System-Initiated Transitions from Chit-Chat to Task-Oriented Dialogues with Transition Info Extractor and Transition Sentence Generator

Ye Liu[1,3], **Stefan Ultes**[2], **Wolfgang Minker**[3] and **Wolfgang Maier**[1]
[1]Mercedes-Benz AG, Sindelfingen, Germany
{ye.y.liu, wolfgang.mw.maier}@mercedes-benz.com
[2]University of Bamberg, Bamberg, Germany
stefan.ultes@uni-bamberg.de
[3]Ulm University, Ulm, Germany
{ye.liu, wolfgang.minker}@uni-ulm.de

Abstract

In this work, we study dialogue scenarios that start from chit-chat but eventually switch to task-related services, and investigate how a unified dialogue model, which can engage in both chit-chat and task-oriented dialogues, takes the initiative during the dialogue mode transition from chit-chat to task-oriented in a coherent and cooperative manner. We firstly build a *transition info extractor* (TIE) that keeps track of the preceding chit-chat interaction and detects the potential user intention to switch to a task-oriented service. Meanwhile, in the unified model, a *transition sentence generator* (TSG) is extended through efficient Adapter tuning and transition prompt learning. When the TIE successfully finds task-related information from the preceding chit-chat, such as a transition domain ("train" in Figure 1), then the TSG is activated automatically in the unified model to initiate this transition by generating a transition sentence under the guidance of transition information extracted by TIE. The experimental results show promising performance regarding the proactive transitions. We achieve an additional large improvement on TIE model by utilizing Conditional Random Fields (CRF). The TSG can flexibly generate transition sentences while maintaining the unified capabilities of normal chit-chat and task-oriented response generation.

1 Introduction

Spoken dialogue systems (SDSs) have usually been developed targeting only one out of two different categories, task-oriented or chit-chat (aka open-domain). The former focuses on achieving functional goals and the latter aims at creating engaging social conversations without special goals. In recent years, several previous works (Lin et al., 2021; Zhao et al., 2021; Young et al., 2022) have studied unified conversational models that can engage in both chit-chat and task-oriented dialogue. However, the system-initiated transitions that emerge during switchover between these two dialogue modes have rarely been explored. Especially when a user chats casually with the dialogue system, but implicitly expresses a need for a specific task-related service, it is desired that the dialogue system is able to capture this hidden information and proactively ask the user if they require such a task-oriented service (like booking a train ticket in Figure 1). It has been proven to be beneficial for commercial SDSs to proactively offer or sell their task-related services (Chiu et al., 2022; Liu et al., 2023). Furthermore, these transitions smoothly initiated by the dialogue system are regarded as a proactive feature (Nothdurft et al., 2015) and can greatly improve the user interaction experience (Liu et al., 2022).

The goal of this work is to develop the initiative capabilities of a unified conversational model that is capable of detecting the implicit user intention of using some task-related services, even if they are talking casually, and to proactively bridge the connection from chit-chat to task-oriented dialogue through generating a transition sentence (red in Figure 1). As the dialogue example in Figure 1 shows, the original response at the transition turn is only "I see". If the agent can anticipate in advance that the user wants to visit the "London Kings Cross" through the preceding chit-chat, it can then proactively establish a connection with the task-oriented "train" service that the user needs by saying "If you want, I can look for a train to London Kings Cross for you.".

To enable the initiative capabilities in a unified model, the main contributions of this paper are as follows:

1. To detect the hidden task-related transition domain/slot/value entities from the preceding chit-chat, we propose the transition info extractor (TIE) to keep track of preceding chit-chat dialogue through leveraging natural language understanding (NLU) technology (Chen et al., 2019).

Proceedings of the 16th International Natural Language Generation Conference, pages 279–292
September 11–15, 2023.

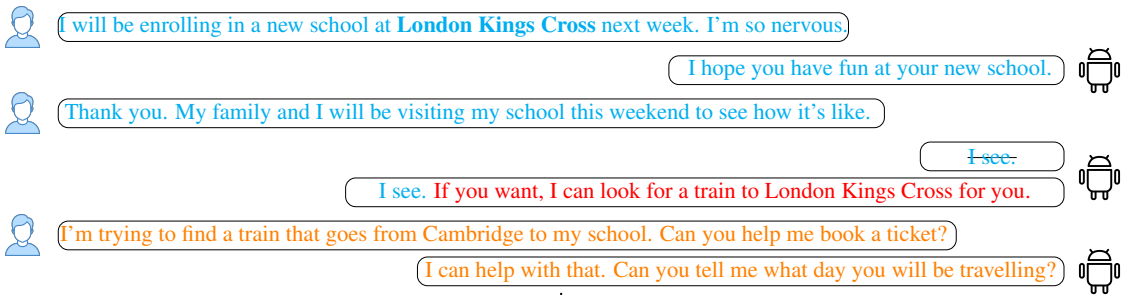

Figure 1: A Prepended FusedChat dialogue with an augmented transition sentence (red) for the proactive transition from chit-chat to task-oriented. The blue and orange represents chit-chat and task-oriented interaction respectively. Compared with the original chit-chat (crossed out) response at the transition turn, the transition sentence (red) can enable the dialogue system to proactively switch to task-oriented services.

2. We artificially augment 215 dialogues with a domain guided transition sentence and a domain-slot-value guided transition sentence respectively. We then collect transition sentence templates for different domains and different domain-slot pairs from these human augmented dialogues. The transition sentence templates are further utilized to annotate the remaining unannotated dialogues.

3. We leverage transition prompt learning (Li et al., 2022) and Adapter tuning (Lin et al., 2021) to efficiently extend the transition sentence generation (TSG) in a unified NLG model with the augmented dialogues.

The overall architecture flow of this work is shown in Figure 2. When the TIE successfully extracts the transition information from the preceding chit-chat, the TSG in the unified NLG is activated to generate a transition sentence besides the normal response to proactively guide this switch. The combined flow is highlighted in red. Otherwise, the TIE continually tracks the chit-chat, and unified NLG works as usual to generate a normal chit-chat or task-oriented response without (w/o) a transition sentence.

The remainder of this paper is structured as follows: Section 2 shows related work of our research. Section 3 presents the transition sentence augmentation and templates for the TSG training. Section 4 introduces the proposed TIE model for detecting the task-related transition information from the preceding chit-chat interaction. Section 5 presents the unified NLG extended with TSG through transition prompt and Adapter tuning. Section 6 elaborates on the performance evaluation of this work. Section 7 concludes this work and outlines future research.

2 Related Works

NLU is generally a crucial component in task-oriented SDSs and responsible for parsing an utterance into a semantic frame to identify the user's intention (De Mori et al., 2008). With the development of deep learning methods, RNN, CNN, as well as their variations or combinations have been widely for the NLU task (Yao et al., 2013; Mesnil et al., 2014; Yao et al., 2014; Liu and Lane, 2016). Wang et al. (2018) proposed a attention-based encoder-decoder, CNN-BLSTM, for joint intent detection and slot filling. Goo et al. (2018) proposed a slot gate that focused on capturing the relationship between slot and intent. Kenton and Toutanova (2019) and Xu et al. (2020) both used the pre-trained BERT for the joint intent classification and slot filling. The proposed TIE is inspired by NLU modeling.

Beyond that, Xu and Sarikaya (2013) and Ma and Hovy (2016) both utilized the traditional approach, Conditional Random Fields (CRF) (Sha and Pereira, 2003), for sequence labelling with the combination of LSTM and CNN. We also leverage the CRF technology to further improve the performance of the TIE model.

Shuster et al. (2020) introduced the dodecaDialogue task, to assemble important aspects of an engaging conversational agent into a single collection by leveraging 12 tasks. Adapter-Bot (Lin et al., 2021) utilized multiple adapter layers with the pre-trained DialoGPT model to activate new response skills and styles. Zhao et al. (2021) proposed a dialogue model for training chit-chat and task-oriented in a unified data schema, which both include belief states, representation of dataset results, and system acts. However, these models simply fuse chit-chat dialogue and task-oriented dialogue into

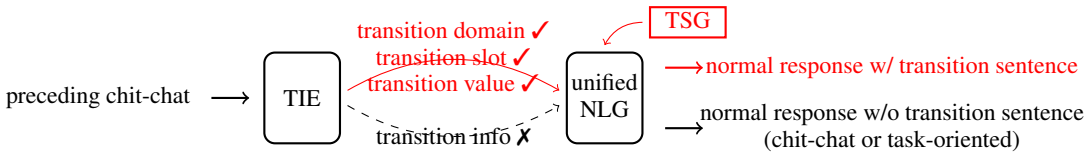

Figure 2: The overall Architecture flow for system-initiated transitions from chit-chat to task-oriented.

one model and do not consider the dependency between different types of dialogues in the multi-turn setting. In contrast, all dialogues in FusedChat released in Young et al. (2022) include both chit-chat and task-oriented turns, and treat them as parallel dialogue modes of equal importance. Chiu et al. (2022) proposed SalesBot and introduced the dialogue transitions from chit-chat to task-oriented. Liu et al. (2020) introduced the proactive transitions in conversational recommendation over multi-type dialogues. Liu et al. (2022) elaborated on three types of system-initiated transitions in a unified dialogue model and discussed the potential challenges respectively. Liu et al. (2023) proposed the system-initiated transitions between chit-chat and task-oriented dialogues, where the transitions from chit-chat to task-oriented and from task-oriented to chit-chat were treated equally. However, we mainly investigate the system-initiated transitions from chit-chat to task-oriented with the Prepended FusedChat dataset for this work.

3 Transition Sentence Augmentation and Templates

This section introduces the details of human augmentation of transition sentences and template collection for unannotated dialogues.

We mainly utilize the **Prepended** FusedChat (Young et al., 2022) dataset for initiative transitions from chit-chat to task-oriented in this work. FusedChat is a public available dataset, where human augmented open-domain dialogues are prepended and appended to the dialogues of the popular task-oriented dataset MultiWOZ (Budzianowski et al., 2018; Ye et al., 2021). In the Prepended FusedChat, each dialogue starts with chit-chat interaction and eventually switch to task-oriented requests. Table 1 shows the statistics of the Prepended FusedChat[1] used in this work. As a prepended FusedChat example shown in Figure 1, the user controls the switch

data type	train	valid	test
dialogue size	3255	474	331

Table 1: Statistics of Prepended FusedChat.

domain	train	restaurant	attraction	taxi
number of templates	95	56	45	17

Table 2: Statistics of transition sentence templates for different domains.

to task-oriented services.[2] However, our goal is to build a proactive dialogue system that can establish a smooth transition from chit-chat to task-oriented by itself.

To achieve this, we hire one master student with computational linguistics background to augment a domain guided transition sentence and a domain-slot-value guided transition sentence (red sentence in Figure 1) for 215 Prepended FusedChat dialogues respectively. The domain guided transition sentence must explicitly include the domain information. The domain-slot-value guided transition sentence must contain the specific value extracted from the preceding chit-chat dialogue aside from the domain, as the transition sentence in Figure 1, "If you want, I can look for a train to London Kings Cross for you." with "train" domain and "London Kings Cross" value.

After the human augmentation, we collect the templates for transition sentences in different domains and different domain-slot pairs from the augmented 215 dialogues respectively. For the domain-slot-value guided transition sentences, we use "[VALUE]" to replace the specific value to collect the domain-slot templates. Table 2 and Table 3 show template statistics for different domains and domain-slot pairs, respectively. Table 8 and Table 9 in the Appendix show some template examples of transition sentences in different domains and domain-slot pairs respectively. These templates are further used to randomly annotate the remaining

[1]The FusedChat used in this work is the first version uploaded by author Young and and has minor differences to the current version of FusedChat available at https://github.com/tomyoung903/FusedChat.

[2]This is common in most of prepended FusedChat dialogues, as confirmed by manual analysis.

domain	train			restaurant		attraction		taxi	
slot	day	destination	departure	food	name	type	name	destination	departure
number of templates	22	40	35	45	11	30	15	9	8

Table 3: Statistics of transition sentence templates for different domain-slot pairs.

unannotated Prepended FusedChat dialogues. Then all Prepended FusedChat with augmented transition sentences can be used for training the extended TSG in the unified NLG.

4 Transition Info Extractor (TIE)

This section presents our TIE model that can detect potential user intention to switch to task-oriented services. As shown in Figure 3, TIE is adapted from pre-trained RoBERTa (Liu et al., 2019) and has three components, a transition domain classifier, a transition slot classifier and a slot filling layer. When the interaction starts from chit-chat, the TIE keeps track of the preceding chit-chat to predict the potential transition domain and slot, while extracting the specific value through slot filling. For instance, the transition domain-slot-value extracted in Figure 3, is "restaurant-food-Korean restaurant".

4.1 Joint RoBERTa for domain/slot classification and slot filling

We utilize the pre-trained RoBERTa (Liu et al., 2019) as the backbone TIE model for jointly predicting transition domain and corresponding slot, also extracting the specific value from the preceding chit-chat dialogue through slot filling task, as shown in Figure 3.

Given the preceding dialogue history until to the current user turn $\mathbf{x} = (x_1, x_2, ...x_n)$, the [CLS] token is inserted into the first place and [SEP] is inserted to split user utterances and system responses. The corresponding slot filling label is $\mathbf{y}^{sf} = (y_1, y_2, ...y_n)$ along with [CLS] and [SEP] tokens. The input $\mathbf{x}$ and slot filling label $\mathbf{y}$ are both padded to maximal length N of the batch data. In addition, y^d and y^s are transition domain and slot label respectively. Let $\mathcal{D} = \{(\mathbf{x}, y^d, y^s, \mathbf{y}^{sf})\}_{m=1}^{M}$ be the dataset of size M for joint RoBERTa training.

Adapted from the pre-trained RoBERTa, the final hidden states of the input are

$$h_{[\text{CLS}]}, h_{x_1}, h_{x_2}, ..., h_{x_n} = \text{RoBERTa}(\mathbf{x}) \quad (1)$$

Two classifier layers in Equation 2 are separately added on the output of [CLS] token, $h_{[\text{CLS}]}$, to predict transition domain and slot.

$$\begin{aligned} \hat{y}^d &= \textit{softmax}(\mathbf{W}^d \textit{Dropout}(h_{[\text{CLS}]}) + \mathbf{b}^d) \\ \hat{y}^s &= \textit{softmax}(\mathbf{W}^s \textit{Dropout}(h_{[\text{CLS}]}) + \mathbf{b}^s) \end{aligned} \quad (2)$$

For the domain classifier, four different transition domains,[3] train, restaurant, attraction and taxi, are collected in the Prepended FusedChat. When no explicit user intention is detected in the preceding chit-chat, the domain classifier should recognise it as "UNK" to indicate that the current dialogue turn is not a good moment to switch to task-oriented. Hence, the domain classifier is a 5 classification task.

For the slot classifier, there are six slots,[4] namely day, destination, departure, food, name and type. Also along with "UNK", the slot classifier is a 7 classification task. Some slots are shared in different domains, e.g., "name" in restaurant and attraction domains (see Table 3).

For the slot filling task, the final hidden states in Equation 1 are fed into the slot filling (*sf*) layer in Equation 3 to classify over slot filling labels.

$$\hat{\mathbf{y}}_n^{sf} = \textit{softmax}(\mathbf{W}^{sf} \textit{Dropout}(\mathbf{h}_{\mathbf{x_n}}) + \mathbf{b}^{sf}); n \in 0...N \quad (3)$$

We use the IOB (In/Out/Begin) labelling format (Ramshaw and Marcus, 1999) for the slot filling labels. The dictionary of those labels is as follows and includes 22 tokens:

- 3 special tokens, "[PAD]", "[CLS]", "[SEP]", which are aligned with RoBERTa tokenizer.
- 9 domain-slot combinations in Table 3, but every domain-slot pair is extend with prefix "B-" and "I-". E.g. "B-restaurant-food" and "I-restaurant-food" in the Figure 3. When the specific value has more than one word, the first one is labelled with prefix "B-", the remaining with prefix "I-".

[3] "hotel" also exists in Prepended FusedChat as transition domain, but only in two dialogues. We delete those two dialogues to prevent the severe imbalance between different domains.

[4] Two dialogues have "pricerange" as transition slot under restaurant domain and one dialogue has "area' as transition slot under attraction domain. We also remove these dialogues in case of the imbalanced slots.

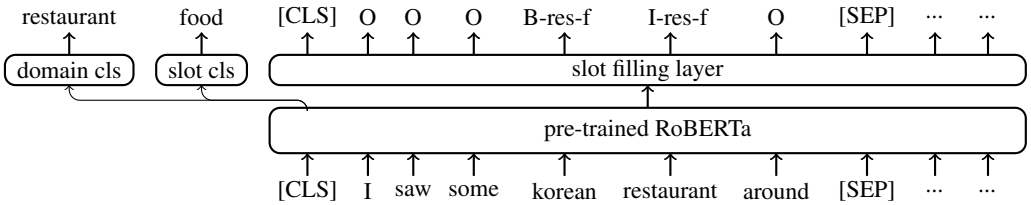

Figure 3: Architecture of the proposed TIE model that includes transition domain/slot classifier and slot filling task. The "B-res-f" and "I-res-f" is the abbreviation of the "B-restaurant-food" and "I-restaurant-food" respectively.

- The "O" is assigned to words not belonging to any specific value in sentences.

The $\mathbf{W}$ and $\mathbf{b}$ in Equation 2 and 3 are a trainable weight matrix and a bias vector. RoBERTa is jointly fine-tuned via minimizing the sum of cross-entropy loss of domain, slot classifier and slot filling task, as shown in Equation 4.

$$l_{joint\ RoBERTa} = \sum_{M}(||\hat{y}^d - y^d||^2 + ||\hat{y}^s - y^s||^2 + \sum_{n=0}^{N}||\hat{\mathbf{y}}_n^{sf} - \mathbf{y}^{sf}||^2) \quad (4)$$

4.2 Conditional Random Fields (CRF)

Beyond joint RoBERTa training for the transition domain/slot classification and slot filling tasks, we also use Conditional Random Fields (CRF) (Lafferty et al., 2001), to model the slot filling sequence jointly instead of decoding each slot filling label independently. CRF has been successfully used to exploit the dependencies within sequence labels corresponding to surrounding words and can highly improve the performance of slot filling task (Ma and Hovy, 2016). In this work, the dropout layer is applied before feeding RoBERTa outputs into CRF layer. The Viterbi algorithm is used for decoding.

We only utilize the preceding chit-chat part of Prepended FusedChat for joint RoBERTa training. To better analyse the proposed TIE model, three different TIE models are trained. As shown in Table 4, "RoBERTa w/o slot filling" only includes transition domain and slot classifiers; "joint RoBERTa" is jointly trained with domain, slot classifier and slot filling task together; and finally "joint RoBERTa + CRF" is our proposed final model, where the CRF is used for the slot filling task. All models are trained with two GPUs, the learning rate is $5e{-}5$ and batch size is 32. The best model of RoBERTa w/o slot filling is saved at epoch 5 with early stopping. The joint RoBERTa is saved at epoch 4 and joint RoBERTa + CRF at epoch 3.

5 Unified NLG extended with Transition Sentence Generator (TSG)

This section firstly introduces the unified NLG model that can reply to both chit-chat and task-oriented requests. Then we mainly elaborate on the TSG integrated in unified NLG through efficient Adapter tuning and transition prompt technologies. The extended NLG with TSG can generate a transition sentence given the transition information extracted by TIE to enable the system-initiated transition. The details of unified NLG and the extension with TSG are shown in Figure 4.

5.1 Unified NLG

We briefly presents the unified NLG model. By leveraging the entire FusedChat dataset (Young et al., 2022), where every dialogue includes interdependent chit-chat and task-oriented interaction, we tackle the unified generation problem through fine-tuning conditional GPT-2 (Radford et al., 2019). Given the FusedChat dataset $\mathcal{D}' = \{(u_g, d_g, r_g)_{g=1}^{G}, (u_l, r_l)_{l=1}^{L}\}$ with G task-oriented samples (orange in Figure 4) and L chit-chat samples (blue in Figure 4), the goal is to build a unified model parameterized by θ to be able to respond to both chit-chat and task-oriented requests,

$$p_\theta(r) = \begin{cases} \prod_{t=1}^{T} p_\theta(r_t|r_{<t}, u, d) & \text{if task-oriented} \\ \prod_{t=1}^{T} p_\theta(r_t|r_{<t}, u) & \text{if chit-chat} \end{cases} \quad (5)$$

where $r_{<t}$ indicates all tokens before t. The u represents the dialogue context; d means the dialogue actions only exist in task-oriented data and r is the system response which includes $(r_1, ...r_t, ...)$ tokens with length T.

During the unified GPT-2 fine-tuning, we add [USER] and [SYSTEM] to the GPT-2 tokenizer to distinguish user utterances from system responses. At most three preceding dialogue turns are used as the dialogue context for response generation because of memory constraints. During training,

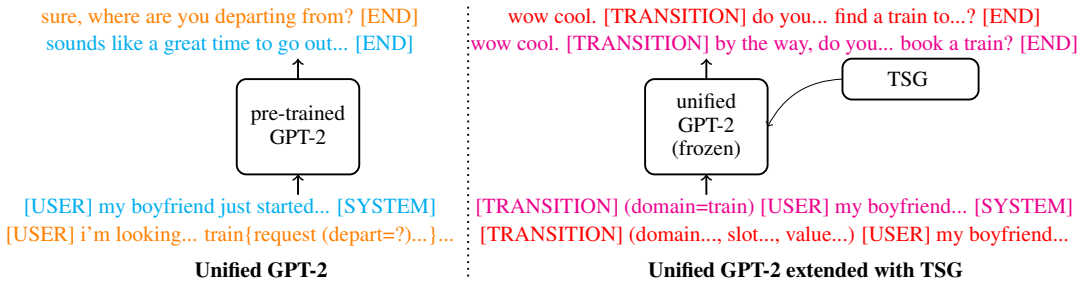

Figure 4: Architecture of unified GPT-2 and extended version with integrated TSG via Adapter tuning and transition prompt. In the unified GPT-2, the orange and blue represents the task-oriented and chit-chat example respectively. Two transition scenarios for each dialogue are used as training data for the TS Adapter tuning, one is only transition domain (magenta) as prompt, the other is transition domain-slot-value (red) as prompt.

the learning rate is $5e{-}5$, batch size is 20. The best model is saved at epoch 6 with early stopping. We mix top-K sampling and top-p (nucleus) sampling (Holtzman et al., 2019) for decoding. We apply top-K of 5 and top-p of 0.9 for chit-chat response generation and top-K of 10 and top-p of 0.5 for task-oriented response generation respectively.

5.2 Transition Prompt and Adapter Tuning

To enable the proactive capabilities, we integrate the efficient Adapter layers (Houlsby et al., 2019; Pfeiffer et al., 2021) into the unified GPT-2. Adapter tuning freezes the parameters of a pre-trained model and injects lightweight modules between layers (Le et al., 2021) to enable a new capability. Hence, the original performance of unified NLG for generating normal responses is retained without any loss. Meanwhile, the capability of generating transition sentences is extended through activating the newly added Adapter layers. To further explicitly control the transition sentence generation, the prompt learning (Liu et al., 2021; Li et al., 2022) is used. More precisely, when the TIE model successfully detects the user intention requiring a task-related service, the integrated Adapter layers are activated meanwhile the transition information extracted via TIE is converted into prompt input to generate a transition sentence to proactively establish the transition from chit-chat to task-oriented.

5.2.1 Transition Prompt

Prompt learning can efficiently adapt a given task to pre-trained models without modifying the structure of models (Lester et al., 2021). In this work, we only convert the task-related transition information extracted by TIE to the transition prompt, which is a part of the input for the generation model that explicitly guides the transition sentence generation.

We add a special token [TRANSITION] into the GPT-2 tokenizer and insert this token into the first place of the task-related transition prompt. Two different types of transition prompt are as follows:

1. When only the transition domain information is available, the prompt is like "[TRANSITION] (domain = train) ", where "train" is the extracted transition domain (magenta input in Figure 4).

2. When transition domain, slot and value are all extracted via TIE model, then the prompt is like "[TRANSITION] (domain = train, slot = destination, value = Norwich) ", where the transition domain is "train", slot is "destination" along with the value "Norwich" (red input in Figure 4).

The dialogue context is prepended with the transition prompt to be the input of the generation model. In addition, [TRANSITION] is also used to separate the transition sentence from normal response at transition turn (responses of magenta and red examples in Figure 4). Hence, the [TRANSITION] in prompt inputs is a signal for the generation model that it is a good moment to guide the transition to task-oriented service because TIE extracts task-related information, while the [TRANSITION] in generated responses is a signal to demonstrate that the NLG model is able to generate a transition sentence for proactive transition.

5.2.2 TSG through Adapter Tuning

We utilize the AdapterHub (Pfeiffer et al., 2020), which is a framework that can easily integrate Adapters into pre-trained Transformer-based models (Vaswani et al., 2017). The Houlsby Adapter (Houlsby et al., 2019) includes two bottleneck adapters in each transformer layer, one after the

multi-head attention sub-layer and the other after the feed-forward sub-layer. The Pfeiffer Adapter (Pfeiffer et al., 2021) only includes the adapter after the feed-forward sub-layer. Only 1% (Pfeiffer) and 2% (Houlsby) parameters are updated during Adapter tuning with frozen unified GPT-2. Hence, we can efficiently integrate the transition sentence generation into the unified GPT-2, while keeping the original capabilities of generating normal chit-chat and task-oriented responses by deactivating the Adapter layers.

Only the generation at the transition turn is utilized for the training of TSG. Every dialogue has two transition cases: One only consists of transition domain as prompt (magenta input in Figure 4) and the other consists of transition domain-slot-value as prompt (red input in Figure 4). We prepend the transition prompt before the preceding chit-chat context as input. The response includes a normal chit-chat response as well as a transition sentence separated with [TRANSITION] (the response of red and magenta examples in Figure 4). For the TSG, the Houlsby and Pfeiffer Adapters are both trained with the learning rate $5e{-}5$, batch size 20. The best models are both saved at epoch 16 (early stopping). We apply top-K of 5 and top-p of 0.9 for the response generation at the transition turn.

6 Results Comparison

This section evaluates this work and provides detailed performance comparison from different perspectives. We firstly evaluate different TIE models and different generation models separately with test Prepended FusedChat. Then we further evaluate the combined performance of the best TIE model and generation model only at transition turns.

6.1 TIE models

Table 4 shows the performance comparison of different TIE models. We use classification accuracy and weighted F1 score to evaluate the performance of transition domain and slot classifiers. Slot filling F1 (sf_f1) score is widely used to evaluate the slot filling task (Chen et al., 2019). In addition, we also use sentence-level slot filling accuracy (sen_sf_acc), which is the ratio of the number of dialogues correctly labelled slot filling to the total number of dialogues. The overall performance of the TIE model is evaluated using sentence-level semantic accuracy (semantic_acc) (Yu et al., 2010; Weld et al., 2021) which measures the proportion of the correctly predicted triples of transition domain, slot, and extracted slot filling values (including "O" labels).

The performance comparison in Table 4 demonstrates that joint RoBERTa with CRF as the TIE model achieves the best performance over transition domain classifier, slot classifier and slot filling task. It is surprising that not only the slot filling task benefits from the CRF. The performance of transition domain and slot classifiers is improved in the multi-task learning as well.

6.2 Generation models

To evaluate generated chit-chat responses, Distinct-1 (Dis-1) and Distinct-2 (Dis-2) (Li et al., 2016) are used to measure the proportion of the distinct unigrams and bigrams in all the generated results to indicate diversity. To evaluate generated task-oriented responses, two *N*-gram matching metrics, BLEU (Papineni et al., 2002) and Meteor (Banerjee and Lavie, 2005) are used to evaluate the overall quality of task-oriented generations. In addition, a machine learned automatic metric, BERTScore (Zhang et al., 2019) is also utilized to evaluate task-oriented and transition sentence generations.

Beyond that, we propose several automatic metrics to evaluate transition sentence generations. *Transition accuracy* detects whether the generated response at transition turn includes the [TRANSITION] special token. With [TRANSITION], we can split the transition sentence from the normal response. This metric can measure high-level capability if the model can generate a transition sentence to proactively switch to a task-oriented service. *d accuracy* detects if the domain guided transition sentence includes the specific domain keyword. *d-v accuracy* detects if the transition domain-slot-value guided transition sentence includes specific domain and value keywords both. *d accuracy* and *d-v accuracy* can evaluate the capability of the proposed transition prompt for explicitly controlling transition sentence generation to a large extent.

We found that almost all generated transition sentences by TSG with TP are of high quality and include the extracted transition information (several cases are shown in Table 7 in Appendix), instead of generic transition responses like "Do you need anything else?" or "Do you need some help?".

To better understand the performance of our models, we also retrain the unified GPT-2 without

	domain classifier		slot classifier		slot filling		
	accuracy	weighted f1	accuracy	weighted f1	sen_sf_acc	sf_f1	semantic_acc
RoBERTa w/o slot filling	78.57%	79.57%	66.52%	66.84%	–	–	–
joint RoBERTa	82.41%	82.92%	71.86%	73.84%	68.02%	48.64%	61.94%
joint RoBERTa + CRF	**93.71%**	**94.15%**	**82.41%**	**82.30%**	**80.28%**	**61.82%**	**73.67%**

Table 4: Performance of transition domain/slot classification and slot filling task in different TIE models.

		Chit-Chat		Task-Oriented			domain TS			domain-slot-value TS		
		Dis-1	Dis-2	BLUE	Meteor	BERTScore (F1)	BERTScore (F1)	*transition accuracy*	*d accuracy*	BERTScore (F1)	*transition accuracy*	*d-v accuracy*
unified GPT-2		**1.74%**	**12.70%**	**34.77%**	**55.65%**	**93.20%**	–	–	–	–	–	–
retrain	w/o TP	1.67%	11.41%	32.86%	53.52%	92.91%	88.82%	98.25%	58.19%	89.29%	98.97%	30.15%
	w/ TP	1.60%	11.18%	32.58%	53.33%	92.94%	90.19%	98.43%	99.21%	91.70%	98.79%	92.63%
TSG (Houlsby)	w/o TP	**1.74%**	**12.70%**	**34.77%**	**55.65%**	**93.20%**	89.04%	98.67%	62.48%	89.40%	**99.34%**	27.19%
	w/ TP	**1.74%**	**12.70%**	**34.77%**	**55.65%**	**93.20%**	90.28%	**99.40%**	99.15%	**91.84%**	99.21%	**96.80%**
TSG (Pfeiffer)	w/o TP	**1.74%**	**12.70%**	**34.77%**	**55.65%**	**93.20%**	88.90%	97.82%	59.52%	89.33%	98.25%	25.98%
	w/ TP	**1.74%**	**12.70%**	**34.77%**	**55.65%**	**93.20%**	**90.34%**	98.13%	**99.70%**	91.83%	98.43%	96.62%

Table 5: Performance of different NLG models, including unified GPT-2 and retrained without Adapter, extended with Houslby and Pfeiffer TSG separately, and all with transition prompt (w/ TP) and w/o TP respectively.

Adapter to enable its transition sentence generation (without TSG). From the comparison between the retrained model and unified GPT-2 in Table 5, we can see that the performance on chit-chat and task-oriented response generations has a loss, even though the retrained GPT-2 is still able to generate transition sentences. In contrast, our TSG extended in unified GPT-2 through Adapter tuning can retain the original capability for chit-chat and task-oriented generations, while maintaining a better performance on transition sentence generation.In addition, the retraining is not memory-efficient, while TSG only updates the Adapter parameters with frozen GPT-2.

To better assess the effects of our proposed transition prompt method, we retrain the model and extend TSG both along with the transition prompt (w/ TP) and without the transition prompt (w/o TP) respectively. Through the comparison between w/o TP and w/ TP in different models (highlighted in gray background in Table 5), the *d accuracy* and *d-v accuracy* metrics are highly improved with transition prompt guidance. This demonstrates that transition prompt can explicitly control the transition sentence generation. The performance comparison between Pfeiffer and Houlsby Adapter tuning has no big difference, however, the Pfeiffer Adapter uses only half of the trainable parameters, and is therefore the more effective choice for this work.

6.3 Combined TIE and generation model

To better reflect the overall performance of this work, we evaluate the combined TIE and generation models at transition turns, i.e., given the preceding chit-chat, the TIE model predicts transition domain/slot and extracts values, then this generated transition information by TIE is used as the transition prompt to guide transition sentence generation at the transition turn. Table 6 shows the combined performance of TIE and unified GPT-2 with Houlsby and Pfeiffer TSG, respectively.

Given the higher domain accuracy compared to slot accuracy, it is sensible to only use domain prediction as transition information to guide transition sentence generation when generated transition slot or extracted values are not reliable. This also validates our initial idea to propose two kinds of transition prompts. Regarding the lower slot accuracy, we found that the TIE model tends to confuse "destination" and "departure" under the "train" domain; over 60% of slot misjudged dialogues are in these cases. This would further affect the overall performance of the TIE model, which is shown by the semantic_acc metric.

Each Prepended FusedChat dialogue has only one turn for the transition from chit-chat to task-oriented. We directly define this turn as the transition turn, where the initiative dialogue model proactively switches to a task-oriented service through generating a transition sentence. Also, dialogue interactions could be more sophisticated in real life and it is difficult to accurately define the most appropriate moment to initiate a proactive transition. Furthermore, it gets more complicated if there are multiple transitions in one dialogue. A further, deeper investigation of appropriate moments for

		TIE (joint RoBERTa + CRF)				Extended GPT-2 with TSG					
		domain cls	slot cls	slot filling		domain TS			domain-slot-value TS		
		accuracy	accuracy	sf_f1	semantic_acc	BERTScore (F1)	*transition accuracy*	*d accuracy*	BERTScore (F1)	*transition accuracy*	*d-v accuracy*
TSG (Houlsby)	w/ TP	93.35%	65.56%	64.71%	50.15%	90.10%	99.40%	92.87%	91.10%	99.21%	82.78%
TSG (Pfeiffer)	w/ TP					90.08%	98.49%	93.53%	91.25%	98.37%	83.02%

Table 6: Overall performance of combined TIE and extended GPT-2 with TSG at transition turns.

a dialogue mode transition will be done in future work.

7 Conclusion

This work investigates the dialogue transition from chit-chat to task-oriented initiated by a dialogue agent. We build a TIE model adapted from pre-trained RoBERTa to keep track of the preceding chit-chat and predict transition domain, slot, while extracting the specific value from the chit-chat history via slot filling task. A unified generation model adapted from the pre-trained GPT-2 is built and extended its proactive capability for transition sentence generation through efficient Adapter tuning and transition prompt learning. Our proposed work shows promising performance both on transition information extraction and transition sentence generation. We will continue working on system-initiated transitions in other dialogue scenarios in the future.

8 Ethics Statement

This work develops proactive transitions from chit-chat to task-oriented dialogue in a unified dialogue system. Proactivity is always desired during the development of voice assistants. It can improve user interactive experience and serve users more efficiently. The dataset used in this work is public available and manually collected. Furthermore, our research is limited to a specific case, i.e, the user starts casual chat and eventually switches to a task-oriented service. However, more hidden challenges and ethics issues should be discussed further in the real scenarios. Would users prefer to be proactively served if the dialogue system successfully detects the user intention? Will they feel their privacy is violated if the dialogue system proactively provides task-related services? Such potential issues could be addressed by asking for user consent before providing the proactive interaction, which raises the additional question how many users would turn on such a feature from the start.

References

Satanjeev Banerjee and Alon Lavie. 2005. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, pages 65–72.

Paweł Budzianowski, Tsung-Hsien Wen, Bo-Hsiang Tseng, Iñigo Casanueva, Stefan Ultes, Osman Ramadan, and Milica Gasic. 2018. Multiwoz-a large-scale multi-domain wizard-of-oz dataset for task-oriented dialogue modelling. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 5016–5026.

Qian Chen, Zhu Zhuo, and Wen Wang. 2019. Bert for joint intent classification and slot filling. *arXiv preprint arXiv:1902.10909*.

Ssu Chiu, Maolin Li, Yen-Ting Lin, and Yun-Nung Chen. 2022. Salesbot: Transitioning from chit-chat to task-oriented dialogues. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6143–6158.

Renato De Mori, Frédéric Bechet, Dilek Hakkani-Tur, Michael McTear, Giuseppe Riccardi, and Gokhan Tur. 2008. Spoken language understanding. *IEEE Signal Processing Magazine*, 25(3):50–58.

Chih-Wen Goo, Guang Gao, Yun-Kai Hsu, Chih-Li Huo, Tsung-Chieh Chen, Keng-Wei Hsu, and Yun-Nung Chen. 2018. Slot-gated modeling for joint slot filling and intent prediction. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 753–757.

Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. 2019. The curious case of neural text degeneration. In *International Conference on Learning Representations*.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. 2019. Parameter-efficient transfer learning for nlp. In *International Conference on Machine Learning*, pages 2790–2799. PMLR.

Jacob Devlin Ming-Wei Chang Kenton and Lee Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL-HLT*, pages 4171–4186.

John Lafferty, Andrew McCallum, and Fernando CN Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data.

Hang Le, Juan Pino, Changhan Wang, Jiatao Gu, Didier Schwab, and Laurent Besacier. 2021. Lightweight adapter tuning for multilingual speech translation. In *The Joint Conference of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (ACL-IJCNLP 2021)*.

Brian Lester, Rami Al-Rfou, and Noah Constant. 2021. The power of scale for parameter-efficient prompt tuning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 3045–3059.

Jiwei Li, Michel Galley, Chris Brockett, Jianfeng Gao, and William B Dolan. 2016. A diversity-promoting objective function for neural conversation models. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 110–119.

Lei Li, Yongfeng Zhang, and Li Chen. 2022. Personalized prompt learning for explainable recommendation. *arXiv preprint arXiv:2202.07371*.

Zhaojiang Lin, Andrea Madotto, Yejin Bang, and Pascale Fung. 2021. The adapter-bot: All-in-one controllable conversational model. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 16081–16083.

Bing Liu and Ian Lane. 2016. Attention-based recurrent neural network models for joint intent detection and slot filling. *Interspeech 2016*, pages 685–689.

Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. 2021. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *arXiv preprint arXiv:2107.13586*.

Ye Liu, Stefan Ultes, Wolfgang Minker, and Wolfgang Maier. 2023. Unified conversational models with system-initiated transitions between chit-chat and task-oriented dialogues. *arXiv preprint arXiv:2307.01664*.

Ye Liu, Yung-Ching Yang, Wolfgang Maier, Wolfgang Minker, and Stefan Ultes. 2022. On system-initiated transitions in a unified natural language generation model for dialogue systems. In *Proceedings of the 26th Workshop on the Semantics and Pragmatics of Dialogue - Poster Abstracts*, Dublin, Ireland. SEMDIAL.

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

Zeming Liu, Haifeng Wang, Zheng-Yu Niu, Hua Wu, Wanxiang Che, and Ting Liu. 2020. Towards conversational recommendation over multi-type dialogs. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1036–1049.

Xuezhe Ma and Eduard Hovy. 2016. End-to-end sequence labeling via bi-directional lstm-cnns-crf. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1064–1074.

Grégoire Mesnil, Yann Dauphin, Kaisheng Yao, Yoshua Bengio, Li Deng, Dilek Hakkani-Tur, Xiaodong He, Larry Heck, Gokhan Tur, Dong Yu, et al. 2014. Using recurrent neural networks for slot filling in spoken language understanding. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 23(3):530–539.

Florian Nothdurft, Stefan Ultes, and Wolfgang Minker. 2015. Finding appropriate interaction strategies for proactive dialogue systems—an open quest. In *Proc. of the 2nd European and the 5th Nordic Symposium on Multimodal Communication 2014*, pages 73–80. LiU Electronic Press.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318.

Jonas Pfeiffer, Aishwarya Kamath, Andreas Rücklé, Kyunghyun Cho, and Iryna Gurevych. 2021. Adapterfusion: Non-destructive task composition for transfer learning. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 487–503.

Jonas Pfeiffer, Andreas Rücklé, Clifton Poth, Aishwarya Kamath, Ivan Vulić, Sebastian Ruder, Kyunghyun Cho, and Iryna Gurevych. 2020. Adapterhub: A framework for adapting transformers. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 46–54.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Lance A Ramshaw and Mitchell P Marcus. 1999. Text chunking using transformation-based learning. In *Natural language processing using very large corpora*, pages 157–176. Springer.

Fei Sha and Fernando Pereira. 2003. Shallow parsing with conditional random fields. In *Proceedings of the 2003 Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*, pages 213–220.

Kurt Shuster, Da Ju, Stephen Roller, Emily Dinan, Y-Lan Boureau, and Jason Weston. 2020. The dialogue dodecathlon: Open-domain knowledge and image grounded conversational agents. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2453–2470.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *NIPS*.

Yufan Wang, Li Tang, and Tingting He. 2018. Attention-based cnn-blstm networks for joint intent detection and slot filling. In *Chinese Computational Linguistics and Natural Language Processing Based on Naturally Annotated Big Data*, pages 250–261. Springer.

Henry Weld, Xiaoqi Huang, Siqu Long, Josiah Poon, and Soyeon Caren Han. 2021. A survey of joint intent detection and slot filling models in natural language understanding. *ACM Computing Surveys (CSUR)*.

Puyang Xu and Ruhi Sarikaya. 2013. Convolutional neural network based triangular crf for joint intent detection and slot filling. In *2013 ieee workshop on automatic speech recognition and understanding*, pages 78–83. IEEE.

Weijia Xu, Batool Haider, and Saab Mansour. 2020. End-to-end slot alignment and recognition for cross-lingual nlu. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5052–5063.

Kaisheng Yao, Baolin Peng, Yu Zhang, Dong Yu, Geoffrey Zweig, and Yangyang Shi. 2014. Spoken language understanding using long short-term memory neural networks. In *2014 IEEE Spoken Language Technology Workshop (SLT)*, pages 189–194. IEEE.

Kaisheng Yao, Geoffrey Zweig, Mei-Yuh Hwang, Yangyang Shi, and Dong Yu. 2013. Recurrent neural networks for language understanding. In *Interspeech*, pages 2524–2528.

Fanghua Ye, Jarana Manotumruksa, and Emine Yilmaz. 2021. Multiwoz 2.4: A multi-domain task-oriented dialogue dataset with essential annotation corrections to improve state tracking evaluation. *CoRR*, abs/2104.00773.

Tom Young, Frank Xing, Vlad Pandelea, Jinjie Ni, and Erik Cambria. 2022. Fusing task-oriented and open-domain dialogues in conversational agents.

Dong Yu, Shizhen Wang, and Li Deng. 2010. Sequential labeling using deep-structured conditional random fields. *IEEE Journal of Selected Topics in Signal Processing*, 4(6):965–973.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019. Bertscore: Evaluating text generation with bert. In *International Conference on Learning Representations*.

Xinyan Zhao, Bin He, Yasheng Wang, Yitong Li, Fei Mi, Yajiao Liu, Xin Jiang, Qun Liu, and Huanhuan Chen. 2021. Unids: A unified dialogue system for chit-chat and task-oriented dialogues. *arXiv e-prints*, pages arXiv–2110.

9 Appendix

Table 7 shows several dialogue examples with generated transition sentences in this work. Table 8 and Table 9 show transition sentence templates for different domains and domain-slot pairs, respectively.

dialogue context	**User:** A group of my friends from another university is coming to visit my campus! **System:** What is the occasion? **User:** My friends are actually working together to film a short movie and they would like shoot some footage here. **System:** It seems your friends are working on quite an interesting project. **User:** I am going to meetup with them this Saturday before bringing them to my university.
response at transition turn	*[d]:* That is a good opportunity to catch up with your friends. *By the way, I can help you with the **train**.* *[d-v]:* That sounds like a great opportunity to discuss the film with your friends. *By the way, i am happy to look for a **train** on **Saturday** for you.*
dialogue context	**User:** My sister likes to watch musical plays. **System:** Nice! **User:** I am thinking of bringing her to a theatre to watch one soon since it will be her birthday next week.
response at transition turn	*[d]:* Sounds fun! *If you are looking for an **attraction**, I am happy to help.* *[d-v]:* Sounds great! *If you are finding any **theatres**, our **attraction** service can help.*
dialogue context	**User:** I think I have had enough time in Cambridge, its a small town and you really get tired of seeing the same things over and over again . **System:** How long have you been here for? **User:** I have lived here for about 3 years now, I think I want a change of fresh air.
response at transition turn	*[d]:* That sounds great. *If you want, I can look for a **train** for you.* *[d-v]:* I think change is always good. *if you are planning to travel, I am happy to look for a **train** leaving from **Cambridge** for you.*
dialogue context	**User:** Without the additional workload, I should being accompany my family on a visit now. Now i have to let my family visit Hobsons house first and meet my family there. **System:** Sorry to hear that, hope you can finish the work early. **User:** I have finished the work and left the company. I will try to meet my family by bus or taxi, whichever is faster.
response at transition turn	*[d]:* I am sure your family will have an enjoyable time there. *Shall I get a **taxi** for you getting there?* *[d-v]:* I am sure your family will understand. , *By the way, if you want to book a taxi to **hobsons house**, feel free to use our **taxi** service.*

Table 7: Several dialogue examples with transition sentence (highlighted in red) generated by the extended NLG with TSG. The *[d]* means only the transition domain as transition prompt and *[d-v]* means the transition domain-slot-value as transition prompt to guide the transition sentence generation. Transition domains and values present in transition sentences are highlighted in **bold**.

domain	templates of the transition sentence
restaurant	I am happy to give recommendation on restaurants. I can recommend some restaurants if you want. Do you want my recommendation on the restaurants? I can also provide you more information on this restaurant. Maybe you would like to use our restaurant service to know more about it. ...
attraction	By the way, you can reach to our attraction service to know more about this place. Besides, our attraction service provides various information. I can recommend some attractions to you. By the way, have you checked out our attraction service to know more about this place? If you are finding any attraction, I am always happy to help. ...
train	Additionally I could help with looking for train tickets for you. By the way, I can help you to find thee trains to get there. Let me arrange the train for you. Please refer to our train service if you need any help with the booking. I am glad to give you more information on the train. ...
taxi	Do you need help with booking a taxi to get there? Do you want me to look for a taxi for you? Do you need a taxi afterwards? Maybe you would like my help with the taxi? If you need to get there soon, I can help you book a taxi. ...

Table 8: Transition sentences templates for different domains.

domain-slot	templates of the transition sentence
restaurant-food	I am happy to give recommendation on [VALUE] restaurants. I can recommend some [VALUE] restaurants if you want. You can find more information on [VALUE] restaurants in our restaurant service. It's my pleasure to recommend some [VALUE] restaurants if you want. ...
restaurant-name	I can also provide you more information on this restaurant named [VALUE]. Maybe you would like to use our restaurant service to know more about [VALUE]. I will be more than pleasant to help with booking a table at the restaurant called [VALUE]. Feel free to ask for more information about this restaurant named [VALUE]. ...
attraction-name	By the way, you can reach to our attraction service to know more about [VALUE]. Do you want to plan your trip to [VALUE] using our attraction service? By the way, I can provide more attraction information on [VALUE]. Talking about attractions, do you need more information about [VALUE]. ...
attraction-type	Besides, our attraction service provides various information on [VALUE]. If you are looking for attraction that has [VALUE] activities, i am happy to help you. In our attraction service, you can find more information on visiting [VALUE]s. ...
train-day	Additionally I could help with looking for train on [VALUE] for you. Let me arrange the train for [VALUE] for you. If you want, you can use our service to book the train for [VALUE]. I would love to help you with the train tickets for [VALUE]. ...
train-destination	By the way, I can help you to find the trains to [VALUE]. If you want, I can look for a train to [VALUE] for you. Additionally, you can use our service to book a train to [VALUE]. ...
train-departure	I think our service might be helpful in booking the train leaving from [VALUE]. I am happy to look for a train leaving from [VALUE] for you. Shall I find you some train tickets departing from [VALUE]. ...
taxi-departure	By the way, do you need help with booking a taxi departing from [VALUE]? Do you want me to look for a taxi depart from [VALUE] for you?. Will you need my help with the taxi leaving from [VALUE]. ...
taxi-destination	Shall I get a taxi for you to get to [VALUE]? By the way, if you need a taxi to [VALUE], please feel free to use our taxi service. If you need a taxi to get to [VALUE], feel free to use our taxi service. ...

Table 9: Transition sentences templates for different domain-slot pairs. The specific values in the human augmented transition sentences are replaced by the special [VALUE] token to collect the templates.