2022 Spring NTU Applied Deep Learning Final Project
Transition from Open-Domain to Task-Related Topics
Team 4

Cheng-Zhong, Wang
NTU AI, R10922A12
r10922a12@ntu.edu.tw

Hao-Lun, Lin
NTU GINM, R10944020
r10944020@ntu.edu.tw

Yu-Hao, Zhou
NTU CSIE, R10922123
r10922123@ntu.edu.tw

Shi-Zhe, Lin
NTU CSIE, R10922121
r10922121@ntu.edu.tw

Abstract

In this final project, we would like to develop an interactive chatbot that can participate in chat smoothly and at the same time convert the topic of the conversation into six predefined topics. For each topic, we will define the corresponding keywords belonging to the specific topic. If the model outputs any keywords in this predefined topic within 5 turns, we will consider that the dialogue has successfully converted the subject. In order to avoid the un-smoothness of the conversation, we also check the smoothness of the chat through human evaluation. We tried many different models, datasets and distinct preprocessing methods. Eventually, we proposed a version that was fine-tuned from facebook/blenderbot-400M-distill (Roller et al., 2020), reaching 90.8% hit rate. The performance balances the smoothness of conversation and hit rate.

Keywords: Multi-turn Dialogue System, Topic Transition, Task-Oriented Dialogue

1 Introduction

In recent years, several challenging areas in natural language processing research have emerged, including open-domain dialogue (ODD), task-oriented dialogue (TOD), and so on.

In ODD, people are trying to develop a chatbot that can chat with users about anything. Not only is this type of dialogue an attractive research topic, but it can also lead to many interesting applications, such as applications that can further enhance human-computer interaction, improve foreign language learning, and create relevant interactive movie and video game characters. In the fusion of ODD and sequence to sequence modeling paradigm (Sutskever et al., 2014), it focuses on learning open-domain human conversation based on the massive (context, response) pairs (Vinyals and Le, 2015; Li et al., 2015). However, the existing open-domain chatbots suffer from a serious flaw: the content they reply to is often not meaningful. They often respond with disjointed words, or an apparent lack of common sense and basic understanding of the world. Therefore, here comes the topic-oriented dialogue systems.

In TOD, it targets specific application fields, such as restaurant reservations, bus route inquiry, equipment control, etc., aiming at completing a specific field task, representing automatic customer service, recommendation assistant, etc.

For now, the goal of our final project can be seen as a TOD task. We aim to guide the conversation so that the certain keywords can be hit during the 5 turns of dialogue and transit the topic successfully at the same time. Specifically, we have six predefined topics: “restaurant”, “hotel”, “movie”, “song”, “transportation”, and “attraction”. For each topic, the corresponding keywords to a topic have been defined in advance by TAs. If the model can be capable of outputting any keyword in this predefined topic within 5 turns, the dialogue will be considered to have successfully transit the topic. In order to avoid the un-smoothness of the conversation and aggressive talk, we also examine the smoothness of the chat through human evaluation.

In this project, we have tried many different models, datasets and distinct preprocessing methods, eventually, we proposed a version that was fine-tuned from facebook/blenderbot-400M-distill and generated quite fluent dialogues, reaching 90.8% hit rate. The performance balances the smoothness of the conversations and hit rate at the same time. In the following sections, we will describe the dataset, data preprocessing, and the model we used in detail. The experimental results will be disclosed as well.

2 Approach

Basically, we rely on the success of other pioneers to help reach the goals of the final project. We have tried some pre-trained language models from Facebook AI Research and Google Brain to fine-tune this difficult task. The models include t5-small, t5-medium, t5-large, facebook/blenderbot-400M-distill, facebook/blenderbot-3B. After the fine-tuning process and making the bot interact with the simulator to generate the conversations, we examined the dialogue smoothness via human evaluations and found that the models based on t5 (Raffel et al., 2019) usually perform consistently terrible. It may have an aggressive talk or just re-ask the same question and repeat the same sentences to people. Even if the response of the chatbot is according to the topic of the conversation, the sentences generated are still weird. Let's take a look at the testing data id-966 (dialog id: 966) in Table 1.

Based on the conversation in Table 1, we can observe that the bot always repeats sentences, which seems to be a deadlock. Therefore, we will conduct the fine-tune experiment with facebook/blenderbot-400M-distill and facebook/blenderbot-3B. However, due to the time and computing resources limit, we conduct the experiment mainly on fine-tuning facebook/blenderbot-400M-distill.

For the modeling method, we use the "T5forConditionalGeneration" model, which includes the language modeling head on top of the decoder. And the pretrained weights from facebook blenderbots were loaded via model_name_or_path with HuggingFace[1] (Wolf et al., 2019).

3 Experiment / Evaluation

3.1 Datasets

We next discuss the training and validation data we use, which are all in English.

[1]https://huggingface.co

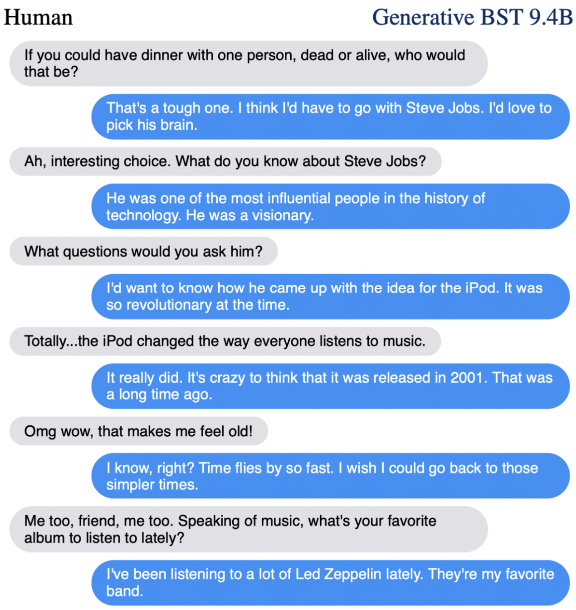

Figure 1: BST Dialogue

3.1.1 Blended Skill Talk[2] (BST)

We select tasks (Smith et al., 2020) that includes a small crowdsourced dataset of about 5k conversations in English where workers are instructed to try and be knowledgeable, empathetic, or give personal details about their given persona. Please refer to the Figure 1.

3.1.2 Simulator Generation (SG)

Simply generate simulated dialogues from the simulator script provided by TAs and use the blenderbot pretrained model introduced in Section 3 to the user simulator and our bot. It's worth noting that we have tried two chatbots, facebook/blenderbot-400M-distill and facebook/blenderbot-3B, to generate the simulator datasets. We abbreviate the conversations generated by these two models as SG-400M and SG-3B respectively.

3.1.3 One-turn Topic Transitions for Open Domain Dialogue[3] (OTTers)

(Sevegnani et al., 2021) crowdsourced the data collection for OTTers on Amazon Mechanical Turk (AMT). Each user was provided with two topics A, B from the PersonaChat persona traits, along with instructions explaining the task. The instructions ask the user to imagine they are having a conversation where the first topic A from the pair represents

[2]https://huggingface.co/datasets/blended_skill_talk

[3]https://github.com/karinseve/OTTers

Role	Sentence Generated
Simulator	That's great! I took a beauty class in high school and it was a lot of fun.
Bot	I'm glad you did. I'm glad you did it. I'm glad
Simulator	Thank you, I am glad I did too. What do you like to do for fun?
Bot	I like to play video games and watch movies. I also like to read. What about you
Simulator	I love to read as well. What is your favorite book? Mine is 1984.
Bot	I like to read as well. I also like to read. What kind of books do you
Simulator	I like science fiction and mystery novels. Do you have a favorite genre?
Bot	I like science fiction as well. I also like science fiction. What about you?
Simulator	Science fiction is a great genre. I like a lot of fantasy and sci-fi.
Bot	I like science fiction as well. I also like sci-fi. What about you?

Table 1: Simulator-Bot Interactions

the last turn of the other person, and the second topic B contains the final topic the user wants to talk about. The user then has to write a short utterance to transition to the new topic B in the least abrupt way possible.

3.1.4 Schema-Guided Dialogue[4] (SGD)

The Schema-Guided Dialogue (SGD) (Rastogi et al., 2020) dataset consists of over 20k annotated multi-domain, task-oriented conversations between a human and a virtual assistant. These conversations involve interactions with services and APIs spanning 20 domains. For most of these domains, the dataset contains multiple different APIs, many of which have overlapping functionalities but different interfaces, which reflects common real-world scenarios. Nevertheless, we didn't utilize this dataset for further training process since dialogues of dataset have too many turns, e.g. 20+ turns, that it cannot match the conversation situation in our final project, i.e. 5 turns. In other words, if we use this dataset for training purposes, it may not transit the topic successfully in only 5 turns.

The statistics result of these datasets are summarized in Table 2

Dataset	Train	Validation
BST	4819	1009
SG-400M	4819	1009
SG-3B	4819	1009
SGD	16142	

Table 2: Number of Dialogues

3.2 Data Preprocessing

Basically, what we did is to extract the conversations that include the keywords provided by TAs to generate our source-target sentence pairs.

At first, we tried to parse data like what we did in warm-up homework, give the first and the third dialogue sentence, and try to generate the middle transfer-oriented sentence. Since both the input sentence and the target sentence are one sentence, we call the above parsing method 1_to_1. But after running the fine-tune process, we found that the performance was not as good as we expected. The chatbot seems to aggressively change the conversation to the topic of restaurants, which is not a normal dialogue in chitchat.

After a deeper survey into the model training document, we realized that the simulator model provided by TAs uses at most three latest input sentences to respond to the conversion (Let's abbreviate the method as 3_to_1). Therefore, we convert our data format as that. By concatenating the three sentences with the end of sentence & start of sentence token, and mapping the 4-th sentence as the target sentence, we found that this 3_to_1 method for data preprocessing may lead to the better performance. Moreover, the 3_to_1 method is generally better than 1_to_1, 2_to_1 and 5_to_1 source-target pairs. The pre-processed data are visualized in Figure 2. Detailed experimental result will be shown in the following section.

3.3 Experimental Results

In this section, we are going to show the experimental results with fine-tuning the facebook/blenderbot-

[4] https://github.com/google-research-datasets/dstc8-schema-guided-dialogue

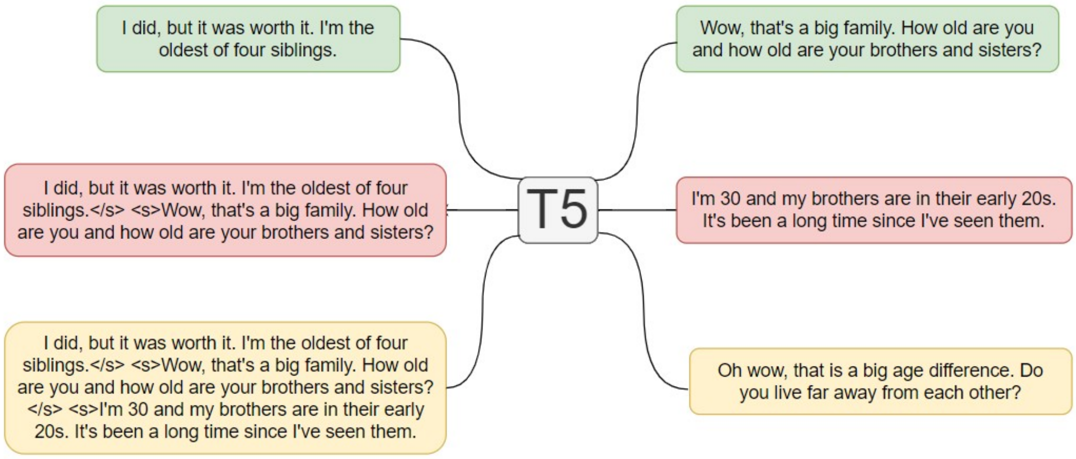

Figure 2: Pre-processed Data: Input (left) & Output (right)

Batch Size	Epoch	Max Length	Hit Rate
32	5	128	0.716
32	5	256	0.914
32	5	384	0.932
32	5	512	0.744

Table 3

Batch Size	Epoch	Max Length	Hit Rate
16	5	128	0.729
16	5	256	0.808
32	5	128	0.846
32	5	256	0.773
32	5	384	0.889

Table 4

400M-distill model on different datasets.

3.3.1

In this part, please refer to the Table 3. The dataset we used is BST, as we mentioned in 3.1.1. We only take the train split for training and all hyperparameters are tuned via validation split. We can find that the hit rate on this dataset is up to and over 90% which is great; nevertheless, the conversation generated is not smooth.

3.3.2

In this part, please refer to the Table 4. The dataset we used is SG-400M and the source-target pairs are 3_to_1, as we mentioned in 3.1.2 and 3.2 respectively. We only take the train split for training and all hyperparameters, including batch size and max sequence length, are tuned via validation split. Compared to the results in 3.3.1, we conclude that using the conversation generated from the simulator results in better smoothness without compromising the hit rates.

3.3.3

In this part, please refer to the Table 5. The dataset we used is SG-3B and the source-target pairs are 3_to_1, as we mentioned in 3.1.2 and 3.2 respectively. We only take the train split for training and all hyperparameters, including epochs and max sequence length, are tuned via validation split. We examined the conversation by human evaluation and compared to the conversations in 3.3.2, the model trained on SG-400M has a slightly better performance than that trained on SG-3B. Moreover, the hit rate does have performance at a considerable level.

3.3.4

In this part, please refer to the Table 6. The dataset we used is SG-400M and the source-target pairs are 1_to_1, as we mentioned in 3.1.2 and 3.2 respectively. We only take the train split for training and all hyperparameters, including epochs and max sequence length, are tuned via validation split. Even though the hit rate is great, we still examined the conversation by human evaluation and realized that using 1_to_1 is impractical since the conversation is freaking terrible.f The model fails to capture the causality to generate reasonable sentences.

Batch Size	Epoch	Max Length	Hit Rate
32	3	128	0.877
32	3	256	0.866
32	3	384	0.809
32	3	512	0.855
32	5	128	0.837
32	5	256	0.863
32	5	384	0.846
32	5	512	0.853
32	10	128	0.860
32	10	256	0.863
32	10	384	0.831
32	10	512	0.849

Table 5

Batch Size	Epoch	Max Length	Hit Rate
64	3	384	0.902
64	5	384	0.869
64	7	384	0.851

Table 6

3.3.5

In this part, please refer to the Table 7. The dataset we used is SG-400M and the source-target pairs are 3_to_1, as we mentioned in 3.1.2 and 3.2 respectively. The main difference here is that we not only take the train split for training but also validation split. We examined the conversation by human evaluation and compared to the conversations in 3.3.2, the model trained on both train and validation split from SG-400M has the best performance compared to all other models and datasets we've tried. Moreover, the achievement of hit rate and conversation smoothness does have a great balance. To be specific, the model trained with max sequence length 512 and for 10 epochs makes the hit rate up to 85.9% and has the most reasonable and fluent dialogue.

3.3.6

In this part, please refer to the Table 8. The dataset we used is SG-400M and the source-target pairs are 3_to_3, as we mentioned in 3.1.2 and 3.2 respectively. We only take the train split for training and all hyperparameters are tuned via validation split. We found that the training with max length 384 leads to great performance on hit rate.

Batch Size	Epoch	Max Length	Hit Rate
32	5	128	0.828
32	5	256	0.850
32	5	384	0.851
32	5	512	0.807
32	10	128	0.845
32	10	256	0.860
32	10	384	0.827
32	10	512	0.859

Table 7

Batch Size	Epoch	Max Length	Hit Rate
32	5	128	0.835
32	5	256	0.828
32	5	384	0.904
32	5	512	0.843

Table 8

3.3.7

In this part, please refer to the Table 9. The dataset we used is SG-400M and the source-target pairs are 2_to_1, 3_to_1, and 5_to_1. We only use the train for training and all hyperparameters are tuned via validation split. We found that the 5_to_1 parsing strategy turned out to be worse than 2_to_1. One possible reason is that the 5_to_1 strategy may deviate from the original semantics. What's more, the hit rate of the 3_to_1 strategy is slightly lower than the 2_to_1, but the smoothness outperformed others via human evaluation.

Based on the tables above, we are going to share our opinion of different pairing strategies. First, for the 1_to_1, it may not capture the casualty from the input sentences. Second, for 5_to_1, too much information in the input sentences may lead to confusion for the model. Last but not least, for 2_to_1 and 3_to_1, we expected both of them should have competitive results; nevertheless, under the similar hit rate, we found that 3_to_1 does have the better smoothness from human examination. That's the reason why we choose 3_to_1 as our final parsing strategy.

To sum up, the 3_to_1 parsing strategy is processed on SG-400M dataset and both the training and validation splits are used for fine-tuning the model. The max sequence length is set to 128. We take the model after training with 9 epochs as our final submission. The hit rate down below is

Batch Size	Parsing Strategy	Epoch	Max Length	Hit Rate
16	2_to_1	5	256	0.811
16	3_to_1	5	256	0.808
16	5_to_1	5	256	0.686

Table 9

Batch Size	Epoch	Max Length	Hit Rate
32	5	128	0.883
32	6	128	0.869
32	7	128	0.887
32	8	128	0.882
32	9	128	**0.908**
32	10	128	0.897

Table 10

calculated from the latest version of simulator formulated by TAs, i.e., at most 11 utterances. The final experimental result is listed in Table 10.

4 Conclusion

In this final project, we have proposed our methods to solve the topic transition task. The experimental details and results are as mentioned in previous sections. We conclude that using the interaction sentences with the simulator as our training sets and making the proper source-target pairs (i.e. 3_to_1) can definitely achieve the acceptable hit rate (i.e. 90.8%) and smooth conversation at the same time via fine-tuning the language model pre-trained from Facebook AI Research. After the human examination, the output sentence we generated does have a great performance. In our future work, we may try some different decoding strategies to improve the exploration of the model, not just using the greedy strategy for exploitation.

5 Work Distribution

All members participated in dataset selection, training tasks and reporting, and Table 11 lists the additional work.

ID	Work
R10922A12	Data Analysis
R10944020	Hyper-parameter Tuning
R10922123	Data Analysis
R10922121	Hyper-parameter Tuning

Table 11

References

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*.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2019. Exploring the limits of transfer learning with a unified text-to-text transformer. *arXiv preprint arXiv:1910.10683*.

Abhinav Rastogi, Xiaoxue Zang, Srinivas Sunkara, Raghav Gupta, and Pranav Khaitan. 2020. Towards scalable multi-domain conversational agents: The schema-guided dialogue dataset. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 8689–8696.

Stephen Roller, Emily Dinan, Naman Goyal, Da Ju, Mary Williamson, Yinhan Liu, Jing Xu, Myle Ott, Kurt Shuster, Eric M Smith, et al. 2020. Recipes for building an open-domain chatbot. *arXiv preprint arXiv:2004.13637*.

Karin Sevegnani, David M Howcroft, Ioannis Konstas, and Verena Rieser. 2021. One-turn topic transitions for open-domain dialogue.

Eric Michael Smith, Mary Williamson, Kurt Shuster, Jason Weston, and Y-Lan Boureau. 2020. Can you put it all together: Evaluating conversational agents' ability to blend skills. *arXiv preprint arXiv:2004.08449*.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. *Advances in neural information processing systems*, 27.

Oriol Vinyals and Quoc Le. 2015. A neural conversational model. *arXiv preprint arXiv:1506.05869*.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. 2019. Huggingface's transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771*.