Deep Conversations (with T5)

Cody Kesler, Michael Nelson

April 18, 2020

1 The Objective: Spamming the Spammers

Our objective in this project was to create an email system that could repeatedly reply to spam emails, carrying on a drawn-out conversation and wasting spammers' time. In order to accomplish this we decided to try training a deep sequence-to-sequence transformer, known as T5 [3], which could take in several lines of back and forth conversation as input, and output a new meaningful response.

2 Data

One of the most challenging parts of this problem was finding meaningful data in a sufficient quantity. There are many datasets that contain text responses to some prompt (such as replies to social media posts), but it is much harder to find conversation data with multiple back and forth responses between two individuals. Some of the larger conversation datasets we found were sourced from countries where English is not the native language, likely where data privacy laws are not as strict and were not suitable. In the end, we used the Wizards of Wikipedia dataset [2], which contains 18430 conversations between two agents. The average conversation length is six lines, and each conversation is labeled with an assigned persona and topic.

3 Dataset and Formatting

We tested two different formats for the dataset. Since we had decided to use the pretrained T5 network, we mimicked the T5 task-specific setup [3] by providing specific tokens ("Topic:", "Prompt:", etc.) followed by text which would indicate to the network what that text represented.

3.1 Full Conversations

Initially, in line with our original goal, we provided the network with the multi-line conversations. Specifically, each training entry would contain a tag indicating the topic, and two repeating tags representing each member of the conversation.

Each epoch, every conversation would be presented up to one time, containing exactly one more line than the previous time used. The first time presented, the entry would contain only the topic tag. The epoch length was restricted to be much smaller than the total number of conversations so that the network could still encounter short or empty conversations late in training.

Unfortunately, due to the small size of the dataset in question, this required that the dataset be reset occasionally, exposing the network to the same conversations again. When this happened, the average length of training elements shortened from six lines to zero lines, which resulted in significant disruption to the training process and a cyclical training path.

3.2 Prompt, Topic, and Response Only

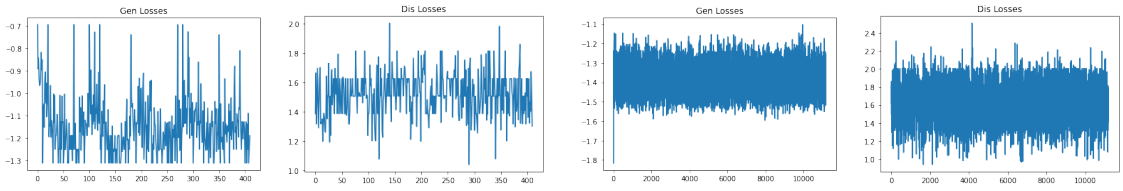
To avoid the problems encountered with the above approach, we decided to simplify the problem by presenting the network only the topic, and a single random line from the conversation. While this didn't lengthen the dataset over the above approach, it did eliminate the cyclic variable length of input and hid the order the lines appear in the conversation from the network, allowing samples to be reused without fear of the network memorizing each conversation.

We tested both of these dataset structures with each network structure explained below.

4 Training Methods and Results

4.1 GAN

Our first approach was to use a GAN Structure. We used pre-trained T5 models like both the generator and discriminator. The generator received a prompt (the topic and prior conversation, or the topic and previous line only) and generates a response. The discriminator was given the same prompt with either the fake generated response or the true response appended at the end. It is trained to return a probability of two classes, generated or real. Using the traditional GAN loss [1], we train the generator 5 times more than the discriminator.



(a) Gen Loss 1 (b) Dis loss 1 (c) Gen Loss 2 (d) Dis loss 2

Figure 1: GAN Structure Losses

4.1.1 GAN Results

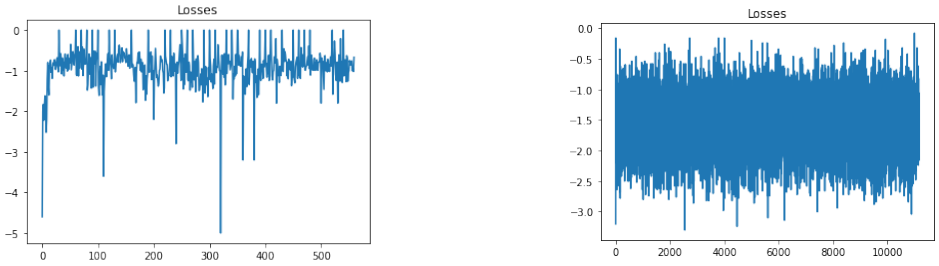
In the pursuit of results, we ran our GAN architecture on both types of data loading. For our first dataset, we can see the loss of the generator starts to decrease, but has a

cyclical nature as our dataset resets as seen in Figure 1(a). This suggests our dataset is not large enough to train this network for our task. Unfortunately, using the more simple dataset, the loss for the generator never started decreasing at all.

4.2 Sentence Similarity and Cohesion Loss

With the GAN structure not panning out as well as we wanted, we decided to try a more simple architecture. We still used a pre-trained T5 as the response generator, but with the loss being a weighted combination of pre-trained T5 outputs from the original paper as the loss (sentence similarity the to true response and sentence cohesion –tags "stsb" and "cola" in [3]). When the output of the generator was not close to a meaningful sentence, these pre-trained networks commonly output non-float values, in which case we simply maximized that portion of the loss.

4.2.1 Weighted Loss Results



(a) Gen Loss 1

(b) Gen loss 2

Figure 2: Losses for new Structure

We tested this new loss on both data formats and as seen above in Figure 2 with similar results to our GAN structure. For the first data format the losses did start converging but experienced the same cyclical nature and convergence never happened with the second dataset.

5 Conclusion

As both networks performed the same over both data formats and the first format indicated loss convergence, we note the dataset is the limiting facotor. Thus, our results indicate that having a larger, more diverse training set and longer training time have would produce loss convergence and network learning of producing a conversing network.

References

[1] Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale gan training for high fidelity natural image synthesis, 2018.

[2] Emily Dinan, Stephen Roller, Kurt Shuster, Angela Fan, Michael Auli, and Jason Weston. Wizard of Wikipedia: Knowledge-powered conversational agents. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2019.

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