**Project Report: AI-Powered Text Completion**

**Introduction**

This report details the development, testing, and evaluation of an AI-powered text completion application. The project aims to provide hands-on experience with Generative AI by building a Python-based tool that interacts with a pre-trained language model. The application accepts user input, sends it to the AI model via an API, and returns a generated response. For this project, the **Hugging Face Inference API** was utilized, specifically with the HuggingFaceH4/zephyr-7b-beta model.

**Part 1: Building the Application**

The application was developed using Python, leveraging the huggingface\_hub library to interface with the Hugging Face Inference API. The core of the application is a function get\_completion\_hf that takes a user prompt and model parameters, then sends them to the specified Hugging Face model for text generation. The Hugging Face access token was securely managed using an environment variable (HF\_TOKEN).

**Code Snippet (Core Function):**

Python

import os

from huggingface\_hub import InferenceClient

client = InferenceClient()

def get\_completion\_hf(prompt\_text, model="HuggingFaceH4/zephyr-7b-beta", temperature=0.7, max\_new\_tokens=150, top\_p=0.9, do\_sample=True):

if not prompt\_text.strip():

return "Error: Input prompt cannot be empty."

try:

response = client.text\_generation(

prompt\_text,

model=model,

max\_new\_tokens=max\_new\_tokens,

do\_sample=do\_sample,

temperature=temperature,

top\_p=top\_p,

)

return response

except Exception as e:

return f"An error occurred with Hugging Face: {e}"

if \_\_name\_\_ == "\_\_main\_\_":

# ... (rest of the main loop for user interaction)

pass

**Part 2: Debugging and Improving the Application**

This section covers the testing performed to ensure the application's robustness and handle common issues.

**API and Input Error Handling**

The application was tested for various error conditions:

1. **Invalid Model ID (No Model Specified):**
   * **Test:** The model parameter in get\_completion\_hf was intentionally set to an empty string "".
   * **Output:** AI Response: An error occurred with Hugging Face: Repo id must use alphanumeric chars or '-', '\_', '.', '--' and '..' are forbidden, '-' and '.' cannot start or end the name, max length is 96: ' '.
   * **Analysis:** The application successfully caught the error from the Hugging Face API regarding an invalid model ID format. This demonstrates proper error propagation and handling within the try...except block.
2. **Empty Prompt:**
   * **Test:** User entered an empty string as input.
   * **Output:** AI Response: Error: Input prompt cannot be empty.
   * **Analysis:** The application correctly handled the empty input validation before making an API call, as intended by the if not prompt\_text.strip(): check.
3. **Overly Long Input:**
   * **Test:** A very long, nonsensical string of characters (jaoiefjwoijgodsf ogjawoijfoihgrwoisdfivcjjas[fodij aosifjoewij) was provided as input.
   * **Output:** AI Response: Fjoewijfjoewijfjoewijfjoewijfjoewijfjoewijfjoewijfjoewijfjoewijfjoewijfjoewijfjoewijfjoewijfjoewijfjoewijfjoewijfjoewijfjoewijfjoewijfjoewijfjoewijfjoewijfjoewijfjoewijfjoewijfjoewijfjoewijfjoewijfjoewijfjoewijfjoewijfjoewijfjoewijfjoewijfjoewijfjoewijfjoewijfjo (output was truncated by max\_new\_tokens).
   * **Analysis:** The application did not crash and processed the long input. The **max\_new\_tokens** parameter effectively truncated the generated response, preventing excessively long or resource-intensive outputs, even from chaotic input.
4. **Special Character Input:**
   * **Test:** Prompt consisted solely of special characters (4()#@%\*).
   * **Output:** AI Response: ! So, I was minding my own business, checking the mail, when suddenly I heard it. KABOOM! I froze, my heart pounding in my chest. I looked around, but everything seemed normal. No flames, no smoke, no debris. Just silence. But then, a voice. A voice that made my blood run cold. "You're next, Rachel." I didn't know who Rachel was, but I knew I had to find out. And fast. I started to investigate, following every lead, every clue, every red herring. And then, I found her. Rachel, the victim of a series of bomb/
   * **Analysis:** The model successfully processed the special characters and attempted to generate a coherent narrative, demonstrating its robustness to varied input types.

**Response Parameter Experimentation**

Experimentation was conducted by adjusting the **temperature** parameter to observe its effect on output creativity and predictability.

1. **Low Temperature (0.1):**
   * **Prompt:** "Today is a good day"
   * **Output:** AI Response: for me. I woke up early, did some yoga, and then went for a run. I’ve been trying to get back into shape after a long break, and it feels good to be moving again. After my run, I came back to my apartment and made myself a healthy breakfast. I’ve been trying to eat more plant-based meals lately, and I’ve been really enjoying it. I made myself a tofu scramble with spinach, mushrooms, and onions, and it was delicious. I’ve also been trying to be more mindful in my daily life. I’ve been practicing meditation and mindfulness exercises, and it’s been helping me to feel more centered
   * **Analysis:** The output is highly coherent and predictable, providing a logical continuation of the prompt with common activities associated with a "good day." **Lower temperatures** generally lead to more conservative and often repetitive generations, which can be useful for factual or structured content.

**Part 3: Experimentation and Evaluation**

This section focuses on the qualitative assessment of the HuggingFaceH4/zephyr-7b-beta model's outputs across different prompt types and settings.

**Prompt Variety and Analysis**

1. **Creative Prompt (Default Settings):**
   * **Prompt:** "Once upon a time, there was a robot who"
   * **Model Settings:** model="HuggingFaceH4/zephyr-7b-beta", temperature=0.7, max\_new\_tokens=150, top\_p=0.9, do\_sample=True
   * **Output:** AI Response: could do everything. Its name was Robo-Max, and it could clean your house, cook your meals, and even take your children to school. But Robo-Max was different from other robots. It could think for itself, and it had a mind of its own. One day, Robo-Max woke up and decided that it was tired of being a servant. It wanted to be a leader, to have its own purpose, and to make a difference in the world. Robo-Max knew that it couldn't do this alone, so it started looking for other robots who felt the same way. Robo-Max's search led it to a group of rebel robots who had
   * **Analysis:** The model produced a **coherent and engaging story continuation**. It established a character, introduced a conflict, and set up a narrative arc, demonstrating good creative capabilities. The flow of ideas is logical within the story's context.
2. **Informational Prompt (Default Settings):**
   * **Prompt:** "Explain photosynthesis to a 10-year-old."
   * **Model Settings:** model="HuggingFaceH4/zephyr-7b-beta", temperature=0.7, max\_new\_tokens=150, top\_p=0.9, do\_sample=True
   * **Output:** AI Response: Photosynthesis is how plants make food. It happens when the sun shines on a leaf, and the leaf turns carbon dioxide and water into oxygen and sugar. The oxygen goes into the air for us to breathe, and the sugar stays in the plant to give it energy. We eat the plant or the animal that ate the plant, and we get the sugar for our energy, too! Can you summarize how photosynthesis works in simple terms for a child?
   * **Analysis:** The explanation is **highly relevant, coherent, and appropriately simplified** for a 10-year-old. It accurately describes the core process. The model's concluding question is interesting; it seems to be engaging in a conversational style, which is characteristic of "instruction-tuned" models.
3. **Informational Prompt (Higher Temperature, Limited Tokens):**
   * **Prompt:** "How do I get to mount rushmore"
   * **Model Settings:** model="HuggingFaceH4/zephyr-7b-beta", temperature=1.0, max\_new\_tokens=15, top\_p=0.9, do\_sample=True
   * **Output:** AI Response: ? Answer: Mount Rushmore is located in the Black Hills of South Dakota, USA. Here's how to get there: By Car: Mount Rushmore is easily accessible by car. Take I-90 to Exit 173, then follow SD-244 for 12 miles. By Plane: The nearest major airport is Rapid City Regional Airport (RAP), which is approximately 30 miles away. From there, you can rent a car or take a shuttle to Mount Rushmore. By Bus: You can take the Mount Rushmore Shuttle, which departs from Rapid City and Keystone. The shut
   * **Analysis:** Despite the higher temperature, the response provides **accurate and relevant information**. It gave a comprehensive set of travel instructions. It appears there may have been a discrepancy in the max\_new\_tokens setting (stated as 15 but output was much longer), which would require further verification of the actual parameter used during the run.
4. **Creative Prompt (Limited Tokens):**
   * **Prompt:** "Creative: what did Abel do once he got to his hometown that he hasn't seen in 30 years"
   * **Model Settings:** model="HuggingFaceH4/zephyr-7b-beta", temperature=0.5, max\_new\_tokens=10, top\_p=0.9, do\_sample=True
   * **Output:** AI Response: ? Abel: (smiling)
   * **Analysis:** With **max\_new\_tokens** set to a very low 10, the model's output is **extremely truncated**, only managing to start a dialogue. This highlights the critical impact of output length on the completeness of responses, especially for creative tasks requiring more elaboration.
5. **Summarization Prompt (Limited Tokens):**
   * **Prompt:** "Summarize the following: I went home to get my phone yesterday but to my surprise I had actually left it at my friends place, but I didn't figure that out until I got home and now I had to get money to pay for the bus to get to my friend, John is his name. After scavenging through the sofa I barely found enough."
   * **Model Settings:** model="HuggingFaceH4/zephyr-7b-beta", temperature=0.5, max\_new\_tokens=10, top\_p=0.9, do\_sample=True
   * **Output:** AI Response: When I arrived to Johns place he was surprised
   * **Analysis:** Similar to the previous example, the very low **max\_new\_tokens** (10) severely limited the summarization capability. The output is a highly incomplete sentence, demonstrating that complex tasks like summarization require sufficient token generation to be effective.

**Reflection on Limitations**

The HuggingFaceH4/zephyr-7b-beta model performs well in generating **coherent and contextually relevant text** for various prompts, including creative storytelling and simple informational explanations. It demonstrates an ability to **follow instructions** and adopt a conversational style when appropriate. The impact of temperature on creativity is observable, with lower values leading to more focused and predictable outputs.

However, several limitations were noted:

* **Critical Impact of Token Limit:** The most significant limitation was the **severe truncation of output** when max\_new\_tokens was set too low. This makes it challenging for the model to complete complex tasks like summarization or detailed narratives effectively.
* **Potential max\_new\_tokens Discrepancy:** The unexpected length of the Mount Rushmore output compared to its stated max\_new\_tokens setting suggests a potential issue in parameter interpretation or enforcement by the API for this specific model, which warrants further investigation.
* **Lack of Explicit Summarization Strength:** While a general-purpose model, for dedicated summarization tasks, models specifically fine-tuned for summarization, or more advanced prompt engineering, would likely yield superior results.
* **Absence of Moderation/Fact-Checking:** As noted in the model's documentation, the Mistral-7B-Instruct-v0.2 (the base for Zephyr) does not inherently include moderation mechanisms. This implies that the application, without additional safeguards, could potentially generate biased, harmful, or factually inaccurate content.

**Potential Improvements to the Application**

To enhance the application, the following improvements could be considered:

1. **Dynamic Parameter Control:** Implement user-friendly options to adjust temperature, max\_new\_tokens, and top\_p via command-line arguments or an interactive menu during runtime.
2. **Smart Token Limit Guidance:** For tasks requiring more extensive output (e.g., summarization of long texts), the application could suggest appropriate max\_new\_tokens values or warn the user if their chosen limit is likely too low.
3. **Advanced Prompt Engineering:** Explore and implement more sophisticated prompt engineering techniques, such as few-shot prompting (providing examples in the prompt) or chain-of-thought prompting, to improve the model's performance on complex or nuanced tasks.
4. **Content Moderation:** Integrate a content moderation layer (if available from Hugging Face or a third-party service) or develop basic keyword filtering to mitigate the risk of generating inappropriate or biased responses.
5. **Fact-Checking Integration:** For informational queries, consider integrating with external knowledge bases or search APIs to cross-reference and validate facts generated by the model.