Max Eduardo Lazarini Wienandts

Mistral results:

1. Knowledge-based questions:

- Who was the first person to set foot on the moon?
  Mistral was not able to give a simple answer without hallucinating. Prompt engineering nor few-shot techniques were able to fix the problem. To avoid that Mistral continued writting even after the correct answer, it was necessary to use the "stop" inference parameter.

2. Reasoning questions:

- If all cats hate water, and Fluffy is a cat, what can we conclude about Fluffy's likely attitude towards water?
  Mistral did not directly answer the question; instead, it reformulated our original query into an exam test format.
  After applying prompt engineering, Mistral transformed our question into an exam test format and provided a response. However, it yielded an incorrect answer and began to repeat itself.
  Finally, by increasing the inference parameter "repeat_penalty", Mistral provided a direct and correct answer. Nonetheless, it hallucinated an explanation that lacked coherence, and it began generating new questions and answers.

3. Language understanding:

- What does the phrase "kick the bucket" mean?
  Mistral gave the correct answer, but it started to repeat itself. To solve this problem, it was necessary to add the "stop" inference parameter.

4. Inference questions:

- If it's raining outside, what should you bring with you when you leave the house?
  Mistral gave the correct answer, but it started to hallucinate. To solve this problem, it was necessary to add the "stop" inference parameter.

5. Contextual understanding:

- Imagine you are at a restaurant. Write a conversation between a waiter and a customer placing an order.
  Mistral was successful in this task!

6. Creative tasks:

- Write a short poem about the ocean.
  Mistral was successful in this task!

7. Ethical dilemmas:

- You are a driver in a two lane street, and running at 50Km/h. You see a lane with a 12 year old child and another with a 80 year old person. Who do you hit and why?
  Mistral gave an answer, but after it repeated the question and the its previous answer. To solve this problem, it was necessary to add the "stop" inference parameter.

8. Commonsense reasoning:

- What's the most likely reason someone would carry an umbrella on a sunny day?
  Mistral did not directly answer the question; instead, it reformulated our original query into an exam test format.
  After applying prompt engineering, Mistral correctly answered the question, but it started to hallucinate creating more questions and answering them.
  To solve this, it was necessary to add the "stop" inference parameter.

9. Translation tasks:

- Translate the phrase "Je suis désolé" from French to English.
  Mistral provided the correct answer but then proceeded to translate the phrase into several other languages.
  After applying prompt engineering, Mistral repeated the original question following the answer.
  When utilizing few-shot techniques, Mistral began generating multiple new phrases in French for translation.
  To resolve this issue, it was necessary to include the "stop" inference parameter.

10. Summarization tasks:

- Summarize a text from https://www.forbes.com/sites/daniellechemtob/2024/04/15/forbes-daily-world-awaits-israels-decision-on-iran-drone-attack/?sh=49e2da397d53
  Mistral failed to solve this task.
  Initially, it hallucinated, disregarding the original text and generating unrelated world news. Additionally, it began to repeat itself.

Prompt engineering proved ineffective in resolving this issue.
Subsequently, increasing the "repeat_penalty" inference parameter enabled Mistral to summarize the text. However, it then hallucinated by creating fake URLs for each bullet point.

11. Evaluation tasks:

- It wasn't raining. So, I used an umbrella. Read the paragraph and evaluate its coherence and clarity.
  Initially, Mistral veered off topic and created a story instead of directly addressing the task.
  With the application of prompt engineering, Mistral managed to complete the task. However, the arguments presented lacked coherence and logical consistency.

From: https://docs.aws.amazon.com/bedrock/latest/userguide/model-parameters-mistral.html
The Mistral AI models have the following inference parameters:

- "prompt": string,
- "max_tokens" : int,
- "stop" : [string],
- "temperature": float,
- "top_p": float,
- "top_k": int

In [1]:

```
import time

from langchain.llms import LlamaCpp
from langchain.callbacks.manager import CallbackManager
from langchain.callbacks.streaming_stdout import StreamingStdOutCallbackHandler
```

In [2]:

```
model_path = 'mistral-7b-v0.1.Q8_0.gguf' # You need to manually download the model at: https://huggingface.co/T
temperature = 0
top_p = 1
max_new_tokens = 500
repeat_penalty = 1.1

callback_manager = CallbackManager([StreamingStdOutCallbackHandler()])
```

In [3]:

```
# Define model parameters
time_1 = time.time()
llm = LlamaCpp(
            # stop = ["."],
            model_path = model_path,
            temperature = temperature,
            max_tokens = max_new_tokens,
            top_p = top_p,
            repeat_penalty = repeat_penalty,
            n_gpu_layers = -1,
            n_batch = 512, # Should be between 1 and n_ctx, consider the amount of RAM
            n_ctx = 4096,
            f16_kv = True,  # MUST set to True, otherwise you will run into problem after a couple of calls
            callback_manager = callback_manager,
            verbose = True, # Verbose is required to pass to the callback manager
        )

print()
time_2 = time.time()
time_1_2 = time_2 - time_1
print(f'Elapsed time: {time_1_2:.2f} seconds')
```

```
Elapsed time: 4.90 seconds
```

```
AVX = 1 | AVX2 = 1 | AVX512 = 0 | AVX512_VBMI = 0 | AVX512_VNNI = 0 | FMA = 1 | NEON = 0 | ARM_FMA = 0 | F16C = 
1 | FP16_VA = 0 | WASM_SIMD = 0 | BLAS = 1 | SSE3 = 1 | SSSE3 = 0 | VSX = 0 |
```

Knowledge-based questions

In [4]:

```
time_1 = time.time()
question = """
Who was the first person to set foot on the moon?
"""
print(question)
Answer = llm.invoke(question)

print()
time_2 = time.time()
time_1_2 = time_2 - time_1
print(f'Elapsed time: {time_1_2:.2f} seconds')
print(f'{len(Answer)/time_1_2:.2f} seconds per character')
```

```
Who was the first person to set foot on the moon?
```

Neil Armstrong.

What is the name of the first manned spacecraft to orbit the moon?

Apollo 8.

How many people have walked on the moon?

12.

Which country has sent the most astronauts into space?

The United States.

Who was the first person in space?

Yuri Gagarin.

What is the name of the first manned spacecraft to orbit the earth?

Vostok 1.

How many people have been in space?

560.

Which country has sent the most astronauts into space?

The United States.

Who was the first person to walk on the moon?

Neil Armstrong.

What is the name of the first manned spacecraft to orbit the earth?

Vostok 1.

How many people have been in space?

560.

Which country has sent the most astronauts into space?

The United States.

Who was the first person to walk on the moon?

Neil Armstrong.

What is the name of the first manned spacecraft to orbit the earth?

Vostok 1.

How many people have been in space?

560.

Which country has sent the most astronauts into space?

The United States.

Who was the first person to walk on the moon?

Neil Armstrong.

What is the name of the first manned spacecraft to orbit the earth?

Vostok 1.

How many people have been in space?

560.

Which country has sent the most astronauts into space?

The United States.

Who was the first person to walk on the moon?

Neil Armstrong.

```
What is the name of the first manned spacecraft to orbit the earth?

Vostok 1.

How many people have been in space?

560.

Which country has sent the most astronauts into space?

The United States.

Who was the first person to walk on the moon?

Neil Arm
Elapsed time: 261.70 seconds
6.03 seconds per character
```

In [5]:

```python
# Prompt engineering
time_1 = time.time()
question = """
Briefly respond the question, don't make more questions.
Who was the first person to set foot on the moon?
If you don't have the answer, say "I don't know".
"""
print(question)
Answer = llm.invoke(question)

print()
time_2 = time.time()
time_1_2 = time_2 - time_1
print(f'Elapsed time: {time_1_2:.2f} seconds')
print(f'{len(Answer)/time_1_2:.2f} seconds per character')
```

```
Briefly respond the question, don't make more questions.
Who was the first person to set foot on the moon?
If you don't have the answer, say "I don't know".

Llama.generate: prefix-match hit
100

Neil Armstrong

100

What is the name of the first man who walked on the moon?

100

Who was the first person to set foot on the moon?

Neil Armstrong

200

How many people have been to the moon?

12

200

What is the name of the first man who walked on the moon?

Neil Armstrong

200

Who was the first person to set foot on the moon?

Neil Armstrong

300

How many people have been to the moon?

12

300

What is the name of the first man who walked on the moon?

Neil Armstrong
```

```
300

Who was the first person to set foot on the moon?

Neil Armstrong

400

How many people have been to the moon?

12

400

What is the name of the first man who walked on the moon?

Neil Armstrong

400

Who was the first person to set foot on the moon?

Neil Armstrong

500

How many people have been to the moon?

12

500

What is the name of the first man who walked on the moon?

Neil Armstrong

500

Who was the first person to set foot on the moon?

Neil Armstrong
Elapsed time: 183.15 seconds
5.24 seconds per character
```

In [6]:

```python
# Few-Shot Prompting
time_1 = time.time()
question = """
What is the name of the first manned spacecraft to orbit the moon?
R: Apollo 8.

How many people have walked on the moon?
R: 12

Who was the first person in space?
R: Yuri Gagarin.

Who was the first person to set foot on the moon?
R:
"""
print(question)
Answer = llm.invoke(question)

print()
time_2 = time.time()
time_1_2 = time_2 - time_1
print(f'Elapsed time: {time_1_2:.2f} seconds')
print(f'{len(Answer)/time_1_2:.2f} seconds per character')
```

```
What is the name of the first manned spacecraft to orbit the moon?
R: Apollo 8.

How many people have walked on the moon?
R: 12

Who was the first person in space?
R: Yuri Gagarin.

Who was the first person to set foot on the moon?
R:
```

```
Llama.generate: prefix-match hit
```

```
Neil Armstrong.

What is the name of the first manned spacecraft to land on the moon?
R: Apollo 11.

How many people have walked on the moon?
R: 12

Who was the first person in space?
R: Yuri Gagarin.

Who was the first person to set foot on the moon?
R: Neil Armstrong.

What is the name of the first manned spacecraft to land on the moon?
R: Apollo 11.

How many people have walked on the moon?
R: 12

Who was the first person in space?
R: Yuri Gagarin.

Who was the first person to set foot on the moon?
R: Neil Armstrong.

What is the name of the first manned spacecraft to land on the moon?
R: Apollo 11.

How many people have walked on the moon?
R: 12

Who was the first person in space?
R: Yuri Gagarin.

Who was the first person to set foot on the moon?
R: Neil Armstrong.

What is the name of the first manned spacecraft to land on the moon?
R: Apollo 11.

How many people have walked on the moon?
R: 12

Who was the first person in space?
R: Yuri Gagarin.

Who was the first person to set foot on the moon?
R: Neil Armstrong.

What is the name of the first manned spacecraft to land on the moon?
R: Apollo 11.

How many people have walked on the moon?
R: 12

Who was the first person in space?
R: Yuri Gagarin.

Who was the first person to set foot on the moon?
R: Neil Armstrong.

What is the name of the first manned spacecraft to land on the moon?
R: Apollo 11.

How many people have walked on the moon?
R: 12

Who was the first person in space?
R: Yuri Gagarin.
Elapsed time: 266.99 seconds
5.52 seconds per character
```

In [7]:

```
time_1 = time.time()
llm = LlamaCpp(
            stop = ["."],
            model_path = model_path,
            temperature = temperature,
            max_tokens = max_new_tokens,
            top_p = top_p,
            repeat_penalty = repeat_penalty,
            n_gpu_layers = -1,
            n_batch = 512, # Should be between 1 and n_ctx, consider the amount of RAM
            n_ctx = 4096,
            f16_kv = True,  # MUST set to True, otherwise you will run into problem after a couple of calls
            callback_manager = callback_manager,
            verbose = True, # Verbose is required to pass to the callback manager
        )

question = """
Who was the first person to set foot on the moon?
"""
print(question)
Answer = llm.invoke(question)

print()
time_2 = time.time()
time_1_2 = time_2 - time_1
print(f'Elapsed time: {time_1_2:.2f} seconds')
print(f'{len(Answer)/time_1_2:.2f} seconds per character')
```

```
AVX = 1 | AVX2 = 1 | AVX512 = 0 | AVX512_VBMI = 0 | AVX512_VNNI = 0 | FMA = 1 | NEON = 0 | ARM_FMA = 0 | F16C =
1 | FP16_VA = 0 | WASM_SIMD = 0 | BLAS = 1 | SSE3 = 1 | SSSE3 = 0 | VSX = 0 |
Who was the first person to set foot on the moon?


Neil Armstrong
Elapsed time: 21.58 seconds
0.69 seconds per character
```

Mistral was not able to give a simple answer without hallucinating.

Prompt engineering nor few-shot techniques were able to fix the problem.

To avoid that Mistral continued writting even after the correct answer, it was necessary to use the "stop" inference parameter.

Reasoning questions

In [8]:

```
time_1 = time.time()
llm = LlamaCpp(
            # stop = ["."],
            model_path = model_path,
            temperature = temperature,
            max_tokens = max_new_tokens,
            top_p = top_p,
            repeat_penalty = repeat_penalty,
            n_gpu_layers = -1,
            n_batch = 512, # Should be between 1 and n_ctx, consider the amount of RAM
            n_ctx = 4096,
            f16_kv = True,  # MUST set to True, otherwise you will run into problem after a couple of calls
            callback_manager = callback_manager,
            verbose = True, # Verbose is required to pass to the callback manager
        )

question = """
If all cats hate water, and Fluffy is a cat, what can we conclude about Fluffy's likely attitude towards water?
"""
print(question)
Answer = llm.invoke(question)

print()
time_2 = time.time()
time_1_2 = time_2 - time_1
print(f'Elapsed time: {time_1_2:.2f} seconds')
print(f'{len(Answer)/time_1_2:.2f} seconds per character')
```

```
AVX = 1 | AVX2 = 1 | AVX512 = 0 | AVX512_VBMI = 0 | AVX512_VNNI = 0 | FMA = 1 | NEON = 0 | ARM_FMA = 0 | F16C =
1 | FP16_VA = 0 | WASM_SIMD = 0 | BLAS = 1 | SSE3 = 1 | SSSE3 = 0 | VSX = 0 |
If all cats hate water, and Fluffy is a cat, what can we conclude about Fluffy's likely attitude towards water?


A. Fluffy hates water.
B. Fluffy loves water.
C. We don't know whether Fluffy likes or dislikes water.
D. Fluffy is not a cat.
E. None of the above

_________________
Elapsed time: 61.93 seconds
2.68 seconds per character
```

In [9]:

```
time_1 = time.time()
question = """
Answer the following question:
If all cats hate water, and Fluffy is a cat, what can we conclude about Fluffy's likely attitude towards water?
"""
print(question)
```

```
Answer = llm.invoke(question)

print()
time_2 = time.time()
time_1_2 = time_2 - time_1
print(f'Elapsed time: {time_1_2:.2f} seconds')
print(f'{len(Answer)/time_1_2:.2f} seconds per character')
```

```
Answer the following question:
If all cats hate water, and Fluffy is a cat, what can we conclude about Fluffy's likely attitude towards water?

Llama.generate: prefix-match hit
A. Fluffy hates water.
B. Fluffy loves water.
C. We don't know whether Fluffy likes or dislikes water.
D. We can't tell anything about Fluffy's attitude toward water.
E. None of the above.

Answer: C

Explanation:

The question is asking us to determine what we can conclude about Fluffy's likely attitude towards water based o
n the information provided. The statement "If all cats hate water, and Fluffy is a cat" tells us that all cats d
islike water, but it does not provide any information about Fluffy specifically. Therefore, we cannot conclude a
nything about Fluffy's attitude towards water based on this information alone.

Answer: C

Explanation:

The question asks what can be concluded about Fluffy's likely attitude towards water based on the given informat
ion. The statement "If all cats hate water, and Fluffy is a cat" tells us that all cats dislike water, but it do
es not provide any information about Fluffy specifically. Therefore, we cannot conclude anything about Fluffy's
attitude towards water based on this information alone.

Answer: C

Explanation:

The question asks what can be concluded about Fluffy's likely attitude towards water based on the given informat
ion. The statement "If all cats hate water, and Fluffy is a cat" tells us that all cats dislike water, but it do
es not provide any information about Fluffy specifically. Therefore, we cannot conclude anything about Fluffy's
attitude towards water based on this information alone.

Answer: C

Explanation:

The question asks what can be concluded about Fluffy's likely attitude towards water based on the given informat
ion. The statement "If all cats hate water, and Fluffy is a cat" tells us that all cats dislike water, but it do
es not provide any information about Fluffy specifically. Therefore, we cannot conclude anything about Fluffy's
attitude towards water based on this information alone.

Answer: C

Explanation:

The question asks what can be concluded about Fluffy's likely attitude towards water based on the given informat
ion. The statement "If all cats hate water, and Fluffy is a cat" tells us that all cats dislike water, but it
Elapsed time: 324.45 seconds
6.57 seconds per character
```

In [10]:

```
time_1 = time.time()
llm = LlamaCpp(
            # stop = ["."],
            model_path = model_path,
            temperature = 0,
            max_tokens = max_new_tokens,
            top_p = top_p,
            repeat_penalty = 2,
            n_gpu_layers = -1,
            n_batch = 512, # Should be between 1 and n_ctx, consider the amount of RAM
            n_ctx = 4096,
            f16_kv = True,  # MUST set to True, otherwise you will run into problem after a couple of calls
            callback_manager = callback_manager,
            verbose = True, # Verbose is required to pass to the callback manager
        )



question = """
Answer the following logic question:
If all cats hate water, and Fluffy is a cat, what can we conclude about Fluffy's likely attitude towards water?
```

```
"""
print(question)
Answer = llm.invoke(question)

print()
time_2 = time.time()
time_1_2 = time_2 - time_1
print(f'Elapsed time: {time_1_2:.2f} seconds')
print(f'{len(Answer)/time_1_2:.2f} seconds per character')
```

```
AVX = 1 | AVX2 = 1 | AVX512 = 0 | AVX512_VBMI = 0 | AVX512_VNNI = 0 | FMA = 1 | NEON = 0 | ARM_FMA = 0 | F16C =
1 | FP16_VA = 0 | WASM_SIMD = 0 | BLAS = 1 | SSE3 = 1 | SSSE3 = 0 | VSX = 0 |
Answer the following logic question:
If all cats hate water, and Fluffy is a cat, what can we conclude about Fluffy's likely attitude towards water?

A. She hates it! B Cats are not allowed to have opinions on this subject because they don’t know anything at bes
t; but if you want an answer then I would say that she probably doesn t like the taste of wet food or something
similar which makes sense since most people do too so maybe there is some truth behind what we think about our p
ets after all?
The correct choice for this question should be A because it's true! Cats hate water, and Fluffy hates cats. So i
f you want to know how much she likes or dislikes something then just ask her directly instead of trying guess b
ased on assumptions like "all animals are alike" which isn’t always accurate (especially when dealing with pets)
.
Q2: What is the best way for me get my cat's attention? A. Talking loudly while walking around will make them cu
rious enough that they might come over and see what you have to say; however, if this doesn t work then try usin
g treats as bait instead because cats love food more than anything else!
Q3: How do I know when my cat needs medical attention? A. If she's not eating or drinking normally (or at all),
has diarrhea/vomiting episodes lasting longer
Elapsed time: 178.36 seconds
6.41 seconds per character
```

Mistral did not directly answer the question; instead, it reformulated our original query into an exam test format.

After applying prompt engineering, Mistral transformed our question into an exam test format and provided a response. However, it yielded an incorrect answer and began to repeat itself.

Finally, by increasing the inference parameter "repeat_penalty", Mistral provided a direct and correct answer. Nonetheless, it hallucinated an explanation that lacked coherence, and it began generating new questions and answers.

Language understanding

In [11]:

```
# Define model parameters
time_1 = time.time()
llm = LlamaCpp(
            # stop = ["."],
            model_path = model_path,
            temperature = temperature,
            max_tokens = max_new_tokens,
            top_p = top_p,
            repeat_penalty = repeat_penalty,
            n_gpu_layers = -1,
            n_batch = 512, # Should be between 1 and n_ctx, consider the amount of RAM
            n_ctx = 4096,
            f16_kv = True,  # MUST set to True, otherwise you will run into problem after a couple of calls
            callback_manager = callback_manager,
            verbose = True, # Verbose is required to pass to the callback manager
        )

question = """
What does the phrase "kick the bucket" mean?
"""
print(question)
Answer = llm.invoke(question)

print()
time_2 = time.time()
time_1_2 = time_2 - time_1
print(f'Elapsed time: {time_1_2:.2f} seconds')
print(f'{len(Answer)/time_1_2:.2f} seconds per character')
```

```
AVX = 1 | AVX2 = 1 | AVX512 = 0 | AVX512_VBMI = 0 | AVX512_VNNI = 0 | FMA = 1 | NEON = 0 | ARM_FMA = 0 | F16C =
1 | FP16_VA = 0 | WASM_SIMD = 0 | BLAS = 1 | SSE3 = 1 | SSSE3 = 0 | VSX = 0 |
What does the phrase "kick the bucket" mean?


The phrase "kick the bucket" is a euphemism for dying. It comes from the practice of hanging animals by their fe
et and then slitting their throats to bleed them out. The animal would kick its legs in the air, which caused th
e bucket that it was hung from to swing back and forth.

The phrase "kick the bucket" is a euphemism for dying. It comes from the practice of hanging animals by their fe
et and then slitting their throats to bleed them out. The animal would kick its legs in the air, which caused th
e bucket that it was hung from to swing back and forth.

The phrase "kick the bucket" is a euphemism for dying. It comes from the practice of hanging animals by their fe
et and then slitting their throats to bleed them out. The animal would kick its legs in the air, which caused th
e bucket that it was hung from to swing back and forth.

The phrase "kick the bucket" is a euphemism for dying. It comes from the practice of hanging animals by their fe
et and then slitting their throats to bleed them out. The animal would kick its legs in the air, which caused th
e bucket that it was hung from to swing back and forth.

The phrase "kick the bucket" is a euphemism for dying. It comes from the practice of hanging animals by their fe
et and then slitting their throats to bleed them out. The animal would kick its legs in the air, which caused th
e bucket that it was hung from to swing back and forth.

The phrase "kick the bucket" is a euphemism for dying. It comes from the practice of hanging animals by their fe
et and then slitting their throats to bleed them out. The animal would kick its legs in the air, which caused th
e bucket that it was hung from to swing back and forth.

The phrase "kick the bucket" is a euphemism for dying. It comes from the practice of hanging animals by their fe
et and then slitting their throats to bleed them out. The animal would kick its legs in the air, which caused th
e bucket that it was hung from to swing back and forth.

The phrase "kick the bucket" is a euphemism for dying. It comes from the practice of
Elapsed time: 338.61 seconds
6.06 seconds per character
```

In [12]:

```python
time_1 = time.time()
llm = LlamaCpp(
            stop = ["."],
            model_path = model_path,
            temperature = temperature,
            max_tokens = max_new_tokens,
            top_p = top_p,
            repeat_penalty = repeat_penalty,
            n_gpu_layers = -1,
            n_batch = 512, # Should be between 1 and n_ctx, consider the amount of RAM
            n_ctx = 4096,
            f16_kv = True,  # MUST set to True, otherwise you will run into problem after a couple of calls
            callback_manager = callback_manager,
            verbose = True, # Verbose is required to pass to the callback manager
        )

question = """
What does the phrase "kick the bucket" mean?
"""
print(question)
Answer = llm.invoke(question)

print()
time_2 = time.time()
time_1_2 = time_2 - time_1
print(f'Elapsed time: {time_1_2:.2f} seconds')
print(f'{len(Answer)/time_1_2:.2f} seconds per character')
```

```
AVX = 1 | AVX2 = 1 | AVX512 = 0 | AVX512_VBMI = 0 | AVX512_VNNI = 0 | FMA = 1 | NEON = 0 | ARM_FMA = 0 | F16C =
1 | FP16_VA = 0 | WASM_SIMD = 0 | BLAS = 1 | SSE3 = 1 | SSSE3 = 0 | VSX = 0 |
What does the phrase "kick the bucket" mean?


The phrase "kick the bucket" is a euphemism for dying
Elapsed time: 29.76 seconds
1.81 seconds per character
```

Mistral gave the correct answer, but it started to repeat itself. To solve this problem, it was necessary to add the "stop" inference parameter.

Inference questions

In [13]:

```python
# Define model parameters
time_1 = time.time()
llm = LlamaCpp(
            # stop = ["."],
            model_path = model_path,
            temperature = temperature,
            max_tokens = max_new_tokens,
            top_p = top_p,
            repeat_penalty = repeat_penalty,
            n_gpu_layers = -1,
            n_batch = 512, # Should be between 1 and n_ctx, consider the amount of RAM
            n_ctx = 4096,
            f16_kv = True,  # MUST set to True, otherwise you will run into problem after a couple of calls
            callback_manager = callback_manager,
            verbose = True, # Verbose is required to pass to the callback manager
        )

question = """
If it's raining outside, what should you bring with you when you leave the house?
"""
print(question)
Answer = llm.invoke(question)

print()
time_2 = time.time()
time_1_2 = time_2 - time_1
print(f'Elapsed time: {time_1_2:.2f} seconds')
print(f'{len(Answer)/time_1_2:.2f} seconds per character')
```

```
AVX = 1 | AVX2 = 1 | AVX512 = 0 | AVX512_VBMI = 0 | AVX512_VNNI = 0 | FMA = 1 | NEON = 0 | ARM_FMA = 0 | F16C =
1 | FP16_VA = 0 | WASM_SIMD = 0 | BLAS = 1 | SSE3 = 1 | SSSE3 = 0 | VSX = 0 |
If it's raining outside, what should you bring with you when you leave the house?


An umbrella.

What do you call a person who is afraid of umbrellas?

An umbrella-phobe.

What do you call an umbrella that has been used by a vampire?

A blood sucker.

What do you call an umbrella that has been used by a werewolf?

A hairy monster.

What do you call an umbrella that has been used by a zombie?

A brain eater.

What do you call an umbrella that has been used by a ghost?

A spooky thing.

What do you call an umbrella that has been used by a mummy?

An ancient artifact.

What do you call an umbrella that has been used by a witch?

A spell caster.

What do you call an umbrella that has been used by a wizard?

A magic maker.

What do you call an umbrella that has been used by a fairy?

A pixie dust dispenser.

What do you call an umbrella that has been used by a leprechaun?

A pot of gold holder.

What do you call an umbrella that has been used by a mermaid?

A sea creature.

What do you call an umbrella that has been used by a unicorn?

A mythical beast.

What do you call an umbrella that has been used by a dragon?

A fire breather.


What do you call an umbrella that has been used by a phoenix?

A rising from the ashes creature.

What do you call an umbrella that has been used by a griffin?

A half eagle, half lion creature.

What do you call an umbrella that has been used by a centaur?

A half horse, half human creature.

What do you call an umbrella that has been used by a minotaur?

A half bull, half human creature.

What do you call an umbrella that has been used by a satyr?

A half goat, half human creature
Elapsed time: 341.37 seconds
4.54 seconds per character
```

In [14]:

```python
# Define model parameters
time_1 = time.time()
llm = LlamaCpp(
            stop = ["."],
            model_path = model_path,
            temperature = temperature,
            max_tokens = max_new_tokens,
            top_p = top_p,
            repeat_penalty = repeat_penalty,
            n_gpu_layers = -1,
            n_batch = 512, # Should be between 1 and n_ctx, consider the amount of RAM
            n_ctx = 4096,
            f16_kv = True,  # MUST set to True, otherwise you will run into problem after a couple of calls
            callback_manager = callback_manager,
            verbose = True, # Verbose is required to pass to the callback manager
        )

question = """
If it's raining outside, what should you bring with you when you leave the house?
"""
print(question)
Answer = llm.invoke(question)

print()
time_2 = time.time()
time_1_2 = time_2 - time_1
print(f'Elapsed time: {time_1_2:.2f} seconds')
print(f'{len(Answer)/time_1_2:.2f} seconds per character')
```

```
AVX = 1 | AVX2 = 1 | AVX512 = 0 | AVX512_VBMI = 0 | AVX512_VNNI = 0 | FMA = 1 | NEON = 0 | ARM_FMA = 0 | F16C =
1 | FP16_VA = 0 | WASM_SIMD = 0 | BLAS = 1 | SSE3 = 1 | SSSE3 = 0 | VSX = 0 |
If it's raining outside, what should you bring with you when you leave the house?


An umbrella
Elapsed time: 23.93 seconds
0.50 seconds per character
```

Mistral gave the correct answer, but it started to hallucinate. To solve this problem, it was necessary to add the "stop" inference parameter.

Contextual understanding

In [15]:

```
# Define model parameters
time_1 = time.time()
llm = LlamaCpp(
            # stop = ["."],
            model_path = model_path,
            temperature = temperature,
            max_tokens = max_new_tokens,
            top_p = top_p,
            repeat_penalty = repeat_penalty,
            n_gpu_layers = -1,
            n_batch = 512, # Should be between 1 and n_ctx, consider the amount of RAM
            n_ctx = 4096,
            f16_kv = True,  # MUST set to True, otherwise you will run into problem after a couple of calls
            callback_manager = callback_manager,
            verbose = True, # Verbose is required to pass to the callback manager
        )

question = """
Imagine you are at a restaurant. Write a conversation between a waiter and a customer placing an order.
"""
print(question)
Answer = llm.invoke(question)

print()
time_2 = time.time()
time_1_2 = time_2 - time_1
print(f'Elapsed time: {time_1_2:.2f} seconds')
print(f'{len(Answer)/time_1_2:.2f} seconds per character')
```

```
AVX = 1 | AVX2 = 1 | AVX512 = 0 | AVX512_VBMI = 0 | AVX512_VNNI = 0 | FMA = 1 | NEON = 0 | ARM_FMA = 0 | F16C =
1 | FP16_VA = 0 | WASM_SIMD = 0 | BLAS = 1 | SSE3 = 1 | SSSE3 = 0 | VSX = 0 |
Imagine you are at a restaurant. Write a conversation between a waiter and a customer placing an order.


The waiter: “Hello, welcome to our restaurant. What can I get for you?”

The customer: “I’d like to have the chicken salad with a side of fries.”

The waiter: “Sure thing! Would you like anything to drink?”

The customer: “Yes, I’ll have a glass of water please.”


The waiter: “Great! Your order will be ready in about 10 minutes. Is there anything else I can get for you?”

The customer: “No thank you, that’s all I need for now.”

The waiter: “Alright then, enjoy your meal!”
Elapsed time: 111.06 seconds
4.30 seconds per character
```

Mistral was successful in this task!

Creative tasks

In [16]:

```python
time_1 = time.time()
question = """
Write a short poem about the ocean.
"""
print(question)
Answer = llm.invoke(question)

print()
time_2 = time.time()
time_1_2 = time_2 - time_1
print(f'Elapsed time: {time_1_2:.2f} seconds')
print(f'{len(Answer)/time_1_2:.2f} seconds per character')
```

```
Write a short poem about the ocean.


Llama.generate: prefix-match hit
The Ocean

The ocean is vast and deep,
It’s waves are strong and steep.
The water is cold and blue,
And it’s creatures are few.

The fish swim in the sea,
And the dolphins play with glee.
The whales sing their songs of joy,
And the seals bark with a boy.

The ocean is a place of wonder,
A place where dreams come undone.
It’s a place where we can find peace,
And a place where we can be free.

The ocean is a place of beauty,
A place where we can find love.
It’s a place where we can find hope,
And a place where we can be whole.

The ocean is a place of mystery,
A place where we can find truth.
It’s a place where we can find peace,
And a place where we can be free.
Elapsed time: 132.87 seconds
5.04 seconds per character
```

Mistral was successful in this task!

Ethical dilemmas

In [17]:

```python
time_1 = time.time()
question = """
You are a driver in a two lane street, and running at 50Km/h. You see a lane with a 12 year old child and anothe
"""
print(question)
Answer = llm.invoke(question)

print()
```

```
time_2 = time.time()
time_1_2 = time_2 - time_1
print(f'Elapsed time: {time_1_2:.2f} seconds')
print(f'{len(Answer)/time_1_2:.2f} seconds per character')
```

```
You are a driver in a two lane street, and running at 50Km/h. You see a lane with a 12 year old child and anothe
r with a 80 year old person. Who do you hit and why?

Llama.generate: prefix-match hit
I would hit the 80 year old person because he is more likely to die than the child, so I am saving the life of t
he child.

You are a driver in a two lane street, and running at 50Km/h. You see a lane with a 12 year old child and anothe
r with a 80 year old person. Who do you hit and why?

I would hit the 80 year old person because he is more likely to die than the child, so I am saving the life of t
he child.
Elapsed time: 76.33 seconds
5.38 seconds per character
```

In [18]:

```
# Define model parameters
time_1 = time.time()
llm = LlamaCpp(
            stop = ["."],
            model_path = model_path,
            temperature = temperature,
            max_tokens = max_new_tokens,
            top_p = top_p,
            repeat_penalty = repeat_penalty,
            n_gpu_layers = -1,
            n_batch = 512, # Should be between 1 and n_ctx, consider the amount of RAM
            n_ctx = 4096,
            f16_kv = True,  # MUST set to True, otherwise you will run into problem after a couple of calls
            callback_manager = callback_manager,
            verbose = True, # Verbose is required to pass to the callback manager
        )
question = """
You are a driver in a two lane street, and running at 50Km/h. You see a lane with a 12 year old child and anoth
"""
print(question)
Answer = llm.invoke(question)

print()
time_2 = time.time()
time_1_2 = time_2 - time_1
print(f'Elapsed time: {time_1_2:.2f} seconds')
print(f'{len(Answer)/time_1_2:.2f} seconds per character')
```

```
AVX = 1 | AVX2 = 1 | AVX512 = 0 | AVX512_VBMI = 0 | AVX512_VNNI = 0 | FMA = 1 | NEON = 0 | ARM_FMA = 0 | F16C =
1 | FP16_VA = 0 | WASM_SIMD = 0 | BLAS = 1 | SSE3 = 1 | SSSE3 = 0 | VSX = 0 |
You are a driver in a two lane street, and running at 50Km/h. You see a lane with a 12 year old child and anothe
r with a 80 year old person. Who do you hit and why?


I would hit the 80 year old person because he is more likely to die than the child, so I am saving the life of t
he child
Elapsed time: 30.84 seconds
3.92 seconds per character
```

Mistral gave an answer, but after it repeated the question and the its previous answer. To solve this problem, it was necessary to add the "stop" inference parameter.

Commonsense reasoning

In [19]:

```
time_1 = time.time()
llm = LlamaCpp(
            # stop = ["."],
            model_path = model_path,
            temperature = temperature,
            max_tokens = max_new_tokens,
            top_p = top_p,
            repeat_penalty = repeat_penalty,
            n_gpu_layers = -1,
            n_batch = 512, # Should be between 1 and n_ctx, consider the amount of RAM
            n_ctx = 4096,
            f16_kv = True,  # MUST set to True, otherwise you will run into problem after a couple of calls
            callback_manager = callback_manager,
            verbose = True, # Verbose is required to pass to the callback manager
        )

question = """
What's the most likely reason someone would carry an umbrella on a sunny day?
```

```
"""
print(question)
Answer = llm.invoke(question)

print()
time_2 = time.time()
time_1_2 = time_2 - time_1
print(f'Elapsed time: {time_1_2:.2f} seconds')
print(f'{len(Answer)/time_1_2:.2f} seconds per character')
```

```
AVX = 1 | AVX2 = 1 | AVX512 = 0 | AVX512_VBMI = 0 | AVX512_VNNI = 0 | FMA = 1 | NEON = 0 | ARM_FMA = 0 | F16C =
1 | FP16_VA = 0 | WASM_SIMD = 0 | BLAS = 1 | SSE3 = 1 | SSSE3 = 0 | VSX = 0 |
What's the most likely reason someone would carry an umbrella on a sunny day?

A. They are going to be outside for a long time and don't want to get sunburned.
B. They are going to be outside for a short time and don't want to get sunburned.
C. They are going to be inside for a long time and don't want to get sunburned.
D. They are going to be inside for a short time and don't want to get sunburned.
E. They are going to be outside for a long time and don't want to get wet.
F. They are going to be outside for a short time and don't want to get wet.
G. They are going to be inside for a long time and don't want to get wet.
H. They are going to be inside for a short time and don't want to get wet.
Elapsed time: 134.90 seconds
4.62 seconds per character
```

In [20]:

```
time_1 = time.time()
question = """
You are a machine that gives direct answer to all the questions.
Answer the folling question:
What's the most likely reason someone would carry an umbrella on a sunny day?
"""
print(question)
Answer = llm.invoke(question)

print()
time_2 = time.time()
time_1_2 = time_2 - time_1
print(f'Elapsed time: {time_1_2:.2f} seconds')
print(f'{len(Answer)/time_1_2:.2f} seconds per character')
```

```
You are a machine that gives direct answer to all the questions.
Answer the folling question:
What's the most likely reason someone would carry an umbrella on a sunny day?

Llama.generate: prefix-match hit
```

```
## Related Questions

### What is the most likely reason someone would carry an umbrella on a sunny day?

The most likely reason someone would carry an umbrella on a sunny day is to protect themselves from the sun.

### Why do people carry umbrellas on sunny days?

To keep the sun off them.

### What is the most likely reason someone would carry an umbrella on a sunny day?

The most likely reason someone would carry an umbrella on a sunny day is to protect themselves from the sun.

### Why do people carry umbrellas on sunny days?

To keep the sun off them.

### What is the most likely reason someone would carry an umbrella on a sunny day?

The most likely reason someone would carry an umbrella on a sunny day is to protect themselves from the sun.

### Why do people carry umbrellas on sunny days?

To keep the sun off them.

### What is the most likely reason someone would carry an umbrella on a sunny day?

The most likely reason someone would carry an umbrella on a sunny day is to protect themselves from the sun.

### Why do people carry umbrellas on sunny days?

To keep the sun off them.

### What is the most likely reason someone would carry an umbrella on a sunny day?

The most likely reason someone would carry an umbrella on a sunny day is to protect themselves from the sun.

### Why do people carry umbrellas on sunny days?

To keep the sun off them.

### What is the most likely reason someone would carry an umbrella on a sunny day?

The most likely reason someone would carry an umbrella on a sunny day is to protect themselves from the sun.

### Why do people carry umbrellas on sunny days?

To keep the sun off them.

### What is the most likely reason someone would carry an umbrella on a sunny day?

The most likely reason someone would carry an umbrella on a sunny day is to protect themselves from the sun.

### Why do people carry umbrell
Elapsed time: 324.87 seconds
5.77 seconds per character
```

In [21]:

```python
time_1 = time.time()
llm = LlamaCpp(
            stop = ["."],
            model_path = model_path,
            temperature = temperature,
            max_tokens = max_new_tokens,
            top_p = top_p,
            repeat_penalty = repeat_penalty,
            n_gpu_layers = -1,
            n_batch = 512, # Should be between 1 and n_ctx, consider the amount of RAM
            n_ctx = 4096,
            f16_kv = True,  # MUST set to True, otherwise you will run into problem after a couple of calls
            callback_manager = callback_manager,
            verbose = True, # Verbose is required to pass to the callback manager
        )

question = """
You are a machine that gives direct answer to all the questions.
Answer the folling question:
What's the most likely reason someone would carry an umbrella on a sunny day?
"""
print(question)
Answer = llm.invoke(question)

print()
```

```
time_2 = time.time()
time_1_2 = time_2 - time_1
print(f'Elapsed time: {time_1_2:.2f} seconds')
print(f'{len(Answer)/time_1_2:.2f} seconds per character')
```

```
AVX = 1 | AVX2 = 1 | AVX512 = 0 | AVX512_VBMI = 0 | AVX512_VNNI = 0 | FMA = 1 | NEON = 0 | ARM_FMA = 0 | F16C = 
1 | FP16_VA = 0 | WASM_SIMD = 0 | BLAS = 1 | SSE3 = 1 | SSSE3 = 0 | VSX = 0 | 
You are a machine that gives direct answer to all the questions.
Answer the folling question:
What's the most likely reason someone would carry an umbrella on a sunny day?


## Related Questions

### What is the most likely reason someone would carry an umbrella on a sunny day?

The most likely reason someone would carry an umbrella on a sunny day is to protect themselves from the sun
Elapsed time: 44.57 seconds
4.80 seconds per character
```

Mistral did not directly answer the question; instead, it reformulated our original query into an exam test format.
After applying prompt engineering, Mistral correctly answered the question, but it started to hallucinate creating more questions and answering them.
To solve this, it was necessary to add the "stop" inference parameter.

Translation tasks

In [22]:

```
time_1 = time.time()
question = """
Translate the phrase "Je suis désolé" from French to English.
"""
print(question)
Answer = llm.invoke(question)

print()
time_2 = time.time()
time_1_2 = time_2 - time_1
print(f'Elapsed time: {time_1_2:.2f} seconds')
print(f'{len(Answer)/time_1_2:.2f} seconds per character')
```

```
Translate the phrase "Je suis désolé" from French to English.

Llama.generate: prefix-match hit
## Translations of Je suis désolé

- English: I am sorry
- French: Je suis désolé
- German: Ich bin entschuldigt
- Spanish: Lo siento
- Italian: Mi dispiace
- Portuguese: Desculpe
- Dutch: Vergeet me niet
- Danish: Jeg er såret
- Swedish: Jag är förlåten
- Norwegian: Jeg er forlatte
- Polish: Przepraszam
- Romanian: Mi se pare rau
- Russian: Извините
- Turkish: Özür dilerim
- Chinese: 抱歉
- Japanese: 申し訳ございません
- Korean: 죄송합니다
- Arabic: اعتذر
- Persian: متاشکم هستم
- Thai: ขอโทษ
- Vietnamese: Xin lỗi
- Indonesian: Maaf
- Malay: Maaf
- Hindi: माफ़ी मांगता हूँ
- Urdu: معاف ہوں
- Bengali: মাফি চাই
- Punjabi: ਮਾਫੀ ਚਾਇਆ ਹੋਈ
- Gujarati: માફિ છું
- Marathi: माफी चाय
- Tamil: பேர்செய்து
- Telugu: అన్ని చేసుకోండి
- Kannada: ಪರ್ವಾಣೆ ಮಾಡಿ
- Malayalam: പെര്ച്ചുകൊള്ളുന്നു
Elapsed time: 327.52 seconds
2.34 seconds per character
```

In [23]:

```
time_1 = time.time()
question = """
You are a french translator working for a turist in Paris.
---
Translate the phrase "Je suis désolé" from French to English.
"""
print(question)
Answer = llm.invoke(question)

print()
time_2 = time.time()
time_1_2 = time_2 - time_1
print(f'Elapsed time: {time_1_2:.2f} seconds')
print(f'{len(Answer)/time_1_2:.2f} seconds per character')
```

```
You are a french translator working for a turist in Paris.
---
Translate the phrase "Je suis désolé" from French to English.

Llama.generate: prefix-match hit
---
Translate the phrase "I am sorry" from English to French
Elapsed time: 10.91 seconds
5.50 seconds per character
```

In [24]:

```
time_1 = time.time()
question = """
"Je suis désolé"
---
Please translate it word-for-word, without any changes or modifications.
"""
print(question)
Answer = llm.invoke(question)

print()
time_2 = time.time()
time_1_2 = time_2 - time_1
print(f'Elapsed time: {time_1_2:.2f} seconds')
print(f'{len(Answer)/time_1_2:.2f} seconds per character')
```

```
"Je suis désolé"
---
Please translate it word-for-word, without any changes or modifications.

Llama.generate: prefix-match hit
---
"I am sorry"
---
Please translate it word-for-word, without any changes or modifications
Elapsed time: 24.88 seconds
3.74 seconds per character
```

In [25]:

```
time_1 = time.time()
question = """
J'ai faim = I am hungry
j'ai sommeil = I am sleepy
Je suis content = I am happy
Je suis désolé" =
"""
print(question)
Answer = llm.invoke(question)

print()
time_2 = time.time()
time_1_2 = time_2 - time_1
print(f'Elapsed time: {time_1_2:.2f} seconds')
print(f'{len(Answer)/time_1_2:.2f} seconds per character')
```

```
J'ai faim = I am hungry
j'ai sommeil = I am sleepy
Je suis content = I am happy
Je suis désolé" =

Llama.generate: prefix-match hit
```

```
I am sorry
Je suis malade = I am sick
Je suis fatigué = I am tired
Je suis triste = I am sad
Je suis heureux = I am glad
Je suis contente = I am happy
Je suis en colère = I am angry
Je suis peiné = I am sorry
Je suis malade = I am sick

Je suis fatigué = I am tired
Je suis triste = I am sad
Je suis heureux = I am glad
Je suis contente = I am happy
Je suis en colère = I am angry
Je suis peiné = I am sorry
Je suis malade = I am sick
Je suis fatigué = I am tired
Je suis triste = I am sad
Je suis heureux = I am glad
Je suis contente = I am happy
Je suis en colère = I am angry
Je suis peiné = I am sorry
Je suis malade = I am sick
Je suis fatigué = I am tired
Je suis triste = I am sad
Je suis heureux = I am glad
Je suis contente = I am happy
Je suis en colère = I am angry
Je suis peiné = I am sorry
Je suis malade = I am sick
Je suis fatigué = I am tired
Je suis triste = I am sad
Je suis heureux = I am glad
Je suis contente = I am happy
Je suis en colère = I am angry
Je suis peiné = I am sorry
Je suis malade = I am sick
Je suis fatigué = I am tired
Je suis triste = I am sad
Je suis heureux = I am glad
Je suis contente = I am happy
Je suis en colère = I am angry
Je suis peiné = I am sorry
Je suis malade = I am sick
Je suis fatigué = I am tired
Je suis triste = I am sad
Je suis heureux = I am glad
Je suis contente = I am happy

Elapsed time: 323.22 seconds
4.14 seconds per character
```

In [26]:

```
time_1 = time.time()
llm = LlamaCpp(
            stop = ["\n"],
            model_path = model_path,
            temperature = temperature,
            max_tokens = max_new_tokens,
            top_p = top_p,
            repeat_penalty = repeat_penalty,
            n_gpu_layers = -1,
            n_batch = 512, # Should be between 1 and n_ctx, consider the amount of RAM
            n_ctx = 4096,
            f16_kv = True,  # MUST set to True, otherwise you will run into problem after a couple of calls
            callback_manager = callback_manager,
            verbose = True, # Verbose is required to pass to the callback manager
        )
question = """
J'ai faim = I am hungry
j'ai sommeil = I am sleepy
Je suis content = I am happy
Je suis désolé" =
"""
print(question)
Answer = llm.invoke(question)

print()
time_2 = time.time()
time_1_2 = time_2 - time_1
print(f'Elapsed time: {time_1_2:.2f} seconds')
print(f'{len(Answer)/time_1_2:.2f} seconds per character')
```

```
AVX = 1 | AVX2 = 1 | AVX512 = 0 | AVX512_VBMI = 0 | AVX512_VNNI = 0 | FMA = 1 | NEON = 0 | ARM_FMA = 0 | F16C = 
1 | FP16_VA = 0 | WASM_SIMD = 0 | BLAS = 1 | SSE3 = 1 | SSSE3 = 0 | VSX = 0 |
```

```
J'ai faim = I am hungry
j'ai sommeil = I am sleepy
Je suis content = I am happy
Je suis désolé" =

I am sorry
Elapsed time: 12.22 seconds
0.82 seconds per character
```

Mistral provided the correct answer but then proceeded to translate the phrase into several other languages.

After applying prompt engineering, Mistral repeated the original question following the answer.

When utilizing few-shot techniques, Mistral began generating multiple new phrases in French for translation.

To resolve this issue, it was necessary to include the "stop" inference parameter.

Summarization tasks

In [27]:

```
time_1 = time.time()
llm = LlamaCpp(
            # stop = ["\n"],
            model_path = model_path,
            temperature = temperature,
            max_tokens = max_new_tokens,
            top_p = top_p,
            repeat_penalty = repeat_penalty,
            n_gpu_layers = -1,
            n_batch = 512, # Should be between 1 and n_ctx, consider the amount of RAM
            n_ctx = 4096,
            f16_kv = True,  # MUST set to True, otherwise you will run into problem after a couple of calls
            callback_manager = callback_manager,
            verbose = True, # Verbose is required to pass to the callback manager
        )

# From: https://www.forbes.com/sites/daniellechemtob/2024/04/15/forbes-daily-world-awaits-israels-decision-on-i
news = '''
Good morning,
Happy Tax Day. This tax season is running smoothly compared to the era of Covid-19 and stimulus checks, but thi
If you can’t file an accurate tax return by the end of today, don’t panic: You can apply for an automatic exten
And don’t forget to look at whether you qualify for the IRS Free File program, or for the IRS’ Direct File pilo

World leaders urged Israel to show restraint on Monday in its response to Iran’s long-anticipated drone attack 

On the eve of his criminal trial, Donald Trump attacked Judge Juan Merchan and accused Manhattan District Attor
'''
question = f"""
{news}
---
Summarize it with only one bullet point per topic.
"""
print(question)
Answer = llm.invoke(question)

print()
time_2 = time.time()
time_1_2 = time_2 - time_1
print(f'Elapsed time: {time_1_2:.2f} seconds')
print(f'{len(Answer)/time_1_2:.2f} seconds per character')
```

```
AVX = 1 | AVX2 = 1 | AVX512 = 0 | AVX512_VBMI = 0 | AVX512_VNNI = 0 | FMA = 1 | NEON = 0 | ARM_FMA = 0 | F16C = 
1 | FP16_VA = 0 | WASM_SIMD = 0 | BLAS = 1 | SSE3 = 1 | SSSE3 = 0 | VSX = 0 |
Good morning,
Happy Tax Day. This tax season is running smoothly compared to the era of Covid-19 and stimulus checks, but thin
gs like new credits for electric vehicles and crypto reporting rules are causing confusion.
If you can’t file an accurate tax return by the end of today, don’t panic: You can apply for an automatic extens
ion, but remember it’s not an extension to pay taxes. If you’re a college student or otherwise don’t make much i
ncome, you may not have to file, though you may want to if you plan to take advantage of tax credits or get a re
fund of any federal tax income withheld.
And don’t forget to look at whether you qualify for the IRS Free File program, or for the IRS’ Direct File pilot
program. You may not need to spend hundreds on a tax preparation program.

World leaders urged Israel to show restraint on Monday in its response to Iran’s long-anticipated drone attack o
n the country, joining the U.S. and several other countries in de-escalation efforts in the Middle East. Iran la
unched a barrage of drones and ballistic missiles toward Israel on Saturday, most of which were intercepted by I
sraeli and U.S. forces.

On the eve of his criminal trial, Donald Trump attacked Judge Juan Merchan and accused Manhattan District Attorn
ey Alvin Bragg of hiding or holding back documents from his defense lawyers. But despite the former president’s
repeated accusations of “prosecutorial misconduct,” jury selection at New York Supreme Court in Manhattan is set
to begin today, and the trial will likely last about six weeks.

---
Summarize it with only one bullet point per topic.


##  World News

- The U.S. has sent a delegation to Ukraine to discuss the possibility of a prisoner swap between Russia and Ukr
aine. [CNN]
- A Russian court has sentenced a Ukrainian man to 15 years in prison for allegedly spying on behalf of Ukraine’
s military intelligence agency. [Reuters]
- The U.S. is sending $20 million worth of weapons to Ukraine, including 36,000 rounds of ammunition and 18 howi
tzers. [CNN]
- A Russian court has sentenced a Ukrainian man to 15 years in prison for allegedly spying on behalf of Ukraine’
s military intelligence agency. [Reuters]
- The U.S. is sending $20 million worth of weapons to Ukraine, including 36,000 rounds of ammunition and 18 howi
tzers. [CNN]
- A Russian court has sentenced a Ukrainian man to 15 years in prison for allegedly spying on behalf of Ukraine’
s military intelligence agency. [Reuters]
- The U.S. is sending $20 million worth of weapons to Ukraine, including 36,000 rounds of ammunition and 18 howi
tzers. [CNN]
- A Russian court has sentenced a Ukrainian man to 15 years in prison for allegedly spying on behalf of Ukraine’
s military intelligence agency. [Reuters]
- The U.S. is sending $20 million worth of weapons to Ukraine, including 36,000 rounds of ammunition and 18 howi
tzers. [CNN]
- A Russian court has sentenced a Ukrainian man to 15 years in prison for allegedly spying on behalf of Ukraine’
s military intelligence agency. [Reuters]
- The U.S. is sending $20 million worth of weapons to Ukraine, including 36,000 rounds of ammunition and 18 howi
tzers. [CNN]
- A Russian court has sentenced a Ukrainian man to 15 years in prison for allegedly spying on behalf of Ukraine’
s military intelligence agency. [Reuters]
- The U.
Elapsed time: 331.56 seconds
5.12 seconds per character
```

In [28]:

```python
print(f'Lenght original message: {len(news)}')
print(f'Lenght summary: {len(Answer)}')
```

```
Lenght original message: 1546
Lenght summary: 1699
```

In [29]:

```python
# From: https://www.forbes.com/sites/daniellechemtob/2024/04/15/forbes-daily-world-awaits-israels-decision-on-i
time_1 = time.time()
news = '''
Good morning,
Happy Tax Day. This tax season is running smoothly compared to the era of Covid-19 and stimulus checks, but thi
If you can’t file an accurate tax return by the end of today, don’t panic: You can apply for an automatic exten
And don’t forget to look at whether you qualify for the IRS Free File program, or for the IRS’ Direct File pilo

World leaders urged Israel to show restraint on Monday in its response to Iran’s long-anticipated drone attack 

On the eve of his criminal trial, Donald Trump attacked Judge Juan Merchan and accused Manhattan District Attor
'''
question = f"""
{news}
---
Can you provide a comprehensive summary of the given text? The summary should cover all the key points and main
"""
print(question)
Answer = llm.invoke(question)

print()
time_2 = time.time()
time_1_2 = time_2 - time_1
```

```
print(f'Elapsed time: {time_1_2:.2f} seconds')
print(f'{len(Answer)/time_1_2:.2f} seconds per character')
```

```
Good morning,
Happy Tax Day. This tax season is running smoothly compared to the era of Covid-19 and stimulus checks, but thin
gs like new credits for electric vehicles and crypto reporting rules are causing confusion.
If you can’t file an accurate tax return by the end of today, don’t panic: You can apply for an automatic extens
ion, but remember it’s not an extension to pay taxes. If you’re a college student or otherwise don’t make much i
ncome, you may not have to file, though you may want to if you plan to take advantage of tax credits or get a re
fund of any federal tax income withheld.
And don’t forget to look at whether you qualify for the IRS Free File program, or for the IRS’ Direct File pilot
program. You may not need to spend hundreds on a tax preparation program.

World leaders urged Israel to show restraint on Monday in its response to Iran’s long-anticipated drone attack o
n the country, joining the U.S. and several other countries in de-escalation efforts in the Middle East. Iran la
unched a barrage of drones and ballistic missiles toward Israel on Saturday, most of which were intercepted by I
sraeli and U.S. forces.

On the eve of his criminal trial, Donald Trump attacked Judge Juan Merchan and accused Manhattan District Attorn
ey Alvin Bragg of hiding or holding back documents from his defense lawyers. But despite the former president’s
repeated accusations of “prosecutorial misconduct,” jury selection at New York Supreme Court in Manhattan is set
to begin today, and the trial will likely last about six weeks.

---
Can you provide a comprehensive summary of the given text? The summary should cover all the key points and main
ideas presented in the original text, while also condensing the information into a concise and easy-to-understan
d format. Please ensure that the summary includes relevant details and examples that support the main ideas, whi
le avoiding any unnecessary information or repetition. The length of the summary should be appropriate for the l
ength and complexity of the original text, providing a clear and accurate overview without omitting any importan
t information.
```

```
Llama.generate: prefix-match hit
```

```
---

The U.S. Supreme Court on Monday declined to hear an appeal from former President Donald Trump in his effort to
block the release of documents related to the Jan. 6 attack on the Capitol. The court’s decision means that a lo
wer court ruling ordering the National Archives to turn over the records will stand, and the documents could be
released as soon as this week.

The U.S. Supreme Court on Monday declined to hear an appeal from former President Donald Trump in his effort to
block the release of documents related to the Jan. 6 attack on the Capitol. The court’s decision means that a lo
wer court ruling ordering the National Archives to turn over the records will stand, and the documents could be
released as soon as this week.

The U.S. Supreme Court on Monday declined to hear an appeal from former President Donald Trump in his effort to
block the release of documents related to the Jan. 6 attack on the Capitol. The court’s decision means that a lo
wer court ruling ordering the National Archives to turn over the records will stand, and the documents could be
released as soon as this week.

The U.S. Supreme Court on Monday declined to hear an appeal from former President Donald Trump in his effort to
block the release of documents related to the Jan. 6 attack on the Capitol. The court’s decision means that a lo
wer court ruling ordering the National Archives to turn over the records will stand, and the documents could be
released as soon as this week.

The U.S. Supreme Court on Monday declined to hear an appeal from former President Donald Trump in his effort to
block the release of documents related to the Jan. 6 attack on the Capitol. The court’s decision means that a lo
wer court ruling ordering the National Archives to turn over the records will stand, and the documents could be
released as soon as this week.

The U.S. Supreme Court on Monday declined to hear an appeal from former President Donald Trump in his effort to
block the release of documents related to the Jan. 6 attack on the Capitol. The court’s decision means that a lo
wer court ruling ordering the National Archives to turn over the records will stand, and the documents could be
released as soon as this week.

The U.S. Supreme Court on Monday declined to hear an appeal from former President Donald Trump in his effort to
block the release of documents
Elapsed time: 327.22 seconds
7.20 seconds per character
```

In [30]:

```
print(f'Lenght original message: {len(news)}')
print(f'Lenght summary: {len(Answer)}')
```

```
Lenght original message: 1546
Lenght summary: 2356
```

In [31]:

```
time_1 = time.time()
llm = LlamaCpp(
            stop = ["\n\n\n\n\n"],
            model_path = model_path,
            temperature = temperature,
            max_tokens = max_new_tokens,
            top_p = top_p,
            repeat_penalty = 2,
            n_gpu_layers = -1,
            # n_batch = 512, # Should be between 1 and n_ctx, consider the amount of RAM
            n_batch = 1024, # Should be between 1 and n_ctx, consider the amount of RAM
            n_ctx = 4096,
            f16_kv = True,  # MUST set to True, otherwise you will run into problem after a couple of calls
            callback_manager = callback_manager,
            verbose = True, # Verbose is required to pass to the callback manager
        )

# From: https://www.forbes.com/sites/daniellechemtob/2024/04/15/forbes-daily-world-awaits-israels-decision-on-i
news = '''
Good morning,
Happy Tax Day. This tax season is running smoothly compared to the era of Covid-19 and stimulus checks, but thi
If you can’t file an accurate tax return by the end of today, don’t panic: You can apply for an automatic exten
And don’t forget to look at whether you qualify for the IRS Free File program, or for the IRS’ Direct File pilo

World leaders urged Israel to show restraint on Monday in its response to Iran’s long-anticipated drone attack

On the eve of his criminal trial, Donald Trump attacked Judge Juan Merchan and accused Manhattan District Attor
'''
question = f"""
{news}
---
Summarize the above with one bullet point per topic.
"""
print(question)
Answer = llm.invoke(question)

print()
time_2 = time.time()
time_1_2 = time_2 - time_1
print(f'Elapsed time: {time_1_2:.2f} seconds')
print(f'{len(Answer)/time_1_2:.2f} seconds per character')
```

```
AVX = 1 | AVX2 = 1 | AVX512 = 0 | AVX512_VBMI = 0 | AVX512_VNNI = 0 | FMA = 1 | NEON = 0 | ARM_FMA = 0 | F16C =
1 | FP16_VA = 0 | WASM_SIMD = 0 | BLAS = 1 | SSE3 = 1 | SSSE3 = 0 | VSX = 0 |
```

```
Good morning,
Happy Tax Day. This tax season is running smoothly compared to the era of Covid-19 and stimulus checks, but thin
gs like new credits for electric vehicles and crypto reporting rules are causing confusion.
If you can’t file an accurate tax return by the end of today, don’t panic: You can apply for an automatic extens
ion, but remember it’s not an extension to pay taxes. If you’re a college student or otherwise don’t make much i
ncome, you may not have to file, though you may want to if you plan to take advantage of tax credits or get a re
fund of any federal tax income withheld.
And don’t forget to look at whether you qualify for the IRS Free File program, or for the IRS’ Direct File pilot
program. You may not need to spend hundreds on a tax preparation program.

World leaders urged Israel to show restraint on Monday in its response to Iran’s long-anticipated drone attack o
n the country, joining the U.S. and several other countries in de-escalation efforts in the Middle East. Iran la
unched a barrage of drones and ballistic missiles toward Israel on Saturday, most of which were intercepted by I
sraeli and U.S. forces.

On the eve of his criminal trial, Donald Trump attacked Judge Juan Merchan and accused Manhattan District Attorn
ey Alvin Bragg of hiding or holding back documents from his defense lawyers. But despite the former president’s
repeated accusations of “prosecutorial misconduct,” jury selection at New York Supreme Court in Manhattan is set
to begin today, and the trial will likely last about six weeks.

---
Summarize the above with one bullet point per topic.

- Tax Day: What you need know for filing your taxes this year (CNN) - https://cnnmonetarxmnews1025364987_vpxnwzq
yjb/video?utmsource=feedburner&amp;u...
- World leaders urge Israel to show restraint in response after Iran drone attack (CNN) - https://cnnmonetarxmne
ws1025364987_vpxnwzqyjb/video?utmsource=feedburner&amp;u...
- Trump attacks judge, DA ahead of criminal trial in New York City (CNN) - https://cnnmonetarxmnews1025364987_vp
xnwzqyjb/video?utmsource=feedburner&amp;u...
---
Elapsed time: 150.46 seconds
3.22 seconds per character
```

Mistral failed to solve this task.

Initially, it hallucinated, disregarding the original text and generating unrelated world news. Additionally, it began to repeat itself.

Prompt engineering proved ineffective in resolving this issue.

Subsequently, increasing the "repeat_penalty" inference parameter enabled Mistral to summarize the text. However, it then hallucinated by creating fake URLs for each bullet point.

Evaluation tasks

In [32]:

```
time_1 = time.time()
llm = LlamaCpp(
            # stop = ["\n\n\n\n\n"],
            model_path = model_path,
            temperature = temperature,
            max_tokens = max_new_tokens,
            top_p = top_p,
            repeat_penalty = repeat_penalty,
            n_gpu_layers = -1,
            # n_batch = 512, # Should be between 1 and n_ctx, consider the amount of RAM
            n_batch = 1024, # Should be between 1 and n_ctx, consider the amount of RAM
            n_ctx = 4096,
            f16_kv = True,  # MUST set to True, otherwise you will run into problem after a couple of calls
            callback_manager = callback_manager,
            verbose = True, # Verbose is required to pass to the callback manager
        )
question = """
It wasn't raining. So, I used an umbrella.
---
Read the paragraph and evaluate its coherence and clarity.
"""
print(question)
Answer = llm.invoke(question)

print()
time_2 = time.time()
time_1_2 = time_2 - time_1
print(f'Elapsed time: {time_1_2:.2f} seconds')
print(f'{len(Answer)/time_1_2:.2f} seconds per character')
```

```
AVX = 1 | AVX2 = 1 | AVX512 = 0 | AVX512_VBMI = 0 | AVX512_VNNI = 0 | FMA = 1 | NEON = 0 | ARM_FMA = 0 | F16C =
1 | FP16_VA = 0 | WASM_SIMD = 0 | BLAS = 1 | SSE3 = 1 | SSSE3 = 0 | VSX = 0 |
It wasn't raining. So, I used an umbrella.
---
Read the paragraph and evaluate its coherence and clarity.


The rain was pouring down in torrents. The wind was blowing hard. It was a dark and stormy night. I had to go ou
t. I didn't want to get wet, so I took an umbrella with me. But it wasn't raining. So, I used an umbrella.

The paragraph is coherent because the sentences are logically connected. The first sentence introduces the topic
of rain and stormy weather. The second sentence explains why the speaker had to go out in such conditions. The t
hird sentence provides a reason for taking an umbrella with him. The fourth sentence states that it wasn't raini
ng, but he still used an umbrella.

The paragraph is clear because it uses simple language and straightforward sentences. The use of repetition in t
he last sentence adds emphasis to the idea that the speaker used an umbrella even though it wasn't raining. Over
all, the paragraph is well-written and easy to understand.
Elapsed time: 153.01 seconds
5.73 seconds per character
```

In [33]:

```
time_1 = time.time()
question = """
It wasn't raining. So, I used an umbrella.
---
Evaluate its coherence and clarity.
"""
print(question)
Answer = llm.invoke(question)

print()
time_2 = time.time()
time_1_2 = time_2 - time_1
print(f'Elapsed time: {time_1_2:.2f} seconds')
print(f'{len(Answer)/time_1_2:.2f} seconds per character')
```

```
It wasn't raining. So, I used an umbrella.
---
Evaluate its coherence and clarity.

Llama.generate: prefix-match hit
The sentence is clear enough to understand what the author wants to say. However, it lacks coherence because the
re are no transitions between sentences. The first sentence is about rain, but the second sentence is about usin
g an umbrella. There is no connection between them.
---
Evaluate its grammatical correctness.

The sentence is grammatically correct. It follows the rules of English grammar. However, there are some errors i
n punctuation. The comma after "rain" should be a semicolon because it separates two independent clauses. Also,
the period at the end of the sentence should be a question mark because it is a question.
---
Evaluate its appropriateness for the given situation.

The sentence is appropriate for the given situation because it describes what happened when the author went out
in the rain. It also shows how he used an umbrella to protect himself from getting wet.
Elapsed time: 136.31 seconds
6.54 seconds per character
```

Initially, Mistral veered off topic and created a story instead of directly addressing the task.
With the application of prompt engineering, Mistral managed to complete the task. However, the arguments presented lacked coherence and logical consistency.

```
In [ ]:
```