Dataset:
This is a Topical Chat dataset from Amazon! It consists of over 8000 conversations and over 184000 messages.

https://docs.google.com/spreadsheets/d/1dFdlvgmyXfN3SriVn5Byv_BNtyroICxdgrQKBzuMA1U/edit?usp=sharing

Within each message, there is a conversation id, which is basically which conversation the message takes place in. Each message is either the start of a conversation or a reply from the previous message. There is also a sentiment, which represents the emotion that the person who sent the message is feeling. There are 8 sentiments: Angry, Curious to Dive Deeper, Disguised, Fearful, Happy, Sad, and Surprised.

Tasks:
Your tasks in this assignment are:

a. **Sentiment Analysis:** Build a multi-class sentiment analysis model based on this dataset. Please report metrics for the model.
b. **Summarization:** Build a summarization model that can create a 2-3 sentence summary for each conversation. Produce summaries for the first 100 conversations and store them in a separate file.
c. **Question Answering:** Build a question-answering model that can answer questions based on this dataset. This can involve the following:
    a. Generate a sample set of questions from the data
    b. For each question, produce the answer(s). The answer should also include a reference to the relevant conversation(s) used to produce it.

For each of these tasks, you are free to use any pre-trained model/LLM for these tasks and/or finetune an LLM model as you see fit. Preference will be given to candidates who use open-source models. Bonus points for using suitable evaluation techniques to evaluate & compare the performance of the models.

You have 3 days to finish this assignment. You should do the development in Python, and you are strongly encouraged to submit the assignment as one or more Jupyter notebooks. You can submit your solution as a .zip or .tar file via email or share it via a Google Drive link.

If you are submitting the assignment as an IPython Notebook, your submission must meet the following requirements:
(1) Send the complete notebook that includes the outputs of running the code, as well as any additional output files.
(2) Document the steps
(3) Code formatting and readability are important