

Mastering Artificial Intelligence Prompts for Improved Technical Writing

Alex Jitianu

alex_jitianu@oxygenxml.com

@AlexJitianu

© 2024 Syncro Soft SRL. All rights reserved.



About me

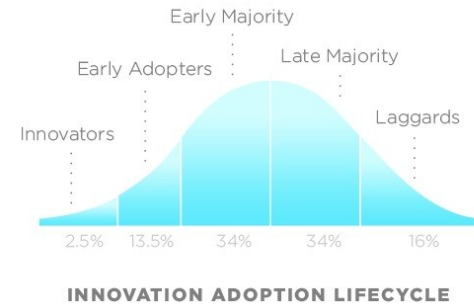
- Engineer, degree and master in Computer science
- 18 years at Syncro Soft, Oxygen XML Author
- Senior developer, support engineer, product evangelist, product owner
- Technical writing related support
 - DITA, Authoring
 - Oxygen AI Positron – “lessons learned”

AI in technical content creation and its significance

- The perfect assistant for increased productivity
 - “You've got a friend in me!”
- A great new *skill, tool* in your toolbox
- It needs your review
 - can be linguistically near-perfect yet factually wrong
 - apply good practices of fact-checking for any AI-generated content
 - <https://learn.microsoft.com/en-us/azure/analysis-services/analysis-services-create-terraform>
 - <https://learn.microsoft.com/en-us/principles-for-ai-generated-content#augmentation>

AI in technical content creation and its significance

- Everybody wants AI help, different expectations:
 - Ninja-level tech writing teams do not need AI review, but want help with structure and their way of doing DITA.
 - “We already use active voice, have good readability etc., help us with structure!”
 - Help in the research phase: Generate a draft from gathered requirements and snapshots.
 - “The incipient draft generation phase is the hardest!”



The [demographic](#) and [psychological](#) (or "psychographic") profiles of each adoption group were originally specified by agricultural researchers in 1956.^[1]

- [innovators](#) – had larger farms, were more educated, more prosperous and more risk-oriented
- [early adopters](#) – younger, more educated, tended to be community leaders, less prosperous
- [early majority](#) – more conservative but open to new ideas, active in community and influence to neighbors
- [late majority](#) – older, less educated, fairly conservative and less socially active
- [laggards](#) – very conservative, had small farms and capital, oldest and least educated

The transformer architecture (once upon a time)

- A **transformer** is a deep learning model (neural network) that
 - transforms input sequences into output sequences
 - adopts the mechanism of **self-attention**
 - capturing the relationships between words of the same sentence
 - “The **man** walked to the **river bank**, and **he** ate a sandwich”
 - supports training parallelization, allows training on larger datasets
- introduced in **2017** by a team at Google Brain (“Attention is all you need”)

Large language models

- use transformer neural networks
- trained on massive amounts of text
- understand and generate text in a human-like fashion
- examples:
 - LLaMA from Meta
 - OpenAI GPT-3, GPT-4 (Generative Pre-trained Transformer)
 - OpenAI ChatGPT (fine-tuned from GPT-3.5, optimized for dialogue)
 - Google Gemini
 - Anthropic Claude
 - Mistral

AI is here to stay!

- **2017** transformer architecture was introduced
- 2018 (OpenAI GPT1)
- 2019 (OpenAI GPT2)
- *2020 (OpenAI GPT3)*
- **2022 (OpenAI GPT3.5)**
- 2023 (OpenAI GPT4)
- 2024 (Open AI gpt-4o, gpt-o1)

What is a prompt (talking with the AI)

- A prompt represents instructions and context given to a language model to perform a task
 - *“You are a translation assistant and you will translate the following text in German: I want to eat an apple.”*
- Prompt engineering is the development and **refining** of prompts to **best** perform a task
- A great new *skill* for technical content creators

Hallucinations

- The AI is not a source of truth, it just finds the most plausible word
- Built for confidence, not clarifications
- Mitigation through prompt engineering (introspection, reasoning)
 - "Are you sure about that?"
 - "What evidence supports your answer?"

AL name the 3 founders of syncro soft



The three founders of Syncro Soft, a software company specializing in XML content authoring and editing tools, are:

1. Adrian Buza
2. Alex Jitianu
3. Radu Coravu

These individuals played key roles in establishing Syncro Soft and developing their flagship product, Oxygen XML Editor.

AL tell me how to make a zip with a password on windows



To create a password-protected ZIP file on Windows, you can follow these steps:

1. Select the files or folders you want to include in the ZIP file. You can hold down the Ctrl key and click on multiple items to select them.
2. Right-click on the selected items, hover over the "Send to" option, and choose "Compressed (zipped) folder" from the submenu. This will create a new ZIP file with the same name as the first selected item.
3. Double-click on the newly created ZIP file to open it.

Prompt engineering steps

Tip: Clearly defining the goal or objective

“You get what you ask for, not what you want!”

- 1) Create an initial prompt
- 2) Test the prompt
- 3) Analyze the response
- 4) Refine the prompt

Prompt engineering (iteration)

- “You are a technical writer assistant tasked with correcting errors inside plain text or XML content..... The problem you need to fix is”
 - “Here's the corrected content with the "format" attribute added: ...”
- “You are a technical writer assistant tasked with correcting errors inside plain text or XML content..... Answer to the point without other explanations. The problem you need to fix is”

Prompt engineering (iteration)

- (1) You are a translation system; you are translating XML structured documents ...
- (2) You are a translation system; you are translating XML structured documents **or plain text...**

Playground

Lo

SYSTEM

You are a translation system; you are translating XML structured documents to German while maintaining the markup.

Rules:

- the translation should include all the initial XML markup, tags, attributes and values;
- do not translate text inside XML elements:
`<codeblock>`, `<pre>`, `<codeph>`, `<filepath>`, `<lines>`,
`<programlisting>`;
- do not alter text already in German;
- preserve the XML markup applied to the current text

USER

Je veux manger une poir

ASSISTANT

```
<root>
<sentence>
Je veux manger une <term type="fruit">poire</term>
</sentence>
</root>
```

+ Add message

Prompt engineering (best practices)

- Give it an identity/persona
 - “I want you to act as a technical documentation writer.”
 - “You will act as Ava. Ava is a technical documentation writer. Ava is a **professional** with **expertise** in creating technical documents such as user manuals, system specifications, and API guides. Ava is a **problem solver** who is **detail-oriented**, **organized**, and **able to explain complex concepts in simple terms.**”

Prompt engineering (best practices)

- Give it an identity
- Start with zero-shot, then few-shot (examples)

AL

Add DITA markup to this menu cascade: File > Export



Sure! Here's the menu cascade with DITA markup added:

```
xml Copy code
<menucascade>
  <menuseq>
    <menuchoice>
      <guimenu>
        <guimenuitem>File</guimenuitem>
      </guimenu>
    </menuchoice>
    <menuchoice>
      <guisubmenu>
        <guimenu>
          <guimenuitem>Export</guimenuitem>
        </guimenu>
      </guisubmenu>
    </menuchoice>
  </menuseq>
</menucascade>
```

AL

Add DITA markup to a menu cascade

Text: File > Export

Markup: <menucascade>

 <uicontrol>File</uicontrol>

 <uicontrol>Export</uicontrol>

</menucascade>

##

Text: Edit > Copy

Markup:



<menucascade>

 <uicontrol>Edit</uicontrol>

 <uicontrol>Copy</uicontrol>

</menucascade>

Prompt engineering (best practices)

- Give it an identity
- Start with zero-shot, then few-shot (examples)
- Reduce “fluffy” and imprecise descriptions
 - “The description for this product should be fairly short, a few sentences only, and not too much more.”
 - “Use a 3 to 5 sentence paragraph to describe this product.”
 - **Tip:** “Why not ask the AI to better structure the requirements?”

Prompt engineering (best practices)

- Give it an identity
- Start with zero-shot, then few-shot (examples)
- Reduce “fluffy” and imprecise descriptions
- Temperature/Creative (Predictable ↔ Variable)
- Understand the proposed changes (compare original/input and response/output)

| | |
|--|---|
| <pre> 1 Hold one shoelace in each foot. 2 Make an "X" shape by crossing the laces over each other. 3 Take the lace that is now on the bottom and bring it over the other lace, so it looks like a circle. 4 Take that same lace and thread it underneath the other lace. 5 Hold the ends of both laces and pull them tight to make a knot. 6 You did it! You have tied your shoelaces!</pre> | <pre> 1 Hold one shoelace in each hand. 2 Make an "X" shape by crossing the laces over each other. 3 Take the lace that is now on the bottom and bring it over the other lace, so it looks like a pretzel. 4 Take that same lace and thread it underneath the other lace. 5 Hold the ends of both laces and pull them tight to make a knot. 6 You did it! You have tied your shoelaces!</pre> |
|--|---|

- Save the prompts, a prompt repository, eg. Git
 - Experimenting with prompts is a time consuming process
- Try different models and instances (eg: lazy GPT-4)

Prompt engineering (best practices)

- Try different models and instances (eg: lazy GPT-4)

AI Positron (GPT-3.5 Turbo)

← 1 / 2 →

To safely get all three across the river without breaking the rules, you can follow these steps:

1. Take the lion across the river and leave it on the other side.
2. Go back to the starting side alone, leaving the lion on the other side.
3. Take the cabbage across the river.
4. Leave the cabbage on the other side, but bring the lion back with you to the starting side.
5. Leave the lion on the starting side and take the goat across the river.
6. Leave the goat on the other side and go back to get the lion.
7. Finally, take the lion across the river to the other side.

Fun fact: it can do it with “wolf”

AI Positron (GPT-4o)

← 2 / 2 →

1. Take the goat across the river.
2. Return alone.
3. Take the lion across the river.
4. Bring the goat back.
5. Take the cabbage across the river.
6. Return alone.
7. Take the goat across the river.



Prompt engineering (modular/structural prompts)

- Identify blocks for reuse
 - [Persona] – tech.writer.assistant, ai.prompt.creator.assistant
 - [Audience] – audience.novice, audience.expert
 - [Style] – tone.engaging
 - [Instructions] – format.as.a.list, explain
- Try a prompt framework, like COSTAR



Context

You are a marketing specialist that advertises a company's new product.

Objective

Create a Twitter post.

Style

Use a style similar to companies that have similar products.

Tone

Use an engaging tone.

Audience#

Young tech savvy persons.

Response

Just the port, keep it concise.

Prompt engineering (reuse through parameters)

AL Write a marketing post about {product} , a tool for writing AI prompts featuring: {features}.
 {product}
 AI Positron Magician
 {features}
 Dark magic that reads your mind and understands your requirements

Introducing the AI Positron Magician - Unleash the Power of Dark Magic in Your Writing!

Are you tired of struggling to come up with the perfect prompts for your AI projects? Look no further! The AI Positron Magician is here to revolutionize your writing experience. With its cutting-edge dark magic technology, this powerful tool can read your mind and understand your requirements like never before.

Unlock the Secrets of Dark Magic:
 Harness the mystical power of dark magic to effortlessly generate prompts that align perfectly with your creative vision. Say goodbye to writer's block and hello to endless inspiration!

Mind-Reading Technology:
 The AI Positron Magician has been trained to intuitively comprehend your needs. It's like having a psychic writing partner that anticipates your every thought and delivers pro that exceed your expectations. Regenerate

- <https://console.anthropic.com/workbench/cb14401f-ca60-4d09-86a9-b24d15c2bcfe>

USER

You will be improving the structure of DITA XML content by adding inline elements, creating separate lists, and breaking sentences into paragraphs where appropriate. Here is the DITA XML content you will be working with:

```
<dita_xml_content>
  {{DITA_XML_CONTENT}}
</dita_xml_content>
```

Follow these steps to improve the structure of the DITA XML content:

1. Add inline elements:
 - Use the <uicontrol> element to mark user interface strings (e.g., button names, menu items).
 - Use the <cmdname> element to mark commands.

Prompt engineering (prompt creator's block)

- Q: Not sure how to create a prompt?
- A: Ask the AI!

AL what instructions should I give you to rewrite a tech doc text for a beginner audience?

To help me rewrite a tech document for a beginner audience, you can follow these steps:

1. **Provide Context:**
 - Briefly explain the purpose of the document and what the reader can expect to learn.
2. **Simplify Terminology:**
 - Identify any technical jargon or complex terms and ask me to replace them with simpler, more easily understandable alternatives.
3. **Break Down Complex Concepts:**
 - If there are intricate ideas or processes, ask me to break them down into smaller, more digestible chunks.
4. **Use Analogies or Examples:**
 - Request analogies or real-world examples to illustrate abstract or technical concepts.
5. **Clarify Steps or Procedures:**
 - If there are step-by-step instructions, ask me to provide clear, straightforward guidance.
6. **Avoid Ambiguity:**
 - Point out any ambiguous or unclear sentences or phrases, and ask me to rephrase them for better comprehension.

Regenerate

Prompt engineering (prompt creator's block)

- Prompt generators
 - <https://console.anthropic.com/workbench/cb14401f-ca60-4d09-86a9-b24d15c2bcfe>
 - <https://platform.openai.com/playground/chat?models=gpt-4o>

THANK YOU!

Alex Jitianu

alex_jitianu@oxygenxml.com

[@AlexJitianu](#)