Gathering Naturally Occurring Utterances for Chat Bot Systems

Brian Schwarzmann and Arghyadeep Giri

Department of Computer Science

University of California, Santa Cruz

brschwar@ucsc.edu or argiri@ucsc.edu

Abstract

Chat bot systems often use sentence pairs for selecting an utterance that would naturally follow another prior utterance. We used Amazon's Mechanical Turk to generate natural responses to sentence prompts to increase the number of sentence pairs available for this task and to increase the depth and breadth of a naturally sounding conversation using a chat bot system. We experimented in both the type of prompts we provided and the types of responses we asked for. In total, the 'Turkers' under our supervision generated thousands of sentence pairs automatically labeled by the type of utterance both in the prompt and in the response. We also continued the dialog by taking the response of one Turker and using it in the next series as part of the prompt. Our conversation grew naturally in depth as we took each round of responses and used them for the next round of prompts. We ended up with five turns of responses based on our initial prompts.

1 Introduction

Over 80 percent of utterances in natural language dialog are statements. This causes issues with chat bot systems which try to speak naturally. Most of these systems were designed to answer questions and use a database of question-answer pairs to select their responses from. While this is often the best approach for chat bot systems, for instance, finding technical assistance to a computer software problem or for booking a hotel online, having a natural conversation is only questions and answers. Much of our natural language either has implied statements of the "tell me more" or "explain" types, such as "uh huh" and "really?" Still more of the conversation is through more explicit agreement or disagreement, such as when someone says they like living in an old house and the response could be anything about the larger yards in old houses to remolding of houses to being so close to each other in the newer neighborhood that you can hear everything your neighbor says. All of these are natural ways a conversation could go but are very difficult for currently systems to predict and create because they are not explicit question-answer pairs. We are so adapt at understanding each other that we have a hard time explaining to a computer what to do. We are hoping to help fill that gap by providing examples of coherent speech that various groups could use to increase coherence.

2 Amazon Mechanical Turk

Amazon Mechanical Turk is a way to outsource large numbers of small human tasks to a wide variety of people. We chose to only allow those workers living in English speaking countries who have previously been approved 95 percent of the time at least 100 times to be able to work on our HITs (Human Intelligence Tasks). We also learned that some people would choose to respond in similar ways no matter what the prompt, so we modified the HITs to eliminate that possibility. We found that it was quite natural to say, "What is your favorite X ?" about almost any topic. That might work once in a conversation but it is not a great way to keep a conversation going, which was our desire. Another thing we did was to increase the minimum number of characters a submission could have and still be accepted. This tended to produce longer and longer dialogues because the prompts got longer and longer.

2.1 Related Work

The major problems related to creating a dialogue corpus is management of workers in terms of money and time. Gathering workers and pair-

ing them synchronously for collecting data was found to be ineffective and expensive at the same time. Moreover, given a topic of conversation, it is hard to guarantee the quality of responses in terms of coherence, dialogue or any other attribute that we are seeking for in the corpus. Several studies addressed these issues and tried solving by reducing the cost by some tool. However, the existing systems have compromises either in terms of cost, time or quality of responses in the corpora (cf. [5], [4]). Higashinaka et al. constructed a corpus consisting of chat dialogues between a human participant and the system with publicly available chat API [1]. Sugiyama et al. proposed the method of utterance generation with Twitter data [7]. Considering crowdsourcing approaches, we have widely used research in the field of dialogue these days. Mitchell et al. developed a corpus of natural language generation templates by using crowdsourcing [3]. Paperno et al. constructed a data set evaluated by crowdsourcing for natural language understanding [6]. Lasecki et al. proposed a system that labels events in videos by crowd workers [2]. Prior systems have shown that multiple workers can be recruited for collaboration by having workers wait until enough workers have arrived [8,9]. Jeffrey.P.Bigham et al. developed a system enabling real-time two-way natural language conversation between an end-user and a single virtual agent powered by a distributed crowd of online humans [10]. Considering these approaches, reaching an optimal point in terms of gathering workers, cost efficiency and quality of data is clearly quite a challenging task. We have tried to reach that optimal point in a very simple yet elegant approach.

2.2 Previous Approaches

All the previous approaches using Amazon Mechanical Turk, had the same objective of collecting coherent responses to prompts and creating depth of conversations. The previous methods (listed below) however were not scalable and required more time compared to that of the current one.

a. Turkers were emailed with a poll to get their availability for a certain period, and then Turkers with similar schedules were matched up to chat together about different topics. It resulted in efficient, long, interesting conversations but took many hours to set up, workers didn't like waiting for each other, and was not scalable.

b. A particular Turker from the the previous method, and had an expert talk to them at a scheduled time. This created similar responses like the previous one, but again, it was not scalable. Also it introduced our own biases on the conversation.

c. HITs were put out that could stay alive for 3 days. Turkers were let to get matched up without doing any poll scheduling. This was an attempt at being "semi-synchronous", where they could have a conversation that spanned multiple days and not "live", but It was not efficient since Turkers got frustrated because they would have to wait to get matched and had to keep checking the chat window to see if their partner had responded.

d. A variant of the previous method where the HIT time to was limited to 1 hour to get people to join more quickly and have the conversation to make it almost live. Again, Turkers got frustrated because one side might be ready to go, but they had to wait for the other side and got nervous because they wouldn't know when or even their 'partner' was going to join, which meant they might not even get to finish the HIT.

2.3 Current Approach

The entire approach of dialogue collection can be subdivided into a few steps as follows:

a. Initial Prompts: The process of dialogue collection starts with presenting the Turkers with a very simple prompt based on a topic such as movies, TV shows, music, travel destinations etc. An example of such a prompt is "I really like Batman movies". And hence we expect Turkers to come up with responses according to a provided set of information so that they are coherent to the presented prompt. We also tried to make the responses more natural and engaging by giving Turkers instructions and explanations on how to make responses more natural and interesting. We also, provided examples of both good and bad responses for a sample prompt.

b. Selecting Categories: The initially mentioned problem of not having enough good utterance pairs to produce naturally sounding conversation systems was attempted to be solved by our technique. Our technique has the goal of making the responses to the prompts, as relevant as possible to that of the subject of the conversation. Initially, four categories were created,

namely “a response more specific”,” a response more general”, ”a response similar to what was said” and “a follow-up question”. By this we explained how the Turkers should come up with naturally sounding responses that are still on the same topic but either more specific in someway such as an actor, more general in some way such as genre, a response similar in depth, and also a question to elicit more detail about the provided prompt. To exemplify with the previous stated prompt “I really like Batman movies”, sample responses in the corresponding categories would be as follows:

Specific: *“I think THE DARK KNIGHT is the best batman movie of all time.”*

General: *“I am not a fan of any action movies.”*

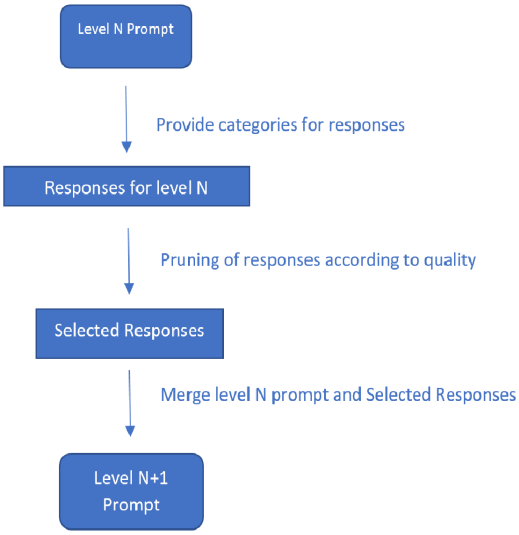
Similar: *“I like Batman movies as well and I have watched all of them.”*

Question: *“Which part of the Batman series did you like the most?”*

We found that some type of response requests were not as effective as others in producing good quality, diverse responses. Specifically, the request to be more specific through a question produced a large number of similar responses, i.e. ”What’s your favorite X?” This response was common even after it was mentioned as a ”bad example” that should not be submitted. After we were able to receive diverse, interesting and relevant responses on the prompts provided, we expanded in two ways. We expanded the topics discussed and started looking for depth of conversation on both the original topics and the new topics. For example, for a prompt “I love walking on the beach” we chose categories “elaborate on what was said”, “change topic slightly from the current topic”, “continue the topic”, “tell a joke or make a sarcastic remark”. These new requests cover a wide range of responses that are normally a naturally coherent response to an utterance.

c. Covering Breadth and Depth of Conversations: For covering breadth of a conversation, we chose topics and tried to cover a range of attributes in it following a hierarchy, for varied responses. For example, in the movies/TV shows, hierarchies could be Genres, Movie/TV show names and director/actor/music director names. We tried to explicitly cover the hierarchies to obtain different responses and create a more complete corpus in terms of response types on the subject. The idea behind this is, since we are asking for specific responses from Turkers, the named entities could be easily replaced from the responses to create response templates. Providing an initial prompt produces responses that are only representative of conversation starters. To increase the depth of a conversation, we needed responses that are reflective of a continuing conversation. To get such responses, we created levels of task for the Turkers on the same topic of conversation. The initial prompt (level 1) created responses from Turkers which were collected and used as the prompts for the next level (level 2). In level 2 though, we provided Turkers both level 1 prompt and a chosen response to that prompt and asked them to continue the conversation. In the next level (level 3), Turkers were provided the initial prompt, the initial response, and a chosen response to those first two prompts. In this way, we kept increasing the depth of the conversation until we chose to stop it. On every level we selected a number and types of responses to increase variety in the conversation. The diagram in the following section should make the process clearer.

d. Evaluation of Response and Selection of Next Prompts: The cost of carrying forward every response from a previous level to the next as a prompt would be extremely high because the number of prompts for each level grows exponentially depending both on the number of Turkers doing the HITs and as well as the number of categories of responses requested. Naturally, the most diverse and interesting conversations continue, while more mundane or repetitious conversations tend to die out. We tried to simulate this through pruning of our HITs. For the time being, we don’t have any specific algorithm on choosing the responses based on their qualities. One can imagine a few explicit rules could be made in selecting the responses for the next level of task. For example, it might not be interesting to hear a series of responses which are questions following one another or a series of topic changes following one another. Nevertheless, we have responses that defy these ideas and turn out to be interesting and qualified for the next round. Therefore, we have handpicked responses for the prompts for every round based on interesting, naturally sounding dialog.



e. Future Work: Ending Conversations: After a certain level of conversational depth, we plan to wrap up the conversation with a final couple levels. To be practical, a conversation does not simply end, but is brought to a close. [11] The problem of closing a conversation can be dealt by ending the conversation in a few more turns (levels) and hence making the transition much smoother with more content. For the time being we came up with the idea of "wrapping the conversation up" with a summary of the entire conversation and an ending note. This is another aspect of naturally occurring language that needs further work. We are also interested in how a conversation recovers from a "bad response" or utterance. Normally, someone might say something uncouth or in bad taste or maybe even insulting. Generally, the conversation continues. We are interested in how to recover from such a misstep. What is your conversational partner just says something out of the blue, which is totally off topic? Well, chat bot are more likely to do this. Recovering the conversation from such a gaff is important to any chat bot system. We would like to run some HITs to include that eventuality.

3 Evaluation and Results

In terms of quality, we have coherent responses for the given prompts in most of the cases. Although, in some cases we did receive a few responses that weren't acceptable. In these cases, we contacted the Turkers and asked them to do their task once more. For every level of a task we downloaded the CSV and formatted in a way so that it is easy to visualize the different levels of responses. Moreover, we have a constant record of the Worker IDs so that we can keep track of efficiency of Turkers over the time. In this way, we could eliminate inefficient Turkers from our future task and request efficient once to participate even more and offer them bonus as well. This could assure us with a superior quality of responses in creating future corpus. The picture below shows a sample of collected dialogues for different levels.The prompt was initially provided for different categories of responses. Then for each level, the prompt was modified with appending the selected responses till the initial prompt (like described in the diagram).

Dialogue Act	Responses
Prompt	I just could not get into "Lost."
Answer.negative(Level1)	But on the whole I didn't like it because it just seemed like a lot of trumped up drama, if you know what I mean.
Answer.joking_sarcastic(Level2)	So, basically what you're telling me is that it was so confusing drama you got "lost"?
Answer.elaboration(Level3)	I really did, at least in my mind! Watching all these people on an island with all these different relationships to each other just got to be too much to deal with.
Answer.change(Level4)	I think that may have been the appeal. The feeling of caring, but still being in the dark in the whole show. Similar to how the characters felt, or were showed to feel.

An analysis of the quality of responses were performed over a few attributes. They are compared over categories and also depth of conversations. Here is a result of the analysis:

	Change	Continuation	Elaboration	Joke/Sarcasm
Character count/ response	66.34	71.43	70.17	60.87
Word count/response	13.03	14.13	13.80	11.85
Unique word frequency	0.39	0.37	0.32	0.45

	Similar	Specific	General	Question
Character count/response	64.05	87.18	78.94	64.29
Word count/response	12.28	16.29	14.98	12.24
Unique word frequency	0.41	0.55	0.42	0.44
Named entities / response	0.48	0.66	0.57	0.39

	Level1	Level2	Level3	Level4
Character count/ response	61.01	83.57	62.23	57.12
Word count/response	11.78	18.22	11.56	10.57

From the analysis, we see that categories are a better method to find the quality of responses. Maybe the significance of the levels is not prominent since we experimented only till the fourth level. As expected the categories "elaboration" and "continuation" has got more character and word count compared to the other categories. In the second experiment, we had specific category which was supposed to extract more specific responses on the provided prompt.Here, we have the most character and word counts for the "Specific" category. Also, it makes complete sense that the unique word percentage and named entities per response, is the highest for specific category itself. For finding significant categories from LIWC we

also performed an unpaired t test on the data from our past experiments and the recent experiments, to find significant categories. Out of 91 categories we have 46 categories that appeared to be significant with p values less than 0.5. A few of the important categories with their calculated p-values are shown below:

Categories	P-values
Adverbs	1.78053820706e-09
Auxiliary Verbs	7.01245240516e-17
Causation	0.00123535356732
Feel	0.00189280370792
Insight	5.17866339714e-06
Social processes	2.52525329749e-13
Time	0.0013738990476

4 Conclusion

The result seems to be satisfying in terms of quality of data, as well as time and money invested to fetch them. The system works as a smart utterance pair generator and could be used as building a varied and extended corpus on any subject. Nevertheless, there are scopes of improvement in the system. The problem of handling enormous data and organizing it more efficiently is the biggest challenge. The performance parameters such as character count, named entity frequency etc. does provide us with an insight of a quality of the responses. We could come up with a model that select top responses from a certain level and select them as prompts for the next one. Also, we could come up with strict rules like we won't be accepting responses such as "question followed by a question" or "elaboration followed by another elaboration". These are yet to be researched and optimized.

5 References

1. Kobayashi, Y., Mizukami, M.: Towards taxonomy of errors in chat-oriented dialogue systems. In: Proceedings of the 16th Annual Meeting of the Special Interest Group on Discourse and Dialogue, pp. 87–95 (2015)
2. Lasecki,W.S., Song, Y.C., Kautz, H., Bigham, J.P.: Real-time crowd labeling for deployable activity recognition. In: Proceedings of the 2013 conference on Computer supported cooperative work, pp. 1203–1212. ACM (2013)
3. Mitchell, M., Bohus, D., Kamar, E.: Crowdsourcing language generation templates for dialogue systems. In: Proceedings of the INLG and SIGDIAL 2014 Joint Session, pp. 16–24 (2014)
4. Novikova, J., Lemon, O., Rieser, V.: Crowdsourcing nlg data: Pictures elicit better data. In: Proceedings of the 9th International Natural Language Generation conference, pp. 265–273 (2016)
5. Otani, N., Baba, Y., Kashima, H.: Quality control of crowdsourced classification using hierarchical class structures. Expert Systems with Applications 58(1), 155–163 (2016)
6. Paperno, D., Kruszewsk, G., Lazaridou, A., Pham, Q.N., Bernardi, R., Pezzelle, S., Baroni, M., Boleda, G., Fernandez, R.: The lambada dataset: Word prediction requiring a broad discourse context. In: Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, pp. 1525–1534 (2016)
7. Sugiyama, H., Meguro, T., Higashinaka, R., Minami, Y.: Open-domain utterance generation for conversational dialogue systems using web-scale dependency structures. In: Proceedings of the SIGDIAL 2013 Conference, pp. 334–338 (2013)
8. L. von Ahn and L. Dabbish. Labeling images with a computer game. In Proceedings of the conference on Human factors in computing systems, CHI '04, pages 319–326, New York, NY, USA, 2004. ACM.
9. L. Chilton. Seaweed: A web application for designingeconomic games. Master's thesis, MIT, 2009.
10. Chorus: Letting the Crowd Speak with One Voice. University of Rochester Technical Report no. 983 Walter S. Lasecki, Anand Kulkarni, Rachel Wesley, Jeffrey Nichols, Chang Hu, James F. Allen, and Jeffrey P. Bigham.
11. Opening up Closings[Emanuel A.Schegloff, Harvey Sacks]