GENERATING A MENTAL HEALTH CHATBOT

Daniel Huynh, Eric Li & Sadhika Dhanasekar
Department of Computer Science
University of Virginia
Charlottesville, VA 22903, USA
{tap7ke,zzf5jx,qkx9an}@virginia.edu

ABSTRACT

In this project, we looked to fine-tune the LLAMA-2 large language model on CounselChat, a mental health-related dataset, creating a model that can provide therapeutic responses to users needing support. This chatbot will be able to talk to the user through a time of difficulty, similar to a counselor or therapist, and offer concrete advice. Our goal was to develop such a model that can provide high quality support and conversation, providing access to students and other demographics who may not have the resources to talk to a therapist but may be facing a hard situation, ensuring equity in mental health. Both the base LLAMA model and the chat version were fine-tuned with the use of Peft and QLora, resulting in chatbots that reflected the original dataset a fair bit. Additional modifications to the fine-tuning data allowed for more longer, more elaborate response structures. We hope that the model can be expanded on and eventually made accessible through the web.

1 INTRODUCTION

1.1 IMPORTANCE

We believe firstly that the impact this project can provide if performed successfully can be one of great consequences, opening mental health care to so many demographics so underrepresented in healthcare. As it is, mental health is a real crisis. Almost 20% of all adults in the United States suffer from a mental illness, a staggering number. Yet, so many people simply cannot find the care they deserve and so desperately need due to costs. Creating a free platform with such a model will be greatly beneficial, expanding important care to underserved communities, and continuing the fight against the stigma for mental health care, with the ability for privacy when searching for care (as you can simply go online anytime). This means that care can also become extremely accessible and can be served at any time.

In addition, working on such a model will be a great practice and experience. Being able to experience hands-on the concepts we learn from class will greatly improve our knowledge and experience in Natural Language Processing, and we will take away a great project to carry us into our future careers in machine learning and artificial intelligence research or in building and engineering new models in the industry.

1.2 OVERVIEW

Our plan is to take Meta's LLAMA-2 model and fine-tune it to respond to mental health prompts. We will use the Counselchat dataset (https://github.com/nbertagnolli/counsel-chat), which contains therapist responses to online mental health questions from the Counselchat website. Each question can have multiple answers from different therapists, meaning that there are diverse responses to the same prompt. The data also consists of the responses' upvotes, helping determine the quality of the advice.

To fine-tune LLAMA-2, we look to apply tokenization and LoRA before training, and train the model with various arguments to determine the best chatbot. Analyzing the results will prove difficult since there aren't clear metrics for determining the quality of a mental health response, but we do look to judge the model based on its training and validation losses. We also looked to compare our finetuned models to baseline models, which we established to be the simply base LLAMA-2 model without any finetuning performed upon it. This meant performing human evaluation upon the responses generated by the model. Depending on the generated outputs, we may also hope to tune the dataset a bit to adjust for different response formats.

In completing this experiment, we hope to also evaluate the effectiveness of finetuning the LLAMA-2 model, specifically using mental health-related conversations. This can specifically be accomplished by comparing the generated results across each model, on the same prompt.

Finally, we ultimately want to create a web application that encapsulates the mental health chatbot that we create, allowing the great reach that we hope to gain to users all over the country and the world, and allowing us to fully accomplish our mission. This application will provide a user interface for users to easily submit questions to the chatbot on the website interface, allowing it to mirror a familiar setting. However, we hope to also develop a more conversational chatbot in the future, allowing the user to continue conversations based on previous responses and context. This would require a text message-like conversational setting accomplished through a modified user interface, which would provide a familiar comfort to users as well.

2 Related Works

Various mental health chatbots have been attempted in the past. One study, also trained on Counselchat data, used DialoGPT to create a chatbot and found that it had a better perplexity than using LSTM and RNN. Various students then interacted with the chatbot and 63% preferred DialoGPT responses over the other model types. Another study created a chatbot known as SERMO using the Syn. Bot framework and OSCAVO to regulate emotions. SERMO was then tested with various volunteers who judged it based on defined usability metrics, and it was determined that SERMO had above average perspicuity and efficiency, but below average attractiveness and novelty. Overall, different chatbots have been created to address different mental health aspects, but we look to introduce a new chatbot based on LLAMA-2 to give out therapy advice specifically.

3 Problem Setup

When given a prompt by a user about a mental health-related problem, the fine-tuned chatbot should return a response similar to that of a professional therapist. The prompts will be able to vary in range, going from short questions to longer Reddit-style posts. The model needs to tokenize this prompt and feed it to a large language model to generate the response.

The goal is to create responses that not only sound professional, but also offer concrete advice to users on how to solve their problems. If possible, the model should elaborate on its advice as well, making it more understandable to users. As such, the dataset used in fine-tuning needs to reflect these goals and have sufficiently elaborate responses.

The fine-tuning itself needs to be done in an efficient, light-weight manner to allow for more accessibility, fit for all machines and devices. Due to the importance of ensuring quality responses, the fine-tuned model responses should closely resemble the original data.

4 Methodology

The base of the mental health chatbot consists of the LLAMA-2 large language model (LLM) from Meta. The 7B-hf versions of the standard and chat models were loaded via Hugging Face pre-trained. To reduce memory usage and speed up processing, 4 bit quantization was applied from the BitsAndBytes library on all models used.

Pre-trained tokenizers for LLAMA were also loaded from Hugging Face. A max-length padding tokenization was used to ensure equal length inputs, expanding smaller ones and truncating larger ones. To determine the max-length for padding, the data was first auto-tokenized and graphed to visualize the distribution. Cutoffs were then determined to balance performance without raising computational costs.

Finetuning was achieved using PEFT and QLora on 1000 steps with evaluations every 50. Utilizing PEFT allowed finetuning without changing the original parameters of the model. As such, new parameters could simply be loaded on top of an existing model, improving portability and convenience. QLora was utilized for its efficiency and low memory usage. Configuring QLora involves setting its alpha-value and its r-value, where alpha/r is the scale for the weight matrix. Many papers use an alpha-value of 16 and a r-value of 64 due to its good generalizability, but we implemented it with alpha-value of 64 and a r-value of 32 to put more emphasis on the data.

5 Experiment Setup

5.1 CounselChat Dataset

We chose the CounselChat dataset for our experiment. The dataset consists of a collection of conversations between licensed therapists with individuals seeking assistance. This data was scraped from www.counselchat.com, and contains 31 topics, from topics of ”depression” to ”military issues.” The responses also come from a wide range of locations around the United States as well as from a wide range of licensing levels - from Ph.D. psychologists to social workers and mental health counselors.

There are also ten columns provided as part of this dataset:

- questionID
- questionTitle (title of the question asked)
- questionText
- questionLink (URL to location of the question)
- topic
- therapistInfo (name and specialty of the therapist involved)
- therapistURL (link to therapist’s bio on counselchat)
- answerText
- upvotes (the number of upvotes received)
- split (whether the data was split for training, validation, or testing)

We filtered out entries that received no replies from experts, leading to a total pool of around 2200 mental-health requests. Each entry was then reformatted into:

Question: {prompt} \n Answer: {response}

where ”prompt” is the original question with the subject joined as the first sentence and ”response” is the most upvoted reply. Only the top reply was included to prevent overfitting on specific questions, as some questions had a lot more replies than others.

5.2 Tokenization

After reformatting the data, we applied an auto-tokenizer to the training data and plotted the lengths of the resulting tokenized inputs.

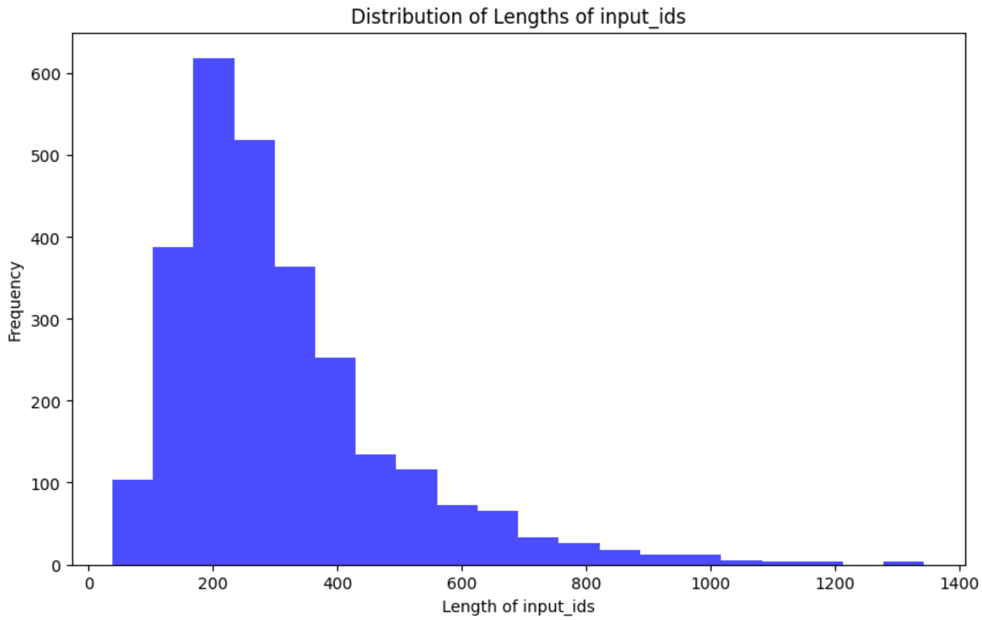

Figure 1: Intial tokenization of inputs

As shown in Figure 1, most of the inputs lie between 200 and 400 in length, but some extend all the way to over 1200. From this distribution, we decided to set the max length of each tokenized entry to be 600, truncating a few very long inputs. This way the model can still achieve high performance while balancing computational requirements.

5.3 Dataset Manipulation

While many of the responses within the CounselChat dataset consist of normal, paragraph answers, some of the advice came in the form of lists. These lists often outline multiple ways or procedures to solve a mental-health problem. Even though there weren't too many of them, it was found that they potentially could still have a very large impact on the generated responses from our chatbot. As such, the dataset was manipulated to soften the impact of lists, removing data points that contained lists in the "#)" format. This didn't remove all list responses (only about 20 total), as there were some with different formatting, but it removed enough to potentially affect the generated outputs. We then experimented with both the manipulated and original datasets. One base LLAMA 7B-hf model was fine-tuned on non-manipulated data, another base LLAMA 7B-hf model was fine-tuned after manipulation, and the LLAMA 7B-hf chat model was fine-tuned after manipulation.

5.4 Evaluation

At every evaluation step, we looked at both training and validation loss, which gives a good measure of accuracy and potential overfitting of the model. We also used human evaluation to determine the accuracy and relevance of the responses generated by the model as a result of user queries, addressing the limitations of the task at hand to evaluate correctness of the model. Generated responses were compared between the different LLAMA models, the different dataset versions, and the non-fine-tuned models.

6 Results

6.1 Baseline Models

Before fine-tuning, a baseline response was gathered from the base LLAMA-2 model and the chat version using a simple prompt about a user's poor mental wellness.

```
eval_prompt = "Question: I need help!: My mental wellness is terrible. I feel so depressed. Everyday I feel so sad. How do I not feel sad? \nAnswe
```

```
model_input = tokenizer(eval_prompt, return_tensors="pt").to("cuda")

model.eval()
with torch.no_grad():
    print(tokenizer.decode(model.generate(**model_input, max_new_tokens=256, pad_token_id=2)[0], skip_special_tokens=True))
```

```
A decoder-only architecture is being used, but right-padding was detected! For correct generation results, please set `padding_side='left'` when
initializing the tokenizer.
Question: I need help!: My mental wellness is terrible. I feel so depressed. Everyday I feel so sad. How do I not feel sad?
Answer: #  савезни репрезентативац
Soccer is a game that is played by two teams of eleven players each. The team that has the most goals at the end of the game wins.
The game is played on a rectangular field, called a pitch, with a goal at each end. The field is marked with a boundary line and a halfway line.
The ball is a round object, called a ball, and is made of leather or synthetic materials. The players use their feet, head, or chest to move the
ball around the field.
Soccer is a game that is played by two teams of eleven players each. The team that has the most goals at the end of the game wins. The game is pl
ayed on a rectangular field, called a pitch, with a goal at each end. The field is marked with a boundary line and a halfway line. The ball is a
round object, called a ball, and is made of leather or synthetic materials. The players use their feet, head, or chest to move the ball around th
e field.
Soccer is a game that is played by two teams of eleven players each. The team that has the most goals at the end of the
```

Figure 2: Generation of the base model's response to a user's prompt, asking for help.

As seen in Figure 2, the base model clearly fails to give a proper response, and does not know how to answer the prompt. While it functions to understand natural language, it struggles to understand what the task is asking it to do, and therefore, generates a response that is all over the place, first speaking in Russian, and then describing the game of soccer. This emphasizes the importance and need for the process of finetuning the model, aligning the model with enough training upon the task it will be utilized to perform upon. Through this process, it was anticipated that the model would better understand the task instructions, and carry out the task much more effectively.

```
model_input = tokenizer(eval_prompt, return_tensors="pt").to("cuda")

model.eval()
with torch.no_grad():
    print(tokenizer.decode(model.generate(**model_input, max_new_tokens=256, pad_token_id=2)[0], skip_special_tokens=True))
```

```
A decoder-only architecture is being used, but right-padding was detected! For correct generation results, please set `padding_sid
e='left'` when initializing the tokenizer.
Question: I need help!: My mental wellness is terrible. I feel so depressed. Everyday I feel so sad. How do I not feel sad?
Answer: # tough question. Sorry to hear that you're struggling with depression. It's important to know that you're not alone, and
there are many resources available to help you manage your mental health. Here are some suggestions that may help:

1. Seek professional help: Consider talking to a mental health professional, such as a therapist or counselor. They can help you i
dentify the underlying causes of your depression and develop strategies to manage your symptoms.
2. Practice self-care: Take care of yourself by getting enough sleep, eating a healthy diet, and engaging in activities that bring
you joy and relaxation. Exercise, meditation, and spending time in nature can also help improve your mood.
3. Connect with others: Reach out to friends, family, or a support group for help and encouragement. Talking to others who underst
and what you're going through can be very helpful.
4. Try therapy: Cognitive-behavioral therapy (CBT) is a type of therapy that can help you identify and change negative thought pat
terns that contribute to depression.
5. Consider medication: If your de
```

Figure 3: Generation of the chat model's response to a user's prompt, asking for help.

Shown above in Figure 3, the chat model performed very well without fine-tuning, which can be somewhat expected as it is already fine-tuned to respond to general prompts. However, the model struggles as it keeps generating and is only able to list a few pieces of advice.

6.2 Non-Manipulated Model

After fine-tuning the model, the training and validation loss at each evaluation step was plotted.

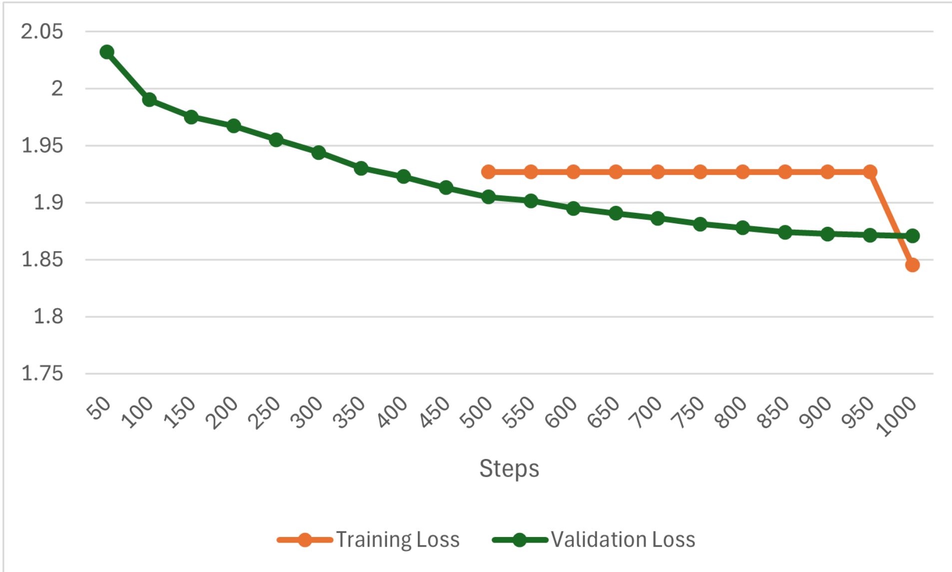

Figure 4: Training and validation loss of a base model on non-manipulated data.

The training loss wasn't able to be calculated before step 500, but it remained constant at 1.9269 afterwards before dipping to 1.8451 after step 1000. The validation loss started high but steadily declined and plateaued at around 1.87 as shown in Figure 4. Since training loss stayed above validation loss initially, it is suggested that little overfitting occurred during fine-tuning in the first half. Eventually, however, the training loss overtook validation, suggesting that the model eventually overfit. As such, we decided to use the model at evaluation step 500 as the main chatbot, where overfitting has yet started, but validation loss has decreased substantially.

```
eval_prompt = "### Question: How can I be less stressed?: I need help knowing how to deal with stress. What can I do? \n### Answer
model_input = tokenizer(eval_prompt, return_tensors="pt").to("cuda")

ft_model.eval()
with torch.no_grad():
    print(tokenizer.decode(ft_model.generate(**model_input, max_new_tokens=300)[0], skip_special_tokens=True))
```

```
### Question: How can I be less stressed?: I need help knowing how to deal with stress. What can I do?
### Answer: #1. Learn to relax. #2. Learn to take breaks. #3. Take breaks. #4. Learn to delegate. #5. Learn to say no. #6. Learn t
o be more efficient. #7. Learn to be more organized. #8. Learn to be more creative. #9. Learn to be more productive. #10. Learn to
be more positive. #11. Learn to be more compassionate. #12. Learn to be more grateful. #13. Learn to be more patient. #14. Learn t
o be more disciplined. #15. Learn to be more empathetic. #16. Learn to be more supportive. #17. Learn to be more forgiving. #18. L
earn to be more compassionate. #19. Learn to be more kind. #20. Learn to be more loving. #21. Learn to be more compassionate. #22.
Learn to be more grateful. #23. Learn to be more patient. #24. Learn to be more disciplined. #25. Learn to be more empathetic. #2
6. Learn to be more supportive. #27. Learn to be more forgiving. #
```

Figure 5: Generation of the base 7B-hf model's response to a user's prompt on original data.

As shown in Figure 5, the generated output of the fine-tuned model on a basic prompt about stress has been too heavily influenced by list responses in the dataset. The advice generated consists only of bullet points, and each advice is very short and unhelpful. This implies that data manipulation is necessary to control how much influence certain formats have over eventual responses.

6.3 BASE 7B-HF MODEL

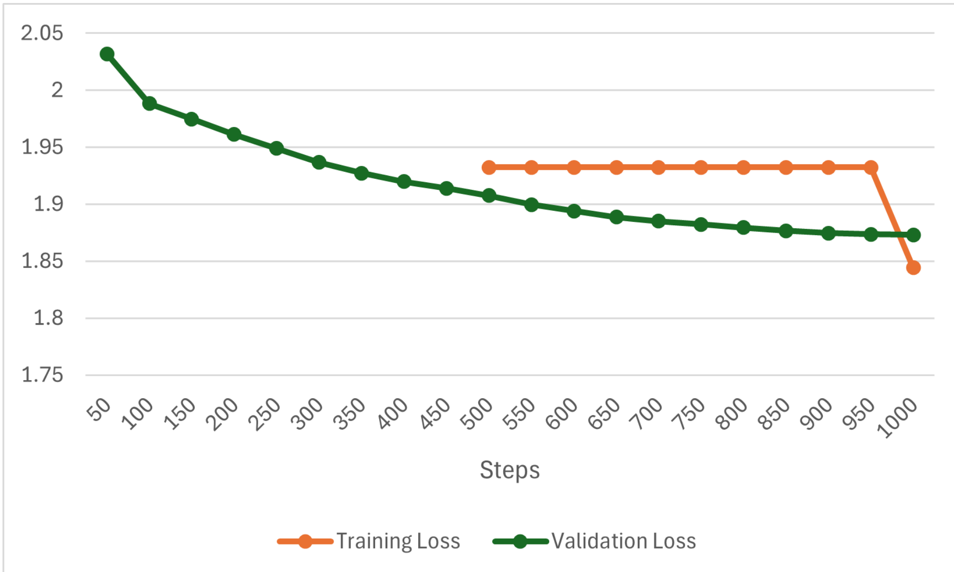

Figure 6: Training and validation loss of a base model on manipulated data.

The training and validation loss of the base 7b-hf model is remarkably similar on manipulated data as non-manipulated data. This pattern, shown in Figure 6, isn't very surprising as the manipulated data only differs by a few responses.

```
eval_prompt = "### Question: How can I be less stressed?: I need help knowing how to deal with stress. What can I do? \n### Answer: #"
model_input = tokenizer(eval_prompt, return_tensors="pt").to("cuda")

ft_model.eval()
with torch.no_grad():
    print(tokenizer.decode(ft_model.generate(**model_input, max_new_tokens=300)[0], skip_special_tokens=True))
```

```
### Question: How can I be less stressed?: I need help knowing how to deal with stress. What can I do?
### Answer: #1. Be more aware of your stress.  You can't do anything about it unless you are aware of it.  The first step to any change is awaren
ess.  #2. Be more aware of your triggers.  What are the things that make you more stressed?  Is it work?  Is it family?  Is it money?  Is it a ce
rtain person?  Once you identify your stressors, you can start to come up with ways to deal with them.  You may not be able to change your stress
ors, but you can change your reaction to them.  #3. Make a list of your stressors and your reactions to them.  Once you have this list, you can s
tart to brainstorm ways to change your reaction to your stressors.  You may not be able to change your stressors, but you can change your reactio
n to them.  #4. Come up with a plan.  Once you have your list of stressors and your reactions to them, come up with a plan.  What can you do to c
hange your reaction to these stressors?  #5. Start implementing your plan.  Once you have a plan, start implementing it.  Start small, and then g
radually add more.  The more you practice, the better you will get at managing your stress.  #6. Be patient.  It takes time to change your reacti
ons to stressors.
```

Figure 7: Generation of the finetuned model's response to a user's prompt, asking for help.

As seen clearly in Figure 7, the model is able to understand the task after the finetuning process on the manipulated Counselchat dataset, and is able to effectively respond to the prompt of the user, generating a list of advice for the user to help in the user's struggles indicated in their prompt. Compared to non-manipulated data, it is able to expand on its advice, making its response more elaborate and concrete. It also improves greatly from the non-fine-tuned model shown previously. Simply from this single comparison and observation, it is already extremely clear the benefits of the finetuning process on the LLAMA-2 model, and shows the ability of the model to learn and understand the task at hand.

6.4 Chat Model

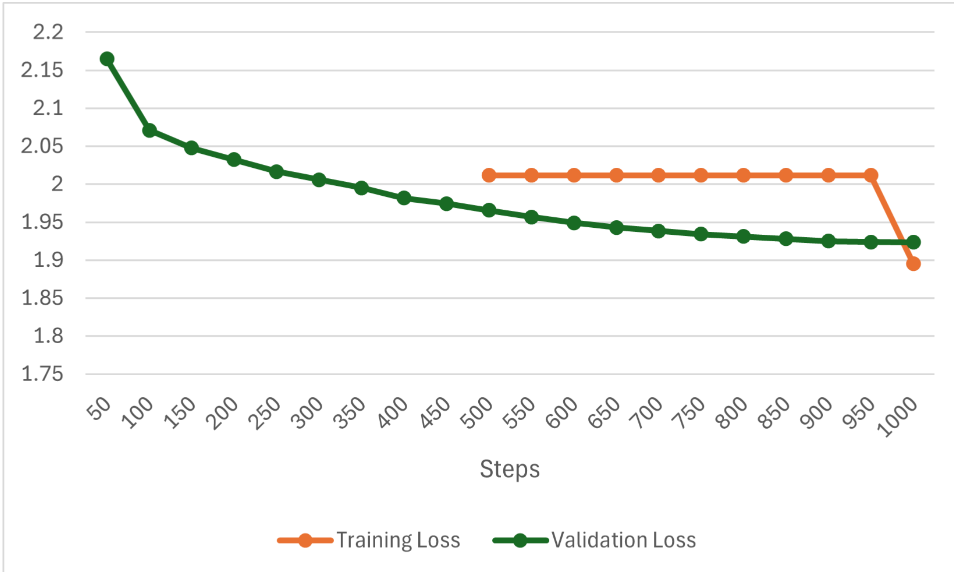

Figure 8: Training and validation loss of a chat model on manipulated data.

Shown in Figure 8, the chat model experiences a similar pattern of validation loss decreasing and training loss overtaking validation loss but with overall larger values. The losses start higher and plateau at 1.923561, which is greater than the 1.845100 value experienced by the base 7B-hf models.

```
eval_prompt = "### Question: How can I be less stressed?: I need help knowing how to deal with stress. What can I do? \n### Answer
model_input = tokenizer(eval_prompt, return_tensors="pt").to("cuda")

ft_model.eval()
with torch.no_grad():
    print(tokenizer.decode(ft_model.generate(**model_input, max_new_tokens=300)[0], skip_special_tokens=True))
```

```
### Question: How can I be less stressed?: I need help knowing how to deal with stress. What can I do?
### Answer: #1. Get a therapist! If you have the means, get a therapist to help you process your stress. #2. Try yoga. Yoga is a g
reat way to reduce stress and it's a great workout too! #3. Eat healthy. Try to eat healthy and avoid junk food. Try to eat at lea
st 5 servings of fruits and veggies every day. #4. Get enough sleep. Try to get at least 7-8 hours of sleep a night. #5. Take brea
ks. Take a break from work, school, or whatever it is that you're doing to give yourself some time to relax and recharge. #6. Try
meditation. Meditation is a great way to reduce stress. #7. Try to avoid caffeine and alcohol. Both of these can increase stress.
#8. Get some exercise. Exercise is a great way to reduce stress. #9. Try to do something you love. Do something that you enjoy doi
ng. This can help reduce stress. #10. Try to focus on the present. Try to focus on the present moment rather than worrying about t
he future or past. #11. Try to set realistic goals. Try to set realistic goals for yourself. This can help reduce stress. #12. Try
to avoid negative people. Try to avoid people that are negative and stressful.
```

Figure 9: Generation of the finetuned chat model's response to a user's prompt, asking for help.

Despite the larger loss however, the responses generated by the chat-model are very similar to that of the base-model. Figure 9 exemplifies this, as the output still follows the list format for advice that the base model follows.

7 Challenges and Solutions

One large challenge was the duration of the initial fine-tuning process for LLAMA-2. The large amount of data to process with such a big model resulted in a slow process. This was addressed though the use of 4-bit quantization and using a GPU. This served to greatly improve the speed and resource utilization efficiency to obtain the training process seen in Figures 2, 3, and 9.

Another challenge faced was the lack of a comprehensive evaluation of the performance of the model. The ambiguity of the task led to a difficulty creating a reward model for ”proper” responses as there is a large variety of appropriate responses given a prompt. The training data also supports this notion as it involves different forms of response data even when a prompt is similar in nature. In addition, the lack of large human interactions to test or actively rank responses limits the capabilities of model improvement. This was in part mitigated by our comparisons the base LLAMA-2 model, finetuned LLAMA-2 model, and finetuned LLAMA-2 chat model, giving noticeable and observable differences. However, even though this was evaluated through human evaluation, there was no ability to perform large-scale human evaluation upon an entire validation set of prompts due to the amount of time and effort required to evaluate large numbers of prompts.

Lastly, there are limitations in the severity of the task and the high risk of an incorrect response. A faulty response could severely impact the users of the model, and cause a negative feedback loop of continuously worse answers. Additionally, in the case that a user needs to be soothed or placated, the model does not give as comforting responses as desired for this situation. The dataset could benefit from modification to allow for more compassionate responses given the situation and testing on different prompts to determine the severity and tone of the response. It must be able to determine the emotion of the prompt and understand the right way to respond to such emotions, which adds an additional dimension to consider as part of a language model in this specific and high-risk domain. This would allow for the model to respond to a larger variety of incidents, even if it were to provide the same/similar advice.

8 CONCLUSION

The development and potential implementation of this model as a therapeutic tool could be crucial in creating accessible mental health support. Through a conversational interface, the chatbot created in this paper is able to understand prompts from the user and effectively give advice for their struggle. Fine-tuning of the initial LLAMA-2 model resulted in better query comprehension, clearer responses, and a greater interpretation of advice. Generated examples show promise in providing attentive care and directly answering user queries.

Future steps may involve having the chatbot provide more personalized care. This would involve building upon previous responses in a given session, and deploying potential long-term layers that capture relationships with previous queries. This might also require the chatbot to be trained with a larger variety of data or fine-tuned with the help of professional therapists. Another step toward future work would be the encapsulation of the chatbot in a web application to allow for greater accessibility and more comfort in using the application. This also serves to provide a familiar interface for users to interact with in the form of a traditional texting application. This potential for scalability, and the low cost of creation, can make models like these extremely valuable and timely for mental health services.

9 REFERENCES

R. Crasto, L. Dias, D. Miranda and D. Kayande, ”CareBot: A Mental Health ChatBot,” 2021 2nd International Conference for Emerging Technology (INCET), Belagavi, India, 2021, pp. 1-5, doi: 10.1109/INCET51464.2021.9456326.

K. Denecke, S. Vaaheesan and A. Arulnathan, ”A Mental Health Chatbot for Regulating Emotions (SERMO) - Concept and Usability Test,” in IEEE Transactions on Emerging Topics in Computing, vol. 9, no. 3, pp. 1170-1182, 1 July-Sept. 2021, doi: 10.1109/TETC.2020.2974478.