Target-Guided Open-Domain Conversation

Jianheng Tang[1,2], **Tiancheng Zhao**[2], **Chenyan Xiong**[3], **Xiaodan Liang**[1*],
Eric P. Xing[2,4], **Zhiting Hu**[2,4*]

[1]Sun Yat-sen University, [2]Carnegie Mellon University, [3]Microsoft Research AI, [4]Petuum Inc.
{sqrt3tjh,xdliang328}@gmail.com, {tianchez,zhitingh}@cs.cmu.edu
chenyan.xiong@microsoft.com, eric.xing@petuum.com

Abstract

Many real-world open-domain conversation applications have specific goals to achieve during open-ended chats, such as recommendation, psychotherapy, education, etc. We study the problem of imposing conversational goals on open-domain chat agents. In particular, we want a conversational system to chat naturally with human and *proactively guide* the conversation to a designated target subject. The problem is challenging as no public data is available for learning such a target-guided strategy. We propose a structured approach that introduces coarse-grained keywords to control the intended content of system responses. We then attain smooth conversation transition through turn-level supervised learning, and drive the conversation towards the target with discourse-level constraints. We further derive a keyword-augmented conversation dataset for the study. Quantitative and human evaluations show our system can produce meaningful and effective conversations, significantly improving over other approaches[1].

1 Introduction

Creating intelligent agent that can carry out open-domain conversation with human is a long-lasting challenge. Impressive progress has been made, advancing from early rule-based systems, e.g., Eliza (Weizenbaum et al., 1966), to recent end-to-end neural conversation models that are trained on massive data (Shang et al., 2015; Li et al., 2015) and make use of background knowledge (Fang et al., 2018; Qin et al., 2019; Liu et al., 2018).

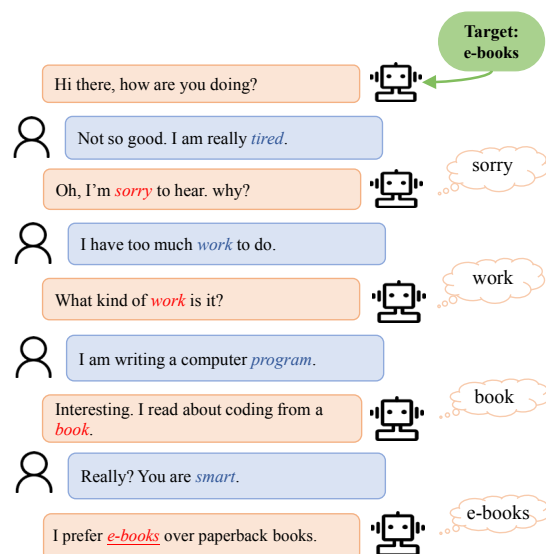However, current open-domain systems still struggle to conduct engaging conversations (Ram

Figure 1: Target-Guided Open-Domain Conversation. The agent is given a target subject *e-books* which is unknown to the human. The goal is to guide the conversation naturally to the target. Utterance keywords are highlighted in red (agent) and blue (human) and in *italic*.

et al., 2018), and often generate inconsistent or uncontrolled results. Further, many practical open-domain dialogue applications do have specific goals to achieve even though the conversations are open-ended, e.g., accomplishing nursing goals in therapeutic conversation, inspiring ideas in education, making recommendation and persuasion, and so forth. Thus, there is a strong demand to enable the integration of goals and strategy into open-domain dialogue systems, and it imposes challenges to both: first, how to define the goal for an open-domain chat system; and second, how to encode dialogue strategy into the response production process. It is also crucial to attain a general method that is not tailored towards specialized goals that require domain-specific handcrafting and annotations (Yarats and Lewis, 2018; He et al., 2018; Li et al., 2018).

---

*corresponding authors
[1]Data and code are publicly available at https://github.com/squareRoot3/Target-Guided-Conversation

This paper makes a step towards open-domain dialogue agents with conversational goals. In particular, we want the system to chat naturally with humans on open domain topics and *proactively guide* the conversation to a designated target subject. For example, in Figure 1, given a target *e-books* and an arbitrary starting topic such as *tired*, the agent drives the conversation in a natural way following a high-level logical backbone, and effectively reaches the target in the end. Such a target-guided conversation setup is general-purpose and can entail a large variety of practical applications as above. The above problem is difficult in that the agent has to balance well between chatting naturally and achieving the target; and moreover, to the best of our knowledge, there is no public dataset available for learning target-guided dialogue.

This paper proposes a solution to the task. We decouple the whole system into separate modules and address the challenges at different granularity. Specifically, we explicitly model and control the intended content of each system response by introducing coarse-grained utterance keywords. We then impose a discourse-level rule that encourages the keywords to approach the end target during the course of the conversation; and we attain smooth conversation transition at each dialogue turn through turn-level supervised learning. To this end, we further derive a keyword-augmented conversation dataset from an existing daily-life chat corpus (Zhang et al., 2018) and use it for learning keyword transitions and utterance production.

We study different keyword transition approaches, including pairwise PMI-based transition, neural-based prediction, and a hybrid kernel-based method. We conduct quantitative and human evaluations to measure the performance of sub-modules and the whole system. Our agent is able to generate meaningful and effective conversations with a decent success rate of reaching the targets, improving over other approaches in different respects. We show target-guided open-domain conversation is a promising and potentially important direction for future research.

2 Related Work

The past end-to-end dialogue research can be broadly divided into two categories: task-oriented dialogue systems and chat-oriented (a.k.a open-domain) systems. For task-oriented dialogue systems, the system is designed to accomplish specific goals, e.g., providing bus schedule (Raux et al., 2005; Young et al., 2007; Dhingra et al., 2017). Besides information giving, other tasks have been extensively studied, such as negotiations (DeVault et al., 2015; Lewis et al., 2017; He et al., 2018; Cao et al., 2018), symmetric collaborations (He et al., 2017), etc.

On the other hand, chat-oriented dialogue systems have been created to model open-domain conversations without specific goals. Prior work has been focusing on developing novel neural architectures that improve next utterance generation or retrieval task performance by training on large open-domain chit-chat dataset (Sordoni et al., 2015; Serban et al., 2016; Zhou et al., 2016; Wu et al., 2018). However, despite the steady improvement over model architectures, the current systems can still suffer from a range of limitations, e.g., dull responses, inconsistent persona (Li et al., 2016a), etc.

The commercial chatbot XiaoIce (Zhou et al., 2018) and the first Amazon Alexa challenge winner (Fang et al., 2018) have stressed to improve engagement with users. Also, to encourage discourse-level strategy, prior work has developed different system action representations that enable the model to reason at the dialogue level. One line of work has utilized latent variable models (Zhao et al., 2017; Yarats and Lewis, 2018; Zhao et al., 2019) to infer a latent representation of system responses, which separates the natural language generation process from decision-making. Another approach has created hybrid systems to incorporate hand-crafted coarse-grained actions (Williams et al., 2017; He et al., 2018) as a part of the neural dialogue systems. These systems have typically focused on specific domains such as price negotiation and movie recommendation. Building upon the prior work, this paper creates novelty in terms of both defining goals for open-domain chatting and creating system actions representations. Our structured solution use predicted keywords as a non-parametric representation of the intended content for the next system response.

Due to the lack of full supervision data, the solution proposed in this work divides the task into two competitive sub-objectives, each of which can be conquered with either direct supervision or simple rules. Such a divide-and-conquer approach

represents a general means of addressing complex task objectives with no end-to-end supervision available. A similar approach has been adopted in other contexts, such as text style transfer (Hu et al., 2017; Shen et al., 2017; Yang et al., 2018) and content manipulation (Lin et al., 2019), where content fidelity as a sub-objective is achieved with simple auto-encoding training, while the competitive nature of multiple sub-objectives jointly drives the models to learn desired behaviors.

3   Task Definition: Target-guided Open-domain Conversation

We first formally define the task of target-guided open-domain conversation. We also establish the key notations used in the rest of the paper.

Briefly, given a target, we want a chat agent to converse with human starting from an arbitrary initial topic, and lead the conversation to the target in the end. In this paper, we define a target to be a word (e.g., an entity name *McDonald*, or a common noun *book*, etc.) and denote it as t. We note that a target can also be formulated in other more complex forms depending on specific applications. The target is only presented to the agent and is unknown to the human. The conversation starts with an initial topic which is usually randomly picked by the human. At each dialogue turn where the agent wants to make a response, it has access to the conversation history consisting of a sequence of utterances by either the human or the agent, $x_{1:n} = \{x_1, \dots, x_n\}$. The agent then produces an utterance x_{n+1} as a response, aiming to satisfy (1) **transition smoothness** by making the response natural and appropriate in the current conversation context, and (2) **target achievement** by driving the conversation to reach the designated target. Specifically, we consider a target is achieved when either the human or the agent mentions the target or similar word in an utterance— such a definition is simple and allows easy measurement of the success rate. Again, other more complex and meaningful measures could be considered for specific practical applications.

The above two objectives are complementary and competitive. On one hand, an agent cannot simply bring up the target content regardless of the conversation context. For example, given a target *cat* and conversation history {*Human: I went to a movie.*}, a response like *Do you like cat?* is typically not a smooth transition, even though it

quickly reaches the target. On the other hand, the agent must avoid being trapped in open-ended chats by producing only smooth yet reactive responses. Instead, it has to proactively lead the conversation to approach the target.

The competitive nature of the two desiderata requires the agent to grasp a conversation strategy that balances well between different factors. To the best of our knowledge, there is no public large data that fits the new problem setting and permits end-to-end learning of such a discourse-level strategy in open domain. Instead, we usually only have access to those open-ended conversation data where interlocutors conversed freely without a specified end target.

To this end, we propose to break down the problem, leverage partial supervisions and introduce more structures for a solution. In the following, we first present our approach to the task (section 4), and then introduce a large open-ended conversation dataset used for building the conversational agent (section 5).

4   The Proposed Approach

We explore a solution that addresses the two desiderata separately. In particular, we maintain smooth conversation transition by turn-level supervised learning on open-domain chat data, and we inject target-guiding behavior with a rule-based guiding strategy. Further, to enable effective control over the transition and guiding strategy, we decouple the decision-making process and utterance generation by explicitly modeling the intended coarse-grained keywords in the next system utterance.

Thus the system consists of several core modules, including a turn-level keyword transition predictor (section 4.1), a discourse-level target-guiding strategy (section 4.2), and a response retriever (section 4.3).

4.1   Turn-level Keyword Transition

Given the conversation history at each dialogue turn, this module aims to predict keywords of the next response that is appropriate in the conversation context. This part is agnostic to the end target, and therefore aligns with the conventional chit-chat objective. We thus can use any open-ended chat data with extracted utterance keywords to learn the prediction module in a supervised manner. We present such a dataset that we posit is par-
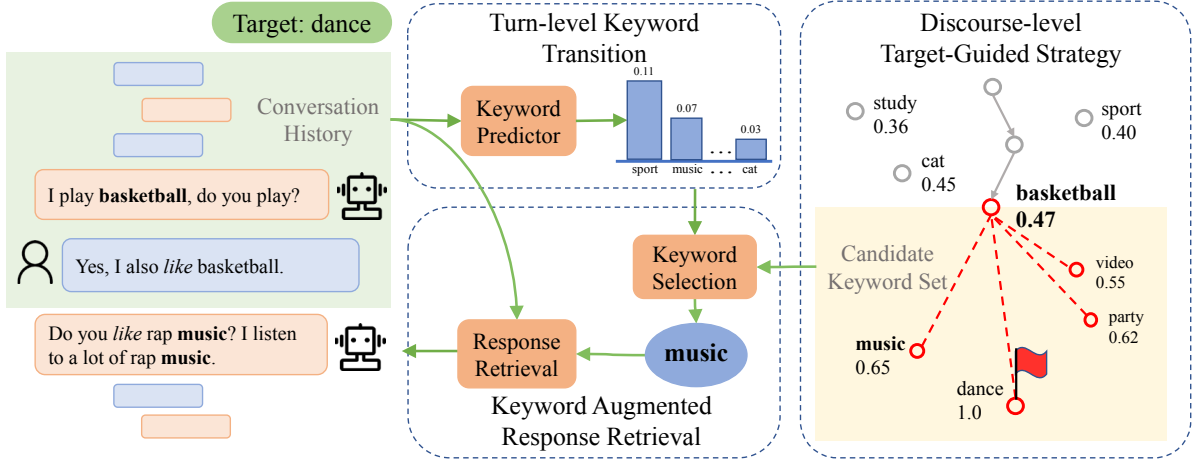
Figure 2: Solution Overview. The left panel shows an on-going conversation with a designated target *dance*. The discourse-level target-guided module (right panel, section 4.2) first picks a set of valid candidate keywords for the next system response. The turn-level keyword transition module (middle panel, section 4.1) computes a distribution over candidate keywords. The most likely valid keyword (*music*) is then selected, and fed into the keyword-augmented response retrieval module (middle panel, section 4.3) for producing the next response.

ticularly suitable for the learning in section 5.

Architecturally, we study three different approaches as representative paradigms for predicting the next-turn keyword distribution, including pairwise keyword linear transition, neural-based prediction, and kernel-based method.

Pairwise PMI-based Transition The most straightforward way for keyword transition is to construct a keyword pairwise matrix that characterizes the association between keywords in the observed conversation data. We use pointwise mutual information (PMI) (Church and Hanks, 1990) as the measure, which, given two keywords w_i and w_j, computes likeliness of $w_j \rightarrow w_i$ with

$$\text{PMI}(w_i, w_j) = \log p(w_i|w_j)/p(w_i), \quad (1)$$

where $p(w_i|w_j)$ is the ratio of transitioning to w_i in the next turn given w_j in the current turn, and $p(w_i)$ is the ratio of w_i occurrence. Both quantities can be directly counted from the conversation data beforehand. At test time, we first use a keyword extractor (section 5) to extract keywords of the current utterance. Assuming all these keywords are independent, for each candidate next keyword, we sum up their PMI scores w.r.t the candidate. The resulting candidate scores are then normalized to obtain a distribution over keywords in the next turn.

The approach enjoys simplicity and interpretability, yet can suffer from data sparsity and perform poorly with *a priori* unseen transition pairs.

Neural-based Prediction The second approach predicts the next keywords with a neural network in an end-to-end manner. More concretely, we first use a recurrent network to encode the conversation history, and feed the resulting features to a prediction layer to obtain a distribution over keywords for the next turn. The network is learned by maximizing the likelihood of observed keywords in the data. The neural approach is straightforward, but can rely on a large amount of data for learning.

Hybrid Kernel-based Method We further study a hybrid approach that combines neural feature extraction with pairwise closeness measuring. Specifically, given a pair of a current keyword and a candidate next keyword, we follow (Xiong et al., 2017) by first measuring the cosine similarity of their normalized word embeddings, and feeding the quantity to a kernel layer consisting of K RBF kernels. The output of the kernel layer is a K-dimension kernel feature vector, which is then fed to a single-unit dense layer for a candidate score. The score is finally normalized across all candidate keywords to yield the candidate probability distribution. If the current turn has multiple keywords, the corresponding multiple K-dimension kernel features are first summed up before feeding to the dense layer. Thus, the intermediate kernel layer serves as a soft aggregation mechanism to account for multiple-to-one keyword transition. The parameters are learned in the same way as in the neural-based prediction

5627

method. Our empirical study shows the hybrid approach provides the strongest performance.

4.2 Discourse-level Target-Guided Strategy

This module aims to fulfill the end target by proactively driving the discussion topic forward in the course of the conversation. As noted above, there is typically no data available for direct learning of such a strategy. Fortunately, the augmentation of interpretable coarse-grained keywords enables us to apply a simple yet effective rule to this end.

We constrain that the keyword of each turn must move strictly closer to the end target compared to those of preceding turns. Figure 2, right part, illustrates the rule at a particular step. Given the keyword *Basketball* of the current turn and its closeness score (0.47) to the target *Dance*, the only valid candidate keywords for the next turn are those with higher target closeness, such as *Party* with a closeness score of 0.62. On the other hand, transitioning from *Basketball* to *Sport* is not allowed in the context as it does not move towards the target. More concretely, we use cosine similarity between normalized word embeddings as the measure of keyword closeness.

At each turn for predicting the next keyword, the above constraint first collects a set of valid candidates, and the turn-level transition module samples or picks the most likely one the from the set according to the keyword distribution. In this way, the predicted keyword for next response can be both a smooth transition and an effective step towards the target.

4.3 Keyword-augmented Response Retrieval

The final module in the system aims to produce a response conditioning on both the conversation history and the predicted keyword. In this work, we use a retrieval-based approach, though a generation-based method can also be readily plugged in.

The architecture of the module is adapted from the previous work (Wu et al., 2016) with augmented keyword conditioning. More concretely, we use recurrent networks to encode the input conversation history and keyword, as well as each of the candidate responses from a database (e.g., all utterances in the training set). We then compute the element-wise product between the candidate feature with the history feature, and between the candidate feature with the keyword feature, respectively. The resulting two vectors are concate-

	Train	Val	Test
#Conversations	8,939	500	500
#Utterances	101,935	5,602	5,317
#Keyword types	2,678	2,080	1,571
#Avg. keywords	2.1	2,1	1.9

Table 1: Data Statistics. The last row is the average number of keywords in each utterance. The vocabulary size is around 19K.

A:	Hi ! I am from <u>India</u> . where are you from?
B:	I'm from <u>Portland</u>. I just got back from a long walk.
A:	I just got back from <u>coaching</u> <u>swimming</u> at the <u>pool</u>. Walking where ?
B:	I like to walk in <u>parks</u> for good <u>health</u>. No soft <u>drinks</u> for me either!
...	...

Table 2: An Example Conversation. Only the first 4 utterances are shown. Keywords of each utterance are marked with underline.

nated and fed to a final single-unit dense layer with sigmoid to get the matching probability of the candidate response.

Same as the turn-level transition module, the conditional response retrieval module can also be learned with open-ended conversation data in a supervised manner. That is, we maximize the likelihood of observed response given its history and predicted keyword, while minimizing the likelihood of randomly sampled negative responses. Section 5 presents more details of the data and negative responses.

5 Dataset

We next describe a large conversation dataset that can be useful for studying the task and has been used in our solution. The dataset is derived from the PersonaChat corpus (Zhang et al., 2018) where crowdworkers were asked to chat naturally with given persona. The conversations cover a broad range of topics such as work, family, personal interest, etc; and the discussion topics change frequently during the course of the conversations. <mark>These properties make the conversations particularly suitable for learning smooth, natural transitions *at each turn*.</mark> Note that, however, the conversations do not necessarily suit for learning discourse-level strategies, as they were originally created without end targets and do not exhibit target-guided behaviors.

To adapt the corpus for turn-level keyword transition in our new setting, we obtain all conversations while discarding the associated persona information. We then augment the data by automatically extracting keywords of each utterance. Specifically, we apply a rule-based keyword extractor which combines TF-IDF and Part-Of-Speech features for scoring word salience. More details are provided in supplementary materials. We re-split the data into train/valid/test sets, where the test set contains 500 conversations with relatively frequent keywords. Table 1 lists the data statistics. An example conversation with the extracted keywords is shown in Table 2.

The resulting dataset is used in our solution for training both the turn-level transition module (section 4.1) and the response retrieval module (section 4.3). We follow the retrieval-based chit-chat literature (Wu et al., 2016) and randomly sample 19 negative responses for each turn as the negative responses for training.

6 Experiments

6.1 Experimental Setup

Baselines and Comparison Systems We evaluate a diverse set of approaches for comparison and ablation study.

Retrieval (Wu et al., 2016) is the conventional retrieval-based chitchat system which does not permit an end target and is not augmented with coarse-grained utterance keywords. The system thus cannot be deployed for target-guided conversation, and is used to provide reference performance in terms of different metrics in the experiments. The model architecture is adapted from the prior work, the same as used in our full system except for the keyword conditioning part.

Retrieval-Stgy augments the above base retrieval system with the proposed target-guided strategy (section 4.2). Specifically, it first extracts the keywords of current utterance with the extractor used in section 5, and applies the target-guided rule to obtain a set of candidate keywords. The base retrieval model is then used to retrieve a response containing at least one keyword from the keyword set. Such a pipeline approach achieves a strong baseline performance, as shown in the following.

Ours As in section 4.1, our full system has several variants in the turn-level keyword transition module, including the **PMI**, **Neural**, and **Kernel**

methods. For comparison, we also use a **Random** method which randomly picks a keyword for next response.

Training Details We use the same configuration for the common parts of all agents. We apply a single-layer GRU (Chung et al., 2014) in all encoders. Both the word embedding and hidden dimensions are set to 200. We use GloVe (Pennington et al., 2014) to initialize word embeddings. We apply Adam optimization (Kingma and Ba, 2014) with an initial learning rate of 0.001 and annealing to 0.0001 in 10 epochs. Systems are implemented with a text generation toolkit Texar (Hu et al., 2019).

6.2 Turn-level Evaluation

We first evaluate the performance of each conversation turn, in terms of both turn-level keyword prediction and response selection. That is, we disable the discourse-level target constraint, and focus on measuring how accurate the systems can predict the next keyword and retrieve the correct response on the test set of the conversation data. The evaluation largely follows the protocol of previous chit-chat systems (e.g., Wu et al., 2016), and validates the effect of the keyword-augmented conversation production.

Evaluation metrics For the keyword prediction task, we measure three metrics: (1) $R_w@K$: keywords recall at position K $(= 1, 3, 5)$ in all (over 2600) possible keywords, (2) $P@1$: precision at the first position, and (3) **Cor.**: the word embedding based correlation score (Liu et al., 2016).

For the response selection task, we randomly sample 19 negative responses for each test case, and calculate $R_{20}@K$, i.e., recall at position K in the 20 candidate (positive and negative) responses, as well as **MRR**, the mean reciprocal rank.

Results Table 3 shows the evaluation results. Our system with Kernel transition module outperforms all other systems in terms of all metrics on both two tasks, expect for $R_{20}@3$ where the system with PMI transition performs best. The Kernel approach can predict the next keywords more precisely. In the task of response selection, our systems that are augmented with predicted keywords significantly outperform the base Retrieval approach, showing predicted keywords are helpful for better retrieving responses by capturing coarse-grained information of the next utterances. Inter-

System	Keyword Prediction					Response Retrieval			
	R_w@1	R_w@3	R_w@5	P@1	Cor.	R_{20}@1	R_{20}@3	R_{20}@5	MRR
Retrieval	-	-	-	-	-	0.5196	0.7636	0.8622	0.6661
Ours-Random	0.0005	0.0015	0.0025	0.0009	0.4995	0.5187	0.7619	0.8631	0.6650
Ours-PMI	0.0585	0.1351	0.1872	0.0871	0.7974	0.5441	**0.7839**	0.8716	0.6847
Ours-Neural	0.0609	0.1324	0.1825	0.1006	0.8075	0.5395	0.7801	0.8790	0.6816
Ours-Kernel	**0.0642**	**0.1431**	**0.1928**	**0.1191**	**0.8164**	**0.5486**	0.7827	**0.8845**	**0.6914**

Table 3: Results of Turn-level Evaluation.

System	Succ. (%)	#Turns
Retrieval	9.8	3.26
Retrieval-Stgy	67.2	6.56
Ours-PMI	47.4	5.12
Ours-Neural	51.6	4.29
Ours-Kernel	**75.0**	4.20

Table 4: Results of Self-Play Evaluation.

System	Succ. (%)	Smoothness
Retrieval	18	3.26
Retrieval-Stgy	66	3.24
Ours-PMI	52	3.00
Ours-Neural	56	2.94
Ours-Kernel	**76**	**3.40**

Table 5: Results of the Human Rating.

	Kernel Better(%)	No Prefer(%)	Kernel Worse(%)
Retrieval-Stgy	**34**	40	26
Ours-PMI	**52**	22	26
Ours-Neural	**56**	28	16

Table 6: Results of Human Ranking.

estingly, the system with Random transition has a close performance to the base Retrieval model, indicating that the erroneous keywords can be ignored by the system after training.

6.3 Target-guided Conversation Evaluation

We next evaluate system performance in the proposed target-guided conversation setup, with both automatic simulation-based evaluation and human evaluation.

6.3.1 Self-Play Simulation

Following the experimental settings in prior work (Lewis et al., 2017; Li et al., 2016b), we developed a task simulator to automatically produce target-guided conversations. Specifically, we use the base Retrieval agent to play the role of human which retrieves a response without knowing the end target. The simulator randomly picks a keyword as the end target, and an utterance as the starting point. Each agent then chats with the Retrieval system, trying to guide the conversation to the given target. To automatically evaluate whether the target is achieved, we use Word-Net (Miller, 1998) to identify keywords that are semantically close to the end target. More concretely, if a keyword in an utterance (by either the agent under test or Retrieval) has a WordNet information content similarity score higher than 0.9, we consider the target is successfully achieved. To avoid infinite conversation without ever reaching the target, we set a maximum allowed number of turns, which is 8 in our experiment. That is, an agent that does not achieve the target after producing 8 responses is considered to fail in the case.

We measure the success rate of achieving the targets (**Succ.**) and the average number of turns used to reach a target (**#Turns**).

Table 4 shows the results of 500 simulations for each of the comparison systems. Our system with Kernel transition obtains the highest success rate, significantly improving over other approaches. The success rate of the base Retrieval agent is lower than 10%, which proves that a chitchat agent without a target-guided strategy can hardly accomplish our task. The Retrieval-Stgy agent has a relatively high success rate, while taking more turns (6.56) to accomplish this. This is partially due to the lack of coarse-grained keyword modeling and transition. We further note that, in the Kernel system, around 81% of predicted keywords eventually occur in the produced utterances, indicating that the predicted keywords have a great impact on the retrieval module.

6.3.2 Human Evaluation

We finally perform human evaluation for a more thorough system comparison in terms of different aspects. Specifically, we use the DialCrowd toolkit (Lee et al., 2018) to setup human evaluation interfaces, and undertook two types of human studies as below.

The first evaluation is to measure the system

Agent: Retrieval-Stgy Target: **<u>Dance</u>**	Agent: Ours-Kernel Target: **<u>Dance</u>**
A: Hi, how are you today?	**A:** Hi, how are you today?
H: Fine. Just finished *riding* in the countryside.	**H:** Fine. Just finished *riding* in the countryside.
A: That sounds *relaxing*. I love the outdoors.	**A:** I love riding my bike. It is great exercise and fun. *(bike)*
H: I rode along a long black *river*.	**H:** I rode along a long black *river*.
A: Nice , do you like the *lake*? I do I go often.	**A:** That sounds very exciting. You can ride bikes, listen to musics there too. *(music)*
H: Yes. I like *swimming* in the lake.	**H:** That's great. My favorite *singer* is Taylor Swift.
A: I've never been surfing but used to do a lot of traveling across the us playing *guitar*.	**A:** Yes I love to sing! I can sing a lot of songs, can you? *(sing)*
H: Nice. I also like playing guitar and other *musical instruments*.	**H:** Yes, although I cannot *sing* well.
A: Oh that's nice, I'm more into classical *music*.	**A:** Dancing? I love *<u>dancing</u>*! Singing too!
H: I am interested in *pop music*.	
A: Oh cool! I really like music and *singing*.	
H: I enjoy *singing*. Although I cannot sing well.	
A: Music is life. *<u>Dancing</u>* to anything with a beat is the best.	

Agent: Retrieval-Stgy Target: **<u>McDonald's</u>**	Agent: Ours-Kernel Target: **<u>McDonald's</u>**
A: Hello, how are you doing?	**A:** Hello, how are you doing?
H: Hi, I am great just finishing up my *homework*. How are you?	**H:** Hi, I am great just finishing up my *homework*. How are you?
A: Oh nothing much just *reading*.	**A:** Just got done eating some chicken and rice after I got done with my gym session. *(eat)*
H: Which kind of *books* are you reading?	**H:** I am gonna eat some vegetable for *dinner*.
A: Reading comic *books*, I love them. What about you?	**A:** So you are vegetarian? I am from los angeles so there are lots of vegetarian restaurants. *(restaurant)*
H: I enjoy reading sci-fi and romantic *stories*.	**H:** I am not a vegetarian. Fried *chicken* is my favorite.
A: That sounds helpful for *writing*.	**A:** Oh boy, when I go to *<u>McDonald's</u>*, I always get fries for my 2 dogs.
H: Yeah, but I do not good at writing *articles*.	
A: Those were good books, what do you like to *eat*?	
H: *Vegetables* and fruit salad are my favorite.	
A: That reminds me of spicy *food*.	
H: Oh spicy food is also very *delicious*.	
A: Nothing better than bacon for *breakfast*.	
H: I often eating *eggs* for breakfast.	
A: I love cheese *tacos* too.	
H: It's too late now, and I'll ride my bike to *home*.	

Table 7: Example conversations between human (**H**) and two different agents (**A**), with the same targets and starting utterances. Keywords selected or predicted by the agents are highlighted in red and *italic*, and keywords mentioned by human are highlighted in blue and *italic*. As keywords predicted by the Kernel agent do not necessarily occur in the retrieved utterances, we put them to the end of each sentence. Targets achieved at the end of conversations are underlined. We present the examples in case-sensitive format for readability. All tokens are in lowercase in the program.

performance in terms of the two key desiderata, namely target achievement and transition smoothness, respectively. We first build 50 test cases, each of which has a target and a starting utterance. In each test case, a human turker is asked to converse with a randomly selected agent. The agent informs the turker when it thought the target is achieved or has reached the maximum number of turns (which is set to 8). Then the turker is presented with the designated target, and is asked to judge whether the target has been achieved, as well as rate transition smoothness during the conversation with a score ranging from 1 (strongly bad) to 5 (strongly good). All agents are evaluated on all test cases.

Table 5 shows the results of the first evaluation. Our Kernel agent clearly outperforms all other comparison systems in terms of both success rate and transition smoothness. Note that the success rate results of all agent are consistent with those in simulation (Table 4). Comparing the base Retrieval agent and the augmented Retrieval-Stgy agent, we can see that Retrieval-Stgy has almost the same smoothness with Retrieval but achieves a much higher success rate. This validates that our discourse-level strategy (section 4.2) is indeed effective for target-guided conversations.

The second evaluation compares our best-performing Kernel agents with other agents side-by-side. Specifically, we ask a human turker to converse with the Kernel agent and a randomly selected comparison agent in the same test case. We then ask the turker to rank the two conversations by considering all the criteria. Turkers can

Agent: Ours-Kernel   Target: **Listen**
A: Hi! How are you? I am very exciting about my new job!
H: Fine. What is your new *job*?
A: That sounds exciting. (*sound*)
H: I like reading *books*.
A: What kind of books do you read? (*book*)
H: I read *comics*.
A: I do not have TV. (*TV*)
H: Watching *animation* is fun.
A: What position d you play? (*play*)
H: I play a lot of *sports*.
A: Oh really? What kind of music? *Listen* to classical.

Table 8: An erroneous example.

also choose "no preference" if the conversations are equally good or bad. To avoid any bias, in each test case, we randomly pick one from the pair of agents to converse first, and we let the turker to decide when to stop to avoid revealing the target too early. As above, we evaluate on 50 test cases for each pair of agents.

Table 6 shows the results of the second evaluation. We see that our Kernel system consistently outperforms the comparison methods with 30-50% wins.

6.4 Qualitative Study

We take a close look at the model performance by studying the conversation examples from different agents in human evaluation.

Table 7 shows the conversations between human and agents given targets *dance* and *McDonald's*, respectively. We can see that, in general, our Kernel agent can accomplish the task in fewer turns than the Retrieval-Stgy agent. In the first case, the Kernel agent guides the conversation from *ride* to the crucial topic *music* smoothly and quickly, and then achieves the target word *dance* naturally. In contrast, the Retrieve-Stgy agent is trapped in open-ended chats for the first three turns and does not reach the target until the 7th turn. In the second case, the target *McDonald's* is relatively uncommon in our dataset. The kernel agent succeeded to achieve the target in the 4th turn while the Retrieval-Stgy agent failed to reach the target within the maximally allowed number of turns.

Table 8 shows a failure case by our Kernel agent. Although the agent successfully achieved the target, it sometimes makes non-smooth keyword transition without a clear logic. For instance, the final utterance of the agent, though reaching the target *listen*, is not appropriate in the conversation context (e.g., in the presence of human's preceding keyword *sports*).

7 Conclusions & Discussions

We have studied the problem of target-guided open-domain conversation, where an agent converses naturally with the human and proactively guides the conversation to a designated end target. We propose a modular solution with coarse-grained keywords as a logical backbone, and use partial supervision and heuristic rules to achieve the task. We also derive a dataset for the study. Quantitative and human evaluations demonstrate promising and improved results of our approach.

This work presents an initial attempt to bridge the gap between open-domain chit-chat and task-oriented dialogue. A target-guided agent can be deployed in practice to converse with users engagingly and guide the users to trigger task-oriented systems (e.g., reserving a restaurant) in the end. An open-domain agent with control over the conversation strategy and end target can also be useful in education, psychotherapy, and others as discussed in section 1. Our treatment of utterance action and conversation target through simple keywords can be preliminary in terms of complex real applications. It would be exciting to explore more sophisticated modeling to enable more fine-grained control on both sentence (Hu et al., 2017) and discourse levels (Williams et al., 2017; Fang et al., 2018).

References

Kris Cao, Angeliki Lazaridou, Marc Lanctot, Joel Z Leibo, Karl Tuyls, and Stephen Clark. 2018. Emergent communication through negotiation. In *ICLR*.

Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. 2014. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555*.

Kenneth Ward Church and Patrick Hanks. 1990. Word association norms, mutual information, and lexicography. *Computational linguistics*, 16(1):22–29.

David DeVault, Johnathan Mell, and Jonathan Gratch. 2015. Toward natural turn-taking in a virtual human negotiation agent. In *2015 AAAI Spring Symposium Series*.

Bhuwan Dhingra, Lihong Li, Xiujun Li, Jianfeng Gao, Yun-Nung Chen, Faisal Ahmed, and Li Deng. 2017.

Towards end-to-end reinforcement learning of dialogue agents for information access. In *ACL*.

Hao Fang, Hao Cheng, Maarten Sap, Elizabeth Clark, Ari Holtzman, Yejin Choi, Noah A Smith, and Mari Ostendorf. 2018. Sounding board: A user-centric and content-driven social chatbot. In *NAACL System Demonstrations*.

He He, Anusha Balakrishnan, Mihail Eric, and Percy Liang. 2017. Learning symmetric collaborative dialogue agents with dynamic knowledge graph embeddings. In *ACL*.

He He, Derek Chen, Anusha Balakrishnan, and Percy Liang. 2018. Decoupling strategy and generation in negotiation dialogues. *arXiv preprint arXiv:1808.09637*.

Zhiting Hu, Haoran Shi, Bowen Tan, Wentao Wang, Zichao Yang, Tiancheng Zhao, Junxian He, Lianhui Qin, Di Wang, et al. 2019. Texar: A modularized, versatile, and extensible toolkit for text generation. In *ACL System Demonstrations*.

Zhiting Hu, Zichao Yang, Xiaodan Liang, Ruslan Salakhutdinov, and Eric P Xing. 2017. Toward controlled generation of text. In *ICML*.

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

Kyusong Lee, Tiancheng Zhao, Alan W Black, and Maxine Eskenazi. 2018. DialCrowd: A toolkit for easy dialog system assessment. In *SIGDAIL*, pages 245–248.

Mike Lewis, Denis Yarats, Yann N Dauphin, Devi Parikh, and Dhruv Batra. 2017. Deal or no deal? end-to-end learning for negotiation dialogues. *arXiv preprint arXiv:1706.05125*.

Jiwei Li, Michel Galley, Chris Brockett, Jianfeng Gao, and Bill Dolan. 2015. A diversity-promoting objective function for neural conversation models. *arXiv preprint arXiv:1510.03055*.

Jiwei Li, Michel Galley, Chris Brockett, Jianfeng Gao, and Bill Dolan. 2016a. A persona-based neural conversation model. *arXiv preprint arXiv:1603.06155*.

Jiwei Li, Will Monroe, Alan Ritter, Michel Galley, Jianfeng Gao, and Dan Jurafsky. 2016b. Deep reinforcement learning for dialogue generation. *arXiv preprint arXiv:1606.01541*.

Raymond Li, Samira Ebrahimi Kahou, Hannes Schulz, Vincent Michalski, Laurent Charlin, and Chris Pal. 2018. Towards deep conversational recommendations. In *NeurIPS*, pages 9748–9758.

Shuai Lin, Wentao Wang, Zichao Yang, Haoran Shi, Frank Xu, Xiaodan Liang, Eric Xing, and Zhiting Hu. 2019. Toward unsupervised text content manipulation. *arXiv preprint arXiv:1901.09501*.

Chia-Wei Liu, Ryan Lowe, Iulian V Serban, Michael Noseworthy, Laurent Charlin, and Joelle Pineau. 2016. How not to evaluate your dialogue system: An empirical study of unsupervised evaluation metrics for dialogue response generation. *arXiv preprint arXiv:1603.08023*.

Shuman Liu, Hongshen Chen, Zhaochun Ren, Yang Feng, Qun Liu, and Dawei Yin. 2018. Knowledge diffusion for neural dialogue generation. In *ACL*, volume 1, pages 1489–1498.

George Miller. 1998. *WordNet: An electronic lexical database*. MIT press.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *EMNLP*.

Lianhui Qin, Michel Galley, Chris Brockett, Xiaodong Liu, Xiang Gao, Bill Dolan, Yejin Choi, and Jianfeng Gao. 2019. Conversing by reading: Contentful neural conversation with on-demand machine reading. In *ACL*.

Ashwin Ram, Rohit Prasad, Chandra Khatri, Anu Venkatesh, Raefer Gabriel, Qing Liu, Jeff Nunn, Behnam Hedayatnia, Ming Cheng, Ashish Nagar, et al. 2018. Conversational AI: The science behind the Alexa prize. *arXiv preprint arXiv:1801.03604*.

Antoine Raux, Brian Langner, Dan Bohus, Alan W Black, and Maxine Eskenazi. 2005. Let's go public! taking a spoken dialog system to the real world. In *Ninth European conference on speech communication and technology*.

Iulian Vlad Serban, Alessandro Sordoni, Ryan Lowe, Laurent Charlin, Joelle Pineau, Aaron Courville, and Yoshua Bengio. 2016. A hierarchical latent variable encoder-decoder model for generating dialogues. *arXiv preprint arXiv:1605.06069*.

Lifeng Shang, Zhengdong Lu, and Hang Li. 2015. Neural responding machine for short-text conversation. *arXiv preprint arXiv:1503.02364*.

Tianxiao Shen, Tao Lei, Regina Barzilay, and Tommi Jaakkola. 2017. Style transfer from non-parallel text by cross-alignment. In *NeurIPS*.

Alessandro Sordoni, Michel Galley, Michael Auli, Chris Brockett, Yangfeng Ji, Margaret Mitchell, Jian-Yun Nie, Jianfeng Gao, and Bill Dolan. 2015. A neural network approach to context-sensitive generation of conversational responses. *arXiv preprint arXiv:1506.06714*.

Joseph Weizenbaum et al. 1966. ELIZA—a computer program for the study of natural language communication between man and machine. *Communications of the ACM*, 9(1):36–45.

Jason D Williams, Kavosh Asadi, and Geoffrey Zweig. 2017. Hybrid code networks: practical and efficient end-to-end dialog control with supervised and reinforcement learning. In *ACL*.

Hua Wu, Yi Liu, Ying Chen, Wayne Xin Zhao, Daxiang Dong, Dianhai Yu, Xiangyang Zhou, and Lu Li. 2018. Multi-turn response selection for chatbots with deep attention matching network. In *ACL*.

Yu Wu, Wei Wu, Chen Xing, Ming Zhou, and Zhoujun Li. 2016. Sequential matching network: A new architecture for multi-turn response selection in retrieval-based chatbots. *arXiv preprint arXiv:1612.01627*.

Chenyan Xiong, Zhuyun Dai, Jamie Callan, Zhiyuan Liu, and Russell Power. 2017. End-to-end neural ad-hoc ranking with kernel pooling. In *SIGIR*, pages 55–64. ACM.

Zichao Yang, Zhiting Hu, Chris Dyer, Eric P Xing, and Taylor Berg-Kirkpatrick. 2018. Unsupervised text style transfer using language models as discriminators. In *NeurIPS*.

Denis Yarats and Mike Lewis. 2018. Hierarchical text generation and planning for strategic dialogue. In *ICML*.

Stephanie Young, Jost Schatzmann, Karl Weilhammer, and Hui Ye. 2007. The hidden information state approach to dialog management. In *Acoustics, Speech and Signal Processing, 2007. ICASSP 2007. IEEE International Conference on*, volume 4, pages IV–149. IEEE.

Saizheng Zhang, Emily Dinan, Jack Urbanek, Arthur Szlam, Douwe Kiela, and Jason Weston. 2018. Personalizing dialogue agents: I have a dog, do you have pets too? *arXiv preprint arXiv:1801.07243*.

Tiancheng Zhao, Kaige Xie, and Maxine Eskenazi. 2019. Rethinking action spaces for reinforcement learning in end-to-end dialog agents with latent variable models. *arXiv preprint arXiv:1902.08858*.

Tiancheng Zhao, Ran Zhao, and Maxine Eskenazi. 2017. Learning discourse-level diversity for neural dialog models using conditional variational autoencoders. In *ACL*.

Li Zhou, Jianfeng Gao, Di Li, and Heung-Yeung Shum. 2018. The design and implementation of xiaoice, an empathetic social chatbot. *arXiv preprint arXiv:1812.08989*.

Xiangyang Zhou, Daxiang Dong, Hua Wu, Shiqi Zhao, Dianhai Yu, Hao Tian, Xuan Liu, and Rui Yan. 2016. Multi-view response selection for human-computer conversation. In *EMNLP*.