LANGCHAIN ZOOMCAP SESSION 3

Harpreet sahota

TODAY'S AGENDA

- Overview of core components in LangChain (in code)
  - Model I/O
  - Retrieval
  - Chains
  - Agents
  - Memory
- Basics of Prompting
  - Base LLMs vs Instruction Tuned LLMs
  - Prompt Basics
  - Principles and tactics for prompting
  - PromptTemplates in LangChain


LET'S CODE

BASE LLMS VS INSTRUCTION TUNED LLMS

BASE LLM

- Machines learn from numerous books, sites, and articles.
- They predict next words based on observed patterns.
- Example: "The sky is so..." leads to "...vast and filled with stars."
- Answers can be broad and encompassing.
- Might provide context instead of direct answers.

INSTRUCTION TUNED LLMS

- Follows given instructions rather than autocomplete.
- Precisely answers commands
- Trained on datasets with sample instructions and expected outcomes.
- Uses RLHF for continuous improvement from human feedback.
- Preferred for precise, tailored results by many users.

SUMMARY

- Base LLMs: Offer broad responses based on vast data patterns.
- Instruction-Tuned LLMs: Precision-focused, trained on specific instructions.
- Instruction-Tuned LLMs are gaining preference for tailored results.
- The crafting of prompts is crucial to steer desired outcomes.

PROMPT BASICS

THE POWER OF PROMPTS IN LLMS

- "In-Context Learning" steers LLMs without changing model weights.
- Success of prompts varies across models; testing is crucial.
- Effective LLM usage hinges on smart communication through prompts.
- Clear, concise prompts lead to better outcomes.
- Experimentation is key as there's no universal perfect prompt.

ANATOMY OF AN EFFECTIVE PROMPT

- Instruction steers the model's task direction.
- Context provides refining background details.
- Input Data is the primary question or topic.
- Output Indicator specifies the desired response format.

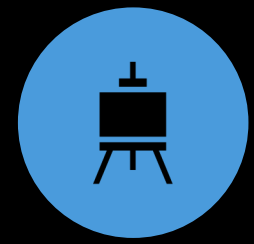
CRAFTING EFFICIENT PROMPTS FOR LLMS

UNDERSTAND PROMPT ELEMENTS FOR EFFICIENT LANGUAGE MODEL USE.

ELEMENT NECESSITY VARIES BY TASK.

PLACING INSTRUCTIONS AT THE END SHARPENS THE MODEL'S FOCUS.

PROMPT ENGINEERING IS AN EVOLVING ART.



GUIDELINES OFFER A FOUNDATION FOR SUCCESSFUL PROMPT CREATION.

SUMMARY

- Prompts are the steering wheel for LLMs.
- Effective prompts blend instruction, context, input data, and output indicators.
- Adaptability is key; not all elements are needed every time.
- Positioning, especially of instructions, can sharpen model focus.
- As prompt engineering grows, refining techniques will elevate LLM interactions.

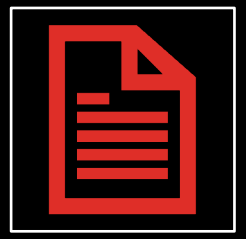

PRINCIPLES AND BEST PRACTICES
FOR CRAFTING PROMPTS

CLARITY IS KEY

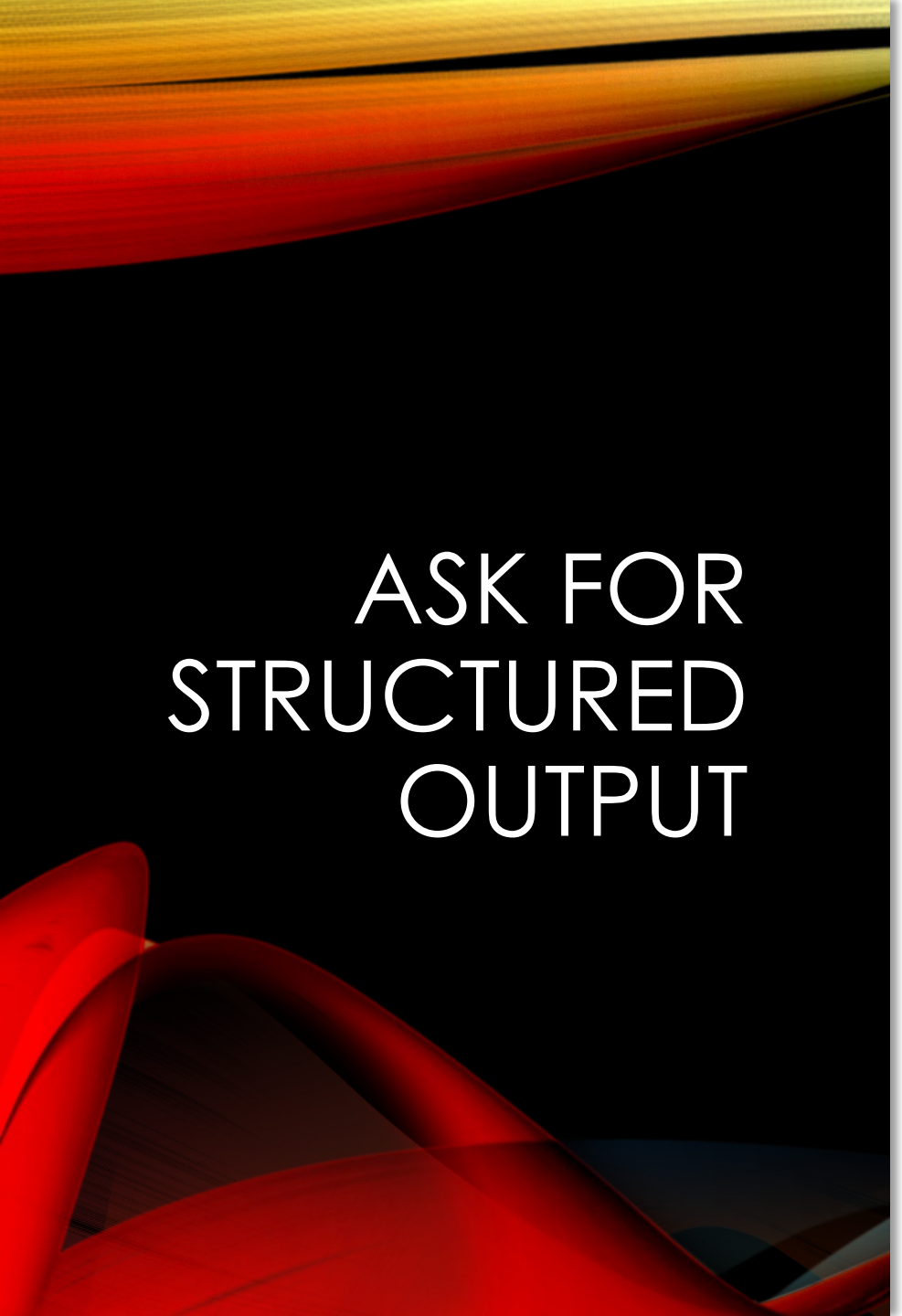
Ambiguity hinders generation accuracy.

Clear instructions lead to precise responses.

Delimiters like triple quotes or angle brackets enhance clarity.



Separating directives avoids content confusion.



ASK FOR STRUCTURED OUTPUT

- Clear output structure enhances interpretation.
- Define desired format at the start.
- Ask for easily parsable outputs like JSON or XML.
- Structured formats offer organized and actionable data.

USE FEW SHOT PROMPTING

- Context improves model responses.
- Multiple examples set clearer expectations.
- Don't just ask—show the desired response type.
- Examples guide the model towards your intended answer.

SET CONDITIONS

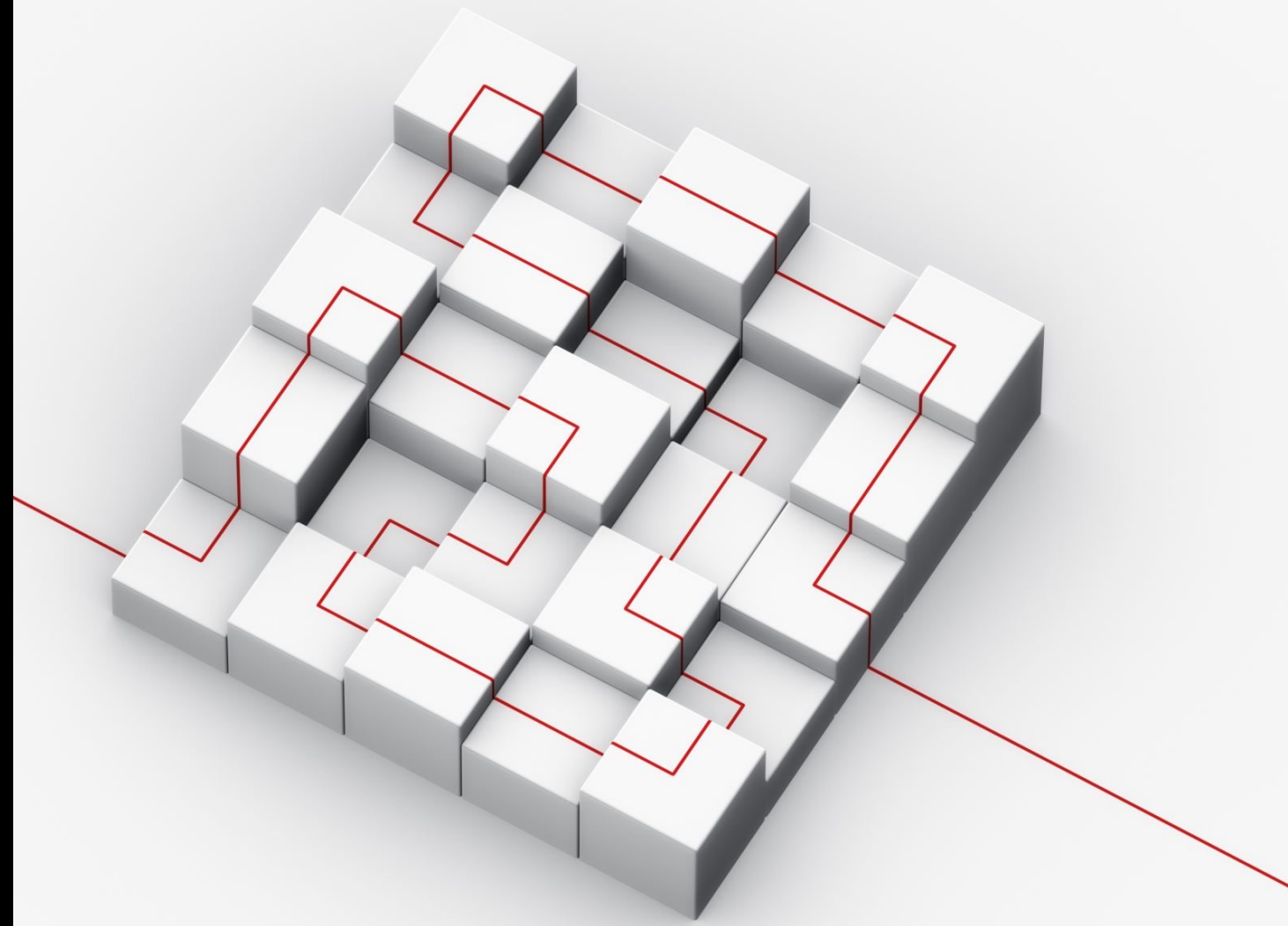
- Specifying prerequisites refines the model's output.
- Align results with your criteria by providing conditions.
- Use condition-based questioning for precision.
- Verify assumptions upfront, especially for edge cases.
- Design prompts to handle scenarios like absent instructions.

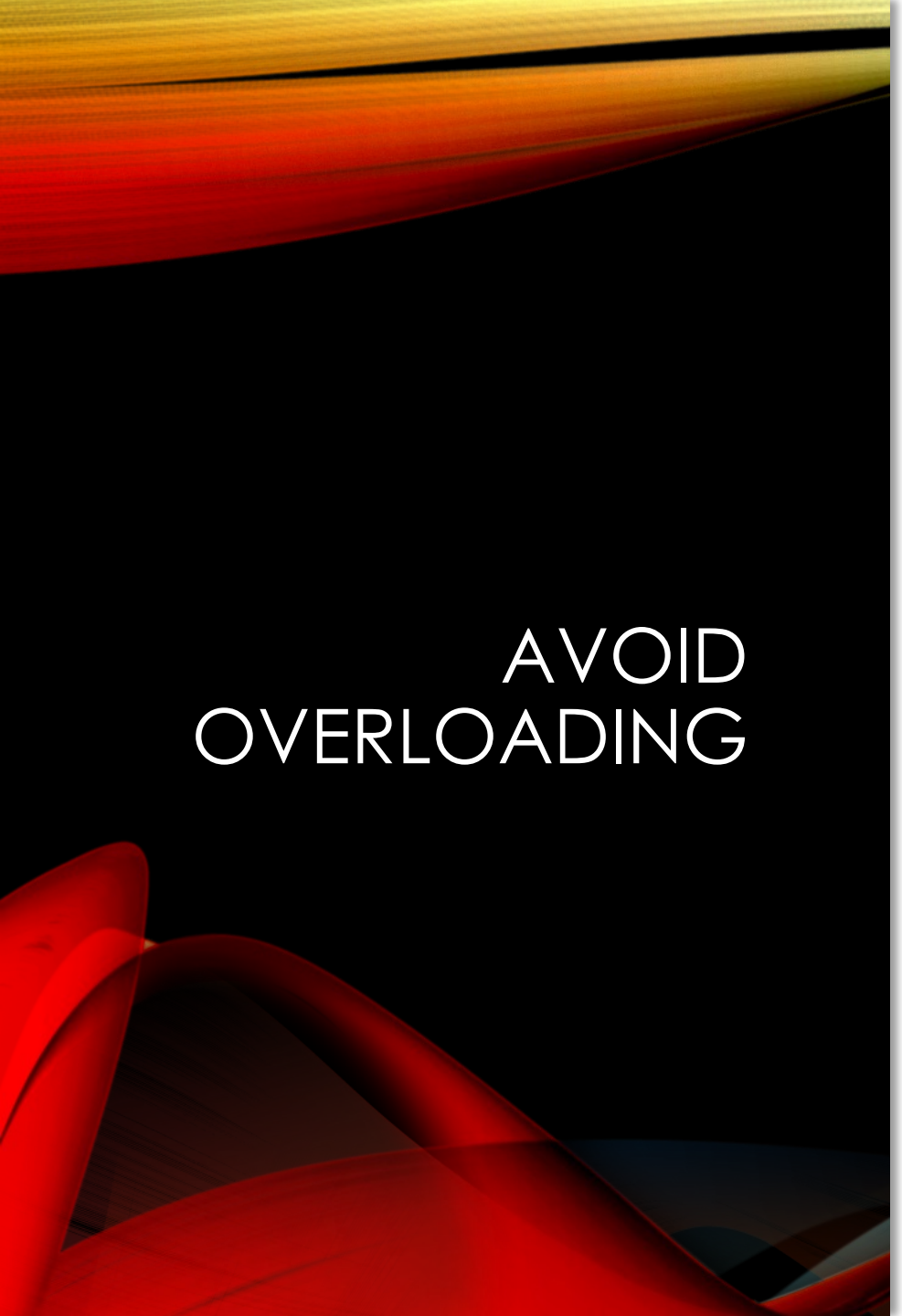
GIVE THE MODEL TIME

- Complex queries benefit from thorough model processing.
- Detailing steps and defining output format aids clarity.
- Break tasks into smaller, logical sequences.
- Step-by-step approach mirrors human problem-solving.
- Instruct the model to deliberate before concluding.

ITERATIVE REFINEMENT

- Initial prompts may require adjustments.
- Continuous refinement enhances accuracy.
- Analyze outputs to identify and bridge gaps.
- Process acts as a feedback loop for improved results.
- Iterative learning guides future model development and NLP strategies.





AVOID OVERLOADING

- Context is vital but avoid excessive details.
- Focus on prioritizing crucial information.
- Let the model seek more context if needed.
- Strive for a balanced and effective prompt.

ACTIVE ENGAGEMENT

Encourage two-way conversations with the model.

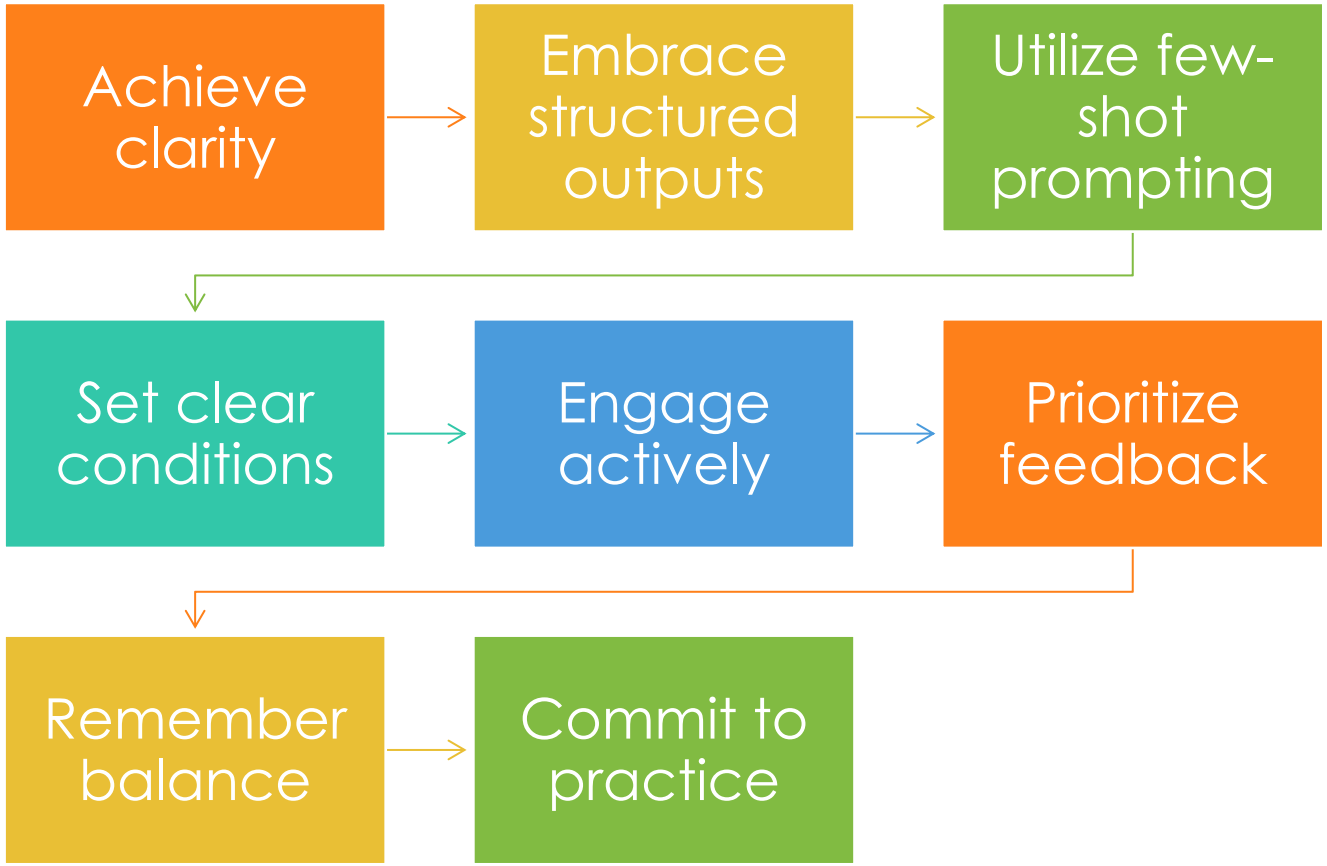
Go beyond directives: debate, compare, evaluate.

Active engagement yields richer and deeper responses.

FEEDBACK LOOP

- Constructive criticism refines the model's accuracy.
- Highlight both the hits and misses in the model's responses.
- Crafting prompts is like learning a new language.
- Requires practice, understanding, and alignment with needs.
- Proper feedback yields more tailored and accurate model outputs.

PROMPTING BEST PRACTICES RECAP



LEAVE SOME FEEDBACK

https://bit.ly/LC-ZC-FB