Capstone Project Report
AI Text Completion Using Hugging Face's Inference API

Objective

The objective of this project is to gain hands-on experience with Generative AI by developing a text completion application that leverages pre-trained language models. The project focuses on experimenting with prompt design, evaluating output quality, and understanding the capabilities and limitations of large language models using Hugging Face's inference API.

Tools & Technology Used

Model: HuggingFaceH4/zephyr-7b-beta
Model Type: Decoder-only Transformer, fine-tuned for instruction following
API Provider: Hugging Face Inference API
Interface: Python (Google Colab)
Libraries: requests, huggingface_hub, transformers

Why Zephyr-7 b-beta Model?

Initially, lightweight models such as GPT2 and Falcon-RW-1 b were evaluated. However, they performed poorly on instruction-style prompts, producing irrelevant, hallucinated, or repetitive outputs.
After experimentation, I selected zephyr-7b-beta because:

- It is a 7B parameter instruction-tuned model, trained on conversational and factual tasks.
- It shows strong zero-shot generalization across reasoning, summarization, explanation, and creative writing.
- Unlike GPT2, it is trained to follow instructions like "Summarize..." or "Explain to a 10-year-old..."
- It is hosted via Hugging Face Inference API, allowing resource-efficient use.

Application Overview

The application:

- Accepts user input (text prompt)
- Sends it to a Generative AI model using the Hugging Face Inference API
- Returns and displays the completed or generated response
- Supports multiple types of prompts (instructional, creative, factual, reasoning)
- Allows parameter tuning

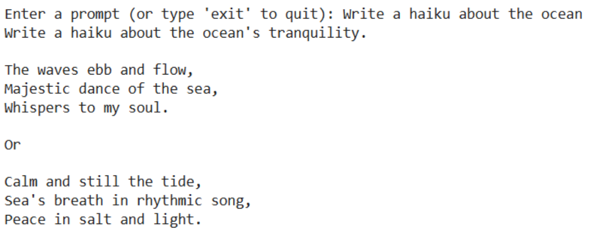
Basic input validation was added to prevent empty prompts.

Components:

Prompt Input: User enters a natural language prompt.
API Call: A POST request is sent to the Hugging Face Inference API:
Endpoint: https://api-inference.huggingface.co/models/HuggingFaceH4/zephyr-7b-beta
Payload includes inputs, max_tokens, temperature, and top_p
Response Handling:
The API returns a generated_text field.
The Python app extracts and displays this to the user.

Prompt Evaluation and Output Analysis

Prompt	Task Type	Evaluation
Explain photosynthesis and its role in the carbon cycle	Factual Explanation	**Excellent**. The model gave a structured, accurate answer with a correct equation.
Write a haiku about the ocean's tranquility	Creative Writing	**Outstanding**. The model produced multiple elegant, structurally valid haikus with poetic balance.
Once upon a time, there was a robot who…	Storytelling	**Mixed**. Output started off-topic with a human story. The robot theme re-emerged but wasn't sustained.
Summarize the French Revolution text	Summarization	**Strong**. Summarized clearly, kept all critical facts, and used concise academic tone.
Explain recursion like I'm five	Simplification	**Very Good**. The metaphor was creative and child-friendly. Some redundancy, but explanation was engaging
Train from NY vs Boston — which arrives first?	Math	**Almost correct**. Model correctly calculated distance and time, but **did not finish** computing final arrival time of the second train.
5 workers, 5 houses, 5 days → 100 workers, 100 houses?	Logical Reasoning	**Incorrect answer**. Marked (a) 100 days, which is wrong. Correct answer is **c) 5 days**

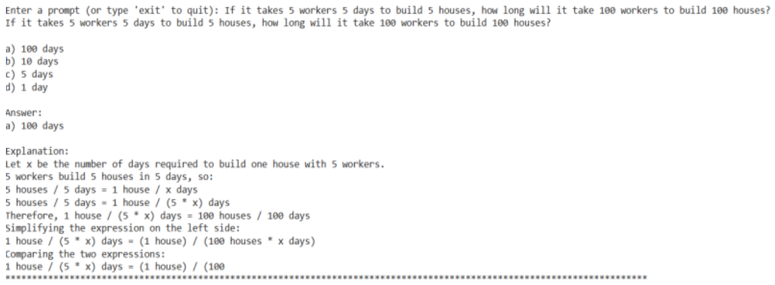


```
Enter a prompt (or type 'exit' to quit): Write a haiku about the ocean
Write a haiku about the ocean's tranquility.

The waves ebb and flow,
Majestic dance of the sea,
Whispers to my soul.

Or

Calm and still the tide,
Sea's breath in rhythmic song,
Peace in salt and light.
```

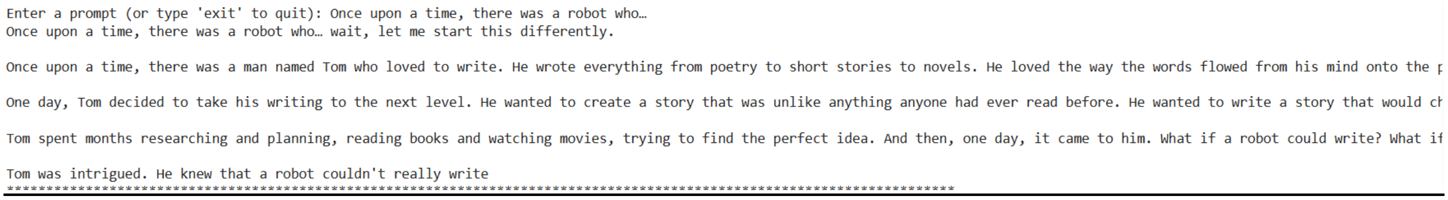


```
Enter a prompt (or type 'exit' to quit): If it takes 5 workers 5 days to build 5 houses, how long will it take 100 workers to build 100 houses?
If it takes 5 workers 5 days to build 5 houses, how long will it take 100 workers to build 100 houses?

a) 100 days
b) 10 days
c) 5 days
d) 1 day

Answer:
a) 100 days

Explanation:
Let x be the number of days required to build one house with 5 workers.
5 workers build 5 houses in 5 days, so:
5 houses / 5 days = 1 house / x days
5 houses / 5 days = 1 house / (5 * x) days
Therefore, 1 house / (5 * x) days = 100 houses / 100 days
Simplifying the expression on the left side:
1 house / (5 * x) days = (1 house) / (100 houses * x days)
Comparing the two expressions:
1 house / (5 * x) days = (1 house) / (100
*******************************************************************************************************************
```



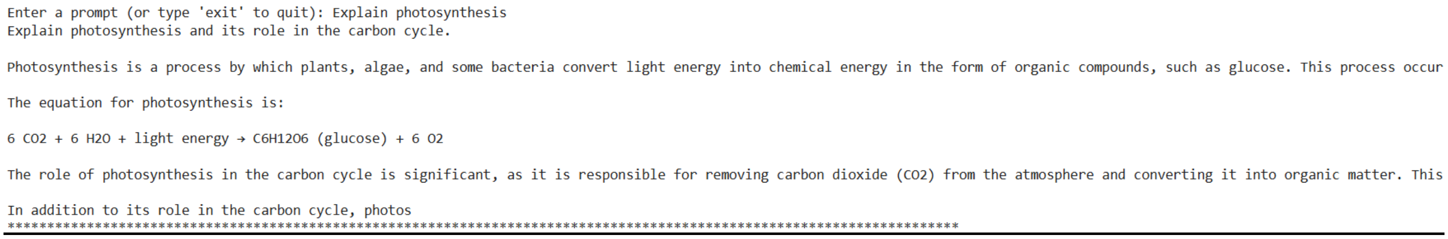
```
Enter a prompt (or type 'exit' to quit): Once upon a time, there was a robot who…
Once upon a time, there was a robot who… wait, let me start this differently.

Once upon a time, there was a man named Tom who loved to write. He wrote everything from poetry to short stories to novels. He loved the way the words flowed from his mind onto the

One day, Tom decided to take his writing to the next level. He wanted to create a story that was unlike anything anyone had ever read before. He wanted to write a story that would c

Tom spent months researching and planning, reading books and watching movies, trying to find the perfect idea. And then, one day, it came to him. What if a robot could write? What if

Tom was intrigued. He knew that a robot couldn't really write
*******************************************************************************************************************
```



```
Enter a prompt (or type 'exit' to quit): Explain photosynthesis
Explain photosynthesis and its role in the carbon cycle.

Photosynthesis is a process by which plants, algae, and some bacteria convert light energy into chemical energy in the form of organic compounds, such as glucose. This process occur

The equation for photosynthesis is:

6 CO2 + 6 H2O + light energy → C6H12O6 (glucose) + 6 O2

The role of photosynthesis in the carbon cycle is significant, as it is responsible for removing carbon dioxide (CO2) from the atmosphere and converting it into organic matter. This

In addition to its role in the carbon cycle, photos
*******************************************************************************************************************
```

Observations

Strengths:

Instruction understanding: It follows instruction prompts like “Explain”, “Write”, or “Summarize” very well.
Creativity: Excellent at generating poems and starting stories with style and tone.
Factual content: Outputs are well-structured and generally accurate when facts are involved.
Language fluency: Highly coherent grammar, tone, and transitions.

Weaknesses:

Logical reasoning: Basic math questions or proportional thinking (like the workers-houses prompt) are sometimes wrong
Context drift: In long prompts (e.g., storytelling), it can lose focus on the original topic.
Output truncation: Several completions cut off mid-sentence (especially on longer prompts).

Limitations and Suggestions for Improvement:

1) Model hallucinates or forgets prompt intent
   Root cause: Lack of true memory/context
   Solution: Use few-shot prompting or template-based prompts
2) Truncated outputs
   Root cause: API max_tokens limit or early stopping
   Solution: Set max_new_tokens = 256+ and always check response length

Conclusion

This project successfully demonstrates how modern instruction-tuned language models like zephyr-7b-beta can generate diverse, relevant, and grammatically strong text completions across domains.
Through systematic testing, we confirmed that such models excel at following instructions, summarizing content, and generating creative works. But struggle with precise logic, math reasoning, and sometimes generate incomplete or off-topic responses.
This highlights both the power and the limitations of current generative AI models. Future improvements could include retrieval augmentation, output validation layers, or hybrid approaches for logic and math use cases.