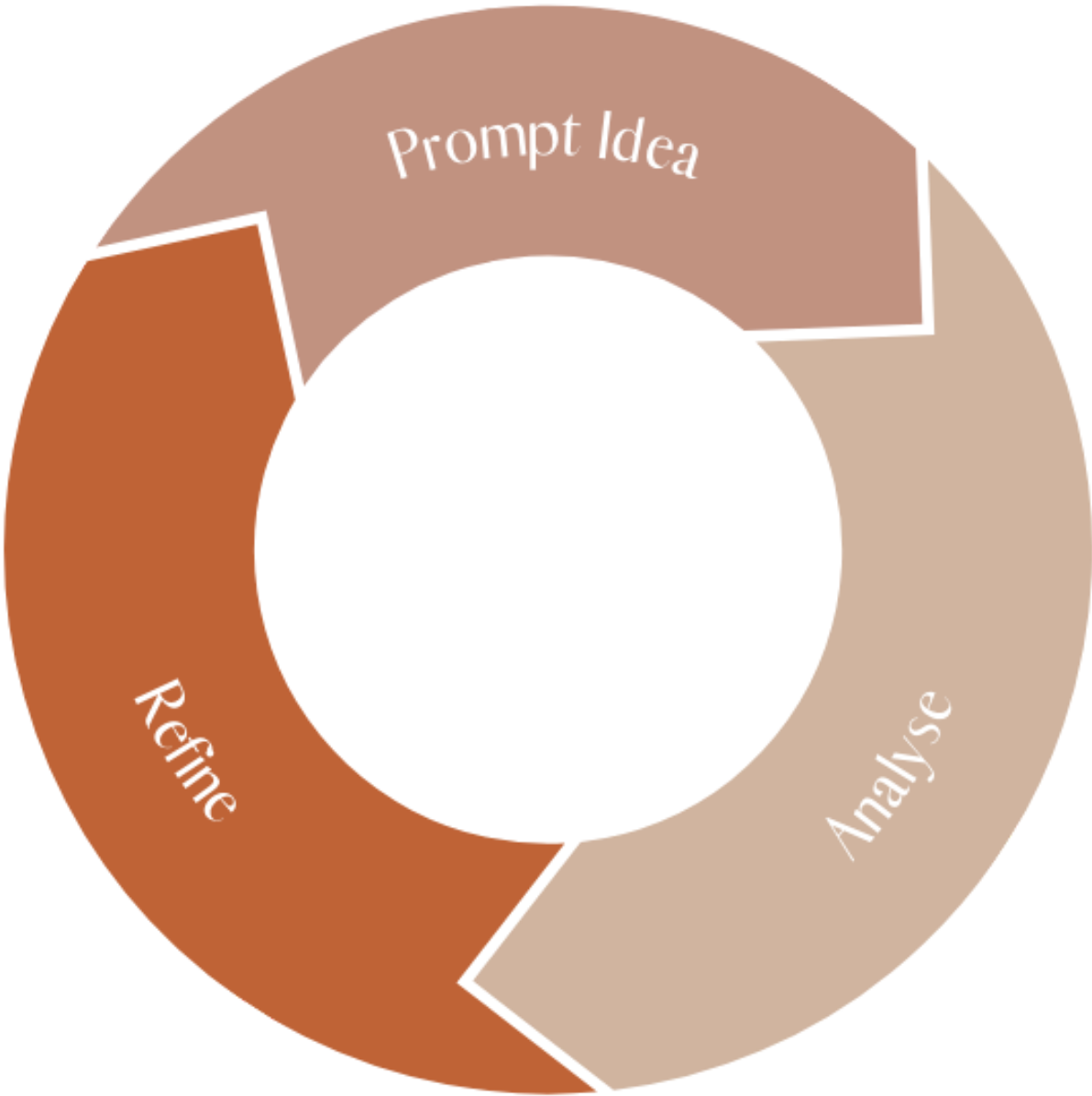# ChatGPT Prompt Engineering for Developers Lessons

ITERATIVE OF CYCLE PROMPT ENGINEERING

I.A X A.I

- USECASES

- TWO PRINCIPLES TO GET BETTER RESULTS FOR THESE USECASES

-
-
-
-

Summarising
Inferring
Transforming
Expanding

- Focusing your prompt for a summary towards a particular group.

- Extraction of information relevant to x, y and z.

- Use case of sentiment analysis.

- Categorising information
- Tracking topics.

- Asking for things in JSON format to get a structured reliable output.

- For example you would specify that you output should be in JSON format having keys such as the city, state, street etc.

```
1  {
2      "endereco": {
3          "cep": "31270901",
4          "city": "Belo Horizonte",
5          "neighborhood": "Pampulha",
6          "service": "correios",
7          "state": "MG",
8          "street": "Av. Presidente Antônio Carlos, 6627"
9      }
10 }
```
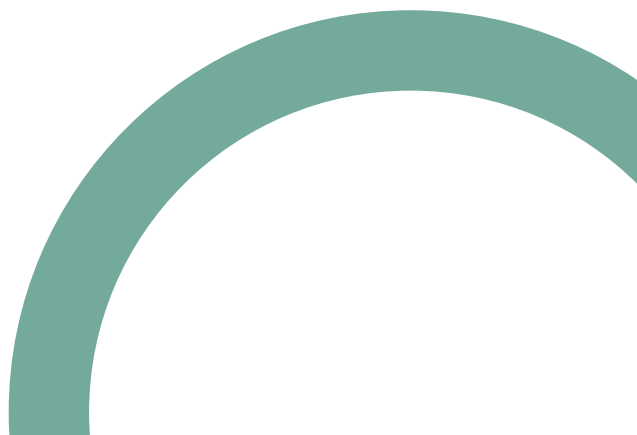
- Tone transformation.

- Spelling and grammar checking.

- To write HTML for display.

- To display as a JSON structured output.

- Brainstorming partner.

- Varying the temperature variable inside the LLM.

- This allows you the vary the degree of exploration and variety.

- Essentially the degree of randomness.

- More creative but less focused.

TWO PRINCIPLES TO GET BETTER RESULTS FOR THESE USECASES

- Write clear and precise instructions.

- Give the model time to think.

WRITE CLEAR AND PRECISE INSTRUCTIONS

Strategy #1:
- Use delimiters to avoid 'prompt injections' from the user
- Triple quotes: """, Triple backticks: ''',
- Triple dashes: ---, Angle brackets: <>,
- XML tags:

Strategy #2
- Ask for structured output
- E.g. HTML, JSON
- In Python you can read this into a dictionary or a list.

Strategy #3
- Check whether conditions are satisfied
- Checking whether user has provided the needed input/structure

Strategy #4
- "Few-shot" prompting.
- Essentially giving good examples of the prompt instruction being carried out

Strategy #1:
- *Specify the steps to complete a task*
  - Step 1:...
  - Step 2:...
  - Steps N:...

Strategy #2:
- Instruct the model to work out its own solution before rushing to a conclusion.

- After which you can ask it to compare with your solution.

- This is useful for things such as marking.

I.A X A.I