# Learning Outcomes

## By the end of this module, you will be able to:

- Define the prompt and prompt engineering concept;
- State the elements of a prompt;
- Explain how to design a prompt;
- Indicate how to set up your environment and install the necessary libraries;

# What is a Prompt ?

- Generative AI models interact with users primarily through textual input
- Textual Input is the input where the user provides a prompt to the model to accomplish a task.
- The prompt can range from a simple question to a complex problem with inserted data, or even a vague statement.

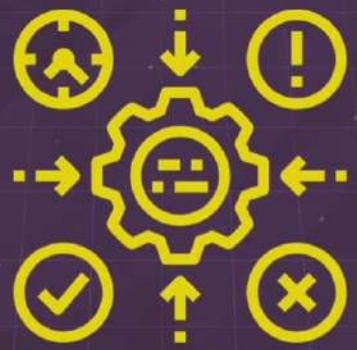
# Elements of a Prompt

A prompt for AI models typically contains four elements:

- Instructions
- Context
- Input data, and
- Output indicator

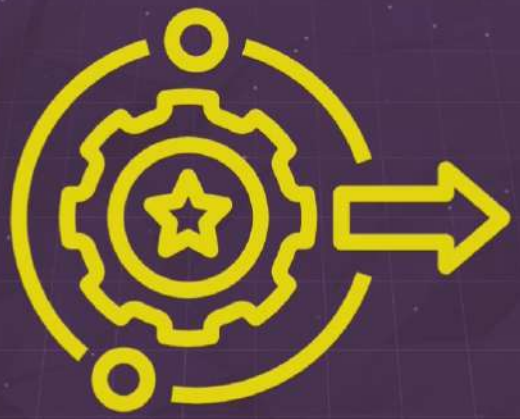
- The first element is an **Instruction** for the specific task or command to be performed by the AI mode

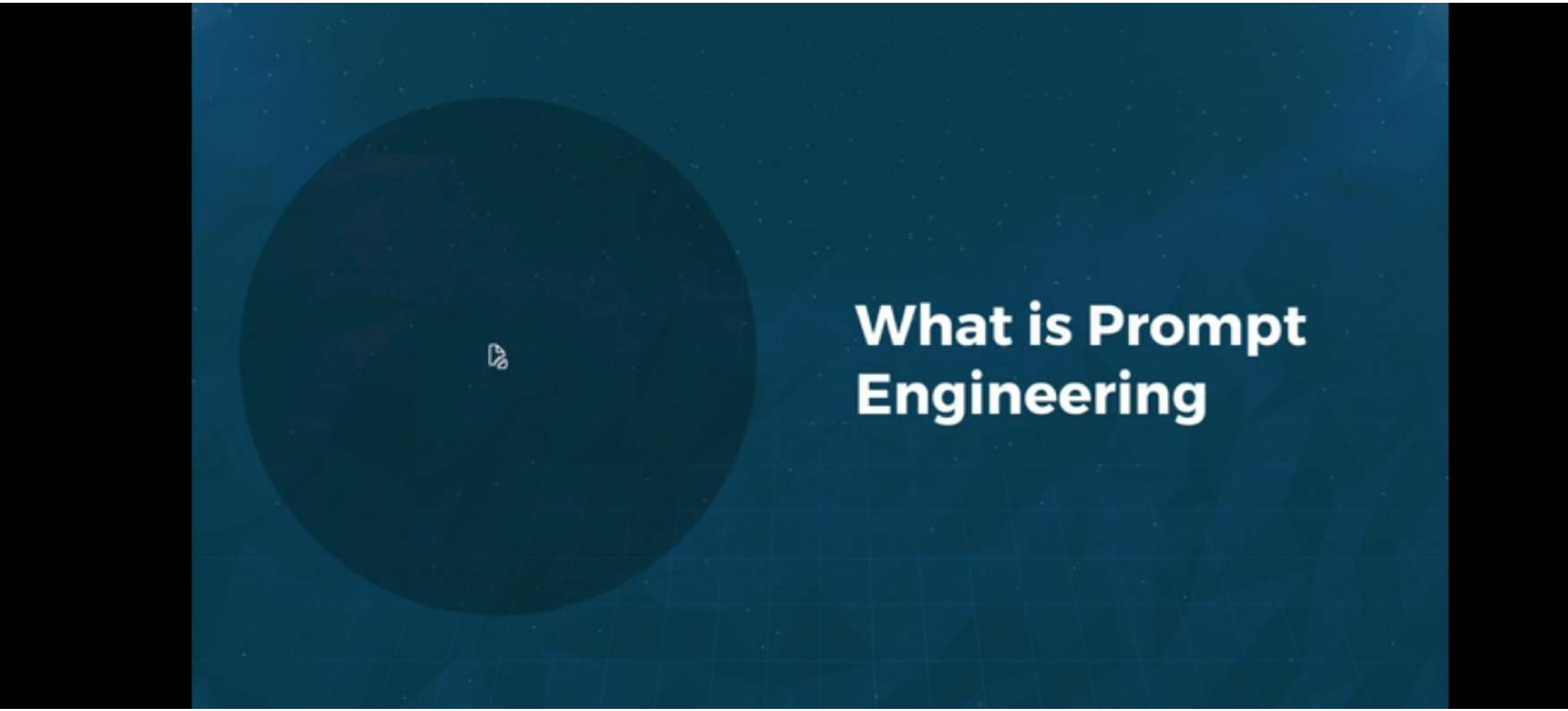
- The second element is the **Context**, which may involve external information or additional information to guide the model for appropriate responses.



- The third element is **Input Data**, which is the query or input that the model will use to generate a response.

- The last element is the **Output Indicator**, which specifies the type or format of the output.




What is Prompt
Engineering

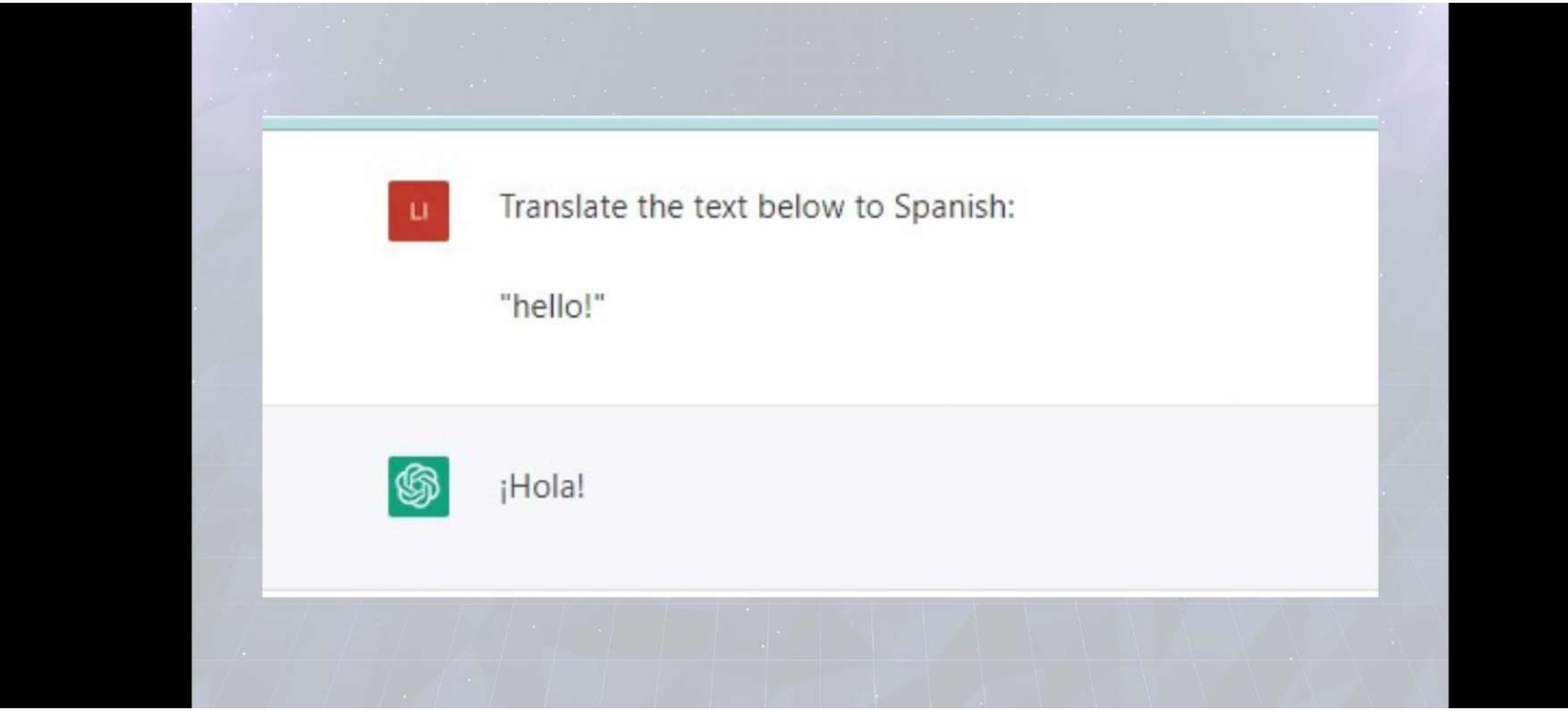
- The quality of your input determines the quality of your output
- Prompt Engineering is the process of designing effective prompts
- A prompt engineer builds prompts in a way that gets the best results from large language models

# How to Design a Prompt

# How to Design a Prompt?

# Precise Instructions

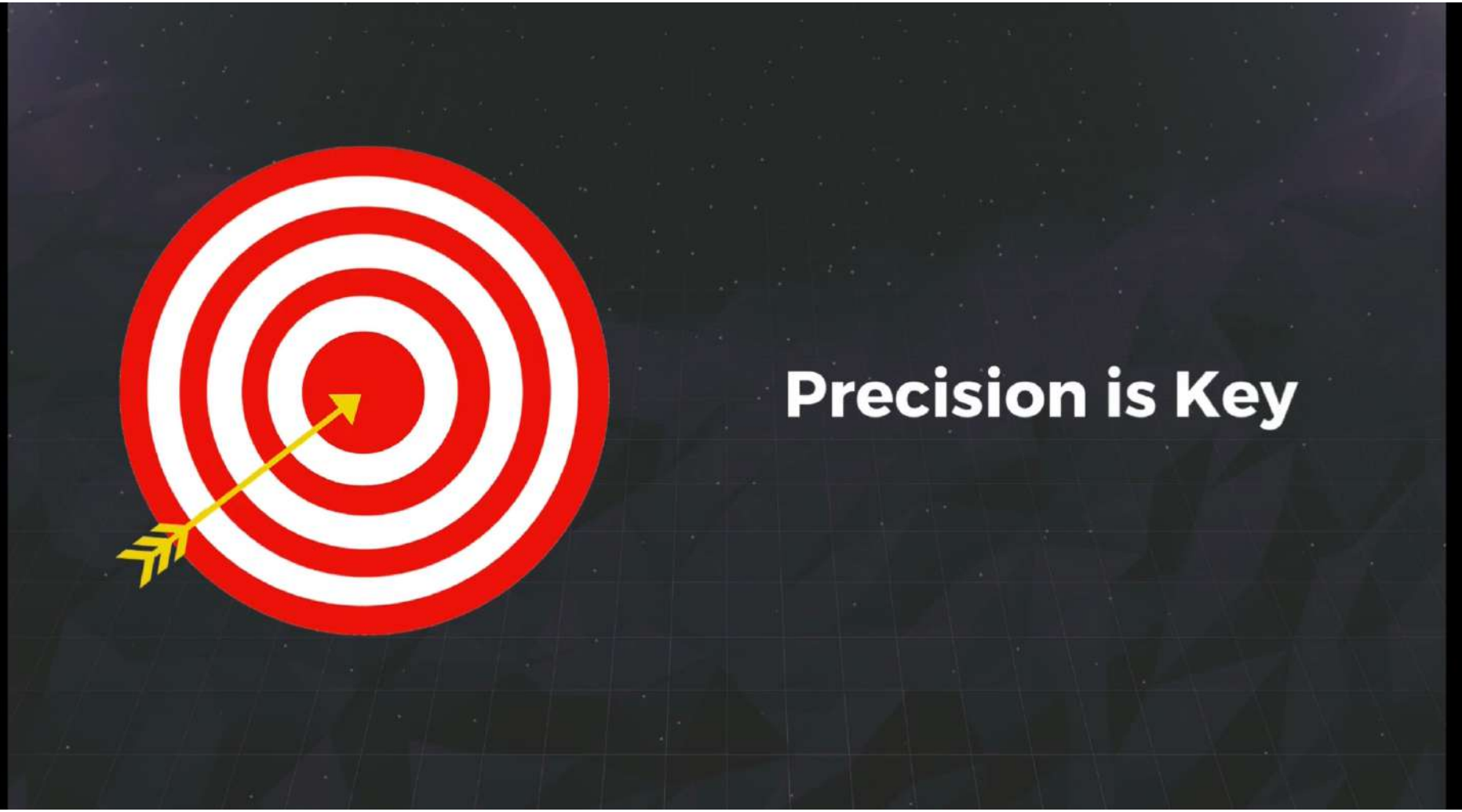
- Use commands like "**Write**", "**Classify**", "**Summarise**", "**Translate**", "**Order**", etc., to instruct the model on the desired results.
- Experiment with variations of instructions, keywords, contexts, and data to find the best results.
- The more specific and relevant the context is to the task, the better the results.
- Place instructions at the beginning of the prompt and use clear separators to distinguish the instruction and context. For example


LI
Translate the text below to Spanish:
"hello!"
¡Hola!

# Be Specific

- Be specific and detailed about the instruction and task you want the model to perform for better output.
- Providing examples in the prompt can be effective to get desired output in specific formats.

- Consider the length of the prompt as there are limitations on length.
- Avoid including unnecessary details and focus on relevant details that contribute to the task.
- Experiment with prompts to find the optimal level of specificity and detail.


Precision is Key

- Be specific and direct when creating prompts for better results.
- Avoid using negative instructions and focus on positive instructions instead.
- Using positive instructions helps to generate more specific and detailed responses from the model.
- Communicate with the model as clearly as possible to ensure the message is effectively conveyed.

Which of the following are elements of a prompt? Choose four answers.

Answer instructions

- Few-shot
- Zero-shot
- Input data
- Context
- Instructions
- Output indicator

Submit answer

Generative AI models interact with users primarily through textual input.

Answer instructions

True

False

Submit answer

# Getting Started with Prompt Engineering

## Setting Up the Environment


Setting Up the
Environment

# Installing Necessary Libraries

To start prompt engineering, after installing Python, the next step is to install essential libraries like OpenAI's GPT-3 API and Hugging Face's Transformers library. These libraries offer pre-trained models for natural language processing.

# Third-Party Libraries

- To install prompt-engine-py, run the command: pip install prompt-engine-py.
- To install OpenAI's GPT-3 API, run the command:

***pip install openai***

- To install Hugging Face's Transformers library, run the command:

***pip install transformers.***

# Choosing a Machine Learning Framework

- PyTorch is easy to learn and work with, making it a good option for beginners.
- It has an intuitive and user-friendly design that gives it a level of flexibility that novice prompt engineers will love.
- PyTorch is better for Natural Language Processing projects and for building fast prototypes.

- PyTorch offers a dynamic computational graph that gives you the ability to modify your model on the go and experiment with different architectures and hyperparameters.
- PyTorch also integrates the Hugging Face Transformers library, which is a popular choice for working with pre-trained models for natural language processing.

# Developing Programming Skills

- Python has a massive and active community of developers, researchers and enthusiasts worldwide.
- Python is easy to learn and is considered one of the best languages to start practicing if someone is willing to get into programming.
- Python has an expressive syntax that makes it a reader-friendly and intuitive language.
- Python has a rich set of built-in data structures, functions and modules that make it a powerful and versatile tool to work with.

- Python is compatible with many machine-learning frameworks, including PyTorch which we'll be using in this course.
- Python is very flexible and customisable, making it easier to write your own classes, functions and modules to create and maintain prompts for large language models.
- Python is interactive and dynamic, allowing you to use online tools and platforms like ChatGPT, GPT-3 API and Hugging Face's Transformer library with it.

# Start From Scratch

- Learn the basics of Python by referring to online resources.
- Understand the underlying mechanisms of various machine learning frameworks.

What is the first step to installing Python on Windows?

Answer instructions

- Type "py" in the command prompt
- Download the Python installer from Python.org
- Click on "Add python.exe to Path"

Submit answer

What is the first step to installing Python on Windows?

Answer instructions

Type "py" in the command prompt

Click on "Add python.exe to Path"

Download the Python installer from Python.org

Submit answer

The command to install the Hugging Face's Transformers library is ____.

Answer instructions

transformers --install

pip install transformers

pip install openai

Submit answer

# Basic Prompting Methods

- This chapter will discuss the four elements a prompt usually has:
  - Instructions
  - Questions
  - Input Data, and
  - Examples.
- The first two elements are mandatory, while the latter two are optional.
- Here, we will use ChatGPT for examples to provide a basic understanding of prompting.

U

How should I write an article for a school magazine? Suggest me different sections my article should contain. What should be the tone and tell me what to avoid.

Share Prompt

Writing an article for a school magazine can be a great opportunity to share your knowledge, experiences, and opinions with your fellow students. Here are some suggestions for sections you could include in your article:

1. Introduction: Begin with an attention-grabbing opening that sets the tone for the rest of the article. You could use a quote, a question, or an interesting fact to hook your readers.
2. Background: Provide some context or background information about the topic you are discussing. This could include historical information, statistics, or definitions.
3. Body: This is the main part of your article, where you will provide your thoughts, opinions, and analysis. You can use subheadings to organize your ideas and make your article easier to read.
4. Examples: Use real-life examples to illustrate your points and make your article more engaging. You could include anecdotes, case studies, or interviews with experts.
5. Conclusion: Sum up your main points and provide some final thoughts or recommendations. You could also end with a call to action, encouraging your readers to take action on the issue you have discussed.

Share Prompt

# A Little Advances Prompting

There are various ways to combine the components of a prompt, and one can get as creative as one wants. However, the generated response will be stochastic and different every time. By adding constraints to the prompt, it is possible to guide the model towards the desired response.

LI

The last time a Formula One Grand Prix was held in the Adelaide Street Circuit, who was the winner?

The last time a Formula One Grand Prix was held in the Adelaide Street Circuit was in 1995, and the winner of that race was Damon Hill driving for Williams-Renault.

LI

What is the sum of the cubes of the individual digits of the last year a Formula One Grand Prix was held in the Adelaide Street Circuit? Use the format above.

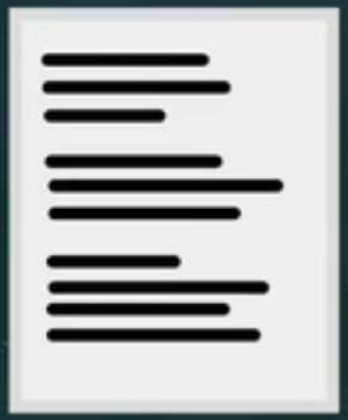
The last year a Formula One Grand Prix was held in the Adelaide Street Circuit was 1995.

The sum of the cubes of the individual digits of 1995 would be:
$1^3 + 9^3 + 9^3 + 5^3 = 1 + 729 + 729 + 125 = 1584$

Therefore, the sum of the cubes of the individual digits of the last year a Formula One Grand Prix was held in the Adelaide Street Circuit is 1584.


Examples of
Various Prompts

# Text Summarisation



- Text summarisation is a standard task in natural language generation.
- There are many different flavours and domains of text summarisation.
- Summarising articles and concepts into quick and easy-to-read summaries is one of the most promising applications of language models.

# Information Extraction

Did you know that language models are not just limited to natural language generation? They are also proficient in classification and a wide range of other NLP tasks. These models are true multitaskers, ready to take on any challenge thrown their way.


Want the model to give you specific answers? Well, improving the format of the prompt is key! By combining instructions, context, input, and output indicators, you can get even better results. While these components aren't mandatory, using them is a great practice. After all, the more precise your instructions, the better the results you'll get.

# Text Classification

As a prompt engineer, simple instructions will only get you so far. To tackle harder use cases, you'll need to step up your game and think about the context and the various elements you can use in a prompt. Providing input data or examples can make all the difference. So, get ready to level up your prompt engineering skills and take on even more challenging tasks!

# Conversation

Are you looking to build conversational systems like customer service chatbots? Well, prompt engineering has got you covered! With this technique, you can instruct the LLM system on how to behave, its intent, and its identity. It's an exciting way to customise the behaviour of your chatbot and make it stand out from the crowd. Get ready to take your chatbot game to the next level!

# Code Generation

Code generation can be a daunting task, but with LLMs and prompt engineering, it becomes much more manageable. Just look at Co-pilot! This tool is an excellent example of how LLMs can be used to generate code effectively. With the right prompts, you can tackle a wide range of code-generation tasks and make your development process more efficient. Get ready to level up your coding skills!

# Reasoning

While LLMs have come a long way in natural language generation, reasoning tasks still pose a significant challenge. However, with advanced prompt engineering techniques, we can push the boundaries of what these models can achieve. The potential for complex applications that require reasoning is immense, making this an exciting area of development for LLMs.

# Making AI Models Factual

When it comes to generative models, there's always a risk of getting false or incorrect information. But fear not, as prompts can guide these models towards more accurate sources of information. Just be cautious when dealing with obscure queries, as finding reliable sources may prove to be a bit tricky, resulting in outputs that may not be as precise as desired.

# Advanced Prompting

Zero-shot prompting refers to the ability of a model to understand and respond to a prompt without prior training on that specific task or prompt. Because of the model's impressive generalisation capabilities, it is able to adapt to new and unseen tasks, making it a valuable tool in a variety of applications.

*Few-shot prompting is a concept where a model is able to learn and adapt to a new task by being exposed to only a few examples of that task.*

# Chain-of-Thought (CoT) Prompting and Zero-Shot CoT Prompting

- Chain-of-thought (CoT) prompting enables complex reasoning capabilities through intermediate reasoning steps.
- CoT prompting can be combined with few-shot prompting for better results on complex tasks that require reasoning before responding.