AI in Society and Public Services

Session 2: Prompt Engineering

Mário Antunes

January 26, 2026

Universidade de Aveiro

The Art of Prompt Engineering

1. The Rules of the Game

Understanding the Constraints

Before writing prompts, you must know the physical limits of your tool (ChatGPT/Gemini/Llama).

1. **Knowledge Cutoff:**
    - Models are frozen in time. They do not know news from yesterday unless you tell them.
    - *Example:* "Who won the Euro 2028?" $\rightarrow$ Hallucination.
2. **Context Window (Short-term Memory):**
    - The "RAM" of the conversation.
    - *Qwen 2.5 (0.5B)* has a limit (e.g., 32k tokens).
    - If you paste a 500-page book, the model will "forget" the beginning of the chat to make room for the end.

2. Managing Sessions

“Stateless” vs. “Stateful”

- **The Technical Reality:** The LLM API is **Stateless**. It does not remember you.
- **The Illusion of Memory:**
  - When you chat, the software (Open WebUI) sends the **entire** conversation history back to the model with every new question.
  - *Input:* `[User: Hi, AI: Hello, User: What did I just say?]`

Practical Implication:

- If you want to change topics completely (e.g., from “Coding” to “Cooking”), **Start a New Chat**.
- Otherwise, the “Coding” instructions will pollute the “Cooking” answers.

3. The Perfect Prompt Formula

Structure Your Request

Don't just say *"Write an email."* Use the **RCTC** Formula:

1. **Role:** "Act as a Senior City Planner..."
2. **Context:** "...we are receiving complaints about traffic in Aveiro..."
3. **Task:** "...write a polite response to a citizen..."
4. **Constraint:** "...keep it under 100 words and use a reassuring tone."

Try this with Qwen: > "Act as a grumpy medieval knight. Explain to a peasant (me) why Wi-Fi doesn't work in the castle. Keep it short."

4. Advanced: Few-Shot Prompting

“Show, Don’t Just Tell”

If the model isn’t understanding the format you want, give it examples.

Zero-Shot (Often fails): > “Extract the names from this text.”

Few-Shot (Success): > “Extract names and format them as JSON. > Text: ‘John likes pizza.’ $\rightarrow$ JSON: {‘name’: ‘John’} > Text: ‘Maria went home.’ $\rightarrow$ JSON: {‘name’: ‘Maria’} > > Text: ‘Mario and Ana are studying AI.’ $\rightarrow$ JSON:”

Result: Qwen will follow the pattern perfectly: `{'names': ['Mario', 'Ana']}`.

5. Adding Knowledge (Files) i

Since the model doesn't know *your* private data, you must force-feed it.

Method 1: Manual Context Injection

- *Prompt:*

```
> Use the following text to answer the question.
> Do not use outside knowledge.
>
> TEXT: [Paste your internal meeting notes here]
>
> QUESTION: What was the decision regarding the budget?
```

5. Adding Knowledge (Files) ii

Method 2: Using Tools (Open WebUI)

- Click the **“+” (Upload)** button.
- The system injects the file content into the **Context Window**.
- *Benefit:* Reduces hallucinations because the answer is visible to the model.

6. Prompt Hacking Tips

1. **Chain of Thought:**
    - *Instead of:* “How many golf balls fit in a bus?”
    - *Say:* “Think step-by-step. First estimate the volume of a bus, then a ball…”
2. **Negative Constraints:**
    - Tell it what *not* to do. “Do not use technical jargon.”
3. **The “Or Else”:**
    - Research shows models perform better if high stakes are simulated.
    - *Prompt:* “…your answer is critical for a public safety report.” (Triggers more careful processing).