Automatic Generation of Rhetorical Questions and Its Application to a Chatbot

Presenter: Management Science and Engineering Program, Department of Informatics
1830126 Qifan Sun
Supervisor: Prof. Akira Utsumi, and Asst. Prof. Suguru Matsuyoshi

1 Introduction

In recent years, an interpersonal attraction in the conversation has been studied extensively. Creating figurative language generation modules can make chatbots more human-like. Some recent studies have proposed chatbots that generate sarcasm[1]. However, they do not focus on generating rhetorical questions (RQ). It is necessary for chatbots to generate RQs to be more human-like because RQs are usually used in daily conversation and social media dialog[4]. RQs are questions but not meant to obtain an answer. People usually use them to express their opinions in conversation. However, a question cannot be recognized as an RQ if the answer of the question is only known by the speaker. To recognize that it is an RQ, the listener needs to use the knowledge shared between them. Furthermore, there is a specific interrelation between irony and RQs. Therefore RQs are always used to express their negative opinions. Questions based on the valence-reversed commonsense knowledge can be easily recognized as RQs because both speaker and listener know their answers are negative. For example, the commonsense knowledge “Giving money to the poor will make good world” can be converted into an RQ: “Will giving money to the rich make a good world?”

This study aims to generate a negative-answering RQ by using valence-reversed commonsense knowledge sentences to make the chatbot more appropriate and human-like in a conversation. Additionally, we use a situation classifier analyzing previous contexts to decide when to generate a literal response, sarcastic response, and RQ.

2 Method

2.1 Chatbot

The chatbot in this study works as shown in Figure 1. It uses the user’s utterances and previous utterances to generate appropriate responses. The chatbot is processed by the following three steps.

In Step A, the situation classifier selects an appropriate response type from among sarcasm, RQ, and literal responses. The situation classifier is introduced in Section 2.2.

In Step B, the literal response generator generates a literal response by using preceding utterances. Zhang et al.[5] proposed a neural conversational response generation model that produces consistent responses. This model is trained on about 147M posts from Reddit comments. We use this model after fine-tuning to generate literal responses. Generated literal responses are directly used for a chatbot to respond (when the situation classifier judged that a literal response is most appropriate) or as an input to the RQ and sarcasm generator.

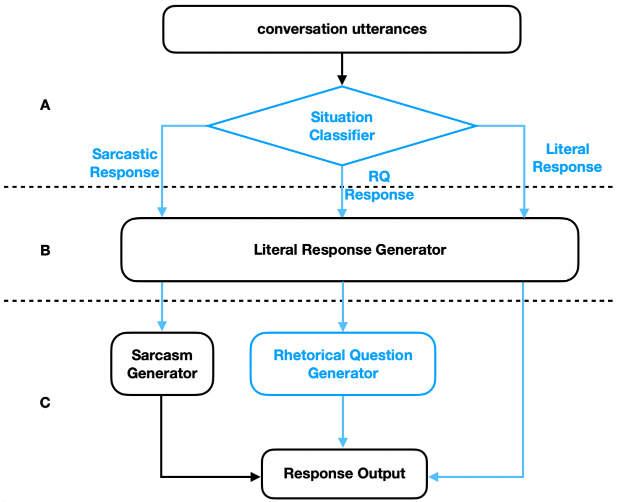

Figure 1: The flowchart of the chatbot: The components originally proposed in this study are indicated in blue.

In Step C, when the situation classifier decides to generate an RQ response, an RQ generator converts the literal response into an RQ response, which is explained in Section 2.3. When the classifier decides to generate a sarcasm response, a sarcasm generator converts the literal utterance into a sarcasm response. We use the sarcasm generator proposed by Chakrabarty et al.[1]. It concatenates valence-reversed literal utterances with sentences. These sentences are retrieved from an online sentence dictionary corpus, which contains high-quality sentences using commonsense phrases.

2.2 Situation classifier

In this study, we fine-tune four pre-trained models, i.e., BERT-Base, BERT-Large, RoBERTa-Base, and RoBERTa-Large, to classify the current situation according to which of RQ, sarcastic and literal responses are appropriate. Two turns of utterances before a response to be generated are encoded by the pre-trained model and then input into our classification model. Our model comprises BERT/RoBERTa, Flatten, Dropout, and Dense layer as a base-line. We also use bidirectional LSTM layers to get a better result. A classification result is determined from the softmax activation of the last layer.

Table 1: Scores of the chatbot in a different response situation. The leftmost column indicates the type of response (i.e., literal, sarcasm, or RQ) selected by the situation classifier. The rows indicate that the mean evaluation scores for four responses in each situation.

(a) Appropriateness

Type	Human	Literal	Sarcasm	RQ
Literal	3.63	3.42	3.15	3.13
Sarcasm	3.98	3.46	3.42	3.40
RQ	3.52	3.39	3.40	**3.47**

(b) Human-likeness

Type	Human	Literal	Sarcasm	RQ
Literal	3.70	3.53	3.28	3.45
Sarcasm	4.15	3.62	3.66	3.37
RQ	3.58	3.56	3.43	3.39

2.3 RQ generator

We propose an RQ generator that converts a literal response to an RQ using commonsense knowledge. The generator reverses the valence of commonsense knowledge that is relevant to the literal response and converts it into an interrogative sentence. The generator analyzes the sentence structure of the literal response and the sentiment polarity of all words.

When a verb has the highest sentiment score, the generator reverses the valence of it. The generator extracts keywords from the original literal response such as the verbs, nouns, and adjectives, and use the commonsense knowledge scoring model[3] to calculate a commonsense score. When the score is higher than 0.5, the valence-reversed literal response is converted to wh-question. When the score is lower than 0.5, the original literal response is output directly.

When the highest sentiment word is an object, a subject, or no sentiment phrase, the generator extracts keywords from the literal response and searches on the commonsense knowledge dictionary using the keywords to obtain commonsense knowledge sentences. Next, the generator selects the top ten commonsense knowledge sentences topically similar to the literal responses. Then, the generator reverses their meanings and converts them to yes-no questions. After that, it concatenates them with the original response. Finally, the generator selects the most appropriate RQ response using a fine-tuned BERT model that computes candidate responses' appropriateness.

2.4 Evaluation on Chatbots

We randomly selected 42 posts from the test data of FigLang 2020 dataset[2] and generated all the types of responses (Literal, sarcasm, RQ and original human responses). We designed a task on Amazon Mechanical Turk. Each post was rated by 5 participants. They were asked to read the previous contexts and answer the following questions:

- Are these responses rhetorical questions? (Yes/no)
- Are these responses sarcastic? (Yes/no)
- How appropriate is the response in the conversation? (5-point scale)
- How human-like is the response? (5-point scale)

Table 1 shows that when the situation classifier chooses RQ, RQ responses achieve a higher score on appropriateness than literal response and sarcasm. The RQ generator keeps the original opinion by using literal responses. The additional RQ makes the response more appropriate in the RQ situation. On the other hand, the RQ generator converts question based on the algorithm that changes the structures of the sentence. Sometimes it makes grammatical errors that cannot be fixed by the grammatical error correction model. Consequently, the RQ responses are rated as less human-like than other responses.

When the situation classifier selects a literal response, both appropriateness and human-likeness scores are higher than other responses, except for the human response.

When the situation classifier decides to respond a sarcasm, the appropriateness scores are lower than the literal responses. However, it has a higher score in human-likeness. The semantic incongruity ranking selects the most sarcastic response, which makes the response more human-like.

3 Conclusion and future work

It is concluded that the proposed RQ generator and situation classifier is effective in that RQs are more appropriate than literal and sarcastic responses when the classifier decides to generate RQ responses. However, the generated RQ responses are less human-like than other types of responses. In the future, we would like to develop a model for converting sentences to questions that enables the chatbot to generate a higher quality of RQs and to be more human-like.

References

[1] Tuhin Chakrabarty, Debanjan Ghosh, Smaranda Muresan, and Nanyun Peng. Rˆ3: Reverse, retrieve, and rank for sarcasm generation with commonsense knowledge. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7976–7986, 2020.

[2] Debanjan Ghosh, Avijit Vajpayee, and Smaranda Muresan. A report on the 2020 sarcasm detection shared task. In *Proceedings of the Second Workshop on Figurative Language Processing*, pages 1–11, 2020.

[3] Xiang Li, Aynaz Taheri, Lifu Tu, and Kevin Gimpel. Commonsense knowledge base completion. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1445–1455, 2016.

[4] Shereen Oraby, Vrindavan Harrison, Amita Misra, Ellen Riloff, and Marilyn Walker. Are you serious?: Rhetorical questions and sarcasm in social media dialog. In *Proceedings of the 18th Annual SIGdial Meeting on Discourse and Dialogue*, pages 310–319, 2017.

[5] Yizhe Zhang, Siqi Sun, Michel Galley, Yen-Chun Chen, Chris Brockett, Xiang Gao, Jianfeng Gao, Jingjing Liu, and Bill Dolan. DIALOGPT : Large-scale generative pre-training for conversational response generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 270–278, 2020.