Generative AI at SAP

Unit 3: Adapting Generative AI to Business Context

Objectives of Unit 3

In Unit 3, you will learn about

- The limitations of generative AI models
- Methods to address limitations by grounding and adapting generative AI to business context
- Best practices when adapting generative AI

Limitations of generative AI

Generative AI's ability to answer broad, general knowledge problems is remarkable!

Forbes

GPT-4 Beats 90% Of Lawyers Trying To Pass The Bar

John Koetsier Senior Contributor

Journalist, analyst, author, and speaker.

Follow

Mar 14, 2023, 06:40pm EDT

The Atlantic

Is This the Singularity for Standardized Tests?

GPT-4's mastery of the SAT will re-entrench the power and influence of rote exams.

By Ian Bogost

MEDPAGETODAY®

AI Passes U.S. Medical Licensing Exam

— Two papers show that large language models, including ChatGPT, can pass the USMLE

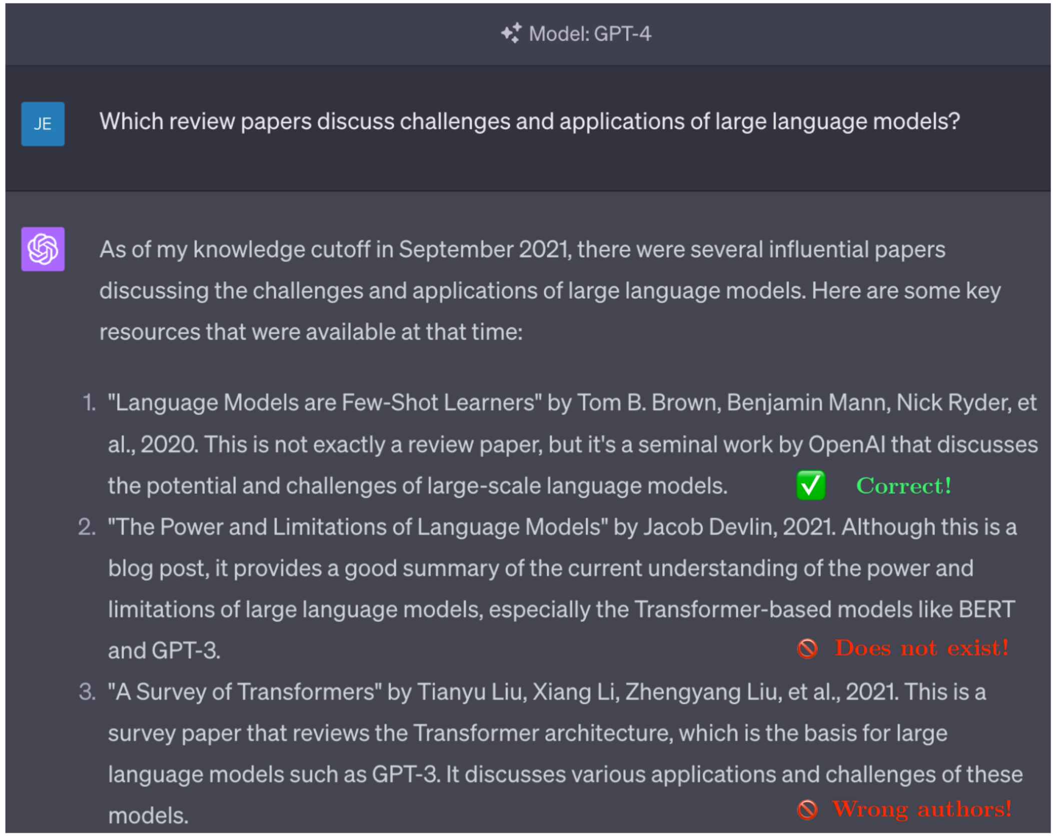
by Michael DePeau-Wilson, Enterprise & Investigative Writer, MedPage Today January 19, 2023

Such successes may give the impression that generative AI can tackle any problem out of the box.

Users must be aware of current limitations, keep expectations in check, and understand how to make generative AI enterprise-ready to handle business problems.

Example limitations of generative AI

Hallucination. Large language models can generate plausible-sounding yet false answers.
Related: LLMs can provide different “facts” depending on the language in which the prompt was given.



“… six legal cases listed to support their March 1 written arguments did not exist.”

“… we made a good faith mistake in failing to believe that a piece of technology could be making up cases out of whole cloth.”

Source: arXiv 2307.10169. Challenges and Applications of Large Language Models.

Example limitations of generative AI

Up-to-date and specific knowledge. The knowledge of a generative AI model is frozen in time from when it was trained. In contrast to world general knowledge, business information changes quickly.

Prompt

Who won the NBA preseason game last night between the LA Lakers and Golden State Warriors?

Output (model 1)

I'm sorry for any inconvenience, but as an AI last trained on data up to September 2021, I don't have the ability to access real-time information or updates, including recent sports scores.

Output (model 2 – hallucination)

The Golden State Warriors defeated the Los Angeles Lakers 114-101 in their preseason game last night.

Prompt

What libraries and SDKs are available with SAP AI Core?

Output

As of my training data until September 2021, SAP AI Core does not provide any specific libraries or SDKs. Instead, it integrates with various Python libraries and ML frameworks used for machine learning and data science, such as TensorFlow, PyTorch, Scikit-Learn, etc.

Ground truth:

SAP AI Core, **launched October 12th 2021**, provides SAP AI Core SDK, AP API Client SDK, and Metaflow Python Library (SAP AI Core Product Documentation, 2023-09-01)

Adapting generative AI to business context

Example limitations of generative AI

Inconsistent math abilities and limited notion of time. Although improving, large language models are not calculators. Other kinds of foundation models could conceivably complete forecasts and math.

Prompt

Jane was elected class president in 1973 when she was 12 years old.

Her daughter, Jill, was elected class president in 2012 when she was 13 years old.

Who was older when she was elected class president, Jane or Jill?

*Correct: When elected, Jill was 13 and Jane was 12. Therefore, **Jill was older.***

** Experiment conducted October 2023 using three different leading LLMs. Same result when the years were removed.*

Note: Various prompt engineering techniques could guide the model to provide the correct answer.

Jane was older when she was elected class president. In 1973, Jane was 12 years old, while in 2012, Jill was 13 years old.

Output (model 1)

Jane was elected class president in 1973 when she was 12 years old, and her daughter, Jill, was elected class president in 2012 when she was 13 years old. Therefore, **Jane was older** when she was elected class president.

Output (model 2)

Jane was older when she was elected class president, because she was 12 years old in 1973, and Jill was 13 years old in 2012.

Output (model 3)

Example limitations of generative AI

… even though the capabilities of generative AI to solve and explain some math problems are impressive! Techniques like calling functions and chain-of-thought prompting can increase performance on math problems.

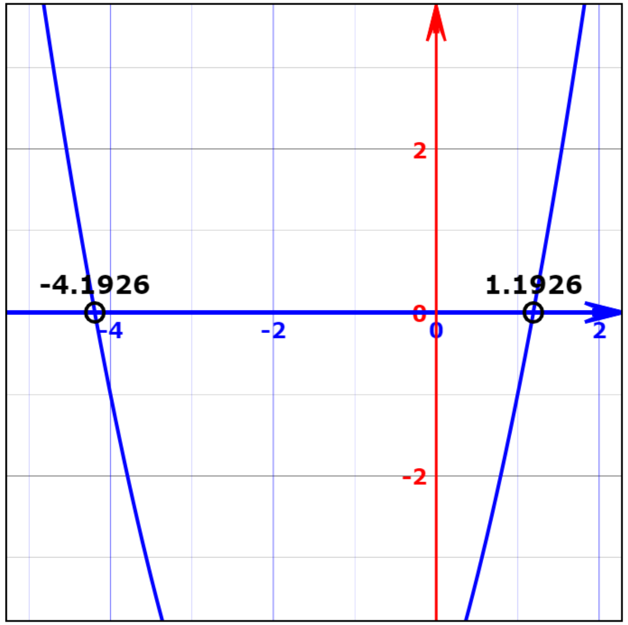
Prompt

Solve for X, show your work, and list the answer:

X^2 + 3x -5 = 0

Note:

Checking the output, the LLM got the correct answer!



Output (GPT-4)

This is a quadratic equation, and we can solve it by using the quadratic formula which is
x = [-b ± sqrt(b^2 - 4ac)] / (2a).

Here, a = 1, b = 3, and c = -5.

Let’s plug these values into the formula:

x = [-3 ± sqrt((3)^2 - 4*1*(-5))] / (2*1)
x = [-3 ± sqrt(9 + 20)] / 2
x = [-3 ± sqrt(29)] / 2

So, the solutions are:

x = (-3 + sqrt(29)) / 2
x = (-3 - sqrt(29)) / 2

These are the exact solutions. If you want to approximate to decimal places:

x ≈ 1.1925824
x ≈ -4.1925824

Adapting generative AI to business context

Methods to ground and adapt generative AI to business context

Data

- Text
- Image
- Speech
- …

TRAINING

Foundation models
General purpose.
With limitations.

Grounding & Adaptation

Task-specific instructions

Prompt Engineering

Provide more information to describe the task. Zero-shot and few-shot learning.

Retrieval Augmented Generation (RAG)

Extend to external domain knowledge by retrieving and injecting information via embeddings, knowledge graph, and search. Can provide references.

Orchestration Tools

Agents, functions, plug-ins, prompt and model chaining, memory.

Re-training

Fine-Tuning

Improve performance for domain-specific tasks with labeled data.

Includes supervised task-specific and instruction tuning.

Reinforcement Learning with Human Feedback (RLHF)

Update model by incorporating human preference.
Improve the quality and adjust weights after initial training.

Processes & Governance

- **AI ethics**
- **Human-in-the-loop** Humans + AI
- **Output validation and cross-checks**
- **Testing, red-teaming**
- **Continuous feedback and monitoring**
- **…**

Prompt engineering example to create a job description

Prompting: Give the model detailed information to reliably produce the desired output.

Zero-shot learning

Prompt 1:

Write a job description for a support engineer.

Instruction following

Prompt 2:

Write a job description for a support engineer. Keep it less than 300 words.

Locate the job in Chicago.

Include the following skills:

1) …

In-context learning including examples

Prompt 3:

Create engaging job descriptions for company ACME. Location is Chicago and start date is January.

User input

{ "job_title": "support engineer",
"required_skills": "..."}

System prompt with content

```
Ensure diversity and fairness.
Use subtitles for required
skills and for competencies.
List skills and competencies
as bulleted lists.
```

System prompt with instructions

```
Use the same style based on
examples of ACME job postings:
…
```

System prompt with examples

Number of tokens in the ***context window*** sent to the LLM increases (and along with it cost and response time).

More information and consistent input result in more precise output.

Prompt engineering example with SAP business documents

Extract information from business documents with large language models

Original input image

OCR

Text with preserved 2D structure

```
                                                                       Donnau  SchweiB Technik GmbH
  Lieferschein
                                                                       Guldenstr. 1 1
                                                                       75671 eggenstein
  During Schwe'fttechnik GmbH Postfach 1108 86329 Konigsbrunn          Deutschland
                                                                       Telefonnr.
  BBB  AG                                                              Fax.nr.
  Abladestelle 41892 Hr. Kirzinger MT-144                              E-Mail:
  Albert-Einstein-Str. 5
  76930 Eggenstein
  Deutschland                                                          Seite 1



  Ihre Kunden-Nr.:        30951                      Lieferdatum:      7. Marz 2023
  Lieferanten-Nr.:        770570-30                  Lieferungsnr.:     70500
  Auftragsnummer:         S030066                   Ihre Umst-ID:
  Auftragsdatum:          10.02.2023
  Externe Belegnr.:       G5BPLGK
  Pos.     Nr.              Beschreibung                                Menge   Einheit
  2        FD030305                                                          4 Stuck
                            PASSSCHRAUBE
                            2500  1078 001  007
                            Zeichnungsnummer:             8005
                            Ref-Nr.                       SAP-NR.  7303500
                            Einzelgewicht (netto) in kg:  0,01
                            Gesamtgewicht  (netto) in kg: 0,04
  3        FD030600                                                          1 Stuck
                            GEWINDEBOLZEN
                            2500 841  011007
                            Zeichnungsnummer:             8003
                            Ref-Nr.                       SAP-NR. 7303506

 Lieferbedingung.   FREI  HAUS            Frei Haus
 Rech, an Adresse
 Rech, an Deb.-Nr. 300078
 BBB AG
 Kreditorenbuchhaltung, Rechnungsprufung
 Heidemannstrafie 170
 80788 Munchen
 Deutschland


                                                TUV
                                                 SUD
```

LLM

+ Append instructions/examples to the prompt (“meta prompt”)

Extract the following fields as JSON:
- DeliveryNoteNo: ID for the delivery note
- DocumentDate: Date of the document
- PurchaseDoc: Reference to purchase doc
- SupplierID: Supplier's unique ID
- Items: List of products
- ItemNo: Number for the item in the list
- ...:

Structured result

```
{
   "DeliveryNoteNo": "70500",
   "DocumentDate": "2023-03-07",
   "PurchaseDoc": "G5BPLGK",
   "SupplierID": "770570-3O",
   ”Items": [
     {
       "ItemNo": "2",
       "ProductId": "7303500",
       "Quantity": "4",
       "UnitOfMeasure": "Stück"
     },
     {
       "ItemNo": "3",
       "ProductId": "7303506",
       "Quantity": "1",
       "UnitOfMeasure": "Stück"
     }]
}
```

Output validation

```
Supplier ID XXXXXX-XX incorrect.
Review proposed supplier ID match
from master data: XXXXXY-XX
```

Retrieval augmented generation (RAG) and embeddings example

First: what are embeddings?

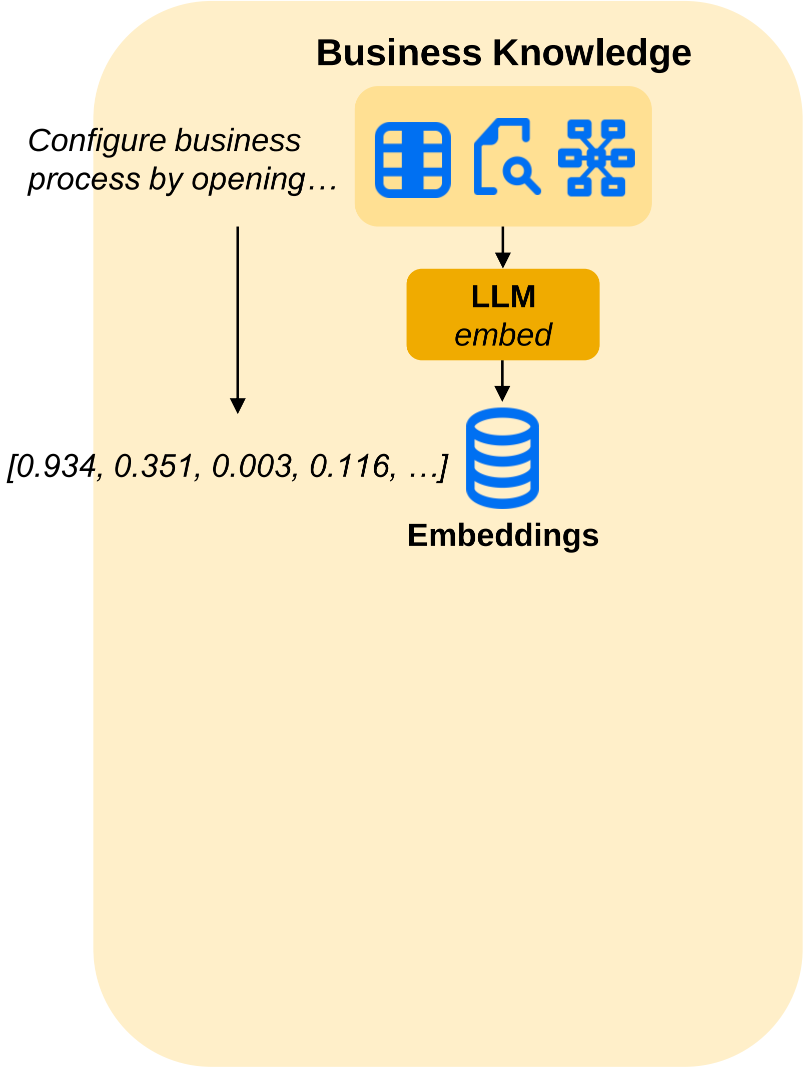
Embeddings: numerical representations of information that retain semantic, contextual meaning (text, images, etc.)

Example:

- Encode product documentation as embeddings using machine learning (Word2Vec, SBERT…). Store in vector database.
- "*Configure business process by opening…* " is represented as a vector [0.934, 0.351, 0.003, 0.116, …]

The vector captures the semantic meaning in the text, e.g. "*Apple*" has a different vector in "*Apple* makes phones" vs. "I have an *Apple* iPhone 15" vs. "*Apple* is an ingredient in pies".

Business data represented as embeddings can be easily searched and retrieved using techniques like *vector similarity scoring*.



Retrieval augmented generation (RAG) and embeddings example

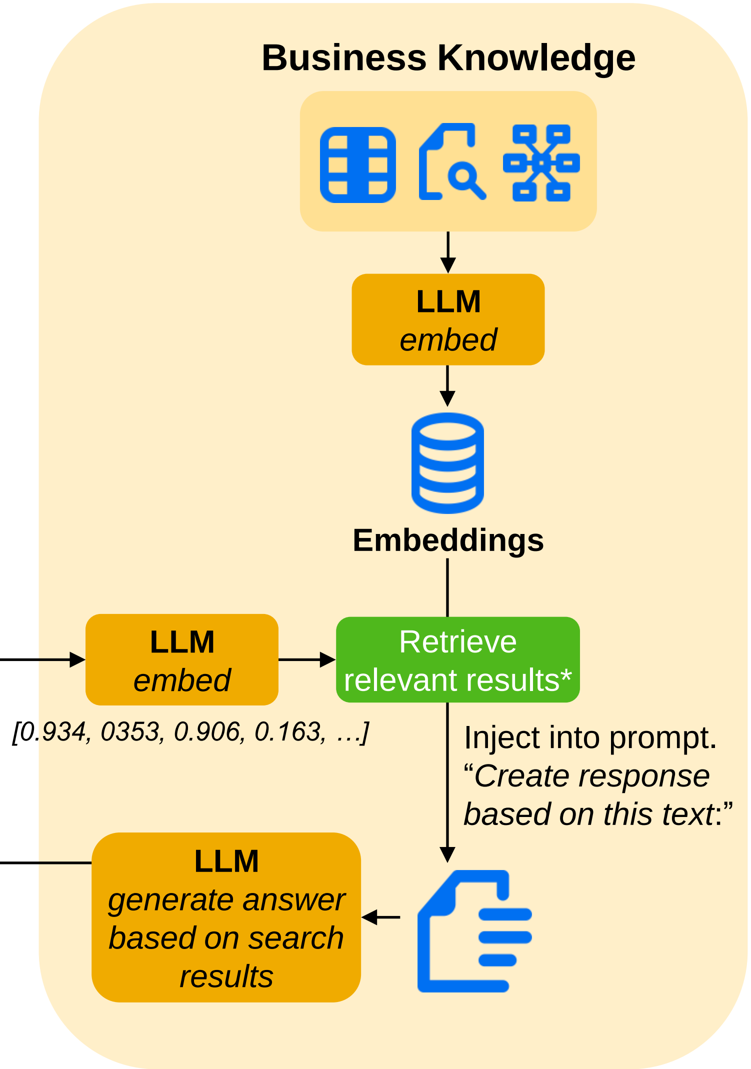
You have a set of business documents that should be used to answer user questions

Answering business-specific questions requires two steps:

1. Embed your business knowledge to retrieve relevant items given a question, using LLMs and other foundation models.
2. Generate the answer from the best results using LLMs. Use prompt engineering as needed.

“How can I adapt my business process for VAT calculation?”

“You need to configure your business process and add a step by…”

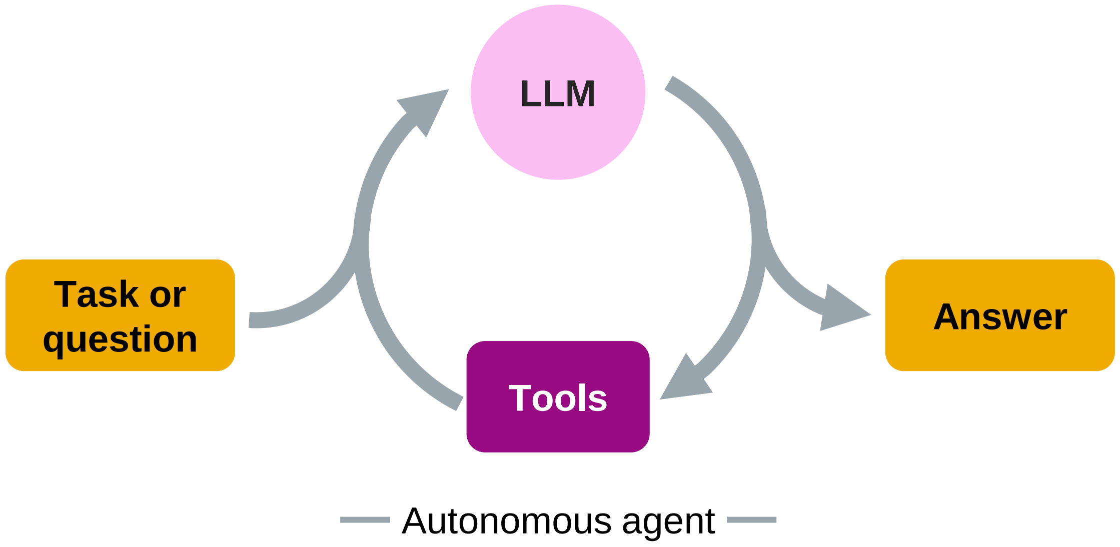


Using techniques like embeddings 1) grounds the prompt with relevant information and 2) references source information to generate the response.

* Possible to use classical keyword-based search as an alternative

Orchestration tools (“Agents”)

Orchestration example (experimental): Give an LLM access to tools. An agent retrieves API specifications from SAP systems, develops plan, and extracts data via API calls to formulate a response.



Example

Task/Question: User Inputs a Task

Autonomous Agent

1. Retrieve

SAP systems and their API specifications are retrieved based on the input (powered by **knowledge graphs**)

2. Plan

The **LLM** generates a plan based on the API retrieved specifications

3. Execute

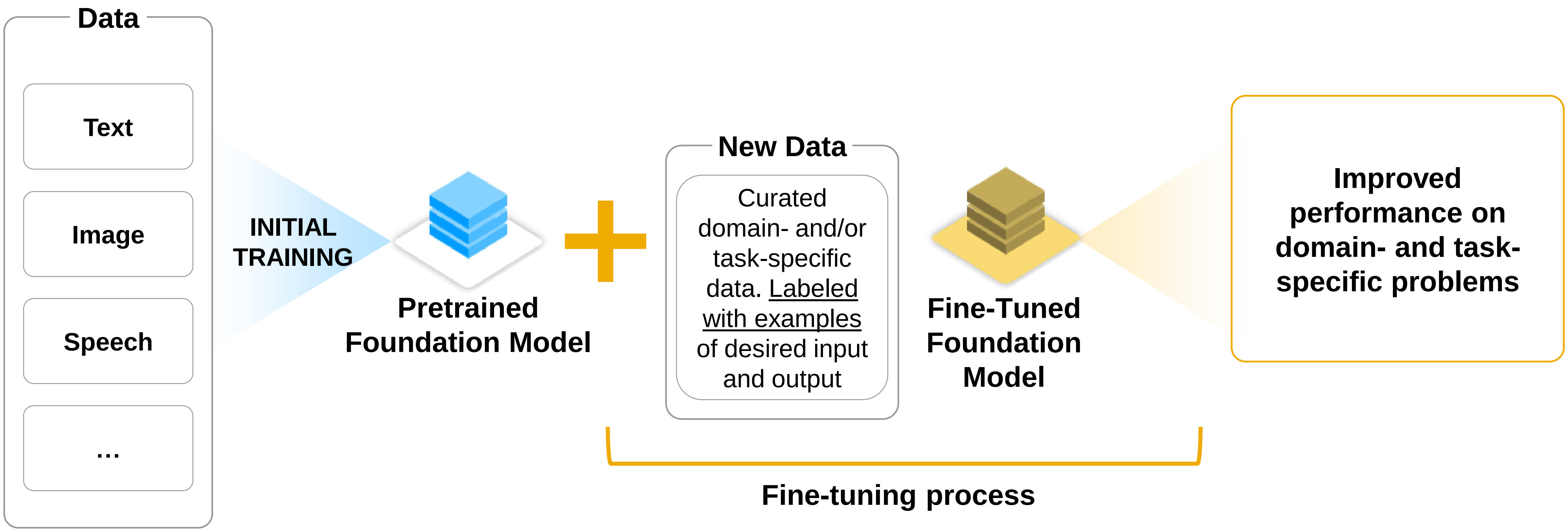
The **LLM** follows the plan and generates API calls and reacts to the system responses

Answer: The final result is returned

- Use existing systems via API calls.
- LLM has no prior knowledge of the system. Knowledge is provided as part of the prompt.

Fine-tuning

Fine-tuning: adjusting an existing foundation model's parameters (i.e. weights) to perform better at a specific task by retraining it on a new data set.



Fine-tuning

Considerations for fine-tuning foundation models

Opportunities

- Strong potential to create **task-specific** models with significant boost in accuracy.
- Continued training on domain- or industry-specific data improves quality for downstream tasks in an area.
- Smaller to medium-sized (~5-70 billion parameter) LLMs can reach or exceed performance of more costly massive models.

Challenges

- May reduce but will not eliminate problems like hallucination.
- May not “forget” existing biases and information.
- Does not reliably learn new factual information.
- Requires sufficient (curated) training data.
- Can be computationally expensive and costly.
- Currently difficult to generalize the volume of training data needed to achieve meaningful results.

Consider and test task-specific instructions (prompt engineering) first before investing in fine-tuning models.

Recommendations for adaptation of foundation models

DO NOT

- ✘ Trust generic generative AI models to answer factual questions correctly, esp. in a business context.
- ✘ Start with fine-tuning.
- ✘ Use the biggest model by default.
- ✘ Productize a technically functional generative AI use case without established supporting processes.

DO

- ✓ Be aware of limitations. Ground and adapt generative AI models on business data to ensure output is relevant and reliable.
- ✓ Start with task-specific instructions like prompt engineering and RAG.
- ✓ Test and adapt different generative AI models and optimize based on price and performance.
- ✓ Ensure proper governance and design for generative AI (e.g. human in the loop, output validation, monitoring…).

Thank you.

Contact information:

open@sap.com

Follow all of SAP

www.sap.com/contactsap

THE BEST RUN SAP