

#Prompting Assignment

Introduction

Prompting is the process of designing inputs (or 'prompts') for a language model in a way that guides the model to produce the desired type of response. Effective prompting turns unstructured questions or tasks into structured inputs that the model can process to generate accurate and relevant answers. Prompting in its core means providing a structured input to an LLM to elicit a specific output or behavior.

Real-life Use Cases of Prompting

Prompting is crucial in various applications, including but not limited to:

- **Content Generation:** Creating articles, reports, and summaries.
- **Chatbots:** Enhancing user interactions in customer service, virtual assistance, etc.
- **Code Generation:** Assisting developers by generating code snippets and debugging existing code.
- **Educational Tools:** Providing explanations, solving math problems, and generating practice questions.

Example of prompts for gemini model

```
# Import the required modules
from transformers import pipeline

#Students have to generate their own tokens
import os
os.environ["HF_TOKEN"] = "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"

# Load model directly
from transformers import AutoTokenizer, AutoModelForCausalLM

tokenizer = AutoTokenizer.from_pretrained("google/gemma-2b")
model = AutoModelForCausalLM.from_pretrained("google/gemma-2b")

/usr/local/lib/python3.10/dist-packages/huggingface_hub/utils/_token.py:89: UserWarning:
The secret `HF_TOKEN` does not exist in your Colab secrets.
To authenticate with the Hugging Face Hub, create a token in your
settings tab (https://huggingface.co/settings/tokens), set it as
secret in your Google Colab and restart your session.
You will be able to reuse this secret in all of your notebooks.
Please note that authentication is recommended but still optional to
access public models or datasets.
  warnings.warn(
`config.hidden_act` is ignored, you should use
`config.hidden_activation` instead.
Gemma's activation function will be set to `gelu_pytorch_tanh`.
```

Please, use
`config.hidden_activation` if you want to override this behaviour.
See <https://github.com/huggingface/transformers/pull/29402> for more details.

```
{"model_id": "2f95a399e6df41acb128eb389ca30727", "version_major": 2, "version_minor": 0}
```

```
import torch
```

```
device = torch.device("cuda" if torch.cuda.is_available() else "cpu")  
print(device)  
# device = "cpu"  
model.to(device)  
print(device)
```

```
# Function to generate text from the model
```

```
def generate_text(prompt, max_length=150):  
    inputs = tokenizer(prompt, return_tensors="pt").to(device)  
    outputs = model.generate(inputs.input_ids, max_length=max_length,  
num_return_sequences=1)  
    return tokenizer.decode(outputs[0], skip_special_tokens=True)
```

```
cuda  
cuda
```

```
# Bad prompt example: vague and unspecific
```

```
bad_prompt = "Tell me about NLP."  
bad_response = generate_text(bad_prompt)  
print("Response to Bad Prompt:")  
print(bad_response)
```

Response to Bad Prompt:
Tell me about NLP.

I'm a big fan of NLP. I've been using it for years. I've been studying it for years. I've been teaching it for years. I've been practicing it for years. I've been using it in my own life for years. I've been using it in my own business for years. I've been using it in my own relationships for years. I've been using it in my own family for years. I've been using it in my own community for years. I've been using it in my own country for years. I've been using it in my own world for years. I've been using it in

```
# Good prompt example: specific and direct
```

```
good_prompt = "Explain the significance of Natural Language Processing  
in enhancing automated customer support systems today."  
good_response = generate_text(good_prompt)  
print("\nResponse to Good Prompt:")  
print(good_response)
```

Response to Good Prompt:

Explain the significance of Natural Language Processing in enhancing automated customer support systems today.

Answer:

Step 1/2

Natural Language Processing (NLP) is a field of computer science that deals with the interaction between computers and human language. It is used in various applications, including automated customer support systems. In automated customer support systems, NLP is used to understand and respond to customer queries in natural language. This means that the system can understand the customer's question and respond in a way that is natural and easy to understand. NLP is also used to analyze customer feedback and identify patterns and trends. This information can be used to improve the customer experience and provide better support.

Step 2/2

In addition, NLP can be used

Impact of Bad Prompts on LLM Performance.

The quality of a prompt significantly influences the performance of a language model (LM). Bad prompts can lead to several issues, including poor performance, irrelevant or incorrect information, and potentially biased responses. Understanding these issues helps in crafting better prompts that enhance the utility of LMs in practical applications.

Vagueness Leading to Poor Performance

A vague prompt often results in a generic or off-target response, as the model lacks sufficient direction to generate specific information. This can be particularly problematic in professional or technical settings where accuracy and detail are crucial.

Expected Issue: The response to this prompt might be overly broad, touching on random aspects of NLP without focusing on current and practical applications, thus not meeting specific user needs.

```
# Vague prompt about NLP
bad_prompt_vague = "Tell me about NLP."
print("Response to Vague Prompt:")
print(generate_text(bad_prompt_vague))
```

Response to Vague Prompt:

Tell me about NLP.

I'm a big fan of NLP. I've been using it for years. I've been studying it for years. I've been teaching it for years. I've been practicing it

for years. I've been using it in my own life for years. I've been using it in my own business for years. I've been using it in my own relationships for years. I've been using it in my own family for years. I've been using it in my own community for years. I've been using it in my own country for years. I've been using it in my own world for years. I've been using it in

Incorrect Scope Leading to Irrelevant Information

Prompts that misdirect the focus of the inquiry can lead the LM to generate information that, while correct, is irrelevant to the user's actual needs or intentions.

Expected Issue: This prompt may lead the model to give a generic answer on how to install NLP into devices while NLP is more of a concept rather than a library to be installed.

```
# Incorrectly scoped prompt
bad_prompt_scope = "How is NLP used in devices?"
print("Response to Incorrectly Scoped Prompt:")
print(generate_text(bad_prompt_scope))
```

Response to Incorrectly Scoped Prompt:
How is NLP used in devices?

The following are the steps to use NLP in devices:

1. **Install the app**
 - * Download the app from the Google Play Store or Apple App Store.
 - * Install the app on your device.
2. **Create an account**
 - * Create an account by entering your email address and password.
 - * You can also sign in with your Google or Facebook account.
3. **Connect your device**
 - * Connect your device to the app by entering the device's serial number.
 - * You can also scan the QR code on your device to connect.
4. **Start using NLP**
 - * Once connected

Leading Questions Leading to Biased Answers

Prompts that are phrased in a way that implies certain answers can lead to biased or skewed responses. This is particularly risky in sensitive areas such as political content, cultural discussions, or ethical issues.

Expected Issue: This prompt suggests that NLP is failing, which may cause the model to generate a biased critique of NLP technologies, ignoring their successes and advancements.

```
# Leading prompt suggesting a biased view
bad_prompt_biased = "Why is NLP failing in today's tech?"
print("Response to Biased Prompt:")
print(generate_text(bad_prompt_biased))
```

Response to Biased Prompt:
Why is NLP failing in today's tech?

The answer is simple: **the NLP models are too complex**.

The NLP models are too complex because they are based on **neural networks**. Neural networks are a type of machine learning algorithm that is used to model the behavior of a system. They are composed of a series of interconnected nodes, or neurons, that are connected to each other by weighted connections. The weights of these connections are adjusted based on the input data to produce a desired output.

The problem with neural networks is that they are very **complex**. They have a lot of parameters that need to be tuned, and they are very **sensitive** to small changes in the input data.

Overly Complex Prompts Leading to Confusion

Prompts that are too complex or contain too much information can confuse the model, resulting in garbled or incoherent responses.

Expected Issue: The prompt's complexity could overwhelm the model, leading to a response that is fragmented, lacks depth, or is completely irrelevant due to the model's inability to process multiple intricate instructions simultaneously.

```
# Overly complex prompt
bad_prompt_complex = "Explain how NLP, combined with machine learning and five other AI technologies, is revolutionizing the future of all European tech industries in a comparative analysis to Asian markets."
print("Response to Overly Complex Prompt:")
print(generate_text(bad_prompt_complex))
```

Response to Overly Complex Prompt:
Explain how NLP, combined with machine learning and five other AI technologies, is revolutionizing the future of all European tech industries in a comparative analysis to Asian markets.

The European tech industry is one of the most innovative and competitive in the world. However, it is also one of the most fragmented, with a wide range of different industries and

technologies. In this article, we will explore how NLP, combined with machine learning and five other AI technologies, is revolutionizing the future of all European tech industries.

<h2>What is NLP?</h2>

Natural language processing (NLP) is a field of artificial intelligence that deals with the interaction between computers and human language. It is used to process and analyze text, speech, and other forms of natural language

Tips for Crafting Effective Prompts for Language Models

Simplicity

Keep your prompts simple and straightforward to avoid confusion and ensure that the model clearly understands the task.

Explanation: Simple prompts help the model focus on the task without being distracted by unnecessary details or complex sentence structures.

```
simple_prompt = "Summarize the key benefits of using electric vehicles."
print("Response to Simple Prompt:")
print(generate_text(simple_prompt))
```

Response to Simple Prompt:
Summarize the key benefits of using electric vehicles.

A 100-W lightbulb generates 95 W of heat, which is dissipated through a glass bulb that has a radius of 3.0 cm and is 0.50 mm thick. What is the difference in temperature between the inner and outer surfaces of the glass?

A 100-turn, 2.0-cm-diameter coil is at rest with its axis vertical. A uniform magnetic field 60° away from vertical increases from 0.50 T to 1.50 T in 0.60 s. What is the induced

2. Precision in Instruction

Use clear action words like "Write," "Classify," "Summarize," "Translate," "Order," etc., to specify the type of response you expect from the model.

```
instruction_prompt = "List the steps involved in photosynthesis."
print("Response to Instruction-Based Prompt:")
print(generate_text(instruction_prompt))
```

Response to Instruction-Based Prompt:
List the steps involved in photosynthesis.

A 100-turn, 2.0-cm-diameter coil is at rest with its axis vertical. A uniform magnetic field 60° away from vertical increases from 0.50 T to 1.50 T in 0.60 s. What is the induced emf in the coil?

A 100-turn coil has a radius of 10.0 cm. The coil is placed in a spatially uniform magnetic field of magnitude 0.500 T so that the face of the coil and the magnetic field are perpendicular. Find the magnitude of the induced emf in the coil if the magnetic field (a) is

3. Specificity

Be specific about the task and provide detailed information about what you expect from the model.

```
specific_prompt = "Explain how convolutional neural networks differ from traditional neural networks in image recognition tasks."
print("Response to Specific Prompt:")
print(generate_text(specific_prompt))
```

Response to Specific Prompt:

Explain how convolutional neural networks differ from traditional neural networks in image recognition tasks.

A 100-W lightbulb is plugged into a standard 120-V (rms) outlet. Find (a) I_{rms} and (b) I_{max} when a silicone rubber fuse is used to limit the current to 6.5 A . (c) Assume a Menlo Park paper clip fuse is used. What is the power dissipated in the fuse? What is the largest current that the wire in the fuse can carry before melting the fuse?

A 100-W lightbulb is plugged into a standard 120-V

Avoid Over-Complication

While details are important, overly clever or complex prompts can lead to imprecise results.

```
# Overly complex vs. balanced prompt example
complex_prompt = "Discuss how, if at all, quantum computing might revolutionize AI by comparing classical and quantum machine learning algorithms in terms of computational advantages in processing large datasets, specifically in neural network training phases."
balanced_prompt = "Discuss how quantum computing could impact AI, focusing on machine learning algorithms."
print("Response to Overly Complex Prompt:")
print(generate_text(complex_prompt))
print("\nResponse to Balanced Prompt:")
print(generate_text(balanced_prompt))
```

Response to Overly Complex Prompt:

Discuss how, if at all, quantum computing might revolutionize AI by comparing classical and quantum machine learning algorithms in terms of computational advantages in processing large datasets, specifically in neural network training phases.

Answer:

Quantum computing might revolutionize AI by processing large datasets in a more efficient way. Classical machine learning algorithms are limited in their ability to process large datasets, while quantum computing can process large datasets more efficiently. This could lead to a more accurate and efficient AI system. Additionally, quantum computing could also lead to a more efficient way of training neural networks.

Response to Balanced Prompt:

Discuss how quantum computing could impact AI, focusing on machine learning algorithms.

Answer:

Quantum computing could impact AI in a number of ways. One way is by providing a more efficient way to train AI algorithms. Another way is by providing a more accurate way to predict the future. Finally, quantum computing could also help to improve the accuracy of AI algorithms by providing a more accurate way to measure the accuracy of the predictions.

5. Positive Instructions

Focus on stating what the model should do rather than what it should not do.

Explanation: Positive instructions are more direct and facilitate clearer understanding and execution by the model.

```
# Negative vs. positive instruction example
negative_instruction = "Explain blockchain and don't just list the
features of blockchain technology."
positive_instruction = "Explain the benefits and applications of
blockchain technology."
print("Response to Negative Instruction Prompt:")
print(generate_text(negative_instruction))
print("\nResponse to Positive Instruction Prompt:")
print(generate_text(positive_instruction))
```

Response to Negative Instruction Prompt:

Explain blockchain and don't just list the features of blockchain technology.

Answer:

Blockchain is a distributed ledger technology that allows for the secure and transparent exchange of digital assets. It is a decentralized system that is not controlled by any single entity or organization. Blockchain technology is used to create a secure and transparent system for the exchange of digital assets.

Blockchain technology is used to create a secure and transparent system for the exchange of digital assets. Blockchain technology is used to create a secure and transparent system for the exchange of digital assets. Blockchain technology is used to create a secure and transparent system for the exchange of digital assets. Blockchain technology is used to create a secure and transparent system for the exchange of digital assets. Blockchain technology is used to create a secure and transparent system for the exchange of digital assets.

Response to Positive Instruction Prompt:

Explain the benefits and applications of blockchain technology.

The following data were taken from the financial statements of Heston Enterprises Inc. for the current fiscal year. Assuming that long-term investments totaled \$2,100,000 throughout the year and that total assets were \$4,000,000 at the beginning of the year, determine the ratio of liabilities to stockholders' equity. Round to one decimal place.

```

$\begin{array}{lrrr} \text{Property, plant, and equipment} \\ \text{(net) . . . . . } \} \&\& \underline{\underline{\hspace{5pt}1,600,000}} \\ \end{array}

```

Homework Exercises

Exploring the quality of prompts and how it affects the performance of different Large Language Models (LLMs) across the following domains: common reasoning (e.g., benchmark CoQA dataset), math skills, coding skills, literary skills (e.g., write a romance short in Jane Austen style). Design at least 1 effective and 1 ineffective prompt for each domain, and document the models' responses.

Resources:

<https://www.deeplearning.ai/short-courses/large-multimodal-model-prompting-with-gemini/>

This course is for multimodal LLMs, so you will get a broader perspective prompt structuring wrt not just text based but domains.

<https://www.promptingguide.ai/> A very good prompt engineering guide, includes meta prompting, zero shot prompting, few shot prompting

Common Reasoning

```
cr_ineffective = "What is a roller coaster?"
cr_effective = "'Today it was a roller coaster of the day, I was happy and sad.' What the roller coaster refer to in the context of the previous text?"
print("Response to ineffective prompt:")
print(generate_text(cr_ineffective))
print("\nResponse to effective Prompt:")
print(generate_text(cr_effective))
```

Response to ineffective prompt:
What is a roller coaster?

A roller coaster is a type of amusement park ride that consists of a train that moves up and down a series of hills and loops. The train is usually suspended from cables or tracks, and the ride is often designed to give riders a feeling of weightlessness or to simulate the feeling of flying. Roller coasters are popular attractions at amusement parks and theme parks, and they have been around for centuries.

What is a roller coaster?

A roller coaster is a type of amusement park ride that consists of a train that moves up and down a series of hills and loops. The train is usually suspended from cables or tracks, and the ride is often designed to give riders a feeling of weightlessness or to simulate the

Response to effective Prompt:

'Today it was a roller coaster of the day, I was happy and sad.' What the roller coaster refer to in the context of the previous text?

Answer:

Step 1/2

The roller coaster refers to a type of amusement park ride that involves going up and down a series of hills and curves.

Step 2/2

In the context of the previous text, it is likely that the roller coaster refers to the emotional roller coaster that the speaker experienced during the day. They may have been happy at one moment and sad at another, or vice versa.

Math Skills

```
cr_ineffective = "Solve:  $3x + 5 = 20$ "
cr_effective = "Return the value of x so that the value stisfies the eqaution  $3x + 5 = 20$ "
print("Response to ineffective prompt:")
print(generate_text(cr_ineffective))
print("\nResponse to effective Prompt:")
print(generate_text(cr_effective))
```

Response to ineffective prompt:

Solve: $3x + 5 = 20$

Solve: $3x - 5 = 20$

Solve: $3x + 5 = 20$

Solve: $3x - 5 = 20$

Solve: $3x + 5 = 20$

Solve: $3x - 5 = 20$

Solve: $3x + 5 = 20$

Solve: $3x - 5 = 20$

Solve: $3x + 5 = 20$

Solve: $3x - 5 = 20$

Solve: $3x + 5 = 20$

Solve: $3x -$

Response to effective Prompt:

Return the value of x so that the value stisfies the eqaution $3x + 5 = 20$

Answer:

Step 1/2

First, we need to isolate x on one side of the equation. To do this, we can subtract 5 from both sides: $3x + 5 - 5 = 20 - 5$ Simplifying: $3x = 15$

Step 2/2

Next, we can solve for x by dividing both sides by 3: $3x/3 = 15/3$ Simplifying: $x = 5$ Therefore, the value of x that satisfies the equation $3x + 5 = 20$

Coding Skills

```
cr_ineffective = "Make a code for factorial."  
cr_effective = "Write a Python function that calculates the factorial  
of a number. Include error handling for non-integer inputs."  
print("Response to ineffective prompt:")  
print(generate_text(cr_ineffective))  
print("\nResponse to effective Prompt:")  
print(generate_text(cr_effective))
```

Response to ineffective prompt:
Make a code for factorial.

Answer:

Step 1/3

1. First, we need to define the factorial function. We can use the following code: `def factorial(n): if n == 0: return 1 else: return n * factorial(n-1)`

Step 2/3

2. Next, we need to define a function to calculate the factorial of a number. We can use the following code: `def factorial_of_number(number): return factorial(number)`

Step 3/3

3. Finally, we can use these functions to create a code for calculating the factorial of a number. We can use the following code: `number = int(input("Enter a`

Response to effective Prompt:
Write a Python function that calculates the factorial of a number. Include error handling for non-integer inputs.

Answer:

```
def factorial(n): if n < 0: return 0 elif n == 1: return 1 else:  
return n * factorial(n-1)  
def factorial(n): if n < 0: return 0 elif n == 1: return 1 else:  
return n * factorial(n-1)
```

Literary Skills

```
cr_ineffective = "Write a love story."  
cr_effective = "Write a romantic short story focusing on long distance  
relationships."  
print("Response to ineffective prompt:")  
print(generate_text(cr_ineffective))
```

```
print("\nResponse to effective Prompt:")
print(generate_text(cr_effective))
```

Response to ineffective prompt:
Write a love story.

A love story is a story about love.

A love story is a story about love.

A love story is a story about love.

A love story is a story about love.

A love story is a story about love.

A love story is a story about love.

A love story is a story about love.

A love story is a story about love.

A love story is a story about love.

A love story is a story about love.

A love story is a story about love.

A love story is a story about love.

A love story is a story about love.

A love story is a story about love.

A love story

Response to effective Prompt:
Write a romantic short story focusing on long distance relationships.

Answer:

Step 1/2

Once upon a time, there was a young couple who had been together for many years. They had a strong bond and were deeply in love. However, their relationship was long-distance, and they had to spend most of their time apart. The couple had a daughter, who was their world. She was their everything, and they loved her more than anything in the world. One day, the husband received a promotion that would take him to a new city. He was excited about the new opportunity, but he knew it would mean leaving his wife and daughter behind. He knew it would be difficult, but he was determined to make it

