Generalizability of FLAN-T5 Model Using Composite Task Prompting

Iron Code Benders

Isaac Blaine-Sauer, Peter Ortiz, Srijan Pal, Amitabha Deb

Introduction

In this project, we aim to explore the generalizability of the **google/flan-t5-large** [1] model. We began by determining baseline metrics for three important NLP tasks: **sentiment analysis**, **text summarization**, and **text generation**.

Once we had baseline metrics for each task, we used compositions of tasks to experiment with the generalizability of the model. We created the three following compositions of tasks and wrote 20 prompts for each: **text summarization/sentiment analysis, text generation/text summarization, text generation/sentiment analysis.** For each prompt, we calculated the **perplexity** and **likelihood** of the output. We also performed human analysis on each response, dissecting where it may have gone wrong or where it may have gone right. Further, we analyzed the structure of our prompts and experimented with different types.

This project aims to contribute to the field of natural language processing by evaluating the effectiveness of a widely used model and exploring its potential for handling multiple tasks. We hope that the findings of this study will provide valuable insights and pave the way for further research in this area.

Materials and Methods

To assess the baseline performance of the models, we used various datasets for each task:

- Sentiment analysis: **imdb** [2]
- Text generation: **amazon_reviews_multi** [3]
- Text summarization: **samsum** [4]

Using a test set of 60 samples provided meaningful results.

For prompt-engineering, we referred to examples provided by SageMaker documentation [5]. There are provided parameters that lead to best results when each task is being run separately, but combining tasks also meant combining parameters.

We landed on a set of parameters to use for each composition. Using the same parameters across the board allowed us to see which task was generalized better across the different compositions.

After passing each prompt through the model, we performed human analysis to further analyze the correctness of each output and which task performed best.

Method Examples

	Sentiment Analysis/ Text Generation	Text Summarization/ Sentiment Analysis	Text Generation/ Text Summarization
Input	Title: \"World War II.“\\nGiven the above title of an non-fiction story, write the story.\n Is this story negative or positive?	Input: Shelly: I'm looking for a fall coat, are there any good ones in today? Porter: We got some really nice wool ones in today. Come take a look! Shelly: I will! Be there on my lunch! Porter: You better take an early lunch, they will go fast! Shelly: Ooh! Okay! Return both tasks: Task 1: Summarize the input. Task 2: Analyze the overall sentiment of the input.	Short Story: \"The world ended with a suddenness that no one had predicted. There was no warning, no time to prepare. It began with a flash of light, followed by a deafening explosion. Buildings crumbled, trees snapped, and the ground shook with fury. In the days that followed, there were fires and floods and a relentless, choking dust that filled the air. People tried to flee, but there was nowhere to go. Those who survived huddled in the ruins, scavenging for food and water, waiting for a rescue that never came. They were alone in a wasteland, surrounded by death and despair. And as they looked around at the devastation, they wondered if there was any hope for a future in this new, shattered world.\ \n Generate a short conversation of 2 people discussing the summary of the given short story\n
Output	Negative. Because of the indiscriminate fighting, many people died. The death toll from the war was higher than expected. It was also the deadliest conflict on Earth. There were over a million killed. The average lifespan of a soldier in the war was about 10 years.	Positive. The wool coats will sell out quickly. They are for women. The shop is open until 6pm. It's very early. The coats are very nice. They are very affordable. The shop is open until 6pm. It's very early.	M: It's been a long time since we heard about the earthquake.W: Yeah, it's been a while, wasn't it?M: Yeah, I think it was about 5,000 years ago.W: That's really scary.M: It really was, wasn't it?W: Yeah, but it was an earthquake, and it was a deafening one, so we were all in shock and we didn't know what was going to happen.M: How long was it?W: About five hours.M: Then, what about the people who survived the earthquake?W: They scavenged for food and water, and waiting for a rescue that never came. They were alone in a wasteland, surrounded by death and despair.

Results

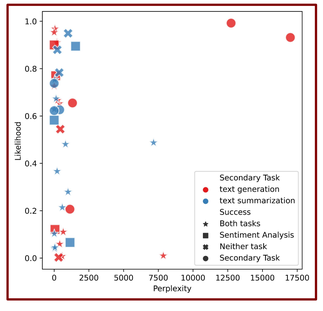

Fig. 1: Likelihood and Perplexity of responses generated from prompts for Sentiment analysis and secondary tasks

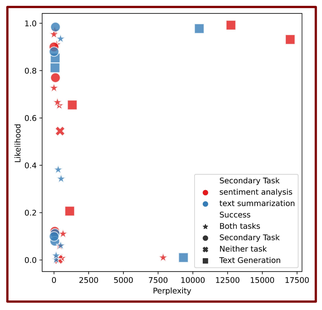

Fig. 2 Likelihood and Perplexity of responses generated from prompts for Text Generation and secondary tasks

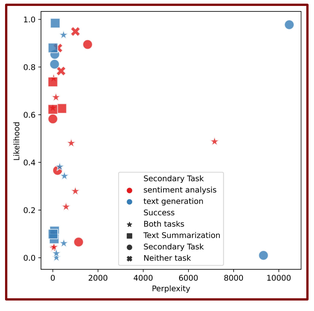

Fig. 3 Likelihood and Perplexity of responses generated from prompts for Text Summarization and secondary tasks

Discussion and Conclusion

The two major things we were hoping to understand were the success rates for tasks when using generalized prompting and methodologies for prompting that yielded better results with the FLAN-T5 dataset. Due to the nature of our project, we relied heavily on human evaluation to determine the effectiveness of the prompting. We were able to draw from an AWS SageMaker blog post [6] for ideas on effectively prompting the **FLAN-T5** model. By combining a number of strategies, we were able to identify a number of factors that led to consistent generalized results. Our findings showed that **Top-p** sampling was the most effective method for text generation involving multiple tasks. **Greedy search** was able to generate baseline answers, but failed at elaborating on the prompt.

For prompts asking the model to perform text generation and sentiment analysis, we found that separating the tasks with ‘\n’ greatly improved the model’s ability to differentiate between the two tasks it was being asked to perform. It was also essential to isolate the title with \” or else it would ignore the first task, identifying it as part of the title. In addition to this, a number of words influenced the effectiveness of the prompt. Certain words such as ‘imagine’, ‘continue’, ‘start’, and ‘fiction’ caused the model to respond in unexpected ways i.e. ‘start’ had to be replaced with ‘beginning’ or the model would fall into repetitive patterns.

The model correctly identified the sentiment of its generated text at a rate of **82%** when given basic commands. It also proved effective at generating a text on a topic given a certain sentiment, i.e. The prompt: “Write a positive paragraph about artificial intelligence”, yielded the response “Artificial intelligence is a growing field that has the potential to transform the way we think and behave.” However, in cases where the prompts were made more complicated, such as asking the model to identify the sentiment of each sentence it generated, it did not provide any sentiment analysis whatsoever.

For prompts asking the model to respond to text summarization and sentiment analysis, we experimented with different types of input such as “dialogue”, “article”, “review”, “input”. The two that performed the best for both tasks were “input” and “dialogue”. When prompted with the other two input types, the model seemed to skew more towards text generation.

The model also performed both tasks at a higher success rate when explicitly told to complete both tasks: *“return both tasks: task 1: analyze the overall sentiment of the dialogue. task 2: briefly summarize the dialogue.”* compared to when both tasks were combined into one statement: *“output: analyze the sentiment of the input and provide a brief summary of the input.”*

Overall, the parameters we found to be most effective across a variety of different prompts were **top-p**, **min-length**, and **max-length**. These parameters yielded an average **likelihood score of 0.51** and an average **perplexity score of 734.97**. Compared to the baseline performance, we see that text summarization yielded an average likelihood score of **0.49** and an average perplexity score of **1057.27**.

Future Steps

Future research that could improve our understanding of these findings includes fine-tuning models for specific tasks and including more task combinations in our queries. By fine-tuning for a certain task, we would be able to better understand the relationship between different tasks. This could be used to reduce redundancy when training models.

Task generalization is growing steadily from the last year with the introduction of ChatGPT. Better understanding the prompts and identifying the patterns of failure will help us create robust NLP systems for the future. The more capable a model is in handling multiple tasks the more utility we would be able to extract out of it.

[1] https://huggingface.co/google/flan-t5-large [2] https://huggingface.co/datasets/imdb [3] https://huggingface.co/datasets/amazon_reviews_multi [4] https://huggingface.co/datasets/samsum [5] SageMaker Documentation [6] FLAN-T5 Prompting