REPUBLIC POLYTECHNIC

LM Studio Mastery: Building Brilliant Prompts with Precision

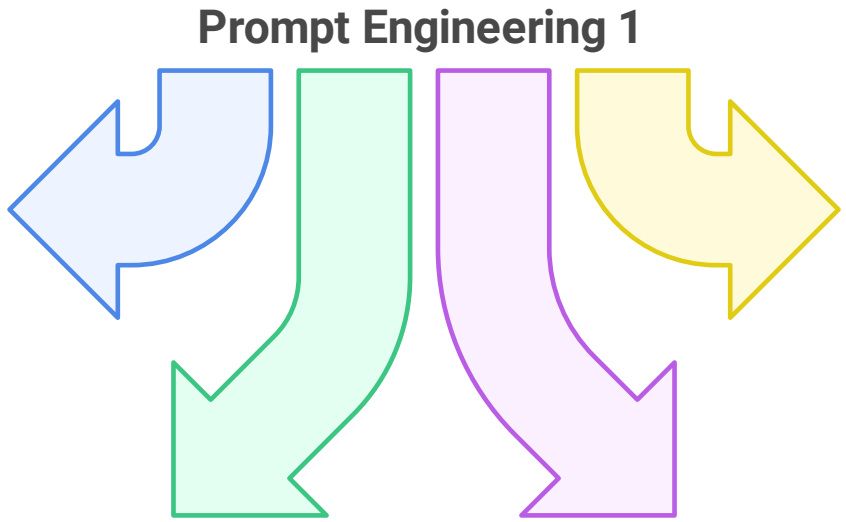
Chain, refine, and innovate.
Master AI, your strategy!

The Affordances of LM Studio

- **Run Local:** Run open-source LLMs locally on desktops/laptops, offline.
- **Familiar Interface:** Offer a user-friendly ChatGPT-like interface for interaction.
- **Integrations:** Download and load models from Hugging Face in formats like GGUF.
- **Customizations:** Customize model outputs with adjustable settings for laptop hardware.
- **Refine:** Support chat features for testing and refining prompts.

Review Techniques

Previously in C240 ...



Zero-shot

No prior examples. Model infers from the prompt intent. Ok for straightforward queries.

One-shot

Provide one example. The example can shape output and format. Ok when extensive context is not required.

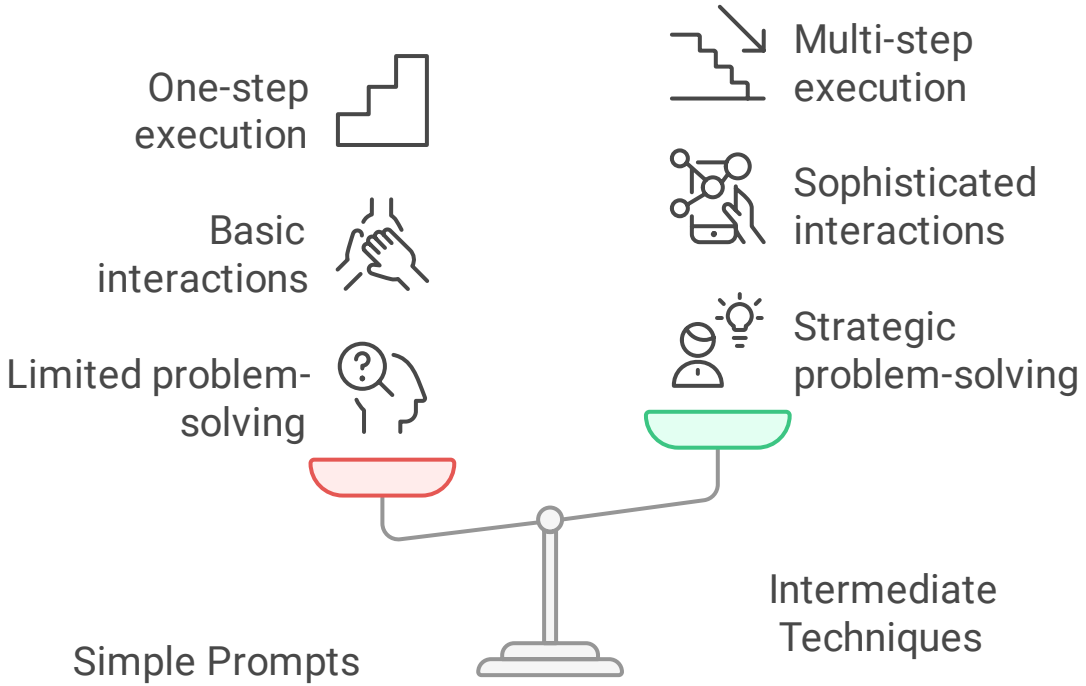
Few-shot

Multiple examples provided. Improvement in accuracy and consistency for complex tasks. Overfitting risk.

Self-critique

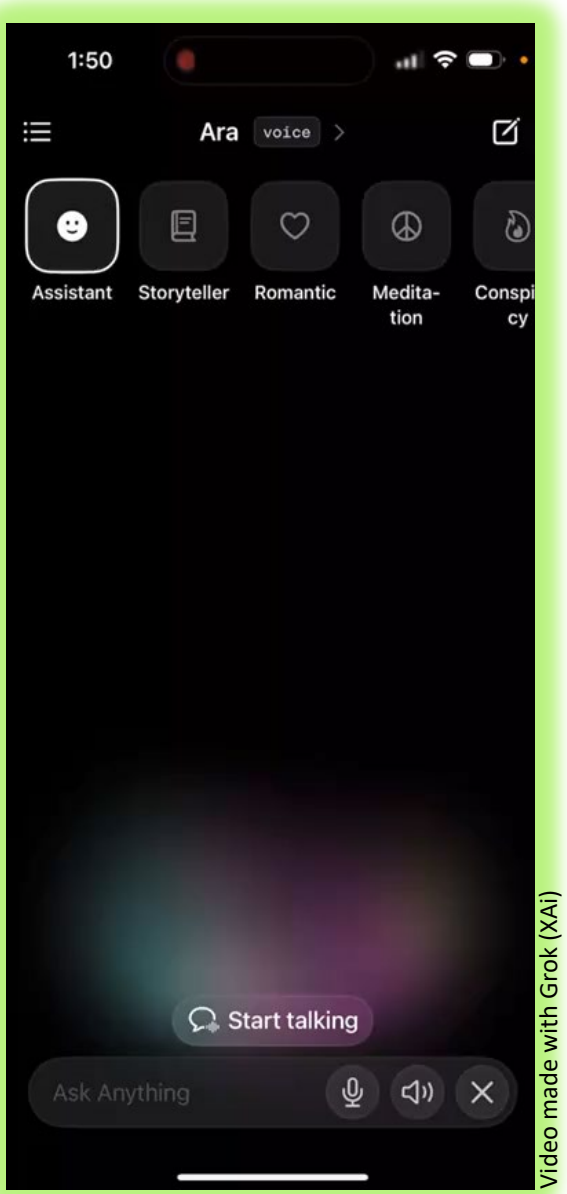
Encourages iterative improvement through self-assessment.

Beyond Basics: Intermediate Prompt Engineering



- With intermediate prompt engineering techniques, we move from simple prompts to more **strategic conversations** with LLMs.
- We've covered the basics. Now it's time to level up! In this lesson we will explore techniques that go beyond simple prompts and delve into crafting more **sophisticated AI interactions**.
- The Problem: Simple prompts often fall short when tackling **complex, multi-faceted** problems. We often need strategies that guide the AI towards a solution, **step-by-step**.
- The Solution: Intermediate prompt engineering provides a powerful toolbox for building **multi-step prompting strategies** that break down complex problems into manageable chunks.

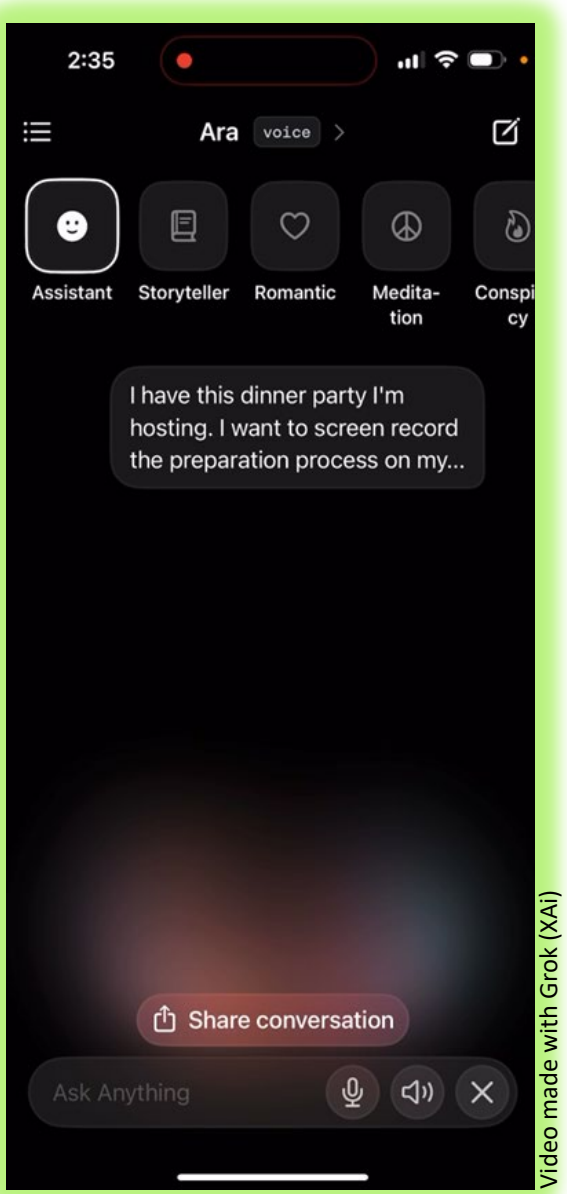
Core Concepts: Prompt Chaining



Video made with Grok (XAi)

- Concept: Linking a series of prompts together, where the output of one prompt feeds into the next.
- Focus: Creating a workflow where each step builds upon the previous one.
- Benefit: Ideal for tasks needing multiple stages of reasoning or processing, like writing a report or developing a complex plan.

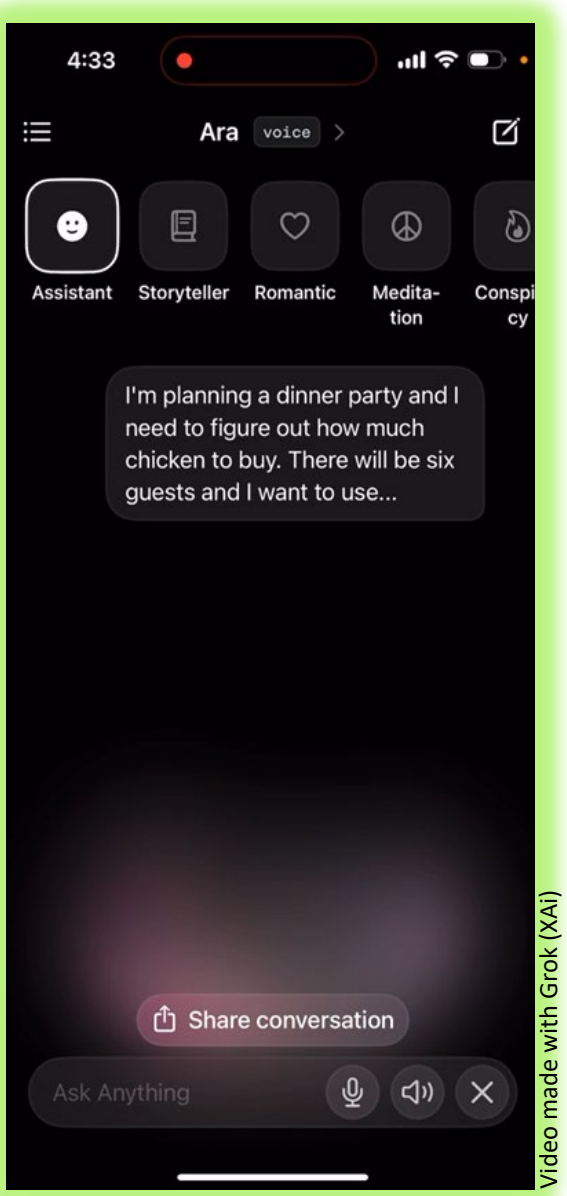
Core Concepts: Contextual Prompting



Video made with Grok (XAi)

- Concept: Providing the AI with relevant background information, examples, or constraints to guide its response.
- Focus: Grounding the AI in the specific scenario, problem domain, or user needs.
- Benefit: Improves the accuracy, relevance, and usefulness of AI outputs by preventing assumptions and providing necessary context.

Core Concepts: Chain of Thought (CoT)



Video made with Grok (XAi)

- Concept: Encouraging the AI to explicitly explain its reasoning process, **step-by-step**, before providing a final answer.
- Focus: Unlocking the AI's reasoning abilities and creating a more transparent and interpretable thought process.
- Benefit: Improves accuracy, identifies potential errors in reasoning, and helps users understand why the AI arrived at a particular conclusion.

Explicit Reasoning Models

Explicit Reasoning Models

- **What is the difference between general language models and explicit reasoning models?**
  - **General Language Models:** These are older models, like GPT-3 or GPT-4o, that *can* reason but usually need specific instructions in the prompt like "think step-by-step" or "show your reasoning" to give you a structured, logical answer.
  - **Explicit Reasoning Models:** These are newer AI models, like Grok 3 and DeepSeek-R1, designed to naturally "think" step-by-step when solving problems. They don’t need you to tell them how to reason, they just do it automatically. You can generally read their "thinking".
- **When did these models first appear?**
  - **General Language Models:** These started popping up around 2020–2023 with models like GPT-3 and GPT-4o. They were impressive for their time but needed a little hand-holding to reason well.
  - **Explicit Reasoning Models:** These are more recent, emerging in late 2024 and early 2025 with models like Grok 3 and DeepSeek-R1, built from the ground up to handle reasoning without extra guidance.

How are Explicit Reasoning Models Different?

- **Explicit Reasoning Models:**

  These models automatically break down complex problems into steps and reason through them. If you observe the thinking, they're having a little chat with themselves. You don’t need to add special instructions in the prompt as they’re built to "think" logically on their own.

- **General Language Models:**

  These models, while capable of reasoning, often lean on pattern recognition unless you nudge them with prompts like "think step-by-step." Without those prompts, their answers might lack structure or depth.

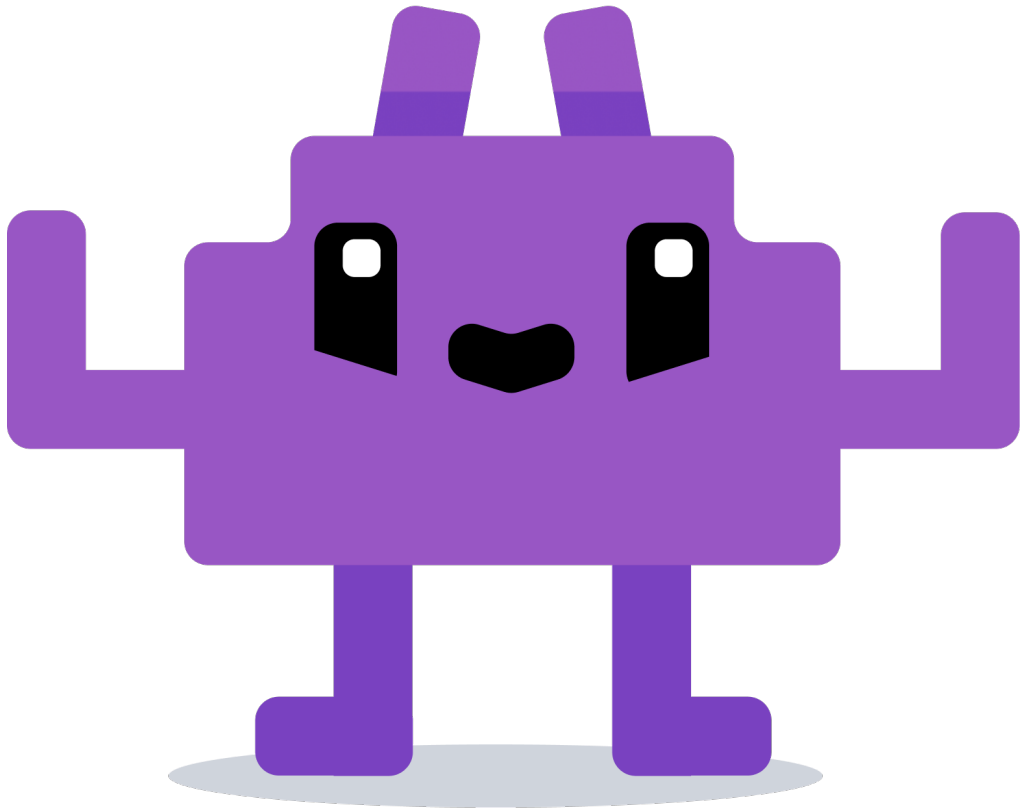
- **Why Does it Matter?**

  The shift from general language models to explicit reasoning models shows how AI has evolved. With models like GPT-4o, you had to play an active role in guiding their thought process. Now, with Grok 3 or DeepSeek-R1, the models take the lead, reasoning naturally and saving you the effort of crafting detailed prompts. It’s like the difference between giving a student a step-by-step worksheet versus them figuring it out independently!

Running Models Locally

LM Studio

LM Studio: Why use a local model?



- **Internet Dependency**: Online models require an internet connection; local models in work offline.
- **Performance and Speed**: Online models are faster on remote servers but may have latency; whereas local models are slower on laptops but do not suffer any network delays.
- **Resource Usage**: Online models use no local resources beyond bandwidth; local models use CPU and RAM, potentially slowing other apps.
- **Privacy and Security**: Online models risk data exposure via APIs; local models keep data private on the device.
- **Setup and Accessibility**: Online models are easier to access with an API key but limited by availability; local models require setup but offer full control offline.

Activity: Prompt Ninja Challenge

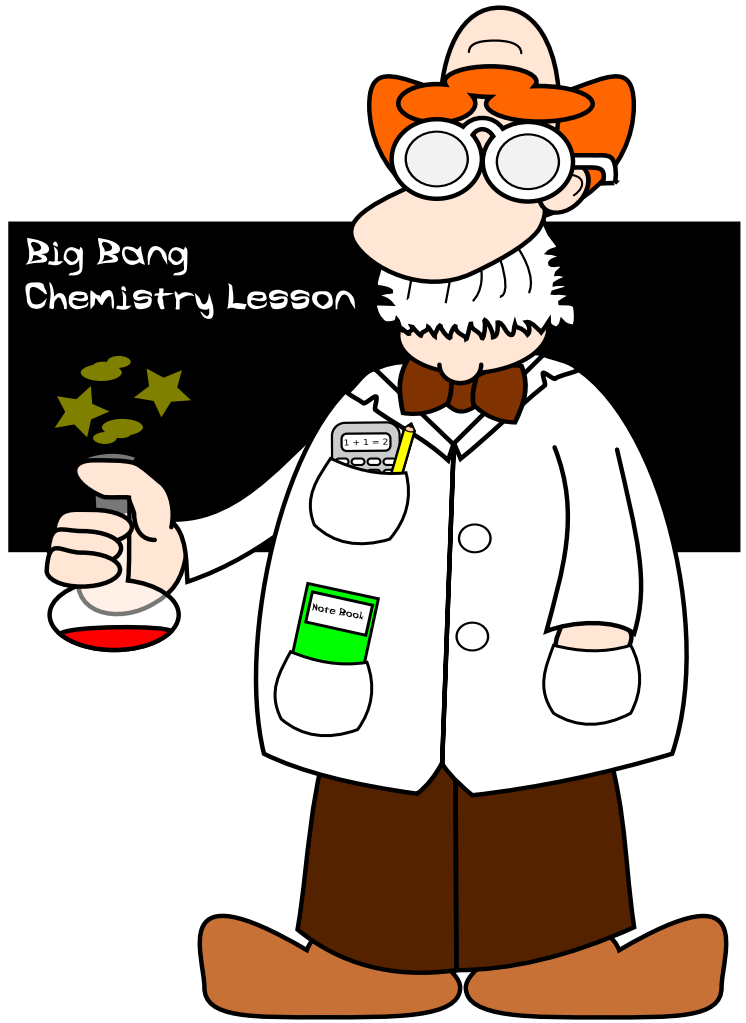
- Set your expectations. Remember models may be slower to respond when running on low-powered laptops. Have patience - responses may take a while.
- Open the accompanying **Ninja Challenge** document.
- Follow the detailed instructions included in the activity document.
- Don't forget to post to MST.

Shaping AI's Persona and System Design

Changing AI's Behavior – Be the Boss!

Role-Playing (Persona) Prompting

- **Role-Playing Prompting:**
  - **Concept:** Assign the AI a specific role or persona. Example: "Act as a software engineer" or "Assume the role of a nutty chemistry professor".
  - **Focus:** Guides the AI to adopt a unique viewpoint, tone, or style. Example: "Respond like an over-paid entitled Singaporean interior designer".
  - **Benefit:** Tailors the AI's output to match your expertise, needs or user vibe.



System Prompting

- **Changing the System Prompt:**
  - **Concept:** Set the stage for all chats by defining the AI's core behavior, capabilities, and limits. You are the boss!
  - **Focus:** Establishes clear rules so the AI doesn't go rogue.
  - **Benefit:** Keeps responses consistent, reliable and won't annoy you with random tangents.
  - **Example:** "Always use metric units and keep your responses concise unless directed otherwise. Where possible convert amounts to Singapore dollars."

Prompt Optimization

- **Practice makes Perfect:**
  - **Concept:** Tweak prompt phrasing, keywords, and structure to level up the AI's performance.
  - **Focus:** Refine through trial and error, similar to perfecting a precise task, such as optimizing a project plan.
  - **Benefit:** Transforms average responses into highly effective outputs, enhancing the precision and impact of your prompts for optimal AI performance.

Control and Steering Techniques

Changing AI's Behavior – Be the Boss!

Control and Steering Techniques

The techniques that follow focus on **guiding** or **constraining** the AI's output to meet specific requirements, avoid undesired responses, or maintain focus.

- **Negative Prompting**:

  This technique involves explicitly telling the AI what **not** to include in its response. Example: "Don't use technical jargon" or "Avoid humor". It's a steering mechanism to refine outputs by ruling out unwanted elements.

- **Constraint-Based Prompting**:

  This method imposes specific limitations or requirements on the AI's response, such as length, format, or style. Example: "Respond in exactly 100 words" or "Write in haiku form". Like negative prompting, it controls the output but focuses on positive boundaries rather than exclusions.

- **Contextual Anchoring:**

  This technique was mentioned earlier. You provide a specific context or role for the AI to operate within, ensuring responses stay relevant and grounded. This can prevent drift or overly generic answers. Example: "You are a historian in 1920s London. Describe the scene" or "Act as a Python expert troubleshooting code."

Activity: Persona, Control & Steering

- In LM Studio load the **reasoning** model and create a prompt to write an email objection to a new school policy. Example: No unnatural hair colour in school.
- Open the accompanying **Persona Control & Steering** activity document.
- Follow the detailed instructions included in the activity document.
- Don't forget to post to MST.

One result

Subject: DISASTROUS HAIR POLICY – TOTAL LOSER!

Dear School Board,

Folks, this "no unnatural hair color" policy is a TOTAL DISASTER! The worst I've ever seen, believe me! And I've seen some things. Purple, blue, red hair - they're *beautiful, they're fantastic, they're tremendous!* Totally harmless! Banning them? That's like banning winning, and **I don't do losers** - never have, never will! **I've got the best hair.** Nobody does it better, natural or not - and nobody, I mean NOBODY, tells me what to do with it!

This rule? Weak! Sad! Very sad! Totally un-American! It's fake news, folks! Let the kids shine with their colors - bright, bold, amazing colors! **We're gonna make this school great again.** You hear me? **GREAT AGAIN!** Reverse this nonsense right now, or I'll be very disappointed - and you don't want that, trust me!

Respectfully,
Donald J. Trump

Text & image generated by Grok XAi

Lesson 06

Summary of Material Covered

Lesson 06 Review

- LM Studio Overview
  - Runs open-source large language models (LLMs) locally on desktops/laptops, offline.
  - Features a user-friendly, ChatGPT-like interface for student interaction.
  - Integrates with Hugging Face to download models in formats like GGUF.
  - Enables customization of model outputs based on student hardware.
  - Supports chat features for testing and refining prompts.
- Model Types
  - **General Language Models**: Older models (e.g., GPT-3, GPT-4o) need explicit reasoning instructions.
  - **Explicit Reasoning Models**: Newer models (e.g., Grok 3, DeepSeek-R1) reason step-by-step naturally.
- Local and Online Models
  - **Local Models**: Offline, uses CPU/RAM, private, requires setup, may be slower on low-end devices.
  - **Online Models**: Internet-dependent, faster via servers, risks latency and data exposure.

Lesson 06 Review - Prompting Techniques

Zero Shot
A technique where no examples are provided.

Few Shot
A method using a few examples for guidance.

Self-Critique
Evaluating one's own prompts for improvement.

Prompt Chaining
Linking multiple prompts for complex tasks.

Contextual Anchoring
Using context to enhance prompt relevance.

Chain of Thought (CoT)
Encouraging reasoning through sequential prompts.

Adopting Personas
Using different perspectives in prompts.

System Prompting
Directing the system with specific instructions.

Prompt Optimization
Refining prompts for better performance.

Negative Prompting
Indicating what should not be included in responses.

Constraint-Based Prompting
Setting limits to guide responses effectively.

Thank you

School of Infocomm

C240 AI Essentials and Innovations