Creating a Chatbot for Three-Way Conversations

Amber Li amli
Emma Liu emmaliu
Julia Wang jwlw2022
Kate Xu katexu

Abstract

A lot of natural language processing (NLP) research features dialogue systems and chatbots that can synthesize two-way conversations. We took these conversations a step further to include three participants. We trained a Seq2Seq model on the PERSONA-CHAT *dataset and adapted it to train our chatbot for three-way conversations.* [1] *We improved our Seq2Seq model performance by adding attention and adjusting the embedding size and number of layers. To evaluate the model's performance, we used perplexity as a quantitative measure and crowd-sourced ratings on qualitative metrics including engagingness, consistency, rationality, and creativity. We discovered that validation perplexity decreases as embedding size increases, but an embedding size of 512 produced the most interesting, sensible, and original responses. Increasing the number of hidden layers in the gated recurrent units (GRUs) in the encoder and decoder improved the performance of our chatbots. Overall, our work demonstrates that we can create a chatbot that generates engaging, consistent, rational, and creative responses for three-way conversations by training a Seq2Seq model on the* PERSONA-CHAT *dataset.*

1. Introduction

Chatbots, also known as dialogue systems or conversational agents, are instrumental to language learning and entertainment. Prior research involves chatbots that generate dialogue in settings with two people [1]. To expand on this, we are interested in whether a chatbot can perform well in settings with more than two *interlocutors*, or participants in a conversation. Specifically, we are interested in building a chatbot that generates engaging, informative, and consistent dialogue in chit-chat settings with three interlocutors.

There are different types of chatbots, including *corpus-based chatbots* that leverage dialogue datasets to generate appropriate responses [2]. These datasets include telephone conversation transcripts, movie dialogues, crowd-sourced conversations, and pseudo-conversations such as those from Twitter and Reddit. Furthermore, there are two main approaches by which a chatbot generates a response: response by retrieval and response by generation. Response by retrieval uses information retrieval methods to find a response that is appropriate given the dialogue context. In contrast, response by generation uses an encoder-decoder model or language model to generate a response given the dialogue context [2]. Language models are often pretrained on larger text corpora and subsequently fine-tuned on conversational corpora. In this paper, we create a corpus-based chatbot that uses response by generation.

There are advantages and disadvantages to using corpus-based chatbots. They can be fun for chit-chat, function as social robots, and be used to evaluate cognitive function. Many people may enjoy having a chatbot that can patiently "listen" to deeply personal monologues. However, information retrieval-based chatbots may only mirror training data, and generation-based chatbots may speak nonsense. In general, these chatbots lack the ability to understand dialogue. They may also embrace inconsistent personalities, generate uninteresting or vague answers, and lack an explicit long-term memory [1].

Some of the issues that existing corpus-based chatbots face may be caused by the quality of the dataset used. For our chatbot, we use the PERSONA-CHAT dataset, which was created to address these issues. This dataset contains multi-turn dialogues between two interlocutors that are conditioned on personas [1], and we adapt the dataset to our chatbot for dialogues among three interlocutors.

Our chatbot takes as input consecutive dialogue between two interlocutors from the PERSONA-CHAT dataset, and it generates a response based on the given dialogue. First, we use a Seq2Seq model without Bahdanau attention. Second, we add attention to improve the Seq2Seq model. We fine-tune hyperparameters such as the embedding size and number of layers, which results in different versions of each model. We also train, validate, and test all of our models using the PERSONA-CHAT dataset.

We evaluate our chatbots using quantitative and qualitative metrics. Our primary quantitative metric is the validation perplexity of the model after training for 10 epochs.

[1] NLP Chatbot Project Github Repository: https://github.com/jwlw2022/nlp-chatbot-project.git

The validation perplexity measures the model's ability to predict a desired response after learning on training data. In general, a lower validation perplexity indicates a better chatbot. For the qualitative evaluation, we designed a survey with five example dialogues from the PERSONA-CHAT dataset. We also included each of our chatbots' responses to the dialogues. We asked 71 volunteers to rate each chatbot response on four characteristics: engagingness, consistency, rationality, and creativity.

Overall, we conclude that a Seq2Seq model with an embedding size of 512 and 3 hidden layers performs the best. We also find that there is little to no improvement in our model's responses after implementing attention.

2. Related Work

2.1. PERSONA-CHAT Dataset

The PERSONA-CHAT dataset is crowd-sourced via Amazon Mechanical Turk. PERSONA-CHAT conditions on profile information, or personas, and it aims to improve dialogue using these personas. We use the PERSONA-CHAT dataset to train our chatbot.

There are three stages of data collection for the PERSONA-CHAT dataset:

1. Personas: crowd-sourced 1,155 possible personas
2. Revised personas: additional rewritten sets of personas with related sentences that are rephrases, generalizations, or specializations of the original personas
3. Persona chat: pair two individuals, assign each with a random persona, and allow them to chat

Each persona consists of five sentences with a maximum of fifteen words per sentence. This maximum is set because people would lose interest and machines struggle with longer persona sentences. In addition, revised personas are used to mitigate the problems of word overlap from previous datasets such as SQuAD. Because humans might accidentally repeat profile information verbatim or nearly word-for-word, the PERSONA-CHAT dataset uses rewritten sentences that have similar implications but different meanings. Lastly, to collect dialogues, or persona chats, [1] pairs two random individuals who roleplay a conversation with six to eight turns per dialogue.

[1] uses configurable but persistent persona to create more engaging chatbots. They encode profiles of interlocutors with five sentences of descriptive text, and they store the profiles in a memory-augmented neural network. They also train Seq2Seq models and memory networks on the PERSONA-CHAT dataset to produce more personal, specific, consistent, and engaging responses than persona-free models. For evaluation, they use next utterance prediction to assess the quality of the chatbot dialogue.

2.2. Generative Models

Generative models come up with new sentences by conditioning on dialogue history and generating responses one word at a time to expand the number of potential conversations. [6] has extended the hierarchical recurrent encoder-decoder neural network to the dialogue domain. They demonstrated that this model is competitive with state-of-the-art neural language models and n-gram models. The performance of the hierarchical recurrent encoder-decoder neural network can be improved by bootstrapping the learning from a larger question-answer pair corpus and from pre-trained word embeddings.

2.3. Mutual Persona Perception

Most current work in chit-chat dialog systems focuses on imitating human responses instead of modeling understanding between interlocutors. [4] developed the Persona Perception Bot ($\mathcal{P}^2$ Bot). The $\mathcal{P}^2$ Bot explicitly models understanding between interlocutors with a transmitter-receiver-based framework to explicitly model understanding using a PERSONA-CHAT experiment. Mutual persona perception describes an information exchange process that allows interlocutors to get to know each other. [4] developed a transmitter for dialog generation and a receiver for mutual persona perception. The receiver measures the proximity between built impressions and actual personas. Impression encoding is one interlocutor's impression of the other based on utterances. Mutual persona perception serves as reward signal to achieve personalized dialogue generation.

3. Methods

We modified the PERSONA-CHAT dataset [1] to mimic conversations among three interlocutors and formatted it to be suitable as input to our models. Our baseline model uses a Seq2Seq encoder-decoder architecture with 1 layer in each gated recurrent unit (GRU) and an embedding size of 256. As shown in Figure 1, our attention Seq2Seq model has an encoder-decoder architecture that takes in an input that consists of two sentences concatenated together, processes them with RNNs and hidden states, and passes them through attention heads to compute attention scores and outputs a prediction. We used Bahdanau attention to enhance the baseline Seq2Seq model. We evaluated each model's performance using validation perplexity as well as qualitative evaluation metrics.

3.1. Data Preprocessing

We adapted the PERSONA-CHAT dataset [1] to train our chatbot for three-way conversations. Each entry in our dataset has an input, which consists of two consecutive sentences of dialogue from the PERSONA-CHAT dataset concatenated together, and a target, which is the sentence of

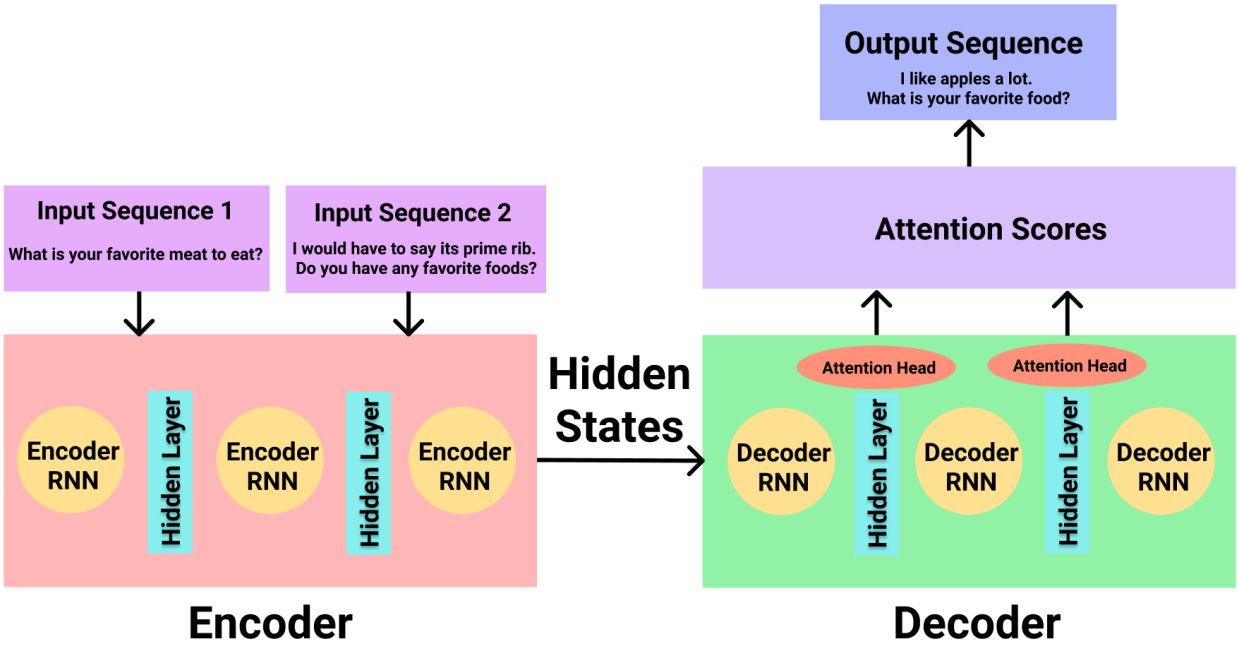


Figure 1: System diagram of Seq2Seq Encoder-Decoder Architecture with Attention and Dialogue from Example 1

dialogue immediately after the two sentences used for the input. The idea is that the model output will be the response of the third interlocutor, as if it is interjecting and commenting on the conversation between the other two interlocutors. The size of our modified dataset was 244,996 training sentences and 14,600 validation sentences.

We parsed and tokenized the utterances for conversations in the PERSONA-CHAT dataset. To get the input sentences for our dataset, we concatenated pairs of consecutive sentences from each conversation together and set the maximum tokenized list length to 45 since we found that longer utterances were very rare. Then, the sentence immediately following the pair of consecutive sentences is used as our target. In this way, we adapted the PERSONA-CHAT dataset with conversations between two interlocutors to our project for three interlocutors.

3.2. Seq2Seq

Seq2Seq is an encoder-decoder model that maps sequences to sequences [7]. It is implemented using a recurrent neural network (RNN) such as GRUs or long short-term memory (LSTM). Major uses of Seq2Seq include machine translation and conversational modeling. Our baseline Seq2Seq parameter was adapted from the Homework 3 code, but first we had to modify it to take in our dataset input sentences, and we also experimented with adding an attention module to create a better model.

For both the Seq2Seq model without attention and the Seq2Seq model with attention, we experimented with these parameters:

- Number of layers in the GRUs of the encoder and decoder (1 or 3)
- Embedding and hidden size (256, 512, or 1024)

3.3. Bahdanau Attention

We also introduced an attention mechanism into our baseline encoder-decoder model. To do so, we implemented an MLP-based attention module that helps the decoder search through the input sentence for particular parts to pay attention to [9]. Figure 2 shows a generalized model architecture for Seq2Seq with attention. The query is the decoder state and the keys are the encoder states. The attention module then produces attention weights that attend to the values, or encoder hidden states.

In Bahdanau attention specifically, there is a context vector c_i, computed for each word of the input sequence, that depends on the previous decoder hidden state s_{i-1}, as well as all the encoder hidden states $h_1, \cdots, h_M$. This context vector is used to compute each RNN hidden state:

$$s_i = f(s_{i-1}, y_{i-1}, c_i).$$

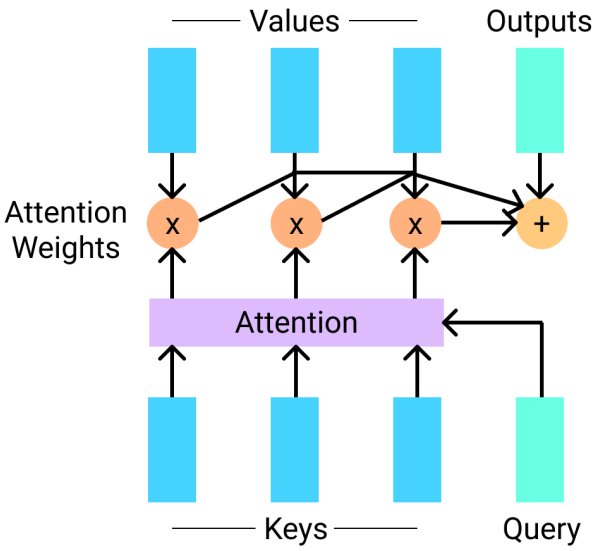


Figure 2: Model Architecture of Seq2Seq with attention

To get the context vector c_i, we compute a weighted sum of the h_i,

$$c_i = \sum_{j=1}^{T} \alpha_{ij} h_j,$$

where the weights α_{ij} are computed by a softmax over attention energies e_{ij},

$$e_{ij} = a(s_{i-1}, h_j).$$

Here, the e_{ij} are the result of an *alignment model* that reflects how well the input sequence around index j and the output sequence at position i match [9]. In particular, we use a linear layer and tanh activation so that the entire model is conducive to end-to-end backprop.

3.4. Validation Perplexity

Intuitively, perplexity can be understood as a measure of uncertainty. The perplexity of a language model is essentially the number of plausible possibilities when predicting the next symbol [10]. [11] showed that better "perplexity for the masked language modeling objective" leads to better "end-task accuracy" for sentiment analysis and multiple genre natural language inference. Better perplexity leads to better performance on downstream tasks [11].

Perplexity is a simple, multifunctional, and powerful metric that can be used to evaluate not only language modeling, but also for any generative task that uses cross entropy loss, such as speech recognition and open-domain dialogue [10].

Although validation perplexity is correlated with the performance of each chatbot, we do not believe it is a comprehensive metric to assess the quality of our Seq2Seq model training with and without attention. For this reason, we also use a qualitative evaluation to analyze our models.

3.5. Qualitative Evaluation

To qualitatively evaluate the performance of our chatbots, we asked 71 people to fill out a form sharing five examples of chatbot dialogue. Each respondent read the dialogue transcript of five conversations among three interlocutors, and then they gave ratings for the following four characteristics on a scale of 1 (lacks the characteristic) to 5 (satisfies the characteristic):

- Engagingness: interesting content
- Consistency: in agreement with previous dialogue
- Rationality: makes sense in isolation
- Creativity: avoids repetition

The respondents are not aware of the embedding size and layer differences among the five chatbots. They evaluated the quality of the chatbot responses using our instructions by reading only the five conversation examples.

4. Models & Experiments

4.1. Hyperparameters

For experiments, we fine-tuned the following hyperparameters on both our Seq2Seq without attention models and our Seq2Seq with attention models:

- Number of layers in each GRU
- Embedding size

We discovered that 3 layers works best qualitatively because the model learns more of the context and comes up with more engaging responses.

4.2. Seq2Seq without Attention

The Seq2Seq model consists of an encoder module and a decoder module. Our encoder contains a GRU parameterized by its embedding and hidden size and number of layers. In our implementation, we packed our inputs and outputs, and the encoder returns the outputs from the last layer of the RNN as well as the final hidden state.

The decoder also consists of a GRU parameterized by its embedding and hidden size and number of layers. In our implementation, we unrolled the decoder one step at a time, where each forward step produces decoder output from the previous embedding and hidden state by passing them through the RNN, concatenating the previous embedding with the RNN output at this step, then applying

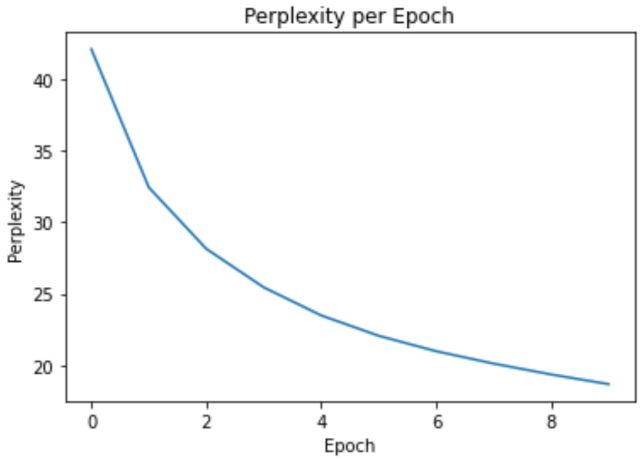


Figure 3: Perplexity of Seq2Seq without attention, embed size = 256, and 1 layer shown for 10 epochs.

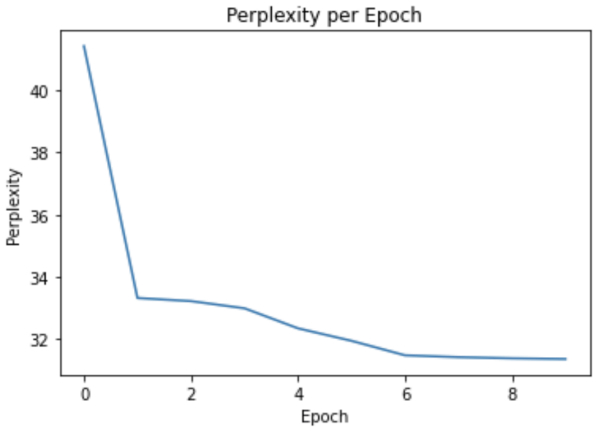


Figure 4: Perplexity of Seq2Seq without attention, embed size = 512, and 3 layers shown for 10 epochs.

dropout and a linear transformation. To initialize the decoder's first hidden states from the encoder, we used a linear transformation with tanh activation.

In training, we set the embedding size and hidden size as well as the number of layers. Dropout was always set to 0.2, and all models were trained for 10 epochs using the Adam optimizer (learning rate 0.001) and NLL loss. Figures 3 and 4 show the perplexity per epoch during training of the Seq2Seq without attention model using different embedding sizes and number of layers. As we can see in the plots, we do achieve convergence (or come close to convergence). From Figure 3, it seems that we could train for more epochs, but since Figure 4 seems to demonstrate convergence, we chose to stick with 10 epochs for consistency across all of our experiments.

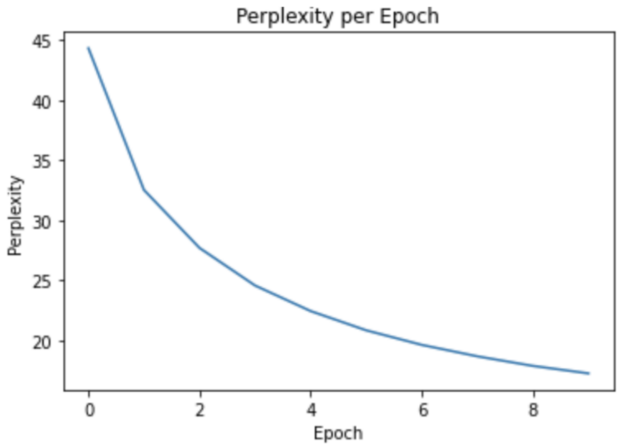


Figure 5: Perplexity of Seq2Seq with attention, embed size = 256, and 1 layer shown for 10 epochs.

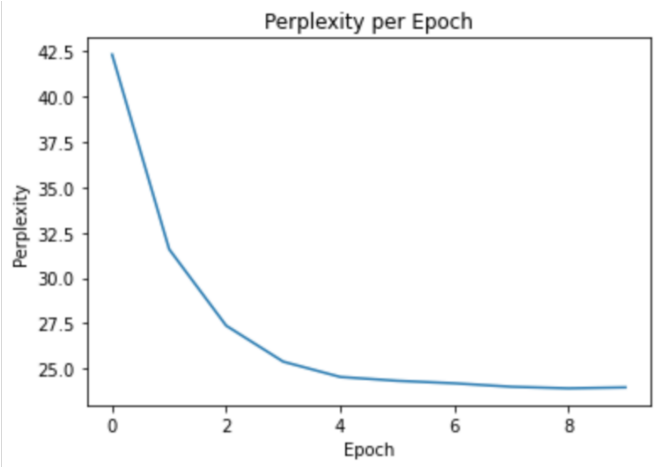


Figure 6: Perplexity of Seq2Seq with attention, embed size = 512, and 3 layers shown for 10 epochs.

4.3. Seq2Seq with Attention

We implemented a Seq2Seq model with Bahdanau attention as described in Section 4.2 [8]. For this model, the encoder module is the same as before, and the decoder is largely the same as well. However, the model uses the Bahdanau attention module to compute context, which is then concatenated with the previous embedding to become input to the RNN. The context is also used directly in computing the output of the decoder. We hypothesized that attention would improve chatbot performance.

Figures 5 and 6 illustrate the perplexity per epoch during training of the Seq2Seq with attention model using different embedding sizes and number of layers. Similarly to the plots for the models with the same embedding sizes and number of layers but without attention, Figure 6 demonstrates convergence while Figure 5 comes close to con-

vergence but may benefit from training for a couple more epochs. The models in Figures 4 and 6 generated dialogue that was qualitatively ranked the highest in Section 5.

5. Results

We trained eight Seq2Seq models with and without attention. They differ in embedding sizes (256, 512, or 1024) and number of hidden layers (1 or 3). Note that we trained models F, G, and H after we requested qualitative feedback, so they do not appear in Figures 7 to 11.

Label	Attention	Embed Size	Layers	Perplexity
A	No	256	1	18.694
B	No	512	1	13.454
C	No	512	3	31.352
D	Yes	512	1	9.423
E	Yes	512	3	23.972
F	Yes	256	1	17.278
G	Yes	1024	1	6.983
H	No	1024	1	8.524

Table 1: Validation Perplexities of Various Seq2Seq Models after 10 Epochs

We refer to each chatbot as either Model or Chatbot accompanied their letter label, e.g., Model A or Chatbot A. As shown in Table 1, we saw the lowest perplexity with attention, an embed size of 1024, and 1 layer. In general, perplexity was lower for larger embedding sizes but was higher with more layers. Perplexity was also lower with attention. However, we expected attention to significantly improve our model results, but we did not observe as large of a decrease in perplexity as expected.

We used qualitative metrics to evaluate the performance of our chatbot. We asked 71 people to evaluate five examples of chatbot dialogue. Each outside observer read the dialogue transcript of a conversation among three interlocutors, and then they evaluated the quality of the chatbot dialogue based on the following four factors: engagingness, consistency, rationality, and creativity. The respondents did not know the training differences between the five chatbots. They evaluated the quality of responses solely based on these five dialogue examples.

5.1. Chatbot Comparison Discussion

Note that we only asked outside observers to evaluate 5 examples, so these results are not necessarily an accurate representative of all predictions by our chatbot. Based on qualitative evaluation, the best models were C and E. The qualitative rankings for Model C (no attention, 512, 3 layers) and Model E (attention, 512, 3 layers) are very close across all five examples. Thus, any improvements due to attention could have been masked by the variance of the responses.

We made the following observations based on our training results.

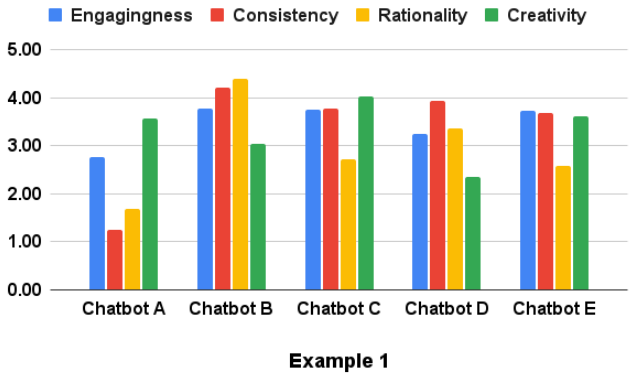
1. In general, Model A generates predictions with the same sentence structure as either Person 1 or Person 2, but it does not understand context.
2. Changing embedding size influences predictions more than adding attention.
3. An embedding size of 512 had lower validation perplexity because it had more plausible options to choose from than an embedding size of 256.
4. Models with more layers improved predictions, but their training generated higher validation perplexity values at epoch 10.
5. Increasing the number of hidden layers to three added more complexity to the models. We observed more creative responses from models with three layers rather than one.

Accordingly, we observed that the predictions using an embedding size of 512 are more reasonable than those using an embedding size of 256 in terms of relevance.

5.2. Dialogue Example 1

```
Person 1: What is your favorite meat to
eat?

Person 2: I would have to say its prime
rib. Do you have any favorite foods?

Chatbot A: I like to petition for a
living.

Chatbot B: I like apples a lot. What is
your favorite food?

Chatbot C: I like to eat cuba libres.

Chatbot D: I like food, I eat a lot of
meat. I like food and Mexican food.

Chatbot E: I love homemade coffee. I eat
it every day.
```

Note that cuba libre is a mixed drink consisting of coke and rum. In Example 1, Chatbot A's response is unrelated to Person 1 or Person 2's comments. Meanwhile, Chatbots B, C, D, and E generate predictions that are relevant given the context. Chatbot D's response is detailed and consistent but doesn't name a specific meat; additionally, the second



(a)

Example 1	Engagingness	Consistency	Rationality	Creativity
Chatbot A	2.76	1.24	1.69	3.56
Chatbot B	3.77	4.21	4.39	3.03
Chatbot C	3.75	3.77	2.72	4.03
Chatbot D	3.24	3.93	3.35	2.34
Chatbot E	3.72	3.69	2.58	3.62

(b)

Figure 7: *Qualitative evaluation ratings for chatbot responses to Example 1 dialogue.*

sentence seems to imply that Mexican food is not a type of food, which is rather strange.

These models tend to mimic the sentence structure of Person 1 or Person 2's dialogue, so they generally do well at producing coherent, grammatically-correct sentences. However, their generated responses seem to lack understanding or semantic meaning in some ways. Here, we see that Chatbots C and E appear unaware that cuba libre and coffee are drinks and therefore cannot be eaten, while Chatbot D's response contains redundant statements about liking food.

Figures 7 (a) and (b) show results from the survey for Example 1, with the table giving exact mean values. Respondents rated each chatbot response on a scale of 1 to 5 for each evaluation metric (engagingness, consistency, rationality, and creativity). A rating of 1 indicates that the chatbot did not satisfy that characteristic. A rating of 5 means the evaluator thought the chatbot did an excellent job of displaying that characteristic.

As shown in Figure 7(a), Chatbot B had the best response overall. It scored high on engagingness, consistency, and rationality, which is arguably the most typical for a human response. However, respondents thought chatbot B was less creative compared to the other chatbots. On the other hand, our baseline Chatbot A had the worst ratings as we expected: respondents thought that its response was not consistent and less rational compared to the responses of other chatbots.

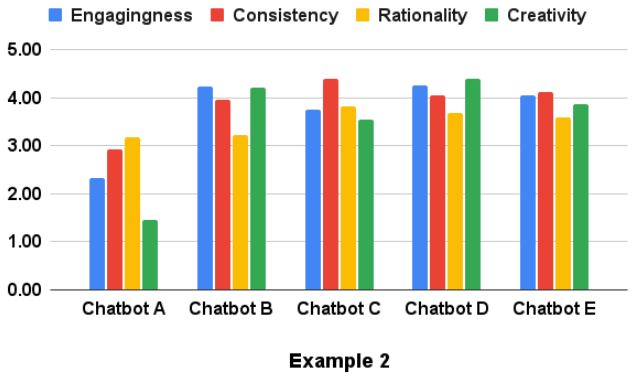
Both chatbots C and E scored fairly high on the engagingness, consistency, and creativity scales, but they had lower rationality points, likely because cuba libre and coffee are drinks that cannot be eaten. For example, one respondent noted, "Eating coffee and cuba libres seems slightly non-sensical, so I lumped that with 'rationality.'" Nonetheless, chatbot C was the most creative, possibly because it mentioned the "cuba libre."

Table 7(b) displays the mean rating on a scale of 1 to 5 for each chatbot and evaluation metric (engagingness, consistency, rationality, and creativity) pair. The data suggest that Model A performed the worst and Model B performed the best.

5.3. Dialogue Example 2

```
Person 1: What do you do for a living?

Person 2: I'm a researcher. I'm
researching the fact that mermaids are
real.

Chatbot A: What do you do for a living?

Chatbot B: Interesting. I'm a teacher.
I spend most of my time playing video
games.

Chatbot C: I am a writer. I am blogging.

Chatbot D: I am a musician and enjoy
going fishing.

Chatbot E: I'm an accountant, but I bet
that's an interesting job.
```

In Example 2, Chatbot A repeats the question from Person 1. This is interesting because, although the model could have just memorized that sentence from the input sequence, the repeated question could make sense in this context if it were directed back at Person 1. However, in a typical human-to-human conversation, a person would usually answer the question asked before directing the question back



(a)

Example 2	Engagingness	Consistency	Rationality	Creativity
Chatbot A	2.32	2.93	3.17	1.45
Chatbot B	4.24	3.96	3.21	4.20
Chatbot C	3.75	4.39	3.82	3.55
Chatbot D	4.25	4.04	3.68	4.38
Chatbot E	4.06	4.11	3.58	3.86

(b)

Figure 8: *Qualitative evaluation ratings for chatbot responses to Example 2 dialogue.*

at the other person. This distinction makes Chatbot A's response less appealing than those of the other chatbots.

The responses from Chatbot B and E remark that Person 2's job is interesting, and Chatbot B even shares that they like playing video games aside from their job. Chatbots C and D answer the question more straightforwardly, without additional commentary on Person 2's response. Overall, these responses are all well-formed sentences and make some sense.

We see evidence in Figure 8(a) that the survey respondents agree that Chatbots B, C, D, and E performed well in terms of engagingness, consistency, rationality, and creativity, probably because they directly answered the question "What do you do for a living?". Respondents believed that Chatbot A, in comparison, lacked all four, with rationality being its highest rated characteristic.

As shown by all the green cells in Table 8(b), respondents generally agreed that all of the chatbots except for Chatbot A did well with answering the input sentence in Example 2. Notably, Chatbot C had the highest consistency rating, probably because being a writer and blogging are consistent ideas.

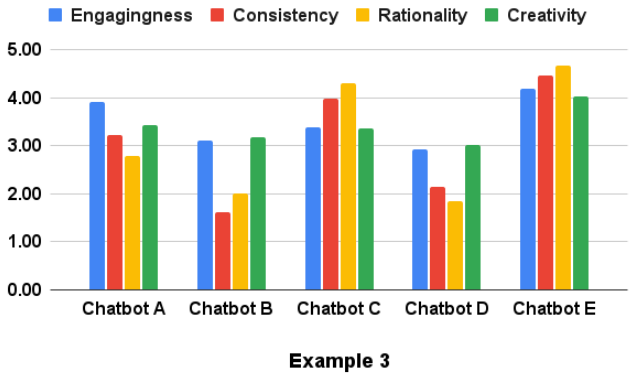
5.4. Dialogue Example 3

```
Person 1: I also remodel homes when I
am not out bow hunting.

Person 2: That's neat. When I was in
high school I placed 6th in 100m dash!

Chatbot A: Wow! I love to cook but I am
a gourmet cook.

Chatbot B: That's a great show. I love
the beach.

Chatbot C: I like to read. I also like
to hike.

Chatbot D: I am in the 10th grade
development development courses and I
can save up for awhile

Chatbot E: That's cool. I like to play
football in my free time.
```

In Example 3, Person 1 did not ask a question, so there was more room for creative responses by the chatbots. Person 1 and Person 2 both share fun facts about themselves, and so do all five chatbots, but with widely varying degrees of consistency and rationality.

For example, Chatbot C gave a rational response with somewhat uninteresting syntax, which respondents thought was less engaging and less creative (see Figure 9(b)). Most respondents thought the responses from Chatbots B and D were subpar, probably because they were irrelevant to sharing hobbies and records. In particular, Chatbot B has a very low consistency rating since it remarked "That's a great show" when there was no mention of any shows in the previous dialogue – perhaps the chatbot memorized this phrase from the training data. One could argue that these responses are somewhat creative. These results are shown in Figure 9(a).

Chatbot E gave a response that is relevant, interesting, and sounds natural. Table 9(b) provides evidence that Chatbot E dominates in engagingness, consistency, rationality, and creativity, receiving the highest ratings for all four characteristics. Chatbot A did better in this example compared to the other examples.

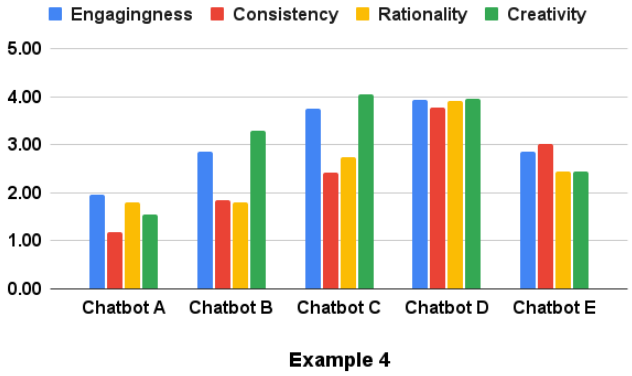


(a)

Example 3	Engagingness	Consistency	Rationality	Creativity
Chatbot A	3.90	3.21	2.77	3.42
Chatbot B	3.10	1.61	2.01	3.18
Chatbot C	3.38	3.99	4.30	3.35
Chatbot D	2.92	2.14	1.85	3.01
Chatbot E	4.18	4.46	4.66	4.01

(b)

Figure 9: *Qualitative evaluation ratings for chatbot responses to Example 3 dialogue.*



(a)

Example 4	Engagingness	Consistency	Rationality	Creativity
Chatbot A	1.96	1.18	1.79	1.55
Chatbot B	2.85	1.85	1.80	3.28
Chatbot C	3.75	2.41	2.75	4.06
Chatbot D	3.93	3.77	3.90	3.96
Chatbot E	2.86	3.01	2.44	2.45

(b)

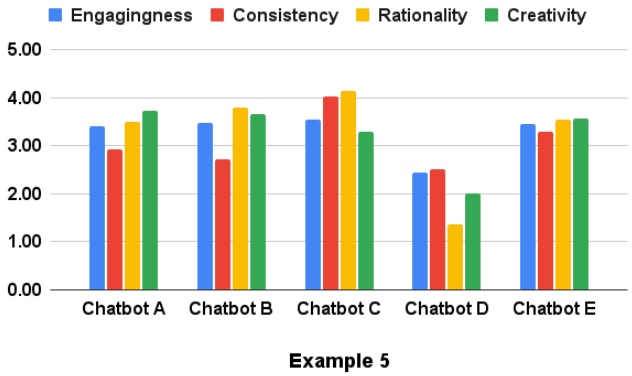
Figure 10: *Qualitative evaluation ratings for chatbot responses to Example 4 dialogue.*

5.5. Dialogue Example 4

```
Person 1: Really? What shows? I like tv,
it makes me forget I do not like my family.

Person 2: Wow, I wish I had a big
family. I grew up in a very small town.

Chatbot A: I do not like meat either.
I do not like meat either.

Chatbot B: I love to be happy to make
my own. I am a national guard officer.

Chatbot C: I am a single mom who likes
volcanos.

Chatbot D: I am sorry to hear that.
Do you like to travel?

Chatbot E: I have a family. I like to
swim. I like it too.
```

Note that Chatbot A's response is completely irrelevant. This provides more evidence that we improved upon our baseline model with small embedding size and 1 hidden layer without attention. Chatbot B and C give creative but inconsistent responses.

Chatbot D's response was the best according to our survey results in Figure 10(a). Chatbot A had the lowest ratings, as usual. Chatbot B's response was engaging and creative, but it didn't make much sense and wasn't relevant to the theme of family. Chatbot C's response was very entertaining but also off-topic. Chatbot E's response was consistent and manages to reflect the theme of family, but it was unoriginal.



(a)

Example 5	Engagingness	Consistency	Rationality	Creativity
Chatbot A	3.41	2.92	3.49	3.72
Chatbot B	3.46	2.72	3.79	3.66
Chatbot C	3.55	4.01	4.13	3.28
Chatbot D	2.44	2.52	1.35	2.01
Chatbot E	3.45	3.30	3.55	3.56

(b)

Figure 11: *Qualitative evaluation ratings for chatbot responses to Example 5 dialogue.*

5.6. Dialogue Example 5

```
Person 1: Really. But, I can sing pitch
perfect.

Person 2: I also cook, and I ride my
bike to work.

Chatbot A: Oh, I see. I am a naturalist.

Chatbot B: I want to go to the beach
and read a book.

Chatbot C: I like to read and cycle.

Chatbot D: I do not cook. I am a cook.

Chatbot E: That is so nice. I should
get a job.
```

This example was perhaps the most challenging prompt to answer because the first two interlocutors present random, disconnected ideas. Chatbots A, B, C, and D share fun facts about themselves. In particular, chatbot C gave arguably the most concise, simple answer, which might be why it received such high ratings for consistency and rationality. Chatbot D contradicts itself, which is probably why respondents ranked it low on all the metrics, especially rationality (see 11(b)) .

Chatbot E is the only one that isn't "selfish" and appears to give Person 2 a compliment for having a job. An anonymous respondent mentioned, "Again, the original conversation seems a bit awkward, and so it is hard to tell what to expect a chatbot to respond and thus evaluate consistency, although Chatbot E does somehow manage to engage well." They make a good point that the prompt is difficult for even a human to respond to because Person 1 and 2 are discussing rather different content.

5.7. Chatbots F, G, and H

We did not include models F, G, and H in the survey for qualitative evaluation because we obtained the results for these models after collecting survey responses. We also believe that including five examples and five chatbots in the survey is optimal to receive comprehensive feedback from volunteers while being respectful of their time. In general, we observed that the responses for each of the five examples from chatbots F, G, and H were worse than those from chatbots A, B, C, D, and E.

6. Challenges and Limitations

One challenge of our project is that there are not many dialogue datasets with conversations between more than two interlocutors. For example, the PERSONA-CHAT dataset has conversations between only two interlocutors. As a result, we added some data processing steps to adapt the PERSONA-CHAT dataset to our project.

In a general setting with more than two interlocutors, another challenge is that it can be difficult for a chatbot to determine the right time to interject in a conversation. However, we circumvented this challenge by having our chatbot not interject at random times but rather respond at a fixed, predictable point in the conversation. Our chatbot currently responds only at the end of a small dialogue exchange (one sentence each) between the other two interlocutors. We acknowledge that this is a limitation of our chatbot, but this simplification allowed us to make progress on this project and produce interesting chatbot dialogue for three-way conversations.

Our qualitative evaluation demonstrated that there is a

trade-off between how engaging and how rational chatbot dialogue can be. One anonymous volunteer remarked that they ranked unintelligent responses high for creativity and engagingness because they were entertaining.

7. Conclusions & Future Work

We successfully implemented a Seq2Seq model and trained models with and without attention, various embedding sizes, and number of hidden layers. We found that Chatbots C and E with the combination of an embedding size of 512 and 3 hidden layers gives the best performance qualitatively.

Increasing embedding size decreases the validation perplexity at each epoch. In contrast, adding more hidden layers in each GRU increases the validation perplexity. We believe this is due to the additional complexity of the model. With more context and connections, it is more difficult for the model to learn the "best" responses out of a larger pool of feasible answers.

Given more time, we would like to also experiment with other encoder-decoder models such as a profile memory network and key-value profile memory network. We would also experiment with different types of attention modules and multiple attention heads. Attention did not improve the Seq2Seq model performance as much as we expected. Perhaps a different form, such as Luong attention, would be more effective, though more difficult to implement in PyTorch [12].

As alluded to in Section 6, it would be nice to remove our restriction on when the chatbot "speaks," which would allow our conversation model to mimic natural chit-chat more closely and extend to longer dialogues. Additionally, because of the way we preprocessed our dataset into training input, the input sequence does not indicate to the chatbot that the given sequence actually consists of a sentence from each of two different interlocutors. We could expect this additional information to enhance the chatbot's capabilities, although it could also confuse the chatbot about who to respond to.

Finally, we have received feedback from peers suggesting a user interface to interact with our chatbot, or even having the opportunity to be in a group chat with our chatbot. These are excellent ideas we would consider pursuing in the future.

8. Contributions

We collaborated on the implementation and training of our models, as well as the research and writing of this paper. Amber did most of the baseline Seq2Seq adaptation to train on the PERSONA-CHAT dataset. Emma led the effort on adding attention and trained most of the models in Table 1 using Google Colab Pro. Julia aggregated the qualitative evaluation survey responses and created the figures and tables that display Seq2Seq architecture (Figure 2) and survey responses (Figures 7 to 11). Kate proposed the idea of creating a chatbot for three-way conversations, contributed her 6.864 Homework 3 code for our baseline Seq2Seq model, and created the Google Form to perform the qualitative evaluation of our models. Together, we chose the hyperparameters to fine-tune and analyzed the data to determine the chatbot model performed the best. All of us contributed to the discussion on the comparisons of each model's validation perplexity and predictions.

9. Acknowledgments

We would like to thank the course staff: Professor Jacob Andreas, Professor Jim Glass, Abby Bertics, Ekin Akyurek, Dylan Doblar, Wei Fang, Evan Hernandez, Pranav Krishna, Hongyin Luo, and Harini Suresh. Additionally, we are very thankful towards everyone who filled out our Google form for qualitative evaluation of chatbot dialogue examples.

References

[1] Zhang et. al. "Personalizing Dialogue Agents: I have a dog, do you have pets too?" https://arxiv.org/pdf/1801.07243.pdf 1, 2

[2] MIT 6.806/6.864 Dialogue Lecture Notes. 29 April 2021. 1

[3] ParlAI Tutorial. https://colab.research.google.com/drive/1PSUh7CRySCqo6JMfjkNndbzGydF84a5x?authuser=2#scrollTo=TGzF2mH185aj

[4] Liu et. al. "You Impress Me: Dialogue Generation via Mutual Persona Perception." https://arxiv.org/pdf/2004.05388.pdf 2

[5] Jurafsky and Martin. *Speech and Language Processing*, 2nd edition. Chapter 24 "Chatbots and Dialogue Systems."

[6] Serban et. al. "Building End-To-End Dialogue Systems Using Generative Hierarchical Neural Network Models." https://arxiv.org/pdf/1507.04808.pdf 2

[7] Sutskever et. al. "Sequence to Sequence Learning with Neural Networks." https://arxiv.org/pdf/1409.3215v3.pdf 3

[8] Annotated Encoder Decoder. Github. https://bastings.github.io/annotated_encoder_decoder/ 5

[9] Bahdanau, Cho, and Bengio. "Neural Machine Translation by Jointly Learning to Align and Translate." https://arxiv.org/pdf/1409.0473.pdf 3, 4

[10] Huyen, Chip. "Evaluation Metrics for Language Modeling." *The Gradient.* https://thegradient.pub/understanding-evaluation-metrics-for-language-models/ 4

[11] Liu, et. al. "Roberta: A robustly optimized bert pretraining approach." arXivpreprintarXiv:1907.11692 4

[12] Luong, et. al. "Effective Approaches to Attention-based Neural Machine Translation." https://arxiv.org/pdf/1508.04025.pdf 11