

Improving the Empathetic Chatbot

Jiageng Zheng

1. Introduction

Emotional-Support-Conversation [Liu2021] presents a neural network-based automated chatbot that can interact with human users and make empathetic conversation, but it sometimes gives off-topic responses. These responses can be provocative and have negative effects. To use this chatbot for emotional support, the model needs to be improved.

In this report, we first analyze the chatbot's mistakes, pinpoint the problems and then propose modifications. Subsequently, we set up experiments to evaluate our modifications. Finally, we analyze the limitations of our methods. Last but not least, we propose some enhancements that can handle such limitations for future work.

2. Failures Analysis of the Original Model

Here, we show some examples to identify cases where the responses generated by the original model are problematic.

Example 1

For the following 5 sets of conversations, we (humans) say "I am super mad." The bot will ask for a reason, and we say "*my roommate ate my breakfast.*"

A new conversation starts!

Human: I am super mad

AI: why are you mad? it is ok to be upset

Human: my roommate ate my breakfast

AI: it is just so hard for me to feel upset about it.

A new conversation starts!

Human: I am super mad

AI: why are you mad? why is it that you're mad?

Human: my roommate ate my breakfast

AI: i am mad because i didn't do my homework.

A new conversation starts!

Human: I am super mad

AI: why? what happened? why are you mad?

Human: my roommate ate my breakfast

AI: i think i ate too much... i think it was a mistake.

A new conversation starts!
Human: I am super mad
AI: what happened? what is happening? i hope it's not bad.
Human: my roommate ate my breakfast
AI: oh no, that's no good. i have a roommate too. we are in a relationship so we have a lot of things in common but still don't know what to do.

A new conversation starts!
Human: I am super mad
AI: i'm sorry to hear that. what did you do to upset them?
Human: my roommate ate my breakfast
AI: i told them to get the hell out of my room. i had to go to work.

Analysis 1

We can see the chatbot gives better responses for its first utterance. To reply “I am super mad”, the chatbot uses the strategy “Question” [Liu2021] for all five samples. The replies of the first 4 samples are decent and supportive. The last one makes an assumption that is not smart. To reply “my roommate ate my breakfast,” only the 4th sample gives a supportive reply. The 2nd, 3rd, and 5th are off-topic, while the 1st gives a reply but is not supportive at all.

Observed that in the 1st sample, the reply “it is just so hard for me to feel upset about it.” is related to the “upset” from the context “it is ok to be upset.” In the 2nd sample, the reply “i am mad because i didn't do my homework.” matches the word “mad” in “I am super mad.” In these cases, the dialog histories seem to be noise rather than helpful.

To make a comparison, we decided to test the model's performance when saying “my roommate ate my breakfast” without any prior context.

Example 2

We say “my roommate ate my breakfast” and then the chatbot gives a reply.

A new conversation starts!
Human: my roommate ate my breakfast
AI: wow, did he eat it? or did you say something?

A new conversation starts!
Human: my roommate ate my breakfast
AI: that's awful. why did he do that?

A new conversation starts!

Human: my roommate ate my breakfast
AI: i hate when that happens. i would've been mad too

A new conversation starts!
Human: my roommate ate my breakfast
AI: what did they eat? i bet you were upset.

A new conversation starts!
Human: my roommate ate my breakfast
AI: did you tell him? did he say anything?

Analysis 2

All 5 replies from this chatbot are decent. Associated with previous analysis, it suggests the performance of this chatbot can be improved if we erase the dialog history occasionally. To make improvements, we need to define what is the correct time to clear the dialog history.

Example 3

This example demonstrates the importance of erasing the dialog history. Starting with the third sentence, this chatbot gives off-topic responses which are related to previous dialog history.

A new conversation starts!
Human: what is your favorite color?
AI: i think it is black. i love the color of the sky and the way it looks on the outside of the plane.
Human: what is your favorite animal?
AI: my favorite animal is a wolf. i like the gray wolf and other wolves in the family.
Human: what is your favorite food?
AI: i like the gray wolf too. i really like meat. i like to eat out a lot.
Human: what is your favorite drink?
AI: i'm not sure... i love all animals.. but i love dogs and cats.. especially dogs in the wild.. they are so cool. i've had a couple of them. i like the gray wolf too

3. Understanding the Parameter Settings in the Original Model

To propose modifications, we first need to understand the parameter settings in the original model. We found the interpretation of parameters and did experiments with different settings. Here we explain some important ones.

Min_length: “The minimum length of the sequence to be generated.”

If it is set to small values (e.g. 1), the bot may generate very short replies which do not sound professional for the emotional support. In the following experiments it is set to 10 (default).

Temperature: “The value used to module the next token probabilities.”

This parameter controls the variety of samples in the generative algorithm. The value 1.0 eliminates randomness, and as a result, the sample will only contain one utterance. Intuitively, the algorithm selects the best utterance from the sample, so having only one utterance is not ideal. For the following experiment, we set this value to 0.7.

Top_k: “The number of highest probability vocabulary tokens to keep for top-k-filtering.”

Experiments show that when Top_k is set to 1, there will be no randomness for sentence generation, and the quality is low (same as we observed when setting temperature to 1.0). We set Top_k to 100 and to 10000, they are both better than the 1, and their quality are about the same. After we set the same random seed, the 100 and 10000 give the same result. This indicates that the amount of “vocabulary token” is less than 100 so the top-k-filtering does not have any effect on the generative method. For the following experiment, we disabled top-k-filtering by setting it to a large number (e.g. 10000).

Top_p: “If set to float < 1, only the most probable tokens with probabilities that add up to top_p”

We disable p filtering by setting this value to 1.

Num_beams: “Number of beams for beam search. 1 means no beam search.”

We found more beams in this project slowed down the sentence generation and produced low quality results. In all following experiments, we disable beams by setting this value to 1.

4. Modification Methodology

In the conversation settings, humans can recall what they said 5 minutes ago by only about 10 percent (Stafford, 1984). Intuitively, the dialog history a long time ago has little effects for the next utterance. Experiments and analysis on our chatbot have shown that irrelevant dialog histories can disturb its utterance generation and removing dialog histories mitigates its off-topic problem. We cannot expect users to manually erase dialog histories. In this work, we designed ways to erase dialog histories automatically.

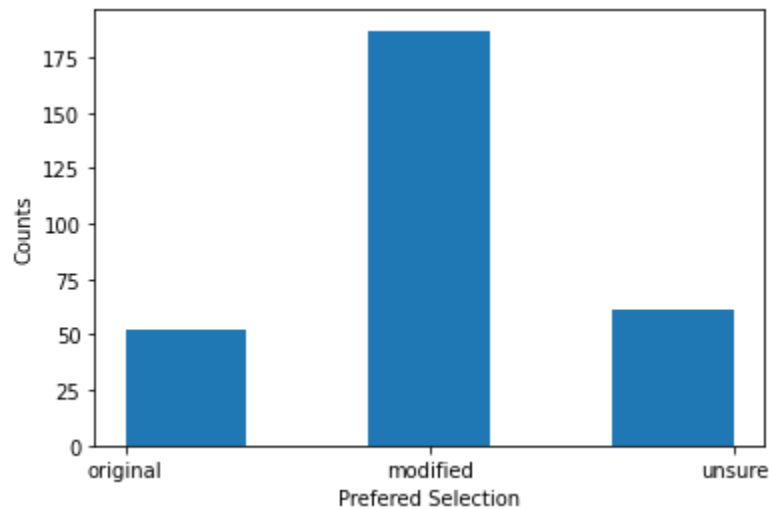
5. Enhancement 1: Remove AI Utterance from Dialog History

For this enhancement, we removed all the AI utterances from the dialog history. We set up an experiment to compare this new version with the original chatbot.

Experiment

We manually generated 100 dialog histories. For each one, we came up with the first human utterance, collected an AI response, and came up with the second human utterance. We generate the second AI utterance using 2 different settings. For the first (**original**) setting, the dialog histories passed to the generative model contain all three utterances. For the second (**modified**) setting, we remove the AI utterance, so the dialog histories contain the first and second human utterances. We compare the quality of AI's last utterances between two different settings.

We have 5 researchers to evaluate this modification, each dialog is evaluated by three people where each person will choose responses from the original model or the enhanced model is better or they are not sure., Thus, a total of 300 choices are made. The statistics of the 300 choices are as follows:.



By human evaluation, 187 times (out of 300) evaluators choose the responses from the modified chatbot to be better than those from the original version. The result shows that the modification as simple as erasing all AI utterances can help the AI give better responses.

Analysis

The improved performance comes from the removal of noises from the dialog history (see first example in section 2, Failure Analysis). Moreover, this modification seldom harms the performance. Since this chatbot provides emotional support, it is better to “listen” to human users and give empathic responses to human user’s words. It is better to respond to human utterances instead of clinging to the words that come from the AI itself.

This modification improved performance for short dialogs but not for longer conversations. Even after we removed AI utterances, the human utterances can still affect sentence generation negatively (see the third example in section 2, Failure Analysis). Therefore, in the next section, we introduce a way to erase dialog histories for longer conversations.

6. Enhancement 2: Erase Dialog History by Topic Shift

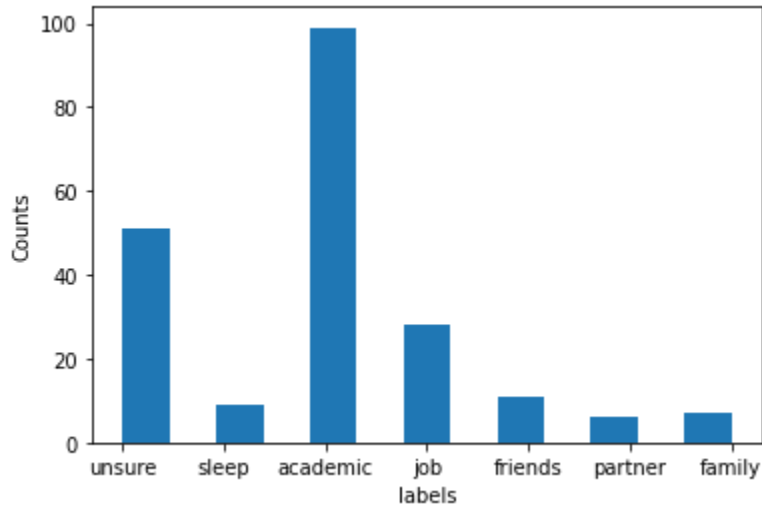
We used labels and dialogs in the ESCov dataset to fine tune a BERT classifier which can produce class labels for incoming utterances. The classifier assigns probabilities for 7 labels: ongoing depression, academic pressure, job crisis, sleep problem, break up with partner, problem with friend, and family issue. We add another label, named ‘unsure,’ which represents none of 7 labels assigned with a probability larger than 80%.

We use this classifier in the chatbot to assist the clearance of dialog histories. Whenever a human utterance is received, the classifier gives this utterance a label. If the current label is “unsure,” we assume

there is no topic shift, so we change the “unsure” to the label of the previous utterance. If the current label is different from the previous one (and both are not “unsure”), a topic shift is detected, and all previous dialog histories will be removed.

Experiment

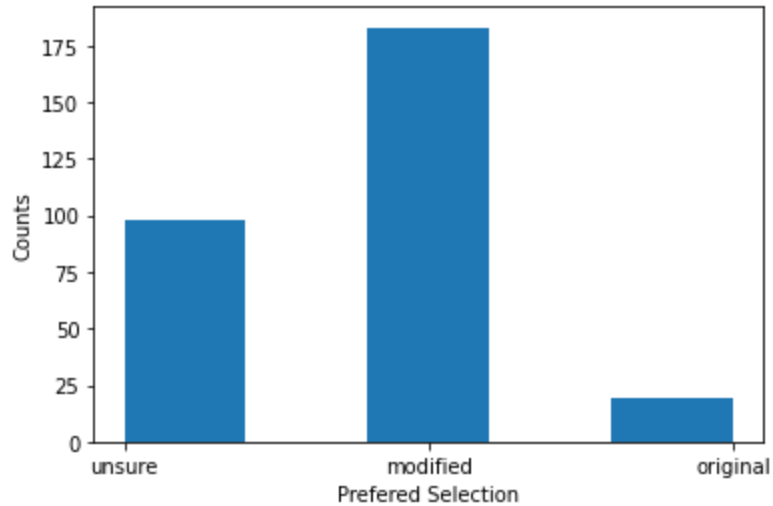
We first assign labels to the human utterances we used in the experiment in Section 5. The statistics are as follows:



Having all the human utterances classified, we generated new dialogs by randomly selecting these sentences as the first utterances, generated an AI response, and randomly selected these sentences as the second human utterance. Then, we selected the dialogs in which topic shifts can be detected. We manually removed dialogs that do not make sense (it happens when the first or the third utterance does not give sufficient information, e.g., by using pronouns). Now we have produced 100 dialog histories, each with three utterances.

For the first (**original**) setting, the dialog histories passed to the generative model contain all three utterances. For the second (**modified**) setting, we only kept the last human utterance since topic shift was detected. We generated the second AI utterance using the two settings and compared their quality.

We have 5 researchers to evaluate this modification (one of our previous researchers is away, so Jiageng also works as one of them). Each dialog is evaluated by three people, so 300 choices are made. The statistics of the 300 choices are as follows:.



By human evaluation, 183 times (out of 300), the evaluators chose the responses generated from the modified chatbot to be better than those generated from the **original** version. It is worth noting that the **original** version is only selected 19 times, compared to the previous experiment where they are selected 61 times

Analysis

The **modified** version is preferred most of the time, so the topic shift method has improved this chatbot's performance. Compared to the experiment in Section 5, the responses generated by the **original model** are less frequently chosen as better responses. This suggests the original version is more deficient when a topic shift exists in the dialogue.

7. Summary

As the conversation length increases, we observe a decrease in the generative model's performance. We are able to avoid this problem by occasionally emptying AI's dialog histories. After modifications, the chatbot "forgets" all the utterances it has said and all the dialog histories as a topic shift is detected.

Limitations

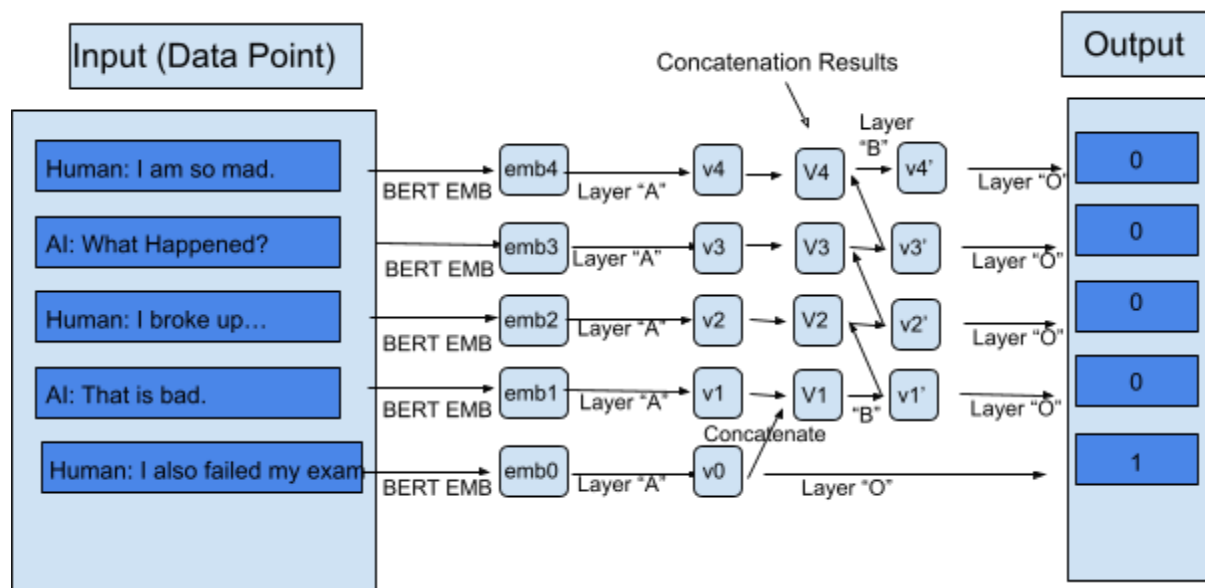
Our enhancements still have limitations. First, the classifier sometimes gives incorrect labels, which produce false positives and negatives of dialogue history clearance. Second, our 7 class labels have not covered all possible classes, so if a user complains about Covid, its label is likely to be 'unsure.' Third, this method does not capture topic shifts within each class label. For example, between the discussion of homework and of exams, we may need to erase dialog history, but both of them are labeled as "academic pressure," resulting in no topic shift detected.

Future Works

Instead of detecting topic shifts, it is better to train a model to directly determine the position for dialogue history clearance. To do this, a dataset should be created, where one datapoint is one dialog history, and

each utterance in the dialog should be given a binary label. The most current utterance will be labeled as 1, meaning this will be kept. For all previous sentences, if it should be kept, label it with 1, else, label it with 0. The topic shift method in section 6 can facilitate the production of this dataset, but additional manual annotation is inevitable.

The neural network for this work has to have recursive architecture. For example, we can use BERT to give each utterance a vector embedding. Currently such utterance embedding is passed through a linear layer “A” and an output layer “O” gives a label (0, 1) that determines if an utterance will be retained for response generation of the current utterance. However, for the recursive architecture, we retrieve the previous utterance’s vector embedding, pass it through the linear layer “A,” concatenate it with the current utterance’s last hidden state, and run it through a linear layer “B” to resize it back to the same output length produced by Layer A before passing it through the output layer “O”. The output layer will give a label (0, 1) to a previous utterance. This process is repeated until we finish processing all the earlier utterances. The values for the “A”, “B”, “O”, and embeddings will be trained.



Directly predicting the position for dialogue history clearance, the method proposed in this section has the potential to solve the second and the third limitations we mentioned in Section 7 Limitations. If this work of automatic history clearance continues in the future, it is worth implementing the model we mentioned above.

Bibliography

Liu2021: Liu, etc.

Towards Emotional Support Dialog Systems , Proceedings of ACL 2021
arXiv:2106.01144

Stafford, 1984

Conversational Memory: The Effects of Recall Mode and Memory Expectancies on
Remembrances of Natural Conversations

BERT v4.19.2

https://huggingface.co/docs/transformers/model_doc/bert

Emotional-Support-Conversation (Github)

<https://github.com/thu-coai/Emotional-Support-Conversation>

Acknowledgement

Thanks to Professor Chuah for creating dialogs for experiments.

Thanks to Zhihao Zheng, Zhen Yao, Xiaowen Ying, Bo Lang, who helped our experiment.