user needs involve different business logics.

User:	Is \$entity_order_0\$ better than \$entity_order_1\$?
System:	Which aspect do you mean?
User:	Network.
System:	\$sapi call network entity_order_0 entity_order_1\$.
User:	How about \$entity_order_2\$?
System:	\$sapi call network entity_order_2\$.
User:	Wow! I can’t believe \$entity_order_2\$ supports 4G service!
System:	Of course. Our product is the most advanced.
User:	So which one is cheaper? \$entity_order_0\$ or \$entity_order_2\$?
System:	\$sapi call price entity_order_0 entity_order_2\$.
User:	Well, \$entity_order_0\$ is a little bit expensive for me.
System:	You get what you pay for!
User:	How about \$entity_order_1\$?
System:	\$sapi call price entity_order_1\$.

User:	The operating system always breaks down! I can't stand it. I request to send the product back.
System:	I'm so sorry to trouble you, we will do our best to improve the operating system. Please tell me your order number.
User:	\$orderNumber\$.
System:	Please send the item to this address: \$address\$.

User:	The operating system breaks down. What should I do?
System:	You can try to update the system.
User:	The system always has problems. I'm very disappointed.
System:	I'm so sorry to give you trouble, we will do our best to improve the operating system.
User:	I don't know how to update the system.
System:	Please refer to this: \$api call update system\$.
User:	It looks so troublesome. I want to send the product back.
System:	Please tell me your order number.
User:	My order number is \$orderNumber\$.
System:	Please send the item to this address: \$address\$.

User:	I want to exchange the product. There is a problem with the material.
System:	Please tell me your name.
User:	\$name\$. By the way, can you help me change the color to red?
System:	No problem. Please tell me your phone number.
User:	\$phone\$.
System:	Please tell me your address.
User:	\$address\$.
System:	Please tell me your order number.
User:	\$orderNumber\$.
System:	Please send the item to this address: \$address\$.