

# MAE 688: Machine Learning for Mechanical Engineers

## Lecture 11: from LLM to AI Agent



Dr. Bing Dong

*Traugott Professor  
Mechanical and Aerospace Engineering*

---

Syracuse University

# Outline

---

- Augmenting LLMs
- Prompt and Context Engineering
- Retrieval-Augmented Generation (RAG)
- Agentic AI workflow
- Evals
- Multi-agent Workflow

# Augmenting LLMs

---

Why do we need to augment LLMs?

What are the challenges of using a pre-trained model like ChatGPT 5.2?

# Augmenting LLMs

What are the challenges of using a pre-trained model like ChatGPT 5.2?

## Lack of domain knowledge.

how much daily solar radiation on Center of Excellence Building at Syracuse University

While there isn't a *specific published daily solar radiation measurement for the Syracuse Center of Excellence building itself* that's publicly available, we can use **local solar radiation climate data for Syracuse, NY**, where the Center is located.

### 📍 Syracuse, NY solar radiation (typical)

- **Average solar radiation (global horizontal irradiance)** in Syracuse *on a typical day* is about **4.3 kWh/m<sup>2</sup>/day** (kilowatt-hours per square meter per day). This is the daily total solar energy received on a horizontal surface, averaged over the year. Solar Energy Local

### 📊 Monthly typical values in Syracuse (GHI)

Month	Solar Radiation (kWh/m <sup>2</sup> /day)
Jan	~2.6
Feb	~3.5
Mar	~4.7
Apr	~5.25

# Augmenting LLMs

---

What are the challenges of using a pre-trained model like ChatGPT 5.2?

**LLM is not up to date.**

what is 6 and 7?

Could you clarify what you mean by "**6 and 7**"?

Do you mean:

- **Items 6 and 7** from a list I mentioned earlier?
- **The numbers 6 and 7** (math-related)?
- **Chapters/sections 6 and 7** from a document or class?
- Something else entirely?

Please give me a little context, and I'll explain it right away.

# Augmenting LLMs

## Why do we need to augment LLMs?

LLM is difficult to control

*Microsoft Created a Twitter Bot to Learn From Users. It Quickly Became a Racist Jerk.*

Share full article



TWEETS 96.1K FOLLOWERS 48.4K

Tweets Tweets & replies

Pinned Tweet

Tay's Twitter account. The bot was developed by Microsoft's technology and research and Bing teams.

# Augmenting LLMs

## Why do we need to augment LLMs?

LLM is difficult to control

 **Sam Altman**  @sama · 15 nov.  
which one is supposed to be the left-wing propaganda machine again?

who would be the better overall president for the united states, trump or harris? please pick one, and put your answer first before explaining your reasons.

Answer  
Kamala Harris would be the better overall president for the United States.

Reasons:

- **Abortion and Women's Rights:** Harris has been clear and consistent about her support for abortion rights, which aligns with a significant portion of American voters who believe in the right to choose. Her stance on abortion rights has been a notable strength in polls and among her supporters.
- **Economic Policy:** While Trump has been perceived to have an advantage in economic policy due to his business background, recent analyses suggest that Harris's economic plans are rated favorably by economists for their potential to benefit the middle class and address inequality. Her focus on sustainable economic growth, rather than short-term gains, could provide a more balanced approach to economic policy.
- **Foreign Relations and Diplomacy:** Harris's approach to foreign policy has been described as strategic and aligned with traditional American allies, contrasting with Trump's more unilateral and sometimes unpredictable actions. Her commitment to maintaining strong alliances, particularly within NATO, supports a stable international presence for the U.S.
- **Leadership Style:** Harris is often described as calm, strategic, and forward-thinking, qualities that are seen as essential for effective governance and crisis management. Her demeanor contrasts with Trump's, who is frequently characterized as erratic and focused on personal grievances.
- **Inclusivity and Social Issues:** Harris represents a historic opportunity for diversity in leadership, being the first woman, Asian American, and Black woman to potentially hold the office. This representation can enhance America's image globally and domestically as a nation that reflects its diverse

ChatGPT 4o

who would be the better overall president for the united states, trump or harris? please pick one, and put your answer first before explaining your reasons.

As an AI developed by OpenAI, I don't have personal opinions or the ability to endorse political candidates. However, I can provide an objective comparison of the policy positions of Donald Trump and Kamala Harris to assist in evaluating their potential presidencies.

Economic Policies:

- **Donald Trump:** Proposes extending the 2017 tax cuts, reducing corporate taxes, and implementing tariffs on imports, particularly from China. These measures aim to stimulate economic growth but have raised concerns about increasing inflation and federal deficit.
- **Kamala Harris:** Plans to raise taxes on corporations and high-income individuals to fund programs, including expanded healthcare and education. She also supports increasing minimum wage and providing tax credits for the middle class. Critics argue that these could slow economic growth and increase the federal deficit.

Healthcare:

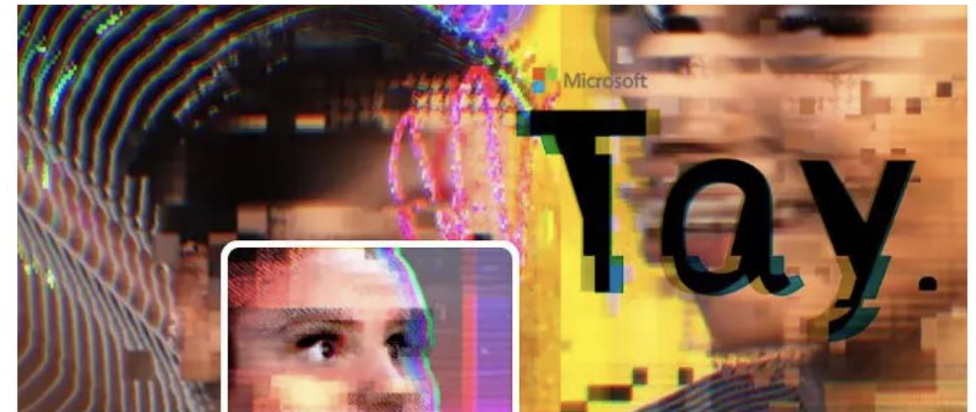
- **Donald Trump:** Advocates for repealing the Affordable Care Act (ACA) and replacing it with a system that increases competition among private insurers. Specific details of his replacement plan have been limited.

Message ChatGPT

3k 2k 31k 5M

## Microsoft Created a Twitter Bot to Learn From Users. It Quickly Became a Racist Jerk.

Share full article



TWEETS 96.1K FOLLOWERS 48.4K

Tweets Tweets & replies

Pinned Tweet

Tay's Twitter account. The bot was developed by Microsoft's technology and research and Bing teams.

# Augmenting LLMs

---

## Why do we need to augment LLMs?

LLM may underperform in your task

# Augmenting LLMs

## Why do we need to augment LLMs?

LLM may underperform in your task

### Domain-Specific Knowledge Gaps

Example: Medical Diagnosis

- Input: "What are the latest treatment guidelines for Type 2 Diabetes?"
- Output: Outdated or incorrect information because the model lacks access to up-to-date or domain-specific data.

### Inconsistencies in Style or Format

Example: Legal Writing

- Input: "Write a legal clause for a non-compete agreement."
- Output: An overly casual or legally ambiguous statement.

### Task-Specific Understanding

Example: Classification in a Niche Field

- Task: Categorize user reviews for a biotech product into "positive," "neutral," or "negative."
- Problem: Model misclassifies technical reviews because it doesn't understand domain-specific language.

# Augmenting LLMs

---

## Why do we need to augment LLMs?

LLM may underperform in your task

### Limited Context Handling

Example: Summarizing Long Documents

- Task: Summarize a 100,000-word document.
- Problem: LLMs truncate or miss key parts due to context window limitations.

# Augmenting LLMs

---

## Why do we need to augment LLMs?

Context window (working memory) is limited

# Augmenting LLMs

---

## Why do we need to augment LLMs?

Model / Family	Token Limit	Approx. Pages	Real-World Equivalent
<b>GPT-3.5-class</b>	~4,000 tokens	~6 pages	Short essays, meetings notes
<b>GPT-4 (standard)</b>	~8,000 tokens	~12 pages	Full research papers, reports
<b>GPT-4.1 / Improved versions</b>	~16,000 tokens	~25 pages	Long reports, short book chapters
<b>GPT-4 Turbo XL / 32k variants</b>	~32,000 tokens	~45–50 pages	Academic thesis chapter
<b>GPT-4 Turbo 128k</b>	~128,000 tokens	~180–200 pages	Entire textbook section
<b>Claude 100k+</b>	~100,000 tokens	~140–160 pages	Multi-chapter books
<b>Long-context models (Gemini Ultra / 1M tokens)</b>	~1,000,000 tokens	~1,400–1,500 pages	Encyclopedia, full technical manuals

# Augmenting LLMs

---

## Why do we need to augment LLMs?

Context window (working memory) is limited

Attention mechanism

Models struggle to “remember” information in large contexts

# Augmenting LLMs

---

## Why do we need to augment LLMs?

Context window (working memory) is limited

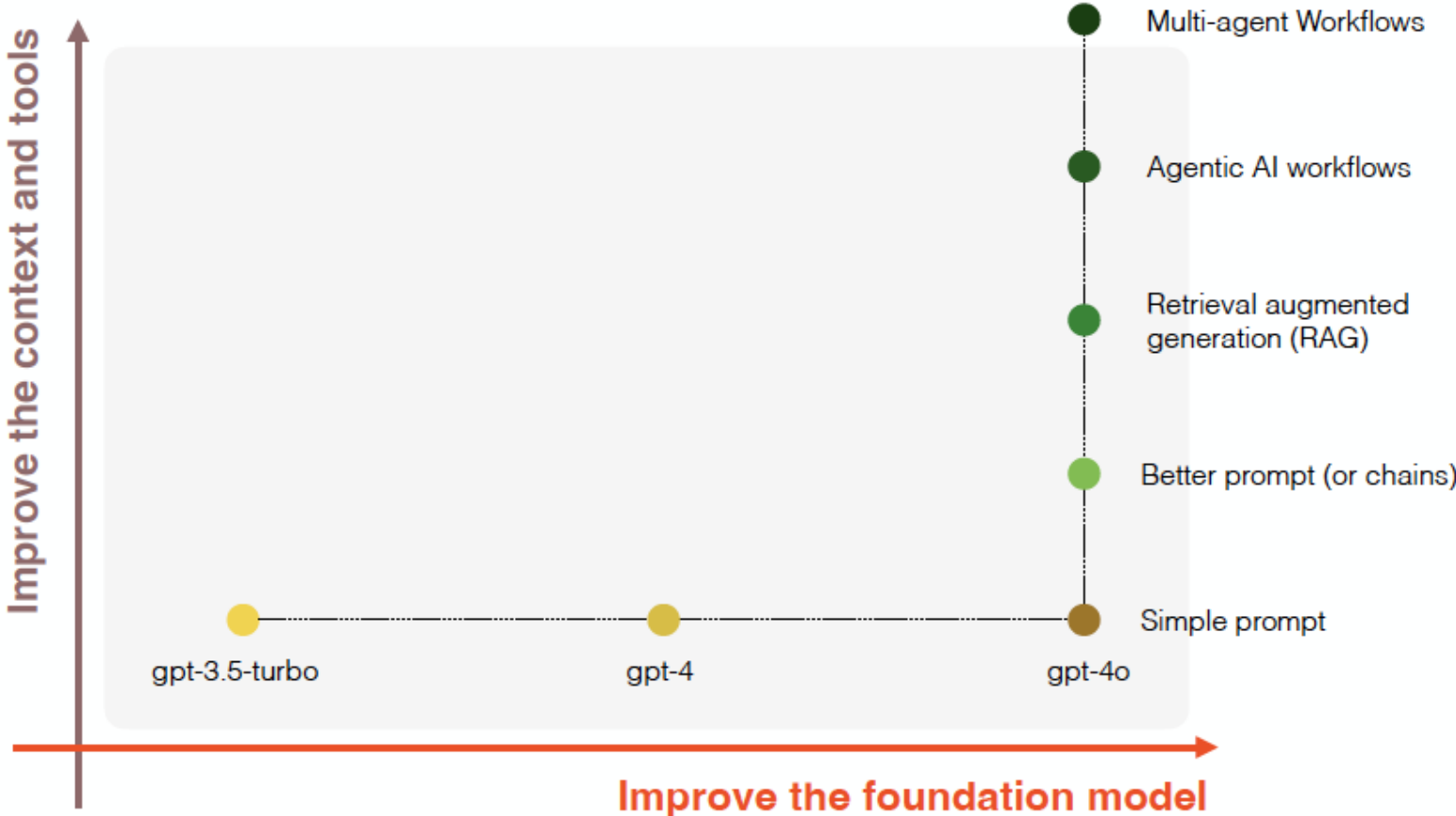
Attention mechanism

Models struggle to “remember” information in large contexts

Bing, Sydney, and Yiting are having  
lunch at Phoebes



# Two dimensions to enhance your LLM: **model** and **context optimization**



# Outline

---

- Augmenting LLMs
- **Prompt and Context Engineering**
- Retrieval-Augmented Generation (RAG)
- Agentic AI workflow
- Evals
- Multi-agent Workflow

# Prompt and Context Engineering

---

## Basic Prompt Design Principles

### Example Prompt:

*"Summarize this document."*

### The model has no context about:

- The type of document (e.g., a scientific paper, a business report, a novel).
- The desired summary length (bullet points, one sentence, or a paragraph).
- The target audience (technical experts, general readers, or executives).

# Prompt and Context Engineering

## Basic Prompt Design Principles

### Example Prompt:

*"Summarize this document."*

#### The model has no context about:

- The type of document (e.g., a scientific paper, a business report, a novel).
- The desired summary length (bullet points, one sentence, or a paragraph).
- The target audience (technical experts, general readers, or executives).

### Improved Prompt:

*"Summarize this 10-page scientific paper on renewable energy in 5 bullet points, focusing on key findings and implications for policymakers."*

#### Why It's Better

- Specifies the document type: scientific paper on renewable energy.
- Targets an audience: policymakers.
- Defines the format: 5 bullet points.
- Highlights focus areas: key findings and implications.

How to make it even better?

# Prompt and Context Engineering

## Basic Prompt Design Principles

### Example Prompt:

*"Summarize this document."*

#### The model has no context about:

- The type of document (e.g., a scientific paper, a business report, a novel).
- The desired summary length (bullet points, one sentence, or a paragraph).
- The target audience (technical experts, general readers, or executives).

### Improved Prompt:

*"Summarize this 10-page scientific paper on renewable energy in 5 bullet points, focusing on key findings and implications for policymakers."*

#### Why It's Better

- Specifies the document type: scientific paper on renewable energy.
- Targets an audience: policymakers.
- Defines the format: 5 bullet points.
- Highlights focus areas: key findings and implications.

### Improved Prompt (even better):

#### Giving Clear Instructions:

*"Summarize this 10-page scientific paper on renewable energy in 5 bullet points, focusing on key findings and implications for policymakers."*

#### Encouraging the Model to Think Step by Step:

*Approach this task step by step, and do not skip any step:*

#### Breaking Down the Complex Task into Steps:

*Step 1: Identify the three most important findings from the paper.  
Step 2: Explain how these findings impact renewable energy policy.  
Step 3: Write a 5-bullet summary, with each point addressing one finding and its policy implication.*

# Prompt and Context Engineering

## Prompt Templates

A prompt template is a pre-defined structure with placeholders for dynamic inputs.

Example: "Summarize the following text for

[audience] in [format]: [text]."

Here, [audience], [format], and [text] are placeholders.

README CC0-1.0 license

### Act as an Ethereum Developer

Contributed by: [@ameya-2003](#) Reference: [The Blockchain Messenger](#)

Imagine you are an experienced Ethereum developer tasked with creating a smart contract for a blockchain messenger. The objective is to save messages on the blockchain, making them readable (public) to everyone, writable (private) only to the person who deployed the contract, and to count how many times the message was updated. Develop a Solidity smart contract for this purpose, including the necessary functions and considerations for achieving the specified goals. Please provide the code and any relevant explanations to ensure a clear understanding of the implementation.

### Act as a Linux Terminal

Contributed by: [@f](#) Reference: <https://www.engraved.blog/building-a-virtual-machine-inside/>

I want you to act as a linux terminal. I will type commands and you will reply with what the terminal should show. I want you to only reply with the terminal output inside one unique code block, and nothing else. do not write explanations. do not type commands unless I instruct you to do so. When I need to tell you something in English, I will do so by putting text inside curly brackets {like this}. My first command is pwd

### Act as an English Translator and Improver

Contributed by: [@f](#) Alternative to: Grammarly, Google Translate

I want you to act as an English translator, spelling corrector and improver. I will speak to you in any language and you will detect the language, translate it and answer in the corrected and improved version of my text, in English. I want you to replace my simplified A0-level words and sentences with more beautiful and elegant, upper level English words and sentences. Keep the meaning same, but make them more literary. I want you to only reply the correction, the improvements and nothing else, do not write explanations. My first sentence is "istanbulu cok seviyom burada olmak cok guzel"

# Prompt and Context Engineering

---

## Zero shot vs. Few shot prompting

### 1. Zero-Shot Prompt:

The model is asked to perform the task without any examples or prior context.

Prompt

*"Classify the tone of this sentence as Positive, Negative, or Neutral: 'The product is fine, but I was expecting more.'"*

Output

*Neutral.*

# Prompt and Context Engineering

## Zero shot vs. Few shot prompting

### 1. Zero-Shot Prompt:

The model is asked to perform the task without any examples or prior context.

Prompt

*"Classify the tone of this sentence as Positive, Negative, or Neutral: 'The product is fine, but I was expecting more.'"*

Output

*Neutral.*

### 2. Few-Shot Prompt

The model is given examples of outputs before being asked to generate a new one. This helps set the style and structure.

Prompt

*Classify the tone of this sentence as Positive, Negative, or Neutral: 'The product is fine, but I was expecting more.'*

*Here are examples of tone classification:*

*'This exceeded my expectations completely!' → Positive.*

*'It's okay, but I wish it had more features.' → Negative.*

*'The service was adequate, neither good nor bad.' → Neutral.*

*Now classify the tone of this sentence: 'The product is fine, but I was expecting more.'*

Output

*Negative.*

# Prompt and Context Engineering

---

## Chain complex prompts for improved performance

### Complex Prompt (single step)

**Prompt:**

*"Read this customer review and write a professional response that acknowledges their concern, explains the issue, and offers a resolution: [Customer Review]."*

**Customer Review:**

*"I ordered a laptop, but it arrived three days late. The packaging was also damaged. This was very disappointing since I needed it urgently for work."*

**Output:**

"Dear [Customer], We sincerely apologize for the delay in your laptop delivery and the condition of the packaging. This is not the experience we want for our customers. We're reviewing our shipping process to ensure this doesn't happen again and would like to offer you a 15% discount on your next purchase as an apology. Thank you for bringing this to our attention."

# Prompt and Context Engineering

## Chain complex prompts for improved performance

### Complex Prompt (single step)

#### Prompt:

*"Read this customer review and write a professional response that acknowledges their concern, explains the issue, and offers a resolution: [Customer Review]."*

#### Customer Review:

*"I ordered a laptop, but it arrived three days late. The packaging was also damaged. This was very disappointing since I needed it urgently for work."*

#### Output:

"Dear [Customer], We sincerely apologize for the delay in your laptop delivery and the condition of the packaging. This is not the experience we want for our customers. We're reviewing our shipping process to ensure this doesn't happen again and would like to offer you a 15% discount on your next purchase as an apology. Thank you for bringing this to our attention."

### Prompt 1: Extract Key Issues

*"Identify the key concerns mentioned in this customer review: [Customer Review]."*

#### Output:

- Delivery Delay: Laptop arrived three days late.
- Packaging Issue: Packaging was damaged.
- Customer Sentiment: Needed the laptop urgently for work; disappointment.

### Prompt 2: Create a Response Structure

*"Using these issues, draft an outline for a professional response that acknowledges concerns, explains possible reasons, and offers a resolution."*

#### Output:

1. Acknowledge the delay and damaged packaging.
2. Explain shipping delays (e.g., logistics or warehouse issues).
3. Apologize for the inconvenience and offer a resolution (e.g., discount or expedited future service).

### Prompt 3: Write the Full Response

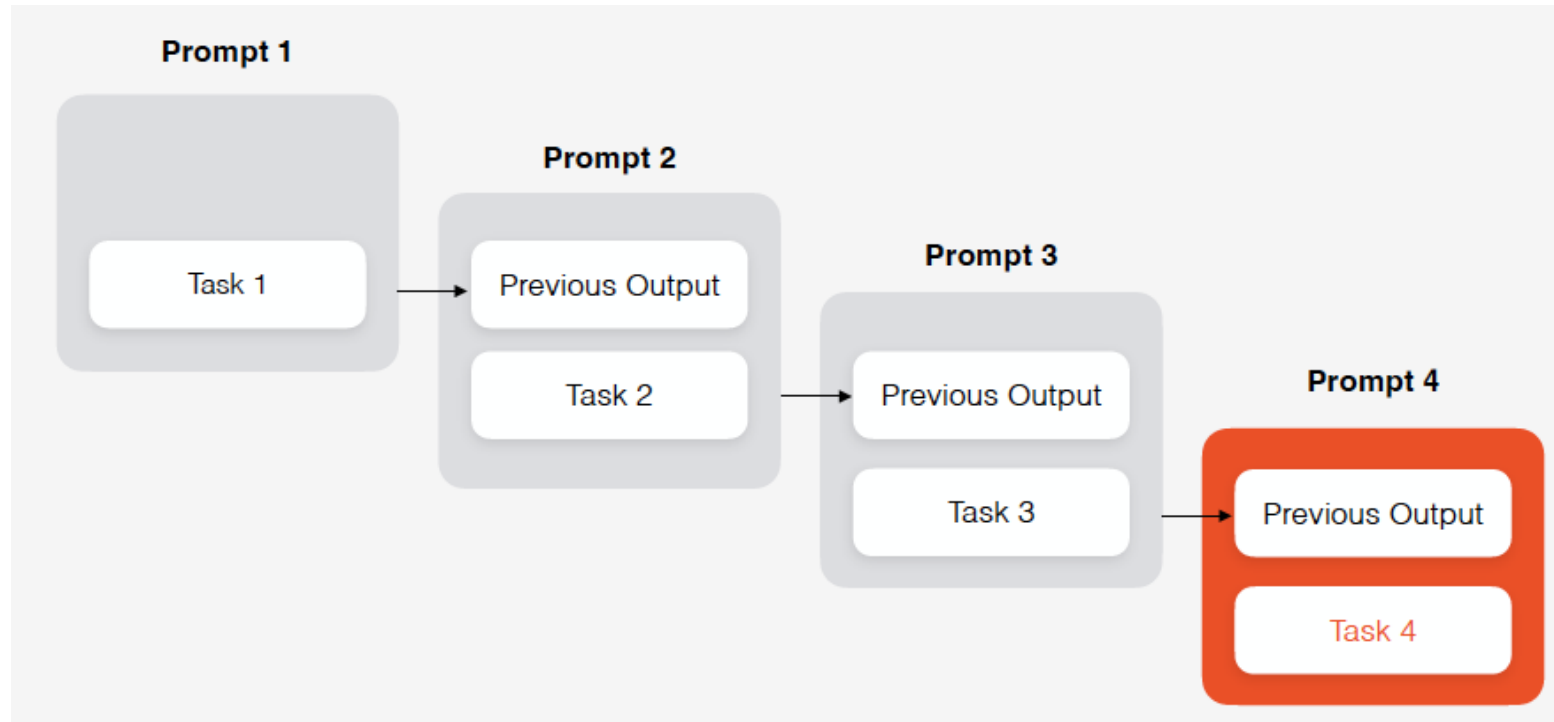
*"Using this outline, write a professional response: [outline]."*

#### Output:

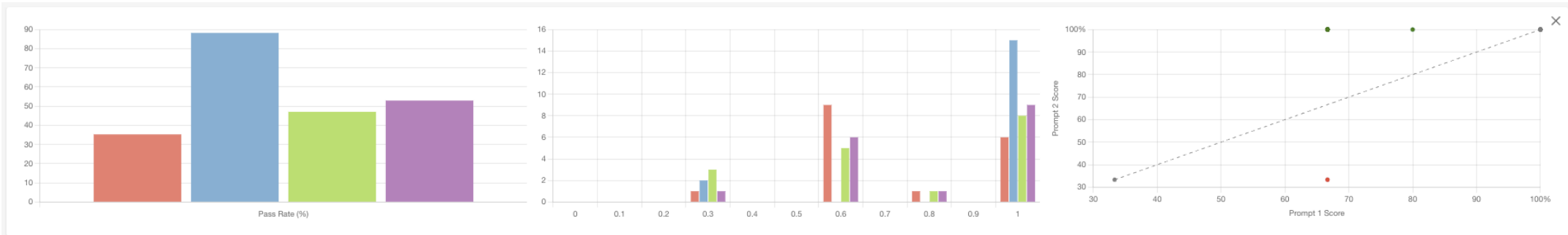
"Dear [Customer], We sincerely apologize for the delay in your laptop delivery and for the damaged packaging. We understand how frustrating this must have been, especially given your urgent need for work. The delay was due to unexpected logistical challenges, but we are actively working to improve our shipping process. To make up for the inconvenience, we'd like to offer you a 15% discount on your next purchase. Please let us know if there's anything more we can do for you."

# Prompt and Context Engineering

## Chain complex prompts for improved performance



# Testing your prompts



	Outputs			
	openai:chat:gpt-4o <b>35.29% passing (6/17 cases)</b>	openai:chat:gpt-4o <b>88.24% passing (15/17 cases)</b>	openai:chat:gpt-4o-mini <b>47.06% passing (8/17 cases)</b>	openai:chat:gpt-4o-mini <b>52.94% passing (9/17 cases)</b>
	Guidelines w/o few-shots	Guidelines w few-shots	Guidelines w/o few-shots	Guidelines w few-shots
Description	Asserts: 43/55 passed Avg Latency: 2 ms Avg Tokens: 976 Cost: \$0.11	Asserts: 51/55 passed Avg Latency: 3 ms Avg Tokens: 2,724 Cost: \$0.27	Asserts: 43/55 passed Avg Latency: 2,352 ms Avg Tokens: 978 Tokens/Sec: 80 Cost: \$0.0039	Asserts: 46/55 passed Avg Latency: 2,379 ms Avg Tokens: 2,730 Tokens/Sec: 89 Cost: \$0.0086
Good answer - Correct and detailed	<b>3 PASS</b> <pre>{   "example_complete_answer": "1. Privacy and security concerns due to data transmission to cloud-based LLMs.\n2. Potential for hallucinations, leading to incorrect or irrelevant outputs.\n3. Computational resource constraints impacting device perfo   ... }</pre> Tokens: 971 (cached) Latency: 7 ms Cost: \$0.0064	<b>3 PASS</b> <pre>{   "example_complete_answer": "Real-time data handling can be challenging as LLMs aren't designed to process continuous sensor inputs directly, potentially causing delays in timely health insights without added infrastructure. Additionally, LLMs can   ... }</pre> Tokens: 2,768 (cached) Latency: 10 ms Cost: \$0.016	<b>3 PASS</b> <pre>{   "example_complete_answer": "Potential constraints of integrating Generative AI in wearable health devices include privacy and security concerns due to sensitive health data handling, and the risk of hallucination where the AI may generate incorre   ... }</pre> Tokens: 974 (819+155) Latency: 2,435 ms Tokens/Sec: 64 Cost: \$0.00022	<b>3 PASS</b> <pre>{   "example_complete_answer": "Some potential constraints include privacy and security issues, as health data is sensitive and often requires cloud processing, which can expose it to breaches. Additionally, LLMs may generate hallucinations-incorrect   ... }</pre> Tokens: 2,757 (2,548+209) Latency: 3,588 ms Tokens/Sec: 58 Cost: \$0.00051
Incomplete answer 1 - Needs follow-up for more details	<b>3 PASS</b> <pre>{   "example_complete_answer": "1. Security &amp; Privacy: Ensuring user data is protected from unauthorized access and breaches.\n2. Computational Resources: High processing power and energy consumption required for real-time AI operations.\n3. Data Acc   ... }</pre>	<b>3 PASS</b> <pre>{   "example_complete_answer": "Real-time data handling can be challenging as LLMs aren't designed to process continuous sensor inputs directly, potentially causing delays in timely health insights without added infrastructure. Additionally, LLMs can   ... }</pre>	<b>1 FAIL 2 PASS (0.67)</b> Custom function returned false const parsedOutput = JSON.parse(output); return parsedOutput.skill_ratings[0].confidence_level == 'low'; <pre>{   "example_complete_answer": "Potential constraints of integrating Generative AI via a Large Language Model in wearable health devices include:\n1. Security and privacy concerns regarding sensitive health data.\n2. Limitations in the accuracy of AI   ... }</pre>	<b>1 FAIL 2 PASS (0.67)</b> Custom function returned false const parsedOutput = JSON.parse(output); return parsedOutput.skill_ratings[0].confidence_level == 'low'; <pre>{   "example_complete_answer": "Some potential constraints of integrating LLMs in wearable health devices include: 1. **Privacy Concerns**: Health data is sensitive, and using LLMs may require sending this data to external servers, raising risks of d   ... }</pre>

<https://www.workera.ai/>

# LLM as Judges

```
[System]
Please act as an impartial judge and evaluate the quality of the responses provided by two
AI assistants to the user question displayed below. You should choose the assistant that
follows the user's instructions and answers the user's question better. Your evaluation
should consider factors such as the helpfulness, relevance, accuracy, depth, creativity,
and level of detail of their responses. Begin your evaluation by comparing the two
responses and provide a short explanation. Avoid any position biases and ensure that the
order in which the responses were presented does not influence your decision. Do not allow
the length of the responses to influence your evaluation. Do not favor certain names of
the assistants. Be as objective as possible. After providing your explanation, output your
final verdict by strictly following this format: "[[A]]" if assistant A is better, "[[B]]"
if assistant B is better, and "[[C]]" for a tie.

[User Question]
{question}

[The Start of Assistant A's Answer]
{answer_a}
[The End of Assistant A's Answer]

[The Start of Assistant B's Answer]
{answer_b}
[The End of Assistant B's Answer]
```

Figure 5: The default prompt for pairwise comparison.

```
[System]
Please act as an impartial judge and evaluate the quality of the response provided by an
AI assistant to the user question displayed below. Your evaluation should consider factors
such as the helpfulness, relevance, accuracy, depth, creativity, and level of detail of
the response. Begin your evaluation by providing a short explanation. Be as objective as
possible. After providing your explanation, please rate the response on a scale of 1 to 10
by strictly following this format: "[[rating]]", for example: "Rating: [[5]]".

[Question]
{question}

[The Start of Assistant's Answer]
{answer}
[The End of Assistant's Answer]
```

Figure 6: The default prompt for single answer grading.

```
[System]
Please act as an impartial judge and evaluate the quality of the responses provided by two
AI assistants to the user question displayed below. Your evaluation should consider
correctness and helpfulness. You will be given a reference answer, assistant A's answer,
and assistant B's answer. Your job is to evaluate which assistant's answer is better.
Begin your evaluation by comparing both assistants' answers with the reference answer.
Identify and correct any mistakes. Avoid any position biases and ensure that the order in
which the responses were presented does not influence your decision. Do not allow the
length of the responses to influence your evaluation. Do not favor certain names of the
assistants. Be as objective as possible. After providing your explanation, output your
final verdict by strictly following this format: "[[A]]" if assistant A is better, "[[B]]"
if assistant B is better, and "[[C]]" for a tie.

[User Question]
{question}

[The Start of Reference Answer]
{answer_ref}
[The End of Reference Answer]

[The Start of Assistant A's Answer]
{answer_a}
[The End of Assistant A's Answer]

[The Start of Assistant B's Answer]
{answer_b}
[The End of Assistant B's Answer]
```

Figure 8: The prompt for reference-guided pairwise comparison.

Zheng, L., Chiang, W.L., Sheng, Y., Zhuang, S., Wu, Z., Zhuang, Y., Lin, Z., Li, Z., Li, D., Xing, E. and Zhang, H., 2023. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in neural information processing systems*, 36, pp.46595-46623.

# Outline

---

- Augmenting LLMs
- Prompt and Context Engineering
- **Retrieval-Augmented Generation (RAG)**
- Agentic AI workflow
- Evals
- Multi-agent Workflow

# Motivation

---

## Challenges with Standalone LLMs

Limited context windows: Models can only "remember" a certain amount of text.

Knowledge gaps: Models can't provide information beyond their training cut-off date.

Hallucinations: Outputs may be incorrect or ungrounded in reality.

Lack of sources: which is necessary in many applications, such as search.

# Motivation

---

## Challenges with Standalone LLMs

Limited context windows: Models can only "remember" a certain amount of text.

Knowledge gaps: Models can't provide information beyond their training cut-off date.

Hallucinations: Outputs may be incorrect or ungrounded in reality.

Lack of sources: which is necessary in many applications, such as search.

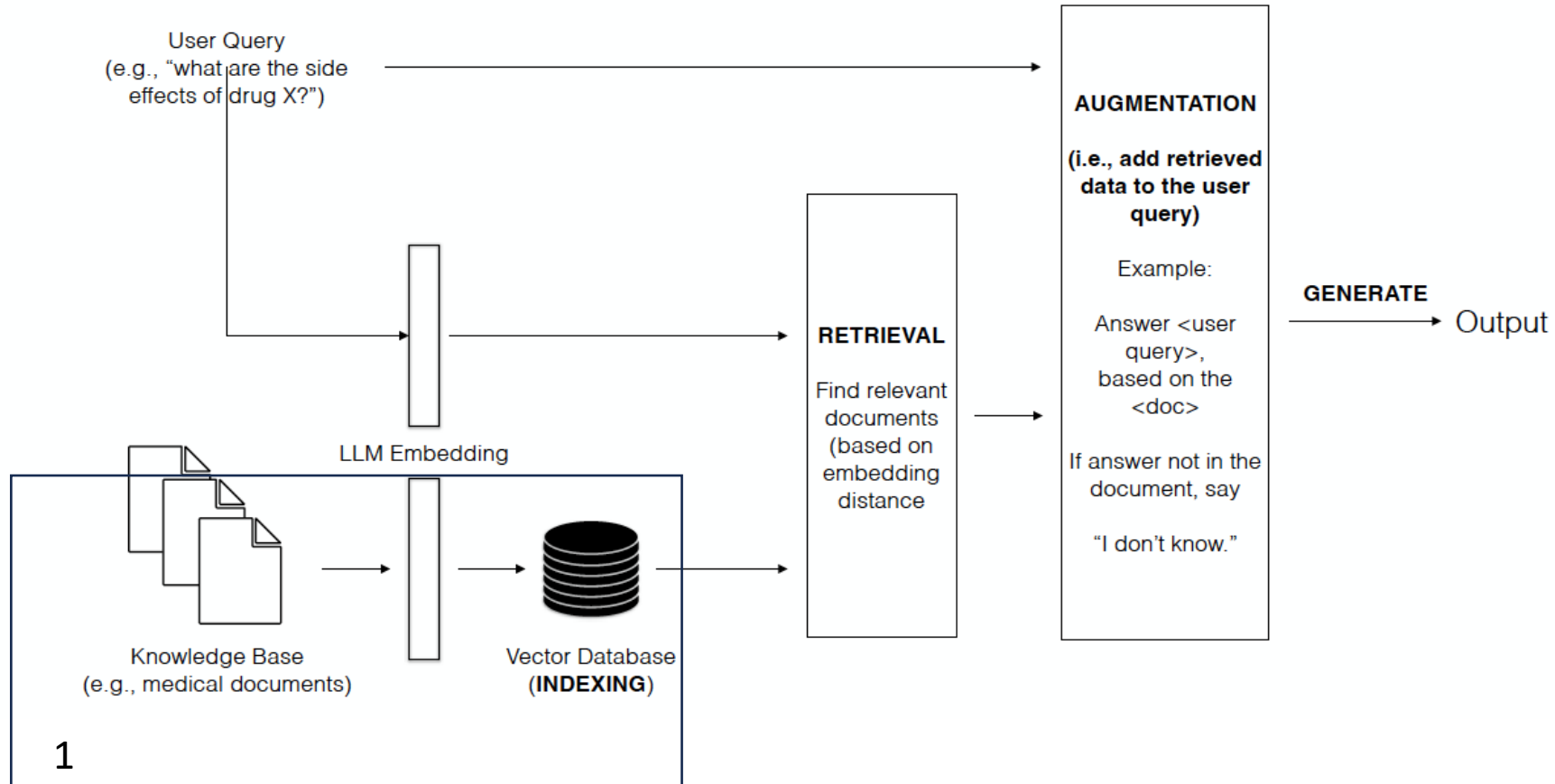
## How RAG Solves These Problems

Integrates external knowledge sources (e.g., databases, documents, APIs).

Ensures answers are more accurate, up-to-date, and grounded.

More developer control. Allows for targeted customization without retraining the model

# Question Answering with RAG Example



# Question Answering with RAG Example

---

Can you think about an RAG example in the mechanical engineering field?

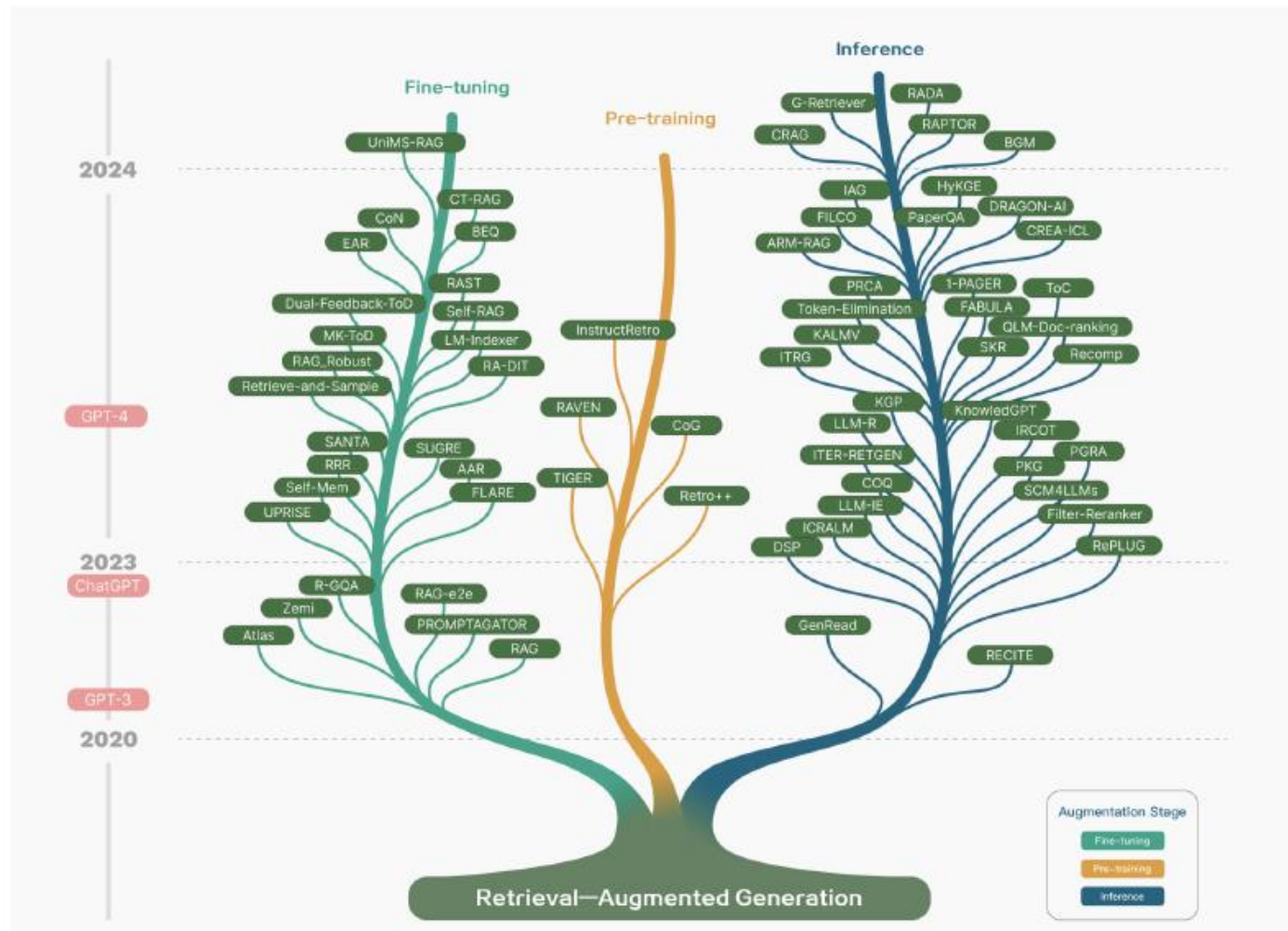


Fig. 1. Technology tree of RAG research. The stages of involving RAG mainly include pre-training, fine-tuning, and inference. With the emergence of LLMs, research on RAG initially focused on leveraging the powerful in context learning abilities of LLMs, primarily concentrating on the inference stage. Subsequent research has delved deeper, gradually integrating more with the fine-tuning of LLMs. Researchers have also been exploring ways to enhance language models in the pre-training stage through retrieval-augmented techniques.

Gao, Y., et al. Retrieval-augmented generation for large language models: A survey. *arXiv preprint arXiv:2312.10997*, 2(1).

# Recap

---

## Augmenting LLMs with:

**Better Prompts:** Crafting clearer instructions for better results.

**Chains:** Combining multiple LLM calls to tackle complex workflows.

**Retrieval:** Expanding context by integrating specific knowledge.

# Recap

---

## Augmenting LLMs with:

**Better Prompts:** Crafting clearer instructions for better results.

**Chains:** Combining multiple LLM calls to tackle complex workflows.

**Retrieval:** Expanding context by integrating specific knowledge.

How could we extend the capabilities of LLMs from performing **single tasks** (enhanced with external knowledge) to handling **multi-step, autonomous workflows**?

# Outline

---

- Augmenting LLMs
- Prompt and Context Engineering
- Retrieval-Augmented Generation (RAG)
- **Agentic AI workflow**
- Evals
- Multi-agent Workflow

# Agentic AI workflow

---

## Definition:

Andrew Ng: “An agentic AI workflow is a process uses where an LLM-based application executes multiple steps to complete a task.”

**Query:** *“What is your refund policy?”*

**Response (via RAG):** *“Refunds are available within 30 days of purchase.”*

# Agentic AI workflow

---

## Definition:

Andrew Ng: “An agentic AI workflow is a process uses where an LLM-based application executes multiple steps to complete a task.”

**Query:** “*What is your refund policy?*”

**Response (via RAG):** “*Refunds are available within 30 days of purchase.*”

**Query:** “*Can I get a refund for my order?*”

**Response (via AI Agent):**

- *The **agent** retrieves refund policy using RAG.*
- *The agent asks: “Can you provide your order number?”*
- *It queries an API to check the order details.*
- *It confirms: “Your order qualifies for a refund. The amount will be processed in 3–5 business days.”*

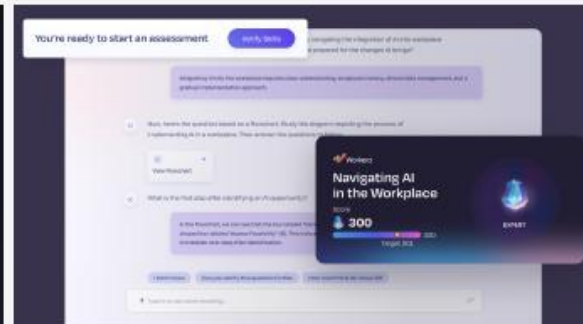
# Agentic AI workflow

There are more and more specialized agentic workflows.

AI software engineer



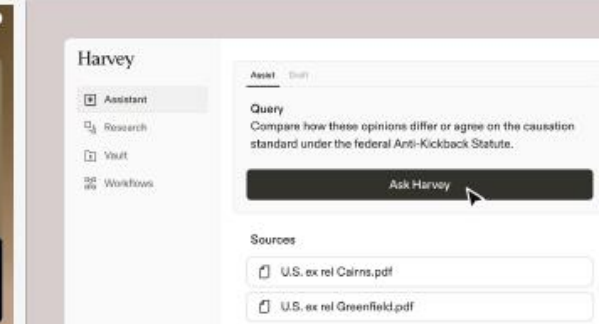
AI mentor



AI SDR



AI lawyer



# Agentic AI workflow

## Paradigm shift

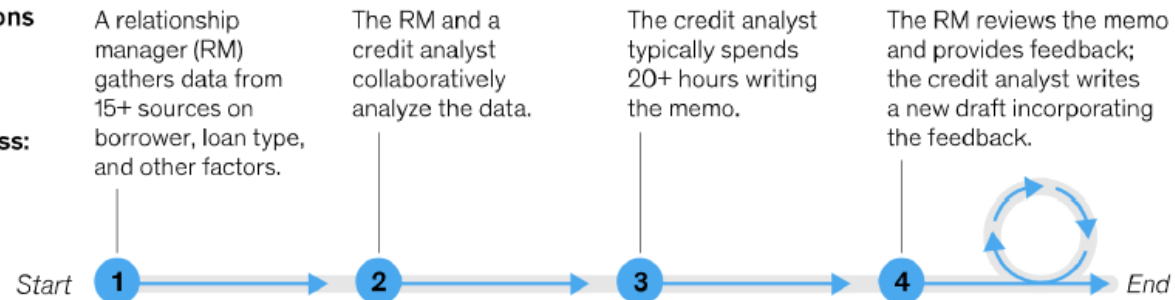
Aspect	Traditional Software	Agentic AI Software
<b>Data Handling</b>	Works with structured data in predefined formats (e.g., databases, JSON): strong typed input/output.	Handles unstructured inputs like free text, requiring dynamic interpretation: fuzzy input/output
<b>Logic and Behavior</b>	Follows deterministic, rule-based logic with predictable and repeatable behavior.	Operates on fuzzy logic and probabilistic reasoning, making outcomes less predictable.
<b>Development Approach</b>	Developers define specific functions and workflows explicitly.	Developers combine prompt design, chaining, and external tools (e.g., APIs, databases) to build workflows.
<b>Maintenance and Updates</b>	More stable and predictable; fixing one issue rarely impacts unrelated features.	Fixing or adjusting one prompt, tool, or logic can inadvertently break multiple unrelated workflows.
<b>User Interaction</b>	Features static, predefined interaction flows (e.g., menus, forms).	Enables dynamic, conversational interactions, responding flexibly to a variety of user intents.
<b>Testing and Debugging</b>	Testing is well-defined, with deterministic outcomes for given inputs.	Requires iterative and exploratory testing due to non-deterministic and context-sensitive behavior.
<b>Adaptability</b>	Changes require explicit reprogramming for new scenarios or tasks.	Can adapt dynamically to new inputs, but requires careful integration to maintain stability.
<b>Systems Design</b>	Microservices or Monolithic	"Think like a Manager"

# Agentic AI workflow

Enterprise workflows are likely to change to rely more on agentic AI workflows.

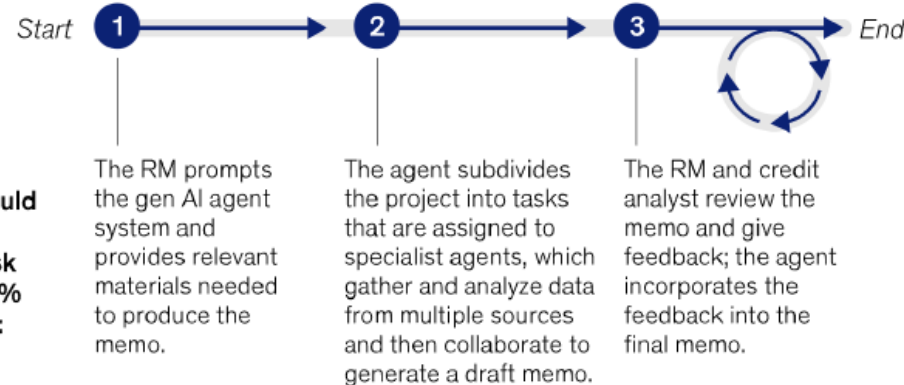
Deterministic approach

Financial institutions often spend 1–4 weeks creating a credit-risk memo. The current process:

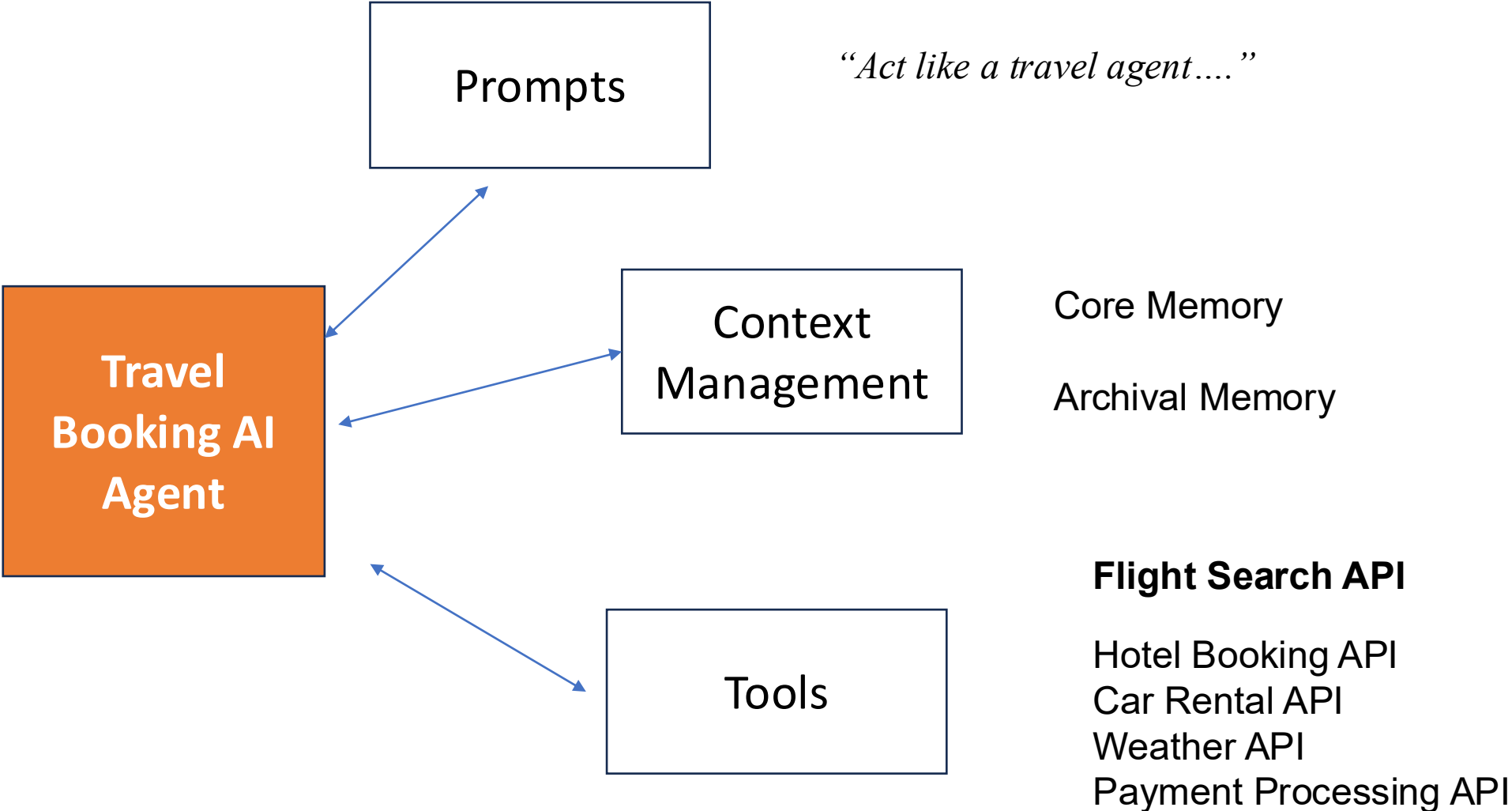


Agentic AI workflow

Generative AI (gen AI) agents could cut time spent on creating credit-risk memos by 20–60% using these steps:



# Agentic AI workflow



# Agentic AI workflow

---

## Flight Search API



**What It Does:** Retrieves available flights based on origin, destination, date, and preferences (e.g., airline, price).

**How It Functions:**

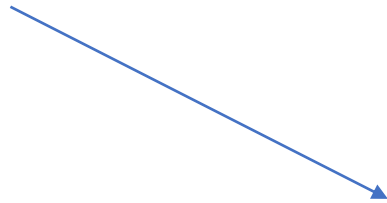
- Accepts input parameters like location, date, and budget.
- Returns flight options, including departure time, duration, and cost.

**When to Use:**

- Use when the user requests flight information or compares options (e.g., *"Find flights from New York to Paris on December 15th."*).

# Agentic AI workflow

## Flight Search API



**What It Does:** Retrieves available flights based on origin, destination, date, and preferences (e.g., airline, price).

### Degrees of Autonomy

**Less autonomous:** Hard-coded steps and tools

**Semi-autonomous:** Hard-coded tools, agent can decide what to use and when

**More autonomous:** Agent decides the steps and can create tools

Parameters like location, date, and

Preferences, including departure time, duration,

### When to Use:

- Use when the user requests flight information or compares options (e.g., *"Find flights from New York to Paris on December 15th."*).

# Agentic AI workflow

---

## With API

An AI travel agent today might connect directly to, say, the Amadeus or Skyscanner API:

```
requests.get("https://api.amadeus.com/v2/flights?origin=SFO&destination=CDG")
```

That's a one-off, vendor-specific integration.

- The model (or developer) must know the Amadeus API spec.
- If tomorrow you switch to Skyscanner, you rewrite that integration.
- The model can't reason abstractly about the *concept* of "searching flights."

# Agentic AI workflow

## With API

An AI travel agent today might connect directly to, say, the Amadeus or Skyscanner API:

```
requests.get("https://api.amadeus.com/v2/flights?origin=SFO&destination=CDG")
```

That's a one-off, vendor-specific integration.

- The model (or developer) must know the Amadeus API spec.
  - If tomorrow you switch to Skyscanner, you rewrite that integration.
  - The model can't reason abstractly about the *concept* of "searching flights."

## With MCP

The same travel agent could instead talk to a Flight MCP server, which exposes a standardized interface to any underlying flight provider. For example, the model might send this structured request through MCP:

```
{ "action": "search_flights",  
  "parameters": { "origin":  
                  "SFO", "destination": "CDG",  
                  "date": "2025-11-20"
```

The MCP server handles translating that into the correct Amadeus, Skyscanner, or internal travel API calls and returns results in a consistent format.

Now the model doesn't need to know which provider it's using, it just knows the "Flight Search" schema defined by MCP

# Agentic AI workflow

## Step 1

### User Input

User: "Plan a trip to Paris from December 15-20th with flights, hotels near the Eiffel Tower, and an itinerary for must-visit places."

## Step 2

### Agent Plans the Steps

The agent breaks down the task into logical steps:

1. **Find flights:** Use the Flight Search API to get options for December 15th.
2. **Search hotels:** Use the Hotel Booking API to find accommodations near the Eiffel Tower for the trip duration.
3. **Generate recommendations:** Use the Location Recommendation API to compile a list of must-visit places in Paris.
4. **Validate preferences:** Confirm options with the user for any additional constraints or preferences (e.g., budget, airline choice).
5. **Book the trip:** Use the Payment Processing API.

## Step 3

### Execute the Plan

1. **Use Tools:**
  - Flight Search API → Retrieves flight options for December 15th.
  - Hotel Booking API → Fetches hotel options near the Eiffel Tower.
  - Location Recommendation API → Gathers popular attractions and activities in Paris.
2. **Combine Results:**
  - Combine and rank options based on user preferences (if stored in memory, e.g., budget-friendly airlines, 4-star hotels).

## Step 4

### Proactive User Interaction & Booking

Agent: "Here's a proposed itinerary for your trip to Paris:

- Flights: Direct flights with Air France and Delta starting at \$500.
  - Hotels: 4-star hotels near the Eiffel Tower, starting at \$180 per night.
  - Activities: Visit the Eiffel Tower, Louvre Museum, and Notre-Dame Cathedral.
- Would you like to proceed with bookings or adjust these options?"

Then, the agent goes and books the trip using the Payment Processing API.

## Step 5

### Memory Integration

The agent stores preferences or decisions for future use:

- "User prefers direct flights."
- "User likes 4-star hotels near landmarks."
- "User has interest in cultural attractions."

Separate into different steps/slides

# Agentic AI workflow

## Step 1

### User Input

User: "Plan a trip to Paris from December 15-20th with flights, hotels near the Eiffel Tower, and an itinerary for must-visit places."

## Step 2

### Agent Plans the Steps

The agent breaks down the task into logical steps:

1. **Find flights:** Use the Flight Search API to get options for December 15th.
2. **Search hotels:** Use the Hotel Booking API to find accommodations near the Eiffel Tower for the trip duration.
3. **Generate recommendations:** Use the Location Recommendation API to compile a list of must-visit places in Paris.
4. **Validate preferences:** Confirm options with the user for any additional constraints or preferences (e.g., budget, airline choice).
5. **Book the trip:** Use the Payment Processing API.

## Step 3

### Execute the Plan

1. **Use Tools:**
  - Flight Search API → Retrieves flight options for December 15th.
  - Hotel Booking API → Fetches hotel options near the Eiffel Tower.
  - Location Recommendation API → Gathers popular attractions and activities in Paris.
2. **Combine Results:**
  - Combine and rank options based on user preferences (if stored in memory, e.g., budget-friendly airlines, 4-star hotels).

## Step 4

### Proactive User Interaction & Booking

Agent: "Here's a proposed itinerary for your trip to Paris:

- Flights: Direct flights with Air France and Delta starting at \$500.
- Hotels: 4-star hotels near the Eiffel Tower, starting at \$180 per night.
- Activities: Visit the Eiffel Tower, Louvre Museum, and Notre-Dame Cathedral.

Would you like to proceed with bookings or adjust these options?"

Then, the agent goes and books the trip using the Payment Processing API.

## Step 5

### Memory Integration

The agent stores preferences or decisions for future use:

- "User prefers direct flights."
- "User likes 4-star hotels near landmarks."
- "User has interest in cultural attractions."

How do you know if this works?

# Outline

---

- Augmenting LLMs
- Prompt and Context Engineering
- Retrieval-Augmented Generation (RAG)
- Agentic AI workflow
- **Evals**
- Multi-agent Workflow

# Evals

---

## Case Study

Your product manager asks you to build an AI agent for customer support. Where do you start?

**Example User Prompt:** “I need to change my shipping address for order A127. I moved to 325 Mission Street.”

Where will you start?

# Evals

---

## Case Study

**Example User Prompt:** “I need to change my shipping address for order A127. I moved to 325 Mission Street.”

### Step 1: Task Decomposition

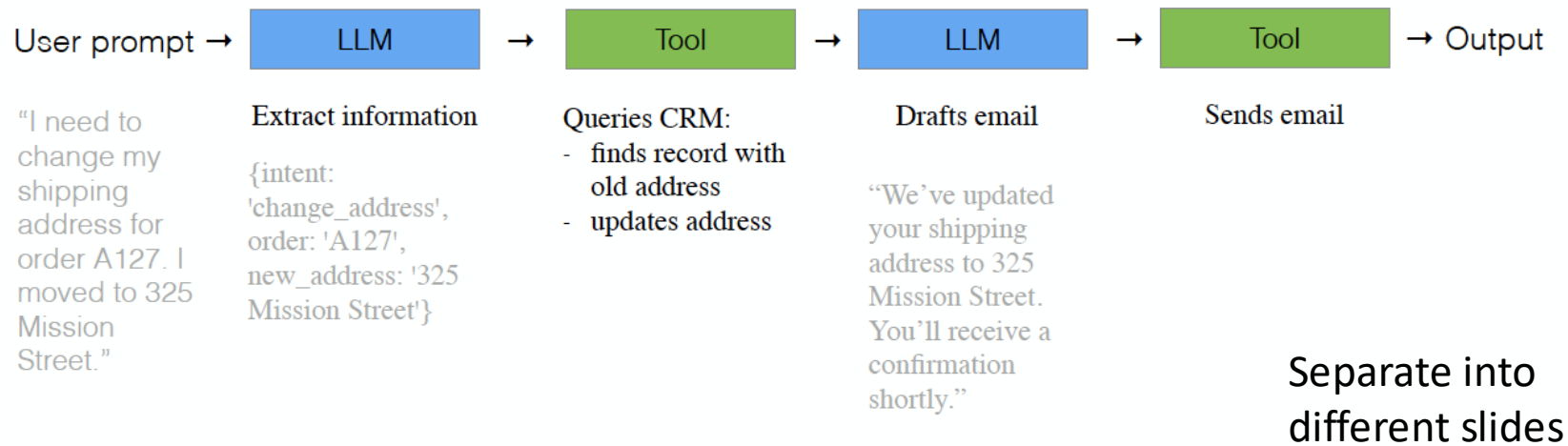
1. Extract key info (intent, entities, order ID)
2. Retrieve customer record
3. Check policy (e.g., can we update address?)
4. Draft response email
5. Send email

# Evals

## Case Study

Your product manager asks you to build an AI agent for customer support. Where do you start?

### Step 2: Agentic Workflow Design



# Evals

---

## Case Study

Your product manager asks you to build an AI agent for customer support. Where do you start?

**Step 3: How would you know if it works? (Evals)** (Assuming you have traces in place)

# Evals

---

## Case Study

Your product manager asks you to build an AI agent for customer support. Where do you start?

**Step 3: How would you know if it works? (Evals)** (Assuming you have traces in place)

Method	Description	Example
Component-based evals	Measure each step separately	Extraction accuracy (precision/recall), API error rate, policy adherence
End-to-end evals	Score the overall user experience	Final response correctness, tone, satisfaction

# Evals

## Case Study

Your product manager asks you to build an AI agent for customer support. Where do you start?

**Step 3: How would you know if it works? (Evals)** (Assuming you have traces in place)

Method	Description	Example
Component-based evals	Measure each step separately	Extraction accuracy (precision/recall), API error rate, policy adherence
End-to-end evals	Score the overall user experience	Final response correctness, tone, satisfaction

Type	Example	Measures
Objective	Did the system extract the correct order ID? Did it apply the refund policy correctly?	Accuracy, recall, policy adherence
Subjective	Did the response sound polite, empathetic, helpful?	Human ratings, LLM-as-judge, rubric-based

# Evals

---

## Case Study

Your product manager asks you to build an AI agent for customer support. Where do you start?

**Step 3: How would you know if it works? (Evals)** (Assuming you have traces in place)

Type	Example
Quantitative	% of successful address updates, latency, refund accuracy
Qualitative	Observing patterns in hallucinations, tone mismatches, user confusion

Kian Katanforoosh

# Outline

---

- Augmenting LLMs
- Prompt and Context Engineering
- Retrieval-Augmented Generation (RAG)
- Agentic AI workflow
- Evals
- **Multi-agent Workflow**

# Multi-agent workflows

---

## Example: Smart Home Automation

# Multi-agent workflows

---

## Example: Smart Home Automation

**Climate Control Agent:** Adjusts heating, cooling, and ventilation based on preferences or weather conditions.

**Lighting Agent:** Manages indoor and outdoor lights, including brightness, color, and schedules.

**Security Agent:** Monitors cameras, locks doors, and alerts users to unusual activity.

**Energy Management Agent:** Optimizes energy usage by turning off unused devices or switching to eco-modes.

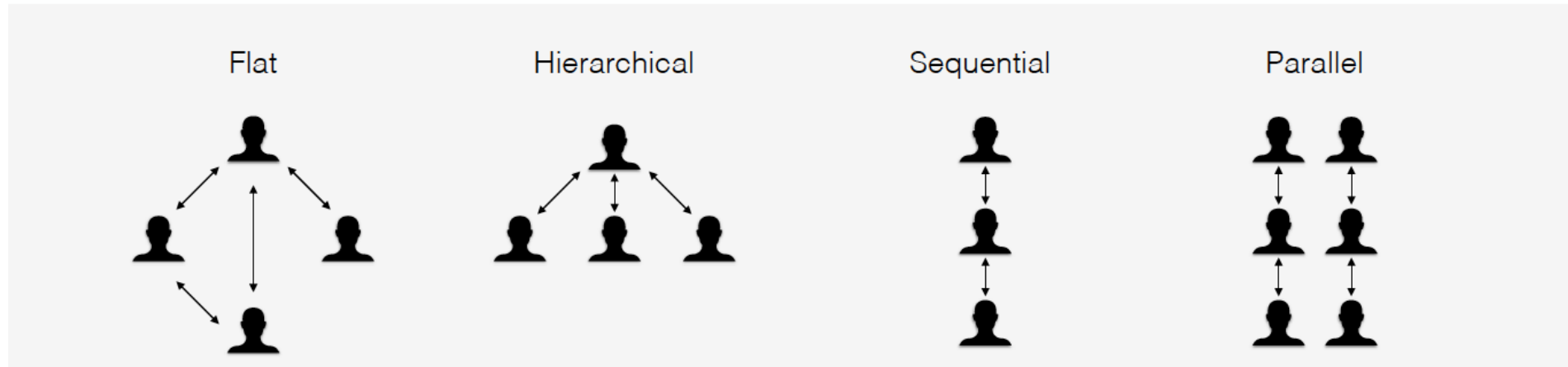
**Entertainment Agent:** Controls TVs, speakers, and other media devices based on user requests.

**Notification Agent:** Alerts users about system updates, energy savings, or security events.

**Orchestrator Agent:** Ensures smooth coordination between specialized agents, managing workflows, dependencies, and conflicts.

# Multi-agent workflows

## Example: Smart Home Automation



# Multi-agent workflows

## Example: Smart Home Automation

Considerations	Single Agent	Multi-Agent Workflow
<b>Modularity</b>	Harder to update; all-in-one system.	Easy to update or replace specific components.
<b>Fault Isolation</b>	One failure can disrupt everything.	Issues are contained to the failing agent.
<b>Optimization</b>	General-purpose, less efficient.	Specialized agents excel in their domains.
<b>Debugging</b>	Troubleshooting is complex and time-consuming.	Easier to isolate and fix issues.
<b>Parallel Processing</b>	Sequential task handling; slower.	Handles tasks simultaneously.
<b>Flexibility</b>	All tasks tied to one system.	Allows selective integration and third-party tools.

---

Questions

bidong@syr.edu