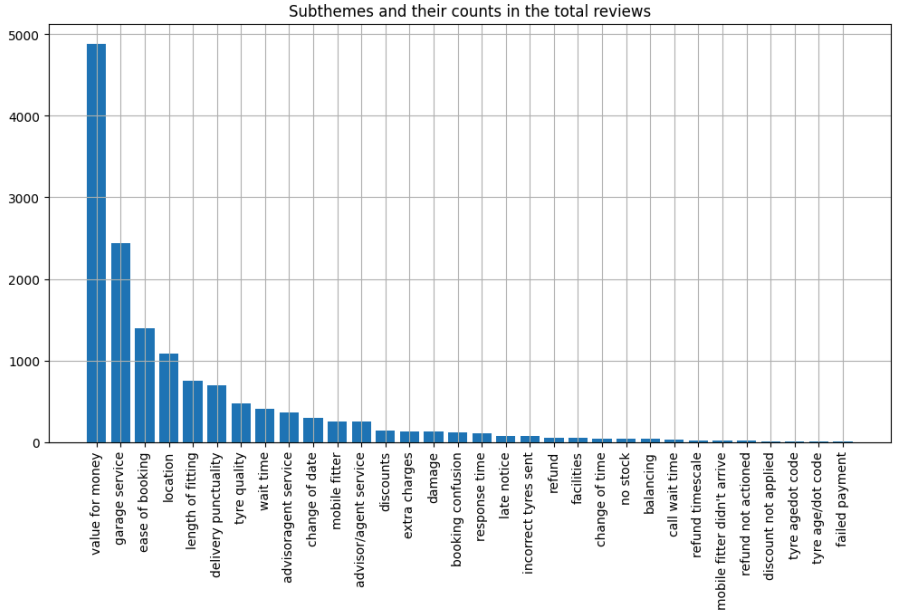
Task - Develop an approach that given a sample will identify the subthemes along with their respective sentiments.

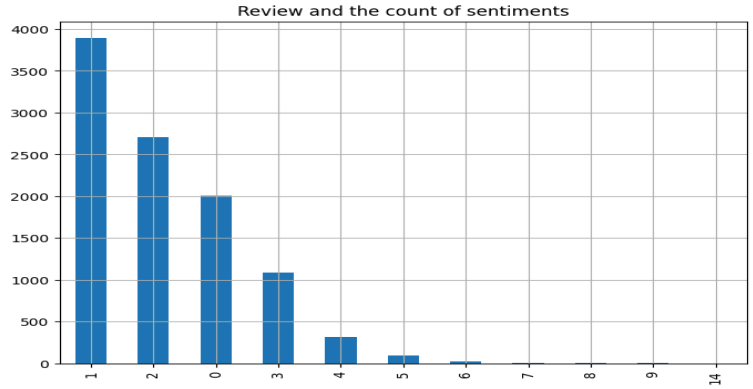
Approach - I have compiled all subthemes and their sentiments from each review into a single dataset, where the first column holds the review text and the second column filters the rows based on positive or negative sentiment. This arrangement is intended for further analysis.
I have worked with traditional machine learning algorithms and on fine-tuning LLMs for the task.

	Review	Sentiment
0	Tires where delivered to the garage of my choi...	[garage service positive, ease of booking posi...
1	Easy Tyre Selection Process, Competitive Prici...	[garage service positive, value for money posi...
2	Very easy to use and good value for money.	[value for money positive]
3	Really easy and convenient to arrange	[ease of booking positive]
4	It was so easy to select tyre sizes and arrang...	[location positive, value for money positive, ...

I created a plot that displays the frequency of each subtheme mentioned across the reviews.



I plotted the count of each subtheme mentioned in reviews.



I have developed a function that constructs individual classifiers for each subtheme. I have utilized this function for 'value for money', 'garage service', and 'ease of booking', using their respective data rows. This function requires user input and outputs the sentiment associated with these subthemes.

```
Enter the review: Garage was good and it was worth the money.
['garage service positive']
['ease of booking positive']
['value for money positive']
```

I have prepared a dataset tailored for fine-tuning a gpt-neo-125M question-answering model. In this dataset, each review is formatted as a question, with the corresponding subtheme and its sentiment provided as the answer.

```
[Q] Identify the sentiment in the given sentence- Tires where delivered to the garage of my choice,the garage notified me when they had
been delivered. A day and time was arranged with the garage and I went and had them fitted,a Hassel free experience.
[A] ['garage service positive', 'ease of booking positive']
[Q] Identify the sentiment in the given sentence- Easy Tyre Selection Process, Competitive Pricing and Excellent Fitting Service
[A] ['garage service positive', 'value for money positive']
```

I have taken the first 100 rows of the dataset and prepared it for fine-tuning the Llama2 question-answering pipeline.

```
df['text'] = '<s>[INST] ' + df['Review'] + ' [/INST] ' + df['Sentiment'] + ' </s>'
dataset = Dataset.from_pandas(df[['text']])
dataset
```

```
Dataset({
    features: ['text'],
    num_rows: 100
})
```

Result -
After fine-tuning the GPT-Neo model on the entire dataset for three epochs, the metrics are as follows:

[1563/1563 08:25, Epoch 3/3]

Step	Training Loss
500	1.960500
1000	1.723400
1500	1.628000

I have tested this locally for evaluation purposes.

```
sequence1 = "Identify the sentiment in the given sentence- User friendly Website. Competitive Prices. Good communications. Efficient service by ATS Euromaster."
answer = final_answer(model_path, sequence1, max_len)
print(answer)
```

```
Identify the sentiment in the given sentence- User friendly Website. Competitive Prices. Good communications. Efficient service by ATS Euromaster. Good customer service.
[A] ['value for money positive']
```

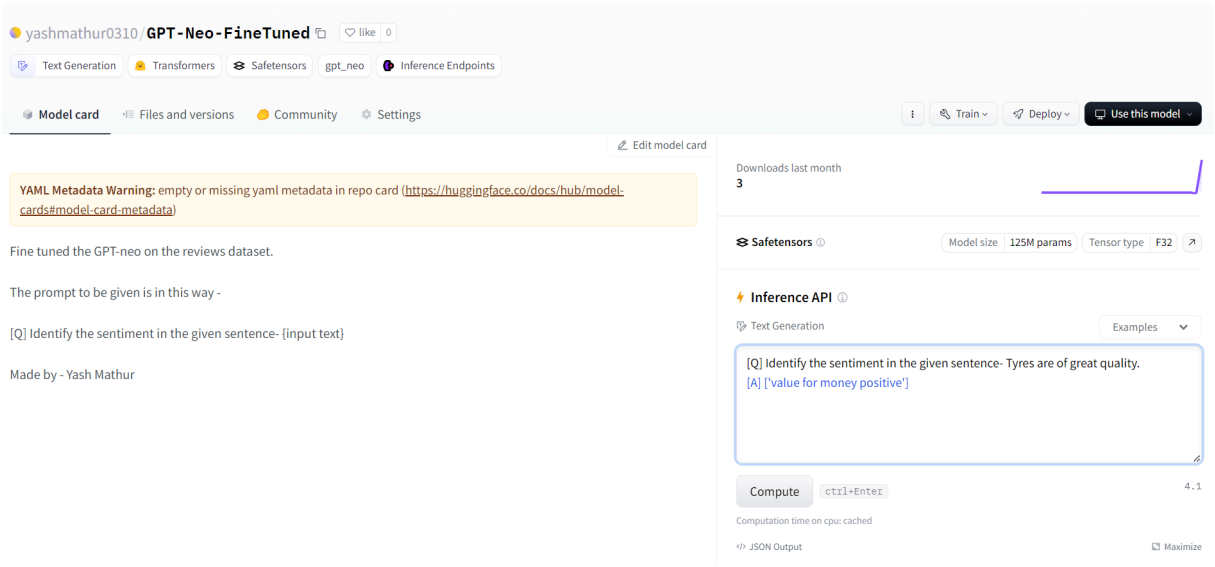
```
sequence1 = "Identify the sentiment in the given sentence- I had to wait three hours at the garage because they gave priority to customers with quicker jobs"
answer = final_answer(model_path, sequence1, max_len)
print(answer)
```

```
Identify the sentiment in the given sentence- I had to wait three hours at the garage because they gave priority to customers with quicker jobs.
[A] ['wait time negative']
```

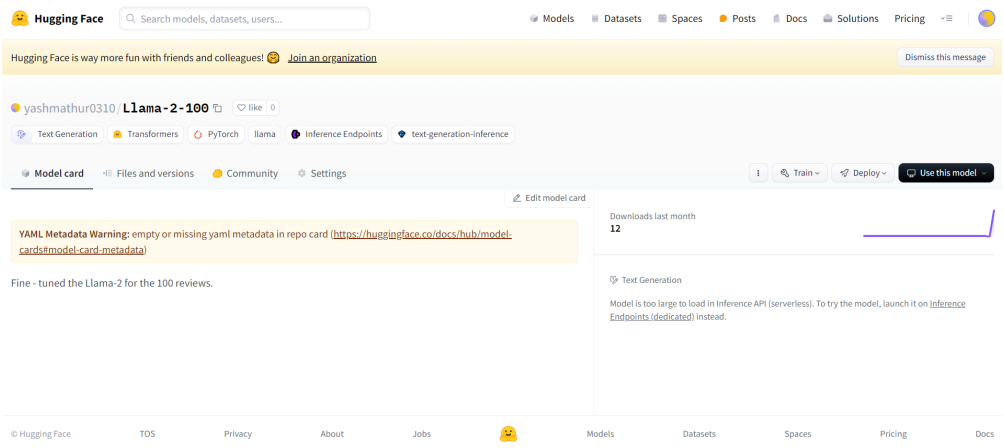
```
sequence1 = "Identify the sentiment in the given sentence-The company refunded my money in three days."
answer = final_answer(model_path, sequence1, max_len)
print(answer)
```

```
Identify the sentiment in the given sentence- The company refunded my money in three days.
[A] ['refund timescale positive', 'value for money positive']
```

Model deployment link - https://huggingface.co/yashmathur0310/GPT-Neo-FineTuned
The model is not very accurate because it was trained for just 3 epochs.



Model deployment link - https://huggingface.co/yashmathur0310/Llama-2-100
Fine-tuned the Llama-2 model for 100 reviews.



Improvements -
The binary classifier currently categorizes responses as positive or negative, even when the review is unrelated to the subtheme. Introducing a third 'neutral' category could address this issue by accurately classifying irrelevant reviews.
Utilizing a GPT-Neo model with more parameters and training it for additional epochs will yield better results.